USER MANUAL

GoPxL for Gocator Laser Profilers

Gocator 2300, 2400, 2500, 2600 Series

Firmware version: 1.1.x.x

Document revision: I

Copyright

Information contained within this manual is subject to change.

This product is designated for use solely as a component and as such it does not comply with the standards relating to laser products specified in U.S. FDA CFR Title 21 Part 1040.

Contact Information

LMI Technologies, Inc.
9200 Glenlyon Parkway
Burnaby BC V5J 5J8
Canada

Telephone: +1 604-636-1011
Fax: +1 604-516-8368

www.lmi3D.com

Table of Contents

Introduction

This documentation describes how to connect, configure, and use a Gocator line profile sensor ("G2 sensor" for short) using the GoPxL software.

	Before using a sensor, you should familiarize yourself with sensor safety and maintenance. For more information, see *Safety and Maintenance* on page 14.

For a breakdown of this documentation, see *Using the Manual* on page 22.

The documentation applies to the following:

- Gocator 2100 and 2300 series (C and D revision only)
- Gocator 2400 series
- Gocator 2500 series
- Gocator 2600 series
- GoMax NX
- X64-based PC (Intel/AMD) running Windows 10 (for running GoPxL on a PC)

⚠	GoPxL only supports GoMax NX, not the pre-NX version.

Notational Conventions

This documentation uses the following notational conventions:

⚠	Follow these safety guidelines to avoid potential injury or property damage.

	Consider this information in order to make best use of the product.

Safety and Maintenance

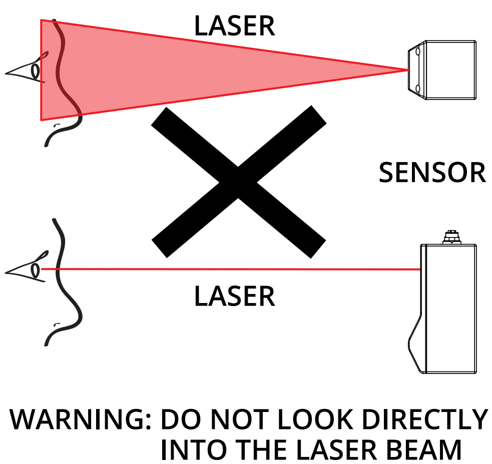
The following sections describe the safe use and maintenance of Gocator sensors.

Laser Safety

Gocator sensors contain semiconductor lasers that emit visible or invisible light and are designated as Class 2, 2M, Class 3R, or Class 3B, depending on the laser option. For more information on the laser classes used in these sensors, *Laser Classes* on the next page.

	To understand sensor part numbers, which contain information on laser class and color, among other things, see *Sensor Part Numbers* on page 25.

Gocator sensors are referred to as *components*, indicating that they are sold only to qualified customers for incorporation into their own equipment. These sensors do not incorporate safety items that the customer may be required to provide in their own equipment (e.g., remote interlocks, key control; refer to the references below for detailed information). As such, these sensors do not fully comply with the standards relating to laser products specified in IEC 60825-1 and FDA CFR Title 21 Part 1040.



Use of controls or adjustments or performance of procedures other than those specified herein may result in hazardous radiation exposure.

References

1. *International standard IEC 60825-1 (2001-08) consolidated edition*, Safety of laser products – Part 1: Equipment classification, requirements and user's guide.
2. *Technical report 60825-10*, Safety of laser products – Part 10. Application guidelines and explanatory notes to IEC 60825-1.

3. *Laser Notice No. 50*, FDA and CDRH (https://www.fda.gov/Radiation-EmittingProducts/ElectronicProductRadiationControlProgram/default.htm)

Laser Classes

Class 2 laser components

Class 2 laser components are considered to be safe, provided that:

- The user's blink reflex can terminate exposure (in under 0.25 seconds).
- Users do not need to look repeatedly at the beam or reflected light.
- Exposure is only accidental.

Class 2M laser components

Class 2M laser components should not cause permanent damage to the eye under reasonably foreseeable conditions of operation, provided that:

- No optical aids are used (these could focus the beam).
- The user's blink reflex can terminate exposure (in under 0.25 seconds).
- Users do not need to look repeatedly at the beam or reflected light.
- Exposure is only accidental.

Class 3R laser components

Class 3R laser products emit radiation where direct intrabeam viewing is potentially hazardous, but the risk is lower with 3R lasers than for 3B lasers. Fewer manufacturing requirements and control measures for 3R laser users apply than for 3B lasers.

- Eye protection and protective clothing are not required.
- The laser beam must be terminated at the end of an appropriate path.
- Avoid unintentional reflections.
- Personnel must be trained in working with laser equipment.

Class 3B laser components

Class 3B components are unsafe for eye exposure.

- Usually only eye protection is required. Protective gloves may also be used.
- Diffuse reflections are safe if viewed for less than 10 seconds at a minimum distance of 13 cm.
- There is a risk of fire if the beam encounters flammable materials.
- The laser area must be clearly identified.
- Use a key switch or other mechanism to prevent unauthorized use.
- Use a clearly visible indicator to show that a laser is in use, such as "Laser in operation."
- Restrict the laser beam to the working area.
- Ensure that there are no reflective surfaces in the working area.

For more information, see *Precautions and Responsibilities* below.

Precautions and Responsibilities

Precautions specified in IEC 60825-1 and FDA CFR Title 21 Part 1040 are as follows:

Requirement	Class 2	Class 2M	Class 3R	Class 3B
Remote interlock	Not required	Not required	Not required	Required*
Key control	Not required	Not required	Not required	Required – cannot remove key when in use*
Power-on delays	Not required	Not required	Not required	Required*
Beam attenuator	Not required	Not required	Not required	Required*
Emission indicator	Not required	Not required	Not required	Required*
Warning signs	Not required	Not required	Not required	Required*
Beam path	Not required	Not required	Terminate beam at useful length	Terminate beam at useful length
Specular reflection	Not required	Not required	Prevent unintentional reflections	Prevent unintentional reflections
Eye protection	Not required	Not required	Not required	Required under special conditions
Laser safety officer	Not required	Not required	Not required	Required
Training	Not required	Not required	Required for operator and maintenance personnel	Required for operator and maintenance personnel

**LMI Class 3B laser components do not incorporate these laser safety items. These items must be added and completed by customers in their system design. For more information, see Class 3B Responsibilities below.*

Class 3B Responsibilities

LMI Technologies has filed reports with the FDA to assist customers in achieving certification of laser products. These reports can be referenced by an accession number, provided upon request. Detailed descriptions of the safety items that must be added to the system design are listed below.

Remote Interlock

A remote interlock connection must be present in Class 3B laser systems. This permits remote switches to be attached in serial with the keylock switch on the controls. The deactivation of any remote switches must prevent power from being supplied to any lasers.

Key Control

A key operated master control to the lasers is required that prevents any power from being supplied to the lasers while in the OFF position. The key can be removed in the OFF position but the switch must not allow the key to be removed from the lock while in the ON position.

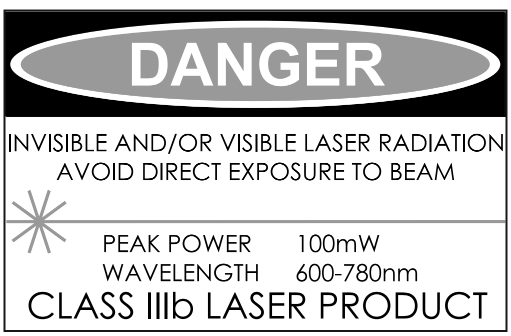
Power-On Delays
A delay circuit is required that illuminates warning indicators for a short period of time before supplying power to the lasers.

Beam Attenuators
A permanently attached method of preventing human access to laser radiation other than switches, power connectors or key control must be employed.

Emission Indicator
It is required that the controls that operate the sensors incorporate a visible or audible indicator when power is applied and the lasers are operating. If the distance between the sensor and controls is more than 2 meters, or mounting of sensors intervenes with observation of these indicators, then a second power-on indicator should be mounted at some readily-observable position. When mounting the warning indicators, it is important not to mount them in a location that would require human exposure to the laser emissions. User must ensure that the emission indicator, if supplied by OEM, is visible when viewed through protective eyewear.

Warning Signs
Laser warning signs must be located in the vicinity of the sensor such that they will be readily observed.

Examples of laser warning signs are as follows:



FDA warning sign example

IEC warning sign example

Nominal Ocular Hazard Distance (NOHD)

Nominal Ocular Hazard Distance (NOHD) is the distance from the source at which the intensity or the energy per surface unit becomes lower than the Maximum Permissible Exposure (MPE) on the cornea and on the skin.

⚠	The laser beam is considered dangerous if the operator is closer to the source than the NOHD.

The following tables provide the NOHD values for each sensor model and laser class, assuming continuous operation of the laser. As a configurable device, a sensor lets you set the laser exposure (laser on-time) independently of the frame period (total cycle time for data acquisition). Continuous operation of the laser means that the laser exposure is configured to be identical to the frame period, which is also referred to as 100% duty cycle. However, in many applications the laser exposure can be smaller than the frame period (less than 100% duty cycle), thereby reducing the NOHD. The tables therefore show the worst-case NOHD.

⚠	In the model numbers in the following tables, an "x" refers to any number that represents an existing model. So for example, 2x80D refers to Gocator 2180, 2380, and 2880 (but not, for example, 2480, which isn't available).

The following table provides NOHD values for *current* hardware versions of sensors.

Current Hardware Versions

Model	Laser Class	Wavelength (nm)	Class I NOHD (mm)	Class II NOHD (mm)
21x0D/23x0D (except 2x80D)	2	660	670	-
	3R	660	3340	1330
2x80D	2	660	1310	-
	3R	660	4700	1850
2350C	3B (NIR laser)	808	19750	-
2375C	3B (NIR laser)	808	13777	-
2x75D	2	660	670	-
	3R	660	3340	1330
2410A 2420A	2M	405	259 [a]	103 [a]
	3R	405	1300 [a]	500 [a]
2430A 2440A	2	660	670	-
	3R	660	3340	1330
	3B	660	6661	2598
2430A 2440A	2	405	554	-
	3R	405	2770	1065
	3B	405	7546	3005
2450A	2	405	559	-
	3R	405	2794	1075
	3B	405	9433	3755
2490A	2	660	1310	-
	3R	660	4700	1850
251xA	2	405	615[a]	-
252xA	2	405	615[a]	-
2530A	2	405	564[a]	-
2540A 2550A	2	405	590[a]	-
	3R	405	2949[a]	1150
	3B	405	6667[a]	2600
2630A 2640A	2	405	615[a]	-

Model	Laser Class	Wavelength (nm)	Class I NOHD (mm)	Class II NOHD (mm)
2650A 2670A				
	3R	405	3078[a]	1200
	3B	405	6667[a]	2600
2690A	2	660	718[a]	-
	3R	660	3590[a]	1400
	3B	660	6667[a]	1700

[a] With exposure time < 10 seconds. For longer exposure times, consult IEC 60825.

The following table provides NOHD values for older hardware version sensors.

Older Hardware Versions

Model	Laser Class	Wavelength (nm)	Class I NOHD (mm)	Class II NOHD (mm)
2120A to C, 2320A to C 2130A to C, 2330A to C 2140A to C, 2340A to C 2150A to C, 2350A to C	2M	660	259	103
	3R	660	900	358
	3B	660	5759	2292
2350A	3B (NIR laser)	808	19750	-
2170A to C, 2370A to C	2M	660	251	100
	3R	660	875	348
	3B	660	3645	1451
2375A	3B (NIR laser)	808	13777	-
2180A to C, 2380A to C	2M	660	245	97
	3R	660	859	342
	3B	660	2645	1052

Systems Sold or Used in the USA

Systems that incorporate laser components or laser products manufactured by LMI Technologies require certification by the FDA.

Customers are responsible for achieving and maintaining this certification.

Customers are advised to obtain the information booklet *Regulations for the Administration and Enforcement of the Radiation Control for Health and Safety Act of 1968: HHS Publication FDA 88-8035*.

This publication, containing the full details of laser safety requirements, can be obtained directly from the FDA, or downloaded from their web site at https://www.fda.gov/Radiation-EmittingProducts/ElectronicProductRadiationControlProgram/default.htm.

Electrical Safety

> ⚠ Failure to follow the guidelines described in this section may result in electrical shock or equipment damage.

Sensors should be connected to earth ground
All sensors should be connected to earth ground through their housing. All sensors should be mounted on an earth grounded frame using electrically conductive hardware to ensure the housing of the sensor is connected to earth ground. Use a multi-meter to check the continuity between the sensor connector and earth ground to ensure a proper connection.

Minimize voltage potential between system ground and sensor ground
Care should be taken to minimize the voltage potential between system ground (ground reference for I/O signals) and sensor ground. This voltage potential can be determined by measuring the voltage between Analog_out- and system ground. The maximum permissible voltage potential is 12 V but should be kept below 10 V to avoid damage to the serial and encoder connections.

For a description of the connector pins, see *Gocator I/O Connector* on page 1060.

Use a suitable power supply
The power supply used with sensors should be an isolated supply with inrush current protection or be able to handle a high capacitive load. Verify the voltage input requirements for your sensor in the sensor's specifications; for specifications, see *Sensors* on page 975.

Use care when handling powered devices
Wires connecting to the sensor should not be handled while the sensor is powered. Doing so may cause electrical shock to the user or damage to the equipment.

Heat Warning

⚠	If a sensor is not adequately heat-sunk, the housing may get hot enough to cause injury.

Sensors should be properly heat-sunk
To avoid injury and to ensure that a sensor functions properly, mount the sensor to a thermally conductive material for good heat-sinking.

Handling, Cleaning, and Maintenance

	Dirty or damaged sensor windows (emitter or camera) can affect accuracy. Use caution when handling the sensor or cleaning the sensor's windows.

Keep sensor windows clean
Use dry, clean air to remove dust or other dirt particles. If dirt remains, clean the windows carefully with a soft, lint-free cloth and non-streaking glass cleaner or isopropyl alcohol. Ensure that no residue is left on the windows after cleaning.

Turn off lasers when not in use
LMI Technologies uses semiconductor lasers in Gocator sensors. To maximize the lifespan of the sensor, turn off the laser when not in use.

Avoid excessive modifications to files stored on the sensor
Sensor settings are stored in flash memory inside the sensor. Flash memory has an expected lifetime of 100,000 writes. To maximize lifetime, avoid frequent or unnecessary file save operations.

Environment and Lighting

Avoid strong ambient light sources

The imager used in this product is highly sensitive to ambient light. Do not operate this device near windows or lighting fixtures that could influence measurement or data acquisition.If the unit must be installed in an environment with high ambient light levels, a lighting shield or similar device may need to be installed to prevent light from affecting measurement.

Avoid installing sensors in hazardous environments

To ensure reliable operation and to prevent damage to sensors, avoid installing the sensor in locations

- that are humid, dusty, or poorly ventilated;
- with a high temperature, such as places exposed to direct sunlight;
- where there are flammable or corrosive gases;
- where the unit may be directly subjected to harsh vibration or impact;
- where water, oil, or chemicals may splash onto the unit;
- where static electricity is easily generated.

Ensure that ambient conditions are within specifications

Except for the sensor models listed below, the sensor operating temperatures range is 0 to 50 °C.

- For Gocator 2500 sensors, operating temperatures are 0 to 40 °C.
- For Gocator 2430, 2440, and 2450 sensors, which are equipped with a Class 2 blue laser, operating temperatures are 10 to 50 °C.

⚠	The sensor must be heat-sunk through the frame it is mounted to. When a sensor is properly heat sunk, the difference between ambient temperature and the temperature reported in the sensor's health channel is less than 15° C. In some applications, to maintain this maximum difference, you may need additional heatsinking. You can monitor this temperature on the **Report** > **Health** page.

For all sensors, the storage temperature is -30 to 70 °C.

For all sensors, relative humidity (non-condensing) is 25% to 85%.

The Master network controllers are similarly rated for operation between 0 and 50 °C.

⚠	Sensors are high-accuracy devices, and the temperature of all of its components must therefore be in equilibrium. When the sensor is powered up, a warm-up time of at least one hour is required to reach a consistent spread of temperature in the sensor.

Protect Master network controllers from dusty environments

⚠	If you are installing a Master 810 or 2410 in a dusty environment, LMI strongly recommends installing the device in a vented cabinet or enclosure to avoid contamination.

Using the Manual

Use the following to help decide which part of this manual you need.

> The tool topics from this manual are available in GoPxL, in the **Tool Help** panel at the bottom of the interface.

Manual breakdown

Category	Description
Safety and Maintenance	Important safety and maintenance information (see *Sensor Management and Maintenance* on page 117).
Getting Started	Hardware overview and installation information (see *Getting Started* on page 24).
Key Concepts	Fundamental Gocator sensor concepts (see *Key Concepts* on page 60).
User Interface Overview	General overview of the GoPxL user interface (see *User Interface Overview* on page 83).
Data Viewer	Description of the data viewer and how to use it (see *Data Viewer* on page 88).
File Formats	Lists the formats used in GoPxL (see *File Formats* on page 842).
Working with Scan Data	Describes how to record scan data and how to play back recordings (see *Working with Scan Data (Toolbar)* on page 112).
Creating, Saving and Loading Jobs	Describes how to work with jobs (see *Creating, Saving and Loading Jobs (Toolbar)* on page 115).
Configuring Systems	Contains subsections describing how to do the following: • Perform general sensor maintenance • Create a sensor system • Align sensors • Configure acquisition • Configure inspection and processing tools • Configure control (including protocol and GoHMI) • Configure reporting For more information, see *Configuring GoPxL* on page 116.
Using a PC Instance of GoPxL	Describes how to start and use a PC instance of GoPxL, as well as GoPxL Manager. You use a PC instance of GoPxL to accelerate a sensor and to examine scan data offline. For more information, see *Running GoPxL on a Windows PC* on page 828.
Development Kits and API	Provides an overview of the GoPxL SDK and the REST API and describes installation of the SDK (see *GoPxL SDK and REST API* on page 871).
GoHMI	Describes how to configure GoHMI in a PC instance of GoPxL. Also describes the GoHMI Designer (see *GoHMI and GoHMI Designer* on page 844).
Integrations	Describes the industrial protocols you can use in GoPxL (see See *Integrations* on

Category	Description
	page 884).
Specifications	Provides sensor specifications and drawings (see *Specifications* on page 973).

Getting Started

The following sections provide hardware and system overviews and describe installation and initial networking setup procedures.

Gocator Overview

Gocator sensors are designed for 3D measurement and control applications.Sensors are configured using a web browser and can be connected to a variety of input and output devices. Sensors can also be configured using the provided development kits.

Before using sensors, see *Safety and Maintenance* on page 14.

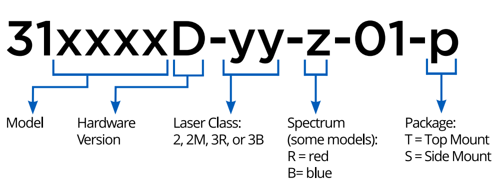
For information on the physical installation and mounting of sensors, as well as a general hardware overview, see *Getting Started* on page 24.

For key concepts on how sensors acquire data and perform measurements, see *Key Concepts* on page 60.

For information on using the PC-based interface (and the differences between running GoPxL on-sensor and on a PC), see *Running GoPxL on a Windows PC* on page 828.

Sensor Part Numbers

Use the following to understand sensor part numbers:



Example: Gocator 2330 with Class 2 laser = 312330D-2-R-01-T

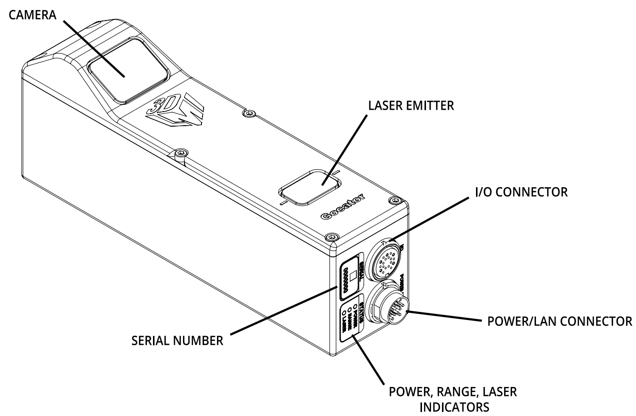
Upgrading from Gocator Firmware

If you are upgrading your sensor from Gocator firmware 6.2 or earlier to GoPxL, and you have created pattern files using Surface Pattern Matching or track files using Surface Track in the Gocator firmware, be sure to add instances of those tools and remove those files manually. These files can take up considerable space.

Hardware Overview

The following sections describe Gocator and its associated hardware.

Gocator Sensor



Gocator 2140 / 2340

Item	Description
Camera	Observes laser light reflected from target surfaces.
Laser Emitter	Emits structured light for laser profiling.
I/O Connector	Accepts input and output signals.
Power / LAN Connector	Accepts power and laser safety signals and connects to 1000 Mbit/s Ethernet network.
Power Indicator	Illuminates when power is applied (blue).
Range Indicator	Not currently used.
Laser Indicator	Illuminates when laser safety input is active (amber).
Serial Number	Unique sensor serial number.

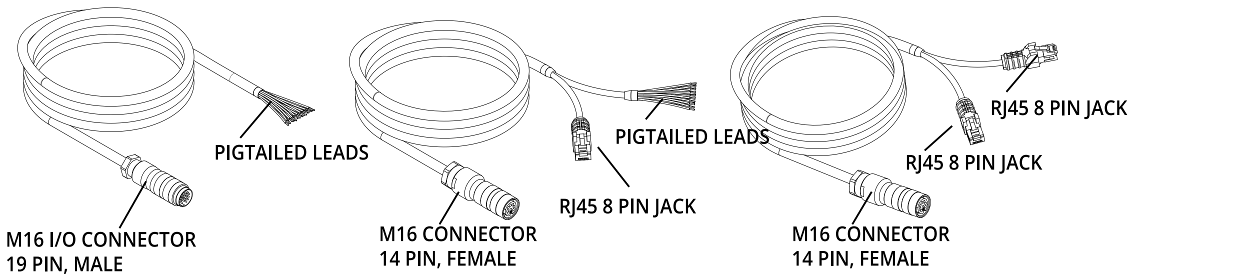
Gocator Cordsets

Gocator sensors use two types of cordsets: the Power & Ethernet cordset and the I/O cordset.

> When connecting cordsets to the sensor's connectors, do not exceed a torque of 2 Nm (18 in-lbs). Ensure that you properly secure the cordset cabling to avoid stress loading on the sensor connectors.

The Power & Ethernet cordset provides power and laser safety to the sensor. It is also used for sensor communication via 1000 Mbit/s Ethernet with a standard RJ45 connector. The Master version of the Power & Ethernet cordset provides direct connection between the sensor and a Master network controller, excluding Master 100 (for more information, see *Master Network Controllers* on page 1066).

The I/O cordset provides digital I/O connections, an encoder interface, RS-485 serial connection, and an analog output.



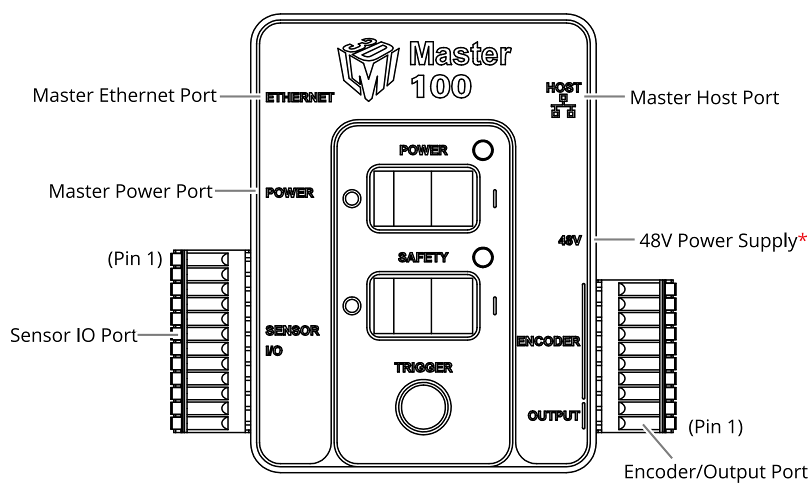
The maximum cordset length is 60 m.

See *Gocator I/O Connector* on page 1060 and *Gocator Power/LAN Connector* on page 1058 for pinout details.

See *Accessories* on page 1087 for cordset lengths and part numbers. Contact LMI for information on creating cordsets with customized lengths and connector orientations.

Master 100

The Master 100 is used by sensors for standalone system setup (that is, a single sensor).



Item	Description
Master Ethernet Port	Connects to the RJ45 connector labeled Ethernet on the Power/LAN to Master cordset.
Master Power Port	Connects to the RJ45 connector labeled Power/Sync on the Power/LAN to Master cordset. Provides power and laser safety to the sensor.
Sensor I/O Port	Connects to the I/O cordset.
Master Host Port	Connects to the host PC's Ethernet port.

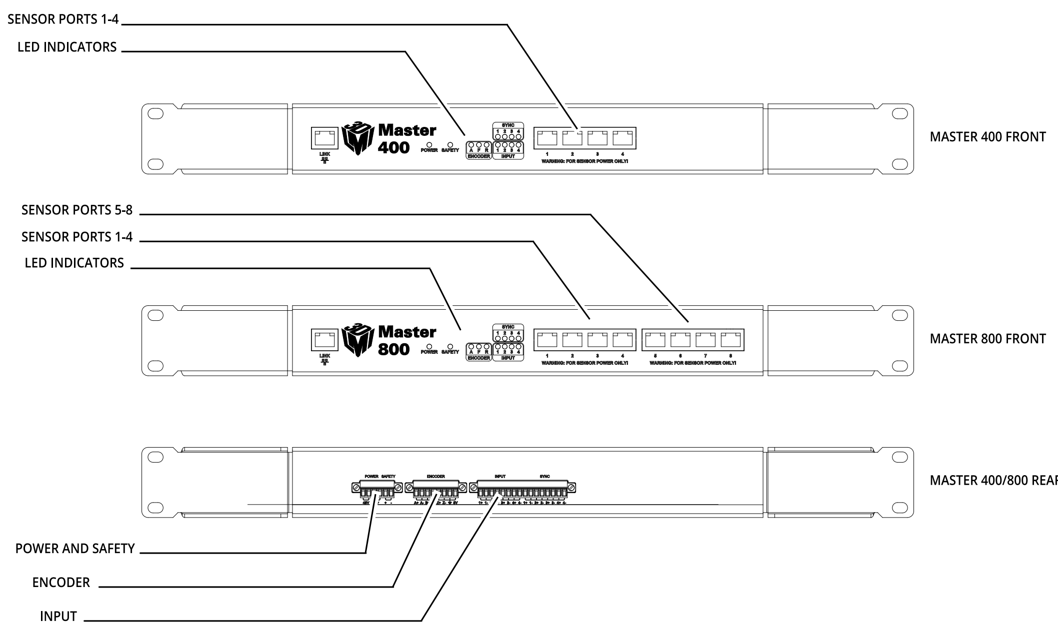
Item	Description
Power	Accepts power (+48 V).
Power Switch	Toggles sensor power.
Safety Switch	Toggles safety signal provided to the sensors [O= off, I= on]. This switch must be set to on in order to scan with laser-based sensors.
Trigger	Signals a digital input trigger to the sensor.
Encoder	Accepts encoder A, B and Z signals.
Digital Output	Provides digital output.

See *Master 100* on page 1066 for pinout details.

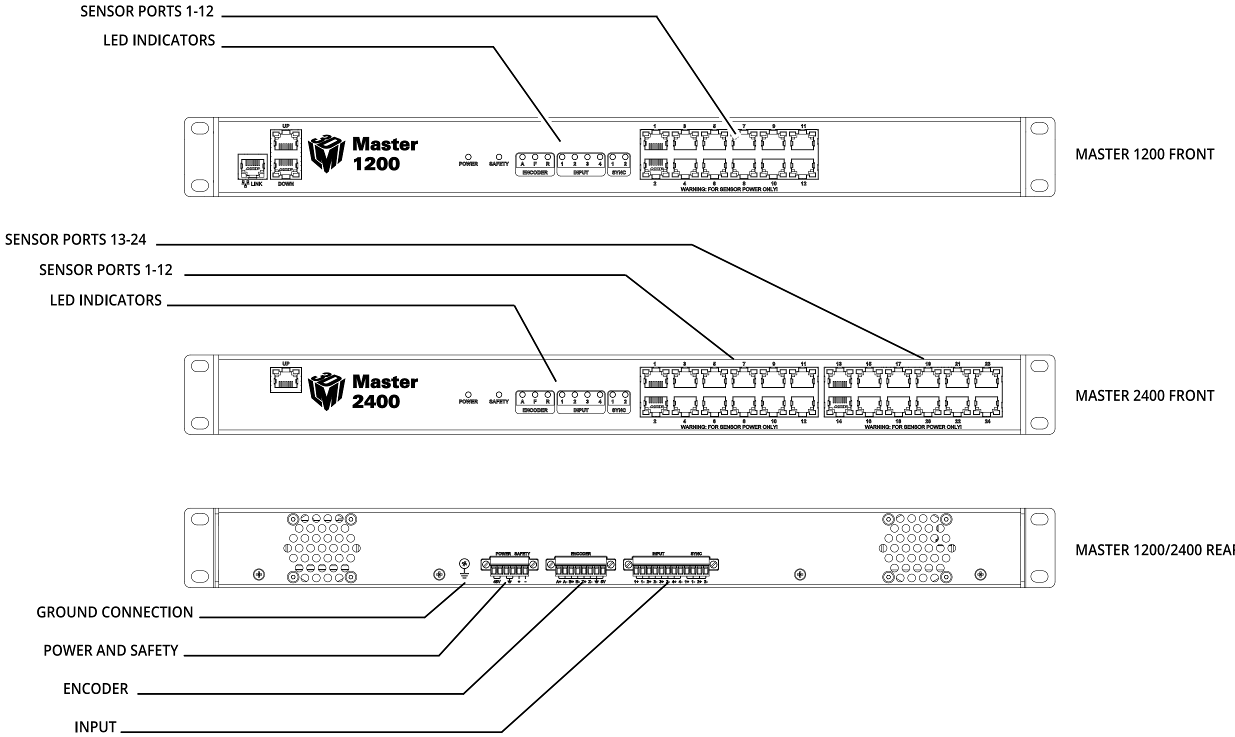
Master 400 / 800 / 1200 / 2400

The Master 400, 800, 1200, and 2400 network controllers let you connect more than two sensors:

- Master 400: accepts four sensors
- Master 800 accepts eight sensors
- Master 1200: accepts twelve sensors
- Master 2400: accepts twenty-four sensors



Master 400 and 800



Master 1200 and 2400

Item	Description
Sensor Ports	Master connection for sensors (no specific order required).
Ground Connection	Earth ground connection point.
Power and Safety	Power and safety connections. Safety input must be high in order to scan with laser-based sensors.
Encoder	Accepts encoder signal.
Input	Accepts digital input.

For pinout details for Master 400 or 800, see *Master 400/800* on page 1068.

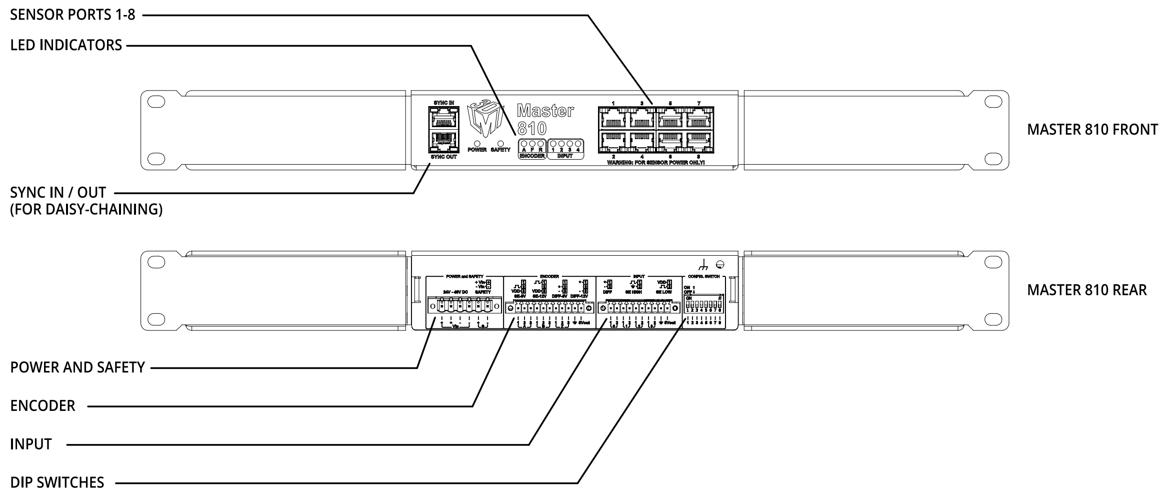
For pinout details for Master 1200 or 2400, see *Master 1200/2400* on page 1083.

Master 810 / 2410

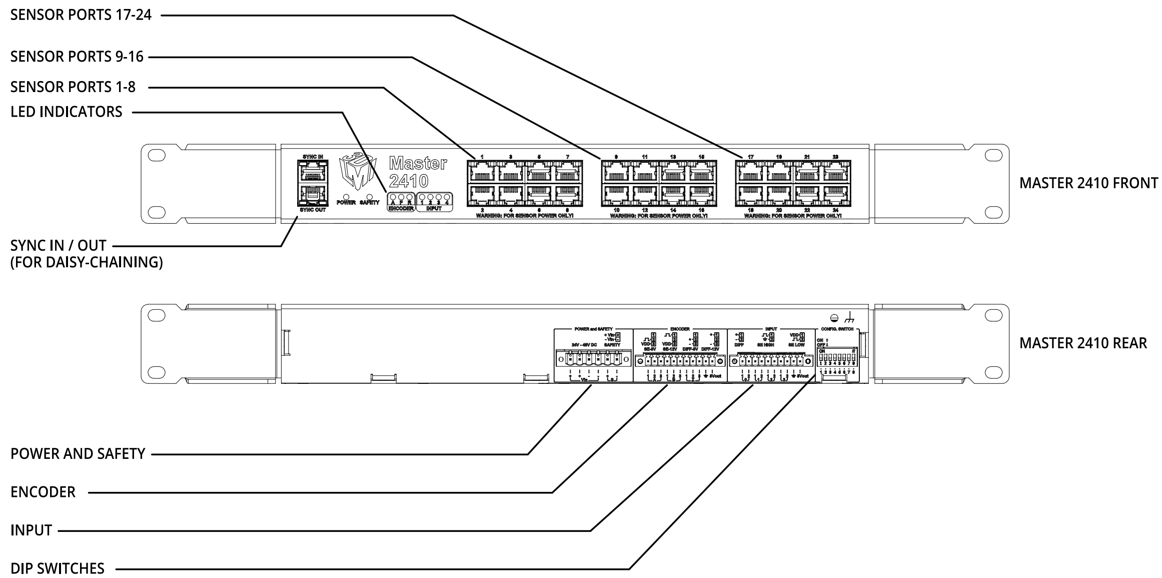
The Master 810 and 2410 network controllers let you connect multiple sensors to create a multi-sensor system:

- Master 810 accepts up to eight sensors
- Master 2410 accepts up to twenty-four sensors

Both models let you divide the quadrature frequency of a connected encoder to make the frequency compatible with the Master, and also set the debounce period to accommodate faster encoders. For more information, see *Configuring Master 810* on page 47. (Earlier revisions of these models lack the DIP switches.)



Master 810



Master 2410

Item	Description
Sensor Ports	Master connection for sensors (no specific order required).
Power and Safety	Power and safety connections. Safety input must be high in order to scan with laser-based sensors.
Encoder	Accepts encoder signal.
Input	Accepts digital input.
DIP Switches	Configures the Master (for example, allowing the device to work with faster encoders). For information on configuring Master 810 and 2410 using the DIP switches, see *Configuring Master 810* on page 47.
LED Indicators	For more information, see *Master 810/2410* on page 1072.

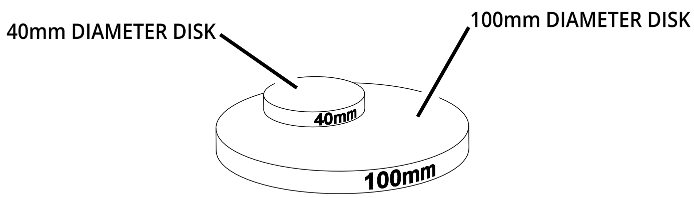
For pinout details, see *Master 810/2410* on page 1072.

Alignment Targets

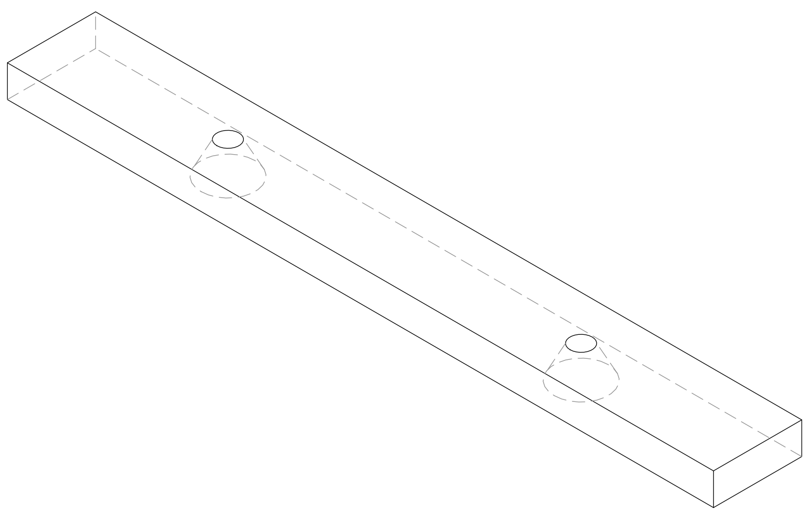
🗋	This section provides a brief overview of the kinds of alignment targets used to set up a sensor system. For details, see the appropriate cross-references below.

Targets are used for aligning sensors (due to mounting inaccuracies) and for calibrating transport systems.

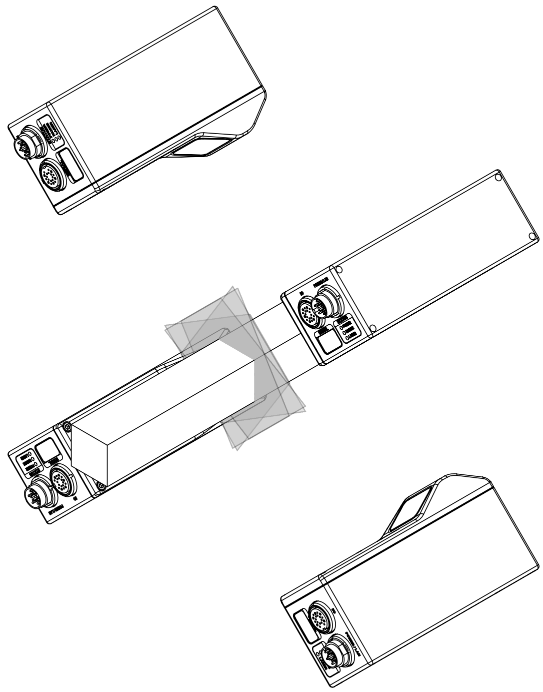
Disks are typically used with systems containing a single sensor and can be ordered from LMI Technologies. Note that disk alignment targets are typically used in demo systems, due to the lack of accuracy in the resulting alignment. When choosing a disk for your application, select the largest disk that fits entirely within the required field of view. See *Accessories* on page 1087 for disk part numbers.



For dual- and multi-sensor systems, where sensors are mounted in rows beside each other, or single-sensor systems where a higher degree of accuracy is required (in the presence of Z angle rotation), an alignment bar is used for alignment. (LMI Technologies does not manufacture or sell bars.) For bar construction requirements, see *Stationary and Moving Bar* on page 148.



For multi-sensor systems in a ring layout, where a lower degree of accuracy is acceptable, or X angle correction is not required, use a polygon-shaped alignment target. The number of corners in the target should correspond with the number of sensors in the system. Sensors should be positioned so that each sensor can scan a corner and surrounding surface. For polygon target construction requirements, see *Stationary Polygon* on page 153.



Finally, you can perform a high-accuracy alignment of ring (360-degree or partial) and wide layouts using special alignment targets and built-in measurement tools. For more information on this type of alignment, see *Aligning Sensors to 6 Degrees of Freedom* on page 157.

For more general information on the alignment process, including how to choose the alignment type for your sensor system, see *Aligning Sensors* on page 132.

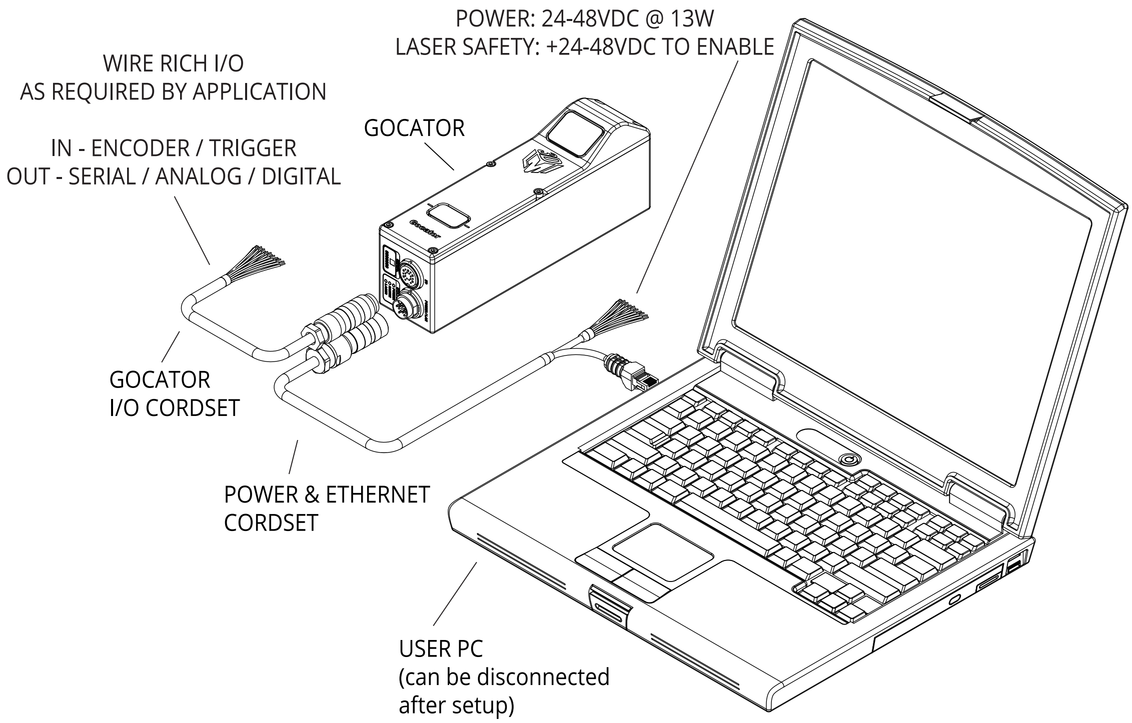
System Overview

Gocator sensors can be installed and used in a variety of scenarios. Sensors can be connected as standalone devices, dual-sensor systems, or multi-sensor systems.

Standalone System

Standalone systems include only a single sensor. The device can be connected to a computer's Ethernet port for setup and can also be connected to devices such as encoders, photocells, or PLCs.

For information on physical installation, see *Installation* on page 36.

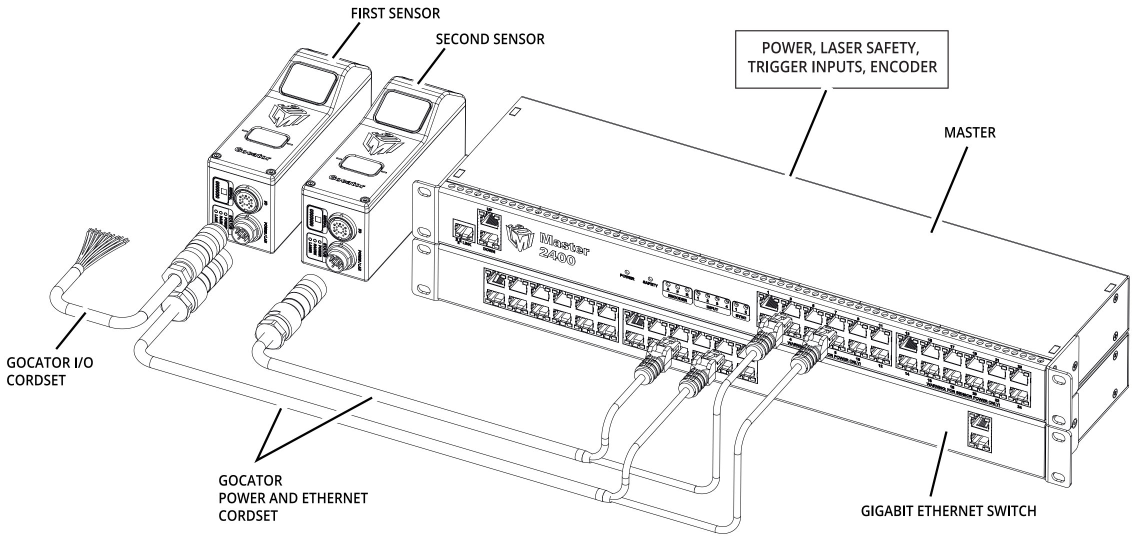


Dual-Sensor System

In a dual-sensor system, two sensors work together to perform data acquisition and output the combined results.

A Master network controller (excluding Master 100) must be used to connect two sensors in a dual-sensor system. Power and Ethernet to Master cordsets are used to connect sensors to the Master.

GOCATOR 2300 SERIES

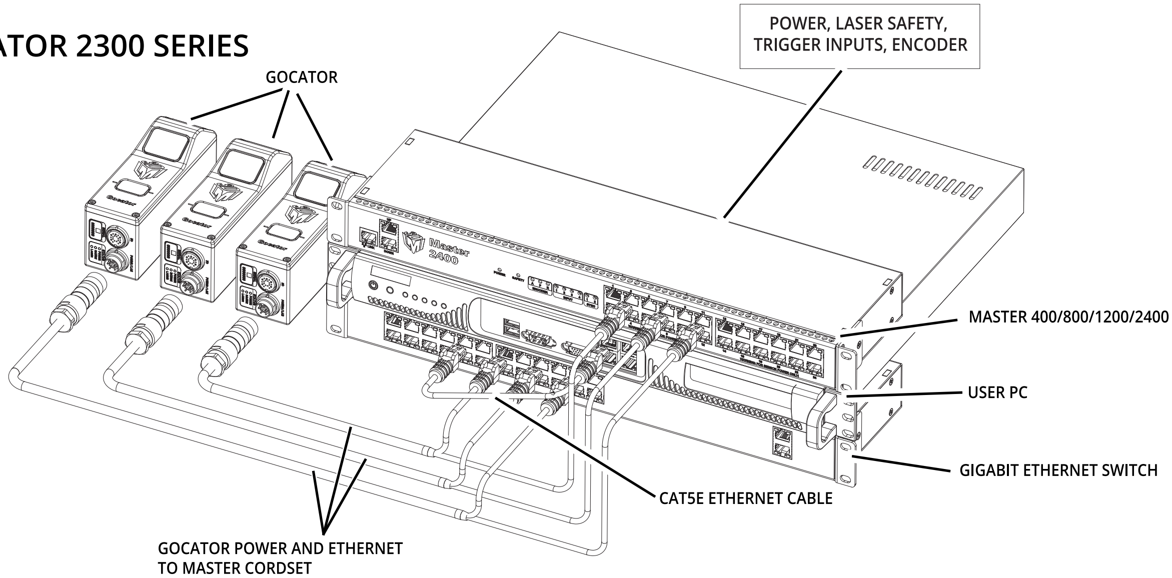


Multi-Sensor System

A Master network controller (excluding Master 100) can be used to connect two or more sensors into a multi-sensor system. Master cordsets are used to connect the sensors to a Master. The Master provides a single point of connection for power, safety, encoder, and digital inputs. A Master 400/800/810/1200/2400/2410 can be used to ensure that the scan timing is precisely synchronized across sensors. Sensors and client computers communicate via an Ethernet switch (1 Gigabit/s recommended).

Master networking hardware does not support digital, serial, or analog output.

GOCATOR 2300 SERIES



Installation

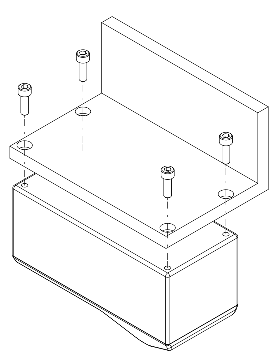
The following sections provide grounding, mounting, and orientation information.

> When connecting cordsets to the sensor's connectors, do not exceed a torque of 2 Nm (18 in-lbs). Ensure that you properly secure the cordset cabling to avoid stress loading on the sensor connectors.

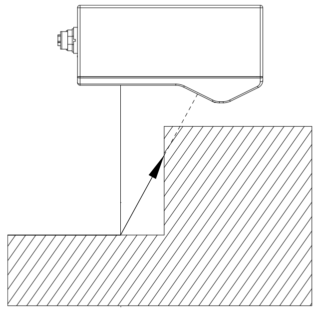
Mounting

Sensors should be mounted using a model-dependent number of screws. Some models also provide the option to mount using bolts in through-body holes. Refer to the dimension drawings of the sensors in *Specifications* on page 973 for the appropriate screw diameter, pitch, and length, and bolt hole diameter.

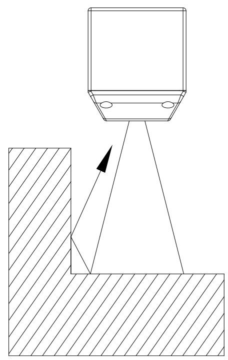
> ⚠ Proper care should be taken in order to ensure that the internal threads are not damaged from cross-threading or improper insertion of screws.

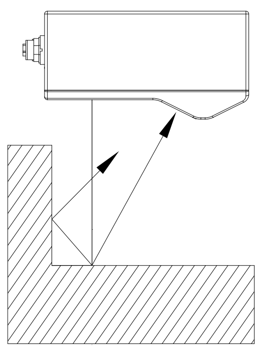


Sensors should not be installed near objects that might occlude a camera's view of the projected light.



Sensors should not be installed near surfaces that might create unanticipated laser reflections.





⚠	The sensor must be heat sunk through the frame it is mounted to. When a sensor is properly heat sunk, the difference between ambient temperature and the temperature reported in the sensor's health channel is less than 15° C.

⚠	Gocator sensors are high-accuracy devices. The temperature of all of its components must be in equilibrium. When the sensor is powered up, a warm-up time of at least one hour is required to reach a consistent spread of temperature within the sensor.

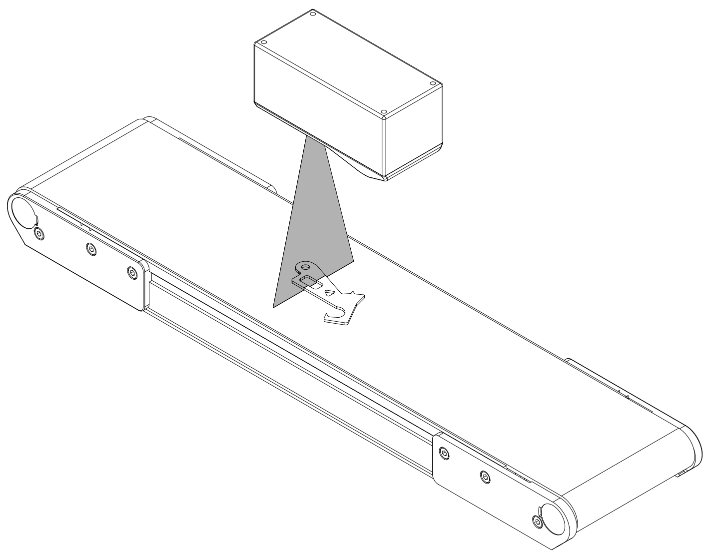
For more information on heat considerations, see *Environment and Lighting* on page 21.

Orientations and Layouts

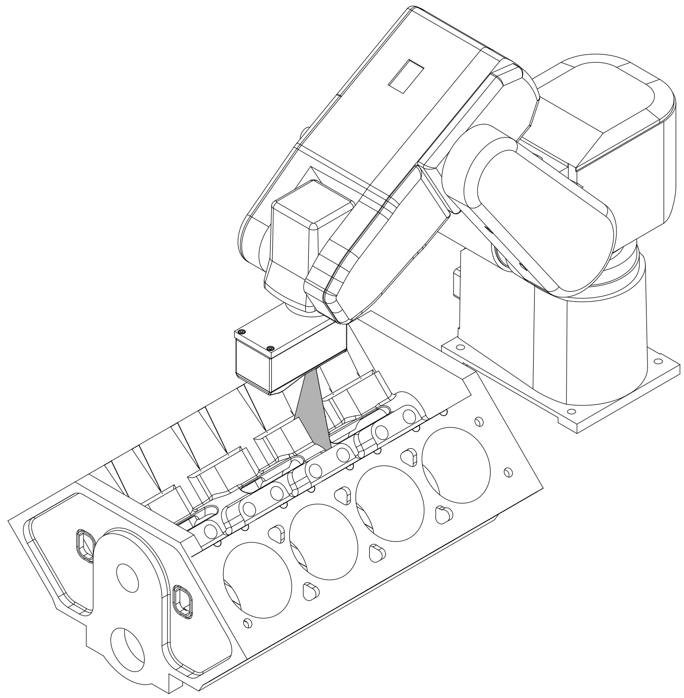
The examples below illustrate some of the possible mounting orientations and layouts for single-sensor, dual-sensor, and multi-sensor systems. The choice of orientation will depend on your application. For more information on orientations and setting them up using the Gocator interface, see *Creating a Sensor System* on page 124.

Typically, you will perform an alignment procedure with sensors using either the flat surface of the conveyor or an alignment target (for an introduction to alignment targets, see *Alignment Targets* on page 31). The choice of alignment target and whether it moves when you perform the alignment depends on the kinds of inaccuracies in sensor mountings. For more information on aligning, see *Aligning Sensors* on page 132.

Standalone Orientations

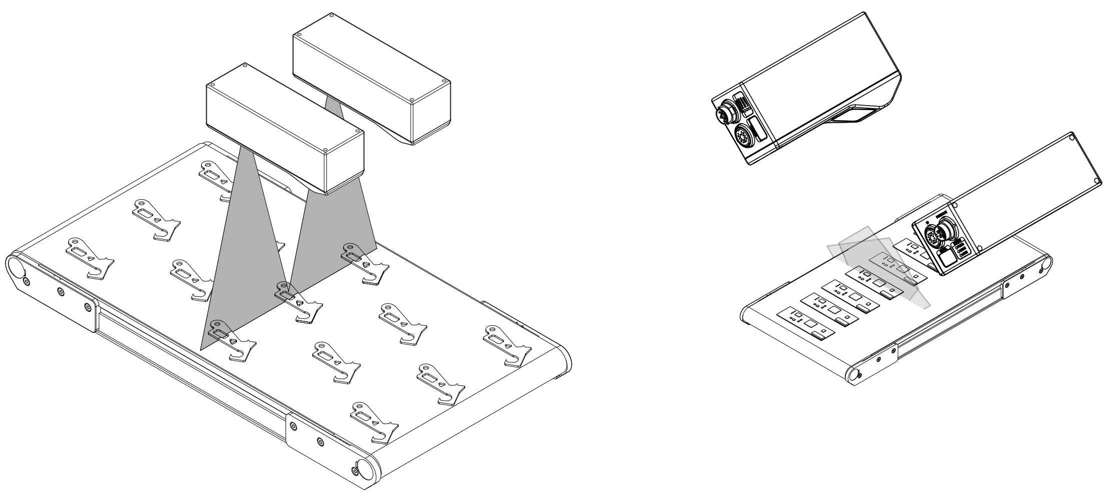


Single sensor above conveyor

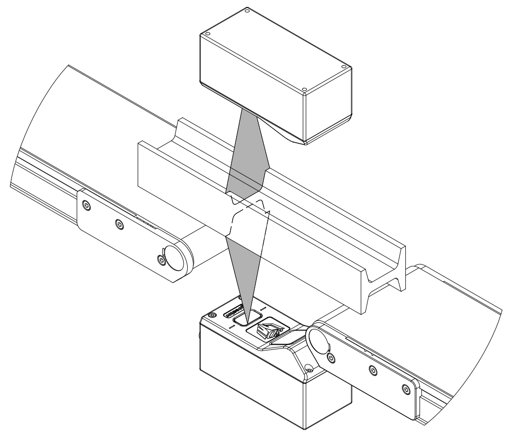


Single sensor on robot arm

Dual-Sensor System Orientations:



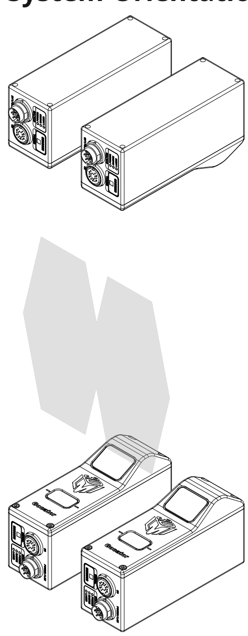
Side-by-side for wide-area measurement (Wide). Sensors can also be angled toward each other, around the Y axis. Sensors can also be mounted with space between their laser lines to scan the width of a large web of material such as metal or rubber (not shown).



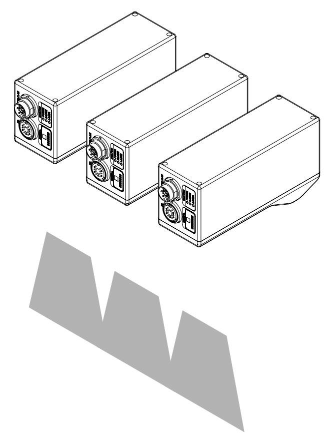
Above/below for two-sided measurement (Opposite)

A multi-sensor system is defined as containing three or more sensors.

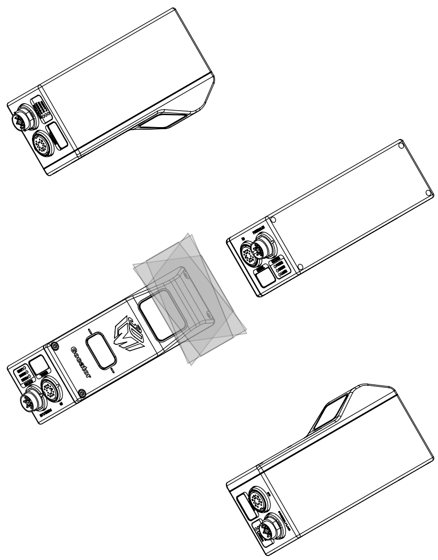
Multi-Sensor System Orientations:



Side-by-side top-bottom (and wide) measurement



Side-by-side for wide-area measurement

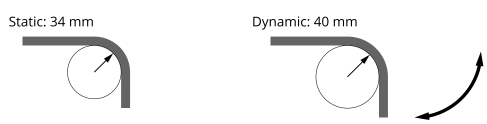


Ring layout for 360-degree scans

Cordset Bend Radius Limits

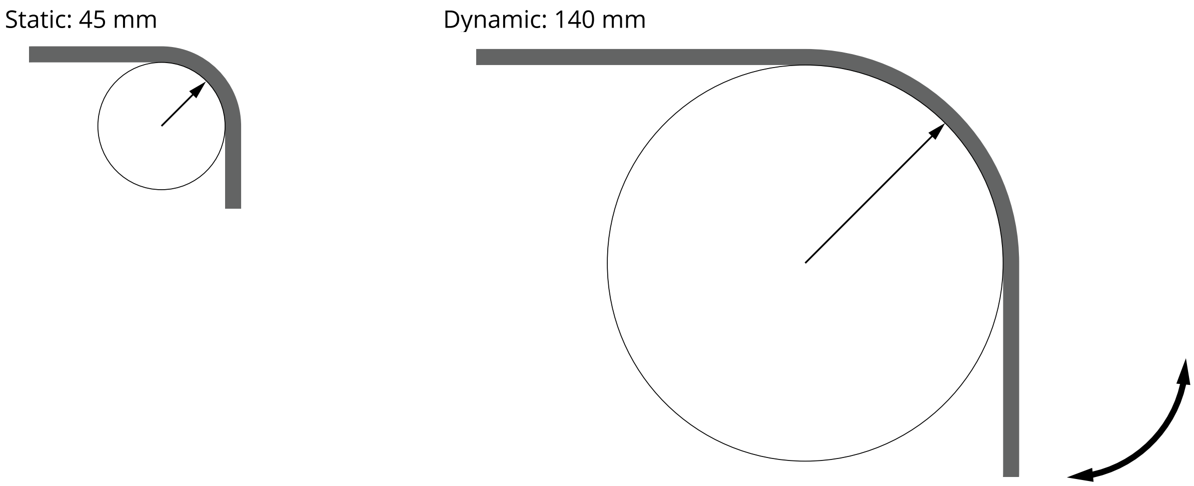
With high flex cordsets of lengths 25 meters and lower, limit bends as follows:

- In installations where a cordset does not bend continuously, limit bending to the static bend radius of 34 mm.
- In installations where a cordset bends continuously, limit bending to the dynamic bend radius of 40 mm.



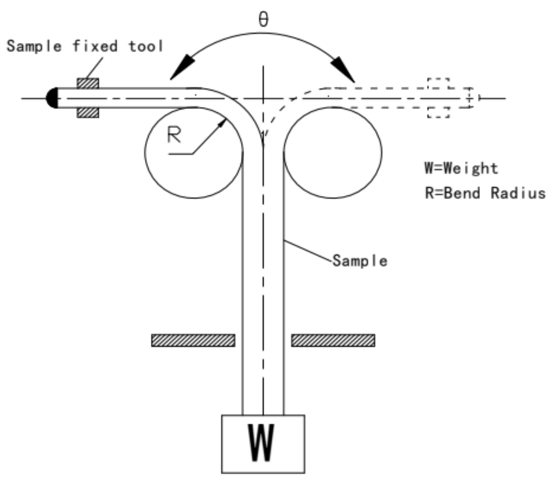
High flex cordset bend radius limits

Custom cordsets between 25 and 60 meters (the maximum length available) have a static bend radius limit of 45 mm and a dynamic limit of 140 mm.

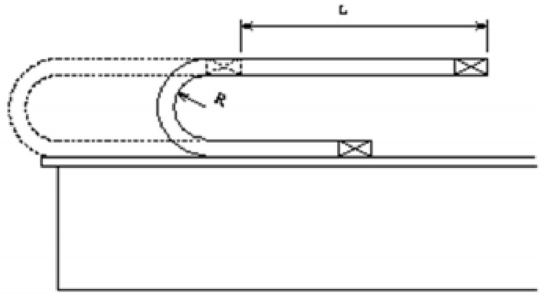


Standard cordset bend radius limits

High flex cordsets are rated for a minimum of 2 million 90° Tick Tock bends and 7 million U-shaped bends, both at the dynamic bend radius limit of 40 mm. The following illustrations show the test setups used to determine the number of bends in high flex cordsets.



Tick-tock test setup (θ = 180°)



U-shape test setup (L = 500 mm).

For cordset part numbers, see *Accessories* on page 1087.

🗋	Standard (non high flex) cordsets, which are no longer available, have a static bend radius limit of 45 mm and a dynamic limit of 140 mm. Standard cordsets are rated for a minimum of 2 million 90° Tick Tock bends.

For more information on cordsets, see *Gocator Cordsets* on page 26.

Grounding

Components of a sensor system should be properly grounded.

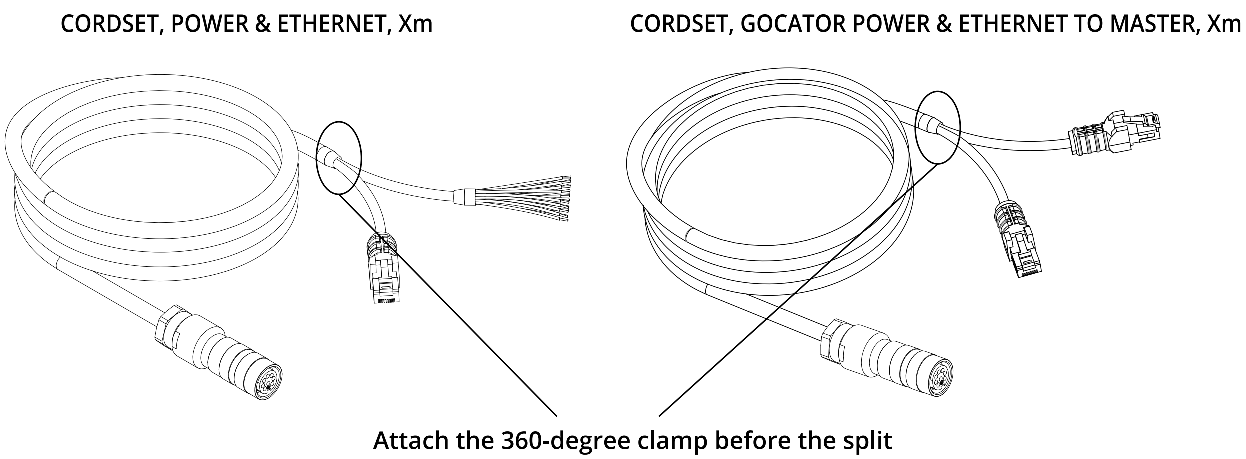
Gocator

Gocator sensors should be grounded to the earth/chassis through their housings and through the grounding shield of the Power I/O cordset. Sensors have been designed to provide adequate grounding through their mounting screws. Always check grounding with a multi-meter to ensure electrical continuity between the mounting frame and the sensor's connectors.

🗋	The frame or electrical cabinet that the sensor is mounted to must be connected to earth ground.

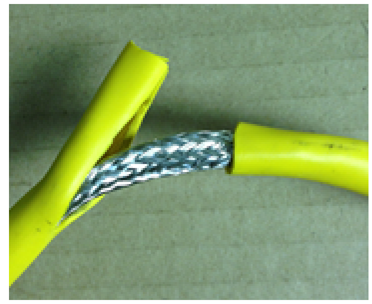
Recommended Practices for Cordsets

If you need to minimize interference with other equipment, you can ground the Power & Ethernet or the Power & Ethernet to Master cordset (depending on which cordset you are using) by terminating the shield of the cordset before the split. The most effective grounding method is to use a 360-degree clamp.



To terminate the cordset's shield:

1. Expose the cordset's braided shield by cutting the plastic jacket before the point where the cordset splits.



2. Install a 360-degree ground clamp.

Master Network Controllers

The rack mount brackets provided with all Masters are designed to provide adequate grounding through the use of star washers. Always check grounding with a multi-meter by ensuring electrical continuity between the mounting frame and RJ45 connectors on the front.

	When using the rack mount brackets, you *must* connect the frame or electrical cabinet to which the Master is mounted to earth ground.

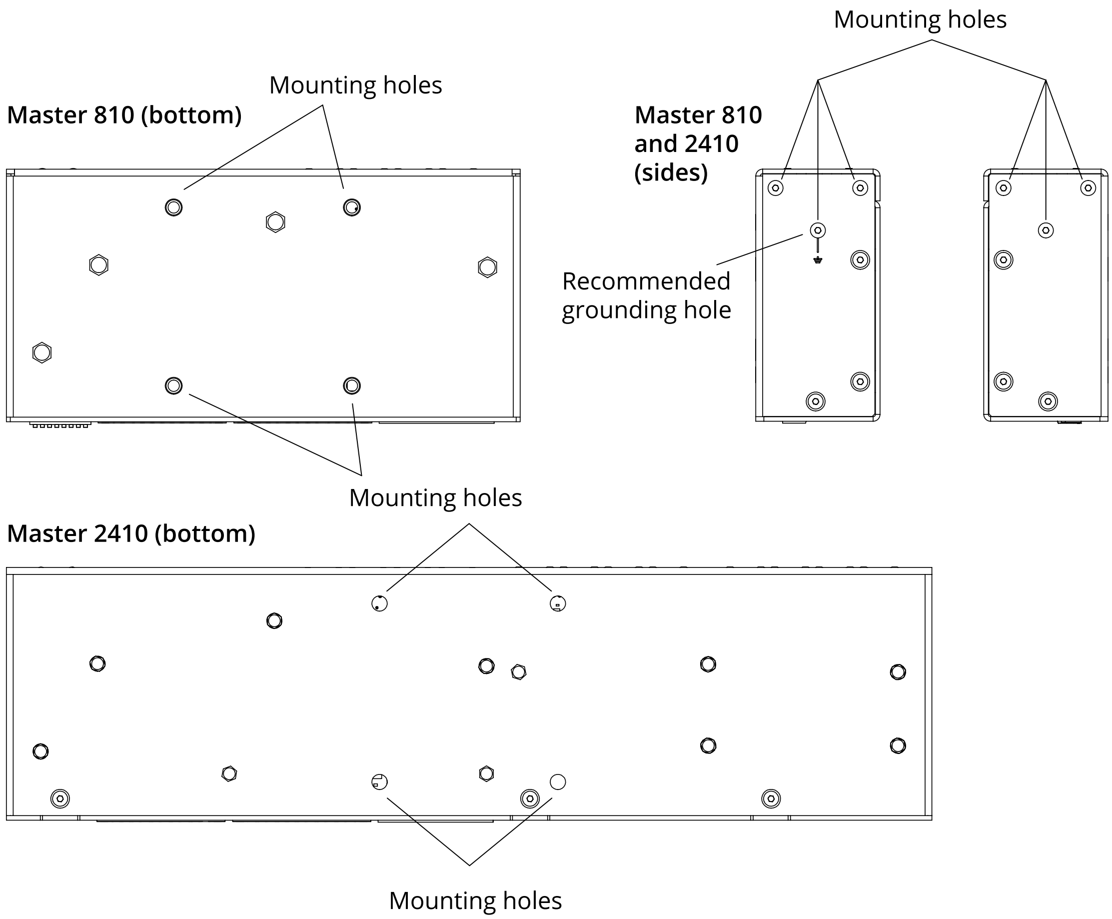
	You *must* check electrical continuity between the mounting frame and RJ45 connectors on the front using a multi-meter.

If you are mounting Master 810 or 2410 using the provided DIN rail mount adapters, you must ground the Master directly; for more information, see *Grounding When Using a DIN Rail (Master 810/2410)* below.

Grounding When Using a DIN Rail (Master 810/2410)

If you are using DIN rail adapters instead of the rack mount brackets, you must ensure that the Master is properly grounded by connecting a ground cable to one of the holes indicated below. The holes on the bottom of the unit accept M4 screws. The holes on the sides of the unit accept M3 screws.

	You can use any of the holes shown below. However, LMI recommends using the holes indicated on the housing by a ground symbol.



An additional ground hole is provided on the rear of Master 810 and 2410 network controllers, indicated by a ground symbol.

Additional Grounding Schemes

Potential differences and noise in a system caused by grounding issues can sometimes cause sensors to reset or otherwise behave erratically. If you experience such issues, see the *Gocator Grounding Guide* (https://downloads.lmi3d.com/gocator-grounding-guide) in the Download center for additional grounding schemes.

Installing DIN Rail Clips: Master 810 or 2410

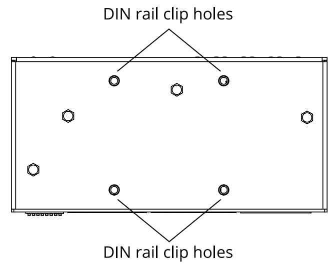
You can mount the Master 810 and 2410 using the included DIN rail mounting clips with M4x8 flat socket cap screws. The following DIN rail clips (DINM12-RC) are included:

Older revisions of Master 810 and 2410 network controllers use a different configuration for the DIN rail clip holes.

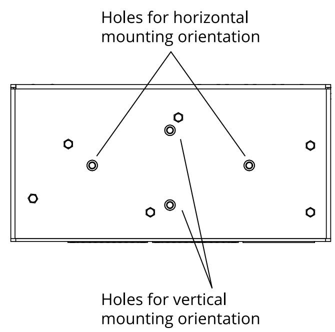
To install the DIN rail clips:

1. Remove the 1U rack mount brackets.
2. Locate the DIN rail mounting holes on the back of the Master (see below).

   Master 810:

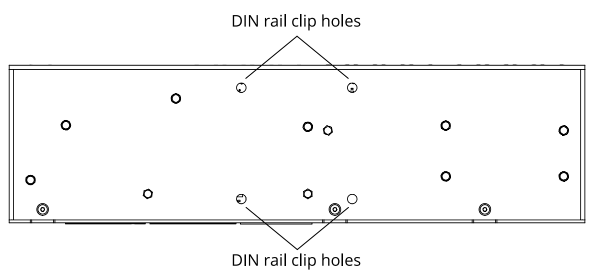


Current revision

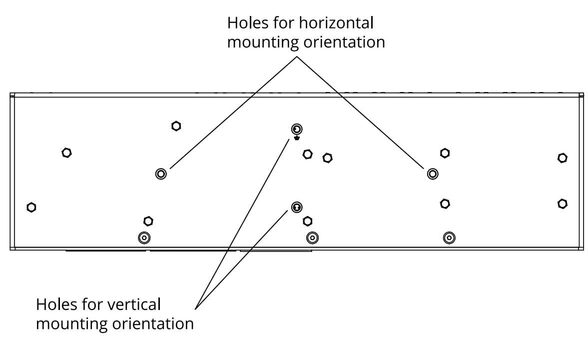


Older revision

   Master 2410:



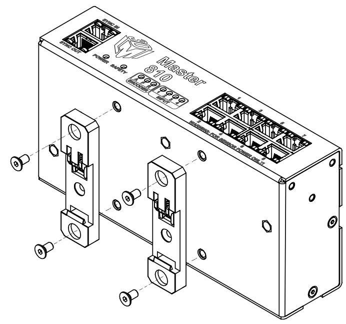
Current revision



Older revision

3. Attach the two DIN rail mount clips to the back of the Master using two M4x8 flat socket cap screws for each one.

   The following illustration shows the installation of clips on a Master 810 (current revision) for horizontal mounting:



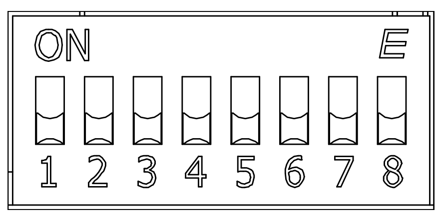
	Ensure that there is enough clearance around the Master for cabling.

Configuring Master 810

If you are using Master 810 with an encoder that runs at a quadrature frequency higher than 300 kHz, you must use the device's divider DIP switches to limit the incoming frequency to 300 kHz.

	Master 810 supports up to a maximum incoming encoder quadrature frequency of 6.5 MHz.

The DIP switches are located on the rear of the device.



	Switches 5 to 8 are reserved for future use.

This section describes how to set the DIP switches on Master 810 to do the following:

- Set the divider so that the quadrature frequency of the connected encoder is compatible with the Master.
- Set the debounce period to accommodate faster encoders.

Setting the Divider

To set the divider, you use switches 1 to 3. To determine which divider to use, use the following formula:

Output Quadrature Frequency = Input Quadrature Frequency / Divider

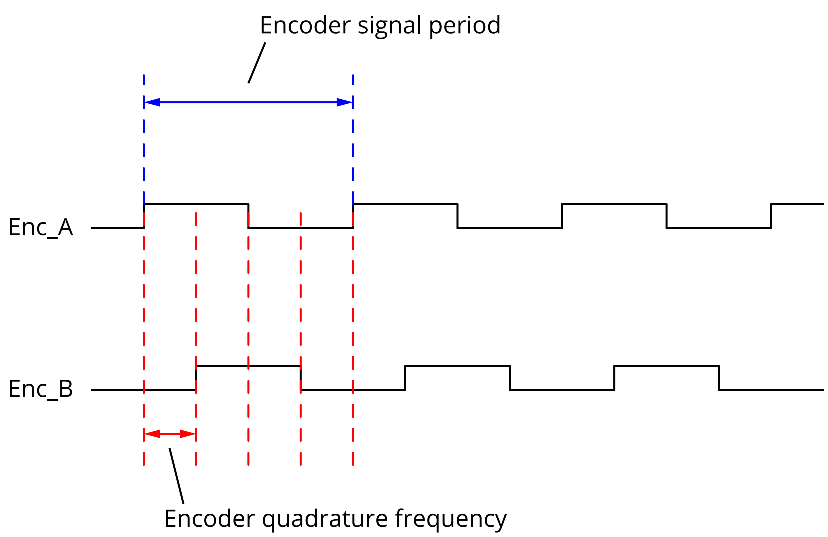
In the formula, use the *quadrature frequency* of the encoder (for more information, see *Encoder Quadrature Frequency* below) and a divider from the following table so that the Output Quadrature Frequency is no more than 300 kHz.

Divider	**Switch 1**	**Switch 2**	**Switch 3**
1	OFF	OFF	OFF
2	ON	OFF	OFF
4	OFF	ON	OFF
8	ON	ON	OFF
16	OFF	OFF	ON
32	ON	OFF	ON
64	OFF	ON	ON
128	ON	ON	ON

> The divider works on debounced encoder signals. For more information, see *Setting the Debounce Period* on the next page.

Encoder Quadrature Frequency

Encoder quadrature frequency is defined as illustrated in the following diagram. It is the frequency of encoder ticks. This may also be referred as the native encoder rate.



You must use a quadrature frequency when determining which divider to use (see *Setting the Divider* on the previous page). Consult the datasheet of the encoder you are using to determine its quadrature frequency.

> Some encoders may be specified in terms of encoder signal frequency (or period). In this case, convert the signal frequency to quadrature frequency by multiplying the signal frequency by 4.

Setting the Debounce Period

If the quadrature frequency of the encoder you are using is greater than 3 MHz, you must set the debounce period to “short.” Otherwise, set the debounce period to “long.”

You use switch 4 to set the debounce period.

Debounce period	Switch 4
short debounce	ON
long debounce	OFF

Network and Sensor Setup

The following sections provide procedures for client PC and sensor network setup.

> DHCP is not recommended for sensors. If you choose to use DHCP, the DHCP server should try to preserve IP addresses. Ideally, you should use static IP address assignment (by MAC address) to do this.

Client Setup

To connect to a sensor from a client PC (using a web browser), you must ensure the client's network card is properly configured. The network ID of the client PC's network card must match that of any sensors you want to connect to.

Sensors are shipped with the following default network configuration:

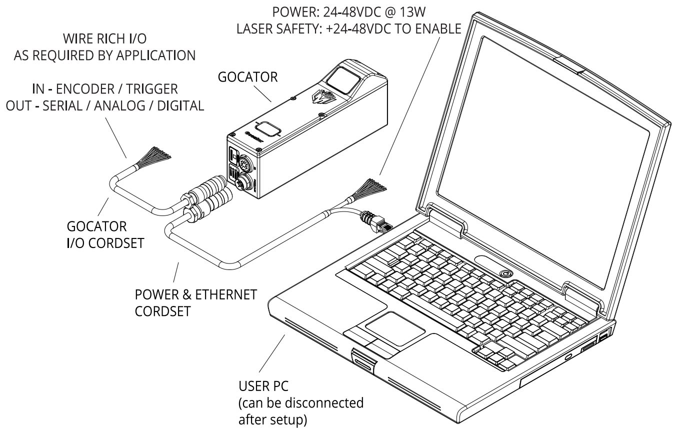
Setting	Default
DHCP	Disabled
IP Address	192.168.1.10
Subnet Mask	255.255.255.0
Gateway	0.0.0.0

> All sensors are configured to 192.168.1.10 as the default IP address. For a dual- or multi-sensor system, the each sensor must be assigned unique addresses before they can be used on the same network. Before proceeding, connect each sensor one at a time (to avoid an address conflict) and use the steps in See *Dual- or Multi-Sensor System* on page 55 to assign each sensor a unique address.

> When connecting cordsets to the sensor's connectors, do not exceed a torque of 2 Nm (18 in-lbs). Ensure that you properly secure the cordset cabling to avoid stress loading on the sensor connectors.

To connect to a sensor for the first time

1. Connect cables and apply power.



For more information on cables, see *Gocator Cordsets* on page 26.

Sensor cabling is illustrated in *System Overview* on page 32.

2. Change the client PC's network settings.

Windows 10

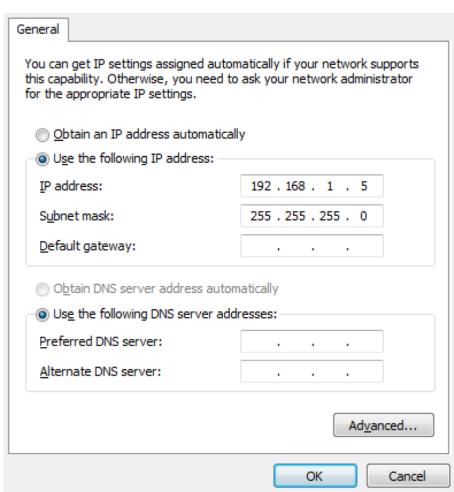
a. From the **Start** menu, launch the Settings app and click **Network & Internet**.

b. Under **Advanced network settings**, click **Change adapter options**.

c. Right-click desired network connection, and then click **Properties**.

d. On the **Networking** tab, click **Internet Protocol Version 4 (TCP/IPv4)**, and then click **Properties**.

e. Select “Use the following IP address” option.



f. Enter IP Address “192.168.1.5” and Subnet Mask “255.255.255.0”, then click **OK**.

Mac OS 11

a. Click Apple menu > **System Preferences**, and then click **Network**.

b. In the list to the left, select **Ethernet**.

c. Click **Advanced**, click **Hardware**, click the **Configure** pop-up menu, and set it to “Manually”.

d. Enter IP Address “192.168.1.5” and Subnet Mask “255.255.255.0”, and then click **Apply**.

⚠ See *Troubleshooting* on page 1089 if you experience any problems while attempting to establish a connection to the sensor.

Gocator Setup

Gocator sensors are shipped with a default configuration that will produce 3D data for most targets.

The following describes how to set up a sensor system for operations. After you have completed the setup, you can perform a scan to verify basic sensor operation.

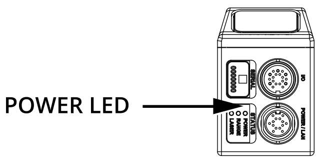
Standalone Sensor System

	When connecting cordsets to the sensor's connectors, do not exceed a torque of 2 Nm (18 in-lbs). Ensure that you properly secure the cordset cabling to avoid stress loading on the sensor connectors.

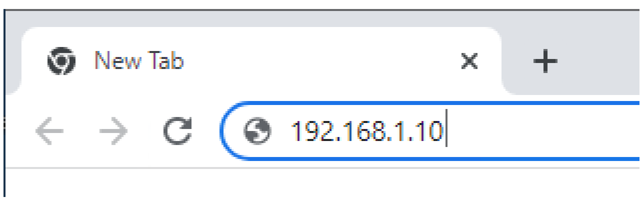
To configure a standalone sensor system

1. Power up the sensor.

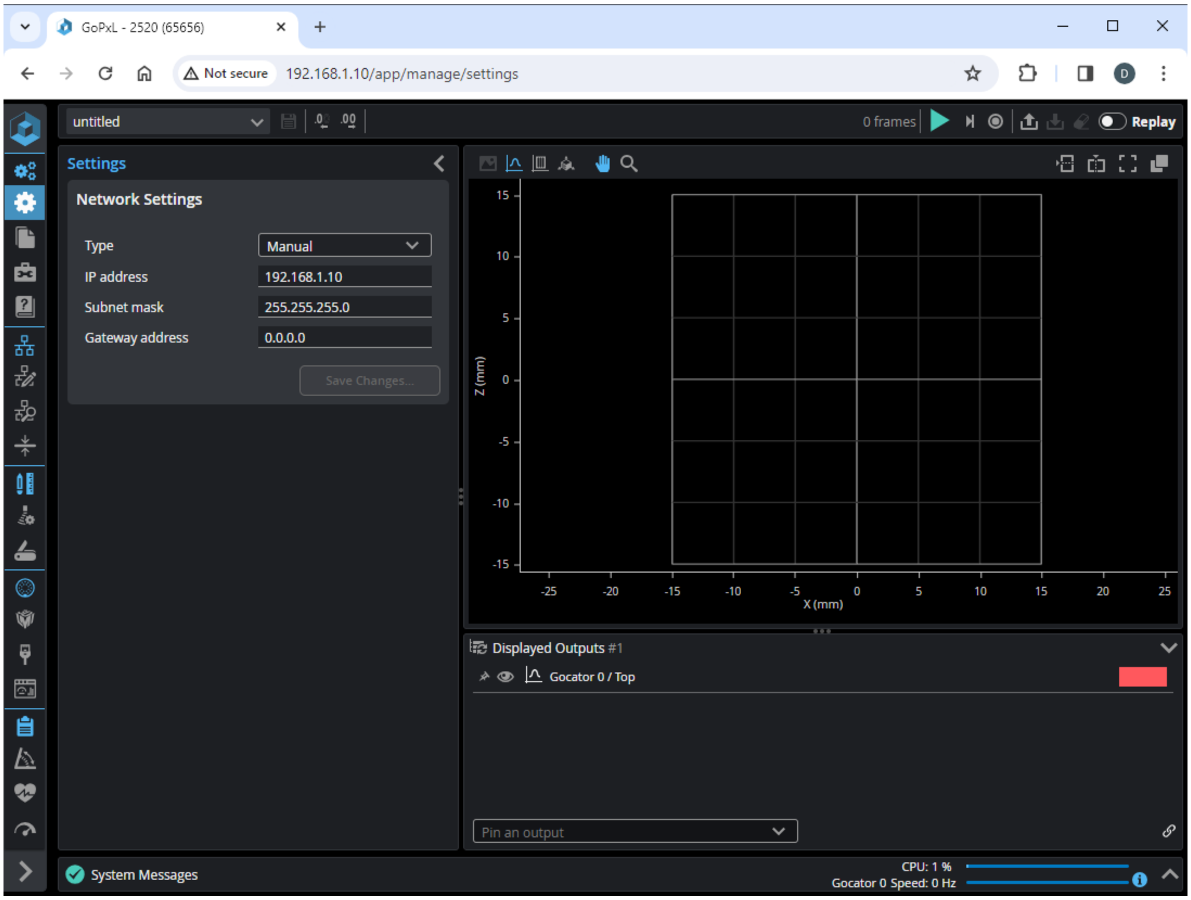
   The power indicator (blue) should turn on immediately.

2. Enter the sensor's default IP address (192.168.1.10) in a web browser.

   The sensor interface loads.



If you can't connect to the sensor's interface, use the Discovery tool to find it on your network (for more information, see *GoPxL Discovery Tool* on page 935) and use the IP address of the sensor.

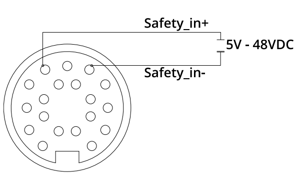
3. Ensure that Replay mode is off

   Replay mode is off when the Replay toggle above the data viewer is to the left.

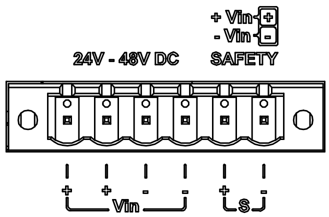
   For an overview of the user interface, see *User Interface Overview* on page 83.

4. Ensure that the Laser Safety input is high.

   For a sensor not connected to a Master:

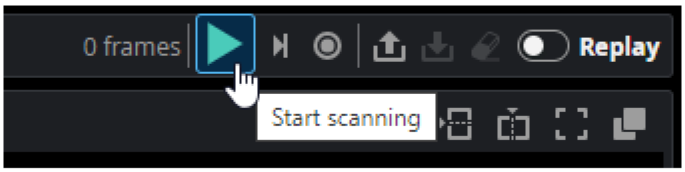


   For a sensor connected to a Master 810 or 2410:

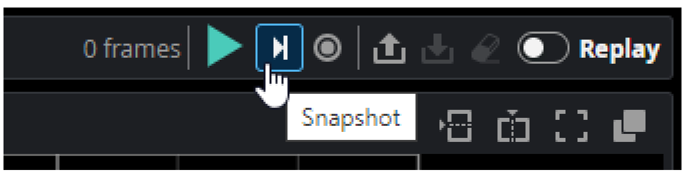


5. Move a target under the sensor within its measurement range and field of view.
6. Press the "Start scanning" or the Snapshot button on the Toolbar to start the sensor (FOV).

   The Start button is used to run a sensor continuously.

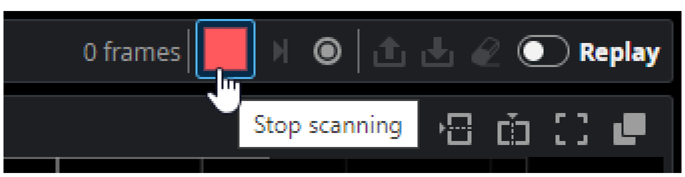
   

   The Snapshot button is used to trigger the capture of a single frame.

   If a target object is within the sensor's measurement range and FOV, the data viewer will display scan data, and the sensor's range indicator will illuminate.

   If no scan data is displayed in the data viewer, see *Troubleshooting* on page 1089.
7. If you started the sensor using the "Start scanning" button, press the Stop button.

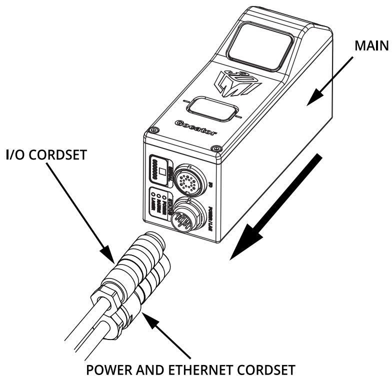
Dual- or Multi-Sensor System

All sensors are shipped with a default IP address of 192.168.1.10. To set up and connect to a dual-sensor or multi-sensor system you must make sure each sensor has a unique IP address.

> When connecting cordsets to the sensor's connectors, do not exceed a torque of 2 Nm (18 in-lbs). Ensure that you properly secure the cordset cabling to avoid stress loading on the sensor connectors.

To configure a dual- or multi-sensor system

1. Make sure all sensors are unplugged.



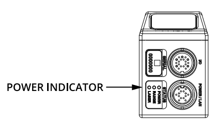
2. Decide which sensor will be the "main" sensor with which you will group other sensors.

   This is the sensor on which GoPxL will run and to which you will connect to configure your multi-sensor system later.

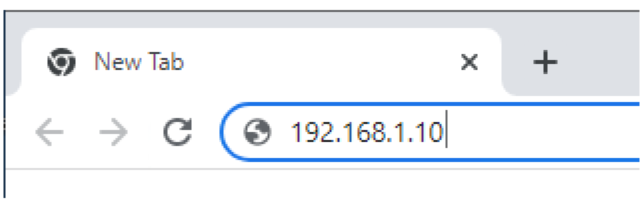
   You can leave this sensor's IP address at its default of 192.168.1.10. If you need to change it to work on your network, follow the instructions below to change this sensor's IP address first.

3. Power up one of the other sensors (or the "main" sensor if you must change its IP address) by connecting the Power and Ethernet cordset.

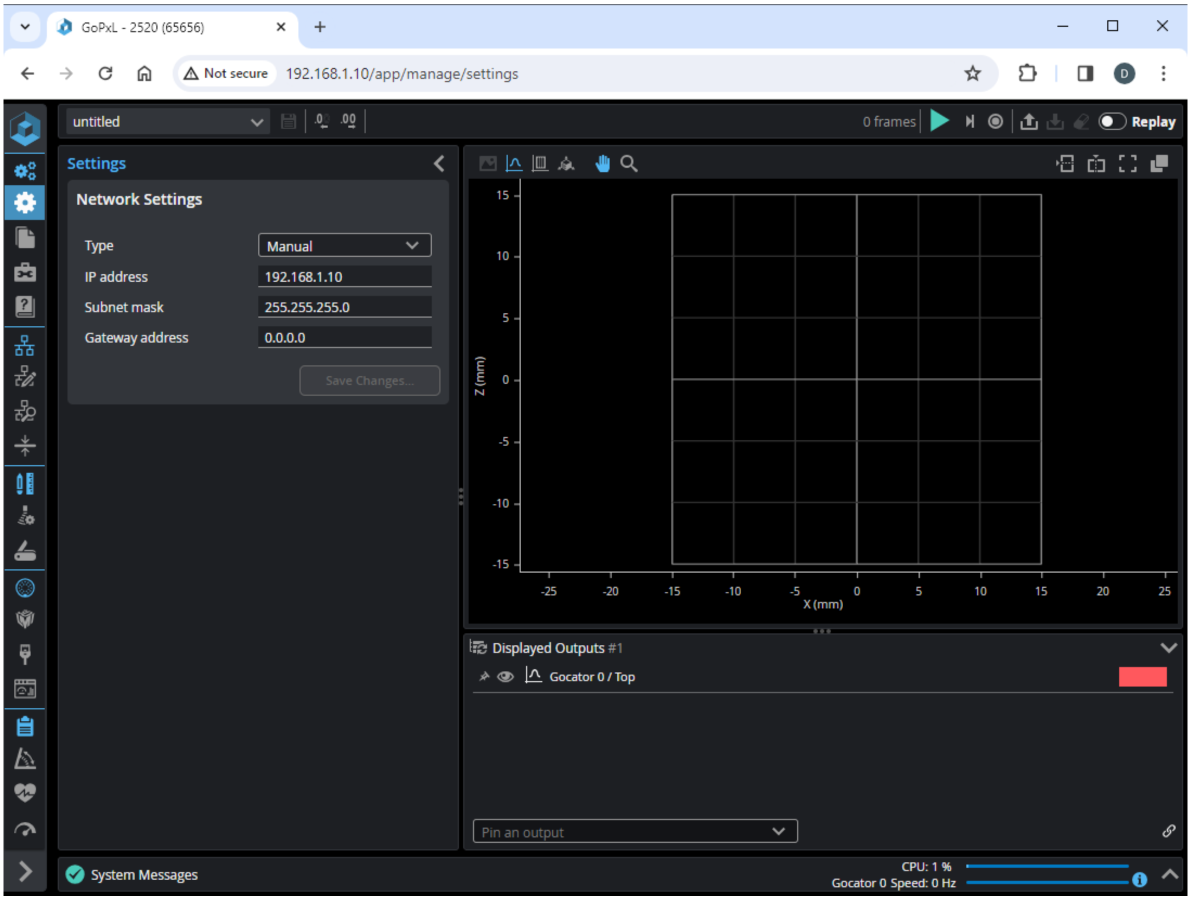
   The power LED (blue) of the sensor should turn on immediately.

4. Enter the sensor's default IP address (192.168.1.10) in a web browser.

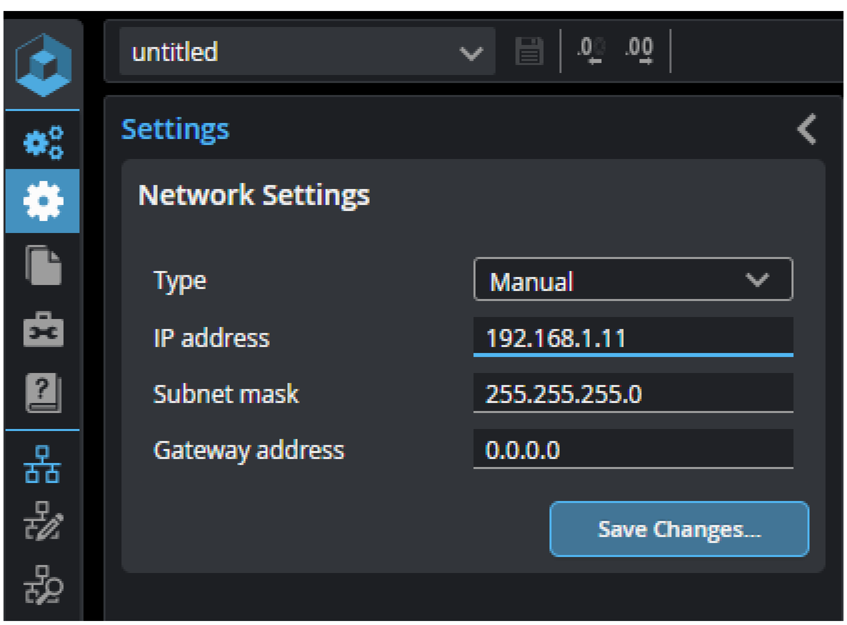
   

   The sensor's web interface loads in the browser.

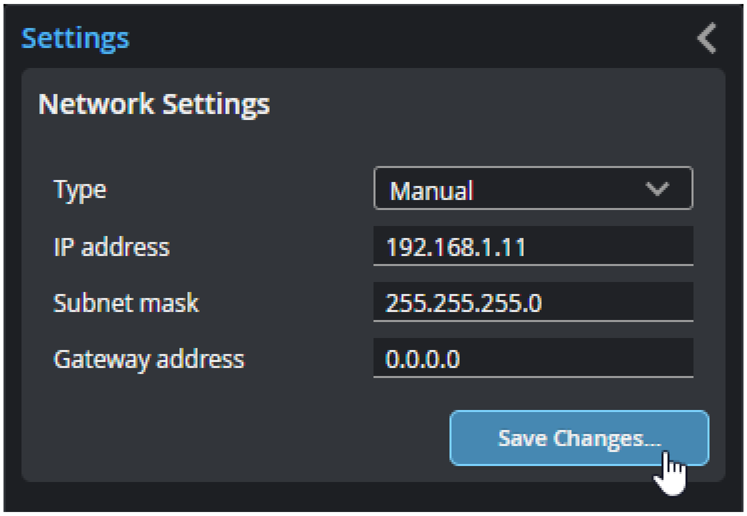


5. Move your mouse over the navigation bar along the left of the interface to expand it, and make sure the Settings page is selected.

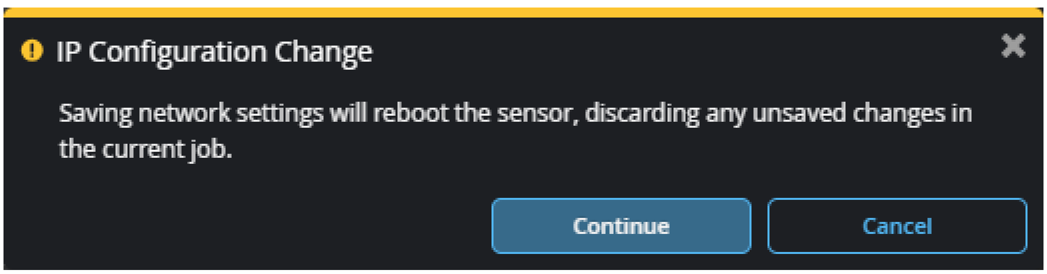
6. In the **Network Settings** panel, change the IP address to the desired address and press Tab or click outside of the **IP Address** field.



7. Make note of or copy the new address in the **IP Address** field.
8. Click **Save Changes**.

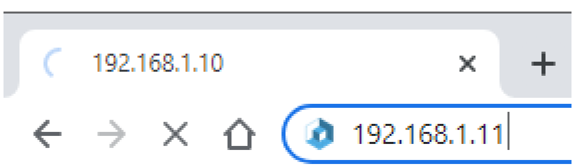


9. To confirm that you want to make the change, click **Continue**.



The sensor restarts.

10. Wait several seconds.
11. Type or paste the new IP address in the web browser's address field.



The sensor's web interface loads, using the new IP address.

12. Power down the sensor
13. For each additional sensor, perform steps 3 to 12.
14. After you have changed the IP addresses of the sensor so that they all have unique IP addresses, see *Creating a Sensor System* on page 124 to group the sensors in GoPxL.

Required Ports

The table below lists the ports used by the following:

- sensors
- the Ethernet-based protocols
- the SDK and the REST API
- PC instances of GoPxL

Use this information to decide whether you or your network administrator needs to open network ports and to understand the traffic that a sensor system will produce over a network.

Ports used by GoPxL

Port	Data Packet Protocol	Description
80	TCP	GoPxL web port for GoPxL GUI. **Sensors and GoMax NX**: All devices use this port. **For GoPxL running on a PC**: Use port 8100; for more information, see below.
502	TCP	Used by the GoPxL Modbus protocol as the listener port.
2222	UDP	Used by the EtherNet/IP protocol for implicit messaging.
3192	TCP	FireSync physical sensor upgrade port.
3194	TCP	FireSync physical sensor health port.
3320	UDP	GoPxL discovery protocol server listener port
3500	UDP	GoPxL Remote Procedure Call port.
3600	TCP	GoPxL control port. **Physical sensors and GoMax NX**: All devices use this port. **GoPxL running on a PC**: By default, the first PC instance starts at this port number, and subsequent instances are offset (for example, 3600, 3620, and so on). You can set custom values in GoPxL Manager. For more information, see *Running GoPxL on a Windows PC* on page 828.
3601	TCP	Gocator Data Protocol (GDP) port. **Physical sensors and GoMax NX**: All devices use this port. **GoPxL running on a PC**: This port is offset from the control port by +1 (for example, 3601, 3621, and so on).
8080	TCP	Default accelerated sensor web server port number.
8100		GoPxL web port for GoPxL GUI for PC instances of GoPxL. (Sensors and GoMax NX units use port 80.) By default, the first PC instance starts at this port number, and subsequent instances are offset (for example, 8100, 8120, and so on). You can set custom values in GoPxL Manager. For more information, see *Running GoPxL on a Windows PC* on page 828.
8190	TCP	Used by GoPxL Ethernet for default ASCII control, data, and health port. Actual ports can be configured by the user.
44818	TCP and UDP	Used by the GoPxL EtherNet/IP protocol as a UDP and TCP listener port for explicit messaging.

Key Concepts

The following sections provide an overview of how devices acquire and produce data, detect and measure parts, and control devices such as PLCs. Some of these concepts are important for understanding how you should mount sensors and configure settings such as active area.

Sensors, Sensor Groups, and Systems

In GoPxL, the term *sensor* refers to a single device. A *sensor group* contains two or more sensors of the same model. The term "system" refers to the sensors or sensor groups as a whole. Currently GoPxL only supports a single sensor group in a system.

> In this document, we will use the term multi-sensor to describe systems with *two* or more sensors.

In a multi-sensor system, the scan data of the individual sensors in a sensor group is combined. So for example, in a multi-sensor G2 system, the profiles from the individual sensors are combined into a single, wider profile.

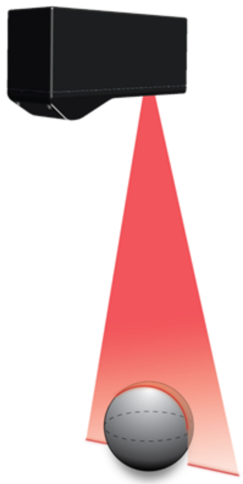
Sensors and sensor groups also have an associated "scan engine." For example, when designing a sensor system containing one or more G2 sensors, the associated scan engine is "Gocator Laser Profiler," with acquisition settings specific to that family of sensors.

> You may still see the term "scanner" in the GoPxL UI, SDK, and REST API. "Scanner" is equivalent to "sensor group."

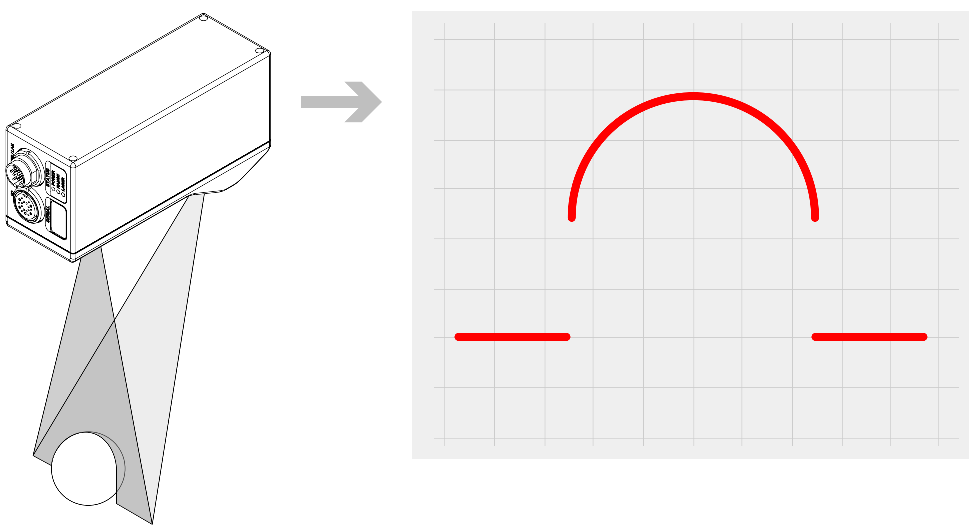
3D Acquisition

After a sensor system has been set up and is running, it is ready to start capturing 3D data.

Laser profile sensors project a laser line onto the target.



The sensor's camera views the laser line on the target from an angle and captures the reflection of the laser light off the target. The camera captures a single 3D profile—a slice, in a sense—for each camera exposure. The reflected laser light falls on the camera at different positions, depending on the distance of the target from the sensor. The sensor’s laser emitter, its camera, and the target form a triangle. The sensor uses the known distance between the laser emitter and the camera, and two known angles—one of which depends on the position of the laser light on the camera—to calculate the distance from the sensor to the target. This translates to the height of the target. This method of calculating distance is called *laser triangulation*.



Target objects typically move on a conveyor belt or other transportation mechanism under a sensor mounted in a fixed position. Sensors can also be mounted on robot arms and moved over the target. In both cases, the sensor captures a series of 3D profiles, building up a full scan of the target. Sensor speed and required exposure time to measure the target are typically critical factors in applications with line profile sensors.

	Gocator sensors are always pre-calibrated to deliver 3D data in engineering units (mm) throughout their measurement range.

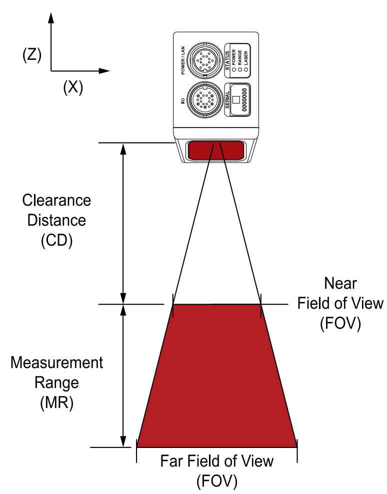
Clearance Distance, Field of View and Measurement Range

Clearance distance (CD), field of view (FOV), and measurement range (MR) are important concepts for understanding the setup of a sensor and for understanding results.

Clearance distance – The minimum distance from the sensor that a target can be scanned and measured. A target closer than this distance will result in invalid data.

Measurement range – The vertical distance, starting at the end of the clearance distance, in which targets can be scanned and measured. Targets beyond the measurement range will result in invalid data.

Field of view – The width on the X axis along the measurement range. At the far end of the measurement range, the field of view is wider, but the X resolution and Z resolution are lower. At the near end, the field of view is narrower, but the X resolution is higher. When resolution is critical, if possible, place the target closer to the near end. (For more information on the relation between target distance and resolution, see *Z Resolution* on the next page.)



Resolution and Linearity

X Resolution

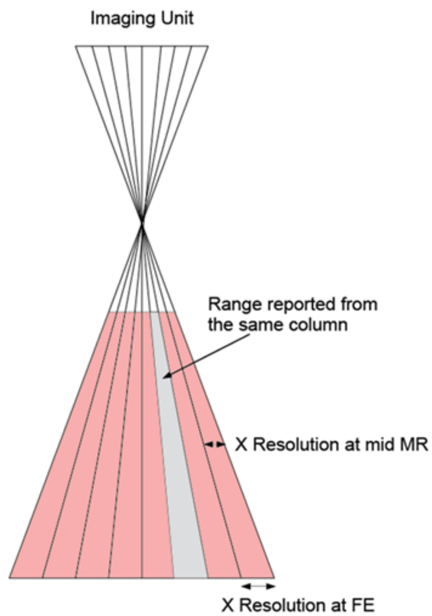
X resolution is the horizontal distance between each measurement point along the laser line. This specification is based on the number of camera columns used to cover the field of view (FOV) at a particular measurement range.

Because the FOV is trapezoidal (shown in red, below), the distance between points is closer at the near range than at the far range. This is reflected in the Gocator data sheet as the two numbers quoted for X resolution.

X Resolution is important for understanding how accurately width on a target can be measured.

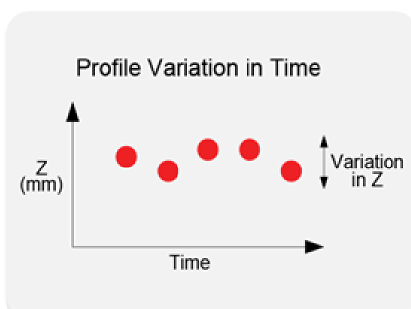
	When the sensor runs in Profile mode and **Uniform Spacing** is enabled, the 3D data is

resampled to an X interval that is different from the raw camera resolution. For more information, see *Uniform Data and Point Cloud Data* on page 70.



Z Resolution

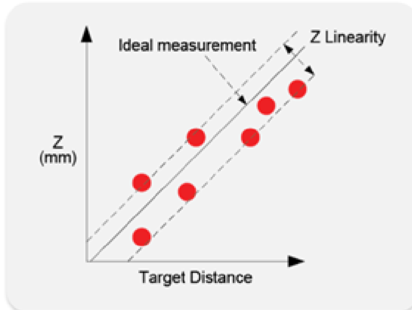
Z Resolution gives an indication of the smallest detectable height difference at each point, or how accurately height on a target can be measured. Variability of height measurements at any given moment, in each individual 3D point, with the target at a fixed position, limits Z resolution. This variability is caused by camera and sensor electronics.



Like X resolution, Z resolution is better closer to the sensor. This is reflected in the Gocator datasheets as the two numbers quoted for Z resolution.

Z Linearity

Z linearity is the difference between the actual distance to the target and the measured distance to the target, throughout the measurement range. Z linearity gives an indication of the sensor's ability to measure absolute distance.



Z linearity is expressed in the Gocator data sheet as a percentage of the total measurement range.

Profile Output

Gocator sensors represent a profile as a series of ranges, with each range representing the distance from the origin. Each range contains a height (on the Z axis) and a position (on the X axis) in the sensor's field of view.

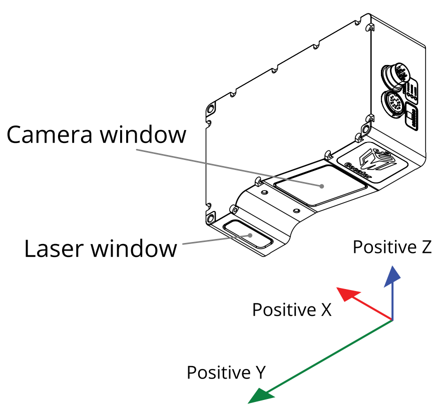
Coordinate Systems

Data points are reported in one of two coordinate systems, which depends on the alignment state of the sensor.

- **Unaligned (sensor) coordinates**: Used on unaligned sensors.
- **Aligned (system) coordinates**: Used on aligned sensors. Applies to either standalone or multi-sensor systems.

Understanding coordinate systems is an important part of understanding measurement results.

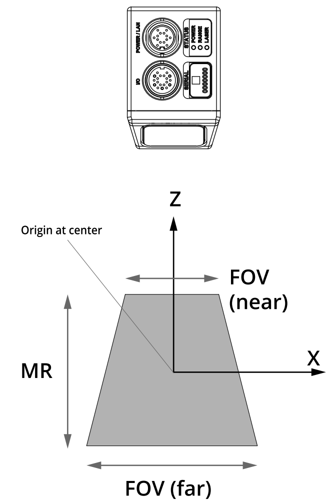
For all G2 Gocator line profile sensors, Y increases moving *from the camera to the laser*. You should *not* use the location of the connectors to determine the orientation of positive Y of your sensor.



For more information, see the illustrations of the coordinate system orientation of your sensor in the specification drawings for your sensor in *Sensors* on page 975.

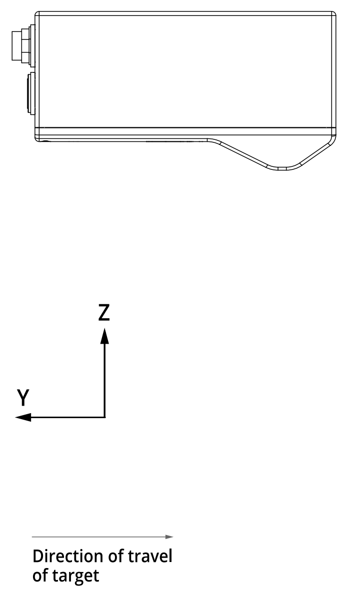
Unaligned Coordinates

Unaligned sensors use *sensor* coordinates, that is, the coordinate system is relative to the sensor itself. The measurement range (MR) is along the Z axis. The sensor’s field of view (FOV) is along the X axis. Most importantly, the origin is at the *center* of the measurement range and field of view, in other words, the center of the scan area.



Gocator 2130/2330 sensor

The Y axis represents the relative position of the target part in the direction of travel. Y position increases as the target moves forward (increasing encoder position). Typically, the direction of travel of the target is opposite the sensor's positive Y axis. If it isn't (due to the orientation of the sensor's mounting), you need to configure the sensor's orientation to be "reversed." For more information, see *Creating a Sensor System* on page 124.



Gocator 2130/2330 sensor

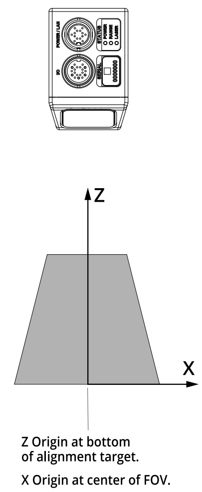
Aligned Coordinates

Understanding aligned coordinates is important for two reasons. First, they are the direct result of performing the built-in alignment procedure. Second, they change how scan data is represented and how measurement results should be interpreted. For more information on aligning sensors, see *Aligning Sensors* on page 132.

The adjustments resulting from alignment are called *transformations* (offsets along the axes and rotations around the axes). Transformations are displayed in the **System** > **Alignment** page. Note that currently, you can ignore Sensor Group Transforms, which represent the transformations between *groups* of sensors; support for this will be available in future versions of the sensor software. For more information on transformations in the web interface, see *Transformations* on page 156.

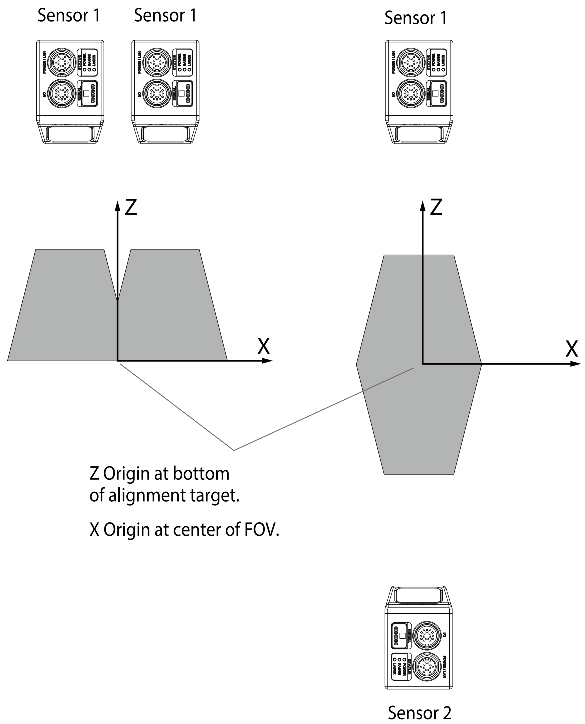
In aligned coordinates, the X axis is parallel to the alignment target surface. The system Z origin is set to the base of the alignment target object. In both cases, alignment determines the offsets in X and Z.

Alignment is used with a single sensor to compensate for mounting misalignment and to set a zero reference, such as a conveyor belt surface.



Gocator 2130/2330 sensor

Additionally, in multi-sensor systems, alignment sets a common coordinate system. That is, scan data and measurements from the sensors are expressed in a unified coordinate system.

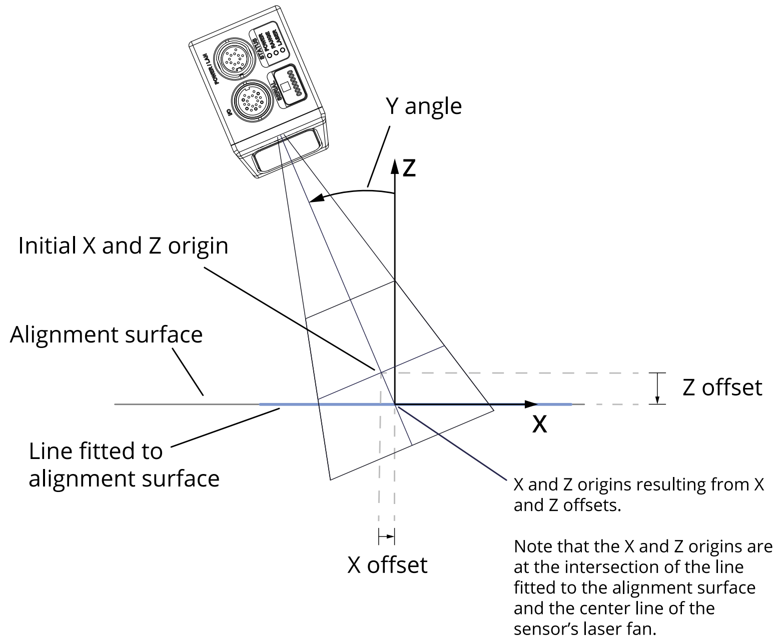


Gocator 2130/2330 sensors

Alignment can also determine offsets along the Y axis. This allows setting up a staggered layout in multi-sensor systems. This is especially useful in side-by-side mounting scenarios, as it provides full coverage for models with a small scan area.

As with sensor coordinates, in system coordinates, Y position increases as the object moves forward (increasing encoder position).

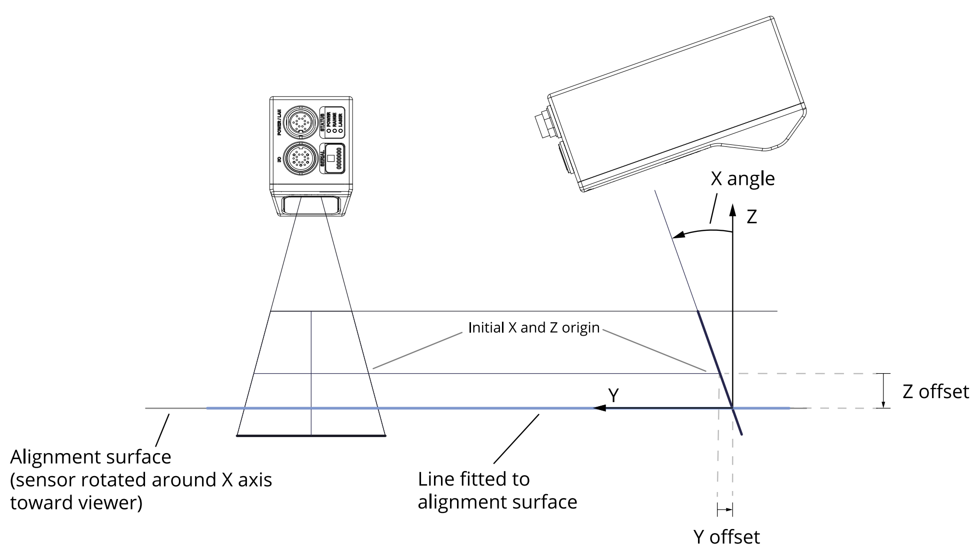
Alignment also determines the Y Angle (angle on the X–Z plane, around the Y axis) needed to align sensor data. This is also sometimes called *roll correction*.



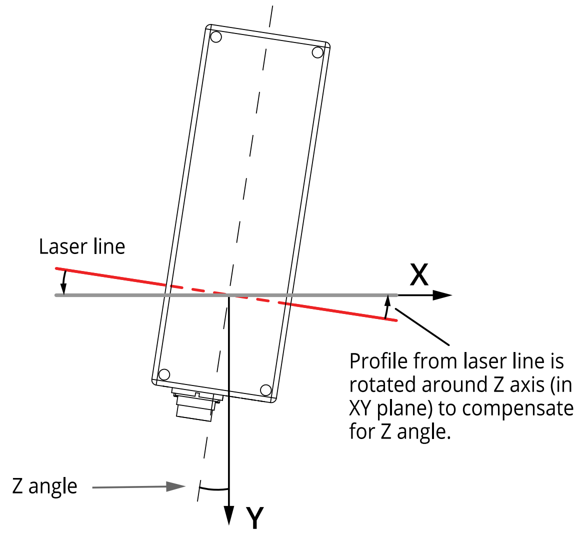
Gocator 2130/2330: Y Angle

Y angle is positive when rotating from positive X to positive Z axis.

Similarly, tilt can be determined around the Z and the X axis, which compensates for the angle in height measurements. These are sometimes called *yaw correction* and *pitch correction*, respectively. Intentional rotation around the X axis is often used for specular mounting, that is, for scanning targets that are shiny or reflective. Note however that X angle correction can't currently be corrected for using the alignment procedure available on the Alignment panel. X angle can only be manually entered in the Transformations panel. For more information on transformations in the web interface, see *Transformations* on page 156.



Gocator 2130/2330: X Angle



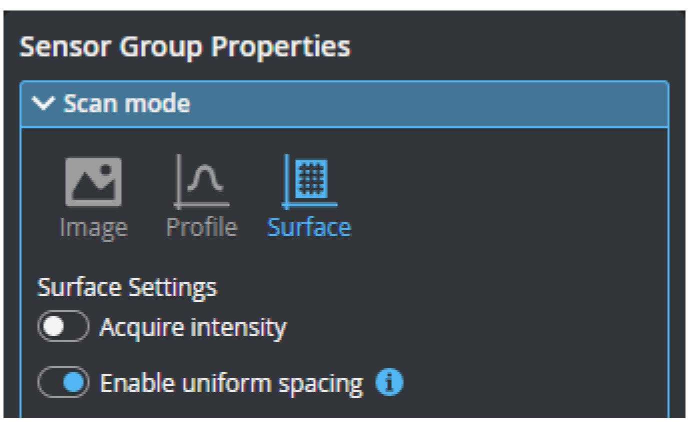
Gocator 2130/2330 sensor: Z Angle

X angle is positive when rotating from positive Y to positive Z. Z angle is positive when rotating from positive X to positive Y.

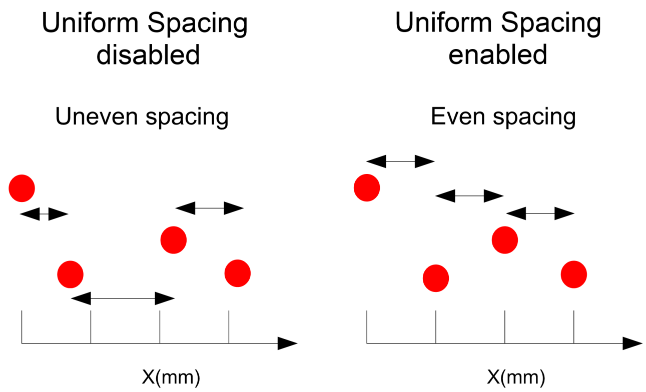
When applying the transformations, the data is first rotated around X (clockwise, with the X axis toward the viewer), then Y (counterclockwise), and then Z (clockwise), and then the offsets are applied.

Uniform Data and Point Cloud Data

The data that a sensor produces in Profile mode is available in two formats: as uniform (resampled) data and as point cloud data (previously called "raw"). The sensor produces uniform data when the **Enable uniform spacing** option is enabled. The sensor produces point cloud data when the option is disabled. The setting is available in **Scan Mode**, on the **Acquire** page.



When **Enable uniform spacing** is enabled, the ranges that make up a profile are resampled so that the spacing is uniform along the X axis. The resampling divides the X axis into fixed size "bins." Profile points that fall into the same bin are combined into a single range value (Z). You can set the size of the spacing interval.



Resampling to uniform spacing reduces the complexity for downstream algorithms to process the profile data from the sensor, but places a higher processing load on the sensor's CPU.

When uniform spacing is not enabled, no processing is required on the sensor. This frees up processing resources in the sensor, but usually requires more complicated processing on the client side. Ranges in this case are reported in (X, Z) coordinate pairs.

A drawback of uniform spacing is that if sensors are angled to scan the sides of a target, data on the "verticals" is lost because points falling in the same "bin" are combined. When **Enable uniform spacing** is disabled, however, all points are preserved on the sides. In this case, the data can be processed by the subset of tools that work on profiles without uniform spacing. Alternatively, the data can be processed externally using the SDK.

🗋	When uniform spacing is enabled, in the Ethernet output, only the range values (Z) are reported. The X positions can be reconstructed through the array index at the receiving end (the client).

For information on enabling uniform spacing, see *Scan Modes and Intensity* on page 195.

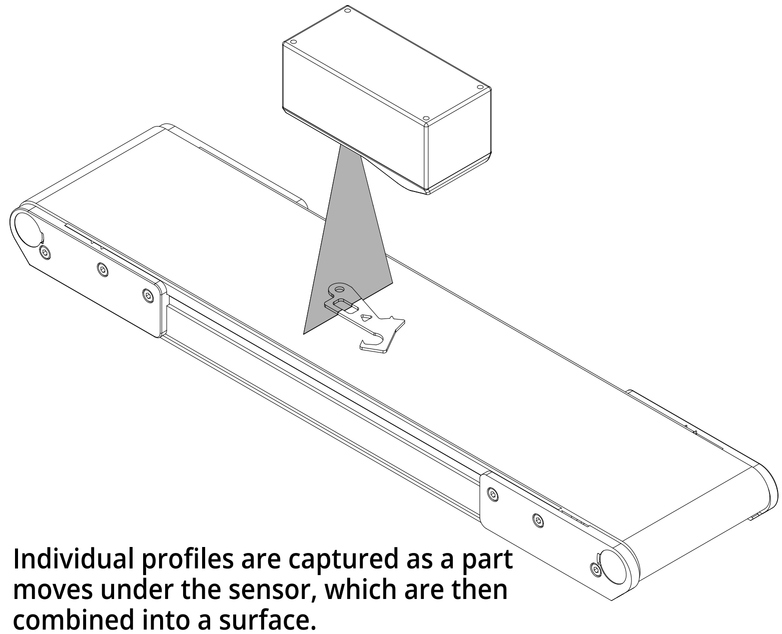
Data Generation and Processing

After scanning a target, a sensor can process the scan data to allow the use of more sophisticated measurement tools. This section describes the following concepts:

- Surface generation
- Part detection
- Sectioning

Surface Generation

Profile sensors create a single profile with each exposure. GoPxL can combine the series of profiles gathered as a target moves under the sensor to generate Surface data of the entire target.

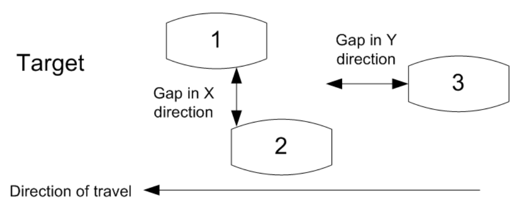


For more information, see *Surface Generation* on page 199.

Part Detection

After a sensor has generated surface data by combining single exposures, GoPxL can isolate discrete parts on the generated surface into separate scans representing parts using the Profile Part Detection tool. For more information, see Continuous in *Surface Generation* on page 199.

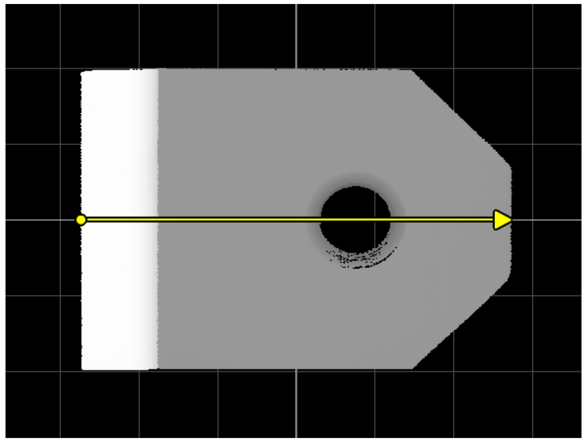
Gocator can then perform measurements on these isolated parts.



Part detection is useful when measurements on individual parts are needed and for robotic pick and place applications.

Sectioning

In Surface mode, the sensor can also extract a profile from a surface or part using a line you define on that surface or part. The resulting profile is called a "section." A section can have any orientation on the surface, but its profile is parallel to the Z axis.

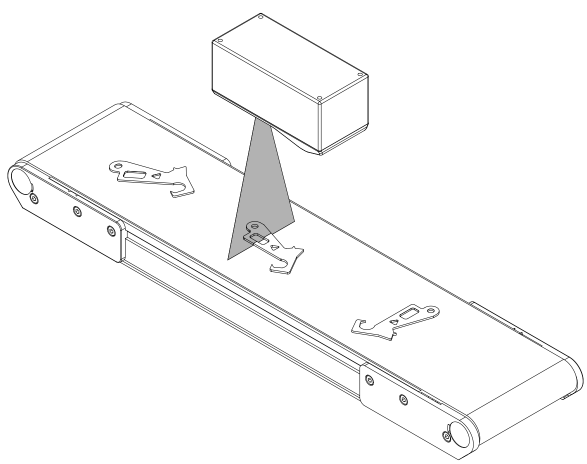


You can use most of Gocator's profile measurement tools on a section, letting you perform measurements that are not possible with surface measurement tools.

For more information on sections, see *Surface Section* on page 628.

Part Matching

The sensor can match scanned parts to the edges of a model based on a previously scanned part using the Surface Pattern Matching tool (for more information, see *Surface Pattern Matching* on page 599) or to the dimensions of a fitted bounding box that encapsulates the part data (for more information see *Surface Bounding Box* on page 466). When parts match, the sensor can rotate scans so that they are all oriented in the same way, using Surface Transform (for more information, see *Surface Transform* on page 672). This allows measurement tools to be applied consistently to parts, regardless of the orientation of the part you are trying to match.



Measurement and Anchoring

After GoPxL scans a target and, optionally, further processes the data, the sensor is ready to take measurements on the scan data.

GoPxL provides dozens of measurement tools, each of which provides a set of individual measurements, giving you dozens of measurements ideal for a wide variety of applications to choose from. The configured measurements start returning pass/fail decisions, as well as the actual measured values, which are then sent over the enabled outputs to control devices such as PLCs, which can in turn control ejection or sorting mechanisms. (For more information on measurements and configuring measurements, see *Tools - Measurement and Processing* on page 232.

A part's position can vary on a transport system. To compensate for this variation, Gocator can anchor a measurement to the positional measurement (X, Y, or Z) or Z angle of an easily detectable feature, such as the edge of a part. The calculated offset between the two ensures that the anchored measurement will always be properly positioned on different parts. For more information, see *Measurement Anchoring* on page 264.

Tool Chaining

GoPxL's measurement and processing tools can be linked together: one tool uses another tool's output as input. This gives you a great deal of control and flexibility when it comes to implementing your application.

The following sections describe the types of output and how you use them as input.

Tool Data

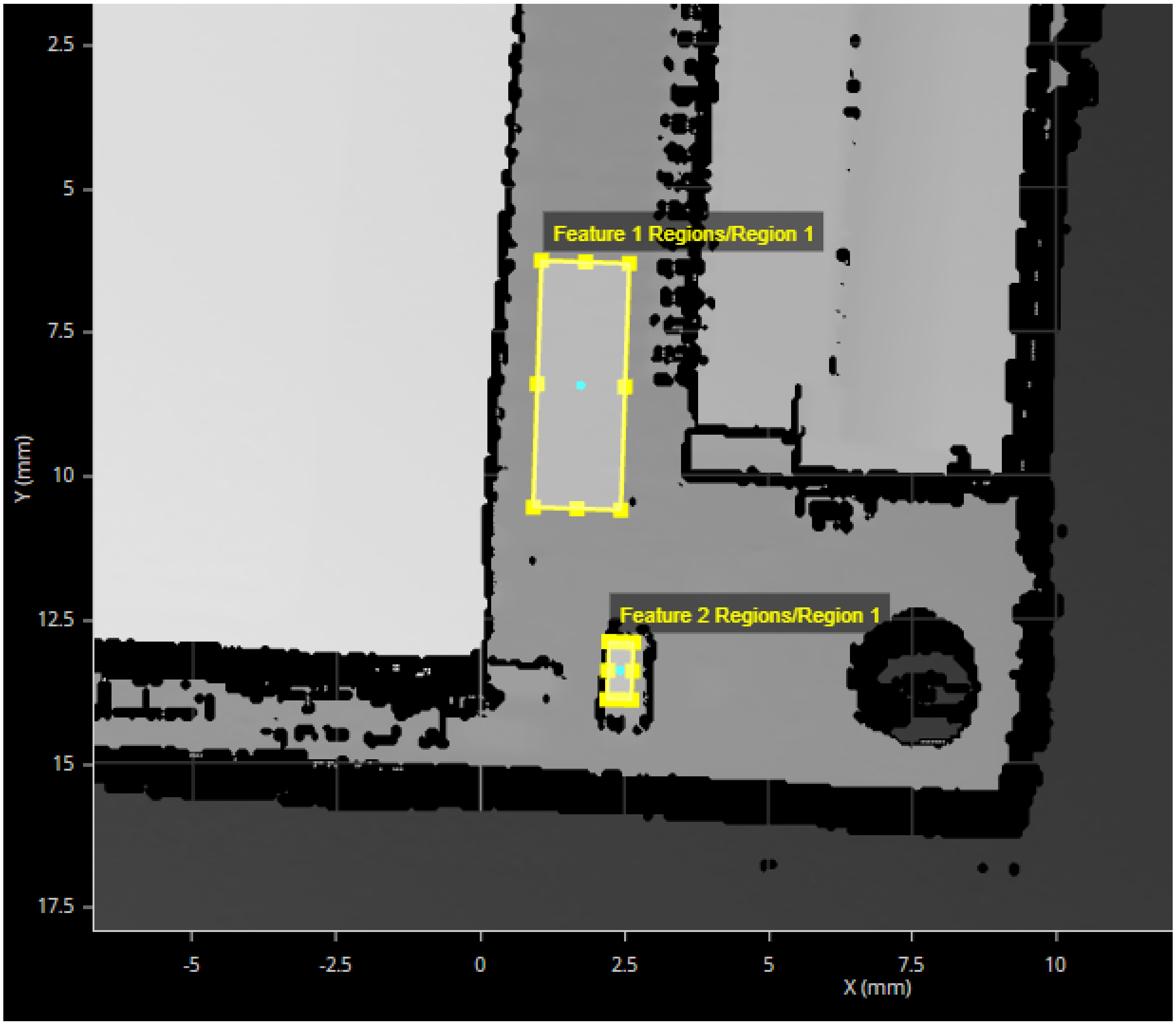
Some measurement and processing tools can output Profile or Surface data, which can be used as input by other tools or SDK applications.

Profile and Surface data produced by a tool is identical in nature to the data produced by a sensor scan, except that they are the processed result from a tool. This kind of data can be used as input in compatible tools. Examples of this kind of this kind of data are the Stitched Surface output from the Surface Stitch tool, or the Filtered Surface output from the Surface Filter tool. Another important kind of data is the Transformed Surface produced by the Surface Transform tool, which transforms (shifting or rotating on the X, Y, and Z axes) the sensor's scan data; the Surface Transform tool supports a full 6 degrees of freedom. For more information, see *Surface Transform* on page 672.

Anchoring Measurements

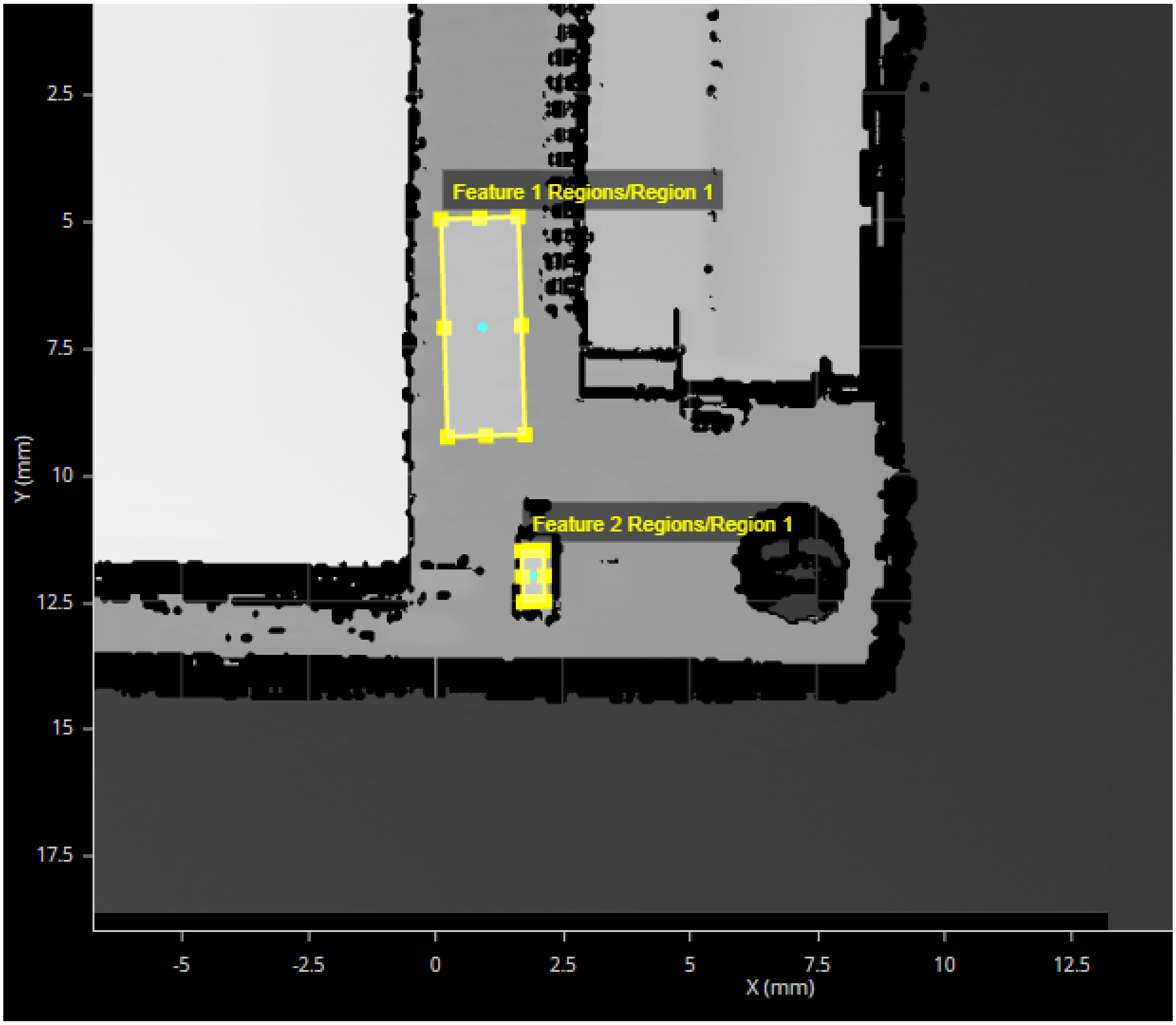
Tools can use the positional measurements (X, Y, or Z) of other tools as anchors to compensate for minor shifts of parts: anchored tools are "locked" to the positional measurements of the anchoring tool's measurements. Some tools can also use a Z Angle measurement as an anchor. Typically, you will use measurements from more easily found features on a target—such as an edge or a hole—as anchors to accurately place other positional and dimensional measurements. This can help improve repeatability and accuracy in the anchored tools. Note that anchoring measurements are used to calculate the offsets of the anchored tools: the results from these measurements are not used as part of the anchored tool's measurements.

Anchoring measurements are rendered as overlays on a tool's input data.

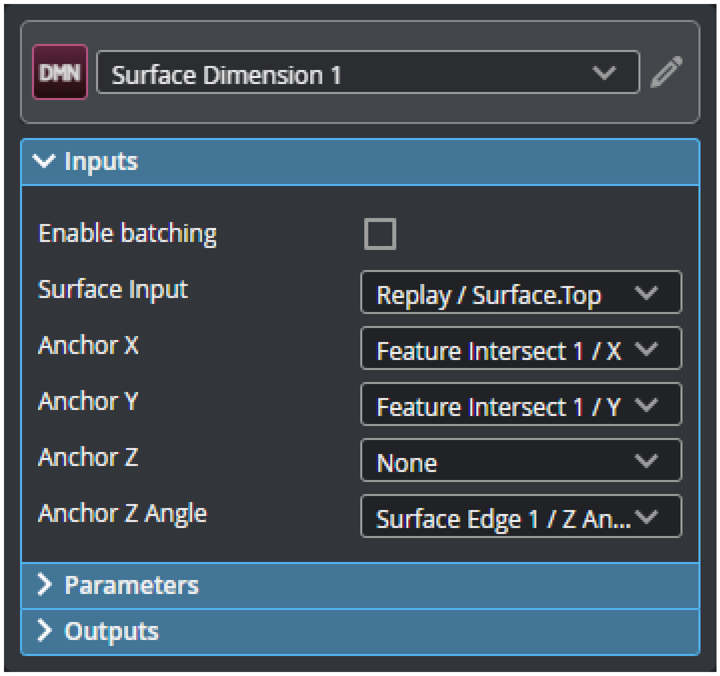


The height of a small PCB component (Feature 2) relative to nearby surface (Feature 1), anchored to positional (X and Y) measurements of the hole (lower right) and to the Z angle of the edge of a larger component to the left.

In the following, the part has rotated, but the tool's measurement regions follow the features, ensuring correct measurement.



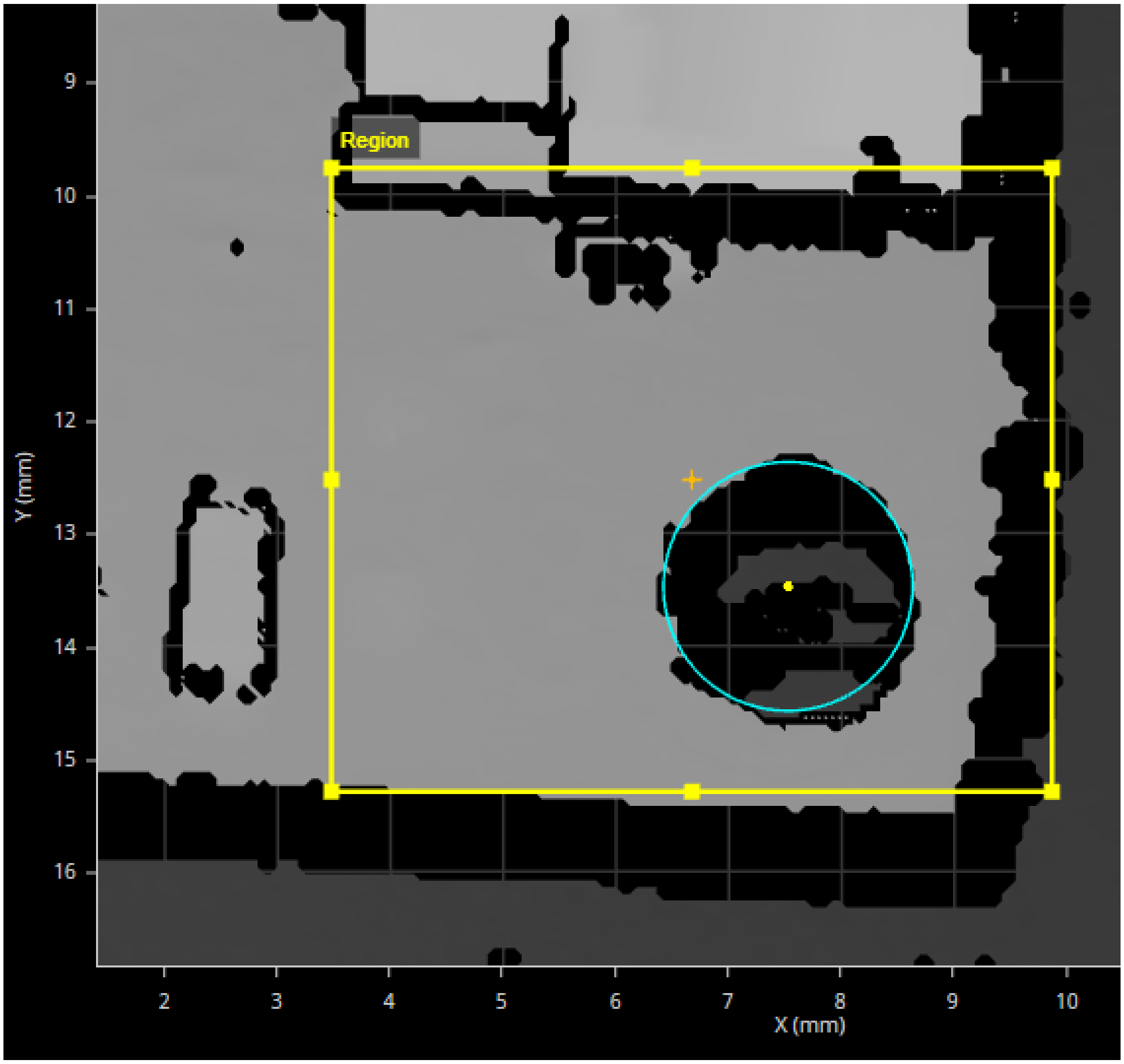
You enable anchoring in the expandable **Inputs** section of a tool:



Geometric Features

Many of GoPxL's measurement tools can output data structures such as points, lines, planes, and circles. These structures are called geometric features and contain the components you would expect: a point geometric feature contains X, Y, and Z components (representing the location of the point in 3D space). Examples of point geometric features output by Gocator's measurement tools are hole center points, the tip and base of studs, or a position on a surface.

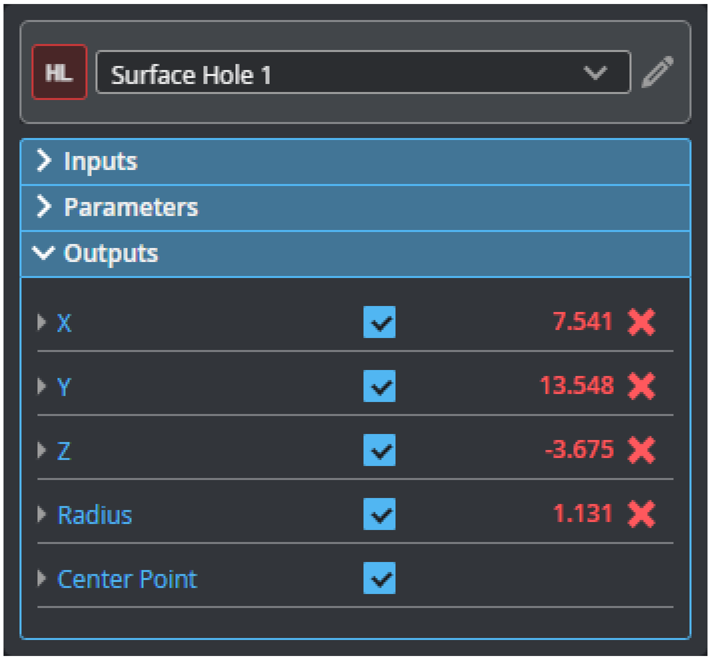
Geometric features are rendered as overlays on a tool's input data.



Point geometric feature (a hole's Center Point) rendered on a tool's input as a small yellow circle

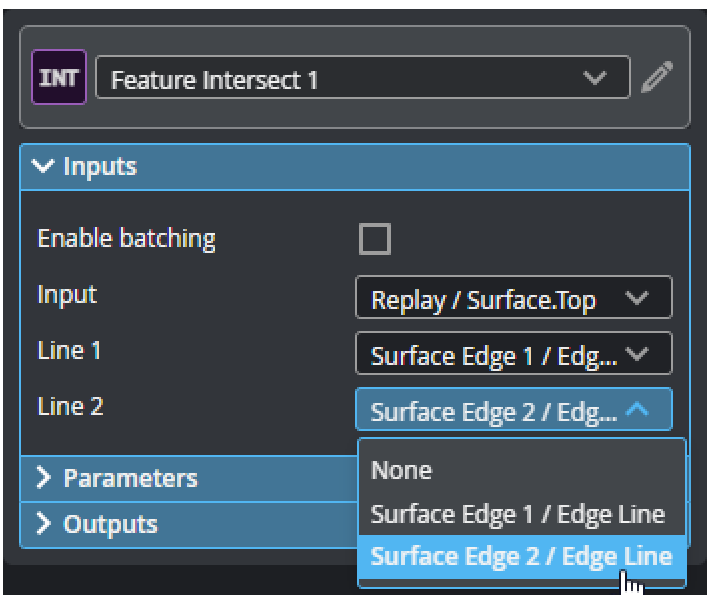
Gocator's "Feature" tools (such as Feature Dimension and Feature Intersect) use geometric features as inputs. For example, because the point geometric feature representing the center of a hole has X, Y, and Z components, you can perform dimensional measurements between it and another geometric feature, such as another hole or an edge. The Feature Create tool takes one or more geometric features as input and generates *new* geometric features (for example, creating a line from two point geometric features). You can then perform measurements on those features directly in the tool or in other Feature measurement tools. You can also use angle measurements on the newly created features for anchoring. For more information on Feature tools, see *Feature Measurement* on page 692.

You enable geometric feature *outputs* in a tool's expandable **Outputs** section:



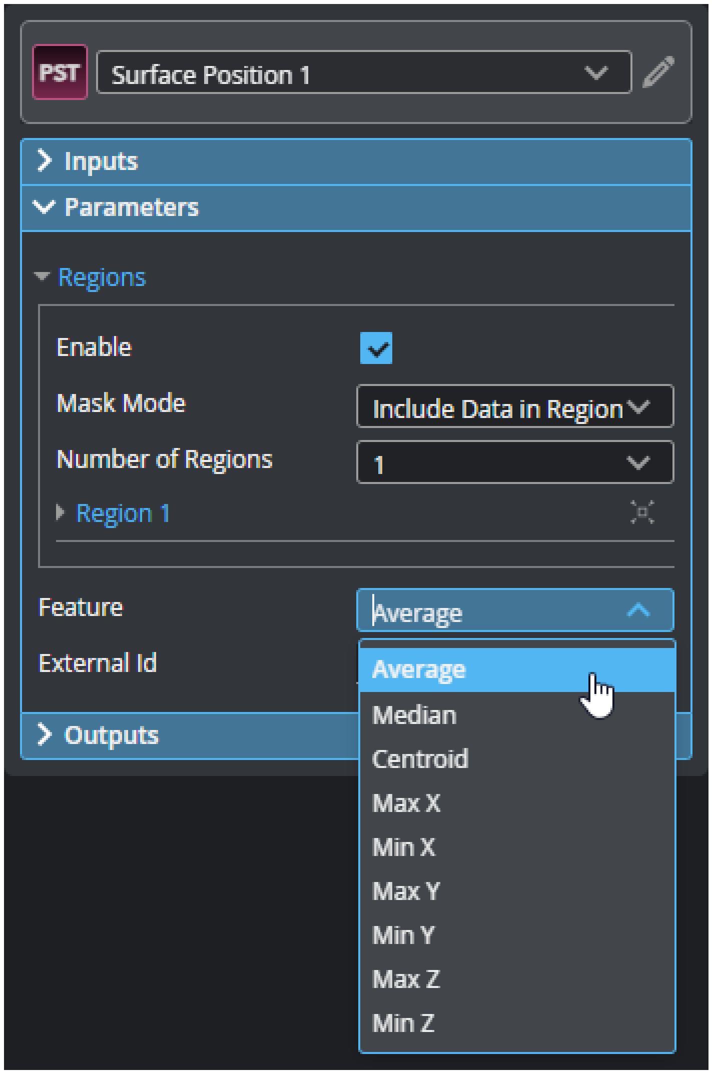
Center Point geometric feature of a Surface Hole tool enabled on Features tab

You enable geometric feature *inputs* in a tool's expandable **Inputs** section:



Setting the Line inputs to two different lines to compute their intersection

Geometric features are distinct from the "feature points" used by certain tools to determine which data point in a region should be used in a measurement, for example, the maximum versus the minimum on the Z axis of a data point in a region of interest:



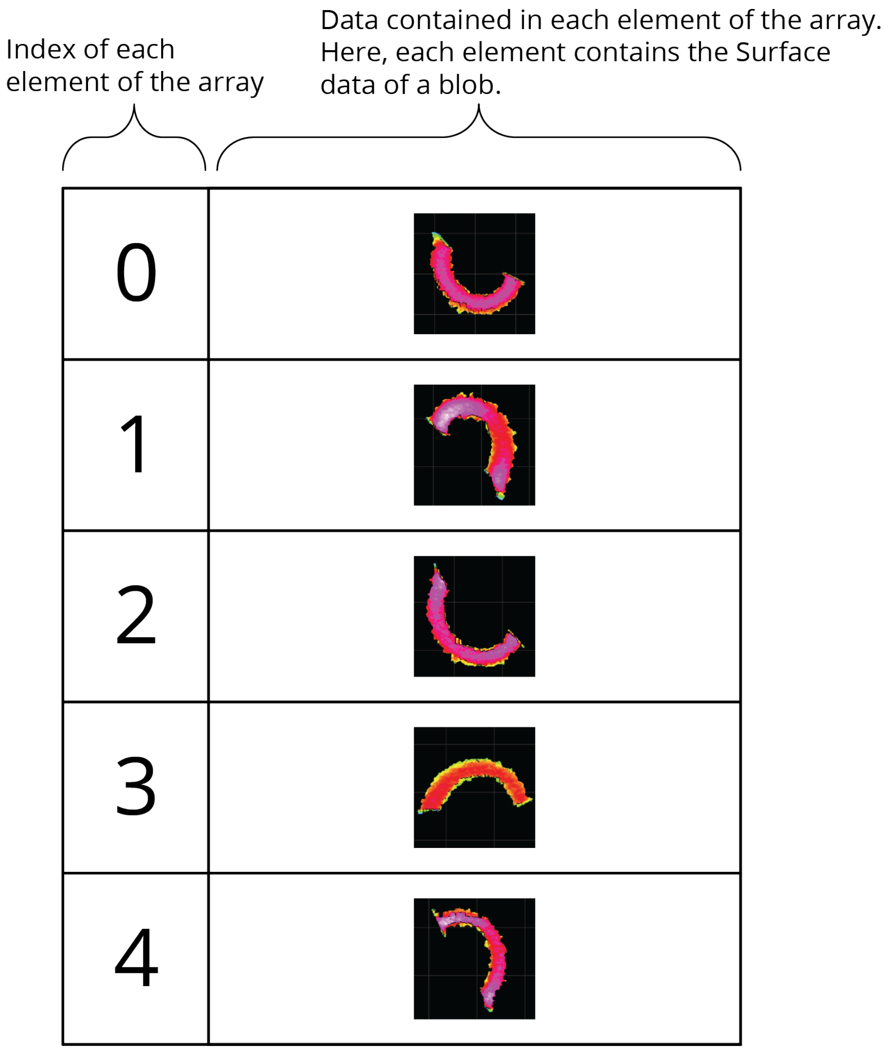
For more information on feature points, see *Feature Points* on page 259.

Arrays, Batching, and Aggregation

GoPxL supports data structures called arrays to help simplify otherwise complicated applications.

An array is a group of data items bundled into a single structure, such as multiple profiles or surfaces, multiple measurements, or multiple features.

For example, the individual parts in a frame of scan data can be isolated and stored in an array. Each part's scan data occupies on element of the array. Note that arrays start at 0. That is, they use zero-based indexing.



Each item, or element, in the array can be processed by other tools in one of two ways:

- Batching: Tools in batching mode process each array element in an array input separately.
- Aggregating: Aggregating tools combine each element in an array output and process it as a single piece of data.

For more information, see *Arrays, Batching, and Aggregation* on page 242.

Output and Control

After the sensor has scanned and measured parts, the last step in the operation flow is to output the results.

One of the main functions of GoPxL is to produce pass/fail decisions, and then control something based on that decision. Typically, this involves rejecting a part through an eject gate, but it can also involve making decisions on good, but different, parts. Currently, GoPxL can communicate using the following over Ethernet:

- With PLCs (programmable logic controllers) using industry-standard protocols (EtherNet/IP, Modbus, PROFINET, and ASCII).
- With the GoPxL SDK and REST API, using the Gocator Communication Protocol.

For more information on configuring control, see *Configuring Control* on page 793.

For more information on the GoPxL SDK and REST API, see *GoPxL SDK and REST API* on page 871.

User Interface Overview

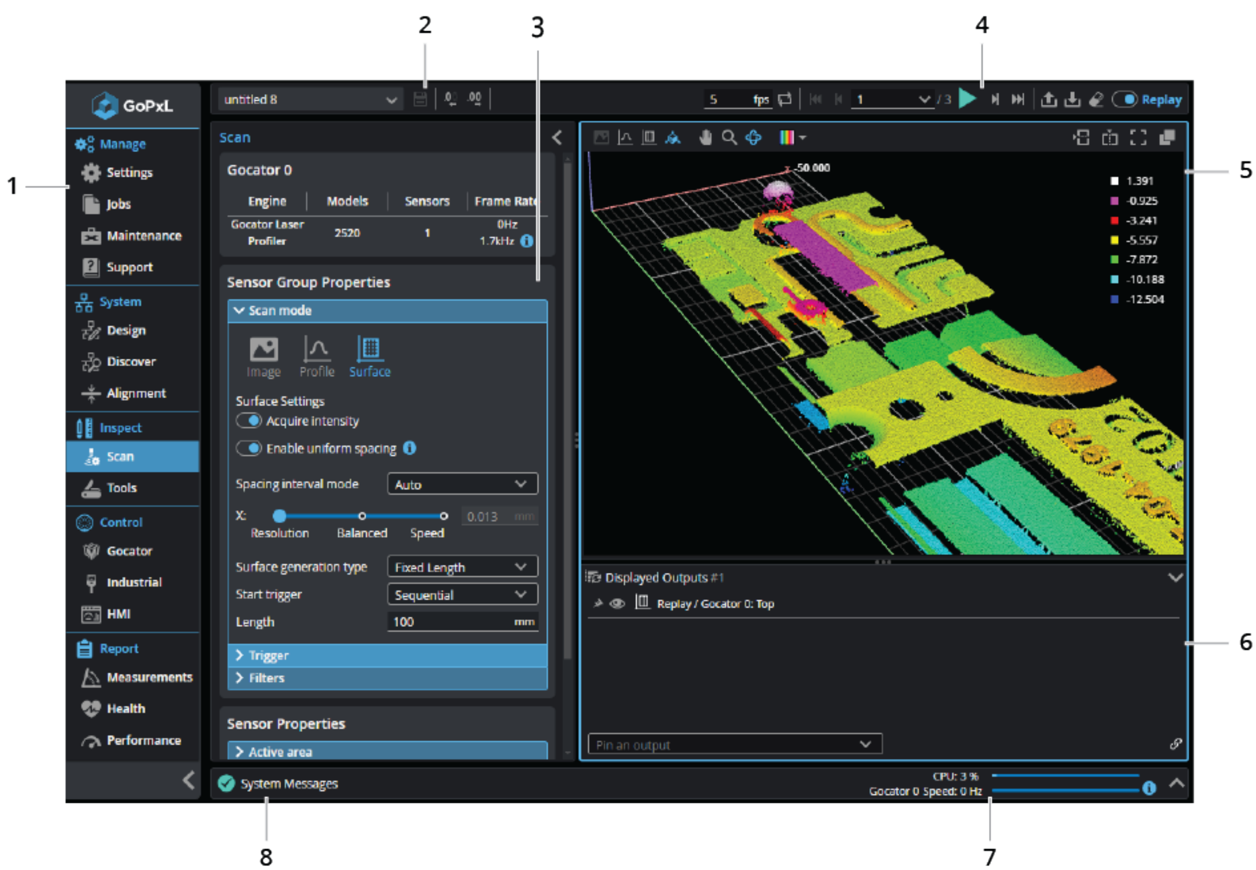
You configure sensors using GoPxL by connecting to an IP address with a web browser.

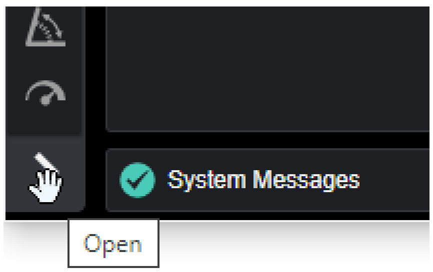
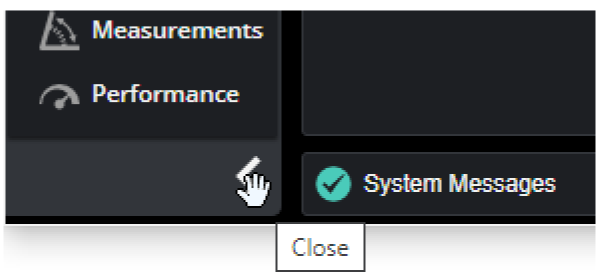
If you are running GoPxL on-sensor, in the browser, you connect to the IP address of the sensor. By default, the IP address of Gocator sensors is 192.168.1.10.

If you are running the Windows-based GoPxL application (to accelerate the sensor on a PC or view previously recorded scans), in the browser, by default you connect to the localhost address (127.0.0.1). For more information on running GoPxL on a PC, see *Running GoPxL on a Windows PC* on page 828.

GoPxL provides categories in a vertical navigation bar on the left of the interface that group similar functions together in pages. Each page contains one or more panels that further categorize functionality. Generally speaking, the order of the categories and the pages reflects the workflow involved in configuring systems for an application for the first time, from the top down.

	Some UI screenshots in this manual may not depict the latest version of GoPxL. These screenshots will be updated soon.



	Element	Description
1	Navigation bar	The navigation bar (sometimes called the "nav" bar) along the far left of the user interface displays categories (for example, from the top, **Manage**, **System**, **Inspect**, and so on), which contain one or more configuration pages (for example, the **Manage** category contains the **Settings** and **Jobs** pages, among others). Each page contains one or more panels. The category and page currently selected in the navigation bar are highlighted. In this manual, we often use a shorthand convention to describe the location of a page, such as "**Inspect** > **Tools** page." By default, the navigation bar is collapsed and only shows the category and page icons. When you move the mouse pointer over the navigation bar, it expands temporarily to show the category and page names. To pin the navigation bar open, click the Open button at the lower left of the interface:  To unpin the navigation bar, click the Close button at the lower left of the interface.  For information on the categories and what you can do with the pages they contain, see *What you can do in the categories* on the next page.
2	Job-related functions	These controls let you work with jobs, for example, creating and saving jobs. For more information on jobs, see *Jobs* on page 119.
3	Configuration area	This area contains the configurations of the selected page and panel. When configuring tools, an additional **Tools Configuration** panel lets you configure a tool's parameters.
4	Scan controls	These controls let you start and stop scanning, and also work with replay data. For information on starting and stopping scanning, see *Starting, Stopping, and Recording* on page 113. For information on working with replay data, see *Playing Back Recorded Data* on page 113.
5	Data viewer	The data viewer (sometimes called the data visualizer) displays scan data,

	Element	Description
		providing several tools in its toolbar to control how data is visualized (such as top view, front view, or perspective view). The data viewer is available on all pages. You can split it horizontally and vertically, and create pop-out windows. When GoPxL is in Replay mode, a blue border is displayed around the data viewer. For more information, see *Data Viewer* on page 88.
6	Displayed outputs pane	This area lets you show, hide, and pin outputs such as measurements. You can also control how outputs are displayed. For more information, see *Pinning Outputs* on page 107.
7	Metrics area	This area provides CPU usage and frame rate, including the calculated maximum frame rate. For more information, see *Metrics Area* on the next page.
8	System messages	This area provides log messages, error alerts and warnings. For more information, see *System Messages* on the next page.

The following lists the categories available in GoPxL and where you can learn how to work with them.

What you can do in the categories

Category	Description
Manage	Lets you configure network settings, manage jobs, upgrade sensors or perform a factory restore, and so on. For more information, see *Sensor Management and Maintenance* on page 117.
System	Lets you design and align sensor systems. For more information system design, see *Creating a Sensor System* on page 124.
Inspect	Lets you do the following: • Configure a sensor's data acquisition. For more information, see *Scan - Configuring Acquisition* on page 195. • Add and configure measurement and processing tools. For more information on tools, see *Tools - Measurement and Processing* on page 232.
Control	Lets you enable and configure the Gocator communication protocol and industrial protocols. Also lets you configure GoHMI. For more information, see *Configuring Control* on page 793. For information on the protocols, see *Integrations* on page 884.
Report	Lets you monitor measurements, as well as sensor health and performance. For more information, see *Reporting* on page 821.

Status Bar

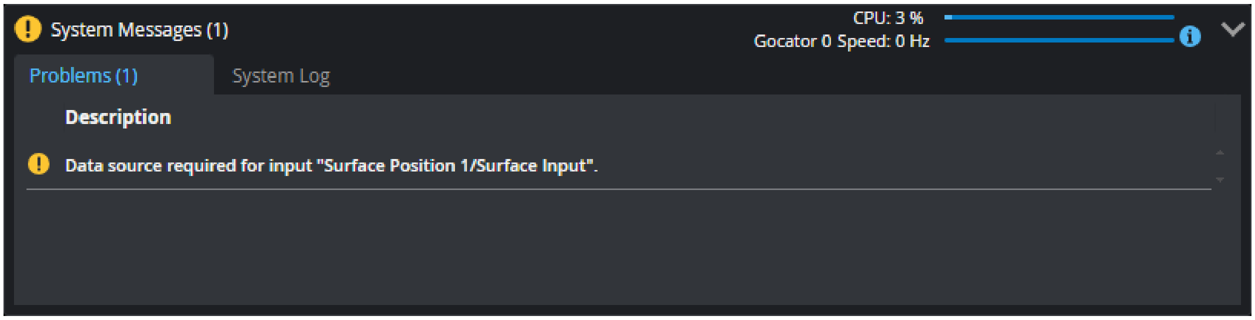
The status bar lets you do the following:

- Access system messages in the in the **System Messages** panel. For details, see *System Messages* on the next page.
- See important system metrics. For details, see *Metrics Area* on the next page.

System Messages

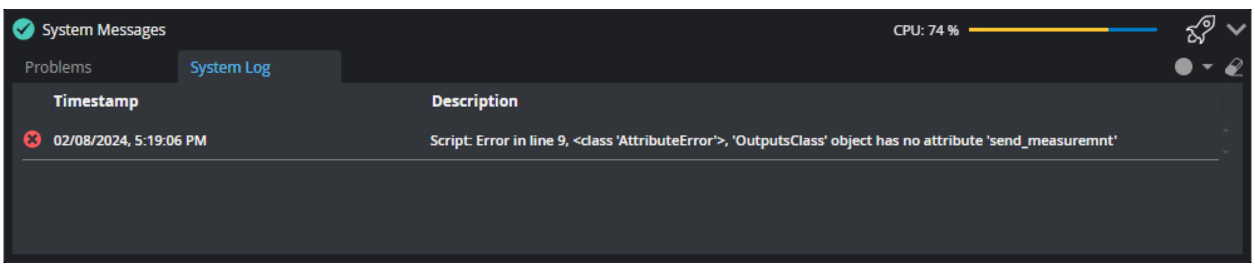
The **System Messages** panel displays current problems and a system log containing errors, warnings, and general information, in two tabs.

GoPxL displays a warning symbol ( ) when there are items in the **Problems** tab. The **Problems** tab lists the *current* problems. Once you fix a problem, it is removed from the **Problems** tab.



Problems tab showing a current problem with the system. The number of problems is indicated in parentheses.

The **System Log** tab displays a complete list of system-level messages. You can filter the log entry types using the filter drop-down to the right of the **System Log** tab, and clear the entries using the eraser button.



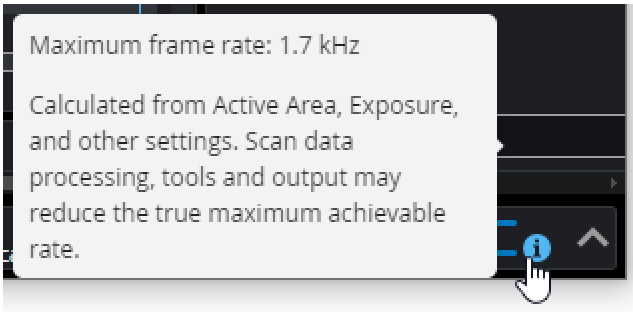
Metrics Area

The Metrics area on the right side of the status bar displays two important performance metrics: CPU load and sensor speed (current frame rate).

The **CPU** bar in the Metrics area displays how much of the CPU is being used.

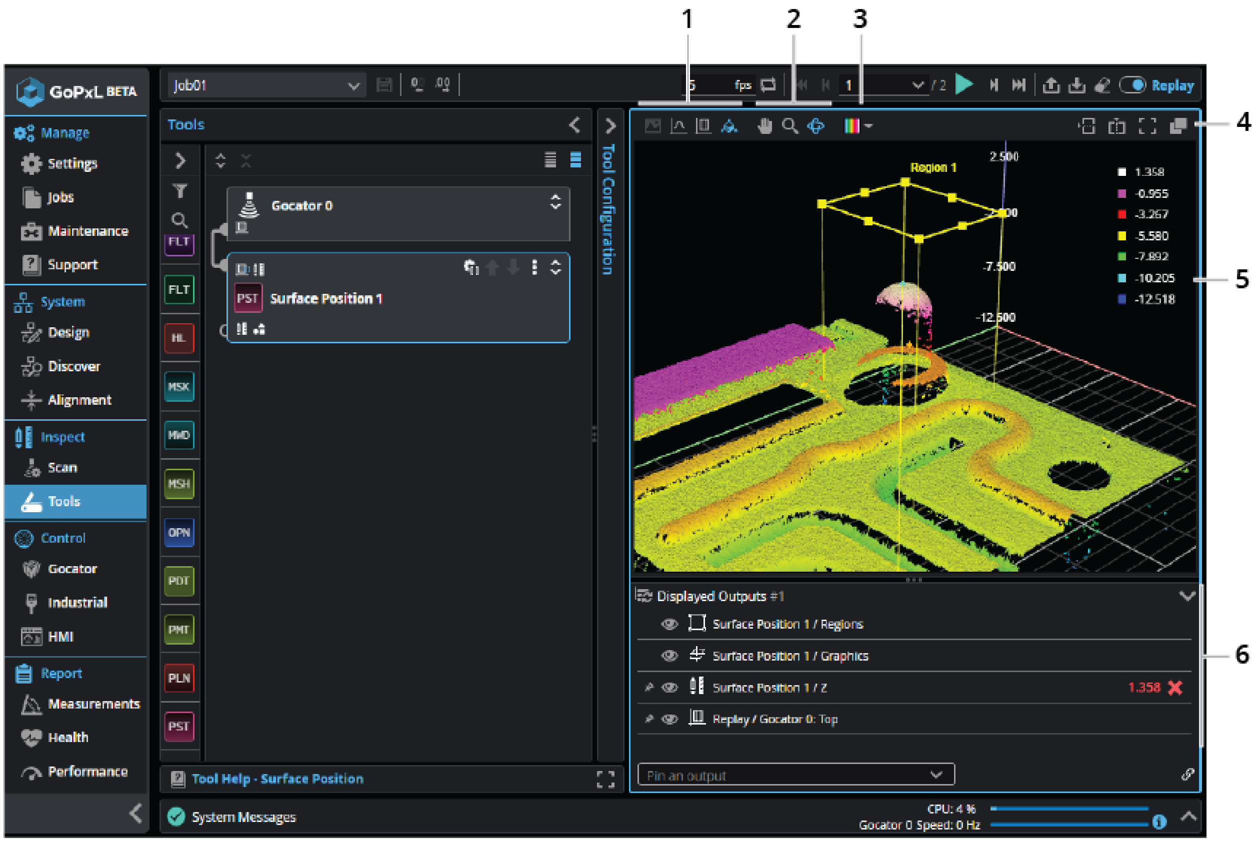
The **Speed** bar displays the frame rate of the sensor.

You can see the maximum calculated frame rate by clicking the info icon to the right of the metrics area.


Maximum frame rate: 1.7 kHz
Calculated from Active Area, Exposure, and other settings. Scan data processing, tools and output may reduce the true maximum achievable rate.

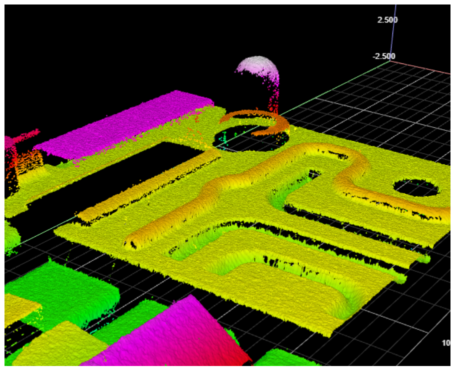
Data Viewer

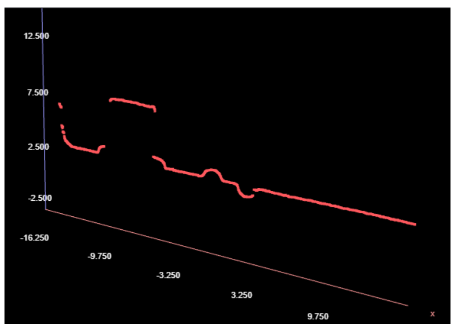
You use a data viewer to observe scan data (such as Profile and Surface data), as well as video images and intensity images. You can also use it to configure measurement tools (see *Tools - Measurement and Processing* on page 232) and a sensor's active area (see *Active Area* on page 215). GoPxL supports multiple data viewers (via both splitting and pop-out buttons).



	Element	Description
1	Display modes	Use these to change the data viewer's display mode. One of the following: **Image** Image mode () lets you view pixel data that the system has acquired in Image mode. You should use Image mode to adjust exposure as an initial step and for diagnostics. For more information, see *Image Mode* on page 92. **Perspective**

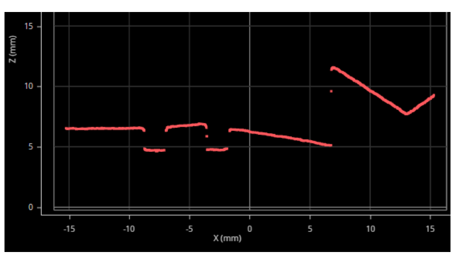
Element	Description
	Perspective mode ( ) lets you view data in a 3D perspective representation. You can move and turn the data using a mouse and the pan, orbit, and zoom buttons ( ). When in orbit mode, hold Shift to temporarily switch to pan mode. The data viewer can display both Surface and Profile data in Perspective display mode. For information on controlling how the data viewer displays Surface data in Perspective mode, see *Surface and Perspective Display Options* on page 100.





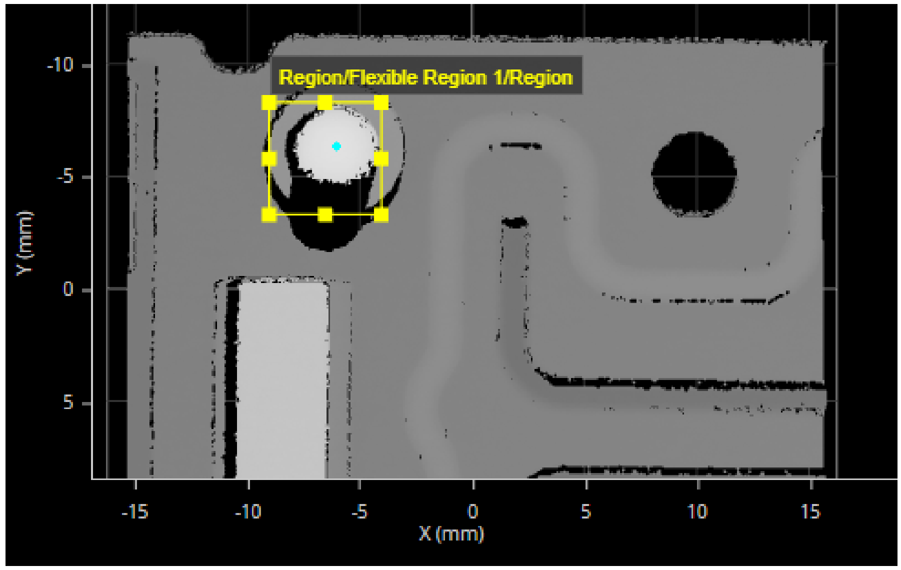
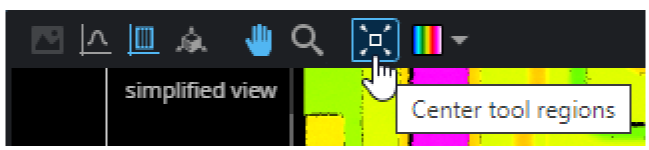
Profile / Front

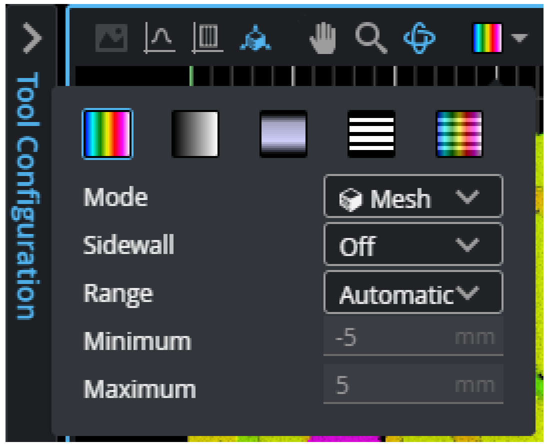
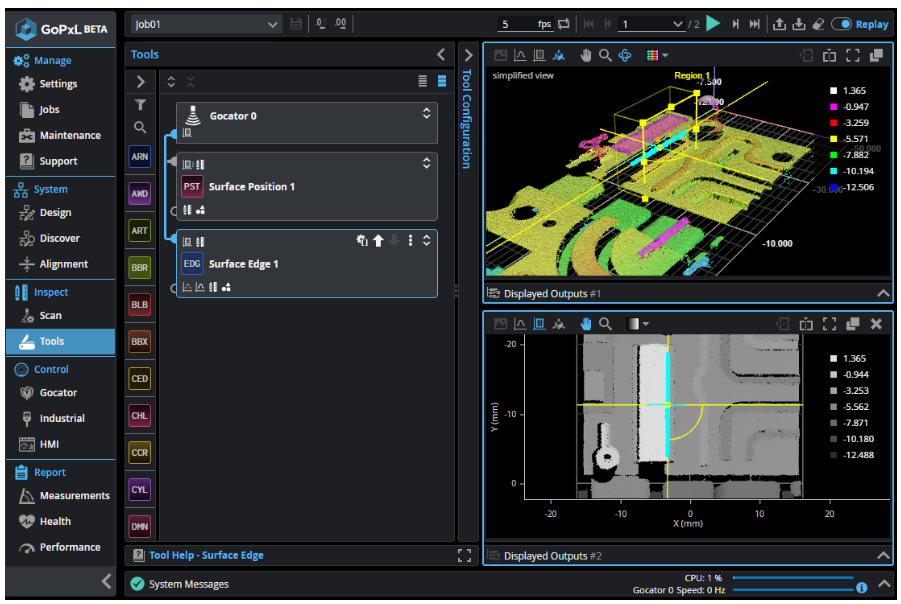
Profile / Front mode ( ) displays profiles in a 2D representation when Profile data is available.



Surface / Top

Surface / Top mode ( ) lets you display surface data in a 2D representation when Surface data is available. For information on controlling how the data viewer displays Surface data in Perspective mode, see *Surface and Perspective*

	Element	Description
		Display Options on page 100.  Note that In Surface / Top display mode (and in Profile / Front mode), a center tool region button is available when a tool is selected in the Tools Diagram panel. Clicking this centers any regions in the middle of the data viewer, resizing them to fit. This can be useful when you have zoomed the data viewer in a lot, and need to quickly bring the regions into view. 
2	Pan, zoom, and orbit	Use these buttons () to move scan data in the data viewer with your mouse. (Orbit is only available in the perspective view mode; see above.) Various shortcuts are available to make working with the data viewer easier. For more information, see *Keyboard Shortcuts* on page 92.
3	Surface display options	Lets you control how scan data is displayed, such as whether Surface scan data is displayed using a color heightmap or a grayscale heightmap overlay. For information on changing how the data viewer displays Surface data, see *Surface and Perspective Display Options* on page 100. For information on how the data viewer displays Profile data, see *Profile Mode* on page 98.

<table>
<tr><th></th><th>Element</th><th>Description</th></tr>
<tr><td></td><td></td><td>
</td></tr>
<tr><td>4</td><td>Split, zoom, and pop-out</td><td> Use the split buttons (and) to create additional data viewers. You can create as many as four data viewers in the main browser window. To close a split window, click its Close icon ().


Use the Pop-out button () to create a new window with a single data viewer. For more information, see Using Multiple Data Viewer Windows on page 106.
Use the Zoom to fit button () to zoom the scan data so that it fits in the data viewer window. </td></tr>
<tr><td>5</td><td>Data viewer window</td><td> This is the main window in the GoPxL interface that shows scan data. When you configure tools or set the active area, you can graphically configure the regions in the data viewer window.
When the view is zoomed out, to reduce processing, the number of data points that are displayed is reduced, as indicated by "simplified view" in the data viewer. When you zoom in enough, the actual data points are all displayed. </td></tr>
</table>

	Element	Description
6	Displayed Outputs pane	Lets you pin and hide outputs. For more information, see *Pinning Outputs* on page 107. Use the link icon ( ) in the lower right of the Displayed Outputs pane to copy the path to the data viewer for use in GoHMI Designer to display a data viewer in your HMI. For more information, see *Using a Data Viewer in an HMI* on page 865.

Keyboard Shortcuts

GoPxL supports the following shortcuts:

Name	Keyboard shortcut	Description
Force pan mode	Shift + Mouse drag	When the data viewer is in orbit or zoom mode, holding Shift and dragging scan data in the data viewer with the mouse switches the data viewer to pan mode. This shortcut is only active when the mouse pointer is over the data viewer window.
Ignore regions	Alt + Mouse drag	Causes the mouse pointer to ignore regions so that you can pan or orbit over a region. This shortcut is only active when the mouse pointer is over the data viewer window.
Toggle demo mode	Alt + Shift + D	Toggles demo mode (automatic camera orbiting) on or off when the data viewer is in Perspective mode.
"Normal" orbit rotation for demo mode	Alt + Shift + N	Sets the demo rotation mode to "normal." If the rotation suddenly moves quickly, press this key combination.
Increase demo speed	Alt + Shift + "+" or Alt + Shift + "="	Increases the demo speed.
Decrease demo speed	Alt + Shift + "-" or Alt + Shift + "_"	Decreases the demo speed.
Undo region modification	Ctrl + Z	Undoes changes to the last modified region (size or location).
Redo region modification	Ctrl + Shift + Z	Redoes a change to the last modified region (size or location).

Image Mode

In Image mode, the data viewer displays images directly from a sensor's camera or cameras, as well as other information.

In this mode, depending on your sensor model, you can use the data viewer to display profile exposure information (see *Exposure Information* on page 96) and spot and dropout information (see

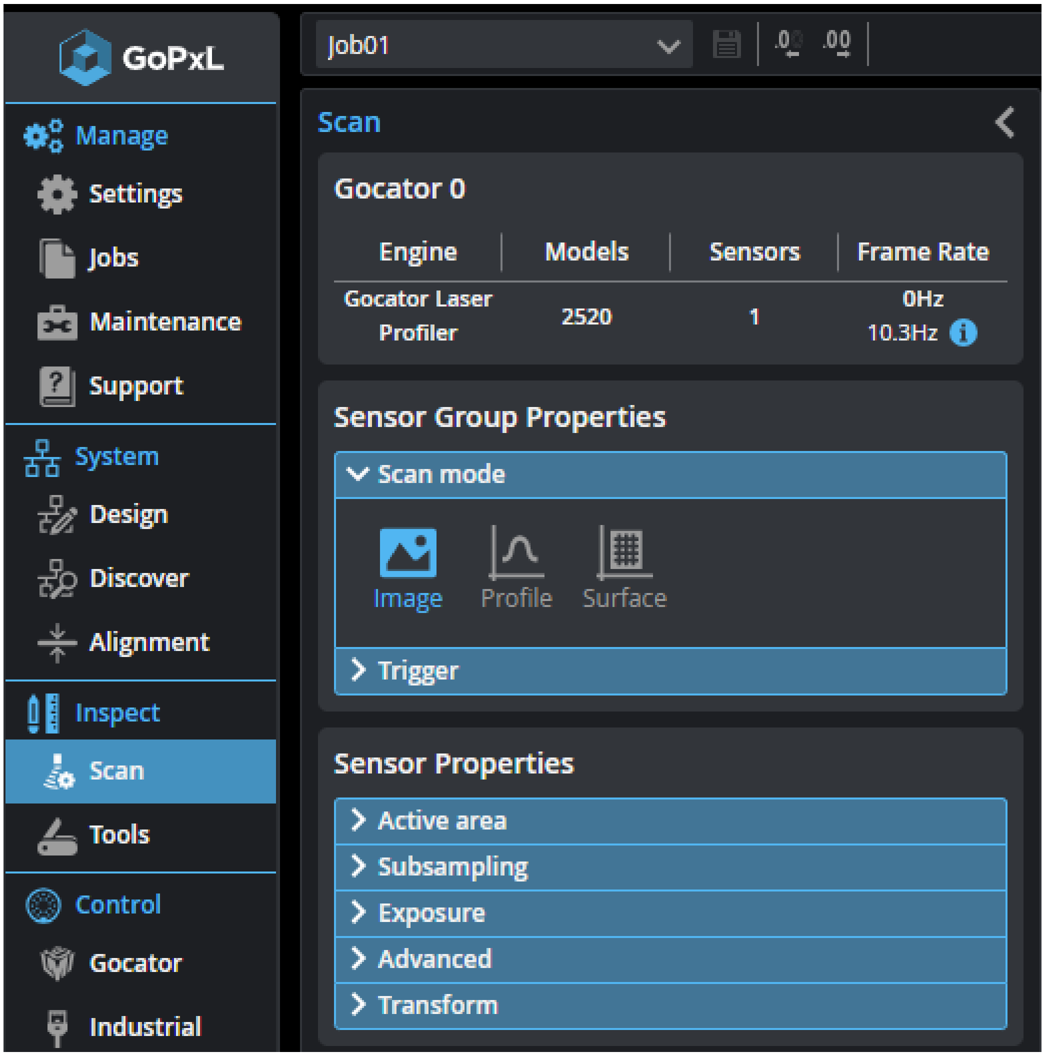
Spots and Dropouts on page 97) that can be useful in properly setting system exposure for scanning, and for troubleshooting stray light or ambient light problems.

Gocator sensors can't generate 3D points in over-saturated areas (indicated with red) or in under-exposed areas (indicated with blue). If it's not possible to set a single exposure to capture the entire object target with minimal red or blue areas appearing in Image mode, you can try enabling the **Multiple** exposure feature; the sensor will then combine the exposures to get the best scan data possible. Choose the exposure you want to examine in the Displayed Outputs panel to view each exposure and tune one high exposure for dark areas on the target and one low exposure for bright areas on the target. Note that multiple exposures reduce the maximum speed the sensor can run at.

	Not all sensor models indicate over-exposed areas (red).

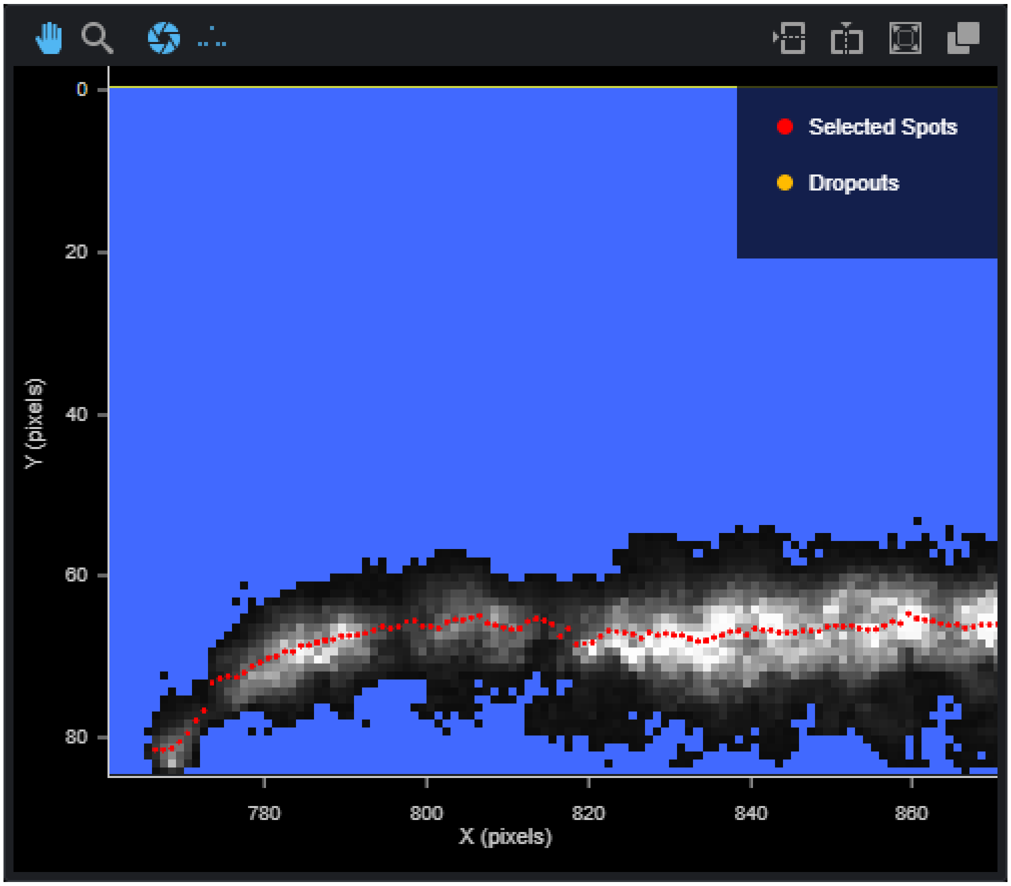
The correct tuning of exposure depends on the reflective properties of the target material and on the requirements of the application. You should carefully evaluate the exposure settings for each application.

To view Image data, you must set the system to Image mode before scanning, in the **Acquire** > **Scan** page > **Scan Mode** section.

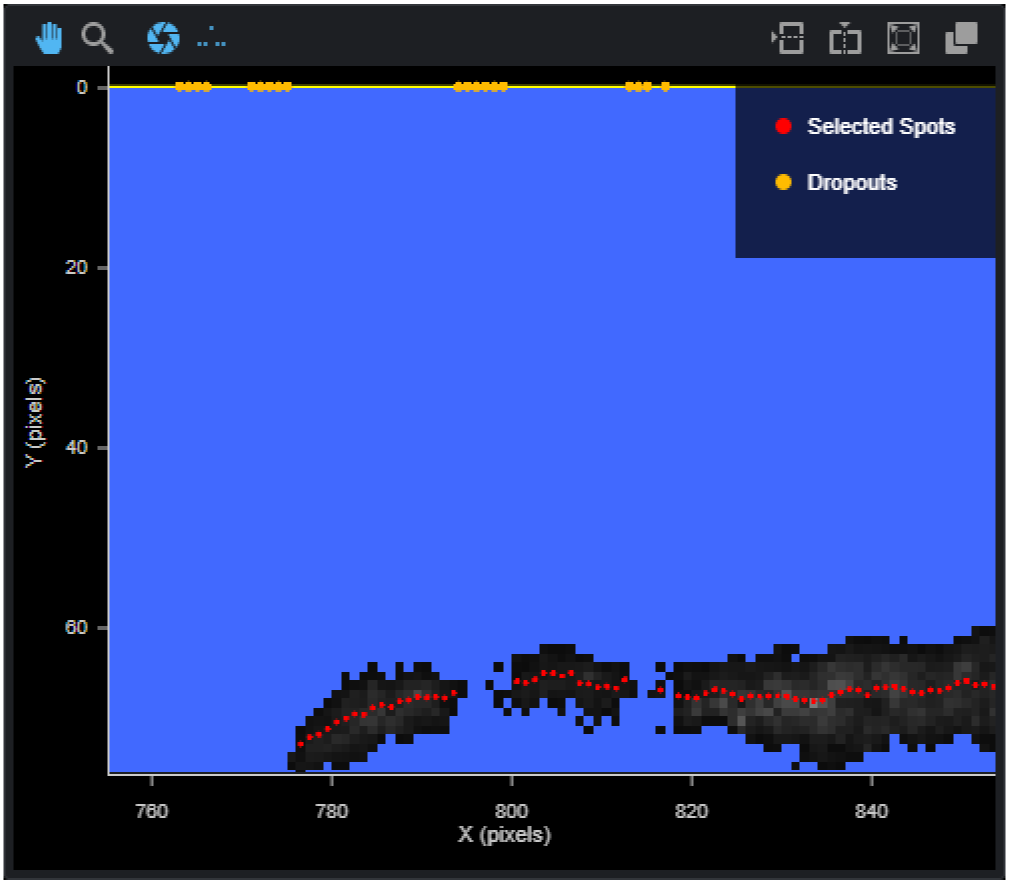


In the following:

- Red dots are the "spots" found by the sensor, which are used to create the data points in scan data.
- The grayscale squares represent the intensity value of a camera pixel.
- Blue squares (pixels) are underexposed. On some sensors, red squares represent pixels that are overexposed.



Yellow dots along the top represent dropouts. For more information on spots and dropouts, see *Spots and Dropouts* on page 97.



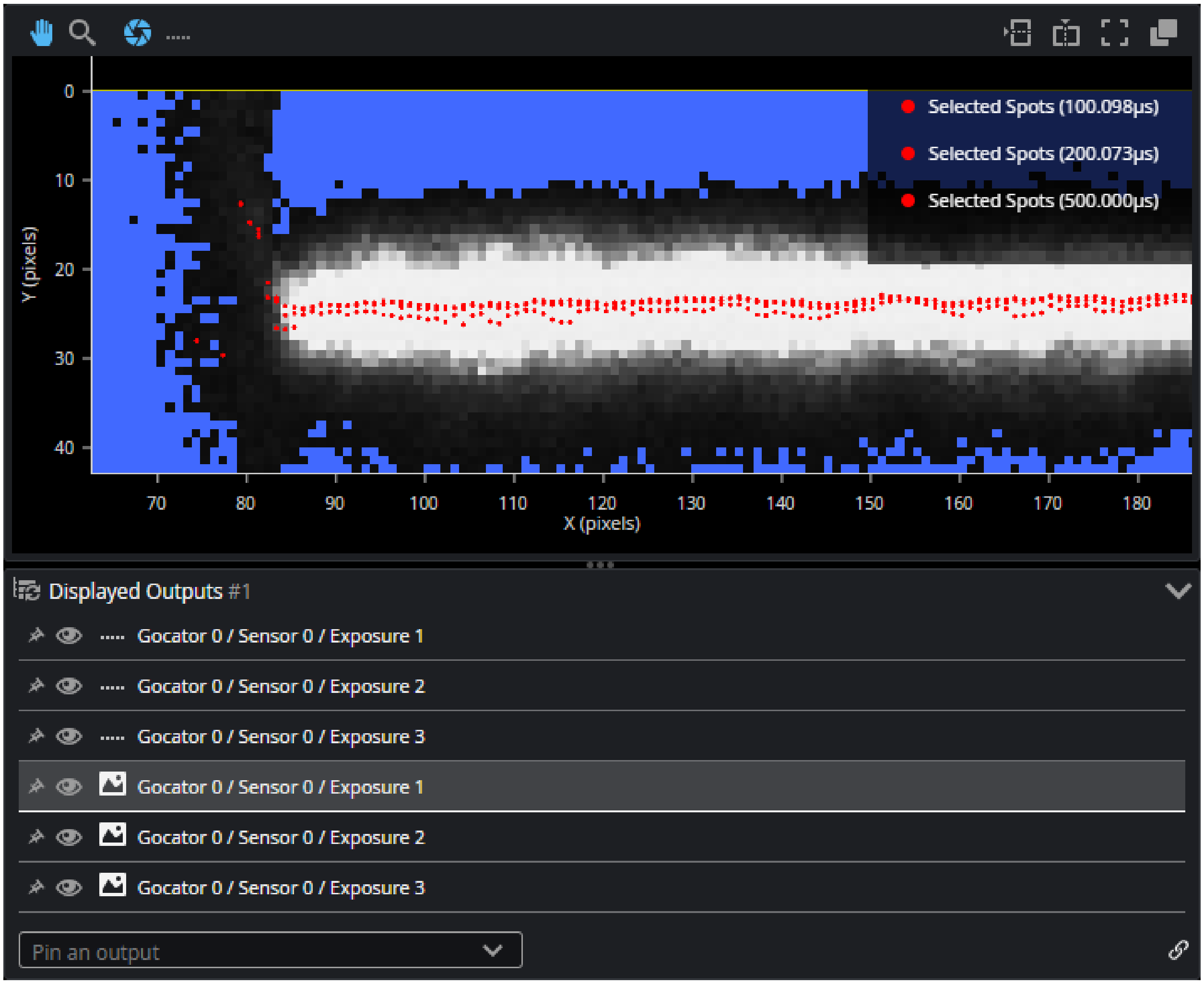
Exposure Information

In Image mode, you can display exposure-related information. This information can help you correctly adjust the exposure. (For information on setting exposure, see *Exposure* on page 219.)

To display exposure information in the data viewer, click the Exposure button () at the upper left of the data viewer.

Exposure information is listed in the **Displayed Outputs** panel at the bottom of the data viewer. If you have set the sensor's exposure to **Multiple** on the **Acquire** > **Scan**page, and have set more than one exposure, each exposure is listed individually (G2) or as an element available in a drop-down (other sensors). Exposure information is displayed in Image () outputs.

To display this information, the output must be set to visible. That is, the "eye" icon must be enabled () instead of disabled (). Note that if you have multiple exposures, only the "top" one is displayed in the data viewer. For example, to see the exposure information for Exposure 3, below, you must hide the Image outputs of Exposure 1 and 2.

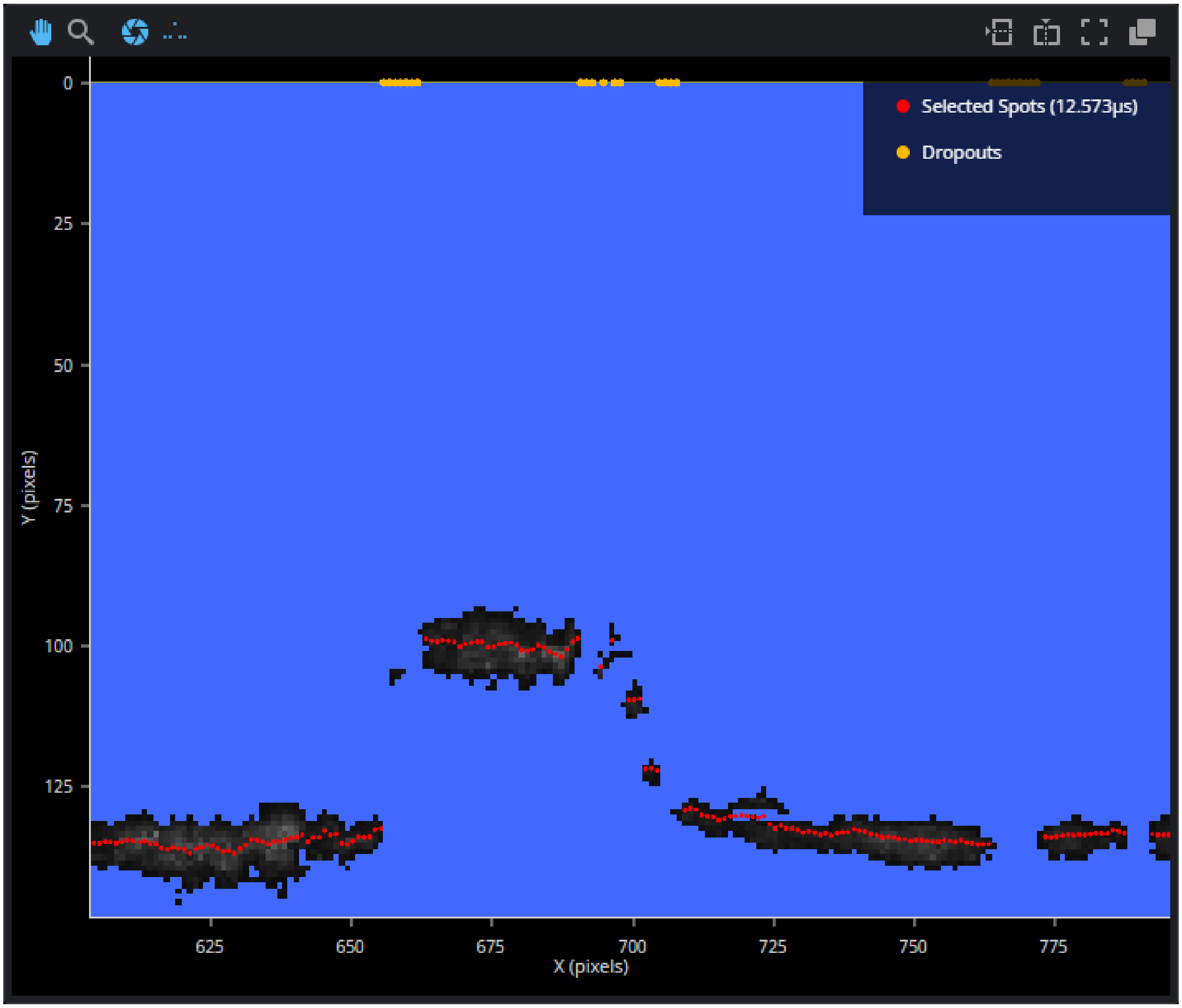


Spots and Dropouts

Various settings related to material types (under **Scan** > **Advanced**, with **Material type** set to Custom) and exposure can interact. In Image mode, you can examine how these settings affect. To do this, in Image mode, check the **Show Spots** option at the top of the data viewer to overlay a representation of the spots in the data viewer.

For more information on the material settings, see *Advanced Settings* on page 224.

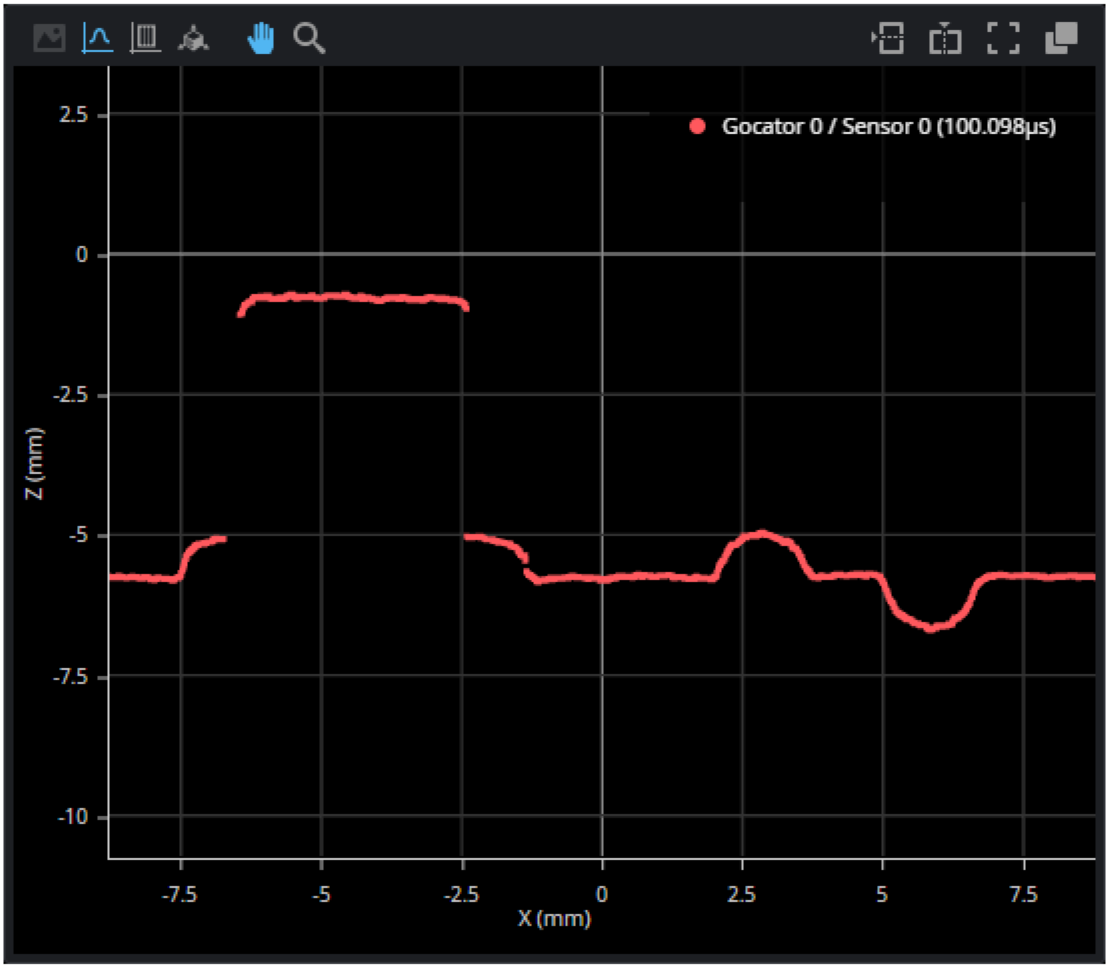
In the image below, the white and gray squares represent the light as it appears on the camera sensor. Spots that GoPxL selects (which roughly represent the center of the line of light on the camera sensor for each column) are displayed as red dots. GoPxL uses these spots to determine where data points in the scan data a sensor produces will be.



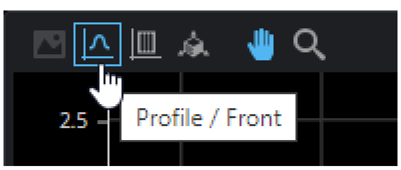
Dropouts (where no spot is detected on the camera sensor in a given column) are depicted at the upper edge of the data viewer as yellow dots.

Profile Mode

When Profile data is available, either because GoPxL is in Profile scan mode or because a Surface Section tool is providing Profile data, the data viewer can displays profile plots.



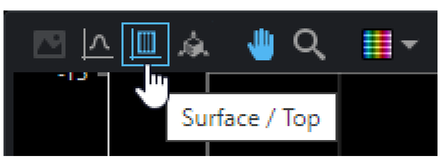
To set the data viewer to Profile mode, click the Profile mode button above the data viewer window:



Surface Mode

When the data viewer is in Surface / Top mode and Surface data is available, the data viewer can display Surface data.

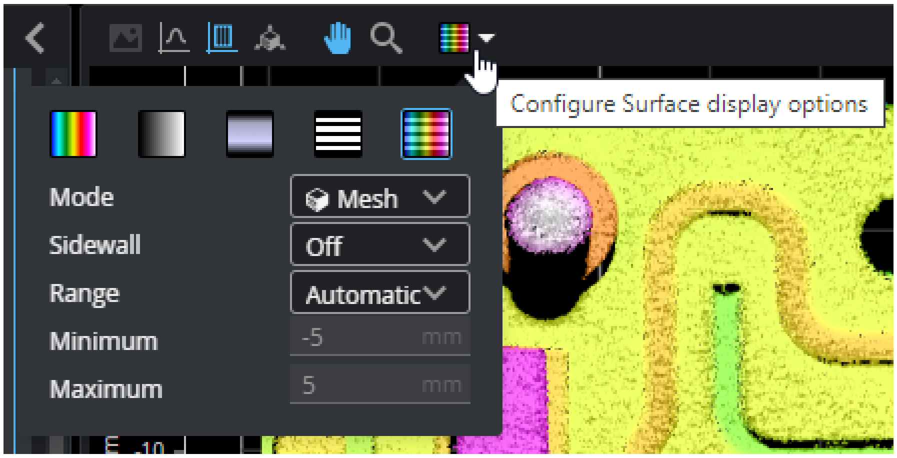
To set the data viewer to Surface / Top mode, click the mode button above the data viewer window:



GoPxL provides several ways of controlling how Surface data is displayed. For more information, see *Surface and Perspective Display Options* below.

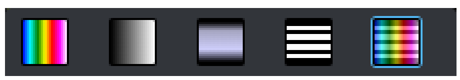
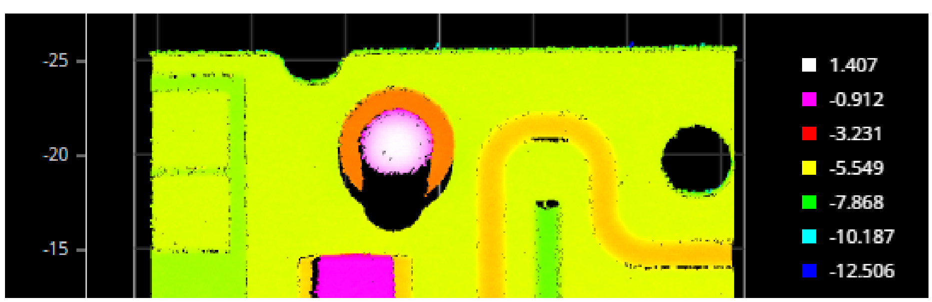
Surface and Perspective Display Options

You control how Surface data displays when the data viewer is in Surface or Perspective mode using the display options available in the Surface display options drop-down at the top-left of the data viewer.



Surface display options.

Display options

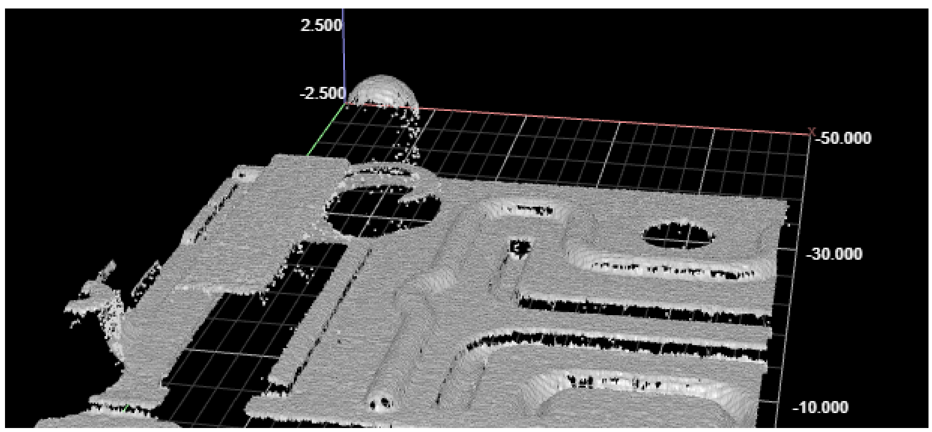
Option	Description
Display options	The output display modes let you choose how GoPxL shades the Surface data when the data viewer is in Surface mode or in Perspective mode.  For Heightmap, Grayscale, and Heightmap + Intensity settings, you can set the range values that map to the heightmap colors (using the **Range** drop-down and the related settings) to make features easier to see. The following output display modes are available: **Heightmap** Heightmap mode () displays a pseudo-color heightmap over the Surface data. A heightmap legend is displayed in the data viewer.  **Grayscale**

Option	Description

Grayscale mode () displays a grayscale heightmap over the Surface data. Working in grayscale mode can make it easier to distinguish the scan data from tool visualizations. A heightmap legend is displayed in the data viewer.

Uniform

Uniform mode () displays a uniform gray color over the Surface data. This mode is mostly useful when you want to focus on shape or geometry.

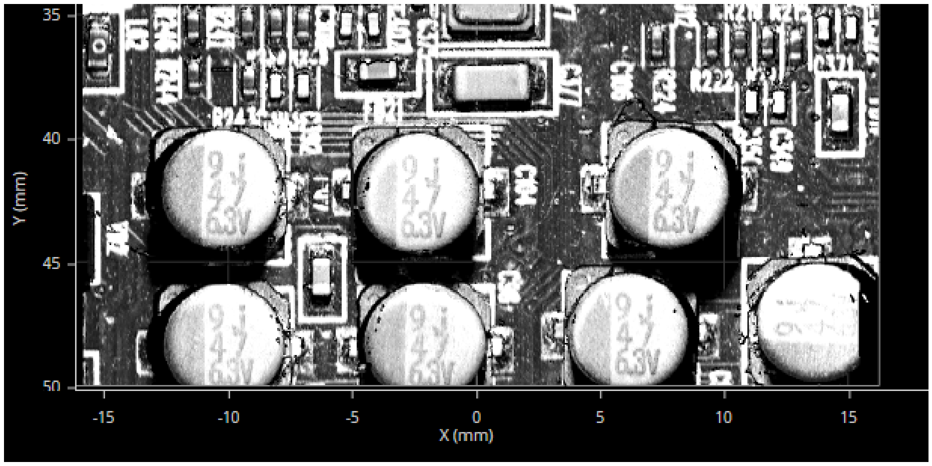
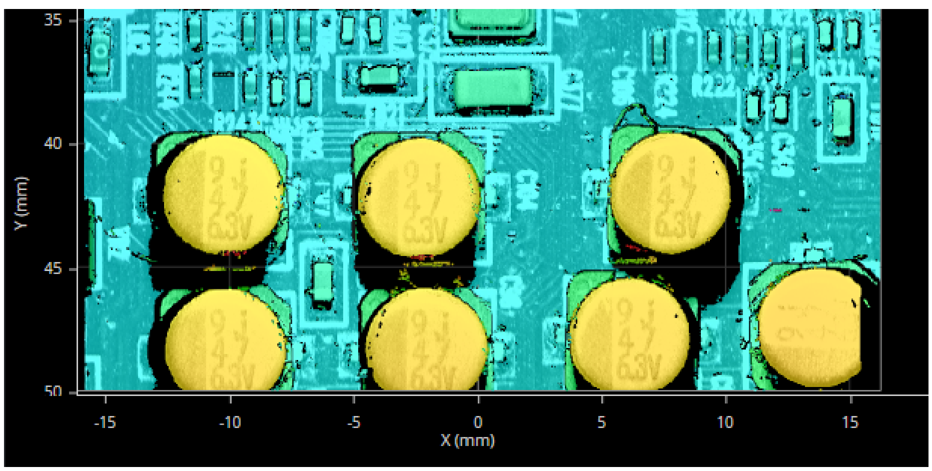


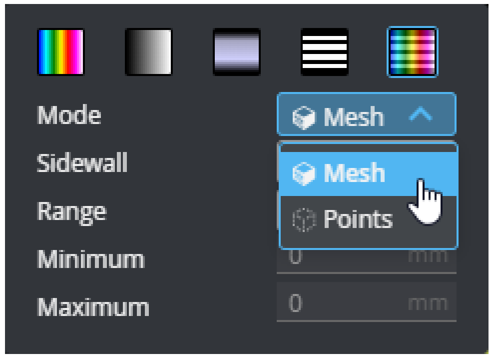
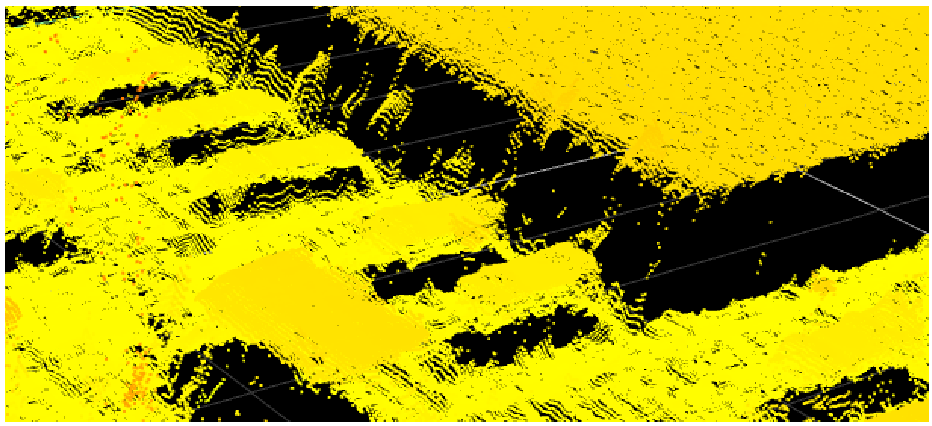
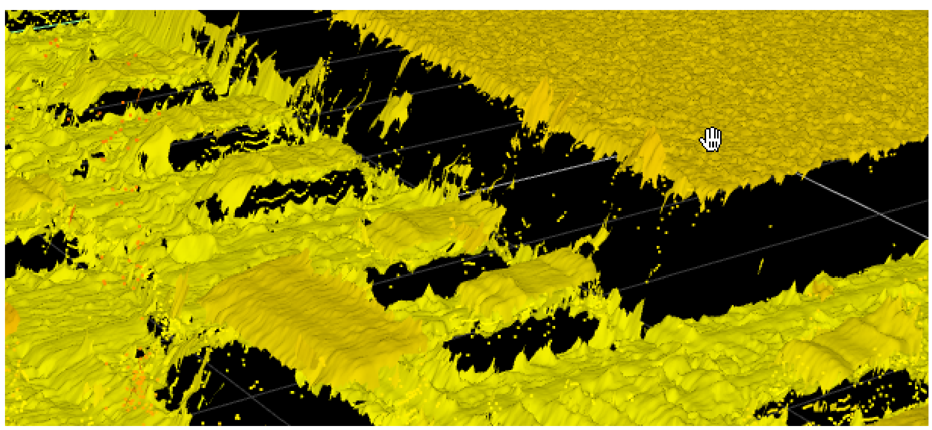
You will typically need to set **Mode** to Mesh in the Surface display options drop-down to use this display option.

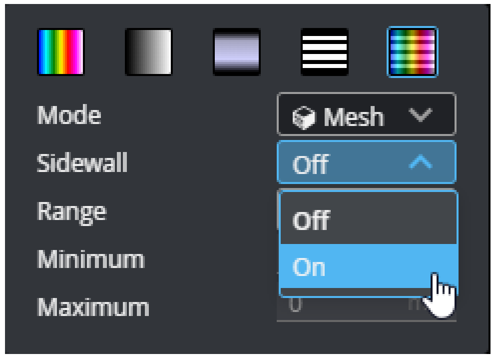
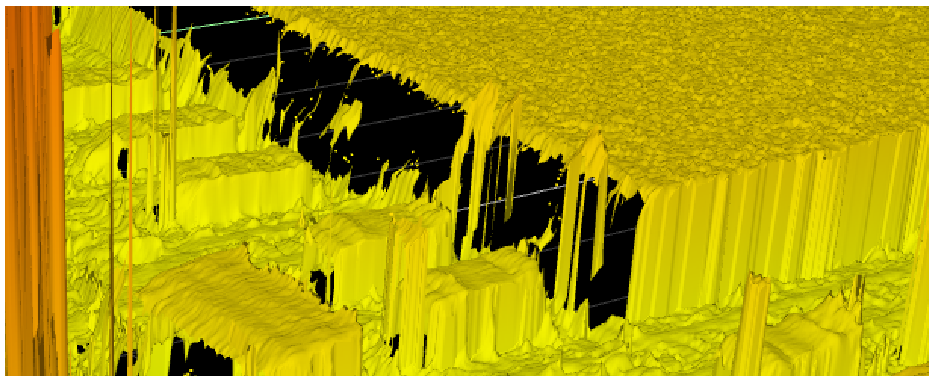
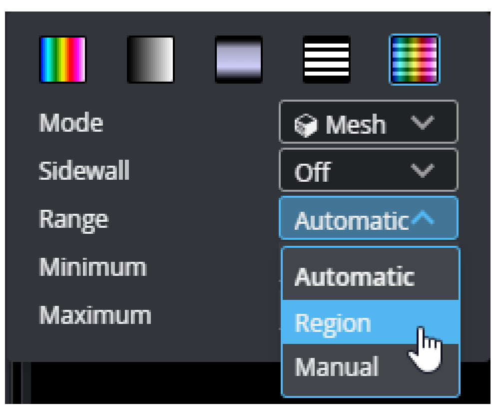
Intensity

Intensity mode () displays intensity data over the Surface data if it is available.

When intensity data is enabled on the **Acquire** > **Scan** page, sensors produce intensity images that measure the amount of light reflected by an object. An 8-bit intensity value is output for each data point in the scan data, where a darker shade represents a smaller value and lighter shade represents a larger value. The sensor applies the same coordinate system and resampling logic as the ranges to the intensity values. For more information, see *Scan Modes and Intensity* on page 195.

Option	Description
	 **Heightmap + Intensity** Heightmap + Intensity mode (icon) displays a color heightmap combined with intensity data on the Surface data. You must enable intensity data before scanning for this mode to be available. For more information, see *Scan Modes and Intensity* on page 195. 
Mesh and Points modes	You can choose whether the data viewer displays Surface data as a mesh or points. Note that the data viewer Mesh mode is unrelated to the Mesh data type. (For information on the Mesh data type, see *Mesh Measurement* on page 747.)

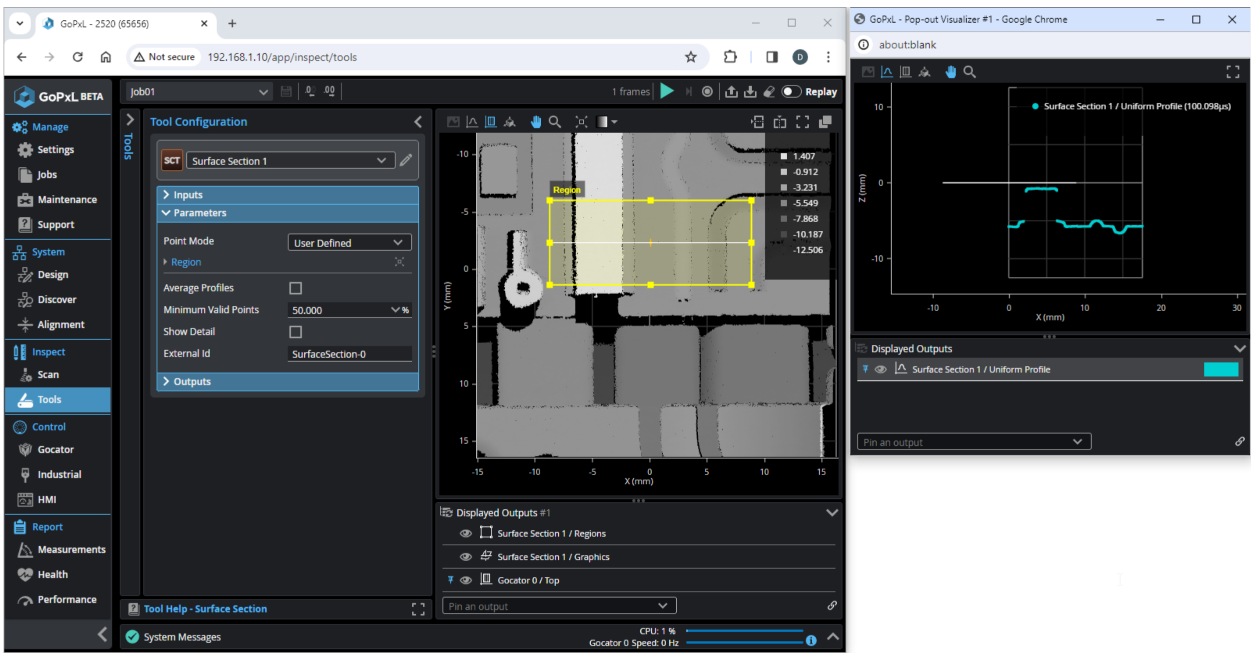
Option	Description
	 Points mode displays the data as discrete data points. This mode is useful in scan data that contains noise around edges, and can show hidden structure.  Mesh mode displays Surface data by connecting the data points with polygons. 
Sidewall mode	You can toggle the data viewer between hiding and showing polygons which are vertical or nearly vertically oriented. This can produce a more realistic-looking part for demonstrations.

Option	Description
	 For example, in the following, the sidewalls are enabled, resulting in the lines shown at the edges of the PCB components.  Note that this setting only affects the *appearance* of scan data in the data viewer. It does not change the scan data and therefore does not affect measurements.
Heightmap range scaling	The data viewer displays heightmap information in pseudo-color or grayscale. The height axis (Z) is color-coded in the legend to the right in the data viewer. You can adjust the scaling of the height map using **Range** and its related settings.  Do one of the following: • To automatically set the scale, choose **Automatic** in the **Range** drop-down. • To automatically set the scale based on sub-region you select on the heightmap,

Option	Description
	choose **Region** in the **Range** drop-down and adjust the yellow region box in the data viewer to the desired location and size. • To manually set the scale, choose the **Manual** in the **Range** drop-down and enter the minimum and maximum height to which the colors will be mapped.

Using Multiple Data Viewer Windows

You can pop out multiple data viewers outside of the main browser window. You can configure data viewers to different views, different display modes, and different sets of pinned outputs. This lets you more easily monitor or set up complex applications.

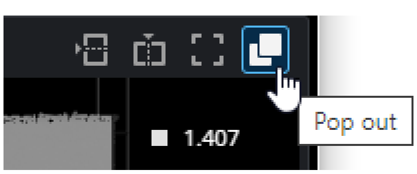


Main view in original browser window showing surface data and a defined section, and a second window showing a Profile tool running on the section.

External data viewer windows provide the same functions as the main data viewer via the toolbar above the viewer (except for the ability to open a new window). External windows also include a Displayed Outputs panel at the bottom and support the pinning of outputs; pinning in external windows is independent from the main view data viewer and other external windows. For more information on pinning outputs, see *Pinning Outputs* on the next page.

To open a new external data viewer window

- In the toolbar of the main view data viewer, click the Pop out button ( ).



A new window opens containing a separate data viewer.

Use the tool bar at the top of the new data viewer to choose and modify the view. For more information, see *Data Viewer* on page 88.

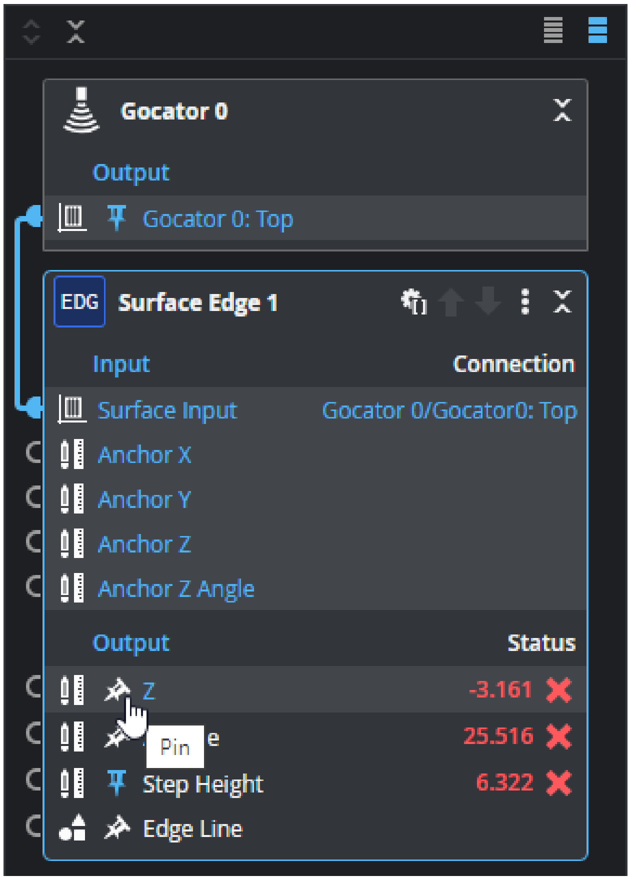
Pin outputs to the new data viewer as in the main view data viewer. For more information, see *Pinning Outputs* below. Any outputs pinned in the main view when you open a new data viewer window appear already pinned in the new window, but pinning in data viewers is otherwise independent.

Pinning Outputs

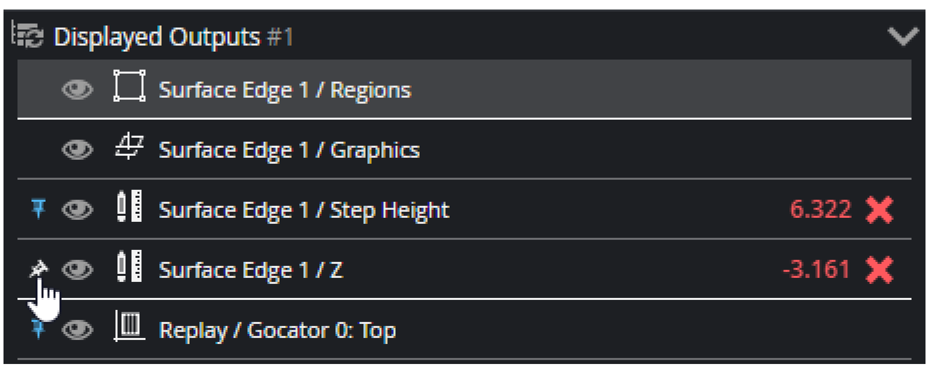
You can “pin” one or more outputs (such as scan data, measurements, and geometric features) to a data viewer. Pinned outputs remain visible in a data viewer at all times, even when you click on a different tool, measurement, or feature in one of the outputs listed in either the data viewer's **Displayed Outputs** panel or in the Tools Diagram. When no tool outputs are pinned, only the currently selected output is displayed in the data viewer. Pin information (for the main data viewer only) is stored in job files.

Pinning outputs is useful if you want to monitor multiple, independent measurements while GoPxL is running in production. Pinning is also useful when setting up tools: you can change the parameters of a tool (such as a filter) earlier in a tool chain and immediately see the impact that change has on another tool (whose output is pinned) later in the chain. This minimizes toggling and clicking between tools and measurements.

Pinned outputs are identified in GoPxL by a vertical blue "thumbtack" icon. Unpinned outputs are identified by a grey thumbtack icon. (See below.) You can pin outputs in the Tools Diagram or in a data viewer's Displayed Outputs pane.

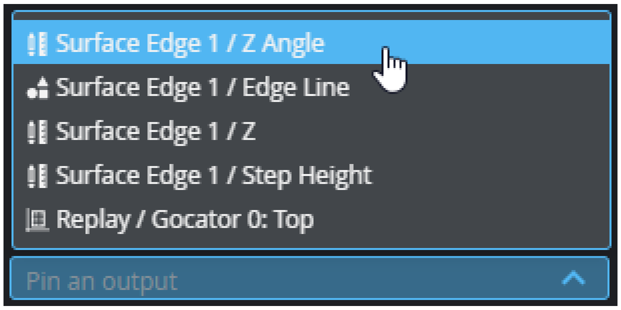


Pinning an output in the Tools Diagram.



Pinning an output in a data viewer's Displayed Outputs panel.

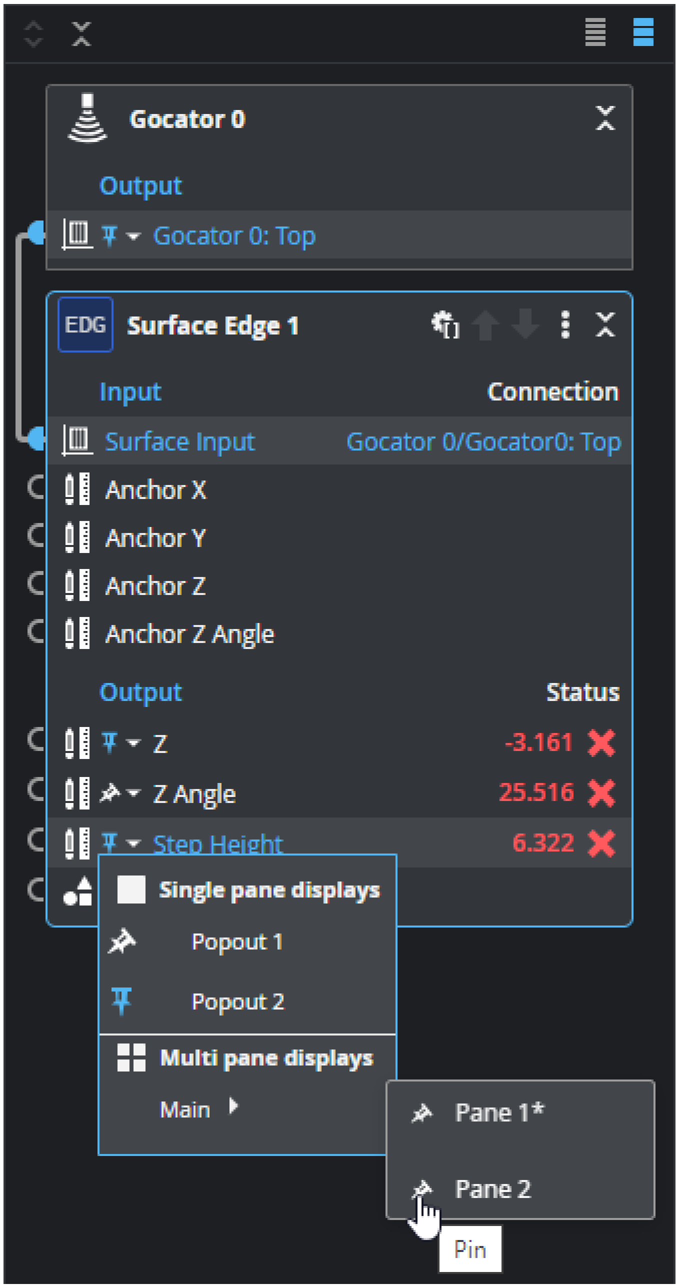
You can also choose an output from the Pin an output dropdown.



If you have created multiple data viewers, either by splitting a data viewer in the main browser window, or by popping out a new data viewer, you can choose which data viewer to pin an output to in the Tools Diagram panel.

"Single pane displays" are data viewers you've created using the "Pop out" button ( ), which display in a new browser window.

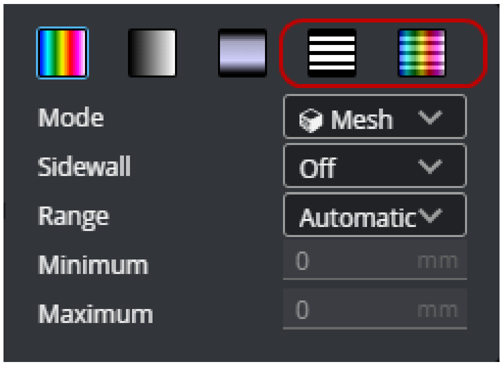
"Multi pane displays" are data viewers you've created using one of the "split" buttons (or). The pane identified by an asterisk is the original data viewer, the one from which you created others.



Intensity Output

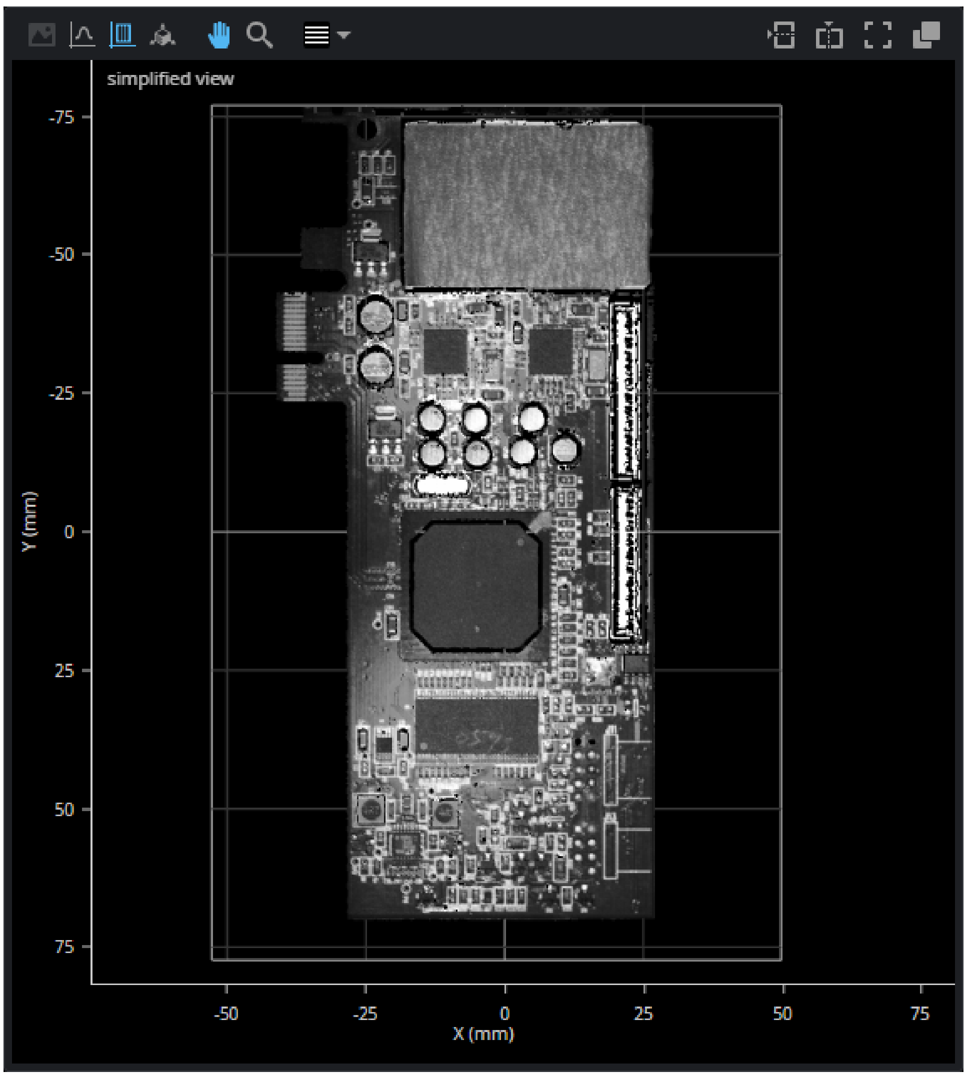
Sensors can produce intensity images that measure the amount of light reflected by an object. An 8-bit intensity value is output for each data point in the scan data. GoPxL uses the same coordinate system as the data points.

To display intensity data, choose either of the last two display options in the data viewer display options. If intensity data is not available, the icons are grayed out. Intensity data can only be shown with Surface data.



Intensity-only and Surface with intensity buttons

To be able to display intensity data, you must enable **Acquire Intensity** in the **Scan Mode** panel before scanning. For more information, see *Scan Modes and Intensity* on page 195.

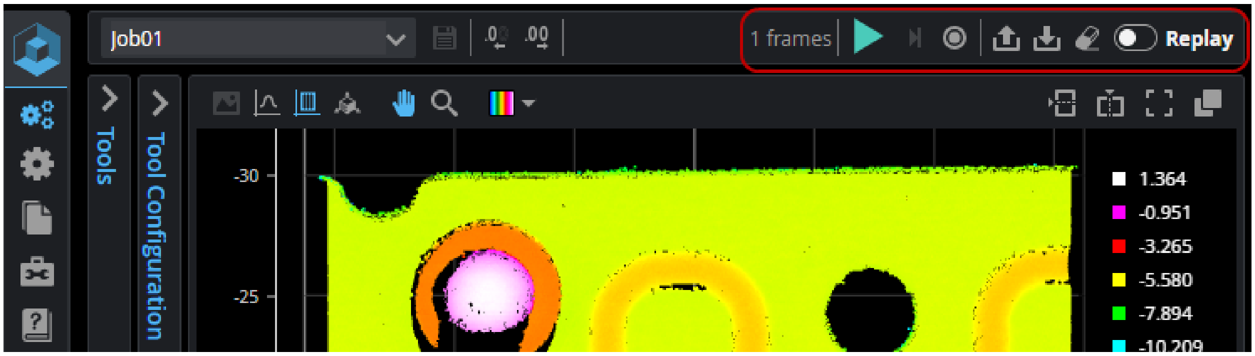


Working with Scan Data (Toolbar)

You use the buttons in the right half of the global toolbar, at the top of the interface, to do the following:

- start and stop scanning
- record replay data
- enable replay mode and play data
- upload, download, and clear replay data

	When switching between scan modes, if you have previously recorded data, you must clear it before recording replay data of a different type.



Depending on whether Replay (4, below) is disabled or enabled, the control buttons (2, below) change. When Replay is off, the controls let you start and stop data acquisition. When Replay is on, the controls let you play back recorded data.

The global toolbar with Replay disabled (top) and enabled (bottom)

Note that when GoPxL is in Replay mode, the data viewer is displayed with a blue border to help

	Element	Description
1	Frames per second and Repeat frames (looping)	These controls let you set the number of frames per second (fps) during playback, and also whether GoPxL repeats frames, by toggling the Repeat frames button (). For more information, see *Playing Back Recorded Data* on the next page.

	Element	Description
2	Data playback and recording controls	These controls let you start and stop scanning, start recording and also play back replay data. The number of frames is also indicated. The controls that are available depends on whether Replay mode is disabled or enabled. For information on starting and stopping scans, see *Starting, Stopping, and Recording* below. For information on data playback, see *Playing Back Recorded Data* below.
3	Replay actions	Lets you upload, download, and clear recorded data. Note that replay data contains any tools you've added, as well as any other configuration changes you've made in GoPxL. The file type of replay data is .gprec.
4	Replay toggle	Lets you toggle between data acquisition (toggle is to the left) and data playback (toggle is to the right).

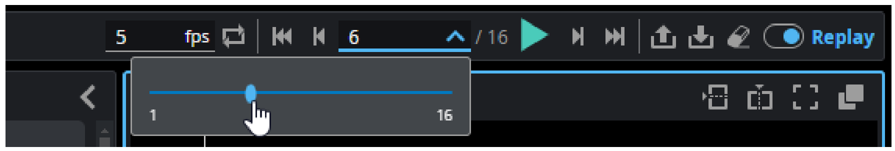
Starting, Stopping, and Recording

You start and stop data acquisition (and toggle recording) using the data control buttons when Replay is disabled.

	Element	Description
1	Start or stop scanning	The sensor starts scanning continually when you click the Start button ( ), and stops when you click the Stop button ( ).
2	Single frame (Snapshot)	The sensor acquires a single frame of data. For line profile sensors, you can't acquire a single frame of data if the sensor is in Surface mode, only in Profile or Image mode.
3	Record data	Toggles recording scan data as it is acquired. The number of frames recorded is displayed next to the Start button. When recording is enabled, the Record data button turns red ( ).

Playing Back Recorded Data

When Replay mode is enabled, you play recorded data back using the data control buttons.

	Element	Description
1	Playback rate and repeating frames	When "Repeat frames" ( ) is enabled, scan data is played back in a loop. This is useful when inspecting data or configuring tools using recorded data, and also for demos.
2	Go to previous / first frame	Displays the previous frame or jumps to the first frame of the recorded data and displays it.
3	Frame position	Lets you go to a specific frame, either by entering a value in the field, or by using a slider. 
4	Start playback	Starts playing back frames of the recorded data, at the specified number of frames per second.
5	Go to next / last frame	Displays the next frame or jumps to the last frame of the recorded data.

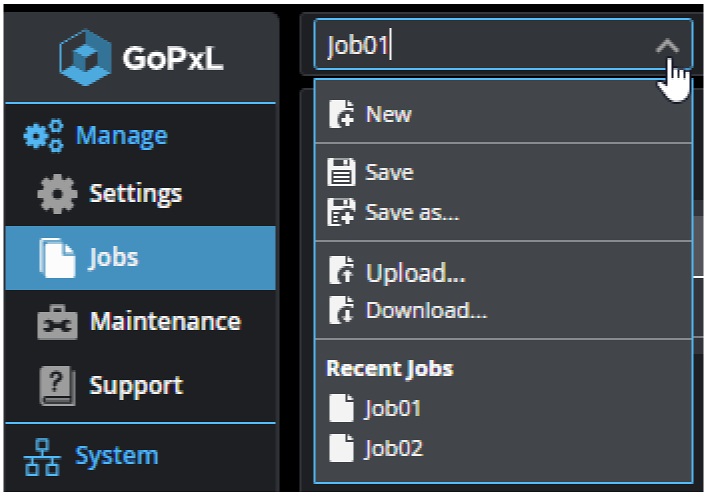
Creating, Saving and Loading Jobs (Toolbar)

Jobs contain sensor, tool, and control configurations for a particular inspection or quality control application. (For more information, see *Configuring GoPxL* on page 116.)

A sensor can store many jobs, subject to sensor storage limitations. Running GoPxL on a PC or using GoMax NX provides more storage. Being able to switch between jobs is useful when a sensor is used with different constraints during separate production runs. For example, width decision minimum and maximum values might allow greater variation during one production run of a part, but might allow less variation during another production run, depending on the desired grade of the part.

Most of the settings that you can change in GoPxL's web interface are temporary until saved in a job file. If there is a job file that is set as the default, it will be loaded automatically when GoPxL starts.

GoPxL lets you perform several job-related operations from the global toolbar at the top of the interface:



Other operations are available on the **Manage** > **Jobs** page. For more information on jobs and working with them, see *Jobs* on page 119.

Configuring GoPxL

The following sections describe how to configure GoPxL using its web interface.

GoPxL can run on a sensor, or you can run GoPxL on a Windows PC and connect to a sensor running GoPxL from the application (in order to increase performance, for example). When GoPxL runs on a PC, we call this a local or PC instance of GoPxL. For applications where performance is critical, you should use the PC version of GoPxL to improve sensor performance. With some sensor models, such as Gocator 4000 and Gocator 5500 series sensors, LMI *strongly* recommends using the PC version of GoPxL (or a GoMax NX accelerator).

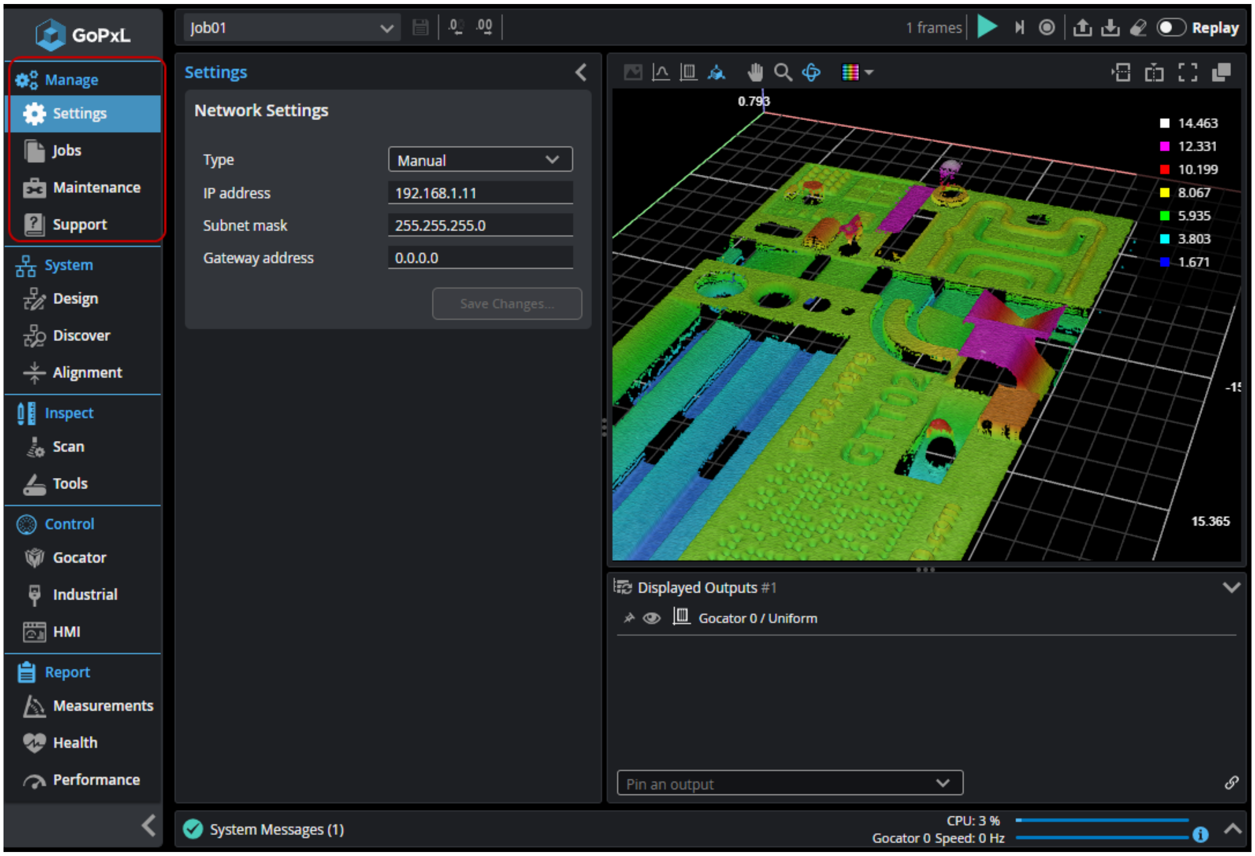
To configure GoPxL when it is running on-sensor, you connect directly to the sensor's IP address (by default, 192.168.1.10). To configure GoPxL when it is running on a PC, you connect to the IP address of the PC instance. If you can't connect to your sensor or the PC instance, use the GoPxL Discovery tool to find it and launch the GoPxL interface (for more information, see *GoPxL Discovery Tool* on page 935).

The interface of the sensor and the PC versions of GoPxL are nearly identical. For this reason, you can use this section to configure sensors whether you are using GoPxL on a sensor or on a PC. If you need to run the PC version of GoPxL, see *Running GoPxL on a Windows PC* on page 828 first.

For information on configuring GoPxL programmatically, see *GoPxL SDK and REST API* on page 871.

Sensor Management and Maintenance

You use the pages in the **Manage** category to perform networking, management, and maintenance tasks (such as software updates and sensor factory restores).

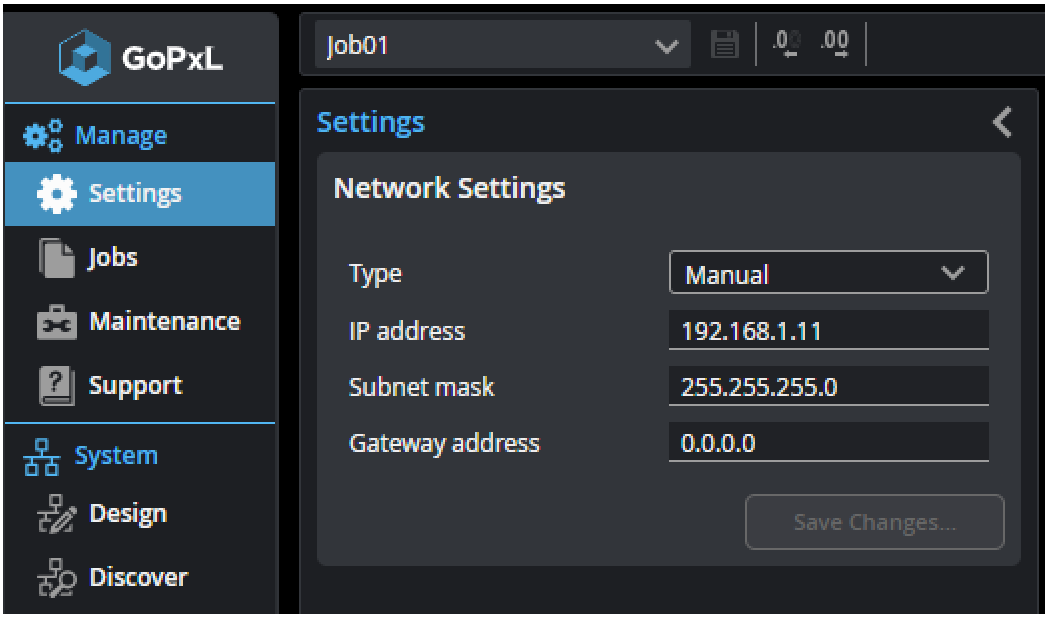


Settings page selected, displaying the panels available on that page. The selected page is highlighted.

Manage Page Name	Description
Settings	The **Settings** page lets you configure the network settings of a sensor. For more information, see *Settings* on the next page.
Jobs	The **Jobs** page lets you create, delete, and manage jobs. For more information, see *Jobs* on page 119.
Maintenance	The **Maintenance** page lets you upgrade a device's software, perform a backup or restore of a device, perform a factory restore of a device, or restart it. For more information, see *Maintenance* on page 121.
Support	The **Support** page provides software version information and lets you download a support file for troubleshooting; it also lets you upload a support file to GoPxL. You can also access and download the GoPxL documentation, download the SDK, and go to LMI's online support page. For more information, see *Support* on page 124.

Settings

The **Settings** page in the **Manage** category provides network settings. Settings must be configured to match the network to which the sensors are connected.

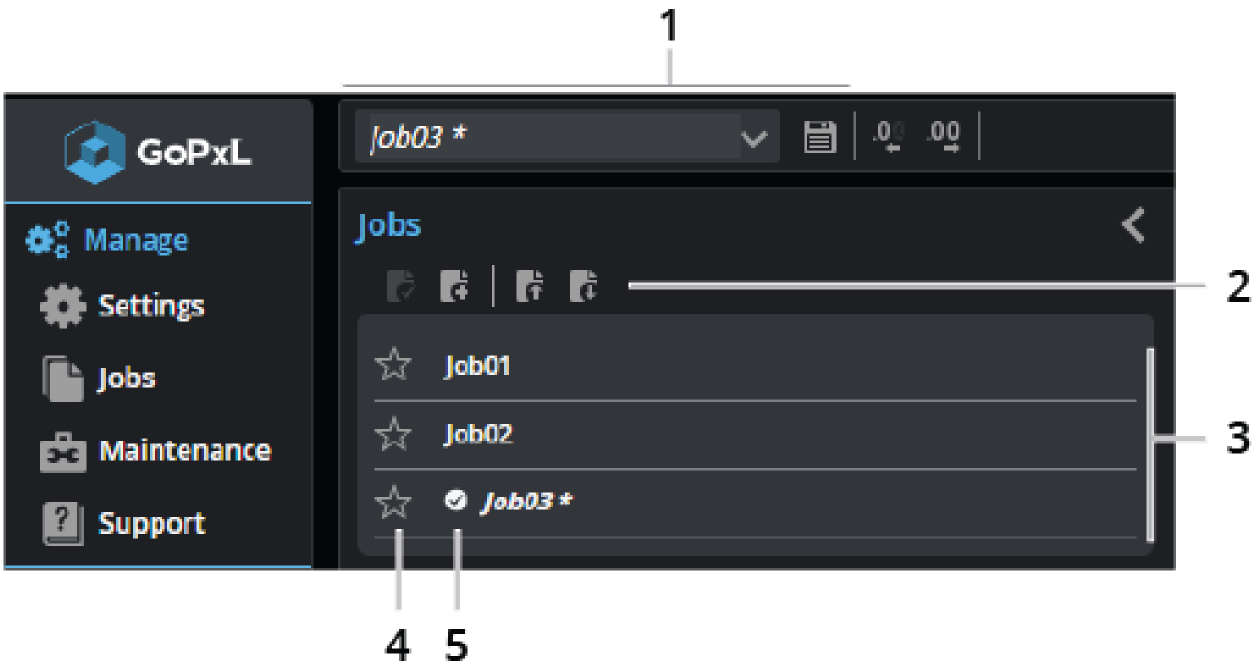


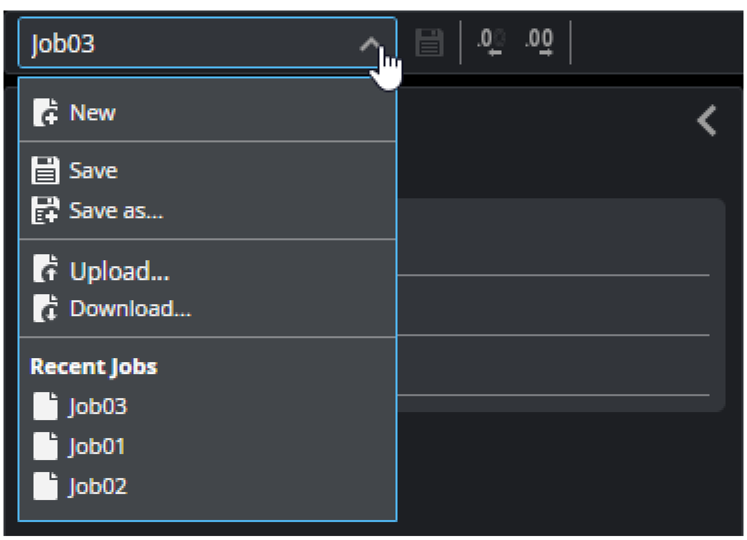
To configure the network settings:

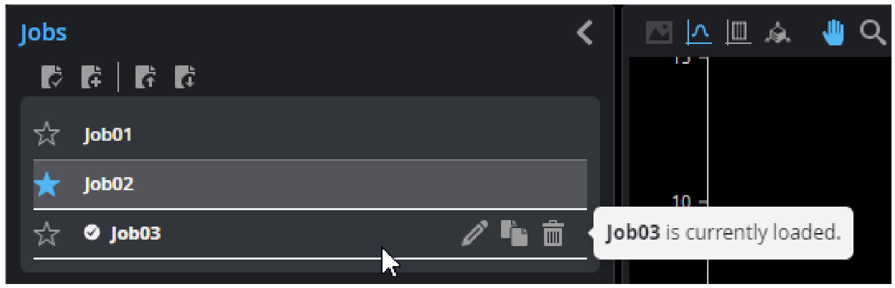
1. Go to the **Manage** > **Settings** page.
2. Specify the type (Manual or DHCP), IP, subnet mask, and gateway settings.
   The sensor can be configured to use DHCP or assigned a static IP address by selecting the appropriate option in the **Type** drop-down.
3. Click **Save Changes...**.
   You will be prompted to confirm your selection.
   The sensor restarts, using the *new* IP address. It may take several seconds for the sensor to become available.

Jobs

The **Jobs** page in the **Manage** category lets you manage jobs. The



	Element	Description
1	Job-related operations on global toolbar	Job menu and Save button. Use the job menu to create new jobs, save jobs, load recent jobs, and also upload or download jobs between the PC and the sensor. You can rename the currently loaded job from the job name field, simply by changing it and pressing enter or clicking outside the field.  When a job has unsaved changes, it is displayed in italics with an asterisk. Click

	Element	Description
		the Save button in the global toolbar to save it.
2	Job-related operations	These controls let you load or reload jobs, create new jobs, and upload or download jobs between the PC and the sensor. For explanations of the job related operations, see *Job-related operations* below.
3	Job list	The list of available jobs. You can rename a job by double-clicking it in the list. When you move the mouse pointer over a job in the list, additional tools are displayed that let you rename the job, duplicate it, or delete it.  When a job has unsaved changes, it is displayed in italics with an asterisk. Click the Save button in the global toolbar to save it.
4	Default job indicator	Clicking the default job indicator ( ) sets the job as the default.
5	Loaded job indicator	The loaded job indicator ( ) shows which job is currently loaded. Note that "untitled" is the default name of a new job file.

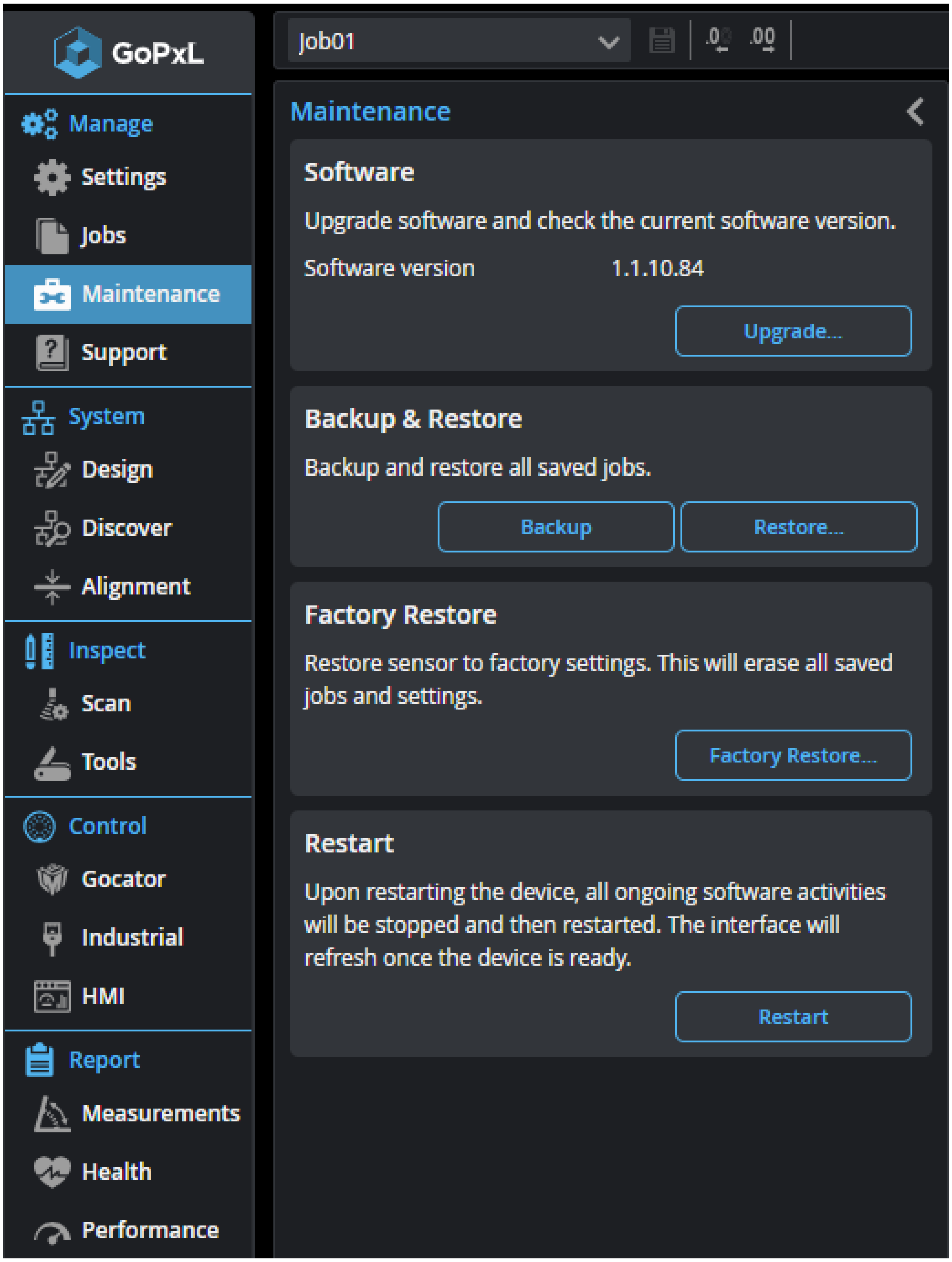
The following table describes the job-related operations.

Job-related operations

Operation	Description
	Loads or reloads the job that is currently selected in the job list.
	Creates a new job called "untitled {n}" and loads it.
	Saves the current job. (Disabled if the job contains no changes.)
	Uploads or downloads a job between GoPxL and the client computer. Job files have a .gpjob extension.
	Lets you edit the name of the job under the mouse pointer. You can also double-click on the job's name to stat editing it.
	Duplicates the job under the mouse pointer.
	Deletes the job under the mouse pointer.

Maintenance

You use the **Maintenance** page in the **Manage** category to perform basic maintenance.



The **Maintenance** panel lets you do the following:

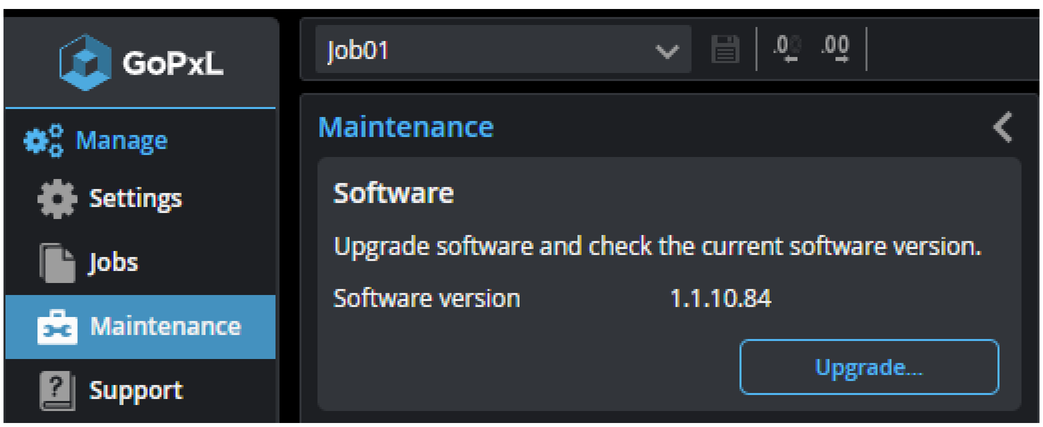
- **Upgrade...**: Upgrade the device's software. Note that you can't upgrade a sensor if you are running it through a PC instance of GoPxL. For more information on upgrading a sensor's software, see *Software Upgrade* on the next page.

- **Backup / Restore...**: Back up and restore all saved jobs and recorded data. Note that backing up and restoring on a PC instance of GoPxL only affects the PC instance, not the sensor. For more information, see *Backup and Restore* below.
- **Factory Restore...**: Restore the sensor (or the PC instance of GoPxL) to factory defaults. This erases *all* saved jobs and settings, as well as files created by tools such as Surface Pattern Matching and Surface Track. Performing a factory restore on a PC instance of GoPxL connected to a sensor *only* affects the PC instance, not the sensor; doing this also disconnects the sensor from the PC instance.
- **Restart**: Restart the sensor. Note that you can't restart a sensor running through a PC instance of GoPxL. To do this, you must remove the sensor from the PC instance; for more information, see *Stopping Acceleration* on page 837.

Software Upgrade

LMI recommends routinely updating software to ensure that sensors always have the latest features and fixes.

Note that if you are running a sensor through a PC instance of GoPxL, you can't upgrade its software. To do this, you must temporarily remove it from the sensor group in the PC instance, access the sensor directly at its IP address, and perform the upgrade. You can then add it back to the sensor group on a PC instance. (Note that it must be the same software version.) For more information, see *Stopping Acceleration* on page 837.

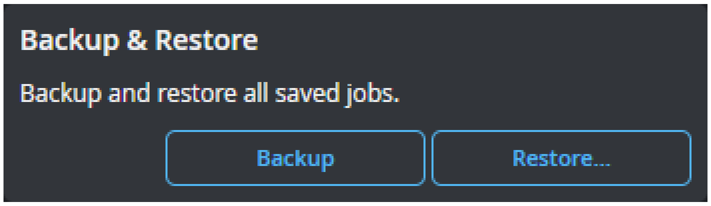


	In multi-sensor systems, all sensors must use the same software version.

Backup and Restore

Sensor backups let you keep the state of a sensor or system in a safe location, and restore a sensor to a previous state in case you have made unwanted changes. You create backups in the **Maintenance** page. Backup files contain the following:

- All jobs
- Current recorded data
- Layout set up on the **System** > **Design** page
- Alignment performed on the **System** > **Alignment** page
- Other configurations such as scan settings and control settings



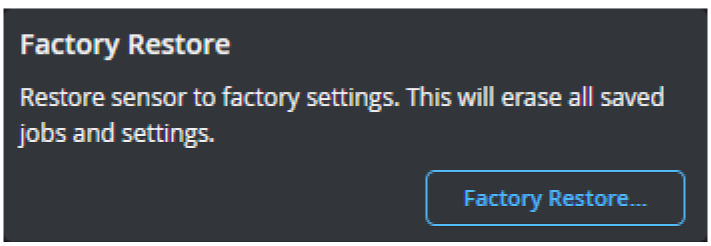
	You should always create a sensor backup file in the unlikely event that a sensor fails and a replacement sensor is needed. If this happens, the new sensor can be restored with the backup file.

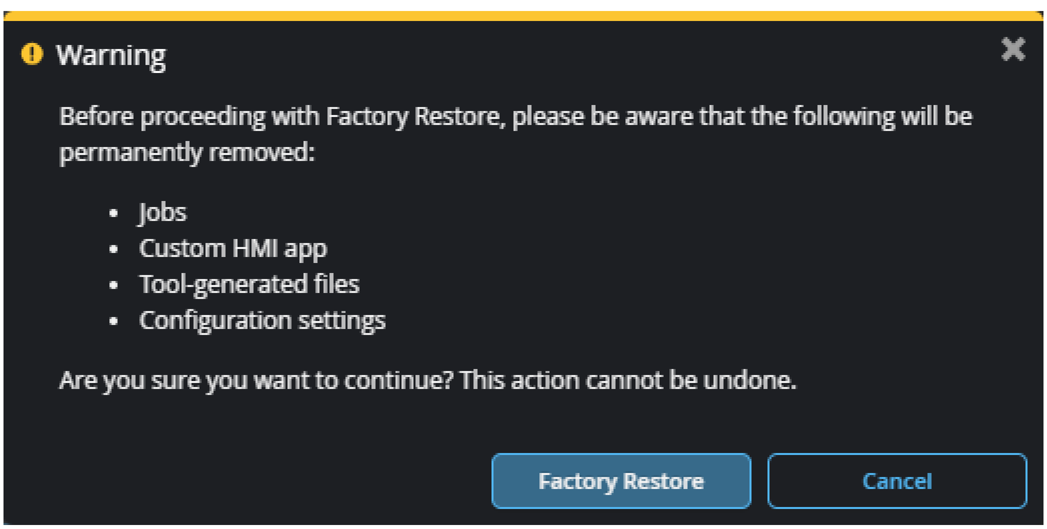
When you create a backup, your browser may ask if you want to allow downloading the file.

If you are running a sensor through a PC instance of GoPxL, performing backup, restore, and factory restore operations only affects the PC instance of GoPxL: jobs and recorded data saved on the sensor are not backed up and are not overwritten by a restore.

Factory Restore

You can perform a factory restore to quickly remove all jobs and configurations, as well as the files created by tools (such as Surface Pattern Matching, Surface Track, and so on). This also removes Custom HMI apps.



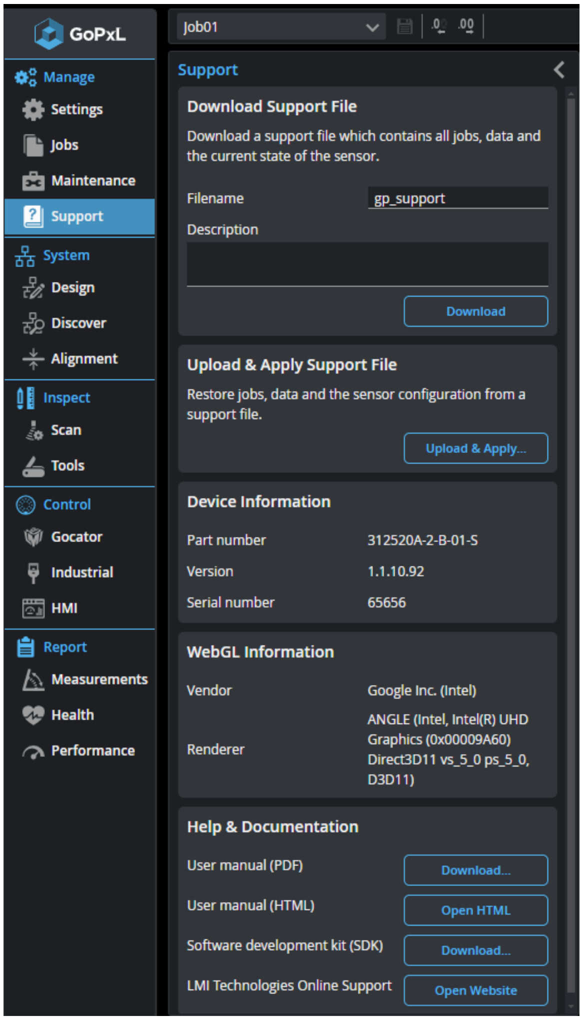


	You should create a backup file of your sensor in case of unwanted changes and in the unlikely event that a sensor fails and a replacement sensor is needed. If this happens, the new sensor can be restored with the backup file.

Support

The **Support** page in the **Manage** category lets you download or upload a support file and get software version information. Support files are useful for supporting development and diagnosing issues.

You can also access the user manual and download the SDK from here.



Support files contain everything in backup files, but also contain *additional diagnostic information*:

- All jobs
- Current recorded data
- Layout set up on the **System** > **Design** page
- Alignment performed on the **System** > **Alignment** page
- Other configurations such as scan settings and control settings

Creating a Sensor System

You use the **Design** page in the **System** category to create a sensor system containing two or more sensors.

GoPxL does not currently support G3 multi-sensor systems.

GoPxL does not currently support mixed-model multi-sensor systems.

The **System** > **Discover** page lets you see all sensors, including those that are unavailable (either because of a GoPxL version mismatch, or because a sensor is already running through another PC instance of GoPxL). For information on running a sensor through a PC instance of GoPxL, see *Running GoPxL on a Windows PC* on page 828.

Adding Sensors and Configuring Multi-sensor Systems

GoPxL lets you create a dual- or multi-sensor system, which combines the scan data (Image, Profile, or Surface data) from individual sensors into a single frame, on which you can run measurement tools.

If you are creating a multi-sensor G2 system in a *ring layout* and intend to perform a high-accuracy alignment (six degree of freedom) using the Surface Align Ring tool (see *Ring Layouts (Surface Align Ring Tool)* on page 182), place *all* sensors in the Top row, even though some of the sensors will be physically below others. The configuration file created in Surface Align Ring and used in Surface Mesh applies the required transformations to the sensors.

If you are using the built-in stationary polygon alignment (five degree of freedom), use the top and bottom rows.

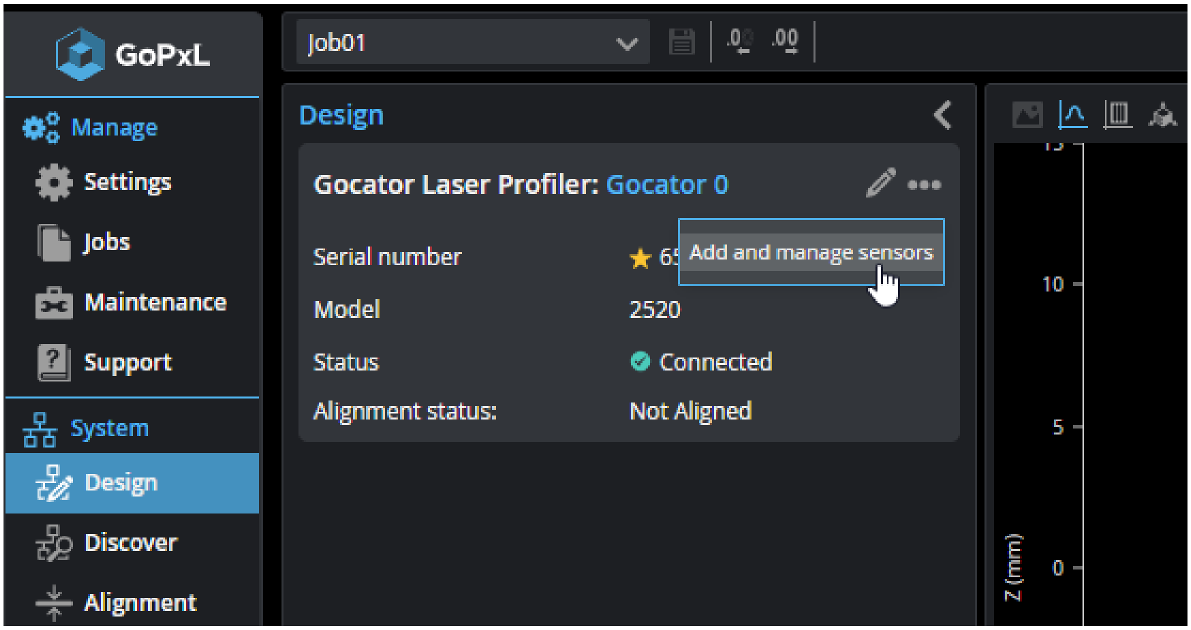
To create a dual- or multi-sensor system

1. In a web browser, connect to the sensor with which you want to group other sensors.

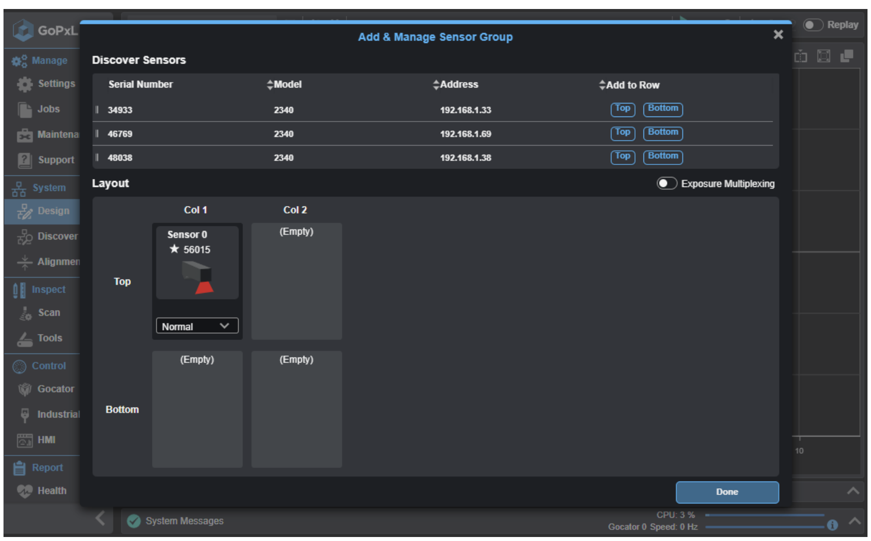
   GoPxL will run on this sensor and control the other sensors.

   The default Gocator sensor IP address is 192.168.1.10. If you don't see your sensor in the browser, see *GoPxL Discovery Tool* on page 935.

2. In the GoPxL interface, go to the **System** > **Design** page, click the three dots menu and click **Add and Manage Sensors**.

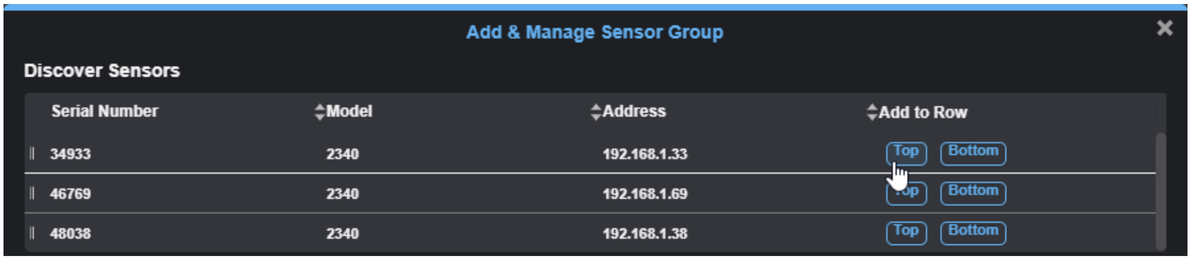


The **Add and Manage Sensors** dialog opens, displaying the available sensors, and the sensor on which you're using GoPxL in the Layout section.

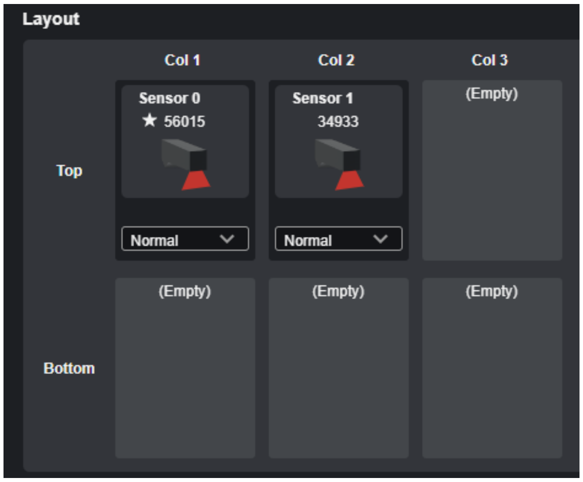


If you don't see sensors that you expect to see, go to the **System** > **Discover** page to see all sensors on the network, or use the Discovery tool (for more information, see *GoPxL Discovery Tool* on page 935).

3. In the dialog, from the list of available sensors at the top, click the **Top** or **Bottom** button next to a sensor to add it to the sensor group in the top or bottom row, respectively.



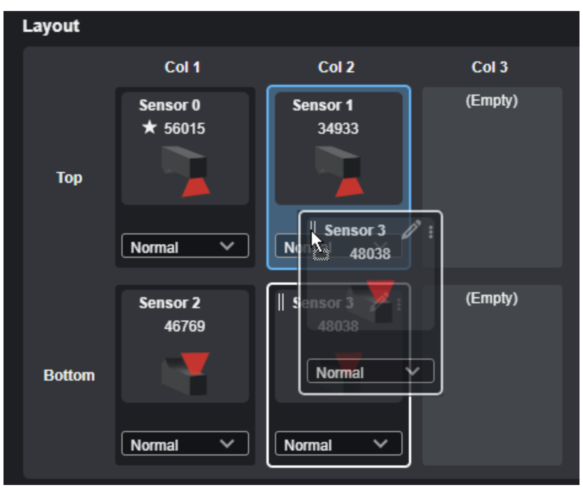
The sensor is added to the sensor group.



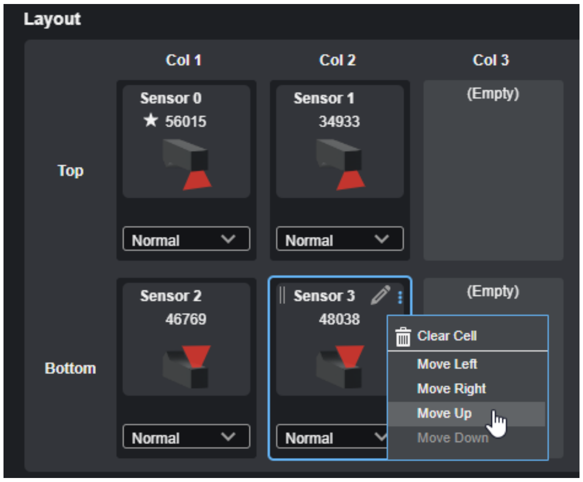
You can also add the sensor to a specific cell in the grid using drag and drop mouse operations, using the grabber icon next to a sensor (‖).

4. (Optional) If you need to reorganize the sensors in the layout, you can hover over a sensor in a grid cell and do one of the following:

   Use drag and drop mouse operations to move a sensor, by clicking on a sensor's grabber and moving it to another cell.



You can also hover over a cell and click the sensor's Show more icon and choose a "Move" command.



5. (Optional) If necessary, change the orientation of sensors.

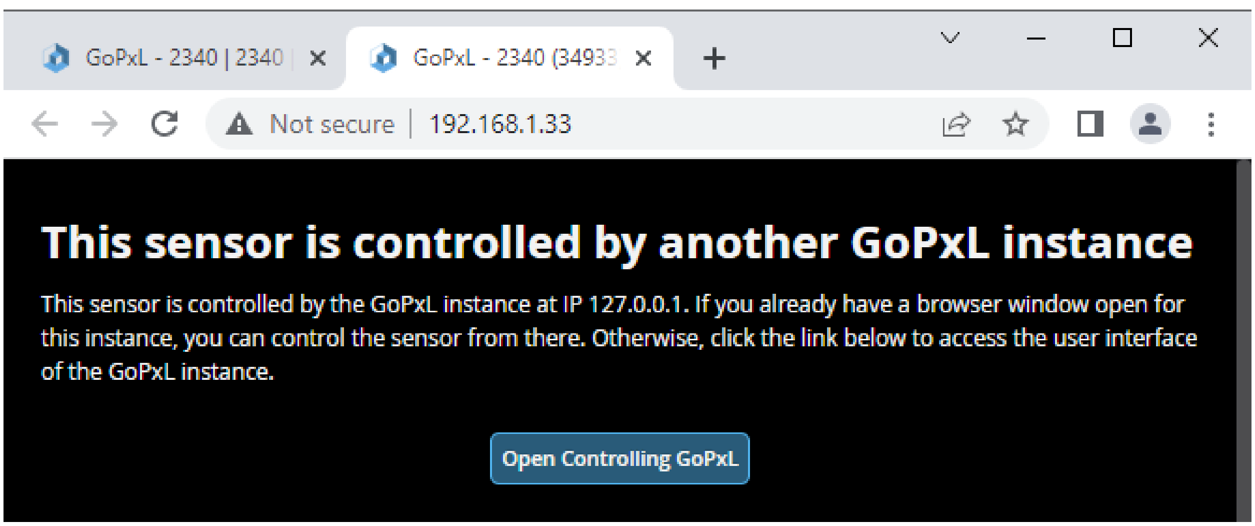
   For more information, see *Changing Sensor Orientation* below.

6. (Optional) If necessary, enable exposure multiplexing.

   For more information, see *Enabling Exposure Multiplexing* on the next page.

7. After you have finished configuring your system, click **Done** and then save your job.

After you have grouped a sensor with another sensor, the grouped (also known as "remote") sensor's web interface will show that it's currently being controlled by another GoPxL instance, namely, GoPxL running on the sensor on which you configured the system.

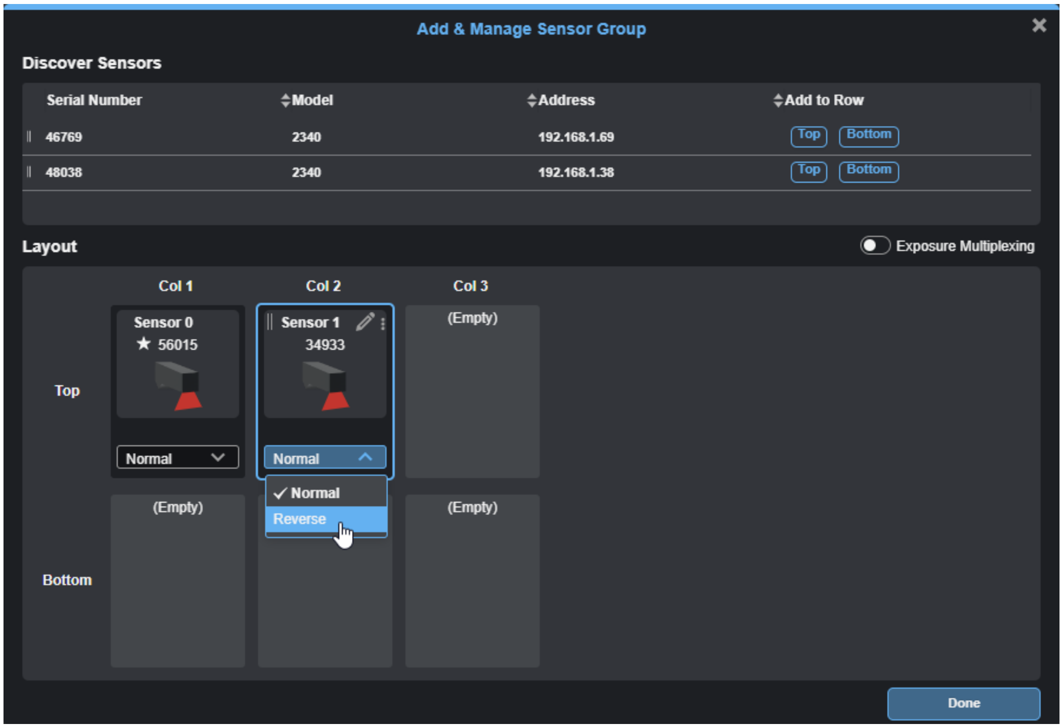


Changing Sensor Orientation

If a sensor's orientation in the layout is such that the sensor's positive Y axis is the *same* as the motion of the target you need to change the sensor's orientation.

To determine the positive Y axis of your sensor, see the *Coordinate System Orientation* diagram for your sensor in the *Sensors* on page 975.

You change a sensor's orientation by changing its orientation to Reverse under a sensor in the layout grid. This information allows the alignment procedure to determine the correct system-wide coordinates for scan data and measurements. For more information on sensor and system coordinates, see *Coordinate Systems* on page 65.



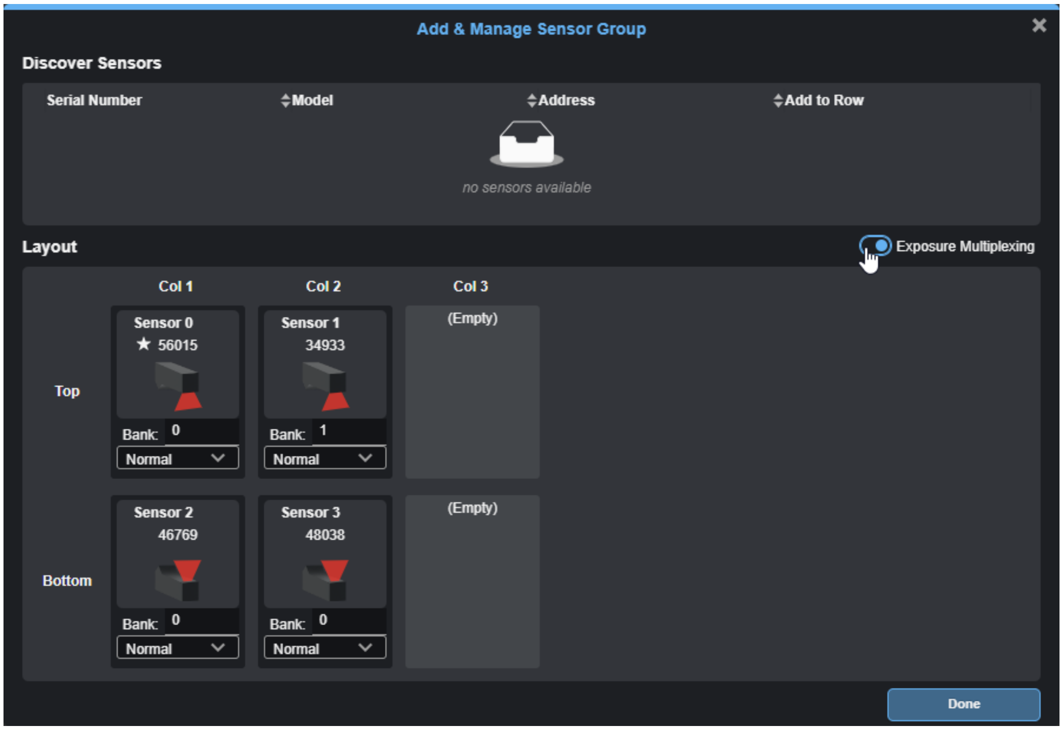
Enabling Exposure Multiplexing

If the sensors in a dual- or multi-sensor system are mounted in a way that allows the camera from one sensor to detect the light from another sensor, you should enable the **Exposure Multiplexing** option in the **Add & Manage Sensor Group** dialog. When the setting is enabled, you assign sensors in the sensor group to different "banks." GoPxL creates a time offset for exposures between the banks, ensuring that they are not triggered at the same time. Using this setting may reduce the maximum frame rate.

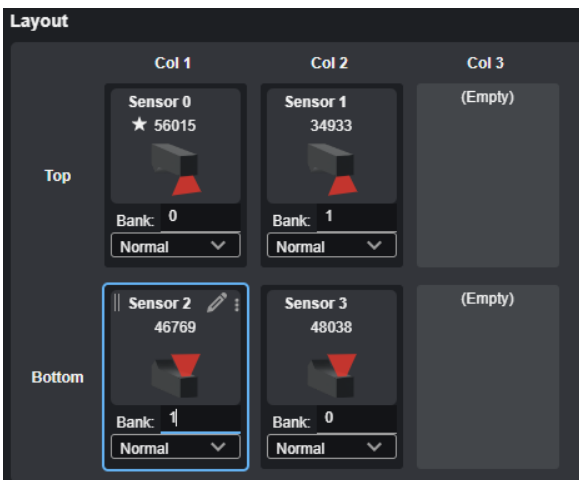
To enable and configure exposure multiplexing

1. In the **Manage Sensor Group** dialog, move the **Exposure Multiplexing** toggle to the right.

   Bank fields display under each sensor previously added to the sensor group in the **Layout** area.

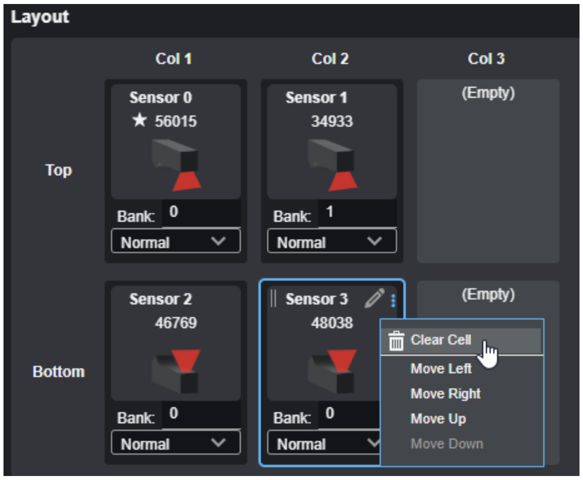


2. Set the values in the **Bank** fields to assign sensors to the desired banks.

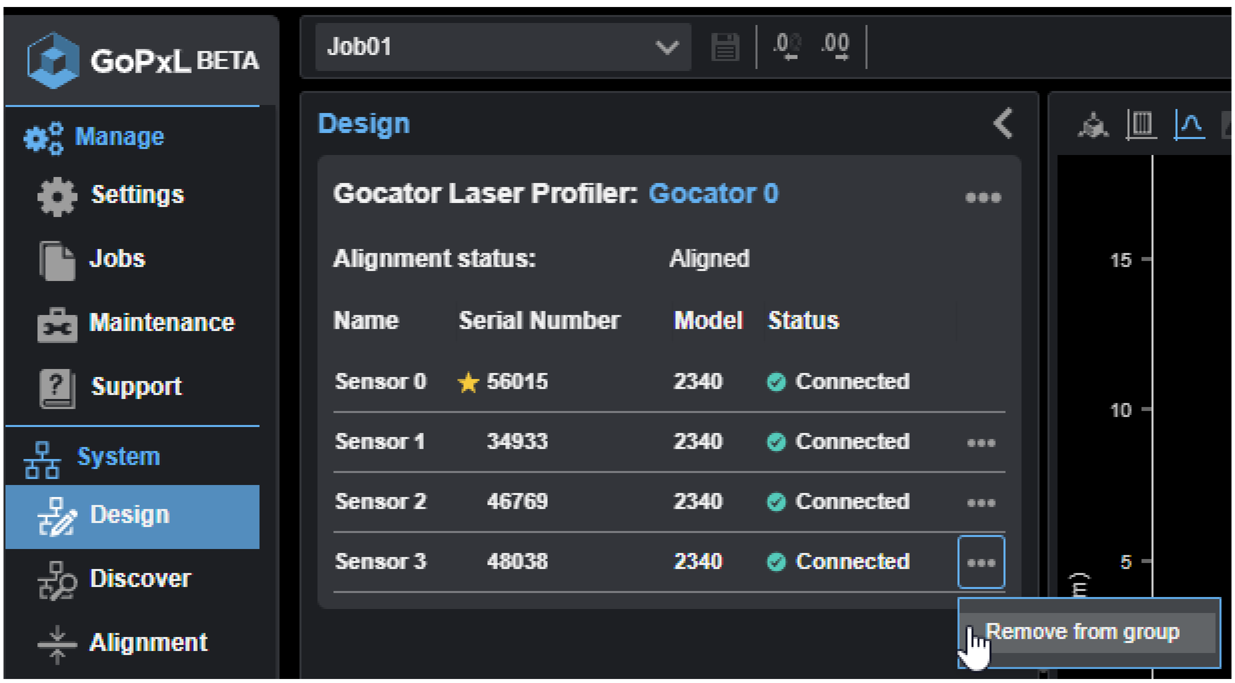


Removing a Sensor from a System

To remove a sensor from a multi-sensor system, in the **Add and Manage Sensor** dialog, click the three dots menu and choose **Clear Cell**. You can also drag a sensor from a layout cell to the sensor list above the layout grid.

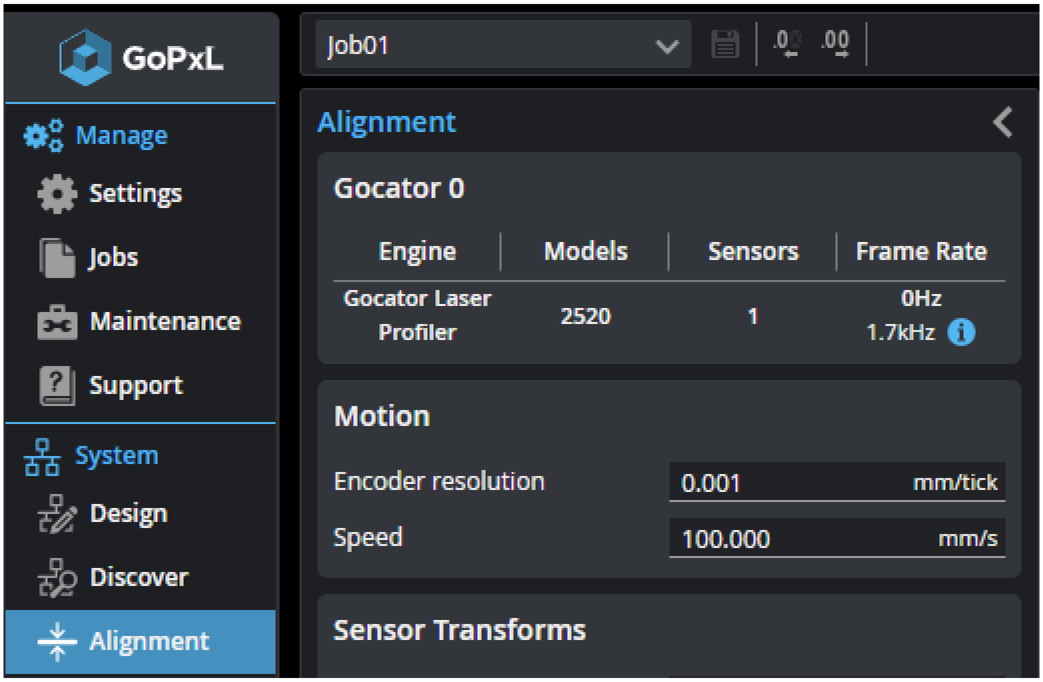


You can also go to the **System** > **Design** page, and in the row of the sensor you want to remove, click the three dots and select **Remove sensor from group**.



Configuring Motion

If your sensor system includes a transport system such as a conveyor, to produce accurate scans, you should configure the settings in the **Motion** section on the **System** > **Alignment** page. Although GoPxL lets you automatically configure motion settings during the sensor alignment routine available on the **Alignment** page (for more information, see *Aligning Sensors* on the next page), it's more accurate to manually enter the encoder resolution or travel speed values provided by the manufacturer of your transport system or its components.



If your transport system includes an encoder, you should configure **Encoder Resolution**. If your transport system does not include an encoder (or if it does and you need to temporarily use time-based triggering for testing purposes), you should configure **Speed**.

Setting Encoder Resolution

Encoder resolution is expressed in millimeters per tick, where one tick corresponds to *one* of the four encoder quadrature signals (A+ / A- / B+ / B-).

> Encoders are normally specified in *pulses* per revolution, where each pulse is made up of the four quadrature *signals* (A+ / A- / B+ / B-). Because the sensor reads each of the four quadrature signals, you should choose an encoder accordingly, given the resolution required for your application.

Setting Travel Speed

Set the value of the **Speed** setting to correctly scale scans in the direction of travel in systems that lack an encoder but have a conveyor system that is controlled to move at constant speed. Travel speed is expressed in millimeters per second.

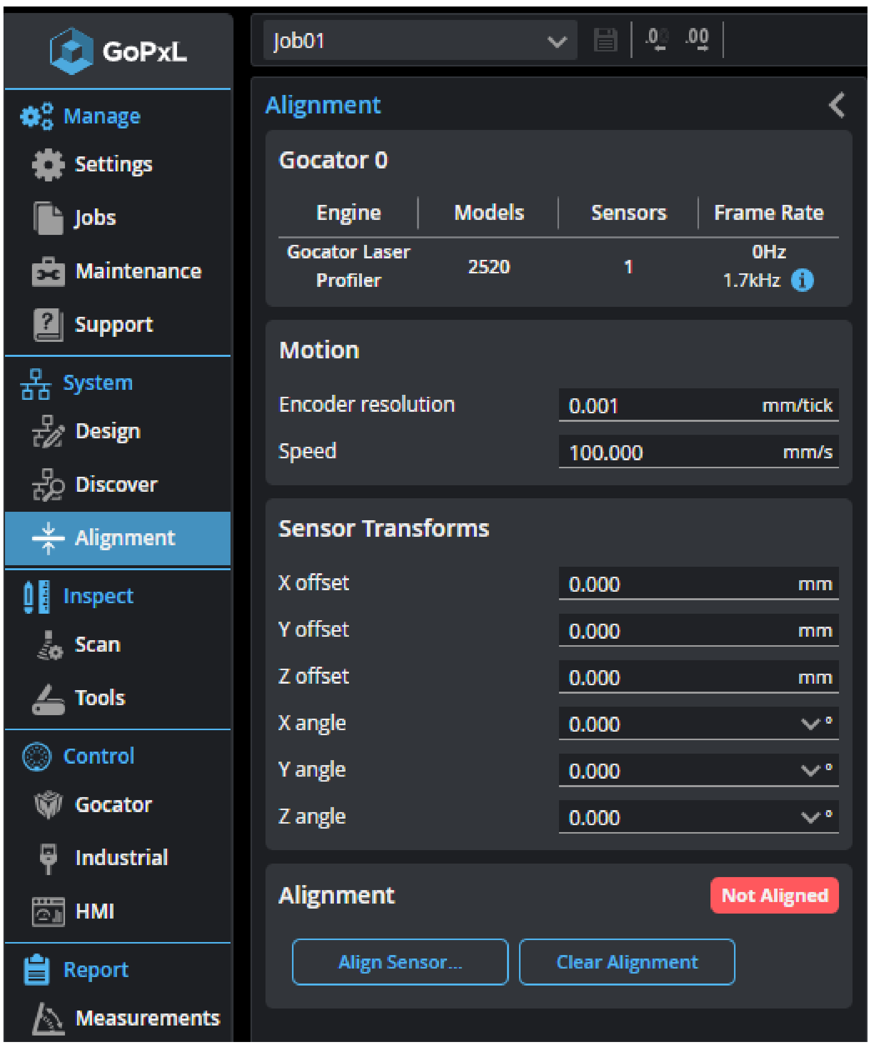
Aligning Sensors

Alignment is the process you use to automatically calculate transformations (rotations and translations / offsets) that are applied to a sensor's scan data while it is scanning targets. If you do not correct for these rotations, scan data may be too distorted for your application, and your measurements may therefore be inaccurate. Alignment is often required for various reasons:

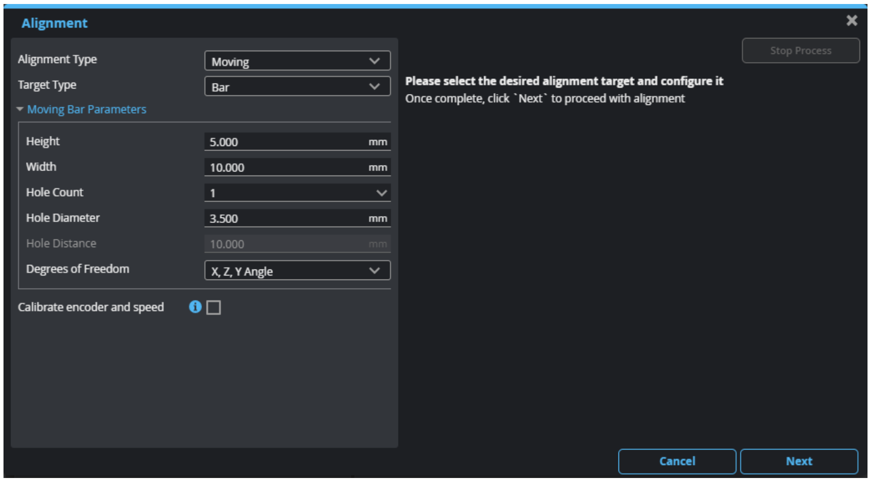
- To compensate for sensor mounting inaccuracies relative to the intended scanning surface, and to other sensors in dual- or multi-sensor systems.

- To set a Z (height) reference plane, using a flat surface or an alignment target.
- To accommodate intentional rotation of sensors, or intentional offsets of sensors in multi-sensor systems.
- To merge profiles in dual- and multi-sensor systems so that the combined profiles can be measured (setting a common coordinate system).
- Optionally, to determine the encoder resolution (if present) and the speed of the transport system. (In many systems, the reference surface is a conveyor belt.) This is only possible using the first of the two methods described below.

With G2 sensors, unless you need 6 degrees of freedom, you should use the alignment procedure on the **System** > **Alignment** page, by clicking the **Align Sensors** button. The routine provides step-by-step instructions.



Once you click the **Align Sensors** button, an alignment "wizard" goes through the steps required.



Alignment page set to Moving Bar alignment, with 3 degrees of freedom selected.

Two ways of performing alignment on sensors are available:

- Using the alignment routines available on the **System** > **Alignment** page. The alignment routines provide step-by-step instructions to perform the alignment. LMI recommends using the alignment routines if your application doesn't require more than 5 degrees of freedom. For information on performing alignments using the alignment routines (for up to 5 degrees of freedom), see *Aligning Sensors with up to 5 Degrees of Freedom* on page 138.
- For multi-sensor G2 applications that require 6 degrees of freedom, using the alignment tools available on the **Inspect** page (the Surface Align Wide and Surface Align Ring). For more information on these tools, see *Aligning Sensors to 6 Degrees of Freedom* on page 157.

If you are not sure which type of alignment you should you for your application, see *Planning Alignment* below and *Choosing an Alignment Method* on page 136.

> Sensors are pre-calibrated and ready to deliver data in engineering units (mm) out of the box. Alignment procedures do not affect sensor calibration.

Planning Alignment

Sensors are aligned to compensate for mounting rotations and offsets of sensors: unaligned sensors, when scanning, produce inaccurate scan data and measurement results. However, depending on your measurement and accuracy requirements, you may not need to perform the built-in alignment procedure. In addition to the time and effort required to prepare alignment targets and perform the procedure, the transformations applied to scan data (the corrections) that result from the alignment procedure can reduce the maximum available frame rate, which in turn determines how fast you can scan and measure parts, or the maximum available precision in measurements.

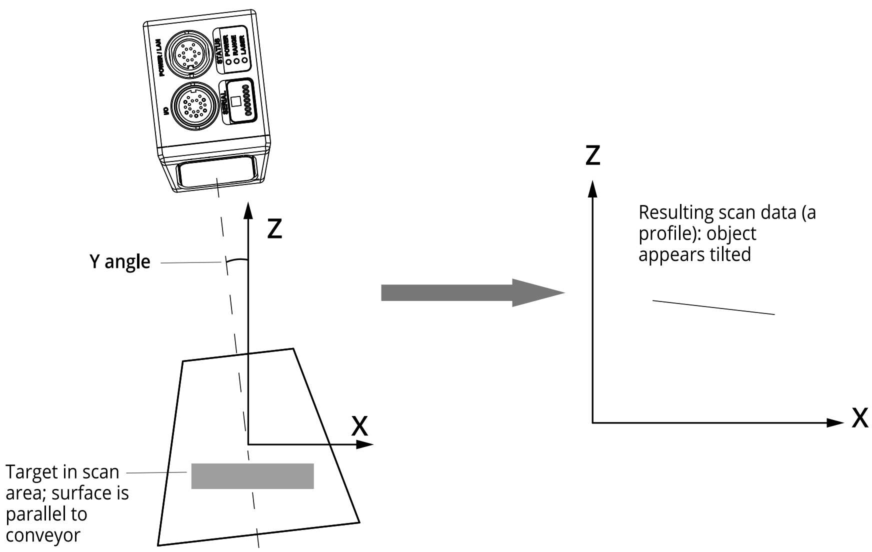
In general, if the inaccuracies are below your required tolerances, or inaccuracies are on an axis that doesn't affect your measurements, you can simply manually set a Z reference within the sensor's scan area (for example, to set the Z = 0 origin to be at the level of the conveyor).

The following sections refer to rotations and offsets on the X, Y, and Z axes. If you are not familiar with the coordinate systems used by Gocator sensors, see *Coordinate Systems* on page 65. Furthermore, when viewing the diagrams below, consult the coordinate system information of your sensor provided in *Sensors* on page 975 to get the correct orientation of the X, Y, and Z axes relative to an unaligned sensor. Note that as a rule of thumb, Y increases moving from the camera to the laser emitter.

The following sections describe the three main effects of not aligning certain degrees of freedom of a sensor; use this information to decide which alignment method to use. Remember that after mounting a sensor, it's unlikely that there will *only* be a mounting inaccuracy on or around a single axis. To clarify the impact of the rotations and offsets we describe below, we touch on them independently.

Y Angle

An unaligned sensor scanning with a Y angle rotation produces data rotated on the XZ plane. It does not distort geometry, unlike Z angle rotation (see below). So for example, with a flat object, data from one side would appear higher than data from the other side:



An exaggerated Y angle of roughly 6 degrees, producing a profile rotated around Y

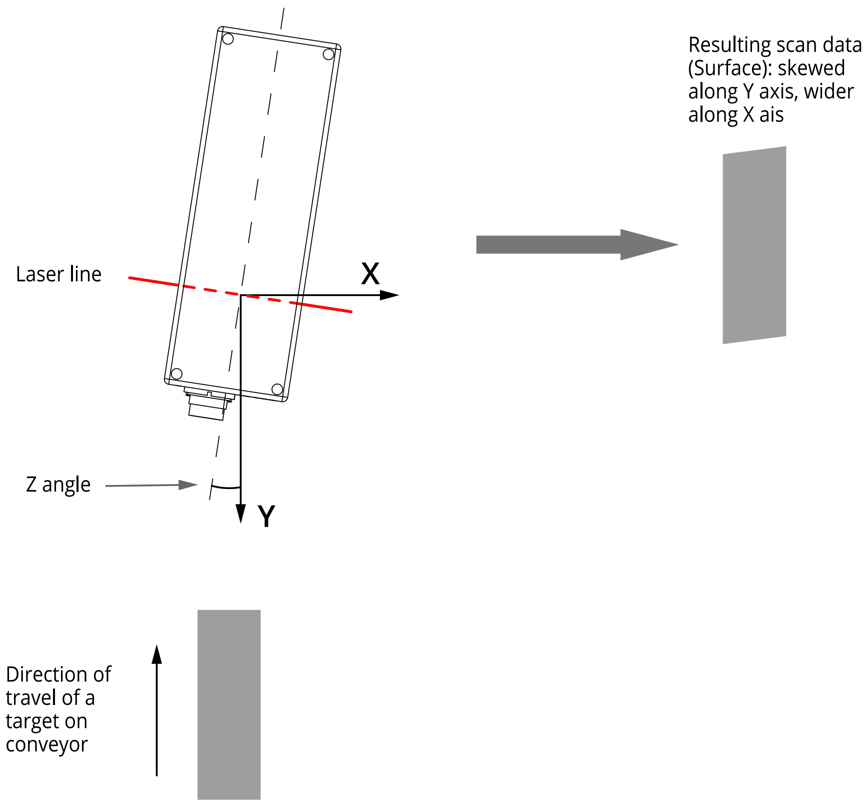
Although transformations to compensate for a Y angle mounting inaccuracy don't affect frame rates, if the resulting Z offset is acceptable in your application, you may be able to save the time and effort of performing the alignment procedure.

Y Offset

Y offset occurs in dual- or multi-sensor systems when sensors are shifted differently along the Y axis, the parts of a combined profile coming from different sensors to be offset along Y. In some situations, sensors are intentionally shifted along the Y axis, for example, with high resolution sensors, whose FOV is too small to get complete coverage when placed side by side.

Z Angle

An unaligned sensor scanning with a Z angle rotation produces data skewed on the XY plane: it creates a Y offset dependent on X position (the Z angle introduces a cosine error). So for example, a rectangular object would appear skewed along the direction of travel, and wider than it actually is.



An exaggerated Z angle of roughly 8 degrees, producing a skewed scan.
Scan data is also slightly wider along X because the laser line produces a longer profile.

However, if your application only involves measuring the height of a feature on the scanned target (so position along the Z axis), although the scan data will be inaccurate, the distortion that Z angle introduces may have no effect on your measurement results.

You can use the sensor itself to determine the mounting angle and the impact on resulting scan data. For example, you can scan a rectangular or square target whose corners are exactly 90 degrees, and then use two Surface Edge tools (for details, see *Surface Edge* on page 521) on adjacent sides to fit an edge line to those edges, and then use the Feature Intersect tool to determine the angle between those lines (for details, see *Feature Intersect* on page 727).

Note that although a Z angle mounting inaccuracy also reduces the effective FOV of a sensor, with Z angles less than 5 degrees, the impact on the FOV is minimal. (To calculate this impact, multiply the FOV by the cosine of the Z angle.)

Choosing an Alignment Method

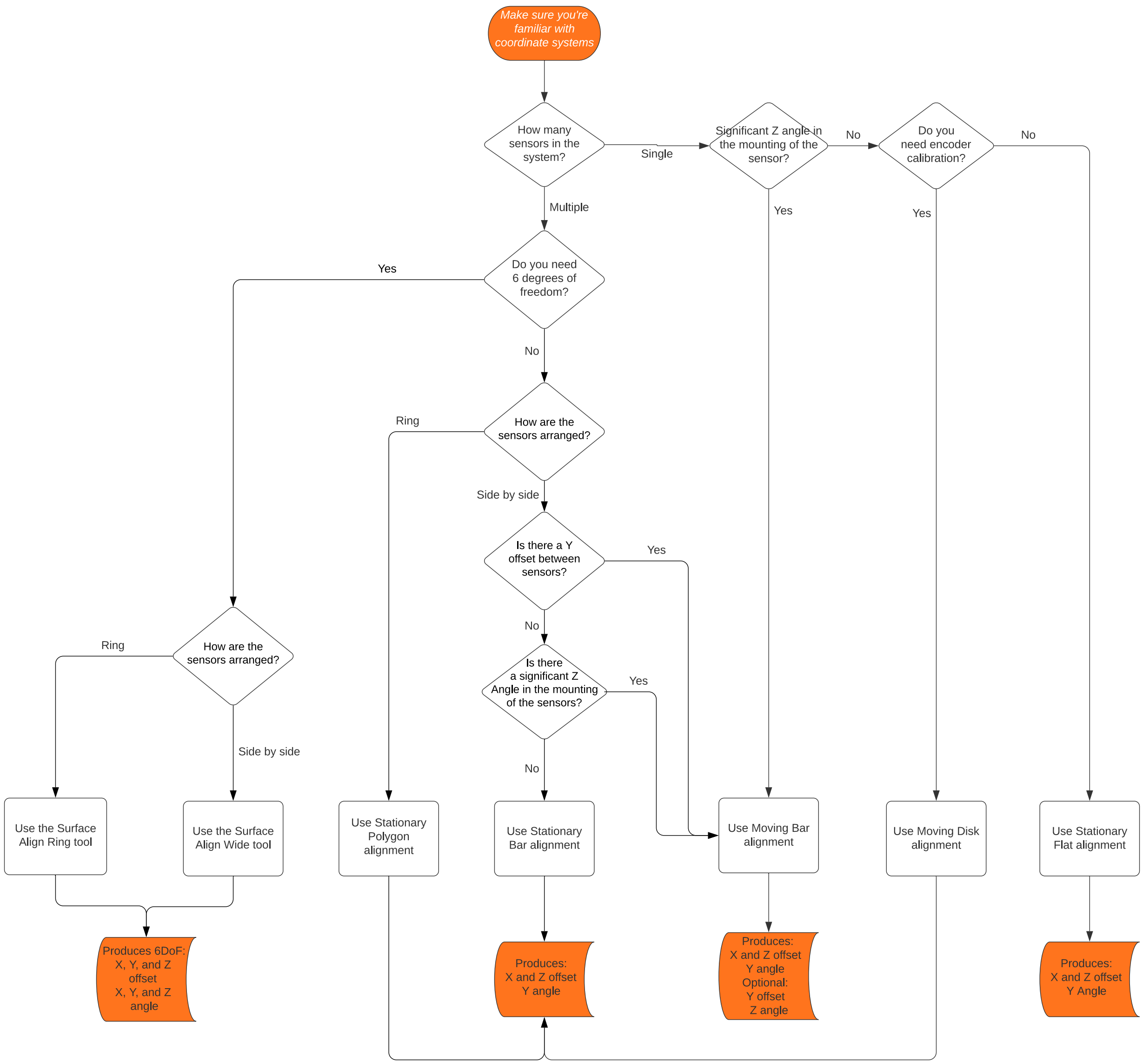
Most alignment methods use a special target that you must fabricate, either a bar with one or more holes, a polygon bar, or a target containing two or more truncated pyramids.

Use the following flowchart to help you decide which alignment method (alignment type and alignment target) to use, and then consult the appropriate sections for the target specifications and procedures relating to the chosen alignment type.

Before you begin, you should be familiar with the basics of coordinate systems and be able to understand concepts such as X / Y / Z offsets and X / Y / Z angles. To understand the transformations resulting from alignment that are then applied to scan data while a sensor is scanning objects in production, see *Coordinate Systems* on page 65.

⚠	Performing alignment using the Surface Align Ring or Surface Align Wide tools (which results in 6 degrees of freedom) involves *considerable* setup effort. First, the 6 DoF alignment targets are more difficult to manufacture than an alignment bar and require a very high degree of accuracy; 3D printed alignment targets are not usually sufficiently accurate. Second, the alignment tools have many parameters that must be properly configured to successfully perform an alignment.

	Disks are typically only used in demo systems.



Whether or not a given rotation or offset should be considered "significant" depends on factors such as your required tolerances. For more information, see *Planning Alignment* on page 134.

For alignment methods involving Surface Align Ring or Surface Align Wide, see *Aligning Sensors to 6 Degrees of Freedom* on page 157.

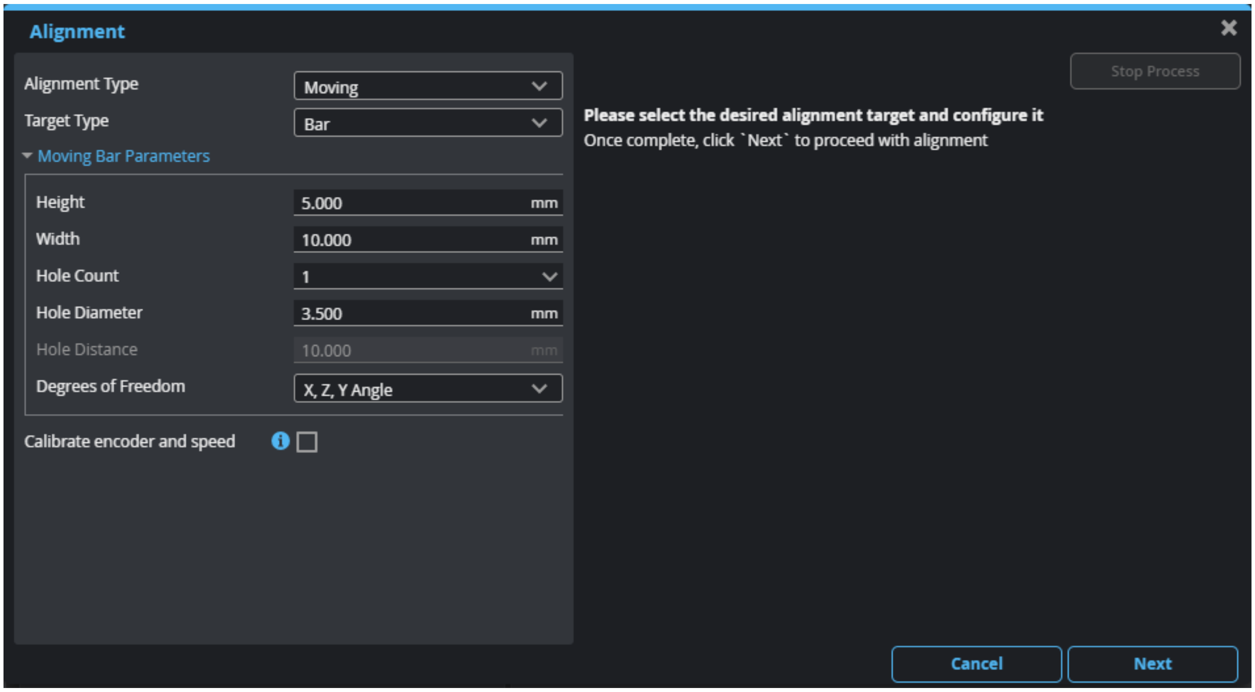
For all other alignment methods, see the appropriate subsection in *Aligning Sensors with up to 5 Degrees of Freedom* below.

Aligning Sensors with up to 5 Degrees of Freedom

Currently, GoPxL only supports up to 5 degrees of freedom using the alignment routines on the **System** > **Alignment** page.

To align single or multi-sensor systems with up to 5 degrees of freedom, you use the alignment routines on the **System** > **Alignment** page. Before proceeding, make sure that you have determined the alignment type (stationary or moving) and alignment target that you need for your system; for more information, see *Choosing an Alignment Method* on page 136.

For information on coordinate systems, see *Coordinate Systems* on page 65.

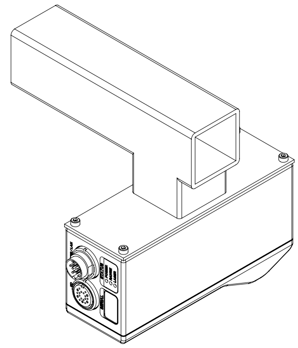
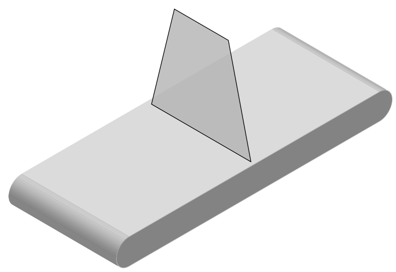


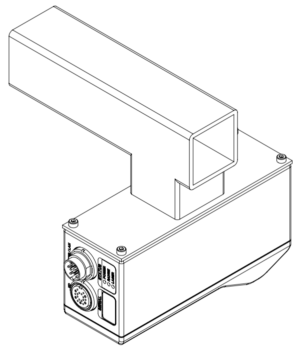
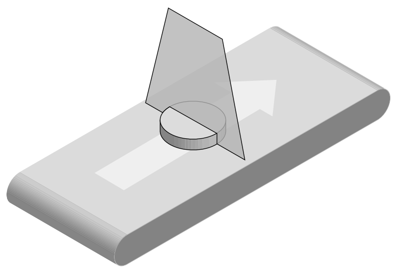
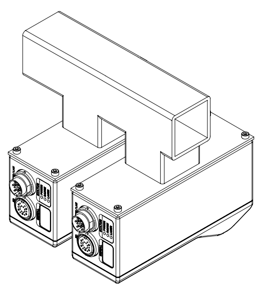
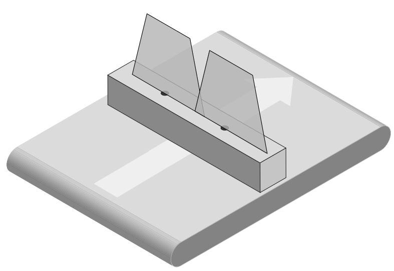
Alignment routine when Moving Bar type is selected

When using the alignment procedure on the **Alignment** page, you choose an alignment type (whether the target moves relative to the sensor) and an alignment target. You choose the combination of type and target based on the types of mounting inaccuracies (mostly minor rotations of the sensor around the X, Y, or Z axis relative to the scanning surface, but also intentional rotations in some situations (such as Y rotation, which is very common), and offsets of sensors in dual- or multi-sensor systems) you need to compensate for, or the reference plane you wish to set. Gocator will calculate different transformations depending on your choice.

Sensors support two types of alignment: stationary or moving.

Type	Description
Stationary	**Stationary** is used when the alignment target does not move during the alignment procedure. This type of alignment can only compensate for mounting position and orientation in the laser plane or light plane (Y angle and X and Z offsets).

Type	Description
	 

Type	Description
Moving	**Moving** is used when the alignment target moves beneath the sensor, for example, on a conveyor. This type allows for Y offset and Z angle alignment, in addition to X and Z offset and Y angle alignment.    

A sensor can be in one of two alignment states: Unaligned and Aligned. An indicator on the **Alignment** panel displays UNALIGNED or ALIGNED, depending on the sensor's state. A sensor's alignment state determines its coordinate system; for more information on coordinate systems, see *Coordinate Systems* on page 65.

> If you perform a high-accuracy tool-based sensor alignment, the **Alignment** panel will still display UNALIGNED. *This is normal.*

Alignment State

State	Explanation
Unaligned	The sensor or sensor system is not aligned. Data points are reported in sensor coordinates.
Aligned	The sensor is aligned using the alignment procedure (described below) or by manually modifying the values under **Sensor Transforms** on the **System** > **Alignment** page (for more information, see *Transformations* on page 156). Data points are reported in system coordinates.

Once you have performed the alignment procedure on the **Alignment** page, the calculated transformation values are displayed under **Transformations** in the **Sensor Transforms** panel on the **Alignment** page. For more information on transformations, see *Transformations* on page 156.

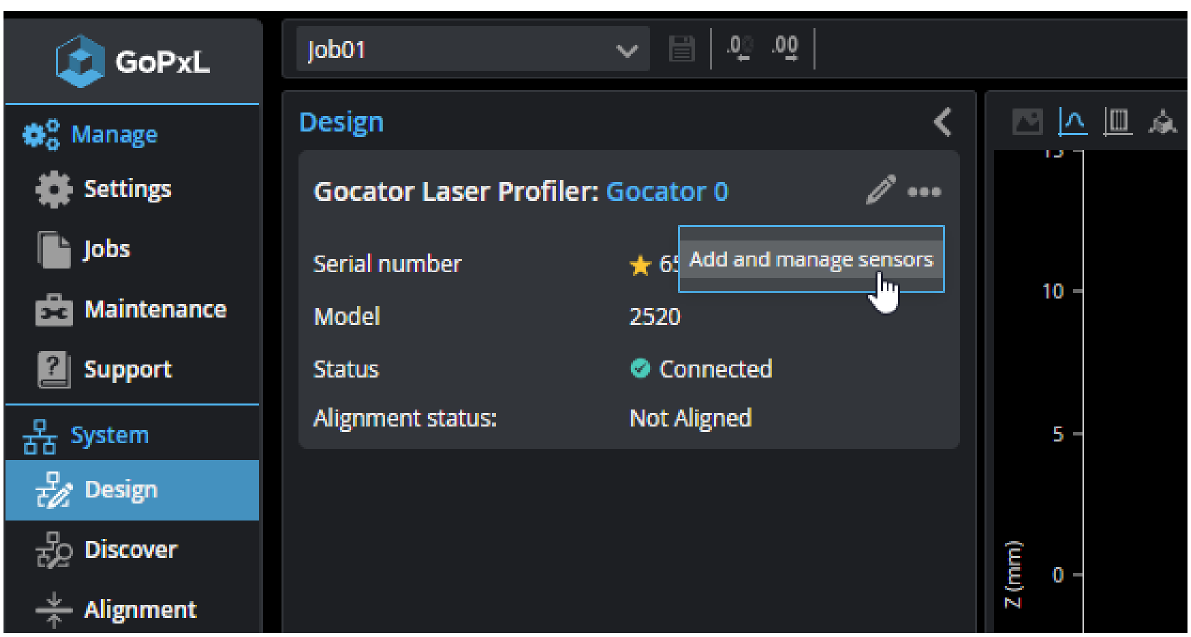
> If you perform a tool-based sensor alignment (using Surface Align Wide or Surface Align Ring) to achieve 6 degrees of freedom, the derived transformation values are *not* displayed under **Transformations** on the **Alignment** page. *This is normal.* (For information on performing tool-based alignment for 6 degrees of freedom, see *Aligning Sensors to 6 Degrees of Freedom* on page 157.)

With certain types of alignment, a **Degrees of Freedom** setting lets you choose the axes on which offsets and rotations are calculated. If the setting is not available, only X and Z offsets, and Y angle rotation, are calculated. That is, alignment is only performed within the profile plane. When the **Degrees of Freedom** setting is available, it generally provides options that let you perform alignment outside the profile plane.

To prepare for alignment

1. For dual- or multi-sensor systems, make sure you have done the following:

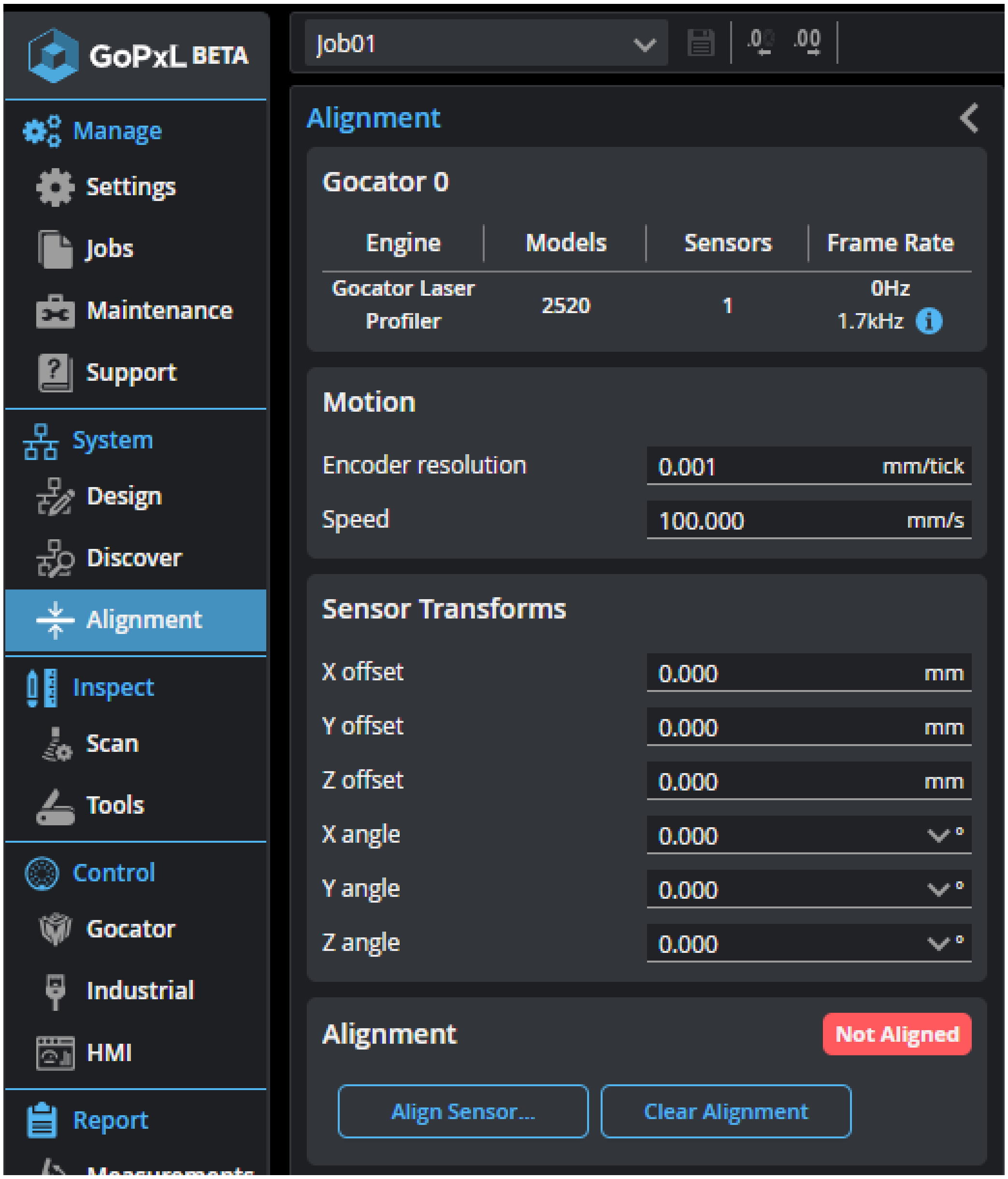
   On the **System** > **Design** page, add sensors to the system. For information, see *Adding Sensors and Configuring Multi-sensor Systems* on page 125.



Configure the system's layout, if necessary. For more information, see *Creating a Sensor System* on page 124 and *Changing Sensor Orientation* on page 128.

If the profiling lines of the sensors overlap, make sure to check the **Exposure Multiplexing** option. Otherwise, the profiling line from one sensor will be detected by other sensors and cause the alignment procedure to fail or be inaccurate; for more information, see *Enabling Exposure Multiplexing* on page 129.

2. Go to the **System** > **Alignment** page.



3. Ensure that all sensors will have a clear view of the alignment target surface.
4. Perform a preliminary scan of the alignment target to evaluate the quality of the scan data.

   Doing this will help ensure that the alignment process succeeds. In the next step, adjust the settings based on the scan data of the alignment target.

5. If necessary, on the **Inspect** > **Scan** page, adjust the sensor settings to get the best data possible from the scans of the alignment target.

   Some examples of the settings you may need to adjust are:

   - Exposure duration (to make sure the target is clearly represented in the scan data). Typically, only a single exposure is needed. For more information, see *Single Exposure* on page 220.
   - Active area. (For example, to exclude the ends of a bar alignment target). For more information, see *Active Area* on page 215.
   - Spacing: Make sure to use the sensor's full X resolution (sub-sampling is set to 1 and spacing interval is set to full resolution). For more information, see *Uniform Spacing* on page 197.

   Because the alignment procedure automatically uses Time triggering, regardless of the settings in the **Trigger** panel, you can leave these settings as is and configure them later. For information on triggering, see *Triggers* on page 206.

6. Based on the decisions made in *Choosing an Alignment Method* on page 136, do one of the following:

   - If you need to perform a stationary alignment, see *Performing Stationary Alignment* below.
   - If you need to perform a moving alignment, see *Performing Moving Alignment* on page 146.

Performing Stationary Alignment

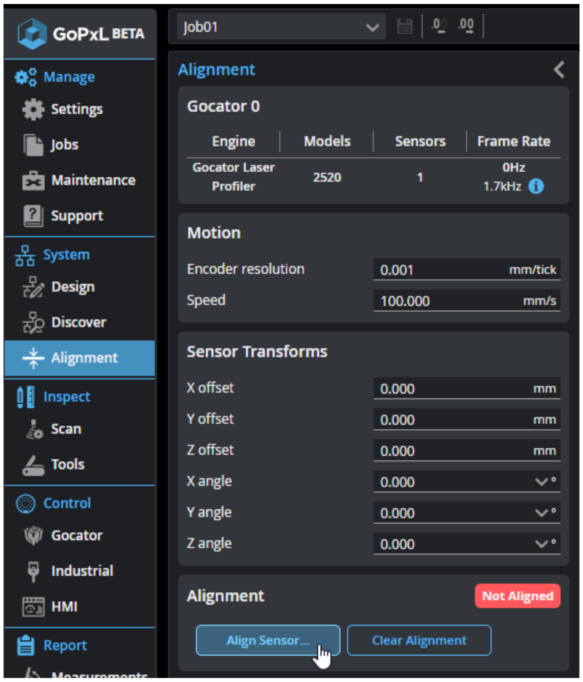
To perform stationary alignment

1. Make sure that the alignment surface (whether it's the surface of a conveyor or of an alignment target) is within the sensor's measurement range.

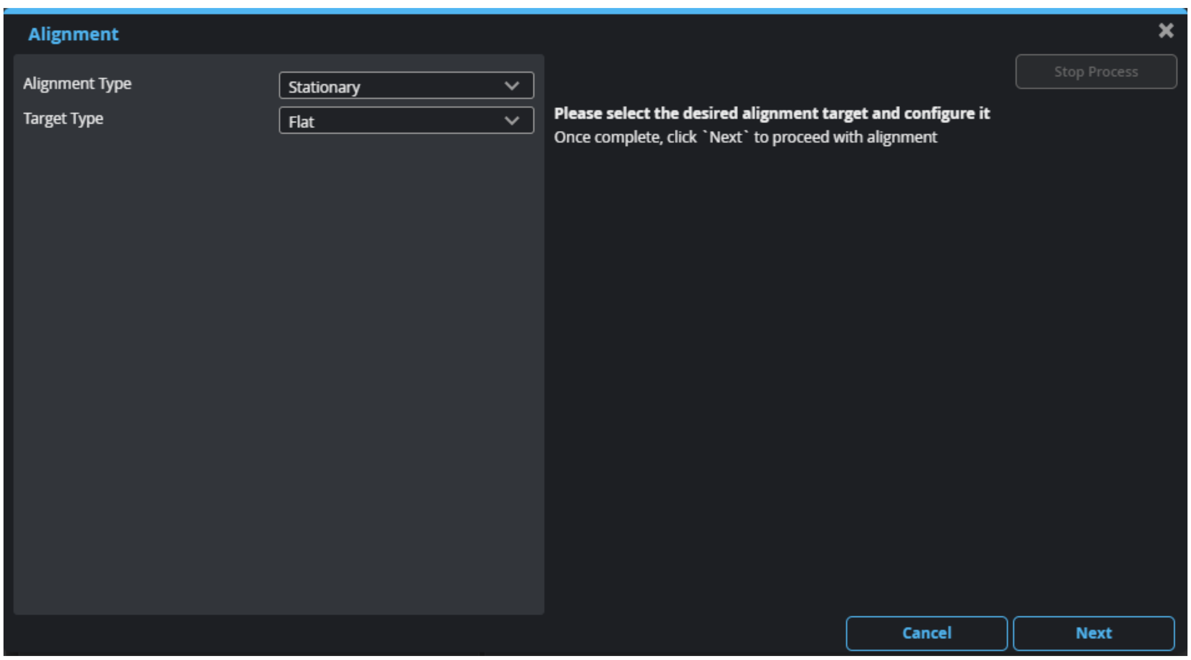
   To determine this, in the sensor's web interface, click **Start** and observe whether the Range LED on the sensor is illuminated. If you have set the trigger source to Encoder under **Inspect** > **Scan** > **Trigger**, temporarily set it to Time for this. Be sure to stop the sensor after confirming the target is in range by clicking the **Stop** button.

   Alternatively, you can determine the correct distance to the scan surface by consulting the sensor's measurement range specifications (see *Sensors* on page 975), and measuring the physical distance between the scan surface and the sensor.

2. Go to the **System** > **Alignment** page.
3. (Optional) If a previous alignment is present, click the **Clear Alignment** button.
4. If you are using an alignment target, place it below the sensor or sensors.
5. On the **System** > **Alignment** page, click **Align Sensors**.



The Alignment dialog appears.



6. Set **Alignment type** to Stationary.
7. Based on the decisions you made in *Choosing an Alignment Method* on page 136, choose the target type in **Target type**.
   - **Flat Surface**: Use this to align to a surface such as a conveyor. For more information, see *Stationary Flat Surface* on page 147.

- **Bar**: Use this to align to a bar alignment target. For information on alignment target requirements, bar-specific settings, and general setup tips, see *Stationary and Moving Bar* on page 148.
- **Polygon**: Use this to align a ring layout setup using a polygon shaped alignment target. For information on alignment target requirements, polygon-specific settings, see *Stationary Polygon* on page 153.

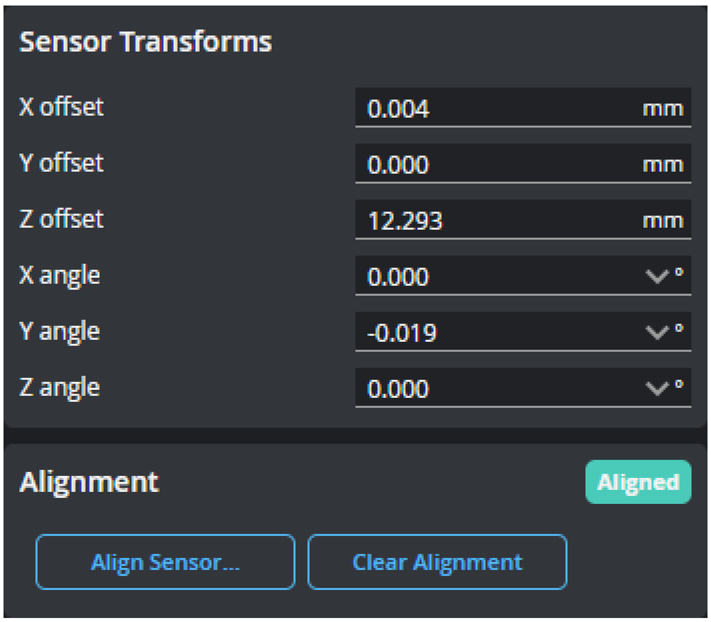
8. Click the **Next** button.

   If the alignment fails, check the settings described in *To prepare for alignment* on page 142 and repeat the steps described here.

9. Inspect alignment results.

   Data points from all sensors should now be aligned to the alignment target surface.

   Check the alignment results under **Sensor Transforms**.



   For information on how alignment affects the coordinate system used by sensors, see *Coordinate Systems* on page 65.

Performing Moving Alignment

To perform moving alignment

1. Go to the **System** > **Alignment** page.
2. (Optional) If a previous alignment is present, click the **Clear Alignment** button.
3. Place the target under the sensor.
4. Make sure that the surface of the alignment target is within the sensor's measurement range.

   You can determine the correct distance to the scan surface by consulting the sensor's measurement range specifications (see *Sensors* on page 975), and measuring the physical distance between the scan surface and the sensor.

5. Set **Alignment Type** to Moving.

6. Based on the decisions you made in *Choosing an Alignment Method* on page 136, choose the target type in **Target Type**.
   - **Disk**: Use this to align to a disk alignment target; disk alignment is typically only used for demos. For information on disk-specific settings, alignment target requirements, and general setup tips, see *Moving Disk* below.
   - **Bar**: Use this to align to a bar alignment target. For information on bar-specific settings, alignment target requirements, and general setup tips, see *Stationary and Moving Bar* on the next page.
7. (Optional) If you need to calibrate the transport system, check the **Calibrate encoder and speed** checkbox.

   The automatic encoder and speed calibration functionality is less accurate than manually specifying the transport system's encoder resolution or travel speed. You should only use this option if you have no other way of getting these values.

   If you do not use the built-in encoder or speed calibration functionality, make sure you have done one of the following:
   - If the transport system includes an encoder, make sure you have configured the encoder resolution.
   - If the transport system does *not* use an encoder (it is a time-based system), make sure you have configured travel speed.

   For more information on configuring motion settings, see *Configuring Motion* on page 131
8. Click the **Next** button.
9. Start the transport system.

   The sensors will start and then wait for the alignment target to pass through the laser plane or light plane. Alignment is performed simultaneously for all sensors. Alignment may take a minute or more.
10. Inspect alignment results.

    If the alignment fails, check the settings described in *To prepare for alignment* on page 142 and repeat the steps described here.

    Data points from all sensors should now be aligned to the alignment target surface.

    Check the alignment results under **Transformation** in the **Active Area** tab in the **Sensor** panel.

    For information on how alignment affects the coordinate system used by sensors, see *Coordinate Systems* on page 65.

Stationary Flat Surface

No additional settings are required for this alignment method. Note however that this type of alignment expects to receive flat scan data. Therefore, if the surface is curved, the alignment will be inaccurate. The surface should also be clear of debris and damage. The alignment results in 3 degrees of freedom (X and Z offset, and Y angle).

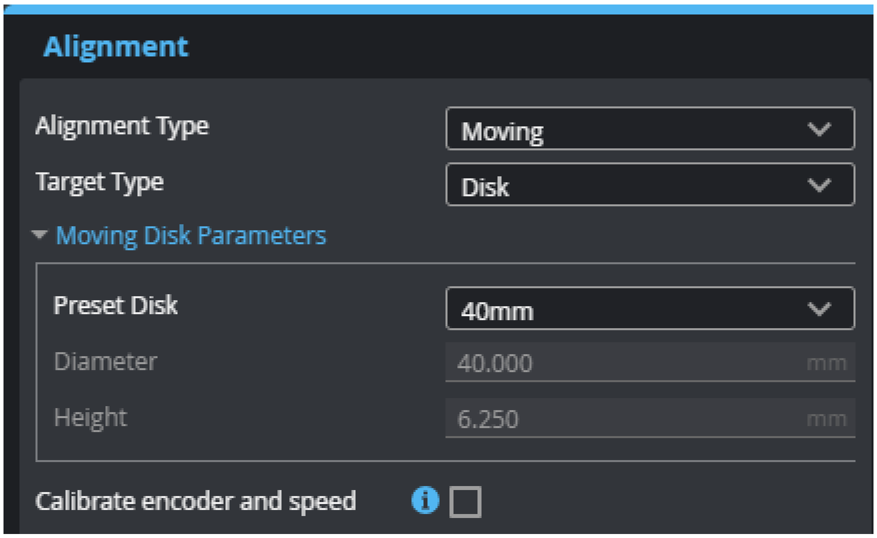
Moving Disk

	Disks are typically only used in demo systems.

Configure the characteristics of the disk target. If you have purchased a 40 mm or 100 mm disk from LMI, you can set their characteristics by choosing the appropriate entry in the **Preset Disk** drop-down. Otherwise, choose **Custom** and provide the dimensions manually.

Diameter defines the expected diameter of the disk.

Height defines the thickness of the disk in the Z direction. The alignment determines the average Z height of the disk's top surface. This height value is used to offset the coordinate system so that the bottom of the alignment disk becomes the Z origin.



Stationary and Moving Bar

For information on bar specifications and procedural requirements, see *Bar Specifications and Procedural Requirements* below.

For information on configuring GoPxL for bar alignment, see *Configuring GoPxL for Bar Alignment* on page 151.

On sensors aligned using Z angle (or sensors with a manually set X angle), and to a lesser extent Y offset, CPU usage increases when scanning, which reduces the maximum scan speed.

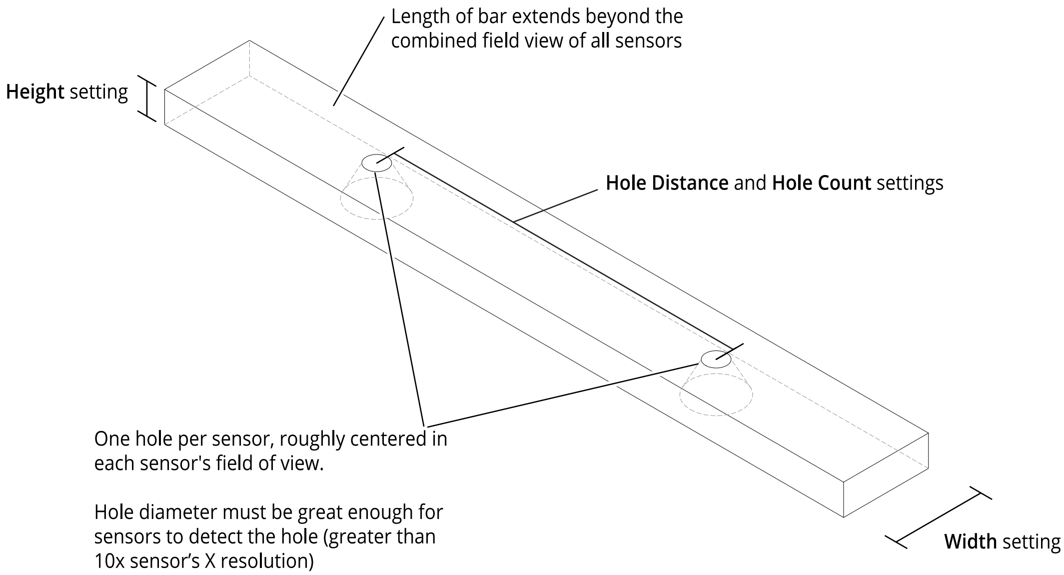
Artifacts may appear in scan data on sensors aligned using Z angle or X angle if encoder trigger spacing is set too high (resulting in a low sampling rate). For more information, see *Trigger Settings* on page 210.

Bar Specifications and Procedural Requirements

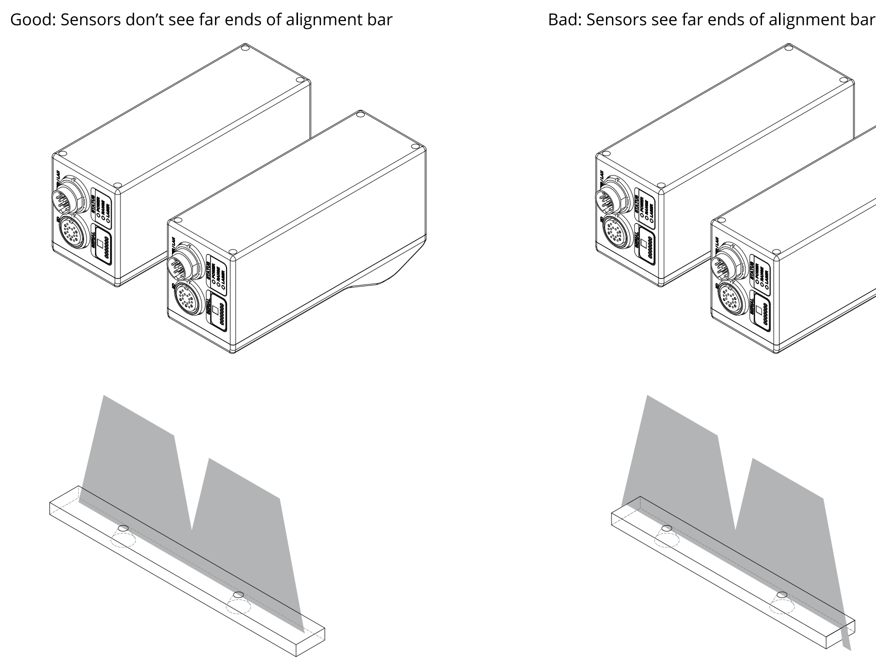
See the following sections for bar specifications and procedural requirements (stationary or moving alignment).

Bar Specifications

Ensure the following:

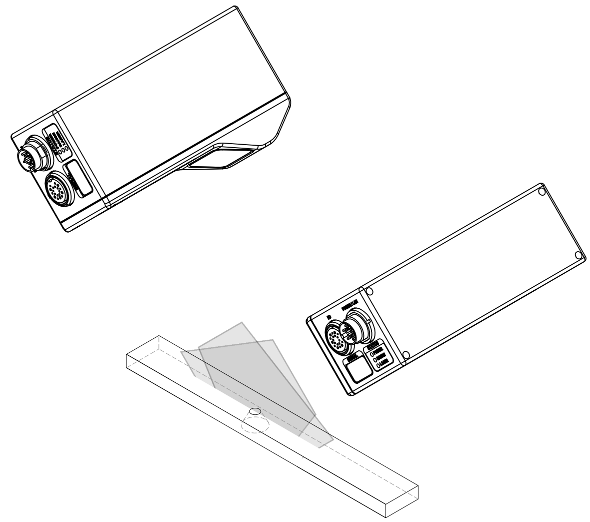


- The bar must extend beyond the outer ends of any profiling line: sensors must not "see" the left or right end of the bar (relative to the direction of travel of the transport system). Alternatively, you can set the active area of sensors that can "see" the ends of the bar to exclude the ends from the scan data; for more information, see *Active Area* on page 215. Otherwise, although the alignment should succeed, it will not be accurate: it may result in unwanted offsets or angles in the transformations.

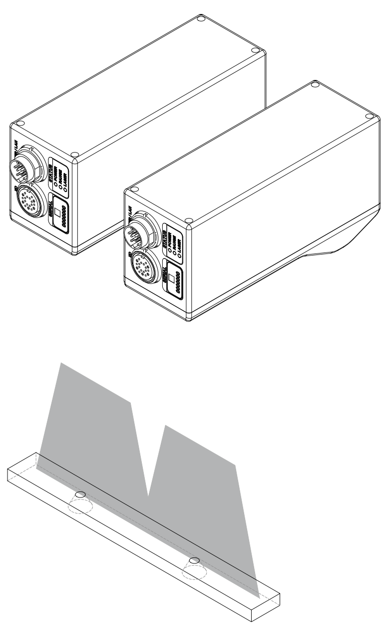


- If the sensor system contains two or more sensors side by side that are *not* intentionally angled toward each other around the Y axis (for example, to reduce occlusions), the bar should have one hole per sensor. Hole spacing should roughly correspond to the distance between the center of the FOVs of the mounted sensors, and holes should be equidistant. Although alignment can be performed if a sensor sees more than one hole (for example, if the profiling lines overlap enough), only the hole nearest to the center of a sensor's FOV is used for that sensor's alignment.

- If the sensor system contains two or more sensors side by side that *are* angled toward each other around the Y axis, a single hole should be used.



- Holes and bar edges must be as sharp as possible: avoid bevels.
- The size of the holes should be more than 10 times the X resolution of the sensor; for the X resolution of your sensor, see specifications of the sensor in *Sensors* on page 975.



- Sensors must capture as little data from the inside of a hole as possible. Either countersink holes from the opposite side of the bar (if no sensors are positioned on the opposite side of the hole in a "Bottom" position), or paint the insides of the holes with a flat black paint. Otherwise, although the alignment should succeed, it will not be as accurate: it may result in unwanted offsets or angles in the transformations.

- The recommended flatness of bar targets for accurate Y angle is roughly the Z resolution rating of the sensor. If the bar target is curved, it will introduce an apparent Y angle in the sensor alignment. For sensor Z resolution, see the specifications for your sensor in *Sensors* on page 975.
- It is not necessary to machine the bar height to a high tolerance. Bar height can instead be controlled during measuring rather than at manufacture. Only flatness and parallelism are important. If the zero level is not critical for the measurement, then standard machining tolerances can be used. Alternatively you can machine to a low tolerance and measure the value to a high precision to save cost.
- Bar width (the dimension along Y, that is, the direction of travel) is used to calibrate the encoder resolution or the travel speed, and is unrelated to Y offset in dual- or multi-sensor systems.
- Bars should be painted with flat light grey or white paint to improve data capture (by reducing the possibility of reflections and improving profile data of the bar surface). Doing this also allows you to reduce the exposure to further reduce the possibility of sensors seeing the interior of a hole. Note that when performing alignment, typically, sensors only need a Single exposure, regardless of whether sensors are going to be configured to use Dynamic or Multiple exposure when scanning in production. For more information on exposure, see *Exposure* on page 219.

Stationary Bar: Visibility of holes and bar

The hole closest to the center of each sensor's field of view is used for the alignment procedure.

Each profiling line must cross the center of a hole.

To do this:

1. Advance or back up the transport system until the sensor profiling line falls on the center of the hole.
2. Continue with step 1 in *To perform stationary alignment* on page 144.

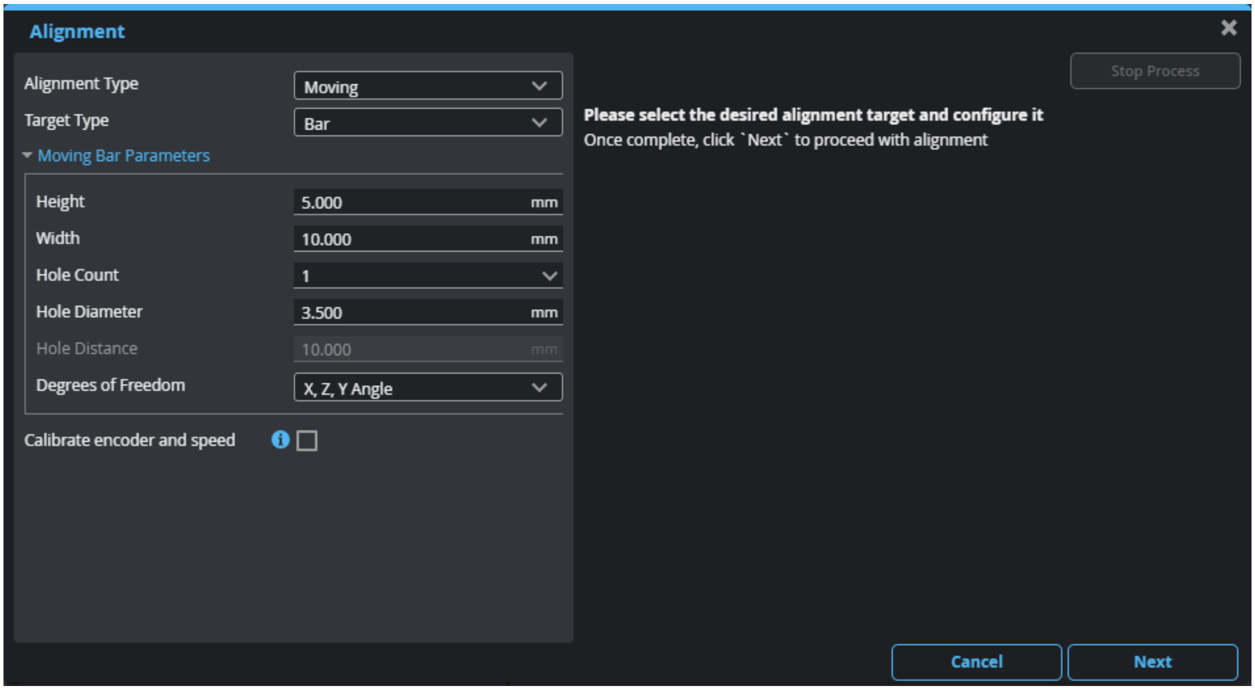
Moving Bar: Visibility of holes and bar

No other edges than the long edges of the bar should be visible during the alignment procedure: if sensors capture data from a conveyor or other structural component, or even debris, edges from these items may be misinterpreted as bar edges, and alignment will result in a false Y offset. Adjust the active area of sensors that see any of these items to prevent them from affecting the alignment; for more information, see *Active Area* on page 215.

Sensors may either see both the bar surface and the surface the bar is on, or only the bar surface (that is, if the supporting surface is beyond the sensor's measurement range): this has no impact on the alignment procedure.

Configuring GoPxL for Bar Alignment

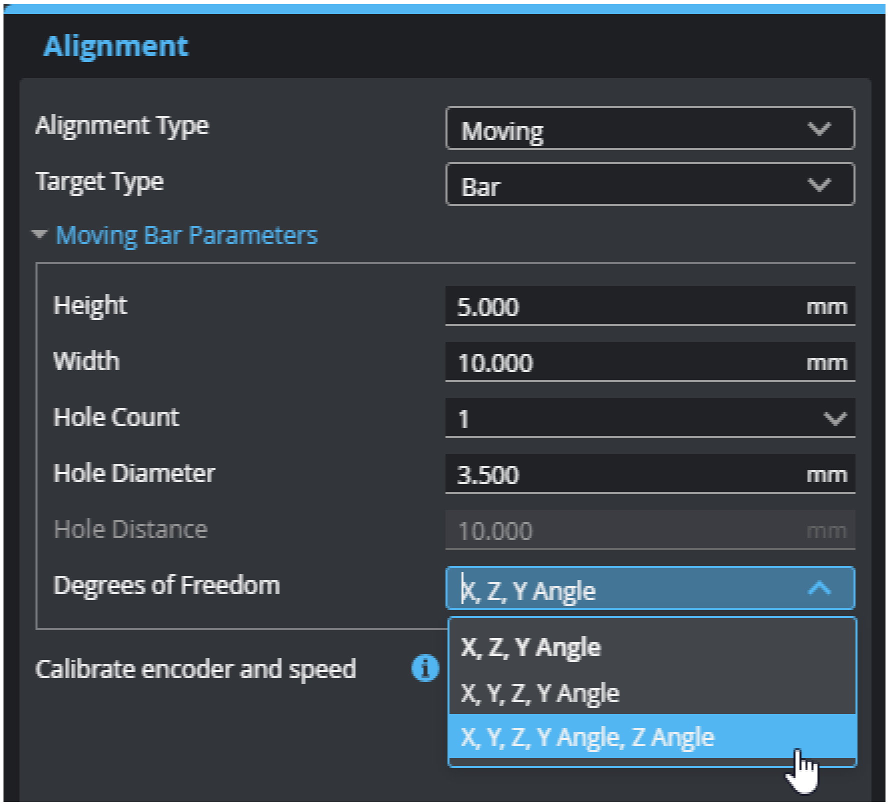
Configure the characteristics of the target (bar dimensions and reference hole layout); for more information on these settings, see below.



For an illustration of the various settings, see *Bar Specifications and Procedural Requirements* on page 148.

- **Height**: The alignment procedure determines the average Z height of the alignment bar's top surface and uses the value specified in **Height** to offset the coordinate system from that average Z height. In effect, the bottom of the alignment target becomes the Z origin (the zero reference level).
- **Width**: Only displayed if you choose a Moving Bar alignment. Sets the width of the bar in the Y direction. The bar width is only used to calibrate encoder resolution and travel speed in conjunction with the **Encoder or Speed Calibration** setting; the width is unrelated to any Y offset between sensors in dual- or multi-sensor system (for more information, see *Encoder Calibration* on page 155). A width of 100 mm is typical.
- **Hole Count**: The number of holes in the bar. For important information on the number of holes you should choose, see *Bar Specifications and Procedural Requirements* on page 148.
- **Hole Diameter**: The diameter of the holes.
- **Hole Distance**: The distance between the centers of the holes. This measurement is critical: you should measure this distance to within the sensor's X resolution. If your bar only has two holes, you can machine the bar to a lower tolerance, measure the true spacing, and use this value for the parameter. But if your bar has more than two holes, you must machine the holes accurately to ensure that the spacing between the holes is the same.
- Under **Degrees of Freedom**, different combinations of offsets (X, Y, and Z) and rotations (Y Angle

and Z Angle) are available.



Stationary Polygon

Polygon target alignment is typically used when you need to scan 360 degrees around a target. A polygon target can also be used with an "arc" of sensors.

Polygon Target Specifications

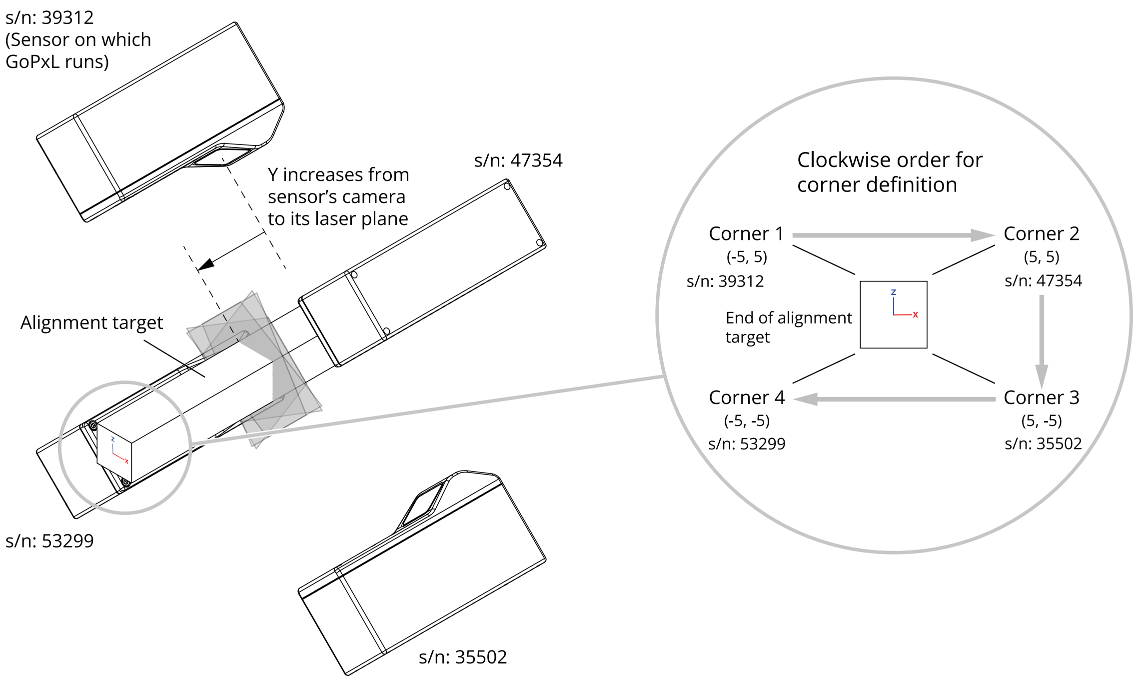
Ensure the following:

- The target must have at least one corner per sensor.
- Corners must have sharp edges.
- The surface adjacent to the corners must be flat.
- Targets should be painted with flat light grey or white paint to improve data capture (by reducing the possibility of reflections and improving profile data of the bar surface).
- Each sensor must clearly see a corner of the polygon target.

Configuring GoPxL for Polygon Alignment

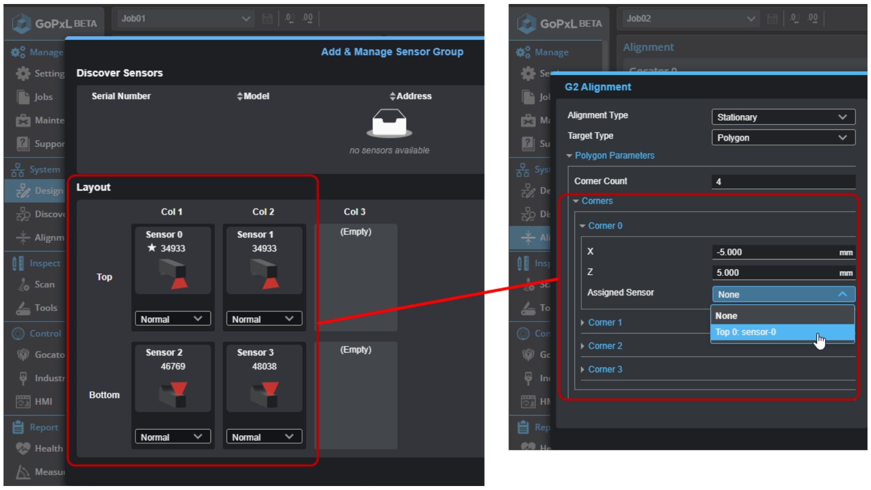
To perform polygon target alignment, you must set the X and Z coordinates of each corner of the alignment target. The coordinates are relative to the *target itself*, and you typically set them such that the X and Z origins are at the *center* of the target.

To properly configure the X and Z values of each corner of the alignment target (and assign sensors to the corners), you must view the sensors and alignment target so that *Y increases toward you*. To determine how to view the sensors and target, refer to the coordinate system orientation information for your sensor model in *Sensors* on page 975, or remember that Y increases moving from the camera to the light emitter. (If any sensors are defined as **Reversed**, use only the non-reversed sensors to determine how to view the sensors; for more information on layout, see *Creating a Sensor System* on page 124.) Starting with the sensor on which GoPxL will run and control the other sensors, for each corner, define the X and Z coordinates and assign the sensor that is viewing that corner, *proceeding in a clockwise order*. You can start with any corner.



Simplified representations of sensors. When looking at the end of the alignment target and non-reversed sensors, Y must increase toward you. In the illustration, an alignment target measuring 10 mm on each side is represented. Therefore, X and Z coordinates are + or - 5 mm.

Refer to the positions of the sensors in the layout grid in the **Layout** section in the **Manage Sensor Group** dialog to help populate the fields for polygon alignment.



You are not required to assign a sensor to every corner.

Encoder Calibration

For systems that use an encoder, encoder calibration can be performed while aligning sensors. The table below summarizes the differences between performing alignment with and without encoder calibration.

The automatic encoder and speed calibration functionality is less accurate than manually specifying the transport system's encoder resolution or travel speed. You should only use this option if you have no other way of getting these values.

	With encoder calibration	Without encoder calibration
Target Type	Calibration disk or calibration bar	Flat surface or calibration bar
Target/Sensor Motion	Linear motion	Stationary
Calibrates Tilt	Yes	Yes
Calibrates Z axis Offset	Yes	Yes
Calibrates X axis Offset	Yes	Yes (Calibration bar required)
Calibrates Encoder	Yes	No
Calibrates Travel Speed	Yes	No

See *Coordinate Systems* on page 65 for definitions of coordinate axes. For descriptions of disks and bars, as well as alignment procedures, see the appropriate sections in *Aligning Sensors with up to 5 Degrees of Freedom* on page 138.

After alignment, the coordinate system for profiles will change from sensor coordinates to system coordinates.

Clearing Alignment

Alignment can be cleared to revert the sensor to sensor coordinates.

To clear alignment:

1. Go to the **System** > **Alignment** page.
2. Click the **Clear Alignment** button.
   GoPxL clears the alignment. Sensors revert to using sensor coordinates instead of aligned system coordinates.

Transformations

The transformation values determine how data is converted from unaligned coordinates to aligned coordinates (for an overview on coordinate systems, see *Coordinate Systems* on page 65). Transformation values are set automatically when you align a sensor using an alignment routine on the **System** > **Alignment** page.

If you perform an alignment using the Surface Align Wide or Surface Align Ring tools, these values are *not* updated. For more information, see *Aligning Sensors to 6 Degrees of Freedom* on the next page.

Parameter	Description
X Offset	Specifies the shift along the X axis. With Normal orientation, a positive value shifts the data to the right. With Reverse orientation, a positive value shifts the data to the left.
Y Offset	Specifies the shift along the Y axis.
Z Offset	Specifies the shift along the Z axis. A positive value shifts the data toward the sensor.
X Angle	Specifies the tilt around the X axis. This creates a skew clockwise around the X axis (pointing toward the viewer).
Y Angle	Specifies the tilt around the Y axis. This rotates profiles counter-clockwise around the Y axis (pointing toward the viewer).
Z Angle	Specifies the tilt around the Z axis. This creates a skew clockwise around the Z axis (pointing toward the viewer).

When applying the transformations, the data is first rotated around X (clockwise, with the X axis toward the viewer), then Y (counterclockwise), and then Z (clockwise), and then the offsets are applied.

Setting **Angle X** or **Angle Z**, and to a lesser extent **Y Offset**, to a non-zero value increases CPU usage when scanning, which reduces the maximum scan speed.

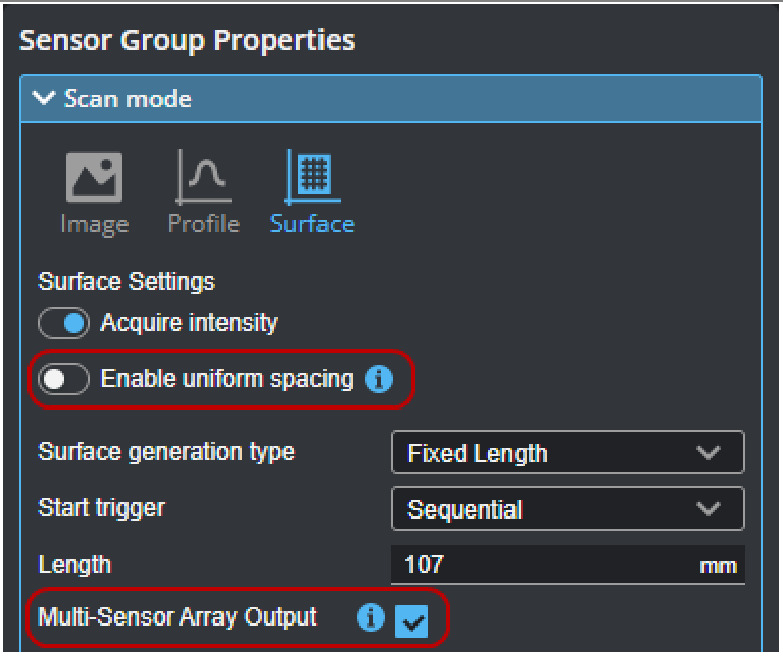
Artifacts may appear in scan data when **Angle Z** or **Angle X** is set to a non-zero value if encoder trigger spacing is set too high (resulting in a low sampling rate). For more information on trigger spacing, see *Trigger Settings* on page 210.

Aligning Sensors to 6 Degrees of Freedom

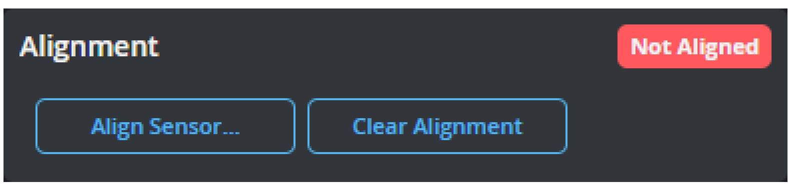
The tools described in the following sections are *only* intended to be used with G2 sensors.

Performing alignment using the Surface Align Ring or Surface Align Wide tools (which results in 6 degrees of freedom) involves *considerable* setup effort. First, the 6 DoF alignment targets are more difficult to manufacture than an alignment bar and require a very high degree of accuracy; 3D printed alignment targets are not usually sufficiently accurate. Second, the alignment tools have many parameters that must be properly configured to successfully perform an alignment.

Before scanning the alignment target, remember to uncheck **Enable uniform spacing** in the **Scan mode** section on the **Scan** page. Also enable **Multi-Sensor Array Output**.

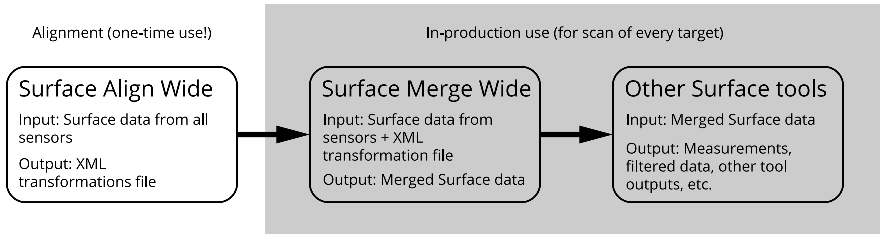


Sensors must not previously have been aligned using the alignment routine on the **System** > **Alignment** page. This is indicated by "Not Aligned" on that page. If necessary, click **Clear Alignment**.

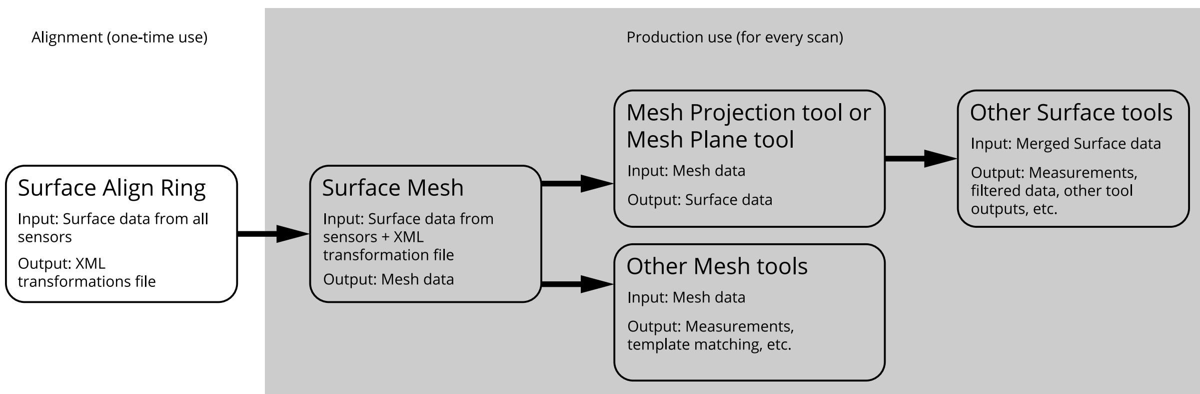


Aligning a system of sensors to 6 degrees of freedom involves the use of one of two Surface measurement tools (Surface Align Wide or Surface Align Ring). These tools create a set of transformations and store them in a configuration file. The resulting alignment is more accurate compared to the other methods available on the **Alignment** panel, and includes compensations for X angle rotations. Note that in order to apply the transformations to scan data, you must use a "stitching" tool that corresponds to the tool used to create the transformations.

- Surface Align Wide: Use this tool if the sensors are in a wide (that is, side-by-side) layout. Sensors may be slightly angled on the Y axis, that is, in an arc above the target. Sensors must be on the same side as the target: no data is supported on the other side. The tool is designed for up to eight sensors. The tool aligns to a multi-column truncated pyramid plate alignment target to produce the transformations necessary to stitch scans of production targets from individual sensors into a single frame of Surface scan data. In a single-sensor system, you can also use the tool to compensate for X angle rotation. (Note that in a single-sensor system, Y offset is not calculated or used.) For more information, see *Wide Layouts (Surface Align Wide Tool)* on page 160. The workflow / data flow is as follows:



- Surface Align Ring: Use this tool if the sensors in a multi-sensor system are in a ring or partial ring layout. The tool aligns to a double-sided truncated pyramid alignment target to produce the transformations necessary to stitch scans of production targets into a single frame of Mesh scan data. For more information on performing this type of alignment, see *Ring Layouts (Surface Align Ring Tool)* on page 182. The workflow / data flow is as follows:



.

Both tools produce configuration files, and can optionally load previously saved configuration files, which is useful if you need to set the system up again. When you run the tools on a PC, the tools store the configuration files in `C:\GoTools\SurfaceAlign\`. On a GoMax device, the tools store the configuration files on the device.

Wide Layouts (Surface Align Wide Tool)

The Surface Align Wide tool performs a high-accuracy alignment of multi-sensor laser line profiler systems in wide layouts (side-by-side). The tool is *only* intended for use with these sensors and system layouts.

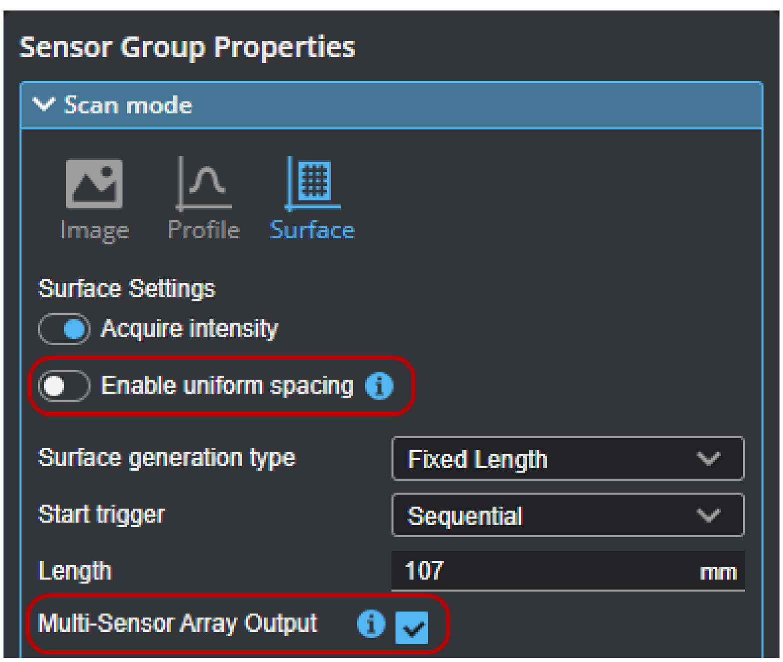
Before using this tool, you must fabricate a special pyramid plate alignment target (for more information, see *Alignment Target and Setup* on page 162), and then scan this target, recording the data for the alignment procedure with this tool.

⚠	Performing alignment using the Surface Align Ring or Surface Align Wide tools (which results in 6 degrees of freedom) involves *considerable* setup effort. First, the 6 DoF alignment targets are more difficult to manufacture than an alignment bar and require a very high degree of accuracy; 3D printed alignment targets are not usually sufficiently accurate. Second, the alignment tools have many parameters that must be properly configured to successfully perform an alignment.

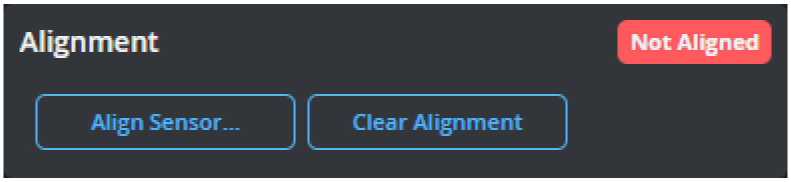
Unlike alignment using the **Alignment** panel on the **Scan** page, the Surface Align Wide determines the X angle rotation, giving you a full six degrees of freedom (for information on coordinate systems, see *Profile Output* on page 65). This method of alignment will produce higher accuracy scans, and allows for higher scan rates, due to the use of a different data-merging algorithm.

⚠	This tool requires acceleration (either by a PC-based application or by GoMax). For more information, see *Running GoPxL on a Windows PC* on page 828.

⚠ Before scanning the alignment target, remember to uncheck **Enable uniform spacing** in the **Scan mode** section on the **Scan** page. Also enable **Multi-Sensor Array Output**.



Sensors must not previously have been aligned using the alignment routine on the **System** > **Alignment** page. This is indicated by "Not Aligned" on that page. If necessary, click **Clear Alignment**.



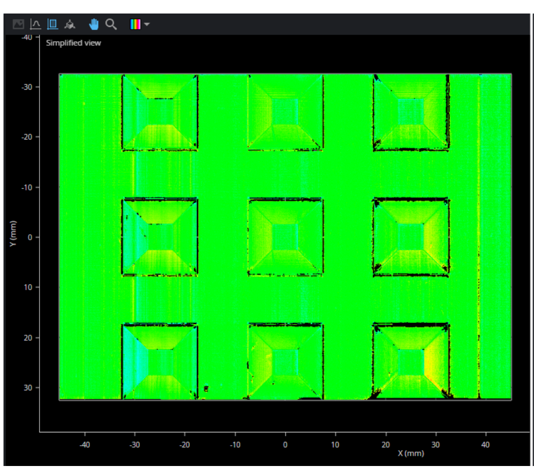
After configuring the alignment tool (see the parameters below), you must check the **Enable Processing** checkbox to start processing of the alignment target scan data. After the tool has finished processing the data and has successfully aligned the sensors, it produces the XML file you load into a Surface Merge Wide tool, as well as a Difference Surface data output that you can use to assess the quality of the alignment. (The tool actually creates two copies of the initialization file: one called Configuration.xml, and another with a date-time stamp in its name.)

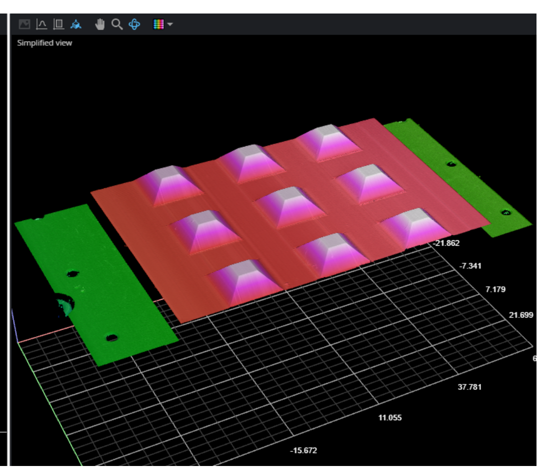
During processing, the tool segments the planes and then performs a best fit of the scanned target to the nominal dimensions you provide in the tool. For this reason, an accurately manufactured alignment target, correct nominal dimensions, and scan data that only includes the target are crucial to success. For more requirements and guidelines, see *Alignment Target and Setup* on the next page.

To determine whether the alignment succeeds and to assess the quality of the alignment, you use the Standard Deviation measurement output and the Difference Surface output.

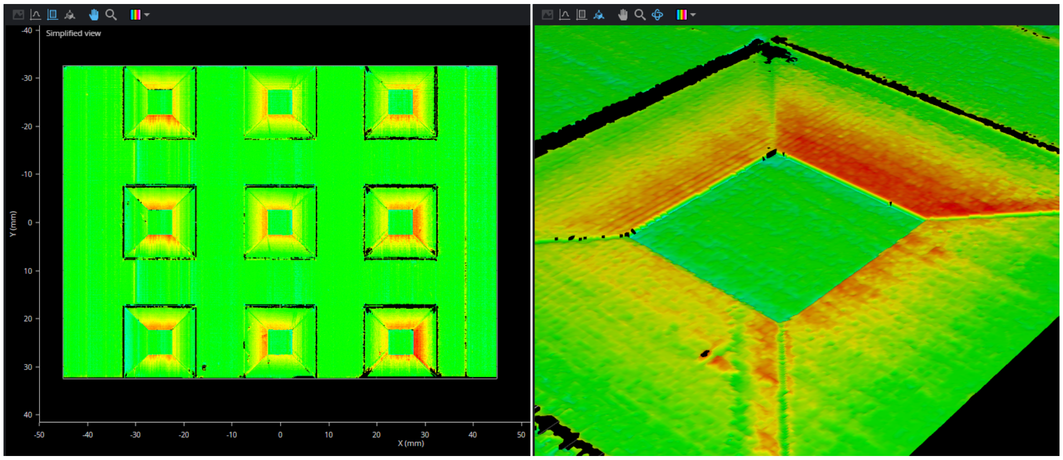
If the Standard Deviation measurement returns Invalid, the alignment has failed. If Standard Deviation returns a numerical value, it represents the "uncertainty" of the alignment process (basically, a numerical assessment of the quality of the alignment). This value should be close to zero.

The Difference Surface output represents discrepancies between the expected, nominal values provided in the tool (the dimensions of the alignment target, as well as the number and positions of the sensors) and the scan of the alignment target. In a "good" Difference Surface output, most of the data points should be relatively close to zero, for example on the left below. (The image on the right is a full, merged scan of the target produced by Surface Merge Wide, to give context.)



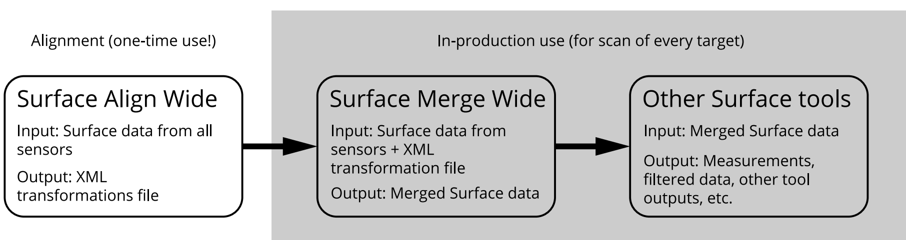


Discrepancies are indicated as height differences above or below zero, that is, how "different" the nominal values and the actual scan data are. In the following images, a discrepancy of -0.15 between the target's top width and the scan data shows up as a "ring" around the top of the pyramid.



For more information on how to interpret the Difference Surface, see *Understanding the Difference Surface* on page 176.

It's important to remember that in order to perform scans and measurements in production using a system aligned using this tool, you *must* include a Surface Merge Wide tool (see *Surface Merge Wide* on page 575) in your scanning job's tool chain, and load the XML initialization file created by this tool into the merging tool: The merging tool uses the transformations in the initialization file to combine the scans from the individual sensors into single frames of Surface scan data. You can then apply any built-in tools to the processed data that Surface Merge Wide produces; for information on the tools, see *Tools - Measurement and Processing* on page 232.

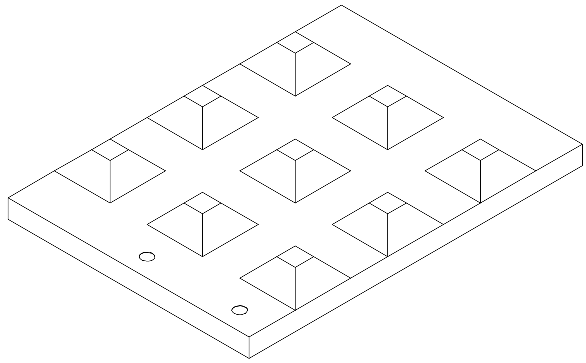


Note that after aligning using this tool, on the **Alignment** panel on the **Scan** page, GoPxL indicates that the sensors are *unaligned*. This is normal.

Alignment Target and Setup

This Surface Align Wide tool requires the use of a pyramid plate alignment target, which consists of rows and columns of truncated pyramid forms. You can find CAD files for an example 3x3 pyramid target under `Tools\Alignment CAD\Pyramid Plate` in the Utilities package (e.g., 14405-x.x.xx.xx_SOFTWARE_GO_Utilities.zip, available on LMI's Product Downloads page). Note that you must adapt the size of the plate and the pyramids to the field of view (FOV) of the sensors in your system: generally speaking, the plate should be scaled so that a truncated pyramid fills most of a sensor's FOV. For details, see the list of requirements below. For definitions of the dimensions of the target as used in the tool, see *Pyramid Plate Target Parameters* on page 172.

> The alignment target LMI provides in a CAD file is appropriate for sensors angled up to roughly 15° from vertical. For information on alignment target requirements, see below.



A 3x3 pyramid plate. Exact dimensions of the plate and the pyramids will depend on the sensors in your system.

The following target requirements must be satisfied for best results:

1. 3D printed targets are not recommended, as they may lack the required accuracy. Ideally, you should machine the alignment target.
2. Scale the target so that each sensor covers the area of only one column. However, if an application requires that sensors must be angled so that their laser planes mostly overlap, two sensors can see the same column.
3. Maximize the target size to ensure that as much of the base area is visible to the sensor. Keep in mind however that the target size should be smaller than the combined FOV.
4. Make sure the alignment target surface is not too shiny or too dark to be scanned. A diffuse surface is best.
5. Edges do not need to be perfectly sharp: The alignment tool performs a plane fit to points within the planar surfaces of the target and excludes data close to the edges.
6. Pyramid heights should be at least 1/5 of the measurement range of the sensor. For example, the measurement range of a Gocator 2520 is 25 mm; for this type of sensor, the pyramid height should be at least 5 mm. (For measurement range specifications, see *Sensors* on page 975.)
7. Pyramid tops should be at least 1/10 of the sensor's FOV, taking into account the location of the target along the Z axis in the sensor's measurement range. For example, at the midpoint in the measurement range of a Gocator 2520, the FOV is 28.75 mm; for this type of sensor, the pyramid tops should be at least 2.875 mm along each side.
8. The pyramid angle (the angle between the side planes and the base plane) must be greater than 30°.
9. There should be no holes in the flat area between sensors.
10. Only one row of pyramids is required for the alignment procedure, but using additional rows improves the accuracy of the alignment.

When mounting sensors over the alignment target and configuring the active area, keep the following in mind:

1. Pyramids should be roughly centered in the FOV of each sensor.
2. For best results, all four sides of a pyramid should be visible to a sensor.
3. Given the size of the top widths of the pyramids and the distance of the sensors to the alignment target, the top widths should be represented by a minimum of 100 to 200 data points along the X axis in the scan data.
4. The flat area between the pyramids must be visible to the sensors.
5. Pyramid sides should not be angled from a sensor's Z axis by more than 60 degrees. So for example, if you use an alignment target with 45° pyramid sides, a sensor's Z axis shouldn't be angled by much more than 15 degrees from vertical. And for sensors angled 30 degrees from vertical, for example, an alignment target whose pyramid sides are angled only 30 degrees would be more appropriate
6. Block any unwanted planar data during scanning, such as the conveyer belt beneath the target. To do this, you can shrink the sensor's active area in the **Acquire > Scan** page to exclude this data while scanning the target. Remember to reset the active area after the alignment. For more information on setting the active area, see *Active Area* on page 215.

After you have scanned the alignment target, verify in the scan data that each of the five planar segments of each pyramid (the top and the four sides) is represented by at least 100 x 100 data points. If the point count along the X axis is less than 100, one alternative way is to scan the target in a smaller **Y Spacing** to meet the 100 x 100 points per segment threshold.

Alignment Procedure

The following provides the steps involved in performing a high-accuracy, tool-based alignment.

> The following assumes you are scanning the alignment target using unaccelerated sensors and then uploading the recording to a PC instance of GoPxL. The procedure for doing this with accelerated sensors is similar.

To perform the alignment:

1. Fabricate an alignment target appropriate for your system (wide layout versus ring layout).

   For details on the alignment target, see *Alignment Target and Setup* on page 162.

2. If you have not already done so, set up and configure the multi-sensor system.

   The following sections describe installing, setting up, and configuring a system:

   Installation on page 36

   Network and Sensor Setup on page 50

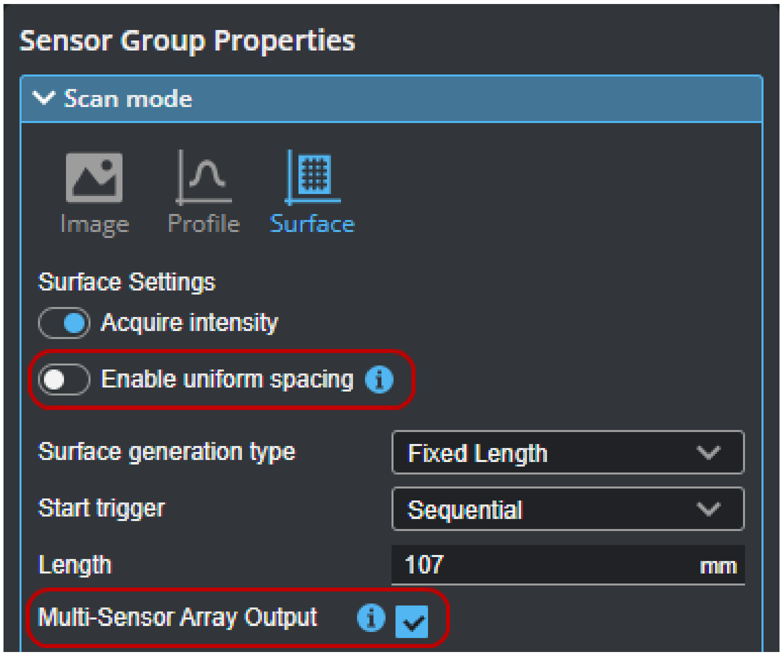
   Sensor Management and Maintenance on page 117

   Creating a Sensor System on page 124

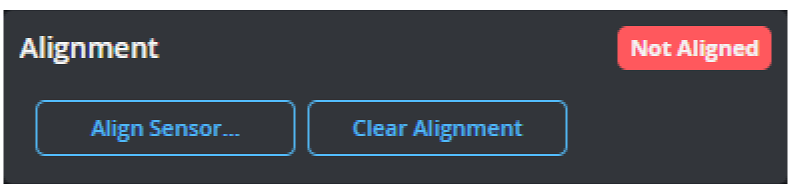
   Scan - Configuring Acquisition on page 195

3. Make sure to do the following.

   Disable **Uniform Spacing** and enable **Multi-Sensor Array Output** in the **Scan mode** section on the **Inspect** > **Scan** page.

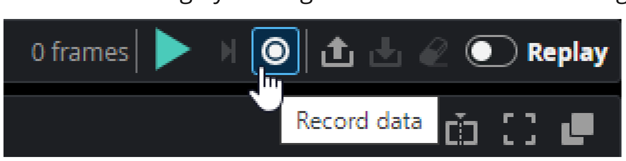
   Sensors must not previously have been aligned using the alignment routine on the **System** > **Alignment** page. This is indicated by "Not Aligned" on that page. If you see "Aligned" here, click **Clear Alignment**,

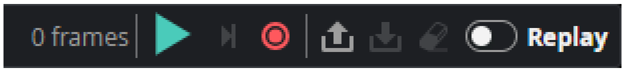
> Although you can scan the alignment target without acceleration, you *must* perform the alignment using PC-based acceleration (for more information, see *Running GoPxL on a Windows PC* on page 828). Because starting acceleration after having performed a scan clears scan data from a sensor, if you are going to perform alignment on-sensor, you should start acceleration before continuing.
>
> You can also optionally download the scan data and perform the alignment on the scanned target using GoPxL on Windows. For more information, see *Working with Scan Data (Toolbar)* on page 112.

4. If any sensors are installed in a "reversed" orientation (that is, such that the conveyor moves in the same direction as the sensor's positive Y axis), be sure to set the sensor's layout to Reverse.

5. Enable recording by clicking the Record button to the right above the data viewer.

GoPxL shows recording is enabled.



6. Start the transport system and then perform a scan of the alignment target.
7. Turn off recording.

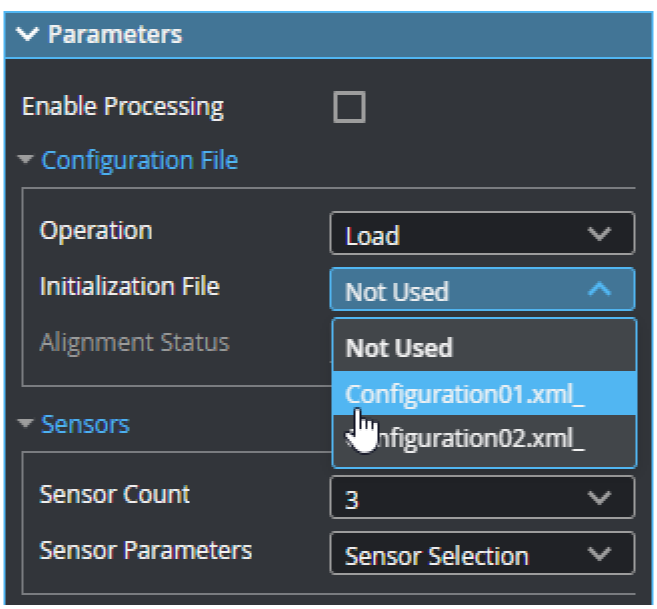
8. Download the recording and make note of the downloaded file's location.

9. Launch a PC instance of GoPxL.

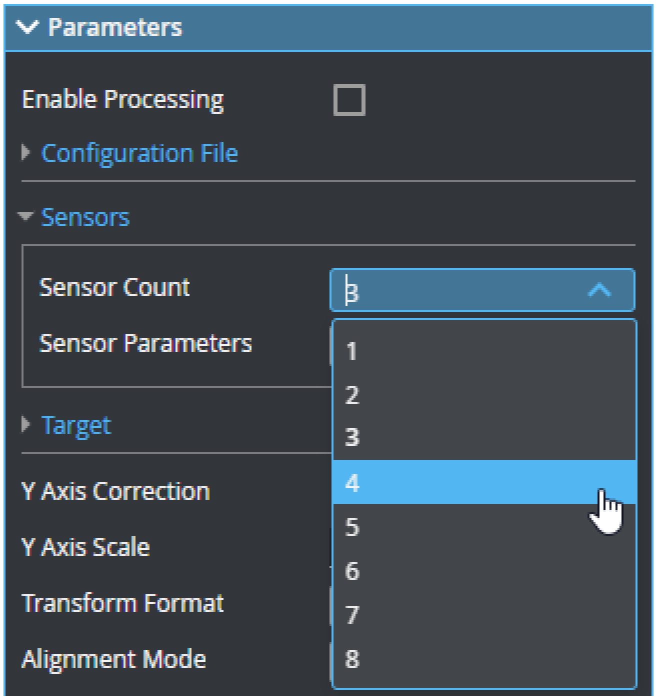
   For more information, see *Running GoPxL on a Windows PC* on page 828.

10. Upload the recording of the alignment target in the PC instance.

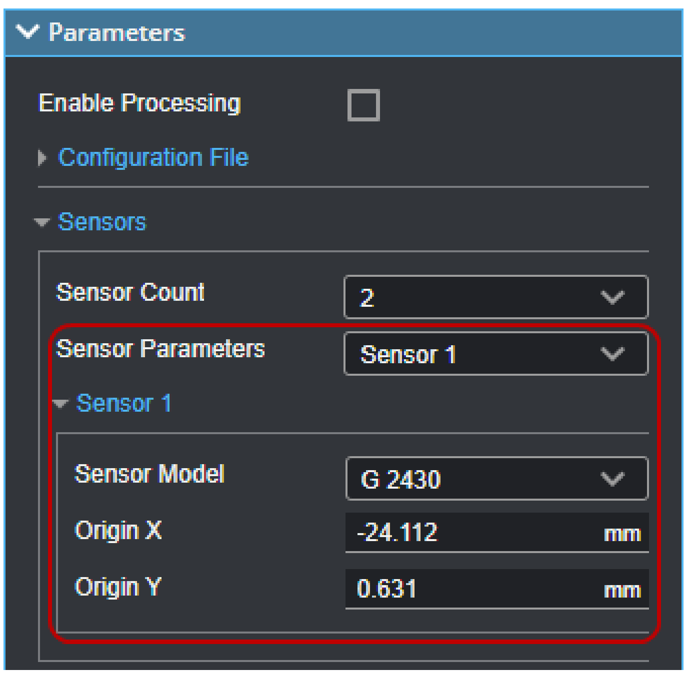
11. On the **Inspect** > **Tools** page, add a Surface Align Wide tool.
12. (Optional) If you have a previously saved configuration file, expand the **Configuration File** section and choose Load from the **Operation** drop-down, load that file, and go to step 16.



13. In the expandable **Sensors** section, set **Sensor Count** to the number of sensors in the system.

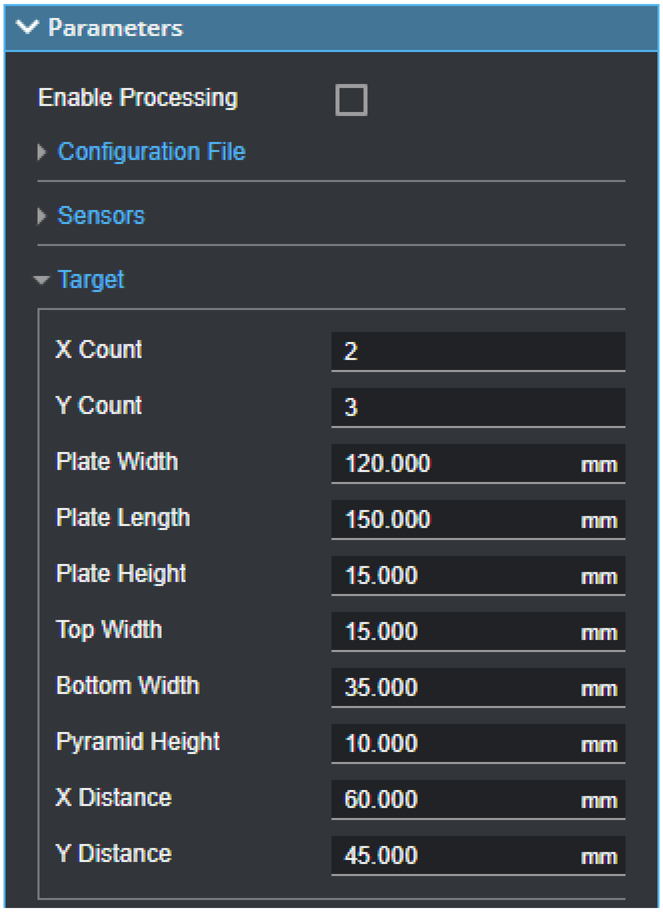


14. Under **Sensor Parameters**, select the sensors, one by one, and configure the parameters related to the sensor's position.



For more information, see *Sensor Parameters* on page 173.

15. Expand the **Target** section and configure the parameters related to the alignment target.



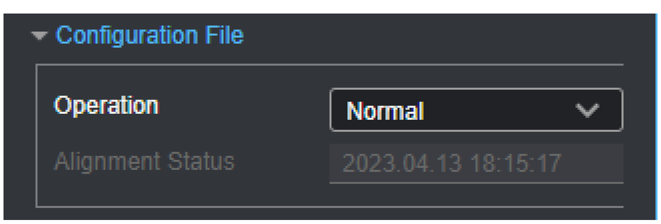
> ⚠ It is *crucial* to set these values correctly. These nominal values must also be accurate with respect to the alignment target.

For more information, see *Pyramid Plate Target Parameters* on page 172.

16. Configure the tool's remaining parameters (see *Parameters* on page 170).
17. Enable any measurements or the Processed Surface outputs if needed.

    These outputs can be useful for diagnostics.
18. Check the **Enable Processing** checkbox.

    The tool processes the scan data, using the provided sensor positions or rotations and alignment target characteristics, and saves a configuration file to C:/GoTools/SurfaceAlign. If the alignment process succeeds, the **Alignment Status** field in the **Configuration File** section displays the time and date of the alignment.

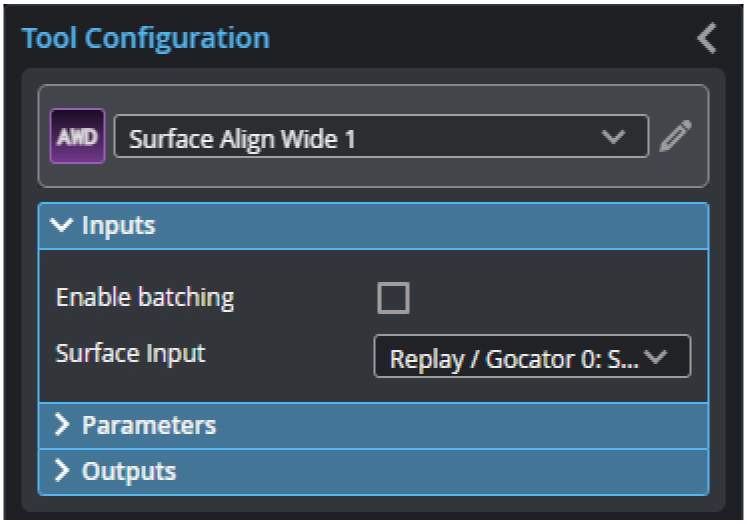


You must load the resulting configuration file in a Surface Merge Wide tool, typically in a separate job, which transforms and merges the multi-sensor scan data into a common coordinate system.

You can then apply other measurement tools to the merged scan data. For more information on the merging tool, see *Surface Merge Wide* on page 575.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

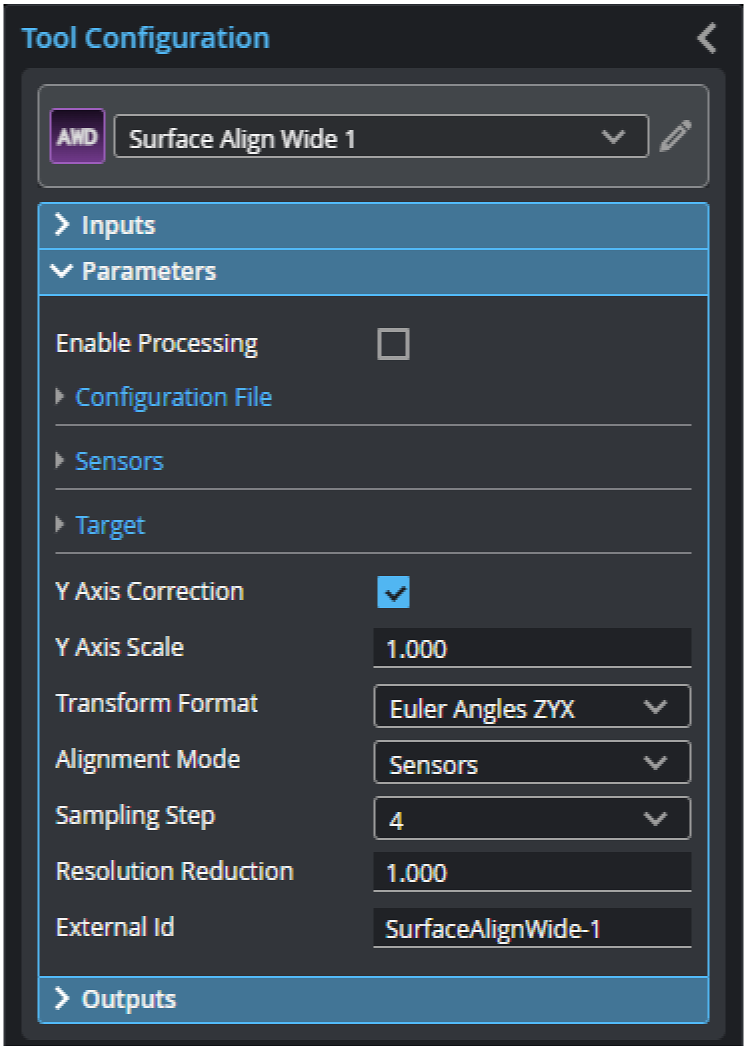


Inputs

Name	Description
Enable Batching	Leave this setting unchecked. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole.
Surface Input	The data the tool applies measurements to or processes. If the scan data in **Surface Input** uses uniform spacing (that is, you scanned the target with **Enable uniform spacing** checked), the tool displays additional sensor parameters that you must set. For more information, see *Using Uniform Spacing During the Alignment* on page 180.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Enable Processing	Causes the tool to perform the alignment. If the alignment is successful, the tool creates an XML alignment (calibration) file containing the transformations of the sensors that you must use with the Surface Merge Wide tool when scanning production targets to merge scan data. Make note of the XML file indicated in the log pane for use with Surface Merge Wide. Make sure to properly configure the tool *before* enabling this option. Disable it after performing the alignment; otherwise, the tool will continue performing the alignment on new frames of data, which will have an impact on performance.
Configuration File	An expandable section that contains configuration file-related parameters.
Operation	Actions that apply to the tool's configuration files. One of the following: • **Normal**: The tool automatically chooses this operation after you have chosen another operation. • **Load**: Displays a list of configuration files you can load. After you select a file,

Parameter	Description
	the tool loads it and displays a message in the log. The settings in the file, such as the number of sensors and their X and Y origin, are updated in the tool's parameters. • **Save**: Saves the tool's settings to a configuration file. Provide the name of the configuration file in the **Configuration Name** parameter (without an extension) and press Enter or Tab. Saving the initialization file saves you time if you need to adjust the positions of the sensors in the system and perform the alignment again as a rough starting point for the alignment procedure. • **Delete**: Deletes the file you select. • **Refresh**: Refreshes the list of files.
Sensors	An expandable section that contains sensor-related parameters.
Sensor Count	Indicates the number of sensors in the system.
Sensor Parameters	A drop-down that displays the settings of the selected sensor. For descriptions of the individual sensor parameters used for the alignment, see *Sensor Parameters* on page 173.
Target	An expandable section that contains parameters related to the pyramid plate target's specifications. For descriptions of the pyramid plate parameters, see *Pyramid Plate Target Parameters* on the next page. For information on alignment targets, see *Alignment Target and Setup* on page 162.
Y Axis Correction	Compensates for errors in encoder settings, which can distort scan data. If you cannot change the encoder settings or if you are working with data recorded with an incorrectly configured encoder, enable this setting. The tool will adjust the Y scaling to compensate and display the value in **Y Axis Scale**.
Y Axis Scale	The Y axis scale the tool calculates to compensate for incorrect encoder settings. Only displayed when **Y Axis Correction** is enabled.
Transform Format	The transformation format the tool uses. Typically, you can leave this at its default setting. Choosing a different format may be useful if you need to compare what the sensors detect to the transformation format used in your CAD package, for example. The setting does not affect alignment. One of the following: • Standard Angles ABC • Euler Angles ZYX • Euler Angles XYZ • Euler Angles ZYZ • Euler Angles ZXZ • Affine Angles YZX
Alignment Mode	The coordinate system used in the aligned system. One of the following: **Sensors** The Y axis is in the direction of motion. The resulting Y rotation of sensors is such

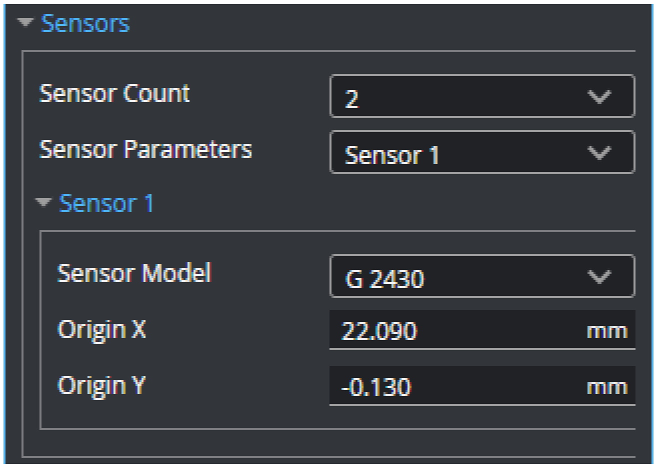
Parameter	Description
	that the average Y rotation is 0. X, Y, and Z = 0 are at average sensor center **Target Y Angle** The resulting Y rotation of sensors is such that the target surface is aligned to be horizontal. This mode is recommended when aligning to a conveyor on which the target is placed flat. X, Y, and Z = 0 are at average sensor center.
Sampling Step	The step in data points in both directions with which the surface is processed. Choosing a higher sampling step reduces the processing time the tool requires, but reduces fit accuracy. Useful if the surface being processed has a large number of data points. Typically, you will want to use as low a sampling step as possible; use a high sampling step only for initial testing purposes.
Resolution Reduction	Reduces the lateral resolution of the heightmap to reduce processing time.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Pyramid Plate Target Parameters

Parameter	Description
X Count Y Count	The count of the truncated pyramids on the pyramid plate, along the X and Y axis, respectively, that are used in the alignment. For example, if you are using only two sensors in a system, and they are each scanning a different column in a 3x3 pyramid plate, **X Count** would be set to 2. In a system where the sensors are angled so that they scan the same column, **X Count** would be set to 1.
Plate Width Plate Length Plate Height	The width, length, and height of the pyramid plate. These settings determine the size of the Difference Surface output; they do not affect the alignment.
Top Width Bottom Width	The width of the top of the truncated pyramids and the base of the pyramids, respectively.
Pyramid Height	The height of the truncated pyramids.
X Distance Y Distance	The distance between the centers of the truncated pyramids along the X and Y axis, respectively.

Setting Origin X and Origin Y

The following settings are available in the expandable **Sensor {n}** sections when **Sensors** > **Sensor Parameters** is set to one of the sensors in the system.

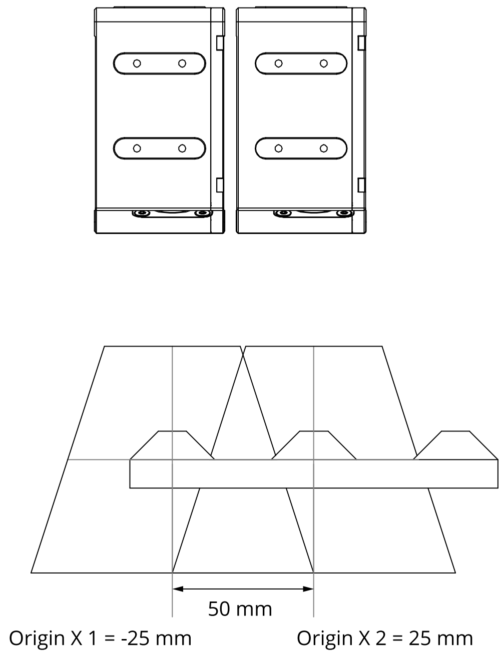


Sensor Parameters

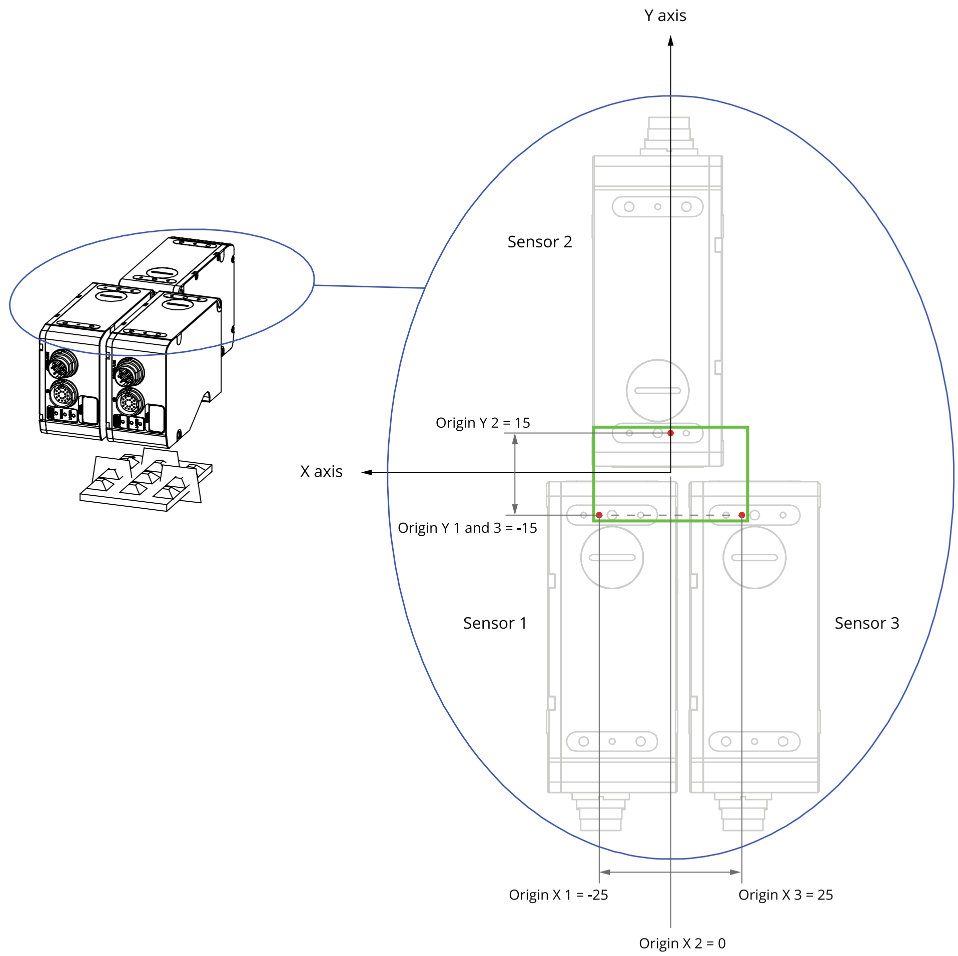
Parameter	Description
Sensor Model	Sets the model of the sensor selected in **Sensor Parameters**. This setting only determines how the sensors FOV is displayed in the data viewer. FOVs are only displayed if a sensor (or "Show All Sensors") is selected in **Sensor Parameters**. It does not affect the alignment.
Origin X {n} / Origin Y {n}	The X and Y origin of sensor {n}. In order for the alignment to succeed, you must enter the rough spatial relationship between the sensors. Use the known values based on physically measuring the mounted sensors, or estimate these values from unaligned scan data. Represents the estimated X and Y position of the intersection of the sensor's Z axis with the pyramid plate, relative to the center of a bounding box that encompasses the intersections of all sensors in the system. The accuracy of the values should be at least 1/10 the distance between pyramids. So, for example, with a distance of 25 mm between pyramids, the minimum accuracy is of these values is 2.5 mm.

The **Origin X {n}** and **Origin Y {n}** settings in the Surface Align Wide tool represent the estimated, real-world, X and Y position of the intersection of a sensor's Z axis with the flat surface of the pyramid plate, relative to the center of a bounding box that encompasses the intersections of all sensors in the system.

For example, in the following example of a two-sensor system, with the sensors placed side by side (no displacement along the Y axis), the X position of the intersection of the left sensor's Z axis and the alignment target's flat surface is roughly -25 mm, and the X position for the right sensor is 25 mm. You would use these values in the **Origin X** parameters of the appropriate sensor.



In the following, a three-sensor system with sensors positioned at different Y locations (viewed from above), the **Origin X** and the **Origin Y** value of each sensor is set to the distance (X and Y position) of the intersections between the alignment target surface and the sensors' Z axes, relative to the center of the bounding box that encompasses the intersections.



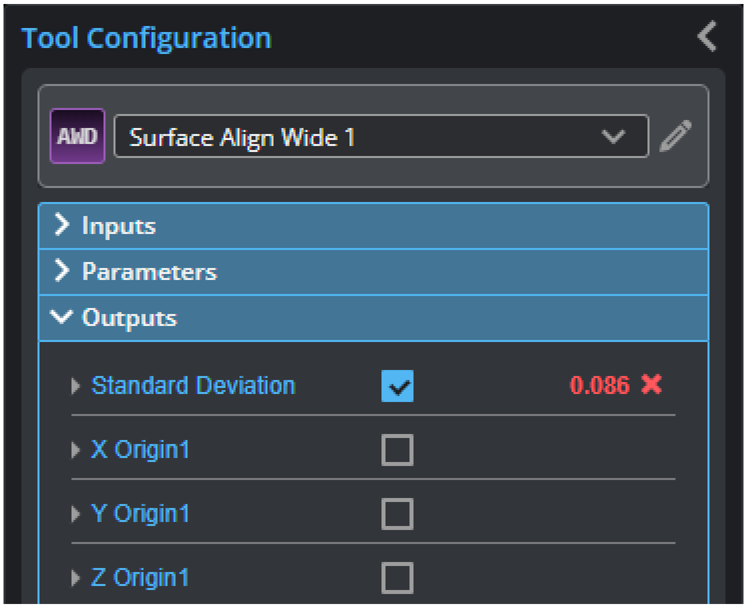
Note however that for sensor systems with sensors positioned at different Y locations, you can set **Origin Y** to 0 if the entire alignment target is scanned during the alignment procedure: the algorithm uses the edges of the alignment target to determine the Y component of the sensors' origin. If you do not scan the entire alignment target during the procedure, you must enter an estimate of the Y position of the sensors' origins.

🗋	It is very important to consult the specifications of the sensors in your system to determine the direction of X and Y. In multi-sensor systems, the Y axis is determined by the Main sensor's orientation.

If your application requires that sensors should be angled, remember to take this into account when estimating the values needed for **Origin X** and the **Origin Y**, paying attention to where the Z axis of each sensor intersects with the alignment target's lower flat surface.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Measurements

Measurement
Standard Deviation Alignment uncertainty (an indicator of alignment quality). Returns Invalid if the uncertainty is very high.
X Origin{n} **Y Origin{n}** **Z Origin{n}** The X, Y, and Z transformations calculated for sensor {n}.
X Angle {n} **Y Angle {n}** **Z Angle {n}** The X, Y, and Z angle transformations calculated for sensor {n}.

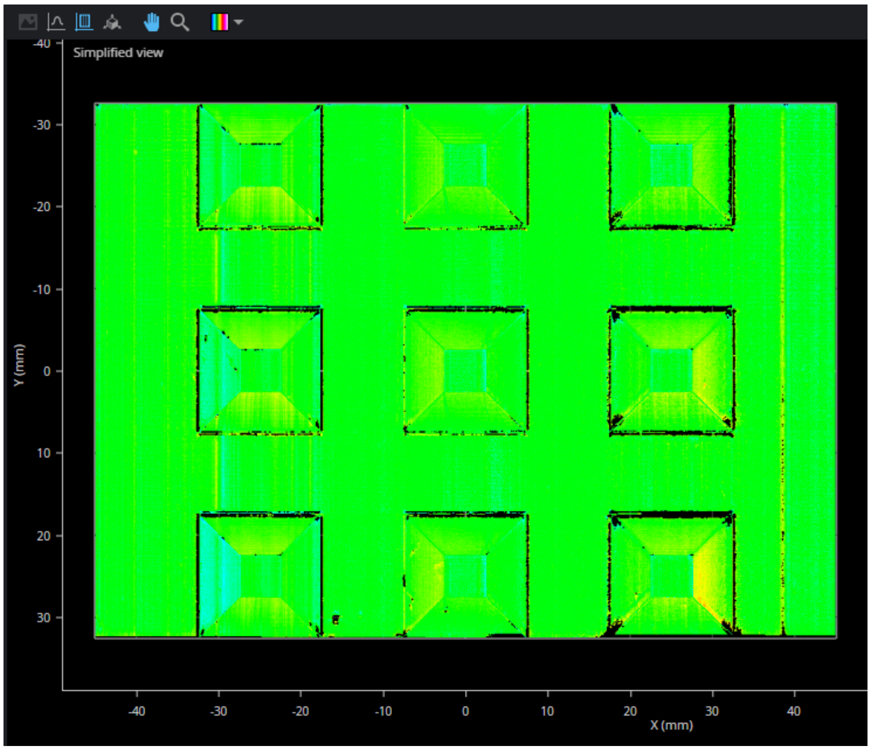
Data

Type	Description
Difference Surface	A heightmap that displays discrepancies and agreement between the nominal values provided in the tool's parameters and the actual dimensions in the scan data. Use this output to assess the quality of the alignment and to determine where there may be issues. For more information, see See *Wide Layouts (Surface Align Wide Tool)* on page 160.

Understanding the Difference Surface

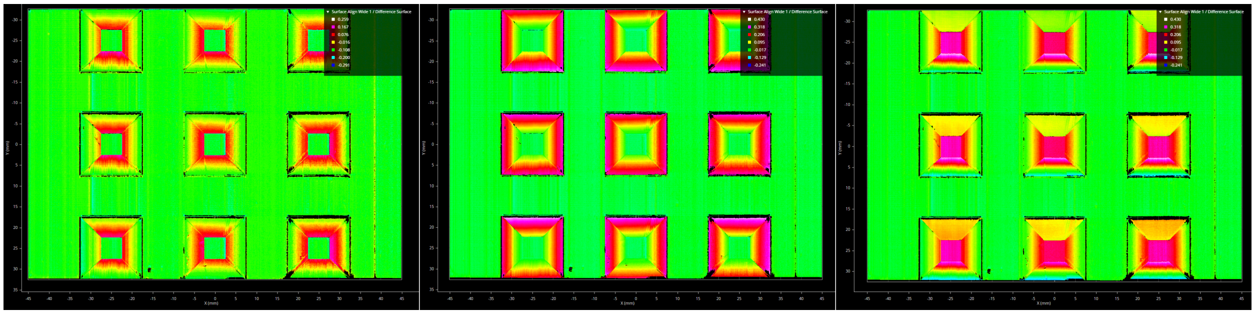
The Difference Surface output is a heightmap that displays discrepancies and agreement between the alignment target dimensions provided in the tool's parameters (the expected dimensions) and the actual dimensions of the target in the scan data. The tool displays areas where the scan data is consistent with the nominal values as close to Z = 0. In areas where the scan data is higher than expected, the tool displays data points with a higher Z value (that is, significantly greater than 0). Where the scan data is lower than expected, the tool displays data points with a lower Z value (that is, significantly less than 0).

A "good" Difference Surface is nearly flat, indicating that the majority of the data points are close to the expected dimensions of the target. The scan data should also contain the entire alignment target, including the area between the pyramids.

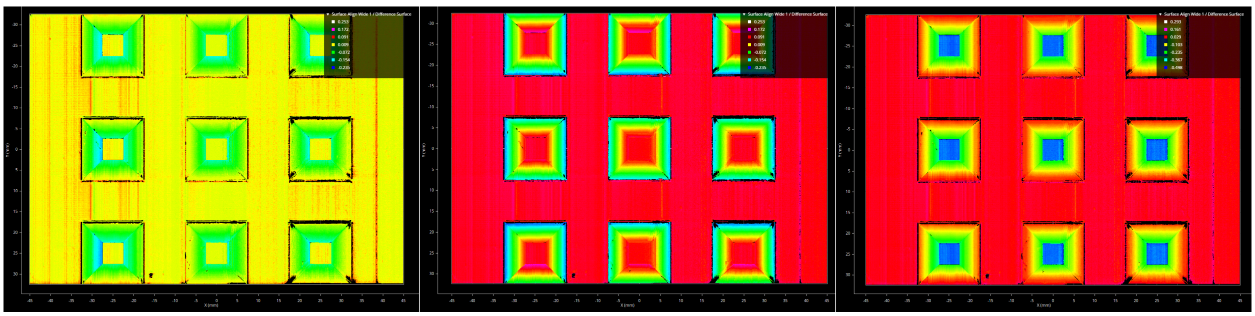


The presence of relatively systematic differences (for example, around the tops or the bases of all of the pyramids) can result from incorrectly entered values in the tool's **Target** parameters, or an incorrectly manufactured alignment target (for example, using incorrect specifications). To properly assess whether there are discrepancies, you may need to zoom in on the scan data in the data viewer. It may also help to adjust the way the colors of the height map are displayed using Region or Manual in the **Range** setting above the data viewer.

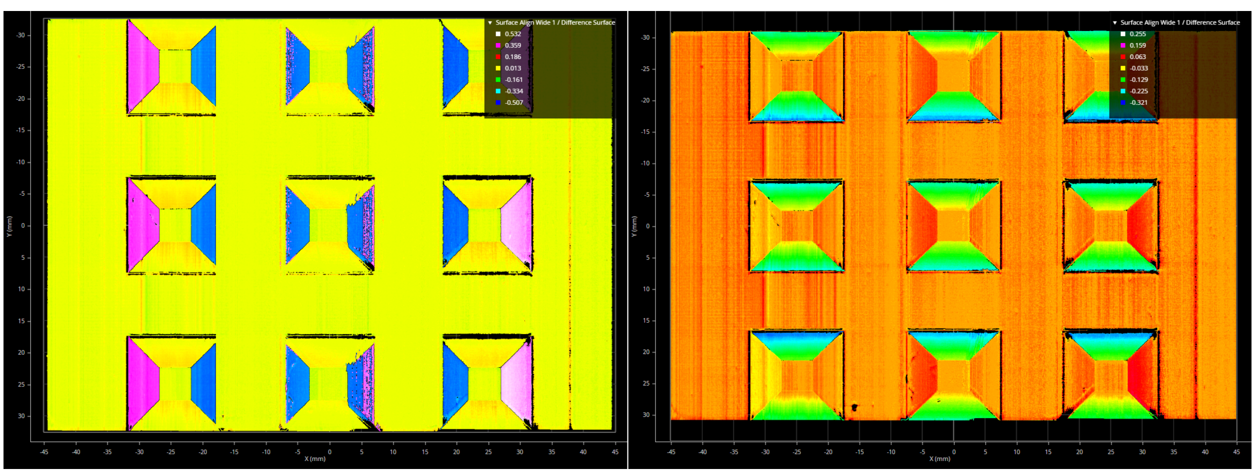
For example in the following, from left to right, the values provided in the **Top Width**, **Bottom Width**, and **Pyramid Height** parameters are roughly 6% lower than the actual dimensions of the alignment target. The height deviations shown in the Difference Surface output are "focused" around the area that is problematic.



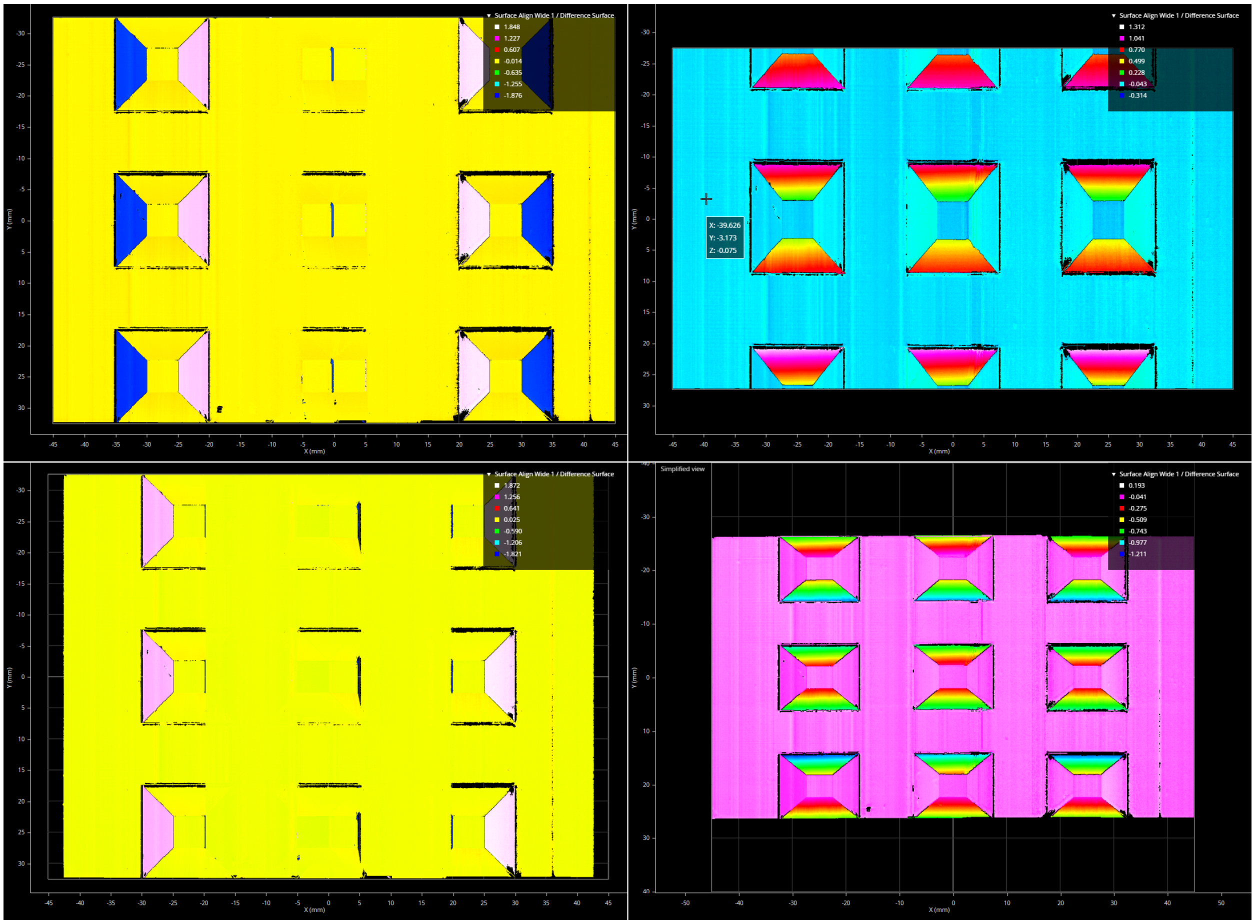
In the following, the same pyramid parameters are set roughly 9% higher than the actual dimensions. The resulting Difference Surface represents these discrepancies as being lower than expected in the heightmap.



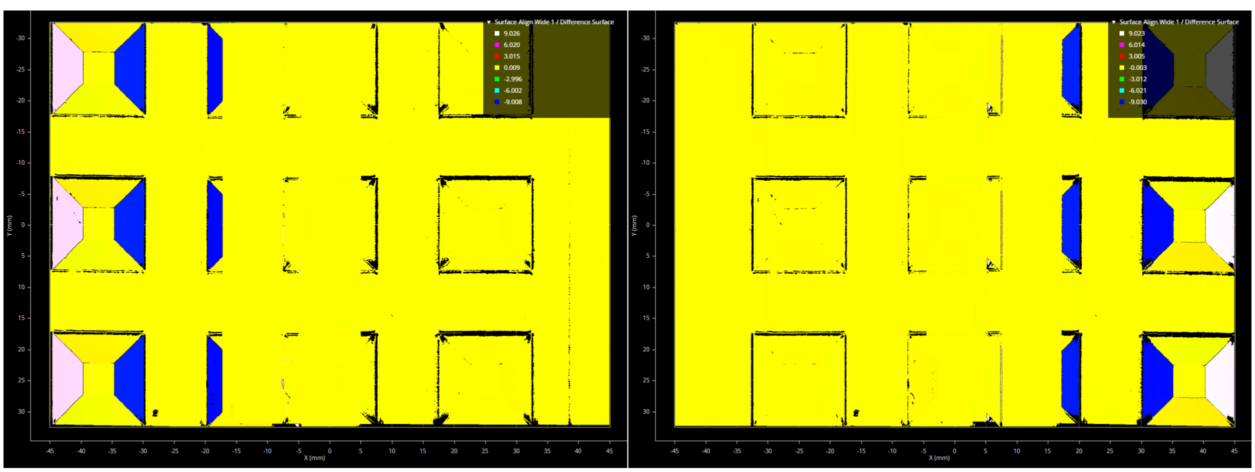
Similarly, relatively systematic deviations along a specific axis can indicate discrepancies between the values provided in **X Distance** or **Y Distance** and the actual target as scanned. In the following Difference Surface outputs, there is a nominal-actual discrepancy along the X axis and Y axis, respectively.



More extreme discrepancies can produce more obvious distortions in Difference Surface. In the following images, on the left, the pyramid tops of the center column are almost missing, resulting from **X Distance** being above and below nominal, respectively. On the right, the pyramids are stretched or compressed along the Y axis, resulting from **Y Distance** being above and below nominal, respectively; the target is also cropped.



An incorrectly set **Origin X** or **Origin Y** parameter, in the **Sensors** section, can result in somewhat systematic discrepancies. In the following images, the **X Origin** of the left and the right sensor, respectively, is incorrectly set. Note that the values used are extreme.



Note that the pyramid plate parameters, in the **Target** section, do not affect the alignment. These parameters only affect the size of the Difference Surface output.

Using Uniform Spacing During the Alignment

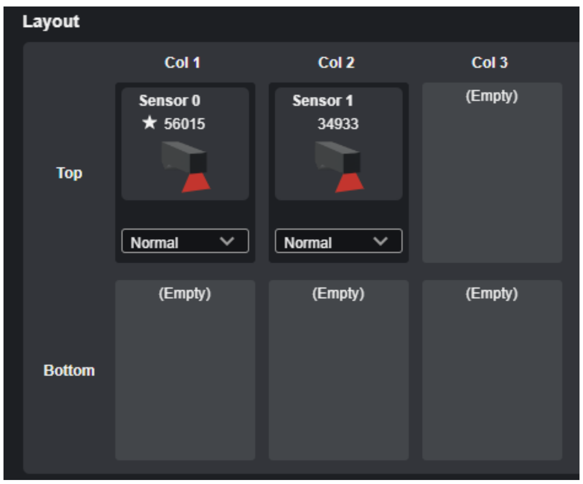
If you choose to enable **Uniform Spacing** when scanning the alignment target, you must perform these additional steps when running the alignment tool on the scanned target. Otherwise, the alignment will fail.

These steps separate the individual sensors' scan data in the heightmap so that the alignment tool can process the scan data from each sensor individually. To do this, you set artificial offsets along the X axis to separate the scan data so that there is no overlap between the sensors.

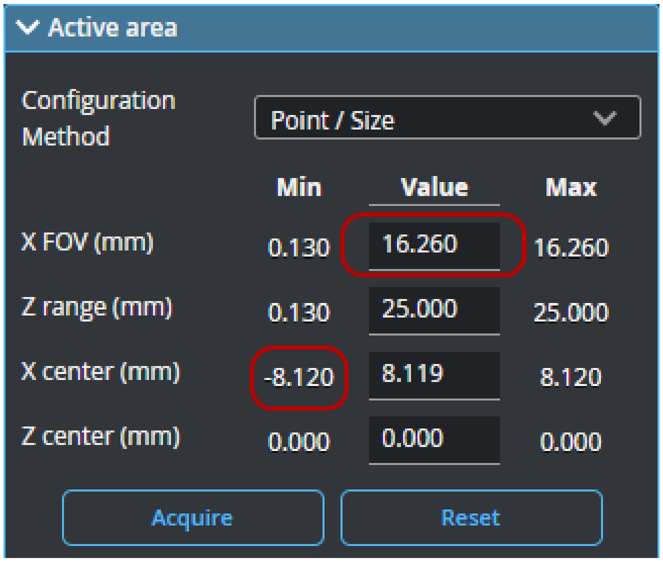
Do the following:

1. (Optional) In the web interface, go to the **System** > **Design** page, click the action menu icon, click **Add and manage sensors**, and make note of the names, serial numbers, and positions of the sensors listed in the **Layout** section.

   This information can be useful when you continue configuring the system in other parts of the web interface.

2. In the web interface, go to the **Scan** page.
3. For each sensor in the system, in the **Active Area** tab, make note of the values for **X Field of View** and the Min value in the **X center** row.



4. For each sensor in the system, in the **Transform** section, set **X Offset** to a value that eliminates any overlap between sensors, and make note of this value.

   For example, for sensors with a 100 mm field of view, you might choose 1 or 2 mm for a blank area. Therefore, for the first sensor, you could set its **X Offset** 0 mm.

   For the Right sensor, you could set its **X Offset** to 101 or 102 mm.

   For systems with more than two sensors, set the offsets in a similar way to ensure that the individual sensors' scan data in the heightmap does not overlap.

5. If you haven't already done so, go to the **Scan** > **Tools** page and add an instance of Surface Align Wide.

6. For each sensor, select the sensor you need to configure in the **Sensors** > **Sensor Parameters** drop-down and copy the "X Field of View," "X Start (Min value)," and "X Offset" values from the **Scan** page into the corresponding parameters in the alignment tool.

   Make sure to do this for *each* sensor.

7. If you have not already done so, finish configuring the rest of the tool's parameters.

Note that the artificial offsets you use to create the combined heightmap for the alignment procedure are unrelated to the values you need to use in **Origin X {n}**; for more information on this setting, see *Setting Origin X and Origin Y* on page 172.

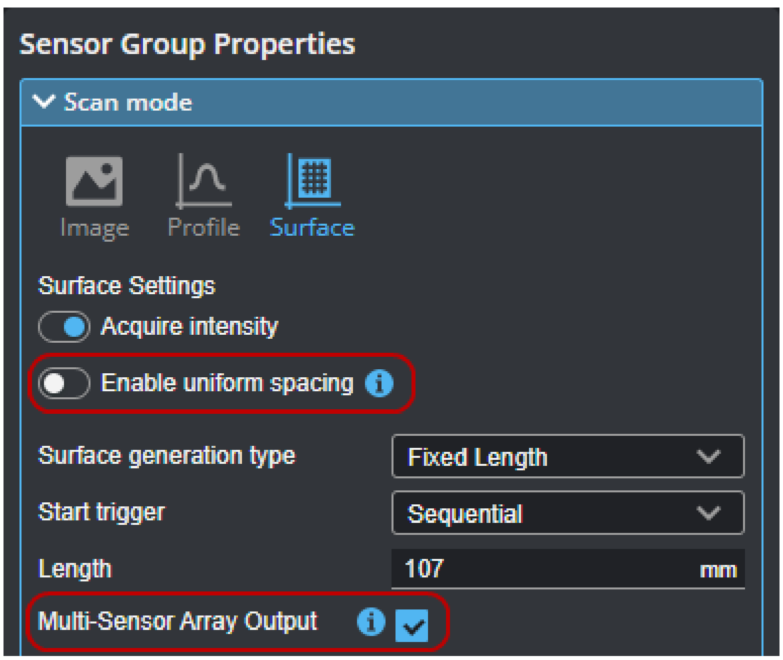
Ring Layouts (Surface Align Ring Tool)

⚠ Performing alignment using the Surface Align Ring or Surface Align Wide tools (which results in 6 degrees of freedom) involves *considerable* setup effort. First, the 6 DoF alignment targets are more difficult to manufacture than an alignment bar and require a very high degree of accuracy; 3D printed alignment targets are not usually sufficiently accurate. Second, the alignment tools have many parameters that must be properly configured to successfully perform an alignment.

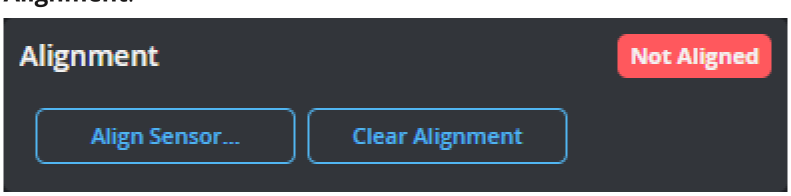
You only need to use this tool to produce a configuration file containing the sensor transformations that Surface Mesh needs to merge scan data. For more information, see *Surface Mesh* on page 582.

This tool requires acceleration (either by a PC-based application or by GoMax). For more information, see *Running GoPxL on a Windows PC* on page 828.

⚠ Before scanning the alignment target, remember to uncheck **Enable uniform spacing** in the **Scan mode** section on the **Scan** page. Also enable **Multi-Sensor Array Output**.



Sensors must not previously have been aligned using the alignment routine on the **System** > **Alignment** page. This is indicated by "Not Aligned" on that page. If necessary, click **Clear Alignment**.

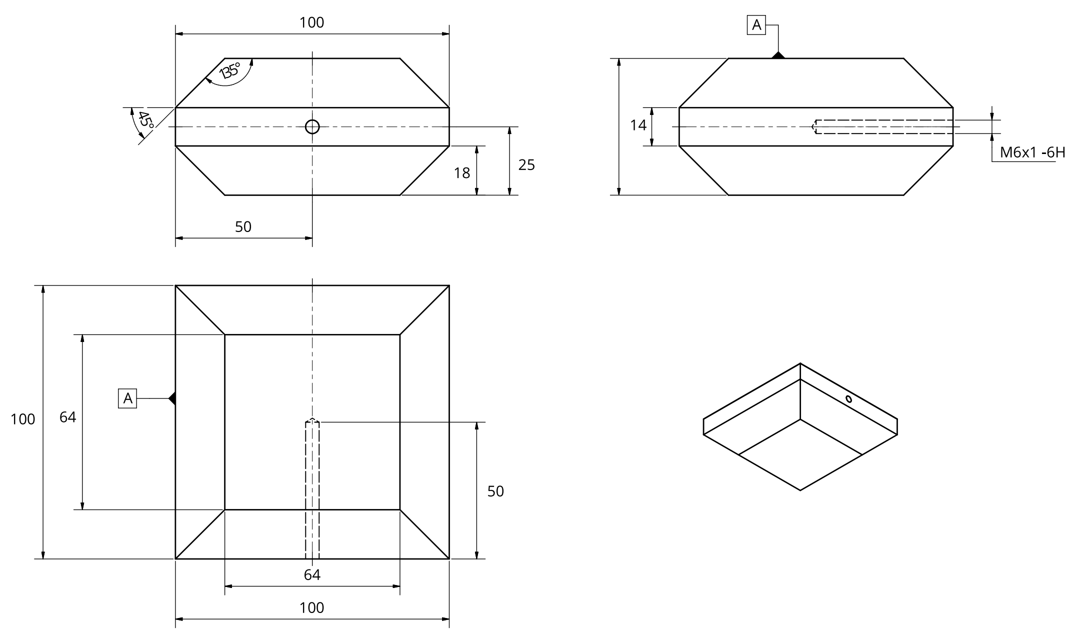


You can use the Surface Align Ring tool to align a G2 multi-sensor system in a ring layout or a dual- or multi-sensor partial ring layout with 6 degrees of freedom. The alignment procedure saves the transformations required for the sensors in a configuration file. Unlike alignment performed using the **Alignment** panel, the tool also compensates for X angle rotation (giving you a full six degrees of freedom).

Note that in order to perform scans in production, you must use the Surface Mesh tool (loading the configuration file created by this tool) to stitch the scans from the individual sensors into Mesh data; for more information on the Surface Mesh tool, see *Surface Mesh* on page 582. You can then either perform measurements directly on the Mesh data using the Mesh measurement tools (see *Mesh Measurement* on page 747) or you can extract Surface data from the Mesh data and apply any built-in or custom GDK-based Surface tools to the resulting data (see *Surface Measurement* on page 444).

Alignment Target

This alignment tool requires the use of a double-sided truncated pyramid alignment target. You can find CAD files for this type of target under `Tools\Alignment CAD\Double Sided Pyramid` in the Utilities package (for example, 14405-x.x.xx.xx_SOFTWARE_GO_Utilities.zip, available on LMI's Product Downloads page). Note that you should adapt the size of the alignment target to the size of the sensors in your system: the target should be scaled so it fills most of the field of view of a sensor.



Example dimensions for mid-size FOV sensors.

Note that after using this tool, on the **Alignment** panel on the **Scan** page, GoPxL indicates that the sensors are unaligned.

The following requirements should be satisfied for best results:

1. Make sure the alignment target surface is not too shiny or too dark to be scanned.
2. Maximize the size: The target should fill the scan volume of each sensor but not extend past the field of view of the sensors.

3. Edges do not need to be perfectly sharp: The alignment tool performs a plane fit to points within the planar surfaces and excludes data close to the edges.
4. No planar surfaces other than those on the alignment target should be visible in the scan results. For example, do not scan the target set on a flat surface such that the surface is included in the scan results.
5. Each sensor in the system must be able to properly scan a minimum of 5 planar surfaces of the alignment target. If necessary, rotate the alignment target to ensure this.
6. Ensure the ratio of X and Y scan resolution does not exceed 5. For more information on setting the X resolution, see *Uniform Spacing* on page 197. For information on setting Y resolution, see *Triggers* on page 206.

Alignment Procedure

The following provides the steps involved in performing a high-accuracy, tool-based alignment.

🗋	The following assumes you are scanning the alignment target using unaccelerated sensors and then uploading the recording to a PC instance of GoPxL. The procedure for doing this with accelerated sensors is similar.

To perform the alignment:

1. Fabricate an alignment target appropriate for your system (wide layout versus ring layout).

   For details on the alignment target, see *Alignment Target* on the previous page.

2. If you have not already done so, set up and configure the multi-sensor system.

   The following sections describe installing, setting up, and configuring a system:

   Installation on page 36

   Network and Sensor Setup on page 50

   Sensor Management and Maintenance on page 117

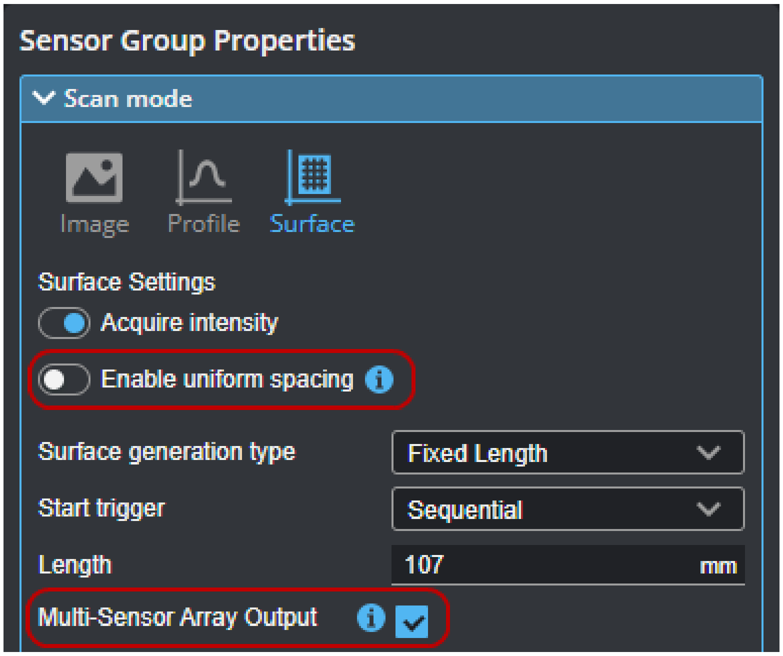
   Creating a Sensor System on page 124

   When configuring the system's layout, put *all sensors* in the *top row*, regardless of their physical position.

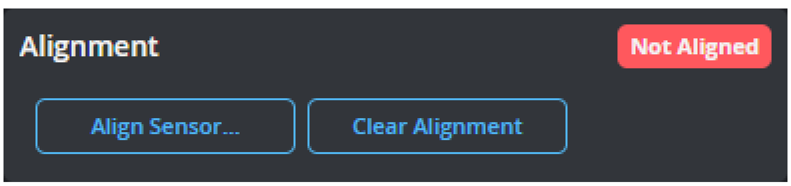
   Scan - Configuring Acquisition on page 195

3. Make sure to do the following.

   Disable **Uniform Spacing** and enable **Multi-Sensor Array Output** in the **Scan mode** section on the **Inspect** > **Scan** page.



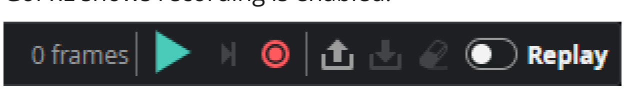
Sensors must not previously have been aligned using the alignment routine on the **System** > **Alignment** page. This is indicated by "Not Aligned" on that page. If you see "Aligned" here, click **Clear Alignment**,



> Although you can scan the alignment target without acceleration, you *must* perform the alignment using PC-based acceleration (for more information, see *Running GoPxL on a Windows PC* on page 828). Because starting acceleration after having performed a scan clears scan data from a sensor, if you are going to perform alignment on-sensor, you should start acceleration before continuing.
>
> You can also optionally download the scan data and perform the alignment on the scanned target using GoPxL on Windows. For more information, see *Working with Scan Data (Toolbar)* on page 112.

4. If any sensors are installed in a "reversed" orientation (that is, such that the conveyor moves in the same direction as the sensor's positive Y axis), be sure to set the sensor's layout to Reverse.
5. Enable recording by clicking the Record button to the right above the data viewer.

GoPxL shows recording is enabled.

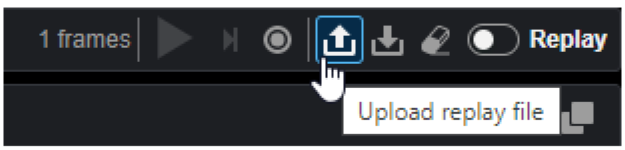


6. Start the transport system and then perform a scan of the alignment target.
7. Turn off recording.

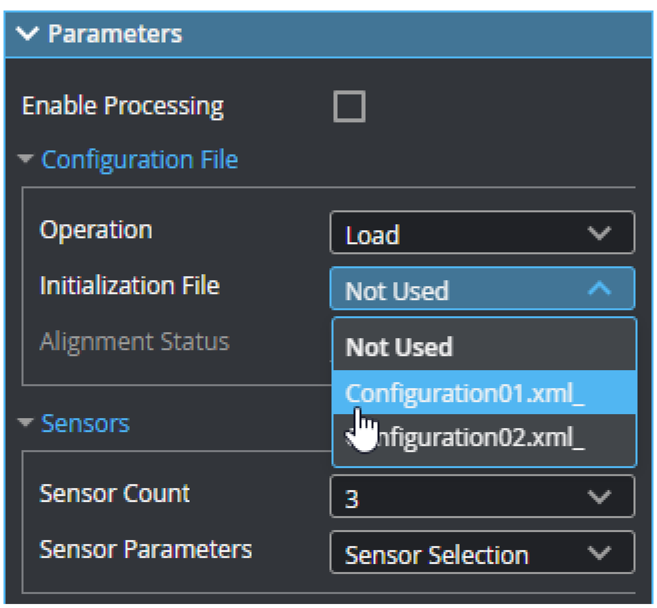
8. Download the recording and make note of the downloaded file's location.

9. Launch a PC instance of GoPxL.

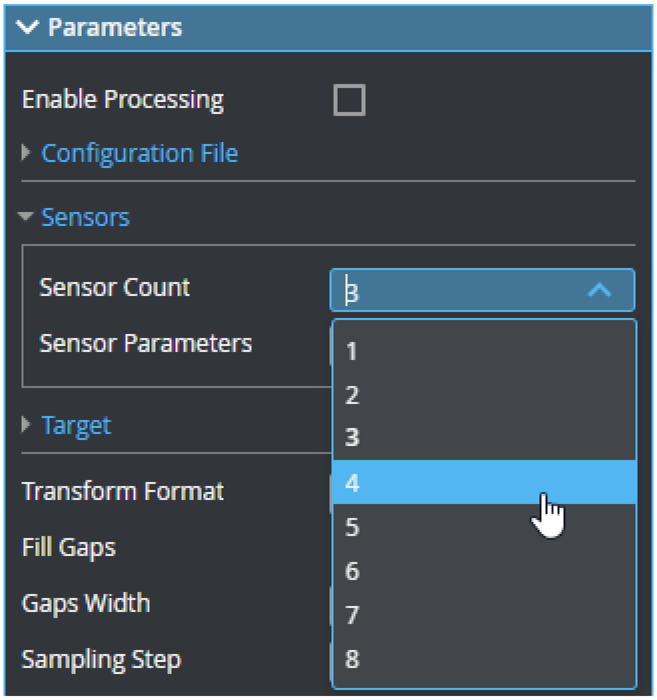
   For more information, see *Running GoPxL on a Windows PC* on page 828.
10. Upload the recording of the alignment target in the PC instance.



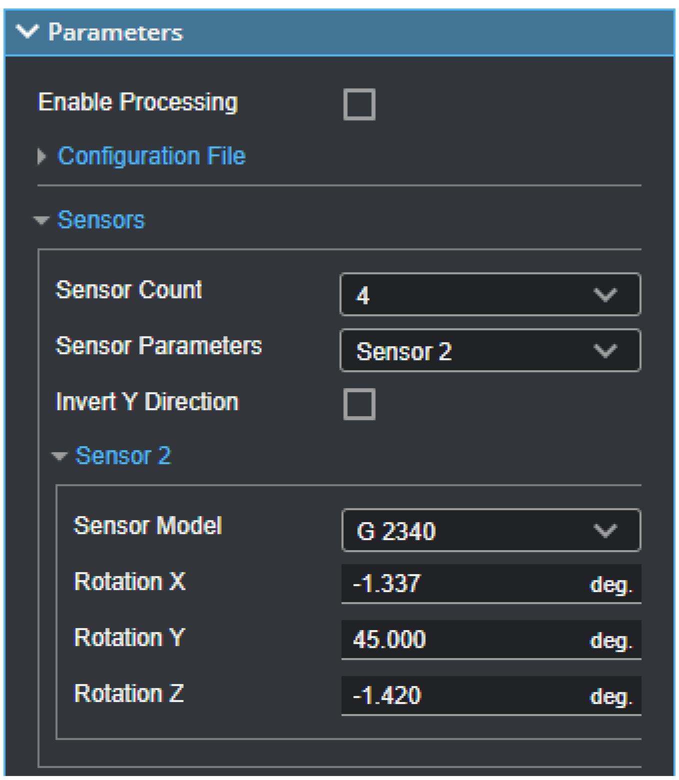
11. On the **Inspect** > **Tools** page, add a Surface Align Ring tool.
12. (Optional) If you have a previously saved configuration file, expand the **Configuration File** section and choose Load from the **Operation** drop-down, load that file, and go to step 16.



13. In the expandable **Sensors** section, set **Sensor Count** to the number of sensors in the system.

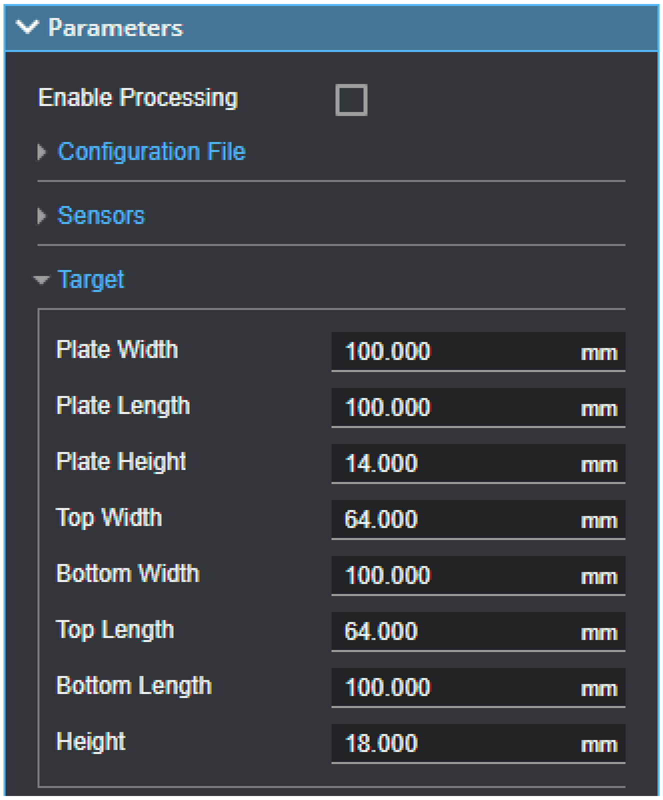


14. Under **Sensor Parameters**, select the sensors, one by one, and configure the parameters related to the sensor's position.



For more information, see *Sensor Parameters* on page 192.

15. Expand the **Target** section and configure the parameters related to the alignment target.

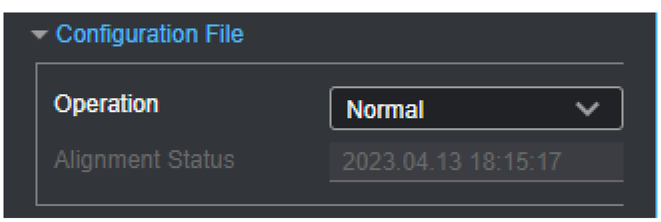


For more information, see *Double-Sided Pyramid Target Parameters* on page 192.

16. Configure the tool's remaining parameters (see *Parameters* on page 190).
17. Enable any measurements or the Processed Surface outputs if needed.

    These outputs can be useful for diagnostics.
18. Check the **Enable Processing** checkbox.

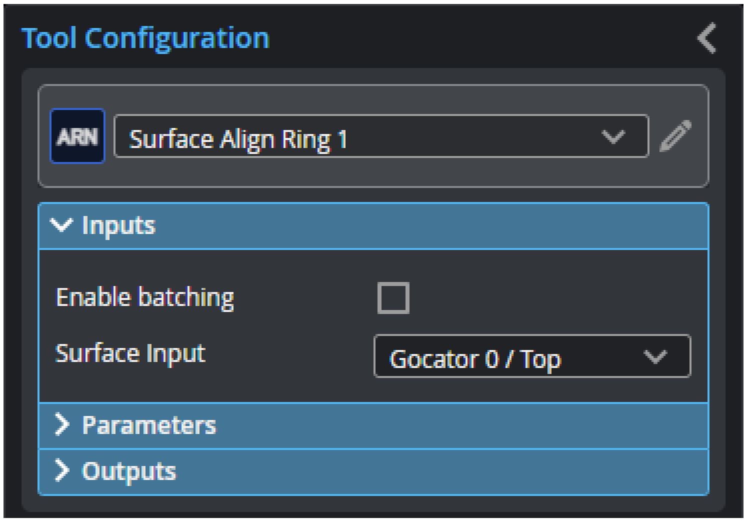
    The tool processes the scan data, using the provided sensor positions or rotations and alignment target characteristics, and saves a configuration file to C:/GoTools/SurfaceAlign. If the alignment process succeeds, the **Alignment Status** field in the **Configuration File** section displays the time and date of the alignment.



You must load the resulting configuration file in a Surface Mesh tool, typically in a separate job, which transforms the multi-sensor scan data into a common coordinate system and produces a Mesh output. You can then apply other measurement tools to the merged scan data. For more information on the merging tool, see *Surface Mesh* on page 582.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

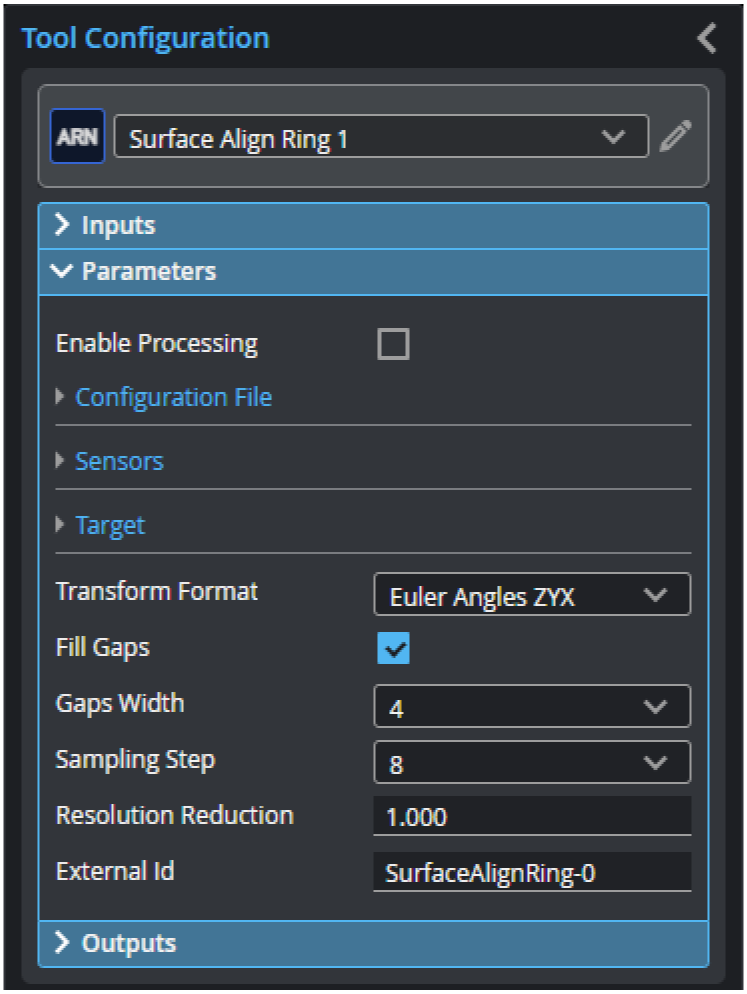


Inputs

Name	Description
Enable Batching	Leave this setting unchecked. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole.
Surface Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

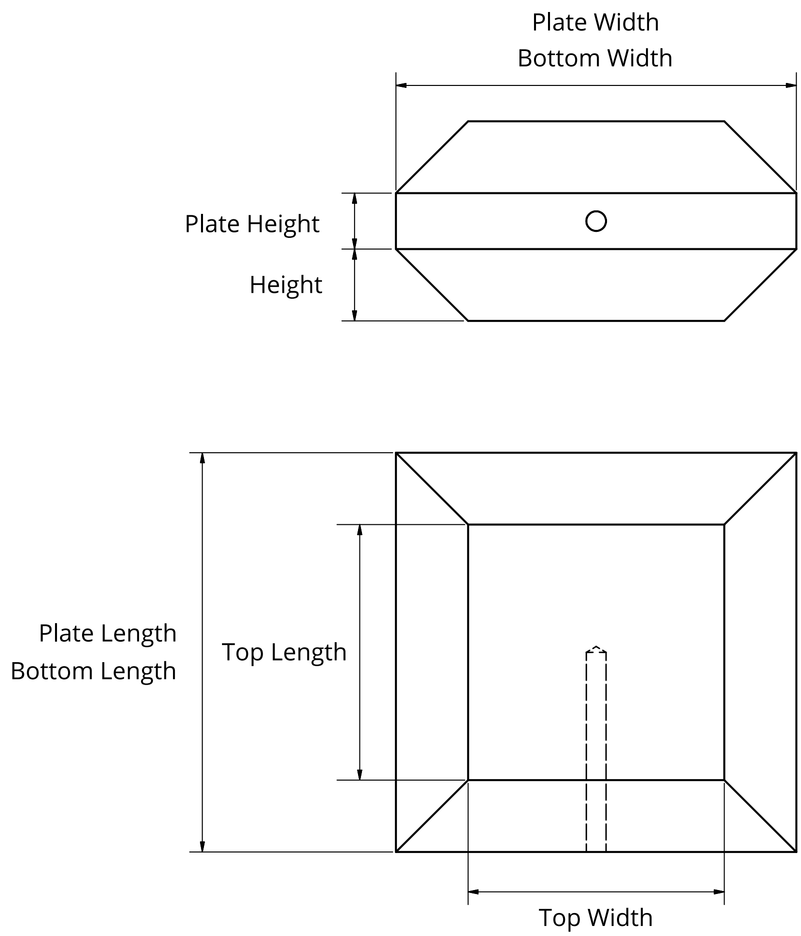
Parameter	Description
Enable Processing	Starts the alignment procedure. Make sure to properly configure the tool *before* enabling this option. Disable it after performing the alignment; otherwise, the tool will continue performing the alignment on new frames of data, which will have an impact on performance. (Ideally, you should use a separate job for the alignment procedure.)
Configuration File	An expandable section that contains file-related parameters.
Operation	Actions that apply to the tool's configuration files. One of the following: • **Normal**: The tool automatically chooses this operation after you have chosen another operation. • **Load**: Displays a list of configuration files you can load. • **Save**: Saves the tool's settings to a configuration file. Provide the name of the configuration file in the **Configuration Name** parameter and press Enter or Tab. You use the file (by later loading it in the tool) to provide a rough initial estimate of sensor orientation and position to the tool when it performs

Parameter	Description
	alignment.. • **Delete**: Deletes the file you select. • **Refresh**: Refreshes the list of files.
Sensors	An expandable section that contains sensor related parameters.
Sensor Count	Indicates the number of sensors in the system.
Sensor Parameters	A drop-down that display the settings of the selected sensor. For descriptions of the individual sensor parameters used for the alignment, see *Sensor Parameters* on the next page.
Invert Y Direction	Converts from a left-handed coordinate system to a right-handed coordinate system. Enable this only if the sensor is mounted in a reverse position and identified as Reversed in the **Add & Manage Sensor Group** dialog in the **System** > **Design** page. This setting applies to all sensors.
Target	An expandable section that contains parameters related to the double-sided pyramid target's specifications. For descriptions of the pyramid parameters, see *Double-Sided Pyramid Target Parameters* on the next page.
Transform Format	The transformation format the tool uses. Choosing a different format may be useful if you need to compare what the sensors detect to the transformation format used in your CAD package, for example. The setting does not affect alignment. One of the following: • Standard Angles ABC • Euler Angles ZYX • Euler Angles XYZ • Euler Angles ZYZ • Euler Angles ZXZ • Affine Angles YZX
Fill Gaps	When this option is enabled, the tool displays a **Gaps Width** parameter (see below).
Gaps Width	The kernel the tool uses to initially calculate the surface normal required for alignment. Typically, a value of 4 works for most applications. If alignment fails and you can't track down the issue, try a different value.
Sampling Step	The step in data points in both directions with which the surface is sampled. Choosing a higher sampling step reduces the processing time the tool requires, but reduces fit accuracy. Typically, you will want to use as low a sampling step as possible; use a high sampling step only for initial testing purposes.
Resolution Reduction	Reduces the lateral resolution of the heightmap to reduce processing time.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Sensor Parameters

Parameter	Description
Sensor Model	Sets the model of the sensor selected in **Sensor Parameters**.
Rotation X {n} Rotation Y {n} Rotation Z {n}	Sets the X, Y, and Z rotations of the sensor selected in **Sensor Parameters**. In order for the alignment to succeed, you must enter the rough orientation of the sensors.

The following image indicates which parameters (see the table below) correspond to which parts of the alignment target.



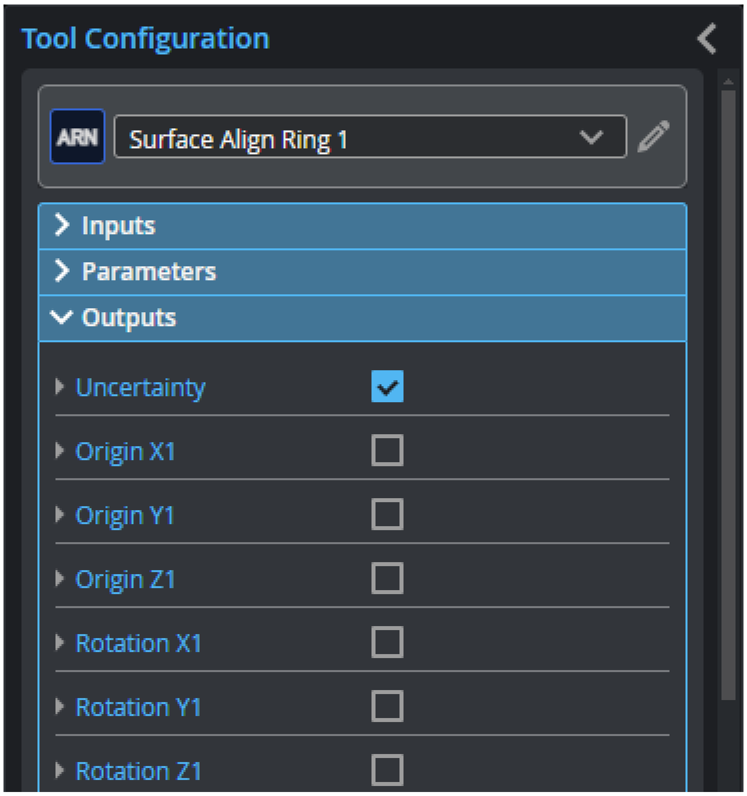
Double-Sided Pyramid Target Parameters

Parameter	Description
Plate Width Plate Length Plate Height	The width, length, and height of the pyramid plate.
Top Width Bottom Width	The width of the top of the truncated pyramid and the base of the pyramid, respectively.

Parameter	Description
Top Length Bottom Length	The length of the top of the truncated pyramid and the base of the pyramid, respectively.
Height	The height of the truncated pyramid.

Outputs

Most tools provide measurements, geometric features, or data as outputs.

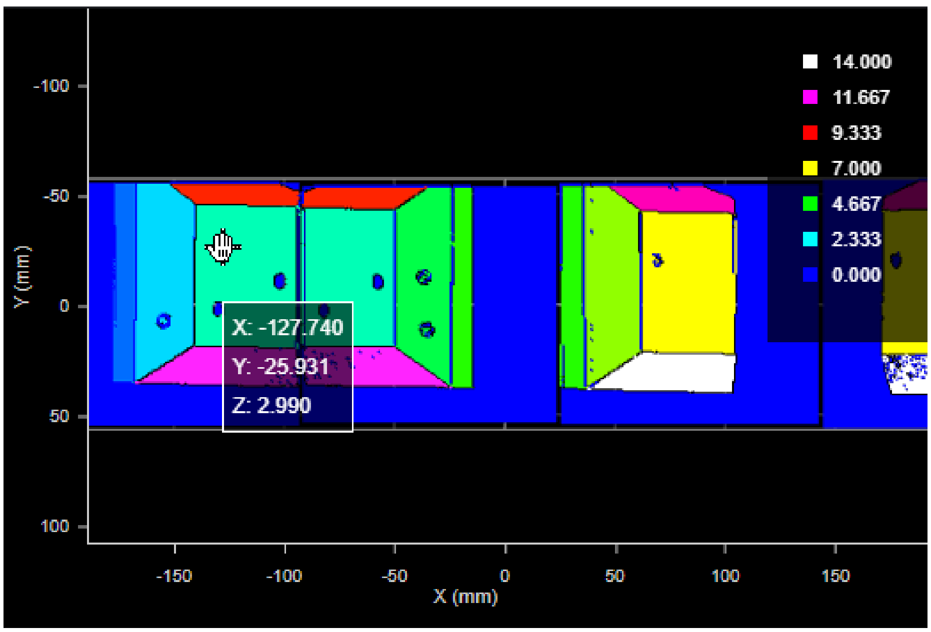


Measurements

Measurement
Uncertainty Alignment uncertainty (an indicator of alignment quality).
Origin X{n} **Origin Y{n}** **Origin Z{n}** The X, Y, and Z transformations calculated for sensor {n}.
Rotation X{n} **Rotation Y{n}** **Rotation Z{n}** The X, Y, and Z angle transformations calculated for sensor {n}.

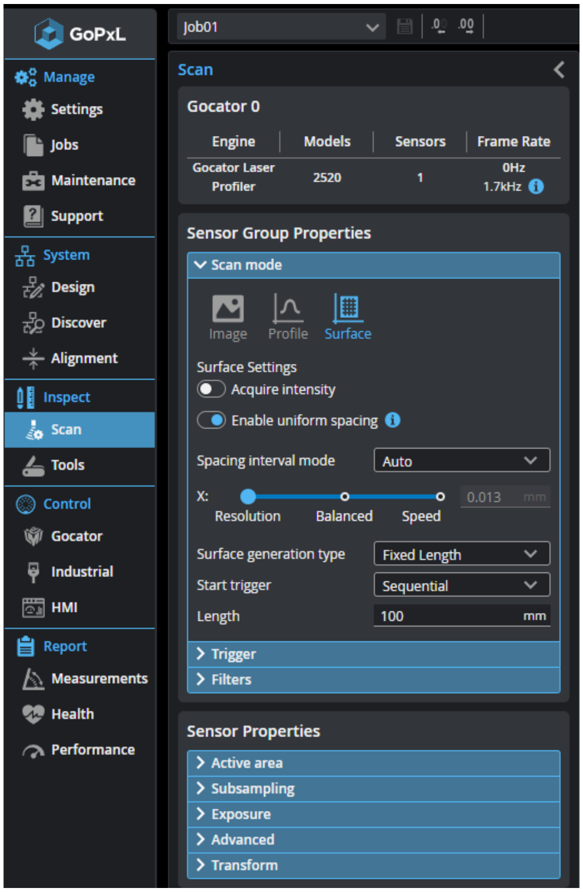
	Enable Processing must be checked to view the following diagnostic data outputs.

Data

Type	Description
Processed Surface	Use this for diagnostic purposes. Shows the Surface data from the individual sensors in a single Surface. The data is not merged.
Difference Surface	Use this for diagnostic purposes. Useful for understanding where inaccuracies may be coming from, for example, if a plane on the alignment target is distorted. This could be caused by inaccuracies in the alignment target, incorrectly entered values in the parameters under **Target** section in the tool, an incorrectly configured transport system encoder, and so on.
Segmentation Surface	Use this for diagnostic purposes. The tool identifies the target's planes in the scan data and assigns a heightmap value to each. So for example, in the following image, the larger rectangular areas of the two pyramids to the left are assigned a height of 2.99 mm, indicating that the tool treats them as belonging to the same plane (that is, views of the same target plane from different sensors). If planes aren't assigned an appropriate value (such as if the same planes on the target are assigned different heightmap values, or different planes are assigned the same values), you may need to adjust the position of the affected sensors slightly, for example, to improve the scan data of the planes involved. 

Scan - Configuring Acquisition

You configure acquisition in the **Scan** page under the **Inspect** category.

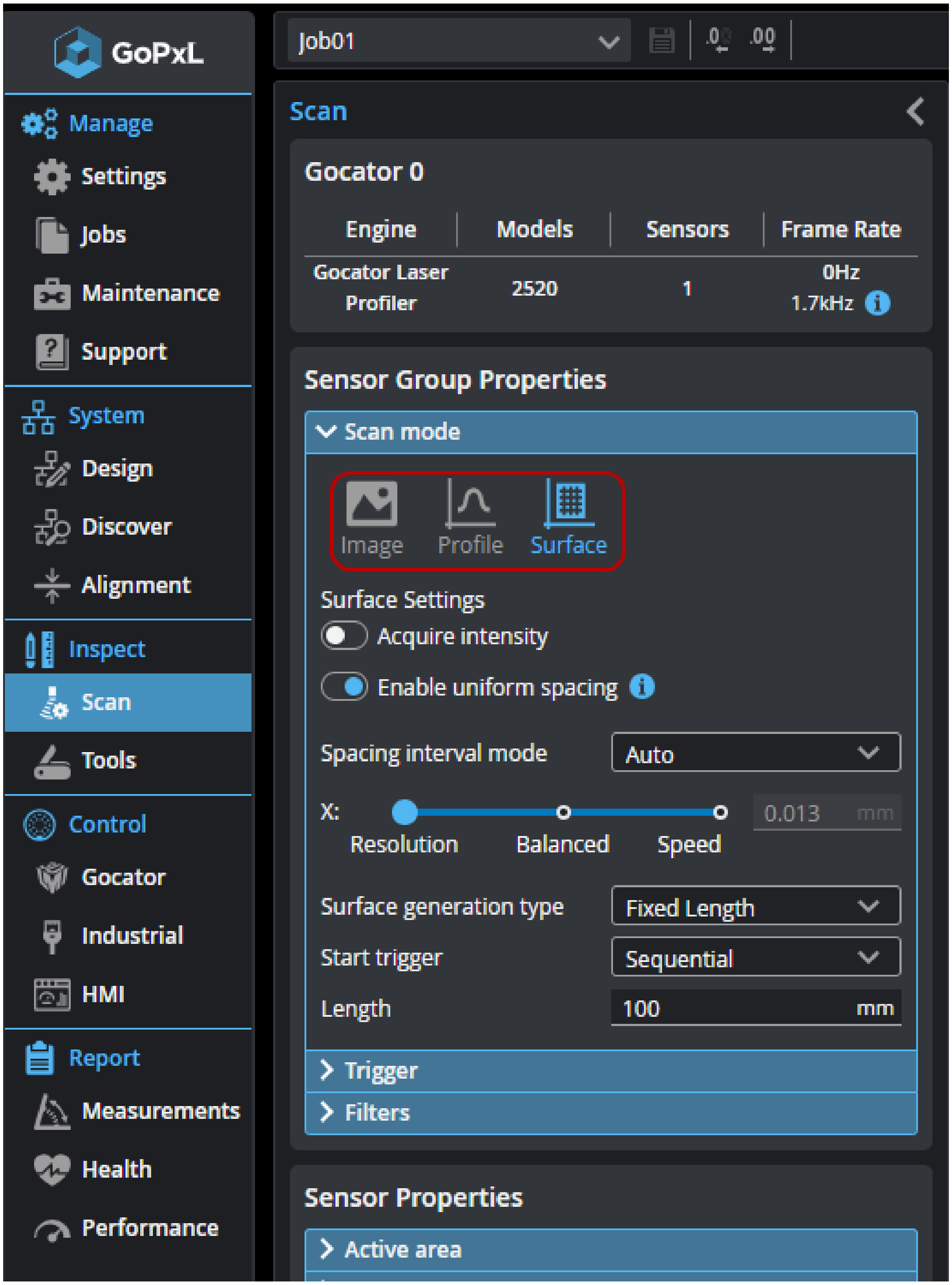


With multi-sensor systems, some settings, such as the scan mode, X spacing intervals, and triggering, apply to all of the sensors in the sensor group. These settings are in the **Sensor Group Properties** section.

Other settings, such as active area, exposure, and sub-sampling, apply to individual sensors in a sensor group. These settings are in the **Sensor Properties** section. To configure these settings in a multi-sensor system, you select a sensor using a drop-down menu.

Scan Modes and Intensity

Sensors supports one or more data acquisition modes, in addition to an image mode. You choose the scan mode on the **Acquire** > **Scan** page > **Scan Mode** section.



	If you have previously recorded scan data and have configured the input of measurement tools, changing the scan mode will display an error, saying the tools inputs are no longer available. For example, if you have scanned and recorded some Surface data, and added and configured a Surface Position measurement tool, but then switch to the Image scan mode to diagnose exposure issues, GoPxL warns you that the measurement tool's input (its data source) is missing.

Scan Mode	Description
Image	Outputs video images from the sensor. This mode is useful for configuring exposure time and troubleshooting stray light or ambient light problems.
Profile	Outputs profiles. Video images are processed internally to produce profiles and cross-sectional measurements.
Surface	Outputs Surface data. The sensor uses various methods to generate a surface (see *Surface Generation* on page 199).
Enable uniform spacing	When this option is enabled, data points are resampled to a uniform spacing (see *Uniform Data and Point Cloud Data* on page 70 for more information). Set the size of the spacing with the Uniform spacing interval settings; for more information, see *Uniform Spacing* below. Disable this option to get data from the sensor at the highest possible rate.
Acquire Intensity	When this option is enabled, an intensity value is produced for each data point. For more information on intensity, see *Intensity Output* on page 110.

Uniform Spacing

When **Enable uniform spacing** is enabled, data points are resampled to a uniform spacing. Set the size of the spacing using the **Uniform spacing interval** parameter (see below).

For more information on uniform spacing, see *Uniform Data and Point Cloud Data* on page 70.

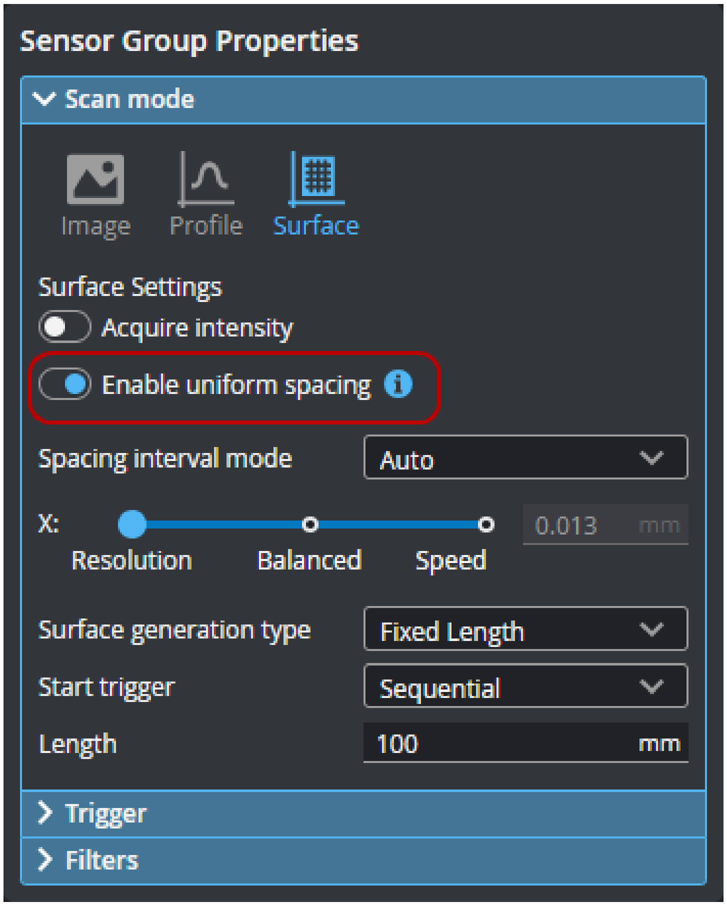
When uniform spacing is disabled, the sensor outputs unprocessed data. The sensor reports data points in (x, z) coordinate pairs. Post-processing is disabled. Only a subset of the measurement tools is available.

Disable uniform spacing to get data from the sensor at the highest possible rate.

> If you are using a layout in which sensors are angled around the Y axis in order to capture "side" data, you must uncheck **Uniform Spacing**.

The spacing interval is the user-configured spacing between data points in uniform data. A larger interval creates profiles with lower X resolution, reduces CPU usage, and potentially increases the maximum frame rate. A larger interval also reduces the data output rate.

You can set the spacing interval to one of three presets or set a custom value.



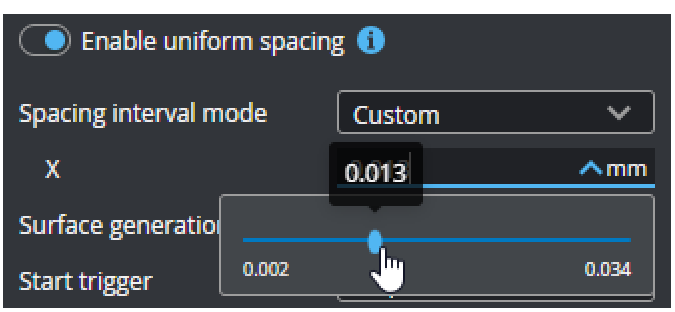
When you set **Spacing interval mode** to Auto, you use the slider to choose between the following options:

Resolution: Uses the highest X resolution within the active area as the spacing interval. This setting maximizes resolution but has higher CPU load and has the highest data output rate (that is, greatest detail).

Balanced: Uses the X resolution at the middle of the active area as the spacing interval. This setting balances CPU load, data output rate, and X resolution.

Speed: Uses the lowest X resolution within the active area as the spacing interval. This setting minimizes CPU usage and data output rate, but the profile has the lowest X resolution.

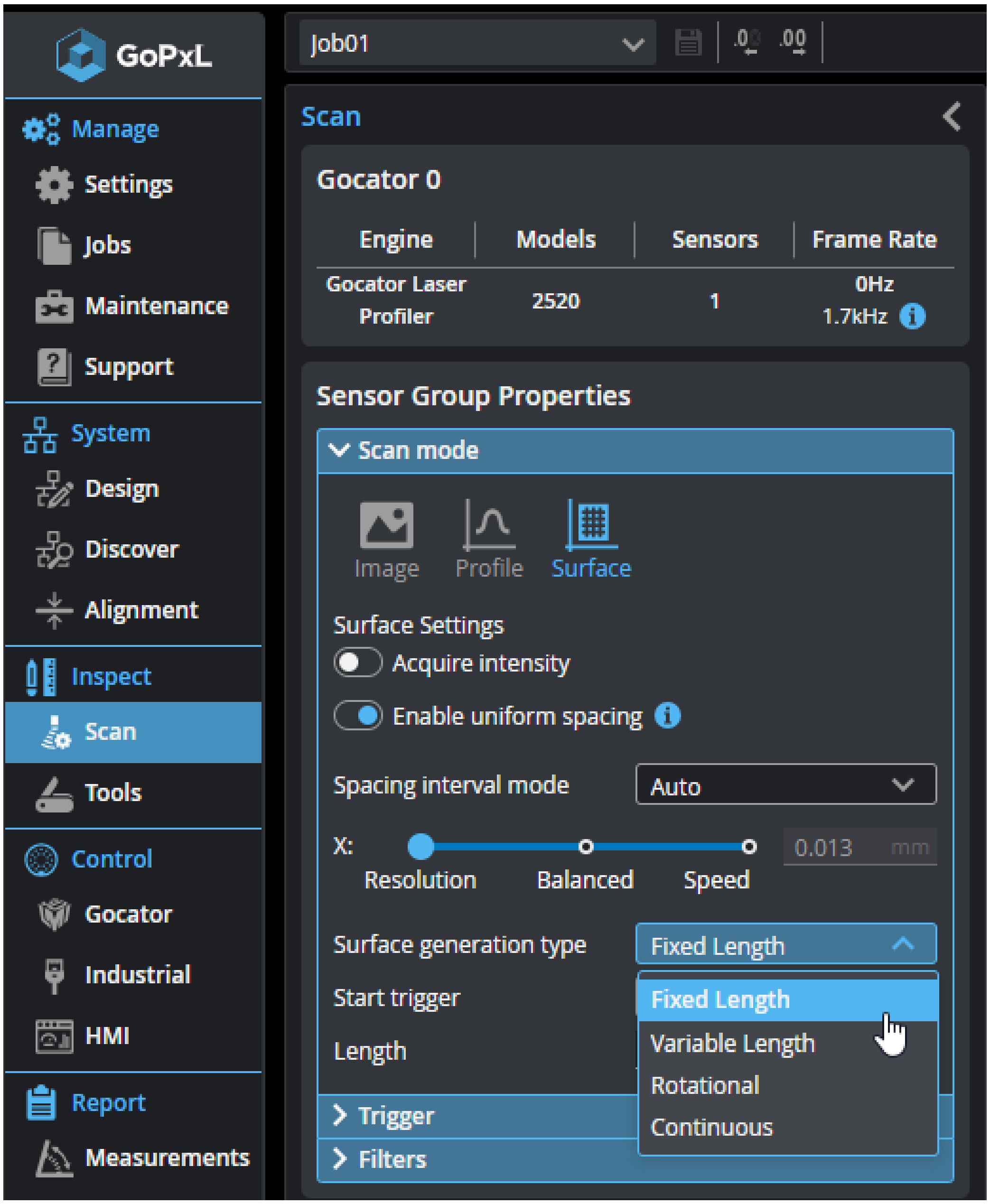
Otherwise, set **Spacing interval mode** to Custom, and set the X parameter to a specific value.



Surface Generation

GoPxL provides different ways to generate Surface data, depending on the needs of your application. You configure these Surface generation methods in the **Scan Mode** section in the **Inspect** > **Scan** page. Four modes are available in **Surface generation type**:

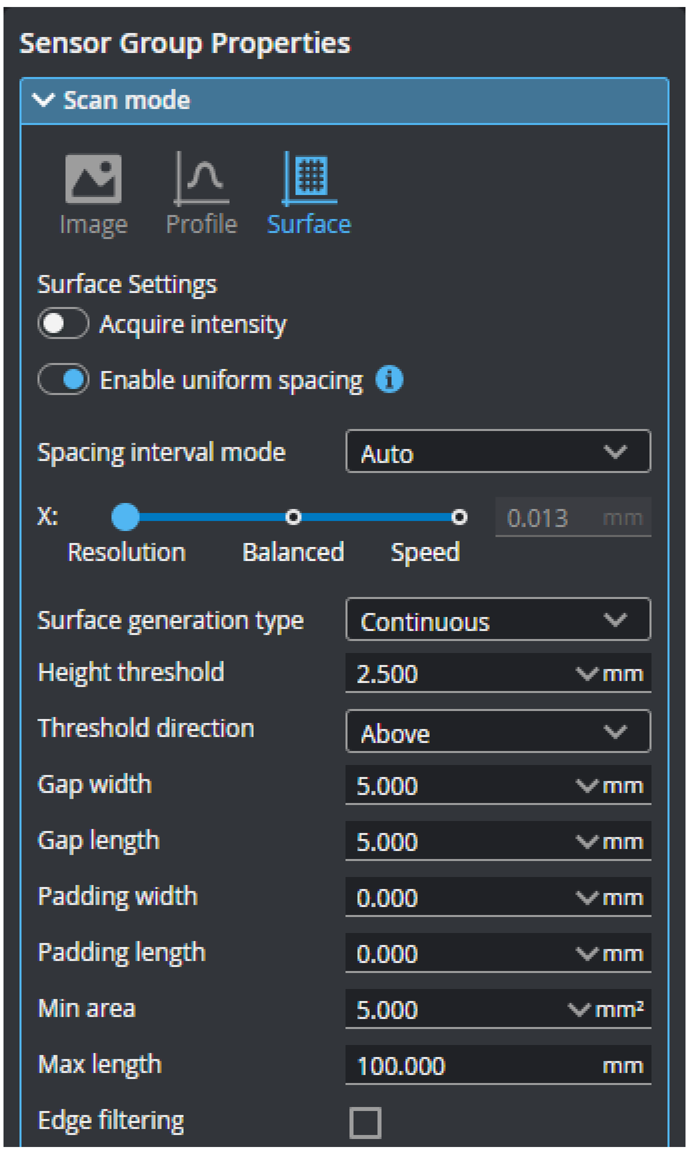
- Continuous
- Fixed Length
- Variable Length
- Rotational

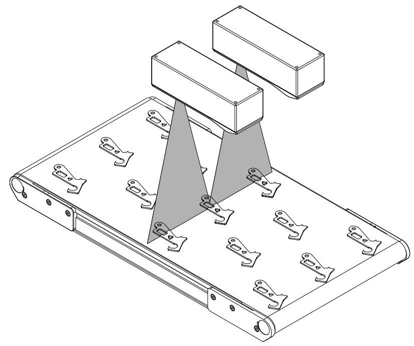


Continuous Surface Generation and Part Detection

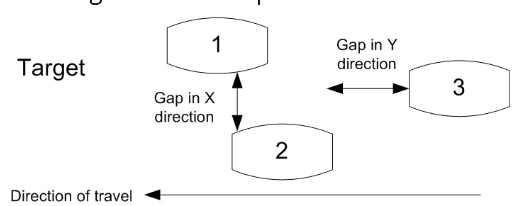
Use this method of surface generation with a transport system such as a conveyor that continuously feeds parts or material under a sensor. The parts or material must have a distinguishable start and stop edge. The sensor continuously generates surfaces of parts that are detected under the sensor. When a sensor is set to continuous Surface generation, you must configure parameters related to part detection.

> GoPxL also provides part detection through the Profile Part Detection tool. The tool uses the same parameters available with continuous generation Surface mode, but also provides diagnostics that can be useful for troubleshooting. After troubleshooting part detection using this tool, you can copy the settings from the tool to the **Scan** page. The diagnostics are output as measurements by the tool.
>
> Note that the Profile Part Detection has a **Frame of Reference** parameter not available when using continuous Surface generation.





Parameters

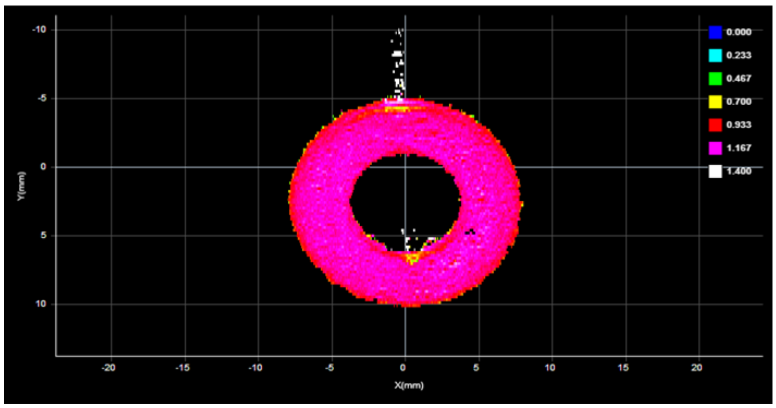
Parameter	Description
Height Threshold	Determines the height threshold for part detection. The setting for **Threshold Type** (see above) determines if parts should be detected above or below the value in **Threshold**. Above is typically used to prevent the belt surface from being detected as a part when scanning objects on a conveyor. In an Opposite layout, the threshold is applied to the difference between the top and the bottom profile. A target thinner than the threshold value is ignored, including places where only one of either top or bottom is detected. To separate parts by gated external input, set **Threshold** to the active area Z offset (that is, minimum Z position of the current active area). In the Trigger panel, set **Source** to **Time** or **Encoder**, and check the **Gate on External Input** checkbox in the **Trigger** panel; for more information, see *Triggers* on page 206.
Threshold Type	Determines if parts should be detected above or below the height threshold.
Gap Width Gap Length	**Gap Width** and **Gap Length** determine the minimum separation between objects on the X and the Y axis, respectively. If parts are closer than the gap interval, they will be merged into a single Surface output. 
Padding Width Padding Length	These parameters are useful when processing part data with third-party software such as HexSight, Halcon, etc. **Padding Width** and **Padding Length** control the amount of additional scan data output in the X and Y directions, respectively. The padding can contain data points that were outside the height threshold and excluded from the initial part detection.
Min Area	Determines the minimum area for a detected part. Set this value to a reasonable minimum in order to filter out small objects or noise.
Max Length	Determines the maximum length of the part object. When the object exceeds the maximum length, it is automatically separated into two parts. This is useful to break a long object into multiple sections and perform measurements on each

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td></td><td>section.</td></tr>
<tr><td>Edge Filtering</td><td>

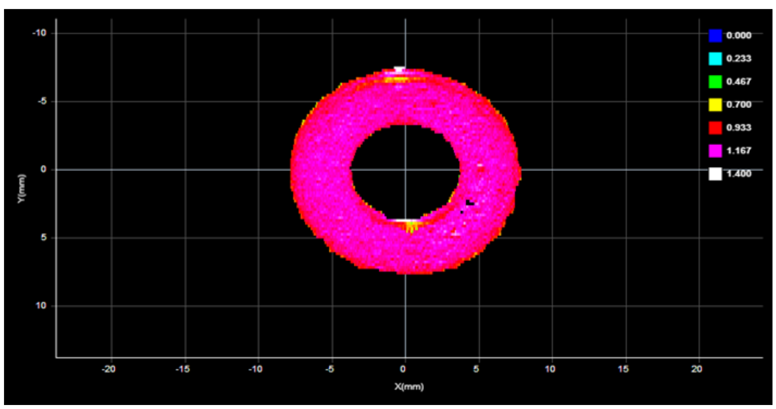
When **Edge Filtering** is enabled, additional parameters are displayed. These settings are described below.

Part scans sometimes contain noise around the edges of the target. This noise is usually caused by the sensor's light being reflected off almost vertical sides, rounded corners, etc. Edge filtering helps reduce edge noise in order to produce more accurate and repeatable volume and area measurements, as well as to improve positioning of relative measurement regions.

(The following screenshots are from a G2 sensor.)

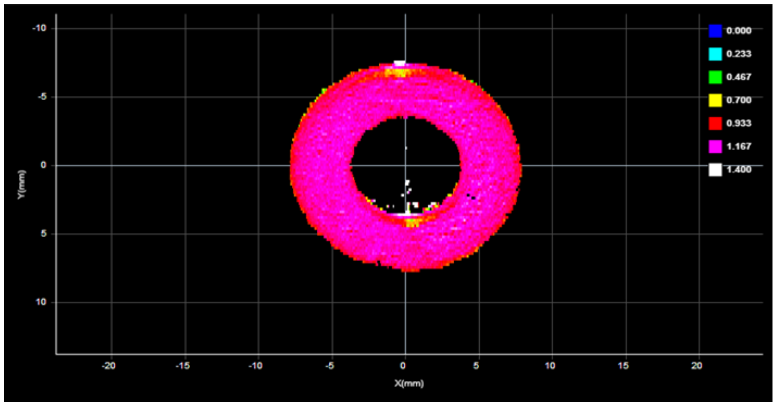


Edge filtering disabled (scan shows reflection noise)



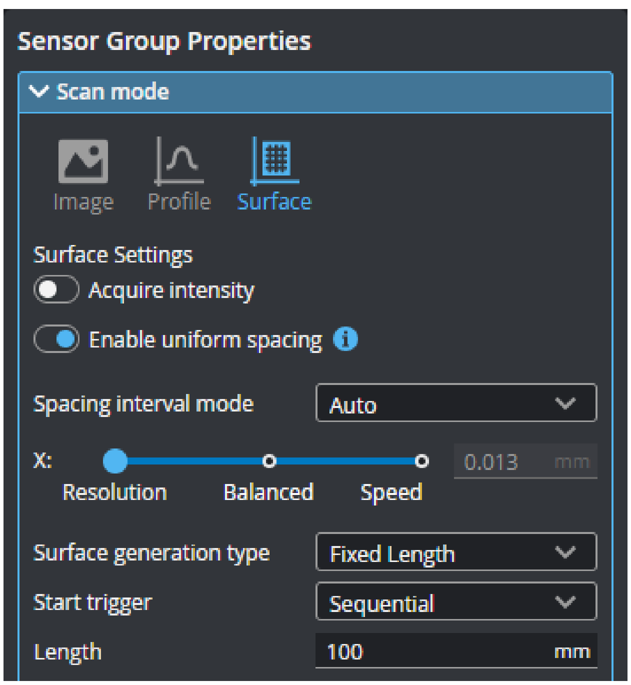
Edge filtering enabled (reflection noise eliminated or reduced)

</td></tr>
</table>

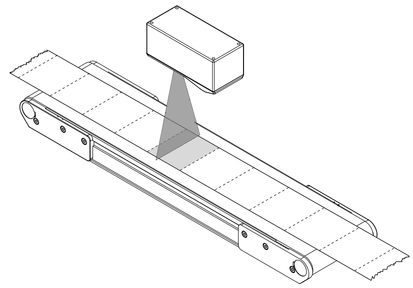
Parameter	Description
	 *Edge filtering enabled, Keep Interior enabled*
Keep Interior	The **Keep Interior** setting limits filtering to the outside edges of the target. Only displayed if **Edge filtering** is enabled.
Edge Width Edge Length	The **Edge Width** and **Edge Length** settings represent the size of the filter on the X axis and the Y axis, respectively. Only displayed if **Edge filtering** is enabled.

Fixed Length

The sensor generates surfaces of a user-configurable fixed length. You must set the length and the start trigger the sensor uses to produce Surface data.



When you set the Surface generation type to **Fixed Length**, the sensor generates surfaces of a user-configurable fixed length. Fixed Length mode is used when material (or parts) continuously pass under the sensor, typically on a conveyor, but do not have distinguishable start and stop edge.



For correct length measurement, you should ensure that motion is calibrated (that is, you have set the encoder resolution for encoder triggers or travel speed for time triggers, in **System** > **Alignment** page.

To use this method of surface generation, under **Inspect** > **Scan**, be sure to select Surface mode and configure the following parameters.

Start Trigger

The following types of start triggers are available:

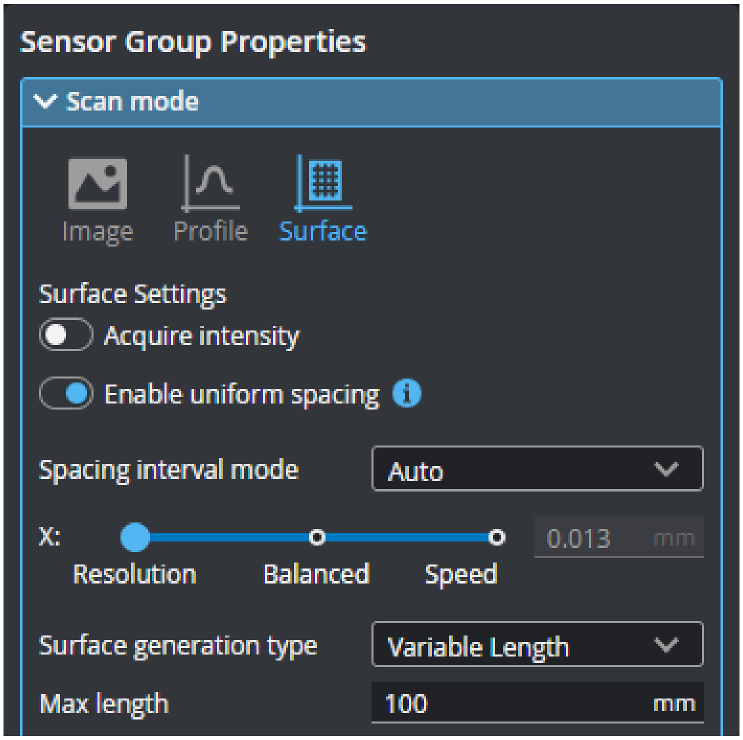
- **Sequential**: Continuously generates back-to-back fixed length surfaces.
- **External Input**: A pulse on the digital input triggers the generation of a single surface of fixed length. For more information on connecting external input to a sensor, see *Digital Input* on page 1061.
- **Software Trigger**: Allows starting fixed length surfaces on command from a PC or a PLC.

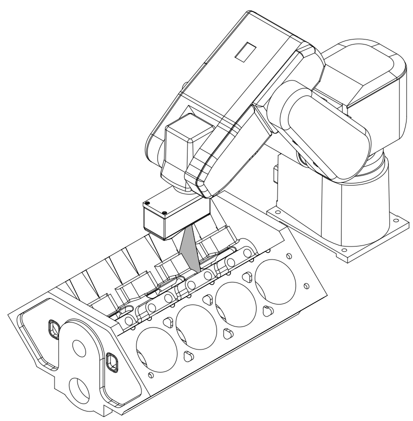
Length

The fixed length of surface to generate.

Variable Length

The sensor generates surfaces of variable length. Profiles collected while the external digital input is held high are combined to form a surface. If the value of the **Max Length** setting is reached while external input is still high, the next surface starts immediately with the next profile. This mode is typically used in robot-mounted applications, for example, measuring the dimensions of different parts on an engine block.

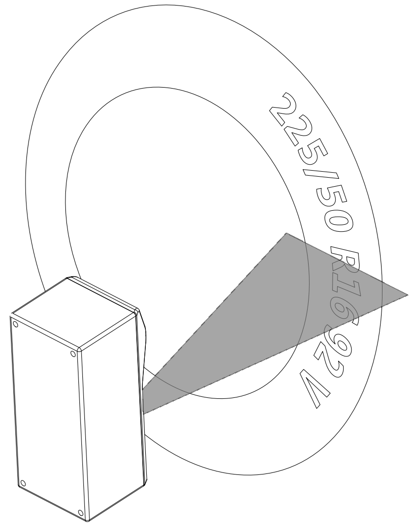


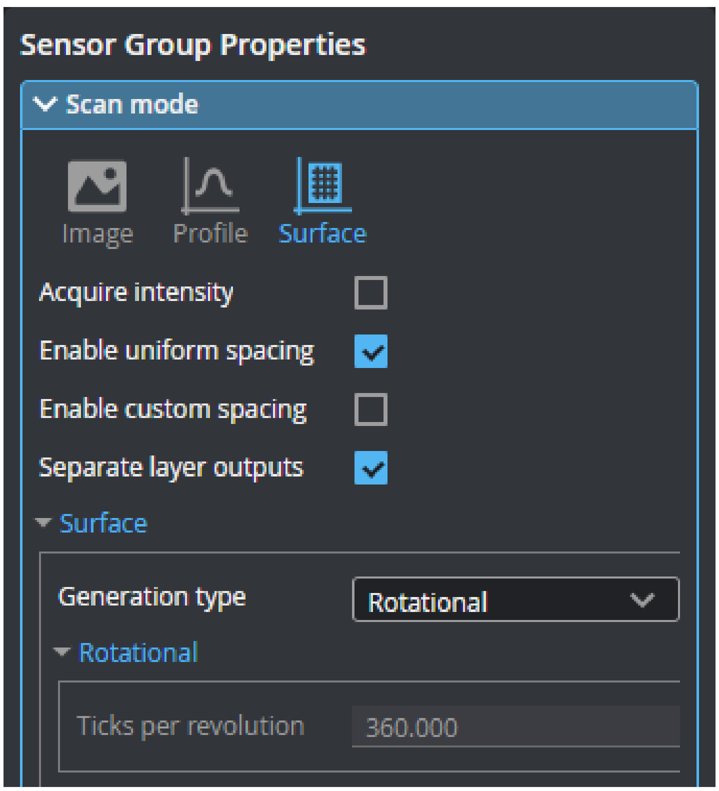


For correct length measurement, you should ensure that motion is calibrated (that is, encoder resolution for encoder triggers or travel speed for time triggers).

Rotational

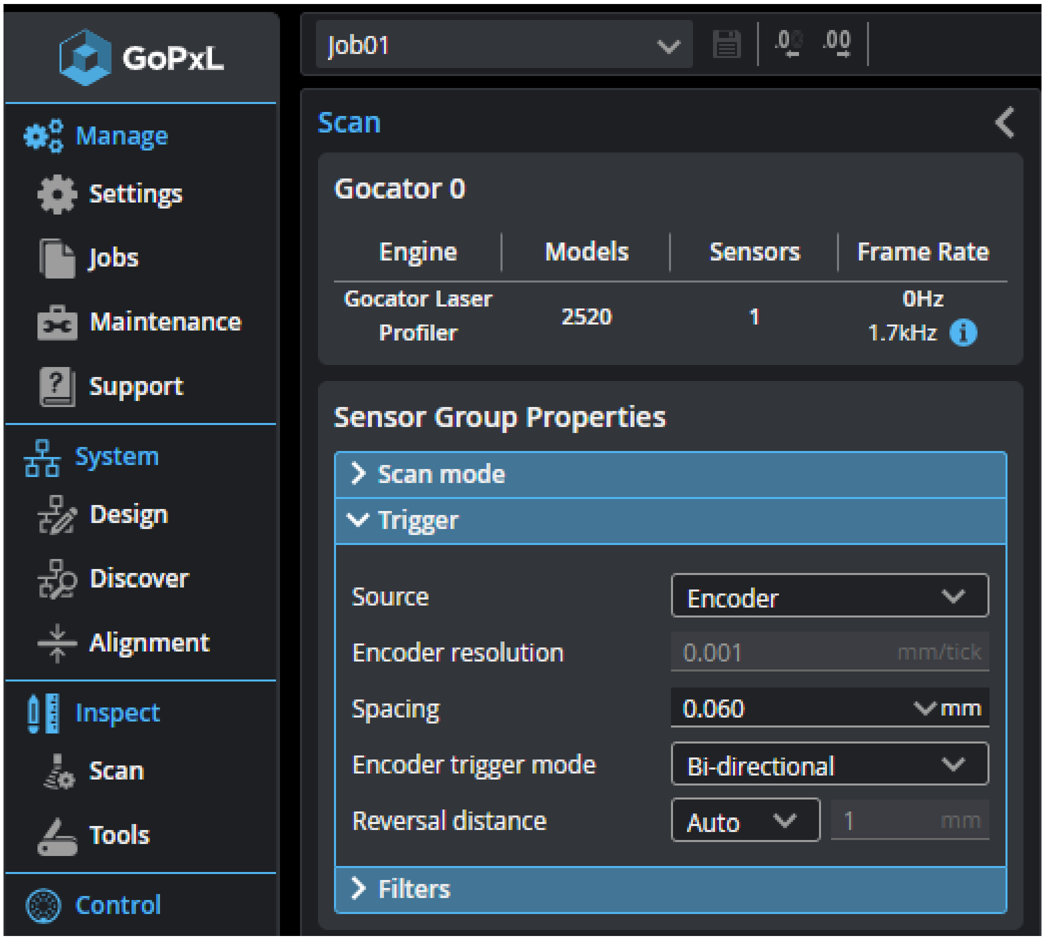
The sensor reorders profiles within a surface to be aligned with the encoder's index pulse. That is, regardless of the radial position the sensor is started at, the generated surface always starts at the position of the index pulse. If the index pulse is not detected and the rotation circumference is met, the surface is dropped and the Encoder Index Drop indicator will be incremented. This mode is typically used in applications where measurements of circular objects or shafts need to be taken, such as tire tread inspection, or label positioning on bottles.





Triggers

A trigger is an event that causes a sensor to take a single image. You configure triggers on the **Acquire** > **Scan** page.



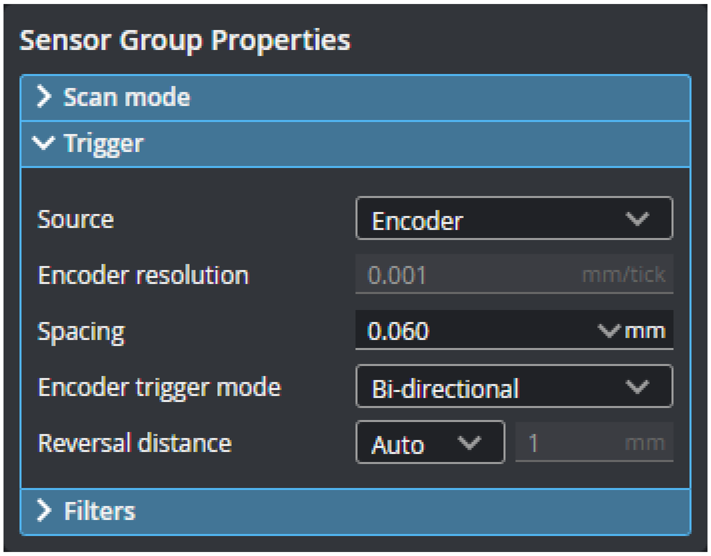
When a trigger is processed, the sensor's laser or LED light strobes and the camera exposes to produce an image. The resulting image is processed inside the sensor to yield scan data. The data can then be used for measurement.

At the top-right of the **Scan** page, the maximum frame rate is displayed, based on the exposure values, active area, the number of projection patterns required, and so on.

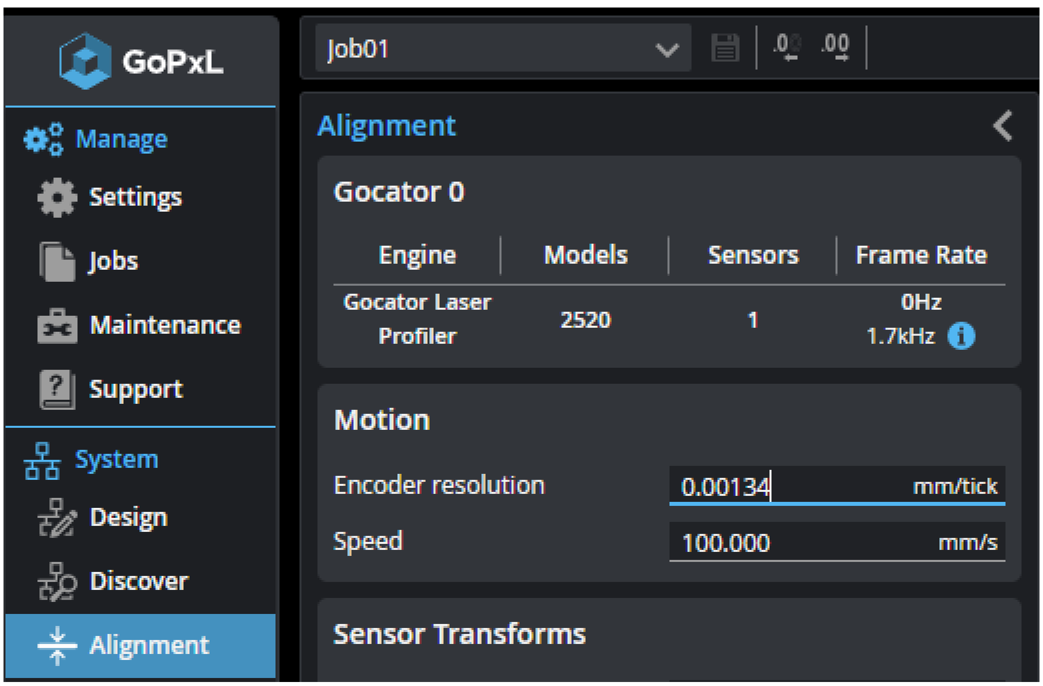
The sensor can be triggered by one of the sources described below.

> If the sensor is connected to a Master 400 or higher, encoder and digital (external) input signals over the IO cordset are *ignored*. The sensor instead receives these signals from the Master; for encoder and digital input pinouts on Masters, see the section corresponding to your Master in *Master Network Controllers* on page 1066.
>
> If the sensor is connected to a Master 100 (or no Master is used), the sensor receives signals over the IO cordset. For information on connecting encoder and digital input signals to a sensor in these cases, see *Encoder Input* on page 1062 and *Digital Input* on page 1061, respectively.

When using encoder triggering (available with line profilers), the current encoder resolution is displayed under the **Source** drop-down as a reference when setting trigger spacing.



To set the encoder resolution, go to the **System** > **Alignment** page and set the value in **Encoder Resolution**.



Trigger source descriptions

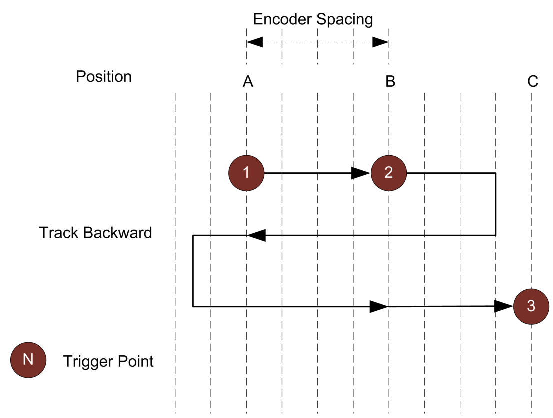
Trigger Source	Description
Time	Sensors have an internal clock that can be used to generate fixed-frequency triggers. The external input can be used to enable or disable the time triggers.

<table>
<tr><th>Trigger Source</th><th>Description</th></tr>
<tr><td>Encoder</td><td>

An encoder can be connected to provide triggers in response to motion. Three encoder triggering behaviors are supported. These behaviors are set using the **Behavior** setting.

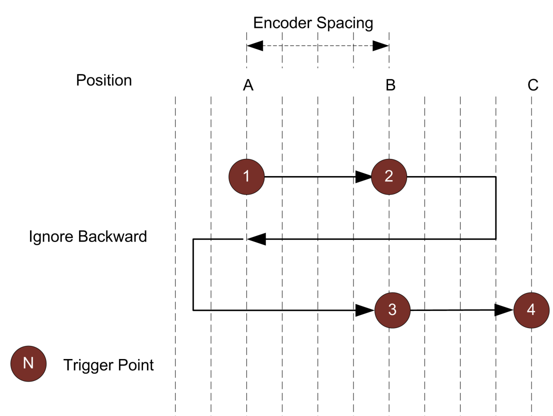
Track Backward

A scan is triggered when the target object moves forward. If the target object moves backward, it must move forward by at least the distance that the target travelled backward (this distance backward is "tracked"), plus one encoder spacing, to trigger the next scan.



Ignore Backward

A scan is triggered only when the target object moves forward. If the target object moves backward, it must move forward by at least the distance of one encoder spacing to trigger the next scan.

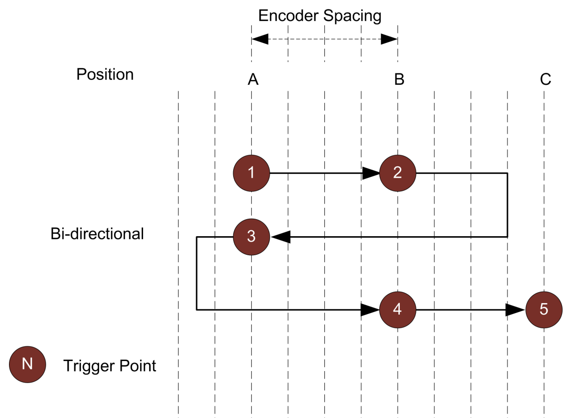


Bi-directional

A scan is triggered when the target object moves forward or backward.

</td></tr>
</table>

Trigger Source	Description



Trigger Source	Description
	When triggers are received at a frequency higher than the maximum frame rate, some triggers may not be accepted. Use the **Trigger Drops** indicator in the **Health** panel on the **Reports** page to check for this condition; for more information, see *Reporting* on page 821. The external input can be used to enable or disable the encoder triggers. For information on the maximum encoder rate, see *Maximum Encoder Rate* on page 213. To verify that the sensor is receiving encoder signals, check whether the **Encoder Value** indicator is changing in the **Health** panel on the **Reports** page (for more information, see *Reporting* on page 821).
External Input	A digital input can provide triggers in response to external events (for example, a photocell). The external input triggers on the rising edge of the signal. When triggers are received at a frequency higher than the maximum frame rate, some triggers may not be accepted. Use the **Trigger Drops** indicator in the **Health** panel on the **Reports** page to check for this condition; for more information, see *Reporting* on page 821. For information on the maximum input trigger rate, see *Maximum Input Trigger Rate* on page 213.
Software	A network command can be used to send a software trigger. For more information, *Protocols (PLCs and other hardware)* on page 884.

Depending on the setup and measurement tools used, the CPU utilization may exceed 100%, which reduces the overall acquisition speed.

For examples of typical real-world scenarios, see *Trigger Examples* on page 212.

Trigger Settings

After you choose the trigger source, configure the other parameters in the **Trigger** section on the **Scan** page.

Parameter	Trigger Source	Description
Source	n/a	Selects the trigger source (Time, Encoder, External Input, or Software).
Frame Rate	Time	Controls the frame rate. Fractional values are supported. For

Parameter	Trigger Source	Description
		example, 0.1 can be entered to run at 1 frame every 10 seconds. Choose one of the following: **Trigger at maximum frame rate**: The sensor triggers at the maximum available frame rate. The frame rate is displayed after this option and also toward the top of the configuration area on the **Scan** page. **Manually configure**: Lets you provide a custom frame rate.
Encoder resolution	Encoder	A read-only field displaying the current encoder resolution. You set this on the **System** > **Alignment** page, or calibrate it during alignment.
Spacing	Encoder	Specifies the distance, along the Y axis, between triggers. Internally the sensor rounds the spacing to a multiple of the encoder resolution.
Encoder trigger mode	Encoder	Specifies how the sensor is triggered when the target moves. Can be Track Backward, Ignore Backward, or Bi-Directional. For more information, see Encoder in *Trigger source descriptions* on page 208.
Reversal Distance	Encoder	When **Encoder trigger mode** is set to **Bi-Directional**, use this setting to ignore jitter or vibrations in your transport system by specifying what distance the target must travel before a direction change is triggered. One of the following: **Auto**: The distance is automatically set. **Custom**: Set the distance (in millimeters). Various functions in the sensor depend on this value to explicitly determine the point where direction change is triggered. Set this value larger than the maximum vibrations you see in your transport system.
Gate on External Input	Time, Encoder, Software	You can use external input to enable or disable data acquisition in a sensor. When this option is enabled, the sensor will respond to time, encoder, or software triggers *only* when the external input is asserted. This setting is only displayed when the sensor is in Profile mode. See *Digital Input* on page 1061 for more information on connecting external input to sensors.
Units	External Input	Specifies the unit.
Trigger Delay	External Input	Specifies the amount of time or the distance the sensor waits before producing a frame after the external input is activated. This is used to compensate for the positional difference between the source of the external input trigger (for example, a photocell) and the sensor.

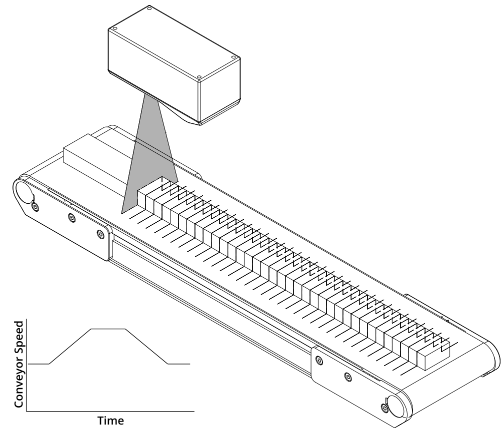
Depending on the surface generation settings, some trigger options may not be available.

Trigger Examples

Example: Encoder + Conveyor

Encoder triggering is used to perform profile measurements at a uniform spacing.

The speed of the conveyor can vary while the object is being measured; an encoder ensures that the trigger spacing is consistent, independent of conveyor speed.

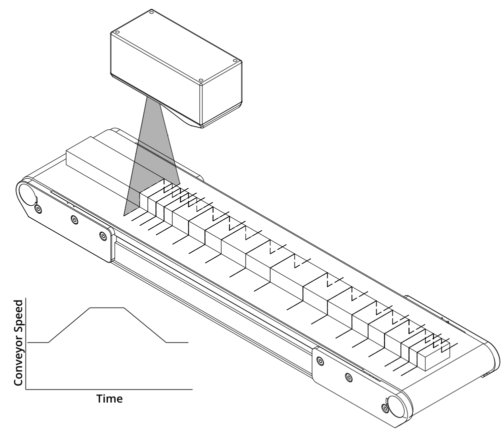


Example: Time + Conveyor

Time triggering can be used instead of encoder triggering to perform profile measurements at a fixed frequency.

Spacing will be non-uniform if the speed of the conveyor varies while the object is being measured.

It is strongly recommended to use an encoder with transport-based systems due to the difficulty in maintaining constant transport velocity.

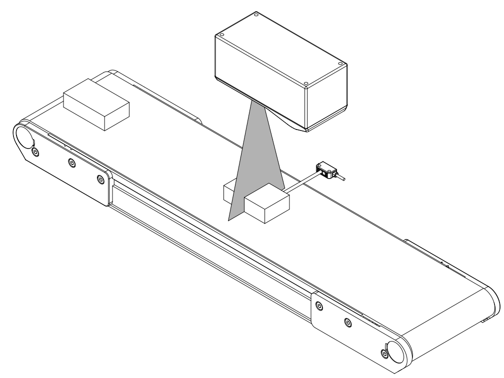


Example: External Input + Conveyor

External input triggering can be used to produce a snapshot for profile measurement.

For example, a photocell can be connected as an external input to generate a trigger pulse when a target object has moved into position.

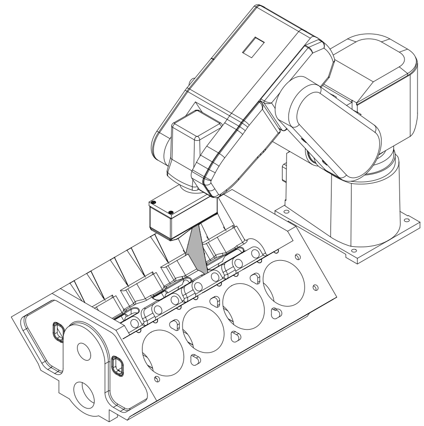
An external input can also be used to gate the trigger signals when time or encoder triggering is used. For example, a photocell could generate a series of trigger pulses as long as there is a target in position.



Example: Software Trigger + Robot Arm

Software triggering can be used to produce a snapshot for profile measurement.

A software trigger can be used in systems that use external software to control the activities of system components.



Maximum Input Trigger Rate

> The maximum external input trigger rate in a system including Master 810 or 2410 is 20 kHz.

When using a standalone sensor or a sensor connected to a Master 100, the maximum trigger rate is 32 kHz. This rate is limited by the fall time of the signal, which depends on the Vin and duty cycles. To achieve the maximum trigger rate, the Vin and duty cycles must be adjusted as follows:

Maximum Speed	Vin	Maximum Duty Cycle
32 kHz	3.3 V	88%
32 kHz	5 V	56%
32 kHz	7 V	44%
32 kHz	10 V	34%

At 50% duty cycle, the maximum trigger rates are as follows:

Vin	Maximum Speed
3.3 V	34 kHz
5 V	34 kHz
10 V	22 kHz

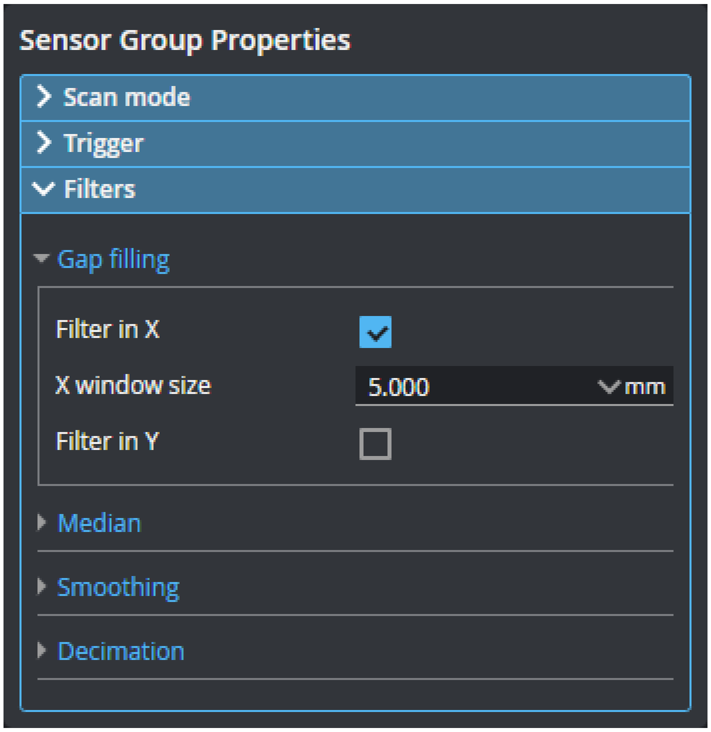
Maximum Encoder Rate

On a standalone sensor, with the encoder directly wired into the I/O port or through a Master 100, the maximum encoder rate is about 1 MHz.

For sensors connected through a Master 810 or higher, with the encoder signal supplied to the Master, the maximum rate is about 300 kHz.

Filters

The filters in the **Filters** panel are used to process scan data along the X or Y axis to remove noise or clean it up *before* it is used by measurement tools or is output. Using the filters can help you get more repeatable measurements. The filters are only available when uniform spacing is enabled.



The following filters are available in the Filter panel.

Filters

Name	Description
Gap Filling	Fills in missing data using information from the nearest neighbor data points, for example, when data is missing due to occlusions. Gap filling also fills gaps where no data is detected, which can be due to low surface reflectivity, for example dark or specular surface areas, or to actual gaps in the surface. The values in **Filter in X** and **Filter in Y** represent the maximum gaps the sensor will fill. Wider gaps are not filled. Gap filling works by filling in missing data points using either the lowest values from the nearest neighbors or linear interpolation between neighboring values (depending on the Z difference between neighboring values), in the specified window. The sensor can fill gaps along both the X axis and the Y axis. In Profile mode, gap filling is limited to the X axis.
Median	Substitutes the value of a data point with the median calculated within the window or windows set in **X Window Size** or **Y Window Size** around the data point. If the number of valid (non-null) data points in the window is even, the median value is simply the value in the center of the sorted list of values. If the number of valid points is odd, the average of the two values in the center is used instead. Missing data points will not be filled with the median value calculated from data points in the neighborhood. With an odd window size, the output is at the center of the window. With an even window size, the output is 0.5 pixels to the right of the center (that is, using

Name	Description
	window / 2-1 values from the left, and window / 2 from the right.
Smoothing	Substitutes a data point value with the mean value of that data point and its nearest neighbors within the window or windows set in **X Window Size** or **Y Window Size**. X smoothing works by calculating a moving average across samples within the same profile. Y smoothing works by calculating a moving average in the direction of travel at each X location. If both X and Y smoothing are enabled, the data is smoothed along X axis first, then along the Y axis. Missing data points will not be filled with the mean value calculated from data points in the neighborhood.
Decimation	Decimation reduces the number of data points along the X or Y axis by choosing data points at the end of a specified window around the data point. For example, by setting **X Window Size** to 0.2, only points every 0.2 millimeters will be used. The filter generates points starting from the leftmost edge of the scan data, stepping in equal steps away from that side.

Tool-based filtering is also available on the **Tools** page. The filters described here and in the filter tool topics use the same algorithms.

Using tool-based filtering provides various advantages:

- Additional filters not available in the **Filters** panel. (This mostly applies to Surface filters.)
- Choosing which tools used in a job take filtered data as input. That is, you can decide to have some tools running on unfiltered data and other tools on filtered data.

For more information on tool-based filters, see *Surface Filter* on page 552 (Surface-based) and *Profile Filter* on page 335 (Profile-based).

The filter window sizes are specified in millimeters (and additionally, in the tools, in data points). To calculate the number of data points that a window covers when the units are millimeters, use the following calculation:

- User-specified window size divided by the X spacing interval (that is, the number of millimeters per point) on the **Spacing** tab in the **Sensor** panel. (For more information on spacing intervals, see *Uniform Spacing* on page 197.)
- *With the exception of the gap filling filter*, round the result of the division to the nearest integer value. With the gap filling filter, filling is performed within the provided window size.

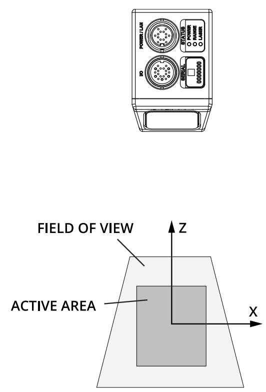
For example, if you set the size of the filter's window to a value between 1.5 mm and 2.49 mm (inclusively), and the X spacing interval is set to 1 mm, the filter covers 2 data points. A filter window size from 2.5 mm to 3.49 mm results in a filter covering 3 data points.

Active Area

Active area refers to the region within the sensor's maximum field of view that is used for data acquisition.

By default, the active area covers the sensor's entire field of view. By reducing the active area, the sensor can operate at higher speeds. You can also reduce the active area to exclude areas that are affected by ambient light, or to avoid supports or parts of a transport system (such as a conveyor).

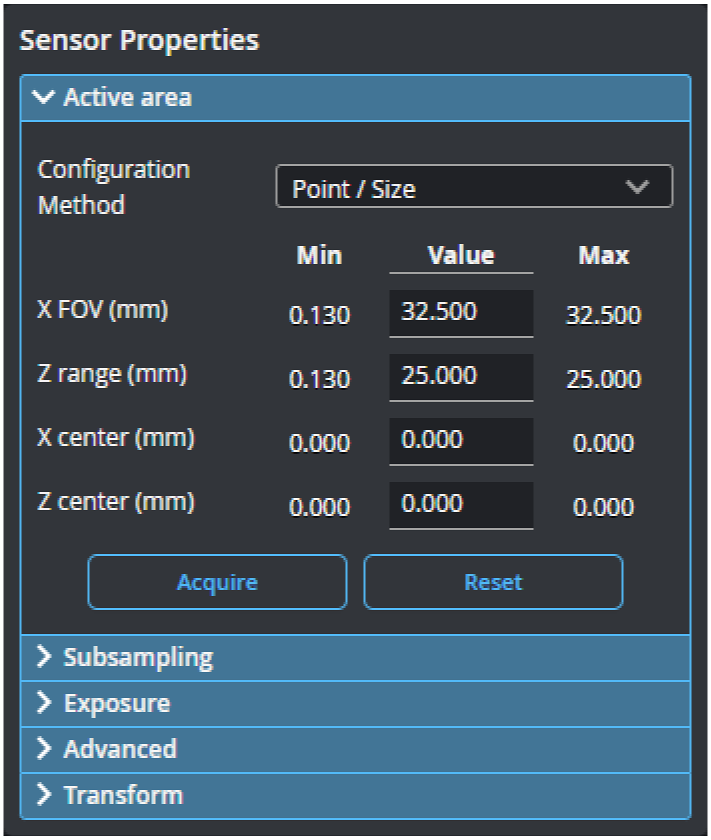
Active area is specified in sensor coordinates, rather than in system coordinates. As a result, if the sensor is already alignment calibrated, press the **Acquire** button to display uncalibrated data before configuring the active area.See *Coordinate Systems* on page 65 for more information on sensor and system coordinates.



Active area in an unaligned G2 sensor's scan area.

You set the active area in the **Inspect** > **Scan** page, in the **Active area** section under **Sensor Properties**. In multi-sensor systems, you set active area for each sensor individually.

Two active area configuration methods are available in the **Configuration Method** parameter: Maximum / Minimum and Point / Size (for the parameters related to each method, see the tables below). For instructions on setting active area, see *To set a sensor's active area* on the next page.



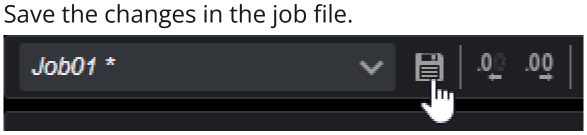
To set a sensor's active area

1. Go to **Inspect** > **Scan** page.
2. (Optional) If you are configuring the active area of a sensor in a multi-sensor system, select the correct sensor.
3. (Optional) Place a sample target in the sensor's scan area, and in the **Active Area** section, click the **Acquire** button to get some scan data.

   Getting a scan while setting the active area can help you determine the necessary size and position of the active area.
4. Choose the active area configuration method in **Configuration Method**.
   One of the following: Point / Size and Minimum / Maximum.
5. Set the active area.
   Adjust the active area graphically in the data viewer using your mouse or enter the values manually in the fields.

   Point / Size: For parameter descriptions, see *Active area parameters - Point / Size method* on the next page.

   Minimum / Maximum: For parameter descriptions, see *Active area parameters - Maximum / Minimum method* on the next page.
6. Save the changes in the job file.

To reset the active area

- Click the **Reset** button.

Active area parameters - Point / Size method

Name	Description
X FOV (mm)	The width of the active area along the X axis. This sets the effective field of view of the sensor.
Z range (mm)	The height of the active area along the Z axis. This sets the effective measurement range of the sensor.
X center (mm) Z center (mm)	The X and Z position in the entire sensor field of view of the *center* of the active area.

Active area parameters - Maximum / Minimum method

Name	Description
Min X Max X	The minimum and maximum X values.
Min Z Max Z	The minimum and maximum Z values.

> Scanning devices are usually more accurate at the near end of their measurement range. If your application requires a measurement range that is small compared to the maximum measurement range of the sensor, mount the sensor so that the active area can be defined at the near end of the measurement range.

Subsampling

Subsampling reduces the number of camera columns or rows used for data acquisition to increase speed or reduce CPU usage, while maintaining the sensor's field of view and measurement range. You can set sub-sampling independently for the X axis and Z axis. You set subsampling on the **Inspect** > **Scan** page, in the **Sensor Properties** section.

When set to a value of less than 1, the **X** subsampling setting decreases the profile's X resolution to lower sensor CPU usage.

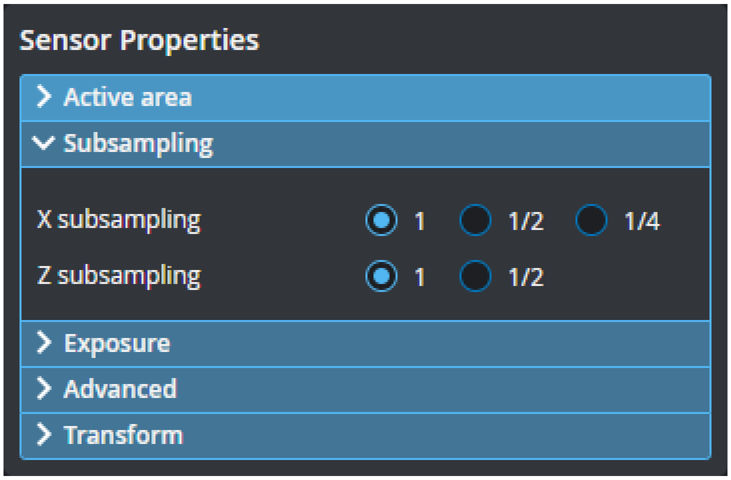
> The **1/4** subsampling setting is not available on Gocator 2100 series sensors.

When set to a value of less than 1, the **Z** subsampling setting decreases the profile's Z resolution, which increases speed. The **Z** setting works by reducing the number of image rows used for profiling.

> Because Z subsampling reduces vertical image resolution, you may need to adjust spot detection parameters when it is enabled.

Subsampling values are expressed as fractions in the GoPxL interface. For example, an X subsampling value of 1/2 means that every second camera column will be used for profiling. When the parameter is set to 1, no subsampling occurs.

Sub-sampling is disabled by default on most sensors (set to 1). As of GoPxL 1.1, Z sub-sampling is enabled by default on Gocator 2600 sensors to provide a series-specific, optimal balance between speed and data quality.



The **CPU** load shown at the bottom of the interface displays how much the CPU is being used.

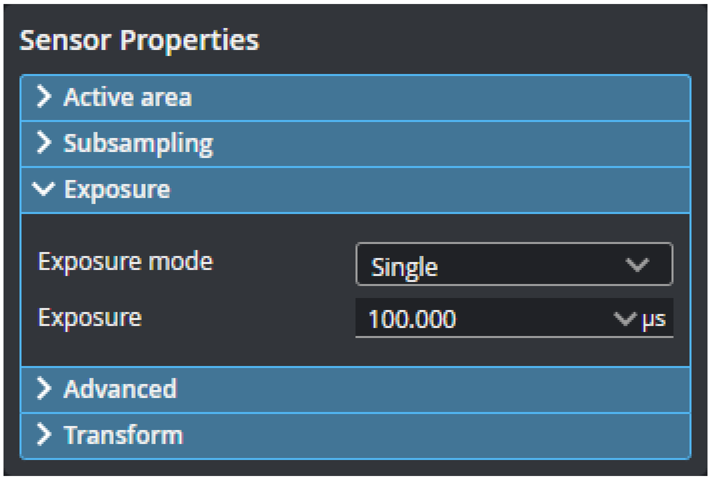
Exposure

Exposure determines the duration of camera and light-source on-time (or pulse width), given in microseconds (µs). Longer exposures can be helpful to detect light on dark or distant surfaces, but increasing exposure time decreases the maximum speed. Different target surfaces may require different exposures for optimal results. In general, shiny and mirror-like surfaces require a short pulse width and low-gloss surfaces a high pulse width.

Sensors provide two or three exposure modes, depending on the model, for the flexibility needed to scan different types of target surfaces.

To properly set exposure, you can acquire scan data in Image mode, and examine the resulting image to confirm whether the features or flaws you need to measure will be well presented in the scan data. For more information on Image mode and the information it provides for configuring exposure, see *Image Mode* on page 92.

You configure exposure in the **Exposure** panel on the **Inspect** > **Scan** page, in the **Sensor Properties** section.

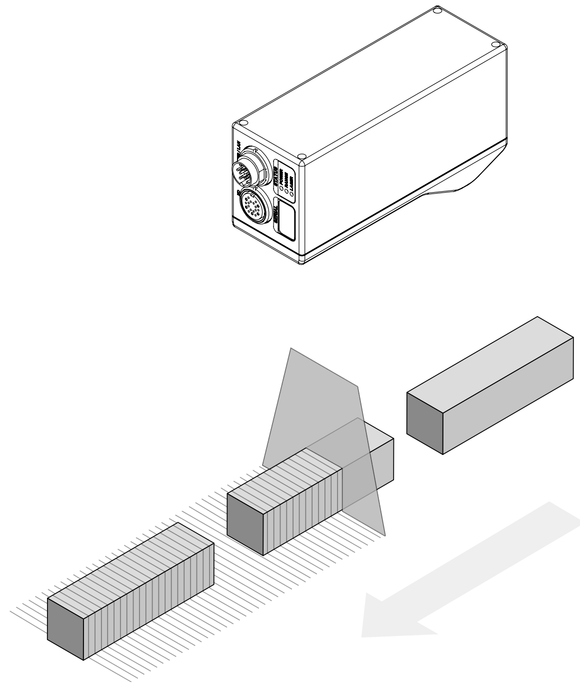


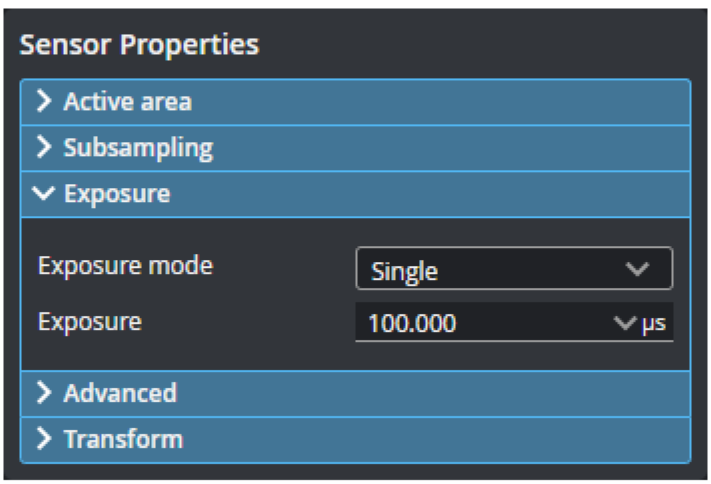
When exposure is set to Dynamic or Multiple, additional parameters are displayed.

Exposure Mode	Description
Single	Uses a single exposure duration for each frame. Use this when the surface is roughly uniform and is the same for all targets. For more information, see *Single Exposure* below.
Dynamic	Automatically adjusts the exposure after each profile based on an analysis of the preceding frame. Used when the target surface varies between scans. For more information, see *Dynamic Exposure* on page 223.
Multiple	Uses multiple exposures to create a single profile. Used when the target surface has a varying reflectance within a single profile. For more information, see *Multiple Exposure* on the next page.

Single Exposure

The sensor uses a fixed exposure in every scan. Single exposure is used when the target surface is uniform and is the same for all targets.





Setting	Description
Exposure Mode	Set to Single.
Exposure	The length of the light emitter's on-time, in microseconds.

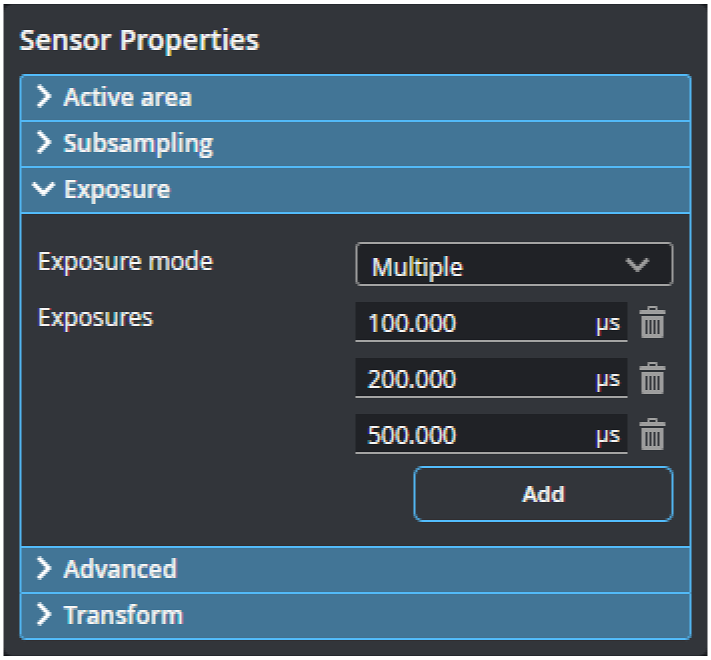
Multiple Exposure

The sensor can combine data from multiple exposures to create a single profile (G2, G4, and G5) or 3D scan (all families). Multiple exposures can be used to increase the ability to detect light and dark materials that are in the field of view simultaneously. Note that you can optionally output the scan data for the different exposures as separate profiles or surfaces, but only if uniform spacing is disabled.

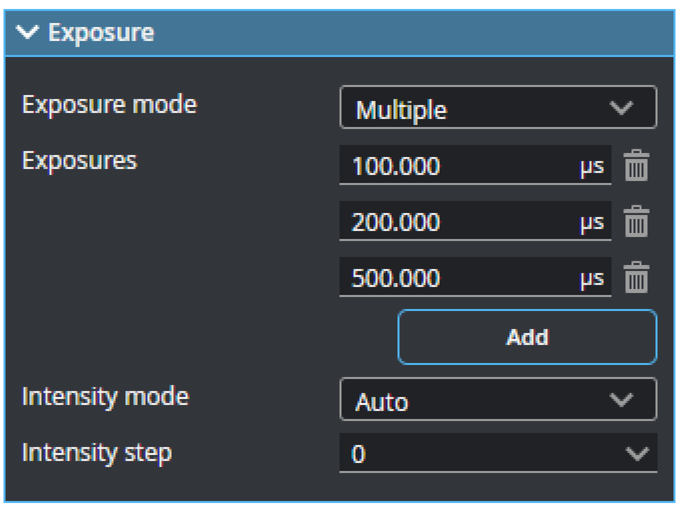
You can define up to five exposures (three for G3 sensors), with each set to a different exposure level. You add exposures using the **Add** button, and remove them using the trashcan icon.

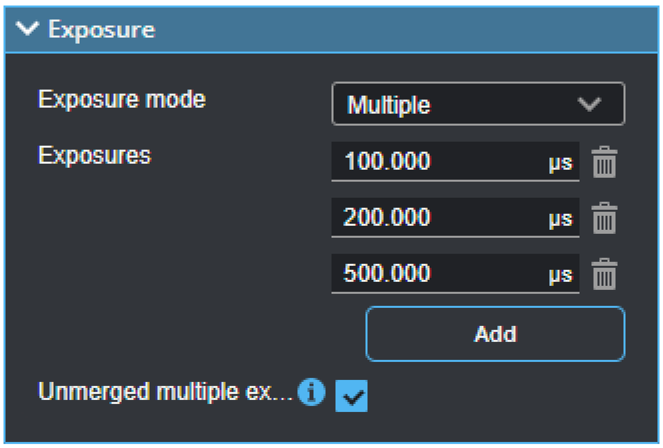
For each exposure, the sensor will perform a complete scan at the current frame rate making the effective frame rate slower. For example, if two exposures are selected, then the speed will be half of the single exposure frame rate. The sensor will perform a complete multi-exposure scan for each external input or encoder trigger.

The resulting scan data is a composite created by combing data collected with different exposures. The sensor will choose profile data that is available from the lowest-numbered exposure step. It is recommended to use a larger exposure for higher-numbered steps.



If you have enabled intensity in the **Scan Mode** panel, you can use **Intensity step** to choose which exposure the sensor uses for acquiring intensity data. This lets you choose the exposure that produces the best image for intensity data.

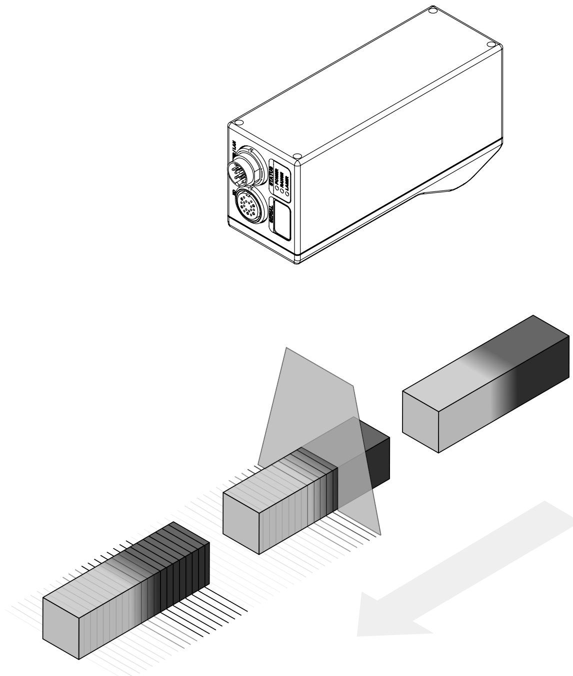


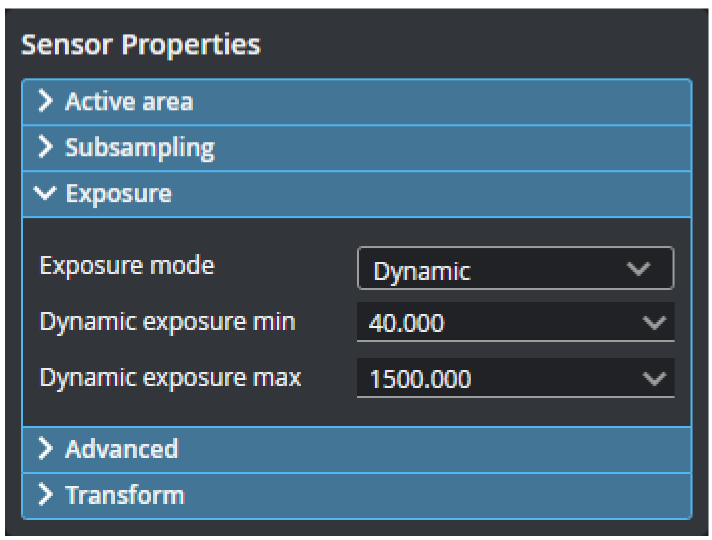
Setting	Description
Exposure Mode	Single, Multiple, or Dynamic. Set to Multiple.
Intensity mode	Auto or legacy mode. Leave this at its default value.
Intensity step	Chooses which exposure is used for intensity data. Choose the step that provides the best data.
Unmerged Multiple Exposure Output	This setting is only available when uniform spacing is disabled (for more information, see *Uniform Spacing* on page 197). When this setting is enabled, the sensor produces separate profiles or surfaces for each exposure, and the data is output as an array. When it is disabled, the profiles or surfaces of the different exposures are combined. 

Dynamic Exposure

When **Exposure mode** is set to Dynamica, the sensor automatically uses previous frames of Profile scan data to adjust the exposure for subsequent exposures to yield the best data. Use this mode when the target surface changes from exposure to exposure.

> You can tune settings that control the exposure that is chosen by dynamic exposure in the Material tab.

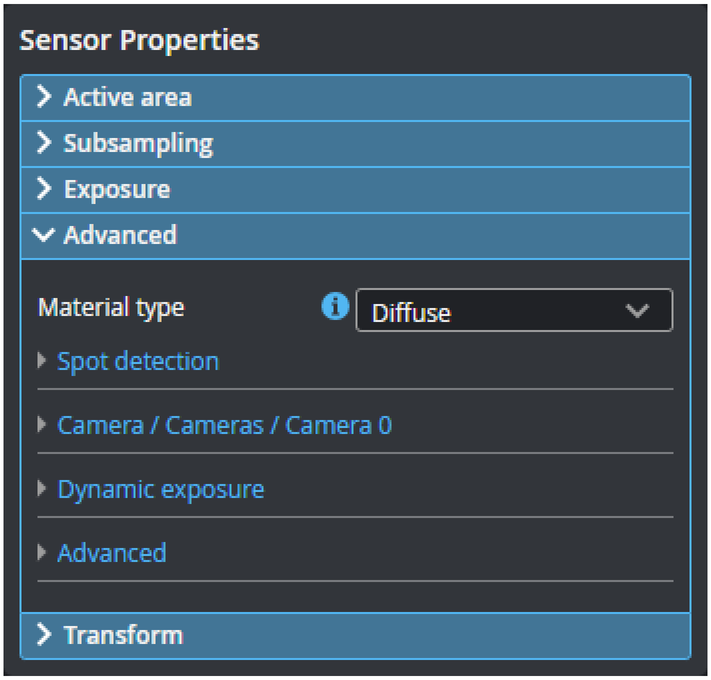




Setting	Description
Exposure Mode	Set to Dynamic.
Dynamic exposure min	Limits the sensor's exposure values to the provided minimum and maximum.
Dynamic exposure max	

Advanced Settings

The settings in the **Advanced** section on the **Inspect** > **Scan** page let you configure material characteristics, camera gain, and dynamic exposure.



Material Type and Spot Detection

You can configure data acquisition to suit different types of target materials. This helps maximize the number of useful profile points produced. For many targets, changing the setting is not necessary, but it can make a great difference with others.

You can select preset material types in the **Material type** setting in the **Advanced** panel on the **Inspect** > **Scan** page. The **Diffuse** material option is suitable for most materials.

When **Material type** is set to **Custom**, you can set camera gain and modify spot detection. Some sensor models support both analog and digital gain, whereas others only support digital gain. For more information, see *Spot Detection* below and *Camera Gain* on page 227.

Spot Detection

When adjusting the spot detection parameters, it's best to observe how the changes you make to these parameters affect the detected spots in Image mode, and also how that affects the resulting profile in Profile mode.

You can only adjust spot detection parameters when **Material type** is set to Custom. Set this to custom if the default values do not provide adequate spot detection.

Various settings can affect how the **Material** settings behave. For more information, *Spots and Dropouts* on page 97.

When **Materials** is set to **Custom**, you can configure the following settings. In order to properly configure the spot-related settings, you should use Image mode (for more information, see *Image Mode* on page 92) to observe the laser line and spots in the data viewer. For information on spots and displaying them in the data viewer, see *Spots and Dropouts* on page 97.

> **Edge window**, **Width threshold**, and **Max spots** are advanced parameters. You should not change them.

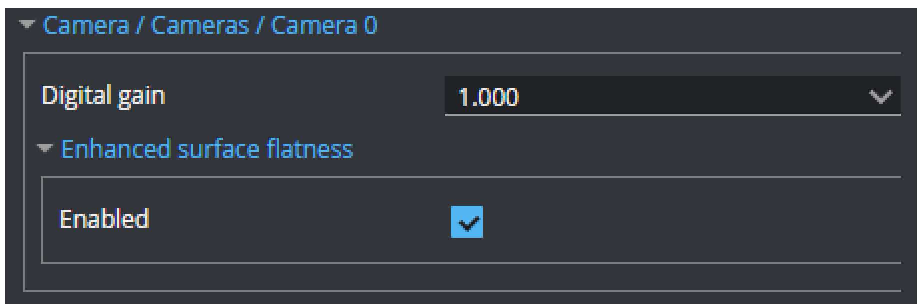
Setting	Description
Threshold	The minimum increase in intensity level between neighboring pixels for a pixel to be considered the start of a potential spot. This setting is useful for filtering false spots generated by sunlight reflection.
Spot Selection	Determines the spot selection method. For details, see *Spot Selection Methods* below.
Min width Max width	The minimum and maximum number of pixels a spot is allowed to span along Y in the data viewer (in Image mode). This setting can be used to filter out data caused by background light if the unwanted light is wider than the laser and does not merge into the laser itself. A lower **Max width** setting reduces the chance of false detection, but limits the ability to detect features/surfaces that elongate the spot. **Min width** can help reduce small speckle noise.
Min sum	Minimum total of pixel intensities over a spot. If the total of pixel values is below this value, the spot will be dropped. This can be used to exclude weak spots, but this is typically undesirable, so it is often set to 1.

Spot Selection Methods

Setting	Description
Best	The **Best** spot selection method selects the strongest or peak spot in a given column on the imager.
Top or Bottom	**Top** selects the topmost spot or the one farthest to the left on the imager, and **Bottom** selects the bottommost spot or the one farthest to the right on the imager. These options can be useful in applications where there are reflections, flying sparks, or smoke that are always on one side of the laser.
None	The **None** selection mode performs no spot filtering. If multiple spots are detected in an imager column, they are left as is. This option is only available if **Uniform Spacing** is disabled in the **Scan Mode** panel on the **Scan** page; for more information on uniform spacing, see *Uniform Data and Point Cloud Data* on page 70. Note that when **Uniform Spacing** is disabled and **Spot Selection** is set to None, both Profile Dimension and Profile Position are unavailable; for more information on enabling and disabling uniform spacing, see *Scan Modes and Intensity* on page 195.
Continuity	The **Continuity** selection mode considers adjacent horizontal data points on the imager to place spots on pixels, giving preference to more complete profile segments. The setting can improve scans in the presence of reflections and noise. When you select the continuity type, you must set the **Min segment size**, **Search window length**, and **Search window width** parameters, which define the minimum length of the segment, as well as the length and width of the segment search window.

Camera Gain

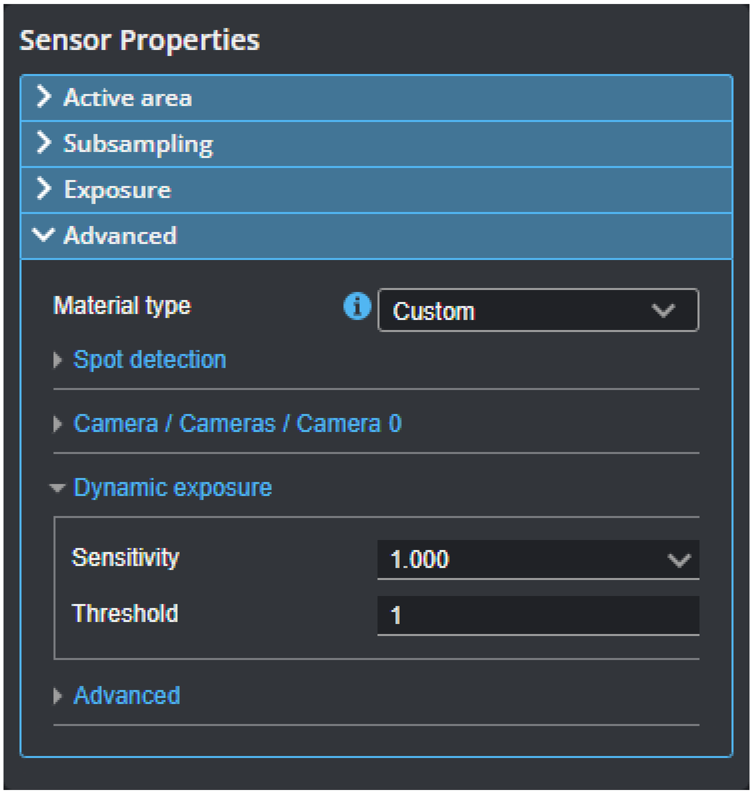
Setting	Description
Camera Gain	
	Analog camera gain can be used when the application is severely exposure limited, yet dynamic range is not a critical factor. Boosts the signal before conversion to digital, so it can potentially allow differentiating dark details that would otherwise be mapped to the same digital value. Not available on all sensor models. **Digital** camera gain can be used when the application is severely exposure limited, yet dynamic range is not a critical factor.
Enhanced Surface Flatness	This setting enables a pre-processing step that reduces spatial noise in video data. This can help the sensor find and measure defects and features more accurately on flat surfaces in scan data. This setting is only available on Gocator 2500 and 2600 series sensors. It is enabled by default on Gocator 2600 sensors.



Material Settings and Dynamic Exposure

You can set dynamic exposure to improve data acquisition.

The settings are in the **Advanced** panel on the **Inspect** > **Scan** page.

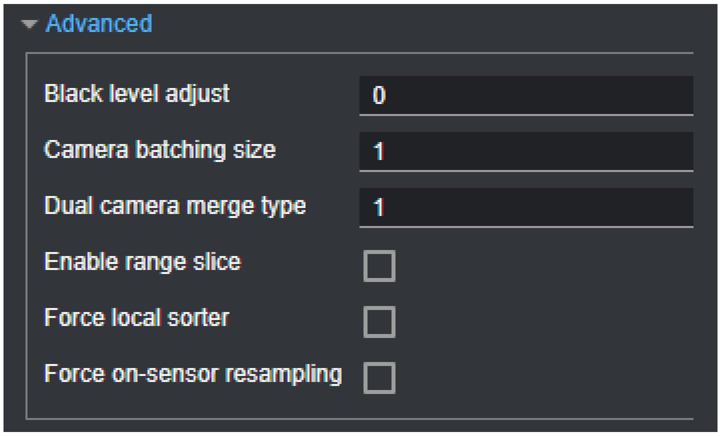


These settings let you set tune how dynamic exposure settles on an exposure for a scan. For more information on Dynamic Exposure, see *Dynamic Exposure* on page 223.

Setting	Description
Dynamic Exposure	
	Sensitivity controls the exposure that dynamic exposure converges to. The lower the value, the lower the exposure the sensor will settle on. The trade-off is between the number of underexposed spots and the possibility of over-exposing. **Threshold** is the minimum number of spots for dynamic exposure to consider the profile point that make up the spot valid. If the number of spots is below this threshold, the algorithm will walk over the allowed exposure range slowly to find the correct exposure. Because this is slow, the parameter's value typically should be kept as low as possible, so this slow search is not used.

Other Advanced Settings

You should leave the settings in the **Advanced** expanding section unchanged unless you are told to do so by LMI support.

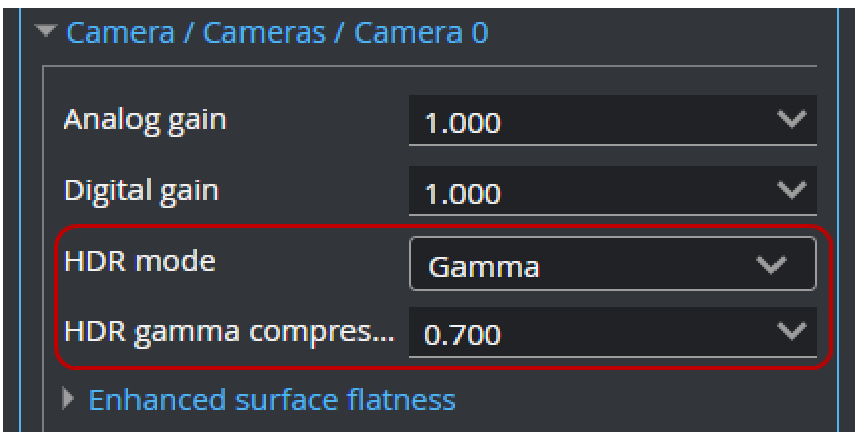


HDR Mode

	This setting is only available on Gocator 2600 series sensors.

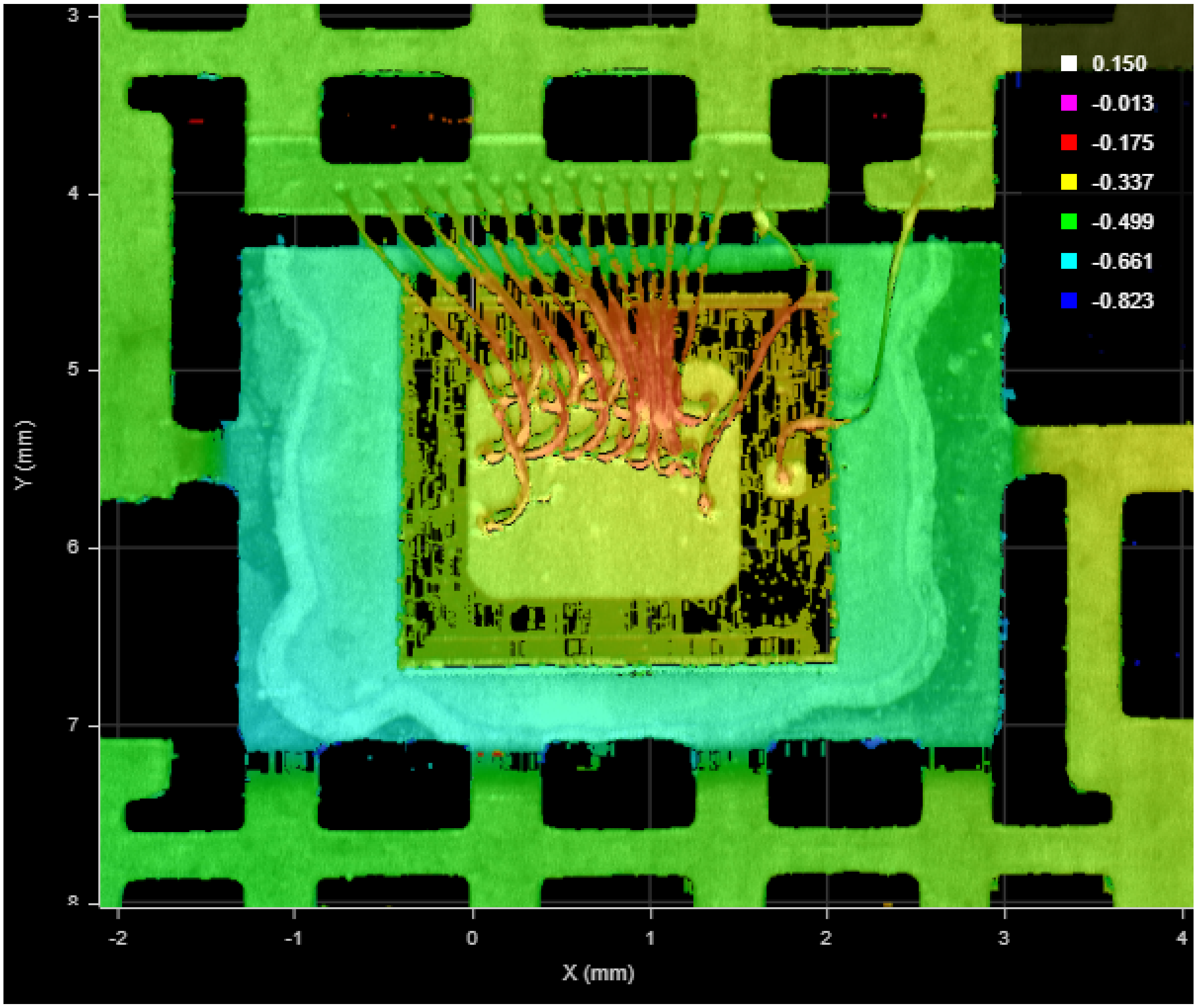
HDR mode increases the dynamic range of supported sensors, improving data quality when scanning challenging targets, such as shiny metal surfaces or surfaces with a combination of materials.

HDR mode is on the **Scan** page, in the **Advanced** section in the **Sensor Properties** panel, under **Camera**.

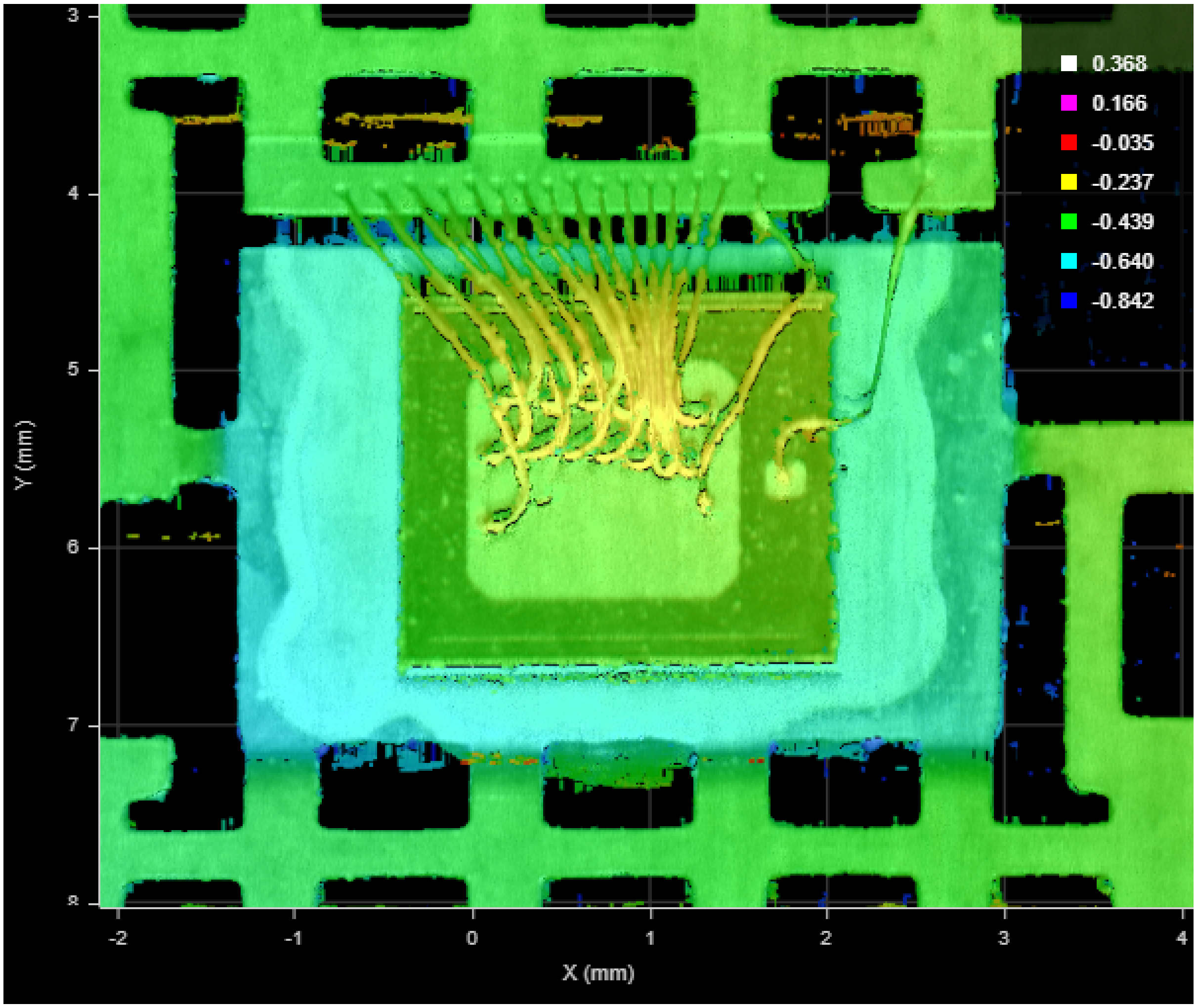


When scanning targets that typically require multiple exposures to capture both dark and light features on the surface, with HDR, you may be able to use a single exposure and improve cycle times. The result is better overall scan quality in shorter time.

In the following, HDR mode is disabled. There is a considerable loss of scan data.



In the following, HDR mode (gamma) is enabled, and its value is set to 0.3. The scan data is greatly improved.



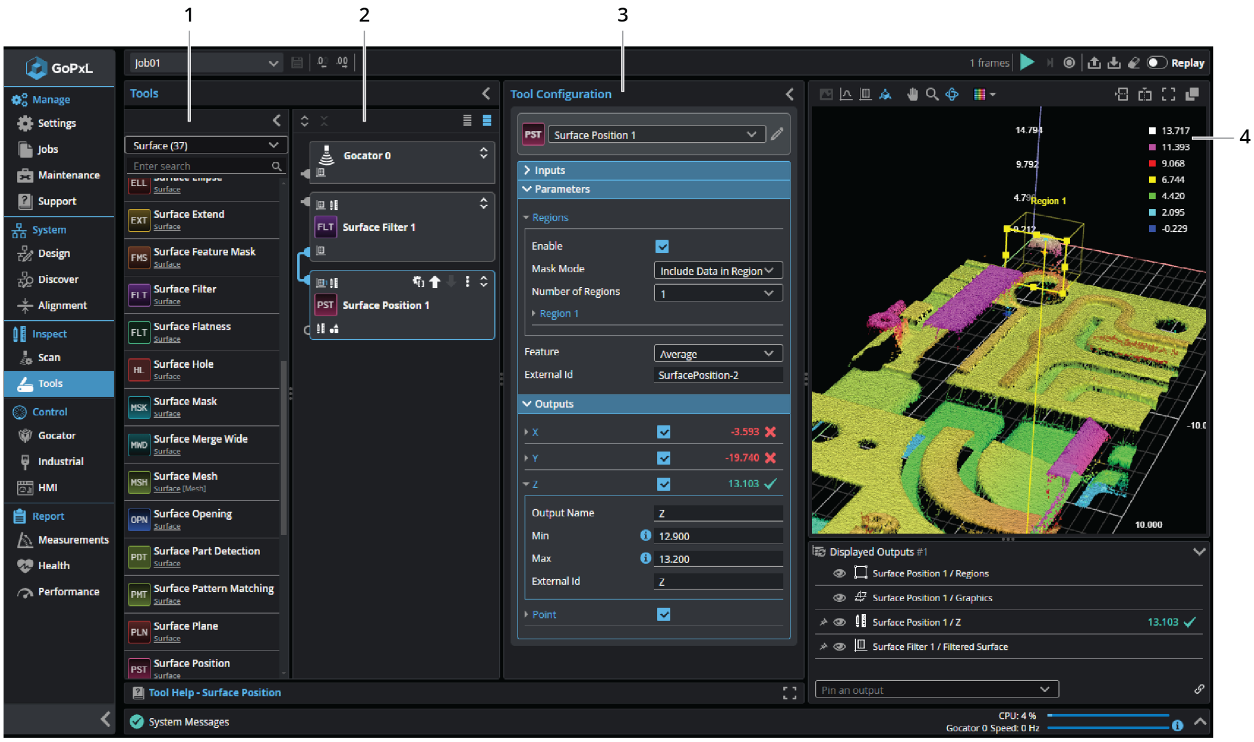
When you enable HDR mode, the sensor applies a gamma curve internally to the video image. High intensity values are compressed to provide better intensity resolution for lower intensity values, which in turn can provide better quality Profile or Surface scan data.

Use the following guidelines to set the value of **HDR Parameters**, which represents the gamma value the sensor uses.

- 1.0: Equivalent to HDR being disabled.
- 0.7 (default): The lowest intensity values are somewhat brighter.
- 0.5: The lowest intensity values are much brighter.

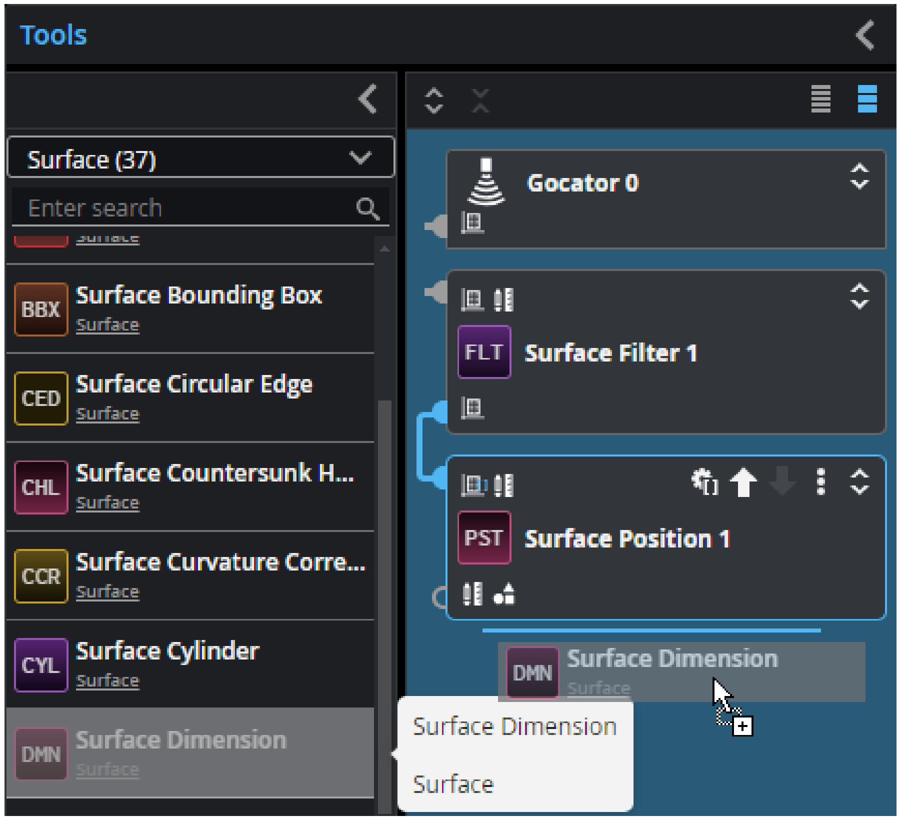
Tools - Measurement and Processing

You add and configure measurement and processing tools on the **Tools** page, in the **Inspect** category.

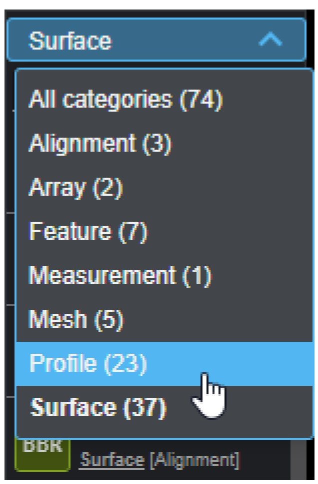


	Element	Description
1	Tool list	Lists all of the available tools. To add a tool, double-click it or drag and drop it in the Tools Diagram panel to the right of the list.

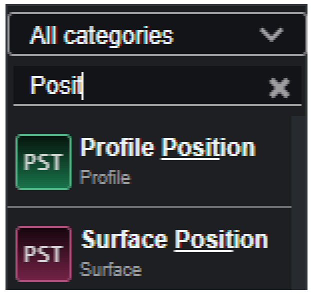
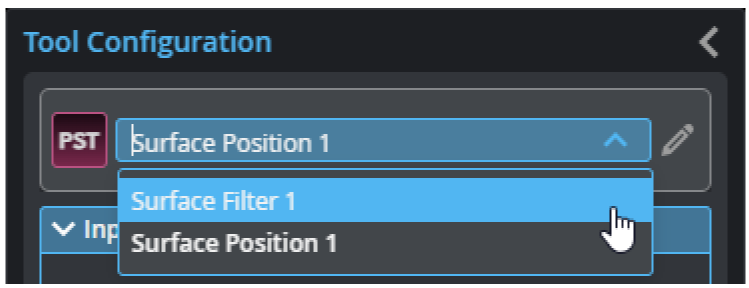
Element	Description



You can filter tools using the filter at the top of the Tool list.



You can also search for tools by typing in the search field. Tools are listed as you type.

	Element	Description
		
2	Tools Diagram	Lists the tools that you have added. The Tools Diagram lets you better visualize the workflow in complex tools chains. For more information, see *Working with Tool Chains* on page 270.
3	Tool configuration	You configure a tool using the **Inputs**, **Parameters**, and **Outputs** expandable sections in the **Tool Configuration** panel. Typically, you configure tools from top to bottom, that is, by setting and configuring the inputs first, then the parameters, and finally the outputs, all in the order they are presented in these sections. You can select which tool you want to configure from the tool selector drop-down, or by selecting the tool in the Tools Diagram to the left.  For information on parameters common to most tools, see *Tool Configuration* on page 240. For information on the different types of tools, see their topics in the following sections. Profile tools: See *Profile Measurement* on page 284. Surface tools: See *Surface Measurement* on page 444. Feature tools: See *Feature Measurement* on page 692. Mesh tools: See *Mesh Measurement* on page 747. Array tools: See *Array Tools* on page 738. Measurement Formula tool: See *Measurement Formula Tool* on page 767. Script tool: See *Script* on page 770. Data Export tool: *Data Export* on page 734.
4	Data viewer	When configuring tools, you can graphically configure regions of interest. The data viewer's toolbar lets you change the display mode (for more

Element	Description
	information, see *Data Viewer* on page 88) and split a data viewer or pop out a data viewer in a new browser window.

Adding a Tool

You can add a tool from the tool list in two ways: by dragging and dropping the tool onto the Tools Diagram or by double-clicking the tool.

GoPxL automatically sets the added tool's input to an appropriate output when possible. The output GoPxL selects is either directly from a sensor or sensor group, or from the output of another tool, depending on various factors (see *Tool Connection Logic* below).

When dragging and dropping a tool, a line in the Tools Diagram shows where the tool will be added when you drop it.

When double-clicking a tool in the tool list, the new tool is added after the currently selected tool in the Tools Diagram.

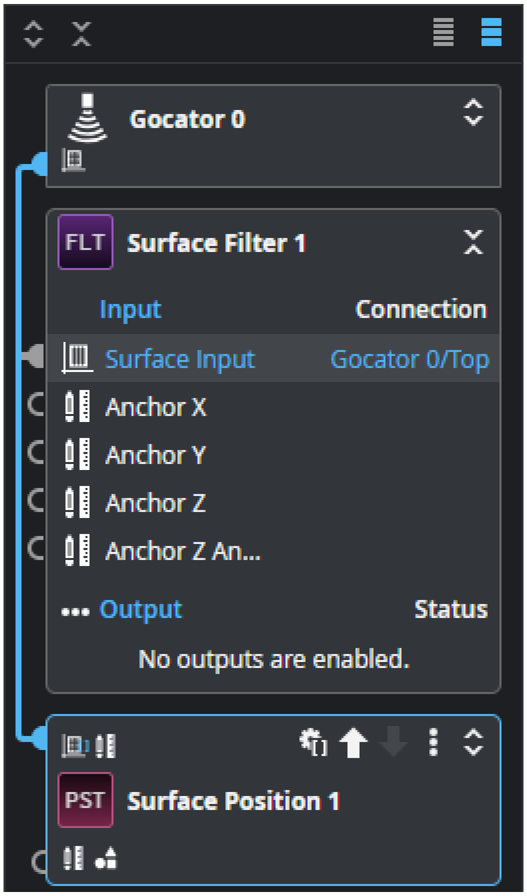
Tool Connection Logic

When you add a tool, GoPxL looks for compatible outputs among the other tools to set the added tool's inputs, starting with the tool or sensor directly "above" the added tool in the list of added tools (in the Tools Diagram panel). By "compatible," we mean, for example, data type (Profile vs. Surface, or uniform vs. non-uniform or point cloud data). GoPxL ignores tools "below" the added tool for this initial search. GoPxL checks the compatibility of all the outputs in a tool before moving up to the next tool.

In the following, a Surface Position tool has been added before the third Surface Filter tool. GoPxL will start searching for compatible outputs in Surface Filter 2 and then move to Surface Filter 1. If no compatible outputs are found in the preceding tools, GoPxL will search for compatible outputs in Gocator 0. Surface Filter 3 is ignored for auto-connection.

When a preceding tool has multiple enabled outputs, GoPxL goes through them in the order they are defined in *the tool* (that is, from the top down).

If no outputs from the preceding tools are compatible with the tool's inputs, GoPxL searches through the sensor or sensor group outputs. In the following, the added Surface Position tool "skips" the Surface Filter tool, whose output has been disabled.

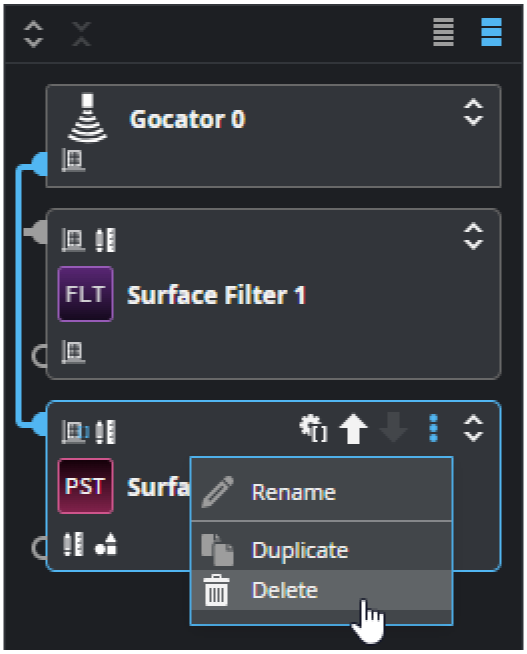


For information on other operations you can perform in the Tools Diagram, see *Working with Tool Chains* on page 270.

Removing a Tool

If you no longer need a tool, you can remove it in the Tools Diagram. Removing an unused tool can increase performance, as it no longer takes up processing time.

To remove a tool, select it in the Tools Diagram and press Delete on your keyboard. You can also open the tool's Action menu ( ) and choose **Delete**.

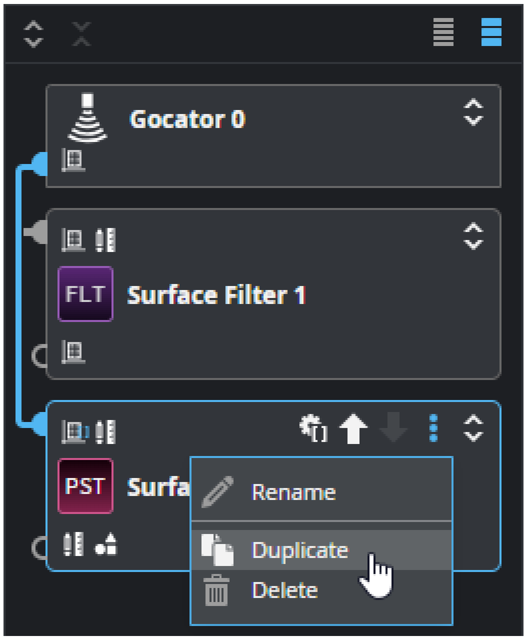


For information on other operations you can perform in the Tools Diagram, see *Working with Tool Chains* on page 270.

Duplicating a Tool

You can quickly create a copy of a tool you have previously added in GoPxL in the Tools Diagram panel. The tool inputs, parameter settings, and enabled outputs of the original are copied. This is useful, for example, when you need almost identical tools with only minor variations, such as different maximum and minimum decision values, or regions of interest shifted from one component to another.

In the Tools Diagram panel, you duplicate a tool by opening the tool's Action menu (⋮) and choosing **Duplicate**.



For information on other operations you can perform in the Tools Diagram, see *Working with Tool Chains* on page 270.

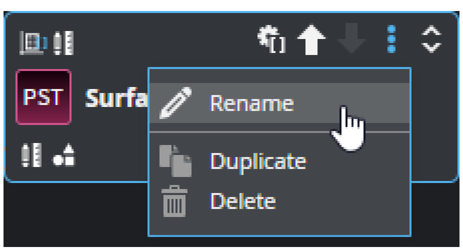
Renaming a Tool

In the Tools Diagram panel, you rename a tool using the action menu of an individual tool.

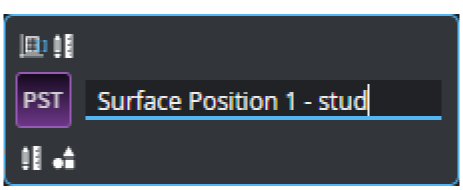
To rename a tool in the Tools Diagram panel

1. Click the action menu icon ( ).

   A context menu containing various actions appears.



2. In the context menu, choose **Rename**.
3. In the tool name field, rename to the tool.

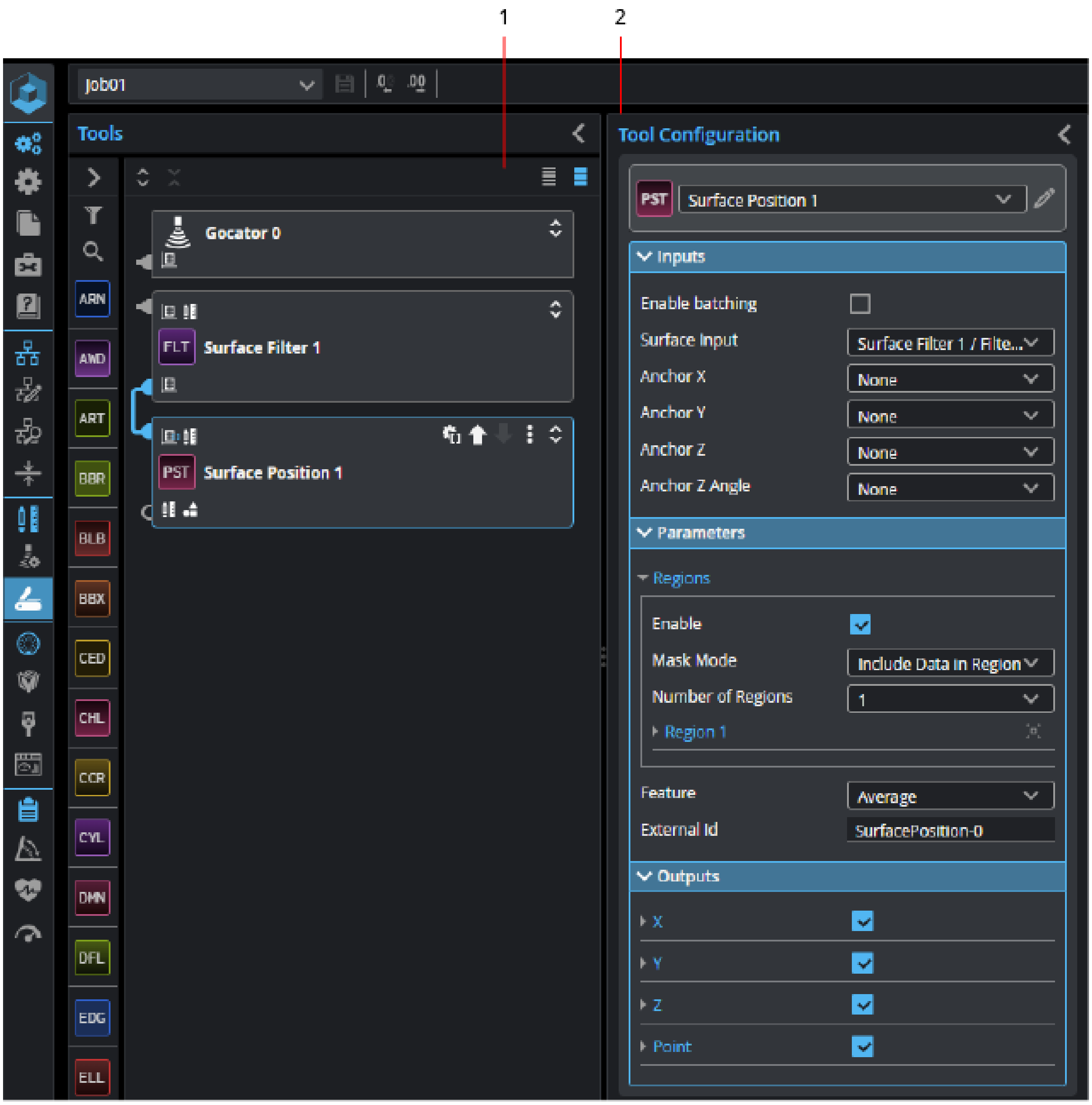


4. Press Enter on the keyboard or click outside the text field.

For information on other operations you can perform in the Tools Diagram, see *Working with Tool Chains* on page 270.

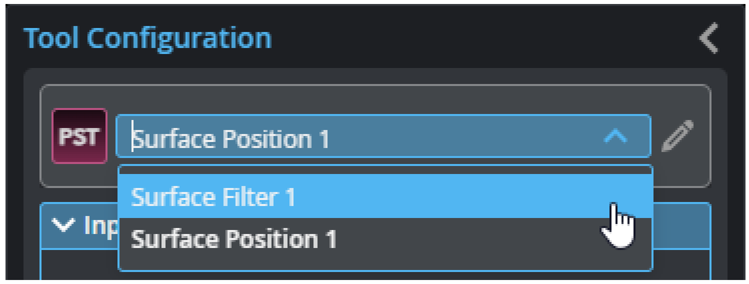
Tool Configuration

You configure a tool's inputs, parameters, and outputs in the tool configuration area to the right of the Tools Diagram.



	Element	Description
1	Tools Diagram	Lists the tools that you have added. The Tools Diagram lets you better visualize the workflow in complex tools chains. For more information, see *Working with Tool Chains* on page 270.
2	Tool configuration	Lets you configure the selected tool using the **Inputs**, **Parameters**, and

Element	Description
	Outputs sections in this area. You can select which tool you want to configure from the tool selector drop-down, in addition to selecting the tool in the Tools Diagram.



The following sections describe the parameters common to most tools.

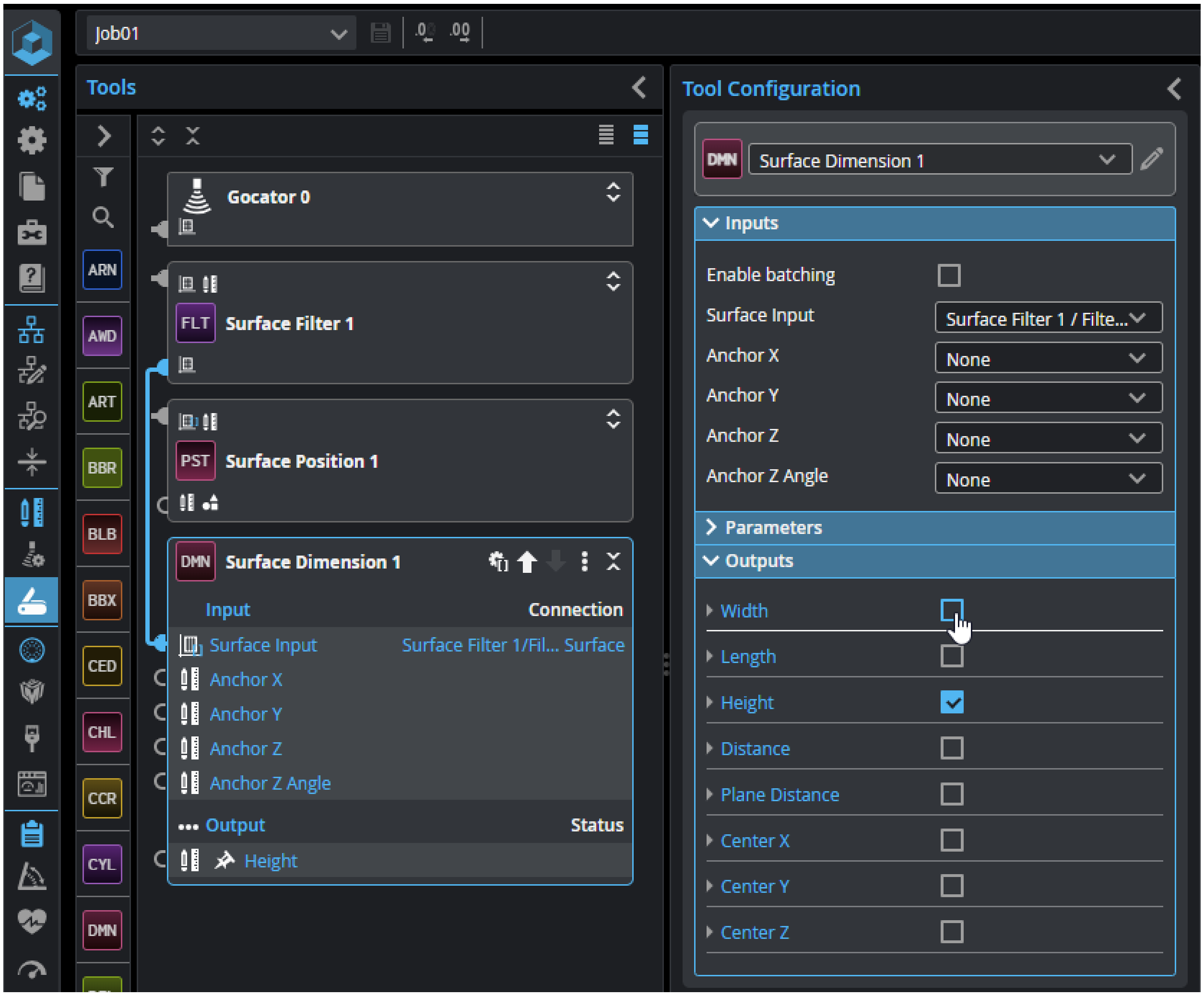
Inputs and Outputs

All tools take one or more inputs, either directly from a sensor or sensor group, or from another tool. All tools produce output, such as measurements or modified scan data.

The simplest case is a tool taking a single input, such as a profile (line profilers only) or surface data. In these cases, the tool performs measurements on the data it receives, or processes it, and produces outputs.

Most tools can also take "arrays" as inputs. For more information on arrays and how tools work with them, see *Arrays, Batching, and Aggregation* below.

All of the outputs (measurements, features, and scan data such as Surface or Profile data) available in a tool are listed in the **Outputs** section of the tool. To enable an output, check its checkbox in the tool's **Outputs** panel.



Enabling the Width measurement. Only the Height measurement is currently enabled, which is displayed in the tool's output section in the Tools Diagram.

Arrays, Batching, and Aggregation

GoPxL supports data structures called arrays to help simplify otherwise complicated applications.

An array is a group of data items bundled into a single structure, such as multiple profiles or surfaces, multiple measurements, or multiple geometric features.

The items, or elements, in the array can be processed by other tools in one of two ways:

- Batching: Tools in batching mode process each array element *separately*. In other words, the tools "loop" through the elements of the array. Typically, tools in batching mode will output an array of outputs, where each element corresponds to the elements in the input. You enable batching by checking the **Enable batching** parameter in a tool's **Inputs** section. For more information, see *Batching* on page 247.
- Aggregating: Aggregating tools combine each element in the array and process it as a *single* piece of data. For more information, see *Aggregating* on page 249.

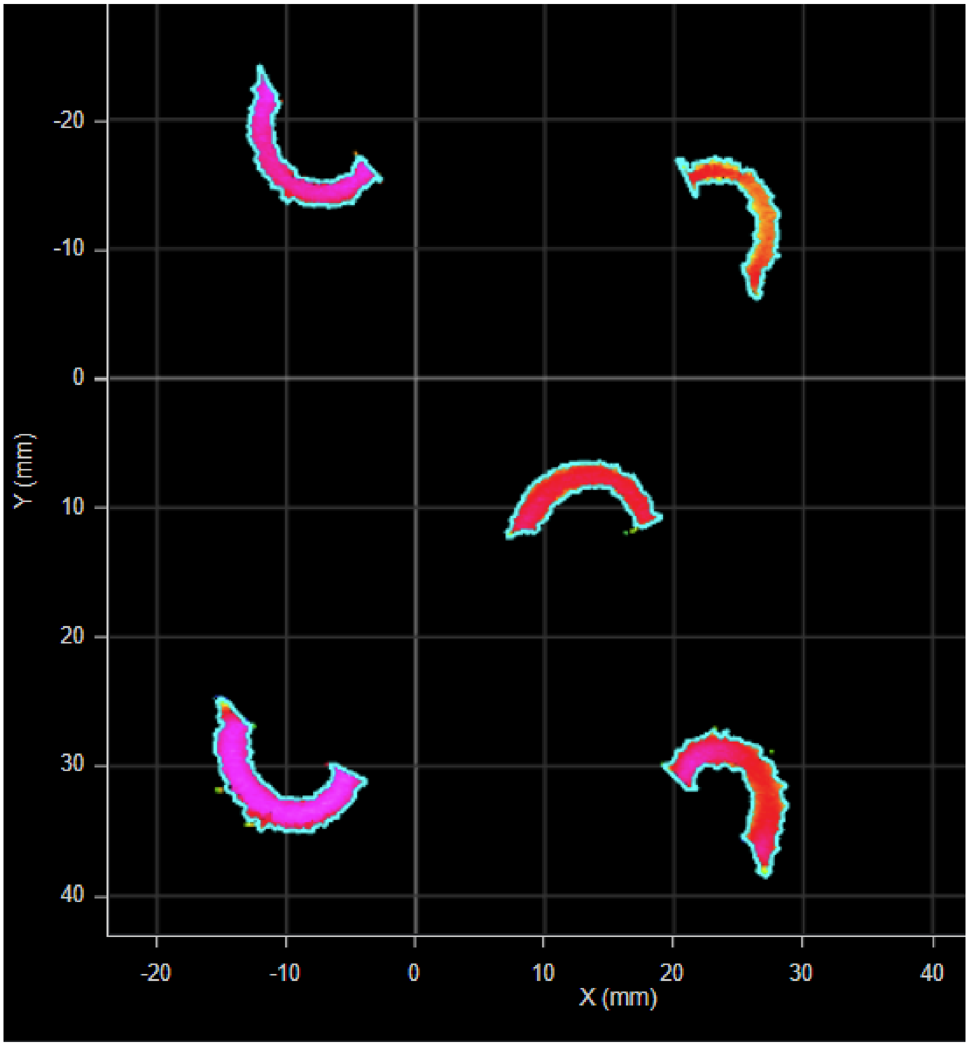
Most tools support the batching of array elements. Some tools support both batching and aggregating.

Array data can come from various sources:

- Outputs from tools like Surface Blob, Surface Segmentation, and Surface Pattern Matching. For more information, see *Arrays from Multi-instance Tools* below.
- G4 and G5 (single- and multi-sensor) and multi-layer output. For more information, see *Arrays from Multi-layer Output* on page 245.
- G2 multi-sensor output. For more information, see *G2 Multi-sensor Output* on page 246.
- The Script tool. For more information, see *Script* on page 770.
- The Array Create tool. For more information, see *Array Create* on page 739.

Arrays from Multi-instance Tools

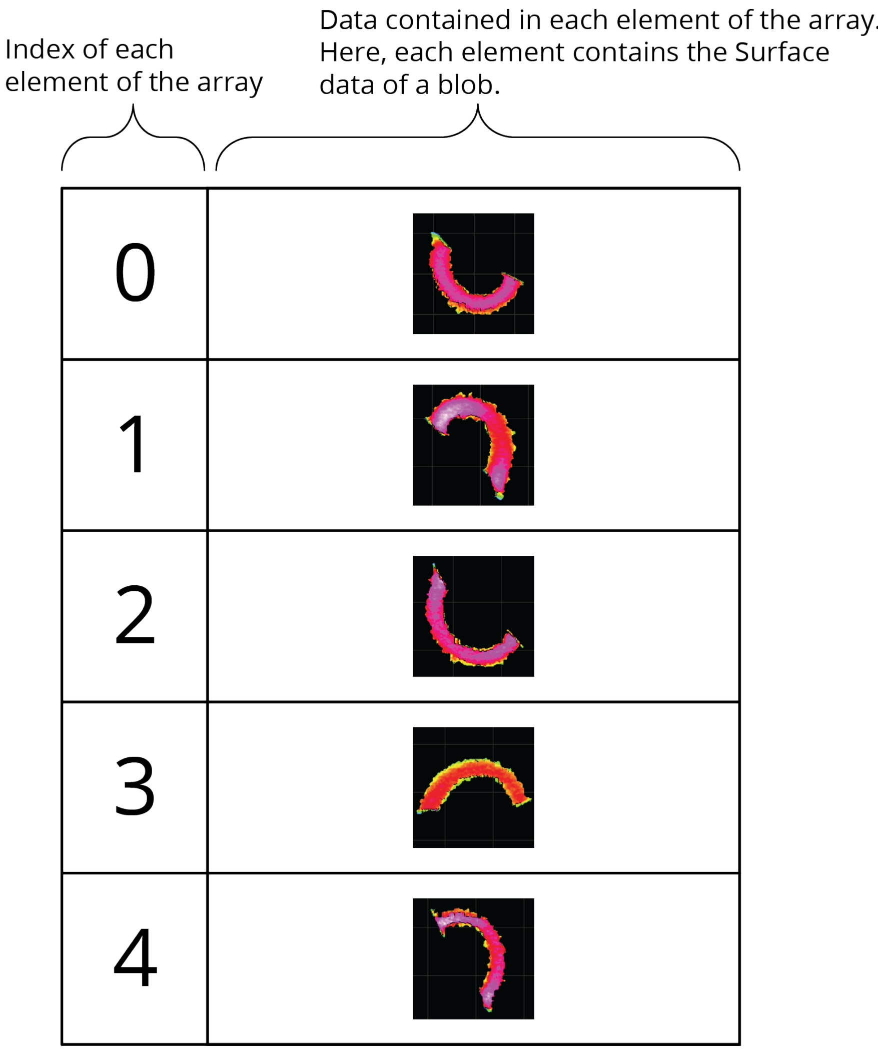
The Surface Blob, Surface Segmentation, and Surface Pattern Matching tools can identify multiple occurrences of features or parts in scan data. When you check **Enable Array Output** in the tool's **Parameters** section, the tool outputs an array containing the occurrences. Otherwise, the tools produce a separate output for each identified feature or part. In the following, a Surface Blob tool has identified five blobs in scan data.



Surface Blob tool identifying blobs in scan data.

When **Enable Array Output** is checked, the Surface Blob tool outputs an array containing all of the blobs it identifies.

An array containing objects identified in some scan data could be represented in the following way. (These are "blobs" from the Surface Blob tool.)



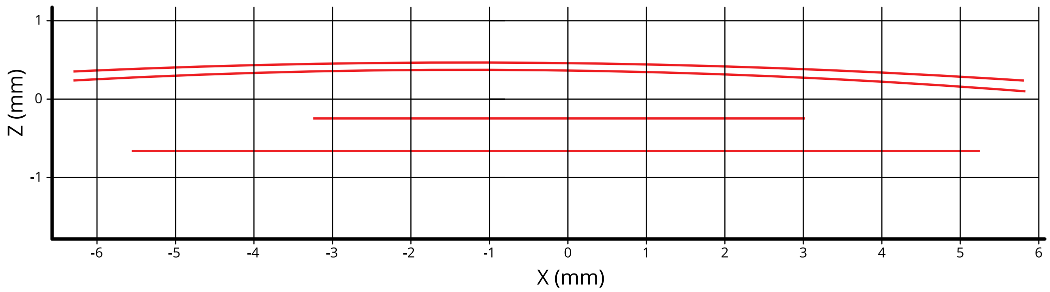
Note that array elements start at 0; that is, arrays are 0-based.

Other tools can then take this array as input. For more information, see *Batching* on page 247 or *Aggregating* on page 249.

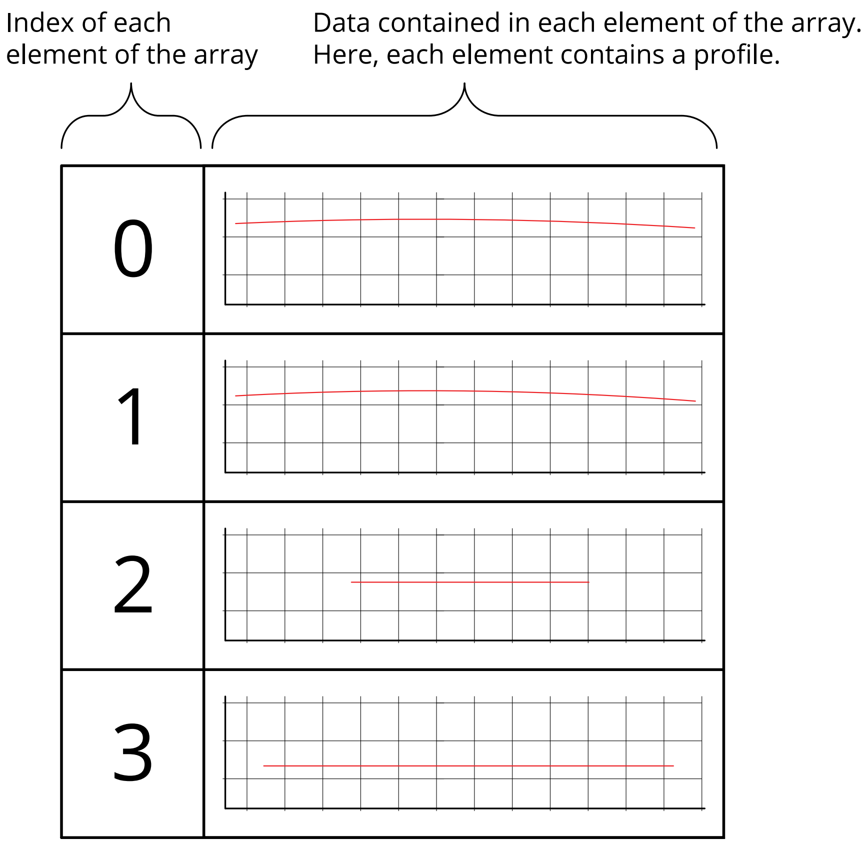
Arrays from Multi-layer Output

When configured for multi-layer scanning, confocal sensors can scan translucent or transparent targets and return a profile or a surface representing each layer. By default, these profiles are output as an array. (Note that you can optionally have GoPxL output the profiles individually by *enabling* **Separate layer outputs** on the **Scan** page; for more information, see *Scan Modes and Intensity* on page 195.)

The following illustration is a representation of multilayer Profile scan data from a confocal sensor, showing four profiles.



The structure of an array containing these profiles could be represented as follows:



Other tools can then take this array as input. For more information, see *Batching* on the next page or *Aggregating* on page 249.

G2 Multi-sensor Output

In G2 multi-sensor systems where sensors are in a top-bottom layout, Profile and Surface data can optionally be output as an array, where top data (from one or more sensors) is one array element, and bottom data (from one or more sensors) is another array element.

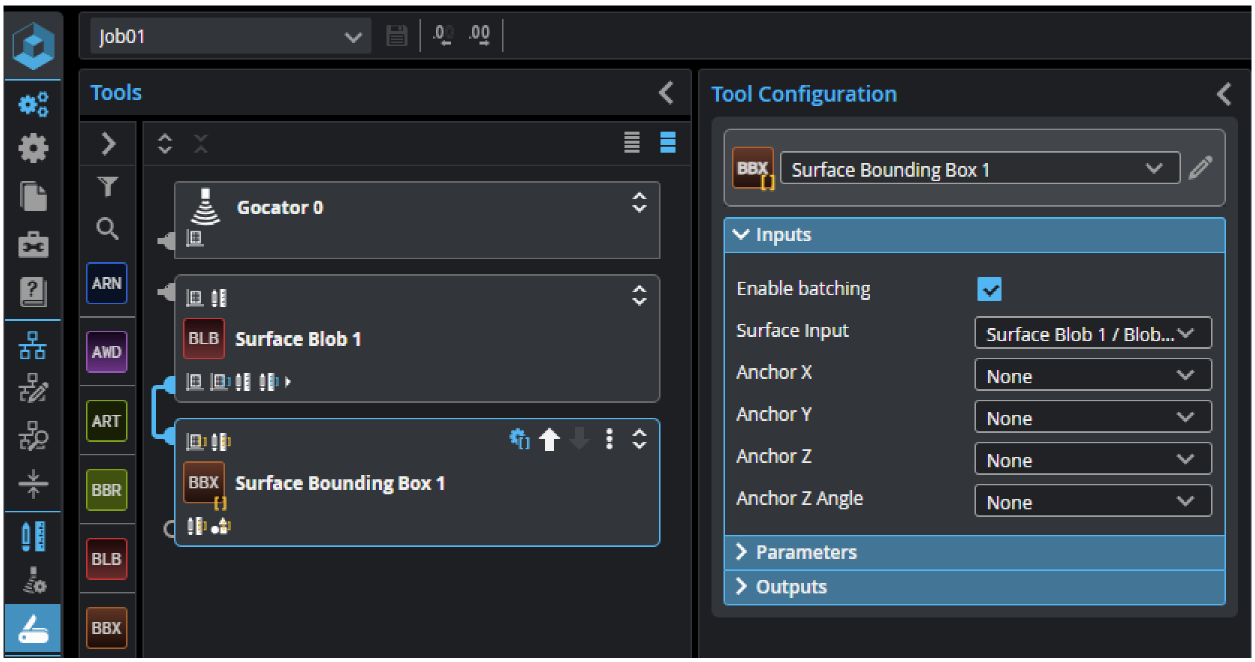
Other tools can then take this array as input. For more information, see *Batching* on the next page or *Aggregating* on page 249.

Batching

When a tool is in *batching* mode, it takes an array as input and processes each item in the input array *individually*. A tool in batching mode also outputs one or more arrays (such as measurements, Profile or Surface data, geometric features, and so on), where each element of the output arrays corresponds to an element in the input array.

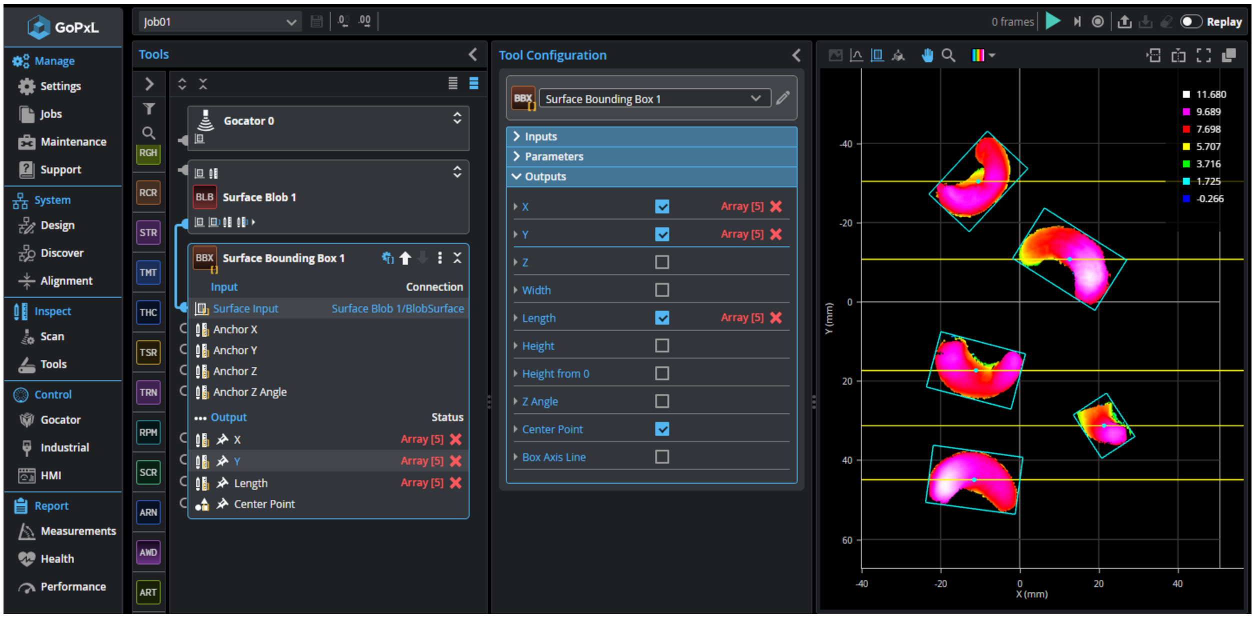
When a tool is in batching mode, there is no limit to the number of elements in the input array (other than any CPU limitations).

Batching is enabled in tools by checking **Enable batching** in the tool's **Inputs** section, or by clicking the Batching icon ( ) in a tool's block in the Tools Diagram panel (see below).

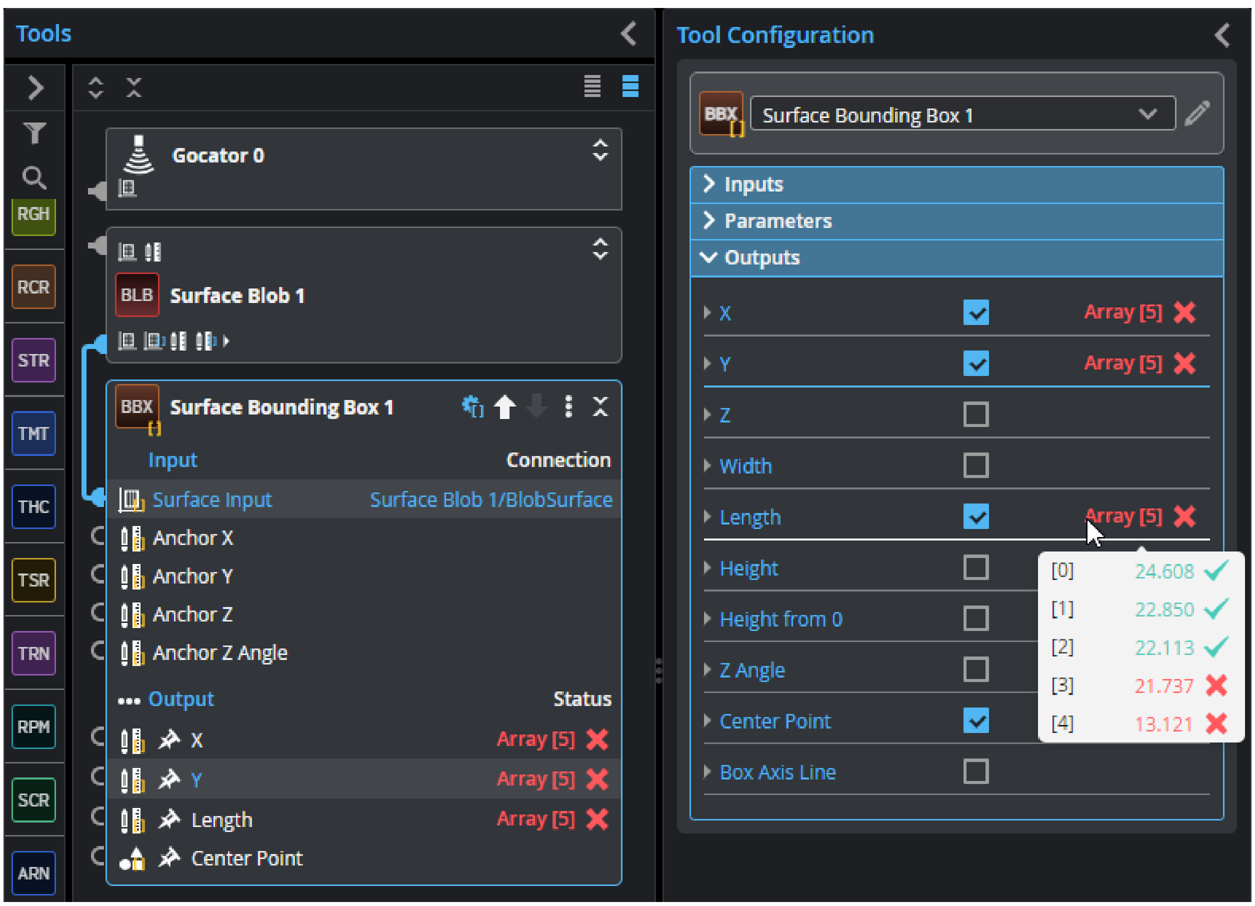


When batching mode is enabled, it's indicated by square bracket ("[ ]") overlays in the Tools Diagram and in the **Tool Configuration** panel. The Batching icon also turns blue.

The following shows the Surface Bounding Box tool in batching mode (taking the Surface Blob tool's array of outputs as intput). The tool calculates the bounding box for each element in the input array, and returns the enabled measurements as arrays. That is, it returns an X, Y, and Z measurement for each object, in the X, Y, and Z arrays.



Hovering over batched outputs in the Tools Diagram or in the **Tool Configuration** panel displays a list of the elements in the array.



When at least one result in the array is a fail, the output is considered a fail and the "Array[n]" label is red. Otherwise, it's green. Outputs within the pop-up list are displayed in green or red, depending on whether they represent pass or fail.

Aggregating

Some tools can take arrays as an input and combine (or "aggregate") the data in the array elements and treat it as a single input (see below for an example). These tools are called "aggregating tools"; see *Aggregating Tools* below for a list of the aggregating tools. In order for a tool to aggregate an array input, its **Enable Batching** parameter must be disabled. Otherwise, it will process the array elements individually. Note that aggregating tools also take non-array input (such as a single scan of Profile or Surface data), and simply process the data as usual.

Some aggregating tools are limited to a maximum of a 2-element array in their input. Other tools can accept an arbitrary number. See *Aggregating Tools* below to determine the limits of a tool. Note that when a tool is in *batching* mode, there is no limit to the number of elements in the input array (other than any CPU limitations), regardless of the array size limitations when the tools is aggregating data.

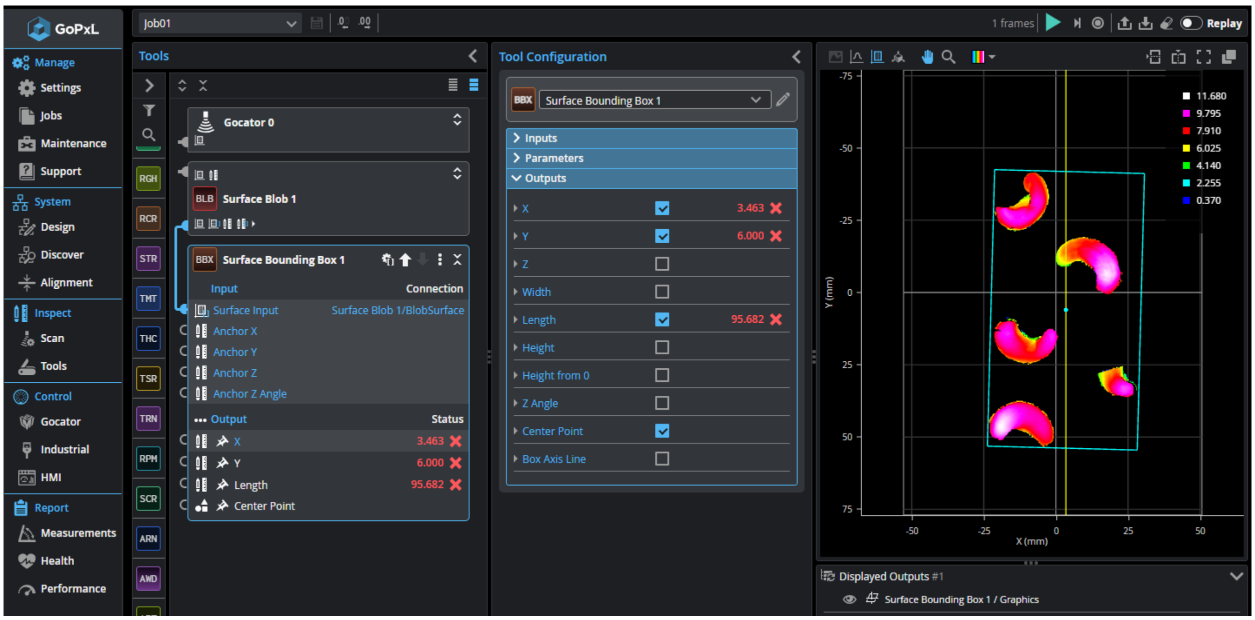
> Remember to uncheck **Enable Batching** if you want a tool to aggregate array inputs.

Aggregating Tools

	Number of array elements supported for aggregation	
Tool	**2 elements maximum**	**arbitrary (2 or more elements)**
Profile Circle Radii		✓
Profile Closed Area		✓
Profile Bounding Box		✓
Profile Bridge Value		✓
Profile Circle		✓
Profile Dimension	✓	
Profile Ellipse		✓
Profile Groove	✓	
Profile Intersect	✓	
Profile Line		✓
Profile Line Intersect		✓
Profile Panel	✓	
Profile Position	✓	
Profile Round Corner	✓	
Profile Strip		✓
Surface Bounding Box		✓
Surface Dimension		✓
Surface Ellipse		✓

	Number of array elements supported for aggregation
Surface Flatness	✓
Surface Plane	✓
Surface Position	✓
Surface Section	✓
Surface Sphere	✓
Surface Stick	✓

Taking the example from above, if batching in the Surface Bounding Box tool is disabled, it will aggregate the scan data in the input array, and fit the bounding around the scan data of all the elements together.

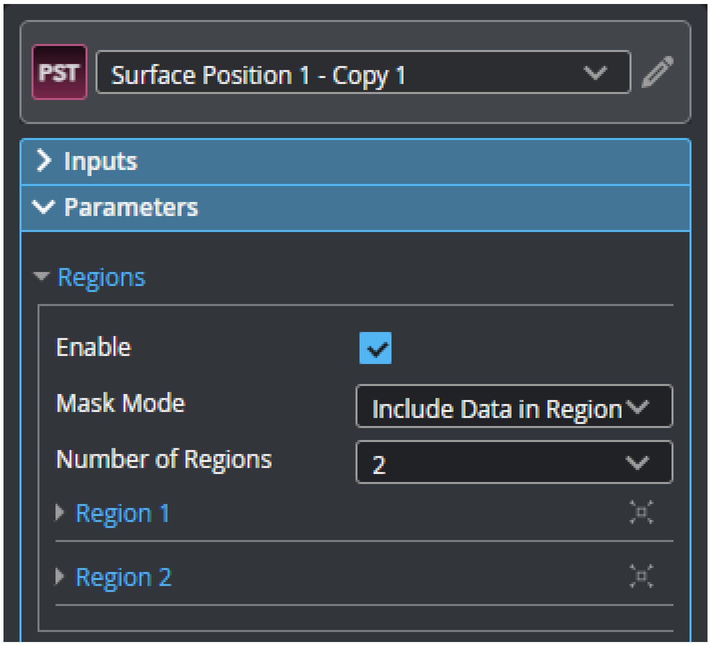


Regions

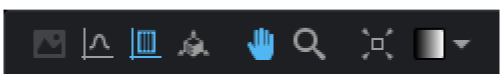
Most measurement tools use user-defined regions, or regions of interest, to limit the area in which measurements occur. Unlike reducing the active area, reducing the region of interest does not increase the maximum frame rate of the sensor.

You can use Ctrl-Z and Ctrl-Shift-Z to undo and redo changes to the last changed region's size and position.

You can also quickly center a region in the data viewer and reset it to its default size; this is useful if you have zoomed in or out a lot. You can center a specific region by clicking its "Center tool region" button ( ) in the tool's **Parameters** section, next to the region expander.



You can center and reset all regions in a tool by clicking the "Center tool region" button ( ) above the data viewer. When a tool has multiple regions, the regions are spaced out in the center of the data viewer.

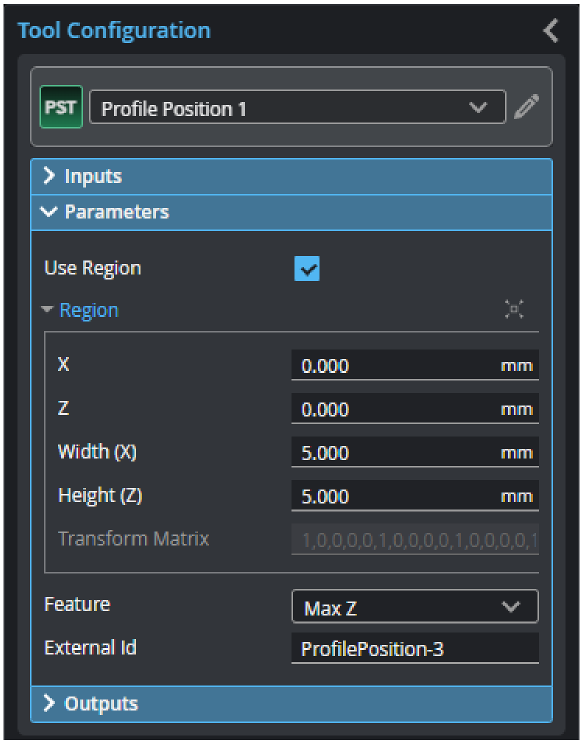


Some tools provide "flexible" regions, which in addition to rectangular regions let you create circular and elliptical regions (which can optionally be annular). Some tools also let you use Surface and Surface Intensity data as masks. For information on setting flexible regions, see *Flexible Regions* on page 253.

Other tools are currently limited to standard, rectangular regions.

Standard Regions

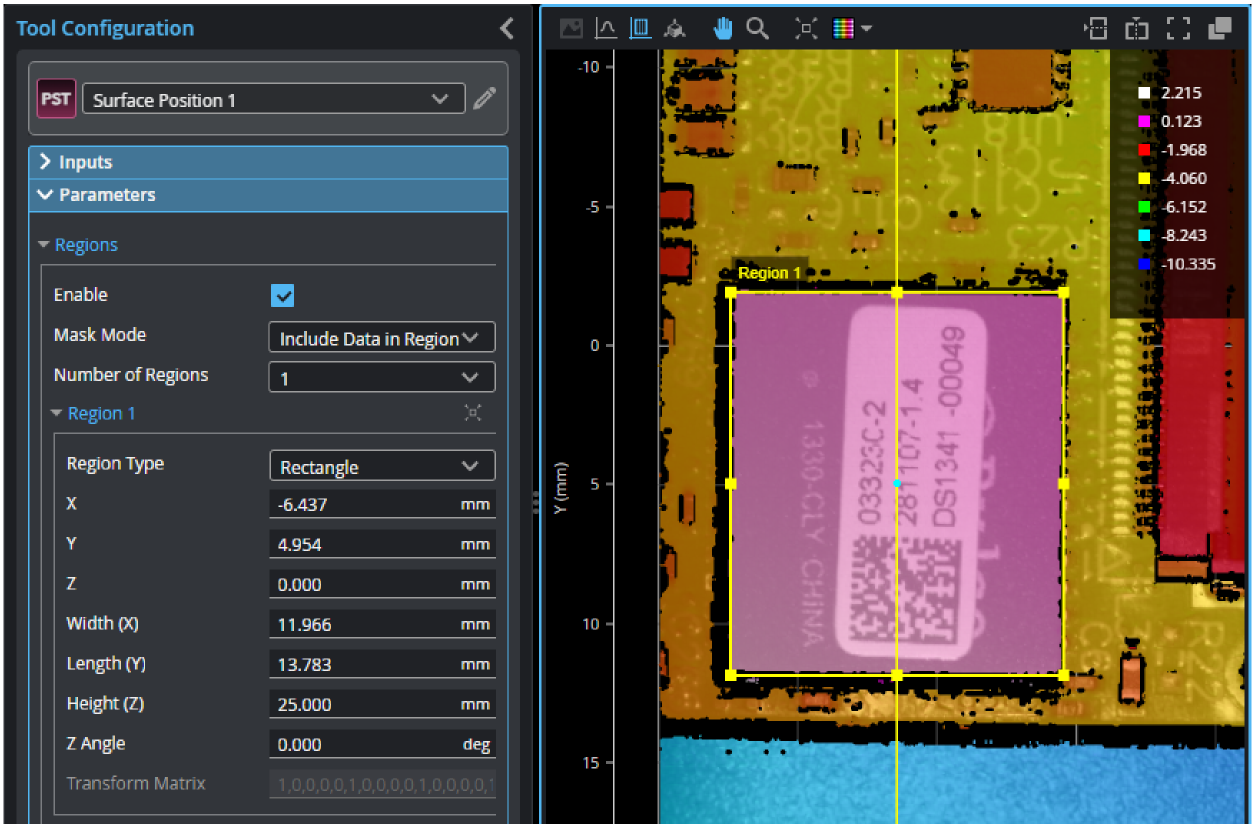
Some tools provide regions limited to boxes/rectangles.

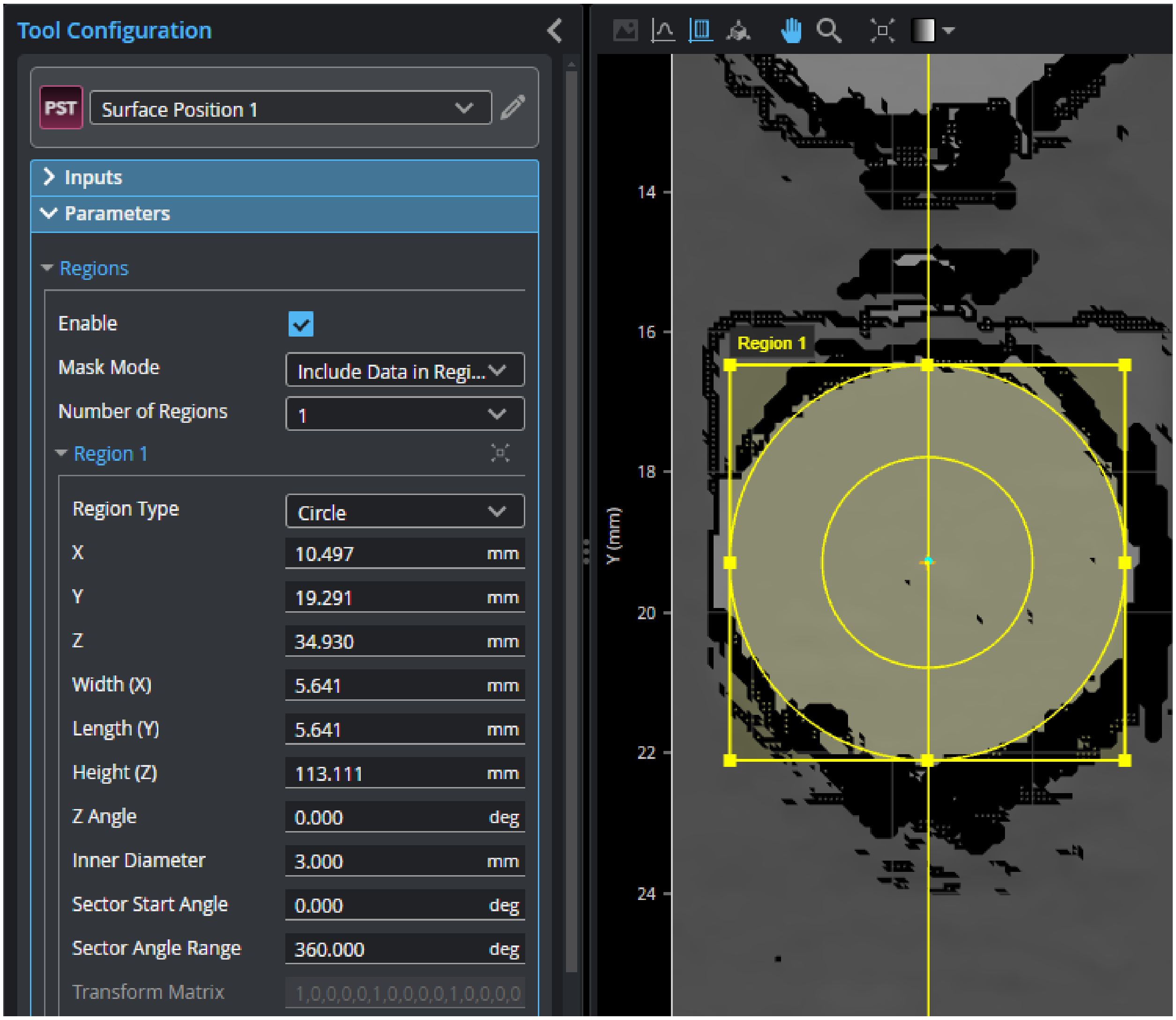


The **Transform Matrix** setting is not currently used.

Flexible Regions

Many tools provide "flexible" regions, which let you choose region types such as circle and ellipse, in addition to rectangle regions. For a complete list and information on the related settings, see *Flexible Region Parameters* on the next page.

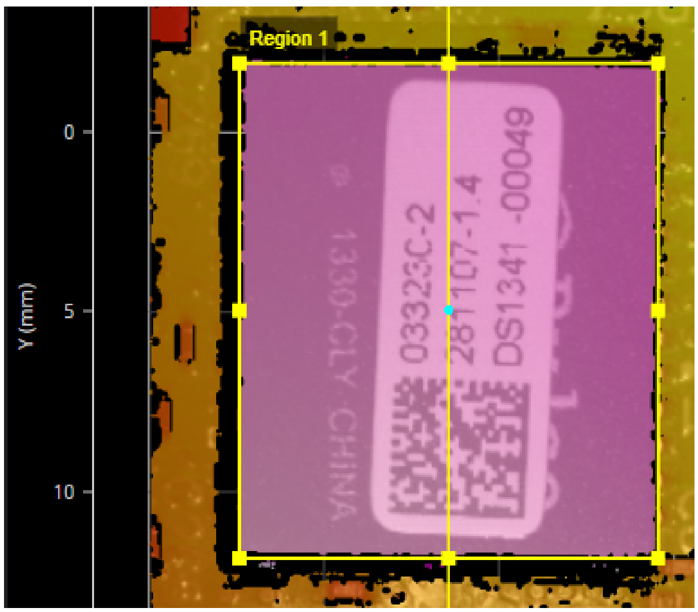




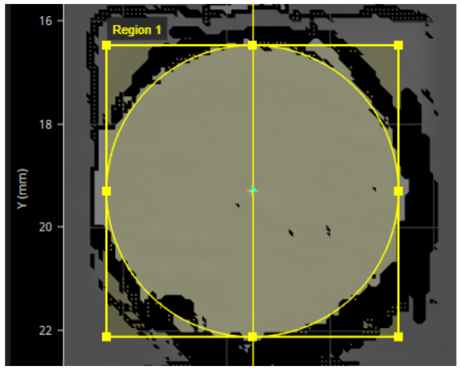
The following parameters are available in tools that support flexible regions.

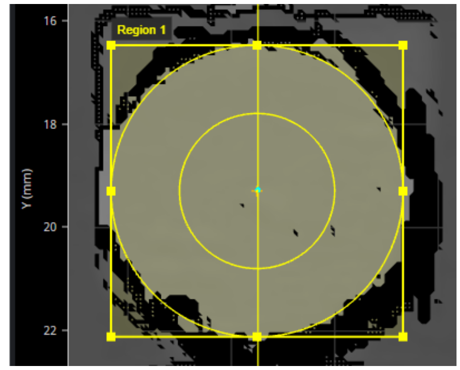
Flexible Region Parameters

Parameter	Description
Mask Mode	One of the following: **Include Data in Region**: The tool uses only the data inside the region or regions. **Exclude Data in Region**: The tool ignores the data inside the region or regions, using only the data outside.
Number of Regions	The number of regions or masks the tool uses to extract surface data. You can define up to 64 regions. This parameter is not available in some tools. To set a number of regions greater than 16, choose "Custom" in this parameter and set **Custom Region Count**. When you specify more than one region, the regions are initially stacked on top of one another, in the same location.

Parameter	Description
Custom Region Count	The number of regions (from 17 to 64) when **Number of Regions** is set to "Custom".
Region {n}	An expandable section containing region definitions for each region.
Region Type {n}	The type of each region. Regions can overlap. (For more information on the settings you use with the Circle and Ellipse types, see *Working with Circular and Elliptical Regions* on page 257.) One of the following: **Rectangle** Extracts a rectangular region from the surface data.  **Circle** Extracts a circular region from the surface data, constrained by a square region. Set the region's inner circle using the **Inner Diameter** parameter to 0 to extract circular data. Set it to a non-zero to extract annular data.

Parameter	Description



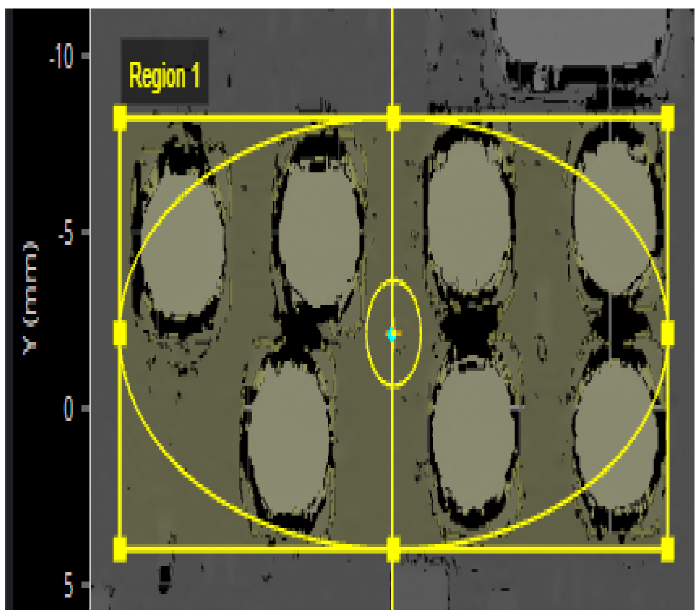


Use the **Sector Start Angle** and **Sector Angle Range** parameters to extract a partial circular or elliptical region. (See below.)

Ellipse

Extracts an elliptical region from the surface data, constrained by a square or rectangular region.

Set the region's inner ellipse (inner cyan ellipse below) using the **Inner Major Axis** and **Inner Minor Axis** parameters to extract annular data.



Use the **Sector Start Angle** and **Sector Angle Range** settings to extract a partial circular or elliptical region. (See below.)

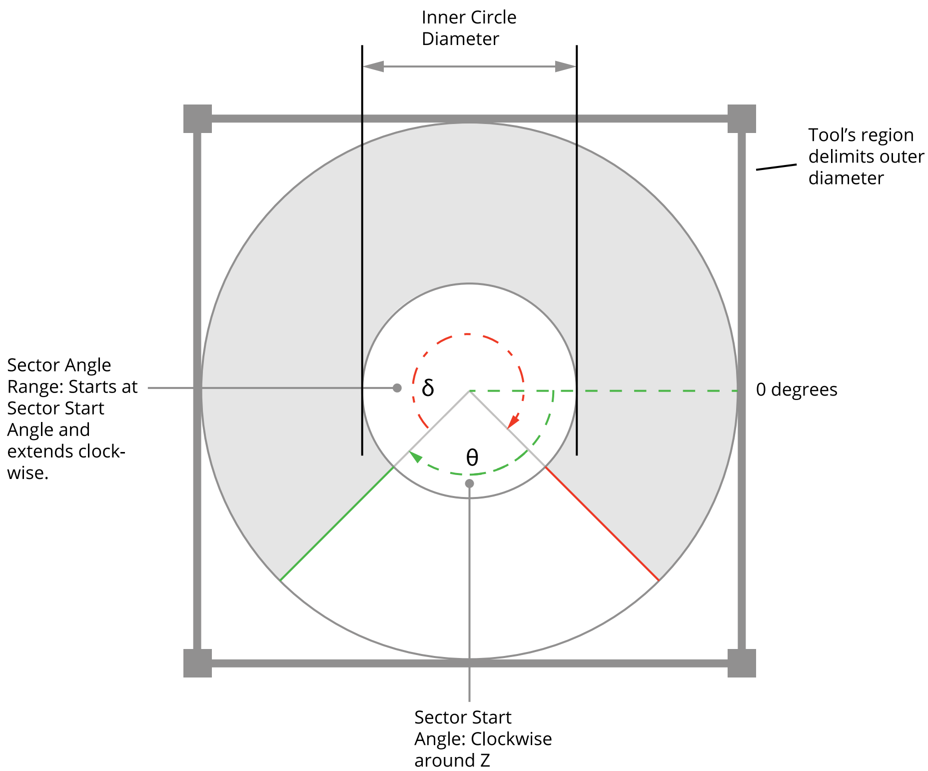
Surface

Uses the Surface data you select in **Mask Source {n}**, in the **Inputs** panel, to create a mask.

Parameter	Description
	Surface Intensity Uses the intensity data you select in **Mask Source {n}**, in the **Inputs** panel, to create a mask. Set the **Low Threshold** and **High Threshold** parameters as required.
X, Y, Z Width (X), Length (Y), Height (Z) Z Angle	These parameters define the size, location (center of the region), and orientation of the region. Not available when **Region Type {n}** is set to **Surface** or **Surface Intensity**.
Inner Diameter	Only available when **Region Type {n}** is set to **Circle**. Defines the diameter of the inner circle. Set this parameter to a value greater than 0 to extract a ring of data. Set this parameter to 0 to extract a circle of data.
Inner Major Axis Inner Minor Axis	Only available when **Region Type {n}** is set to **Ellipse**. These parameters define the major and minor axes of the inner ellipse, respectively Set this parameter to a value greater than 0 to extract a ring of data. Set this parameter to 0 to extract an elliptical disk of data.
Sector Start Angle Sector Angle Range	Only available when **Region Type {n}** is set to **Circle** or **Ellipse** Use these parameters together to extract a partial ring of data. **Sector Start Angle** controls the starting angle of the data, whereas **Sector Angle Range** controls the length of the arc. Note that the angles and ranges in these parameters are measured clockwise around Z, where 0 degrees is along the positive X axis. For more information on how these settings work together, see *Working with Circular and Elliptical Regions* below.
High Threshold Low Threshold	The high and low thresholds the tool uses in combination with the intensity mask. Only available when **Region Type {n}** is set to **Surface Intensity**.

Working with Circular and Elliptical Regions

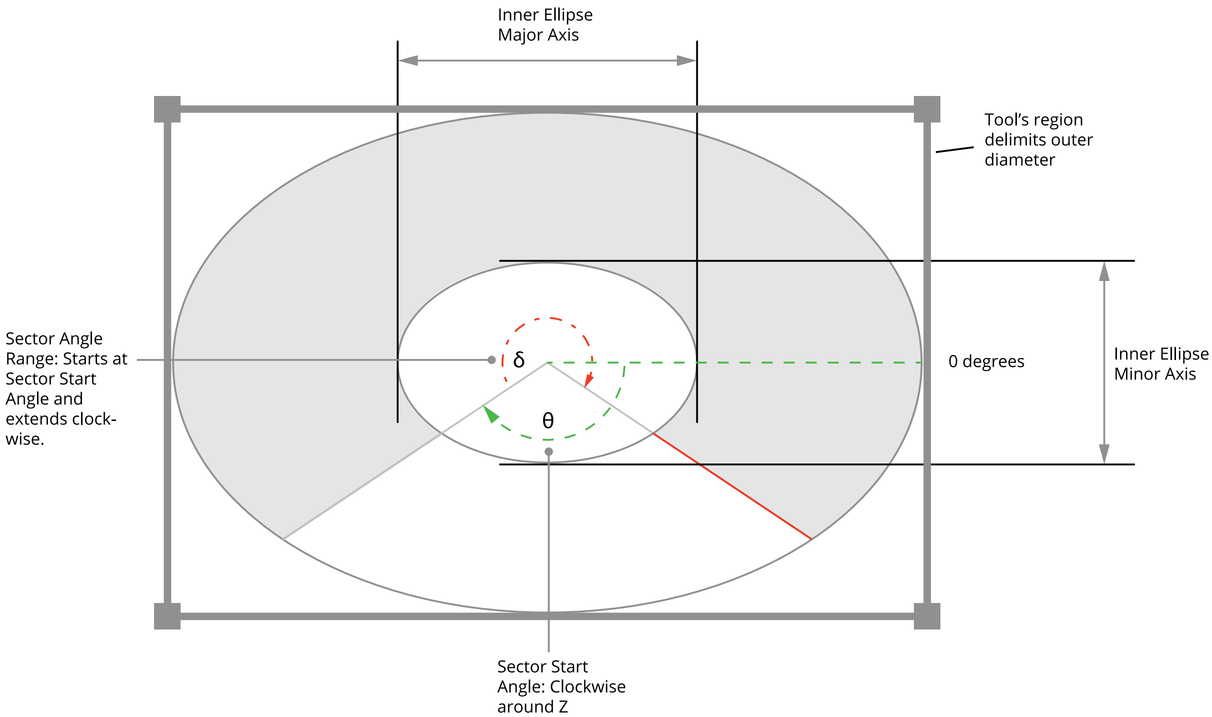
When you set a region's type to Circle or Ellipse, the tool displays several additional settings that work together to define the region. **Sector Start Angle** and **Sector Angle Range** work together to define the start and end of a partial circular/elliptical region (solid or annular). A region will be annular if **Inner Circle Diameter** is non-zero. Note that the "length" of the partial region extends *from* the start angle. In the following illustration, the start angle (θ) is 135 degrees relative to the 0-degree point indicated below, and the region extends 270 degrees (δ) from that, clockwise around Z.



Sector Start Angle starts at the 0-degree point around Z.

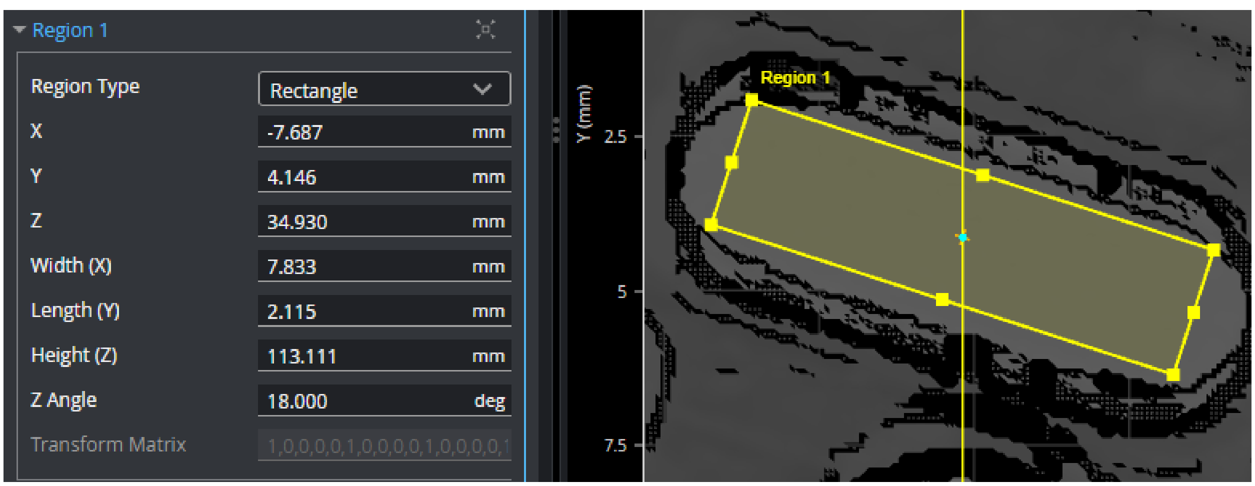
Note that the angles defining a partial circular/elliptical region are relative to the *region*, and not the sensor's coordinate system. So a region rotated 30 degrees using its **Z Angle** setting rotates the start angle and angle range by 30 degrees.

When you set a region type to Ellipse, instead of the inner circle diameter, you must set the major and minor axes of the inner ellipse.



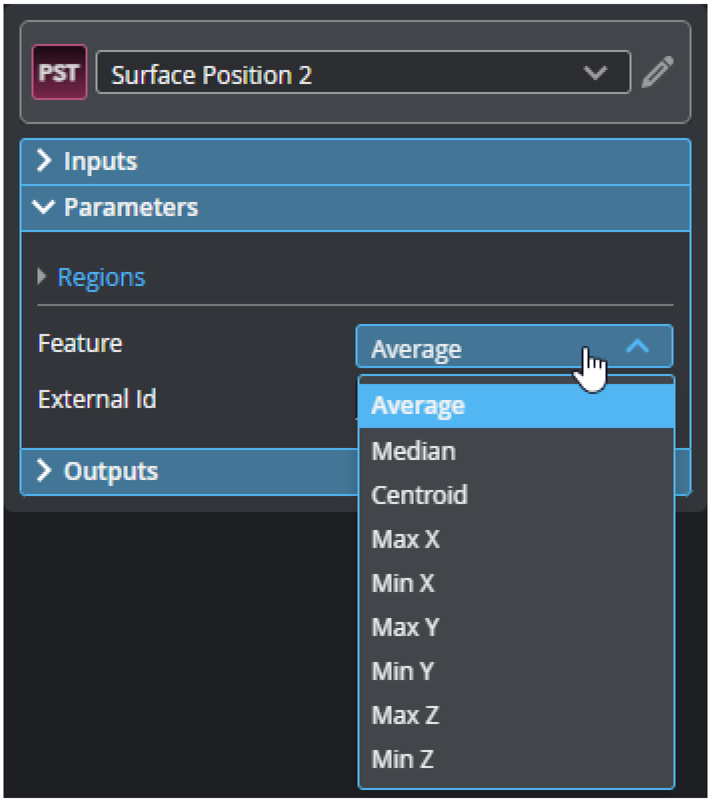
Region Rotation

The measurement region of some tools can be rotated by setting the region's **Z Angle** to better accommodate features that are on an angle on a target. By rotating the measurement region, data not related to the feature can often be excluded, improving accuracy of measurements. You can modify the size and position of a rotated region.



Feature Points

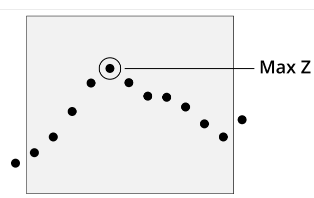
Dimensional and positional measurements detect *feature points* found within the defined region (whether it's a measurement region, feature region, reference region, and so on) and then compare measurement values taken at the selected point with minimum and maximum thresholds to produce a pass / fail decision. Feature points are selected in one or more **Feature** parameters in a tool's **Parameters** section, and are used for all of the tool's measurements.

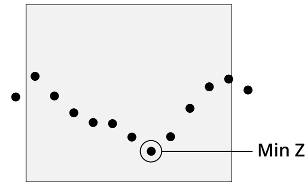
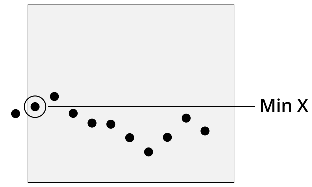
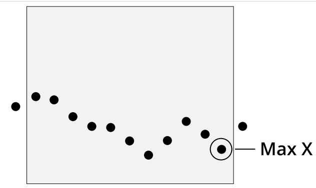
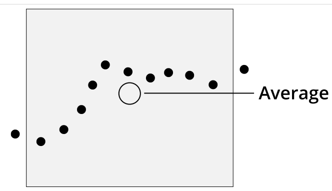
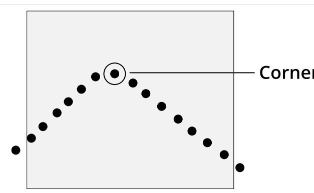
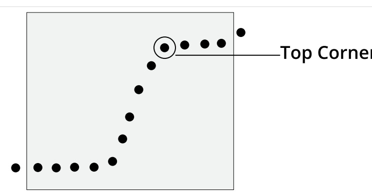
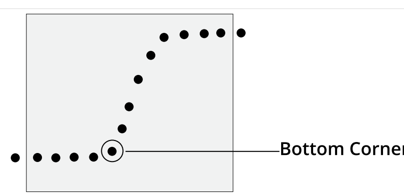


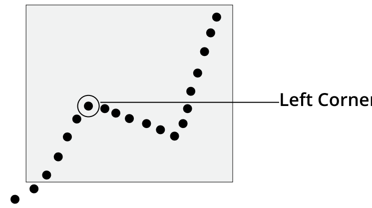
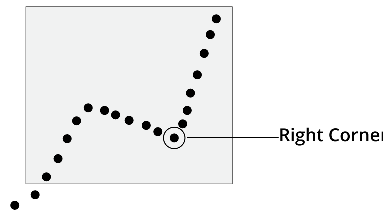
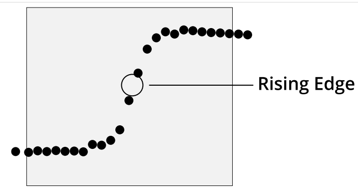
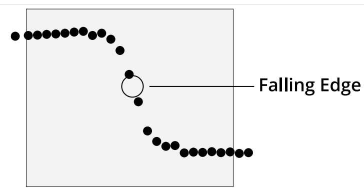
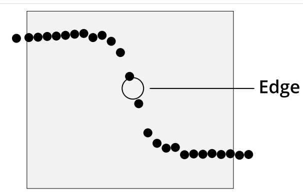
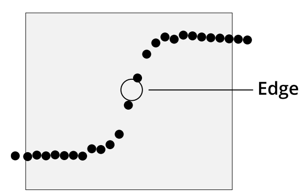
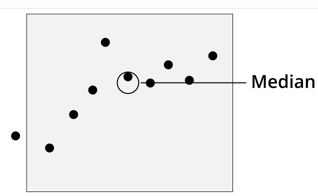
It is important to note that the feature points described here are distinct from the geometric features generated by some tools, and which can be taken as inputs by other tools. Feature points are simply the location in the scan data at which a measurement tool performs a measurement. So, for example, if you set a Profile Position tool's **Feature** parameter to Max Z, it will return the X and Z positions of the point that is at the highest Z value. Geometric features are a data type produced as output or taken as input by a tool. (For more information on geometric features, see *Geometric Features* on page 262.)

The following types of points can be identified in a measurement region. The availability of a feature point type is tool-dependent.

	The following are represented as profiles for clarity. Many Surface tools use feature points.

Point Type	Examples
Max Z Finds the point with the maximum Z value in the region of interest.	

Point Type	Examples
Min Z Finds the point with the minimum Z value in the region of interest.	
Min X Finds the point with the minimum X value in the region of interest.	
Max X Finds the point with the maximum X value in the region of interest.	
Average Determines the average location of points in the region of interest.	
Corner Finds a dominant corner in the region of interest, where corner is defined as a change in profile slope.	
Top Corner Finds the top-most corner in the region of interest, where corner is defined as a change in profile shape.	
Bottom Corner Finds the bottom-most corner in the region of interest, where corner is defined as a change in profile shape.	

Point Type	Examples
Left Corner Finds the left-most corner in the region of interest, where corner is defined as a change in profile shape.	
Right Corner Finds the right-most corner in the region of interest, where corner is defined as a change in profile shape.	
Rising Edge Finds a rising edge in the region of interest (moving from left to right).	
Falling Edge Finds a falling edge in the region of interest (moving from left to right).	
Any Edge Finds a rising or falling edge in the region of interest.	 
Median Determines the median location of points in the region of interest.	

Geometric Features

Most Surface tools, and many Profile tools, can output features that Feature tools can take as input to produce measurements. These features are called *geometric features*. Feature tools use these entities to produce measurements based on more complex geometry. (For more information on Feature tools, see *Feature Measurement* on page 692.)

The measurement tools can currently generate the following kinds of geometric features:

Points: A 2D or 3D point. Can be used for point-to-point or point-to-line measurements.

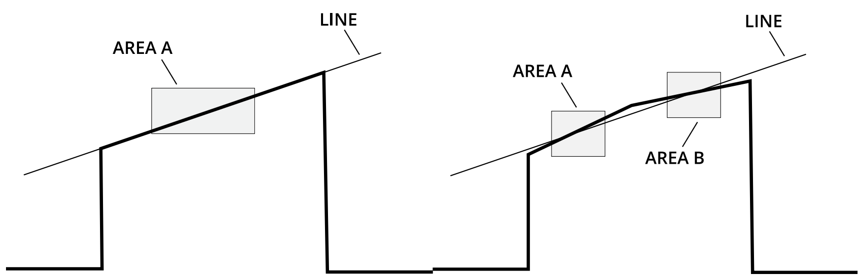
Lines: A straight line that is infinitely long. Useful for locating the orientation of an enclosure or part, or to intersect with another line to form a reference point that can be consumed by a Feature tool.

Planes: A plane extracted from a surface. Can be used for point-to-plane distance or line-plane intersection measurements.

Circles: A circle extracted from a sphere.

Fit Lines

Some measurements involve estimating lines in order to measure angles or intersection points. A fit line can be calculated using data from either one or two fit areas.

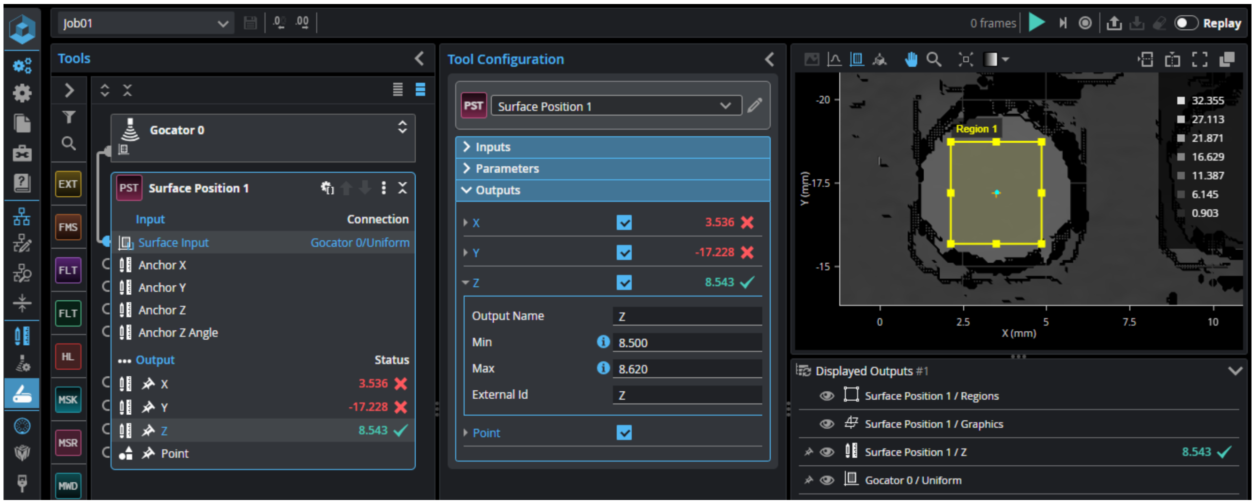


A line can be defined using one or two areas. Two areas can be used to bypass discontinuity in a line segment.

Decisions

Results from a measurement can be compared against minimum and maximum thresholds you set in a tool's **Outputs** panel to generate *pass* / *fail* decisions. The decision state is *pass* if a measurement value is within the minimum / maximum range; the range is inclusive (a measurement value equal to a Min or Max threshold value is a pass). Output values that pass are green in the interface. Output values that fail are red in the interface.

Note that measurement values are output with six decimal places. However, you can configure the GoPxL interface to hide some decimal places, using the "Decrease precision" and "Increase precision" buttons (and , respectively), from zero to six decimal places. Keep this in mind when configuring thresholds.



*Decision maximum and minimum of Surface Position Z set in the **Tool Configuration** panel (center). Pass values for this measurement are displayed in green. Fail measurements are displayed in red.*

Decisions (and measurements) can be sent to external programs and devices. For more information on sending values and decisions as output, see *Configuring Control* on page 793.

Measurement Anchoring

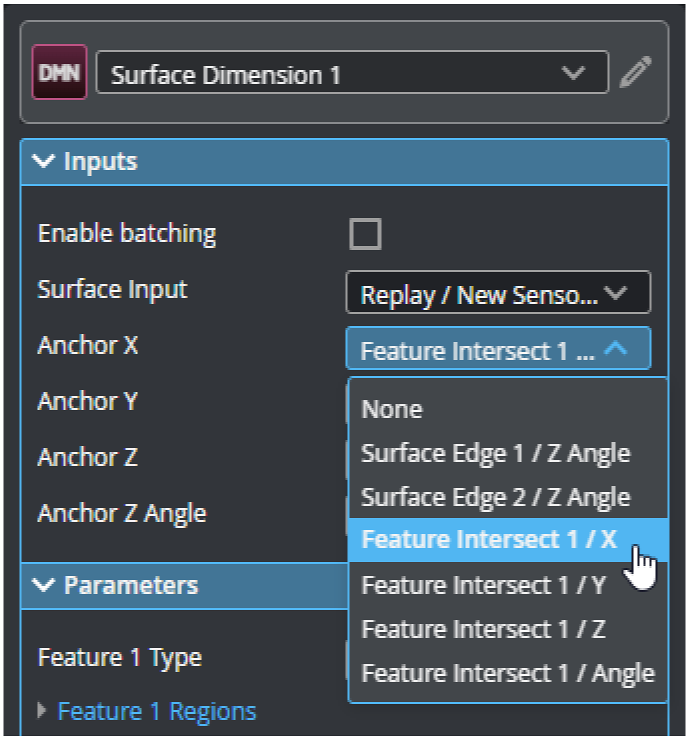
The position of parts moving on a transport mechanism such as a conveyor typically changes from part to part in one or both of the following ways:

- along the X, Y, and Z axes
- around the Z axis (orientation angle)

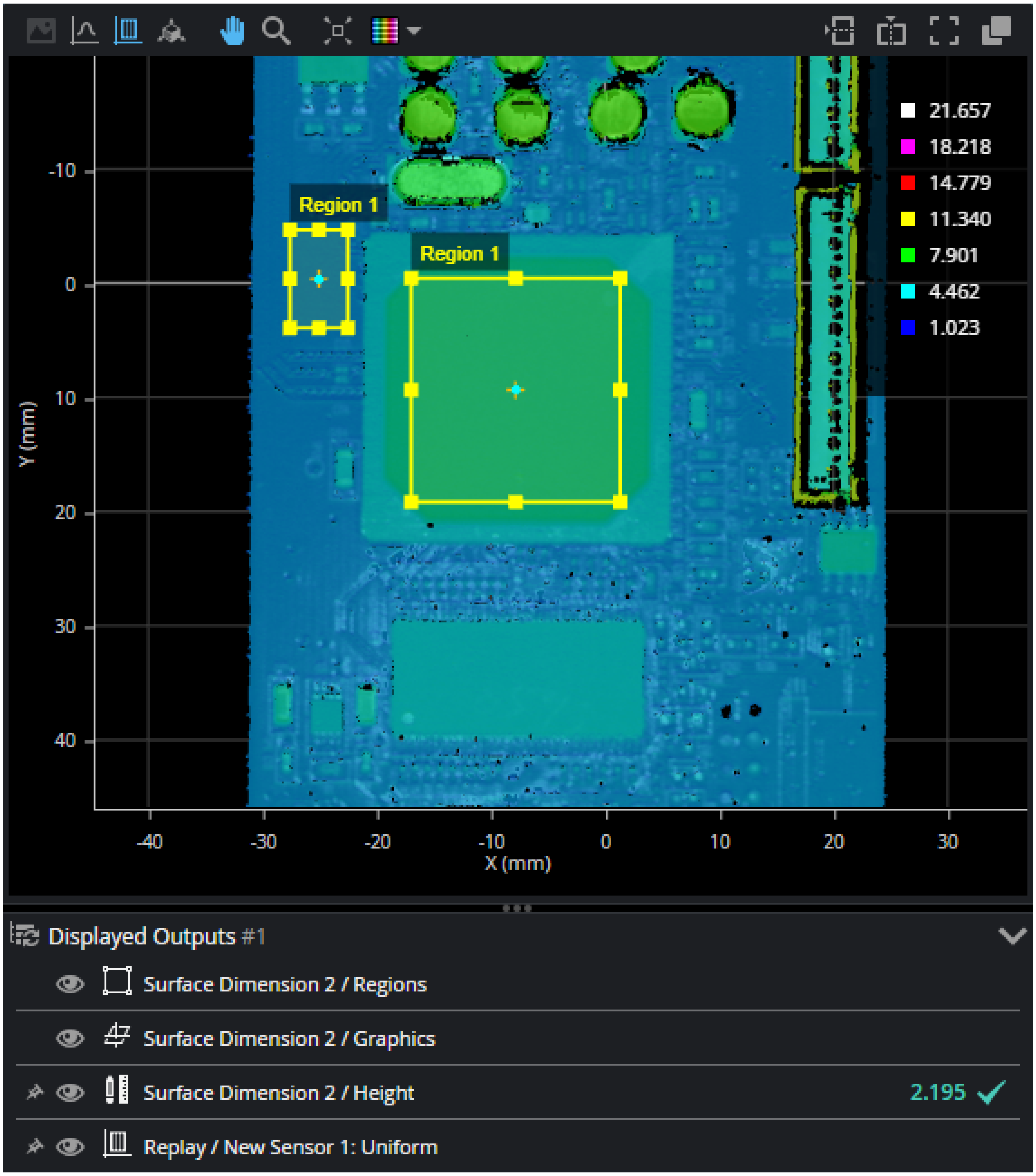
When the position and angle variation between parts is minor—for example, when scanning electronic parts in trays—you can anchor one tool to one or more measurements from another tool to compensate for these minor shifts. As a result, GoPxL can correctly place the anchored tool's measurement regions on each part. This increases the repeatability and accuracy of measurements.

> For cases where movement from part to part is more drastic, you can often use the Surface Pattern Matching tool to compensate. For more information, see *Surface Pattern Matching* on page 599.

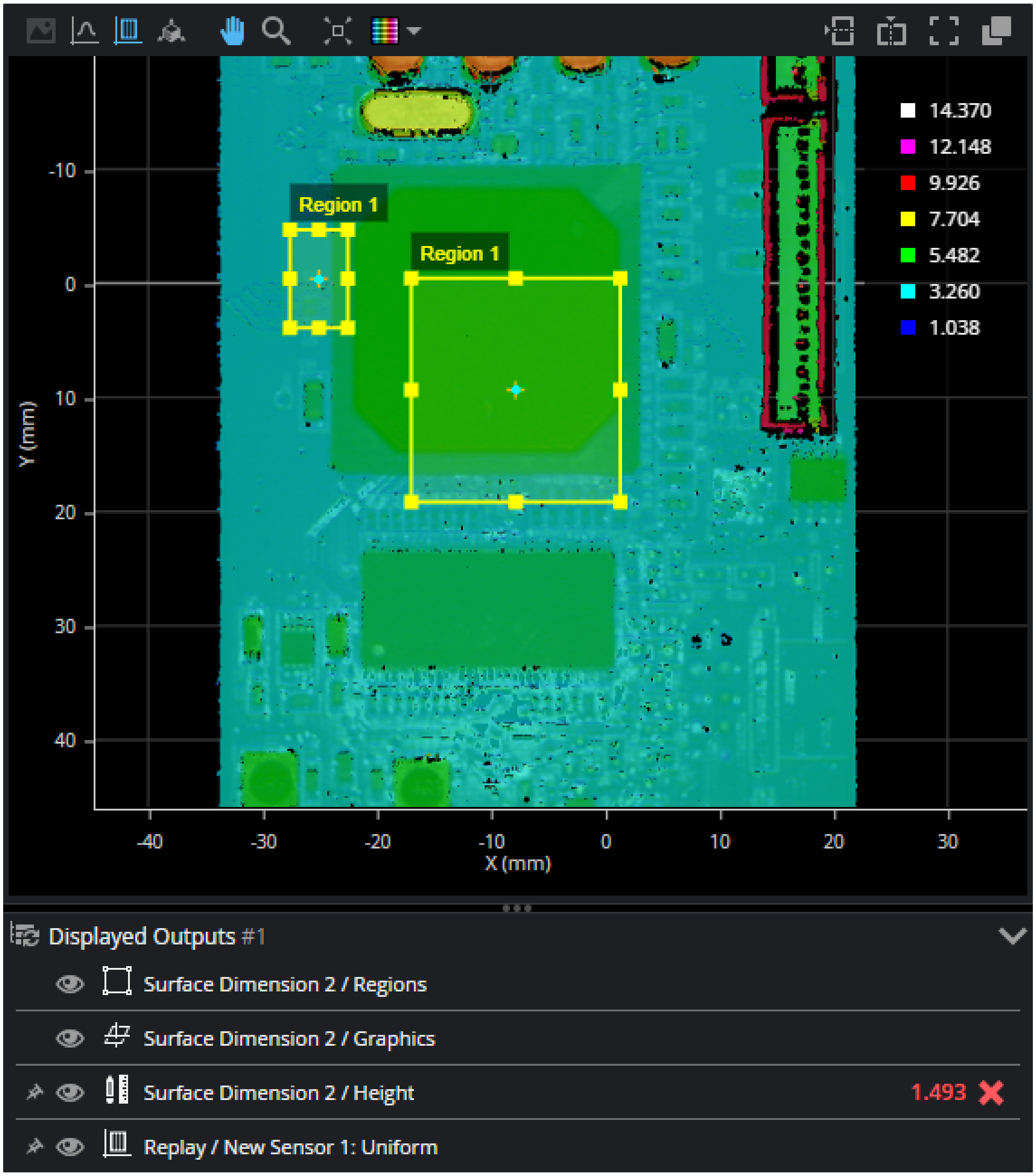
You set a tool's anchors in its **Inputs** section, by setting **Anchor X**, **Anchor Y**, **Anchor Z**, or **Anchor Z Angle** to the appropriate measurement from another tool:



For example, the following shows a surface scan of a PCB. A Surface Dimension height measurement returns the height of a component in the measurement region relative to a nearby reference region; the height is between the measurement's thresholds and is a pass (the green value in the **Displayed Outputs** pane).

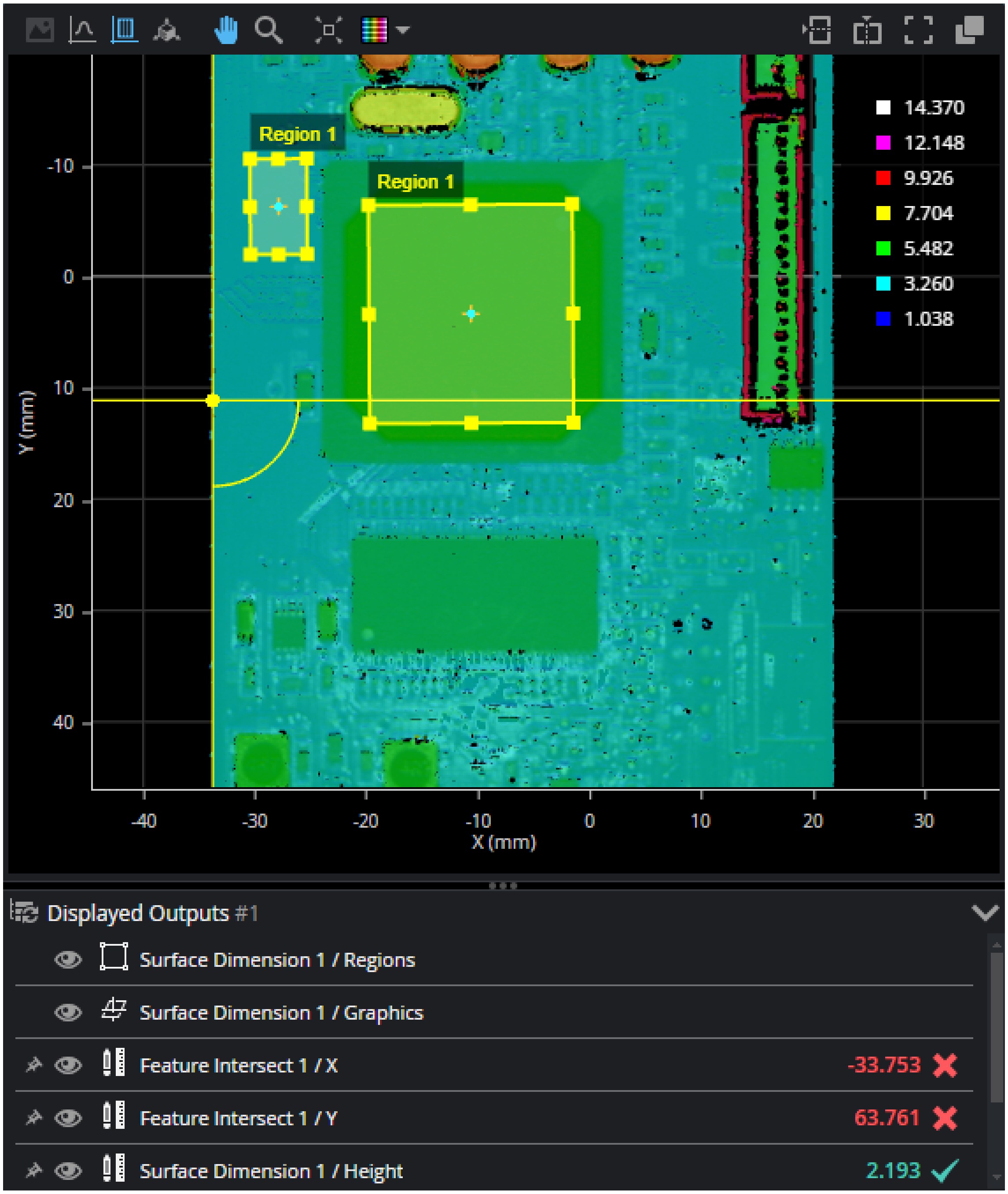


In the following scan, the part has shifted, but the measurement regions remain where they were originally configured, in relation to the sensor or system coordinate system, so the measurement returned is incorrect, and the measurement fails:



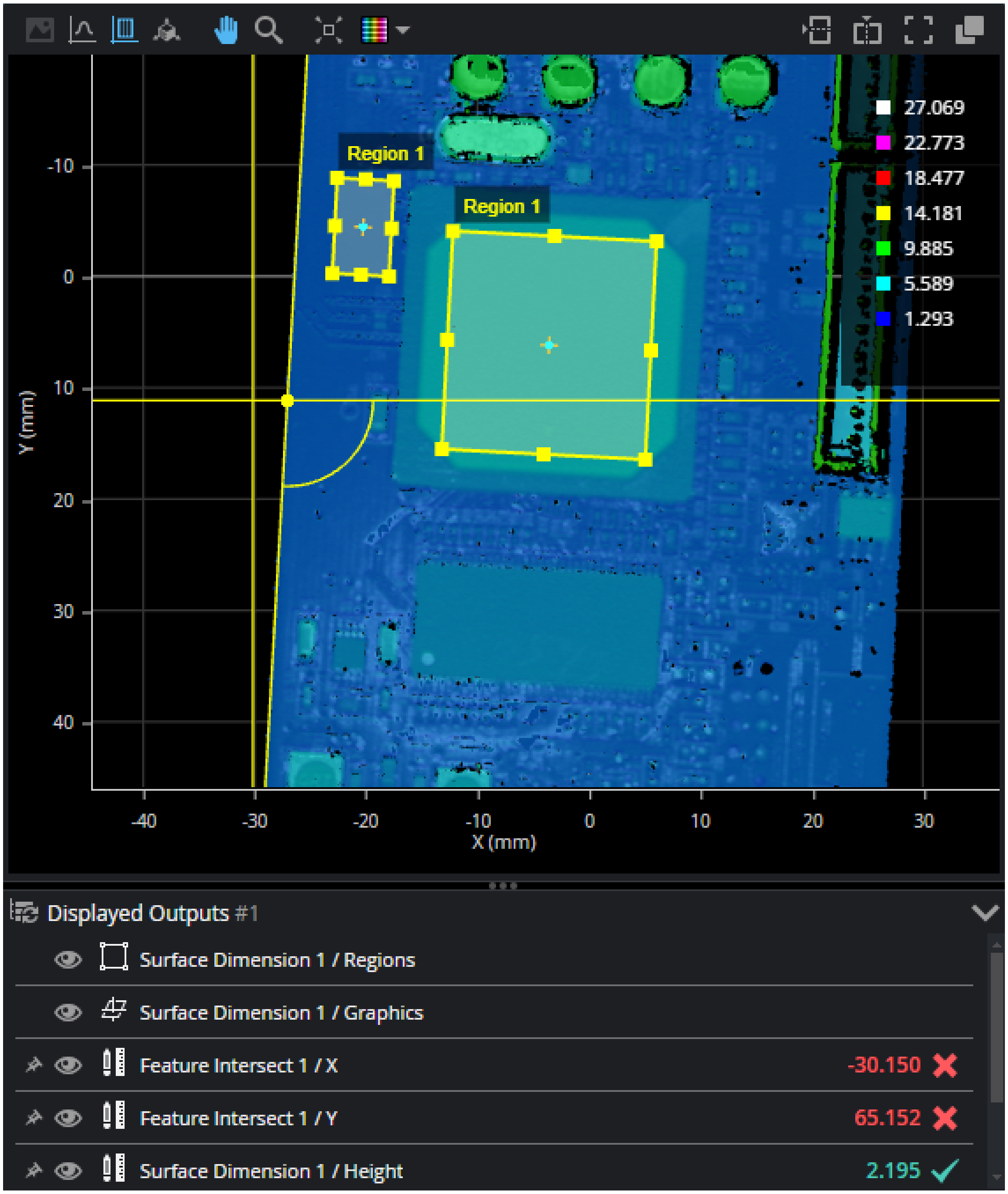
When you set any of a tool's anchor sources, an offset is calculated between the anchored tool and the anchor source. This offset is used for each frame of scanned data: the anchored tool's measurement region is placed in relation to the anchor source, at the calculated offset.

In the following image, after the Surface Dimension tool is anchored to the X and Y measurements from two Surface Edge tools (placed over the left and bottom edges of the PCB), GoPxL compensates for the shift—mostly along the X and the Y axis in this case—and returns a correct measurement, despite the shift.



You can combine the positional anchors (X, Y, or Z measurements) with an angle anchor (a Z Angle measurement) for optimum measurement placement. For example, in the following scan, the part has not only shifted on the XY plane but also rotated around the Z axis. Anchoring the Surface

Dimension tool to the Z Angle measurement of a Surface Edge tool compensates for the rotation, and the anchored tool returns a correct measurement.



	If Z Angle anchoring is used with both X and Y anchoring, the X and Y anchors should come from the same tool.

	If Z Angle anchoring is used without X or Y anchoring, the tool's measurement region rotates around its *center*. If only one of X or Y is used, the region is rotated around its center and then shifted by the X or Y offset.

Working with Tool Chains

The Tools Diagram panel between the tool list and Tool Configuration panel, provides a visual representation of the data flow in a system (the output from a sensor group, and the input and output of tools). It lets you create and view complex tool chains with drag-and-drop and other mouse operations, letting you quickly and easily implement and maintain applications requiring multiple, interconnected tools.

All data types (if available) and their relationships between tools are displayed:

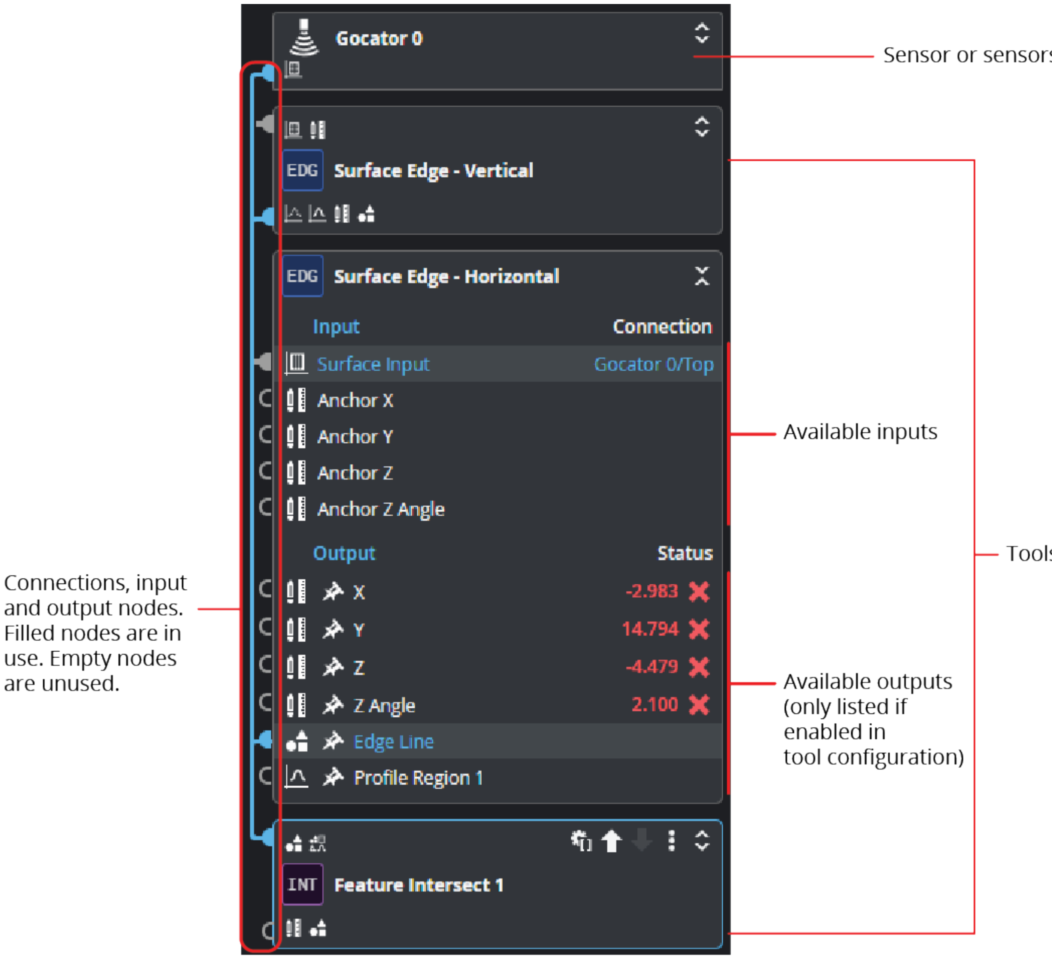
- Profile data (either directly from a sensor group's output or from tool output)
- Surface data (either directly from a sensor group's output or from tool output)
- Measurements (for use as anchors)
- Geometric features
- Tool data (some data outputs are intended to be consumed only by SDK applications and can't be used as part of a tool chain)

For details on how the Tools Diagram panel displays information, see *Understanding the Data Flow in Tool Chains* on page 275.

For details on how to connect and disconnect, see *Connecting Tools* on page 280 and *Disconnecting Tools* on page 282.

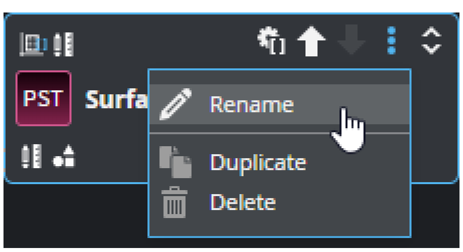
The Tools Diagram panel is open by default. The inputs, parameters, and outputs of the tool selected in the Tools Diagram panel are displayed to the right of the Tools Diagram panel.

The following shows the main features of the Tools Diagram panel.



Tools Diagram panel showing sensor group, tools, outputs/inputs, and data flow connections.

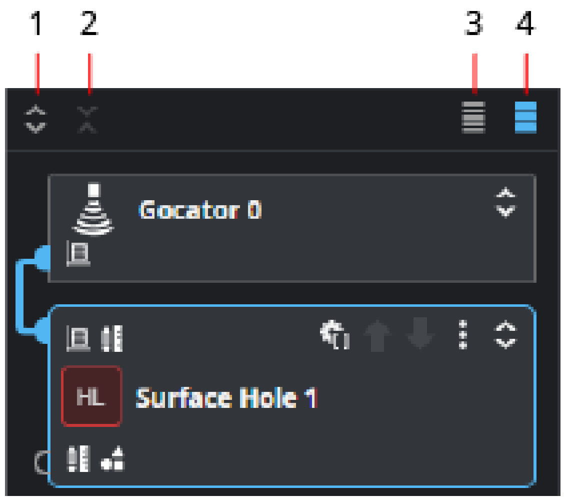
At the top of a tool, an action menu provides functions to rename, duplicate, and delete the current tool.



Action menu (collapsed tool)

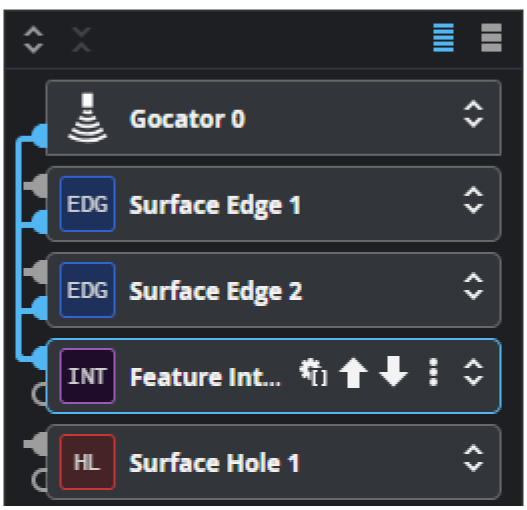
Changing Tool Display Options

The buttons at the top of the Tools Diagram panel let you control how the panel displays sensor groups, tools, and the data flow (tool chain). Buttons at the top of individual tools let you organize the tools in the list, as well as name, duplicate, and delete them.

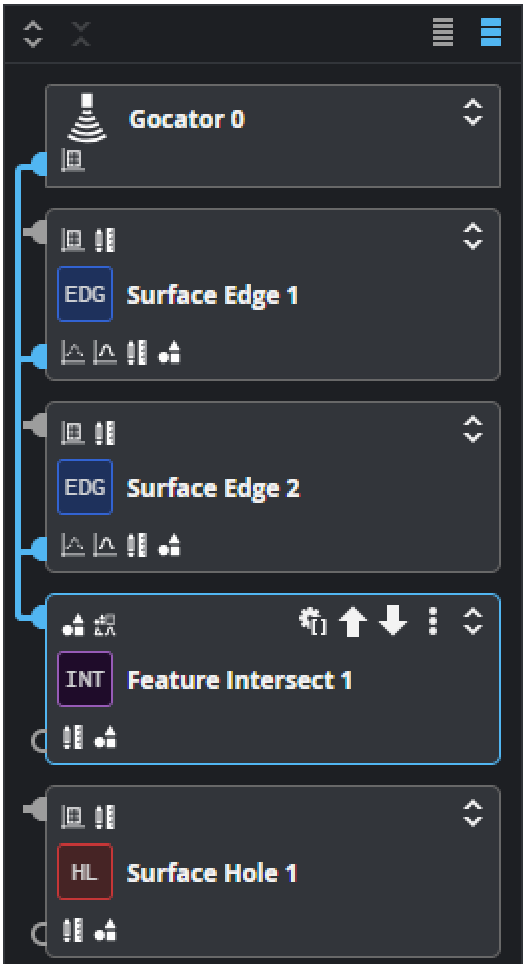


The following describes the toolbar's functions:

1. Open all: Expands the sensor or sensor group block and all tools in the Tools Diagram panel, displaying a list of available inputs and enabled outputs for each one.
2. Close all: Collapses all items in the Tools Diagram panel.
3. Compact view: Hides the list of small input and output icons that indicate the types of the inputs and outputs the sensor or a tool has.



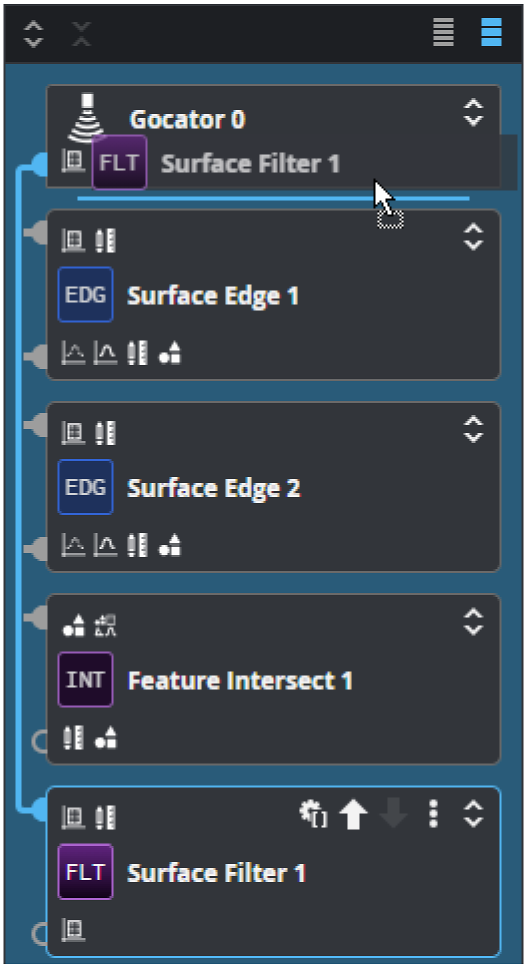
4. Standard view: Shows small icons that indicate the types of the inputs and outputs the sensor group or a tool has. For a list of inputs and outputs, see *Data Types* on the next page.



Reordering Tools

You can reorder tools in the Tools Diagram to organize them more logically or to better represent the data flow. For example, you could group tools that output geometric features with the tools that use them, or group tools you use as anchors with the tools that use those anchors. Note that the order of tools in the Tools Diagram panel does not affect the flow.

You can reorder tools you have added by dragging and dropping them to the position you want in the list of tools, or by using the up and down arrows.

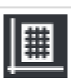
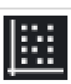


Moving the Surface Filter tool to the top of the tool chain.

Data Types

GoPxL represents the data types of inputs and outputs in the Tools Diagram panel with an icon.

Note that most types can be modified with a "[ ]" badge, which means the tool can be batched (blue) or the input / output is an array (yellow).

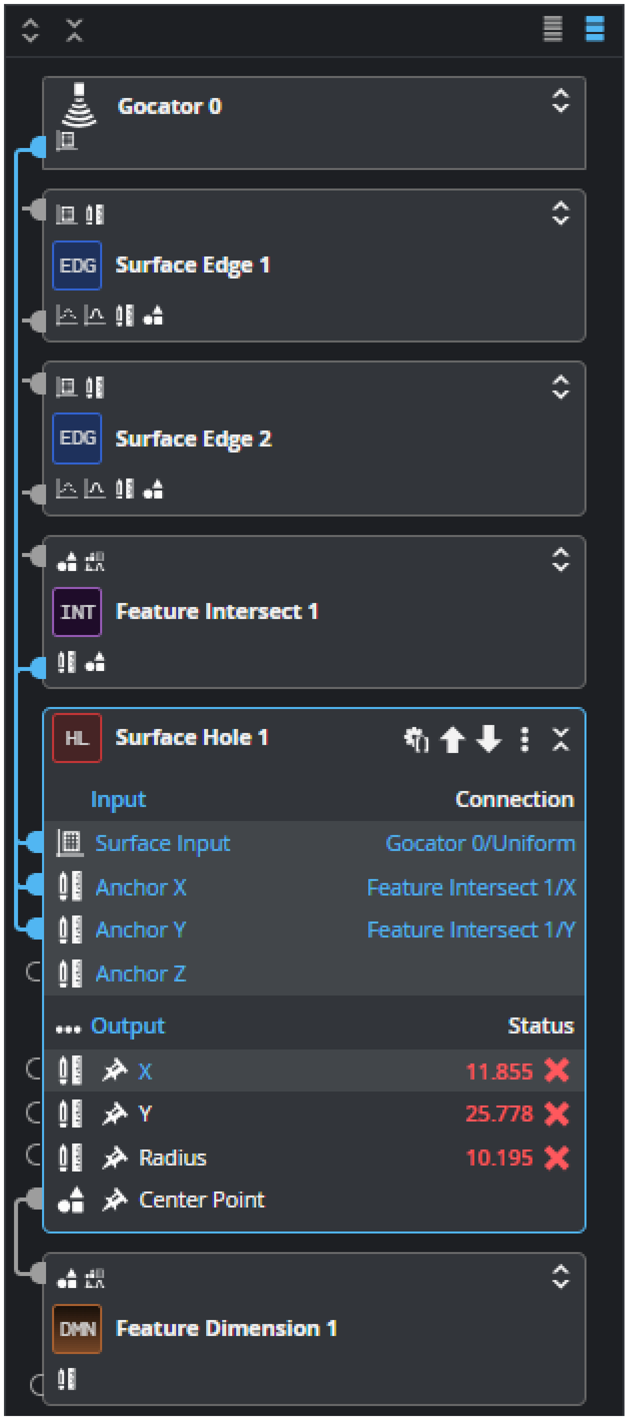
Icon	Description
	Uniform Surface data
	Point Cloud Surface data (non-uniform)
	Uniform Profile data

	Point Cloud Profile data (non-uniform)
	Measurement
	Geometric feature
	Multiple input types

Understanding the Data Flow in Tool Chains

The rectangular blocks displayed in the Tools Diagram represent a sensor or group of sensors at the top, and any tools you have added below that. Sensor groups display output connection nodes, whereas tools display both input and output connection nodes.

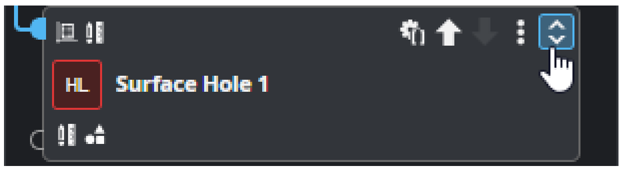
The appearance of nodes changes depending on whether they are connected and whether they are selected. Connections that are used are sold. Connections that are not used are empty. When a sensor or tool is expanded, you can see which specific inputs or outputs are used as part of the tool chain. For example, in the expanded Surface Hole tool below, we can see that the first three inputs (*Surface Input* and the X and Y anchors, receiving their input from the sensor group at the top and from Feature Intersect, respectively) and the Center output are involved in the chain of sensor group and tools.



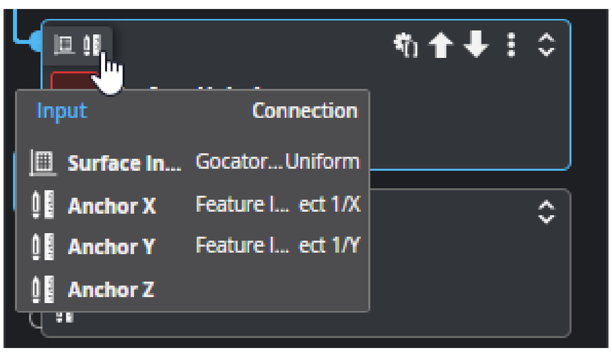
When a tool is collapsed, however, you only know that at least one input or output is used (or none at all). For example, looking at the collapsed Feature Dimension tool at the bottom, we know that at least one input (the connection node at the top) is used, and that none of the tool's outputs are used. Also, we know that inputs and outputs of the three collapsed tools at the top are used, but not exactly which ones.

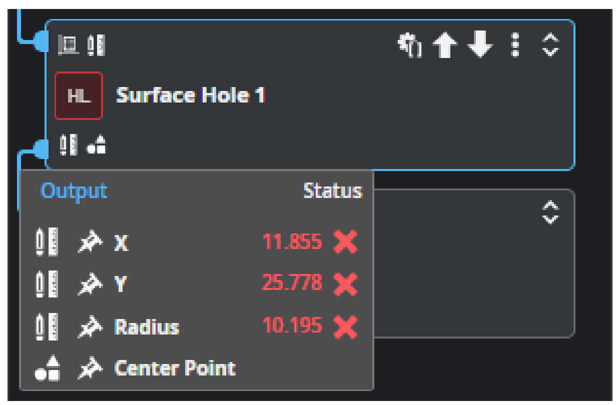
In both cases (collapsed or expanded), the data flow of the selected tool or sensor block is indicated by blue connection lines. For more information, see below.

By default, sensors and tools are collapsed, but you can expand them individually by clicking the expand / collapse button at the top right of a tool to display the complete list of available inputs and outputs. Note that for an output to be listed in the Outputs section, it must be enabled in the tool's configuration: in the tool's Output list, only enabled outputs are listed.

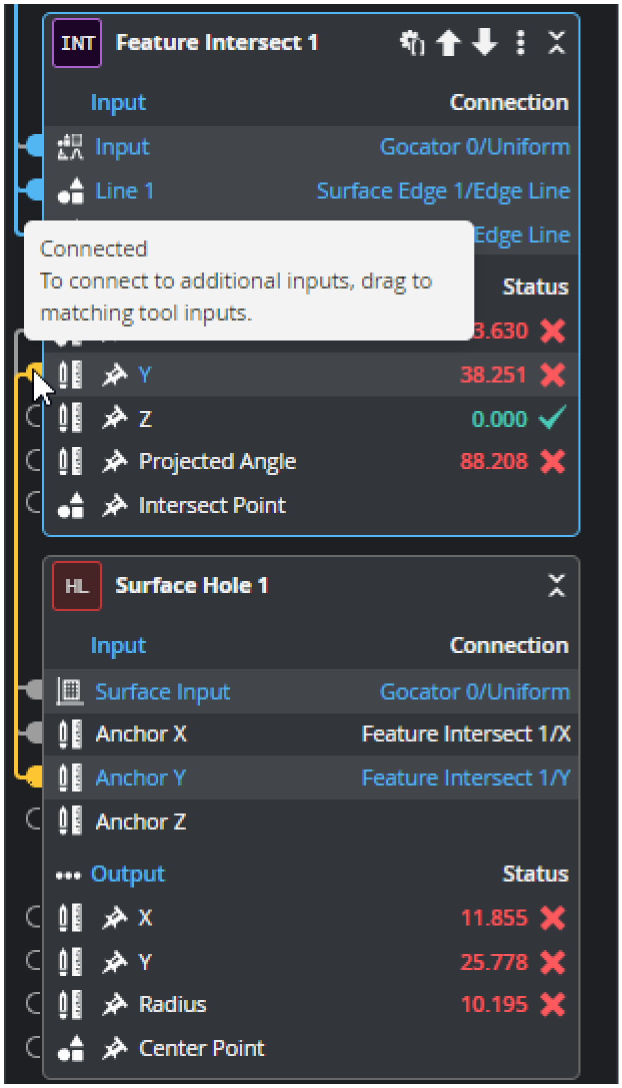


When a tool is collapsed, you can see which inputs are connected and to what source, and also see which outputs are enabled and their values, by clicking one of the rows of small icons at the top or bottom of the block. You can also pin outputs from here; for more information on pinning, see *Pinning Outputs* on page 107.

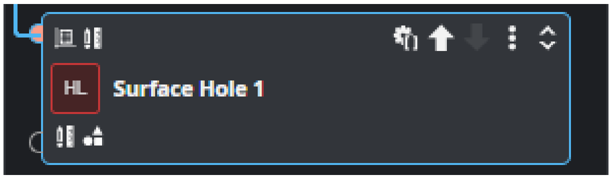


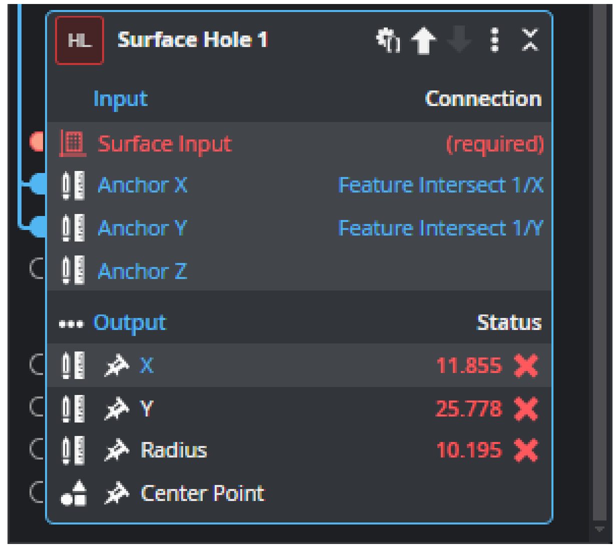


If you hover the mouse pointer over a blue connected node, a part of the blue connection line is highlighted to indicate what it is connected to. In the image below, you can see that by hovering over an output (the Y measurement of the Feature Intersect tool at the top) is used as an input (the Y anchor) of the Surface Hole tool at the bottom.



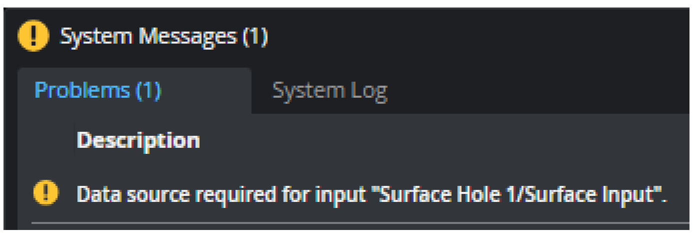
If a tool's input is missing, that input is displayed in red in the Tools Diagram panel to show that you must connect it to something.



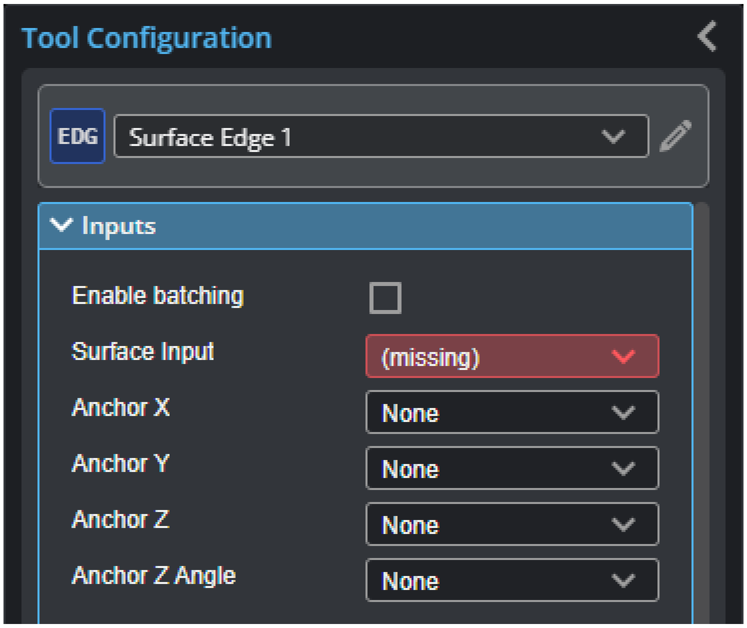


Collapsed tool and expanded tool with a missing input

You can also consult the **Problems** tab in the **System Messages** panel at the bottom of the interface to see which tools are missing inputs.



In the Tool Configuration panel, a tool whose input is no longer available (for example, you have removed the preceding tool in the tool chain) displays "(missing)" in the input parameter.



Connecting Tools

The Tools Diagram panel lets you quickly connect tools using drag-and-drop operations.

In the following, we connect a measurement output from one tool to an anchor input of another tool. However, the same procedure applies when connecting other kinds of outputs to inputs, such as when connecting a geometric feature output from one tool to the input of another tool, or when connecting Surface output (such as the output from the Surface Filter tool) to the input of another tool.

To connect a tool's output to another tool's input

1. Make sure you have added at least two tools and that you have configured the tools higher in the tool chain.

   The output you want to connect must be enabled in the first tool. You can enable a tool's output in the tool's **Outputs** section in the **Tool Configuration** panel, or by using the **Output** menu in the tool's block in the Tools Diagram panel.

   For information on adding tools, see *Understanding the Data Flow in Tool Chains* on page 275.

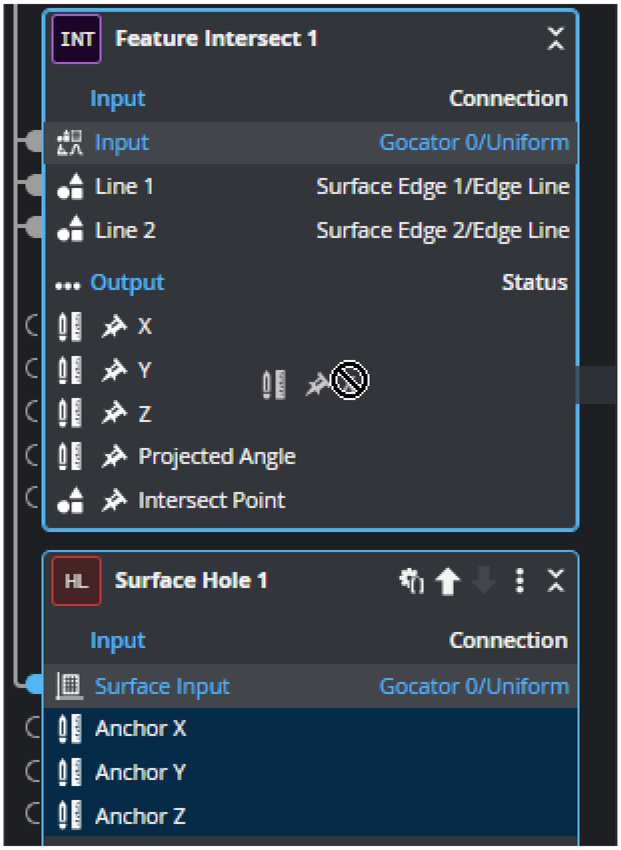
2. Locate the tool whose output you want to use (the "source" tool).

3. Do one of the following:

   From an expanded tool

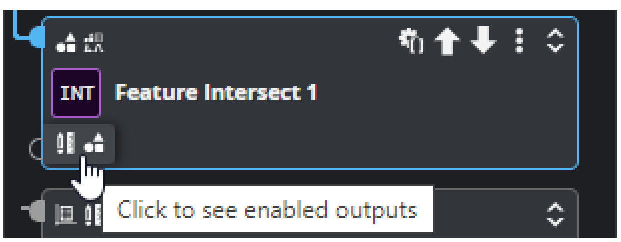
   a. Click and hold the output you want to connect to the other tool's input and drag it to the input.

      Here, we are dragging the X measurement from the Feature Intersect tool. Tools with compatible inputs (here, the anchor inputs of the Surface Hole tool), are highlighted with a dark color.

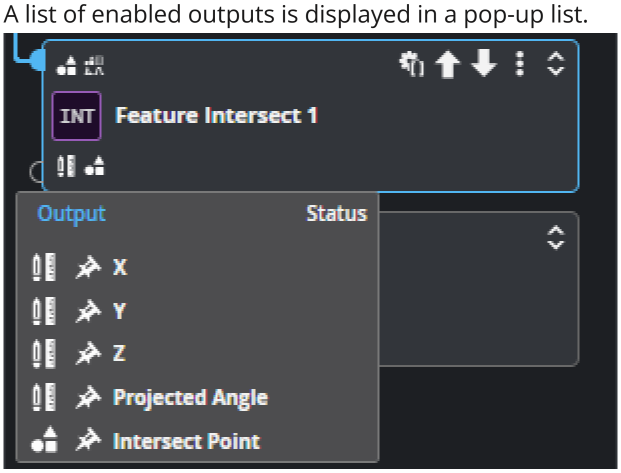


From a collapsed tool

a. Click the small row of icons at the bottom of the tool to expand the list of

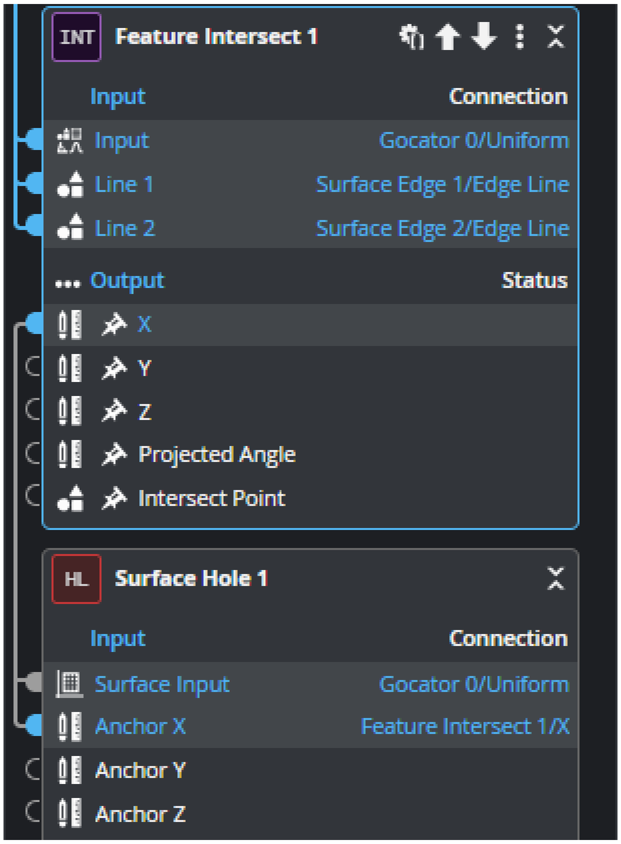


A list of enabled outputs is displayed in a pop-up list.



4. Drop the output on the desired input.

   A new connection appears between the first tool's output and the second tool's input (below, between the Surface Hole tool's Center Point output and the Point input in the Feature Dimension tool).

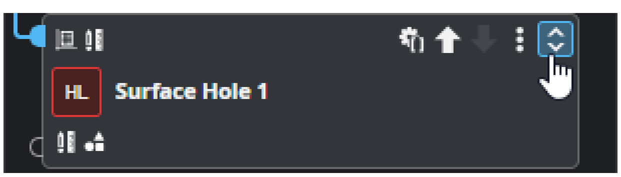


Disconnecting Tools

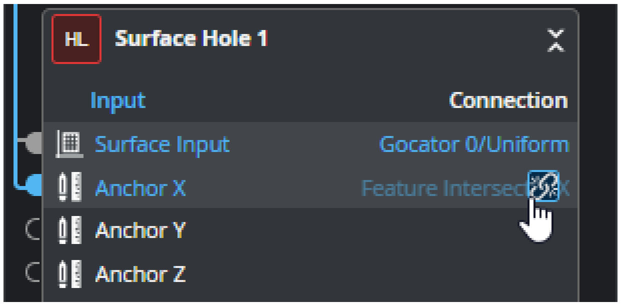
You can quickly disconnect an input in the Tools Diagram panel, but only if the tool containing the input is expanded.

To disconnect an input in a tool:

1. If the tool isn't expanded, click the Expand button at the top of the tool.



2. In the expanded tool, move the mouse pointer over the input you want to disconnect and move it to the right until the pointer is over the Disconnect icon.



3. Click the Disconnect icon.

   The input is disconnected from the other tool's output.

Profile Measurement

This section describes the profile measurement tools available in Gocator sensors.

Only a subset of the Profile tools is available when **Enable uniform spacing** is disabled, that is, when the sensor is producing point cloud data.

For more information on the **Uniform Spacing** setting and resampled data, see *Uniform Data and Point Cloud Data* on page 70.

Profile measurement tools can be used on sections extracted from Surface data. For more information on sections, see *Surface Section* on page 628.

Profile Advanced Height

The Advanced Height tool provides highly accurate and repeatable master (template) comparison and step height measurements (up to 16 in a tool instance).

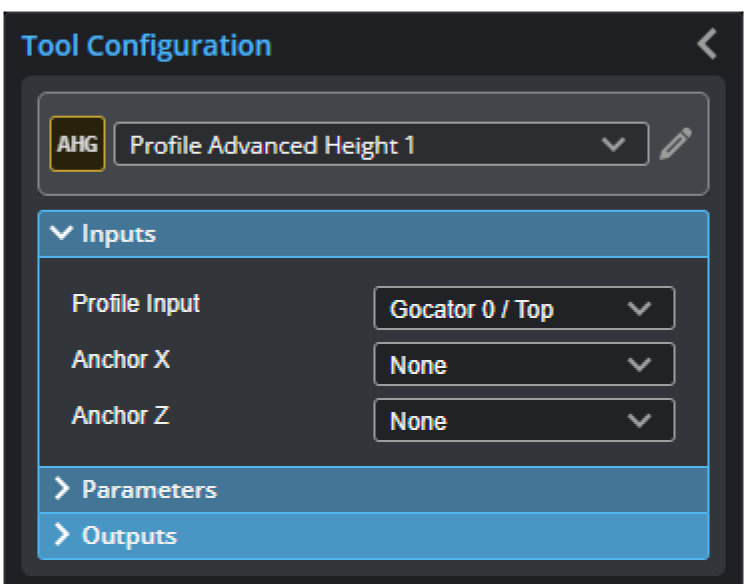
> All instances of the Advanced Height tool share the same template file set in **File**. For this reason, you must be careful when editing or removing template files shared by another instance of the tool.

Height measurements can be made relative to a reference line. Reference line sets the measurement direction (perpendicular to the reference line). A separate base line can also be set so that height measurements are between the base line and a profile feature, rather than the reference line (which in this case is used for angle correction).

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



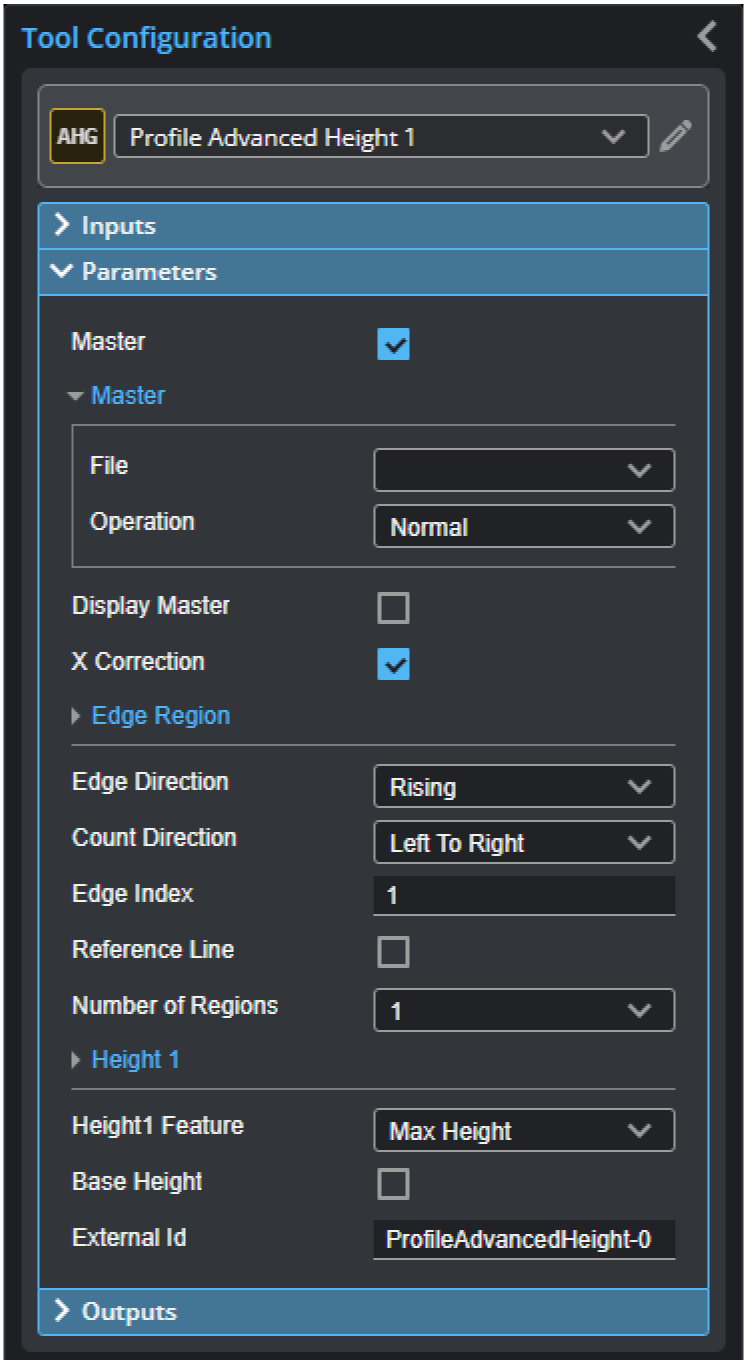
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

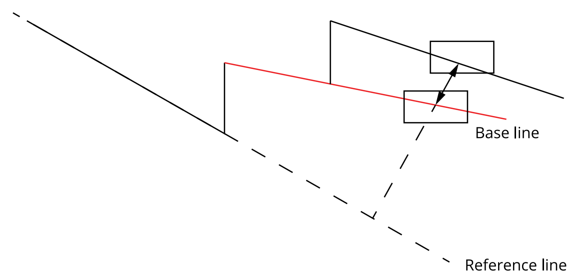
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Master	Toggles a set of settings related to master comparison. For more information, see *Master Comparison* on the next page.
Reference Line	Toggles a set of settings related to the reference line. For more information, see *Reference Line* on page 288.
Number of Regions	Sets the number of height region measurements the tool returns. For each height

Parameter	Description
	region, the tool displays a **Height {n}** section containing the region's position and size. The tool also displays a **Height{n} Feature** drop-down that lets you select the type of feature for that height region.
Base Height	Use base height to "set" the Z axis: when enabled, height values are offset from the base. This is useful if you need to measure between two features, rather than between a feature and the reference line.  When enabled, the tool displays settings related to the base height: size and position of the base height's region (**Base Height** section) and the base height's feature.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Master Comparison

When you check the **Master** option, the tool displays several additional settings and disables measurement anchoring from other tools.

The **File** and **Operation** parameters are in an expandable **Master** section.

Master Parameters

Parameter	Description
File	The file containing the master (template) profile, created by choosing **Save** from the **Operation** drop-down.
Operation	Contains operations related to the master file. One of the following: • **Normal**: Selected by the tool after you perform another file operation. • **Create**: Saves the *current profile* as the master. Use this when the master does not exist. • **Load**: Loads the master file selected in **File**. • **Save**: Saves the current profile as the master. If the file already exists, the master is overwritten with the new data. • **Delete**: Deletes the master file selected in **File**. • **Refresh**: Refreshes the list of files in **File**.
Display Master	Overlays the master profile, in white, on the current profile.

Parameter	Description
X Correction	Enables settings related to X correction (left or right movement) of the profile compared to the master profile. Only displayed if **Master** is enabled. For more information, see *X Correction* below.

X Correction

When you check the **Master** option and enable **X Correction**, the tool displays several additional settings.

X Correction Parameters

Parameter	Description
Edge Region	Lets you set the edge region. You can also edit this region in the data viewer.
Edge Direction	Determines the direction of the edge. One of the following: **Falling** or **Rising**.
Count Direction	Indicates how edges are counted. One of the following: **Left to Right** or **Right to Left**.
Edge Index	Indicates which edge the tool uses.

Reference Line

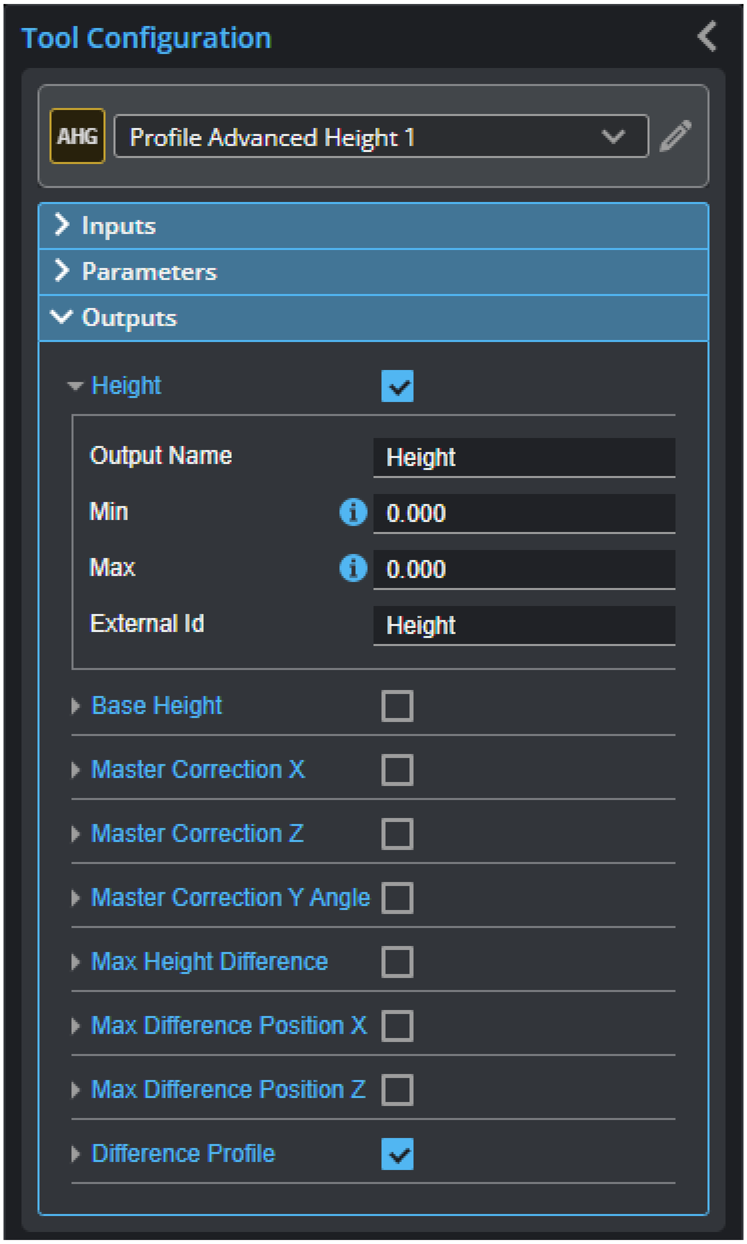
When you enable **Reference Line**, the tool displays several additional settings. The reference line is used to set the measurement direction (perpendicular to the reference line).

Reference Line Parameters

Parameter	Description
Line Region	The number of line regions the tool uses.
Line {n}	Contains settings that let you edit the size and position of the line's region. You can also edit this region in the data viewer.
Fitting Method	Indicates the fitting method the tool uses. One of the following: **Simple** or **Robust**.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Height {n} The height measured in height region {n}. Height is measured perpendicular Will be Invalid if the appropriate number of height regions has not been set in **Height Region**.

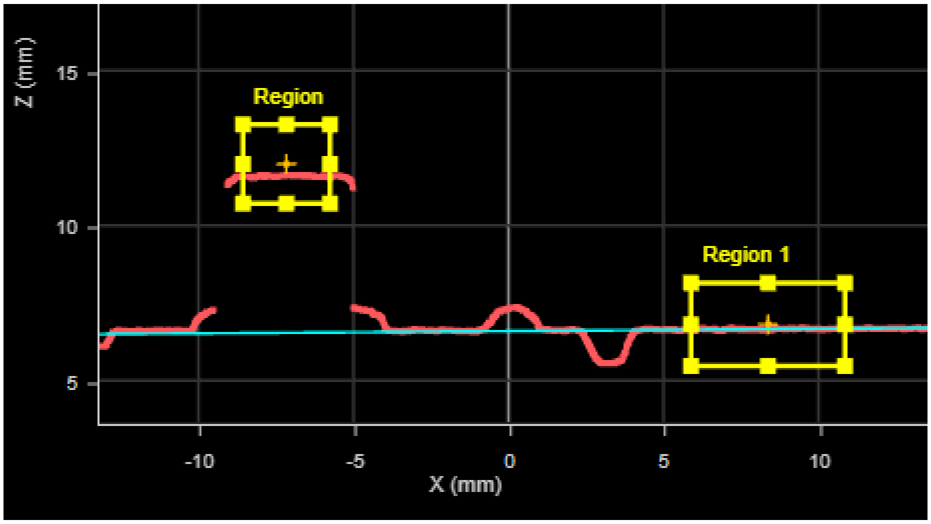
Measurement
Master Correction X **Master Correction Z** **Master Correction Y Angle** The amount of correction applied to the profile with respect to the master.
Max Height Difference The maximum height difference.
Max Difference Position X **Max Difference Position Z** The X and Z positions of the maximum height difference.

Data

Type	Description
Difference Profile	A profile representing the difference between the master and the current frame's profile, available for use as input in the **Stream** drop-down in other tools.

Profile Area

The Area tool determines the cross-sectional area within a region.

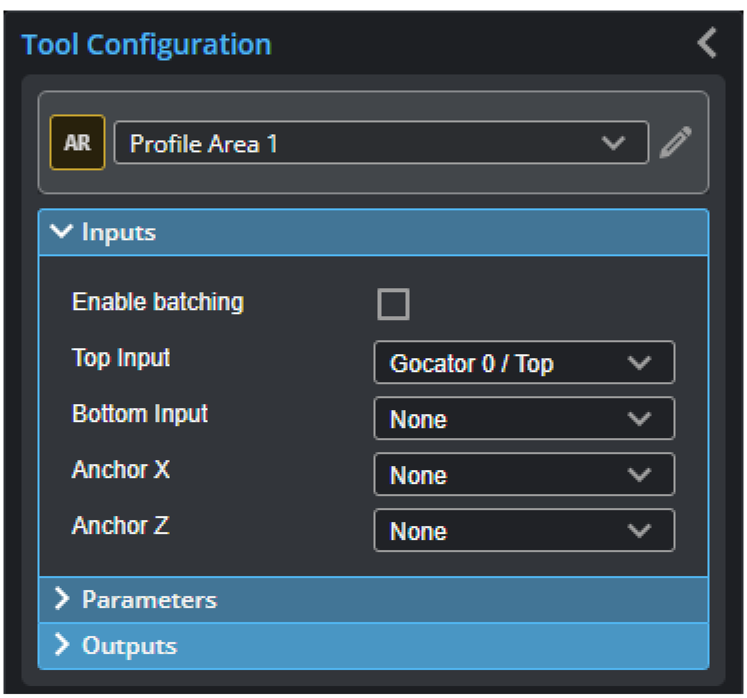


Profile Area tool with baseline set to the best-fitted line in Region 1

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool

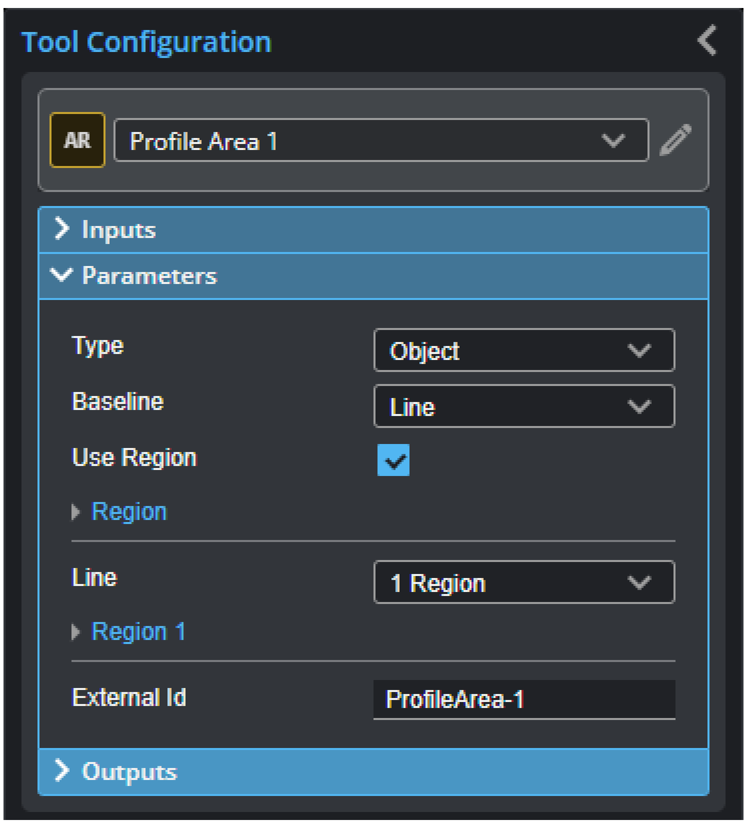
	providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Top Input	The data the tool applies measurements to or processes.
Bottom Input	This tool can optionally take a second, bottom input.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



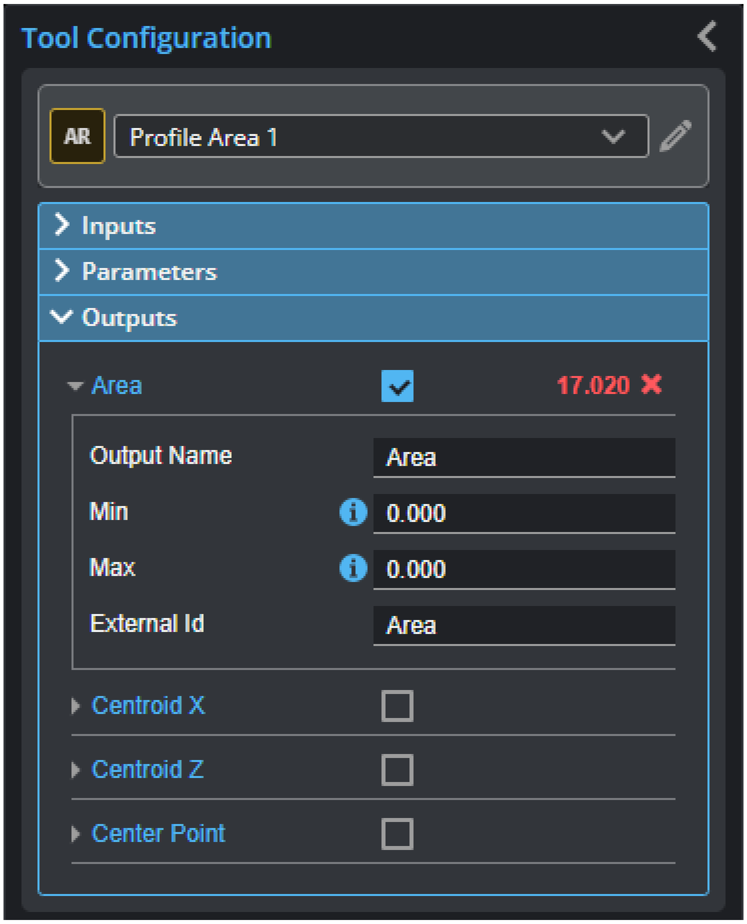
Parameters

Parameter	Description
Type	**Object** area type is for convex shapes above the baseline. Regions below the

Parameter	Description
	baseline are ignored. **Clearance** area type is for concave shapes below the baseline. Regions above the baseline are ignored.
Baseline	Baseline is the fit line that represents the line above which (Object clearance type) or below which (Clearance area type) the cross-sectional area is measured. When this parameter is set to **Line**, you must define a line in the Line parameter. See *Fit Lines* on page 263 for more information on fit lines. When this parameter is set to **X-Axis**, the baseline is set to z = 0.
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Line	When **Baseline** (see above) is set to **Line**, set this to one of the following: **1 Region** or **2 Regions**: Lets you set one or two regions. The tool uses the data in those regions to fit a line. **All Data**: The tool uses all of the data in the scan data. For more information on regions, see *Regions* on page 250). For more information on fit lines, see *Fit Lines* on page 263.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

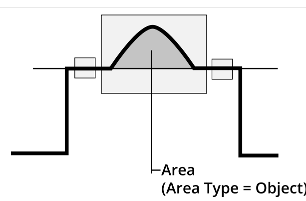
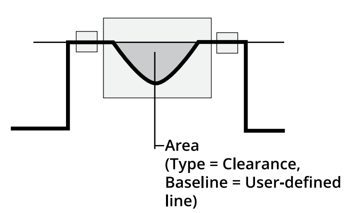
Most tools provide measurements, geometric features, or data as outputs.

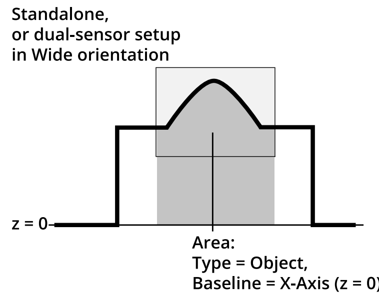
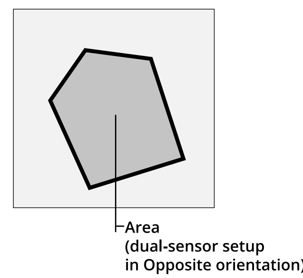
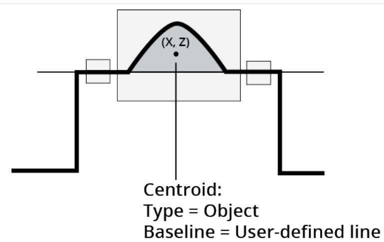


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Area Measures the cross-sectional area within a region that is above or below a fitted baseline.	 

Measurement	Illustration
	 
Centroid X Determines the X position of the centroid of the area.	
Centroid Z Determines the Z position of the centroid of the area.	

Features

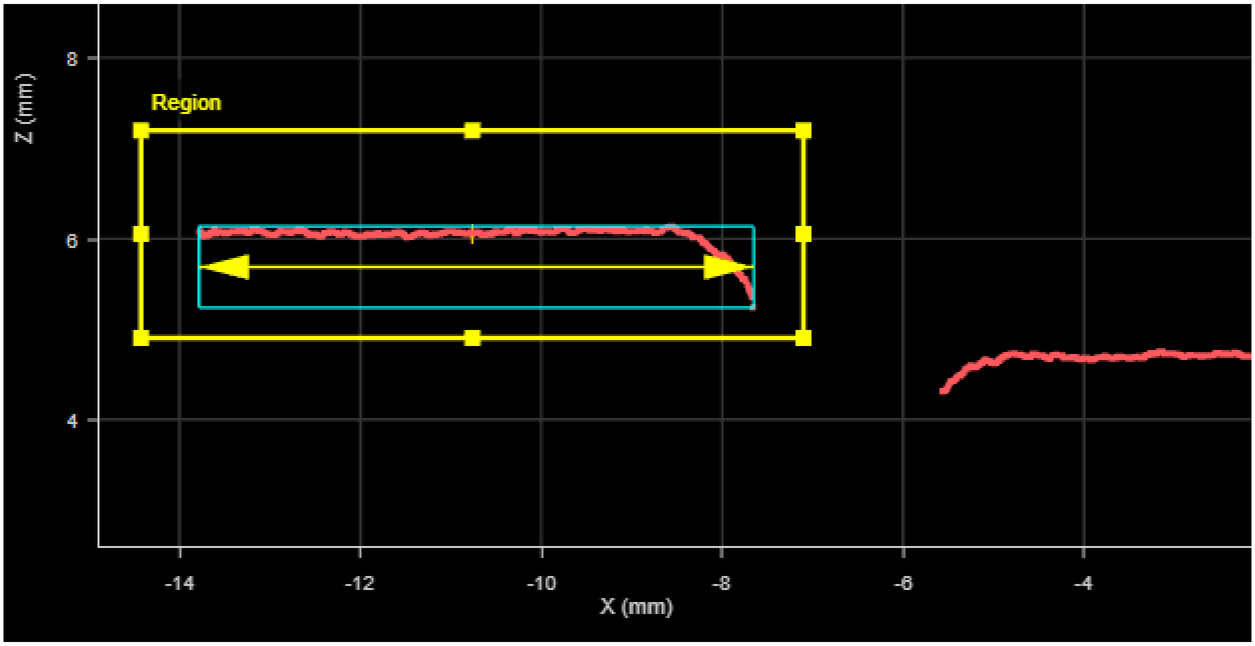
Type	Description
Center Point	The center point of the area.

For more information on geometric features, see *Geometric Features* on page 262.

Profile Bounding Box

The Bounding Box tool provides measurements related to the smallest box that contains the profile (for example, X position, Z position, width, etc.).

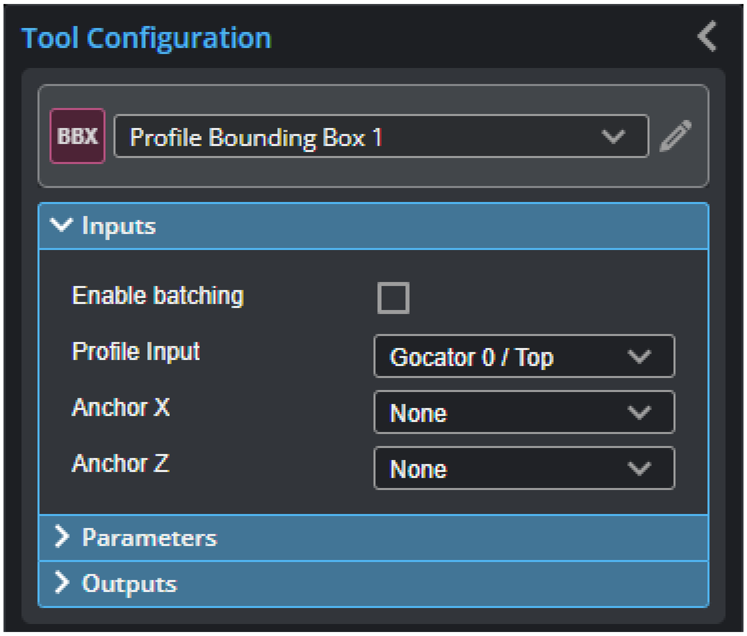
The bounding box provides the absolute position from which the Position centroids tools are referenced.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



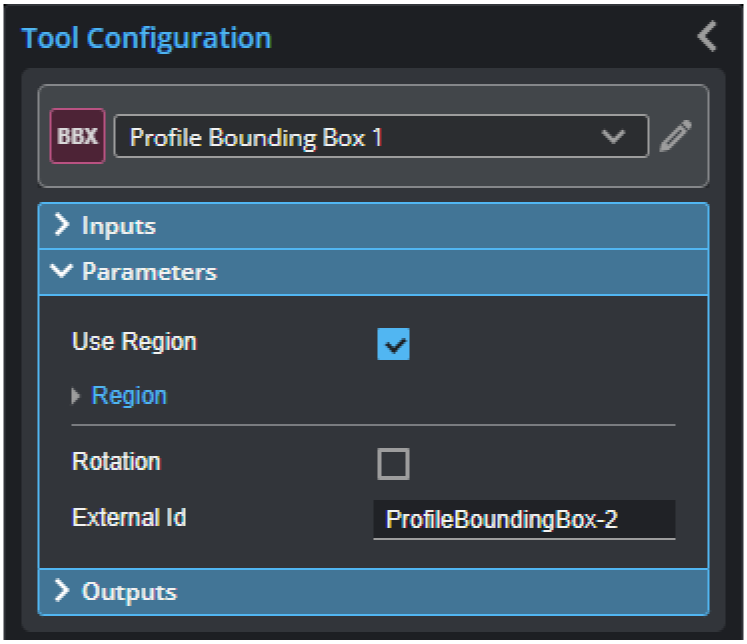
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

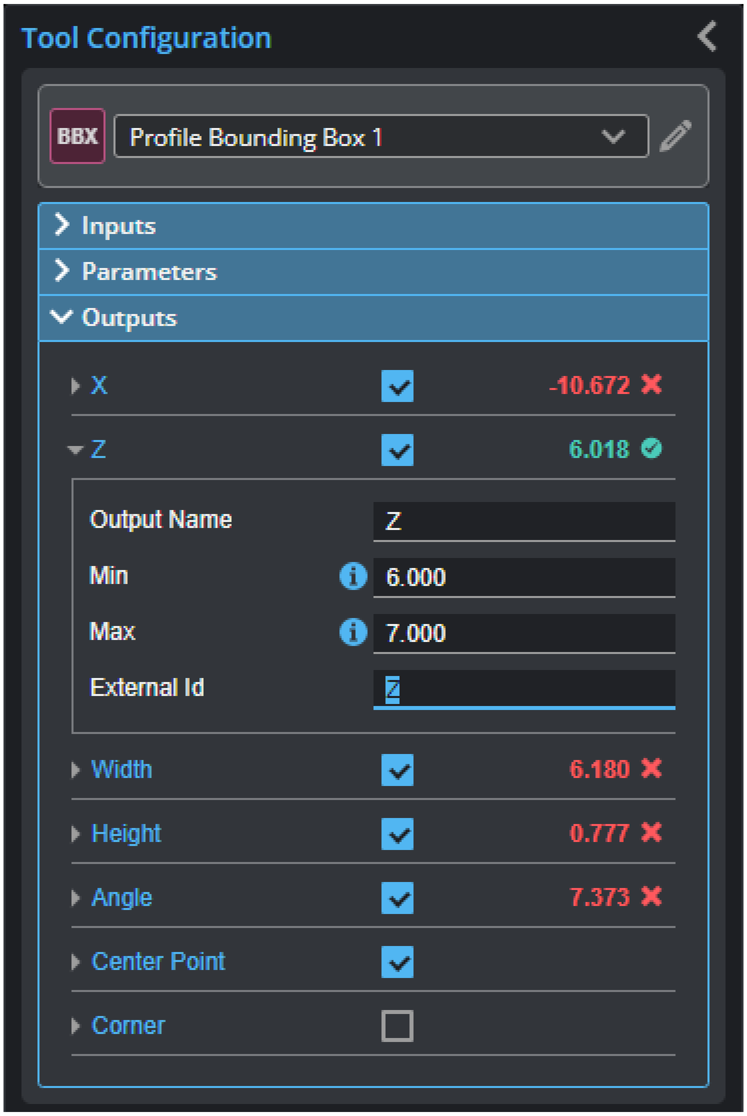


Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Rotation	A bounding box can be vertical or rotated. A vertical bounding box provides the absolute position from which the part's Position centroid measurements are referenced. When enabled, allows rotation of the bounding box. Only displayed if **Use Region** is enabled.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

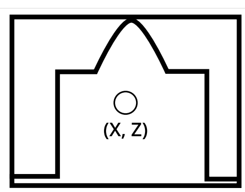
Most tools provide measurements, geometric features, or data as outputs.

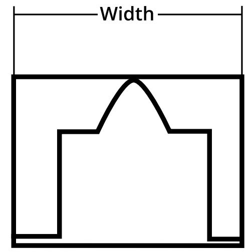
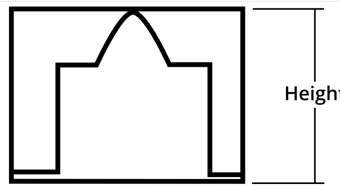


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Determines the X position of the center of the bounding box that contains the profile. The value returned is relative to the profile.	
Z Determines the Z position of the center of the bounding box that contains the profile.	

Measurement	Illustration
The value returned is relative to the profile.	
Width Determines the width of the bounding box that contains the profile. The width reports the dimension of the box in the direction of the minor axis.	
Height Determines the height (thickness) of the bounding box that contains the profile.	
Angle Determines the angle of the longer side of the bounding box around the Y axis, relative to the X axis. This measurement is only available if the **Rotation** parameter is enabled.	

Features

Type	Description
Center Point	The center point of the bounding box.
Corner Point	The lower left corner of the bounding box.

For more information on geometric features, see *Geometric Features* on page 262.

Profile Bridge Value

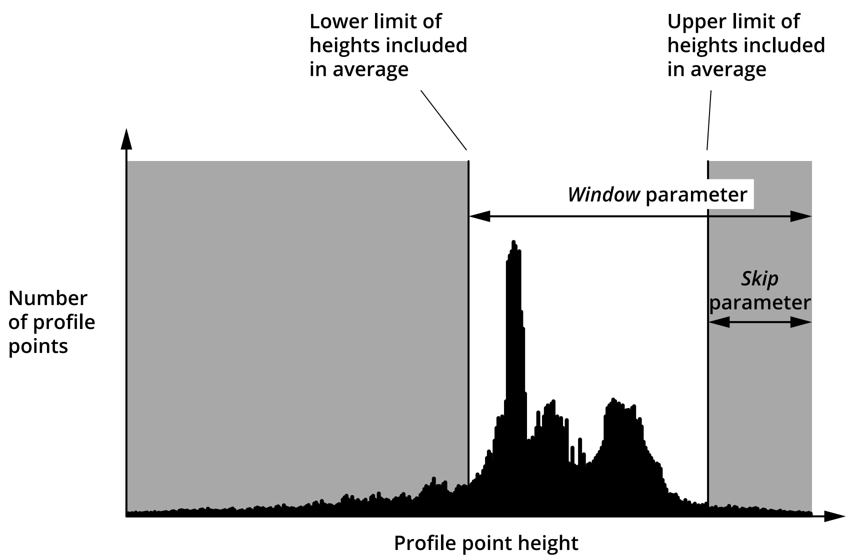
The Bridge Value tool calculates the "bridge value" and angle of a scanned surface. A bridge value is a single, processed range that is an average of a laser line profile that has been filtered to exclude user-definable portions of highs and lows in the profile. The resulting value represents a "roughness calculation." A bridge value is typically used to measure road roughness, but can be used to measure the roughness of any target.

The tool provides two additional measurements (Window and StdDev) that can help determine whether the scanned data is valid; for more information, see *Measurements* on page 305.

> The Bridge value tool is *only* available when **Uniform Spacing** (in the **Scan Mode** panel on the **Scan** page) is unchecked, as the tool only works with unresampled data. For more information, see *Uniform Data and Point Cloud Data* on page 70.

Understanding the Window and Skip Settings

The Bridge Value tool measurements work on a histogram of the ranges that make up the profile. The **Window** and **Skip** parameters together determine what segment of the heights in the histogram is used to calculate the bridge value. The following diagram illustrates the portion of the points of a histogram that would be included for calculating the bridge value, where **Window** is roughly 85% *of the total points of the histogram*, and **Skip** is roughly 15% of the points.



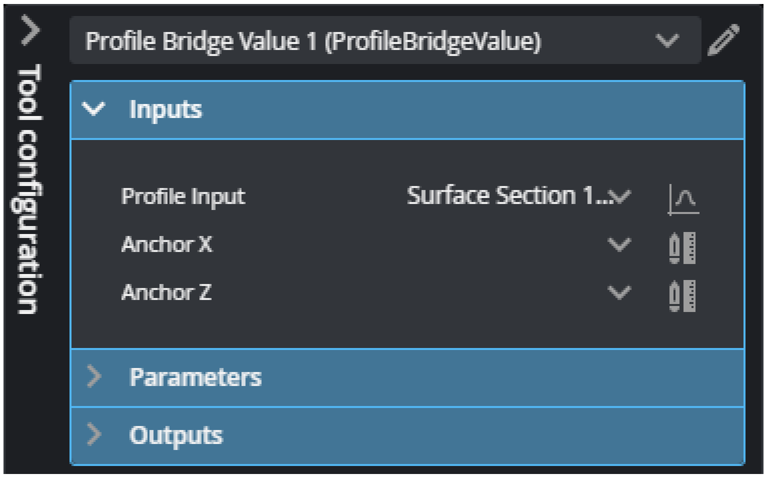
Profile point heights in the white area are included in the calculation of the average. Profile point heights in the grey area are excluded. By adjusting the **Window** and **Skip** parameters, you can exclude profile point heights that correspond to unwanted features on the target. In road roughness applications, for example, you could exclude rocks (profile points higher than the road surface), cracks or tining valleys (profile points lower than the road surface), and so on, to get an accurate representation of the tire-to-road interface.

For more information on parameters, see the *Parameters* table below.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



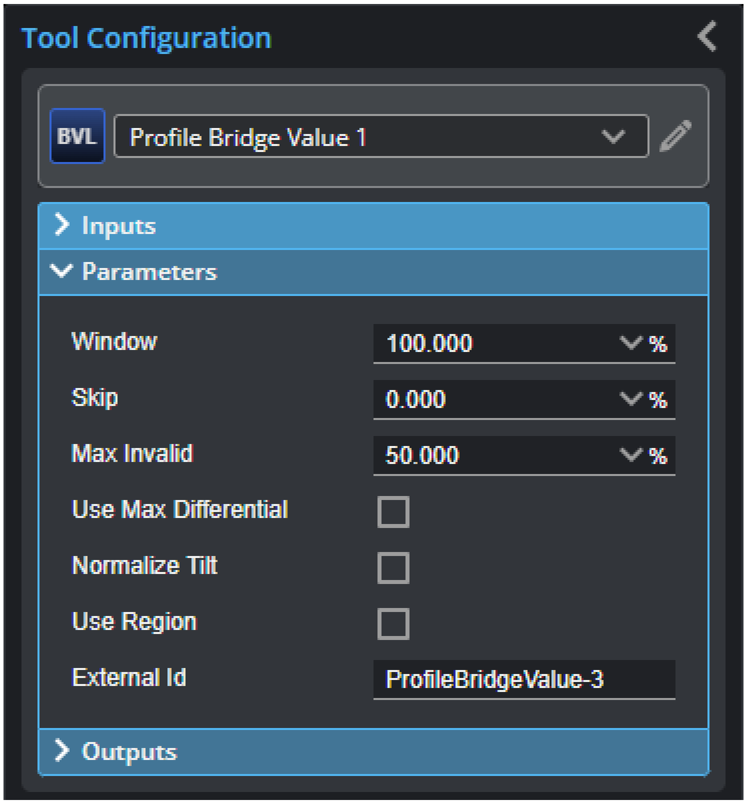
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



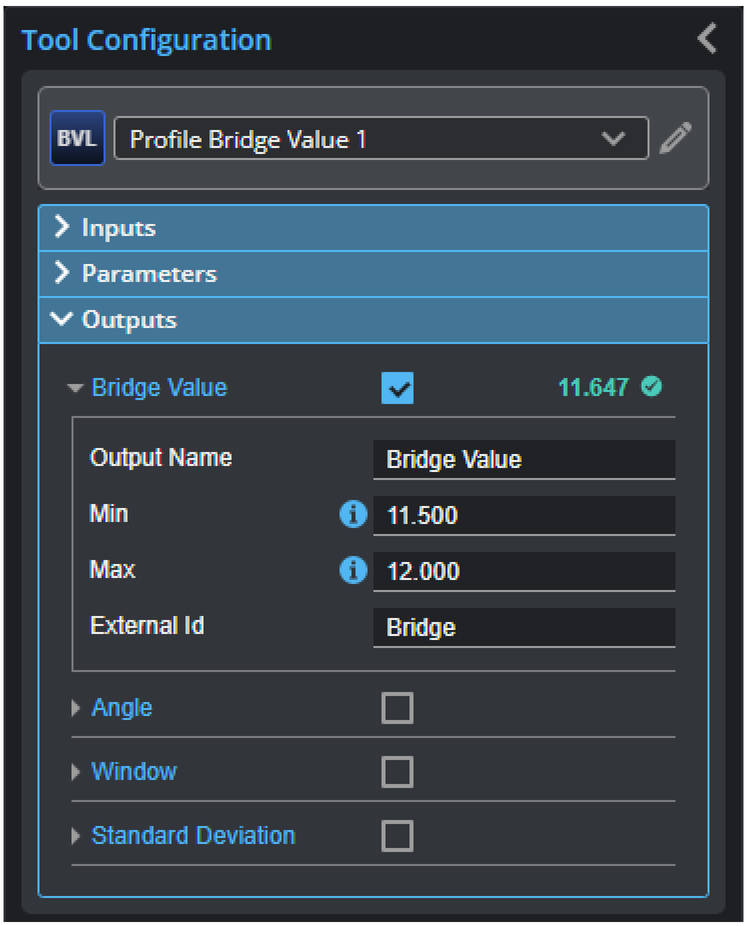
Parameters

Parameter	Description
Window	A percentage of the profile point heights in the histogram, starting from the highest point, to include in the average. For example, a setting of 50% would include the highest 50% of the heights. The **Skip** parameter then determines the actual portion of the profile point heights used to calculate the average. The **Window** setting in effect sets the lower limit of the profile point heights in the histogram to be used in the average. Use the setting to exclude lower parts of a profile that you do not want to include in the measurement.
Skip	A percentage of the profile point heights in the histogram, starting from the highest points, to *exclude* from the average. The **Skip** setting basically sets the upper limit of the profile point heights in the histogram to be used in the average. Use the setting to exclude higher parts of a profile that you do not want to include in the measurement. If **Skip** is greater than **Window**, an invalid value is returned.
Max Invalid	The maximum percentage of invalid points allowed before an invalid result is returned.
Use Max Differential	Enables the **Max Differential** parameter. See below.
Max Differential	The maximum difference between the maximum and minimum histogram values

Parameter	Description
	before an invalid measurement value is produced.
Normalize Tilt	Fits a line to the profile and shears the points in the Z direction by the angle between the fitted line and the X axis. The **Window** and **Skip** settings are applied to the histogram of the transformed data. Useful for surfaces that are tilted.
Use Region	Enables the region settings.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.

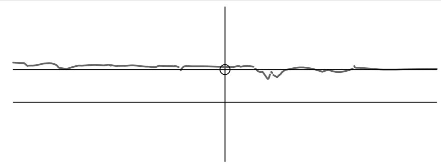
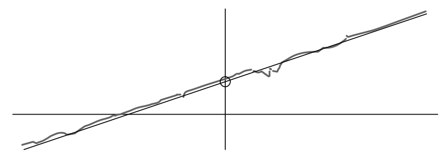
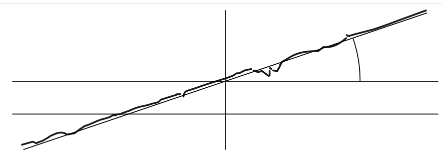
Outputs

Most tools provide measurements, geometric features, or data as outputs.



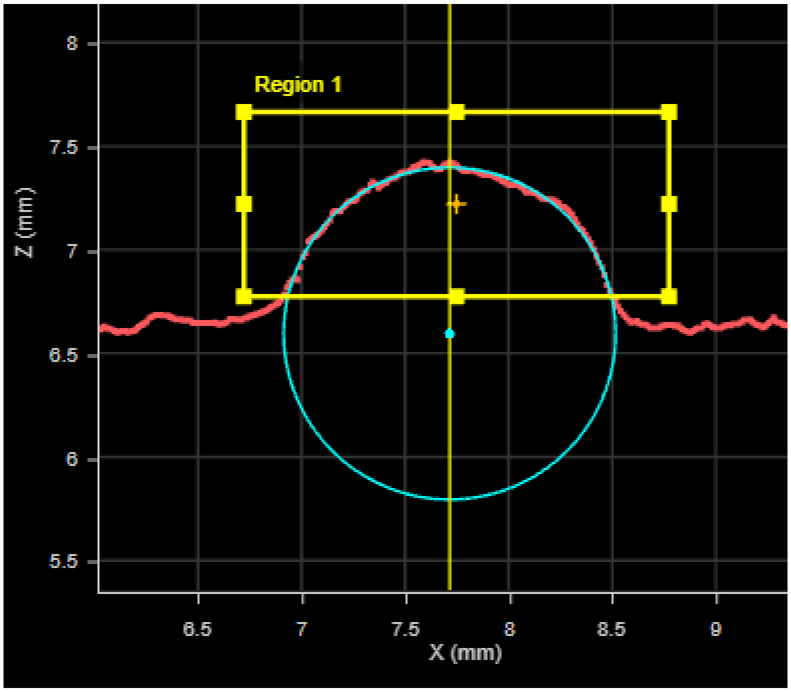
Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurement	Illustration
Bridge Value Determines the bridge value of the non-excluded part of the profile.	 
Angle Determines the angle of the line fitted to the profile. When **Normalize Tilt** is unchecked, the measurement always returns 0.	
Window Returns the height of the area on the profile resulting from the **Window** and **Skip** settings. If you are using this measurement on a Gocator 2342 sensor, see *Profile Bridge Value* on page 301 for more information.	
StdDev Returns the standard deviation of the data in the area on the profile resulting from the **Window** and **Skip** settings. If you are using this measurement on a Gocator 2342 sensor, see *Profile Bridge Value* on page 301 for more information.	

Profile Circle

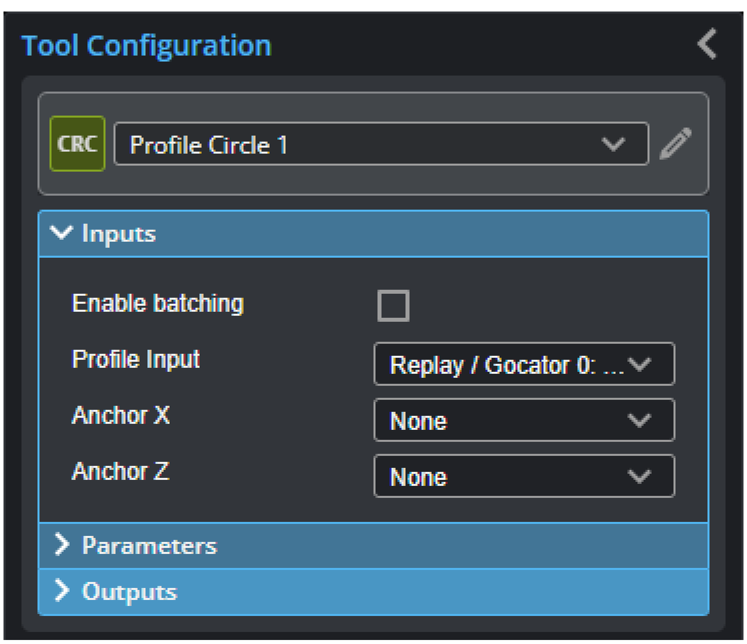
The Circle tool calculates the best-fit circle for a profile and returns measurements of various characteristics of the circle. The tool uses the iterative robust least squares fitting method.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



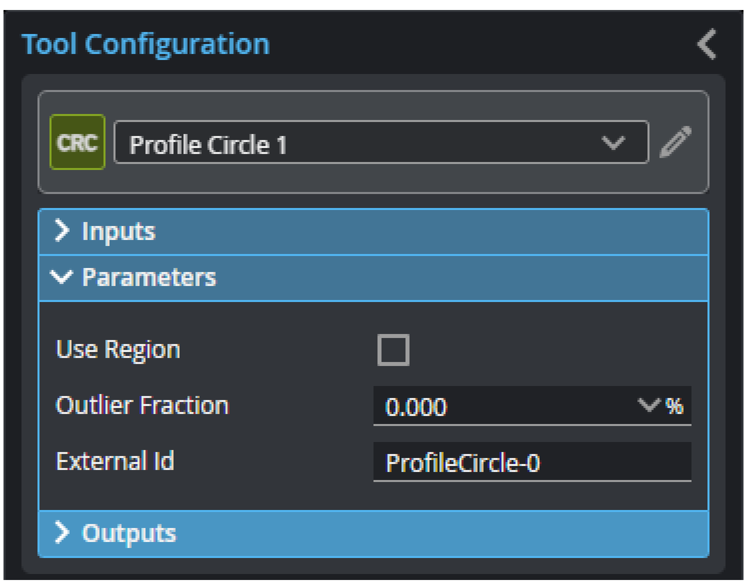
🗋	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



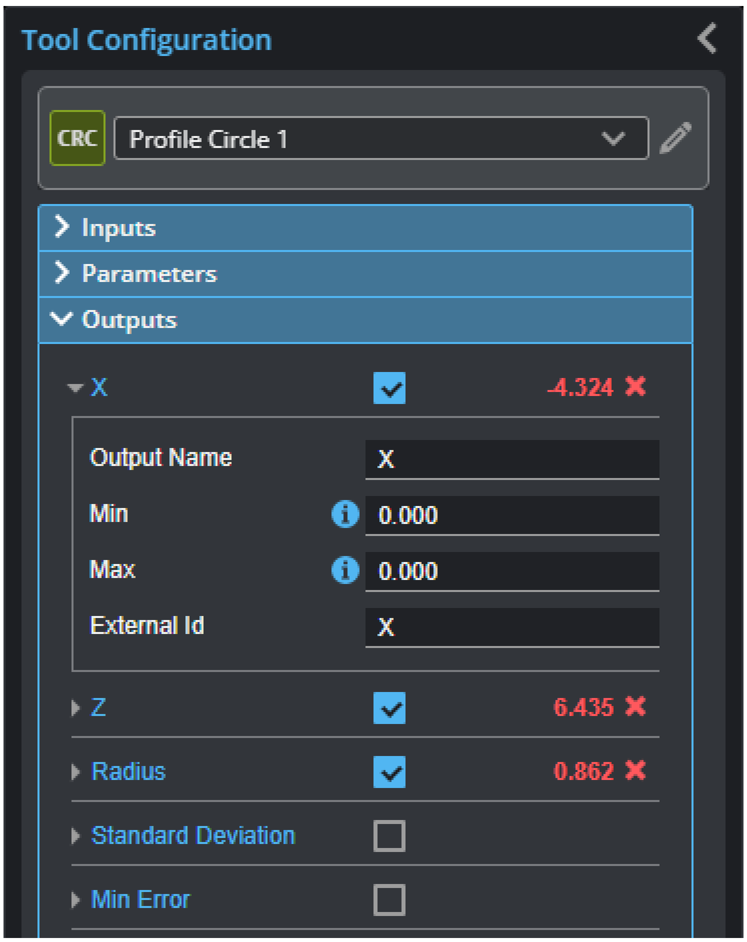
Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.

Parameter	Description
Region {n}	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Outlier Fraction	The percentage of the profile points that are considered noise. A range from 0% to 80%. For example, suppose a profile contains `m` points. If **Outlier Fraction** is not equal to 0, then `Outlier Fraction*m` points will be considered noises and be removed iteratively.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

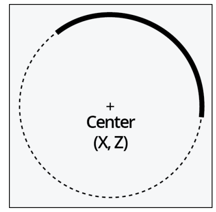
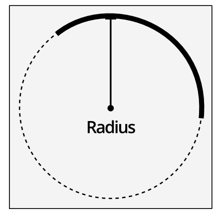
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Finds the circle center position in the X axis.	
Z Finds the circle center position in the Z axis.	
Radius Measures the radius of the circle.	
Standard Deviation Returns the standard deviation of the data points with respect to the fitted circle.	
Min Error **Max Error** The minimum and maximum error among the data points with respect to the fitted circle.	
Min Error X **Min Error Z** The X and Z position of the minimum error.	
Max Error X **Max Error Z** The X and Z position of the maximum error.	

Features

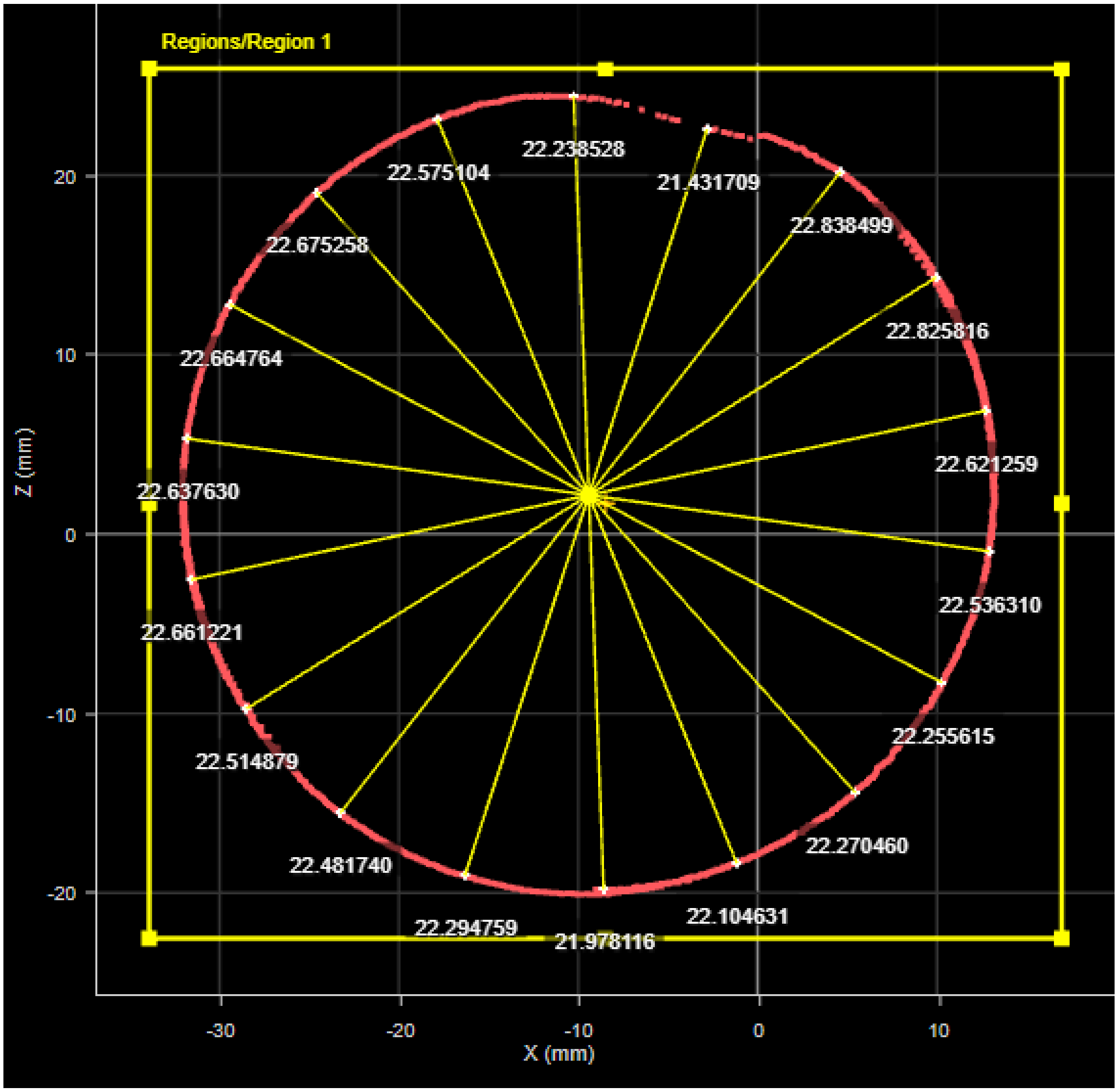
Type	Description
Point	The center point of the fitted circle.
Circle	The fitted circle.

For more information on geometric features, see *Geometric Features* on page 262.

Profile Circle Radii

The Profile Circle Radii tool lets you measure radii and diameters at specified angle steps, given a specified center point. The tool draws rays from the center point and returns radii or diameter measurements for each ray. The center point comes from either the center of a bounding box or a point geometric feature from another tool. The tool also provides settings to compensate for missing data and for rough surfaces or noise.

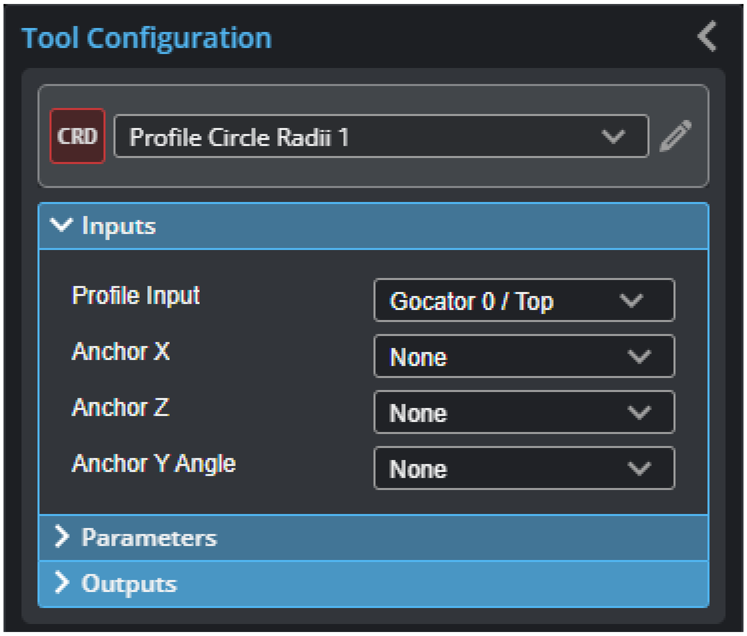
For example, in the following scan of an exhaust pipe by a four-sensor system, the tool is showing a shorter radius measurement at roughly 70 degrees that indicates a dent in the pipe.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



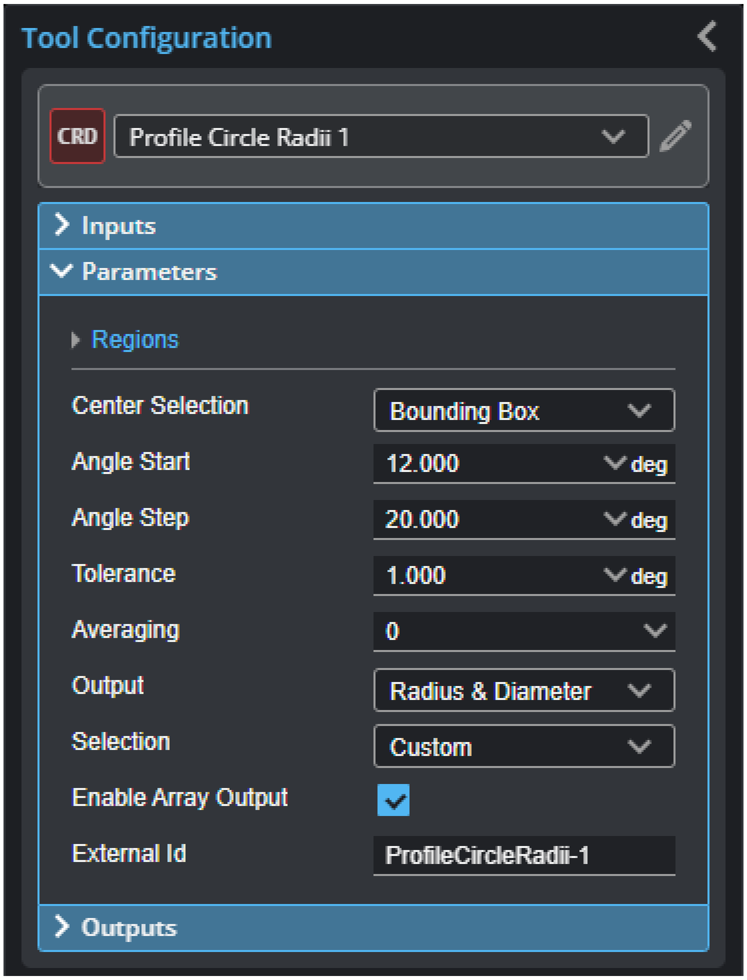
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Y Angle	The Y angle measurement of another tool that this tool uses as a rotational anchor. Rotational anchors are optional.
Center	The point geometric feature output by another tool that the Circle Radii tool uses as the center point from which rays are drawn to search for data points. The parameter is only available when **Center Selection** is set to **Feature Input**.

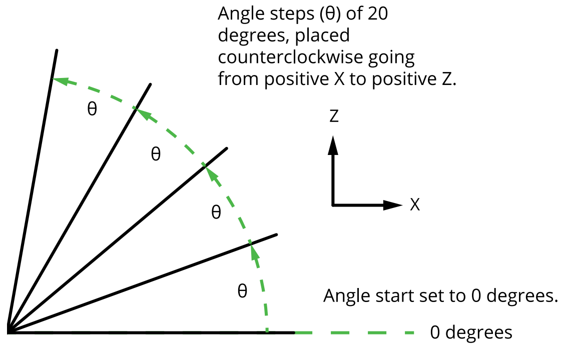
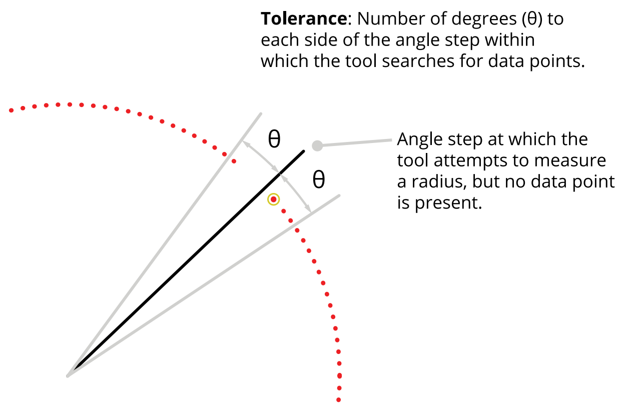
Parameters

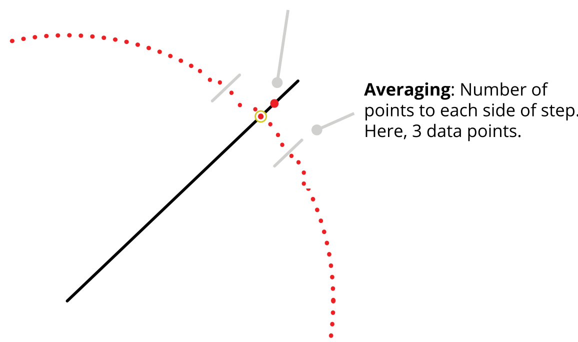
You configure the tool's parameters in the expandable **Parameters** section.



Parameters

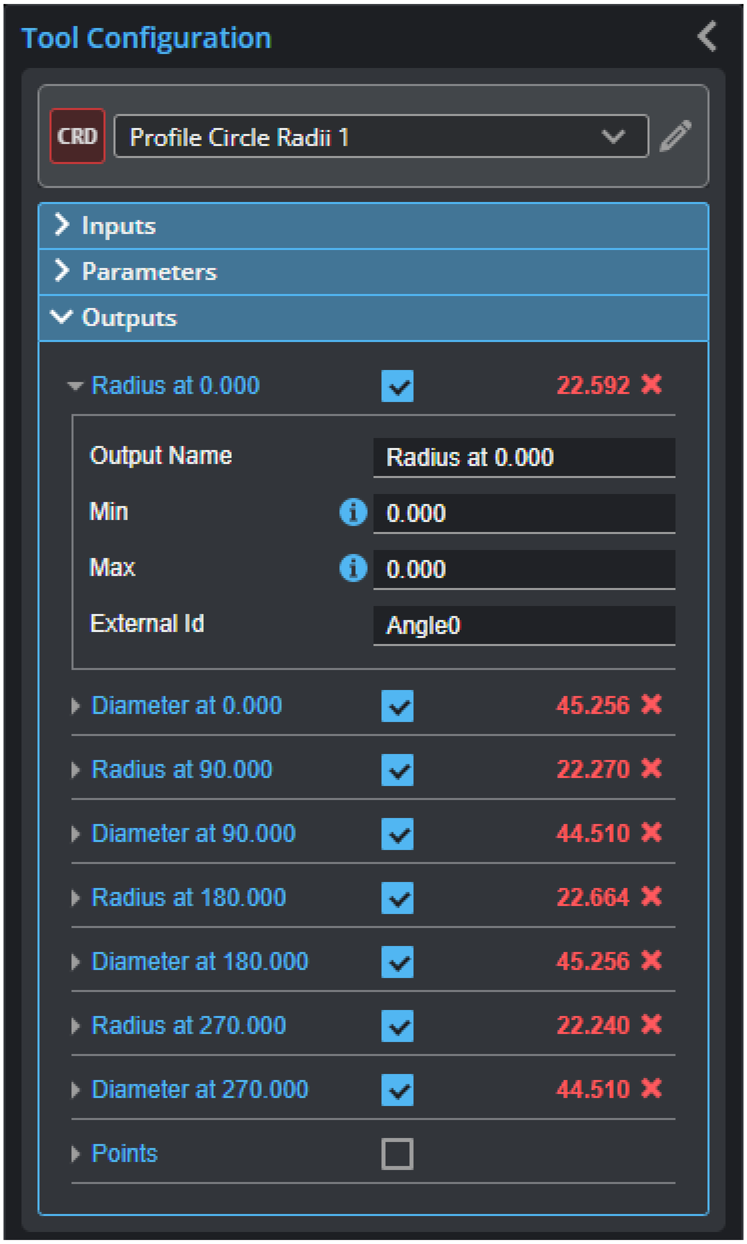
Parameter	Description
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Center Selection	The source for the point geometric feature the tool uses as a center point. One of the following: **Bounding Box** – Uses the center of the bounding box that encloses the scan data. If regions are enabled (**Enable** in the **Regions** expander is checked), the tool places a bounding box only around the data in the region. If **Enable** is unchecked, the tool places a bounding box around all scan data; this will include any outliers in the bounding box, which could produce an undesired center point. **Feature Input** – A point geometric feature provided by another tool, such as the

Parameter	Description
	center point from a Circle tool.
Angle Start Angle Step	**Angle Start**: The angle at which ray steps start. **Angle Step**: The angle step in degrees. The following shows how these settings work together:  The tool searches for a data point at each angle step and returns the radius from the center point or the diameter.
Tolerance	If no data point is found at the angle step, the tool searches within the specified number of degrees to each side of the step to find a data point. Useful to compensate for gaps in the data.  The graphic above shows how the tool searches to each side of the angle step until it finds a data point (circled and in yellow).
Averaging	The number of data points to each side of the point the tool uses to average. Use this to compensate for noise or rough surfaces.

Parameter	Description
	 The graphic above shows how the tool averages the data point at the angle step with the number of data points specified in **Averaging** to each side of the angle step, replacing the original data point with the average (circled and in yellow).
Output	Selects whether to output the radius, diameter, or both for each step.
Selection	Lets you quickly enable or disable all measurements.
Enable Array Output	When this parameter is checked, the tool outputs arrays containing the radius and diameter measurements. If the parameter is disabled, measurements are output for the radius or diameter at each step.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.

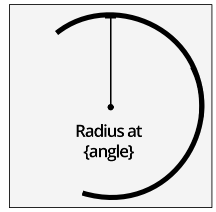
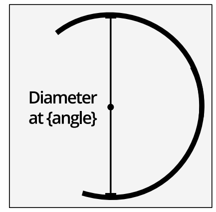


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

	If you check **Enable Array Output** in the tool's parameters, the tool outputs radius and diameter measurements in arrays, called Radius and Diameter, respectively. Otherwise, individual measurements are output, such as "Radius at 90.000," "Radius at 180," and so on.

Measurements

Measurement	Illustration
Radius at {angle} Returns the radius at {angle}.	
Diameter at {angle} Returns the diameter at {angle}.	

Data

Type	Description
Points	An array of the points at the end of the rays.

Profile Closed Area

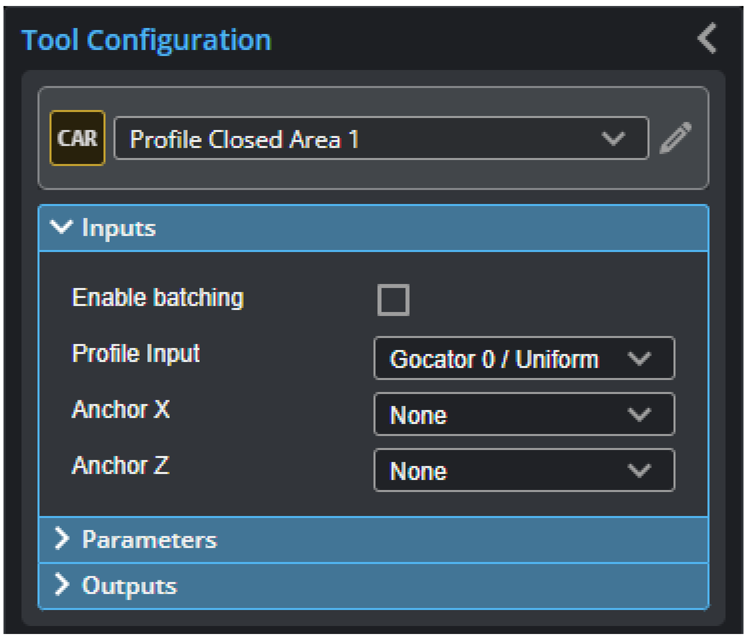
The Closed Area tool determines the cross-sectional area within a region using point cloud data (that is, with **Enable uniform spacing** disabled in the **Acquire** > **Scan** page > **Scan Mode** panel) from a dual- or multi-sensor system.

The tool renders a polygon corresponding to the profile in the data viewer. Use this polygon to decide whether the tool can correctly calculate an acceptable representation of the profile. Minor gaps in the profile are permitted; the size of these gaps is configurable.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

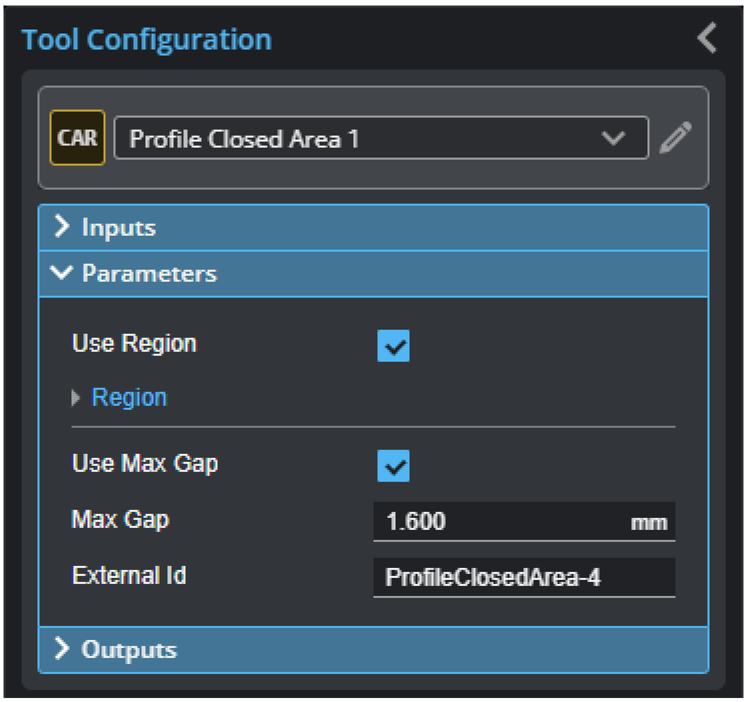
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching,*

Name	Description
	and Aggregation on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

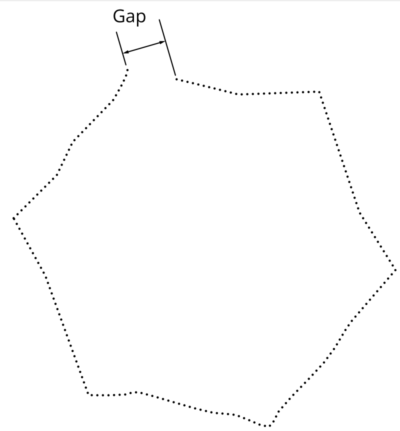
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



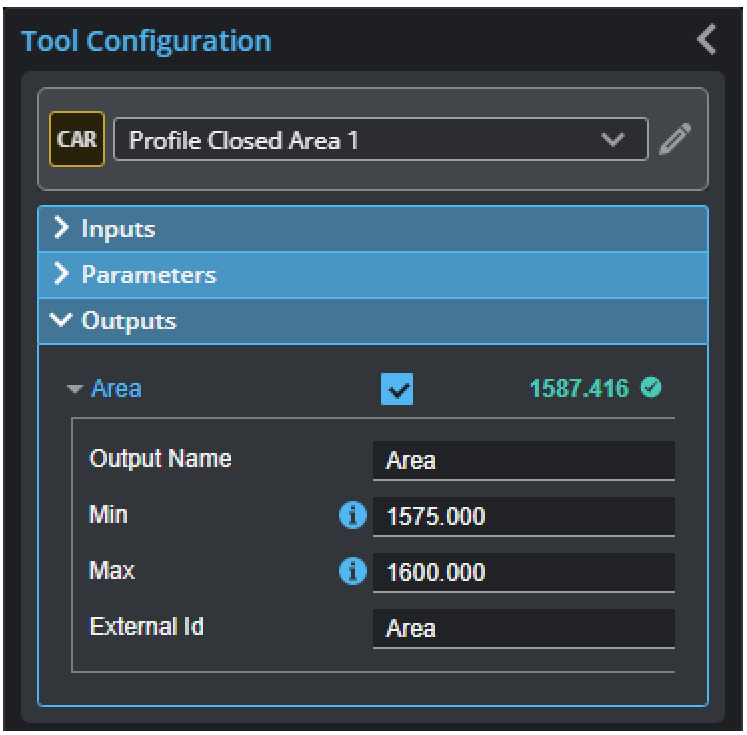
Parameters

Parameter	Description
Use Region	Indicates whether the tool uses a user-defined region. If this option is not checked, the tool uses data from the entire active area.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Use Max Gap	Indicates whether the tool uses the **Max Gap** setting (see below).
Max Gap	The maximum gap allowed between any two profile points on the contour of the target, in millimeters. In the following illustration of a profile, if the gap were greater than the value set in **Max Gap**, the tool would return an invalid value.

Parameter	Description
	
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.

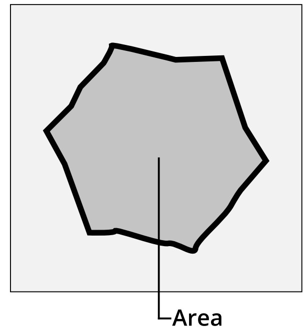


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

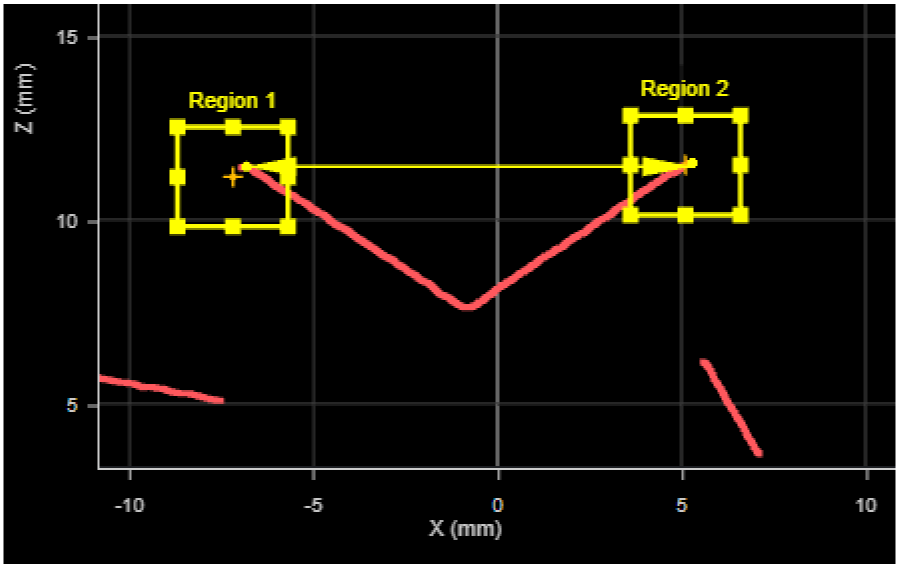
All outputs provide an external ID (available by expanding the output in the Outputs panel) for optional use in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Measurements

Measurement	Illustration
Area Measures the cross-sectional area within a region using data from a dual- or multi-sensor system.	

Profile Dimension

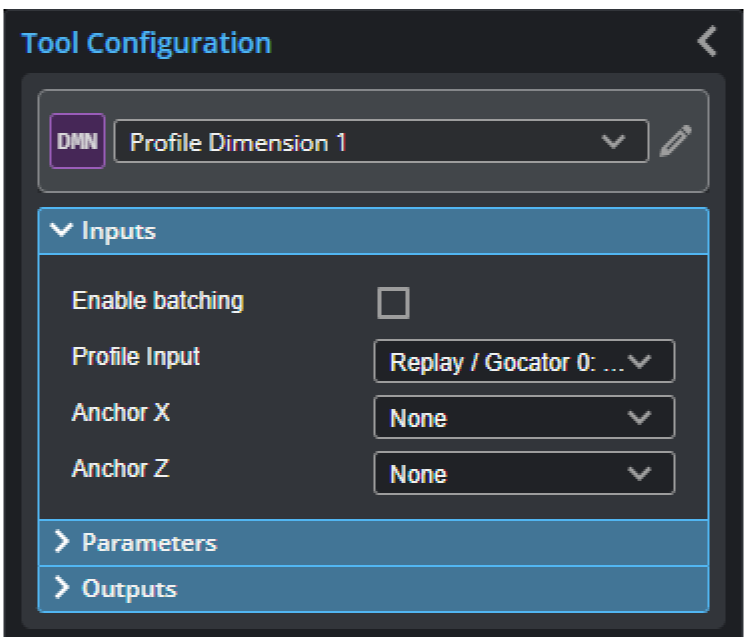
The Dimension tool provides Width, Height, Distance, Center X, and Center Z measurements.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



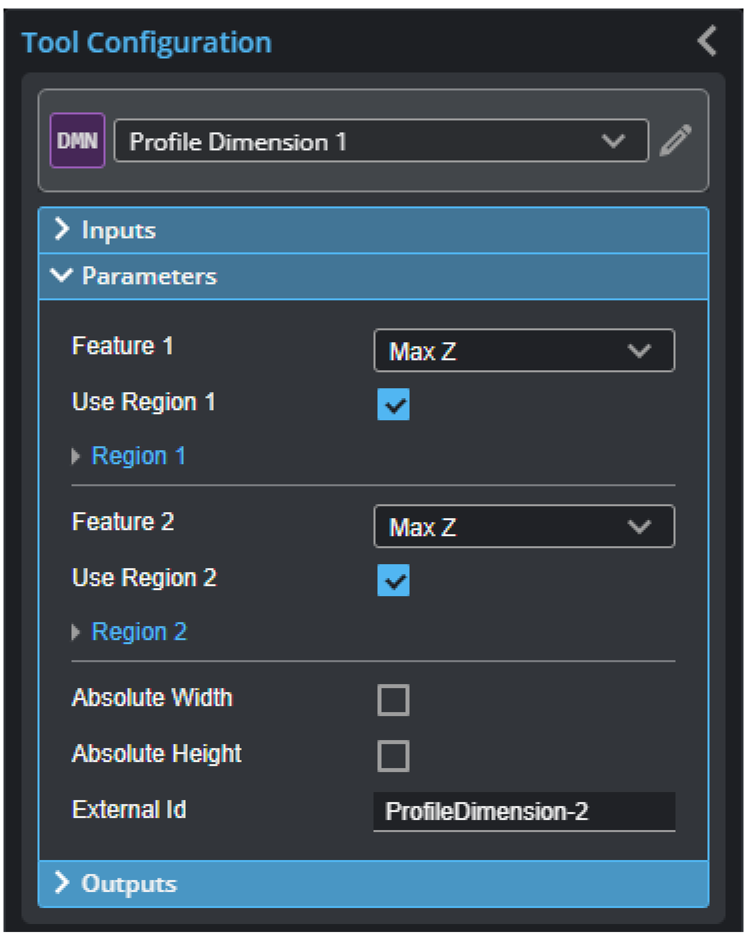
	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. If **Enable Batching** is *disabled* and the passed array contains more than two elements, GoPxL displays an error. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

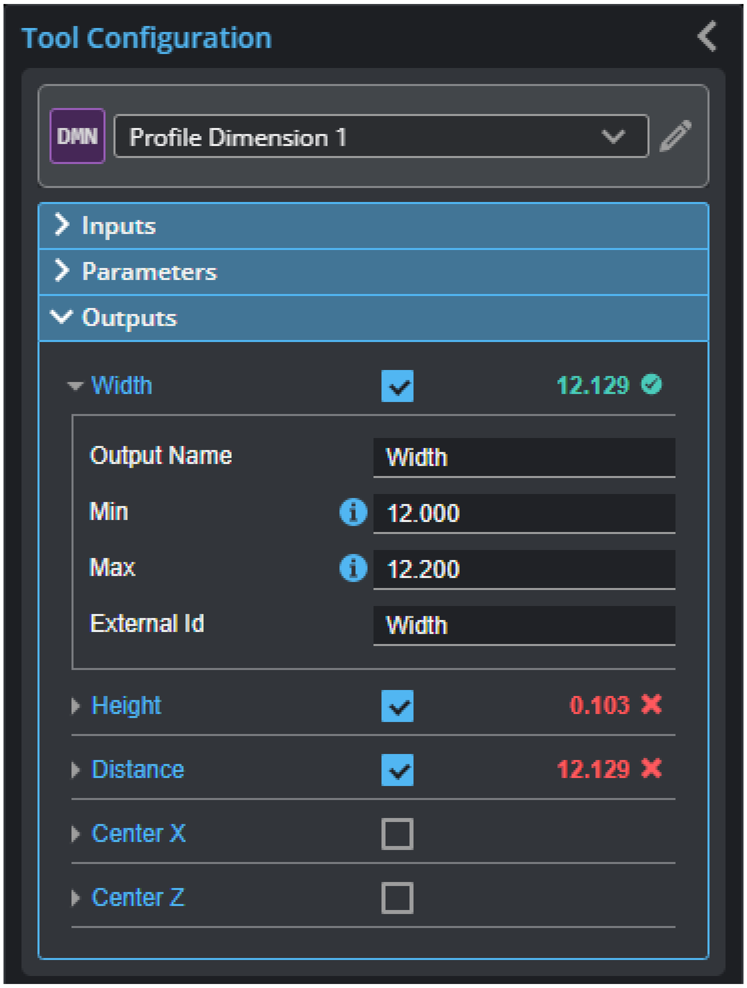


Parameters

Parameter	Description
Feature 1 Feature 2	The **Feature 1** and **Feature 2** settings represent the two features the tool uses to perform measurements. For each, one of the following: • Max Z • Min Z • Max X • Min X • Corner • Average • Rising Edge • Falling Edge • Any Edge • Top Corner • Bottom Corner • Left Corner • Right Corner • Median For more information on feature points, see *Feature Points* on page 259.
Use Region 1 Use Region 2	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region 1 Region 2	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Absolute Width Absolute Height *(Width and Height measurements only)*	Determines if the result will be expressed as an absolute or a signed value.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

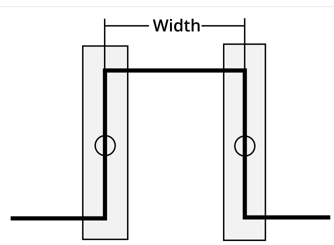
Most tools provide measurements, geometric features, or data as outputs.

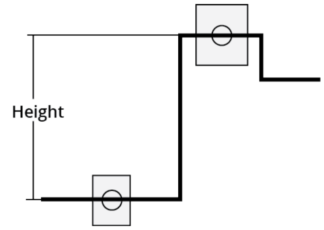
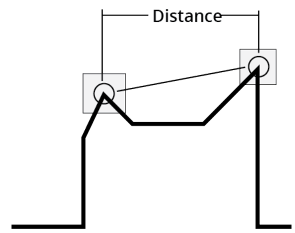
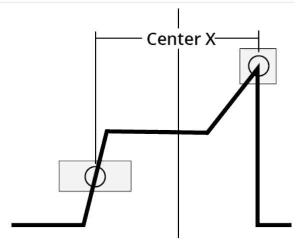
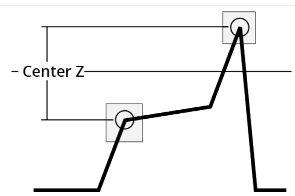


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

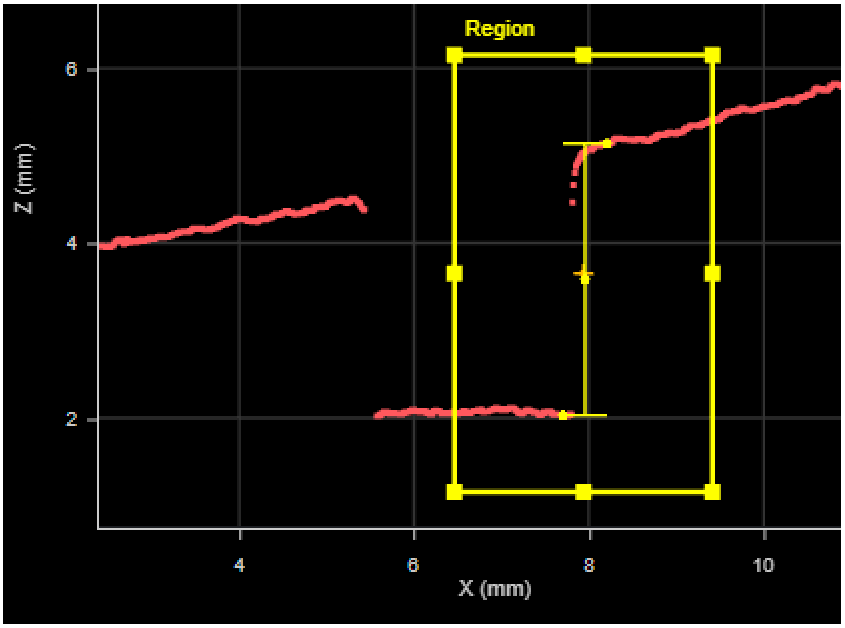
Measurements

Measurement	Illustration
Width Determines the difference along the X axis between two feature points. The difference can be calculated as an absolute or signed result. The difference is calculated by: *Width = Feature* $2_{X\ position}$ – *Feature* $1_{X\ position}$	

Measurement	Illustration
Height Determines the difference along the Z axis between two feature points. The difference can be expressed as an absolute or signed result. The difference is calculated by: *Height = Feature 2$_{Z\ position}$ – Feature 1$_{Z\ position}$*	
Distance Determines the direct, Euclidean distance between two feature points.	
Center X Finds the average location of two features and measures the X axis position of the average location	
Center Z Finds the average location of two features and measures the Z axis position of the average location.	

Profile Edge

The Profile Edge tool finds an edge on a profile, searching from left to right. The tool's settings help fit the edge point when multiple potential edges are in the profile. You can configure the tool to locate a step or a corner (that is, for cases where there is no clear step in the profile but instead a smooth slope), as well as rising or falling steps.



Step height of a step

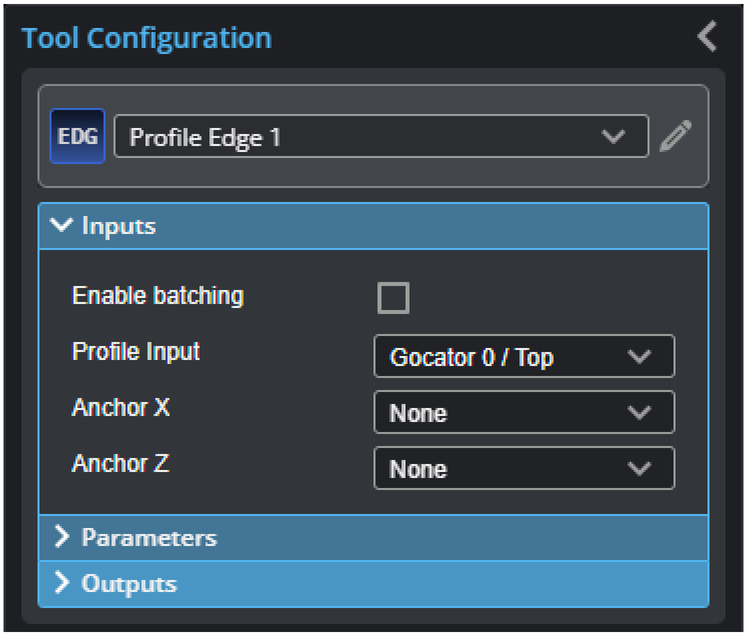
After the tool locates an edge, it returns the position (X and Z) of the edge. For steps, it also returns the step height.

The tool can also generate a point geometric feature corresponding to the center of the step that Feature tools can take as input for measurement. For more information on Feature tools, see *Feature Measurement* on page 692.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



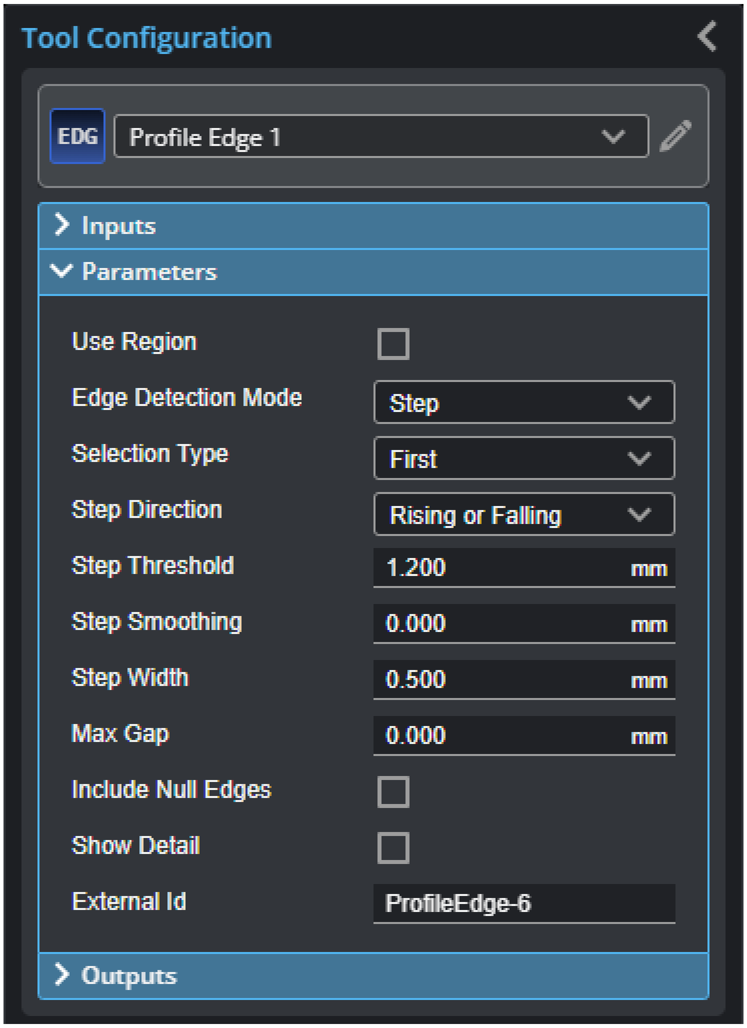
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

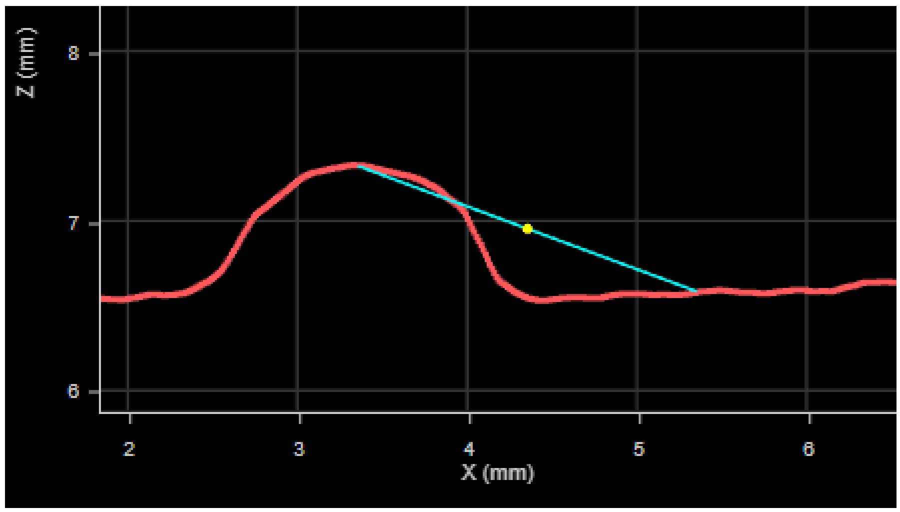
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



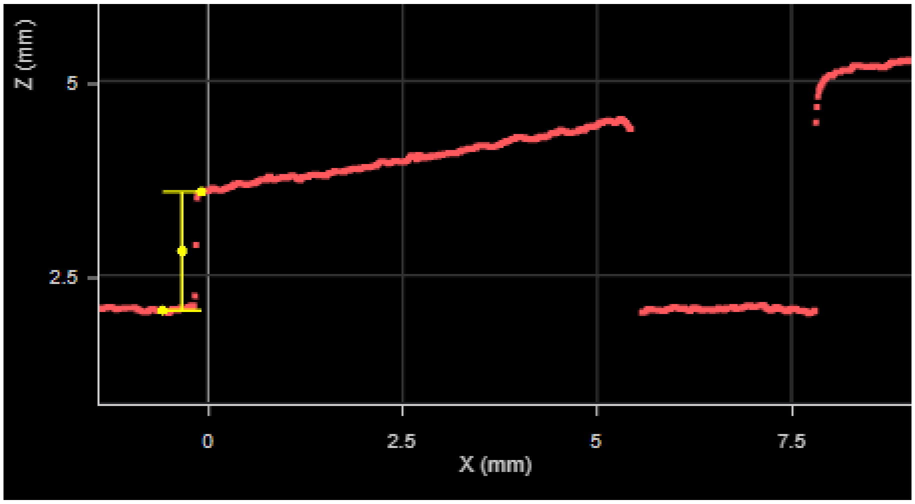
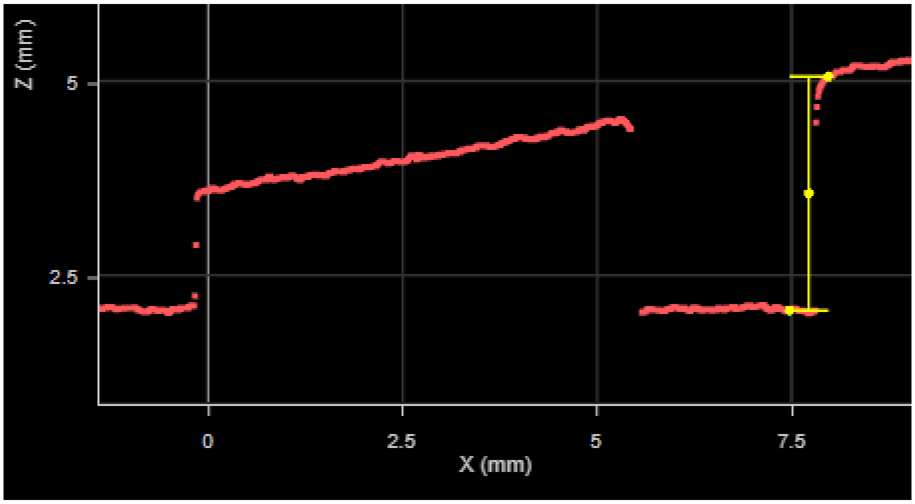
Parameters

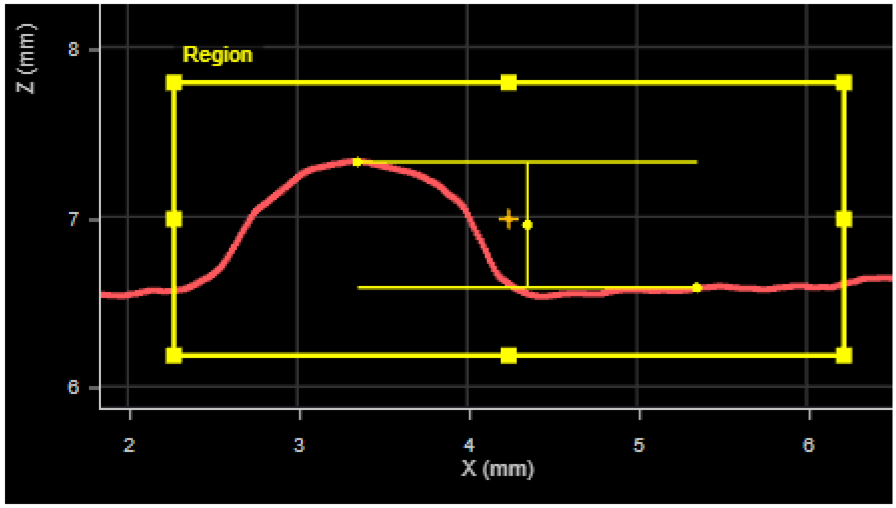
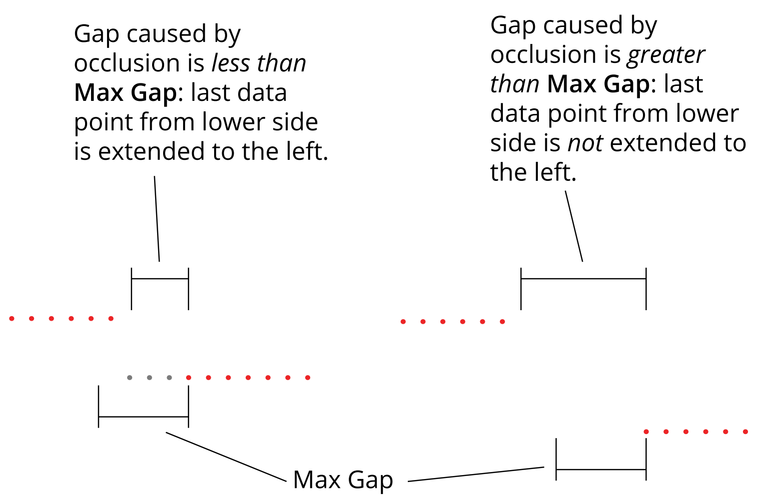
Parameter	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Edge Detection Mode	One of the following: Step or Corner. **Step**: Searches for steps on each path profile. When the edge detection mode is set to Step, you must set several additional parameters. For more information, see *Step Edge Detection Mode Parameters* on the next page.

Parameter	Description
	Corner: Searches for slopes on each path profile. When the edge detection mode is set to Corner, you must set the **Corner Type** parameter. For more information, see *Corner Edge Detection Mode Parameters* on page 332.
Show Detail	When the parameter is enable, the tool displays a line joining the points used to calculate the edge. This can be useful when viewing the edge's center point on a step. Note that enabling this parameter can have an effect on processing. For high frequency scanning, be sure to disable it. 
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Step Edge Detection Mode Parameters

Parameter	Description
Selection Type	Determines which step the tool uses when there are multiple steps in the profile. An edge point is placed on the chosen step. Steps must satisfy the tool's **Step Threshold** and **Step Direction** settings. **Best**: Selects the greatest step on the profile. **First**: Selects the first step on the profile. **Last**: Selects the last step on the profile.
Step Direction	Determines whether the expected step rises or falls, moving from left to right along the profile. Either **Rising**, **Falling**, or **Rising or Falling**.
Step Threshold	The minimum step accepted as an edge candidate. Steps on the profile are treated as absolute values when compared to this setting. In the following profile, with **Step Threshold** set to 1.2 (and **Selection Type** set to First), the tool accepts the rising step to the left, with a step of 1.525 mm, because it is the first above the step threshold.

Parameter	Description
	 In the following, when **Step Threshold** is increased to 2.7, the tool excludes the rising step to the left, because it is no longer above the step threshold, and instead uses the step to the right. 
Step Smoothing	The size of the (moving) window along the profile used to calculate averages for the data points used to calculate the step. The setting is useful for averaging out noise. If **Step Smoothing** is set to 0, no averaging is performed.
Step Width	The distance, along a profile, separating the points the tool uses to find steps on a profile. In the following, a step width of 2 mm causes the tool to consider profile points that distance apart as steps. The curved portion of the profile is not used to measure the step.

Parameter	Description
	 The setting is useful when you must detect a slope as an edge, rather than a sharply defined edge: setting **Step Width** to a value greater than the width of the slope ensures that the tool measures the height difference between the flat regions on either side of the slope. As a result, the height of the step is accurately measured, and the edge is correctly located. Setting **Step Width** wider than necessary can reduce the accuracy of edge location.
Max Gap	Fills in regions of missing data caused by an occlusion near the desired edge. Use this setting when continuity on the target is expected. When **Max Gap** is set to a non-zero value, the tool holds and extends the last data point on the low side next to an edge across a gap of null points, up to the distance specified in **Max Gap**.  The tool uses data points "filled in" by **Max Gap** before data points filled in by **Null Fill Value** (see below).
Include Null Edges	Indicates whether null points (points where no height value is available, due to

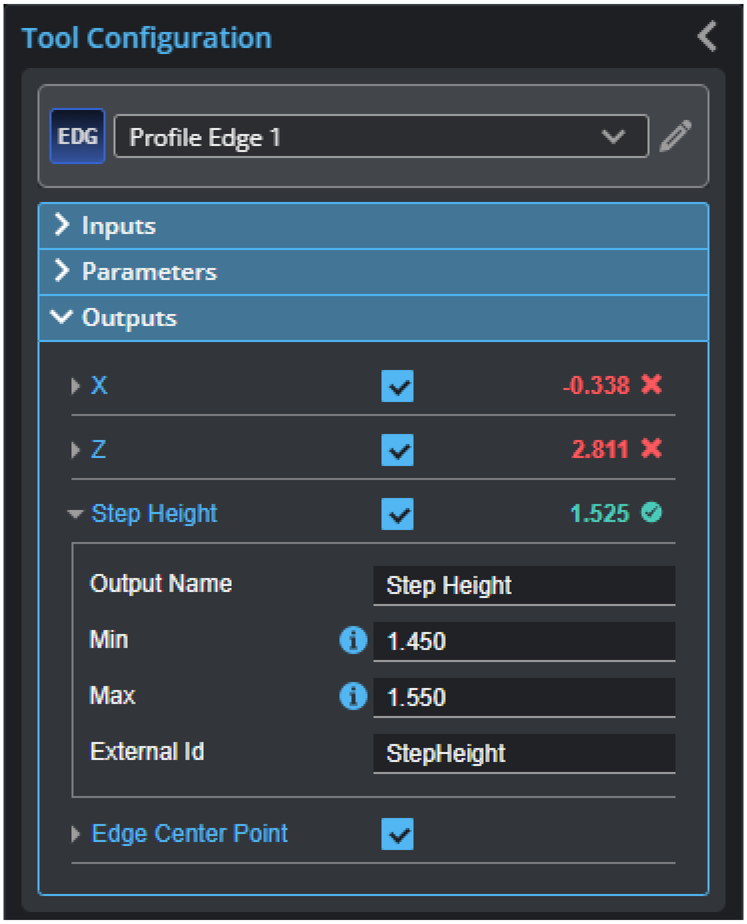
Parameter	Description
	dropouts or regions outside of the measurement range) are filled with the value in **Null Fill Value** as a general “background level.” To find an edges next null points, you must use either this option and an appropriate value in **Null Fill Value** or **Max Gap**. Otherwise, only edges within areas of contiguous data will be detected.
Null Fill Value	The height value (in mm) used to replace null points when **Include Null Edges** is enabled. If both **Null Fill Value** and **Max Gap** fill in null points at the same position, the tool uses the value extended by **Max Gap**, regardless of the value of **Null Fill Value**.

Corner Edge Detection Mode Parameters

Parameter	Description
Corner Type	Determines the corner type the tool searches for. One of the following: **Best**: Selects the greatest step in the profile data. **Left**: Selects the left-most step in the profile data. **Right**: Selects the right-most step in the profile data. **Top**: Selects the step that is the highest in the profile data. **Bottom**: Selects the step that is lowest in the profile.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
X **Z** These measurements return the X and Z position of the edge point, respectively. The edge point is located half-way between the upper and lower data points of the step.
Step Height Returns the height of the step on the profile. Only available if **Edge Detection Mode** is set to Step.

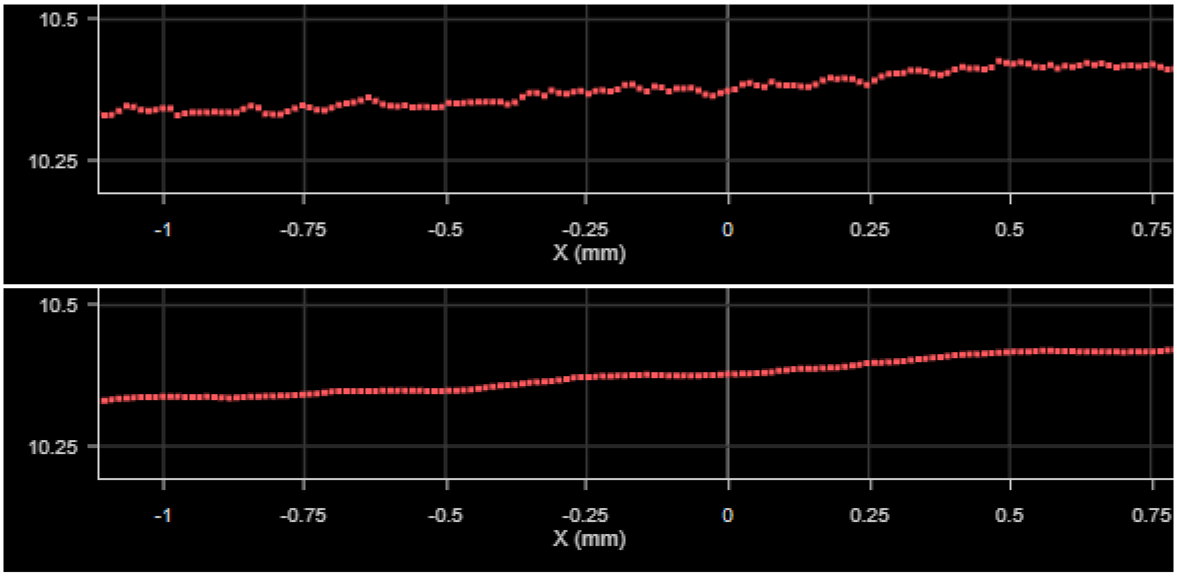
Features

Type	Description
Edge Center Point	The edge point.

For more information on geometric features, see *Geometric Features* on page 262.

Profile Filter

The Profile Filter tool provides processing filters that you can apply to a uniform profile (not point cloud profiles), letting you process scan data to get more repeatable measurements. You can enable up to seven of the filters at once, in any order. Filters in the tool are chained together. Any Profile tool can use the resulting filtered profile as input.



Profile with no filtering (top) and with averaging filter (bottom)

For a list of the filters, see *Filters* on page 338.

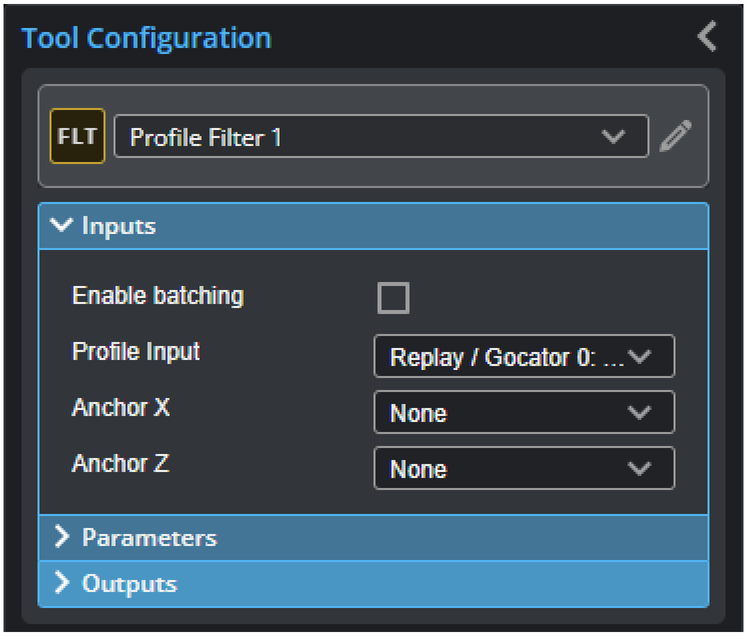
The Filter tool provides no measurements or decisions. It only outputs processed profile data.

> A limited set of filters is also available on the **Scan** page. These filters let you process scan data without needing to add tools. This can be useful if you are using a sensor mostly as an acquisition device.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



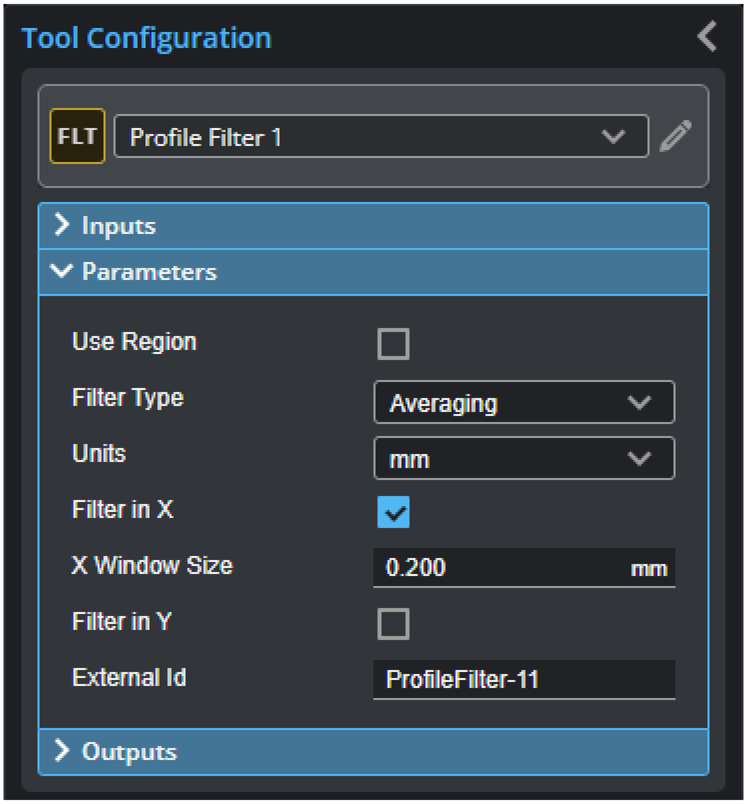
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Number of Regions Region {n}	Lets you set the number of regions, and for each region, the position and dimension. For more information on regions, see *Regions* on page 250.
Filter Type	The type of filter. For more information on the available filters, see *Filters* on the next page.
Units	The units the filter uses for the window or windows: points or distance (mm). Not available with all filters. This parameter is only displayed after enabling **Filter in X** or **Filter in Y**.
Filter in X Filter in Y	When enabled, filters along X and Y, respectively, and displays **X Window Size** or **Y Window Size** parameters you use to set the window size, using the unit set in **Units**. (The gap filling and Gaussian filters only let you filter along the X axis.) The filter window sizes are specified in millimeters (and additionally, in the tools, in data points). To calculate the number of data points that a window covers when the units are millimeters, use the following calculation: • User-specified window size divided by the X spacing interval (that is, the number of millimeters per point) on the **Spacing** tab in the **Sensor** panel. (For more information on spacing intervals, see *Uniform Spacing* on page 197.)

Parameter	Description
	• *With the exception of the gap filling filter*, round the result of the division to the nearest integer value. With the gap filling filter, filling is performed within the provided window size. For example, if you set the size of the filter's window to a value between 1.5 mm and 2.49 mm (inclusively), and the X spacing interval is set to 1 mm, the filter covers 2 data points. A filter window size from 2.5 mm to 3.49 mm results in a filter covering 3 data points.
Sigma	The Gaussian curve's sigma value. (Only displayed with the Gaussian filter and when **Filter in X** is enabled.)
Percentile Type	Displayed when **Filter Type** is Percentile. (Be sure to set the **High Percentile** and **Low Percentile** as required.) One of the following: **Point Count**: Removes points based on the count of valid points in the profile. **Point Height Range**: Removes points based on the height range of valid points in the profile.
High Percentile Low Percentile	The range for used when you set **Filter Type** to Percentile. You must also set **Percentile Type**.

The following filters are available in the Profile Filter tool.

Filters

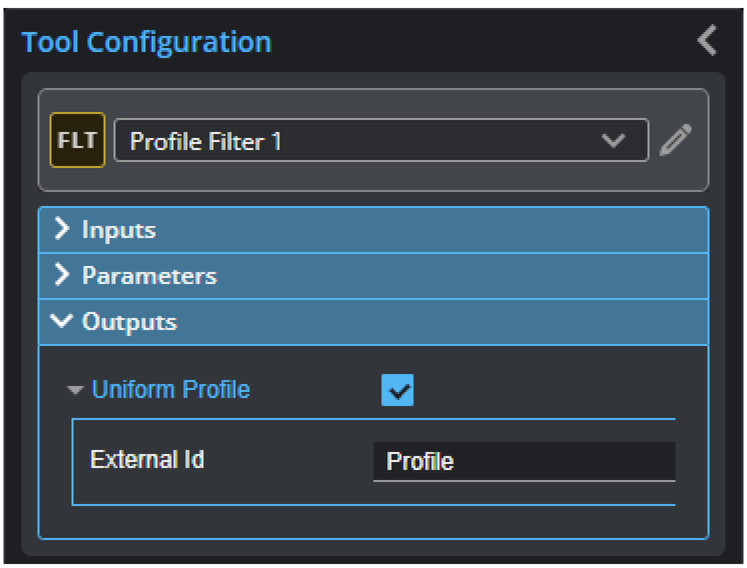
Name	Description
Gap Filling	Fills in missing data using information from the nearest neighbor data points, for example, when data is missing due to occlusions. Gap filling also fills gaps where no data is detected, which can be due to low surface reflectivity, for example dark or specular surface areas, or to actual gaps in the surface. The values in **Filter in X** and **Filter in Y** represent the maximum gaps the sensor will fill. Wider gaps are not filled. Gap filling works by filling in missing data points using either the lowest values from the nearest neighbors or linear interpolation between neighboring values (depending on the Z difference between neighboring values), in the specified window. The sensor can fill gaps along both the X axis and the Y axis. In Profile mode, gap filling is limited to the X axis.
Median	Substitutes the value of a data point with the median calculated within the window or windows set in **X Window Size** or **Y Window Size** around the data point. If the number of valid (non-null) data points in the window is even, the median value is simply the value in the center of the sorted list of values. If the number of valid points is odd, the average of the two values in the center is used instead. Missing data points will not be filled with the median value calculated from data points in the neighborhood. With an odd window size, the output is at the center of the window. With an even window size, the output is 0.5 pixels to the right of the center (that is, using window / 2-1 values from the left, and window / 2 from the right.

Name	Description
Averaging	Substitutes a data point value with the mean value of that data point and its nearest neighbors within the window or windows set in **X Window Size** or **Y Window Size**. X smoothing works by calculating a moving average across samples within the same profile. Y smoothing works by calculating a moving average in the direction of travel at each X location. If both X and Y smoothing are enabled, the data is smoothed along X axis first, then along the Y axis. Missing data points will not be filled with the mean value calculated from data points in the neighborhood.
Decimation	Decimation reduces the number of data points along the X or Y axis by choosing data points at the end of a specified window around the data point. For example, by setting **X Window Size** to 0.2, only points every 0.2 millimeters will be used. The filter generates points starting from the leftmost edge of the scan data, stepping in equal steps away from that side.
Gaussian	A Gaussian filter applied over the specified kernel using the provided sigma. Enable **Filter in X** and set the **Sigma** parameter.
Percentile	Filters points based on **Percentile Type**, as well as **High Percentile** and **Low Percentile**.

The filters described here that are available in the Filter panel topic use the same algorithms.

Outputs

The tool provides the following output.

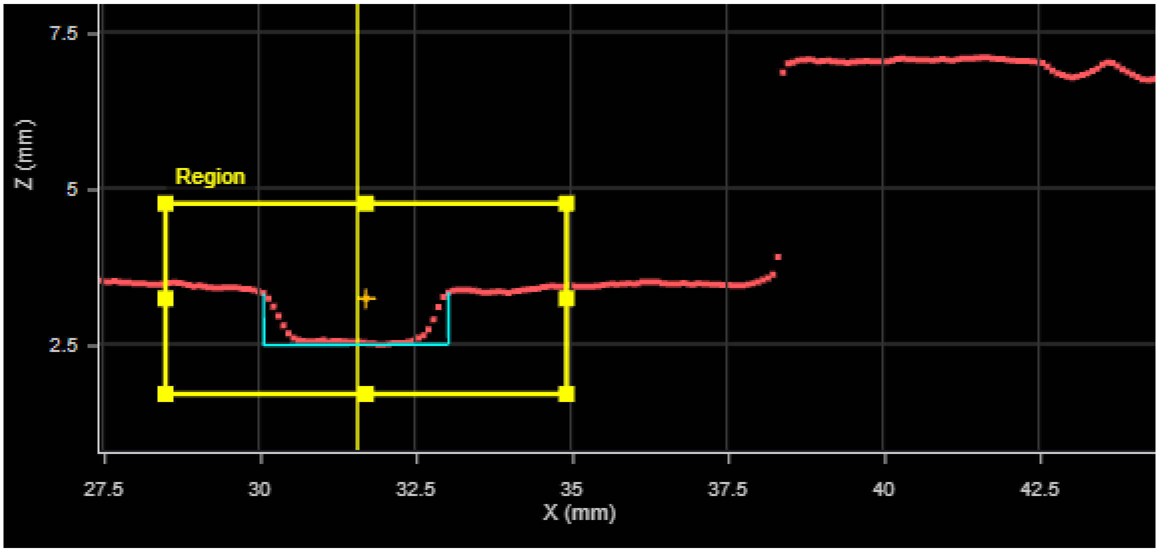


Data

Type	Description
Uniform Profile	The filtered uniform profile.

Profile Groove

The Groove tool provides measurements of V-shape, U-shape, or open-shape grooves.



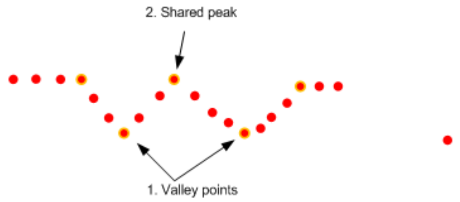
The tool lets you optionally set an index to return the measurements of a specific groove when more than one groove is visible in the region of interest. If you need to return multiple grooves from a profile, add a Profile Groove tool for each groove, configure it to represent the desired groove, and set its **Selection Index** parameter to the desired groove (0-based index).

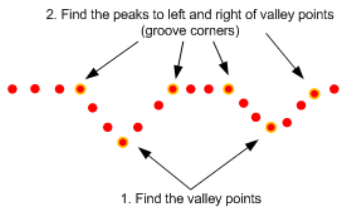
The Groove tool uses a complex feature-locating algorithm to find a groove and then return measurements. For a detailed explanation of the algorithm the tool uses, see *Groove Algorithm* below. The behavior of the algorithm can be adjusted by changing the tool's parameters.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

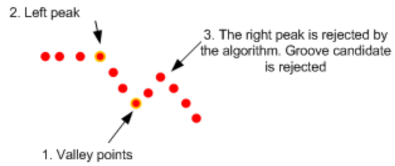
Groove Algorithm

The Groove measurement tool first locates a valley along the profile line. The bottom point of a valley, the valley point, is the first estimation of the position of the groove bottom. For each valley, the algorithm searches for the corner to the left and to the right to find the groove corners. A groove candidate is found when the groove corners are located on the left and right before the next valley is reached. Two groove candidates may share the same corner as shown in the right image below.

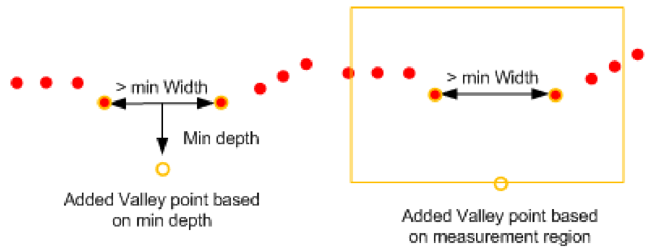




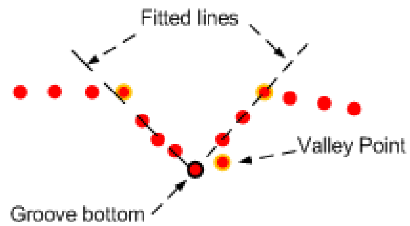
The algorithm derives search parameters from the user settings to prevent noise from triggering false detections. When detecting multiple grooves, an adaptive algorithm is used to ensure that candidate grooves are approximately the same scale.



The valley points of open grooves may not be visible or may fall outside of the measurement region. Voids in the data (regions with no profile data) between pairs of valid points are detected. A valley point is added midway between the pair of valid points. The Z position of the valley point is either the minimum groove depth below the lower of the corners or the bottom edge of the measurement region. The algorithm then proceeds as if to find a U-shaped groove.

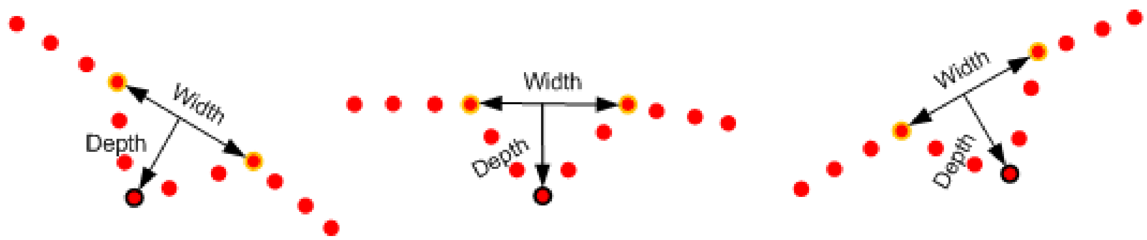


The actual groove bottom is calculated differently for different shapes. For a V-shaped groove, a line is fitted to the sides of the valley points starting from the corners, up to (but not including) the valley point. The groove bottom is the intersection of the left and right lines. Line fitting is used such that an accurate groove bottom can be found even when the real bottom is not visible (i.e., blocked by reflections).



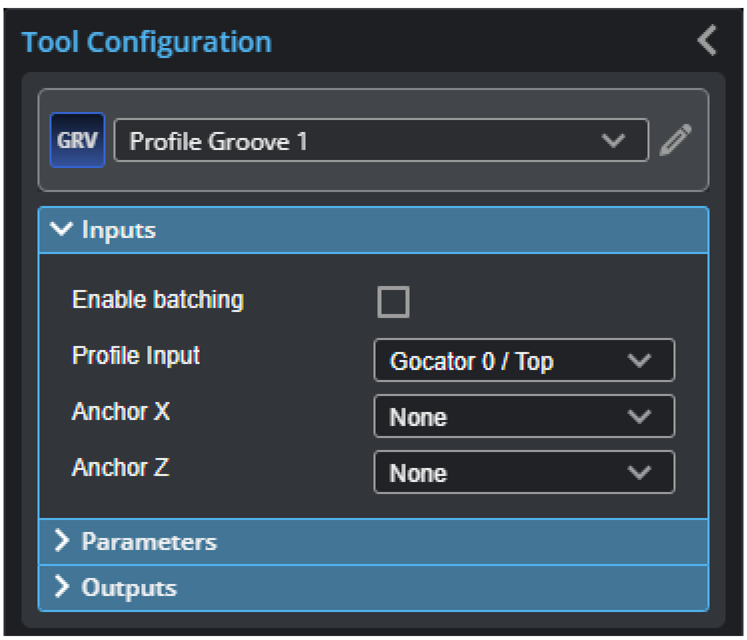
For U-shaped and open groove, the X and Z positions are at the bottom of the groove.

Groove candidates that do not meet the minimum and maximum width and depth settings are rejected. The width and depth measurements are invariant to the groove rotation. The width is the distance between the groove corners and the depth is perpendicular distance of the groove bottom from the groove width.



Inputs

You configure the tool's inputs in the expandable **Inputs** section.



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

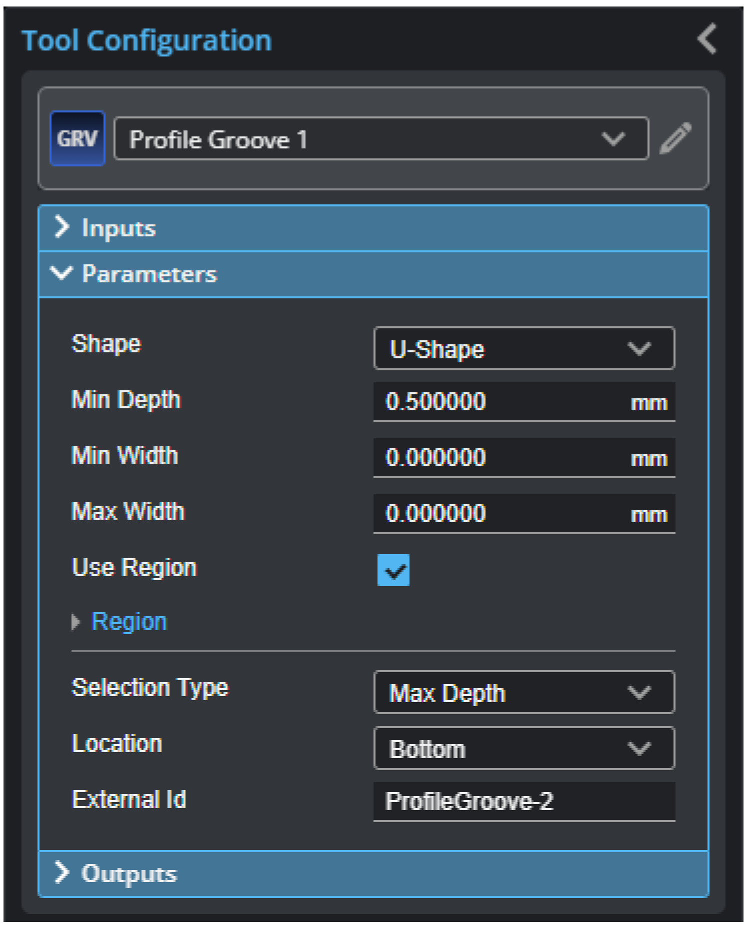
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor.

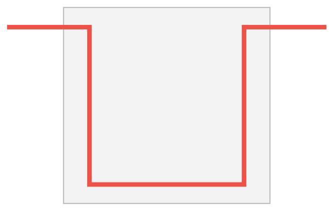
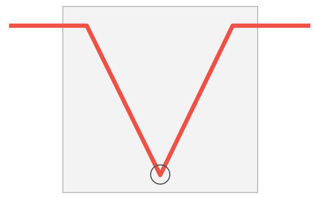
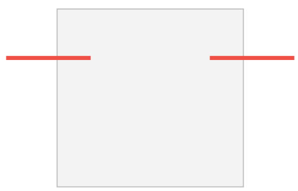
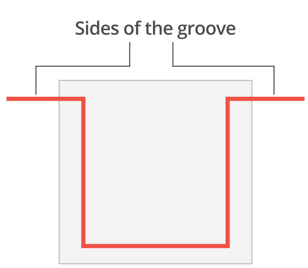
Name	Description
	When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. If **Enable Batching** is *disabled* and the passed array contains more than two elements, GoPxL displays an error. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



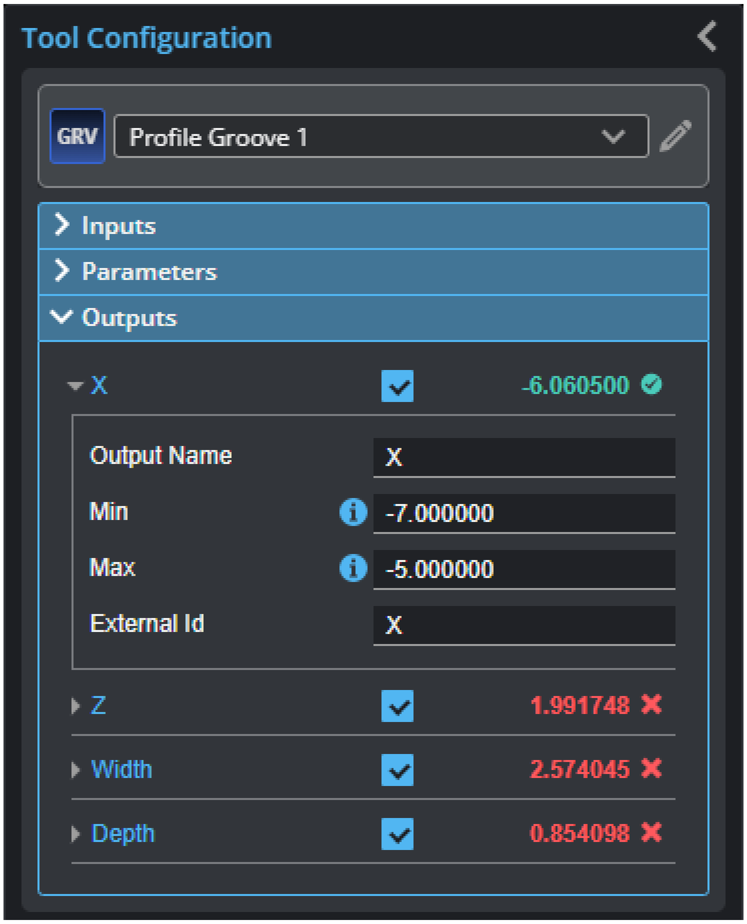
Parameters

Parameter	Description
Shape	One of the following: U-Shape, V-shape, or Open. U-Shape  V-Shape  Open Shape 
Min Depth	Minimum depth for a groove to be considered valid. When set to 0, any groove depth is accepted.
Min Width	Minimum width for a groove to be considered valid. The width is the distance between the groove corners. When set to 0, any groove width is accepted.
Max Width	Maximum width of a groove to be considered valid. If set to 0, the maximum is the width of the region of interest, or the entire profile if **Use Region** is disabled.
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The measurement region defines the region in which the tool searches for a groove. For a stable measurement, the measurement region should be large enough to cover some data on the left and right sides of the groove.  For more information on regions, see *Regions* on page 250.

Parameter	Description
Selection Type	Specifies how a groove is selected when there are multiple grooves the profile data. One of the following: **Maximum Depth**: The groove with the maximum depth. **Index from The Left**: A 0-based groove index, counting from left to right. **Index from the Right**: A 0-based groove index, counting from right to left. When you choose either of the "Index from" selection types, a **Selection Index** parameter is displayed.
Selection Index	The 0-based groove index the tool uses to select a groove when multiple grooves are visible in the data. The direction depends on whether you set **Selection Type** to **Index from the Left** or **Index from the Right**.
Location *(For the X and Z measurements only)*	Specifies the location type to return. One of the following: **Bottom**: The groove bottom. For a U-shape and open-shape groove, the X position is at the center of the bottom of the groove. For a V-shape groove, the X position is at the intersection of lines fitted to the left and right sides of the groove. For more details, see *Groove Algorithm* on page 340. **Left Corner**: The groove's left corner. **Right Corner**: The groove's right corner.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

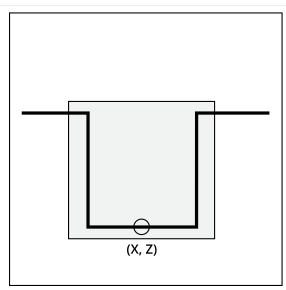
Most tools provide measurements, geometric features, or data as outputs.

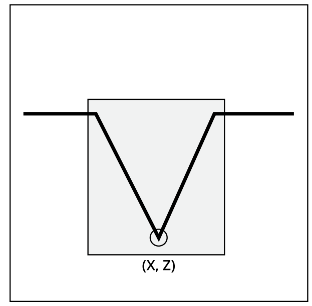
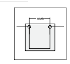
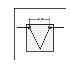
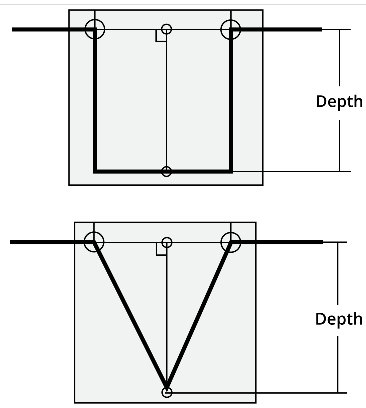


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X, Z Measures the X and Z position, respectively, of the bottom of a groove.	

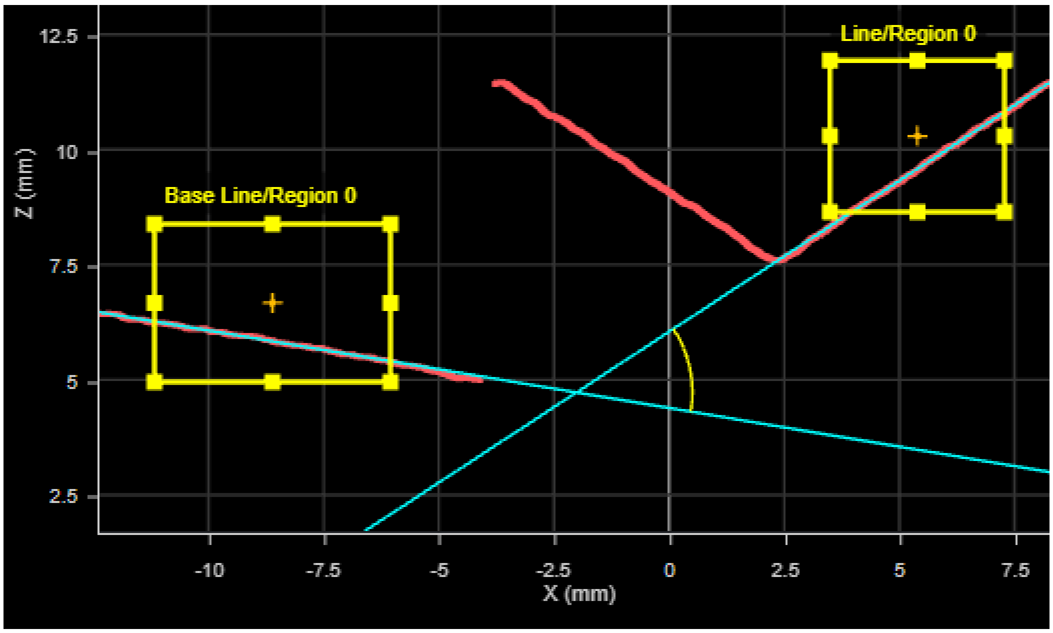
Measurement	Illustration
	
Width Measures the width of a groove.	 
Depth Measures the depth of a groove as the maximum perpendicular distance from a line connecting the edge points of the groove.	

Profile Intersect

The Intersect tool determines intersect points and angles.

The Intersect tool's measurements require two fit lines, one of which is a reference line set to one of the following:

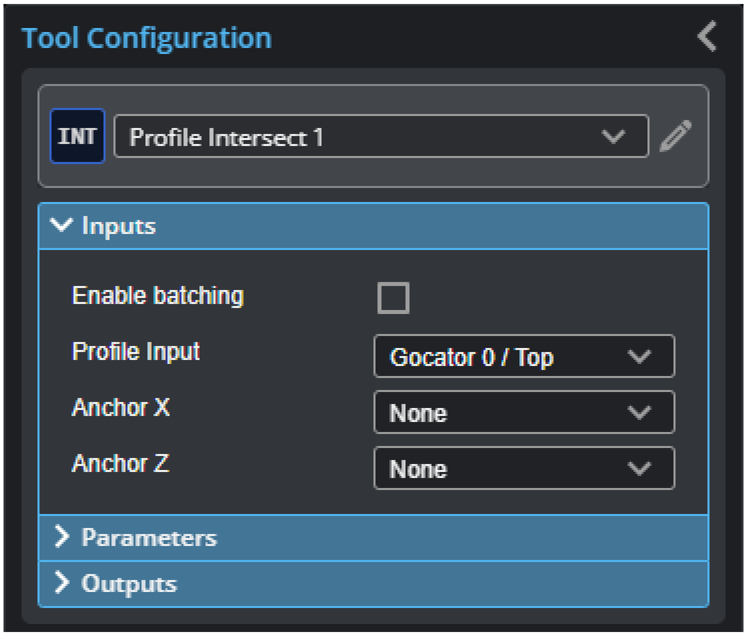
- the X axis (z = 0)
- the Z axis (x = 0)
- a user-defined line



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



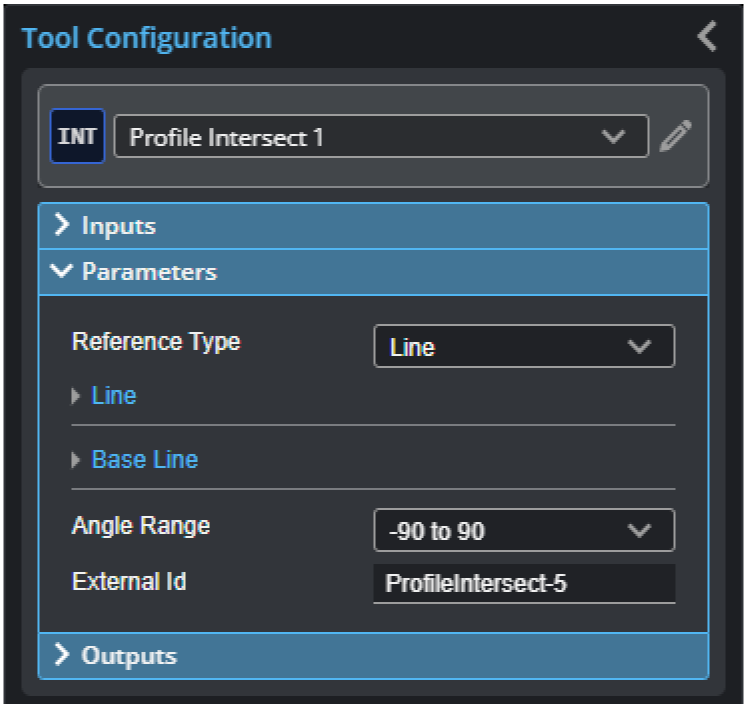
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. If **Enable Batching** is *disabled* and the passed array contains more than two elements, GoPxL displays an error. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

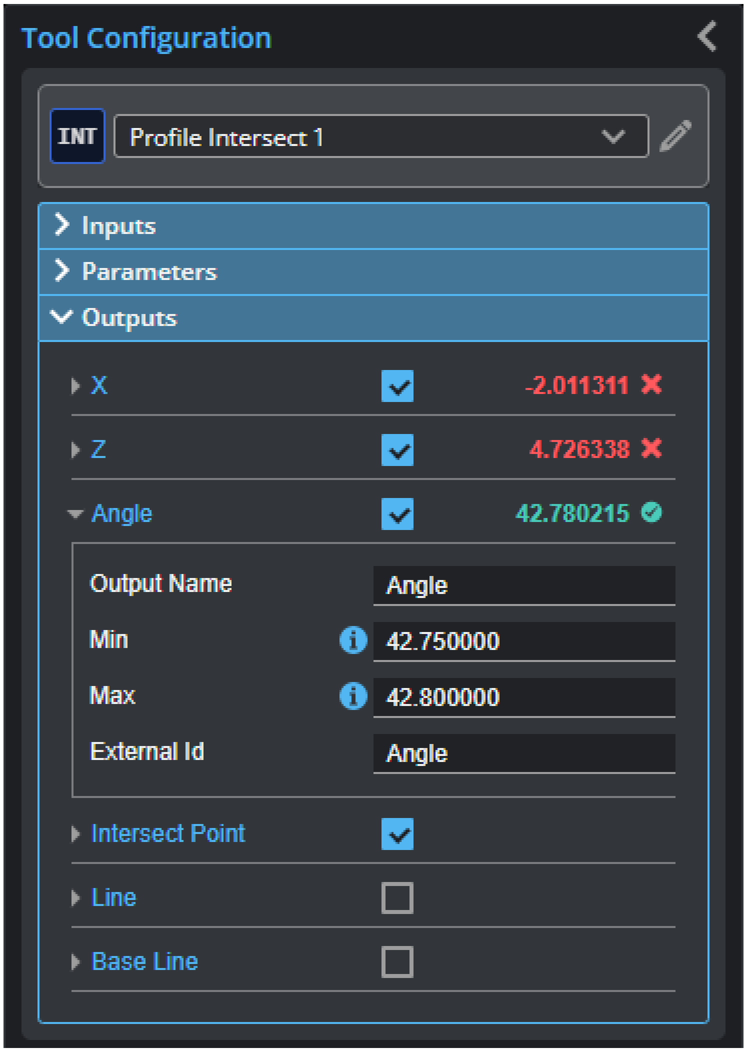


Parameters

Parameter	Description
Reference Type	Determines the type of the reference line. One of the following: X Axis, Z Axis, or Line. **X-Axis**The reference line is set to the X axis. **Z-Axis**: The reference line is set to the Z axis. **Line**: You define the reference line manually using the parameters in the **Base Line** section. You can define the line using one or two regions.
Line	You can use one or two fit areas for the fit line. To set the region (or regions) of the fit line, adjust it graphically in the data viewer, or expand the feature and enter the values in the fields. For more information on regions, see *Regions* on page 250. For more information on fit lines, see *Fit Lines* on page 263.
Base Line	Used to define the reference line when the **Reference Type** parameter is set to **Line**. For more information on regions, see *Regions* on page 250. For more information on fit lines, see *Fit Lines* on page 263.
Angle Range	Determines the range returned by the Angle measurement. One of the following: **-90 – 90** **0 – 180**
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

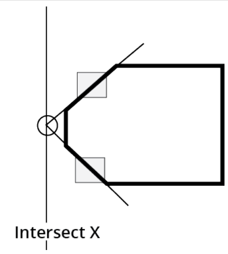
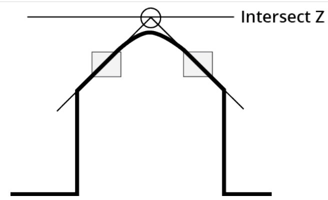
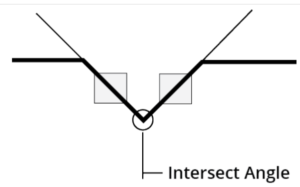
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Finds the intersection between two fitted lines and measures the X axis position of the intersection point.	
Z Finds the intersection between two fitted lines and measures the Z axis position of the intersection point.	
Angle Finds the angle subtended by two fitted lines.	

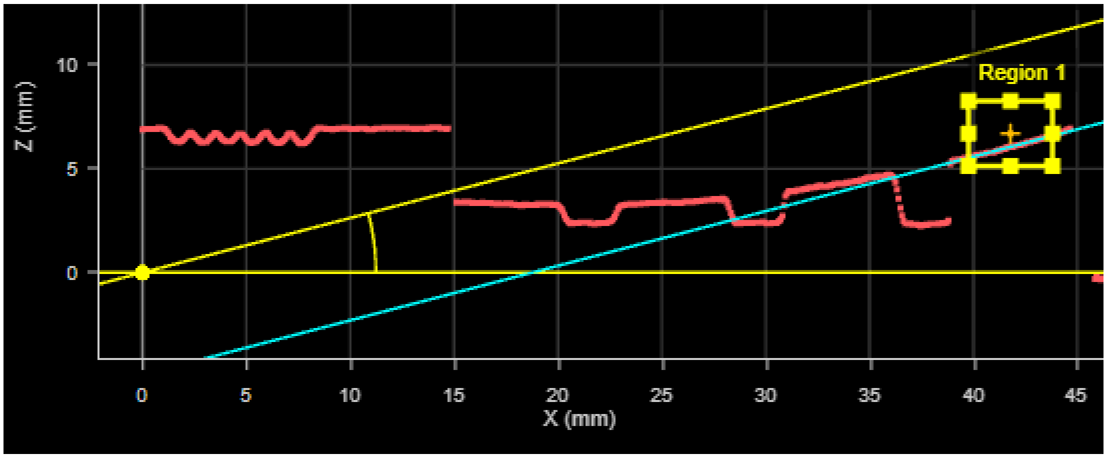
Features

Type	Description
Intersect Point	The point of intersection.
Line	The intersect line.
Base Line	The base line.

For more information on geometric features, see *Geometric Features* on page 262.

Profile Line

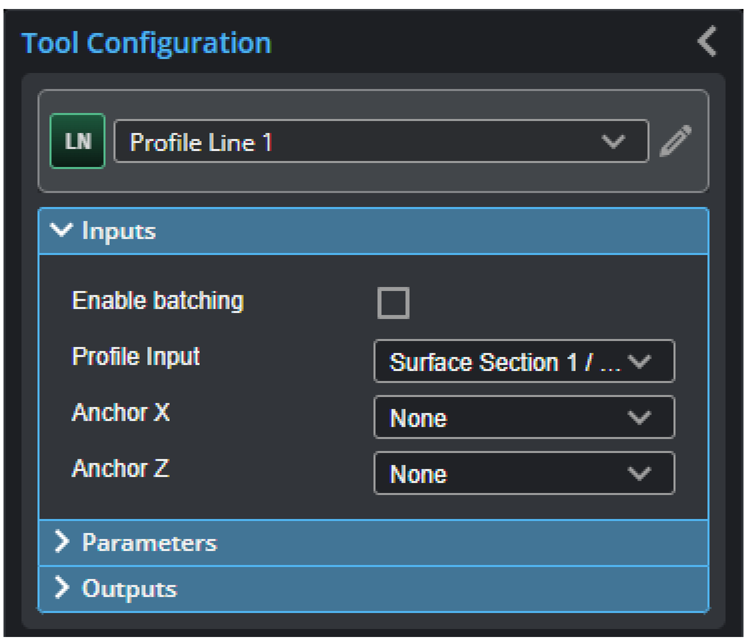
The Line tool fits a line to the profile and measures the deviations from the best-fitted line. The tool also provides two “roughness parameter” measurements: Arithmetic Average and Rz. For a complete list of the measurements the tool provides, see *Measurements* on page 357.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



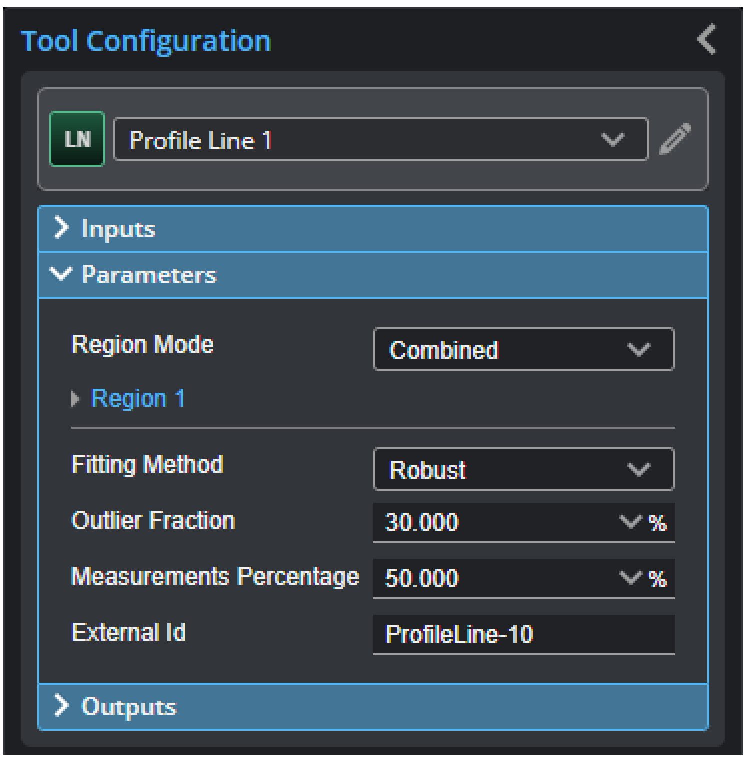
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

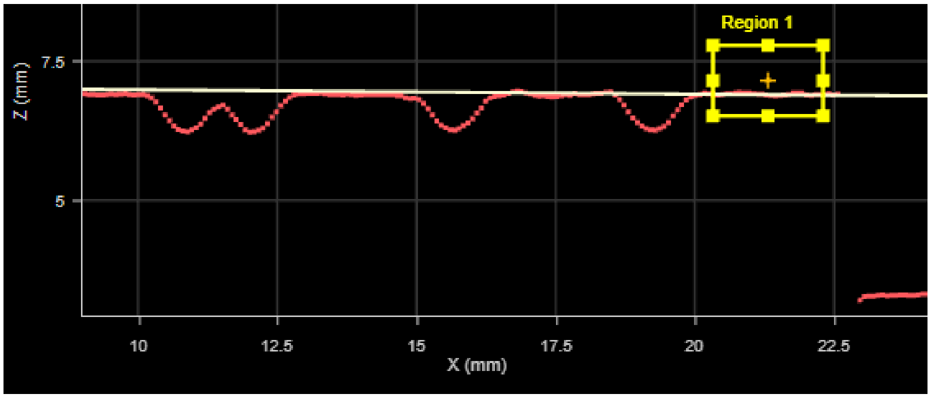
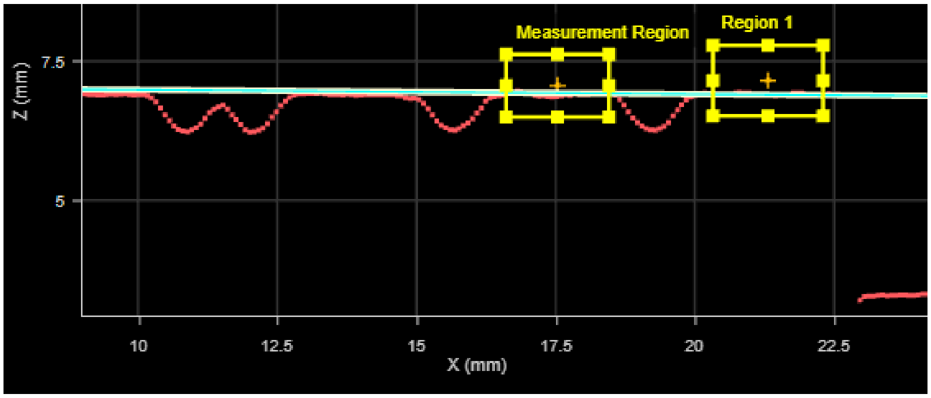
Parameters

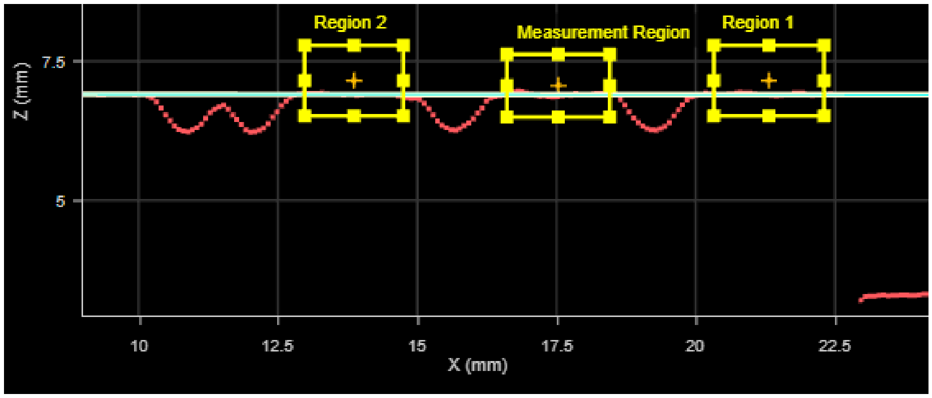
You configure the tool's parameters in the expandable **Parameters** section.



Parameters

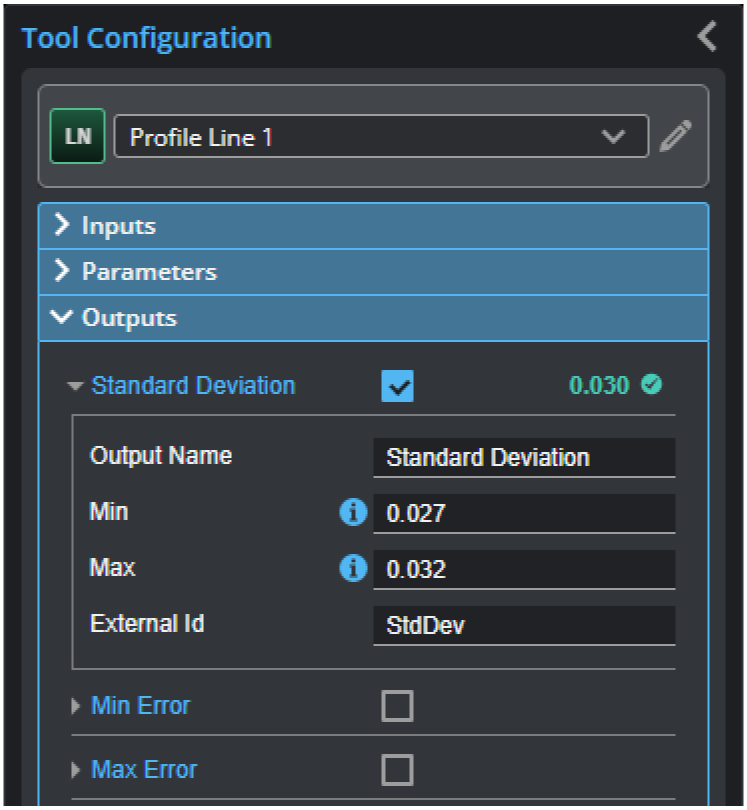
Name	Description
Region Mode	Whether the fitting and measurement regions are used, and whether they are combined or separate. One of the following:

Name	Description
	None The tool uses the entire profile both to fit the line and to perform measurements. **Combined** The tool uses a single, user-defined region to fit the line, in which it also performs measurements.  **Separated** The tool uses one or two regions to fit the line (selected using the **Number of Regions** parameter), and a single, separate region in which it performs measurements. (Offset and Angle measurements are related to the fitted line and are associated with the fitting region, rather than the measurement region.) In the following, the tool uses a single region to the left to fit the line, and performs measurements in the measurement region to the right:  In the following image, the uses two regions to the left to fit the line, and performs measurements in the measurement region to the right:

Name	Description
	
Number of Regions	When **Region Mode** is set to Separated, lets you choose the number of regions.
Region {n} Measurement Region	Use these expandable region sections to define the position and size of the fitting and measurement regions precisely. (Use the mouse to position and size them more roughly.) The number of region parameters displayed depends on the mode selected in **Region Mode**.
Fitting Method	Determines how the tool fits the line to the profile. One of the following: **Simple** Uses a less accurate but faster line-fitting method. **Robust** An iterative line-fitting method that removes points and attempts to fit a line until only the fraction set in **Outlier Fraction** of the original profile data points is left. More accurate but takes longer. When **Fitting Method** is set to Robust, set **Outlier Fraction** (see below).
Outlier Fraction	The fraction of outlier points to be removed during line fitting. Adjust this value based on how much noise is present in the profile. By default, 30%. Only displayed when **Method** is set to **Robust**.
Measurements Percentage *(Percentile measurement only)*	The specified percentage of points around the best-fitted line that the Percentile measurement uses.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

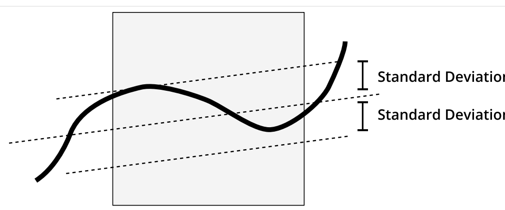
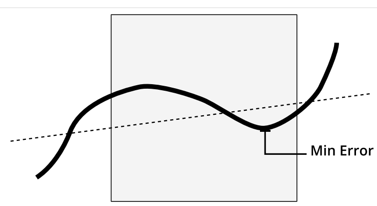
Most tools provide measurements, geometric features, or data as outputs.

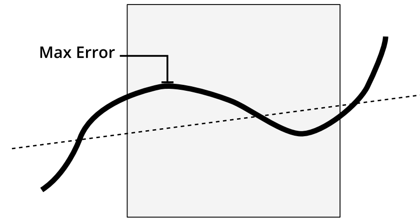
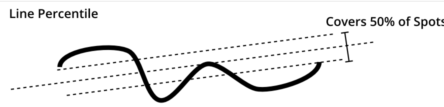
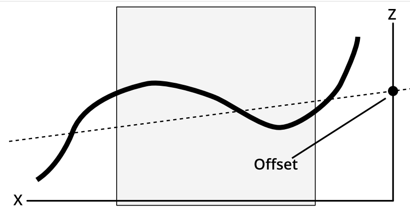
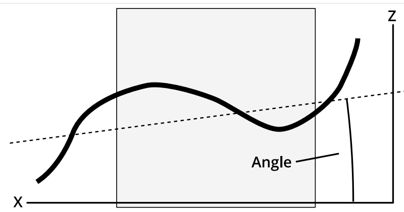
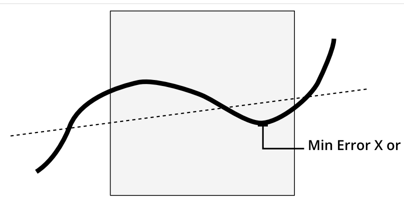
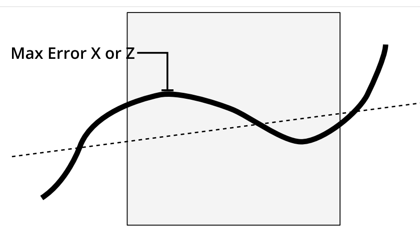
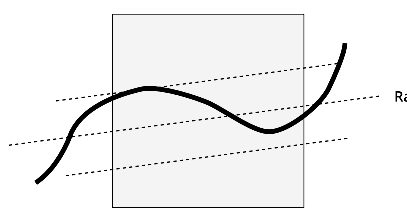


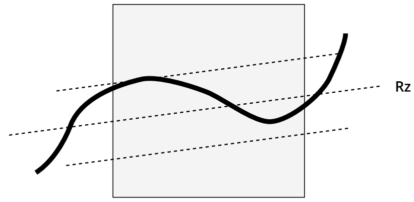
Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Standard Deviation Finds the best-fitted line and measures the standard deviation of the data points from the line.	
Min Error Finds the best-fitted line and measures the minimum error from the line (the maximum distance below the line).	

Measurement	Illustration
Max Error Finds the best-fitted line and measures the maximum error from the line (the maximum distance above the line).	
Percentile Finds the best-fitted line and measures the range (in Z) that covers a percentage of points around the line.	
Offset Finds the best-fitted line and returns the intersection point between that line and the Z axis.	
Angle Finds the best-fitted line and returns the angle relative to the X axis.	
Min Error X **Min Error Z** Finds the best-fitted line and returns the X or Z position of the minimum error from the line (the maximum distance below the line).	
Max Error X **Max Error Z** Finds the best-fitted line and returns the X or Z position of the maximum error from the line (the maximum distance above the line).	
Arithmetic Average Returns the roughness average of the profile data. Ra = SUM_N(abs(distance)) / (N) where N is the total number of non-outlier points, and the distance is from each point to the fitted line.	

Measurement	Illustration
Rz Returns the maximum height of the profile data. Rz = abs(smallest distance) + abs(biggest distance) The distance is from each non-outlier point to the fitted line.	

Features

Type	Description
Line	The fitted line.
Min Error Point	The point of minimum error.
Max Error Point	The point of maximum error.

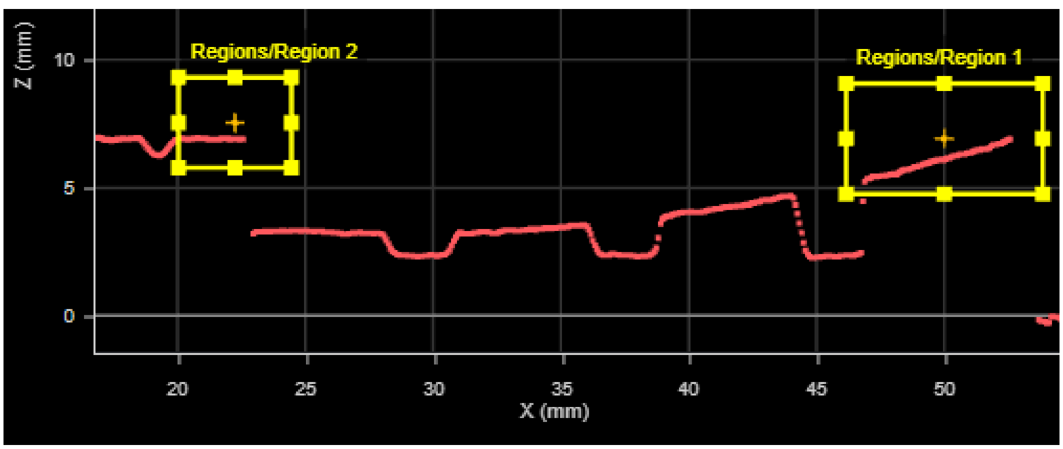
For more information on geometric features, see *Geometric Features* on page 262.

Profile Mask

The Profile Mask tool lets you define up to 16 regions to extract data from a profile. Each region's size, position, and shape (circular, elliptical, and rectangular) can be individually configured, and regions can overlap. The tool can also exclude inner data of circular and elliptical regions, letting you avoid measuring noise or unwanted areas of profile data. Extracted data is output in a single profile.

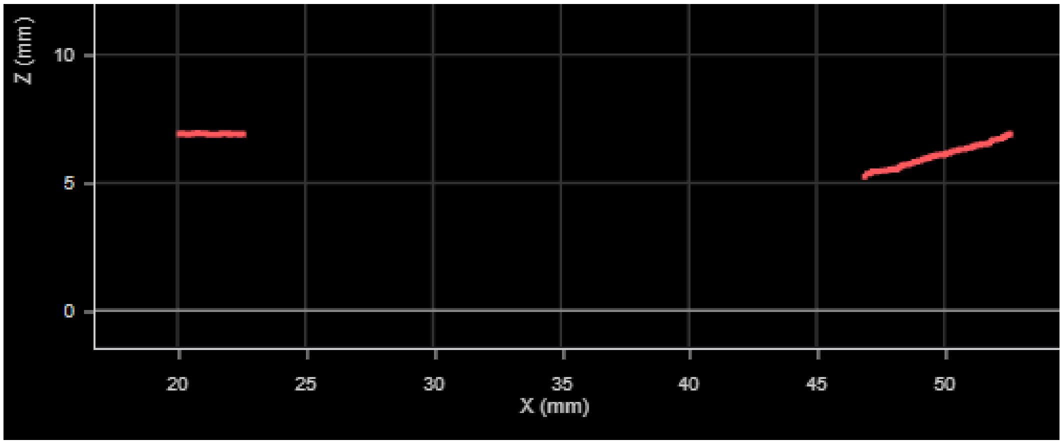
The resulting profile can then be further processed or measured by other tools.

For example, given the following scan data:



Two mask regions defined on a profile (original profile, all data included)

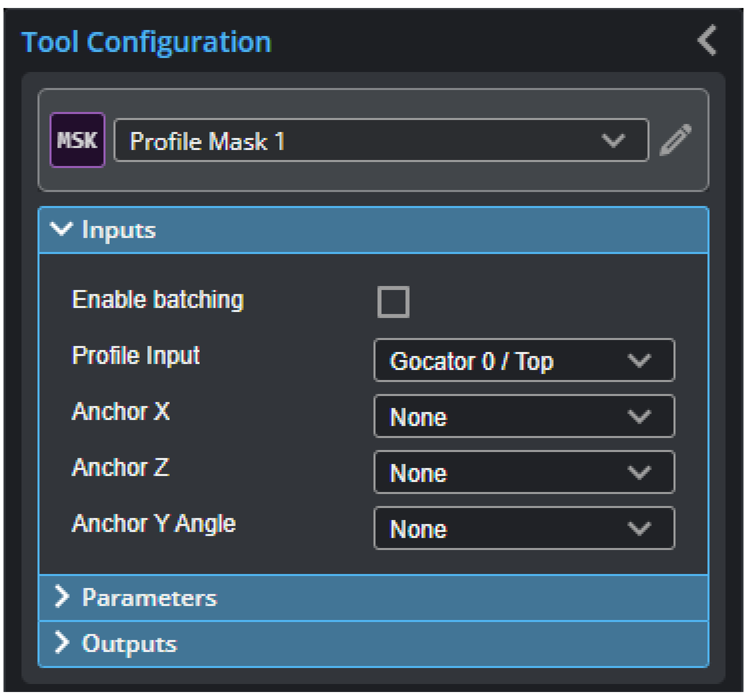
The image below shows the extracted data. The extracted profile data can then be further processed by other tools, or measurements can be applied to the surface data.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

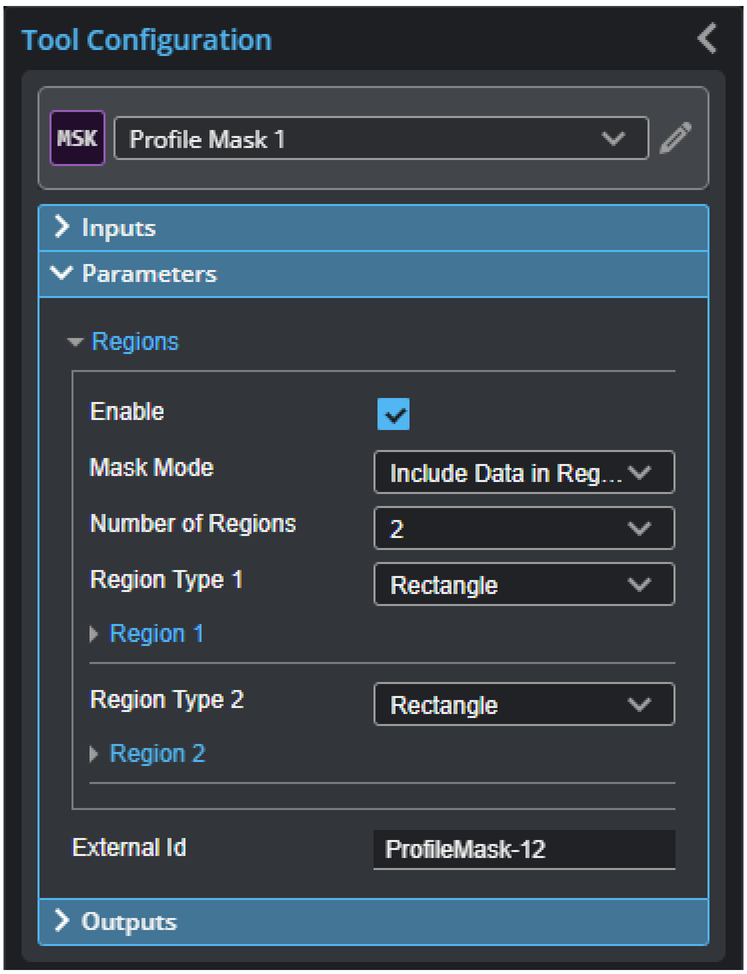
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other

Name	Description
	than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Y Angle	The Y angle measurement of another tool that this tool uses as a rotational anchor. Rotational anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



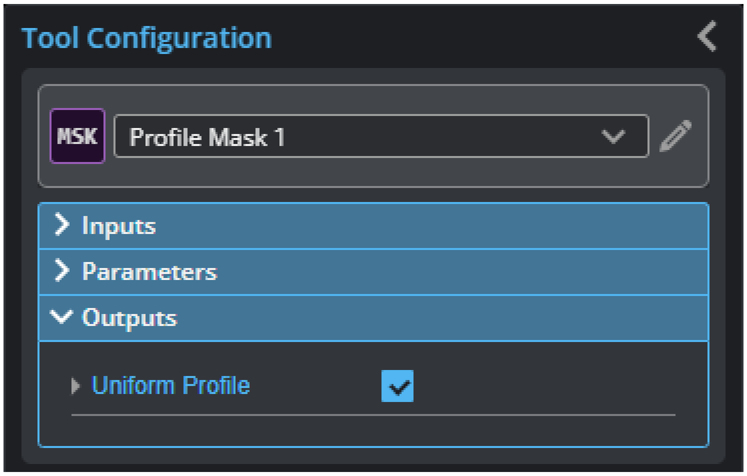
Parameters

Parameter	Description
Regions	When expanded, displays region-related settings. For details on the regions and their settings, see *Flexible Regions* on page 253.

Parameter	Description
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Data

Type	Description
Uniform Profile Point Cloud Profile	The profile containing the extracted region or regions.

Profile Panel

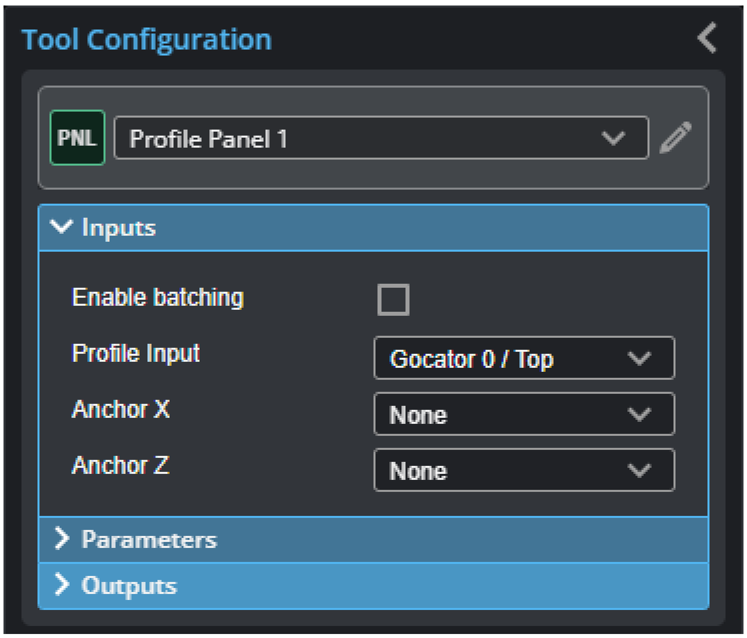
The Panel tool provides Gap and Flush measurements.

The Panel tool uses a complex feature-locating algorithm to find the gap or calculate flushness and return measurements. The behavior of the algorithm can be adjusted by changing the tool's parameters. For a detailed explanation of the algorithm, see *Gap and Flush Algorithms* on page 371.

> You must make sure that there are enough data points to define the edge in the profile, by properly setting up exposure, etc. If not, the algorithm will not function.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

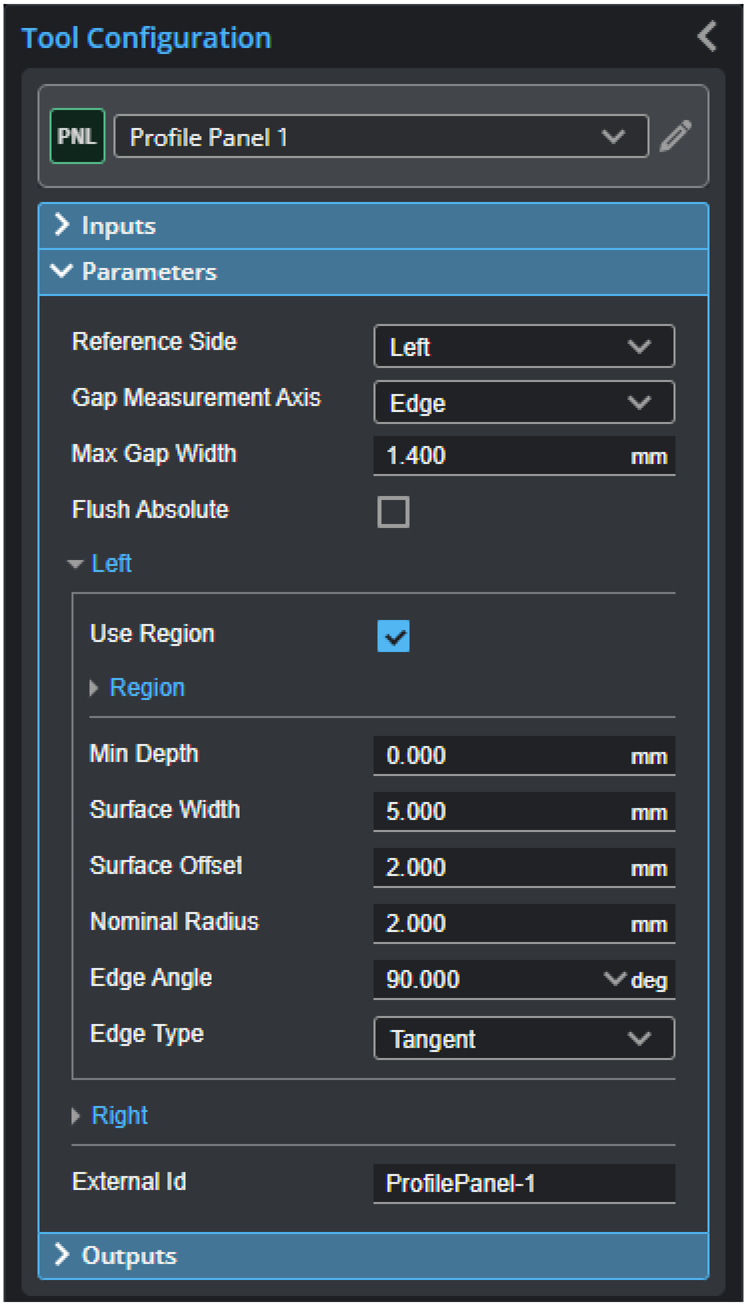
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.

Name	Description
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. If **Enable Batching** is *disabled* and the passed array contains more than two elements, GoPxL displays an error. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

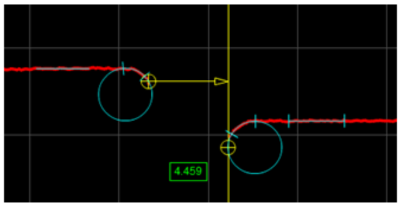
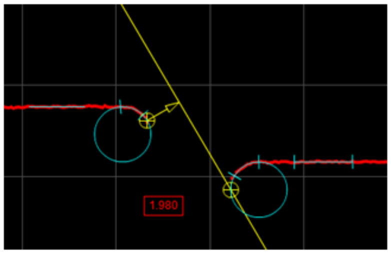
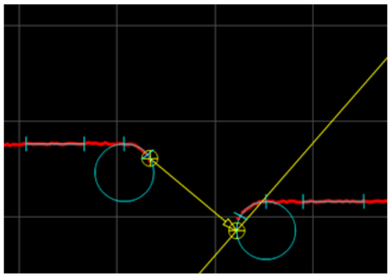
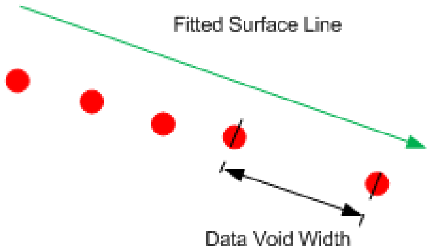
Parameters

You configure the tool's parameters in the expandable **Parameters** section.

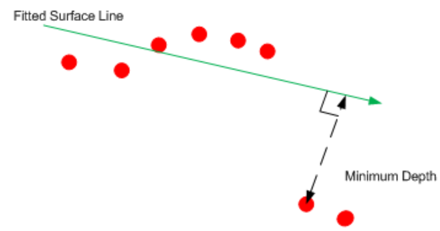
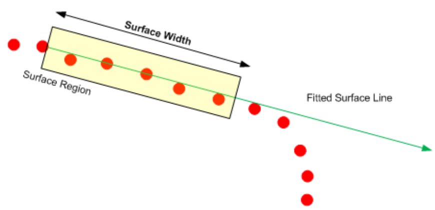
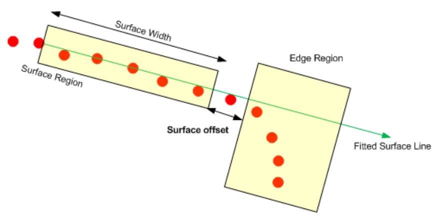
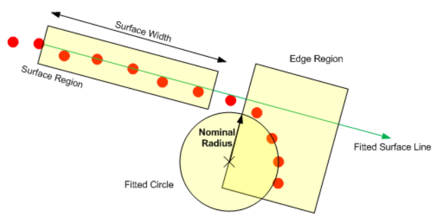


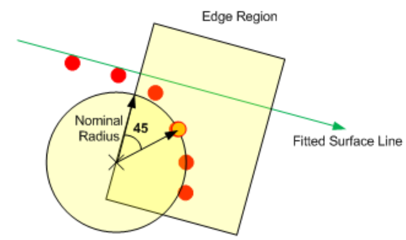
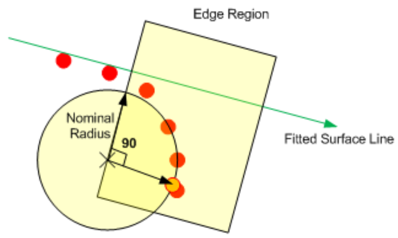
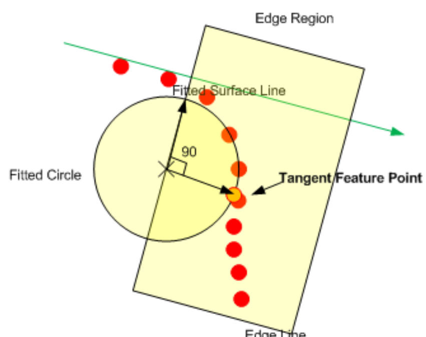
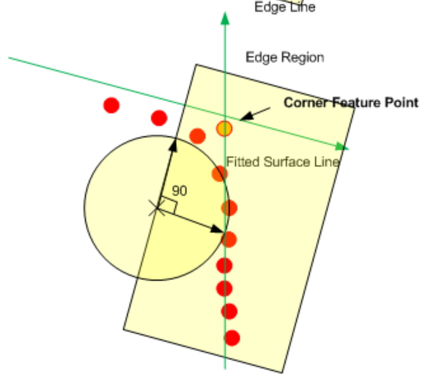
Parameters

Parameter	Description
Reference Side	Defines the side used to calculate the measurement axis (see below) rounded corner.
Gap Measurement Axis *Gap measurement only*	Defines the direction that the gap is calculated, in relation to the reference side (see above). **Surface**: In the direction of the fitted surface line of the reference surface. **Edge**: In the direction perpendicular to the edge of the reference surface.

Parameter	Description
	Distance: The Cartesian distance between the two feature locations. Surface Axis  Edge Axis  Distance Axis 
Max Gap Width	The maximum width of the gap. Allows the tool to filter gaps greater than the expected width. This can be used to single out the correct gap when there are multiple gaps in the field of view. 
Flush Absolute *Flush measurement only*	When enabled, the Flash measurement returns an absolute value instead of a signed value.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

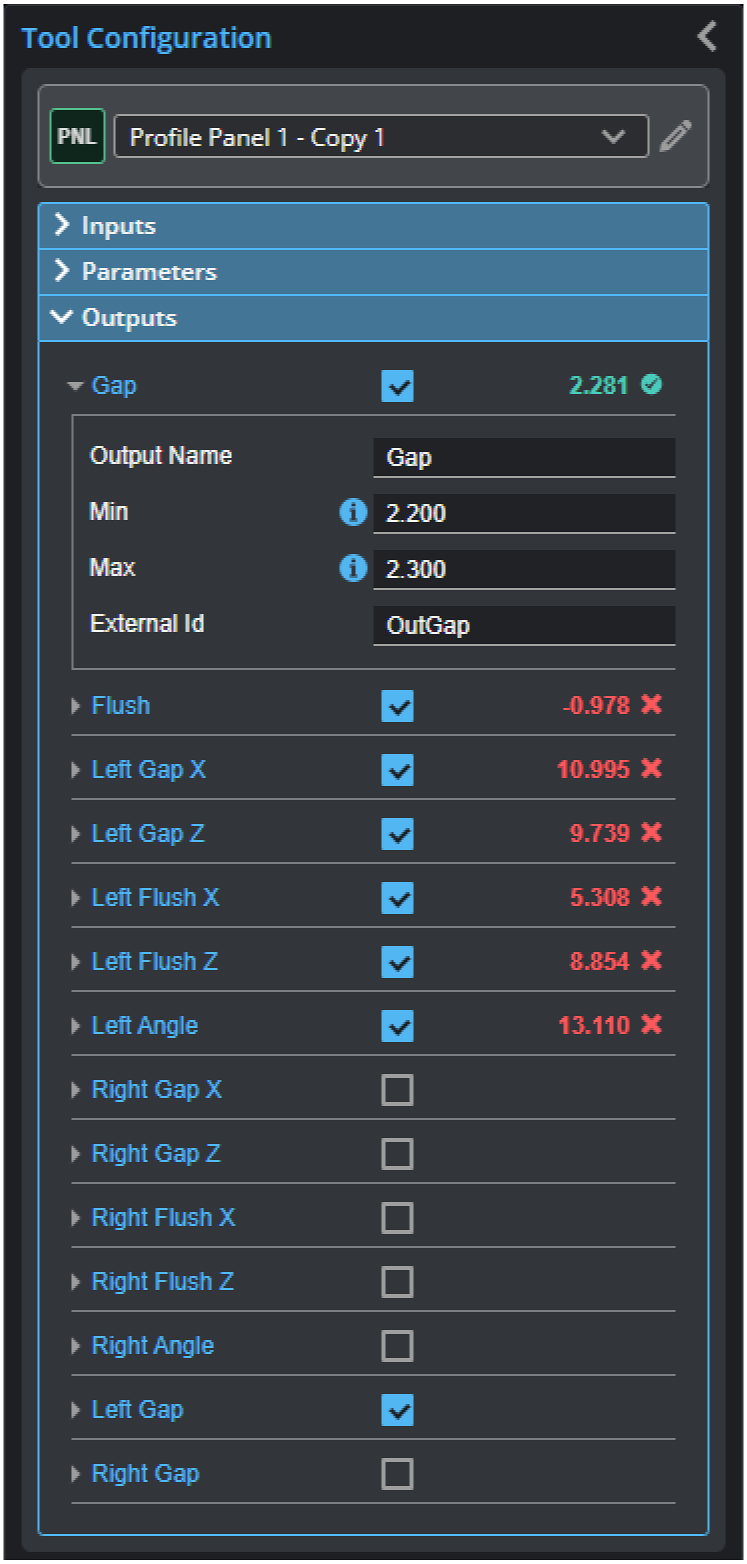
Left and Right Parameters

Parameter	Description
Min Depth	Defines the minimum depth before an opening can be considered to have a potential edge. The depth is the perpendicular distance from the fitted surface line. 
Surface Width	The width of the surface area in which data is used to form the fitted surface line. This value should be as large as the surface allows. Make sure the surface width does not go beyond the edge of the region of interest (that is, with **Region** enabled). 
Surface Offset	The distance between the edge region and the surface region.  Setting a small value allows the edge within a tighter region to be detected. However, the measurement repeatability could be affected if the data from the edge are considered as part of the surface region (or vice versa). A rule of thumb is to set **Surface Offset** equal to **Nominal Radius**. Make sure the surface offset does not move the surface width (see above) beyond the end of the region of interest (that is, with **Region** enabled).
Nominal Radius	The radius of the curve edge that the tool uses to locate the edge region. 
Edge Angle	A point on the best fit circle the tools uses to calculate the feature point. The selected point is on the circumference at the specified angle from the start of the

Parameter	Description
	edge region.   The angle is measured from the axis perpendicular to the fitted surface line.
Edge Type	Defines the type of feature point to use for the edge (Corner or Tangent).   A tangent edge point is the point selected based on the defined Edge Angle. A corner edge point is the intersect point between the fitted surface line and an edge line formed by interpolating the points at and after the tangent within the edge region.

Outputs

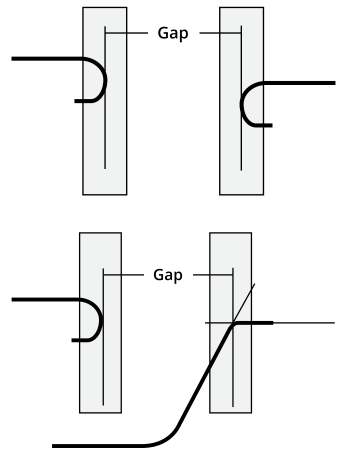
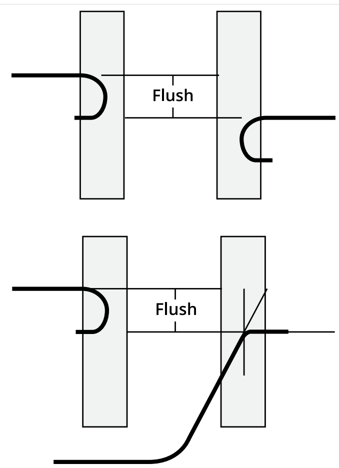
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Gap Measures the distance between two surfaces. The surface edges can be curved or sharp.	
Flush Measures the flushness between two surfaces. The surface edges can be curved or sharp.	
Left Gap X **Left Gap Z** The X and Z position of the edge feature on the left side used to measure the gap.	
Left Flush X **Left Flush Z** Returns the X and Z position of the feature on the left side used to measure flushness.	
Left Angle **Right Angle** The angle of the left and right side surface relative to the X axis.	

Measurement	Illustration
Right Gap X **Right Gap Z** Returns the X and Z position of the edge feature on the right side used to measure the gap.	
Right Flush X **Right Flush Z** Returns the X and Z position of the feature on the right side used to measure flushness.	

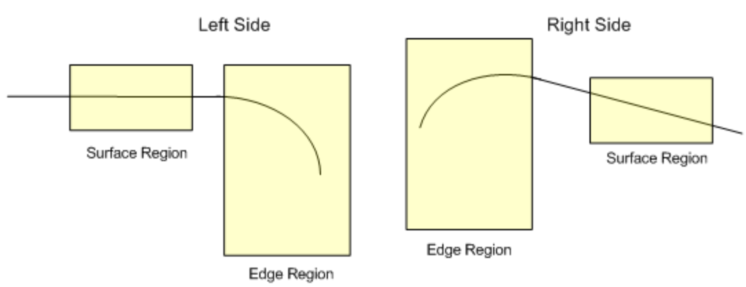
Data

Type	Description
Left Gap Right Gap	Point geometric features representing the point defined by Left Gap X and Left Gap Z, and Right Gap X and Right Gap Z, respectively. Use a side's **Edge Angle** parameter to choose where on the fitted circle the points fall.

Gap and Flush Algorithms

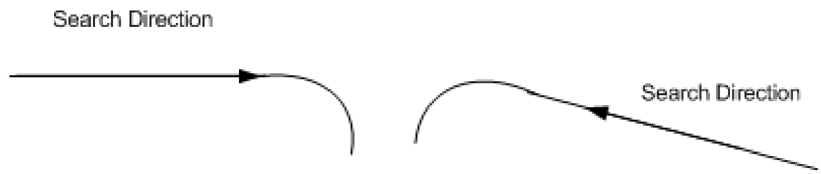
The Panel measurement tool uses the same algorithm to find a feature using either the Gap or the Flush measurement. The Round Corner tool uses the same algorithm, but applies it only to the left or the right; you must choose the side in the tool.

This algorithm first searches for two regions on a side: a surface region and an edge region. (See the tables below for the parameters used by the algorithm.)

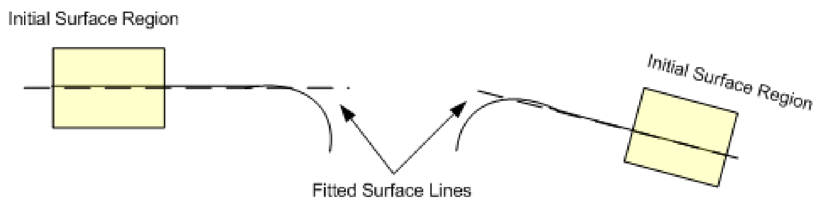


After the algorithm finds the regions, it places a feature point on the surface region based on a set of parameters. You can control the measurement regions, which contain the surface and the edge regions, for the left and the right side. A measurement region also defines the region in which the measurement tool will search for the feature points. Feature points are located on a side using the following algorithm.

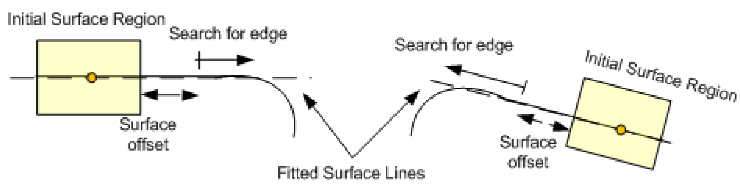
1. On the left side, search from left to right to find a surface region with data that covers at least the value specified in the **Surface Width** setting. For the right side, do the same, searching from right to left.



2. If a surface region is found, fit a line, called the surface line, using the data within the area.

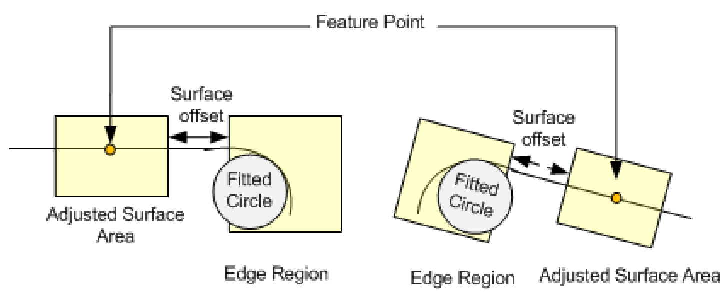


3. Search for a valid edge region that is located at least the distance specified in the **Surface Offset** setting from the end of the surface region. If a surface region is not found, move along the search direction and repeat step 1.



A valid edge region is detected when an edge matches the value in the **Nominal Radius** setting or when the depth exceeds the value in the **Min Depth** setting.

4. If a valid edge region is detected, a model fit is applied to the surface and edge regions to accurately determine the region positions and feature point locations. The model fit takes into account the **Surface Width**, **Surface Offset**, **Edge Angle** and the **Edge Type** parameters.

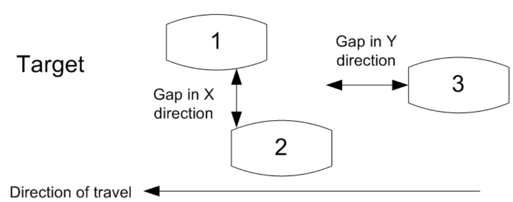


Profile Part Detection

> Profile sensors can produce Surface data for applications where a transport system such as a conveyor continuously feeds discrete parts or material under a sensor. Typically, you will set a sensor to Surface mode on the Scan page, set **Surface generation type** to Continuous, and then configure the part detection parameters on the Scan page. The sensor will then output Surface data for each part it detects.
>
> The Profile Part Detection tool uses the same parameters available with continuous generation Surface mode, but also provides diagnostics that can be useful for troubleshooting. After troubleshooting part detection using this tool, you can copy the settings from the tool to the **Scan** page. The diagnostics are output as measurements by the tool.
>
> Note that the Profile Part Detection has a **Frame of Reference** parameter not available when using continuous Surface generation.
>
> For more information on Surface generation, see *Surface Generation* on page 199.
>
> Remember that when using this tool, the sensor is set to *Profile* mode.

The Profile Part Detection tool takes a series of profiles as input, analyzes the characteristics of each profile based on the configuration of the tool's parameters, and identifies discrete objects and then outputs them as Surface data. Surface measurements can then be performed on each object. This method of surface generation is typically used when a transport system such as a conveyor continuously feeds parts or material under a sensor. The parts or material must have a distinguishable start and stop edge. The sensor continuously generates surfaces of parts that are detected under the sensor.

Multiple parts can pass through the laser or LED light at the same time and will be individually tracked. Parts can be separated along the profile line (X axis), in the direction of travel (Y axis), or by gated external input.



> Gocator also lets you isolate and then measure parts using one of two Surface measurement tools (for more information on these tools, see *Surface Blob* on page 455 and *Surface Segmentation* on page 635).

Part detection can be performed when **Source** on the **Acquire** > **Scan** page in the **Trigger** panel is set to **Time** or **Encoder**. To use the **Time** trigger source, the travel speed must be calibrated. To use the **Encoder** trigger source, the encoder resolution must be calibrated.

> The Profile Part Detection tool is only available when **Enable uniform spacing (resampling)** is checked: the tool does not support point cloud profile data.

In addition to a Part Surface output, the tool also returns the Point geometric feature of center of the detected part, and several measurement tool that you can use as diagnostics (for example, to see the number of parts accepted, the number of parts rejected and why, the part's length, and so on). For a complete list of the diagnostics, see below.

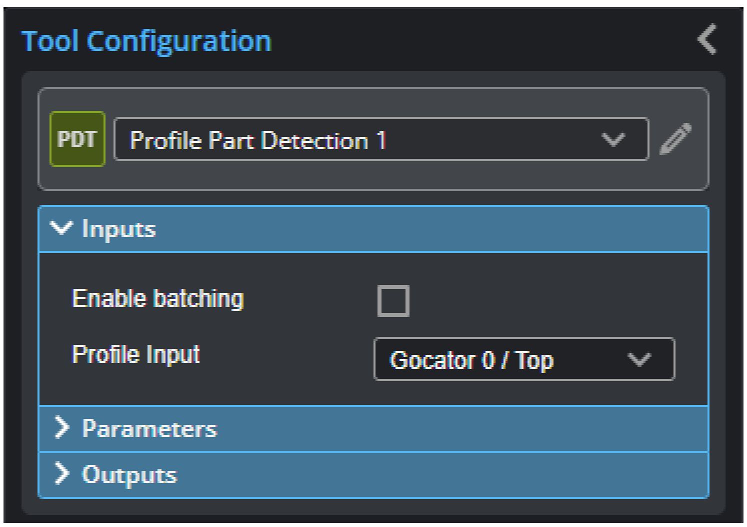
When working with previously recorded Profile scan data, you can set the FPS replay setting to a value high enough that parts are displayed in a reasonable amount of time. You can also optionally enable the Continuous Loop to loop through parts continuously.

🗋	In order to see a detected part's Surface data when the Profile Part Detection tool is selected in the list of added tools, the Part Surface must be selected in the list of outputs or pinned below the data viewer.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

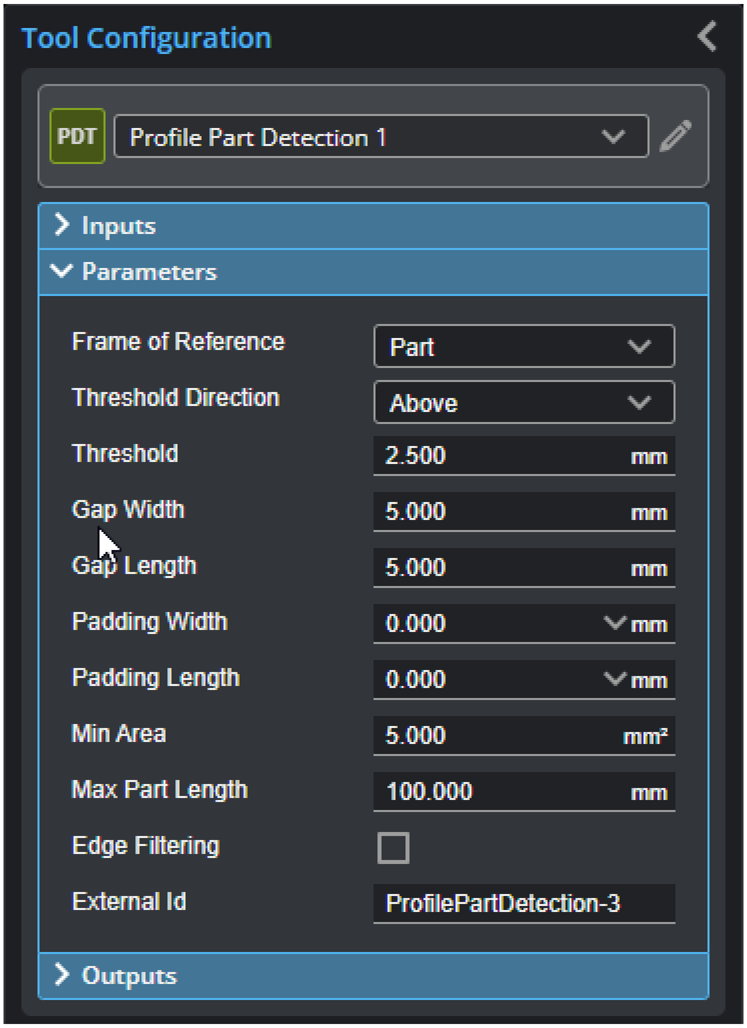


Inputs

Name	Description
Profile Input	The source of profiles for the tool.

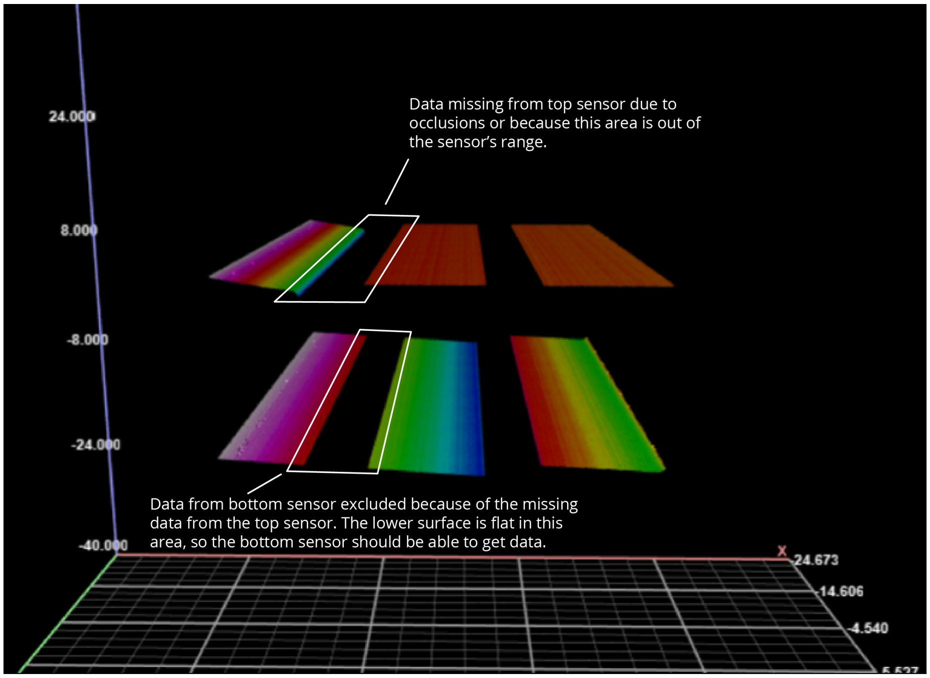
Parameters

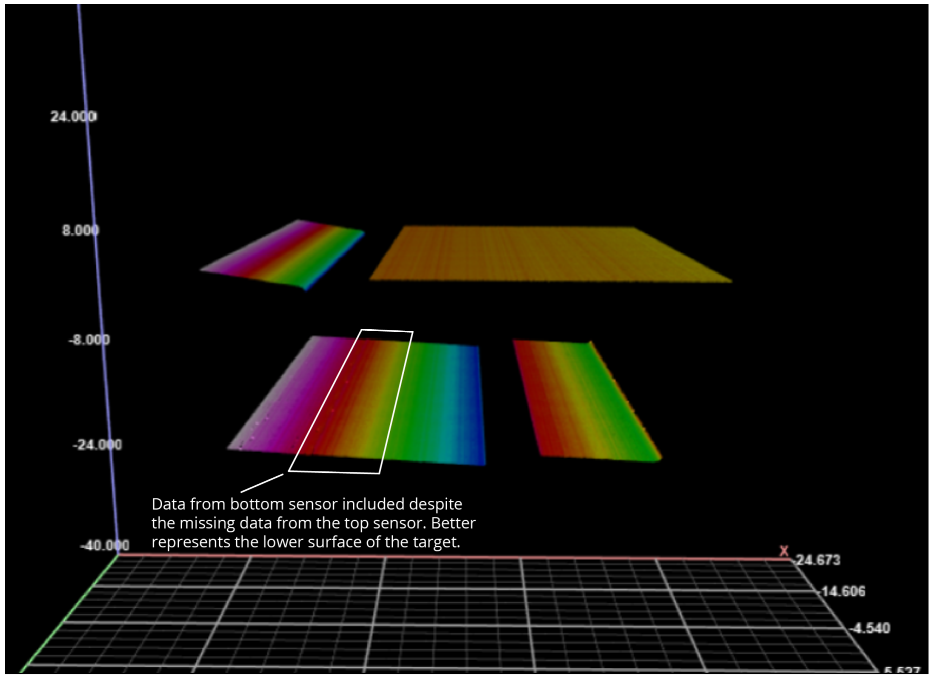
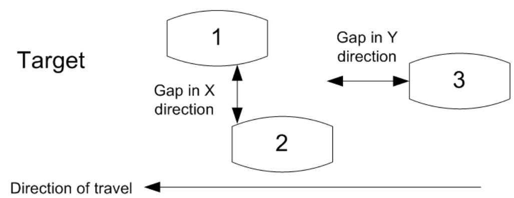
You configure the tool's parameters in the expandable **Parameters** section.

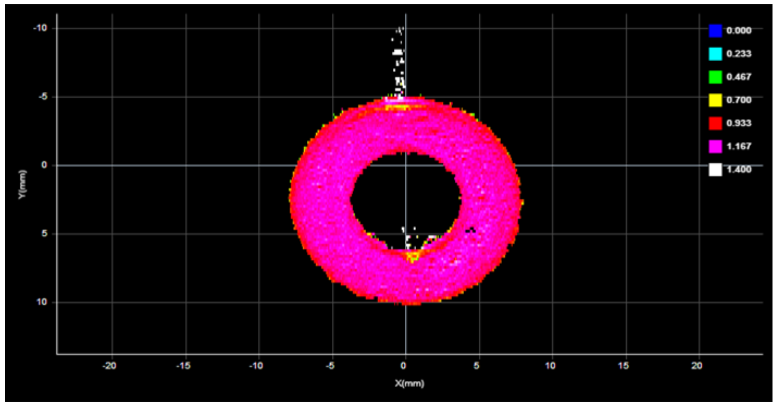
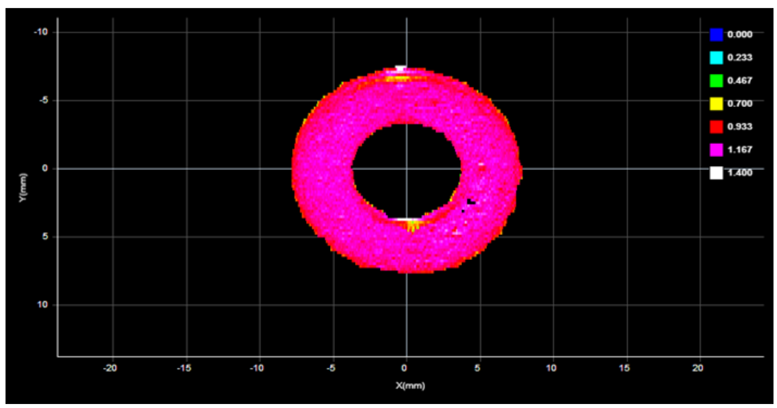
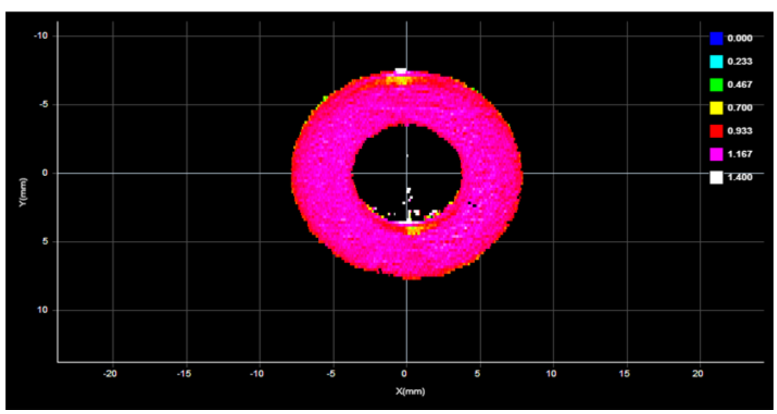


Parameters

Parameter	Description
Height threshold	Determines the height threshold for part detection. The setting for **Threshold Type** (see above) determines if parts should be detected above or below the value in **Threshold**. Above is typically used to prevent the belt surface from being detected as a part when scanning objects on a conveyor. In an Opposite layout, the threshold is applied to the difference between the top and the bottom profile. A target thinner than the threshold value is ignored, including places where only one of either top or bottom is detected. To separate parts by gated external input, set **Threshold** to the active area Z offset (that is, minimum Z position of the current active area). In the Trigger panel, set **Source** to **Time** or **Encoder**, and check the **Gate on External Input** checkbox in the **Trigger** panel; for more information, see *Triggers* on page 206.
Threshold direction	Determines if parts should be detected above or below the height threshold.
Include one-sided data	This option is only displayed with dual-sensor systems in Opposite layout, or multi-

Parameter	Description
	sensor systems in Grid layout with at least one sensor in the Bottom row. When the option is *disabled*, data points from a sensor are excluded if the points directly opposite from the other sensor are missing (due to occlusions, drop-outs, and so on). When the option is *enabled*, data points are included even if data points from the other sensor are missing. The following image shows surface data from a dual-sensor system in which the sensors are mounted facing each other. In this case, **Include one-sided data** is disabled.  The data on the upper left is missing, due to the shape of the target: getting data from this area is difficult or impossible, due to occlusions or simply because this part of the upper surface is beyond the top sensor's measurement range. Data is missing on the left of the lower surface, even though the target is flat in this area. In the following image, **Include one-sided data** is enabled. The result is that data from the lower left is included in the scan data, better representing the actual target. (The same situation is occurring on the right side of the surfaces.)

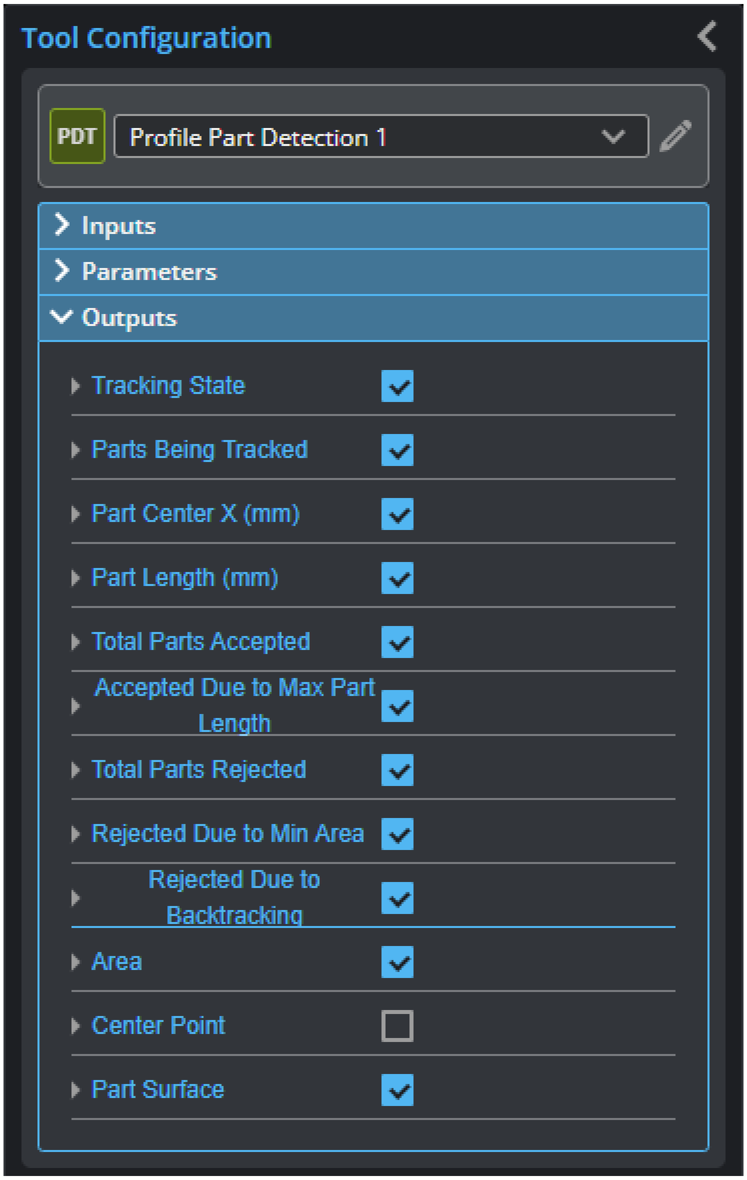
Parameter	Description
	 In general, you should leave this setting enabled.
Gap Width Gap Length	**Gap Width** and **Gap Length** determine the minimum separation between objects on the X and the Y axis, respectively. If parts are closer than the gap interval, they will be merged into a single Surface output. 
Padding Width Padding Length	These parameters are useful when processing part data with third-party software such as HexSight, Halcon, etc. **Padding Width** and **Padding Length** control the amount of additional scan data output in the X and Y directions, respectively. The padding can contain data points that were outside the height threshold and excluded from the initial part detection.
Min Area	Determines the minimum area for a detected part. Set this value to a reasonable minimum in order to filter out small objects or noise.
Max Length	Determines the maximum length of the part object. When the object exceeds the maximum length, it is automatically separated into two parts. This is useful to break a long object into multiple sections and perform measurements on each section.
Edge Filter	When **Edge Filter** is enabled, an **Edge Filter** expanding section is displayed that contains the settings described below.

Parameter	Description
	Part scans sometimes contain noise around the edges of the target. This noise is usually caused by the sensor's light being reflected off almost vertical sides, rounded corners, etc. Edge filtering helps reduce edge noise in order to produce more accurate and repeatable volume and area measurements, as well as to improve positioning of relative measurement regions.  *Edge Filter disabled (scan shows reflection noise)*  *Edge Filter enabled (reflection noise eliminated or reduced)*  *Edge Filter enabled, Keep Interior enabled*
Keep Interior	The **Keep Interior** setting limits filtering to the outside edges of the target.

Parameter	Description
Edge Width Edge Length	The **Edge Width** and **Edge Length** settings represent the size of the filter on the X axis and the Y axis, respectively.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

One of the most common issues when setting up part detection is that surface data is not generated after scanning the target or targets. The Profile Part Detection tool's measurements provide information on the status of the part detection engine. You can use these measurements to diagnose why the tool is not detecting parts.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Tracking State Part detection state for largest currently tracking part. The possible values returned by are the following: 0: Not In Part 1: In Part, Min area not achieved 2: In Part, Min area achieved 3: In Gap, Min area not achieved

Measurement
4: In Gap, Min area achieved
Parts Being Tracked The number of parts the engine is currently tracking.
Part Center X The center of the partial part, midway between the minimum X and maximum X detected for the part.
Part Length The length of the part. In cases of backtracking, the number decreases.
Total Parts Accepted The number of parts that meet the part detection criteria (based on the configuration of the tool's parameters).
Due to Max Part Length The number of parts accepted because they have reached the value in the **Max Part Length** parameter (see above). If too many parts are being accepted, increase **Max Part Length**.
Total Parts Rejected The number of parts that fail to meet the part detection criteria.
Rejected Due to Min Area The number of parts rejected because they are below the value in the **Min Area** parameter (see above). If too many parts are being rejected, reduce **Min Area**.
Rejected Due to Backtracking The number of parts rejected due to backtracking, for example, when the user reverses the direction of the transport mechanism while the sensor is actively scanning a part. Only applicable when the encoder trigger behavior has been set to Bi-Directional; for more information on trigger behaviour, see *Triggers* on page 206.
Area The area of the current part.

Features

Type	Description
Center Point	The center point of the detected part.

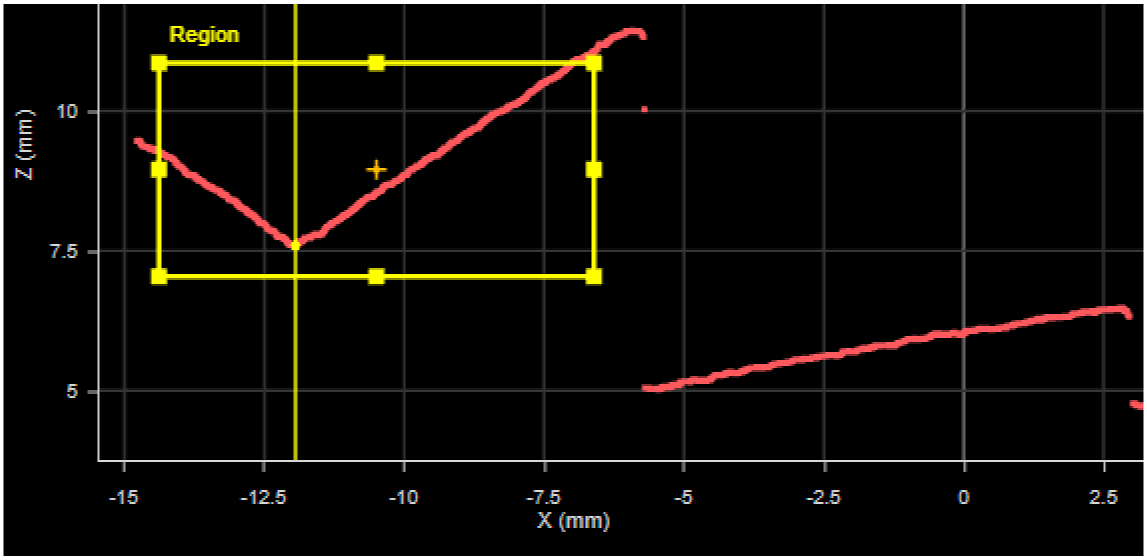
For more information on geometric features, see *Geometric Features* on page 262.

Data

Type	Description
Part Surface	The Surface data of the detected part.

Profile Position

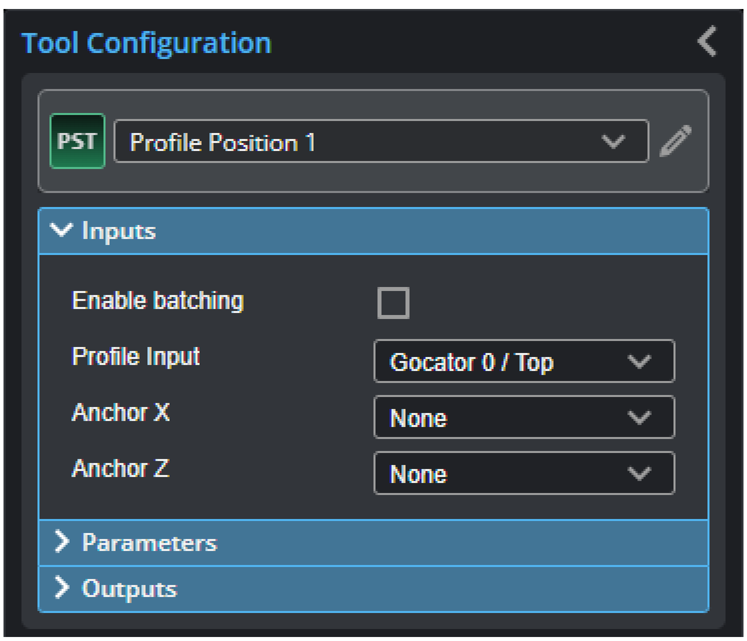
The Position tool finds the X or Z axis position of a feature point. The feature type must be specified (see below).



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



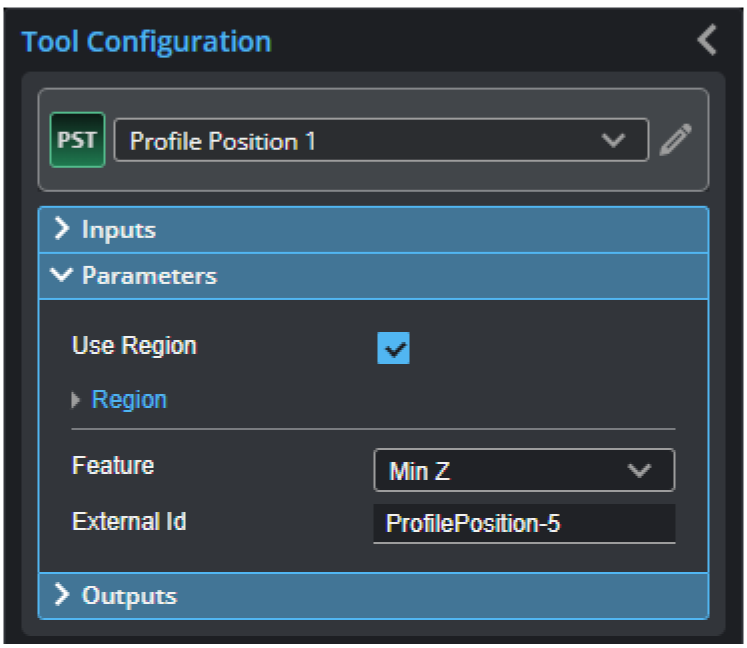
	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. If **Enable Batching** is *disabled* and the passed array contains more than two elements, GoPxL displays an error. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



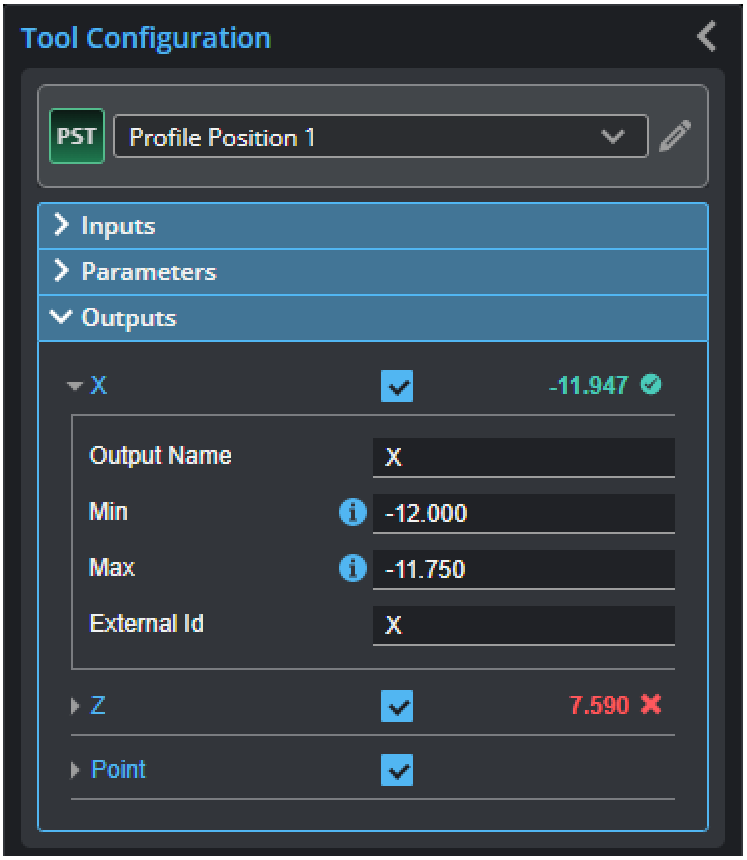
Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.

Parameter	Description
Feature	The feature the tool uses for its measurements. One of the following: • Max Z • Min Z • Max X • Min X • Corner • Average • Rising Edge • Falling Edge • Any Edge • Top Corner • Bottom Corner • Left Corner • Right Corner • Median For more information on feature points, see *Feature Points* on page 259. To set the region of a feature, adjust it graphically in the data viewer, or expand the feature in the **Parameters** section and enter the values in the fields. For more information on regions, see *Regions* on page 250.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

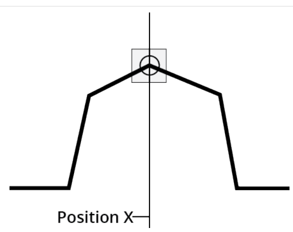
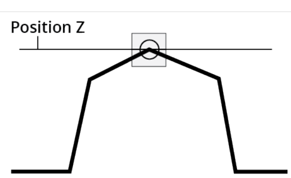
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Finds the position of a feature on the X axis.	
Z Finds the position of a feature on the Z axis.	

Features

Type	Description
Point	The returned position.

For more information on geometric features, see *Geometric Features* on page 262.

Profile Roughness

The Profile Roughness tool calculates multiple measurements of profile roughness according to different industry standards, including the following:

ISO 21920

ISO 4287

ASME B46.1

JIS B0601

ISO 13565

ISO 12085

VDA 2006

VDA 2007

Parameters and measurements differ with standards. Choose the standard appropriate for your application.

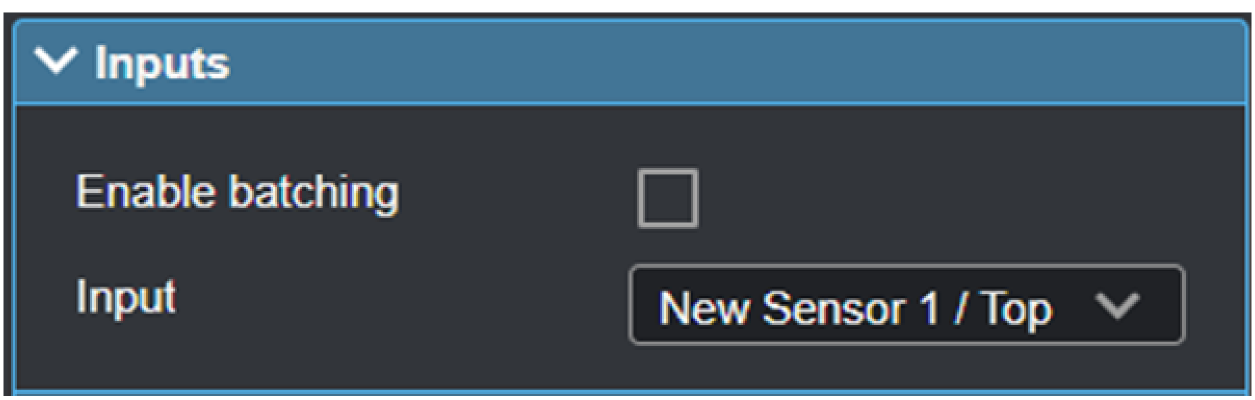
> There cannot be a large invalid point range for Profile roughness. Use the region to restrict the valid range in the tool for the calculation.

> Because the accuracy of a roughness measurement is limited by the X resolution of a sensor, this tool is typically only used on confocal sensors, whose high resolution is better suited for roughness applications.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

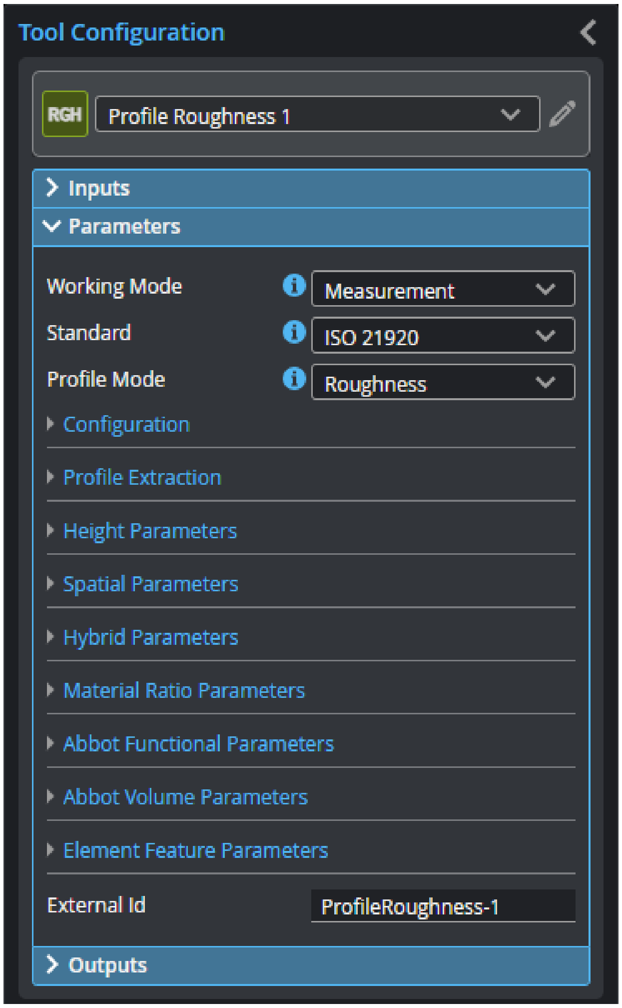
Inputs

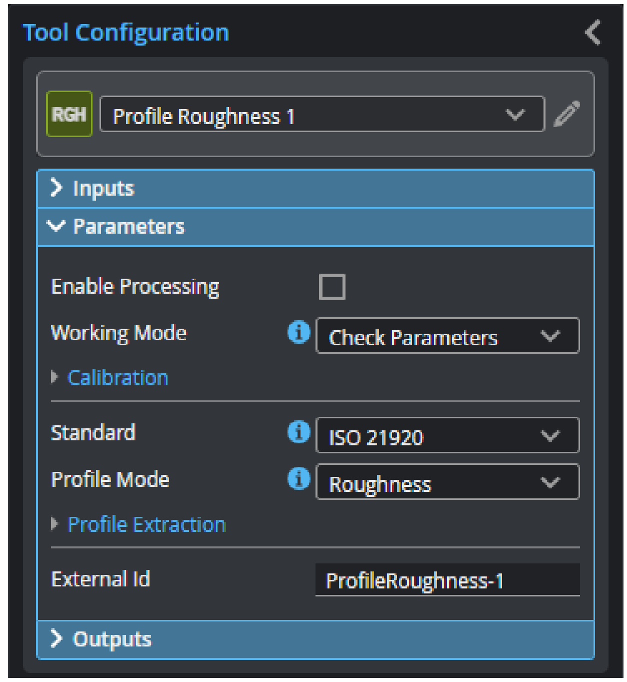
Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes.

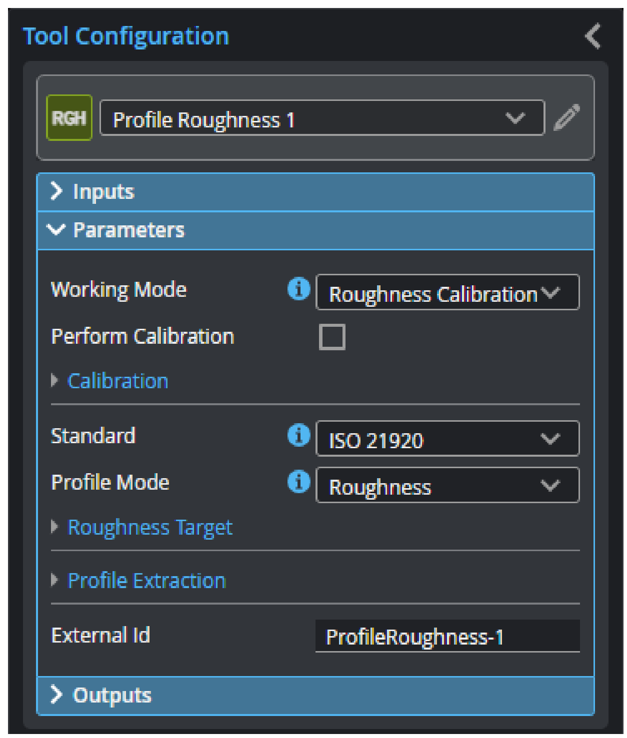
Parameters

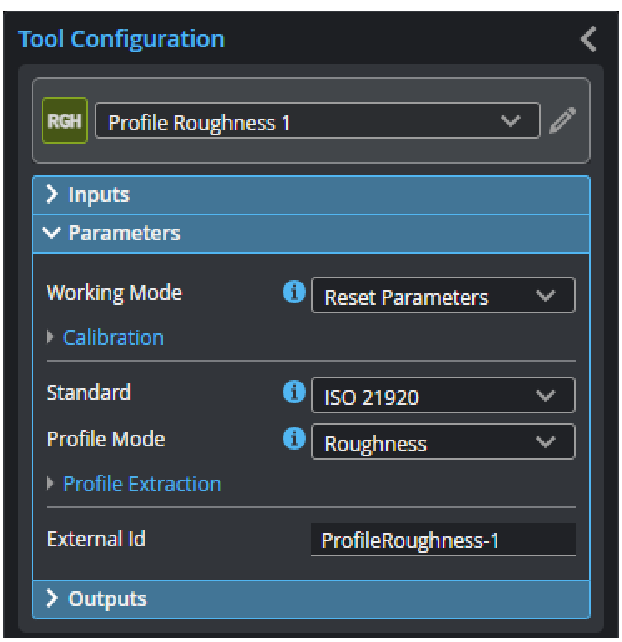
You configure the tool's parameters in the expandable **Parameters** section.

Note that the tool displays different sets of parameters based on what you set **Working Mode** to.


Tool Configuration
RGH
Profile Roughness 1
Inputs
Parameters
Working Mode
Measurement
Standard
ISO 21920
Profile Mode
Roughness
Configuration
Profile Extraction
Height Parameters
Spatial Parameters
Hybrid Parameters
Material Ratio Parameters
Abbot Functional Parameters
Abbot Volume Parameters
Element Feature Parameters
External Id
ProfileRoughness-1
Outputs

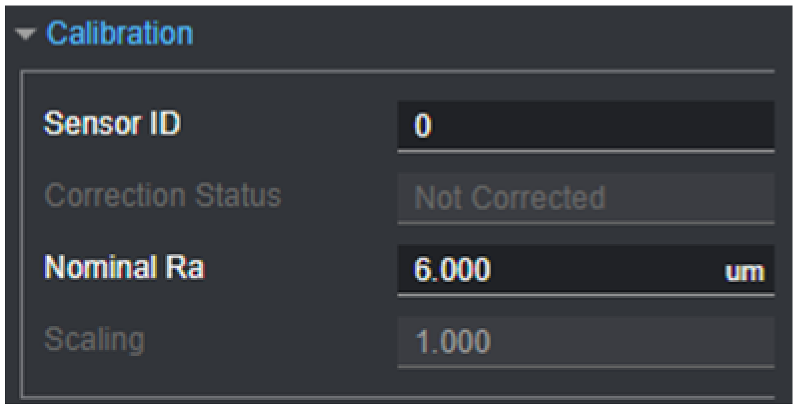

Tool Configuration
RGH
Profile Roughness 1
Inputs
Parameters
Enable Processing
Working Mode
Check Parameters
Calibration
Standard
ISO 21920
Profile Mode
Roughness
Profile Extraction
External Id
ProfileRoughness-1
Outputs

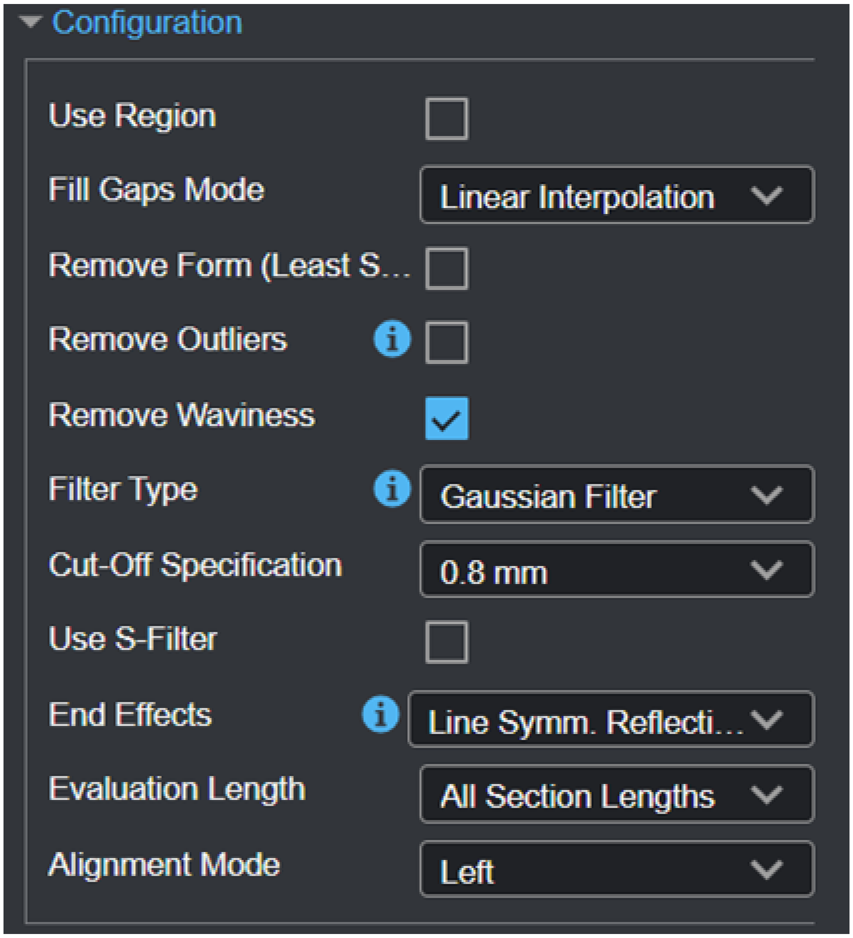

Tool Configuration
RGH
Profile Roughness 1
Inputs
Parameters
Working Mode
Roughness Calibration
Perform Calibration
Calibration
Standard
ISO 21920
Profile Mode
Roughness
Roughness Target
Profile Extraction
External Id
ProfileRoughness-1
Outputs


Tool Configuration
RGH
Profile Roughness 1
Inputs
Parameters
Working Mode
Reset Parameters
Calibration
Standard
ISO 21920
Profile Mode
Roughness
Profile Extraction
External Id
ProfileRoughness-1
Outputs

Parameters

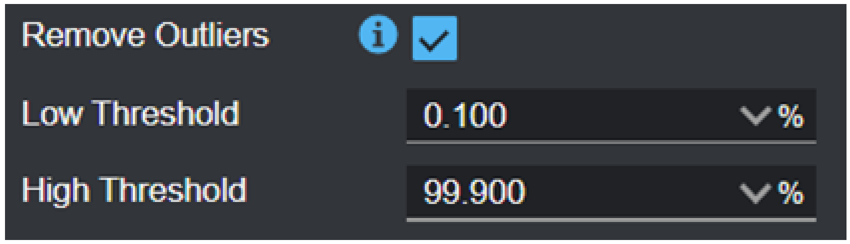
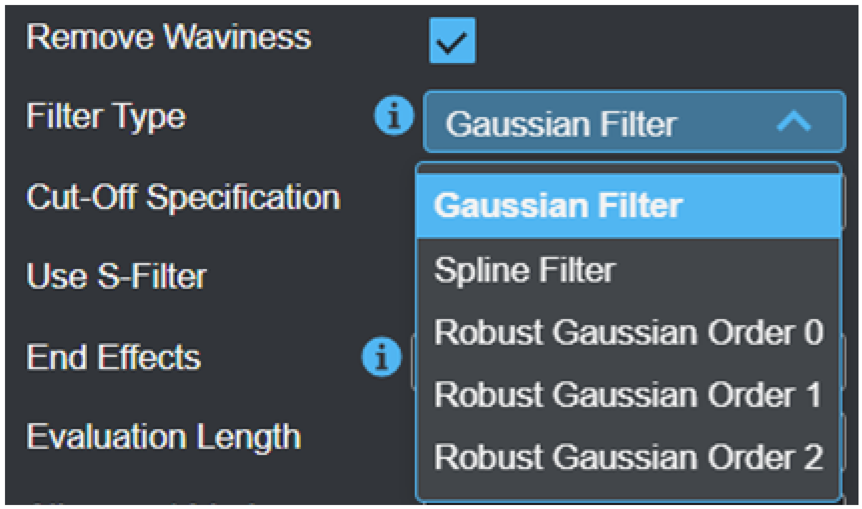
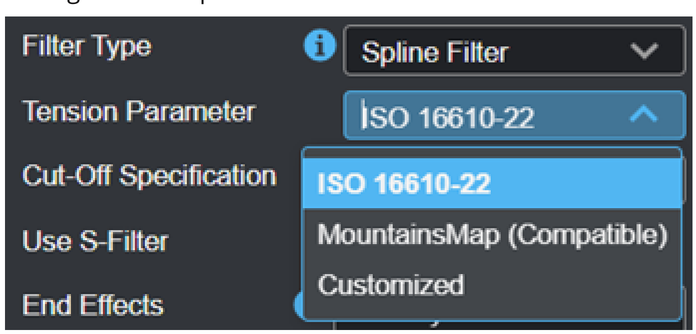
Parameter	Description
Enable Processing	The calculation starts only when this is checked. Only displayed when **Working Mode** is set to Check Parameters or Roughness Calibration. (Processing happens automatically in other modes.)
Working Mode	Support the following four working modes: • Measurement • Check Parameters • Reset Parameters • Roughness Calibration Before starting the calibration, make sure to remove the old calibration.
Perform Calibration	Performs the calibration. Only displayed when **Working Mode** is set to Roughness Calibration.
Standard	The standard the tool uses. Some standards change how options in **Evaluation Length** behave. For some standards, you should follow the recommendations below. **ISO 21920** Set **All Section Lengths** and **All Sections (no Averaging)**. These parameters behave the same with this standard. **ISO 4287** Typically, you will set **Evaluation Length** to "n Section Lengths", and set **Section Count** to 5. You can also choose "All Section Length" with this standard. **ASME B46.1** Typically, you will set **Evaluation Length** to "n Section Lengths", and set **Section Count** to 5. If you use a different **Evaluation Length** option, note that "All Section Lengths" and "All Sections (no Averaging)" are treated the same with this standard. **JIS B0601** According to the original definition, “All Section Lengths” should be used. **ISO 13565** “Total Profile Length” should be used. **ISO 12085** The same as definitions in ISO 4287, or using “Total Profile Length”.

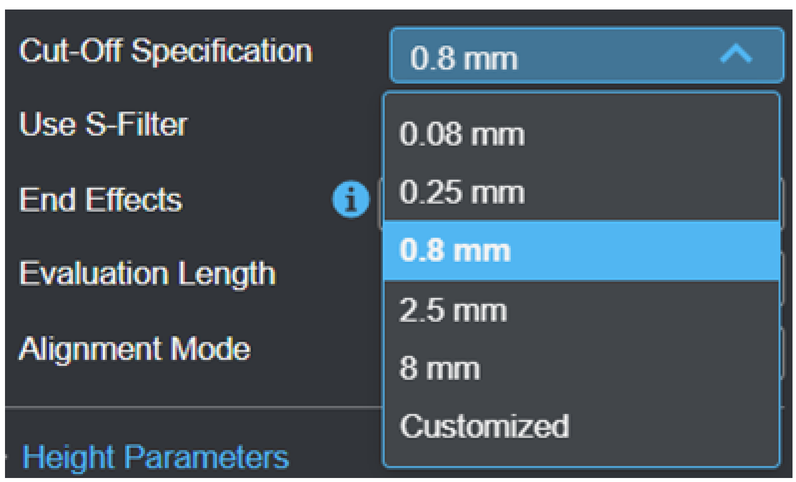
Parameter	Description
	VDA 2006 “All Section Lengths” and "All Sections (no Averaging)" are the same. **VDA 2007** “Total Profile Length” should be used.
Profile Mode	In the standards, many parameters are also defined in relation to Primary or Waviness Profile. Select the profile mode after setting a standard.
Configuration	Shows when **Working Mode** is set to "Measurement". See *Configuration* on the next page for details.
Calibration	Shows when **Working Mode** set to any mode other than "Measurement” 
Roughness Target	Shows when **Working Mode** is “Roughness Calibration”. The parameters are the same as the **Configuration** group.
Height Parameters	Shows different parameters depending on what you select in **Standard**. See *Height Parameters* on page 397.
Spatial Parameters	See *Spatial Parameters* on page 399 for details.
Hybrid Parameters	See *Hybrid Parameters* on page 399 for details.
Material Ratio Parameters	See *Material Ratio Parameters* on page 400 for details.
Abbot Functional Parameters	See *Abbot Functional Parameters* on page 400 for details.
Abbot Volume Parameters	See *Profile Roughness* on page 387 for details.
Element Feature Parameters	See *Element Feature Parameters* on page 402 for details.
Motif Parameters	See *Motif Parameters* on page 403 for details.

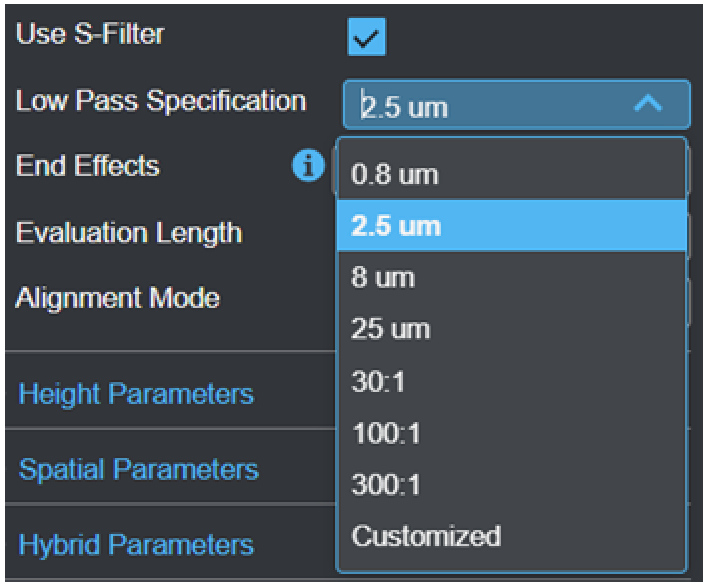
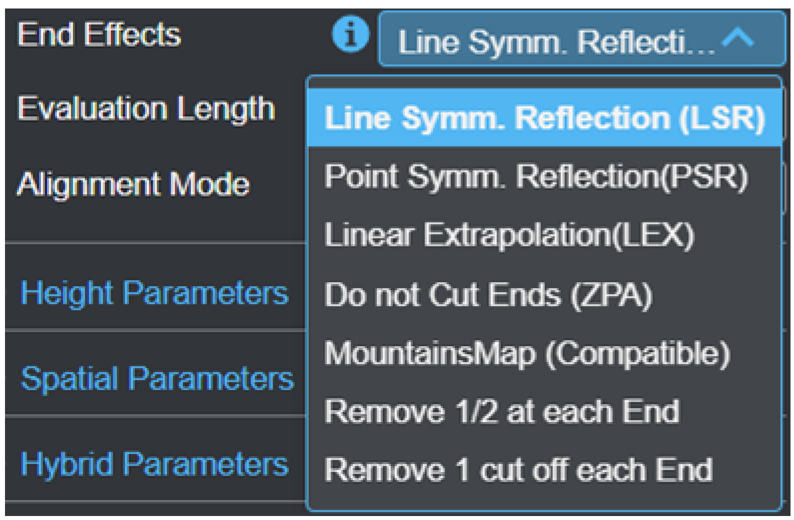


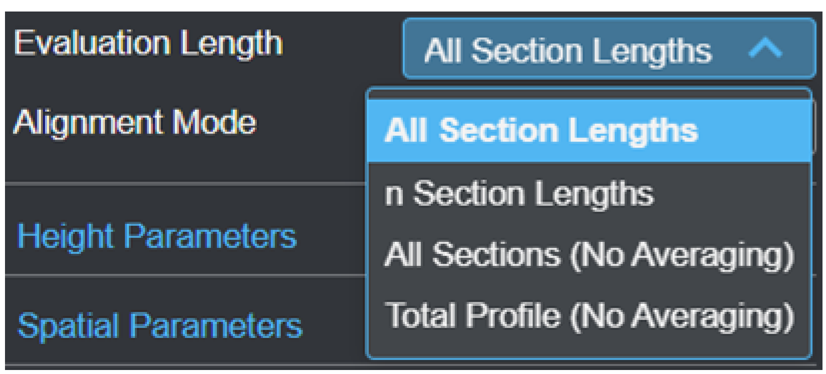
Configuration

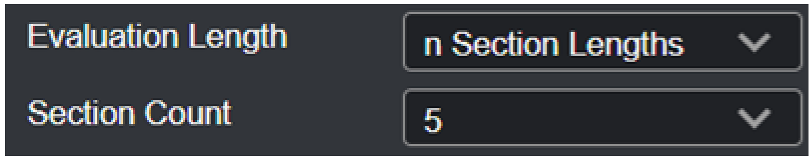
Parameter	Description
Use Region Region	Check to set region size.
Fill Gaps Mode	Choose **Linear Interpolation** to fill gaps in the profile but keep the profile as natural as possible.
Remove Form (Least Squares)	When checked, displays a drop-down containing one of the following: • Linear Alignment • Polynomial Subtraction: Show "Polynomial Order (<21)" for edit if selected • Circular Unwrapping • Circle With Nominal Radius: Show "Nominal Radius" for edit if selected.
Remove Outliers	Check to remove outliers from the profile and set Low Threshold and High Threshold. The outliers are defined according to the Sigma Rule in terms of the given percentage multiplied by 1.5.

Parameter	Description
	
Remove Waviness	Check to remove the waviness of the profile. Select the filter in **Filter Type**.  The tool provides the following filters according to the ISO 16610 standards series, which allow different filter kernels having different shapes. Choose the filter depending on your application and its sensitivity to the curvature of the waviness. **Gaussian Filter**: According to ISO 16610-21, ISO 16610-28 If selected, set choose the appropriate option in **End Effects** (see *End Effects* on page 395). **Spline Filter**: According to ISO 16610-22, ISO 16610-28. If selected, set **Tension Parameter** to control how tightly the spline curve fits through the data points. 

Parameter	Description
	Robust Gaussian Order 0: According to ISO 16610-31, ISO 16610-28. Using Savitzky-Golay filter coefficients: $\alpha = \sqrt{ln(2)/\pi}$ Retaining the zeros moment means to reduce end effects. **Robust Gaussian Order 1**: According to ISO 16610-31, ISO 16610-28. Using Savitzky-Golay filter coefficients: $\alpha = \sqrt{ln(2)/\pi}$ Retaining the first moment means to reduce end effects. **Robust Gaussian Order 2**: According to ISO 16610-31, ISO 16610-28. Using Savitzky-Golay filter coefficients $\gamma = \sqrt{(-1 - W(-1/(2e))/\pi}$  The cut-off is selected depending on the workpiece surface either according to the valley spacing, or the expected roughness values. At the same time the total evaluation length and the corresponding total profile length are defined according to standards. Deviations are necessary if the workpiece does not allow the required total profile length. If the actual possible total profile length on the workpiece surface is not enough for lt, the number of sampling lengths is reduced accordingly and specified in the drawing. If the actual available total profile length is less than a sampling length, the total height of profile Pt of the primary profile is evaluated instead of Rt or Rz.
Use S-Filter	Check to use a low pass filter with the cut-off wavelength defined in **Low Pass**

Parameter	Description
	Specification. 
End Effects	Shows when **Remove Waviness** is checked. The end effect is the unintended changes in filtration response in the both end sections of an open profile, and the end effect correction methods are according to standard ISO 16610-28.  **Line Symm. Reflection (LSR)** A measured profile is extended by horizontal reflection on the left hand and right hand, respectively. **Point Symm. Reflection (PSR)** A measured profile is extended by horizontal reflection in conjunction with vertical

Parameter	Description
	reflection on the left and right hand respectively. Both reflection lines shall intersect at the respective end point of the profile. **Linear Extrapolation (LEX)** In the case of linear extrapolation, a least-squares line is fitted to the profile within the left and right end effect regions. **Do not Cut Ends (ZPA)** Zero padding is a simple method for retaining the total length after filtering the profile. Profile z(x) is padded with zeros over the total length. **MountainsMap (Compatible)** Tries to make filtering compatible with MMP "Manage End Effect". This setting only applies to Guassian filters. **Remove 1/2 at each end** No end effects treatment required as the critical regions are not involved in the calculation. **Remove 1 cut off each end** No end effects treatment required as the critical regions are not involved in the calculation.
Evaluation Length	The evaluation length. One of the following:  **All Section Lengths** The maximum available sections will be used. **n section lengths** Set the number of sections in **Section Count**. The effective section number is always limited by the available length.

Parameter	Description
	 **All Sections (no Averaging)** All parameters as long as the definition allows are calculated in the entire evaluation length. **Total Profile (no averaging)** All parameters as long as the definition allows are calculated in the total profile length.
Alignment Mode	 Note: The alignment in MountainsMap is left-justified.

Height Parameters

Parameter	Description	Standard
Ra	**Roughness average** Ra is the arithmetic average of the absolute values of the roughness profile ordinates, in µm.	ISO 21920 ISO 4287 ASME B46.1 JIS B0601 VDA 2006 ISO 13565-2
Rq	**RMS roughness** Rq is the root mean square average of the roughness profile ordinates, in µm.	ISO 4287 ISO 21920 JIS B0601
Rz	**Mean roughness depth** Rz is the arithmetic mean value of the single roughness depths Rzi of consecutive sampling lengths, in µm.	ISO 21920 JIS B0601
Rz1max	**Maximum height** Rz1max is the greatest Rz value from the n sampling lengths lr, in µm.	ISO 4287

Parameter	Description	Standard
Rz(JIS)	**The ten point height** RzJIS is defined on each sampling length as being the sum of the 5 highest peaks and the 5 deepest holes, divided by 5. The result is expressed in the length unit of the Z-axis, in µm.	JIS B0601
Rz(n)	Maximum height inside the nth section length, in µm.	ISO 4287 JIS B0601
Rmax	**Maximum roughness depth** Rmax is the largest single roughness depth within the evaluation length, in µm.	
Rt	**Total height of profile** Rt is the distance between the highest peak and the deepest valley of the profile of the total evaluation length ln, in µm.	ISO 21920 ISO 4287 JIS B0601
Rsk	**Skewness** Rsk is a measure of the asymmetry of the amplitude density curve. A negative skewness value indicates a surface with good bearing properties. (No unit.)	ISO 21920 ISO 4287 JIS B0601
Rku	**Kurtosis** Rku is a measure of the peakedness of the amplitude density curve. (No unit.)	ISO 21920 ISO 4287 JIS B0601
Rp Rpm	Rp is the height of the highest profile peak of the roughness profile within one sampling length. According to ASME, the Rp mean value (average calculated over the evaluation length) is called Rpm. Both in µm.	ISO 21920
Rv	Rv is the depth of the deepest profile valley of the roughness profile within one sampling length, in µm.	ISO 21920
Rz	The sum of Rp + Rv is the single roughness depth Rzi, in µm.	JIS B0601
Rzx	**Maximum height** (peak-to-valley), within a section length, in µm.	ISO 21920
Rc	**Mean height of profile elements** A horizontal discrimination of 1% of the sampling length and a vertical discrimination of 10% of Rz are applied to avoid taking into account segments with outlying values due to noise. In µm.	ISO 4287 JIS B0601
Rp1max	**Maximum local profile peak height** Rp1max is the Maximum of Rpi values which are calculated within a sampling length, in µm.	ISO 4287 JIS B0601
Rv1max	**Maximum local profile valley depth** Rv1max is the Maximum of Rvi values which are calculated within a sampling length, in µm.	ISO 4287 JIS B0601
Rz1max	**Maximum local height of the profile**	

Parameter	Description	Standard
	Rz1max is the Maximum of Rzi values which are calculated within a sampling length, in µm.	
Rpt	Maximum hill height, in µm.	ISO 21920
Rvt	Maximum dale depth, in µm.	ISO 21920

Spatial Parameters

Parameter	Description	Standard
Ral	Autocorrelation length Ral is the horizontal distance of the autocorrelation function (tx) which has the fastest decay to a specified value s, with 0 < s < 1. The default value for s in the software is 0.2. This parameter expresses the content in wavelength of the surface. A high value indicates that the surface has mainly high wavelengths (low frequencies). In mm.	ISO 21920
Rsw	Dominant spatial wavelength Rsw is the wavelength corresponding to the largest value of the FFT. In mm.	ISO 21920
WDsm	Mean horizontal values determined from the amplitude spectrum Size of the profile elements (mean period length of the dominant waviness). In mm.	VDA 2007
WDc	Average of the heights of the profile elements within the evaluation length, in µm.	VDA 2007
WDt	Vertical difference between the highest and lowest point of the WD profile within the evaluation length, in µm.	VDA 2007

Hybrid Parameters

Parameter	Description	Standard
Rdq	**Root mean square gradient** Rdq is the root mean square value of the ordinate slopes dZ/dX, within the sampling length, in degrees.	ISO 21920
Rda	Arithmetic mean absolute gradient, in degrees.	ISO 21920
Rdt	Maximum absolute gradient, in degrees.	ISO 21920
Rdl	**Developed length** Rdl is the Length measured following the profile curve, in mm.	ISO 21920
Rdr	**Developed length ratio** Rdr is the Ratio between the length measured following the profile curve ("real profile"), and the profile length, expressed as a percentage in excess of 100%.	ISO 21920

Material Ratio Parameters

Parameter	Description	Standard
Rmr	**Material ratio** Rmr indicates what ratio the totaled length in the material has assumed relative to the evaluation length (in %). The comparison is made in the specified section height c and the total evaluation length ln. The material ratio curve indicates the material ratio as a function of the section height.	ISO 21920 ISO 4287 JIS B0601
Rmr (Rz/4)	**Automatic relative material ratio** Rmr (Rz/4) is the Material ratio calculated at cut level c = 0.25 Rz, relative to a specified reference level, as a percentage.	ISO 4287 JIS B0601
Rmc	**Inverse material ratio** Rmc is the height distance between highest level and the cut level c corresponding to a given material ratio, in µm.	ISO 21920
Rdc	**Profile section height difference** Rdc is the Profile section height difference or bearing height. Gives the vertical distance between two cut levels given by their bearing ratio, in µm.	ISO 4287 ISO 21920 JIS B0601

Abbot Functional Parameters

Parameter	Description	Standard
Rk	**Core roughness height** Rk is the effective roughness of the profile. It is the height of the roughness profile, without taking into account the elevated hills or the very deep dales, in µm.	ISO 13565-2 ISO 21920
Mr1	**Upper material ratio** Mr1 is the smallest material ratio at the limits of the roughness core area, as a percentage.	ISO 13565-2 ISO 21920
Mr2	**Lower material ratio** Mr2 is the highest material ratio at the limits of the roughness core area, as a percentage.	ISO 13565-2 ISO 21920
Rpk	**Reduced peak height** Rpk is the mean height of the peaks protruding from the roughness core profile, in µm.	ISO 13565-2 ISO 21920
Rpkx Rpk*	**Maximum peak height** Rpkx is the roughness height of the hills, before area correction, in µm.	ISO 13565-2 ISO 21920
Rvk	**Reduced valley depth** Rvk is the mean depth of the valleys protruding from the roughness core profile, in µm.	ISO 13565-2 ISO 21920

Parameter	Description	Standard
Rvkx Rvk*	**Maximum dale depth** Rvkx is the roughness depth of the dales, before area correction, in µm.	ISO 13565-2 ISO 21920
A1	**Upper area** A1 is calculated as the area of the triangle equivalent to the peaks, $\mu m^2/mm$.	ISO 13565-2
A2	**Lower area** A2 is calculated as the area of the triangle equivalent to the holes, $\mu m^2/mm$.	ISO 13565-2
CV	**Crevice volume** Equivalent to A2, but expressed in specific units ($\mu m^3/\mu m^2$). Note: 10 $\mu m^3/\mu m^2$ = 1 mm^3/cm^2, so the CV value is 10 times the A2 value.	
APH	**Peak height** APH is the maximum height of the protruding peaks above the roughness core profile, in µm.	ISO 13565-2
AVH	**Valley depth** AVH is the maximum depth of the profile valleys below the roughness core profile, in µm.	ISO 13565-2
Rvk/Rk	**Valley-core depth ratio** Rvk/Rk is the ratio of the reduced valley depth Rvk over the core roughness depth Rk. (No unit.)	ISO 13565-2
Rvq	**Valley root-mean square roughness** Rvq is the slope of the linear regression calculated on the dale zone, in µm.	ISO 13565-3 ISO 21920
Rpq	**Plateau root-mean square roughness** Rpq is the slope of the linear regression calculated on the hill zone, in µm.	ISO 13565-3 ISO 21920
Rmq	**Material ratio** Rmq is the Intersection of both regression lines. Gives information about the depth of the intersection between fine surface finish (plateau zone) and rough surface finish (valley zone), as a percentage.	ISO 13565-3 ISO 21920

Abbot Volume Parameters

Parameter	Description	Standard
Rvm	**Material volume** Rvm is the volume of material at a material ratio p (as a percentage), in $\mu m^3/mm^2$.	ISO 21920
Rvv	**Void volume**	ISO 21920

Parameter	Description	Standard
	Rvv is the volume of voids at a material ratio p (as a percentage), in $\mu m^3/mm^2$.	
Rvmp	**Hill material volume** Rvmp is the volume of material in the hills, between 0% material ratio and a material ratio p (as a percentage), calculated in the zone above c1, in $\mu m^3/mm^2$.	ISO 21920
Rvmc	**Core material volume** Rvmc is the volume of material in the core, between two material ratios p and q (as a percentage), calculated in the zone between c1 and c2, in $\mu m^3/mm^2$.	ISO 21920
Rvvc	**Core void volume** Rvvc is the volume of void in the core, between two material ratios p and q (as a percentage), calculated in the zone between c1 and c2, in $\mu m^3/mm^2$.	ISO 21920
Rvvv	**Dale void volume** Rvvv is the volume of void in the dales, between a material ratio p (as a percentage) and 100% material ratio, calculated in the zone below c2, in $\mu m^3/mm^2$.	ISO 21920

Element Feature Parameters

Parameter	Description	Standard
Rsm	**Mean width of the profile elements** Rsm gives information about the spacing of motifs (length of the period), in mm.	ISO 4287 ISO 21920 JIS B0601
Rsmx	Maximum profile element width, in mm.	ISO 21920
Rsmq	**Standard deviation of the profile element widths** Rsmq is the root mean square deviation of widths, in mm.	ISO 21920
Rc	Mean height of the profile elements, in µm.	ISO 21920
Rcx	Maximum height of the profile elements, in µm.	ISO 21920
Rcq	**Standard deviation of the profile element heights** Rcq is the root mean square deviation of heights, in µm.	ISO 21920
RPc	**Peak count parameter** RPc corresponds to the number of mean widths of profile elements (XSm) inside a given length (default length 10 mm). (No unit.)	ISO 4287 ISO 21920 JIS B0601

The principle of the Motif standard consists of looking for local peaks and valleys in the primary profile, and associating one valley with the closest preceding and following peaks in order to create a Motif. Several iterative combinations of two Motifs each assure that the most important Motifs, the width of which fall below the limit A, are considered. If not otherwise specified, the default value is A = 0.5 mm. The limit A has a similar function as the cut-off in the Gaussian filtering.

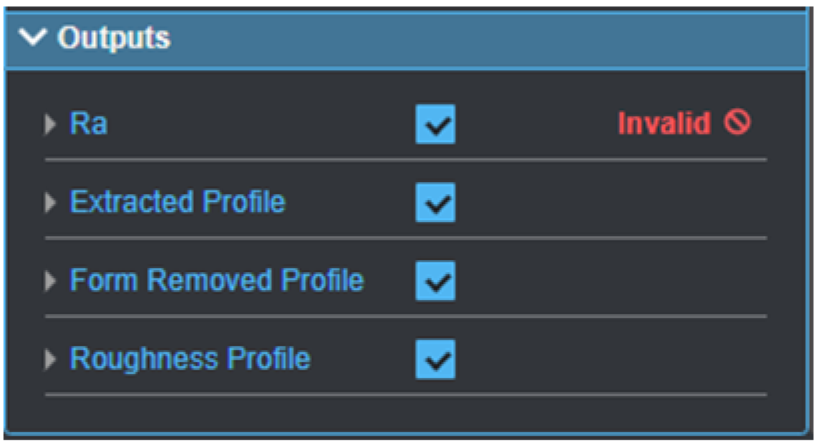
Motif Parameters

Parameter	Description	Standard
R	**Mean depth of roughness Motifs** R is the arithmetic mean value of the depths Hj of the roughness Motifs within the evaluation length, in µm.	ISO 12085
AR	**Mean spacing of roughness Motifs** AR is the arithmetic mean value of the lengths ARi of the roughness Motifs within the evaluation length, in mm.	ISO 12085
Rx	**Maximum depth of the roughness motifs** Rx is the deepest depth Hj within the evaluation length, in µm.	ISO 12085
Rt	Total height of the profile, in µm.	ISO 12085
Kr	Mean slope of the roughness motifs. (No unit.)	ISO 12085
Nr	**Number of roughness motifs** If the number of roughness motifs is less than 3, only the Rx and the Pt can be computed. (No unit.)	ISO 12085
SR	**Standard deviation of the depths of the roughness motifs** SR computes the dispersion of the Ri values, in µm.	ISO 12085
SAR	**Standard deviation of the spacings of the roughness motifs** SAR computes the dispersion of the ARi values, in mm.	ISO 12085
W	**Mean depth of waviness Motifs** W is the arithmetic mean value of the depths Hj of the roughnes Motifs within the evaluation length, in µm.	ISO 12085
AW	**Mean spacing of waviness Motifs** AW is the arithmetic mean value of the lengths ARi of the roughness Motifs within the evaluation length, in mm.	ISO 12085
Wx	**Maximum depth of the waviness motifs** Wx is the biggest of all Wi within the evaluation length, in µm.	ISO 12085
Wte	Total depth of waviness, in µm.	ISO 12085
Kw	Mean slope of the waviness motifs. (No unit.)	ISO 12085
Nw	**Number of waviness motifs** If the number of roughness motifs is less than 3, only the Wx and the Wte can be computed. (No unit.)	ISO 12085
SW	**Standard deviation of the depths of the waviness motifs** SW calculates the dispersion of the Wi values, in µm.	ISO 12085
SAW	**Standard deviation of the spacings of the waviness motifs** SAW calculates the dispersion of the AWi values, in mm.	ISO 12085
Trc	**Micro Geometric material ratio** Trc is the bearing fraction at a given depth c of the envelope leveled	ISO 12085

Parameter	Description	Standard
	profile, as a percentage.	
HTrc	Micro geometric criterion, in μm.	
Rke	Kernel roughness of the envelope leveled profile, in μm.	
Rpke	Reduced height of the envelope leveled profile, in μm.	
Rvke	Reduced depth of the envelope leveled profile, in μm.	

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

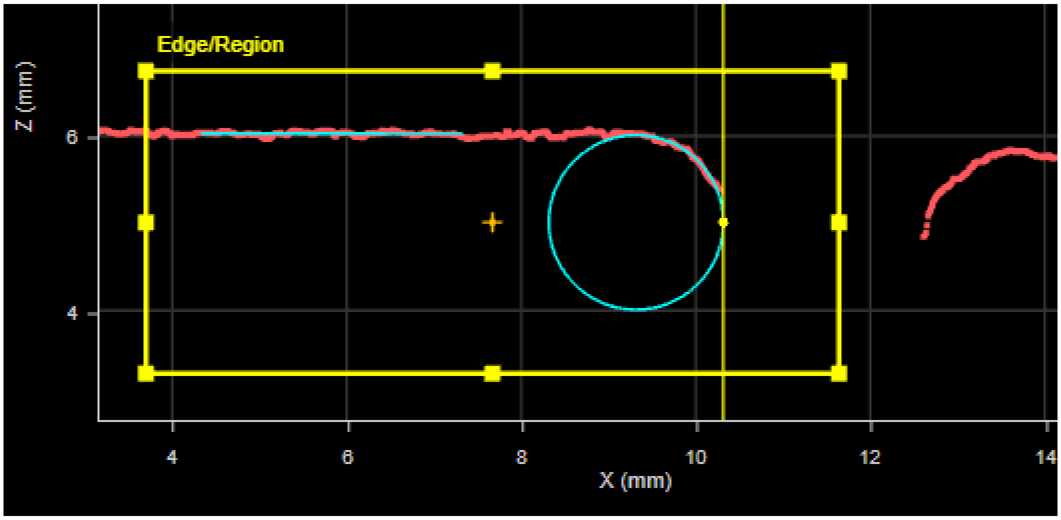
Measurement
Ra Arithmetic average of the absolute values of the roughness profile ordinates. For the other measurements, see the parameter details above.

Data

Type	Description
Extracted Profile	Extraction of the measured profile with the defined region.
Form Removed Profile	Obtained after removal of the nominal form (F-operation).
Roughness Profile	Obtained after filtering including a (high-pass) L-filter (λc) rejecting long wavelengths. When presenting the roughness profile, the mean line is the zero line.

Profile Round Corner

The Round Corner tool measures corners with a radius, returning the position of the edge of the corner and the angle of adjacent surface with respect to the X axis.



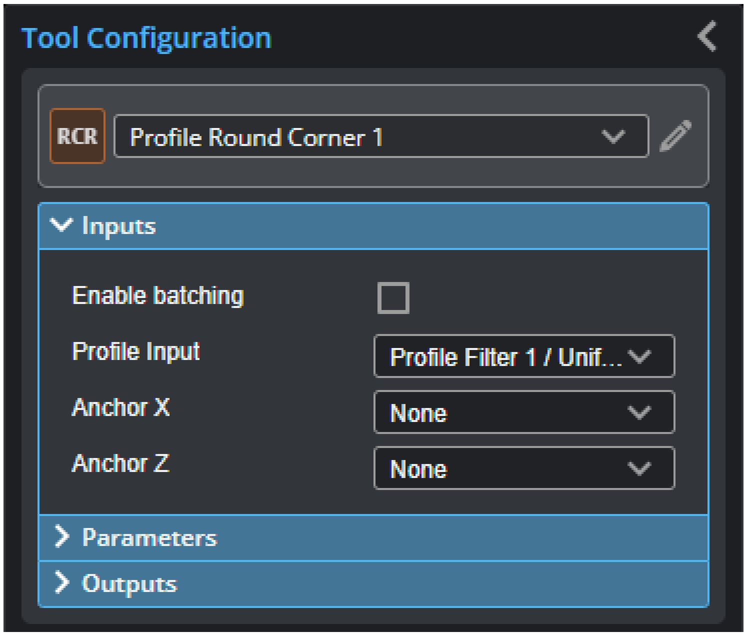
For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

The Round Corner tool uses a complex feature-locating algorithm to find the edge and return measurements. The behavior of the algorithm can be adjusted by changing the tool's parameters. For a detailed explanation of the algorithm, see *Gap and Flush Algorithms* on page 371.

	You must make sure that there are enough data points to define the edge (proper exposure, etc.). If not, the algorithm will not function.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



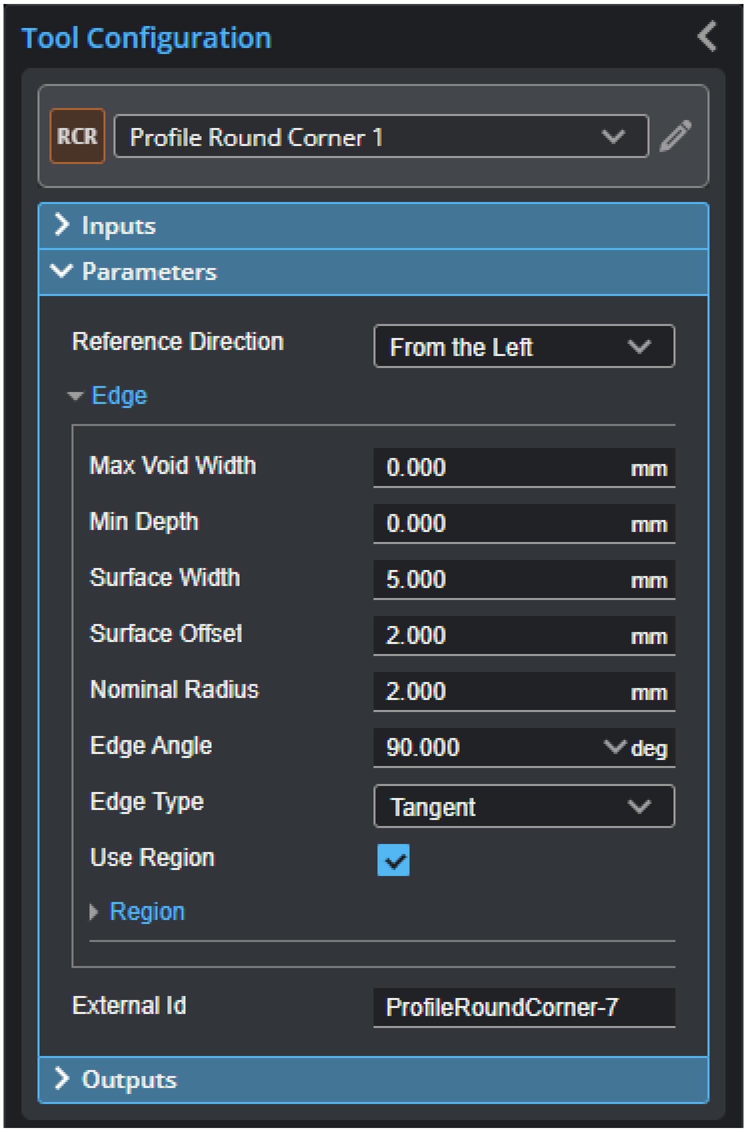
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. If **Enable Batching** is *disabled* and the passed array contains more than two elements, GoPxL displays an error. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

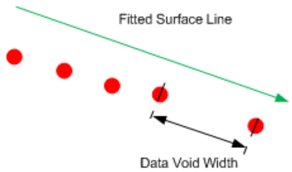
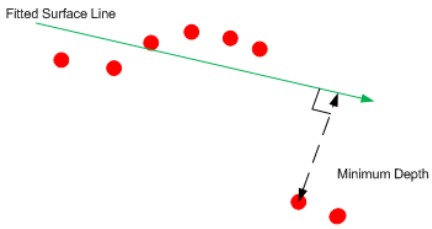
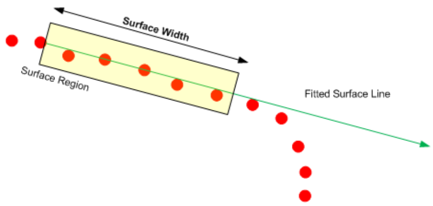
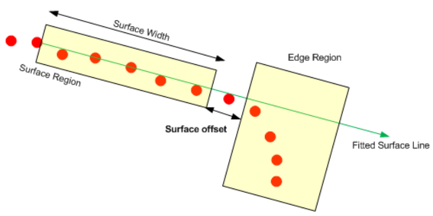
Parameters

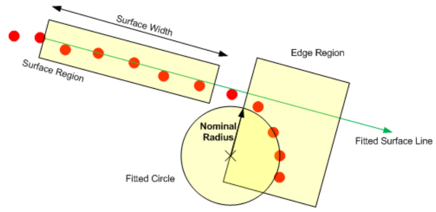
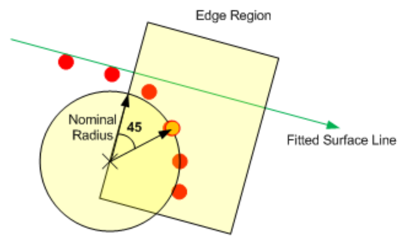
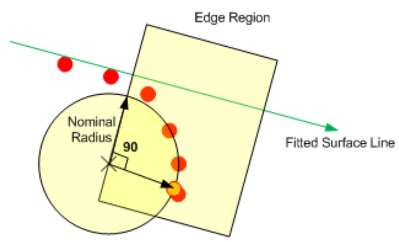
You configure the tool's parameters in the expandable **Parameters** section.

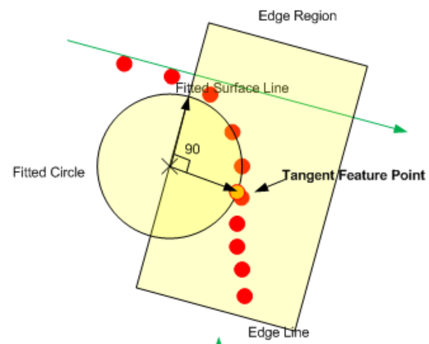
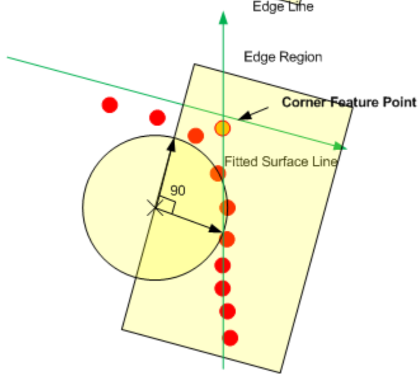


Parameters

Parameter	Description
Reference Direction	Defines the side used to calculate the rounded corner.
Edge	When expanded, displays the parameters for defining the edge (see below).
Max Void Width	The maximum allowed width of missing data caused by occlusion or data dropout. A larger value prevents the algorithm from registering a section of missing data as an edge. Setting the value to 0 causes the algorithm to try to detect an edge in every missing data section.

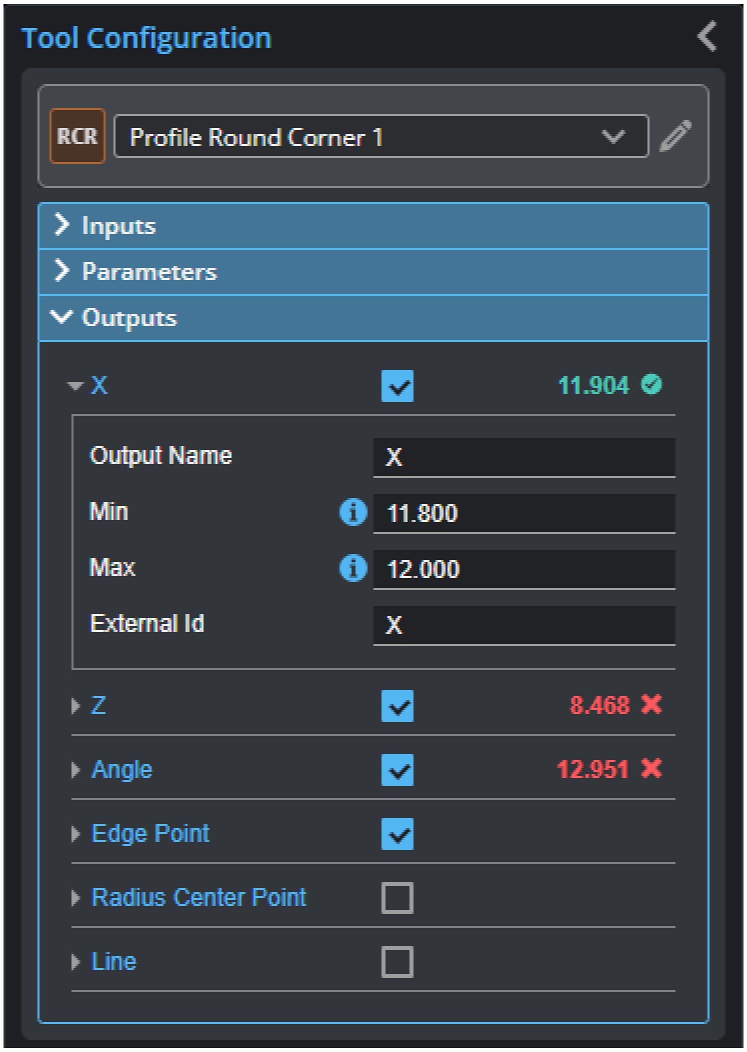
Parameter	Description
	
Min Depth	Defines the minimum depth before an opening could be considered to have a potential edge. The depth is the perpendicular distance from the fitted surface line. 
Surface Width	The width of the surface area in which data is used to form the fitted surface line. This value should be as large as the surface allows. 
Surface Offset	The distance between the edge region and the surface region. Setting a small value allows the edge within a tighter region to be detected. However, the measurement repeatability could be affected if the data from the edge are considered as part of the surface region (or vice versa). A rule of thumb is to set **Surface Offset** equal to **Nominal Radius**. 
Nominal Radius	The radius of the curve edge that the tool uses to locate the edge region. The algorithm searches for a start position in which the remaining data most resemble a circle of the specified nominal radius.

Parameter	Description
	
Edge Angle	A point on the best fit circle to be used to calculate the feature point. The selected point is on the circumference at the specified angle from the start of the edge region. The angle is measured from the axis perpendicular to the fitted surface line.  
Edge Type	Defines the type of feature point to use for the edge (Corner or Tangent). A tangent edge point is the point selected based on the defined Edge Angle. A corner edge point is the intersect point between the fitted surface line and a edge line formed by interpolating the points at and after the tangent within the edge region.

Parameter	Description
	
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

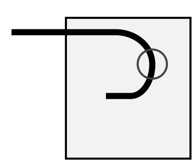
Most tools provide measurements, geometric features, or data as outputs.

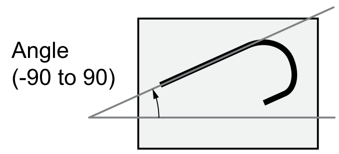
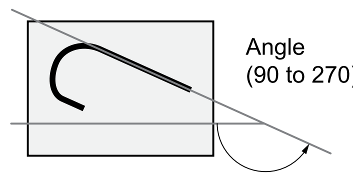


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Measures the X position of the location where the tangent touches the edge, or intersect of the tangent and the line fitted to the surface used by the measurement (see **Reference Side**, below).	
Z Measures the Z position of the location where the tangent touches the edge, or intersect of the tangent and the line fitted to the surface used by the	

Measurement	Illustration
measurement (see **Reference Side**, below).	
Angle Measures the angle of the line fitted to the surface next to the corner (see **Reference Side**, above), with respect to the X axis. Left edge angles are from -90 to 90. Right edge angles are from 90 to 270.	 

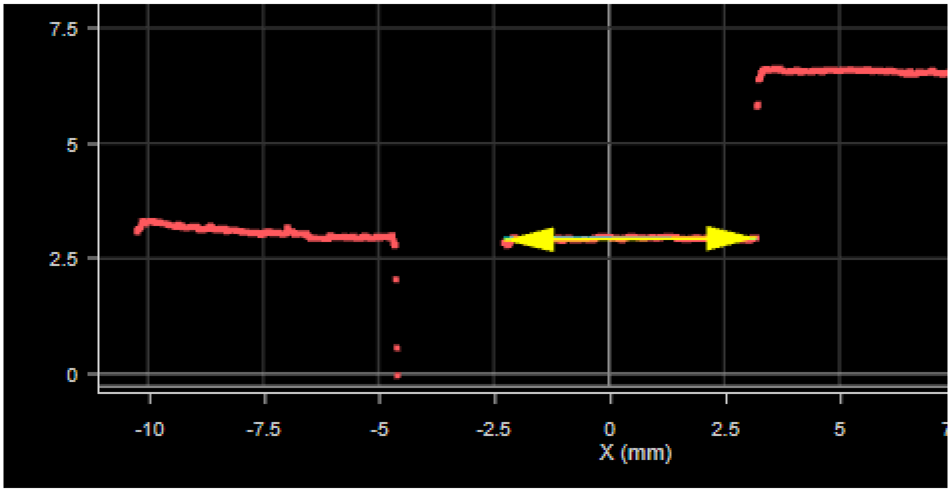
Features

Type	Description
Edge Point	The position of the edge.
Radius Center Point	The center of the radius.
Line	The fitted surface line.

For more information on geometric features, see *Geometric Features* on page 262.

Profile Strip

The Strip tool returns various measurements related to a strip, such as its X and Z positions, width, and height.



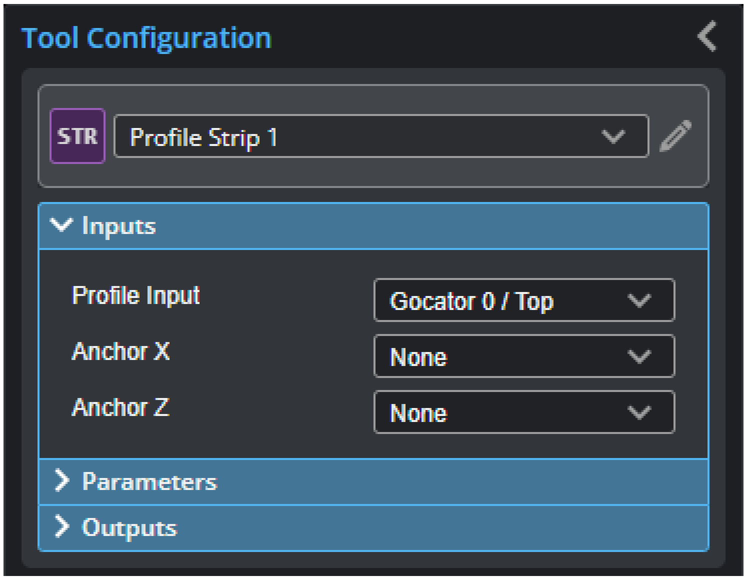
The tool lets you optionally set an index to return the measurements of a specific strip when more than one is visible in the region of interest. If you need to return multiple strips from a profile, add a Profile Strip tool for each strip, configure it to represent the desired strip, and set its **Selection Index** parameter to the desired strip (0-based index).

The Strip tool uses a complex feature-locating algorithm to find a strip and then return measurements. For a detailed explanation of the algorithm, see *Strip Algorithm* on page 420. The behavior of the algorithm can be adjusted by changing the tool's parameters.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



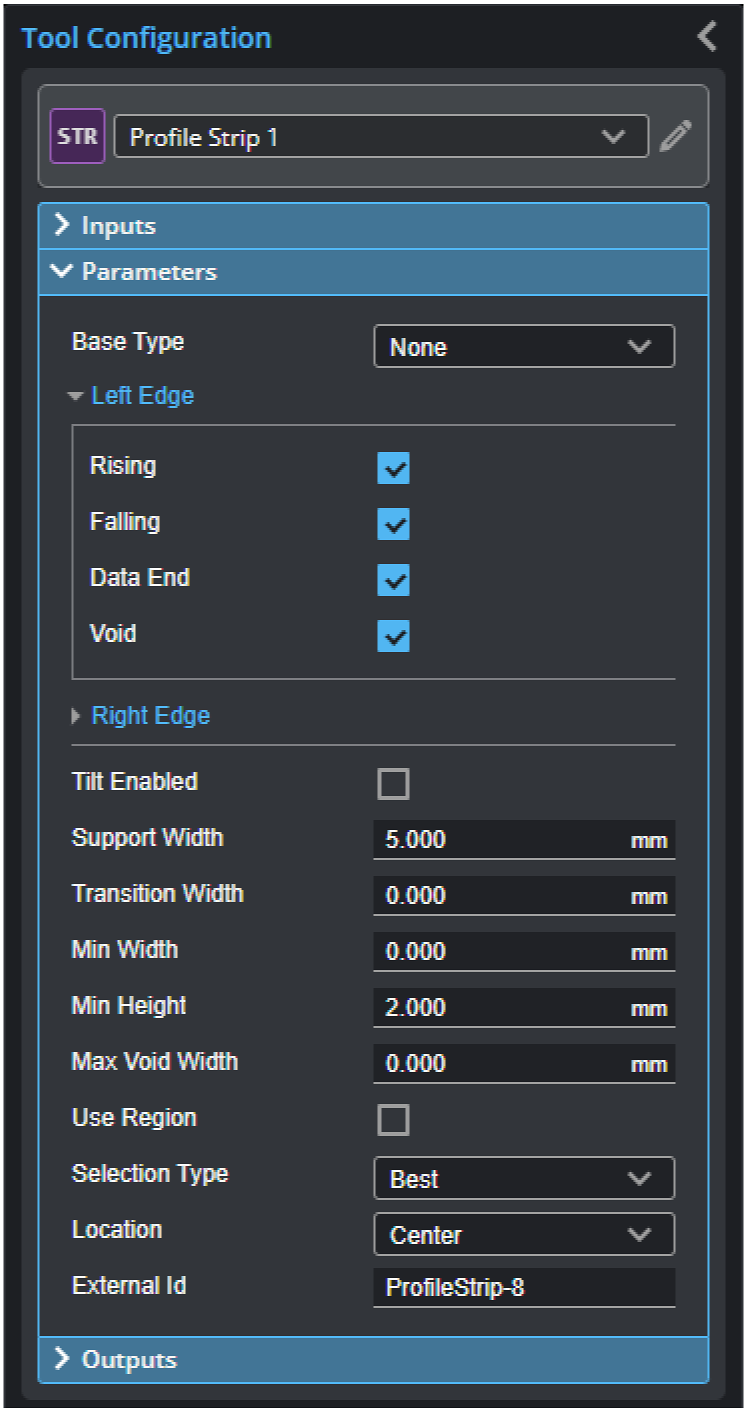
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

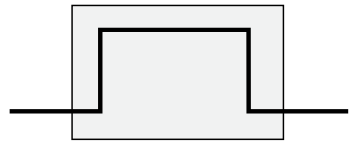
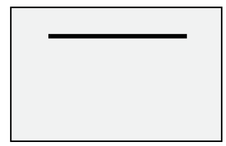
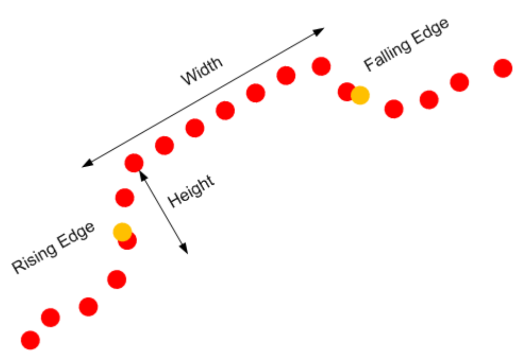
Parameters

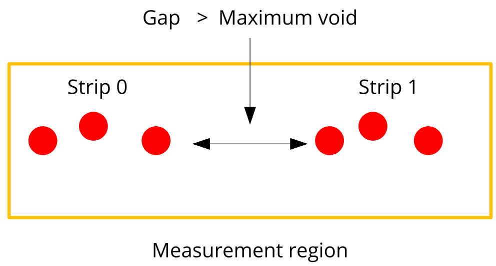
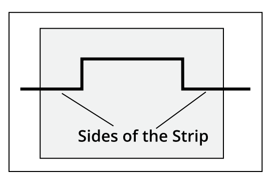
You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Base Type	Affects detection of rising and falling edges.

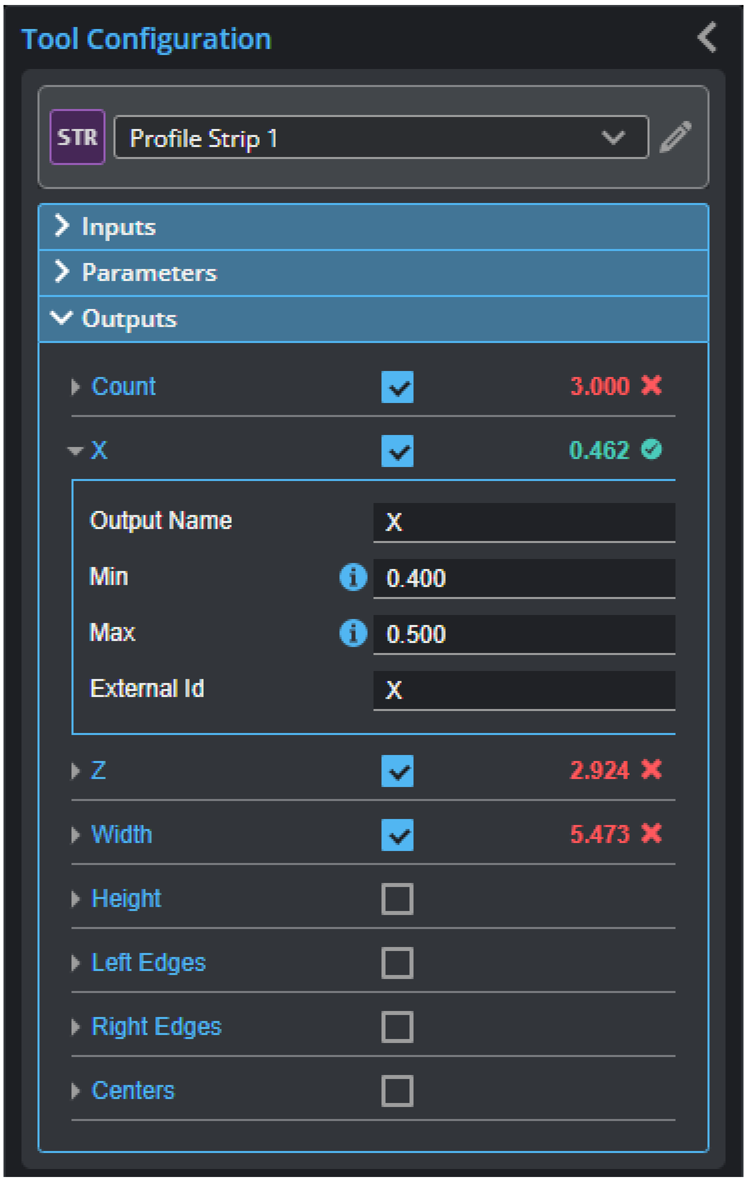
Parameter	Description
	Base Type = Flat  **Base Type = None**  When **Base Type** is set to **Flat**, both strip (raised area) and base support regions are needed. When set to **None**, only a point that deviates from a smooth strip support region is needed to find a rising or falling edge.
Left Edge Right Edge	Specifies the features that will be considered as the strip's left and right edges. You can select more than one condition. **Rising** - Rising edge detected based on the strip edge parameters. **Falling** - Falling edge detected based on the strip edge parameters. **Data end** - First valid profile data point in the measurement region. **Void** - Gap in the data that is larger than the maximum void threshold. Gaps connected to the measurement region's boundary are not considered as a void. For the definitions of these conditions, see *Strip Start and Terminate Conditions* on page 420.
Tilt Enabled	Enables and disables tilt correction. The strip may be tilted with respect to the sensor's coordinate X axis. This can be caused by conveyor vibration. If the Tilt option is enabled, the tool will report the width and height measurements, corrected by the tilt angle of the strip. 
Support Width	Specifies the width of the region around the edges from which the data is used to calculate the step change. For information on how this parameter is used by different base types, see *Strip Step Edge Definitions* on page 422.
Transition Width	Specifies the nominal width needed to make the transition from the base to the

Parameter	Description
	strip. For information on how this parameter is used by different base types, see *Strip Step Edge Definitions* on page 422.
Min Width	Specifies the minimum width for a strip to be considered valid.
Min Height	Specifies the minimum deviation from the strip base. For information on how this parameter is used by different base types, see *Strip Step Edge Definitions* on page 422.
Max Void Width	The maximum width of missing data allowed for the data to be considered as part of a strip when **Void** is selected in the **Left** or **Right** parameter. This value must be smaller than the edge **Support Width**.  When occlusion and exposure causes data drops, use a gap filling filter in a Profile Filter tool to fill the gaps; for more information, see *Profile Filter* on page 335.
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The measurement region defines the region in which to search for the strip. If possible, the region should be made large enough to cover the base on the left and right sides of the strip.  For more information, see *Regions* on page 250.
Location *(Strip Height, Strip X, and Strip Z measurements only)*	Specifies the strip position from which the measurements are performed. **Left** - Left edge of the strip. **Right** - Right edge of the strip. **Center** - Center of the strip.
Selection Type	Specifies how a strip is selected when there are multiple strips within the measurement area. **Best** - The widest strip. **Index Left** - 0-based strip index, counting from left to right. **Index Right** - 0-based strip index, counting from right to left.
Index	0-based strip index. Only displayed when **Selection Type** is set to an

Parameter	Description
	"Index" option.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

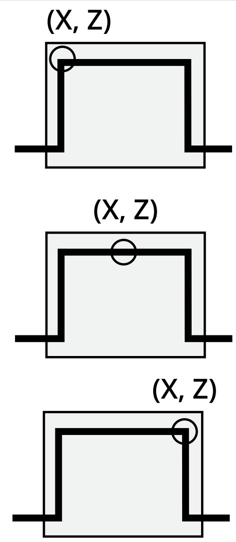
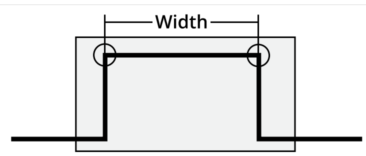
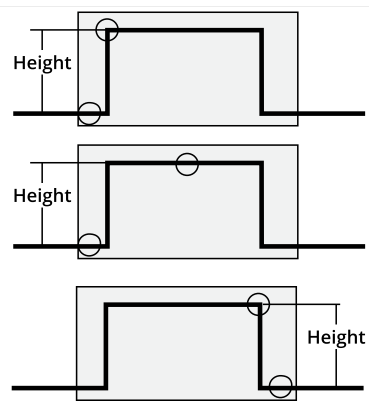
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

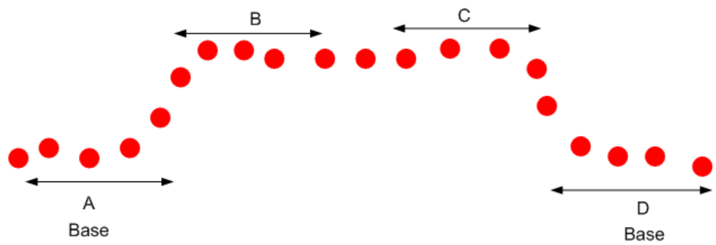
Measurement	Illustration
Count The number of strips.	
X, Z Measures the X and Z position, respectively, of a strip.	
Width Measures the width of a strip.	
Height Measures the height of a strip.	

Features

Type	Description
Left Edges Right Edges Centers	Arrays of Point geometric features representing the left edges, right edges, and centers of all strips, from left to right.

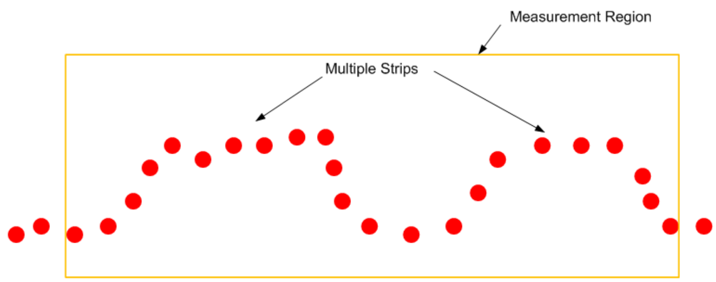
Strip Algorithm

A strip is a flat region bounded on the left and on the right by edges. The Strip tool can measure the edge positions, width, and height of a strip. The Strip tool assumes that regions outside the strip, referred to as the base regions (Region A and D below), deviate in height from the start and end parts of a strip (Region B and C).



When the target is sitting on the surface, the base is lower than the strip (as shown above). Alternatively for a groove the base is above the strip surface. The base could be missing when the target is hanging in the air or the surface holding the target falls outside the sensor's active area. You can control the base type in the measurement panel.

The Strip tool can detect multiple strips. You can select a region of interest, referred to as the measurement region, from which the algorithm search for multiple strips.

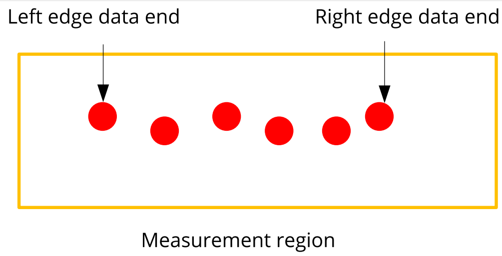
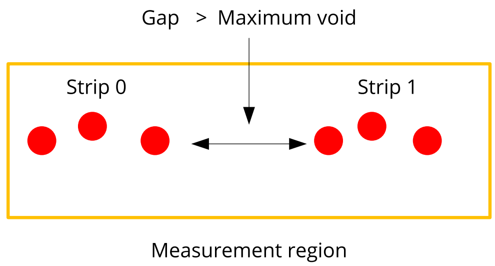
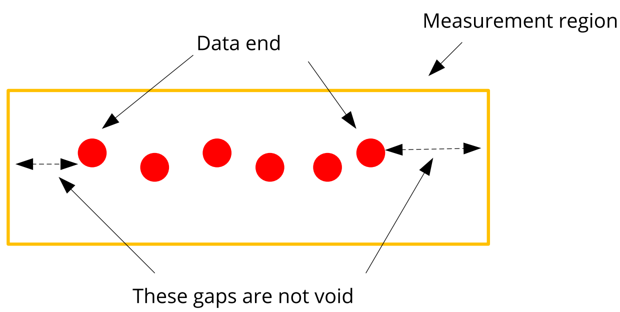


Strip Start and Terminate Conditions

The Strip tool allows you to define how a strip starts and ends. The Left Edge parameter controls how a strip starts and the Right Edge parameter controls how a strip ends.

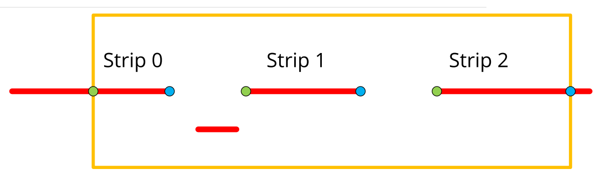
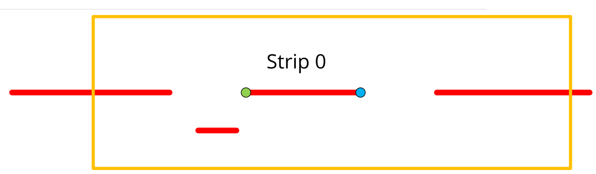
Start / terminate conditions

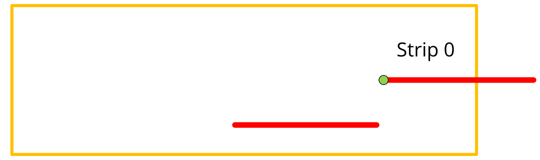
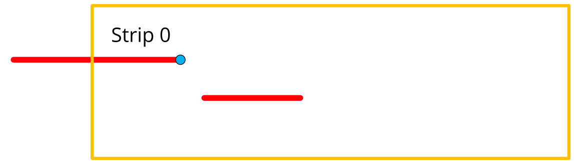
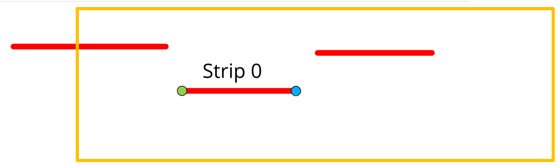
Condition	Description
Rising	Rising step edge detected based on the strip edge parameters. For details on how the step edge is detected, *Strip Step Edge Definitions* on page 422.
Falling	Falling step edge detected based on the strip edge parameters. For details on how the step edge is detected, *Strip Step Edge Definitions* on page 422.
Data end	The first (for the left edge) or the last (for the right edge) valid profile data point in the measurement region.

Condition	Description
	
Void	Gaps in the data that are larger than the maximum void threshold.  Gaps at the ends of the measurement region's boundary are not considered as a void. 

The following examples show how the parameters affect the strip detection in different scenarios.

Left and Right Edge conditions

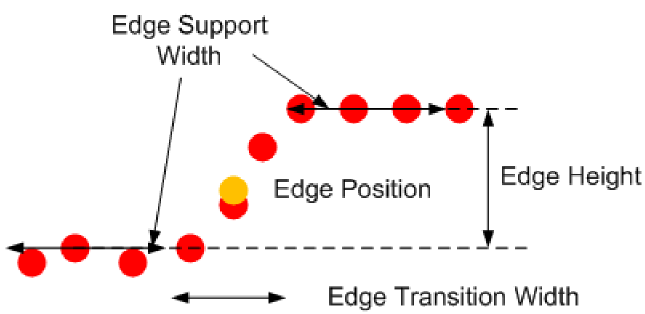
Condition	Example
Left: Rising, data end, void Right: Falling, data end, void	
Left: Rising, void Right: Falling, void	

Condition	Example
Left: Rising Right: Data end, void	
Left: Data end, void Right: Falling	
Left: Falling Right: Rising	

Strip Step Edge Definitions

The Strip tool detects step edges based on the parameters Base Type, Edge Transition Width, Edge Support Width, and Minimum Edge Height.

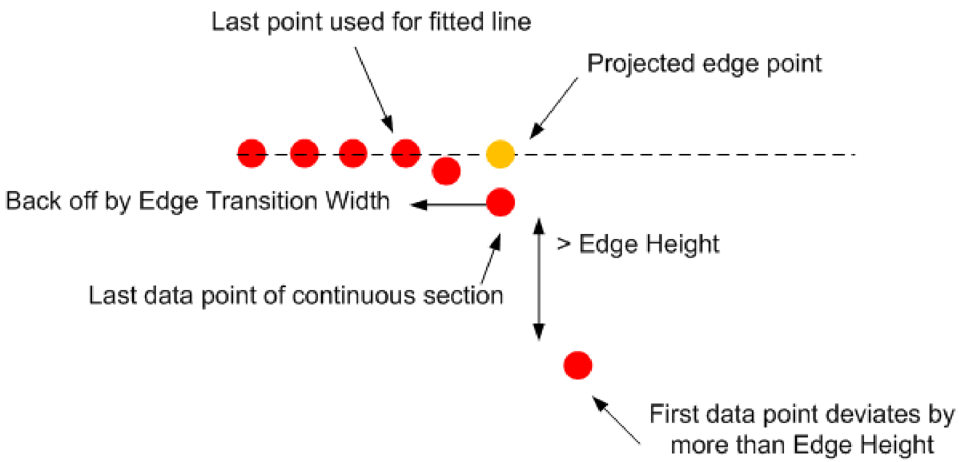
When Base Type is set to Flat, the regions around the edges are visible and the edge positions are between the base and the strip surface.



The Minimum Edge Height parameter defines the size of the step edge. The Edge Transition Width parameter specifies the nominal width of the transition, from the base to the strip surface.

The Edge Support Width parameter defines the width of the region around the edges from which the data is used to measure the step change. To improve noise immunity, the height level of the Edge Support Width parameter is calculated by averaging the data within the region.

When the base is set to None, the tool looks for continuous sections that are wider than the Edge Support Width parameter and have no data points that deviate positively or negatively more than the value of the Minimum Edge Height parameter. The data in the strip support region (the raised area) must be smooth. The height level of the continuous region is calculated based on the fitted line as shown below.



The algorithm then backs off by the value of the Edge Transition Width parameter and uses the data up to the back-off point to create the fitted line and projects the edge point on the line. This step prevents the points near the end of a rounded strip from affecting the height of the strip.

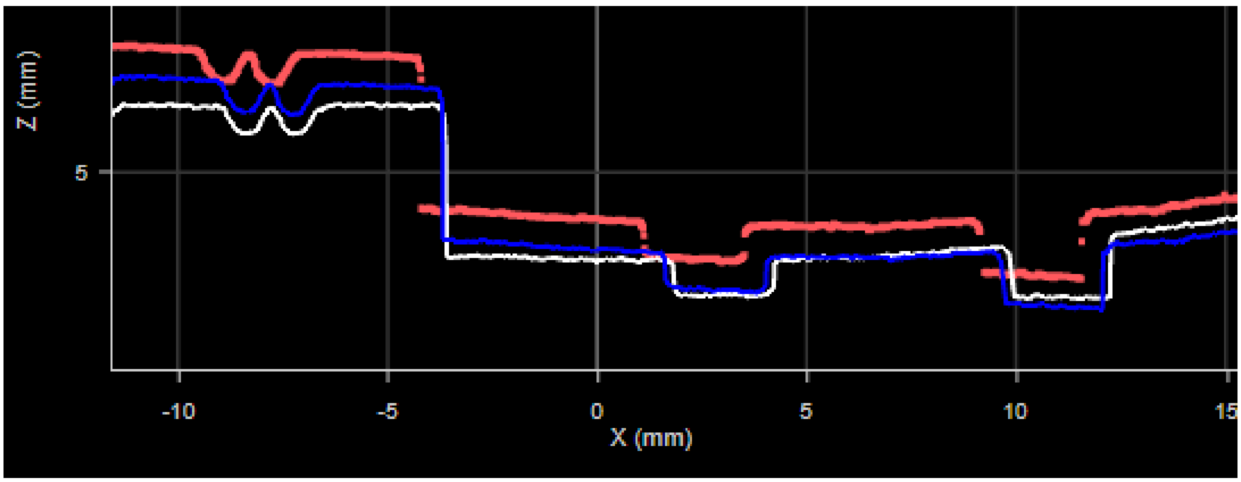
Profile Template Matching

The Profile Template Matching tool lets you align a profile to a "master" template profile you create in the tool (a "golden template"), compensating for movement of the target from frame to frame. As a result, you can perform measurements on a "stabilized" profile.

The tool returns measurements that represent differences between the profile and the master, letting you perform simple defect detection and location from within the tool.

The tool also outputs an aligned profile that other Profile measurement tools can use as input. Finally, the tool produces a "difference" profile on which you can similarly perform measurements.

In the data viewer, the profiles are rendered using different colors.



The master profile is rendered in white. The aligned profile is rendered in blue. The current profile is rendered in red.

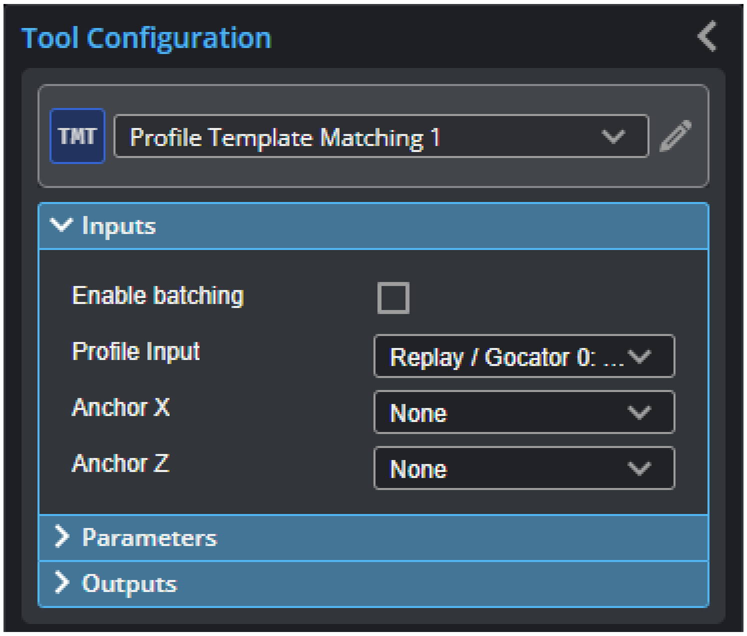
Note that in the image above, the tool is performing only a rough alignment to ensure that the different profiles are clearly visible. Typically, the aligned profile will be on top of the master profile.

The actual colors that are used for aligned and difference profiles *outputs*, as opposed to the profiles toggled with the **Display Master** and **Display Aligned Profile** parameters, will depend on how each profile's color is configured in the Displayed Outputs panel.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

If no profile alignment is performed (both **Coarse Align** and **Fine Align** are disabled), for example, if the targets are sufficiently fixed from profile to profile, the following measurements return 0.000:

- Transform X
- Transform Z
- Transform Y Angle

Master Compare must be enabled for the following measurements; otherwise, they return Invalid values:

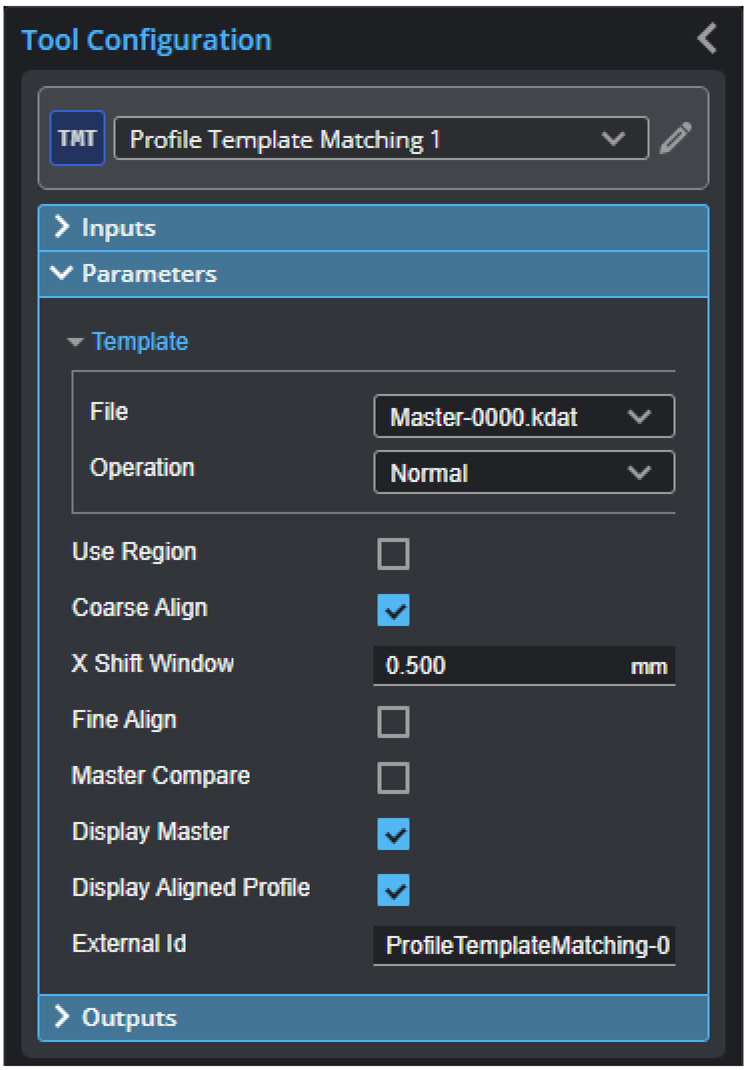
- Max Height Difference
- Max Difference Position X
- Max Difference Position Z

- Standard Deviation
- Difference Average
- Difference Sum
- Variance
- Matching Score

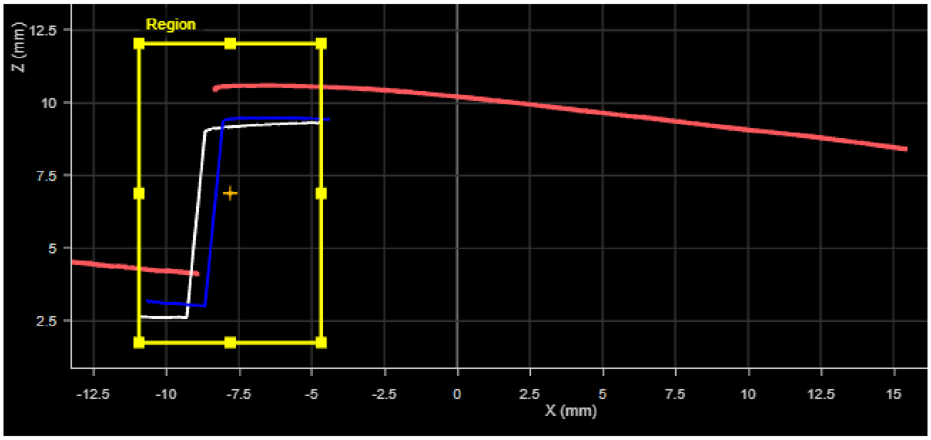
Also, for these "master compare" measurements, if the profile has been aligned to the master (either **Coarse Align** or **Fine Align** is enabled), the measurements compare the *aligned* profile and the master. If the profile has not been aligned (both alignment parameters are disabled), the measurement compares the *original* (unaligned) profile and the master.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



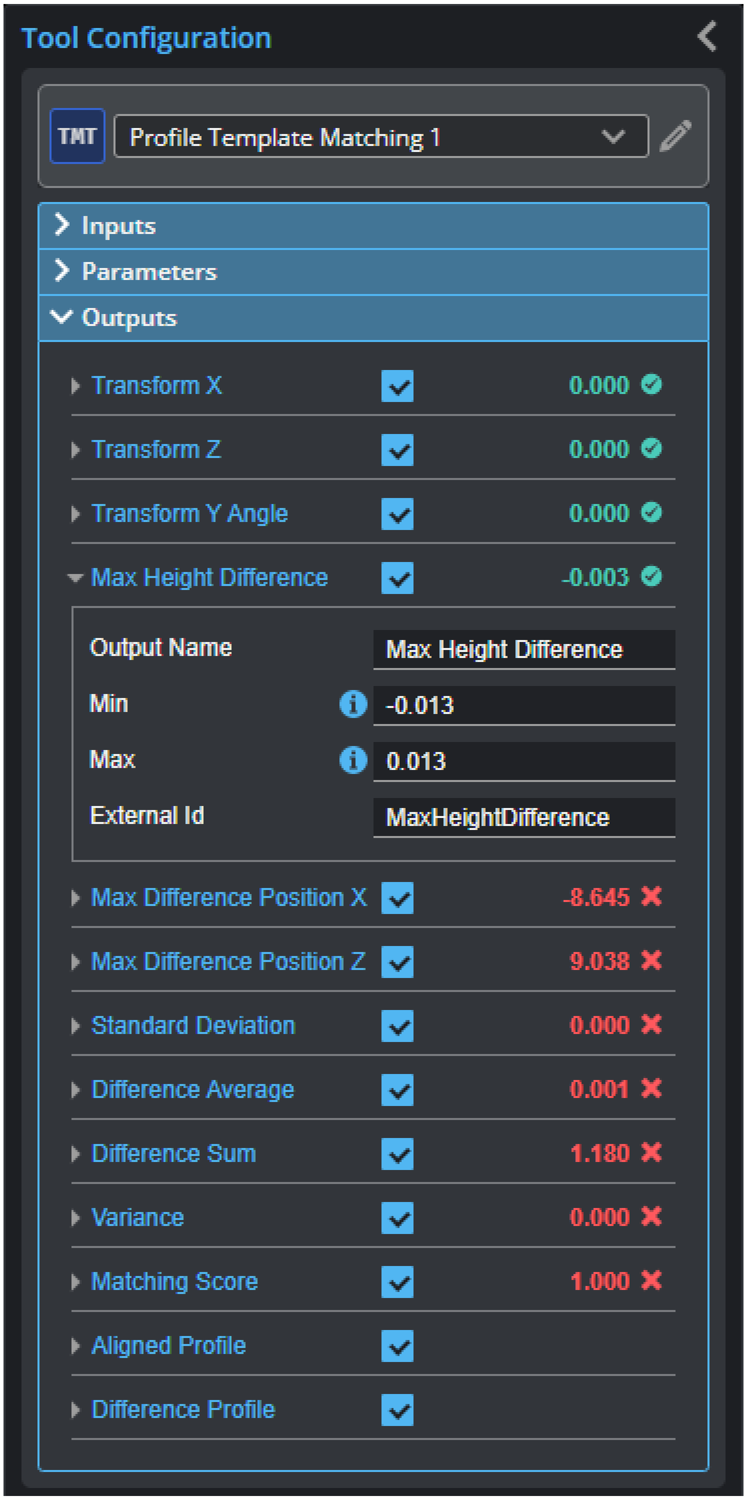
Parameters

Parameter	Description
Template	When expanded, displays **File** and **Operation** parameters.
File	A list of templates available to the tool. The template containing the profile the tool uses as a master profile for alignment and comparisons. Use the **Operation** parameter to add and remove templates to this list.
Operation	Provides operations related to profile template files (masters). One of the following: • **Normal**: Selected by the tool after you perform another file operation. • **Create**: Creates a new template file. • **Load**: Loads the template selected in **File**. • **Save**: Saves the current profile to a template file in the local file system and adds it to the list in **File**. Multiple templates can be available. • **Delete**:Deletes the template file selected in **File**. • **Refresh**: Refreshes the list of files.
Use Region	Whether the tool uses a user-defined region to perform *matching*. (The tool uses *only* the data profile and master data in this region to perform matching.) If this option is not checked, the tool performs matching using data from the entire active area. For more information on regions, see *Regions* on page 250.
Region	The size and position of the region in which the matching (alignment) is performed. This can be useful if only a small part of the profiles is suitable for alignment. Master comparison measurements however are applied to the entire profile (current profile and master). For example, in the following image, the tool limits matching to the data in the match region. But the measurement (Max Height Difference in this case) is calculated on the data outside the region. 
Coarse Align	When enabled, shows the **X Shift Window** parameter. Use this setting by itself if you expect targets will only move along the X and Z axes (that is, you don't expect

Parameter	Description
	rotation). Otherwise, when combined with **Fine Align**, it provides a good initial start position for fine alignment.
X Shift Window	The maximum distance on the X axis the tool can move the current profile in order to align it. Should be set to the maximum amount the part is expected to shift left or right. (Enabled using the **Coarse Align** parameter.)
Fine Align	When enabled, lets you set the **Max Iteration** and **Match Window** parameters for fine alignment. This alignment method is more accurate than coarse alignment but takes more time to run.
Max Iteration	The maximum number of iterations the tool uses to perform fine alignment of the profile to the master.
Match Window	The region in which points are evaluated for a match. It there's a larger difference between the current profile and the master than the match window size, it would ignore the point.
Master Compare	Causes the tool to compare the current profile to the master profile and return results in some of the tool's measurements. (See list above.) When disabled, the measurements that compare the profile to the master return invalid values.
Difference Profile Median Size	Defines the size of the window the tool uses to smooth out noise in the Difference Profile data output.
Tolerance	The difference tolerance for the master comparison.
Display Master	Displays the Master template (white profile).
Display Aligned Profile	Displays the aligned (blue profile).
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Transform X **Transform Z** The distance the profile has shifted on the X and Z axis after alignment to the master, respectively.
Transform Y Angle The rotation of the profile around the Y axis after alignment.
Max Height Difference The maximum height difference between the profile and the master.
Max Difference Position X **Max Difference Position Z** The X and Z positions of the maximum height difference between the profile and the master.
Standard Deviation The standard deviation between the profile and the master.
Difference Average The average difference on the Z axis between the profile and the master.
Difference Sum The sum of the differences on the Z axis between the profile and the master.
Variance Returns the variance of a difference profile calculated by subtracting the current profile from the master.
Matching Score Returns a value between 0 and 1 that is the is the percentile of standard deviation of a difference profile (calculated by subtracting the current profile from the master) from the tolerance.

Data

Type	Description
Aligned Profile	The profile aligned to the master.
Difference Profile	A profile representing the differences between the profile and the master. Z values in the difference profile above 0 represent data points higher in the profile than in the master. Z values in the difference profile below 0 represent data points lower in the profile than in the master. Z values in the difference profile at 0 represent data points that are the same in the profile and the master.

Raw Profile Matching

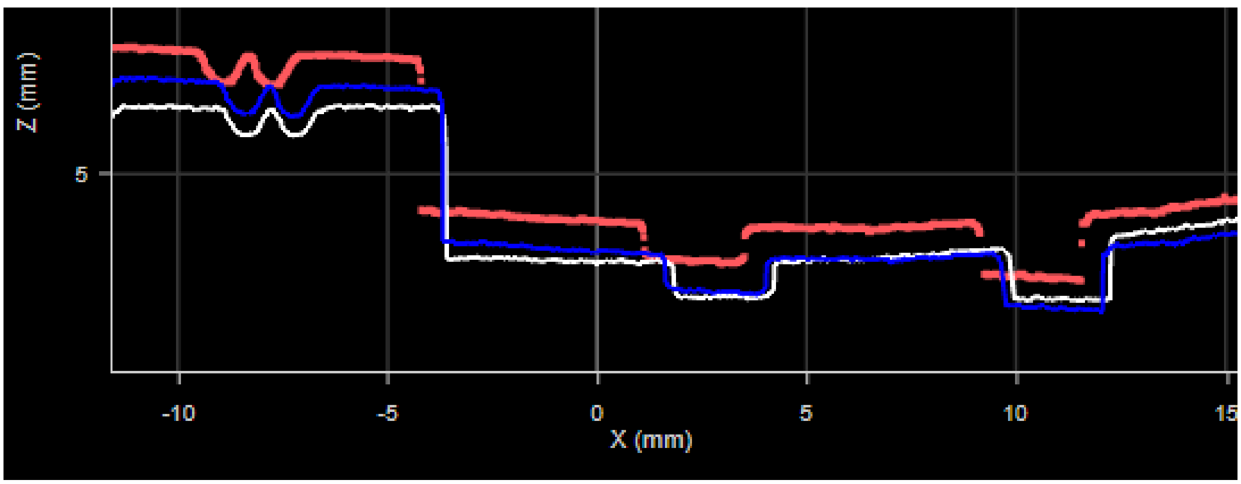
🗋	The Raw Profile Matching tool takes unresampled profile data as input (the **Enable uniform spacing** setting is disabled on Acquire > Scanners).

The Raw Profile Matching tool lets you align a profile to a "master" template profile you create in the tool (a "golden template"), compensating for movement of the target from frame to frame. As a result, you can perform measurements on a "stabilized" profile.

The tool returns measurements that represent differences between the profile and the master, letting you perform simple defect detection and location from within the tool.

The tool also outputs an aligned profile that other Profile measurement tools can use as input (via their **Stream** parameter).

In the data viewer, the profiles are rendered using different colors:



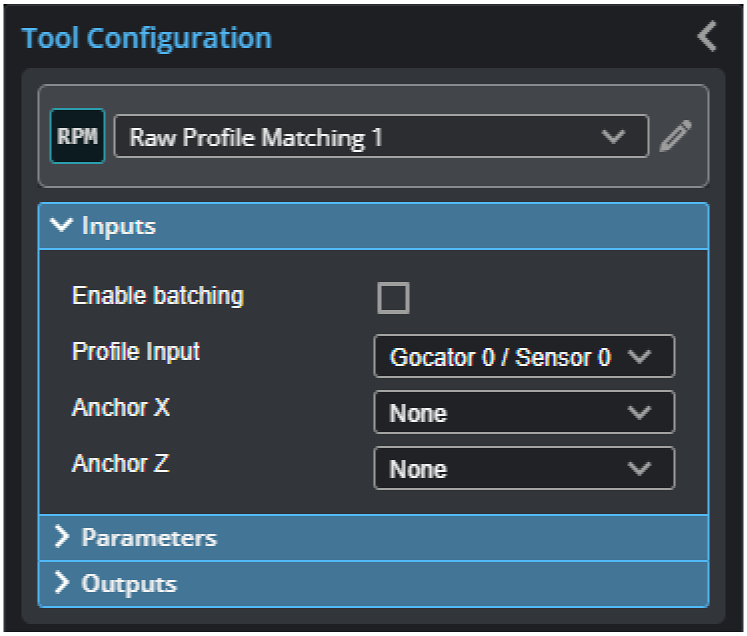
The master profile is rendered in white. The aligned profile is rendered in blue. The current profile is rendered in red.

Note that in the image above, the tool is performing only a rough alignment to ensure that the different profiles are clearly visible for illustration purposes. Typically, the blue aligned profile will be on top of the white master profile.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



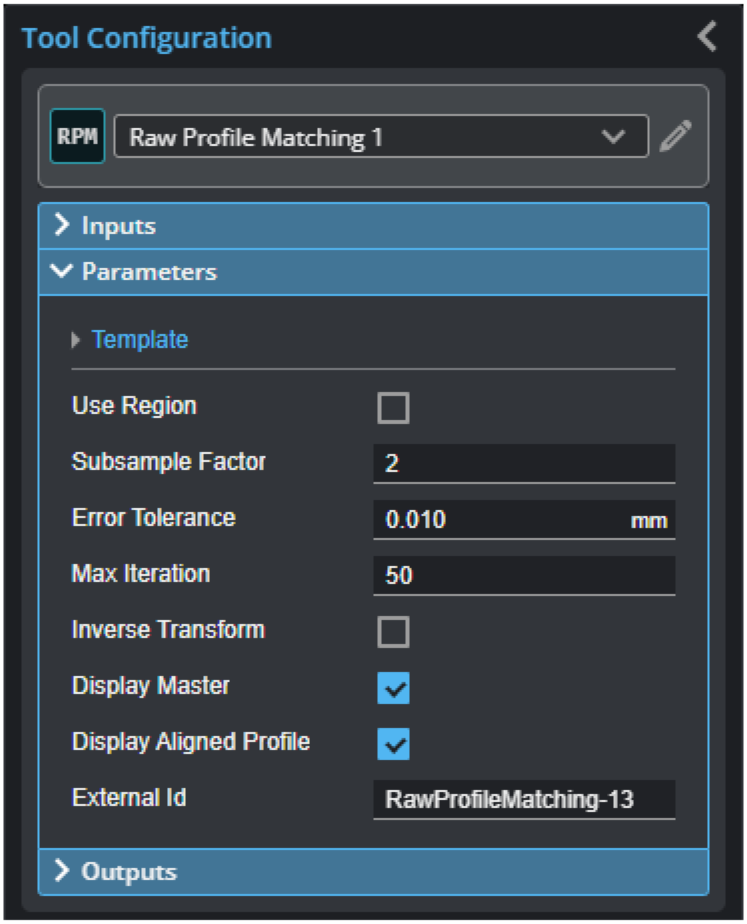
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes.
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

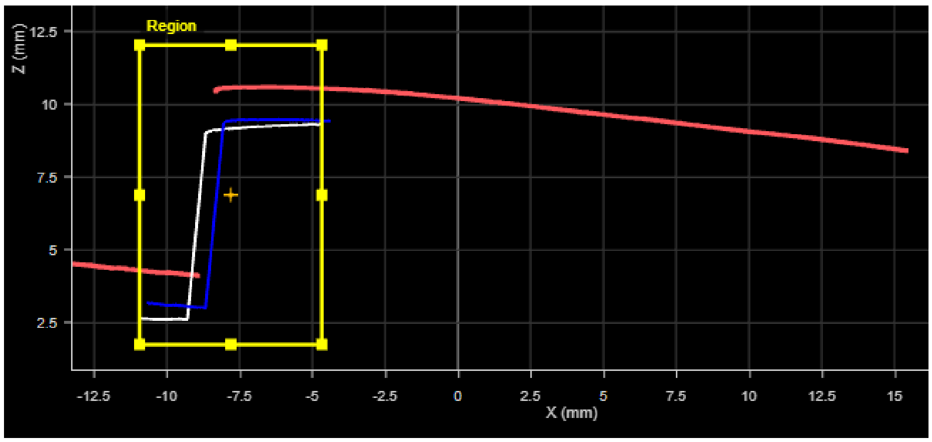
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



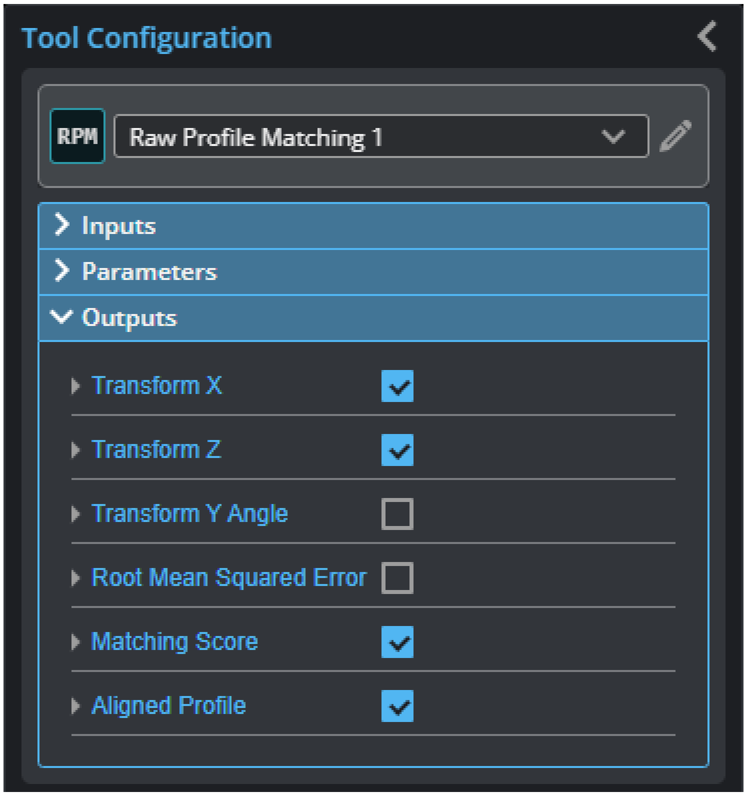
Parameters

Parameter	Description
Template	When expanded, displays **File** and **Operation** parameters.
File	A list of templates available to the tool to use as a master profile for alignment and comparisons. Use the **Operation** parameter to add and remove templates.
Operation	Provides operations related to profile template files (masters). One of the following: • **Normal**: Selected by the tool after you perform another file operation. • **Create**: Creates a new template file. • **Load**: Loads the template selected in **File**. • **Save**: Saves the current profile to a template file in the local file system and adds it to the list in **File**. Multiple templates can be available. • **Delete**:Deletes the template file selected in **File**. • **Refresh**: Refreshes the list of files.
Use Region	Indicates whether the tool uses a user-defined region to perform *matching*. (The tool uses *only* the data profile and master data in this region to perform matching.)

Parameter	Description
	If this option is not checked, the tool performs matching using data from the entire active area. For more information on regions, see *Regions* on page 250.
Match Region	When **Use Region** is enabled, provides settings for the size and position of the region in which the matching (alignment) is performed.. Master comparison measurements however are applied to the entire profile (current profile and master). For example, in the following image, the tool limits matching to the data in the match region. But the measurement (Max Height Difference in this case) is calculated on the data outside the region.  (The dashed lines are added to illustrate the hidden aligned profile and master.)
Subsample Factor	The sampling interval of the input profile point cloud.
Error Tolerance	The difference tolerance for the master comparison.
Max Iteration	The maximum number of iterations the tool uses to perform iterative closest point (ICP). The tool stops iterating once the input profile is considered within the error tolerance (see above).
Inverse Transform	When enabled, inverts the output transformation.
Display Master	Displays the Master template (white profile).
Display Aligned Profile	Displays the aligned profile (blue profile).
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

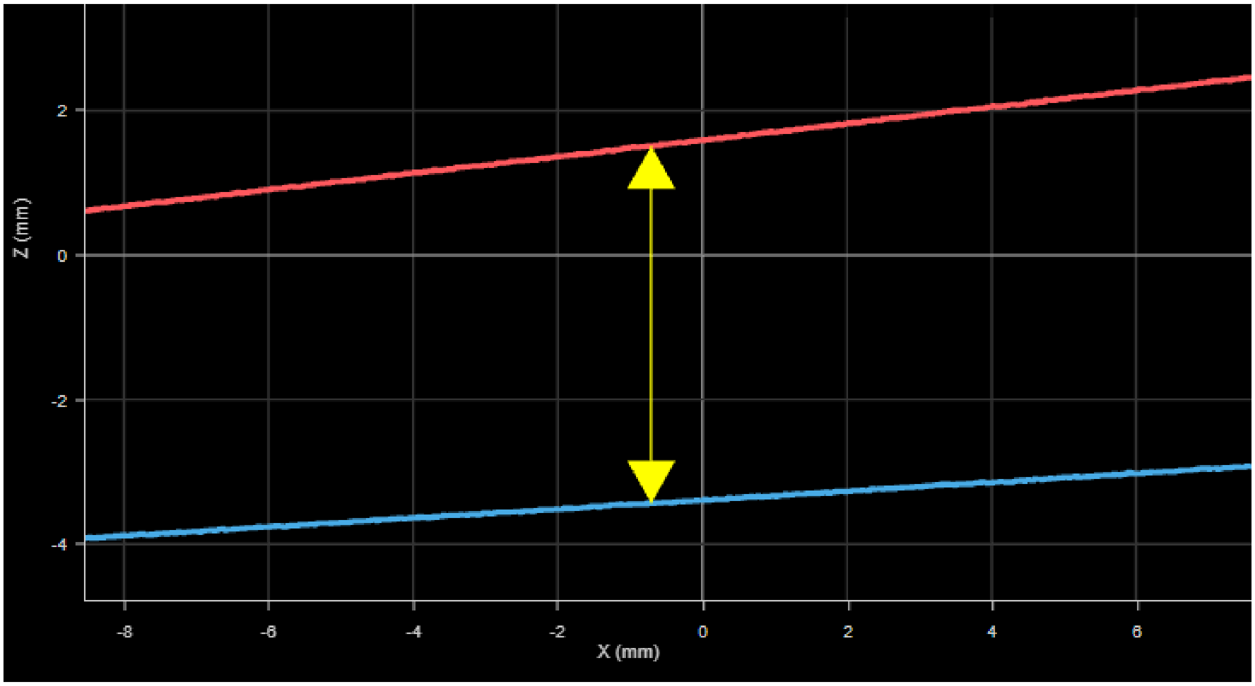
Measurement
Transform X **Transform Z** The distance the profile has shifted on the X and Z axis after alignment to the master, respectively.
Transform Y Angle The rotation of the profile around the Y axis after alignment.
Root Mean Squared Error The root mean squared error of the alignment.
Matching Score Returns a value between 0 and 1 representing the percentile of standard deviation of a difference profile (calculated by subtracting the current profile from the master) from the tolerance.

Data

Type	Description
Aligned Profile	The profile aligned to the master.

Profile Thickness

The Thickness tool lets you perform thickness measurements between the two profiles you specify.

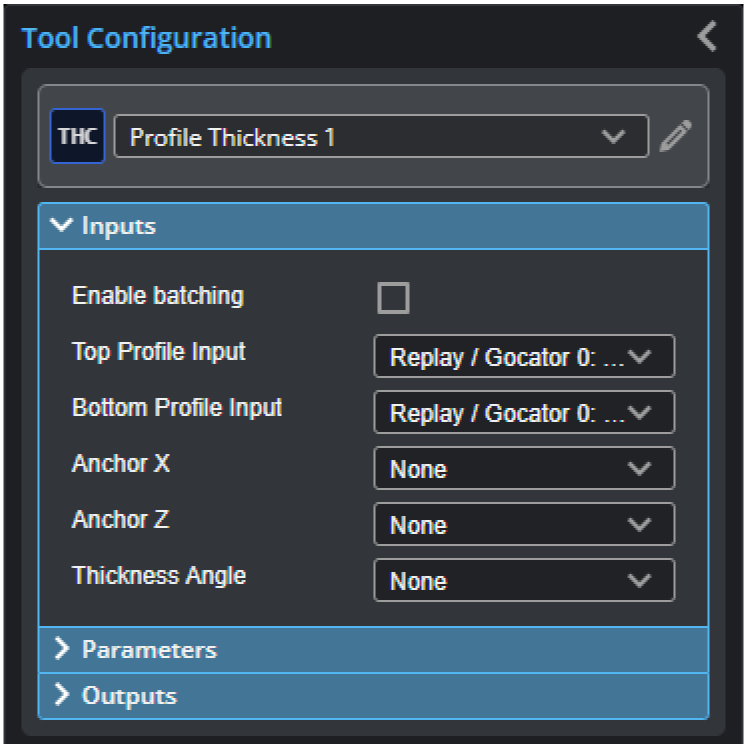


The tool also lets you specify an angle measurement relative to which the tool performs the thickness measurements, up to a maximum of 20 degrees, to compensate for minor tilt of the target. To get the compensation angle, you could first fit a straight line to the top profile using the Profile Line tool, and pass that tool's Angle measurement to Profile Thickness. (For information on Profile Line, see *Profile Line* on page 353.)

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

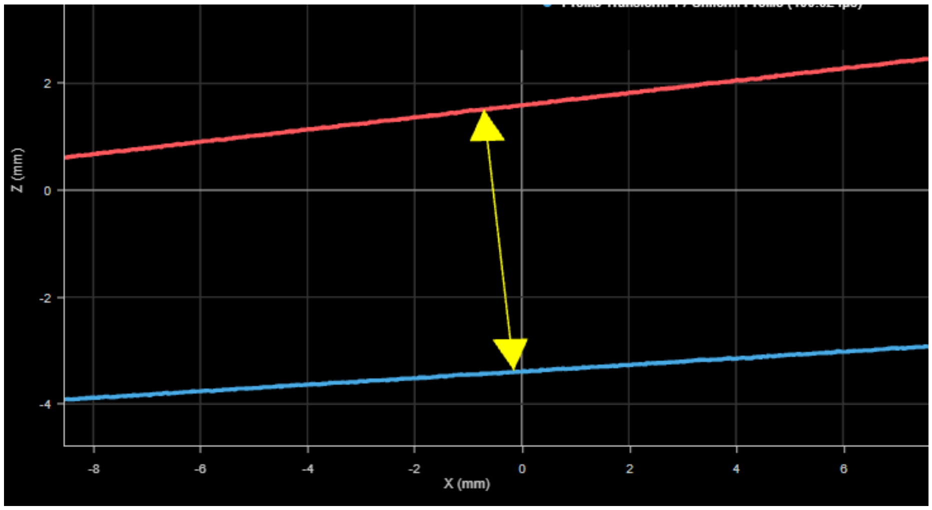
You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

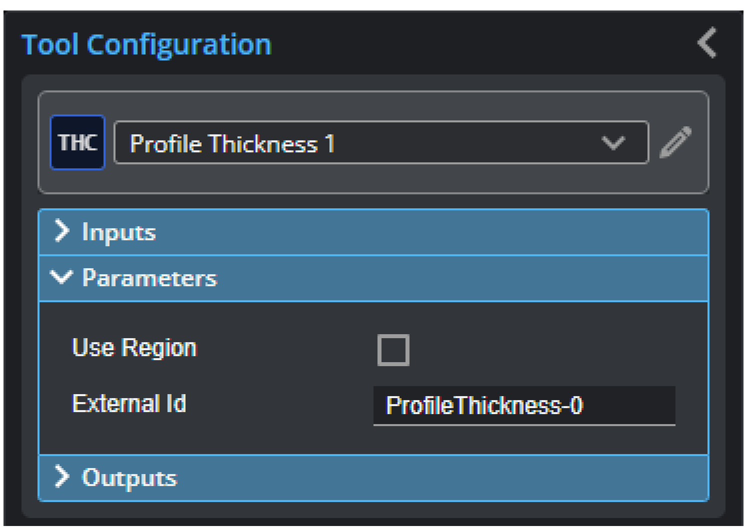
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Top Profile Input	The top and bottom profiles the tool uses to perform thickness measurements. If **Enable Batching** is checked, each input takes an array.

Name	Description
Bottom Profile Input	
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Thickness Angle	An angle measurement from another tool that the Thickness tool uses for the angle of the thickness measurement. If no angle is provided, the thickness measurement is perpendicular to the X axis. 

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



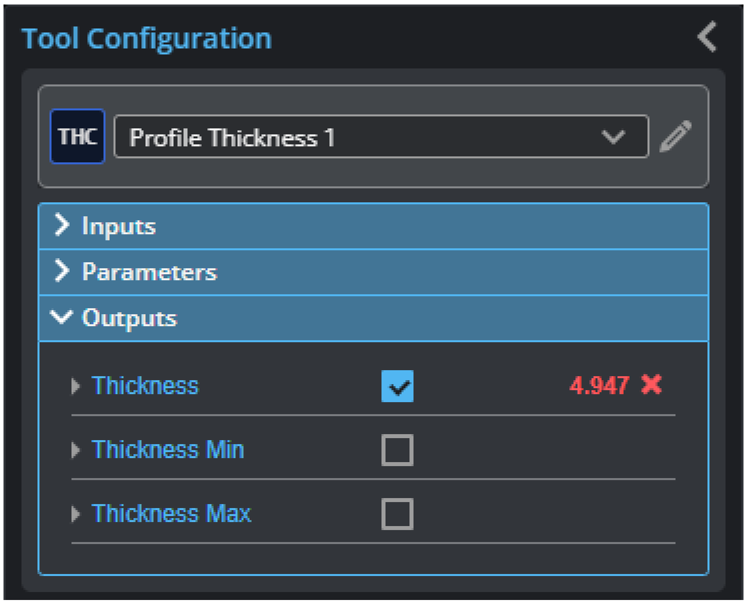
Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.

Parameter	Description
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

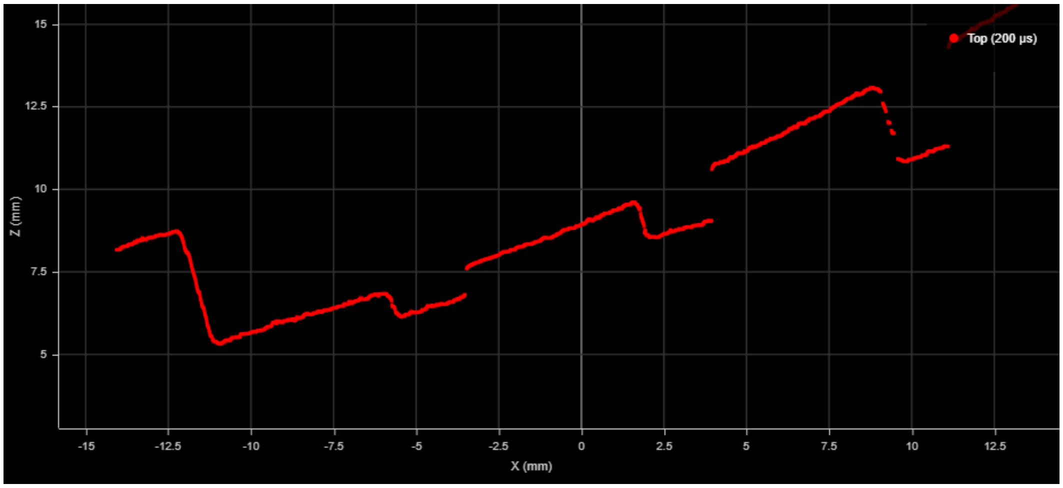
Measurements

Measurement
Thickness The thickness measured in the scan data.
Thickness Min The minimum thickness measured in the scan data.
Thickness Max The maximum thickness measured in the scan data.

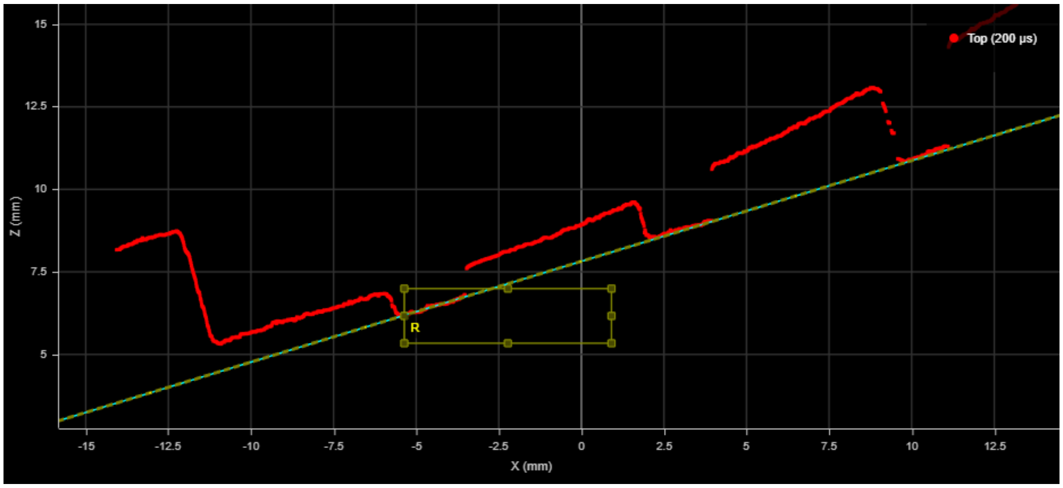
Profile Transform

The Profile Transform creates a new profile based on the coordinate system of geometric features the tool uses as input and/or fixed input values. The tool accepts a Line geometric feature (rotating the profile so that the line is parallel to the X axis) and/or a Point geometric feature (using it as the X and Z origin).

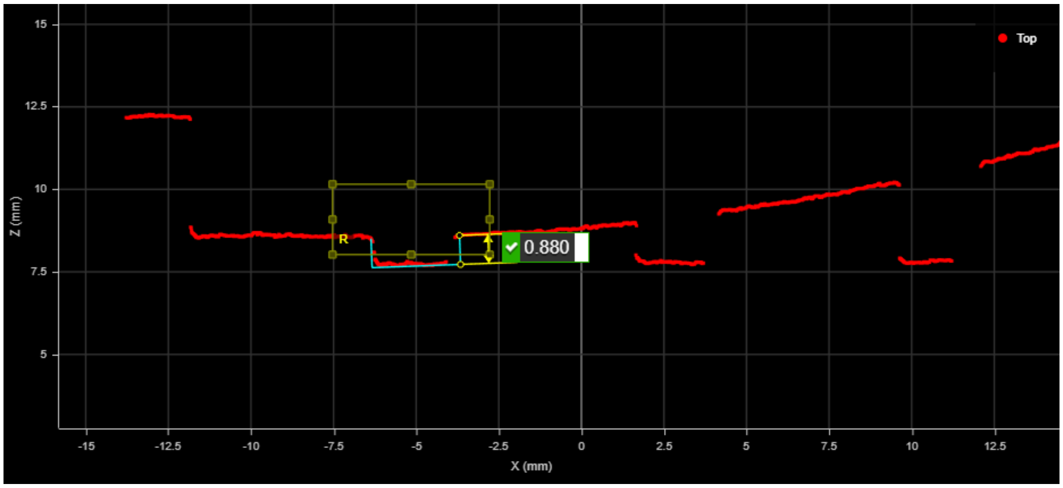
For example, in the following, if you want to measure the characteristics of the first groove on the left, you can use the tool to rotate the profile.



First, a Profile Line tool is added and configured on the bottom of the groove. The tool is configured to output a line geometric feature.



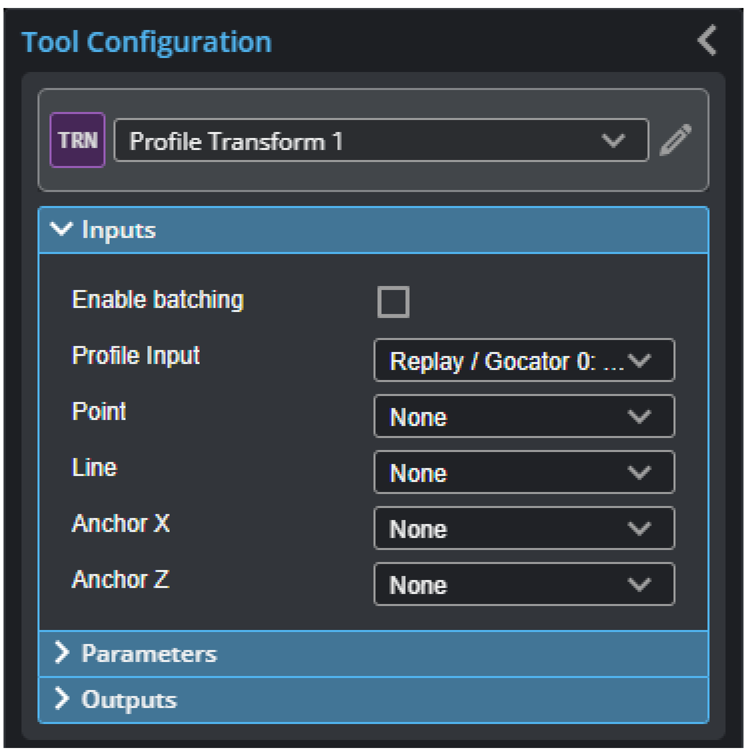
A Profile Transform tool takes the line geometric feature as input, and the transformed profile from that tool is used as input for a Profile Groove tool, which measures the groove's characteristics:



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



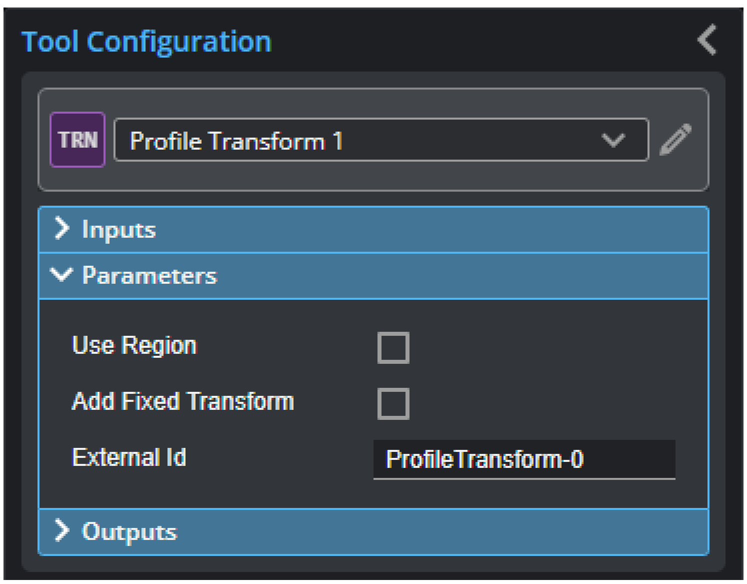
	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Profile Input	The data the tool applies measurements to or processes.
Point	The Point geometric feature the tool uses to offset a profile to an X and Z origin of 0.
Line	The Line geometric feature the tool uses to position (X and Z) and to rotate a profile Y rotation).
Anchor X or Anchor Z	The X or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



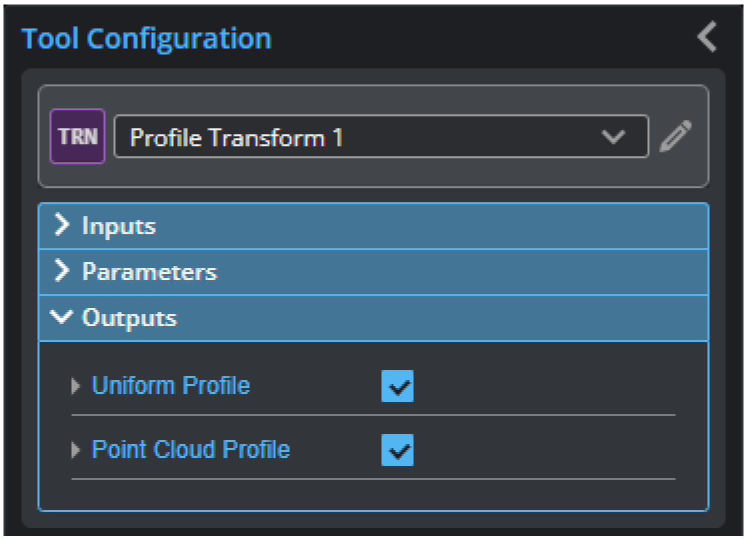
Parameters

Parameter	Description
Use Region	Whether the tool should limit the transformed profile that it outputs to a user-defined region. When enabled, the tool displays a **Region** parameter you use to define the region. If this option is not enabled, the tool transforms the entire profile.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Add Fixed Transform	Enables **X Offset**, **Y Offset**, and **Angle** parameters you can use to manually set a

Parameter	Description
	transformation. Useful if you know the profiles in the scan data will always be in a certain position or orientation.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.

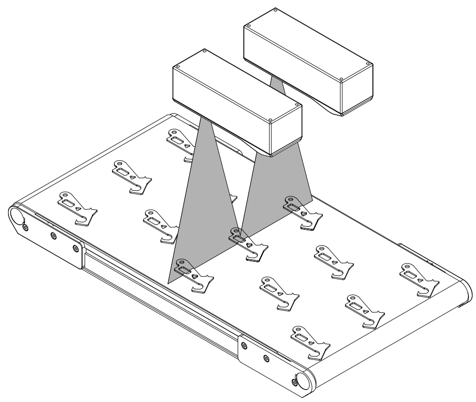


Data

Type	Description
Profile Uniform	The transformed profile.
Profile Point Cloud	The outputs will contain the same profile data, but the type (uniform vs. point cloud) will differ, so any Profile tool will be able to take this tool's output as input.

Surface Measurement

Surface measurement involves capturing scan data, optionally identifying discrete objects, and measuring properties of the surface or the objects, such as the volume of the object or the height at a certain position of the object. All volumetric tools have the ability to operate either on the entire surface or the full object, or within a region of interest at a certain position in relation to the surface or an object.



Multiple measurements can be performed on the entire surface or each discrete object, limited only by the available CPU resources.

Isolating Parts from Surface Data

Gocator lets you isolate and then measure parts in different ways. This section describes some of the differences between these methods.

- For line profiler sensors, on the **Scan** page, by enabling continuous Surface generation and configuring the part detection settings. (For more information, see *Surface Generation* on page 199.)
- By using the Surface Blob or Surface Segmentation measurement tools (for more information, see *Surface Blob* on page 455 and *Surface Segmentation* on page 635).
- For snapshot sensors (and G2, G4, or G5 sensors when Surface generation is *not* set to Continuous), by using the Surface Part Detection tool.
- For line profile sensors, by using the Profile Part Detection tool.

The table below lists several differences between the different methods. A key difference however is that Scan page part detection extracts scan data that is identified as a "part" and outputs it as a *separate frame*. This lets you use any measurement tool on parts individually. Note however that parts must be clearly separated and be relatively consistently spaced for the part detection algorithm to separate the parts. In general, if you can successfully isolate parts using part detection, use this method rather than the Surface tools.

With Surface Blob and Surface Segmentation, on the other hand, areas are not extracted as individual frames. They are extracted either as individual outputs or as elements in an array. (For more information on arrays, see *Arrays, Batching, and Aggregation* on page 242.) These tools also provide measurements such as length, width, and area, which lets you handle common pass/fail needs.

Main Differences Between Part Detection, Surface Blob, and Surface Segmentation

	Part Detection	Surface Blob	Surface Segmentation
Allows output of individual surfaces to separate frames	Yes	No	No
Allows separating touching objects	No	Yes - Limited Through Open filter, some connections between parts can be separated, but the control is more limited than with Surface Segmentation.	Yes
Supports background present	Yes Height threshold must be set above or below the background	Yes Height threshold must be set above or below the background	Yes
Supports background with significant tilt or intensity gradient	No Fixed height threshold is used	No Fixed height threshold is used	Yes Adaptive threshold is used
Integrated Width/Length/Area measurements	N/A	Yes	Yes
Includes circularity and convexity filtering	No	Yes	No
Fast operation	Yes	Yes	No
Finds objects above or below background	Yes	Yes	Yes But requires careful region placement

Surface Align Ring

This tool is only intended for performing a high-accuracy alignment of ring and partial ring layouts with G2 sensors, using recorded scan data. LMI recommends performing alignment using the routines available on the **System** page, under **Alignment**. For information on aligning sensors using this tool, see *Ring Layouts (Surface Align Ring Tool)* on page 182.

⚠	Performing alignment using the Surface Align Ring or Surface Align Wide tools (which results in 6 degrees of freedom) involves *considerable* setup effort. First, the 6 DoF alignment targets are more difficult to manufacture than an alignment bar and require a very high degree of accuracy; 3D printed alignment targets are not usually sufficiently accurate. Second, the alignment tools have many parameters that must be properly configured to successfully perform an alignment.

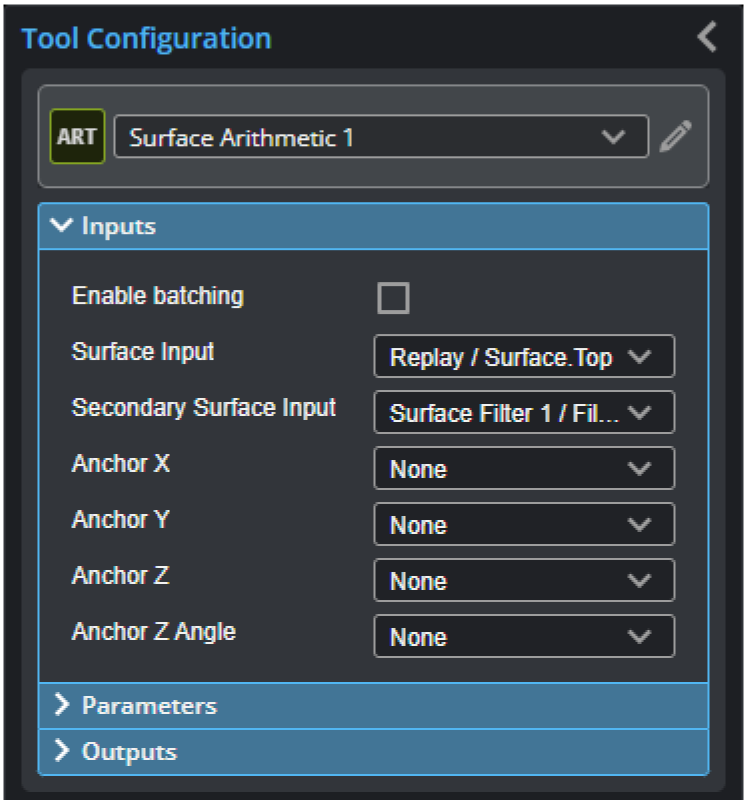
Surface Align Wide

This tool is only intended for performing a high-accuracy alignment of side-by-side (wide) layouts with G2 sensors with six degrees of freedom, using recorded data. LMI recommends performing alignment using the routines available on the **System** page, under **Alignment**. For information on using the tool to trouble-shoot alignment, see *Wide Layouts (Surface Align Wide Tool)* on page 160.

Surface Arithmetic

The Surface Arithmetic tool lets you perform various operations on a pair of surfaces. For example, you can use the tool to perform dynamic masking from frame to frame. The tool performs bitwise operations (AND or OR) on the corresponding data points in the source surfaces, and also combines height and intensity data with add, subtract, average, and mask operations.

Inputs



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

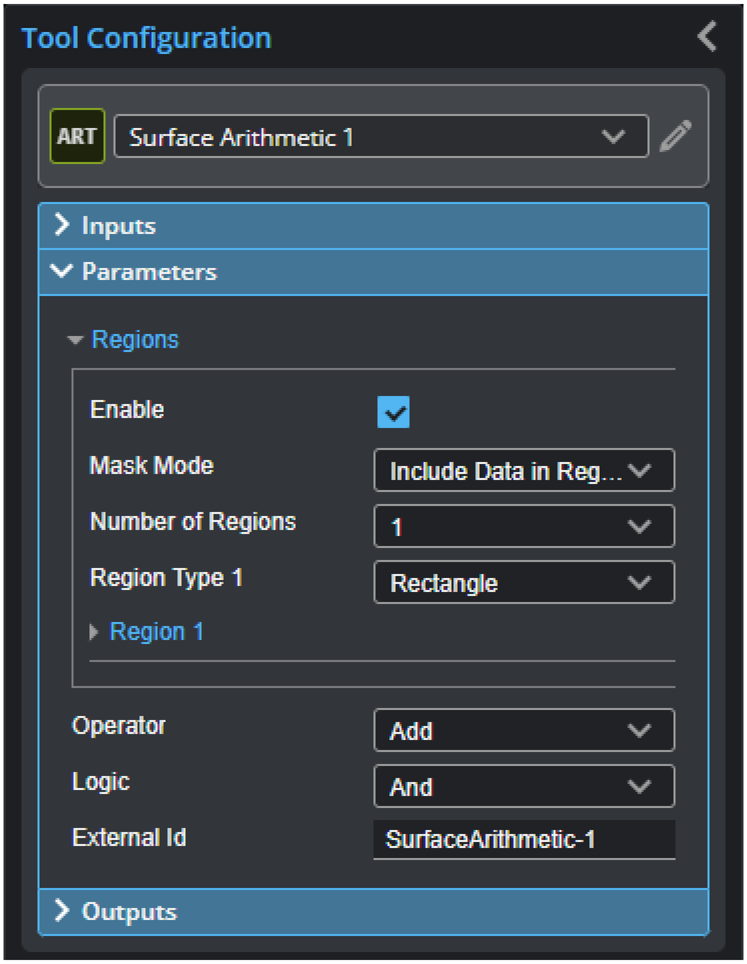
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Secondary Surface Input	The secondary input the tool uses to perform arithmetic.
Anchor X	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Name	Description
Anchor Y Anchor Z	
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.

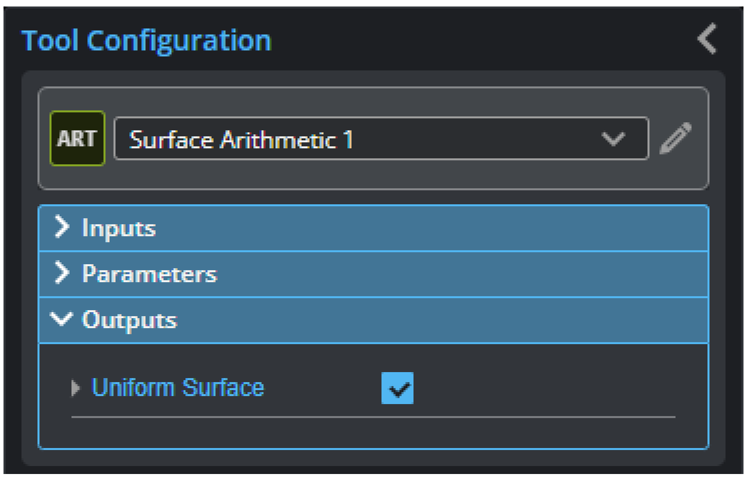


Parameters

Parameter	Description
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and

Parameter	Description
Region {n}	"flexible" regions, see *Regions* on page 250.
Operator	One of the following: **Add**: Adds the height values of the corresponding data points in the two sources. **Subtract**: Subtracts the height values of the corresponding data points in the two sources. **Average**: Averages the height values of the corresponding data points in the two sources. **Mask**: Uses the secondary source as a mask.
Logic	Performs bitwise-operations on the source and secondary source surface data. One of the following: **And** or **Or**.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



All outputs provide an external ID (available by expanding the output in the Outputs panel) for optional use in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Data

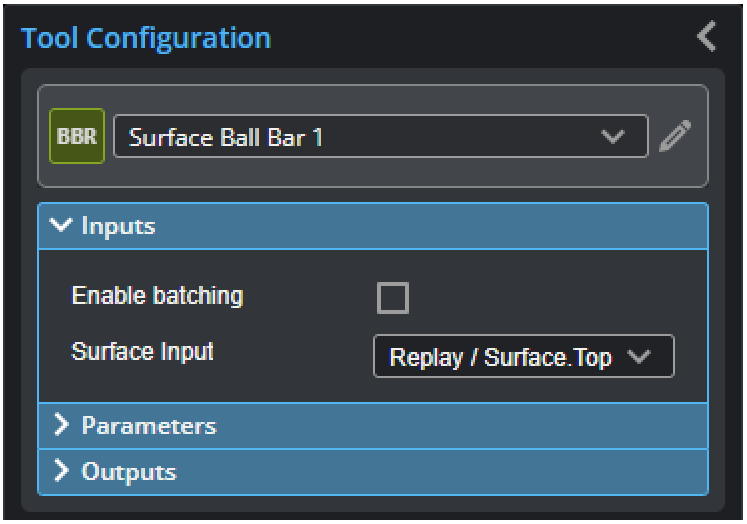
Type	Description
Uniform Surface	The processed surface data.

Surface Ball Bar

The Surface Ball Bar tool returns measurements useful for calibrating systems using a ball bar, particularly systems that include a robot.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

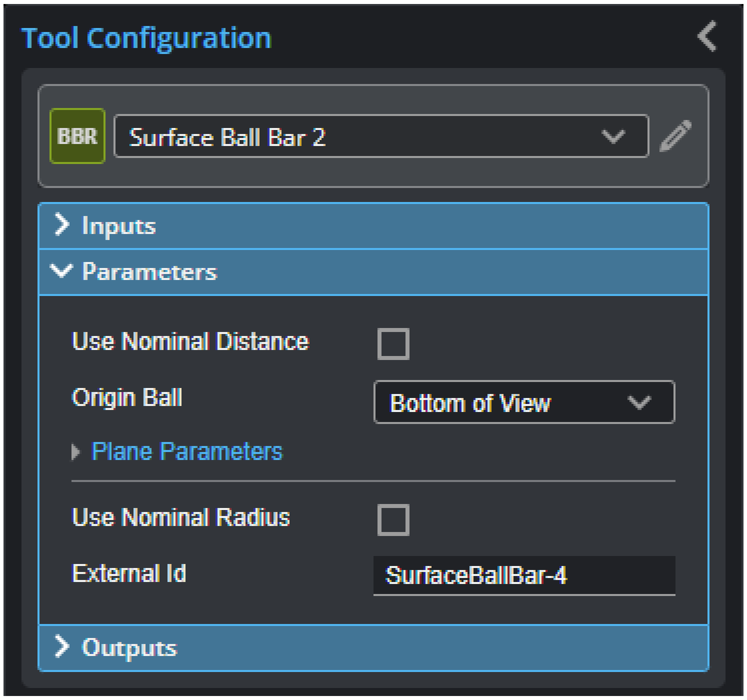


Inputs

Name	Description
Enable Batching	Leave this setting unchecked.
Surface Input	The data the tool applies measurements to or processes.

Parameters

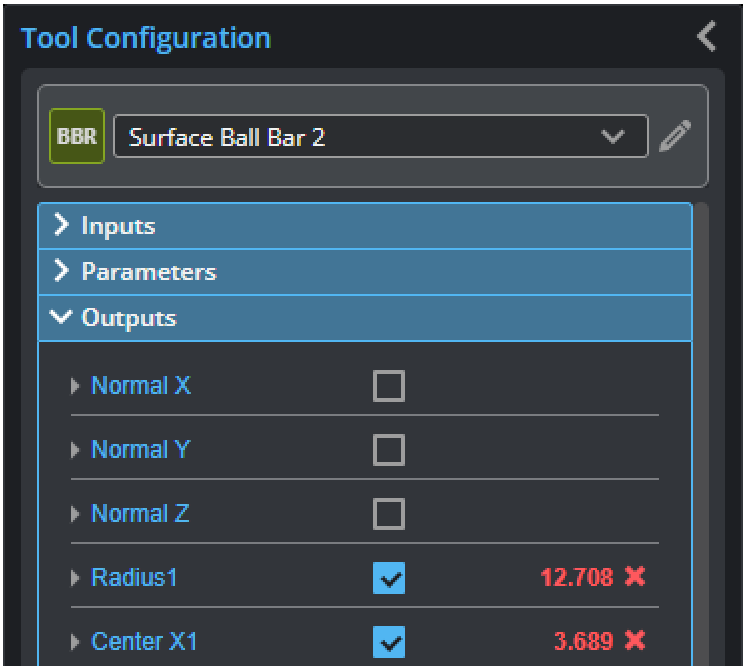
You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Use Nominal Distance	When enabled, displays **Nominal Distance** and **Distance Tolerance** settings. Set these to the distance between the balls of the ball bar (refer to the specifications of the ball bar) and the tolerance you need. This can be useful to ensure invalid results due to false or inaccurate detection are rejected.
Origin Ball	Determines which ball is used as the origin. The **Bottom of View** option selects the ball at the bottom of the data viewer in the web interface.
Plane Parameters	Enables advanced plane settings. For UR integration, you should leave the settings at their default. These parameters allow ensuring the plane detection is accurate and robust to variations.
Use Nominal Radius	When enabled, displays **Nominal Radius {n}** settings. Set these to the radius of the balls of the ball bar (refer to the specifications of the ball bar) and the tolerance you need. This can be useful to ensure invalid results due to false or inaccurate detection are rejected. The tool uses these as a starting point and will not necessarily reject targets based on these.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



Measurements

Measurement
Normal X / Y / Z These measurements return the X, Y, and Z components of the normal vector of the surface surrounding the calibration target.
Radius1 **Radius2** These measurements return the radius of each ball.
Center X1 / Y1 / Z1 **Center X2 / Y2 / Z2** These measurements return the X, Y, and Z positions of the centers of the spheres fitted to the balls. Ball 1 (Center X1 / Y1 / Z1) is always used as the origin. (Corresponds to the values returned in Tx / Ty / Tz.)
Distance 3D The direct distance between the centers of the spheres fitted to the balls.
Ix / Iy / Iz **Jx / Jy / Jz** **Kx / Ky / Kz** These measurements return the X, Y, and Z components of the I, J, and K unit vectors defining the coordinate system orientation.
Tx / Ty / Tz These measurements return the X, Y, and Z components of the translation vector defining the coordinate system origin location.

Data

Type	Description
Difference Surface	The difference between the scan data and the provided nominal dimensions. Used for diagnostics.

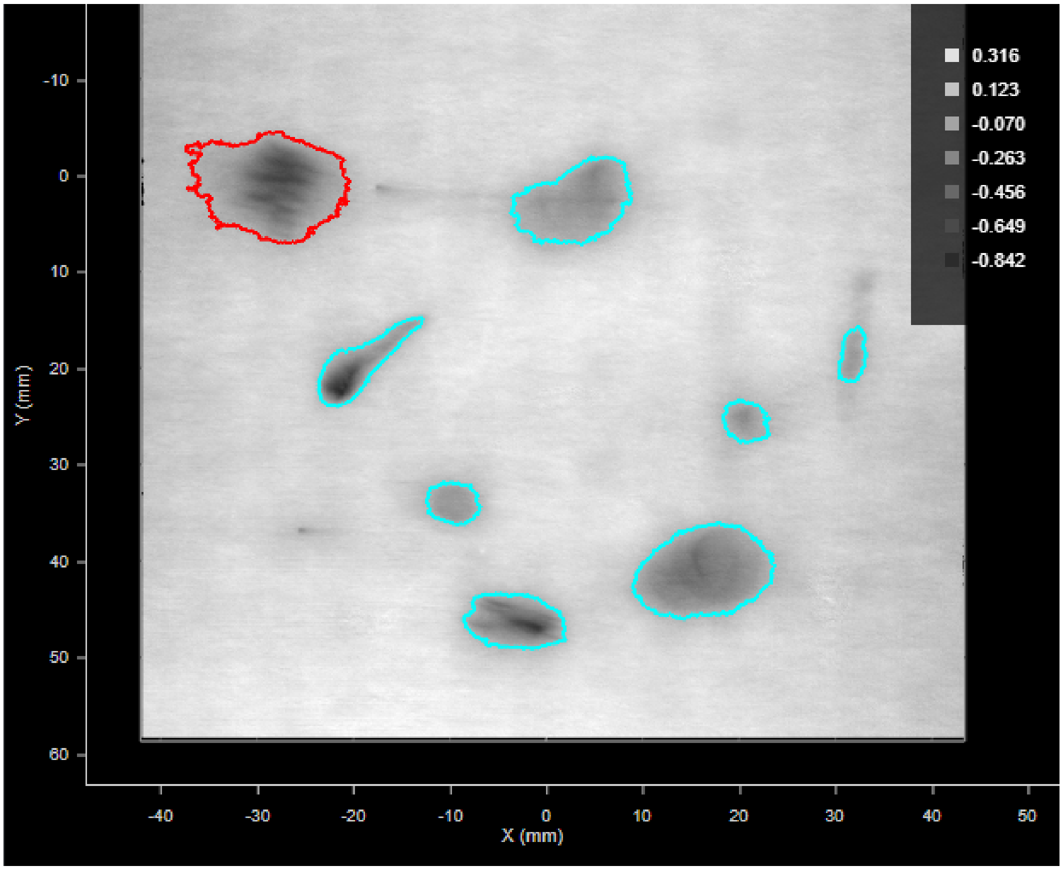
Surface Blob

The Surface Blob tool lets you detect surface defects, such as uneven or excess material, gouges, or blemishes, on a relatively uniform or flat background, in either 3D height map data or intensity data. It can also extract targets from the surface. The tool optionally lets you set its height threshold relative to a user-defined reference region. It also lets you use a reference plane to correct for a minor tilt of the target surface (up to 10 degrees); this lets you detect low or shallow defects that would otherwise not be detectable due to a tilt.

> The Surface Blob tool provides functionality similar to the Surface Segmentation tool. For information on the Surface Segmentation tool, see *Surface Segmentation* on page 635.
>
> For a comparison of these tools and the part detection capabilities you can configure on the **Scan** page, see *Isolating Parts from Surface Data* on page 444.

The tool first filters data based on a height or intensity threshold (above or below it), and then uses configurable morphological operations to better isolate parts. Finally, the tool uses various size- and shape-based filters that let you exclude or include the expected defects or the targets you need (potential blobs).

The tool lets you configure the maximum number of "blobs" to output, and returns the total blob count, and for each blob, the X and Y center, the width and length, and the area. The center point of each blob is available as a geometric feature.



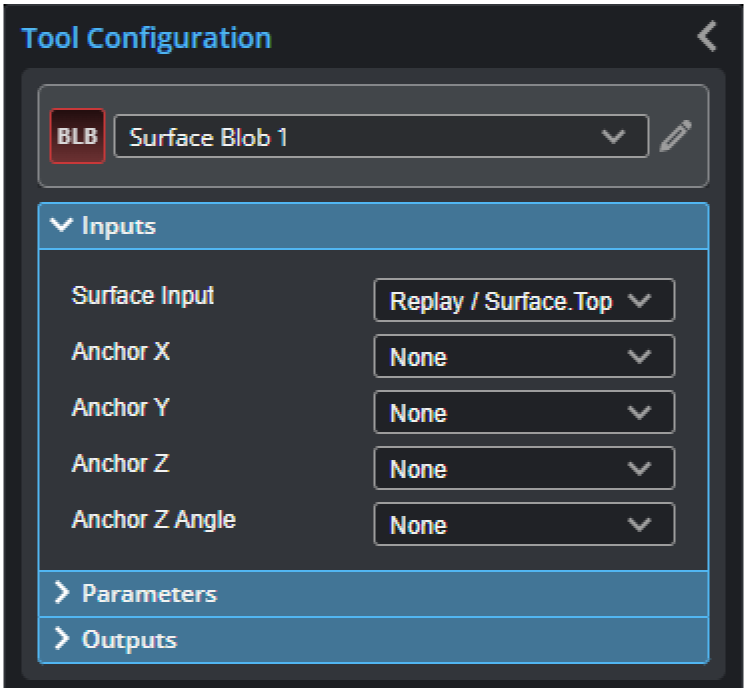
Several dents outlined by the tool on a surface in cyan. The currently selected blob is outlined in red. (Grayscale heightmap mode is used to better see the outlines.)

Note that knowing the rough size and shape of the kinds of detects you expect is important when you are configuring the open and close kernels and the tool's filters.

> When **Enable Array Output** is checked, the tool outputs the blobs in an array called Blob Surface. Be *sure* to enable Blob Surface in the **Outputs** section after enabling this parameter.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs



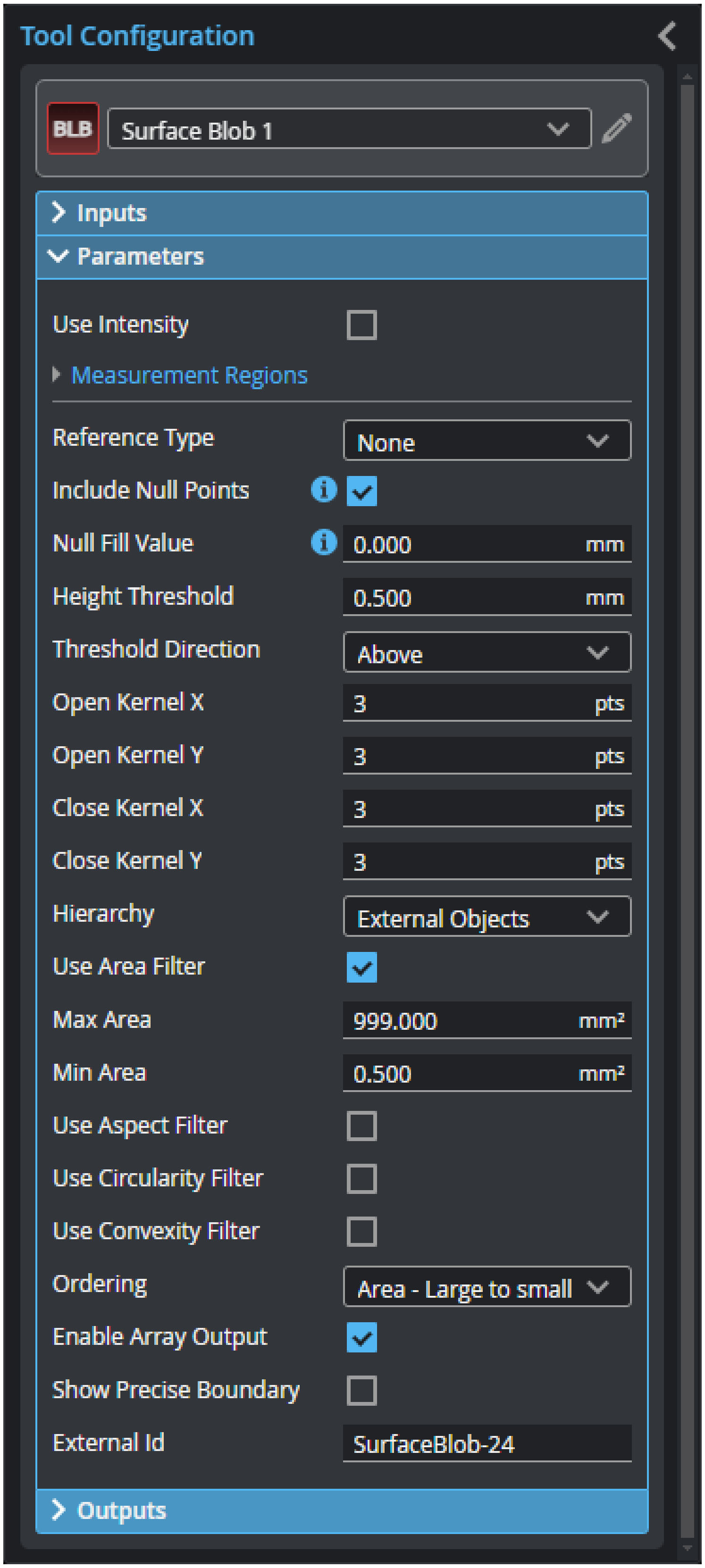
> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

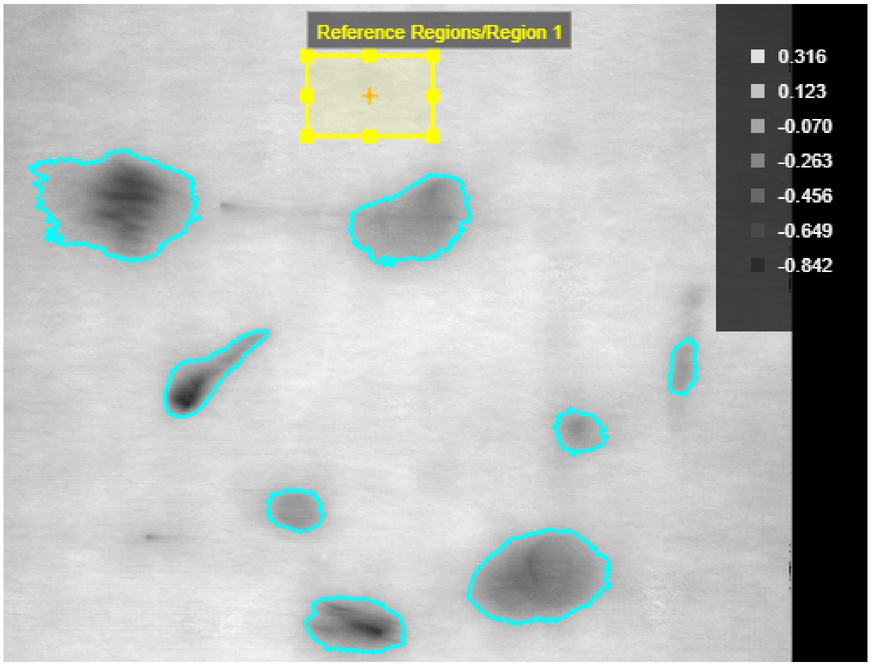
Name	Description
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.
Plane	The Plane geometric feature the tool uses when **Reference Type** is set to Reference Plane.

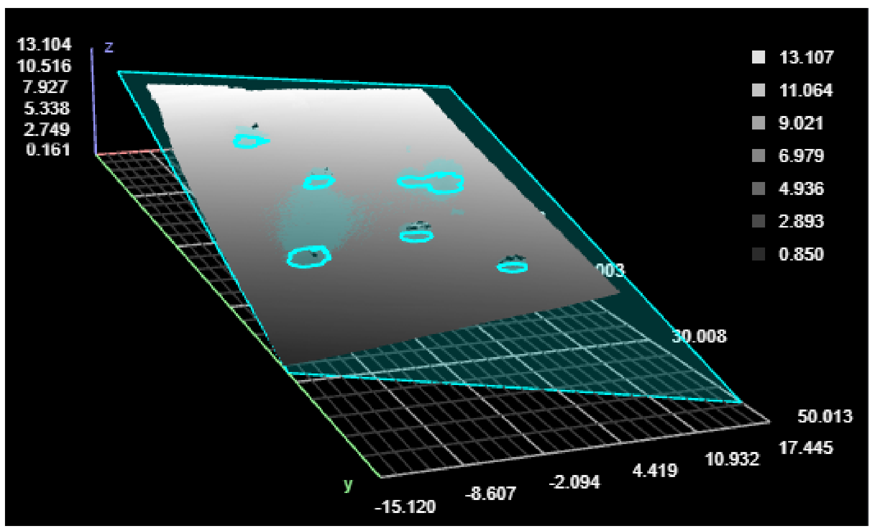
Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.

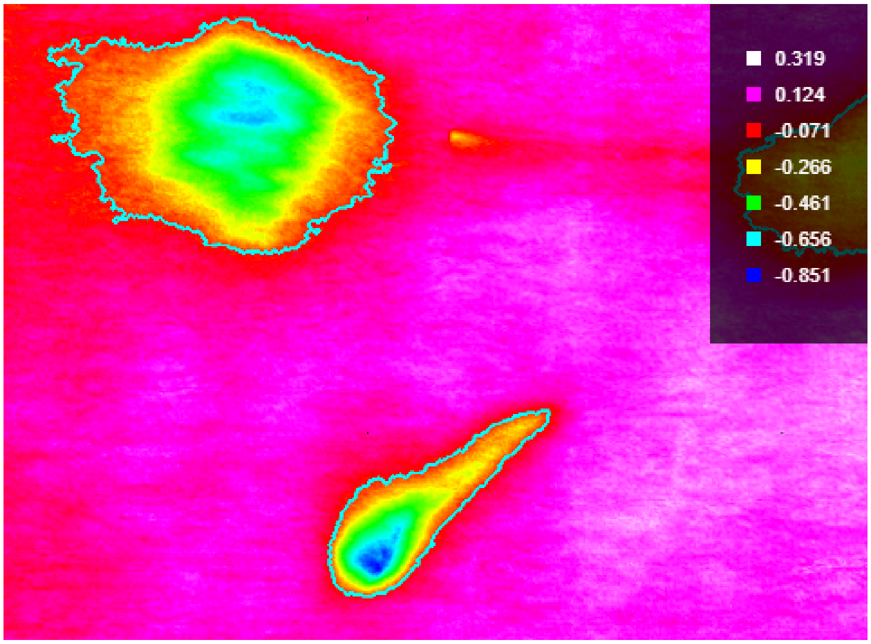
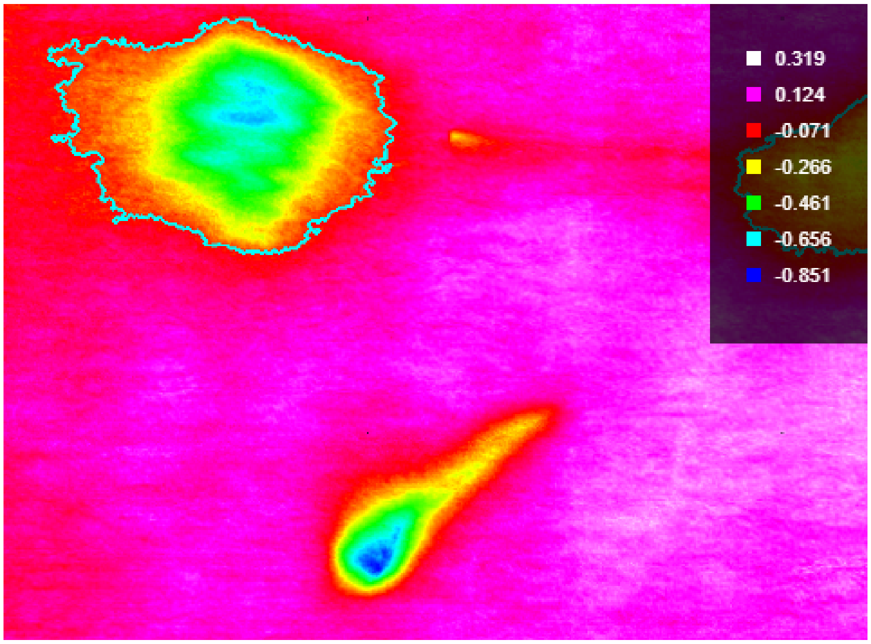

Tool Configuration
BLB
Surface Blob 1
Inputs
Parameters
Use Intensity
Measurement Regions
Reference Type
None
Include Null Points
Null Fill Value
0.000
mm
Height Threshold
0.500
mm
Threshold Direction
Above
Open Kernel X
3
pts
Open Kernel Y
3
pts
Close Kernel X
3
pts
Close Kernel Y
3
pts
Hierarchy
External Objects
Use Area Filter
Max Area
999.000
mm²
Min Area
0.500
mm²
Use Aspect Filter
Use Circularity Filter
Use Convexity Filter
Ordering
Area - Large to small
Enable Array Output
Show Precise Boundary
External Id
SurfaceBlob-24
Outputs

Parameters

Parameter	Description
Use Intensity	If enabled, the tool uses intensity data instead of heightmap data. Only available if **Acquire intensity** is enabled on the **Inspect** > **Scan** page during scanning; for more information, see *Scan Modes and Intensity* on page 195.
Measurement Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Reference Type	Provides three options: None, Reference Region, and Reference Plane. If the reference type is set to None, the **Height Threshold** setting is absolute (relative to zero). For the Reference Region and Reference Plane options, see the descriptions of the **Reference Region Type** and **Reference Plane** parameters below.
Reference Region Type	If you set **Reference Type** (see above) to Reference Region, the tool displays a drop-down that lets you configure the reference region. (For details, see *Flexible Regions* on page 253.) The tool calculates an average height or intensity of the data in the reference region. **Height Threshold** is relative to this value. For example, in the following, blobs are detected using a relative height threshold of -0.2 mm, relative to the average in the reference region:  For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.

Parameter	Description
Reference Plane	If you set **Reference Type** (see above) to Reference Plane, the tool uses the specified plane geometric feature to correct for a tilt of the target. Note however that using a reference plane to correct the tilt distorts the scan data: it sheers the data by the same angle as the tilt. The maximum tilt angle with which you can use the tool therefore depends on how much sheer angle you can tolerate in your application (which can effect the tool's ability to detect blobs). Typically, you add and configure a Surface Plane tool to generate a plane (for more information, see *Surface Plane* on page 609). For information on geometric features, see *Geometric Features* on page 262. For applications where sheer distortion can't be tolerated, use Surface Transform to correct the tilt (see *Surface Transform* on page 672), and use the latter tool's output as the input for Surface Blob. For example, in the following, despite the overall tilt of the target, the tool detects the flaws on the surface. (Note the gradient of the heightmap colors, indicating a height difference of roughly 9 millimeters between the lower and higher areas near the dents on the target's surface.) 
Include Null Points	Indicates whether null points (points where no height or intensity value is available, due to dropouts or regions outside of the measurement range) are filled with the value in **Null Fill Value** as a general "background level" or to fill gaps to aid in isolating blobs. If **Use Intensity** is enabled, the value in **Null Fill Value** is an intensity.
Height Threshold Intensity Threshold	The threshold above or below which data is considered for being a blob. Use the **Threshold Direction** setting to determine whether data above or below the threshold is considered. If **Use Intensity** is enabled, this setting is **Intensity Threshold**. Otherwise, it is named **Height Threshold**.
Threshold Direction	Determines whether data above or below the threshold is considered as being a

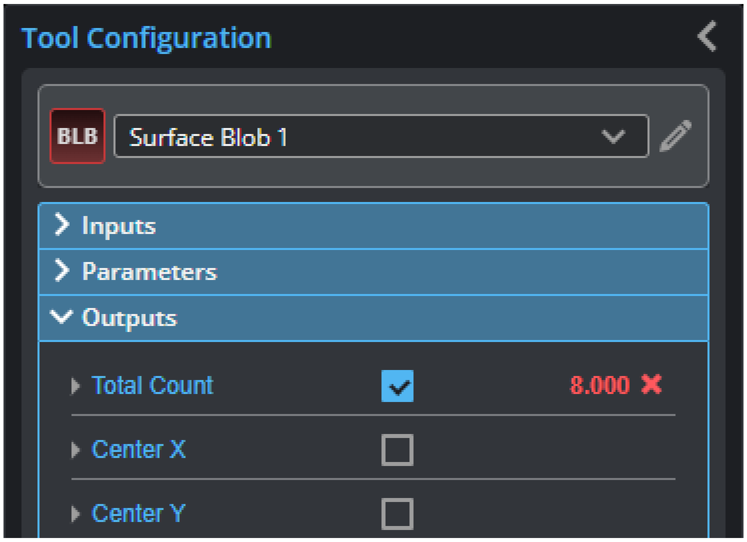
Parameter	Description
	blob. **Below**: The **Height Threshold** value is the maximum that will be considered as part of a blob (for example, a dent below the surrounding surface). **Above**: The **Height Threshold** value is the minimum that will be considered as part of a blob (so a raised feature).
Open Kernel X Open Kernel Y	The X and Y kernel size, respectively, for morphological opening to remove small areas of data. Use these settings, for example, to remove bridges between areas to properly isolate them or to remove small areas entirely (perhaps caused by noise). Use different values of X and Y to use a non-rectangular filter to adapt the kernel to the kinds of unwanted data you see in the scan data.
Close Kernel X Close Kernel Y	The X and Y kernel size, respectively, for morphological closing to fill in holes smaller than the specified kernel size. Use these settings, for example, to fill small areas within potential blobs that may be caused by drop-outs. Use different values of X and Y to use a non-rectangular filter to adapt the kernel to the kinds of holes you see in the scan data.
Hierarchy	Provides options to let you find either external objects only or all objects. **External Objects** Use this option to ignore smaller objects in larger objects. Only the outermost objects are returned. **All objects** Use this option to include smaller objects in larger objects. **Holes** Use this option to get holes inside objects. **All Contours** Use this option to get all detected contours.
Use Area Filter	If **Use Area Filter** is enabled, the tool applies an area filter to potential blobs using the values in **Max Area** and **Min Area**.
Use Aspect Filter	If **Use Aspect Filter** is enabled, the tool applies an aspect filter (ratio of length and width) to the rotated bounding box that would encapsulate the area, using the values in **Max Aspect** and **Min Aspect**. For example, the lowest dent in a surface is included as a blob if the aspect values are set to 1 and 0.350, respectively (the rotated bounding box encapsulating would be 13.461 mm x 4.782 mm).

Parameter	Description
	 In the following, the same dent is excluded if **Min Aspect** is set to a value greater than 0.354. 
Use Circularity Filter	If **Use Circularity Filter** is enabled, the tool applies a circularity filter to potential blobs to measure how close to a circle the blob is, using the values in **Max Circularity** and **Min Circularity** . Circularity is determined from area within the contour of the blob and the perimeter of its contour. With increasing perimeter for the same area, circularity is reduced.

Parameter	Description
Use Convexity Filter	If **Use Convexity Filter** is enabled, the tool applies a convexity filter to potential blobs, using the values in **Max Convexity** and **Min Convexity** . Convexity is defined as the (Area of the Blob / Area of its convex hull), and "convex hull" of a shape is the tightest convex shape that completely encloses the shape.
Ordering	Orders the measurements, features, and surface data of the individual blobs output by the tool. One of the following: • Position - X increasing • Position - X decreasing • Position - Y increasing • Position - Y decreasing • Area - Large to small • Area - Small to large
Enable Array Output	When **Enable Array Output** is unchecked, the tool outputs each blob as an individual Surface output. When **Enable Array Output** is checked, the tool outputs the blobs in an array called Blob Surface. Be *sure* to enable Blob Surface in the **Outputs** section after enabling this parameter. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Number of Blob Outputs	Determines the number of blobs the tool outputs as measurements, features (center points of blobs), and surface data. Currently limited to 200 blobs. If **Enable Array Output** is checked, this parameter is hidden.
Show Precise Boundary	Displays the boundaries of the blobs so that they appear on the Surface data, rather than floating above the Surface. This makes it easier to evaluate the blob boundaries in relation to the Surface data. This setting does not affect measurements or other outputs. Disable this setting for maximum performance.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

All outputs provide an external ID (available by expanding the output in the Outputs panel) for optional use in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

If **Enable Array Output** is checked, the tool outputs arrays for the center position, length, width, and area measurements, as well as the Center Point output.

If **Enable Array Output** is unchecked, the tool outputs individual, numbered measurements. Use the **Number of Blob Outputs** setting to determine the number of measurements listed in the **Measurements** tab.

Measurements

Measurement
Total Count Returns the total number of blobs identified, based on the tool's parameters.
Center X {n} / Center X **Center Y {n} / Center Y** The X and Y positions of the center of mass of a blob extracted from the surface.
Average Z {n} / Average Z The average Z positions of a blob extracted from the surface.
Min Z {n} / Min Z **Max Z {n} / Max Z** The minimum and maximum Z positions of a blob extracted from the surface.
Width {n} / Width **Length {n} / Length** The length and width of the rotated bounding box that encapsulates the blob extracted from the surface.

Measurement
These are always the major and minor axis of a blob, respectively.
Area {n} / Area The area of a blob. The area is calculated using the contour of the blob and resampling. For this reason, areas calculated using the Surface Volume tool will produce different measurements; for more information, see *Area* on page 691.

Features

Type	Description
Center Point {n} / Center Point	The point representing the center of a blob.

For more information on geometric features, see *Geometric Features* on page 262.

Data

Type	Description
Diagnostic Surface	Surface data you can use to evaluate the impact of the tool's parameters, after filters are applied, to properly separate the areas corresponding to the defects or targets you need to detect.
Surface {n} / Blob Surface	Surface data corresponding to each blob. When the tool is configured to output arrays, the blob surfaces are in Blob Surface.

Surface Bounding Box

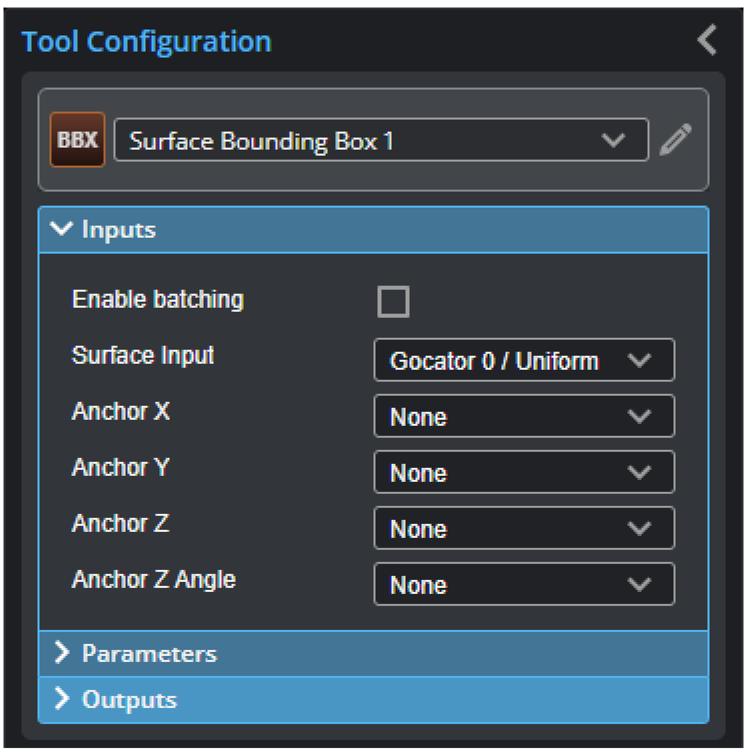
The Bounding Box tool provides measurements related to the smallest box that contains the *scan data* from a part (for example, X position, Y position, width, length, etc.). The tool also lets you get the height of bounding box relative to the Z origin (typically the conveyor on which the target is sitting). This lets you determine, for example, the height of a box or other container on the conveyor as part of a product packaging process. Various settings let you easily filter out noise that can affect height, width, and length measurements.

A bounding box can be vertical or rotated. A vertical bounding box provides the absolute position from which the Position centroids tools are referenced.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

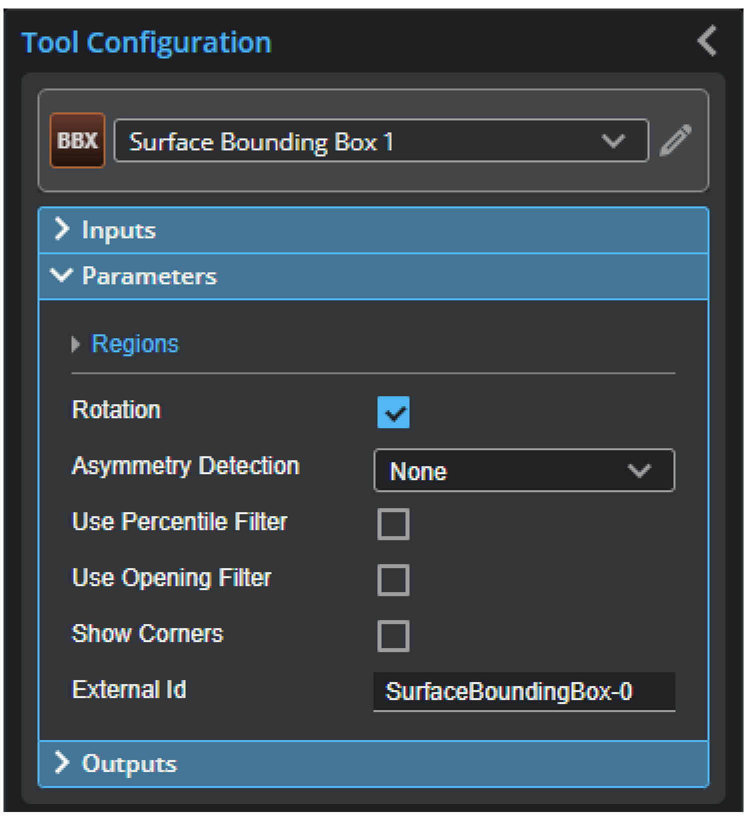
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array,

Name	Description
	other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

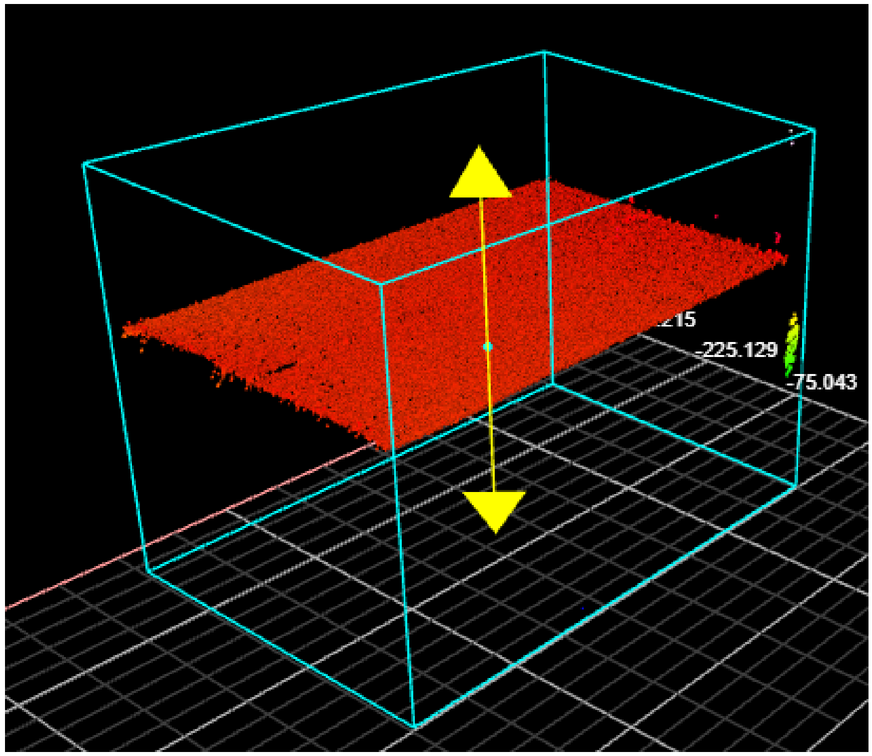
Parameters

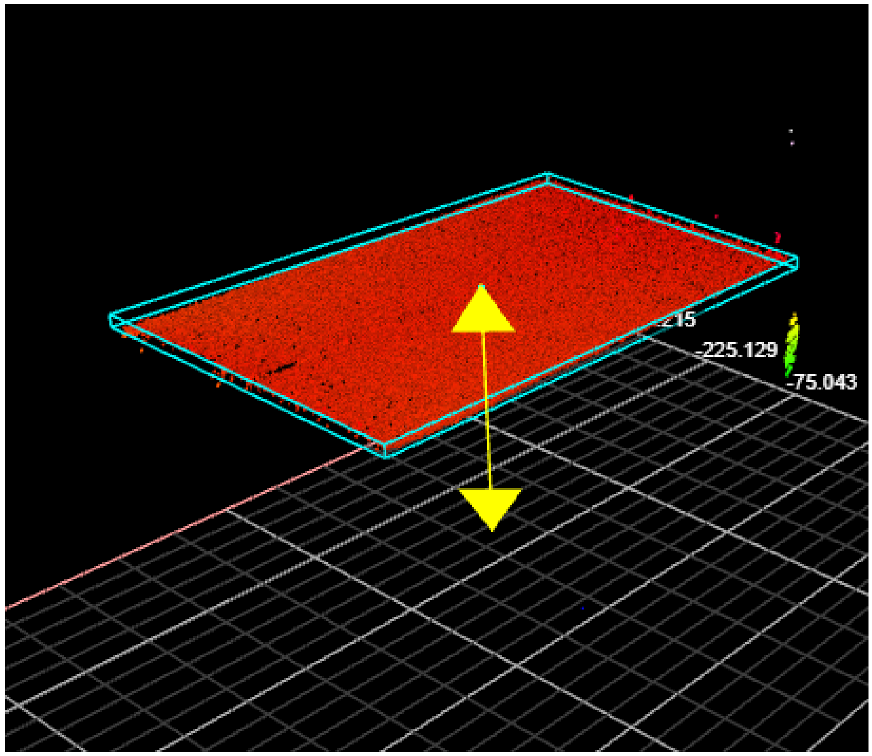
You configure the tool's parameters in the expandable **Parameters** section.

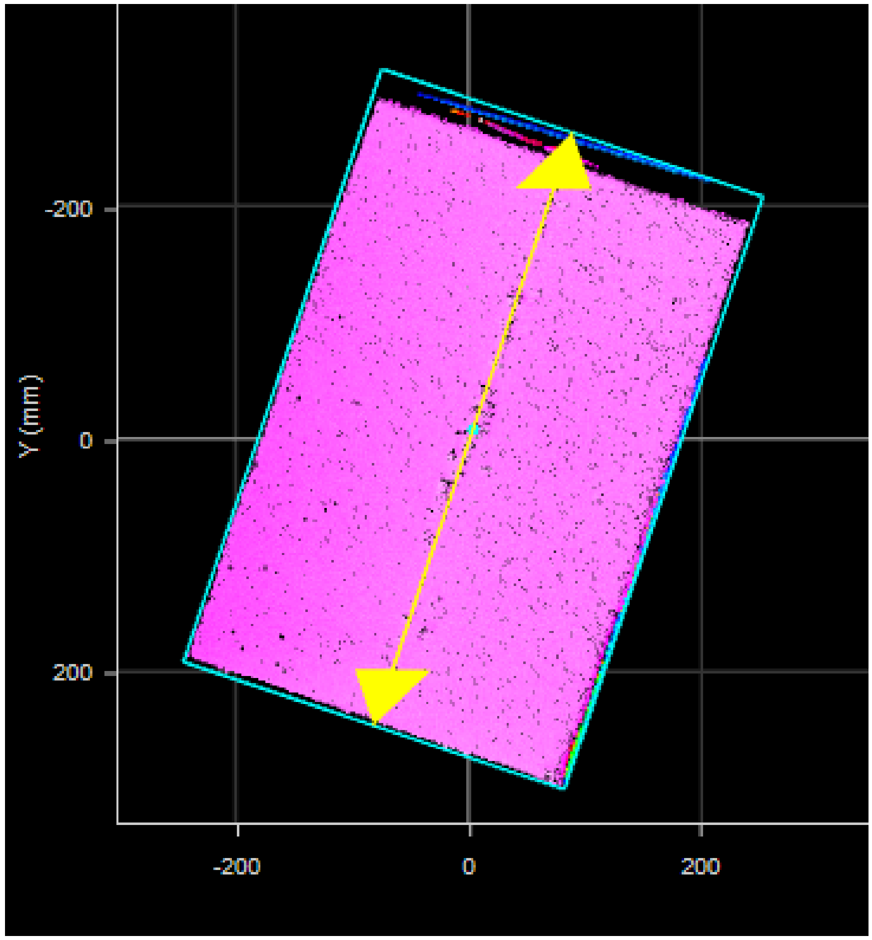


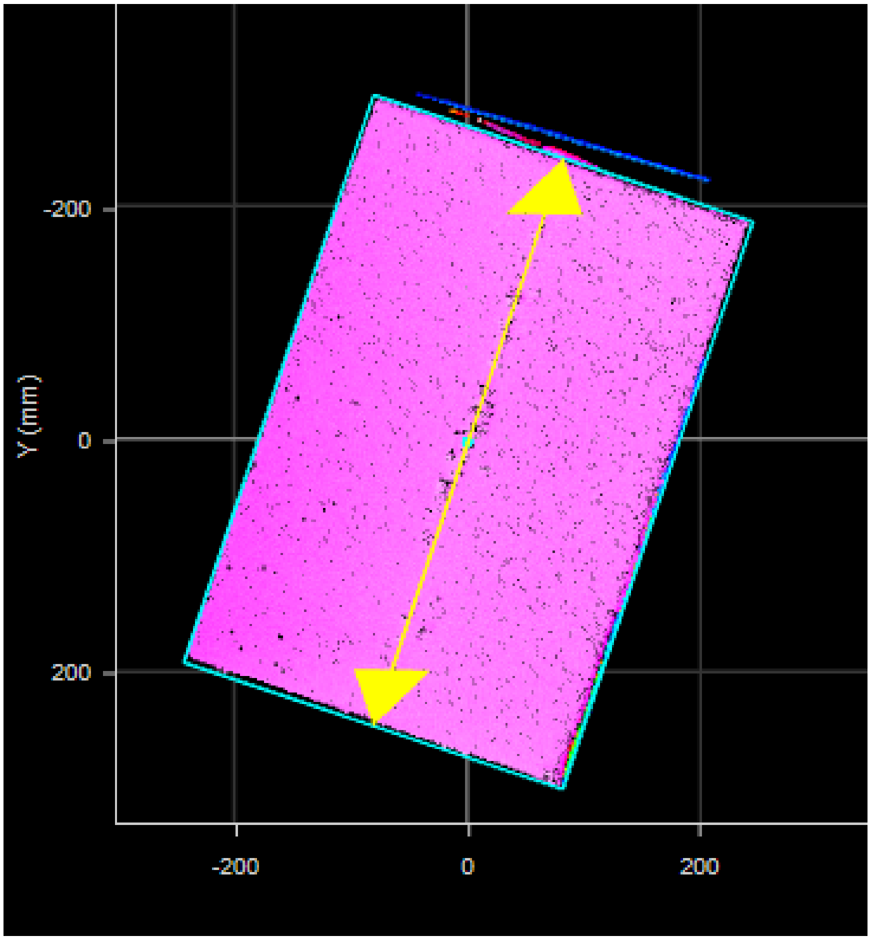
Parameters

Parameter	Description
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Region Type 1 Region 1	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Rotation	A bounding box can be vertical or rotated. A vertical bounding box provides the absolute position from which the part's Position centroid measurements are referenced. Check the **Rotation** setting to select rotated bounding box. When this setting is enabled, an Asymmetry Detection setting is displayed.
Asymmetry Detection	Resolves the orientation of an object over 360 degrees. The possible values are: 0 – None 1 – Along Major Axis 2 – Along Minor Axis This setting is only visible if **Rotation** is checked.
Use Percentile Filter	Limits the bounding box to data points along the Z axis between the values you set in **High Percentile** and **Low Percentile**, which are displayed when you choose this option. Use this setting to obtain more "robust" height measurements. This setting is useful to exclude noise that would otherwise cause inaccurate height measurements. For example, in the following scan of a box, without excluding a small percentage of the highest data points, data points caused by noise to the upper right produces an inaccurate height measurement (from the Z origin) of the box of 406.457 mm.

Parameter	Description
	 When **High Percentile** is set to 99%, the highest 1 percent of data points is excluded from the placement of the bounding box, and an accurate height of the target box of 270.477 mm is returned.

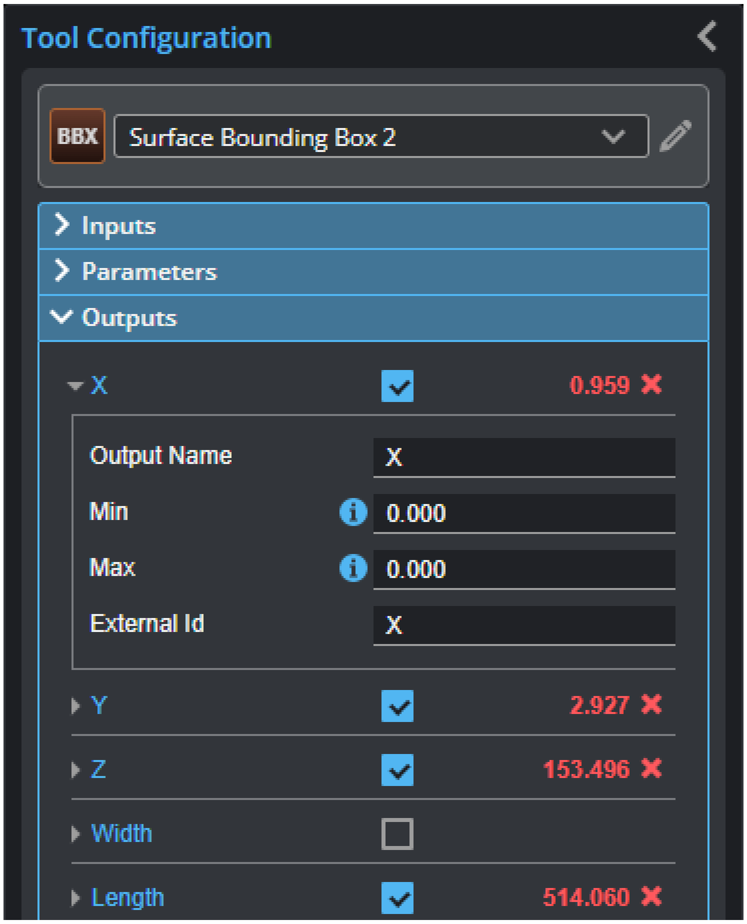
Parameter	Description
	
High Percentile Low Percentile	See **Use Percentile Filter** above.
Use Opening Filter	When enabled, this setting lets you set the value of **Kernel Size** for an *opening* morphological operation applied to the scan data on the XY plane, letting you achieve "robust" width and length measurements. This filter removes noise or small objects from scan data, while keeping the shape and size of the larger objects in the scan data. For example, in the following, noise along the edge at the top of the data viewer results in an inaccurate length measurement.

Parameter	Description
	 When the filter is set to an appropriately sized kernel (here, 11 points), the noise is excluded from the calculation of the bounding box, and an accurate length is returned.

Parameter	Description
	 Use the Diagnostics Surface on the **Data** tab to evaluate the impact of the open filter, to avoid removing too much data.
Kernel Size	See **Use Opening Filter** above.
Show Corners	When this setting is enabled, the tool outputs a Point geometric feature for each corner of the box.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

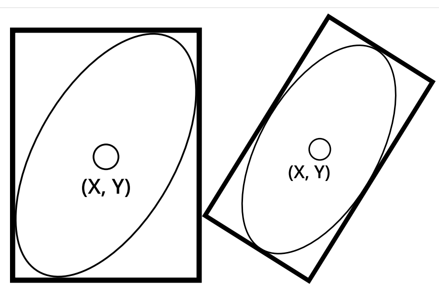
Most tools provide measurements, geometric features, or data as outputs.

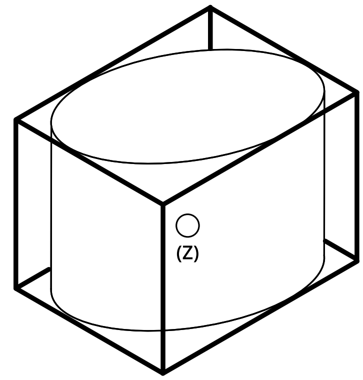
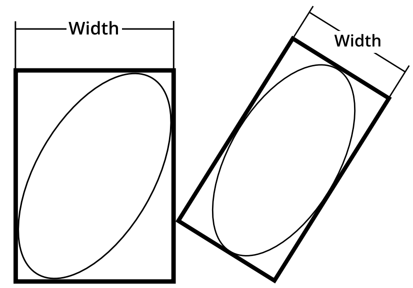
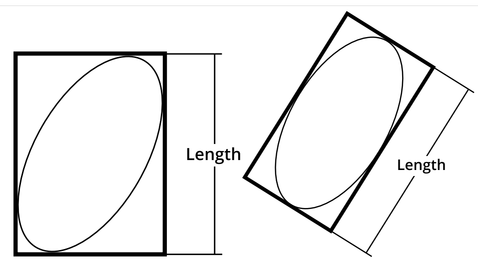
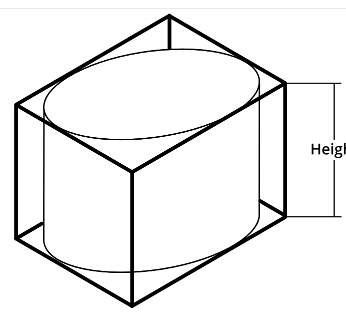


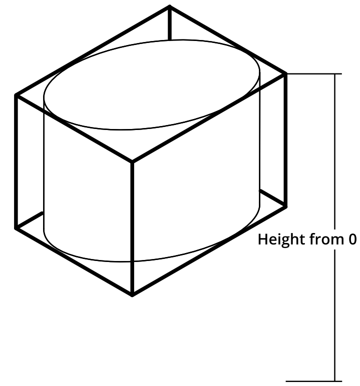
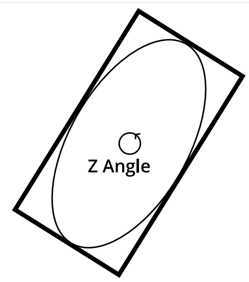
Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Determines the X position of the center of the bounding box that contains the part. The value returned is relative to the *part*.	
Y Determines the Y position of the center of the bounding box that contains the part. The value returned is relative to the *part*.	

Measurement	Illustration
Z Determines the Z position of the center of the bounding box that contains the part. The value returned is relative to the *part*.	
Width Determines the width of the bounding box that contains the part. When the **Rotation** setting is disabled, the bounding box is the smallest rectangle whose sides are parallel to the X and Y axes. Width is on the X axis. When **Rotation** is enabled, the width is the smaller side dimension.	
Length Determines the length of the bounding box that contains the part. When the **Rotation** setting is disabled, the bounding box is the smallest rectangle whose sides are parallel to the X and Y axes. Length is on the Y axis. When **Rotation** is enabled, the length is the longer side dimension.	
Height Determines the height of the bounding box that contains the part.	

Measurement	Illustration
Height from 0 Determines the distance from the top of the bounding box to the Z origin (Z = 0).	
Z Angle Determines the rotation around the Z axis and the angle of the longer side of the bounding box relative to the X axis. If **Rotation** is not enabled, the measurement returns 90.000 degrees. In order to use this measurement for angle anchoring, you must enable **Rotation**; for more information on anchoring, see *Measurement Anchoring* on page 264.	

Features

Type	Description
Center Point	The center point of the bounding box.
Box Axis Line	The axis of the bounding box.
Corner {n}	Point geometric features representing the corners of the box. Only displayed when **Show Corners** is checked.

For more information on geometric features, see *Geometric Features* on page 262.

Surface Circular Edge

The Circular Edge tool fits a circle to a circular edge in the scan data, using either height map or intensity data. The edge can be the outer edge of a disc-like feature or the inner edge of a hole. The tool can optionally work with partial data, as little as 1/4 of a circle, letting it work with rounded corners.

The tool lets you measure the position and radius of the circular feature and determine its roundness error. The feature is expected to be relatively round and not, for example, ovoid.

The tool also returns minimum and maximum error points to either side of the circle and gives the error distance measurement of minimum/ maximum points to the circle.

The tool uses one of four standard methods to calculate roundness. The choice of method affects the other measurements.

- Least Square Circle (LSC)
- Minimum Zone Circle (MZC)
- Maximum Inscribed Circle (MIC)
- Minimum Circumscribed Circle (MCC)

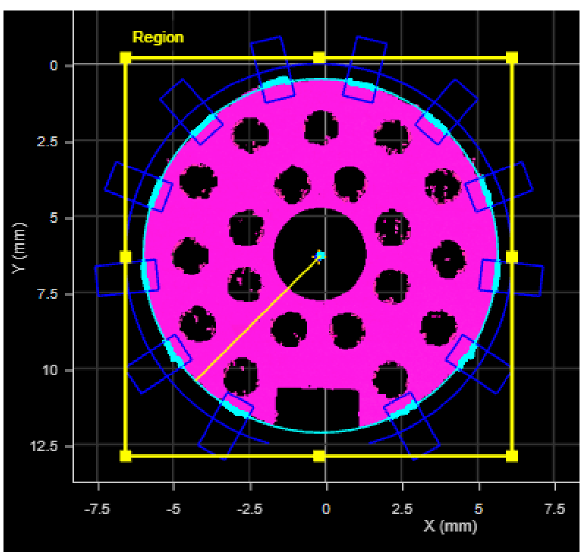
The tool can also generate circle and center point geometric features that Feature tools can take as input for measurement. For more information on Feature tools, see *Feature Measurement* on page 692.

Some of the tool parameters are hidden unless **Show Advanced Parameters** is checked.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Calipers, Extracted Paths, and Edge Points

To fit a circle to the scan data, the Surface Circular Edge tool starts by overlaying evenly spaced calipers along a circular path constrained by the region of interest.



Rectangular calipers (dark blue) placed along circular path (dark blue), constrained by the region

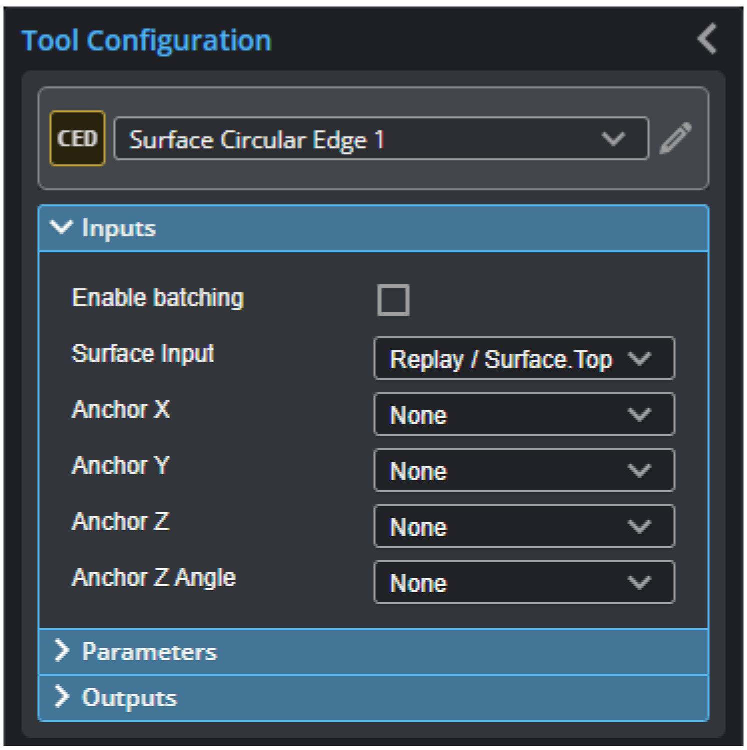
The circular path can optionally be partial, and starts at a defined orientation around the Z axis. The circular path can be as short as 1/4 of a circle, letting it work with rounded corners. Calipers extend vertically to fill the entire region of interest.

Internally, the tool extracts profiles from the data within each caliper, running from the end of the caliper closest to the center of the tool's region of interest to the end farthest from the center. The tool then searches for steps in each profile that meet the criteria set by the tool's settings, such as minimum height, direction (whether it is rising or falling), and so on.

The tool places an edge point on each selected step. The tool then uses the edge points in all the calipers to fit a circle. The various characteristics of the fitted circle are then returned as measurements.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

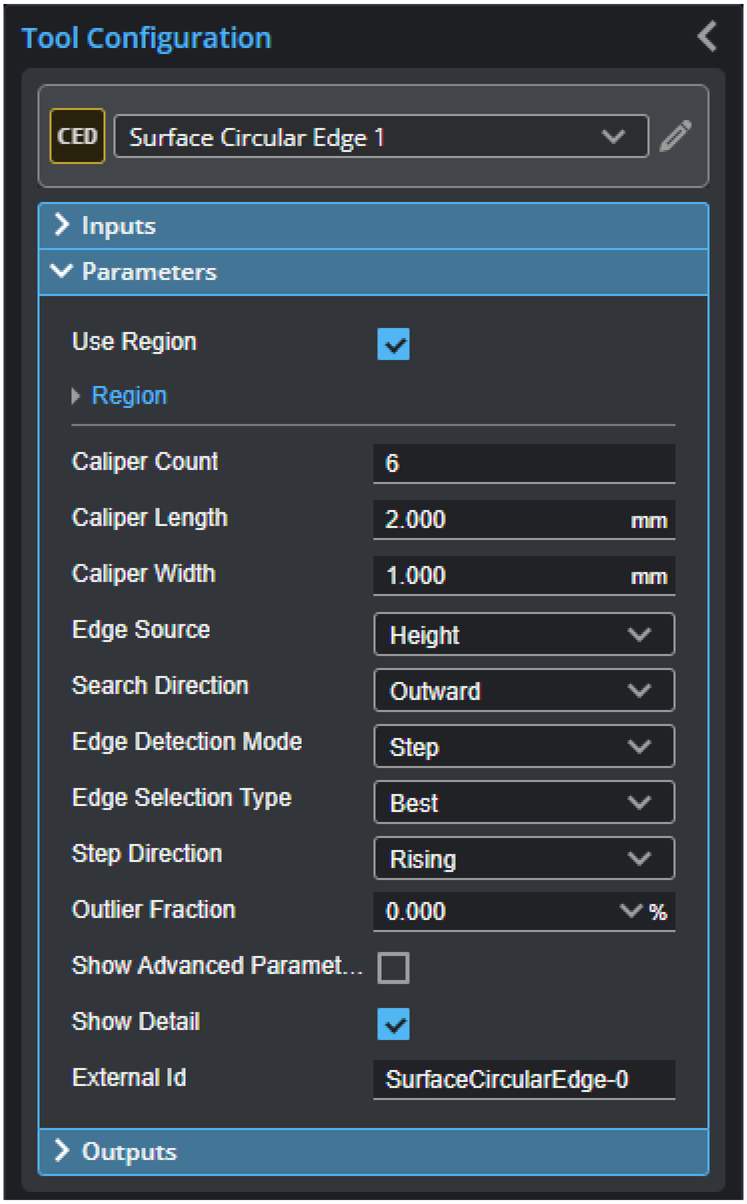
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor.

Name	Description
	For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.
Reference Plane	Uses the feature output of a Surface Plane tool as a reference plane. Useful to correct the scan data if the target is slightly tilted. Only displayed when **Show Advanced Parameters** is enabled.

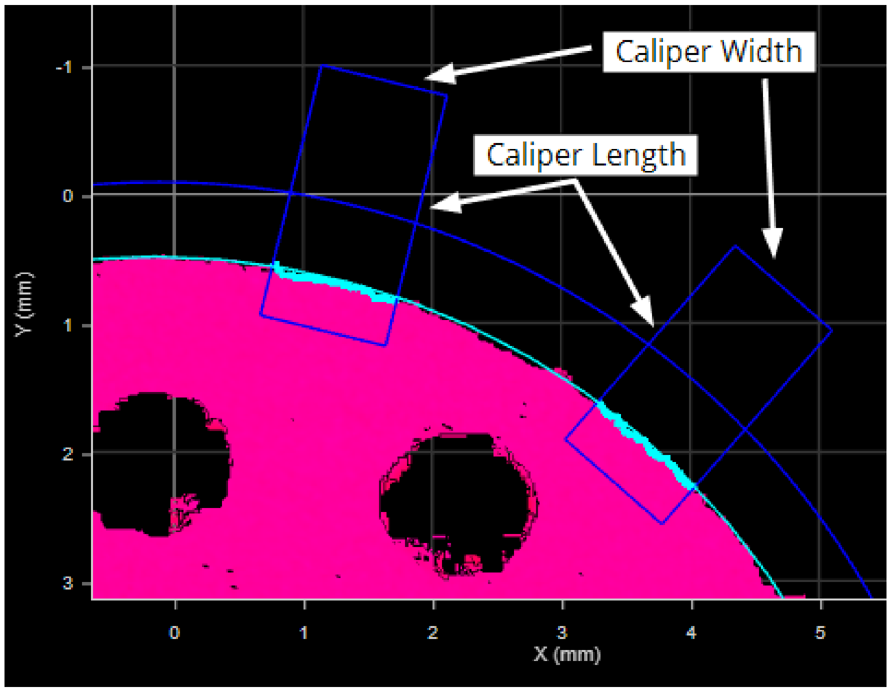
Parameters

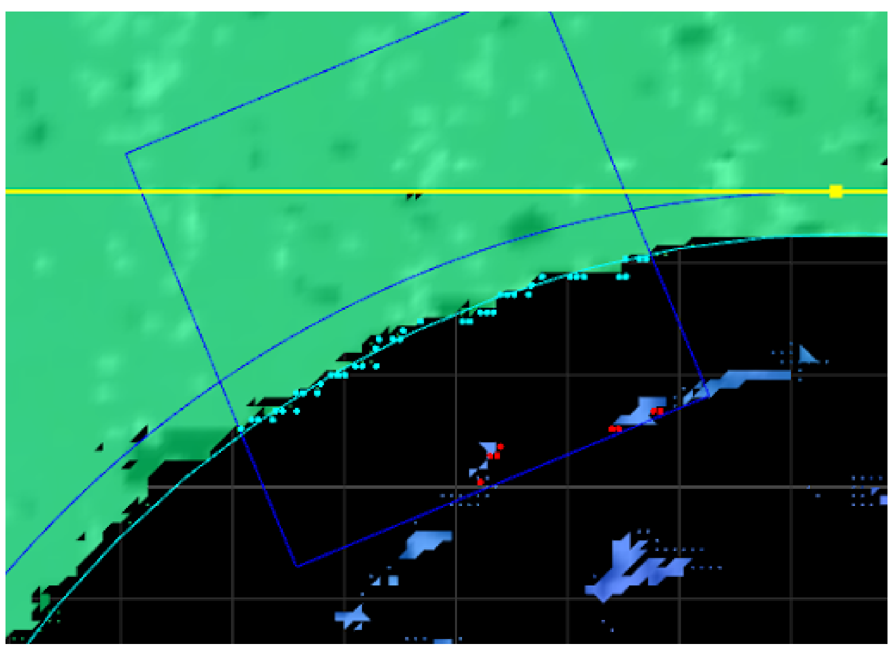
You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Caliper Count	The number of calipers the tool places along the circular path. Using a higher number of calipers increases the amount of data available to the tool, but also increases the amount of time the tool takes to run. Choose a balance between the runtime of the tool and the number of calipers needed to get enough edge points to properly fit the circle to the scan data.

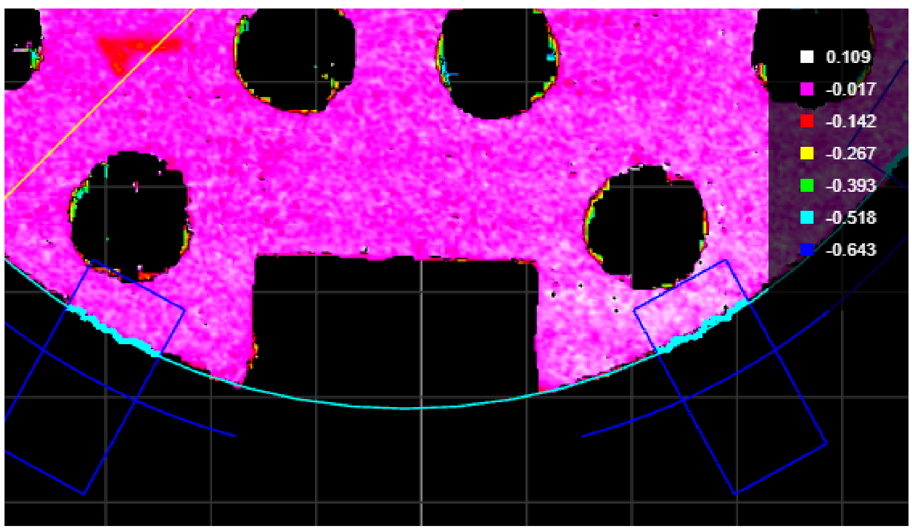
Parameter	Description
Caliper Length Caliper Width	**Caliper Length** is the length of the calipers (extending perpendicular to a tangent on the circular caliper path, centered on the path). The length of the calipers determines the length of the extracted profiles the tool examines for steps. Longer calipers increase the amount of data the tool must analyze and therefore the time the tool takes to run; longer calipers can also include unwanted steps when the tool searches for the edge. **Caliper Width** is the width of the calipers (extending parallel to a tangent on the circular path). A wider caliper increases the time the tool takes to run. It does however increase the number of edge points, which may help the tool fit the circle. 
Edge Source	Specifies the type of data the tool uses. Either Height or Intensity. Use intensity data as the edge source when contrast differences on a flat area of a target, which would not be detected using height map data, are distinct, letting the tool use the detected edge to fit the circle.
Search Direction	Specifies the search direction along the calipers. Either Inward (toward the center of the region of interest) or Outward.
Edge Detection Mode	One of the following: Step or Corner. **Step**: Searches for steps on each path profile. **Corner**: Searches for slopes on each path profile. When you choose this mode, several of the tool's parameters are hidden.

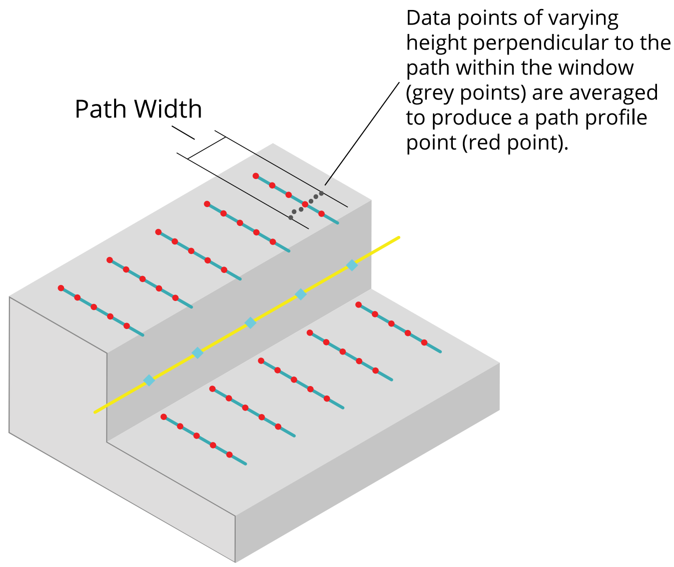
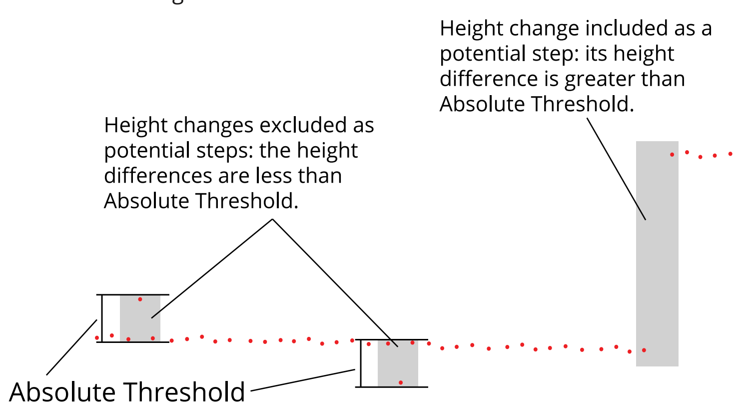
Parameter	Description
Edge Selection Type	Determines which step the tool uses on each of the profiles internally extracted from the calipers when there are multiple steps. An edge point is placed on each chosen step, and is used to fit the circle. Steps must pass the criteria of the tool's settings, such as threshold and outlier exclusion. **Best**: Selects the greatest step in the search direction on each profile. **First**: Selects the first step in the search direction on each profile. **Last**: Selects the last step in the search direction on each profile.
Corner Type	Determines which corner in the search direction on each profile the tool selects. One of the following: **Best**: Selects the greatest corner. **First**: Selects the first corner. **Last**: Selects the last corner. **Top**: Selects the top corner. **Bottom**: Selects the bottom corner.
Step Direction	Determines whether the expected step in the data rises or falls, or moves from valid to null or null to valid. Note that this setting depends on the **Edge Search Direction** setting for its interpretation of what "rises" and "falls." One of the following: **Rising & Falling**: Searches for edge points on rising or falling edges. **Rising**: Searches for edge points only on rising edges. **Falling**: Searches for edge points only on falling edges.
Outlier Fraction	The percentage of outlier points to exclude. Setting this to a small value can help the tool fit the circle better to the edge. 

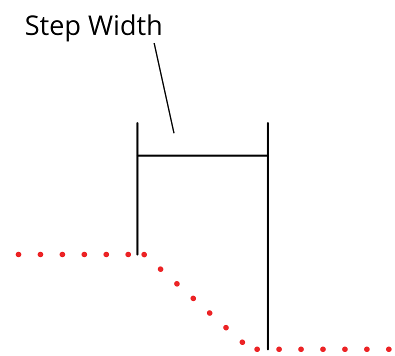
Parameter	Description
Show Advanced Parameters	When enabled, displays advanced settings. Note that most of these settings are applied *even when they are hidden*. For information on these settings, see *Advanced Parameters* below.
Show Detail	When disabled, hides the calipers and caliper path, as well as the edge points.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

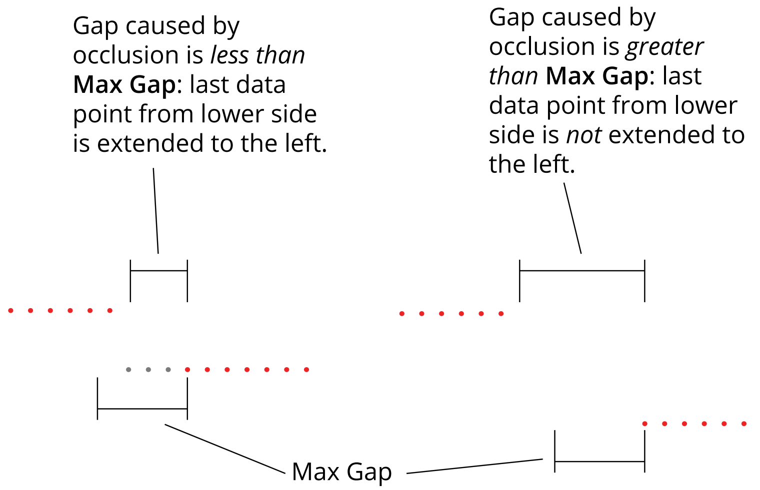
The following parameters are hidden when **Show Advanced Parameters** is unchecked. All advanced parameters, *except***Reference Plane**, are applied when they are hidden. Mask regions are not rendered, even though they are applied.

Advanced Parameters

Parameter	Description
Angle Start Angle Span	These settings work together to let you set a partial path and exclude part of the data. In the following close-up image of a circular feature, the dark blue path starts to the right of the notch, continues counter-clockwise around the circular feature, and ends to the left of it.  **Angle Start** is the starting angle, around the Z axis on the XY plane, for the circular path along which calipers are placed. Setting this to 0 aligns the start angle with the positive direction of the X axis. **Angle Span** is the length of the circular path along which calipers are placed.
Path Spacing	Sets the spacing between paths in the calipers used to extract the profiles that determine the edge. A higher number of paths results in a higher number of edge points, which makes the fitting of the edge line more accurate. However, a higher number of edge points results in a greater tool execution time. When **Path Spacing** is set to 0, the resolution of the scan data is used as the basis for spacing.

Parameter	Description
Path Width	The size of the windows perpendicular to the path used to calculate an average for each data point on a path profile. Useful to average out noise along the path caused by reflections, and so on.  If **Path Width** is set to 0, no averaging is performed (only the data point under the path is used). For averaging along the path, use **Step Smoothing** (see below).
Absolute Threshold	When **Use Intensity** is disabled, the setting specifies the minimum *height* difference between points on a path profile for that step to be considered for an edge point. The setting can be used to exclude smaller steps on a part that should not be considered for an edge, or to exclude height differences caused by noise. When used in conjunction with **Relative Threshold**, **Absolute Threshold** is typically set to a small value, greater than the general surface roughness.  When **Use Intensity** is enabled, the setting specifies the minimum difference in intensity. (**Acquire Intensity** must enabled in the **Scan Mode** panel.)
Use Relative Threshold	When this option is enabled, the **Relative Threshold** field is displayed.

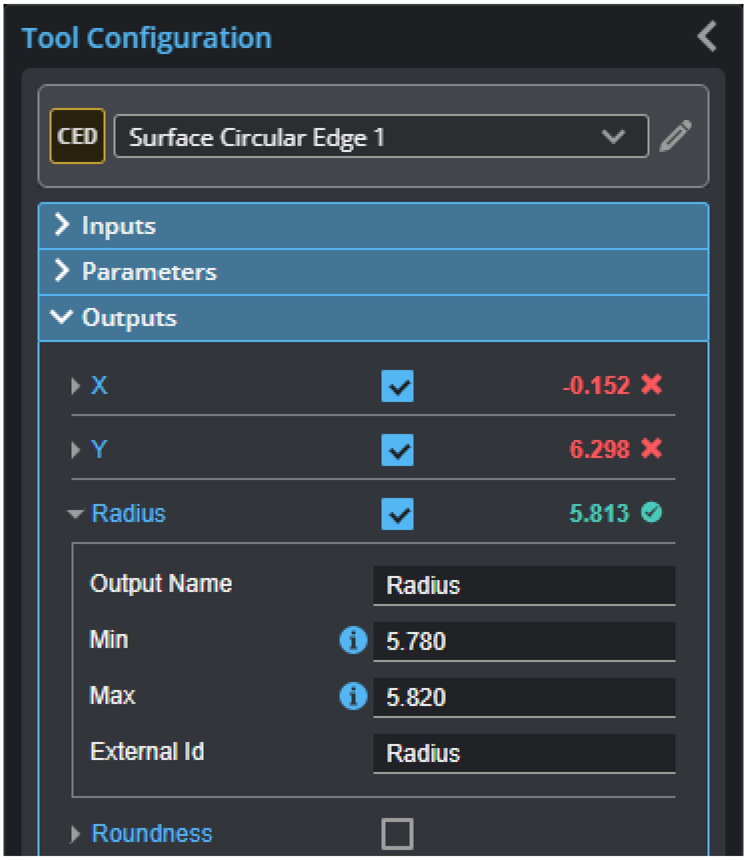
Parameter	Description
Relative Threshold	The value for the relative threshold. The tool calculates a relative threshold by scaling the greatest height or intensity difference found on the path profiles by the percentage in **Relative Threshold**. This lets you configure the tool without knowing the actual step height in advance, and is useful for edges with varying step height. For a height or intensity difference to be considered a valid step, both **Absolute Threshold** and **Relative Threshold** must pass.
Step Smoothing	The size of the windows along the path used to calculate an average for each data point on a path profile. The setting is useful for averaging out noise. If **Step Smoothing** is set to 0, no averaging is performed (only the data point under the path is used). For averaging perpendicular to the path, use **Path Width** (see above).
Step Width	The distance, along a path profile, separating the points used to find steps on a path profile.  The setting is useful when you must detect a slope as an edge, rather than a sharply defined edge: setting **Step Width** to a value greater than the width of the edge ensures that the tool measures the height difference between the flat regions on either side of the edge. As a result, the height of the step is accurately measured, and the edge is correctly located. Setting **Step Width** wider than necessary can reduce the precision of edge location.

Parameter	Description
Max Gap	Fills in regions of missing data caused by an occlusion near the desired edge. Use this setting when continuity on the target is expected. When **Max Gap** is set to a non-zero value, the tool holds and extends the last data point on the low side next to an edge across a gap of null points, up to the distance specified in **Max Gap**. 
Include Null Edges	Indicates whether null points (points where no height or intensity value is available, due to dropouts or regions outside of the measurement range) are filled with the value in **Null Fill Value** as a general "background level." If **Use Intensity** (see above) is enabled, the intensity value in **Intensity Null Fill Value** is also used. A typical example is a discrete part produced by the part detection of an object sitting on a flat background (for more information on part detection, see *Profile Part Detection* on page 373). The background is not visible in the part, so the tool assumes that any null region are at the background level. To find edges along a region of null points, you must use either this option and an appropriate value in **Null Fill Value** (and **Intensity Null Fill Value** if **Use Intensity** is enabled) or **Max Gap**. Otherwise, only edges within areas of contiguous data will be detected.
Null Fill Value	The height value (in mm) used to replace null points not filled by **Max Gap** when **Include Null Edges** is enabled.
Intensity Null Fill Value	The intensity value (0-255) used to replace null points when **Include Null Edges** and **Use Intensity** are enabled.

Parameter	Description
Mask Regions	Lets you enable up to five regions that you can use to mask data you want the tool to ignore. You can resize and reposition the mask regions using the mouse in the data viewer, or by configuring values manually in the **Mask Region** sections the tool displays in the tool settings for each region. You can only set the rotation of the mask regions manually by modifying the region's **Z angle** parameter. By default, when you add multiple mask regions, they are initially placed in the same position, one on top of the other.
Reference Plane	Uses the output of a Surface Plane tool as a reference plane. Useful to correct the scan data if the target is slightly tilted. When **Show Advanced Parameters** is unchecked and **Reference Plane** is set to a plane, the plane is *ignored*.
Fit Type	The method the tool uses to calculate the roundness of the feature. One of the following: **Least Square Circle (LSC)** **Minimum Zone Circle (MZC)**: If you choose this method, set the circle the tool uses with the **Which Circle** parameter. **Maximum Inscribed Circle (MIC)**: Typically used to measure the inner edge of a circular feature, such as a hole. **Minimum Circumscribed Circle (MCC)**: Typically used to measure the outer edge of a circular feature.
Which Circle	Tells the tool which circle (Inner or Outer) to use when Minimum Zone Circle is the fit method in **Fit Type**.

Outputs

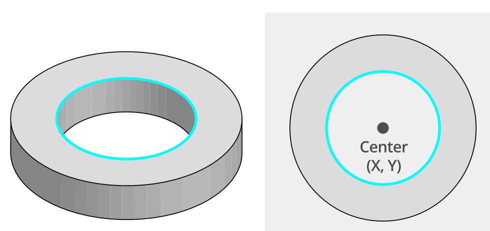
Most tools provide measurements, geometric features, or data as outputs.

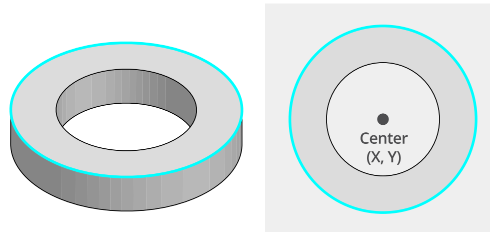
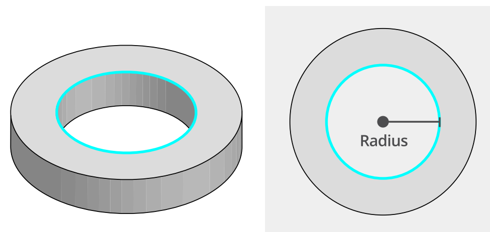
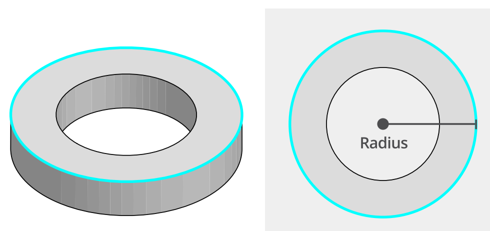
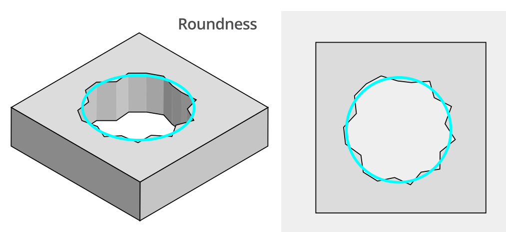
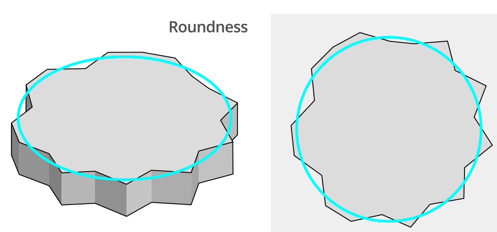


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X **Y** Returns the X and Y position of the center of the fitted circle, respectively.	

Measurement	Illustration
	
Radius Returns the radius of the fitted circle.	 
Roundness Returns the roundness or circularity of the edge points with respect to the reference circle of the selected roundness error method set in **Fit Type**.	 
Min Error **Max Error** These measurements return information on the points	

Measurement	Illustration
furthest inside and outside the fitted circle, respectively.	

Features

Type	Description
Center Point	The center of the fitted circle.
Circle	The fitted circle.
Min Error Point	The point of minimum error which is furthest inside of the circle.
Max Error Point	The point of maximum error which is furthest outside of the circle.

For more information on geometric features, see *Geometric Features* on page 262.

Surface Countersunk Hole

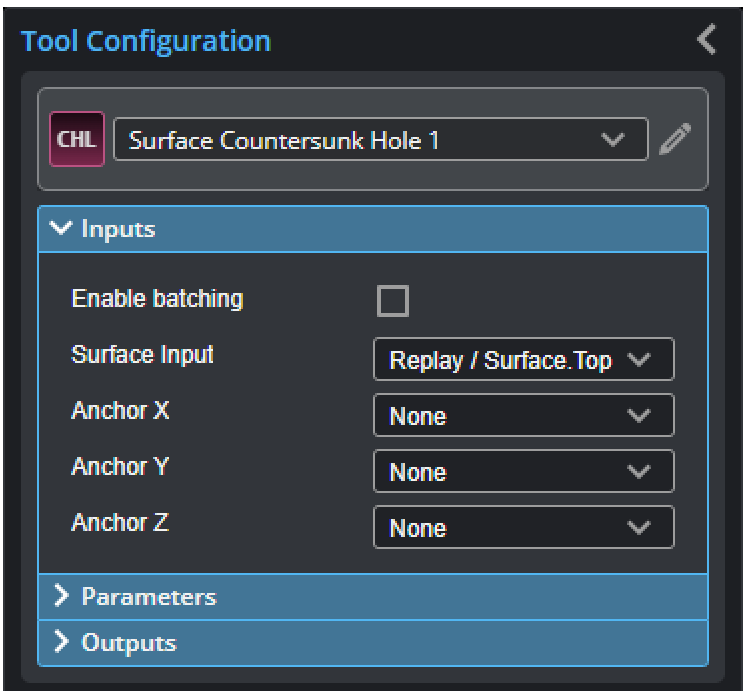
The Countersunk Hole tool locates a countersunk circular opening within a region of interest on the surface and provides measurements to evaluate characteristics of countersunk holes, including the position (X, Y, and Z) of the center of the hole, outside radius of the hole, hole bevel angle, and the depth of the hole. The countersunk hole can be on a surface at an angle to the sensor. The tool also supports measuring holes drilled at an angle relative to the surrounding surface.

> The tool does not search for or detect the feature. The tool expects that the feature, conforming reasonably well to the defined parameters, is present and that it is on a sufficiently uniform background.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

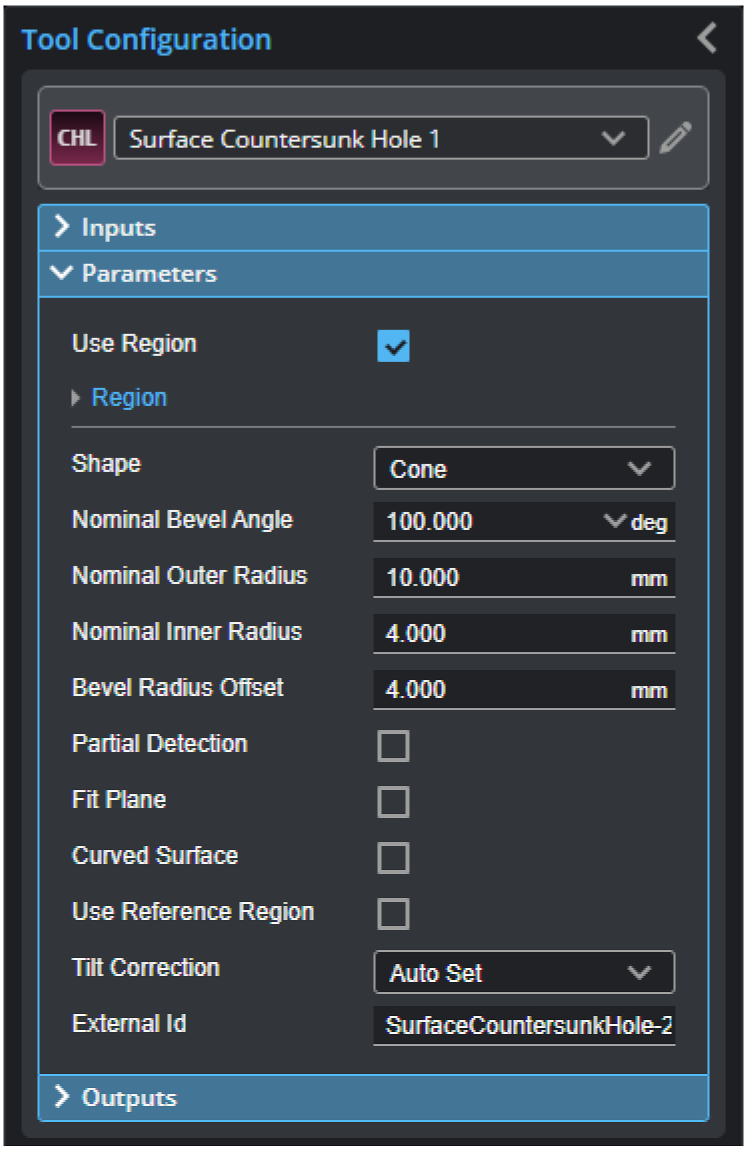
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor.

Name	Description
	For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

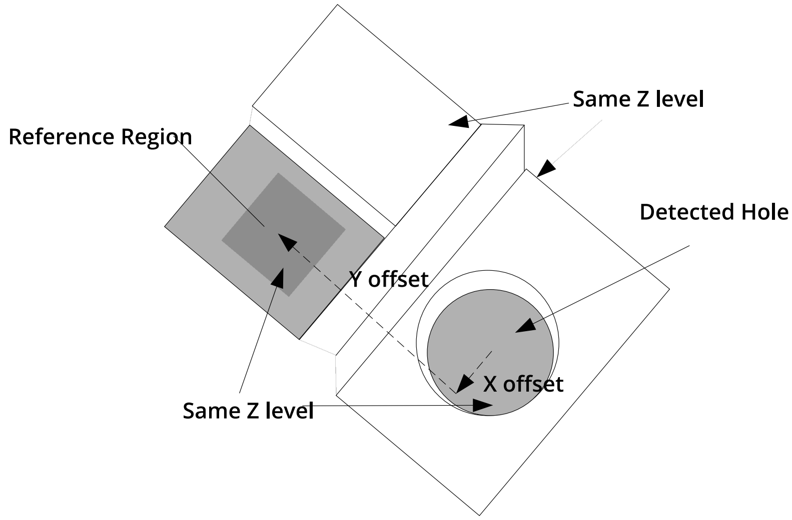
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



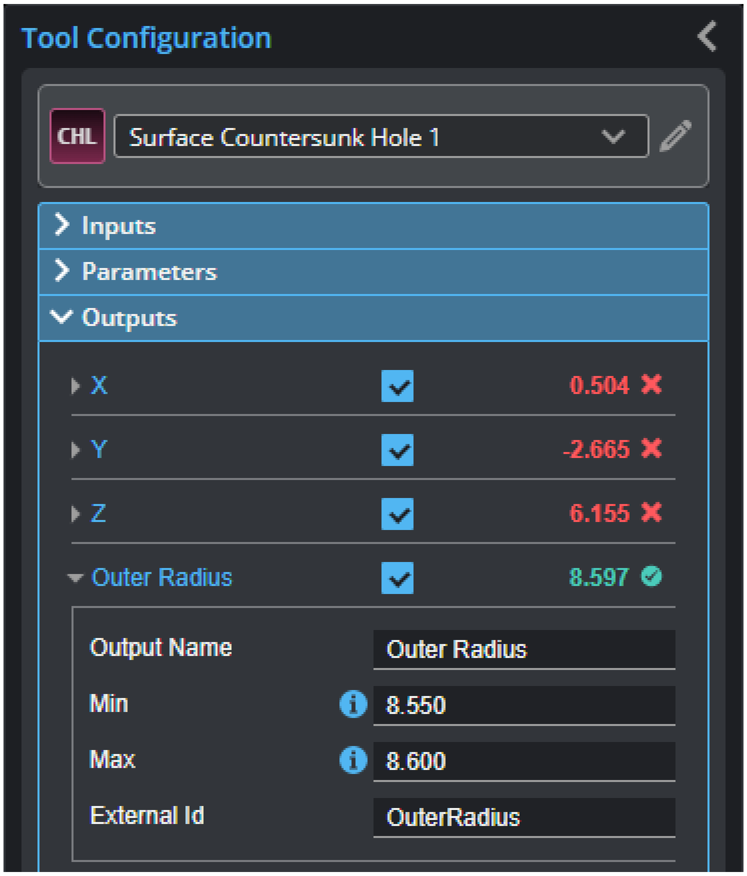
Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Shape	The shape of the countersunk hole. (See illustrations below.) 0 – Cone 1 – Counterbore
Nominal Bevel Angle	The expected bevel angle of the countersunk hole.
Nominal Outer Radius	The expected outer radius of the countersunk hole.
Nominal Inner Radius	The expected inner radius of the countersunk hole.
Bevel Radius Offset	The offset, relative to the surface that the countersunk hole is on, at which the bevel radius will be measured.
Partial Detection	Enable if only part of the hole is within the measurement region. If disabled, the hole must be completely in the region of interest for results to be valid.
Fit Plane	When enabled, excludes data beyond the distance from the plane surrounding the hole, using the value you provide in the **Plane Fit Range** parameter. You can use this setting to exclude surfaces close to the countersunk hole that step down from the plane surrounding the hole that could make measurement of the hole less reliable.
Plane Fit Range	The distance from the plane surrounding the hole beyond which data is excluded.
Curved Surface	Whether the surface that the countersunk hole is on is curved.
Curve Orientation	The orientation of the curvature in degrees. Only visible when **Curved Surface** is enabled.
Use Reference Region	When enabled, displays settings you use to configure reference regions.
Reference Type	The tool uses the reference regions to calculate the Z position of the hole. It is typically used in cases where the surface around the hole is not flat.

Parameter	Description
	 When this option is set to **Autoset**, the algorithm automatically determines the reference region. You can also set this option to one or two regions, which you manually set. Note that the location of manual reference regions is relative to the detected center of the hole and positioned on the nominal surface plane. When **Reference Region** is disabled, the tool measures the hole's Z position using all the data in the measurement region, except for a bounding rectangular region around the hole.
Tilt Correction	Tilt of the target with respect to the alignment plane. **Autoset**: The tool automatically detects the tilt. The measurement region to cover more areas on the surface plane than other planes. **Custom**: You must enter the X and Y angles manually in the X Angle and Y Angle parameters (see below).
X Angle Y Angle	The X and Y angles you must specify when **Tilt Correction** is set to **Custom**. You can use the Surface Plane tool's X Angle and Y Angle measurements to get the angle of the surrounding surface, and then copy those measurement's values to the **X Angle** and **Y Angle** parameters of this tool.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

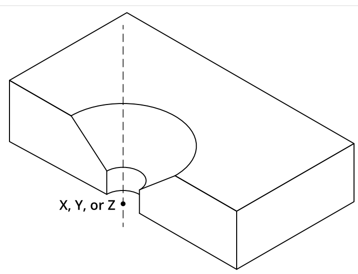
Most tools provide measurements, geometric features, or data as outputs.

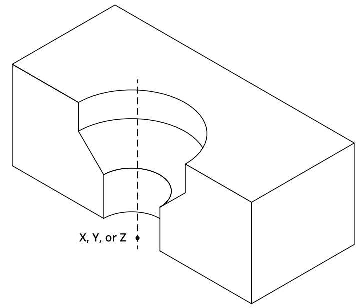
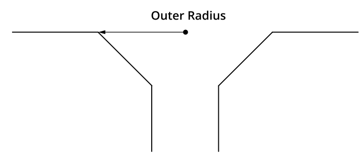
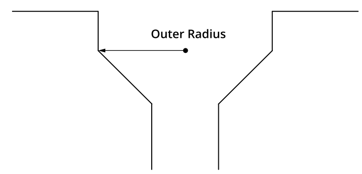
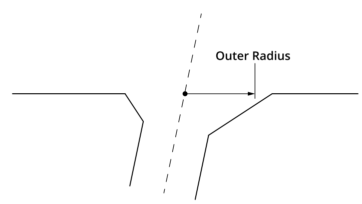
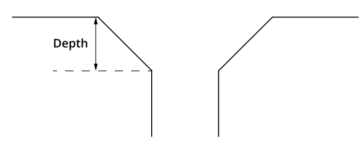
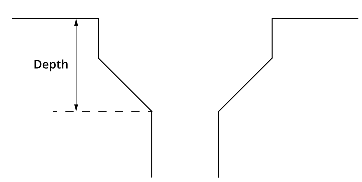


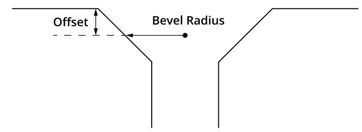
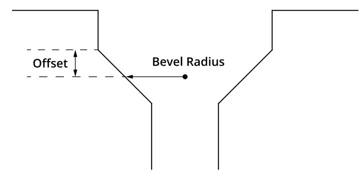
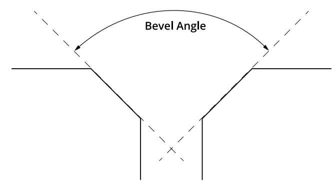
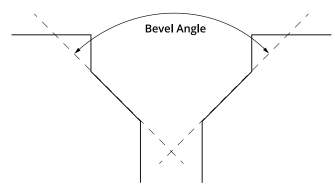
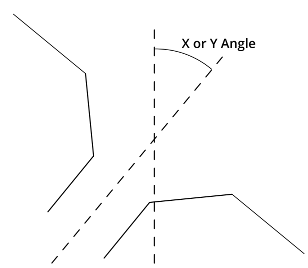
Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

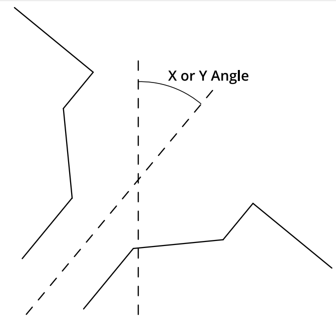
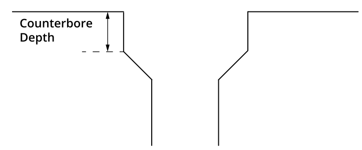
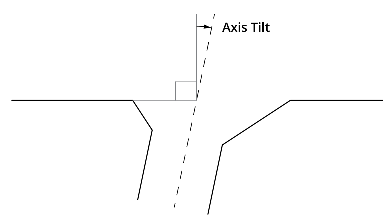
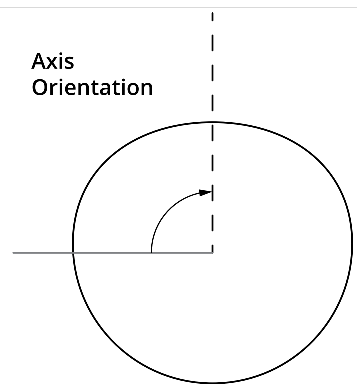
You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Determines the X position of the center of the countersunk hole.	

Measurement	Illustration
	
Y Determines the Y position of the center of the countersunk hole.	
Z Determines the Z position of the center of the countersunk hole.	
Outer Radius Determines the outer radius of the countersunk hole. When a hole is cut at an angle relative to the surrounding surface, the outer radius is calculated as if the hole were not cut at an angle. To convert the radius to a diameter, use an instance of the Measurement Formula too with the Formula field set to M1*2, for example. For more information, see *Measurement Formula Tool* on page 767.	  
Depth Determines the depth of the countersunk hole relative to the surface that the countersunk hole is on.	 

Measurement	Illustration
Bevel Radius Determines the radius at a user-defined offset (**Bevel Radius Offset** setting) relative to the surface that the countersunk hole is on. To convert the radius to a diameter, use an instance of the Measurement Formula too with the Formula field set to M1*2, for example. For more information, see *Measurement Formula Tool* on page 767.	 
Bevel Angle Determines the angle of the hole's bevel.	 *Cone*  *Counterbore*
X Angle Determines the angle the hole relative to the X axis.	 *Cone*

Measurement	Illustration
Y Angle Determines the angle of the hole relative to the Y axis.	 *Counterbore*
Counterbore Depth Determines the depth of a counterbore.	
Axis Tilt Measures the tilt of the axis of the hole relative to the surface surrounding the hole. This measurement is not supported when **Shape** is set to **Counterbore**.	
Axis Orientation Measures the angle of the axis of the hole around the normal of the surface surrounding the hole, relative to the X axis. This measurement is not supported when **Shape** is set to **Counterbore**.	

Features

Type	Description
Center Point	The center point of the countersunk hole. The Z position of the center point is at the Z position of the surrounding surface.

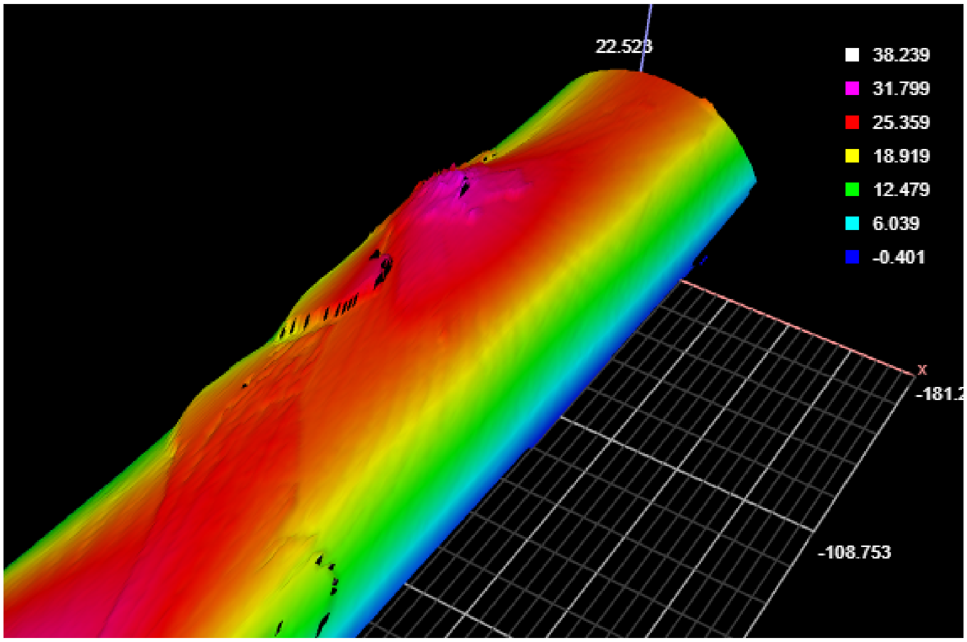
For more information on geometric features, see *Geometric Features* on page 262.

Surface Curvature Correction

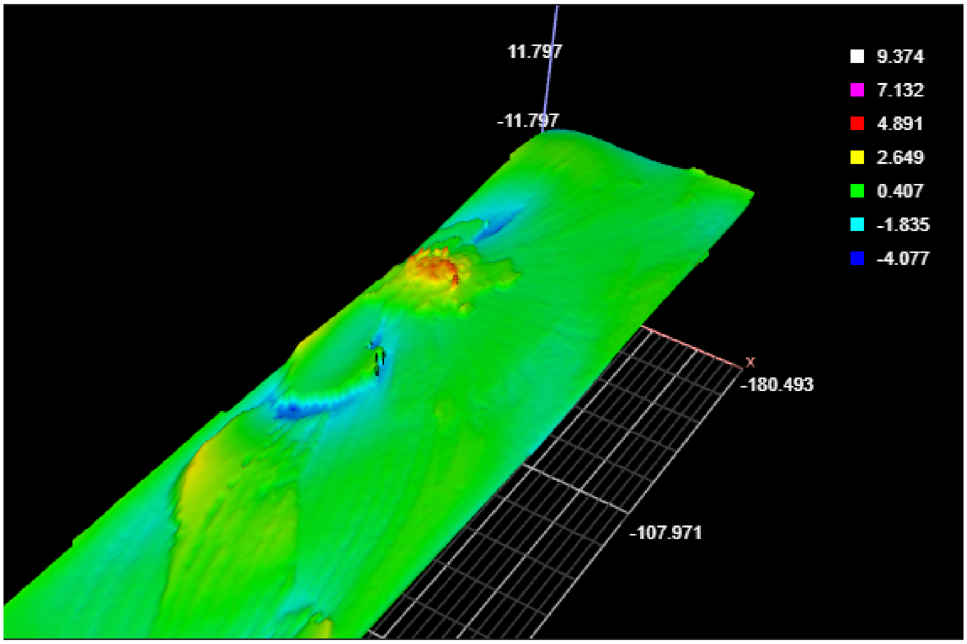
The Surface Curvature Correction tool removes curvature from curved surfaces while preserving surface features or defects, using a configurable polynomial order (the tool performs a 2D polynomial fit on X and Y to process surfaces). You can then use the tool's output as input to other measurement tools to apply measurements to the "flattened" surface.

The tool does not support rotational scans (that is, polar "unwrapping").

In the following images, a curved surface (top) is flattened out (bottom), preserving the surface detail.

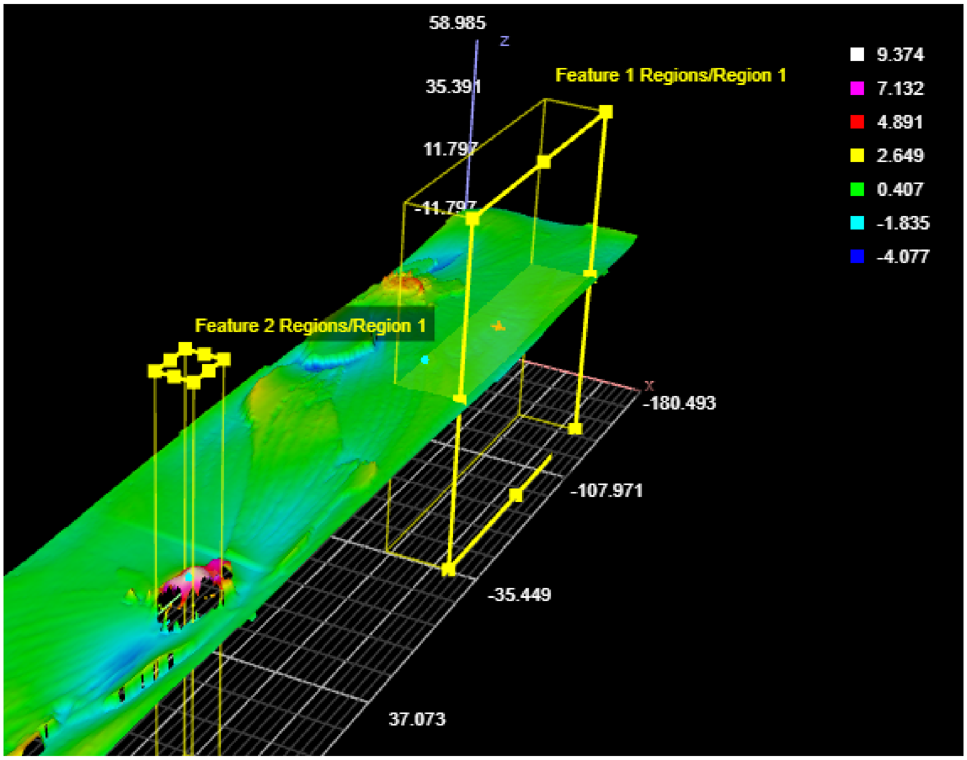


The original, curved scan of a target.



The "flattened" surface data (the tool's Difference Surface data output).

In the following image, a Surface Dimension tool's height measurement runs on the "flattened" output (the Surface Curvature tool's Difference Surface output) to determine the height of one of the raised areas:

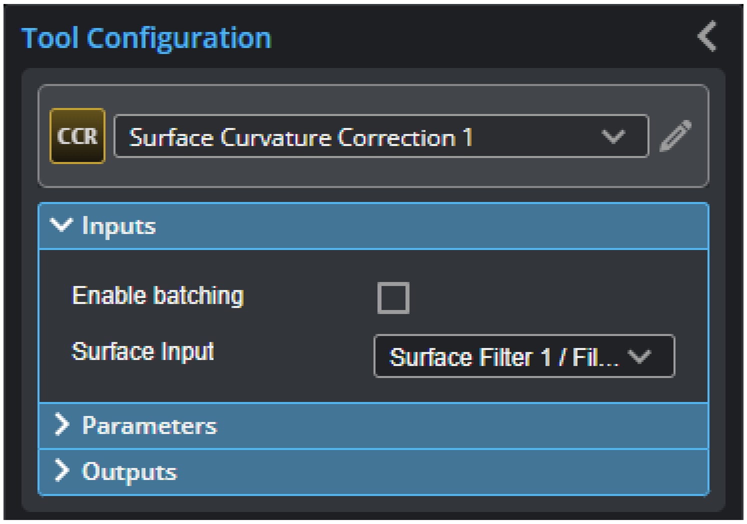


Height of a raised feature relative to the flattened surface.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

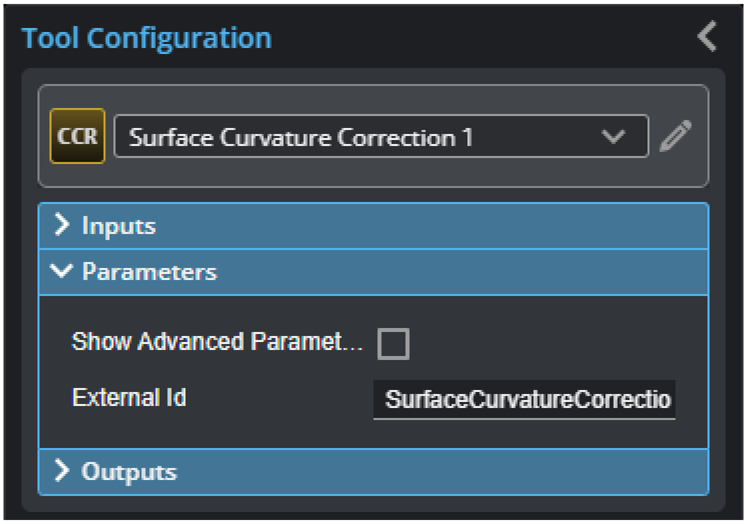


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

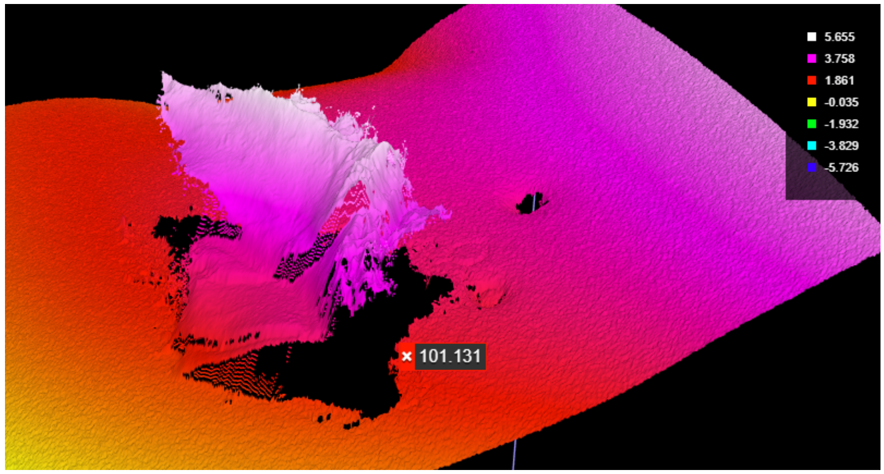
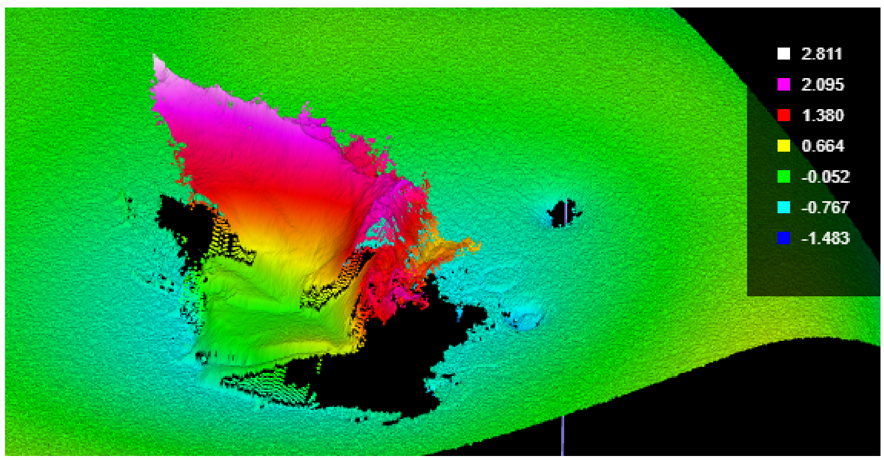


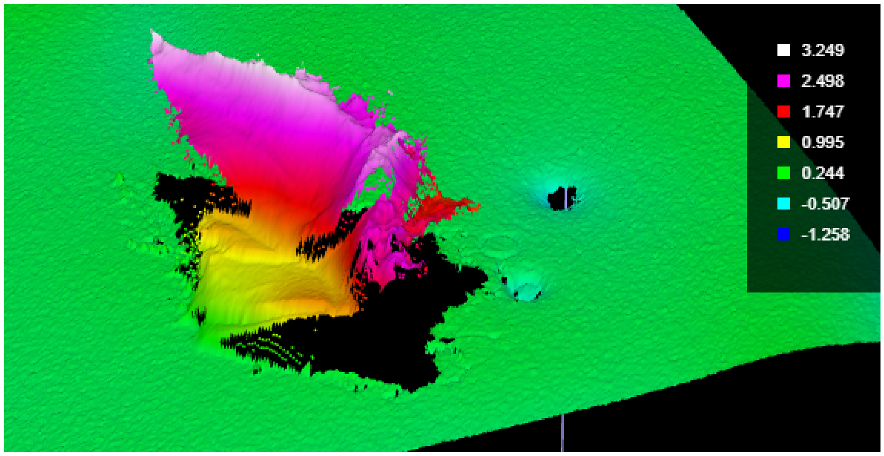
Parameters

Parameter	Description
Show Advanced Parameters	Enables a set of advanced parameters. (See *Advanced Parameters* below.)
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Advanced Parameters

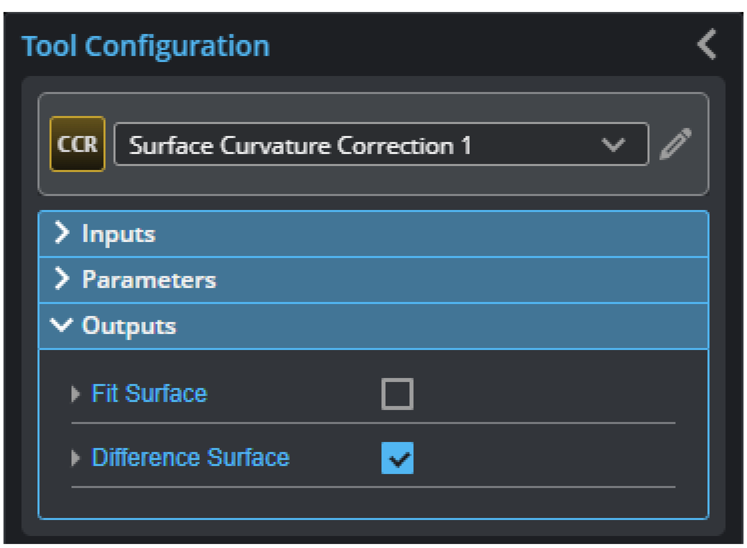
Parameter	Description
Number of Regions	Lets you specify and configure one or more regions the tool will process. Use this parameter to limit the tool to specific areas on the target.
Exclude Features	Lets you exclude features or surface details from the polynomial fit. This can allow you to get a better fit on the surrounding surface. Checking this option enables the **Negative area**, **Positive area**, and **Iterative Steps** parameters. (See below.) For example, in the following scan data, we would like to accurately measure the circular divots and the small hole near the center of the data on the curved surface.

Parameter	Description
	 If the large feature to the left is *not* excluded for the polynomial fit, the measurements on the smaller features will be inaccurate. In the following "flattened" scan data, without excluding the larger feature, the smaller features would be difficult to accurately measure:  When the larger feature is excluded from the polynomial fitting, the surrounding surface and the smaller features are more properly "flattened."

Parameter	Description
	
Polynomial Order	Selects the order (or degree) of the polynomial to be fit to the surface. A higher order results in a better fit but increases processing time.
Sampling Step	The step in data points in both directions with which the surface is sampled. Choosing a higher sampling step reduces the processing time the tool requires, but reduces fit accuracy. Useful if the surface being processed has a large number of data points.
Iterative Steps	The number of times the tool repeats the feature exclusion calculation (see **Exclude Features**, above).
Negative area Positive area	These settings exclude the specified percentage of a histogram of the height values of the scan data from the bottom up (**Negative area**) and from the top down (**Positive area**), respectively.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.

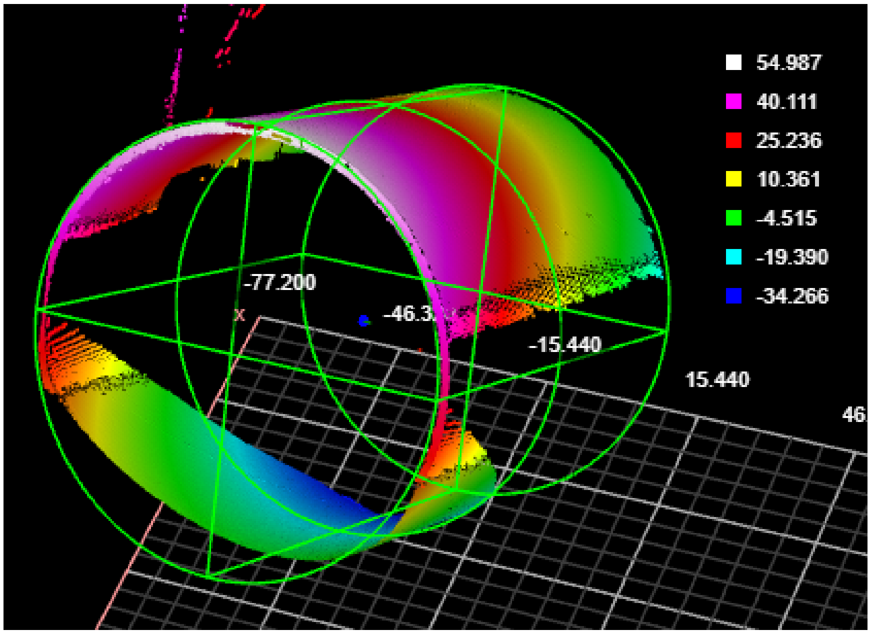


Data

Type	Description
Fit Surface	The fitted polynomial the tool uses to flatten the original surface. Use this as a diagnostic to better fit the polynomial.
Difference Surface	The “flattened” surface: this is the original surface with the fitted polynomial removed.

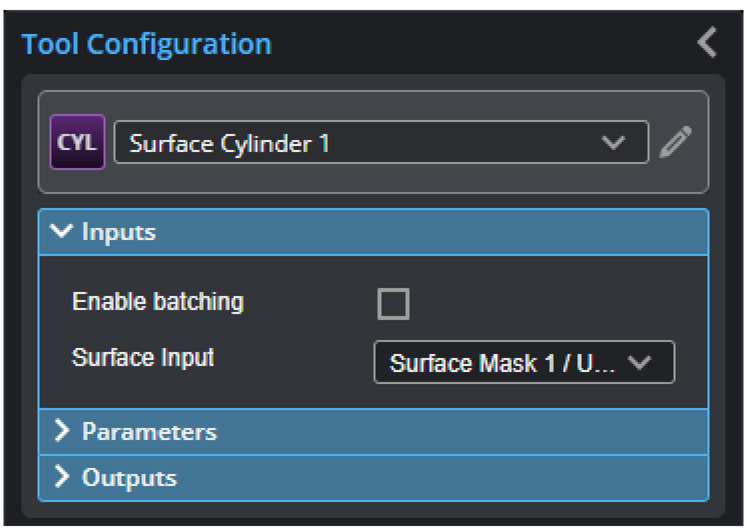
Surface Cylinder

The Surface Cylinder fits a cylinder to scan data and returns measurements and geometric features related to the fitted cylinder. Unlike the Surface Stud tool, the Surface Cylinder tool does not rely on a flat surface perpendicular to the cylindrical object. Examples uses of the tool include fitting to the outside of a cylinder and fitting inside a drilled hole.



Inputs

You configure the tool's inputs in the expandable **Inputs** section.



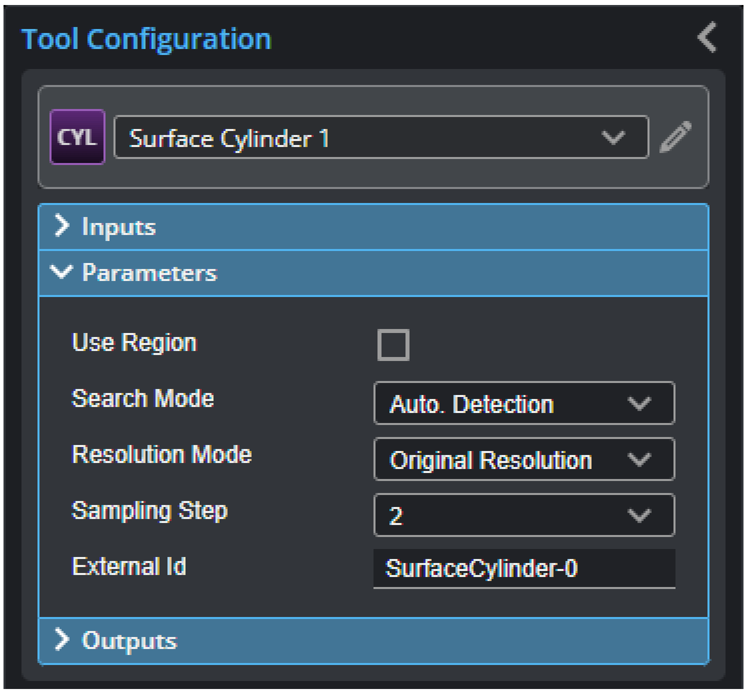
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array,

Name	Description
	other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

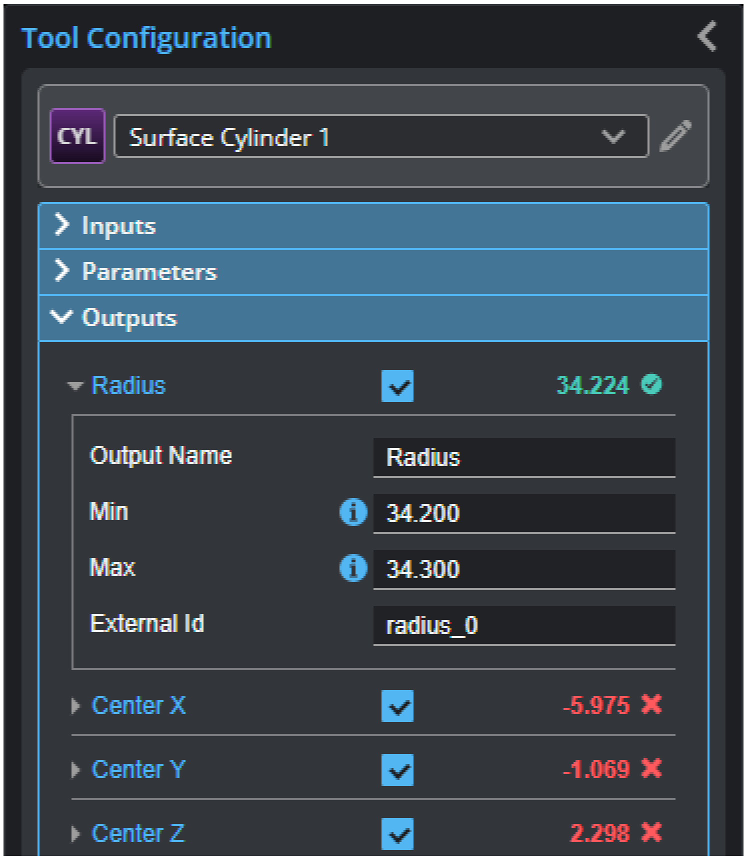


Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Search Mode	Indicates the expected orientation of the cylindrical target's axis around the Z axis. One of the following: **Auto Detection** – The cylindrical target can be in any orientation. **Axis in X Direction** or **Axis in Y Direction** – The cylindrical target's axis is expected to be roughly parallel to the X or the Y axis, respectively. Variation typically must be less than +/- 3 or 4 degrees. **Advanced Detection** – As with Auto. Detection, the cylindrical target can be in any orientation. This mode can handle some complex scenarios, such as when trying to fit a cylinder to the inside of a hole. Processing time is greater with this search

Parameter	Description
	mode.
Resolution Mode	On G3 sensors, leave this set to the default **Original Resolution**. Determines whether the tool scales the X or Y resolution so that they are the same (a 1:1 ratio), or leaves the X and Y resolutions as the original. One of the following. • **Optimal (uniform)** Brings the X/Y resolution ratio to 1:1 while preserving the pixel area. Best for random rotation around Z. Provides a balance between the highest and lowest possible resolutions, requiring an average amount of memory and processing time compared to the **High Oriented (uniform)** or **Low Oriented (uniform)** options. • **High Oriented (uniform)** Interpolates the lower resolution to match the higher resolution (between X and Y) in the input. Choose this option when increased resolution is preferred over speed and low memory usage. (This can result in a very high resolution output, creating a lot of data for subsequent tools to process. This can in turn result in slower processing.) • **Low Oriented (uniform)** Decimates the higher resolution to match the lower resolution (between X and Y) in the input. Choose this option when speed and low memory usage is preferred over resolution. (It can result in significant data quality reduction with large Z rotations if the X and Y resolutions of the input are very different.) • **Original Resolution** Keeps the original X and Y resolution of the scan. Use this option only when you expect little or no Z rotation. Otherwise, with X/Y resolution ratios that are not 1:1, large rotation around Z results in severe data quality reduction.
Sampling Step	The step in data points in both directions with which the surface is sampled. Choosing a higher sampling step reduces the processing time the tool requires, but reduces fit accuracy. Useful if the surface being processed has a large number of data points.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Radius Returns the radius of the fitted cylinder.
Center X **Center Y** **Center Z** The X, Y, and Z position of the center of a circle place in the middle of the fitted cylinder
Tilt Angle The angle of the cylinder relative to the XY plane. A cylinder parallel to the XY plane has an angle of 90 degrees.
Direction Angle The angle of the cylinder's axis around the Z axis. An angle of 0 degrees is parallel to the X axis.
Normal X

Measurement
Normal Y **Normal Z** These measurements return the X, Y, and Z components of the direction vector of the cylindrical target.

Features

Type	Description
Point	A point representing the center of a circle at the midpoint of the fitted cylinder
Line	A line representing the axis of the fitted cylinder.

For more information on geometric features, see *Geometric Features* on page 262.

Data

Type	Description
Fit Surface	The Surface data the tool uses to fit the cylinder (for example, excluding data outside the defined region). Use this output, for example, to confirm that the tool is using the correct part of the target to fit the cylinder.
Difference Surface	Deviations in the Surface from the fitted cylinder. Use this output, for example, to evaluate if blemishes or distortions in the Surface data are preventing the tool from properly fitting the cylinder.

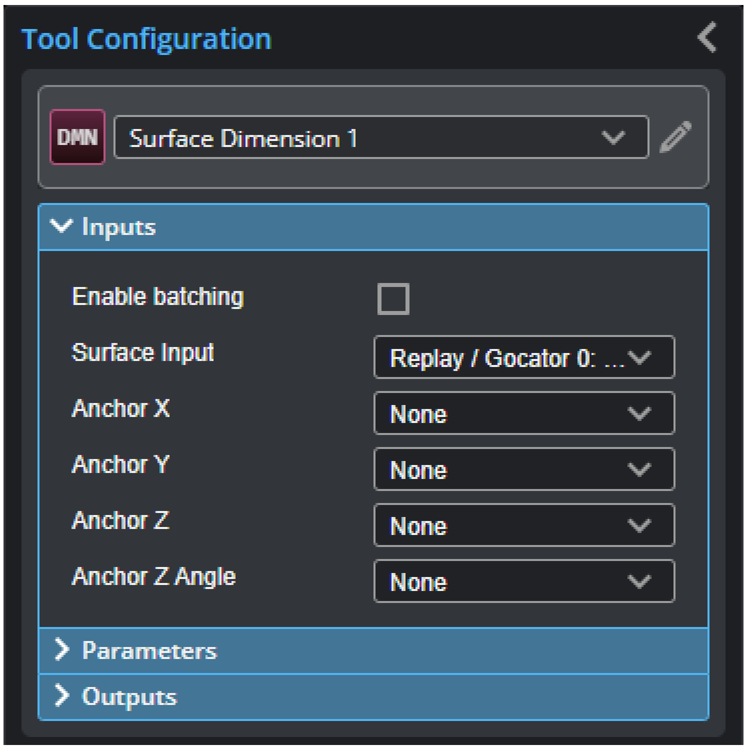
Surface Dimension

The Dimension tool returns various dimensional measurements of a part. You must specify two feature types (see below).

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

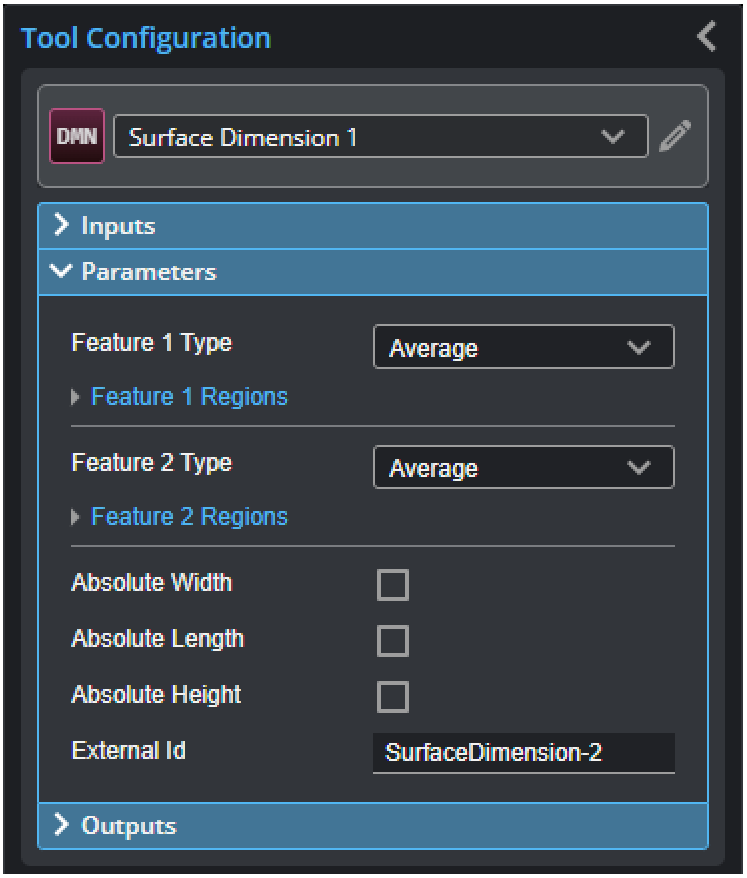
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Name	Description
Anchor Z	
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



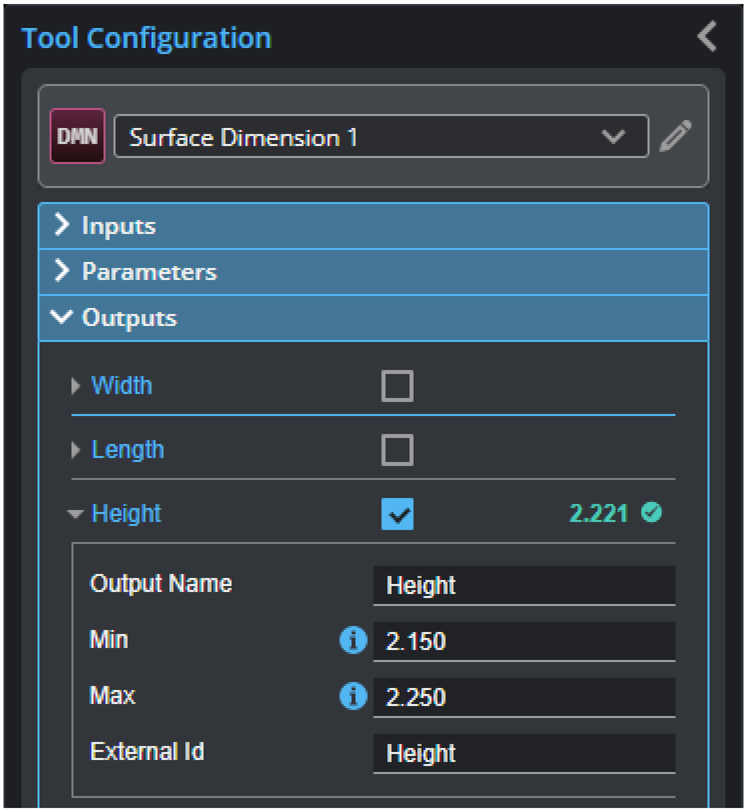
Parameters

Parameter	Description
Feature 1 Type Feature 2 Type	The feature point types the tool uses. For each, one of the following (for more information, see *Feature Points* on page 259): • Average • Median • Centroid • Max X • Min X • Max Y • Min Y • Max Z

Parameter	Description
	• Min Z When more than one point is at minimum Y or maximum Y (and the feature is set to Min Y or Max Y, respectively), the rightmost point (the one at greater positive X) is selected by the tool. When more than one point is at minimum X or maximum X (and the feature is set to Min X or Max X, respectively), the bottom point (the one at greater positive Y) is selected by the tool.
Feature 1 Regions Feature 2 Regions	When these are expanded, they display the region- and mask-related settings associated with the two features the tool uses to perform measurements. The feature parameters associated with these regions are **Feature 1** and **Feature 2**, respectively; for more information, see below.
Enable	When enabled, the region- and mask-related settings are displayed.
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Absolute Width Absolute Length Absolute Height	Enable these parameters to return an absolute value, regardless of the relative positions of the first and second regions of interest. (Dimensions are calculated by subtracting the position of the feature point in region 1 from that of the feature point in region 2.)
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

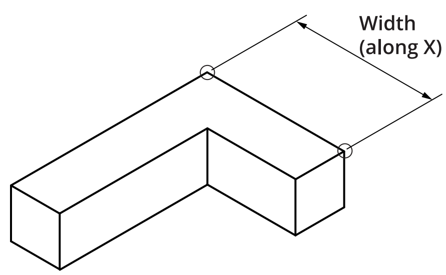
Most tools provide measurements, geometric features, or data as outputs.

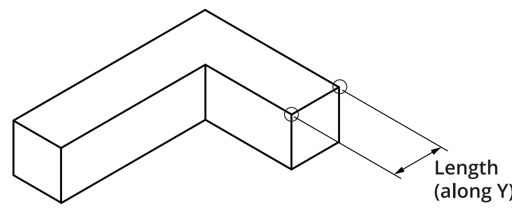
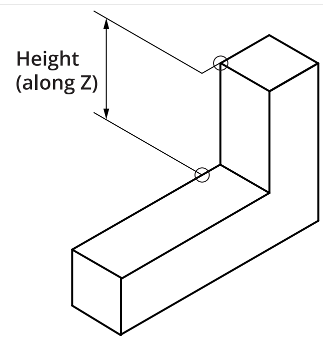
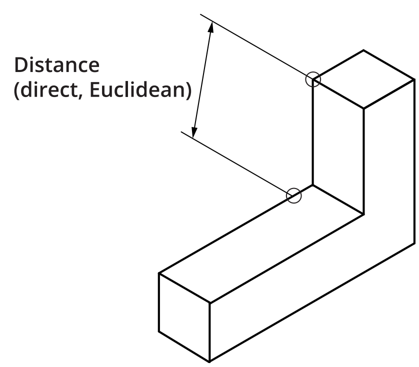


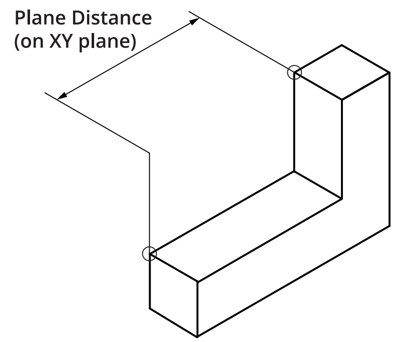
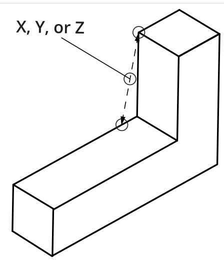
Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Width Determines the distance between the selected features along the X axis.	

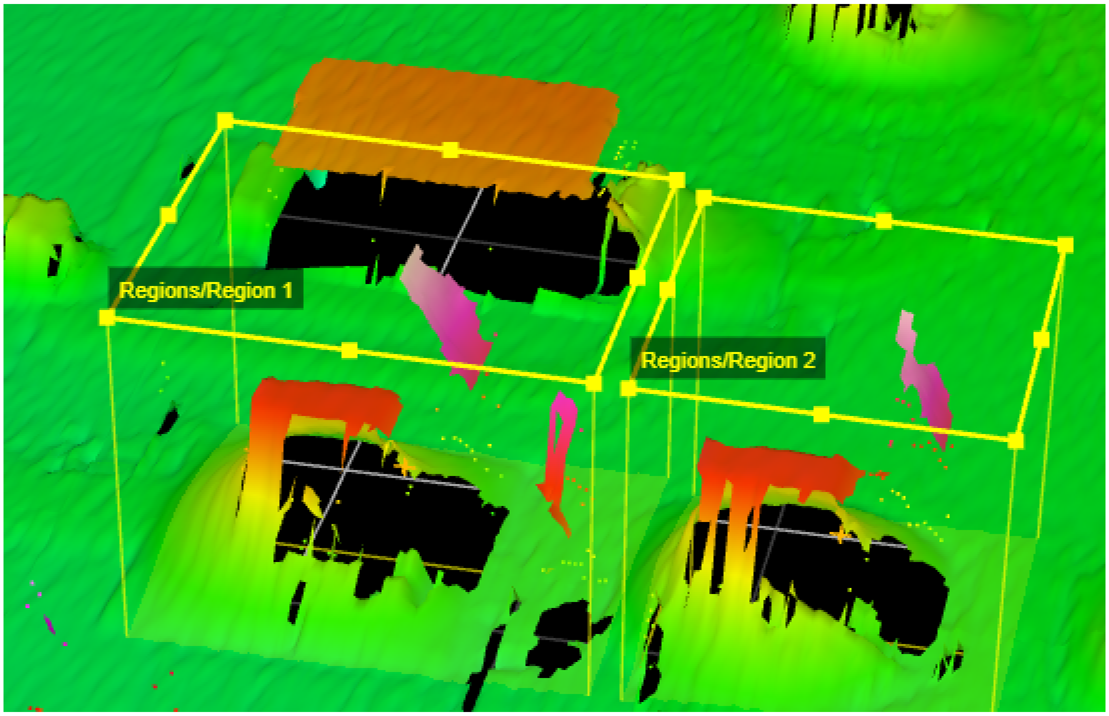
Measurement	Illustration
Length Determines the distance between the selected features along the Y axis.	
Height Determines the distance between the selected features along the Z axis.	
Distance Determines the direct, Euclidean distance between the selected features.	

Measurement	Illustration
Plane Distance Determines the distance between the selected features. The position of the lowest feature point is projected onto the XY plane of the highest feature point.	
Center X **Center Y** **Center Z** These measurements determine the X, Y, or Z position, respectively, of the center point between the selected features.	

Surface Direction Filter

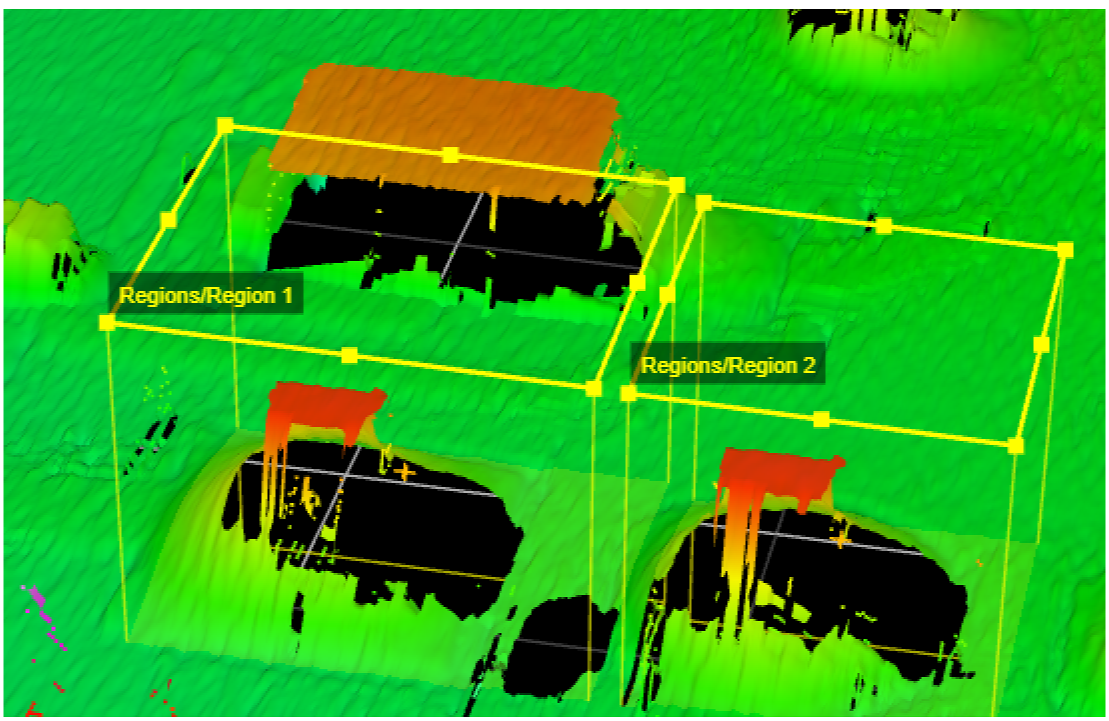
The Surface Direction Filter helps exclude unwanted data points based on their “orientation” (the polar angle relative to surrounding data points) in 3D space, for example, data points resulting from reflections. The tool can provide better results than median or height based filters. The tool lets you define up to 16 regions to which the filtering is applied.

For example, in the following scan data, noise appears to the right of two surface mount components on a PCB (pink areas). In this case, the "direction" of the noise is roughly 75 to 85 degrees, relative to Z.



Surface before direction filtering.

In the following scan data, the tool has removed the noise.



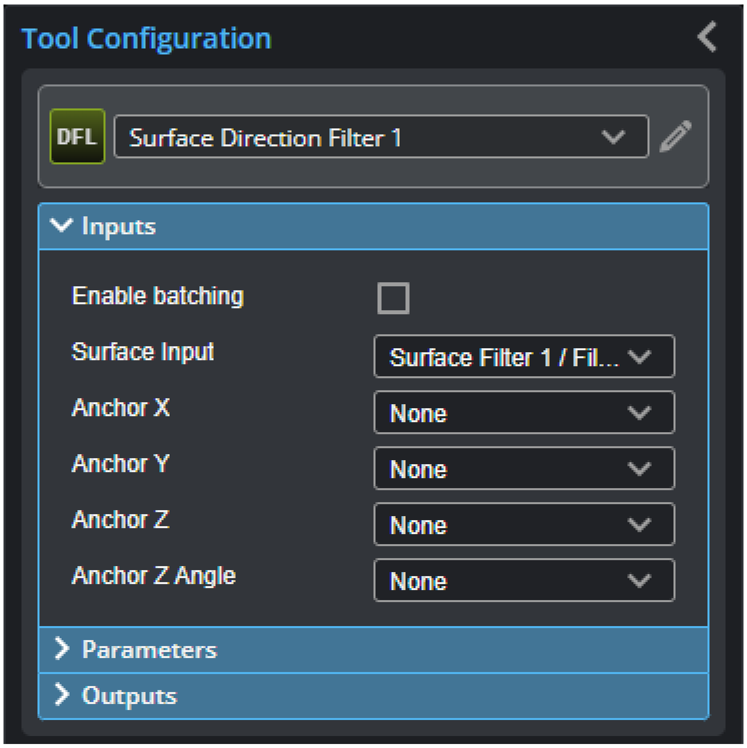
Surface after direction filtering.

Note that the tool's filtering parameters apply to all enabled regions. To apply different filtering parameters, you can add additional instances of the tool and chain them together, setting the input of subsequent instances to the output of the preceding instance.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

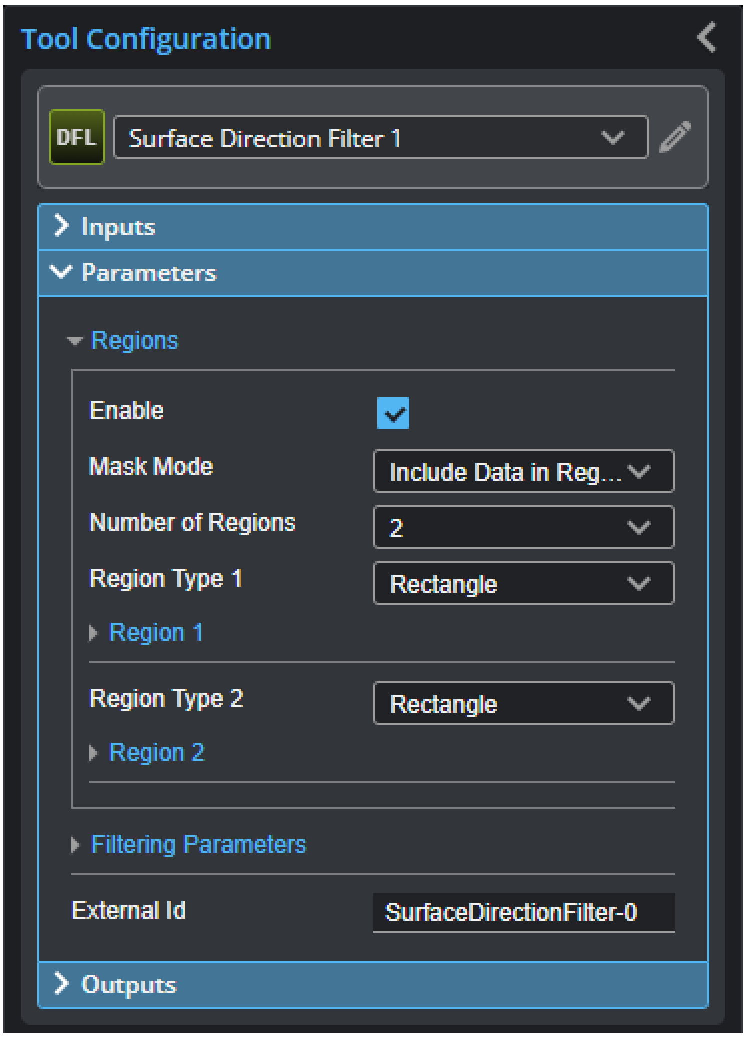
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

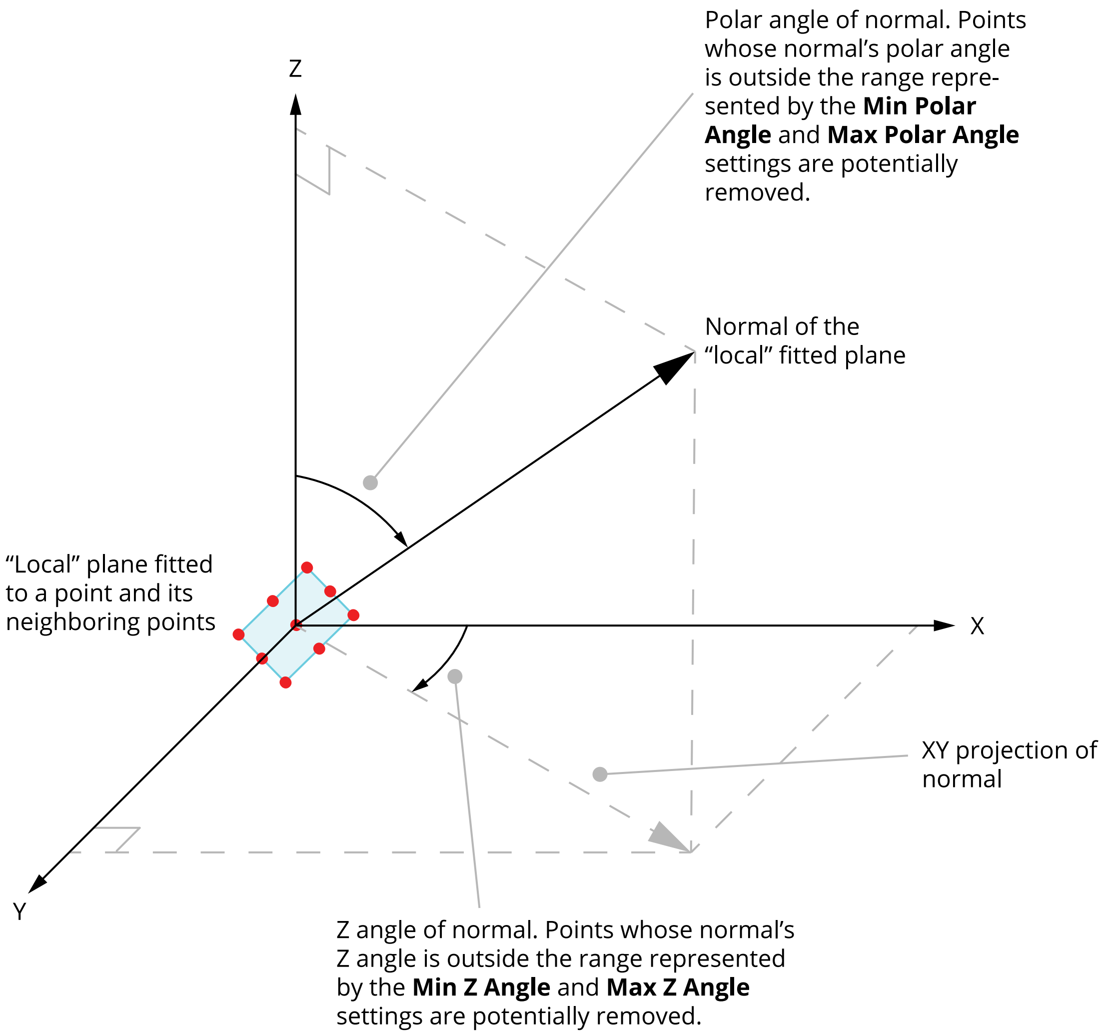
Name	Description
Anchor Z	
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



The following illustrates the angle parameters that control which data points are excluded in scan data:



The number of neighboring points shown above is for illustrative purposes only.

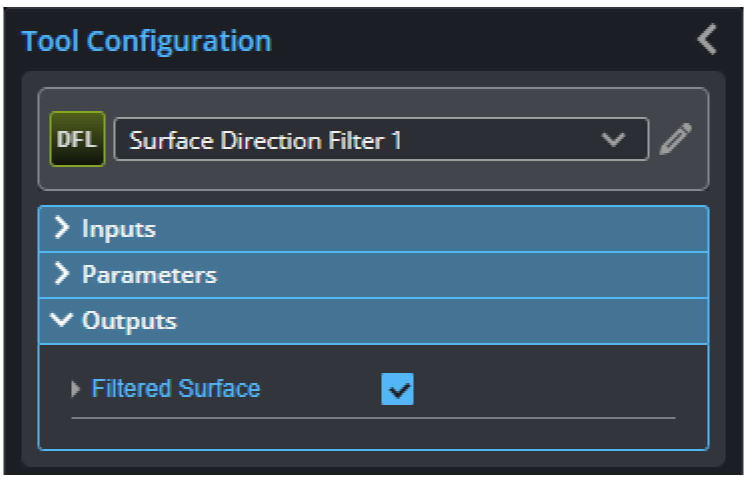
Parameters

Parameter	**Description**
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Filtering Parameters	When expanded, displays the filtering parameters the tool uses.
Min Z Angle Max Z Angle	The minimum and maximum acceptable angles around the Z axis of the XY projection of the normal of the surface surrounding a data point, where 0 degrees is defined as positive X and positive rotation is clockwise around the Z axis.
Min Polar Angle	The minimum and maximum acceptable angles of the normal of the surface

Parameter	Description
Max Polar Angle	surrounding a data point with respect to the Z axis.
Smooth Size	A mean filter applied to the surface data before calculating the normals in order to avoid abrupt normal changes due to noise.
Noise Removal	Eliminates noise that can be introduced by the tool's normal calculation.
Preserve Data Outside Region	When enabled, data outside the regions is preserved in the output. Otherwise, only the data in the regions is output.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Data

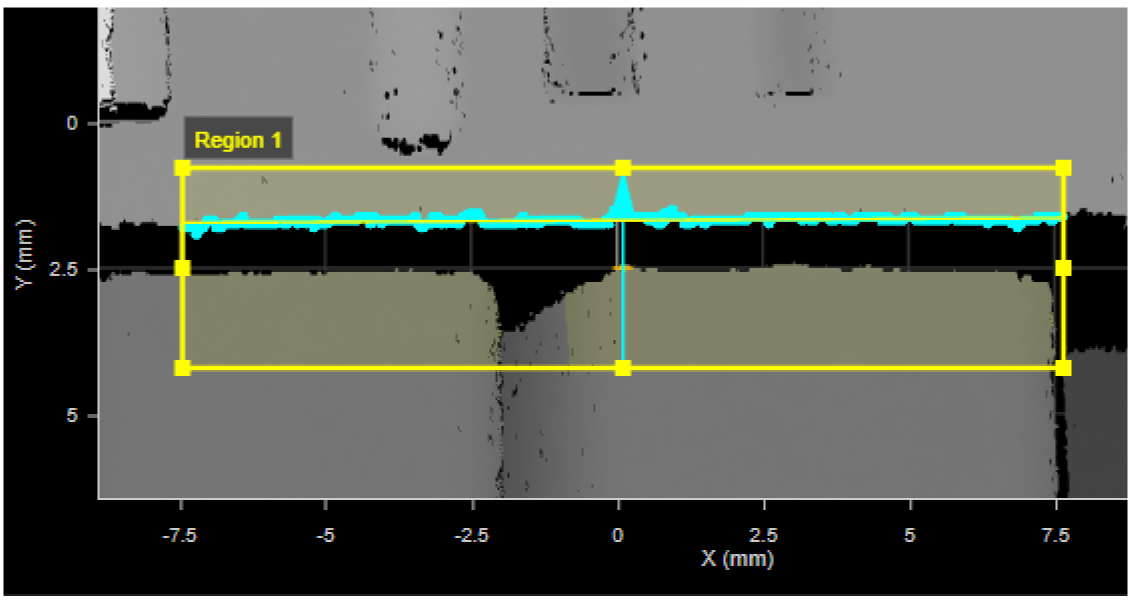
Type	Description
Filtered Surface	The surface after filtering.

Surface Edge

The Edge tool fits a line to a straight edge in the scan data, using either height map or intensity data. The tool can search for an edge using either a step (an abrupt change in the data) or a corner (a contiguous change in the shape of surface). The tool's settings help fit the line when multiple potential edges are in the region of interest. After the tool locates an edge, it returns the position (X, Y, and Z) of the center of the edge line in the region of interest. The tool also returns its angle around the Z axis, the step height between the upper and lower surfaces adjacent to the edge, minimum and maximum error points to either side of the line, and a point count.

You can use the Z Angle measurement of the edge line with some tools to perform angle anchoring, compensating for minor part rotations around the Z axis, greatly increasing repeatability between part scans; for more information see *Measurement Anchoring* on page 264.

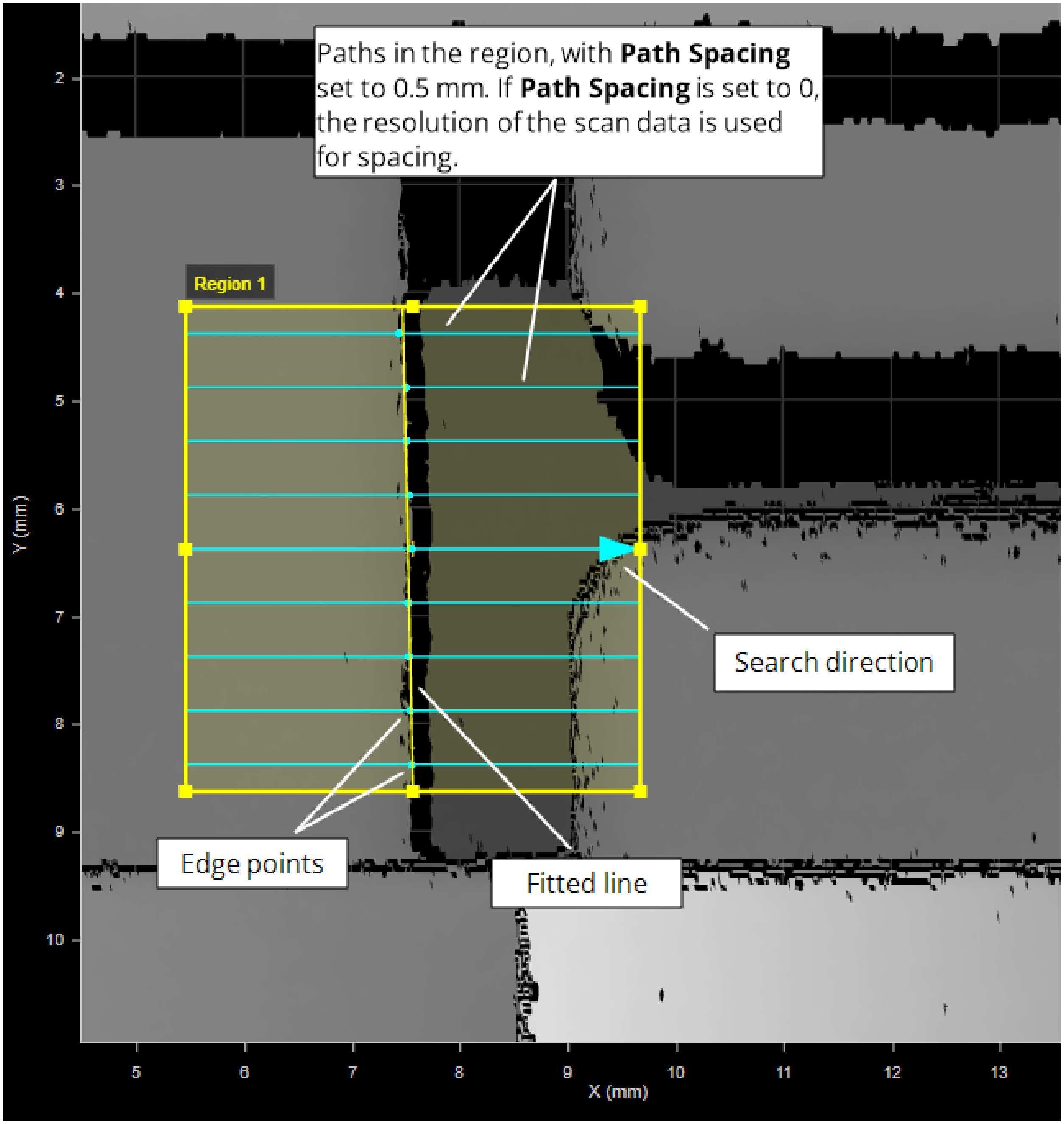
The tool can also generate edge line and center point geometric features that Feature tools can take as input for measurement. For more information on Feature tools, see *Feature Measurement* on page 692.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

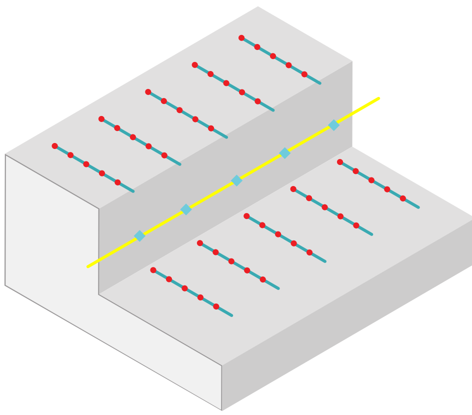
Paths and Path Profiles

To fit an edge line to the scan data, the Surface Edge tool overlays evenly spaced, parallel *paths* (light blue lines in the interface; see below) in the defined region of interest. (Note that the paths are only displayed if **Path Spacing** is greater than 0.)



For each path, a profile is generated internally from the height map's data points that fall under or, optionally, near the path. The tool then examines each path profile for steps (changes in height) that meet the criteria set by the tool's settings, such as minimum height, direction (whether it is rising or falling), and so on.

Red dots are data points from the scan data that fall under paths (light blue lines).

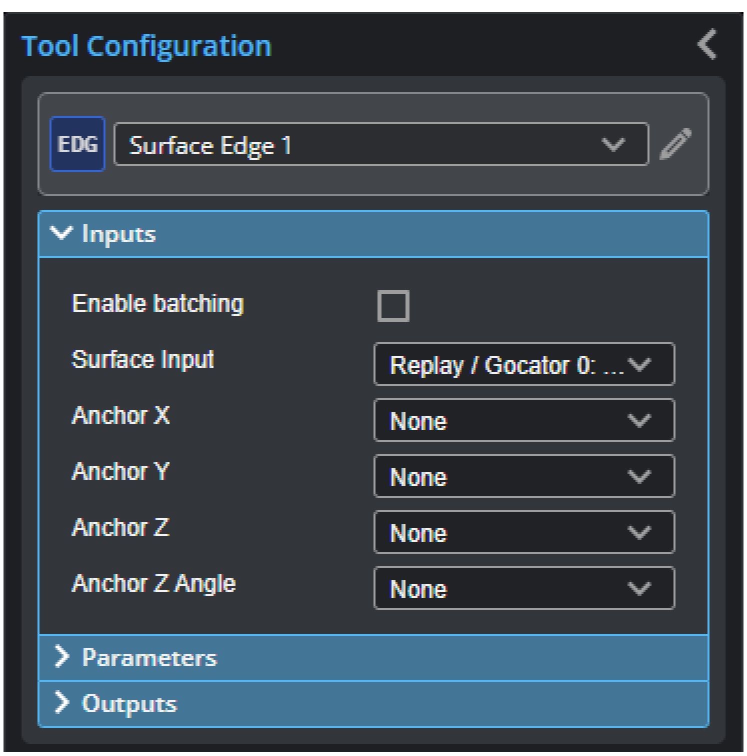


A single path profile extracted from a path.

For the step on each path profile that matches the settings, the tool places an edge point between the upper and lower area (light blue diamonds in the interface). The tool then fits a line to those edge points (yellow line in the interface). You can choose the orientation of the paths around the Z axis to accommodate different edge orientations.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



🗋	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

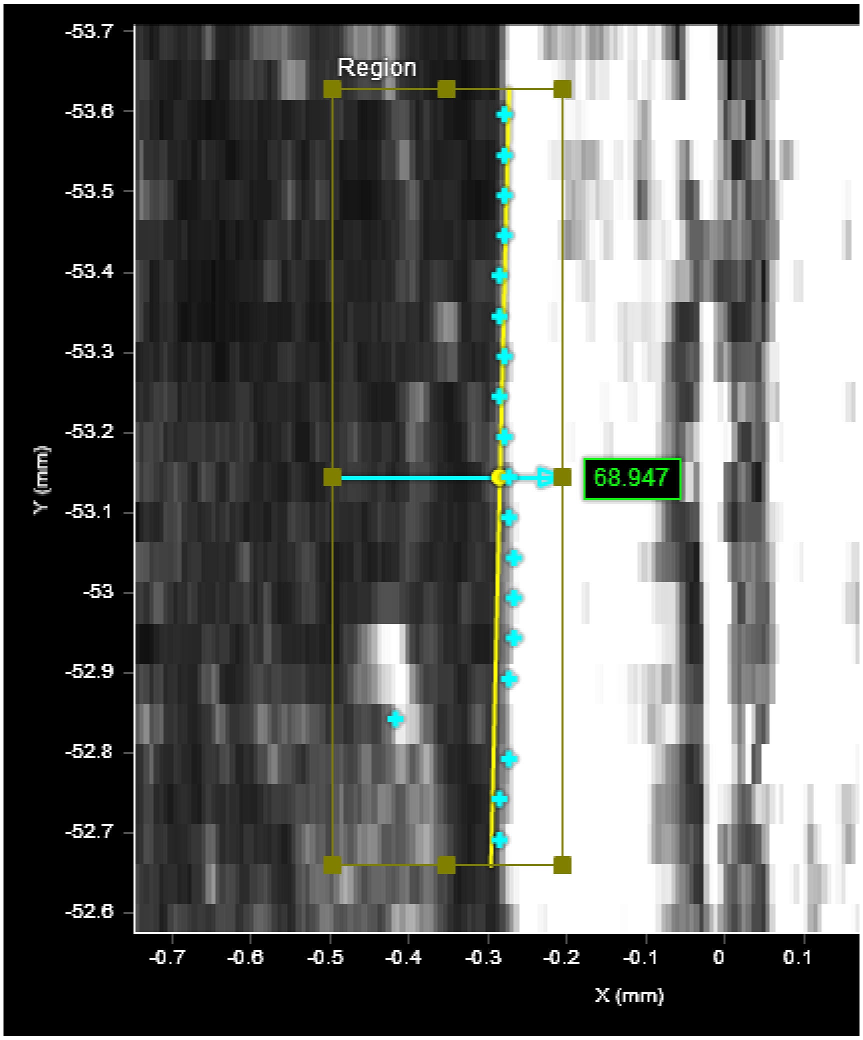
Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

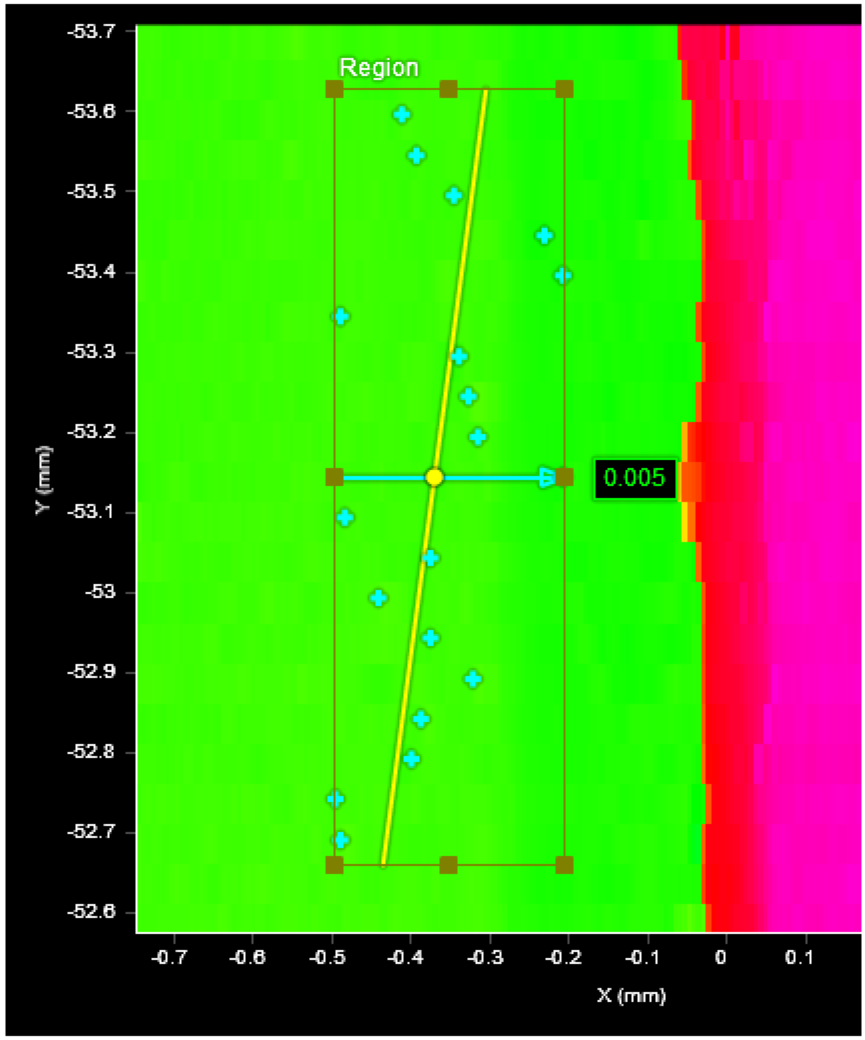
Parameters

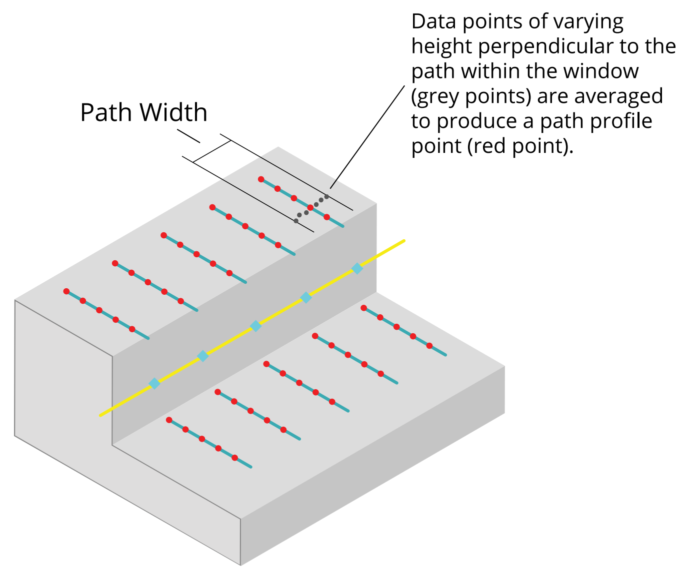
You configure the tool's parameters in the expandable **Parameters** section.


Tool Configuration
EDG
Surface Edge 1
Inputs
Parameters
Use Intensity
Number of Regions
1
Region 1
Search Direction
0 degrees
Fixed Angle
Path Spacing
0.500 mm
Path Width
0.000 mm
Outlier Fraction
0.000 %
Edge Detection Mode
Step
Selection Type
Best
Step Direction
Falling
Absolute Threshold
0.000 mm
Use Relative Threshold
Step Smoothing
0.000 mm
Step Width
0.000 mm
Max Gap
0.000 mm
Include Null Edges
Null Fill Value
0.000 mm
Edge Mode
Projected
Show Detail
External Id
SurfaceEdge-0
Outputs

Parameters

Parameter	Description
Use Intensity	Uses intensity data rather than height data to find an edge. Useful when color differences on a flat area of a target, which would not be detected using height map data, are distinct, letting you use the detected "line" as an anchor source or perform geometric feature measurements.  ***Use Intensity** enabled (intensity view): Surface Edge tool finds the edge using intensity data.*

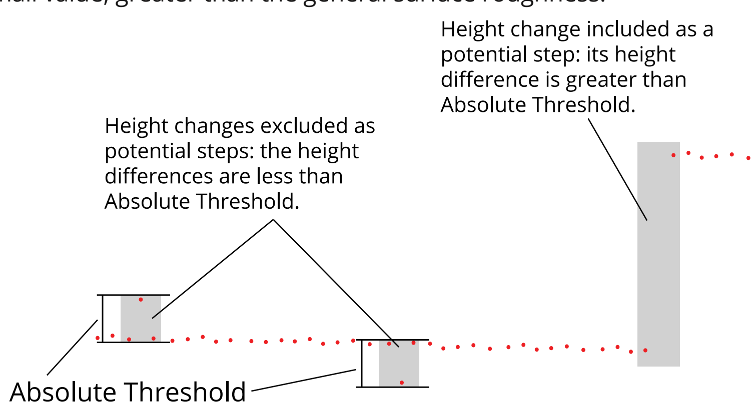
Parameter	Description
	 *__Use Intensity__ disabled (heightmap view of the same area): Surface Edge tool unable to find edge using height data.*
Number of Regions	The number of regions the tool will use to fit the line. You must configure each region (see *Region {n}* below). Using multiple regions allows you to fit a line to an edge that is not straight along its entire length or that is not continuous.
Region {n}	The region or regions the tool uses to fit a line. For more information, see *Regions* on page 250. The **Search Direction** setting applies to all of the regions. You can configure the **Z Angle** of each region independently to accommodate the particularities of the feature or target (for example, to exclude unwanted scan data next to one of the regions in the fitting of the line to the edge).
Search Direction	The search direction for steps, specified as an orientation around the Z axis, relative to the X axis. Can be 0, 90, 180, or 270 degrees. Choose a value that is

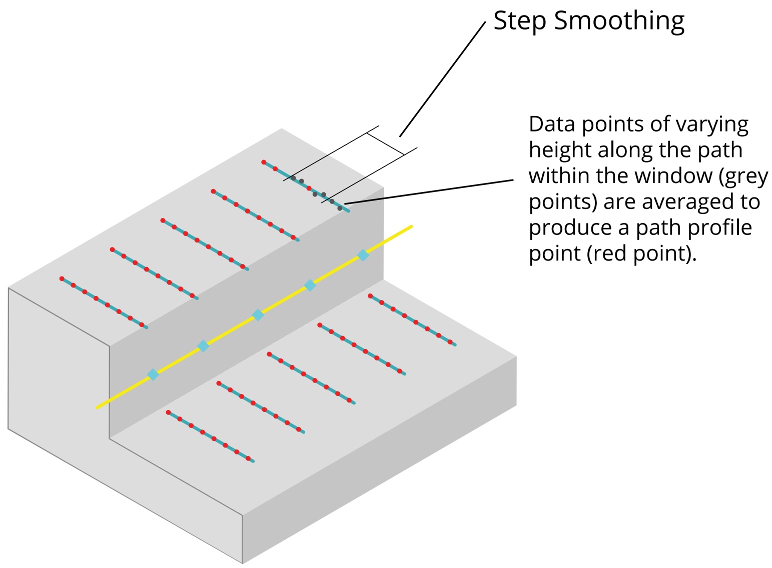
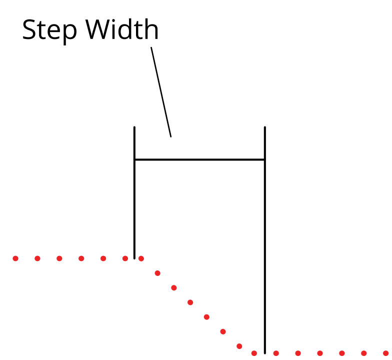
Parameter	Description
	roughly perpendicular to the edge on the target. The direction is indicated by a light blue arrow in the data viewer.
Fixed Angle *Dependent settings* Fixed Angle Value	When this option is enabled, the value in **Fixed Angle Value** replaces the value the Z Angle measurement returns. Useful when the angle of the feature is known and noise in the scan data could otherwise cause the measurement to return an incorrect angle.
Path Spacing	Sets the spacing between paths in the measurement region used to extract the profiles that determine the edge. A higher number of paths results in a higher number of edge points, which makes the fitting of the edge line more accurate. However, a higher number of edge points results in a greater tool execution time. When **Path Spacing** is set to 0, the resolution of the scan data is used as the basis for spacing. No paths are displayed in the data viewer in this case.
Path Width	The size of the windows perpendicular to the path used to calculate an average for each data point on a path profile. Useful to average out noise along the path caused by reflections, and so on.  If **Path Width** is set to 0, no averaging is performed (only the data point under the path is used). For averaging along the path, use **Step Smoothing** (see below).
Outlier Fraction	The percentage of outlier points to exclude. Setting this to a small value can help the tool fit the line better to the edge.
Edge Detection Mode	One of the following: Step or Corner. **Step**: Searches for steps on each path profile. For additional settings when you choose this mode, see *Step Edge Detection Mode Parameters* on the next page. **Corner**: Searches for slopes on each path profile. When you choose this mode, several of the tool's parameters are hidden.

Parameter	Description
Selection Type Corner Type	Determines which step (when **Edge Detection Mode** is set to Step) or corner (when **Edge Detection Mode** is set to Corner) the tool uses on each path profile when there are multiple steps or corners in the profile. An edge point is placed on each chosen step or corner. Steps must pass the criteria of the tool's **Absolute Threshold**, **Step Direction**, and **Relative Threshold** settings (see *Step Edge Detection Mode Parameters* below). **Best**: Selects the greatest step or corner on each path profile. **First**: Selects the first step or corner on each path profile. **Last**: Selects the last step or corner on each path profile. When **Edge Detection Mode** is set to Corner, the following additional options are available in Corner Type: **Top**: Selects the top-most corner on each path profile. **Bottom**: Selects the bottom-most corner on each path profile.
Show Detail	When disabled, hides the light blue path lines and edge points.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

The following parameters are only displayed if you set **Edge Detection Mode** to Step.

Step Edge Detection Mode Parameters

Step Direction	Determines whether the expected step rises or falls along the path, moving from left to right. Either **Rising**, **Falling**, or **Rising or Falling**.
Absolute Threshold	When **Use Intensity** is disabled, the setting specifies the minimum *height* difference between points on a path profile for that step to be considered for an edge point. The setting can be used to exclude smaller steps on a part that should not be considered for an edge, or to exclude height differences caused by noise. When used in conjunction with **Relative Threshold**, **Absolute Threshold** is typically set to a small value, greater than the general surface roughness.  When **Use Intensity** is enabled, the setting specifies the minimum difference in intensity. (**Acquire Intensity** must enabled in the [**Scan Mode** panel](#).)

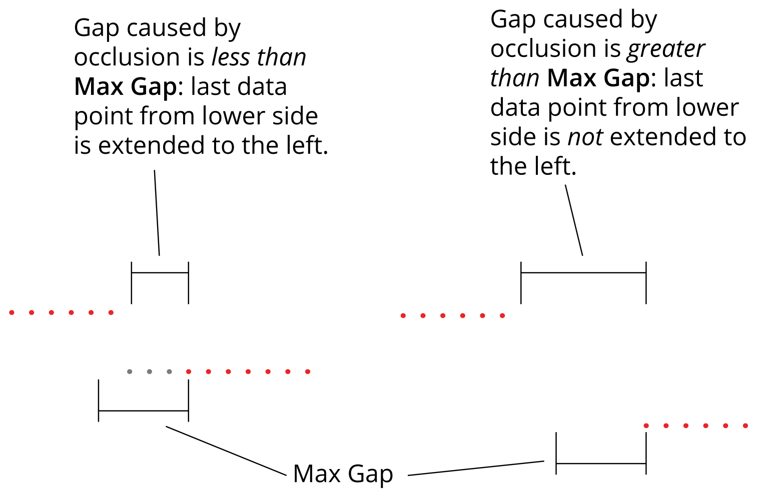
Use Relative Threshold	When this option is enabled, the **Relative Threshold** field is displayed.
Relative Threshold	The value for the relative threshold. The tool calculates a relative threshold by scaling the greatest height or intensity difference found on the path profiles by the percentage in **Relative Threshold**. This lets you configure the tool without knowing the actual step height in advance, and is useful for edges with varying step height. For a height or intensity difference to be considered a valid step, both **Absolute Threshold** and **Relative Threshold** must pass.
Step Smoothing	The size of the windows along the path used to calculate an average for each data point on a path profile. The setting is useful for averaging out noise.  If **Step Smoothing** is set to 0, no averaging is performed (only the data point under the path is used). For averaging perpendicular to the path, use **Path Width** (see above).
Step Width	The distance, along a path profile, separating the points used to find steps on a path profile.  The setting is useful when you must detect a slope as an edge, rather than a sharply defined edge: setting **Step Width** to a value greater than the width of the

edge ensures that the tool measures the height difference between the flat regions on either side of the edge. As a result, the height of the step is accurately measured, and the edge is correctly located.

> Setting **Step Width** wider than necessary can reduce the precision of edge location.

Max Gap

Fills in regions of missing data caused by an occlusion near the desired edge. Use this setting when continuity on the target is expected. When **Max Gap** is set to a non-zero value, the tool holds and extends the last data point on the low side next to an edge across a gap of null points, up to the distance specified in **Max Gap**.



Include Null Edges

Indicates whether null points (points where no height or intensity value is available, due to dropouts or regions outside of the measurement range) are filled with the value in **Null Fill Value** as a general “background level.” If **Use Intensity** (see above) is enabled, the intensity value in **Intensity Null Fill Value** is also used.

A typical example is a discrete part produced by the part detection of an object sitting on a flat background (for more information on part detection, see *Profile Part Detection* on page 373). The background is not visible in the part, so the tool assumes that any null region are at the background level.

> To find edges along a region of null points, you must use either this option and an appropriate value in **Null Fill Value** (and **Intensity Null Fill Value** if **Use Intensity** is enabled) or **Max Gap**. Otherwise, only edges within areas of contiguous data will be detected.

Null Fill Value

The height value (in mm) used to replace null points not filled by **Max Gap** when **Include Null Edges** is enabled.

Intensity Null Fill Value

The intensity value (0-255) used to replace null points when **Include Null Edges** and **Use Intensity** are enabled.

Edge Mode

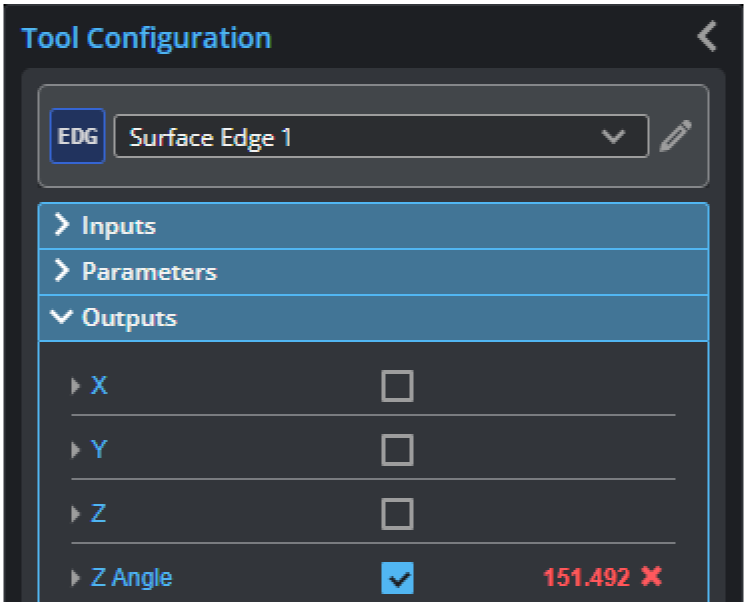
One of the following:

Projected: The line fitted to the edge is projected onto the XY plane. This mode is typically used with an edge that is parallel to the XY plane.

3D: The line fitted to the edge follows the slope of the edge. This mode is typically used with a sloped edge.

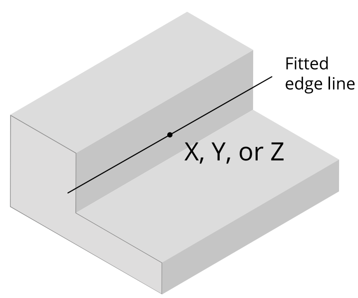
Outputs

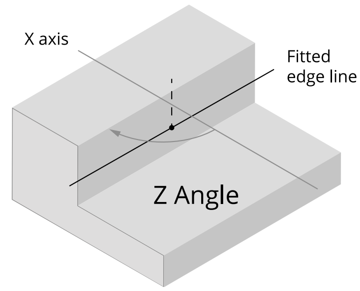
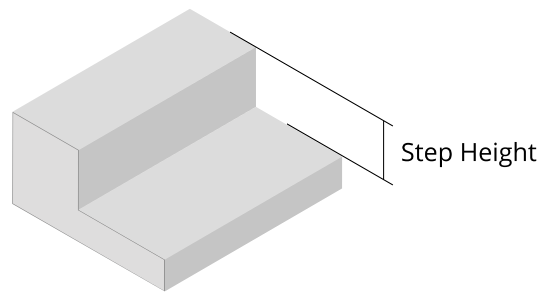
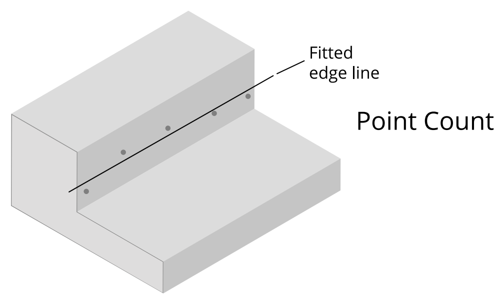
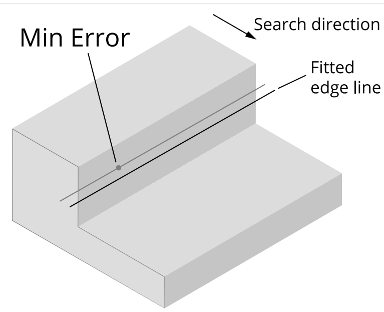
Most tools provide measurements, geometric features, or data as outputs.

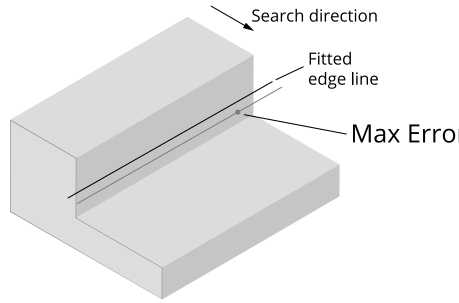


You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Returns the X position of the center point of the fitted edge line.	
Y Returns the Y position of the center point of the fitted edge line.	
Z Returns the Z position of the center point of the fitted edge line.	

Measurement	Illustration
Z Angle Returns the rotation, around the Z axis, of the fitted edge line. Rotating the measurement region has no impact on the angle that is returned unless a different edge is detected. Useful for using minor variations in the rotation of an edge on target as an anchor for other measurements. For more information, see *Measurement Anchoring* on page 264.	
Step Height Returns the height of the step, calculated by averaging the step heights of all of the path profiles. (When **Use Intensity** is enabled, the value returned is the difference in intensity.) This measurement returns Invalid when **Edge Detection Mode** is set to Corner.	
Point Count The number of points used to fit the line. Useful for determining if the number of points is above an acceptable minimum.	
Min Error **Max Error** These measurements return the distances of the point furthest before the line (Min Error) and the point furthest after the line (Max Error), based on the search direction specified in the tool. The measurements ignore points excluded using the **Outlier Fraction** parameter.	

Measurement	Illustration
	

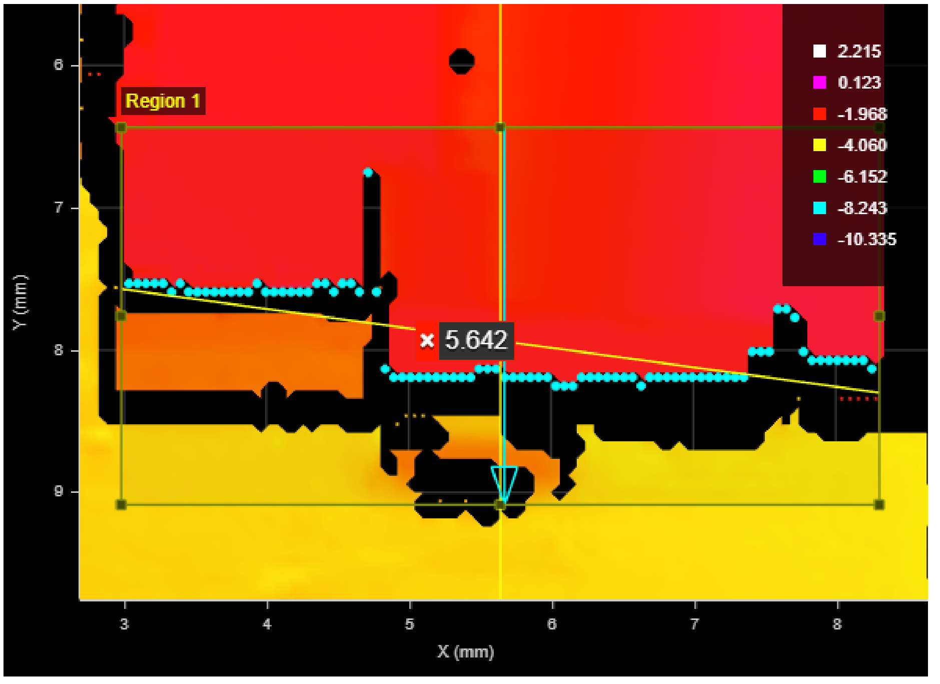
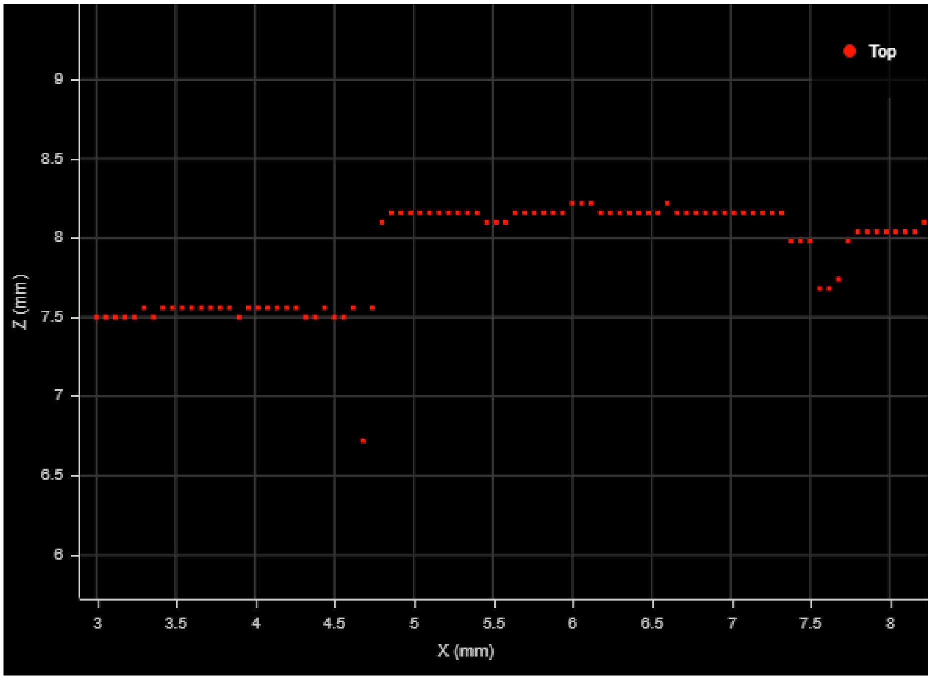
Features

Type	Description
Edge Line	The fitted edge line.
Center Point	The intersection point of the fitted edge line and the line representing the search direction through the center of the region of interest.
Edge Plane	A plane on the XZ axes at the fitted edge line.

For more information on geometric features, see *Geometric Features* on page 262.

Data

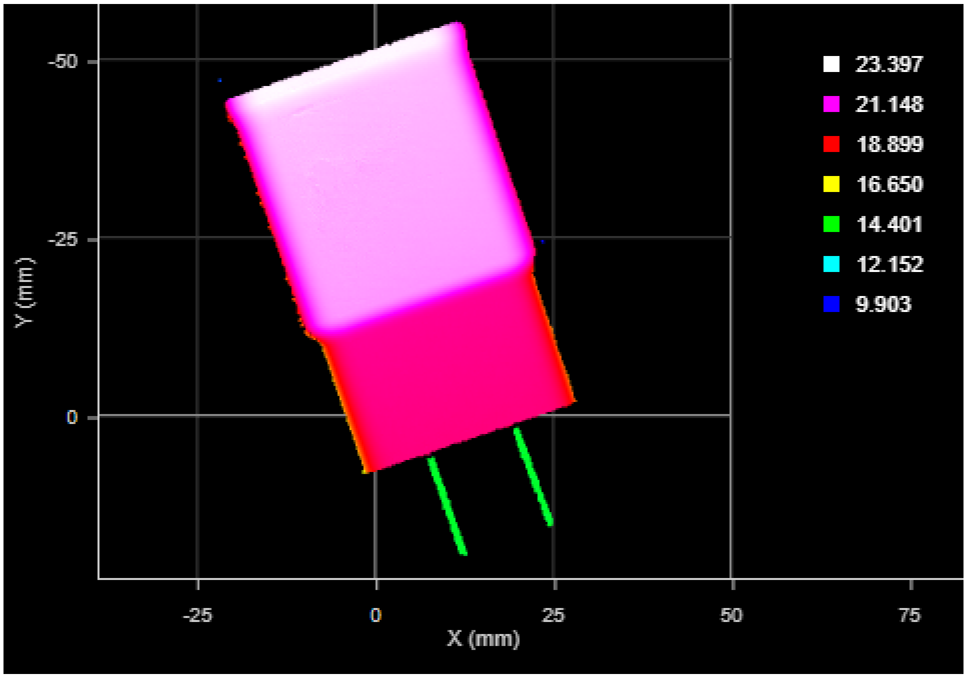
Type	Description
Profile Point Cloud Profile Region {n}	A point cloud profile (Profile Point Cloud) and one or more uniform spacing profiles (Profile Region {n}) representing the edge, respectively, made up of the tool's edge points. The XY positions of the edge points on the surface (cyan dots below) are represented as the XZ positions of the profile points, where X => X and Y => Z. Given the following edge, the resulting profile is shown further below:

Type	Description
	 The profile is mirrored vertically when compared to the edge: Note how the single edge point toward the top of Region 1 in the surface data above is at the bottom of the extracted profile (below). 

Surface Ellipse

The Ellipse tool provides measurements for the major and minor axis lengths of an ellipse roughly aligned to the part's shape in the XY plane, and also for the ratio of the major and minor axis lengths and for the orientation angle of the ellipse. The tool is typically used to find the general orientation of a part, for example, potatoes on a conveyor that are longer in one dimension than the other.

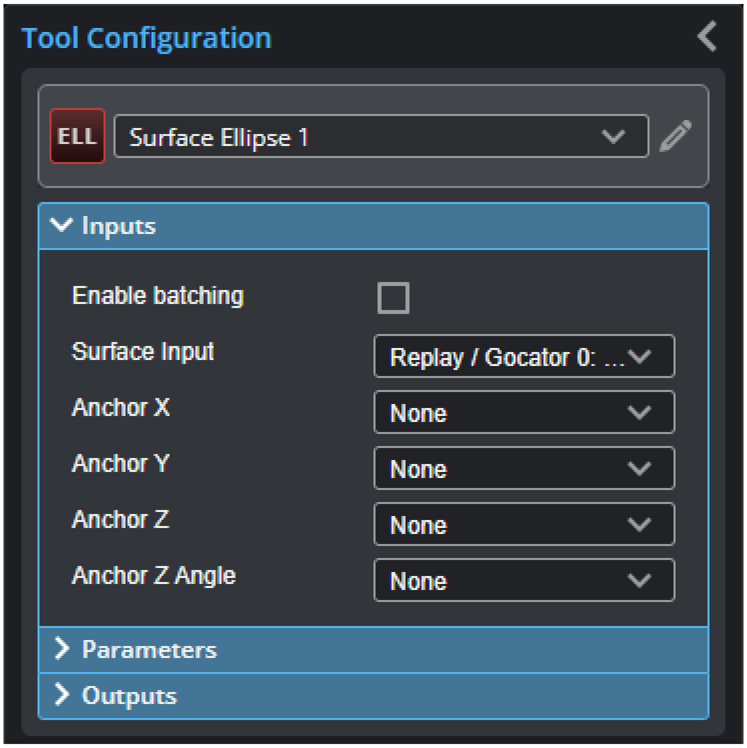
Note that the ellipse fit is not the minimum area ellipse around the data. (Technically, it is the ellipse with matching moments as the data.) For surfaces with no holes, this results in an ellipse approximately the same size and orientation of the part. But for surfaces with holes, the resulting ellipse can be larger than the part.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



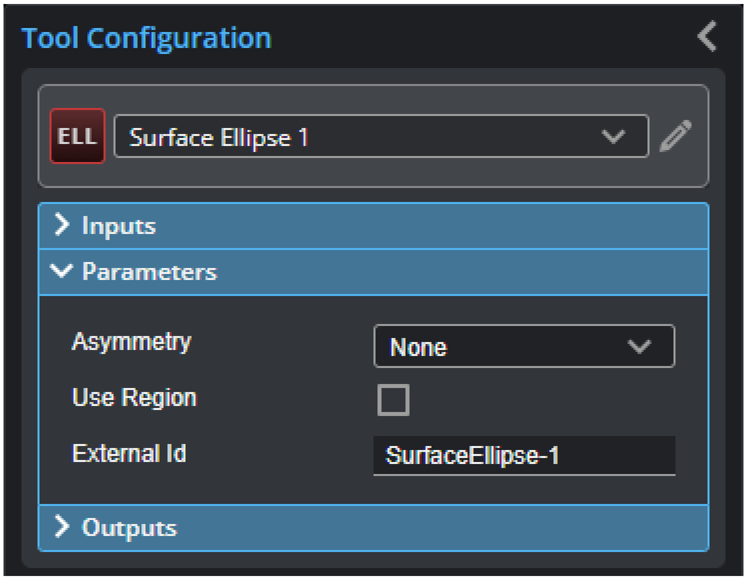
🗋	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

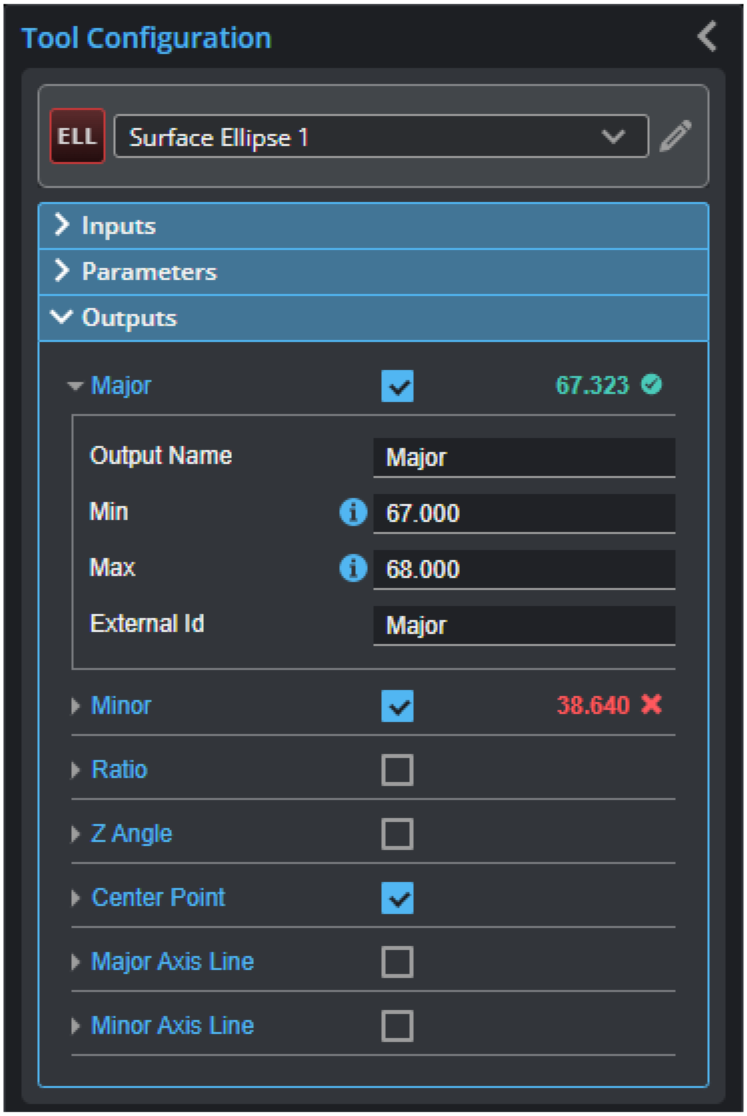


Parameters

Parameter	Description
Asymmetry	Resolves the orientation of an object over 360 degrees. The possible values are: 0 – None 1 – Along Major Axis 2 – Along Minor Axis
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

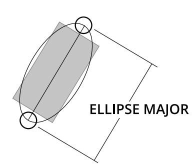
Most tools provide measurements, geometric features, or data as outputs.

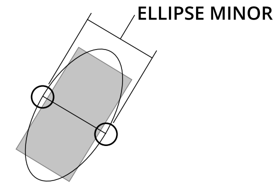
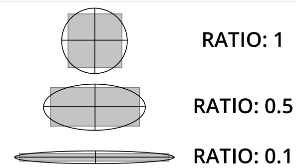
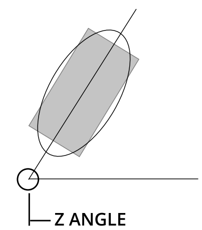


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Major Determines the major axis length of an ellipse fitted to the part's area in the XY plane.	

Measurement	Illustration
Minor Determines the minor axis length of an ellipse fitted to the part's area in the XY plane.	
Ratio Determines the minor/major axis ratio of an ellipse fitted to the part's area in the XY plane.	
Z Angle Determines the orientation angle of an ellipse fitted to the part's area in the XY plane.	

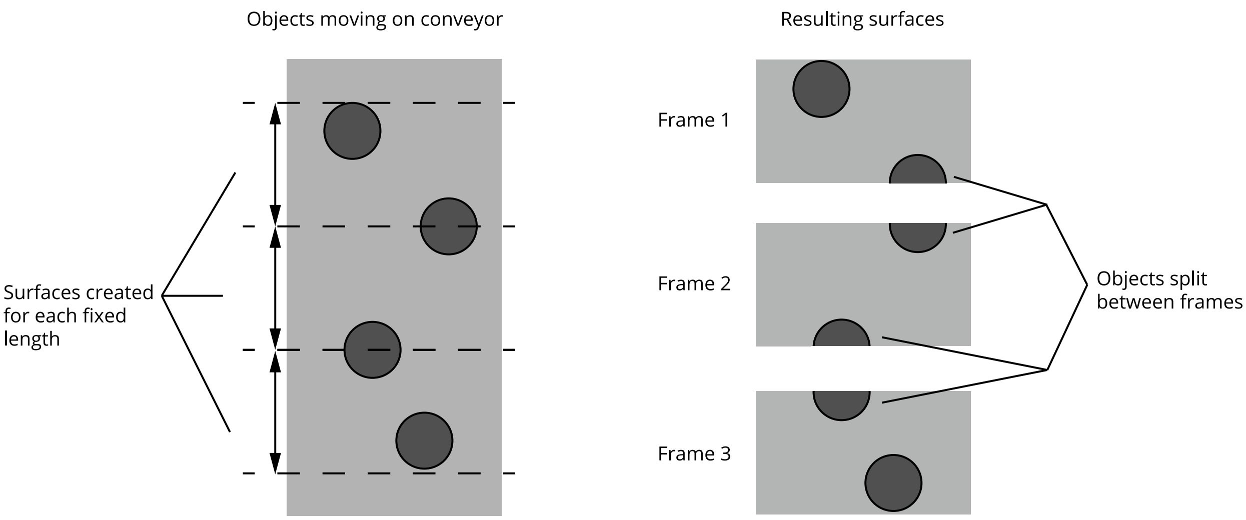
Features

Type	Description
Center Point	The center point of the fitted ellipse.
Major Axis Line	A line representing the major axis of the fitted ellipse.
Minor Axis Line	A line representing the minor axis of the fitted ellipse.

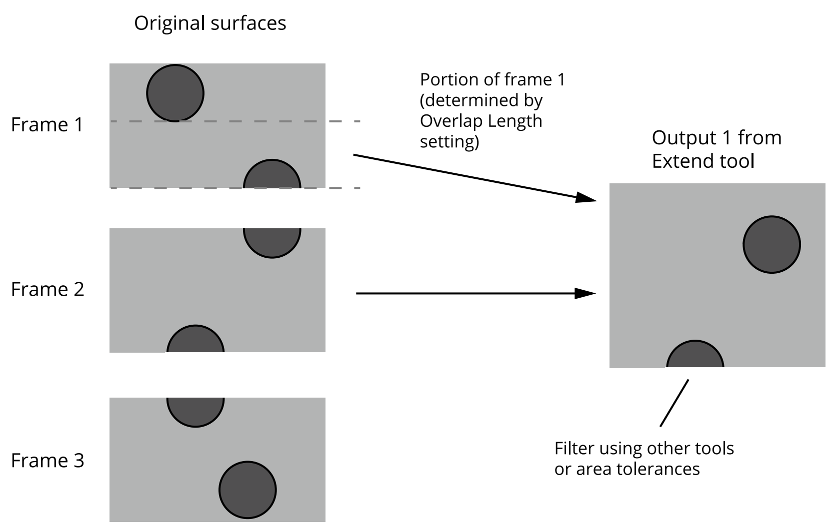
For more information on geometric features, see *Geometric Features* on page 262.

Surface Extend

The Extend tool creates a new surface by appending part of the previous frame's data to the current frame's data. The tool outputs the new surface data, which can be used as input by other tools. The tool is especially useful when scans are performed using fixed length surface generation, where parts might be split between two frames. (For more information on surface generation, see *Surface Generation* on page 199.)



The following shows how the tool combines data:

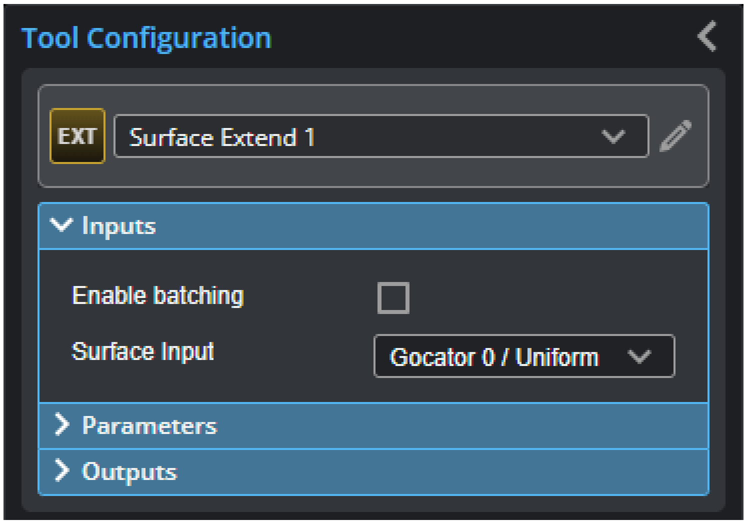


Data is only appended in one direction. Partial objects in the resulting surface output from the tool must be filtered out using downstream tools, for example, excluding them based on the expected area.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



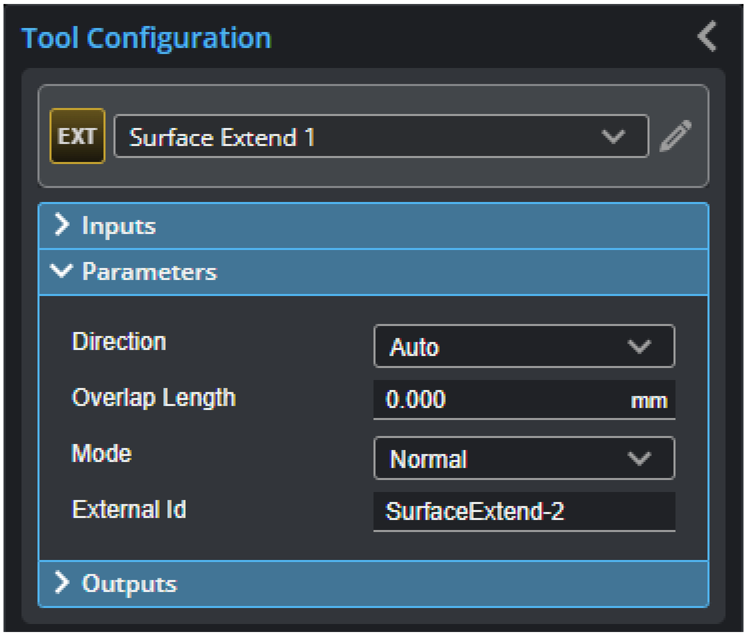
	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

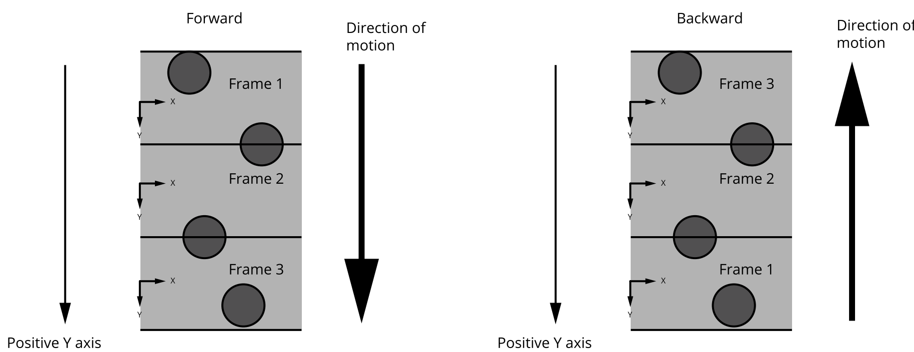
Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>Direction</td><td>
Determines whether the previous frame's data is appended above or below the current frame's data.


One of the following. Note that these settings depend on whether the trigger source has been set to Encoder (see Triggers on page 206) and the orientation of the sensor.

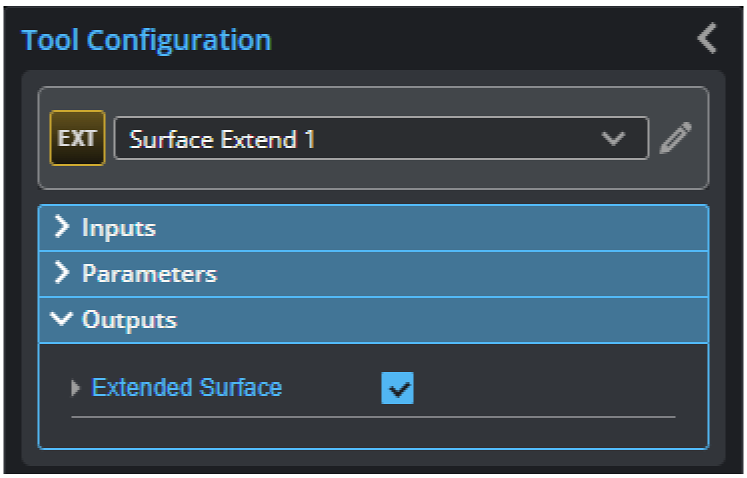
Auto: Choose this when Encoder is selected as the trigger source, in which case the tool will detect the direction of travel relative to encoder increase / decrease.
Forward: Choose this option when the trigger source is not set to Encoder and the direction of motion matches the sensor's positive Y axis.
Backward: Choose this option when the trigger source is not set to Encoder and the direction of motion is the opposite of the sensor's positive Y axis.

</td></tr>
<tr><td>Overlap Length</td><td>The amount, in millimeters, of the previous frame's data to append to the current frame's data. The combination will be output as tool data. Choose the overlap length to accommodate the size of your scan targets.</td></tr>
</table>

Parameter	Description
Mode	Determines the mode of the tool. One of the following: • **Normal**: The tool's output is unlocked and functions normally. • **Lock**: Lets you lock the current processing and outputs of the tool. Useful when you need to add another tool that will use this tool's output (for example, a Surface Section tool). If you do not lock the tool, as soon as you add the other tool, the output is cleared, which means you must re-execute the combined output again to configure the additional tool. Be sure to unlock the tool after you have configured any other tools.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Data

Type	Description
Extended Surface	Data containing an extended surface, available for use as input in the **Stream** drop-down in other tools.

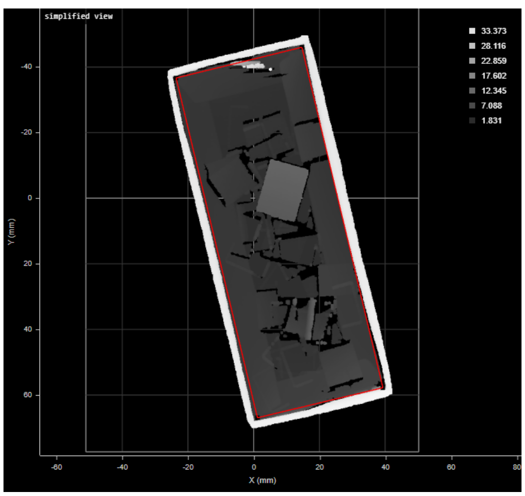
Surface Feature Mask

The Surface Feature Mask dynamically defines a polygon or circle mask using multiple Point geometric features produced by other tools to crop a uniform surface and output it. You can use this tool in void fill applications with bins of varying sizes and varying orientations.

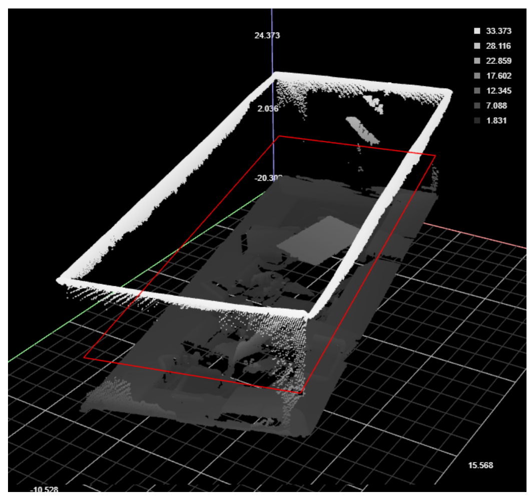
The following is a possible tool chain for calculating the fill volume of rectangular bin using Surface Feature Mask:

- Surface Bounding Box tool: Finds the orientation and position of the center of a box that must be filled. You would use the tool's X, Y, and Z Angle measurements as anchors for the other tools.
- Surface Edge tools (one for each edge of the bin): Using the Bounding Box tool's measurements as anchors, these tools find the edges of the bin.
- Feature Intersect tools: Using pairs of lines from the Surface Edge tools as inputs, these tools return Point geometric for the corners of the bin.
- Surface Feature Mask tool: Using the Point geometric features from the Feature Intersect tools as inputs, this tool outputs Surface data delimited by corners of the bin. To exclude the edges of the bin itself, use the **Padding Distance** parameter to exclude the bin walls.

When only two Point geometric features are passed as input to the tool, it creates a circular mask, where one point is the circle's center point, and the other is a point on the circle.



2D View

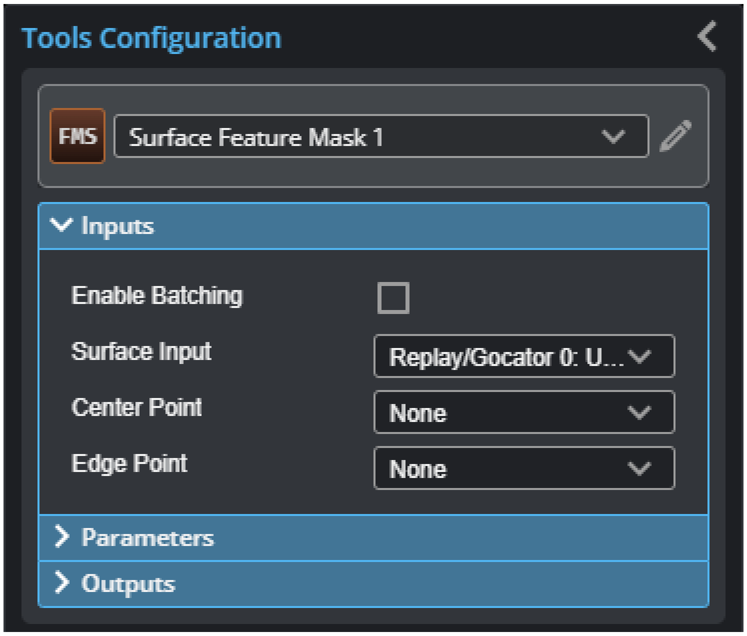


3D View

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



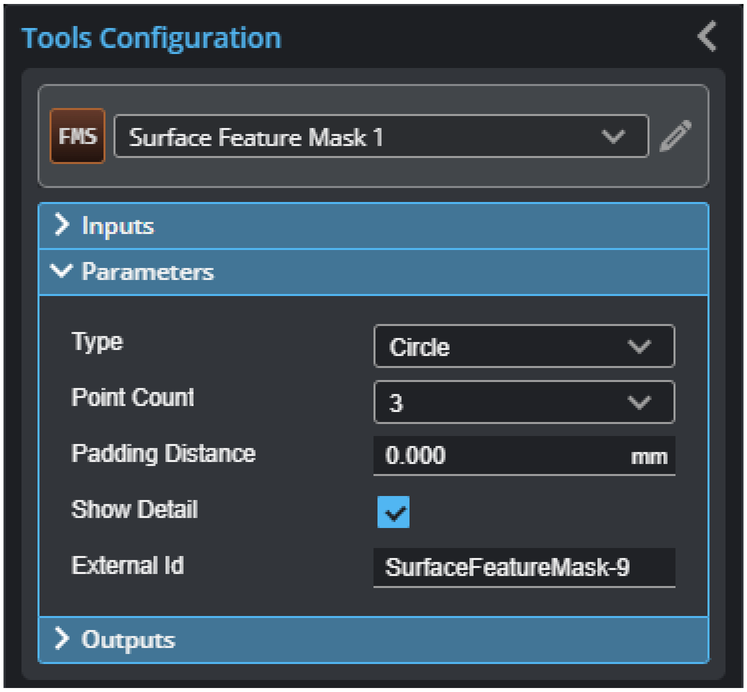
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Point {n}	The nth Point geometric feature. Only displayed if you set **Point Count** to 3 or higher.
Center Point	The center Point geometric feature of the circle. Only displayed if you set **Type** to Circle and **Point Count** to 2.
Edge Point	The Point geometric feature on the circle. Only displayed if you set **Type** to Circle and **Point Count** to 2.
Surface Input	The data the tool applies measurements to or processes.

For more information on geometric features, see *Geometric Features* on page 262.

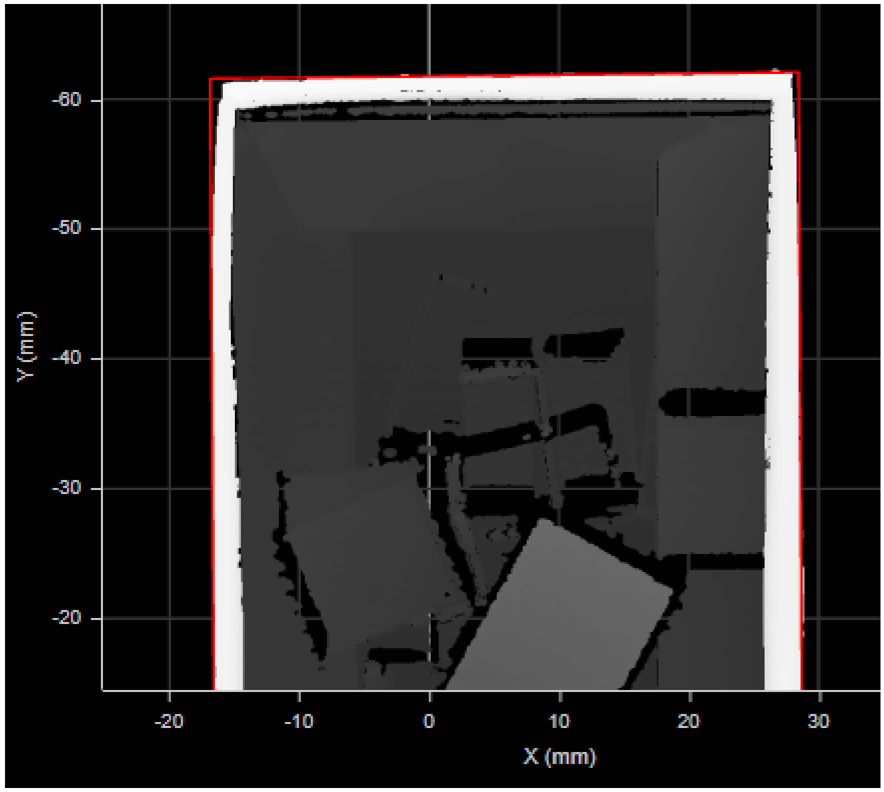
Parameters

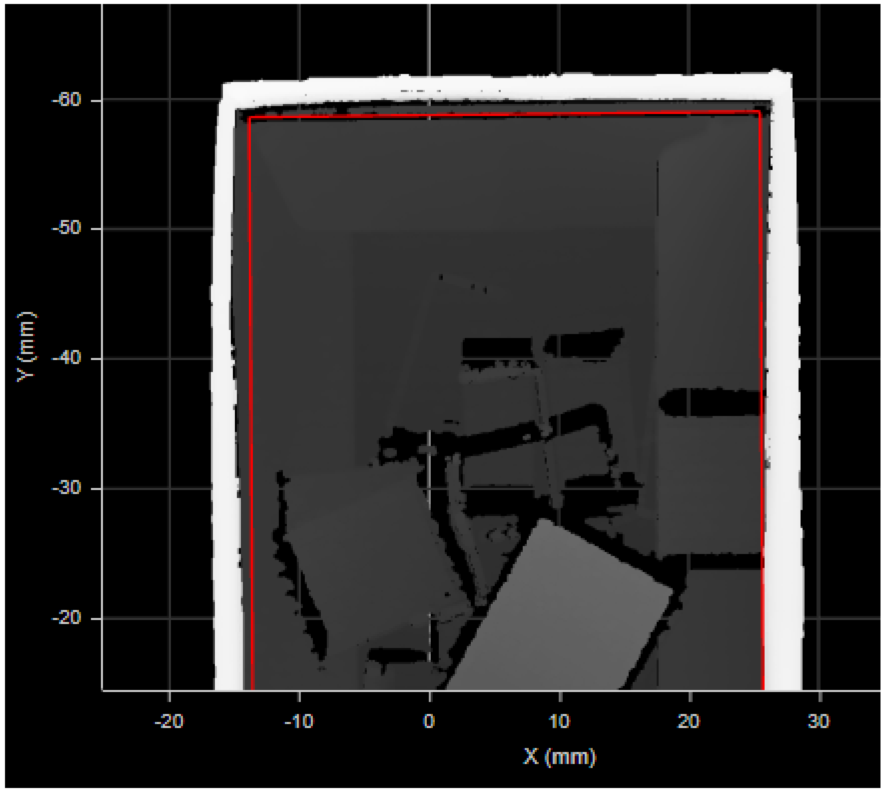
You configure the tool's parameters in the expandable **Parameters** section.

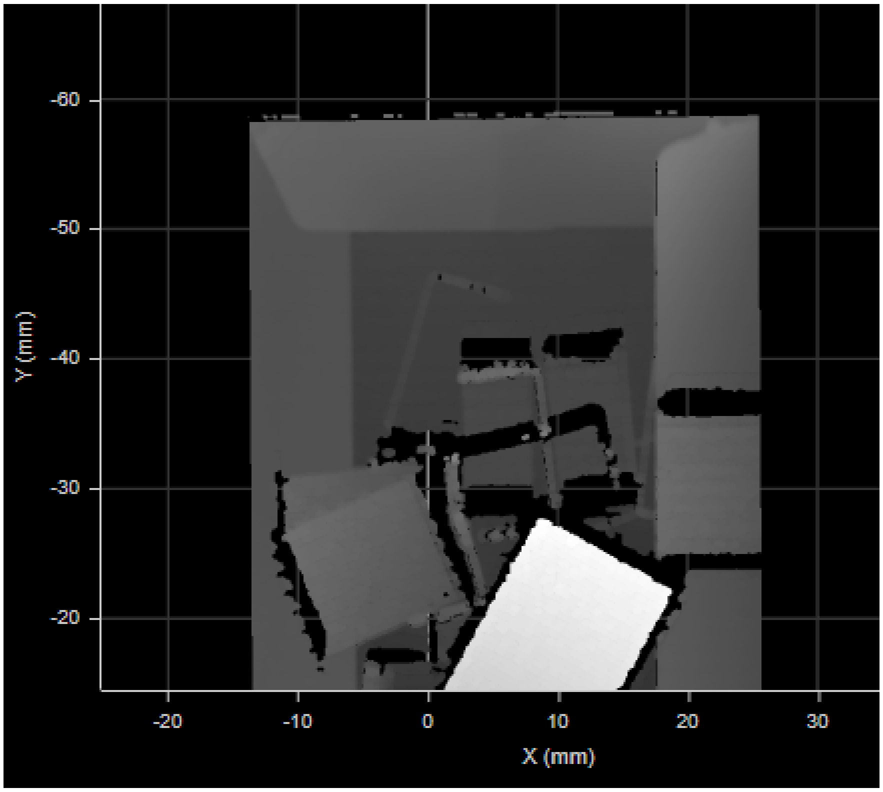


Parameters

Parameter	Description
Type	The shape type of the mask. It supports two options: **Polygon** - The tool needs at least 3 Point geometric features. **Circle** - The tool needs 2 or 3 point geometric features.
Point Count	The count of the input point features. For the Polygon **Type** option, the tool supports integers in the range of 3 to 10. For the Circle **Type** option, the tool supports integers of 2 or 3. After setting this parameter, set the point inputs in the Inputs section to the appropriate Point geometric features.
Padding Distance	The offset from the mask. When the value is greater than 0, the offset is outward. When the value is less than 0, the offset is inward. You can use this for example to exclude a bin that contains parts from the output Surface data. For example, in the following, before padding is set to a non-zero value, the bin is included in the output.

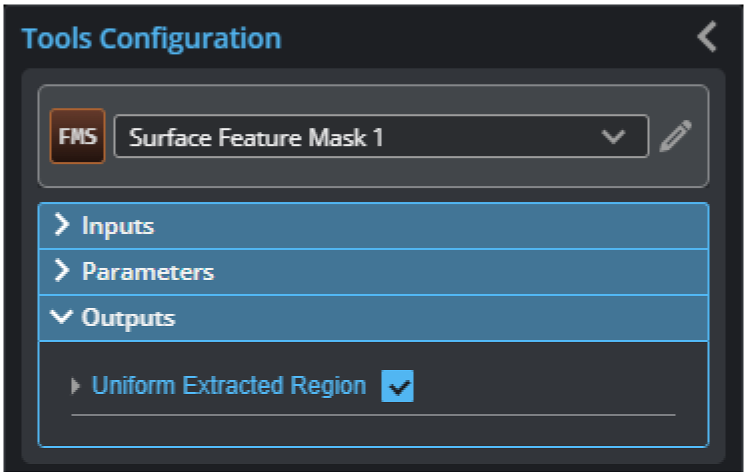
Parameter	Description
	 In the following, padding has been set to -3, which moves the mask "inward," excluding the bin.

Parameter	Description
	 The resulting Uniform Extracted Region outpu contains only the content of the bin.

Parameter	Description
	
Show Detail	Toggles whether to display the graphics of the mask region.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

All outputs provide an external ID (available by expanding the output in the Outputs panel) for optional use in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Data

Type	Description
Uniform Extracted Region	The cropped Surface data.

Surface Filter

The Filter tool provides several common vision processing filters that you can apply to surface data, as well as two "cropping" filters that output a subset of the surface data, letting you pre-process scan data to get more repeatable measurements. Any Surface or Feature tool can use the resulting filtered surface data as input.

For a list of the filters, see *Filters* on page 555.

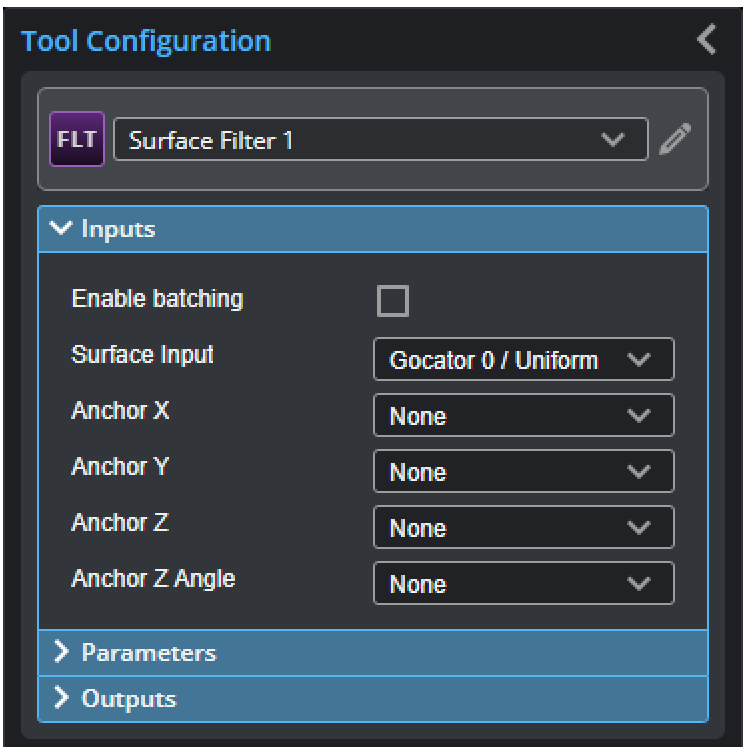
The Filter tool provides no measurements or decisions, as its only purpose is to output processed surface data.

> A limited set of filters is also available on the **Scan** page. These filters let you process scan data without needing to add tools. This can be useful if you are using a sensor mostly as an acquisition device.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



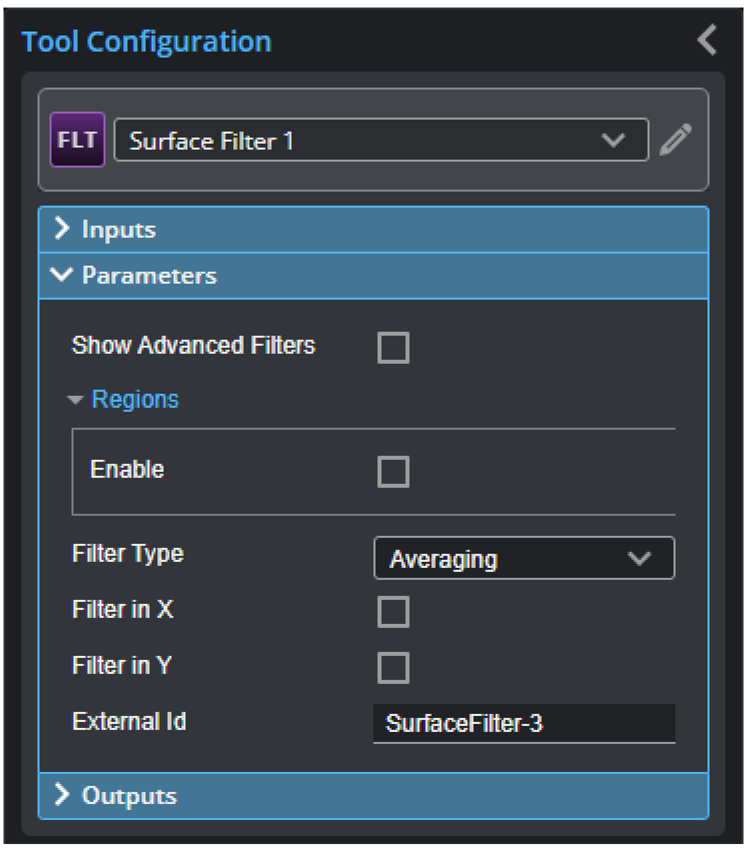
> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Show Advanced Filters	When enabled, displays advanced filters in **Filter Type** (see below), in addition to

Parameter	Description
	the standard filters (Gap Filling, Median, Averaging, and Decimation).
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Use Intensity	If enabled, the tool uses intensity data *instead of* Surface data. Only usable if **Acquire Intensity** is enabled on the **Acquire** > **Scan** page during scanning. Only available in the advanced filters (see above); for information on the available filters, see *Filters* on the next page. If disabled, the tool filters using Surface data; intensity data corresponding to filtered heightmap data will also be removed. For more information on scan modes, see *Scan Modes and Intensity* on page 195.
Units	Specifies whether filters use data points (**pts**) or millimeters (**mm**). Not available with all filters.
Filter Type	The type of the filter. For information on the available filters, see *Filters* on the next page.
Preserve Data Outside Region	Enable this setting when **Mask Mode** is set to **Include Data in Region** to include data outside the region in the Filtered Surface data output. Otherwise, the Filtered Surface data output only includes the data in the region. This setting is only displayed if regions are enabled.
Threshold	The threshold that the filter uses. (Not available on all filters.)
Symmetry	One of the following: Symmetrical, Horizontal, or Vertical. (Not available on all filters.)
Kernel Size	The kernel size that the filter uses. (Not available on all filters.)
Filter in X Filter in Y	These parameters enable filtering along the X and the Y axis, respectively. Set the corresponding window sizes in **X Windows Size** and **Y Window Size**. The filter window sizes are specified in millimeters (and additionally, in the tools, in data points). To calculate the number of data points that a window covers when the units are millimeters, use the following calculation: • User-specified window size divided by the X spacing interval (that is, the number of millimeters per point) on the **Spacing** tab in the **Sensor** panel. (For more information on spacing intervals, see *Uniform Spacing* on page 197.) • *With the exception of the gap filling filter*, round the result of the division to the nearest integer value. With the gap filling filter, filling is performed within the provided window size.

Parameter	Description
	For example, if you set the size of the filter's window to a value between 1.5 mm and 2.49 mm (inclusively), and the X spacing interval is set to 1 mm, the filter covers 2 data points. A filter window size from 2.5 mm to 3.49 mm results in a filter covering 3 data points.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

The following filters are available in the Filter tool. Filters described below as advanced are only displayed if **Show Advanced Filters** is enabled.

Filters

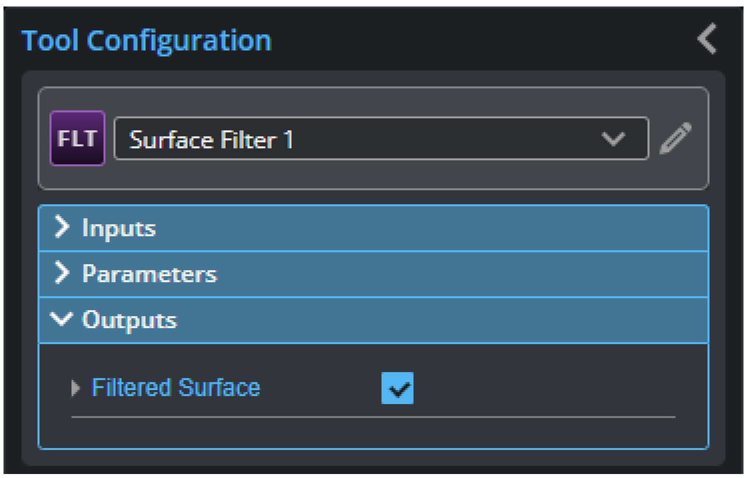
Name	Description
Gap Filling	Fills in missing data using information from the nearest neighbor data points, for example, when data is missing due to occlusions. Gap filling also fills gaps where no data is detected, which can be due to low surface reflectivity, for example dark or specular surface areas, or to actual gaps in the surface. The values in **Filter in X** and **Filter in Y** represent the maximum gaps the sensor will fill. Wider gaps are not filled. Gap filling works by filling in missing data points using either the lowest values from the nearest neighbors or linear interpolation between neighboring values (depending on the Z difference between neighboring values), in the specified window. The sensor can fill gaps along both the X axis and the Y axis. In Profile mode, gap filling is limited to the X axis.
Median	Substitutes the value of a data point with the median calculated within the window or windows set in **X Window Size** or **Y Window Size** around the data point. If the number of valid (non-null) data points in the window is even, the median value is simply the value in the center of the sorted list of values. If the number of valid points is odd, the average of the two values in the center is used instead. Missing data points will not be filled with the median value calculated from data points in the neighborhood. With an odd window size, the output is at the center of the window. With an even window size, the output is 0.5 pixels to the right of the center (that is, using window / 2-1 values from the left, and window / 2 from the right.
Averaging	Substitutes a data point value with the mean value of that data point and its nearest neighbors within the window or windows set in **X Window Size** or **Y Window Size**. X smoothing works by calculating a moving average across samples within the same profile. Y smoothing works by calculating a moving average in the direction of travel at each X location. If both X and Y smoothing are enabled, the data is smoothed along X axis first, then along the Y axis. Missing data points will not be filled with the mean value calculated from data points in the neighborhood.

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Decimation</td><td>Decimation reduces the number of data points along the X or Y axis by choosing data points at the end of a specified window around the data point. For example, by setting X Window Size to 0.2, only points every 0.2 millimeters will be used. The filter generates points starting from the leftmost edge of the scan data, stepping in equal steps away from that side.</td></tr>
<tr><td>Gaussian (advanced)</td><td>A Gaussian filter. Set the Sigma, Units, and Preserve Data Outside Region parameters as required.</td></tr>
<tr><td>Opening (advanced)</td><td>Erosion followed by dilation.</td></tr>
<tr><td>Closing (advanced)</td><td>Dilation followed by erosion.</td></tr>
<tr><td>Erosion (advanced)</td><td>Applies an erosion filter. Lets you specify the direction of the erosion; one of the following:
Horizontal
Vertical
Symmetrical</td></tr>
<tr><td>Dilation (advanced)</td><td>Applies a dilation filter. Lets you specify the direction of the dilation; one of the following:
Horizontal
Vertical
Symmetrical</td></tr>
<tr><td>Morph Gradient (advanced)</td><td>Applies a morphological gradient. The difference between dilation and erosion.</td></tr>
<tr><td>Sobel Magnitude (advanced)</td><td>Applies a Sobel magnitude filter.
Lets you specify the direction of the filter; one of the following:
Horizontal
Vertical
Symmetrical</td></tr>
<tr><td>Laplacian (advanced)</td><td>Applies a Laplacian filter. Useful for detecting areas of distinct edges. Uses the following kernel:<table><tr><td>0</td><td>-1</td><td>0</td></tr><tr><td>-1</td><td>4</td><td>-1</td></tr><tr><td>0</td><td>-1</td><td>0</td></tr></table></td></tr>
<tr><td>Negative (advanced)</td><td>Inverts the height or intensity values in the scan data. When Use Nulls is enabled, null values in the input are replaced by the highest height value.</td></tr>
<tr><td>Equalize (advanced)</td><td>Normalizes the norm or value range of an array.</td></tr>
<tr><td>Binarize (advanced)</td><td>Sets height values to a fixed value for each point that is present in the data. Can be used with a region Z offset to threshold points above/below a Z value.
When Use Intensity is enabled, the tool sets data points with an intensity value</td></tr>
</table>

Name	Description
	over
Percentile (advanced)	Limits the scan data to points between the values you set in **High Percentile** and **Low Percentile**, which are displayed when you choose this option.
Relative Threshold (advanced)	Crops scan data based on user-specified minimum and a maximum heights. Use **Reference Region** to set the heights relative to a reference region.
Crop only (advanced)	Crops the scan data to the user-defined region.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Data

Type	Description
Filtered Surface	The filtered data, available for use as input in the **Stream** drop-down in other tools.

Surface Flatness

The Flatness tool returns various measurements related to the flatness of one or more regions on the surface of your target. The tool is ideal for general fit and finish inspection.

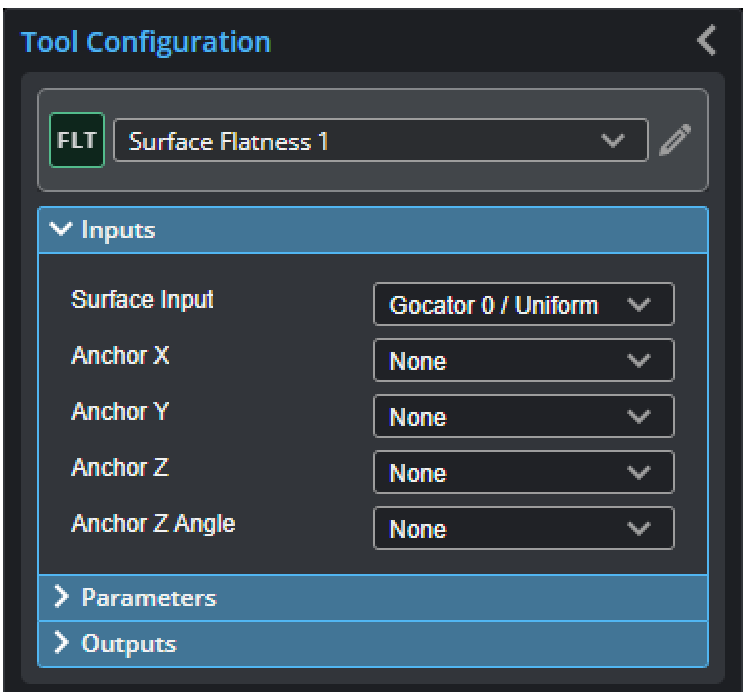
The tool lets you set a grid with user-defined cell sizes over a region, or more flexibly with multiple individual regions manually. In each case, "local" minimum and maximum heights, as well as flatness indicators (maximum - minimum), are returned. In addition, "global" minimum, maximum, and flatness measurements, which combine data from all flatness measurement areas, can also be returned. The tool measures the maximum and minimum distances from a best-fit plane for each cell or flexible region, and from another plane fit to all data for the "global" measurements.

You can control how many data points the tool uses in its calculations to account for noise or smooth data, or otherwise exclude unwanted data.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

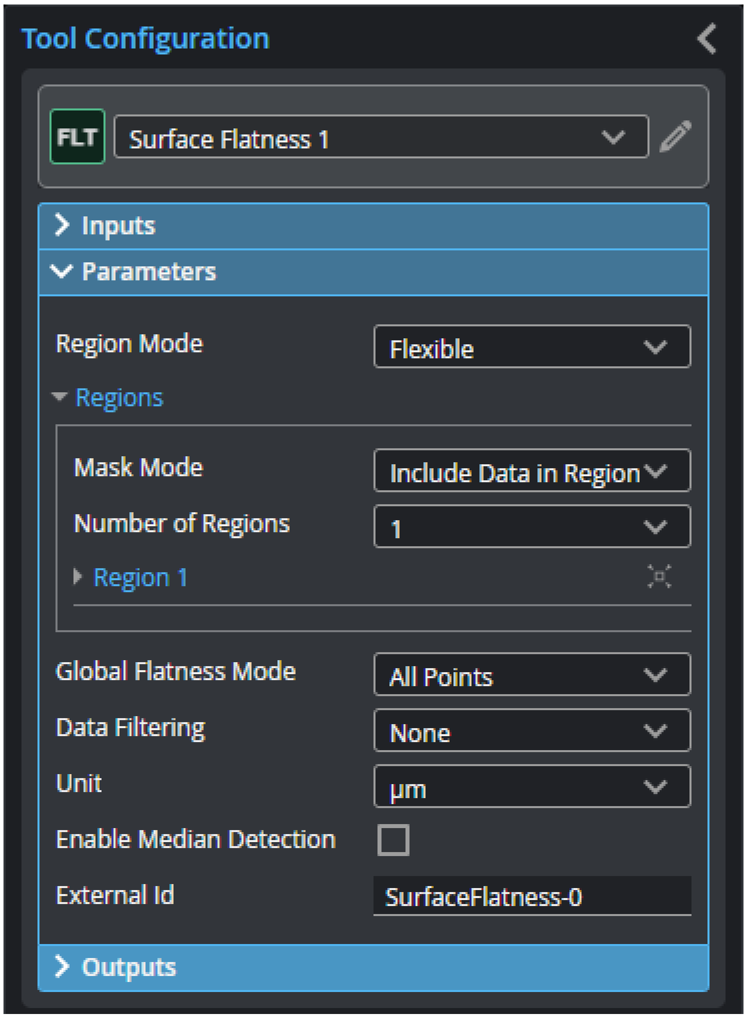
Inputs

Name	Description
Surface Input	The data the tool applies measurements to or processes.

Name	Description
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

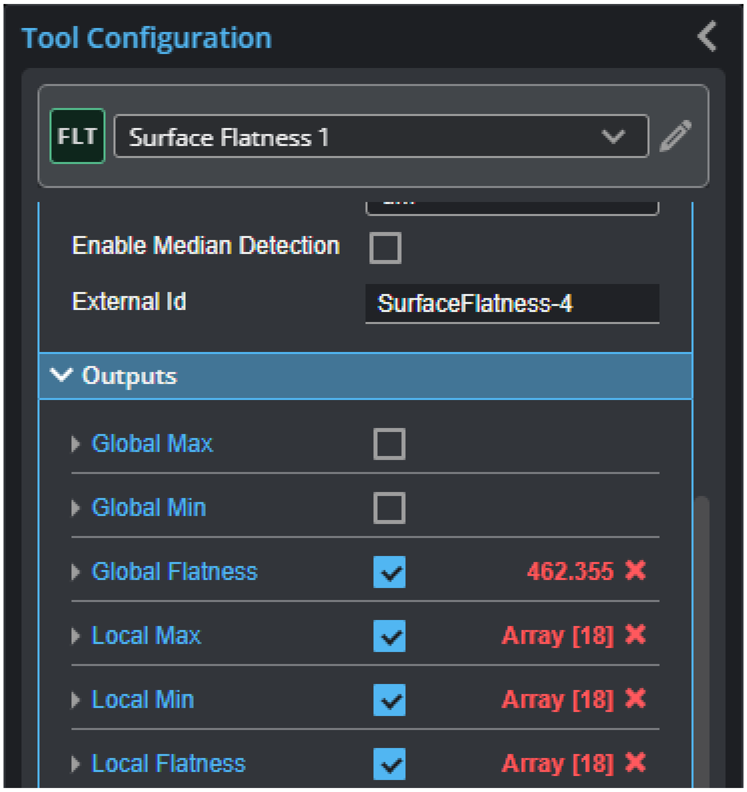
Parameter	Description
Region Mode	Determines how flatness measurement areas are set up on the target. One of the following: **Grid Pattern**: The tool determines flatness in a grid you define on the target. This

Parameter	Description
	option enables settings that let you set the size and location of a region that contains the grid (**Grid Region** setting), as well as the width and length of the grid cells (**Grid Width** and **Grid Length**). The combination of the values of these settings determines the number of cells in the grid region. **Flexible**: The tool determines flatness using one or more regions that you define individually on the target.
Grid Region *(used with Grid Pattern region mode)*	Determines the size of the grid region. (See details under **Grid Pattern** in **Region Mode** above.)
Grid Width (X) Grid Length (Y) *(used with Grid Pattern region mode)*	These settings determine the size of the cells in the grid. (See details under **Grid Pattern** in **Region Mode** above.)
Regions	When expanded, displays the region- and mask-related settings.
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Global Flatness Mode	Chooses which points the tool uses to calculate global flatness. One of the following: **All Points**: The tool uses all points in the measurement area (all flexible regions or the grid pattern in the region). **Single Average Point**: The tool uses an average of the points in the measurement area. When you choose this option, the global measurements require at least four data points to calculate the plane and statistics. This means that if you set **Region Mode** to **Flexible**, you must choose a minimum of four regions; if you set Region Mode to Grid Pattern, the size of the grid and the cells must result in at least four cells.
Data Filtering	Lets you filter scan data before the tool performs its calculations. When you use Local Percentile or Global Percentile, you must set **High Percentile** and **Low Percentile**. **None** - The tool performs no filtering. **Local Percentile** - Applies the percentile filter to local masks first, and then merges to a global mask. **Global Percentile** - Merges to a global mask first, and then applies the percentile filter to the global mask.
Unit	Lets you choose which units the tool uses for measurement results. One of the following:

Parameter	Description
	• um (micrometers) • mm (millimeters)
Enable Median Detection	When enabled, Point geometric features for the global median and the local median are output.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

> The unit of measurement values is set with the **Unit** drop-down.

Measurements

Measurement
Global Max **Global Min** **Global Flatness**

Measurement
The maximum distance, minimum distance, and flatness (maximum - minimum) calculated using the valid data points from *all* the cells in the grid (when **Region Mode** is set to **Grid Pattern**), or *all* the individual regions (when **Region Mode** is set to **Flexible**).
Local Max **Local Min** **Local Flatness** Arrays containing the maximum distance, minimum distance, and flatness (maximum - minimum) calculated using the valid data points from the grid cells (when **Region Mode** is set to **Grid Pattern**), or the individual regions (when **Region Mode** is set to **Flexible**). For more information on arrays, see *Arrays, Batching, and Aggregation* on page 242.

Features

Type	Description
Global Plane	The plane fitted to the valid data points from *all* the cells in the grid (when **Region Mode** is set to **Grid Pattern**), or *all* the individual regions (when **Region Mode** is set to **Flexible**).
Local Plane	An array of the planes fitted to the valid data points from the grid cells (when **Region Mode** is set to **Grid Pattern**), or those from the individual regions (when **Region Mode** is set to **Flexible**).
Global Average Point Global Max Point Global Min Point Global Median Point	Point geometric features representing the global average, the global maximum / minimum, and, if **Enable Median Detection** is enabled, the global median.
Local Average Point Local Max Point Local Min Point Local Median Point	Arrays of point geometric features representing the local average, local maximum / minimum, and, if **Enable Median Detection** is enabled, local median.

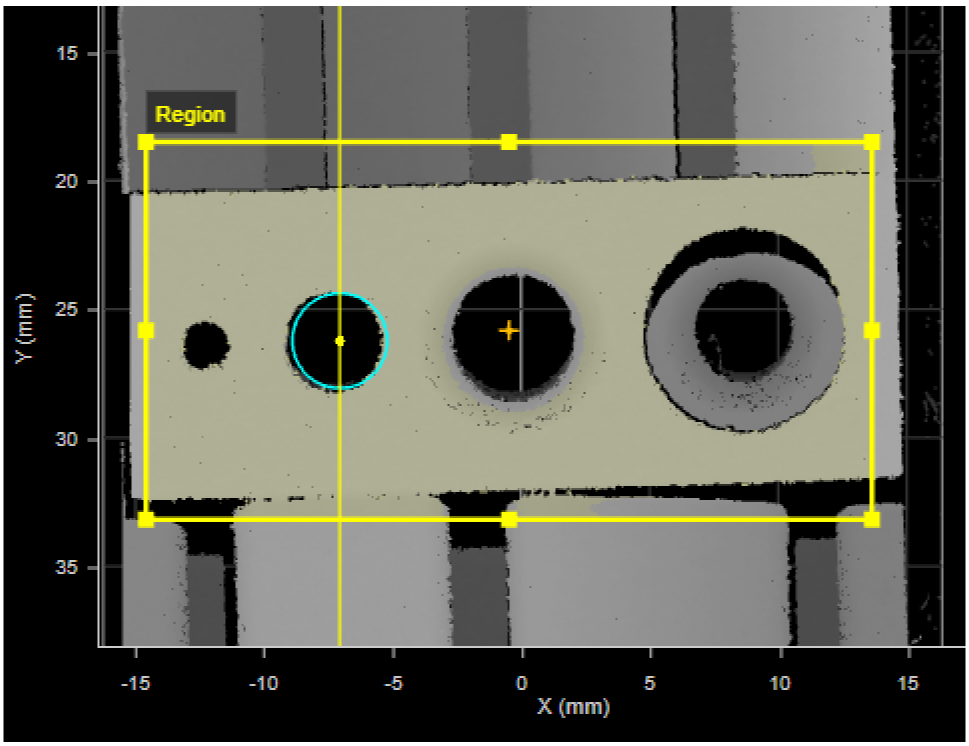
Surface Hole

The Hole tool measures a circular opening within a region of interest on the surface and returns its position and radius.

	The tool does not search for or detect the feature. The tool expects that the feature, conforming reasonably well to the defined parameters, is present and that it is on a sufficiently uniform background.

The hole can be on a surface at an angle to the sensor.

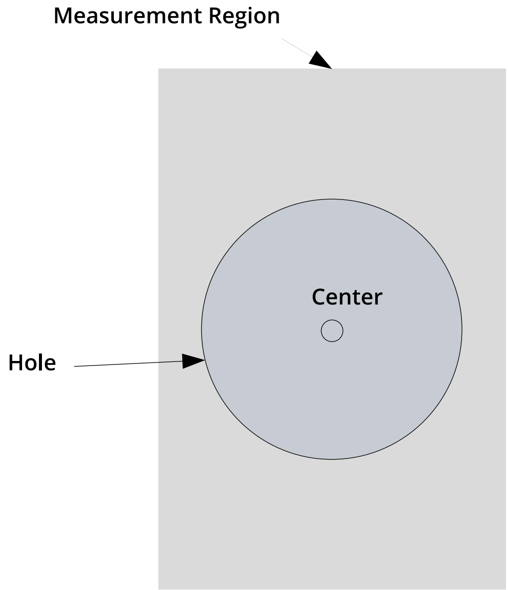
The tool uses a complex feature-locating algorithm to find a hole and then return measurements. For a detailed explanation of the algorithm, see *Hole Algorithm* on the next page. The behavior of the algorithm can be adjusted by changing the tool's parameters.

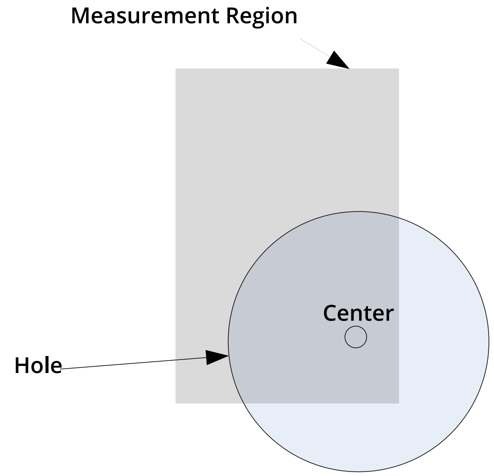


For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Measurement Region

The center of the hole must be inside the measurement region, even if the Partial Detection option is enabled. Either of the following is acceptable.





Hole Algorithm

The Hole tool processes the data in three phases: Search, Measure, and Filter. The algorithm can separate out background information that appears inside the hole. It can also detect holes that only partially appear in the data.

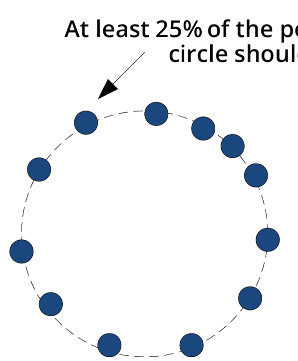
See the tool's parameters for an explanation of the options that affect the tool's algorithm.

Search phase - The tool searches for coarse data transitions (edge data) and performs a coarse fitting of the hole model (specified by the orientation angles and the nominal value) to determine the most likely candidate. If **Tilt Correction** is set to **Autoset**, the algorithm uses the data within the measurement region to estimate the orientation of the part.

Measure phase - A more rigorous edge detection algorithm is applied to precisely determine the edges around the feature. Edge detection at this stage will reject outliers and noise. The algorithm requires at least 25% of the data around the hole for the candidate to remain valid.

The accuracy of the algorithm improves when the points are spread more evenly along the hole's circumference.

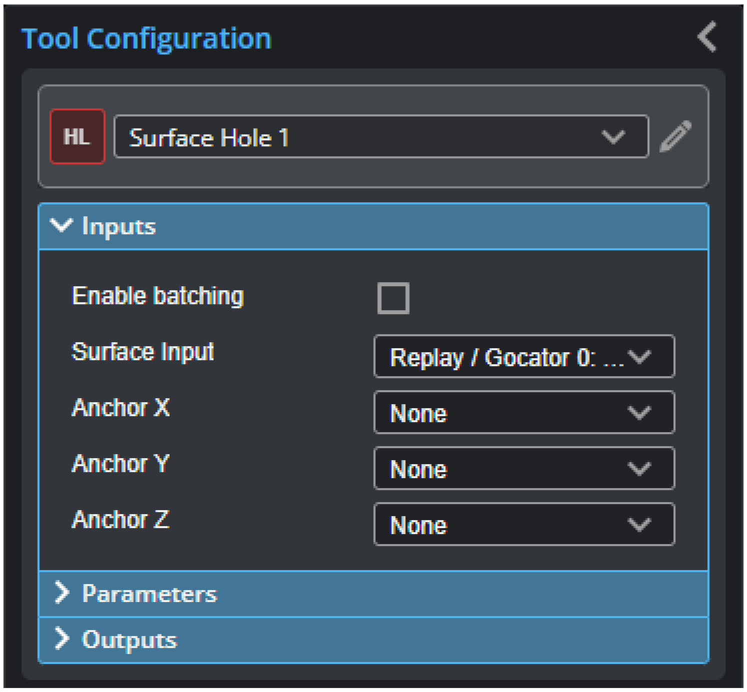
The set of refined edges is then used to locate and inspect the feature. If the Reference Regions option is enabled and set to AutoSet, the edges are also used to calculate the location of the reference regions.



Filter phase - The detected location and dimensions are then compared to the nominal and tolerance settings. If the refined feature falls within the measurement region and its measurements fit within the specified tolerance, the results are reported. If not, an invalid result is returned.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



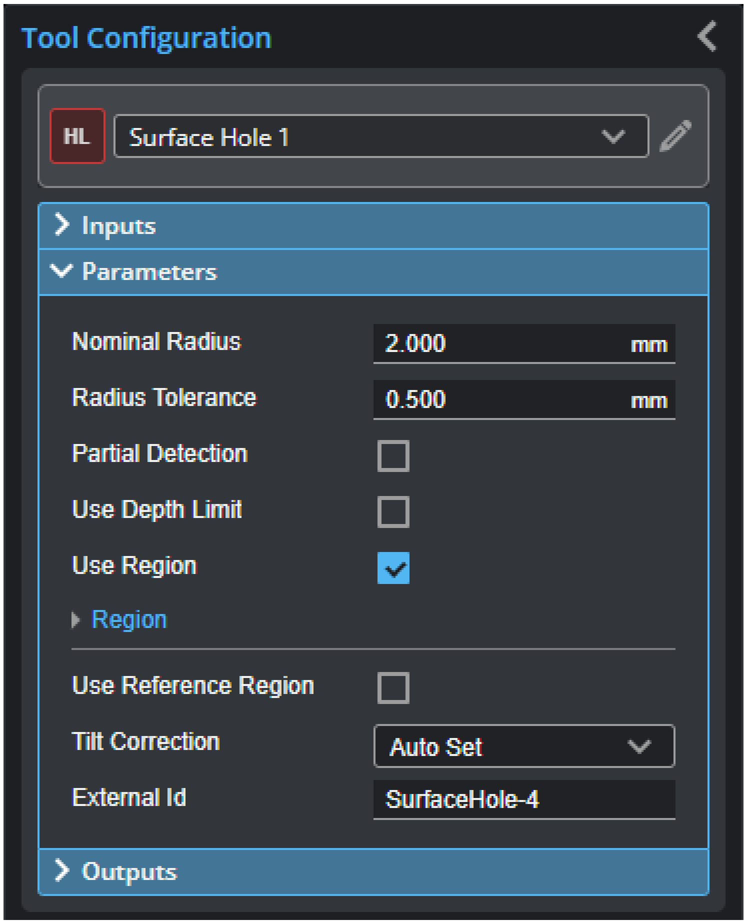
> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

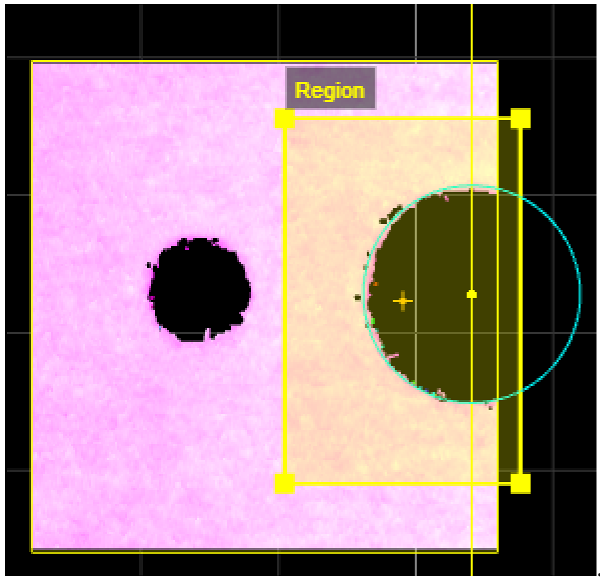
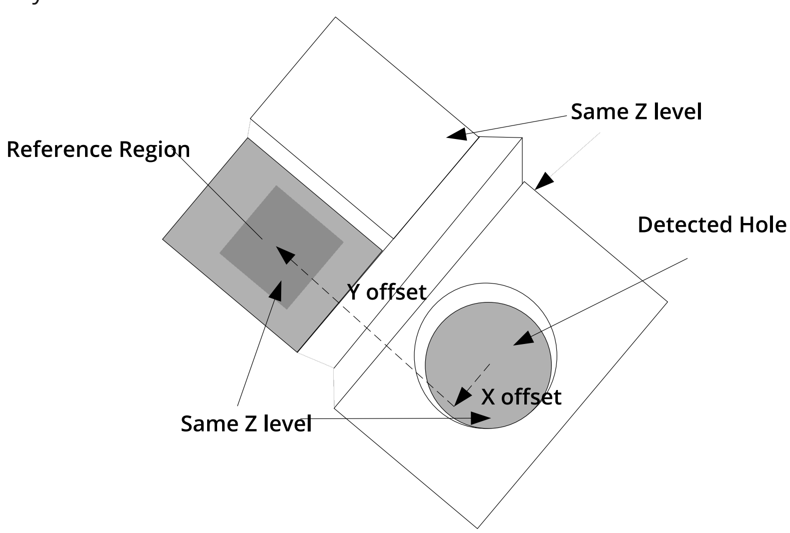
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

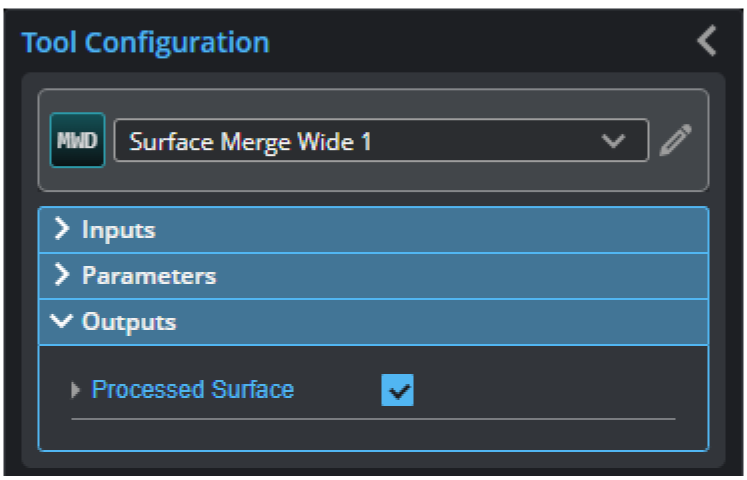
Parameter	Description
Nominal Radius	Expected radius of the hole.
Radius Tolerance	The maximum variation from the nominal radius (+/- from the nominal radius).
Partial Detection	Enable if only part of the hole is within the measurement region; the centre of the hole must be in the measurement region to be located. If disabled, the hole must be completely in the region of interest for results to be valid.

Parameter	Description
	
Use Depth Limit	When **Use Depth Limit** is enabled, data below the value set in **Depth Limit** (relative to the surface) is excluded from the hole calculations.
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Use Reference Region	When enabled, displays additional settings to let you set a reference region (see below).
Reference Region	The tool uses the reference regions to calculate the Z position of the hole. It is typically used in cases where the surface around the hole is not flat.  When this option is set to **Autoset**, the algorithm automatically determines the

Parameter	Description
	reference region. When the option is set to **1 Region** or **2 Regions**, you must manually define the reference regions. The location of the reference region is relative to the detected center of the hole and positioned on the nominal surface plane. When **Reference Region** is disabled, the tool measures the hole's Z position using all the data in the measurement region, except for a bounding rectangular region around the hole.
Tilt Correction	Tilt of the target with respect to the alignment plane. **Autoset**: The tool automatically detects the tilt. **Custom**: You must enter the X and Y angles manually in the X Angle and Y Angle parameters (see below).
X Angle Y Angle	The X and Y angles you must specify when **Tilt Correction** is set to **Custom**. You can use the Surface Plane tool's X Angle and Y Angle measurements to get the angle of the surrounding surface, and then copy those measurement's values to the **X Angle** and **Y Angle** parameters of this tool. For more information, see Plane.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

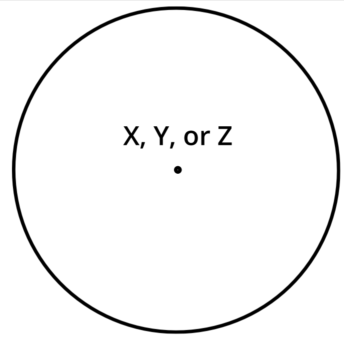
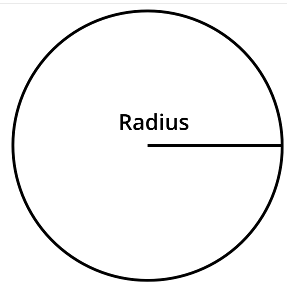
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Determines the X position of the hole center.	
Y Determines the Y position of the hole center.	
Z Determines the Z position of the hole center.	
Radius Determines the radius of the hole.	

Features

Type	Description
Center Point	The center point of the hole. The Z position of the center point is at the Z position of the surface surrounding surface.

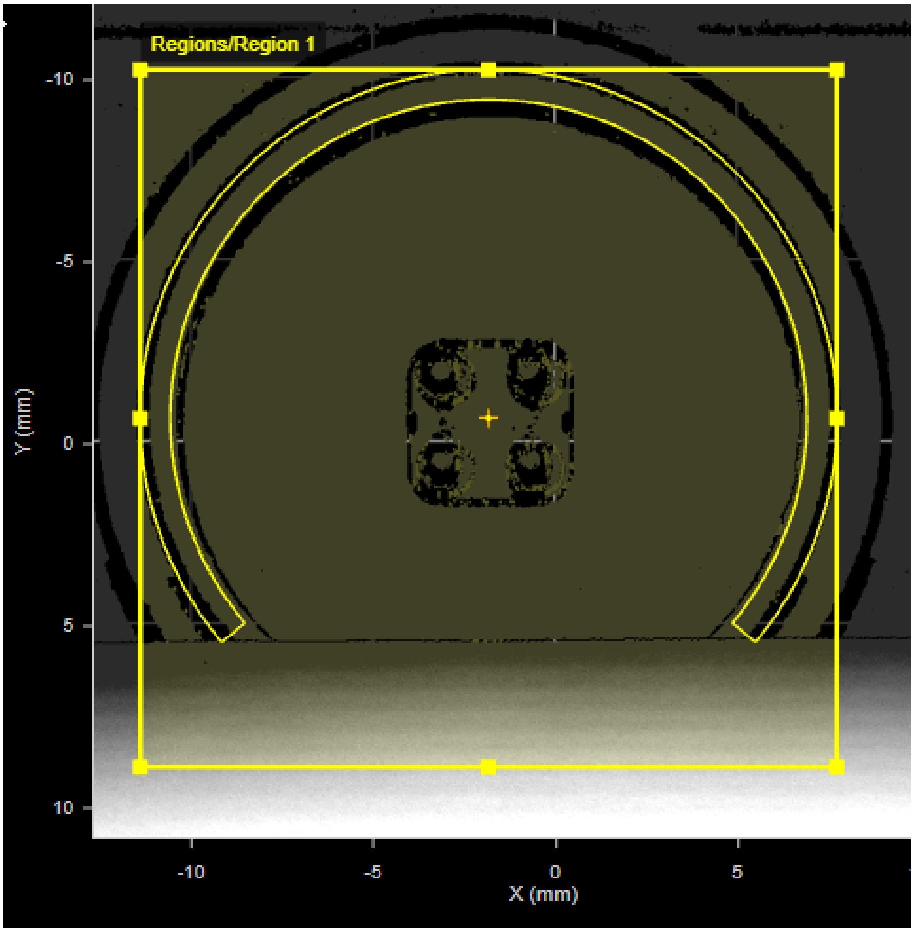
For more information on geometric features, see *Geometric Features* on page 262.

Surface Mask

The Surface Mask tool lets define up to 16 regions to extract data from a surface. Each region's size, position, and shape (circular, elliptical, and rectangular) can be individually configured, and regions can overlap. The tool can also exclude inner data of circular and elliptical regions, letting you extract rings of surface data. Extracted data is output in a single surface.

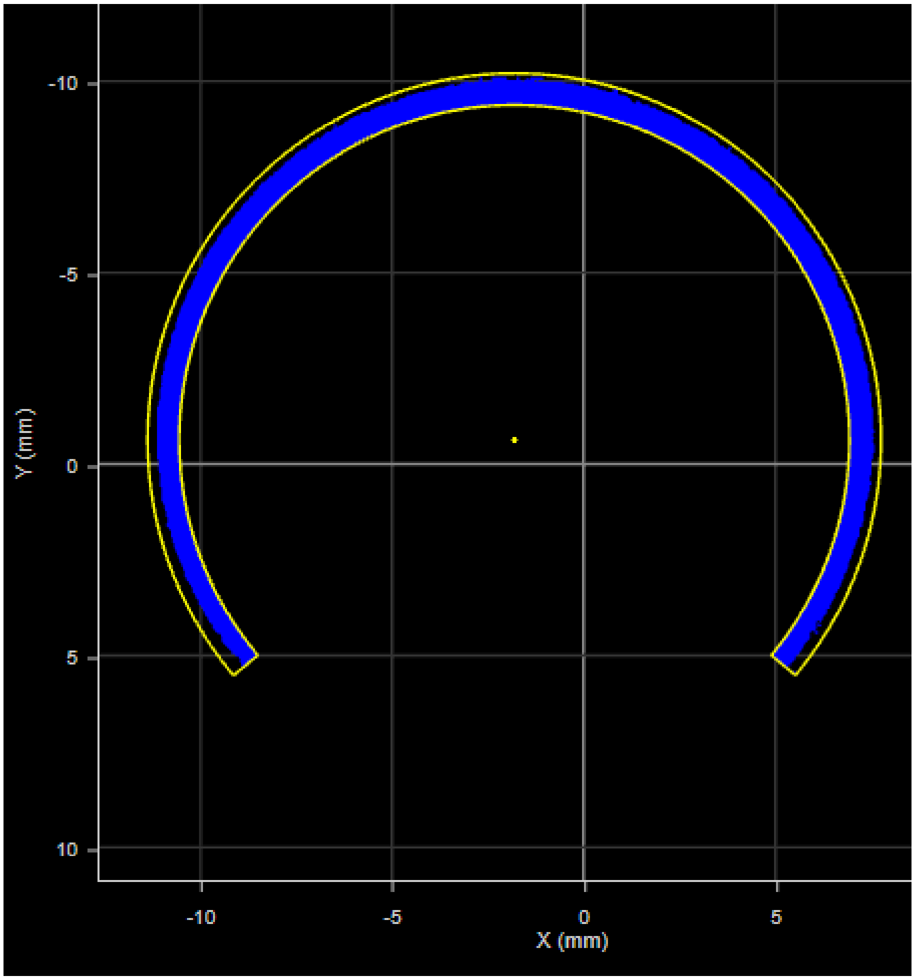
The resulting surface can then be further processed or measured by other tools.

For example, given the following scan data:



A circle region box containing a partial ring (cyan)

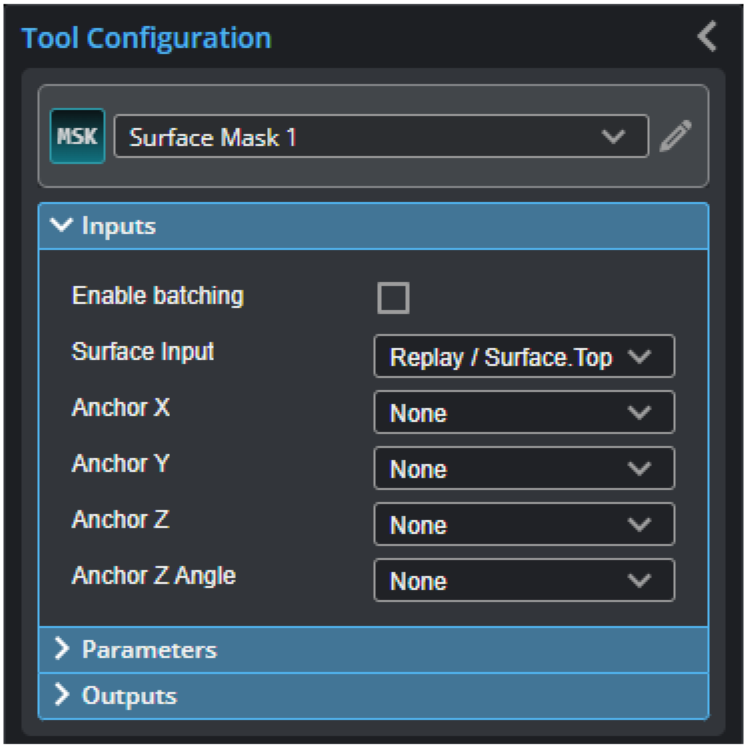
The image below shows the extracted data. The extracted surface data can then be further processed by other tools, or measurements can be applied to the surface data.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



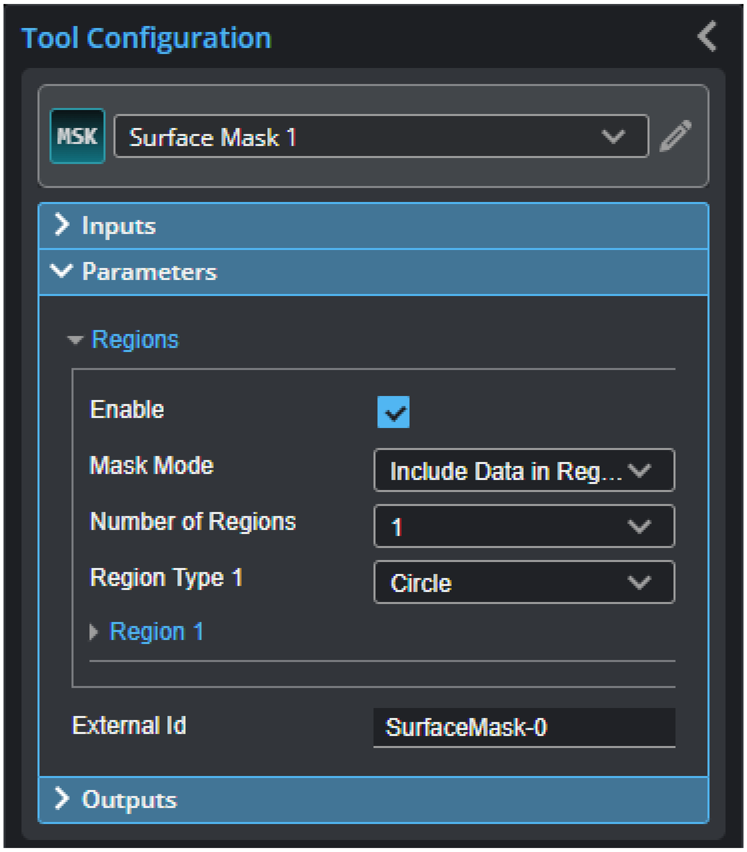
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

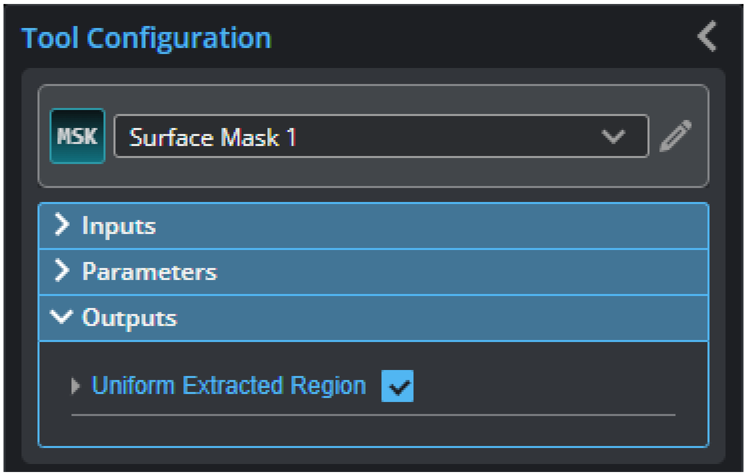


Parameters

Parameter	Description
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



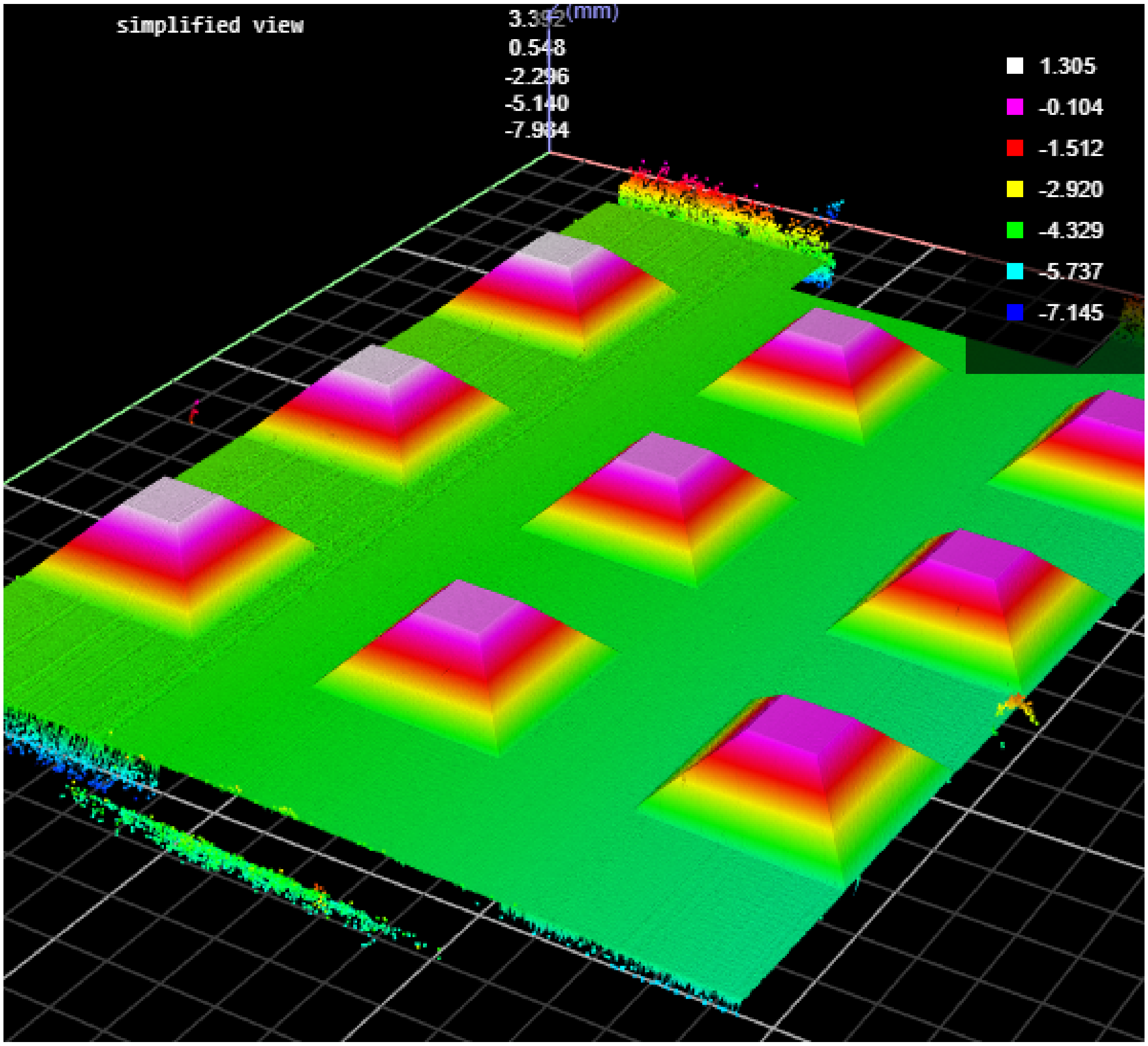
Data

Type	Description
Uniform Extracted Region Point Cloud Extracted Region	The surface containing the extracted region or regions. (Uniform and point cloud data, respectively.)

Surface Merge Wide

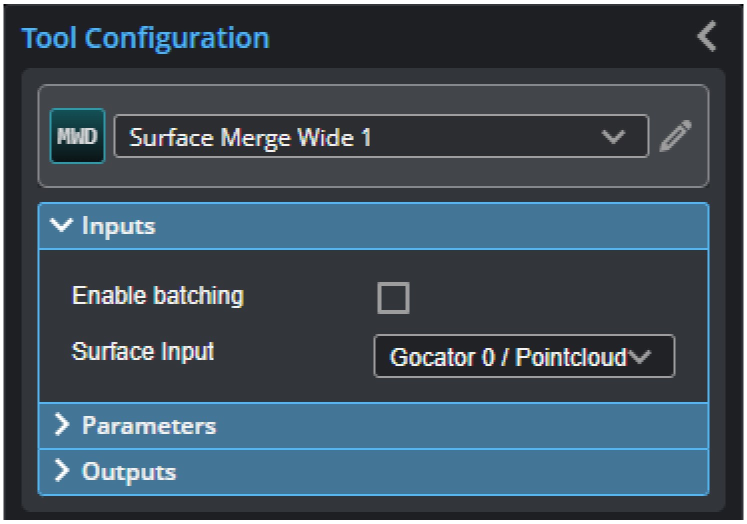
	This tool is *only* intended for use with G2 sensors.

The Surface Merge Wide tool uses the XML transformation file you choose that was produced by the Surface Align Wide tool (see *Wide Layouts (Surface Align Wide Tool)* on page 160) and combines scan data from multiple sensors into a single surface. You can use any built-in or GDK-based Surface measurement tool to perform measurements on the resulting merged Surface scan data.



Inputs

You configure the tool's inputs in the expandable **Inputs** section.

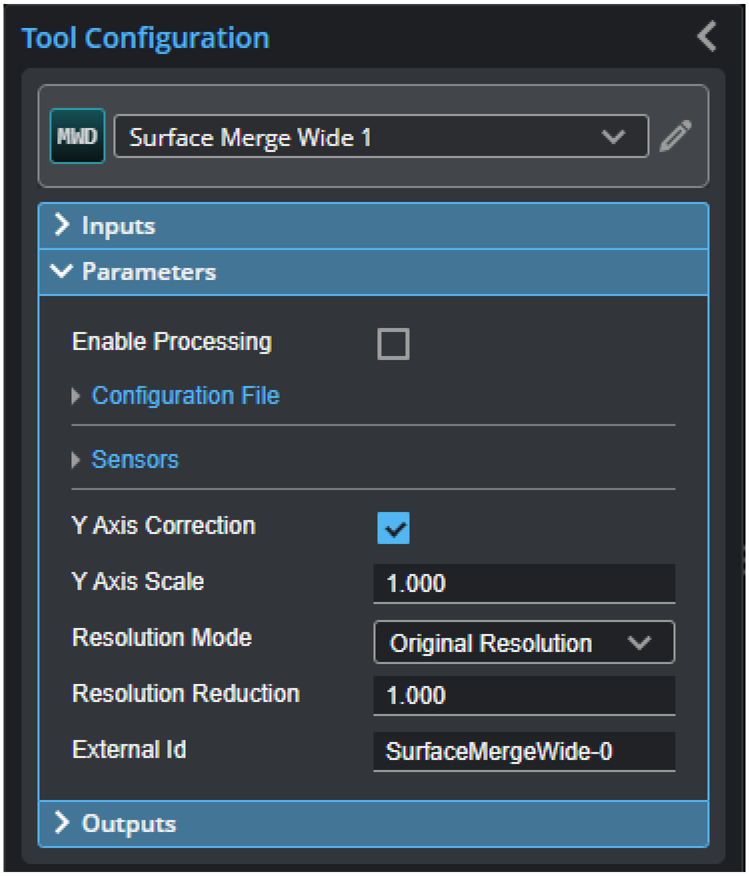


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Enable Processing	Causes the tool to start merging data from the individual sensors. Make sure to properly configure the tool *before* enabling this option.
Configuration File	Expandable area containing settings related to the XML configuration file.
Operation	Actions that apply to the XML configuration file the tool uses to perform merging. You must first create a configuration file using the Surface Align Wide tool (see *Wide Layouts (Surface Align Wide Tool)* on page 160)One of the following: • **Normal**: The tool automatically switches to this operation after performing another operation. • **Load**: Displays a list of configuration files you can load. After you select a file, the tool loads it and displays a message in the log. • **Save**: Saves manual changes in the **Sensor Parameters** settings, using the provided configuration file name. You do not typically need to make manual changes in this tool. Use this only for troubleshooting. • **Delete**: Deletes the configuration file you select. • **Refresh**: Refreshes the list of configuration files.
Alignment Status	Indicates whether the tool has aligned the sensors. Either "Not Aligned" or the date

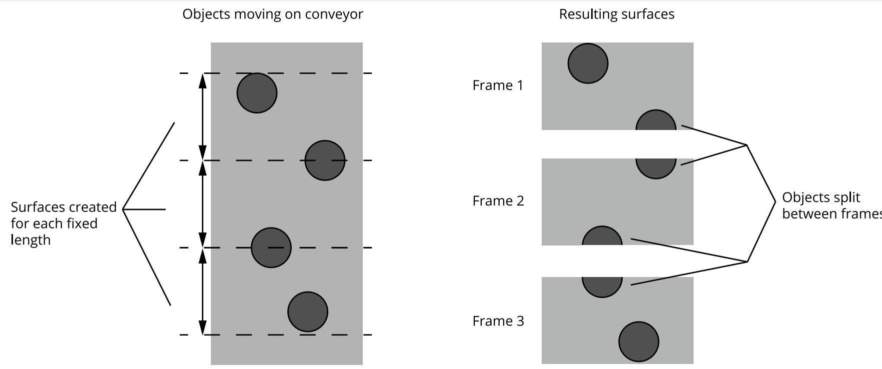
Parameter	Description
	of the alignment.
Sensors	Expandable section containing sensor-related settings
Sensor Count	Indicates the number of sensors in the system.
Sensor Parameters	A drop-down that lets you display the settings of a specific sensor. You do not usually need to change these settings, as they are set when you load the configuration file produced by the Surface Align Wide tool using the **Operation** drop-down. The values are intended for diagnostics only. For information on the parameters, see *Sensor Parameters* on page 173.
Y Axis Correction	Compensates for errors in encoder settings, which can distort scan data. If you cannot change the encoder settings or if you are working with data recorded with an incorrectly configured encoder, enable this setting. The tool will adjust the Y scaling to compensate and display the value in **Y Axis Scale**. When enabled, displays a **Y Scaling** setting. For more information, see below.
Y Axis Scale	Typically used to compensate for errors in encoder settings, which can distort scan data. If you cannot change the encoder settings or if you are working with data recorded with an incorrectly configured encoder, use this setting to adjust the Y scaling to compensate. The value used in this tool must match the value used in Surface Align Wide; for more information, see *Wide Layouts (Surface Align Wide Tool)* on page 160. Only displayed when **Y Axis Correction** is enabled (see above).
Resolution Mode	Determines whether the tool scales the X or Y resolution so that they are the same (a 1:1 ratio), or leaves the X and Y resolutions as the original. One of the following. • **Optimal (uniform)** Brings the X/Y resolution ratio to 1:1 while preserving the pixel area. Best for random rotation around Z. Provides a balance between the highest and lowest possible resolutions, requiring an average amount of memory and processing time compared to the **High Oriented (uniform)** or **Low Oriented (uniform)** options. • **High Oriented (uniform)** Interpolates the lower resolution to match the higher resolution (between X and Y) in the input. Choose this option when increased resolution is preferred over speed and low memory usage. (This can result in a very high resolution output, creating a lot of data for subsequent tools to process. This can in turn result in slower processing.) • **Low Oriented (uniform)** Decimates the higher resolution to match the lower resolution (between X and Y) in the input. Choose this option when speed and low memory usage is preferred over resolution. (It can result in significant data quality reduction with large Z rotations if the X and Y resolutions of the input are very different.) • **Original Resolution**

Parameter	Description
	Keeps the original X and Y resolution of the scan. Use this option only when you expect little or no Z rotation. Otherwise, with X/Y resolution ratios that are not 1:1, large rotation around Z results in severe data quality reduction. • **Customized** Lets you set a custom resolution mode using the **Scale X**, **Scale Y**, **Output Width**, and **Y Offset Buffering** parameters this option displays. For more information on Y Offset Buffering, see *Y Offset Buffering Parameters* below.
Resolution Reduction	Reduces the lateral resolution of the heightmap to reduce processing time.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

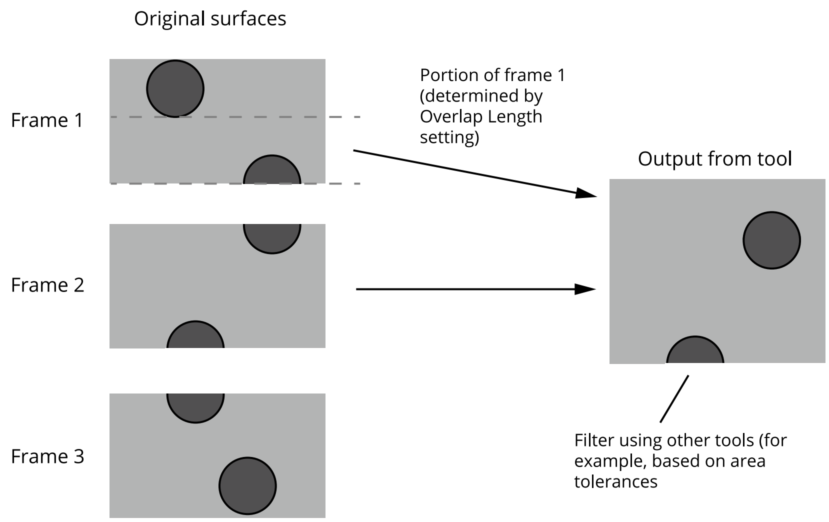
Y Offset Buffering Parameters

Parameter	Description
Y Offset Buffering	Displayed when **Resolution Mode** is set to Customized. The following options are available: **Disabled** The tool performs no offset buffering. **Fixed Length** The tool performs Y offset buffering. Use this mode if some sensors in the sensor system are offset along the Y axis and the system uses Fixed Length surface generation to ensure that the complete, combined Surface scan is output. (For more information on Fixed Length surface generation, see *Surface Generation* on page 199.) The tool outputs the new surface data in the Processed Surface data output. Choosing this option displays two additional, optional parameters: **Overlap Length** and **Reset Time Limit**. • **Overlap Length**: Useful for applications where parts might be split between two frames. Sets the amount of the previous frame's data, in millimeters, to append to the current frame's data. The combined Surface data will be output in the Processed Surface tool data. Choose the overlap length to accommodate the size of your scan targets. The following images show how parts could be split between frames:

Parameter	Description



The following shows how the tool combines data:

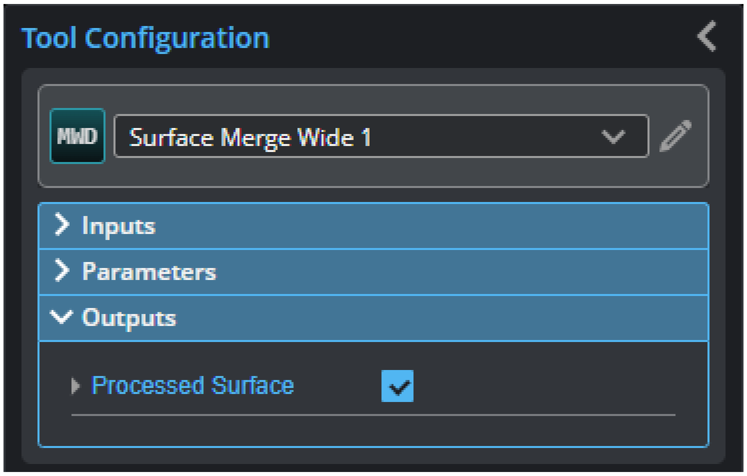


Data is only appended in one direction. Partial objects in the resulting surface output from the tool must be filtered out using downstream tools, for example, excluding them based on the expected area.

- **Reset Time Limit**: The maximum time the tool waits until resetting the buffer. When it is set, it will check the timestamp of neighboring frames to ensure that not too much time has elapsed. If the time difference is above the limit, the buffering is reset. When set to 0, the check is disabled and input surfaces are assumed to follow each other.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Data

Type	Description
Processed Surface	The Surface data resulting from combining the scan data of the individual sensors. You can use any Surface measurement tool to perform measurements on the combined data.

Surface Mesh

	This tool is *only* intended for use with G2 sensors.

	You *must* enable the **Enable Processing** parameter for this tool to produce output.

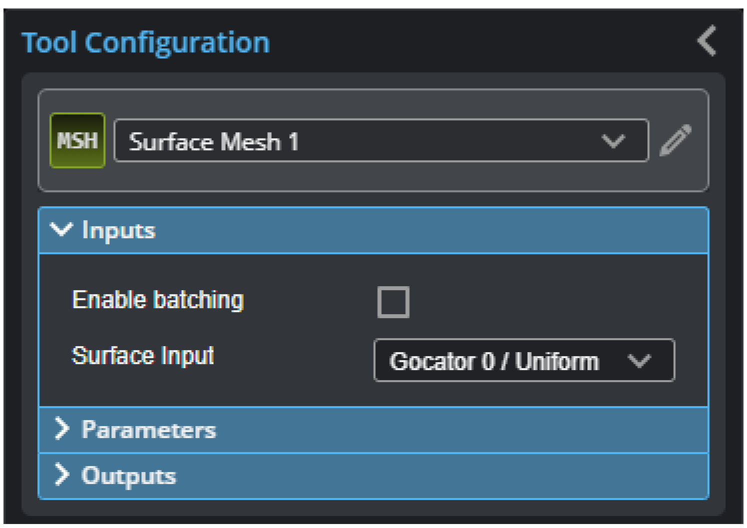
	Always set this tool's input to the Surface data being produced by the sensors.

The Surface Mesh tool takes in an XML transformation file produced by the Surface Align Ring tool (see *Ring Layouts (Surface Align Ring Tool)* on page 182), and stitches scan data from multiple sensors into a single *mesh*, which is typically a 360-degree scan. (That is, when sensors are in a ring configuration.) You can apply some measurements directly to the resulting Mesh scan data, or you can use the Mesh Projection tool to extract a surface from any angle of the Mesh data, and apply any of the other Surface measurement tools to the extracted surface.

Note that the tool's settings (most of which are visible only if you select a sensor from the **Sensor Parameters** drop-down) are populated by loading the XML transformation file produced by Surface Align Ring.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

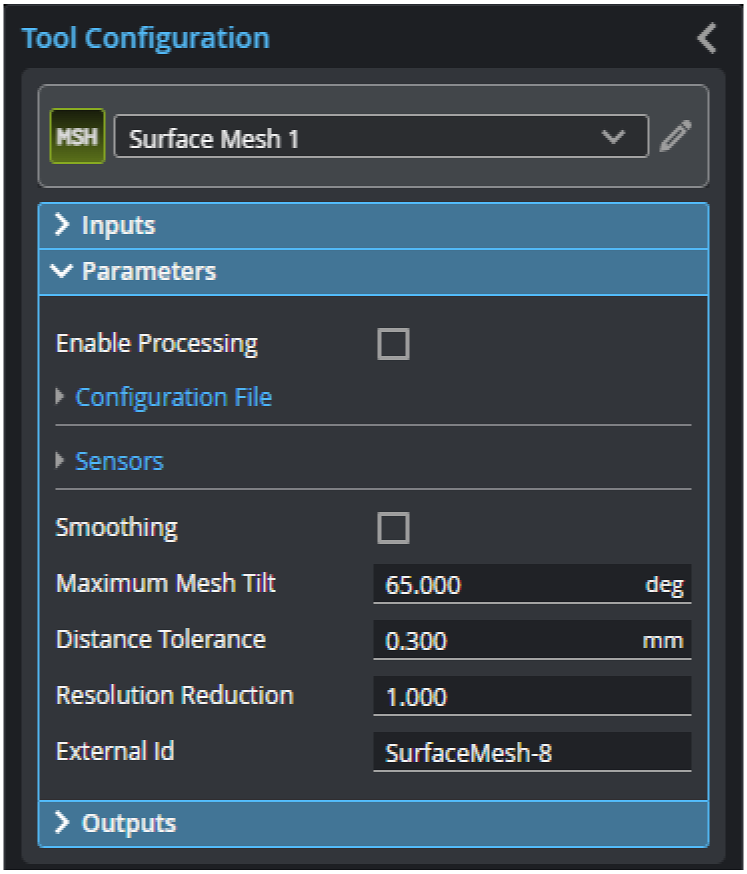


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.

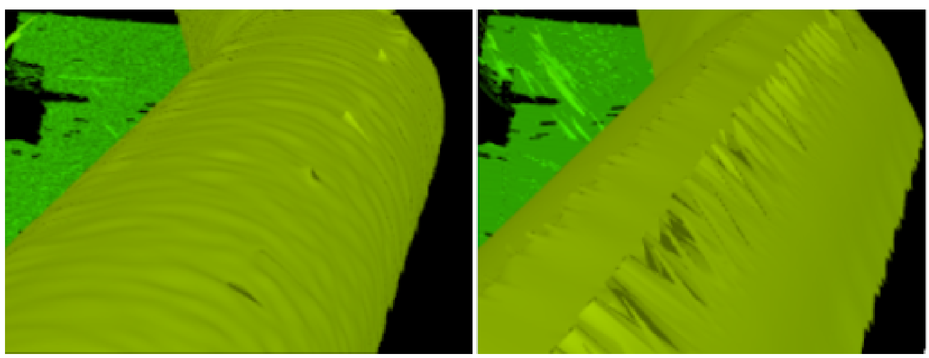
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



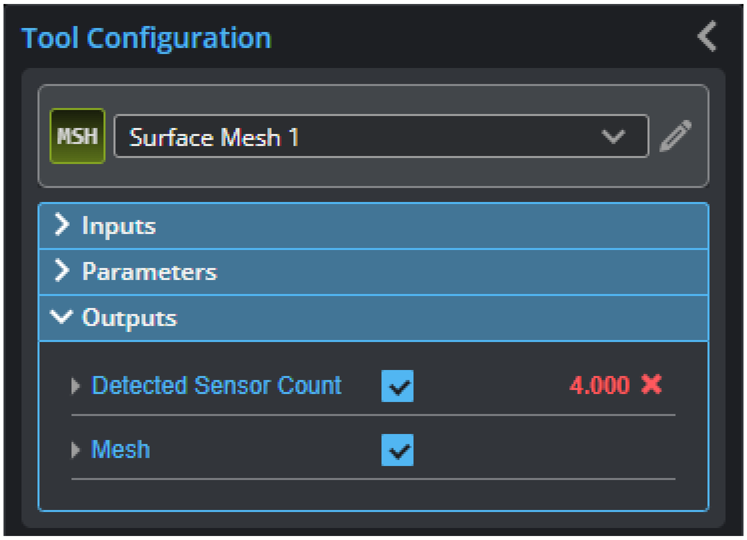
Parameters

Parameter	Description
Enable Processing	Causes the tool to start processing scan data from individual sensors, combining it into a Mesh data output. Make sure to load the XML
Configuration File	Expandable section containing configuration file operations and alignment status.
Operation	Actions that apply to the tool's XML configuration files. The files are located in `C:\GoTools\SurfaceAlign`. One of the following: • **Normal**: The tool automatically chooses this operation after you have chosen another operation. • **Load**: Displays a list of configuration files you can load. After you select a file, the tool loads it and displays a message in the log. The settings in the file, such as the number of sensors and their X and Y origin, are updated in the tool's parameters. • **Save**: Saves the sensor alignment information to a configuration file, using the

Parameter	Description
	new name you provide in the **Configuration Name** field that displays when you choose this option. This lets you save the alignment information if you have made manual changes to the sensor positions or orientations using the **Sensor Parameters** settings. • **Delete**: Deletes the file you select. • **Refresh**: Refreshes the list of files.
Alignment Status	Indicates whether the tool has aligned the sensors. Either "Not Aligned" or the date and time of the alignment.
Sensors	Expands to show sensor-related settings.
Sensor Count	Sets the number of sensors in the system.
Sensor Parameters	A drop-down that lets you display the settings of a specific sensor. You do not usually need to change these settings, as they are set when you load the XML file produced by Surface Align Ring using the **Operation** drop-down. The values are intended for diagnostics only. For information on the parameters, see *Sensor Parameters* on page 192.
Smoothing	If enabled, applies smoothing to each sensor's frame before stitching them together. Set the smoothing level in **Smooth Window**. For low density surfaces, you should disable smoothing to preserve the edge details.
Smooth Window	Sets the smoothing level. In the following, smoothing level is disabled on the left, and set to 5 on the right. 
Maximum Mesh Tilt	The maximum acceptable angle between a sensor's XY plane and the generated mesh triangles. Using a smaller value can result in a cleaner transition where the Surface scan data of individual sensors are stitched, reducing the inclusion of outlier data points or the creation of artifacts.
Distance tolerance	The maximum acceptable distance between the data points in overlapping Surface data.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Measurements

Measurement
Detected Sensor Count The number of sensors detected in the system.

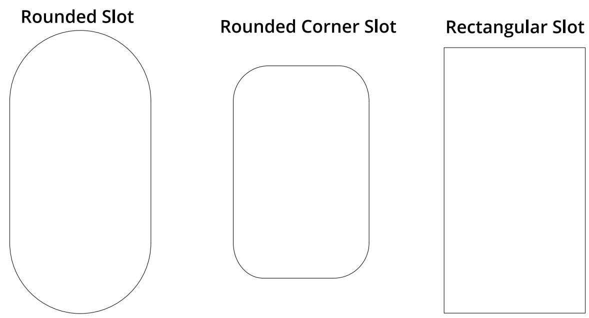
Data

Type	Description
Mesh	The Mesh data resulting from combining the scan data of the individual sensors. This data output can be taken as input by the Mesh tools (see *Mesh Measurement* on page 747). Use Mesh Projection or Mesh Plane to extract Surface data from this output, which can then be measured using any Surface measurement tool.

Surface Opening

The Opening tool locates rounded, rectangular, and rounded corner openings. The opening can be on a surface at an angle to the sensor.

	The tool does not search for or detect the feature. The tool expects that the feature, conforming reasonably well to the defined parameters, is present and that it is on a sufficiently uniform background.

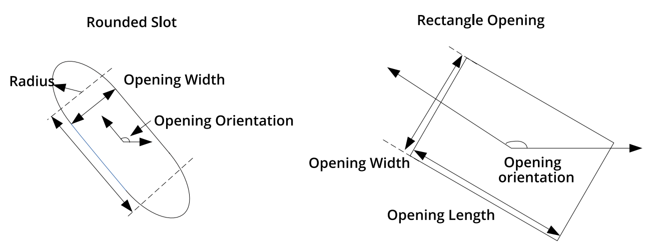


The tool uses a complex feature-locating algorithm to find a hold and then return measurements. For a detailed explanation of the algorithm the tool uses, see *Opening Algorithm* on the next page. The behavior of the algorithm can be adjusted by changing the tool's parameters.

The tool can separate out background information that appears inside the opening. It can also detect a slot that only partially appears in the data.

The shape of the opening is defined by its type and its nominal width, length, and radius.

The orientation defines the rotation around the normal of the alignment plane.

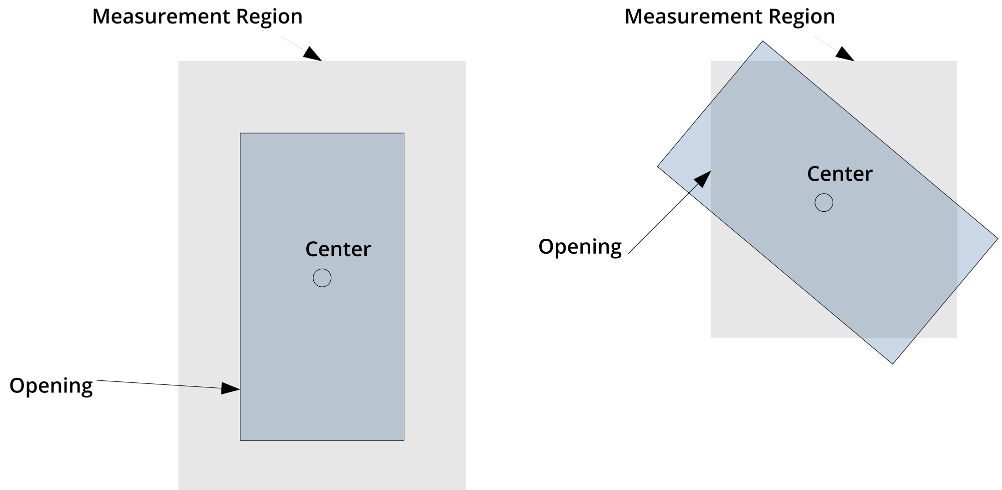


Measurement Panel

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Measurement Region

The center and the two sides and ends of the opening must be within the measurement region, even if **Partial Detection** is enabled.



Opening Algorithm

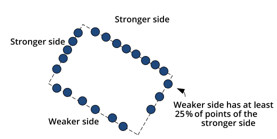
The Opening tool processes the data in three phases: Search, Measure, and Filter.

See the tool's parameters for an explanation of the options that affect the tool's algorithm.

Search phase - The tool searches for coarse data transitions (edge data) and performs a coarse fitting of the opening shape (specified by the orientation angles and the nominal dimensions) to determine the most likely candidate. If **Tilt Correction** is enabled, the algorithm uses the flat surface in the measurement region to estimate the orientation of the part.

Measure phase - A more rigorous edge detection algorithm is applied to precisely determine the edges around the feature. Edge detection at this stage will reject outliers and noise. The algorithm requires opposite sides and ends to be associated with a comparable number of edge pixels, with the weaker side or end having at least 25% of the stronger.

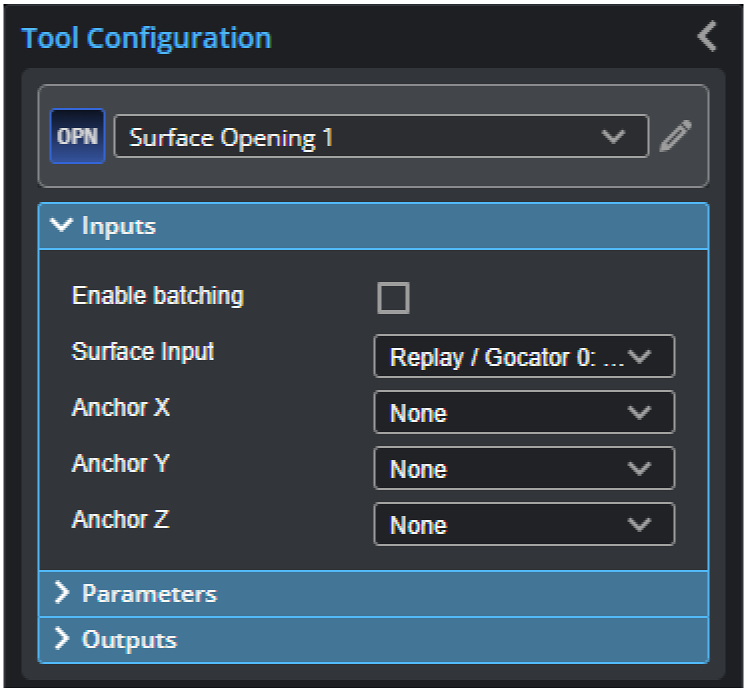
The set of refined edges is then used to locate and inspect the feature. If the **Reference Regions** setting is enabled, the edges are also used to calculate the location of the reference regions.



Filter phase - The detected location and dimensions are compared to the nominal and tolerance settings. If the refined feature falls within the measurement region and its measurements fit within the specified tolerance, the results are reported. If not, an invalid result is returned.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



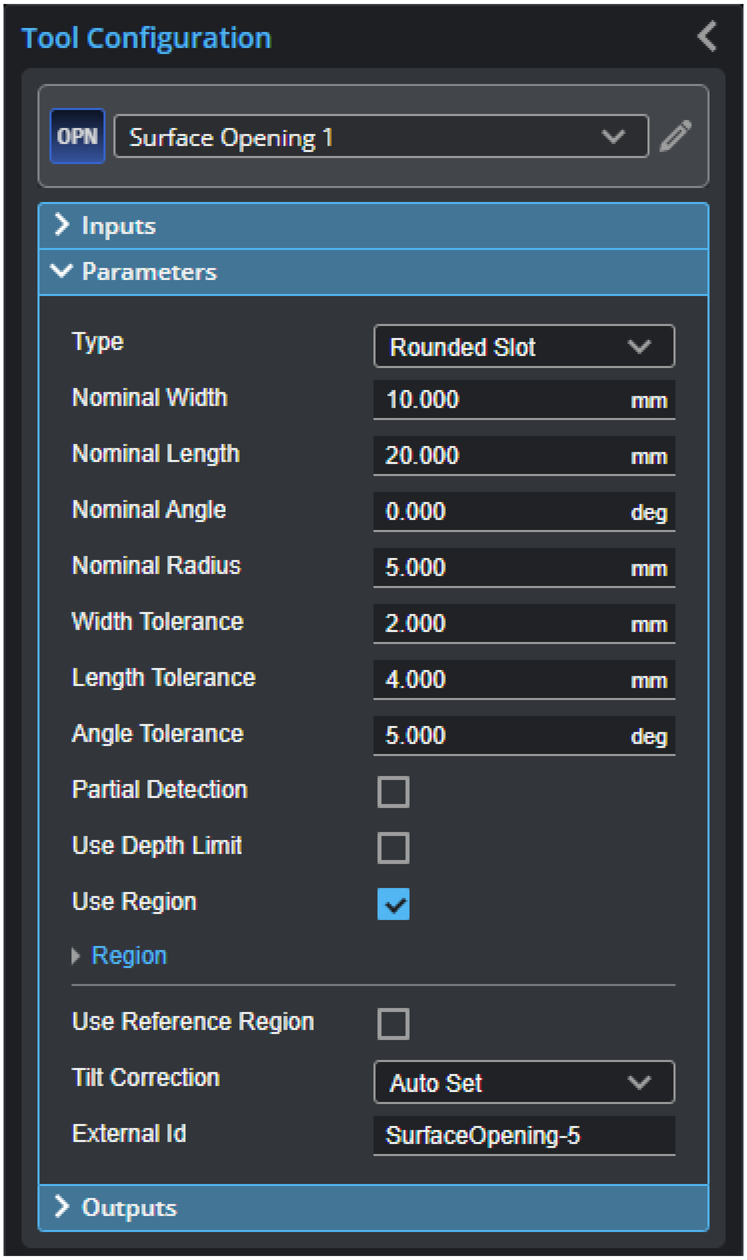
	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

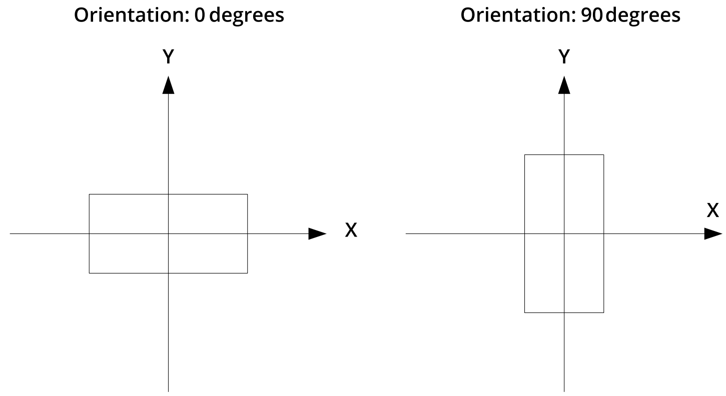
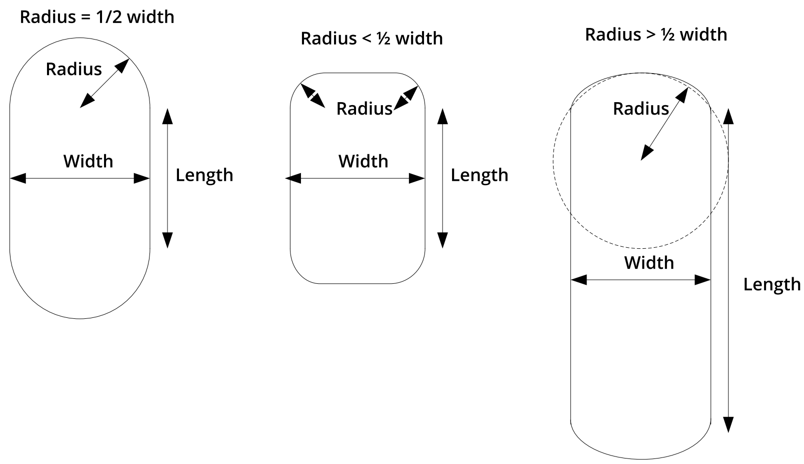
Parameters

You configure the tool's parameters in the expandable **Parameters** section.

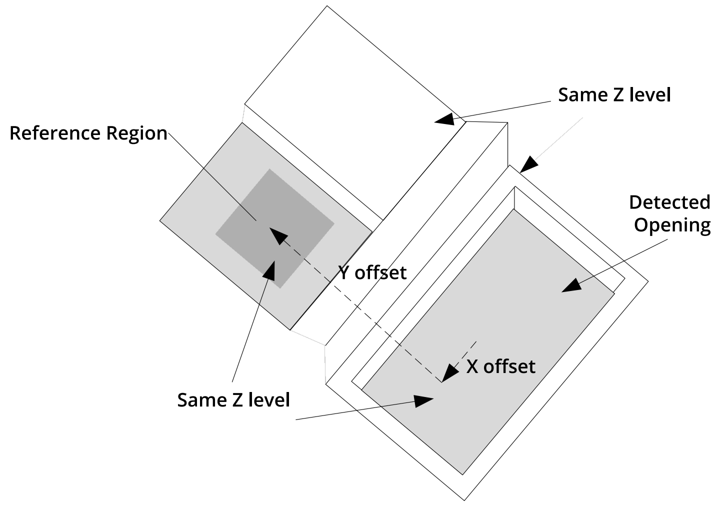


Parameters

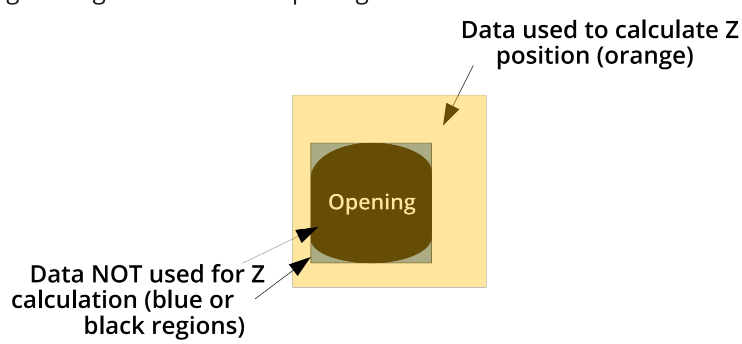
Parameter	Description
Type	One of the following: Rounded Slot, Rectangle.
Nominal Width	Nominal width of the opening.
Nominal Length	Nominal length of the opening.
Nominal Angle	Nominal angle of the opening. The default orientation is the length of the opening along the X axis.

Parameter	Description
	
Nominal Radius	Nominal radius of the opening ends. If the opening type is set to rectangular, the radius setting is disabled. The opening has an oval shape if the radius is equal to ½ of the width. The opening is a rounded rectangle when the radius is less than ½ of the width. 
Width Tolerance Length Tolerance Angle Tolerance	The maximum variation from the nominal width, length, and angle (+/- from the nominal value).
Partial Detection	Enable if only part of the opening is within the measurement region. If disabled, the opening must be completely in the region of interest for results to be valid.
Use Depth Limit Depth Limit	When **Use Depth Limit** is enabled, data below the value set in **Depth Limit** (relative to the surface) is excluded from the opening calculations.
Use Region	If enabled, displays an expandable region section where you define the region.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Use Reference Region Reference Type Reference Region {n}	When **Use Reference Region** is enabled, the tool uses reference regions to calculate the Z position of the opening. Reference regions are relative to the center location of the feature. This option is typically used in cases where the

Parameter	Description
	surface around the opening is not flat.



When the Reference Regions setting is disabled, the tool measures the opening's Z position using all data in the measurement region, except for a bounding rectangular region around the opening.



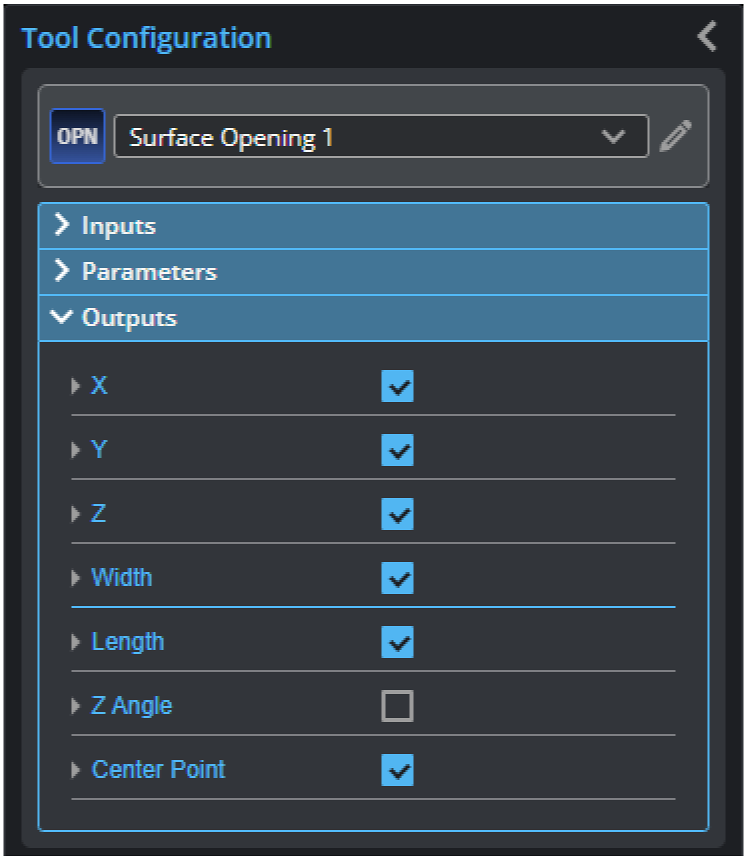
With one or more reference regions, the algorithm calculates the Z positions as the average values of the data within the regions.

When you place the reference region manually, all of the data is used, whether the data is inside or outside the opening. You should place the reference region carefully.

Parameter	Description
Tilt Correction	Tilt of the target with respect to the alignment plane. **Autoset**: The tool automatically detects the tilt. The measurement region to cover more areas on the surface plane than other planes. **Custom**: You must enter the X and Y angles manually in the X Angle and Y Angle parameters (see below).
X Angle Y Angle	The X and Y angles you must specify when **Tilt Correction** is set to **Custom**. You can use the Surface Plane tool's X Angle and Y Angle measurements to get the angle of the surrounding surface, and then copy those measurement's values to

Parameter	Description
	the **X Angle** and **Y Angle** parameters of this tool. For more information, see [Plane]().
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

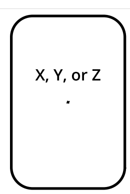
Outputs

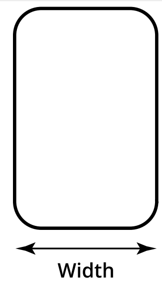
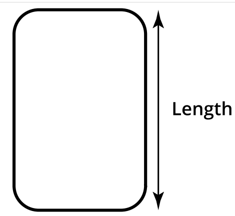
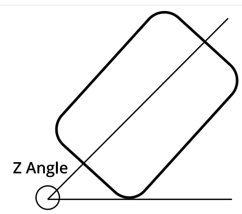


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Determines the X position of the opening's center.	
Y Determines the Y position of the opening's center.	
Z Determines the Z position of the opening's center.	

Measurement	Illustration
Width Determines the width of the opening.	
Length Determines the length of the opening.	
Z Angle Determines the angle (rotation) around the normal of the alignment plane.	

Features

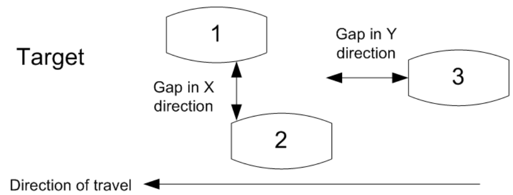
Type	Description
Center Point	The center point of the opening. The Z position of the center point is at the Z position of the surrounding surface.

For more information on geometric features, see *Geometric Features* on page 262.

Surface Part Detection

	The Surface Part Detection tool is usually used with G3 sensors.

The Surface Part Detection tool identifies discrete objects in Surface data and then outputs them as Surface data. Surface measurements can then be performed on each object.



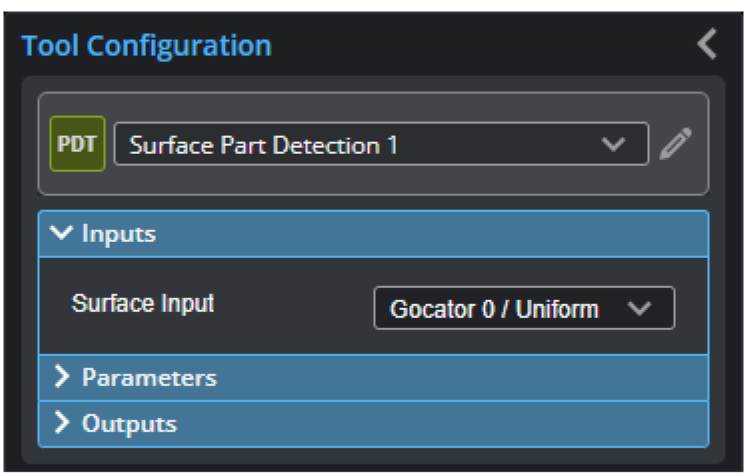
	You can isolate and then measure parts using Surface Blob and Surface Segmentation tools (for more information on these tools, see *Surface Blob* on page 455 and *Surface Segmentation* on page 635). For a comparison of part detection and these tools, see *Isolating Parts from Surface Data* on page 444.

Part detection can be performed when **Source** on the **Acquire** > **Scan** page in the **Trigger** panel is set to **Time** or **Encoder**. To use the **Time** trigger source, the travel speed must be calibrated. To use the **Encoder** trigger source, the encoder resolution must be calibrated.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

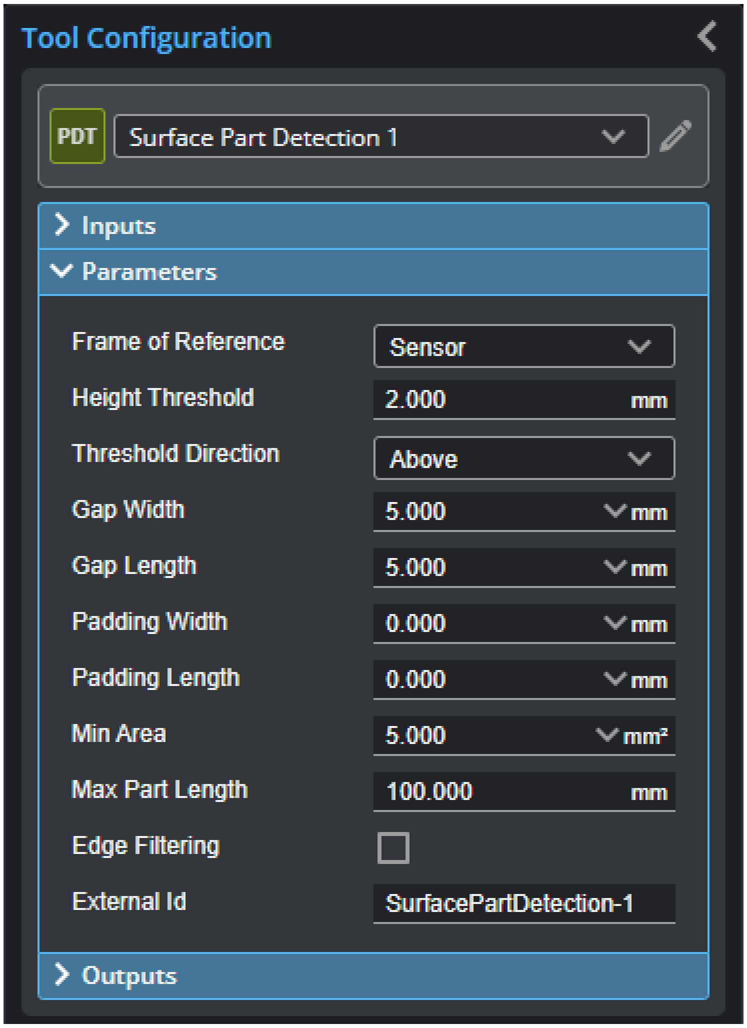


Inputs

Name	Description
Surface Input	The source of data for the tool.

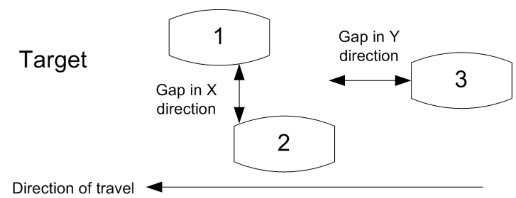
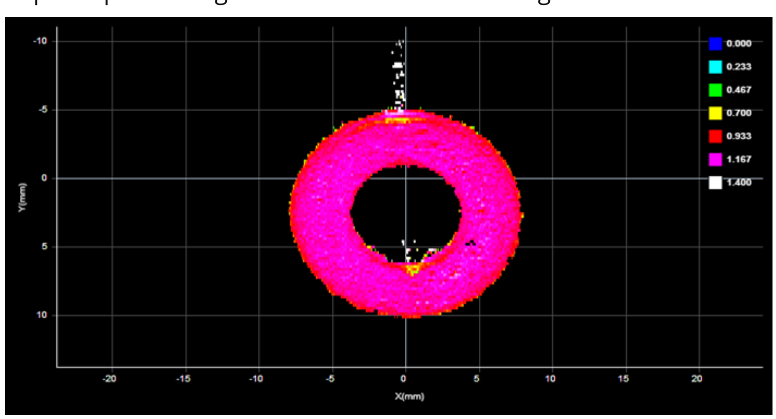
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



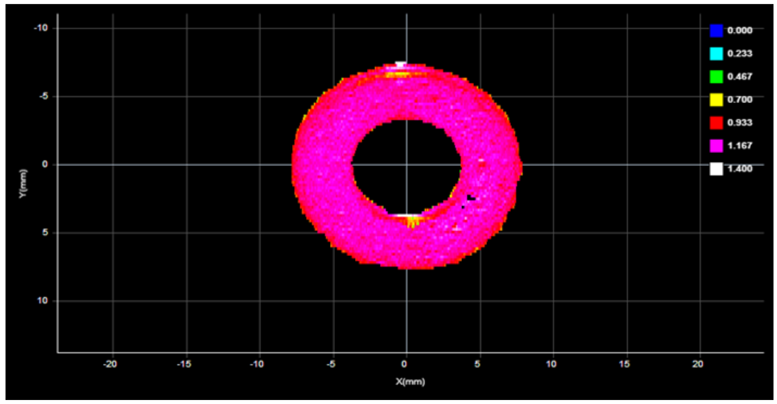
Parameters

Parameter	Description
Frame of Reference	Determines the coordinate reference for surface measurements. **Sensor** When **Frame of Reference** is set to **Sensor**, the sensor's frame of reference is used. **Part** When **Frame of Reference** is set to **Part**, all measurements are relative to the center of the bounding box of the part. For Bounding Box X and Y, the measurement values are always relative to the sensor frame of reference (see *Surface Bounding Box* on page 466).
Height Threshold	Determines the height threshold for part detection. The setting for **Threshold Direction** (see below) determines if parts should be detected above or below the

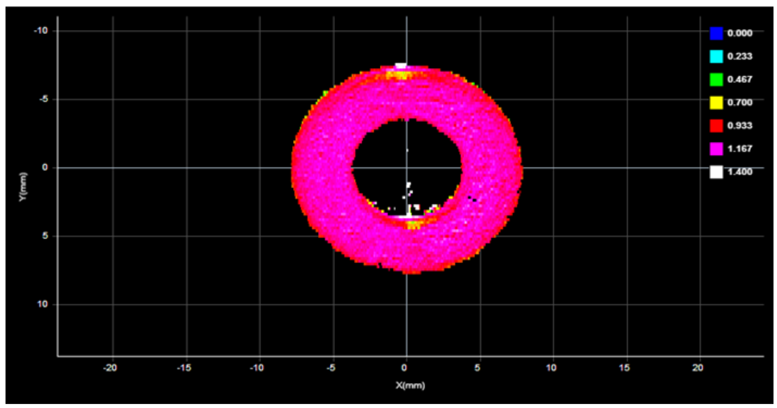
Parameter	Description
	value in **Height Threshold**. Above is typically used to prevent the belt surface from being detected as a part when scanning objects on a conveyor.
Threshold Direction	Determines if parts should be detected above or below the height threshold.
Gap Width Gap Length	**Gap Width** and **Gap Length** determine the minimum separation between objects on the X and the Y axis, respectively. If parts are closer than the gap interval, they will be merged into a single Surface output. 
Pad Width Pad Length	These parameters are useful when processing part data with third-party software such as HexSight, Halcon, etc. **Pad Width** and **Pad Length** control the amount of additional scan data output in the X and Y directions, respectively. The padding can contain data points that were outside the height threshold and excluded from the initial part detection.
Min Area	Determines the minimum area for a detected part. Set this value to a reasonable minimum in order to filter out small objects or noise.
Max Length	Determines the maximum length of the part object. When the object exceeds the maximum length, it is automatically separated into two parts. This is useful to break a long object into multiple sections and perform measurements on each section.
Edge Filter	When **Edge Filter** is enabled, an **Edge Filter** expanding section is displayed that contains the settings described below. Part scans sometimes contain noise around the edges of the target. This noise is usually caused by the sensor's light being reflected off almost vertical sides, rounded corners, etc. Edge filtering helps reduce edge noise in order to produce more accurate and repeatable volume and area measurements, as well as to improve positioning of relative measurement regions. 

Parameter	Description

Edge Filter disabled (scan shows reflection noise)



Edge Filter enabled (reflection noise eliminated or reduced)

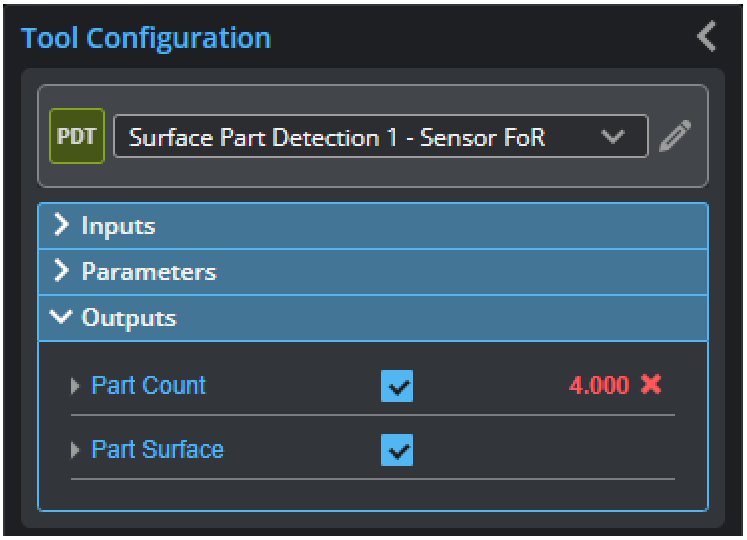


Edge Filter enabled, Keep Interior enabled

External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Measurements

Measurement
Part Count The number of parts the engine detects.

Data

Type	Description
Part Count	The number of parts detected.
Part Surface	The Surface data of the detected part.

Surface Pattern Matching

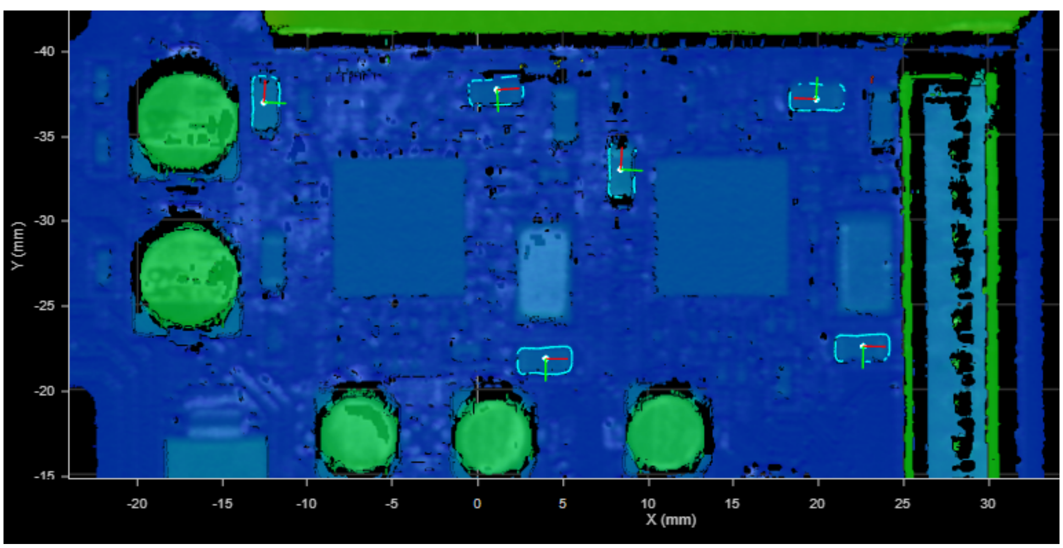
The Surface Pattern Matching tool locates parts and features by comparing 2D contours (on the XY plane) found in scan data to pattern models you have defined; note that the tool does not use height data in its algorithms. Models represent contour-based "golden parts" or "golden features." (Models can be modified in the provided standalone model editor; for more information, see *Pattern Editor* on page 951.)

The tool can process multiple occurrences of a part or feature in a frame of scan data. For each matching part or feature (called an instance), the tool returns X and Y position and rotation (angle) measurements, which you can use to anchor other measurements (for more information on anchoring, see *Measurement Anchoring* on page 264).

The tool also returns a point and a line geometric feature for each instance, which you can use in conjunction with Surface Transform tools to shift and rotate scan data to reliably position the target.

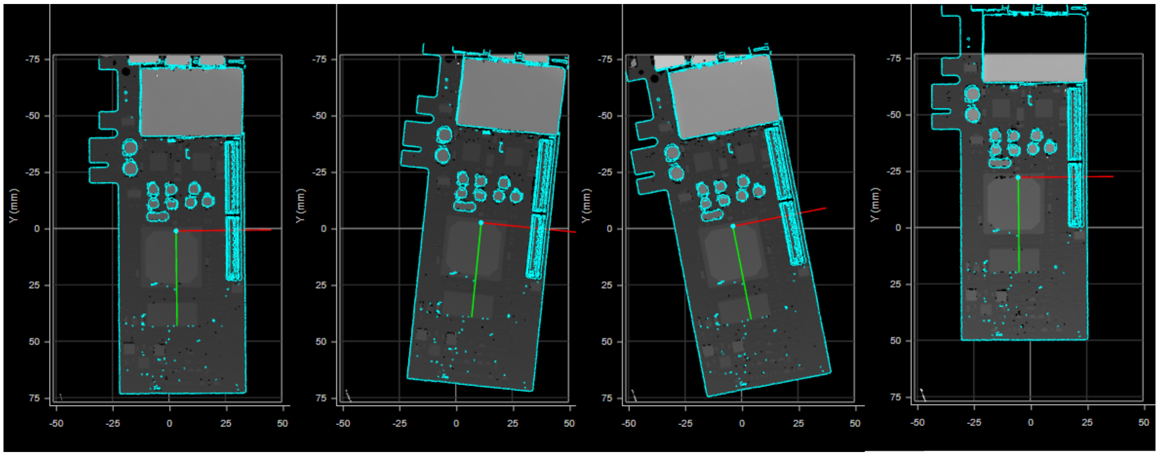
Finally, the tool returns a match quality that you can use as a general conformity measure for matching instances (for example, checking for dents in a target), as well as a count of located instances.

> In order to create a template for a *feature* on a target, you typically need to enable the **Use Region** checkbox to limit the tool to the contours related to that feature. After that, when running the tool to find instances of the feature, you should either modify the region to limit it to areas of the target that might contain the feature you are looking for or *disable* **Use Region** so that the tool can locate instances of the feature in all of the scan data. You can also use the **Use Region** parameter when creating a template to limit it to a unique portion of an outer edge of a target.



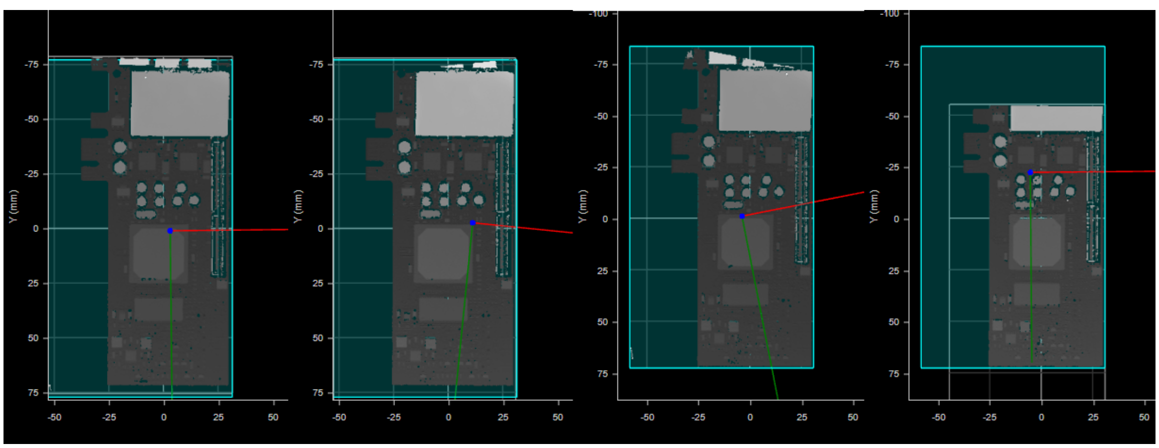
For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Note that when you use the geometric features with a Surface Transform tool to transform the scan data from frame to frame, you can often avoid the need to anchor other measurements, because the transforms ensure that any features you are interested are always in the same location. This can save considerable setup time and reduce the complexity of an application. For example, in the following frames of scan data, in which a PCB shifts from frame to frame, a Surface Pattern Matching tool successfully locates the entire PCB using its outer contours and the contours of various components on the PCB, as indicated by a dark blue outline. Note the “missing” data in the second and fourth frames, on the lower right and left edges, respectively: the tool still locates the PCB, despite the occlusions.



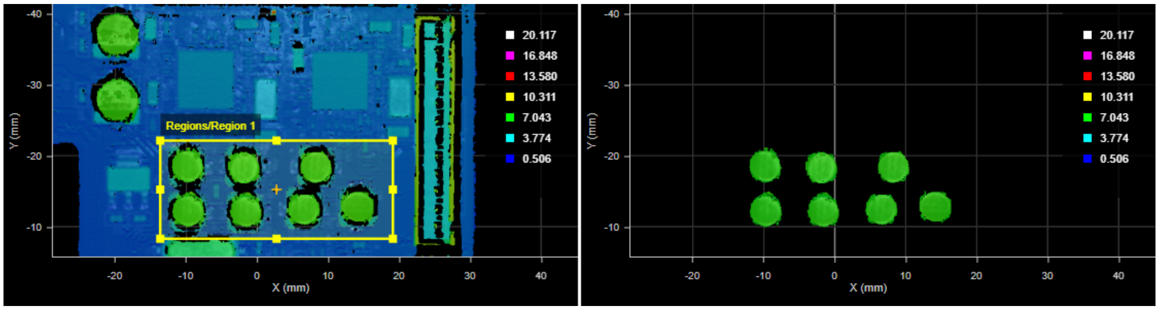
Four frames of scan data. Light blue outline represents the matching template. The first frame was used to create the template.

When the tool’s Point and Line geometric features are passed to a Surface Transform tool, the transformed scan data ensures that features are always in the same location and orientation.



Transformed scan data of the four frames.

Other measurement tools can then be placed over the capacitors, without needing to anchor them. In the following image (the third frame, which was significantly shifted and rotated), a Surface Filter tool isolates the capacitors based on height. Subsequent tools can perform measurements on the isolated data to verify that all capacitors are present, are seated properly, and so on.

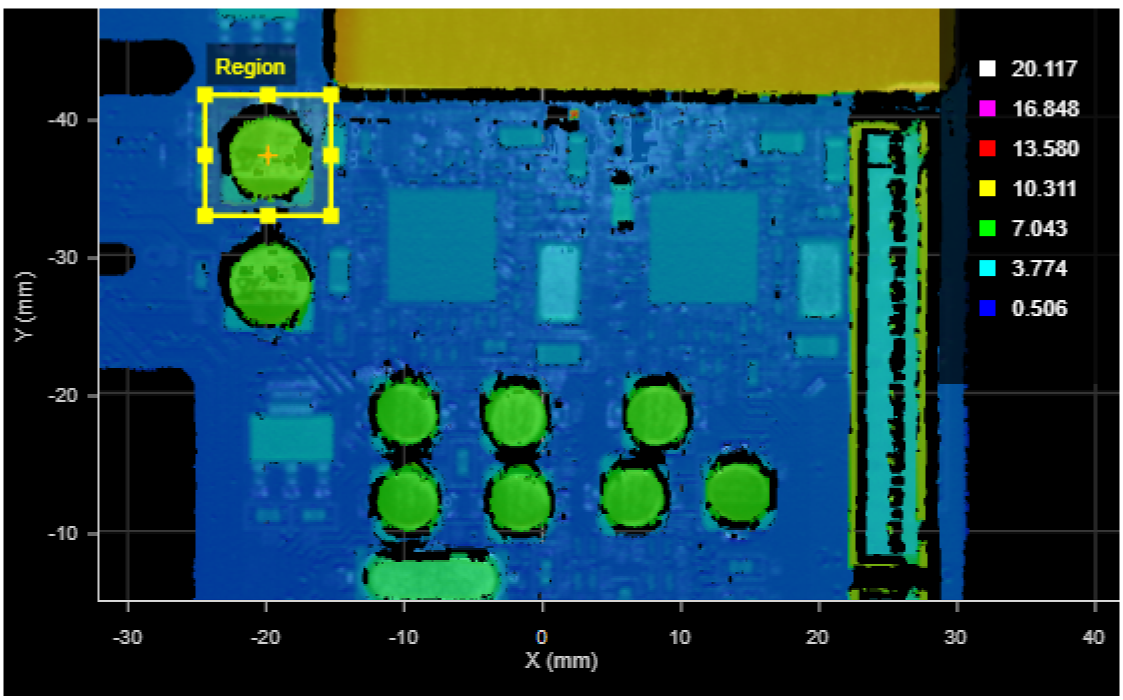


By adding multiple Surface Pattern Matching tools to a job and defining different templates for each, you can match multiple types of features or parts, for example, matching different parts moving on a conveyor, or different types of features on a single target.

Creating a Template

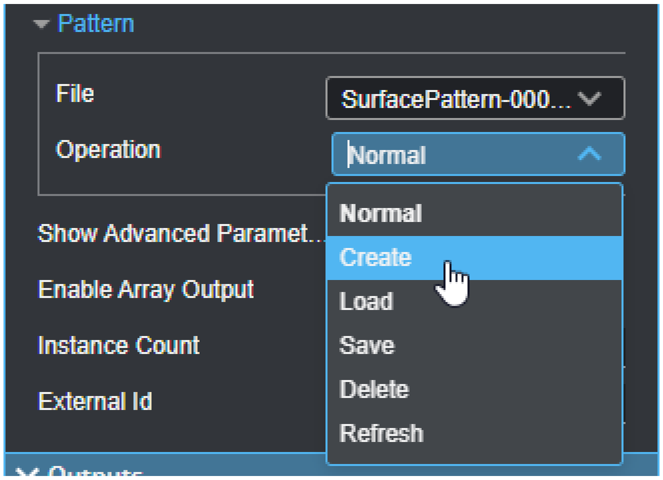
To create a template:

1. Scan a part that is representative of a "good" part (no damage, all features are present, and so on).
2. Add an instance of Surface Pattern Matching.
3. If you need to perform pattern matching on a feature on the part, enable **Use Region** and position the region over the feature.

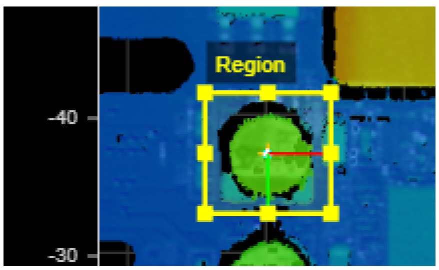


Otherwise, if you need to match the entire part, for example to orient parts from frame to frame, leave **Use Region** disabled to perform pattern matching on all of the scan data. This only works if no unwanted data is in the frame, such as the conveyor.

4. In the **Operation** drop-down, choose **Create**.



The tool creates a model and saves it either to the PC or to the sensor. The tool also "matches" the pattern.

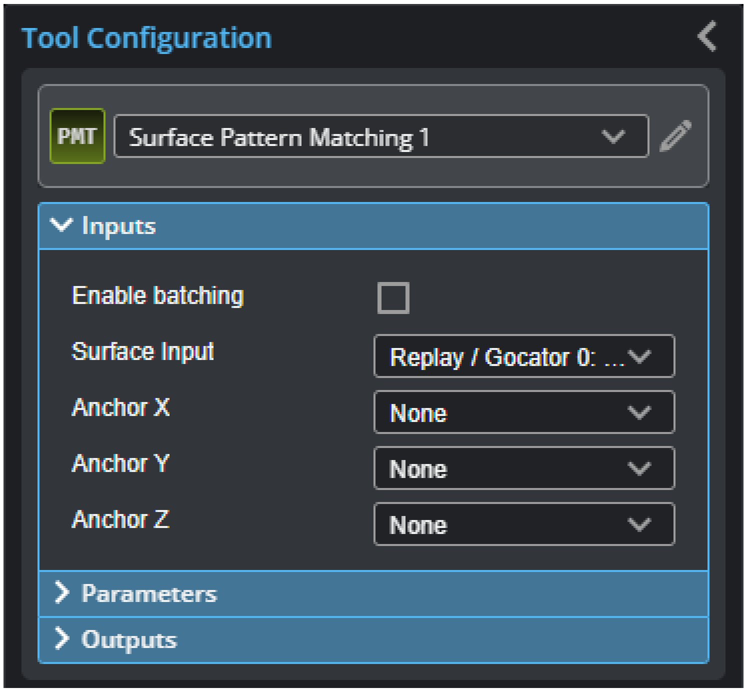


5. Disable, resize, or reposition the region as necessary to match the pattern in parts you will be scanning in production.

After creating a template, configure the tool's parameters (see below) for use during production runs.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



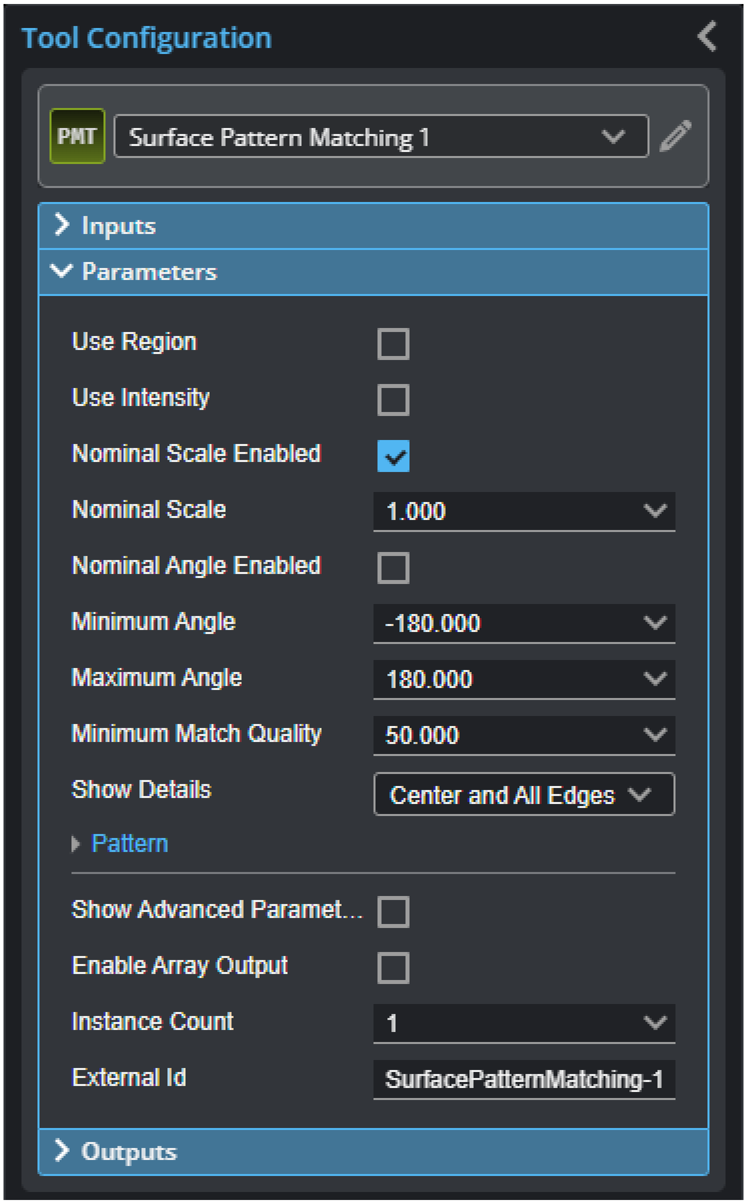
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Use Region	Determines whether the tool uses a user-defined region to search for matching instances, or, when first creating a template, whether the tool limits establishing template contours to the data in the ROI.
Region	Settings related to the region. For more information, see *Regions* on page 250.
Use Intensity	Determines whether the tool uses intensity instead of heightmap data to locate instances or create templates.
Nominal Scale Enabled	If **Nominal Scale Enabled** is checked, the tool displays the **Nominal Scale** setting and the tool uses the user-defined nominal scale. Otherwise, the tool displays **Minimum Scale** and **Maximum Scale** settings and uses the user-defined range.

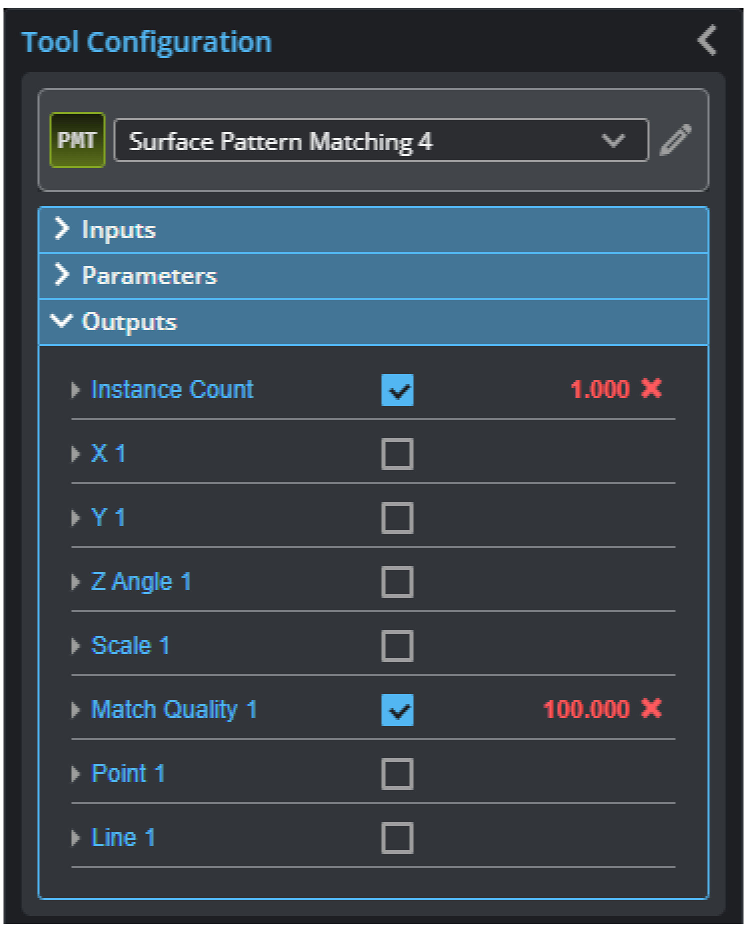
Parameter	Description
Nominal Scale	The scale factor the tool requires to recognize an instance. Displayed when **Nominal Scale Enabled** is enabled.
Minimum Scale Maximum Scale	The minimum and maximum scale factors set the acceptable scaling range of an instance. Displayed when **Nominal Scale Enabled** is disabled.
Nominal Angle Enabled	If enabled, displays the **Nominal Angle** setting and the tool uses the user-defined nominal angle. Otherwise, the tool displays **Minimum Angle** and **Maximum Angle** settings and uses the user-defined range.
Nominal Angle	The angle the tool requires to recognize an instance. Displayed when **Nominal Angle Enabled** is enabled.
Minimum Angle Maximum Angle	The minimum and maximum angles allowed for the tool to recognize an instance, respectively. Displayed when **Nominal Angle Enabled** is disabled.
Minimum Match Quality	Minimum percentage of template contours that must match in the scan data for the tool to consider the object instance as valid.
Show Detail	Toggles whether to overlay tool details. **Center only**: The center of the found pattern and its XYZ axes. **Center and All Edges**: The center of the found pattern and its XYZ axes, as well as the outline of the full pattern template. **Center and Matched Edges**: The center of the found pattern and its XYZ axes, as well as the matched edges only.
Pattern	When expanded, displays parameters related to template files. **File** A drop-down containing the currently available templates. **Operation** The operation to perform on the currently selected template in the File drop-down. One of the following: **Normal**: The default value after having performed another operation. **Create**: Creates a new template based on the current frame of scan data. Limited to the region if **Use Region** is enabled. **Load**: Loads the currently selected template. **Save**: Saves contour data to the currently selected template, overwriting its contour data. **Delete**: Deletes the currently selected template. **Refresh**: Refreshes the list of templates. This is useful if you have renamed a template (using the Pattern Editor or manually in the client PC's file system).

Parameter	Description
Show Advanced Parameters	When enabled, the tool displays the following additional advanced parameters. **Sorting** The sort order of instances the tool uses. One of the following: • Evidence: The instances are ordered according to their hypothesis strength. • Quality: The instances are ordered according to their match quality. This setting can significantly increase the tool's search time because it can't output instance results until it has found and compared all instances to determine their order. The time required to output the first instance corresponds to the total time needed to search the image and analyze all the potential instances. The time for additional instances is zero since the search process is already complete. • Position - X increasing • Position - X decreasing • Position - Y increasing • Position - Y decreasing **Recognition Level** The "effort" the tool will expend on recognizing an instance in scan data. Ranges from Fast to Accurate (that is, there is a trade-off between accuracy and speed). Only used during pattern matching (and not during pattern template creation). **Positioning Level** How accurately the tool determines the position of the instance. Ranges from Fast to Accurate. Only used during pattern matching (and not during pattern template creation). **Match Tolerance Mode** The tolerance the tool uses when attempting to match contours in the scan data to the loaded template; this is the maximum distance between the contours in the template and in the scan data. Either Pattern-based (a looser tolerance, the default), Resolution-based (a tighter tolerance), or Manual. Typically, the Pattern-based option should produce a better match quality. However, if you are getting too many false matches, try setting the mode to Resolution-based or set the value manually. The tolerances calculated from the pattern-based and resolution-based modes are displayed in the mode drop-down. Note that the resolution-based mode results in a much tighter tolerance with high resolution scan data. **Add Border**

Parameter	Description
	Considers a drop to NULL, outside the region, as an edge. Use this when performing part detection or when there is no data around the part. If there is nothing in the region, then there will be nothing in the template either.
Enable Array Output	When **Enable Array Output** is disabled, the tool produces individual outputs for each instance. (For example, X 1 and Y 1 for the first instance, X 2 and Y 2 for the second instance, and so on.) When **Enable Array Output** is enabled, the tool produces an array for each output. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Instance Count	The maximum number of instances the tool will locate.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Instance Count Returns the number of parts or features matching the loaded template up to the value set in the Instance Count parameter.
X {n} **Y {n}** The X and Y position of the center of matched instance {n}.
Z Angle {n} The angle of matched instance {n} relative to the sensor's coordinate system.
Scale {n} The scale of matched object {n} relative to the loaded template.
Match Quality {n} Percentage of matched model contours for the selected object instance. Match quality ranges from 0 to 100, with 100 being the best quality. A value of 100 means 100% of the model contours were successfully matched to the actual contours detected in the scan data. Use the **Minimum Match Quality** parameter to set the minimum acceptable value.

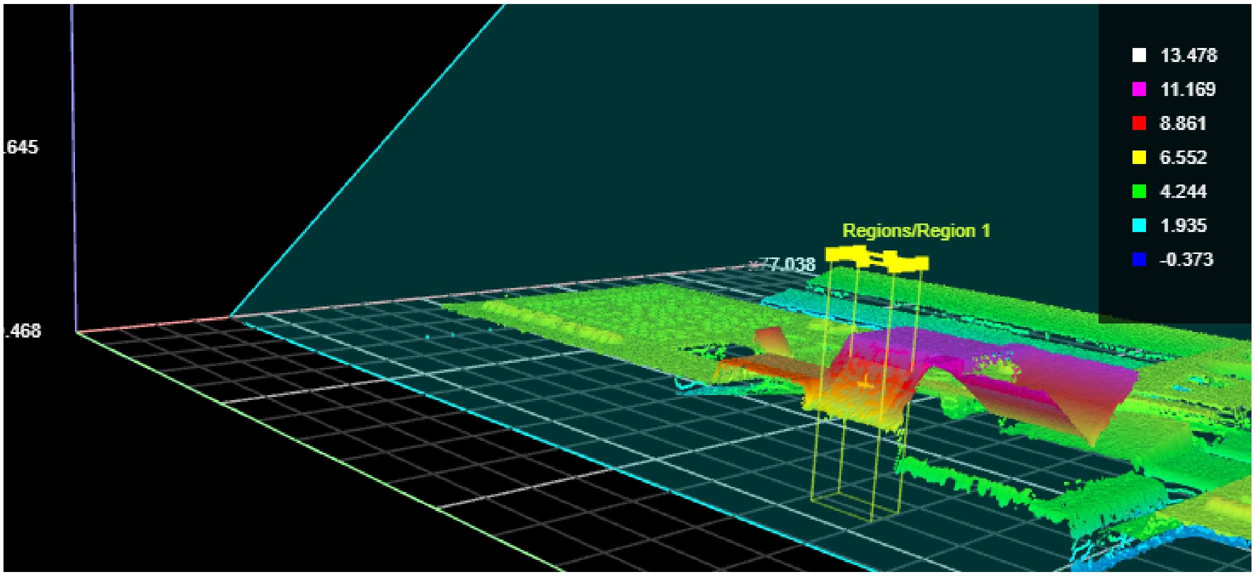
Features

Type	Description
Point	A point representing the center of the region used when creating a template and the template's default reference point. (Note that the reference point of a template can be changed in the model editor.)
Line	A line parallel to the X axis passing through the Point feature.

For more information on geometric features, see *Geometric Features* on page 262.

Surface Plane

The Plane tool provides measurements that report a plane's position and orientation (X Angle, Y Angle, Z Offset, Normal, Distance), as well as the maximum and average deviations from the plane.



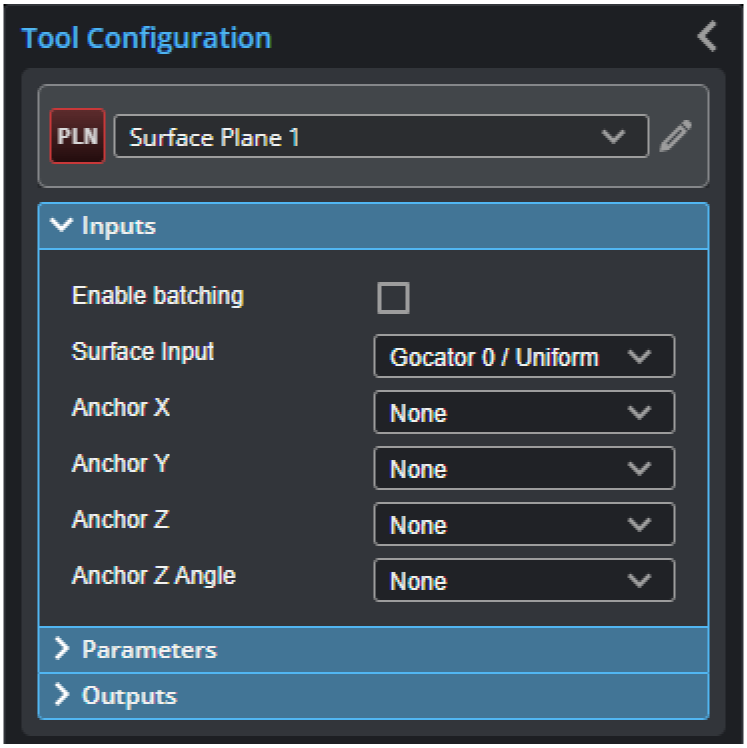
The Z offset reported is the Z position at zero position on the X axis and the Y axis.

The results of the Angle X and Angle Y measurements can be used to manually customize the tilt angle in the Hole, Opening, and Stud tools.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



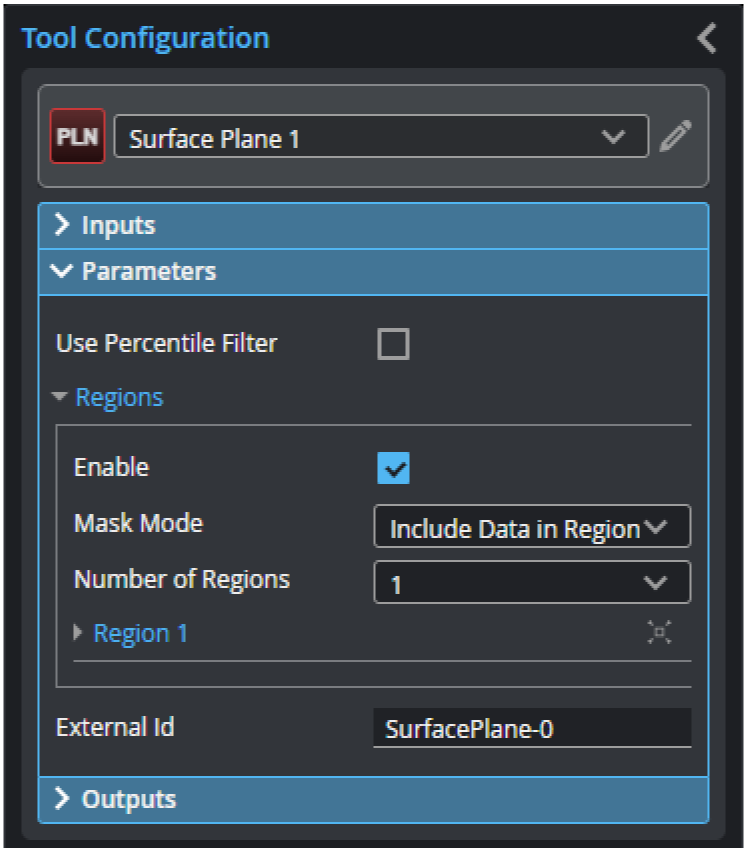
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

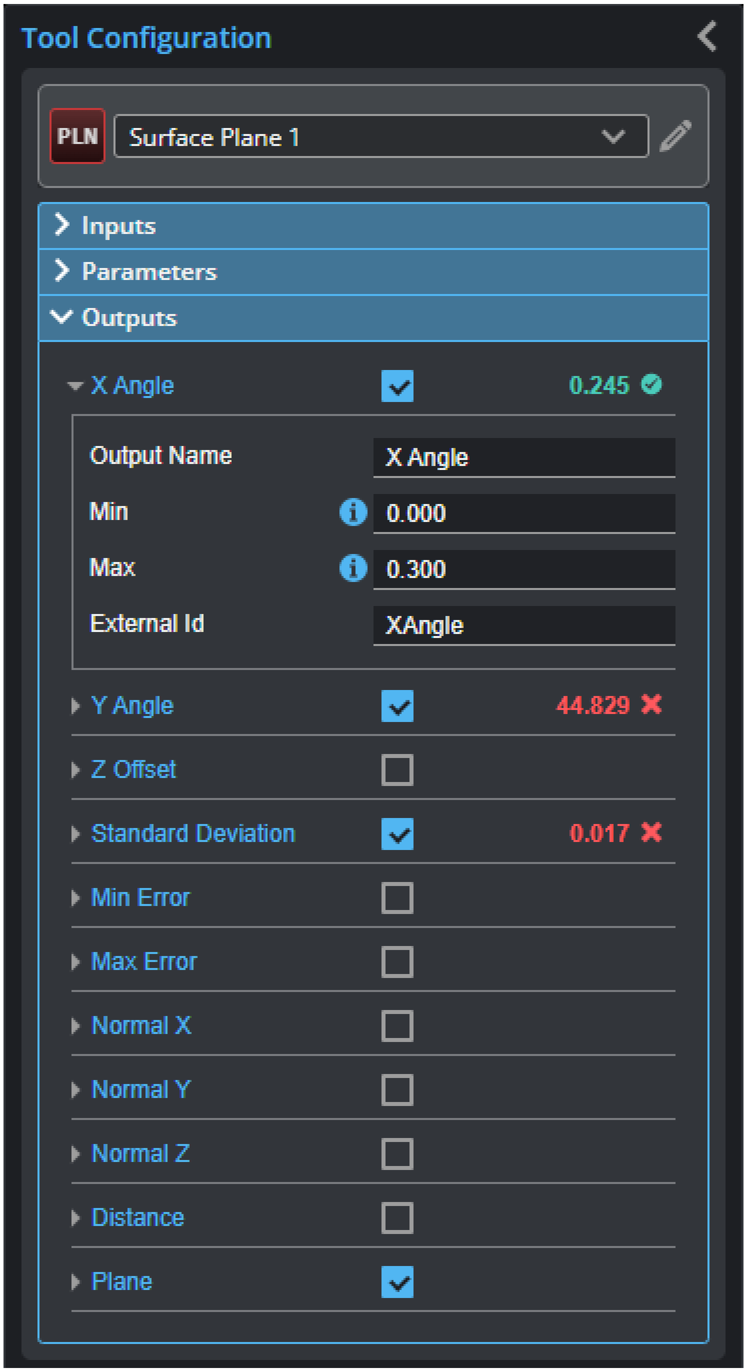
The following parameters are in the expandable **Parameters** section in the tool's configuration.



Parameters

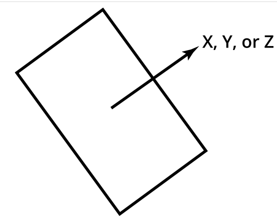
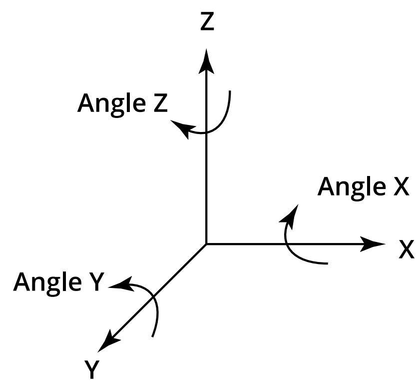
Parameter	Description
Use Percentile Filter High Percentile Low Percentile	When **Use Percentile Filter** is checked, displays parameters to set the low and high percentile. Use this to remove outliers.
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurement	Illustration
X Angle Determines the X angle of the surface with respect to the alignment target.	
Y Angle Determines the Y angle of the surface with respect to the alignment target.	
Z Offset Determines the Z value of intersection of the plane and the Z axis.	
Standard Deviation Measures the standard deviation of the points of the surface from the detected plane within the specified region or regions.	
Min Error Measures the minimum error from the detected plane (the maximum distance below the plane, perpendicular to the plane) within the specified region or regions.	
Max Error Measures the maximum error from the detected plane (the maximum distance above the plane, perpendicular to the plane) within the specified region or regions.	
Normal X Returns the X component of the surface normal vector.	
Normal Y Returns the Y component of the surface normal vector.	
Normal Z Returns the Z component of the surface normal vector.	

Measurement	Illustration
Distance Distance from the origin to the plane.	

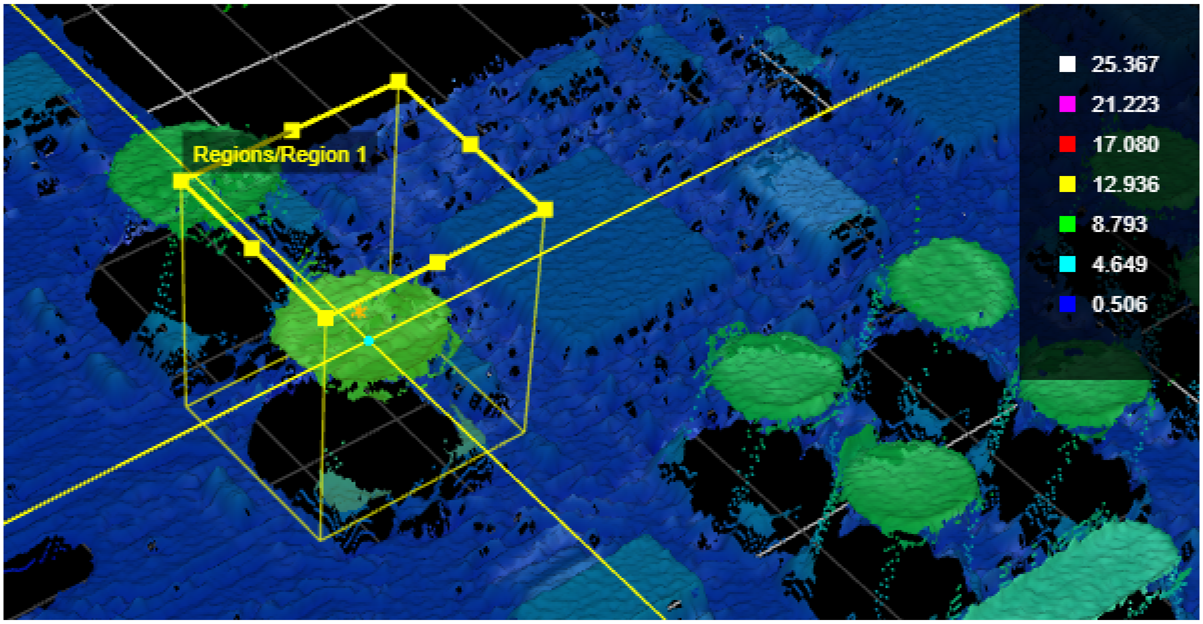
Features

Type	Description
Min Error Point Max Error Point	Points representing the minimum and maximum errors.
Plane	The fitted plane.

For more information on geometric features, see *Geometric Features* on page 262.

Surface Position

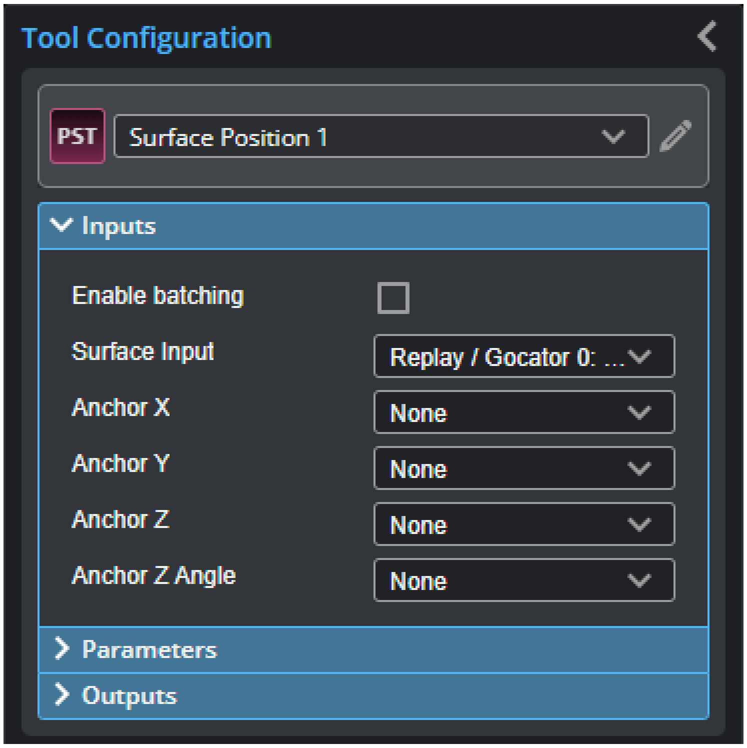
The Position tool reports the X, Y, or Z position of a part. You must specify the feature type specified and is one of the following: Average (the mean X, Y, and Z of the data points), Median (median X, Y, and Z of the data points), Centroid (the centroid of the data considered as a volume with respect to the z = 0 plane), Min X, Max X, Min Y, Max Y, Min Z, or Max Z.



Position X and Y on a PCB component

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs



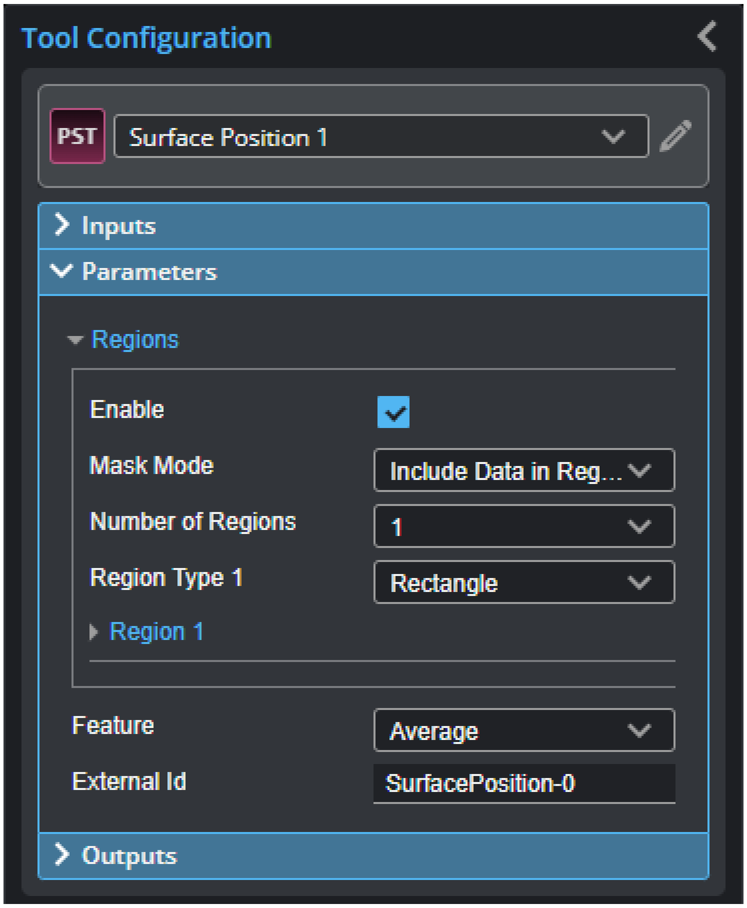
> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.

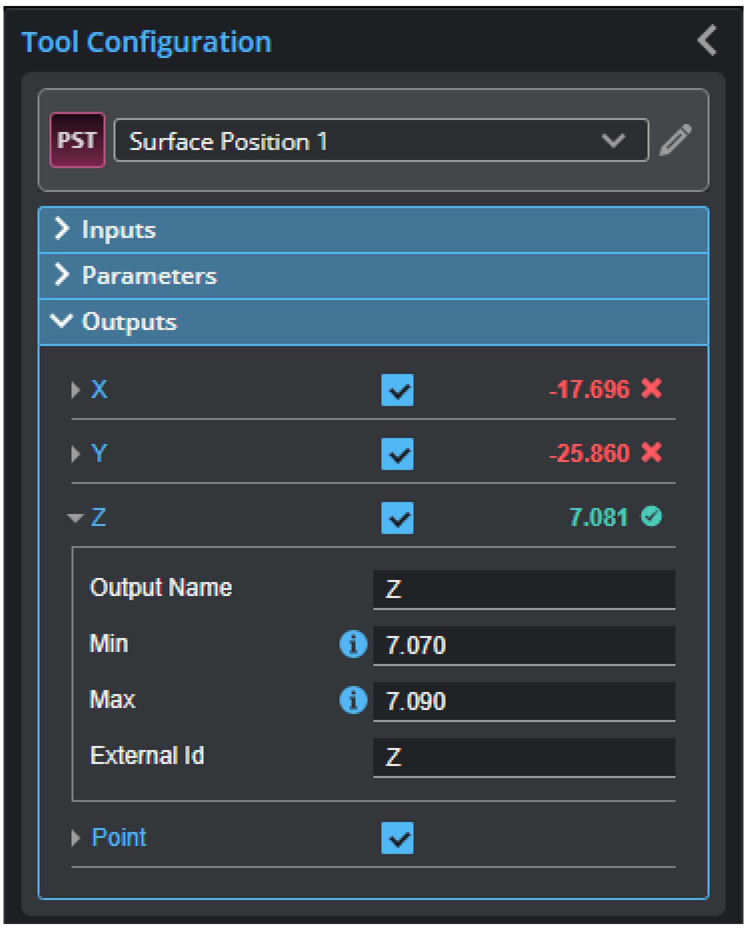


Parameters

Parameter	Description
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Feature	The feature point types the tool uses. For each, one of the following (for more information, see *Feature Points* on page 259): • Average • Median • Centroid • Max X • Min X

Parameter	Description
	• Max Y • Min Y • Max Z • Min Z When more than one point is at minimum Y or maximum Y (and the feature is set to Min Y or Max Y, respectively), the rightmost point (the one at greater positive X) is selected by the tool.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

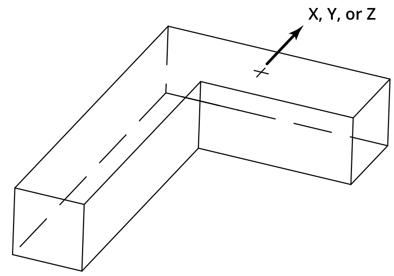
Outputs



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
X Determines the X position of the selected feature type.	
Y Determines the Y position of the selected feature type.	
Z Determines the Z position of the selected feature type.	

Features

Type	Description
Point	A Point geometric feature representing the returned position.

For more information on geometric features, see *Geometric Features* on page 262.

Surface Roughness

The Surface Roughness tool generates measurements of surface roughness.

> Because the accuracy of a roughness measurement is limited by the X resolution of a sensor, this tool is typically only used on confocal sensors, whose high resolution is better suited for roughness applications.

The following measurements are available:

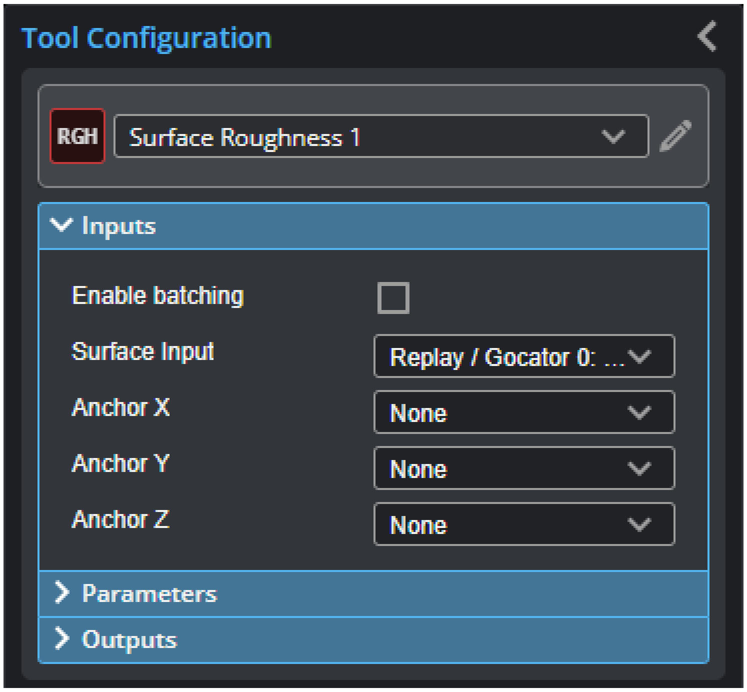
Sa	Sk
Sq	Spk
Sp	Svk
Sv	Smr1
Sz	Smr2
Sku	Sal
Ssk	Str

The tool offers five filter operations that can be enabled or disabled individually as required. The following filters are available:

- Filling Gaps: Designed to fill any holes in the surface and on the edge while trying to keep the surface as natural as possible.
- Remove Form: Offers both homogeneous and inhomogeneous polynomial filters from 1 to 8 orders.
- Remove Outliers: Used to remove outliers which are defined as a percentage of points in relation to the total number of points at the furthest distance from the height center.
- Gaussian and Cubic Spline filters: The tool offers Gaussian and Cubic Spline filters as standard filters for removing waviness. The Gaussian filter is used by default. The tool offers 4 conventional "end-effect" treatment options that are recommended by ISO 16610-28.
- The tool offers Gaussian filters with small Cut-Off wavelength (3 μm) as a low pass filter.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs



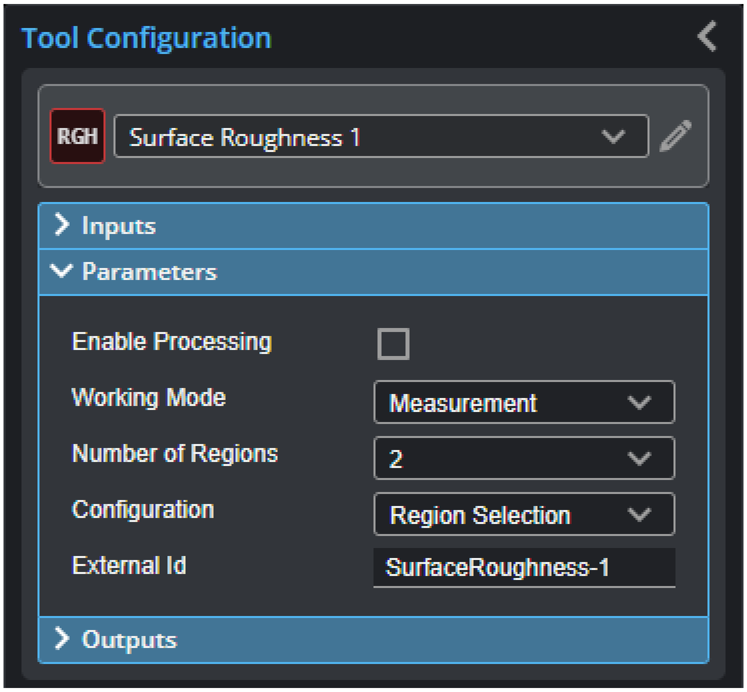
> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.

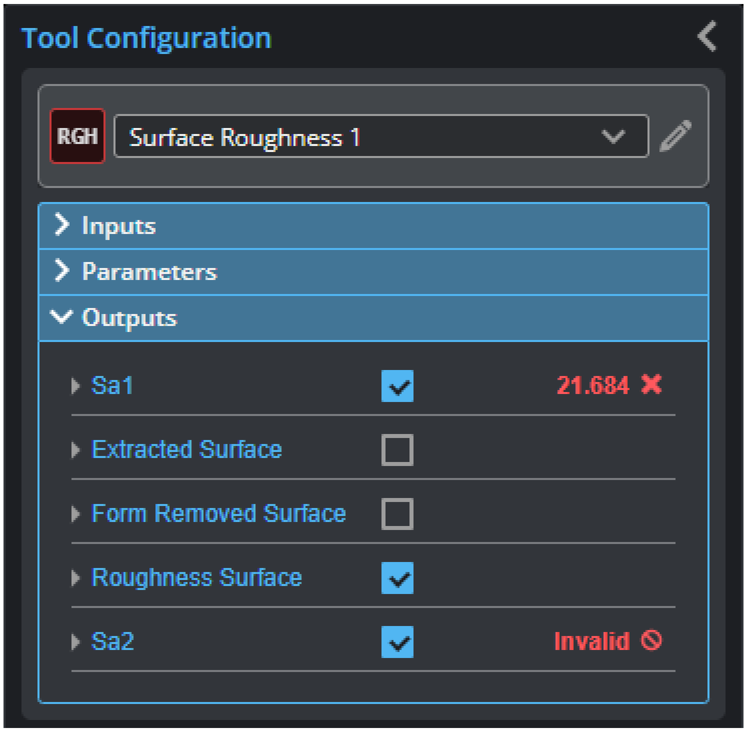


Parameters

Parameter	Description
Enable Processing	When enabled, the tool starts performing measurements.
Working Mode	The tool's working mode. One of the following: • Measurement • Check Parameters: Use this to check the calibration parameters. When you set the tool to this mode, an expandable **Calibration** section displays, containing additional parameters. For more information, see *Calibration* below. • Reset Parameters: If the calibration parameters are no longer used or a new calibration is carried out, select **Reset parameters** once. • Roughness Calibration: See *Roughness Calculation Procedure* on page 627. See *Calibration Procedure* on page 627.
Number of Regions	Sets the number of regions to be calculated, currently up to 16. (Two additional regions are reserved for calibration.)
Calibration	An expandable section displayed when **Working Mode** is set to **Check Parameters**. The tool gives the summed mean values and the recorded sample number after each measurement. For mirror calibration, the mean values of the background noise of Sa and Sq and the number of measurements are returned. For roughness calibration, the nominal values of Sa and Sq must be specified before the measurement starts.

Parameter	Description
	The mean scaling value and the number of measurements are returned.
Configuration	Lets you choose which region to configure in the expandable Region {n} section. Regions are independent. The two regions for mirror and roughness calibration targets are suspended after the specified number of evaluation regions, that is, after changing the number of evaluation regions, the settings for “Mirror” and “Roughness” must be checked again.
Region	An expandable section containing the parameters for the region selected in **Configuration**.
Use Region	If enabled, uses the region you define.
Number of Regions	Select the number of regions to be calculated, currently up to 18. Two regions are available for selection (the last 2 are reserved for calibration).
Fill Gaps Mode	Set to Linear Interpolation to fill any holes in the surface and on the edge while trying to keep the surface as natural as possible.
Remove Form	Check to remove form from the surface with homogeneous or inhomogeneous polynomial filter.
Polynomial Order	Order of polynomial function.
Remove Outliers	The outlier is defined here as a percentage of points in relation to the total number of points at the furthest distance from the height center. When using the integrated histogram curve, the outliers outside the threshold range are replaced by the corresponding limit values, which are calculated by the points within the threshold range.
Remove Waviness	Check to remove waviness using Gaussian or Cubic Spline filter.
Filter Type	Choose Gaussian filter or Spline Filter.
End Effect	Methods for dealing with end-effects. One of the following: • Zero Padding criterion (ZPA) • Linear Extrapolation (LEX) • Line Symmetric Reflection (LSR) • Point Symmetrical Reflection (PSR)
Use Low Pass Filter Low Pass Cut-off	Check **Use Low Pass Filter** to use low pass filter with cut-off wavelength defined in **Low Pass Cut-off**.
Margins	Custom margin distance.
Output Parameters	Some combination of measurement outputs can be selected. Dynamic measurement management is used.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

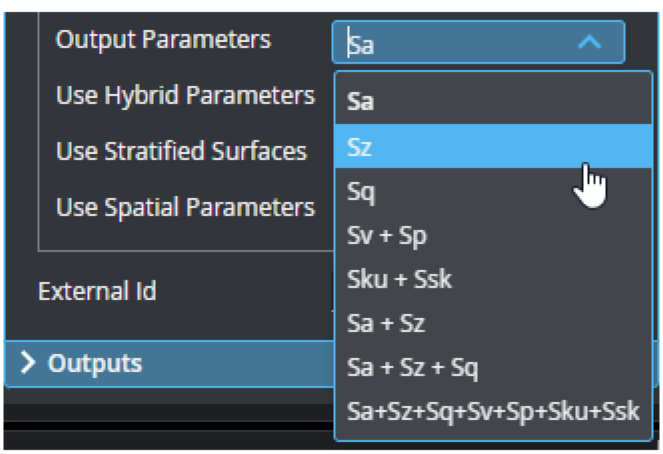
Outputs



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

You select the following parameters (or a combination of the parameters) in the **Output Parameters** drop-down to list them in the tool's **Outputs** section.



Measurement	Formula
Sa {n} Arithmetical mean height in µm, which is the difference in height of each point compared to the arithmetical mean of the surface.	$Sa = \frac{1}{A} \iint_A \lvert z(x,y) \rvert dxdy.$
Sz {n} Maximum height in µm, which is defined as the sum of the largest peak height value and the largest pit depth value within the defined area.	$Sz = Sp + \lvert Sv \rvert$
Sq {n} Root mean square height in µm, which is equivalent to the standard deviation of heights.	$Sq = \sqrt{\frac{1}{A} \iint_A z^2(x,y) dxdy}$
Sv {n} Maximum pit Height in µm, which is the absolute value of the height of the largest pit within the defined area.	$Sv = \lvert \min_A z(x,y) \rvert$
Sp {n} Height of the highest peak within the defined area in µm.	$Sp = \max_A z(x,y)$
Sku {n} Kurtosis, which is a measure of the sharpness of the roughness profile. (No unit.)	$Sku = \frac{1}{Sq^4} \frac{1}{A} \iint_A z^4(x,y) dxdy.$
Ssk {n} Skewness, which represents the degree of bias of the roughness shape (asperity). (No unit.)	$Ssk = \frac{1}{Sq^3} \frac{1}{A} \iint_A z^3(x,y) dxdy.$
Sdq {n}, Sdr {n} Areal roughness hybrid parameters. To enable these parameters, check **Use Hybrid Parameters**. Sdq is the root mean square gradient. (No unit.) Sdr is the developed interfacial area ratio, as a percentage.	
Sk {n}, Spk {n}, Svk {n}, Smr1 {n}, Smr2 {n} Areal parameter for stratified surfaces.	

Measurement	Formula
Sk is the core roughness depth, in µm. Spk is the reduced summit height, in µm. Svk is the reduced valley depth, in µm. Smr1 is the upper bearing area, as a percentage. Smr2 is the lower bearing area, as a percentage.	

To add the following parameters in the tool's **Outputs** section, check **Use Hybrid Parameters**. The parameter outputs are enabled by default.

Measurements (Hybrid Parameters)

Measurement	Formula
Sdq {n}, Sdr {n} Areal roughness hybrid parameters. To enable these parameters, check **Use Hybrid Parameters**. Sdq is the root mean square gradient. (No unit.) Sdr is the developed interfacial area ratio, as a percentage.	

To add the following parameters in the tool's **Outputs** section, check **Use Stratified Surfaces**. The parameter outputs are enabled by default.

Measurements (Areal Parameters)

Measurement	Formula
Sk {n}, Spk {n}, Svk {n}, Smr1 {n}, Smr2 {n} Areal parameters for stratified surfaces. Sk is the core roughness depth, in µm. Spk is the reduced summit height, in µm. Svk is the reduced valley depth, in µm. Smr1 is the upper bearing area, as a percentage. Smr2 is the lower bearing area, as a percentage.	

To add the following parameters in the tool's **Outputs** section, check **Use Spatial Parameters**. The parameter outputs are enabled by default.

Measurements (Areal Parameters)

Measurement	Formula
Sal {n}, Str {n} Spatial parameters. Sal is the auto-correlation length, in µm. Str is the texture aspect ratio. (No unit.)	

Data

Type	Description
Extracted Surface	Surface data after filling gaps.
Form Removed Surface	Surface data after subtracting the polynomial pattern.
Roughness Surface	Surface data after a high pass filtering.

Calibration Procedure

1. Deactivate "Enabled Processing" to get into the editing mode.
2. Use "Check Parameters" in the "Working Mode" option to check the last calibration.
3. Use Reset Parameters to ensure that all parameters from the last calibration are reset. If no correction is used, this action can also be used.
4. Change the "Working Mode" to "Mirror calibration".
5. Editing position and size of the region and other settings.
6. Use the input data measured with a mirror.
7. Activate "Enabled Processing" to start the mirror calibration.
8. Repeat the measurement as necessary. The tool gives the summed up mean value of background noise and the recorded sample number after each measurement. 25 measurements is recommended, but there is no upper limit to the number of measurements.
9. Deactivate "Enabled Processing" to get into the editing mode.
10. Change the "Working Mode" to "Roughness calibration".
11. Enter the nominal values of Sa and Sq of the calibration target. Depending on requirements, Sa, Sq can be calculated separately or together, (since the two correlate very well, a mean value of the correction factor can be formed if the two are activated).
12. Editing position and size of the region and other settings.
13. Use the input data measured with a roughness calibration target.
14. Activate "Enabled Processing" to start the roughness calibration.
15. Repeat the measurement as necessary. The tool gives the summed up mean value of correction factor and the recorded sample number after each measurement. 25 measurements is recommended, but there is no upper limit to the number of measurements.
16. It is assumed that a job file is created after the calibration, so the information is preserved in the job file.

Roughness Calculation Procedure

1. Deactivate "Enabled Processing" to get into the editing mode.
2. Select the number of regions to be calculated, currently up to 20 - 2 regions are available for selection (the last 2 are reserved for calibration).
3. Configure individual regions. The default settings correspond roughly to the settings when using Mountains Map, so only the region should be set. Otherwise, all available configurations for each region should be set according to specification.

4. Activate the measurements and output surfaces.

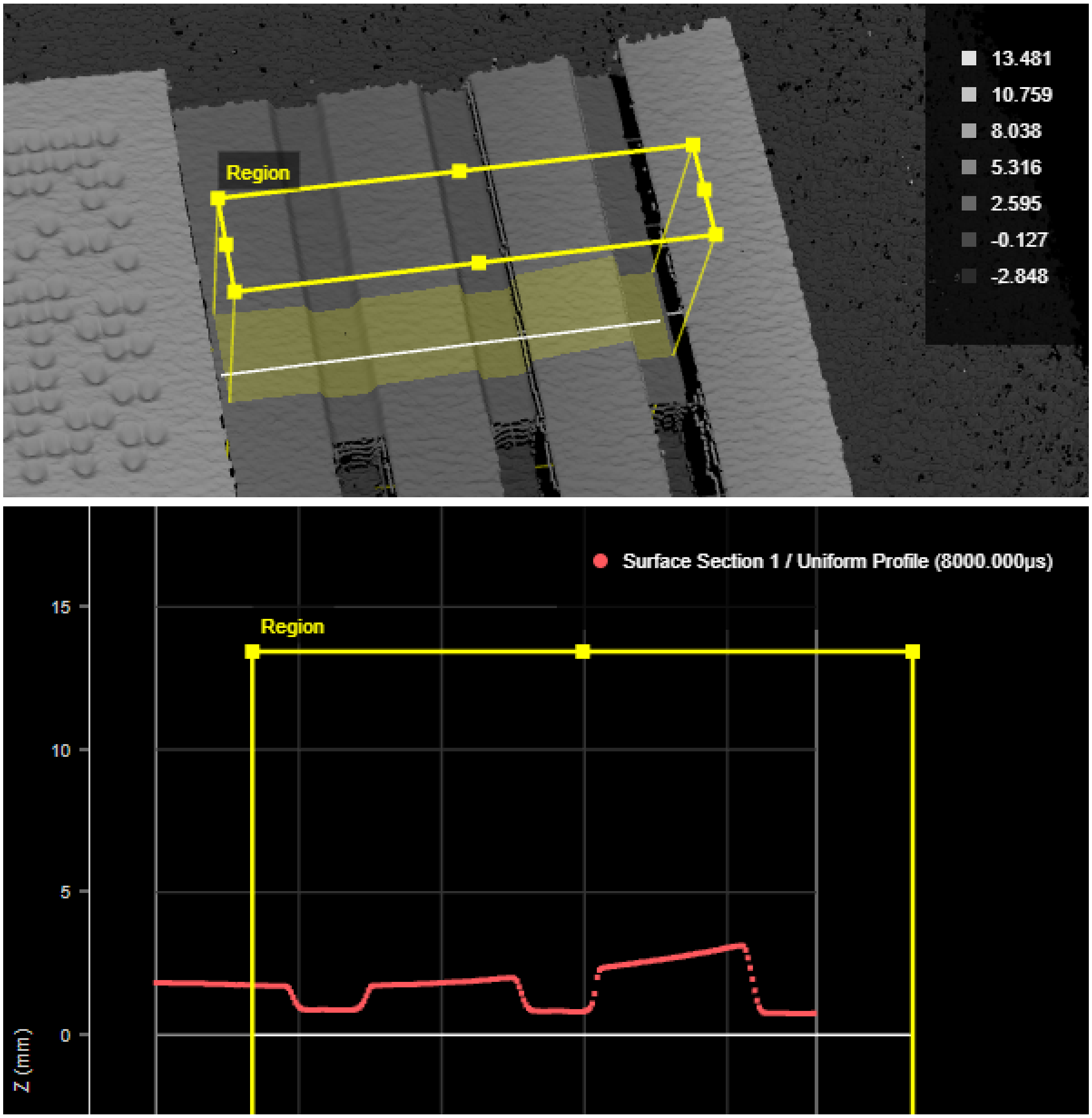
5. Activate "Enabled Processing" to start the calculation.

Surface Section

The Surface Section tool lets you define a line on a surface (a "section") from which the tool extracts a profile. You can then apply any Profile tool to the resulting profile (see *Profile Measurement* on page 284). A section can have any XY orientation on the surface. The resulting profile is on the XZ plane.

Using sections and the profile measurements, you can therefore use measurements that are not otherwise possible in Surface mode, for example:

- Gap and flush measurements
- Surface radius measurements (for example, rounded edges or corners)
- Intersections
- Point-to-point dimension measurements between profile features



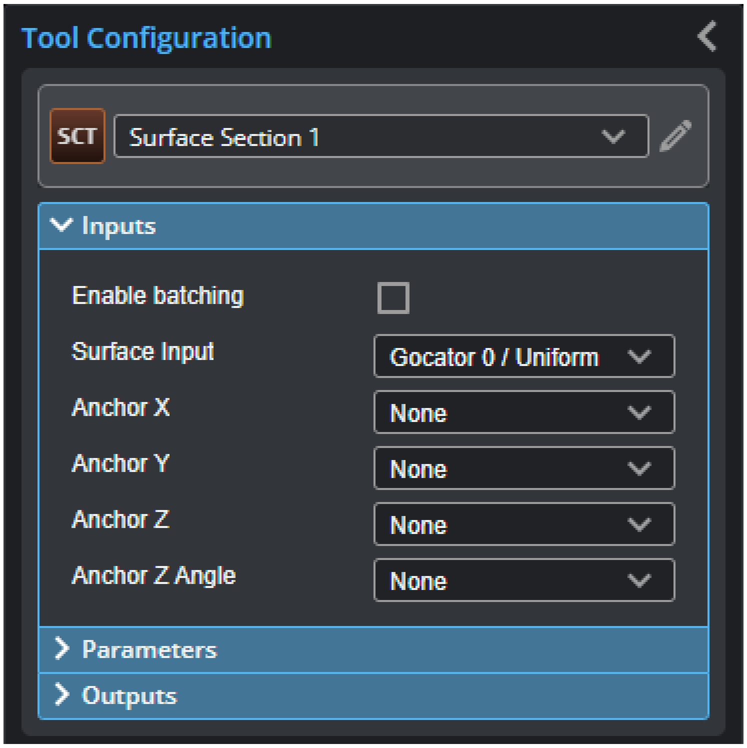
The resulting profile. The leftmost point of the profile is at the X origin. The region is positioned showing the X and Z position of the center of the region on the surface.

You can anchor the tool to some other easily identifiable feature on the scan target, which "shifts" the section in relation to that feature in case the target moves. This increases repeatability. For more information, see *Measurement Anchoring* on page 264.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

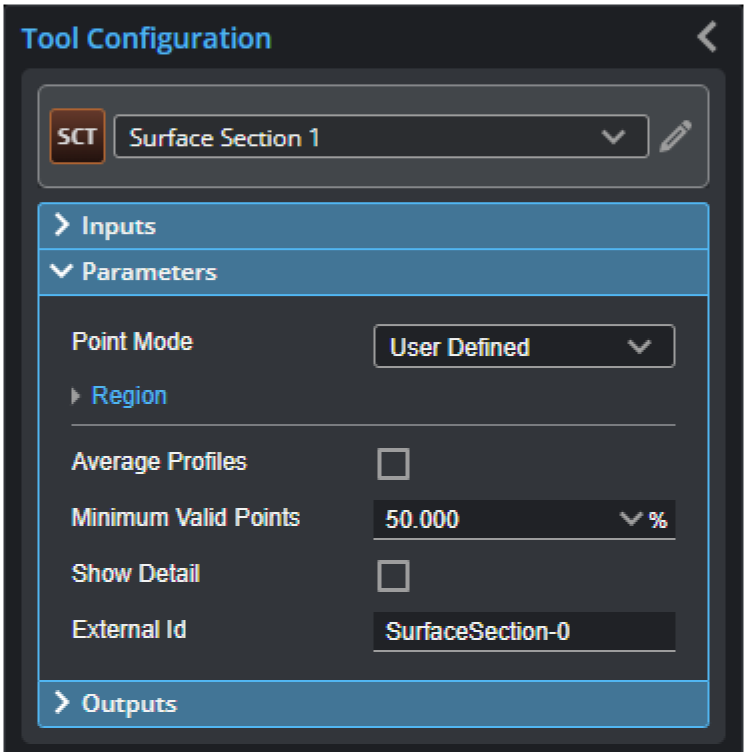
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Name	Description
Point 1 Point 2	The Point geometric features the tool uses to define the end points of the section. Only displayed if **Point Mode** is set to Feature Points.

Parameters

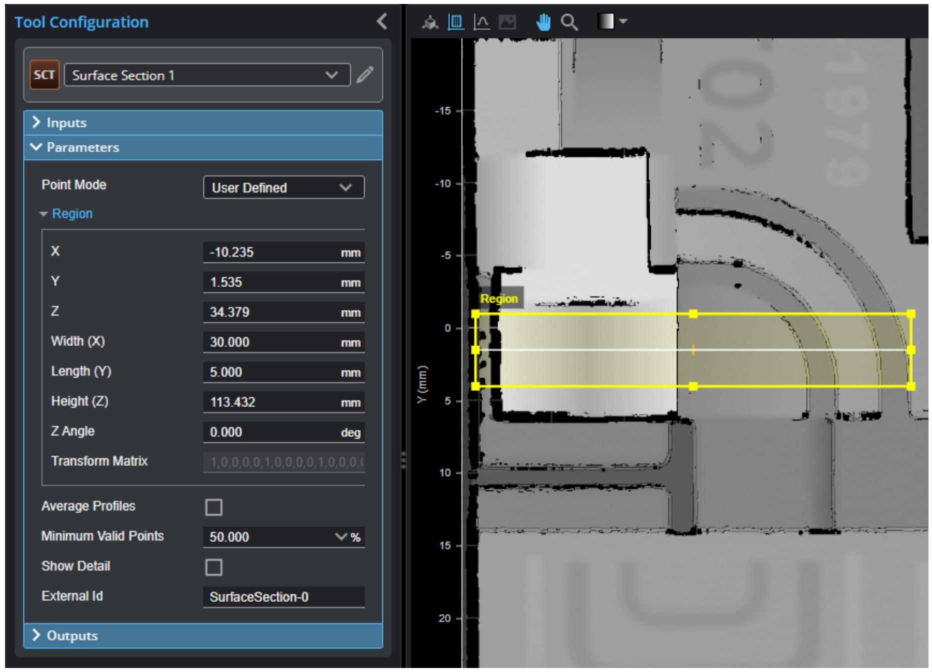
You configure the tool's parameters in the expandable **Parameters** section.



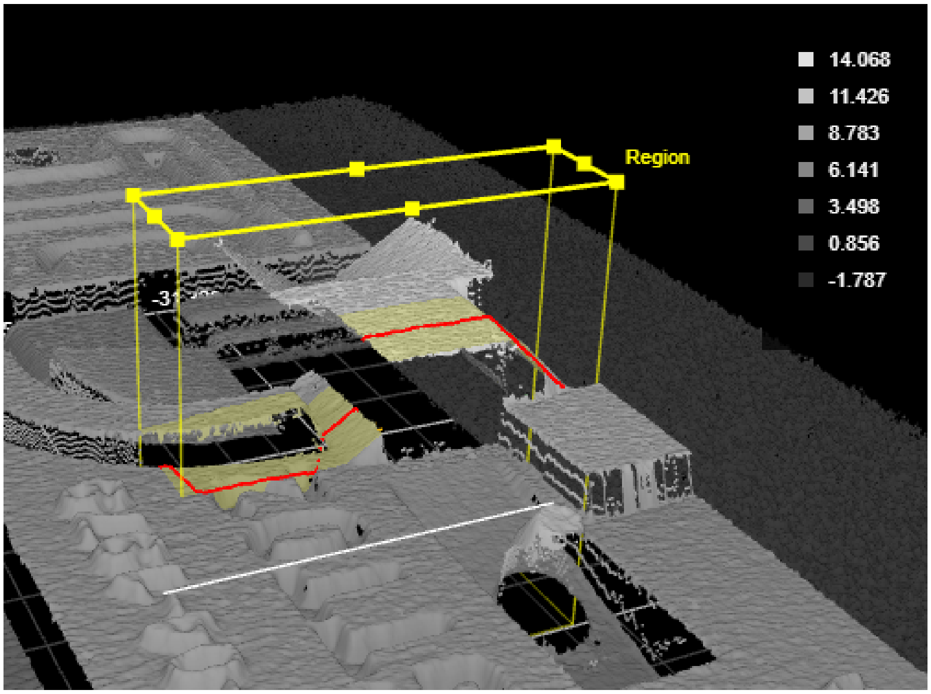
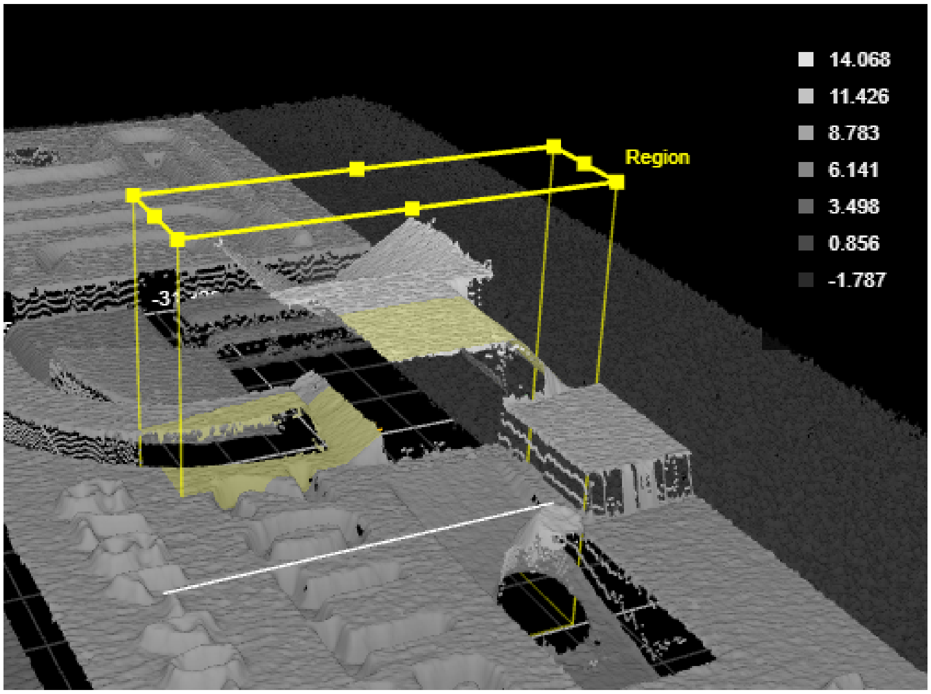
Parameters

Parameter	Description
Point Mode	Selects how the tool defines the end points of the section. One of the following: **User-Defined**: Enables a region (see **Region** below) whose size, position, and rotation you define. The section is placed along the X axis in the middle of the region. **Feature Points**: Uses two Point geometric feature inputs as the end points of the section.
Region	A rectangular region to define the section to extract. By default, the section is taken along the X axis of the region. The length of the region (along the Y axis) is the averaging width for the section. In the following, the region is set to a length of 5 mm and a width of 30 mm. The tool will therefore use a 5 mm averaging window, perpendicular to the X axis (width) of the region, to calculate the data point using its neighbor. If **Average Profiles** is enabled, the tool averages the points along the entire length of the

Parameter	Description
	region.

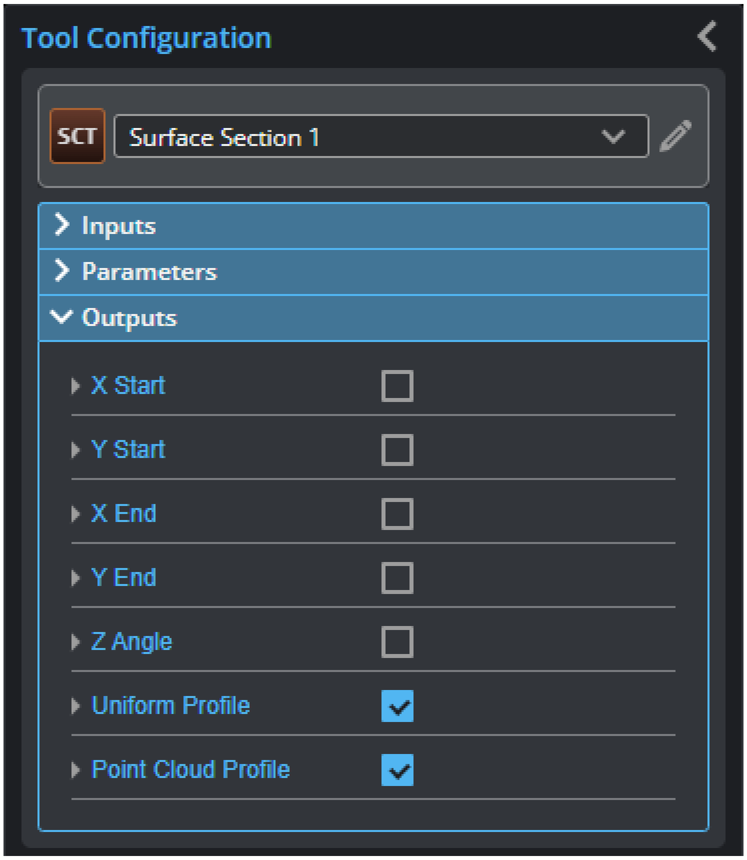


Parameter	Description
	For more information, see *Regions* on page 250.
Average Profiles	When enabled, averages the profiles in the region to produce the section profile. Otherwise, the tool only uses the data points under the line running parallel to the X axis in the middle of the region.
Minimum Valid Points	When **Average Profiles** is enabled, the minimum percentage of neighboring points across the averaging width (perpendicular to the section) that need to be valid for a point to be output on the resulting profile.
Show Detail	Determines whether data points (in red) are displayed under the section in the data viewer. If this setting is disabled, only the line representing the defined section is displayed. **Show Details** enabled:

Parameter	Description
	 **Show Details** disabled: 
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

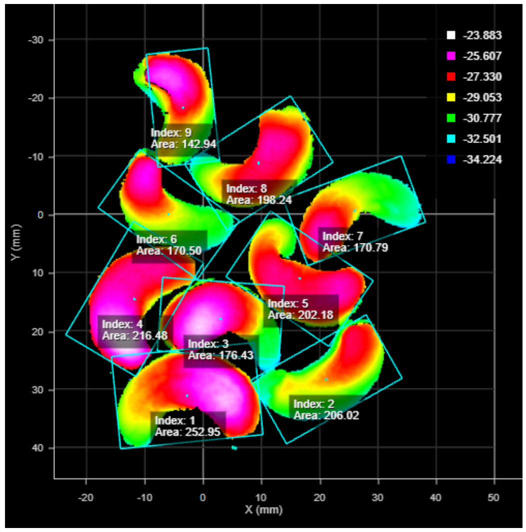
Measurement
X Start **Y Start** These measurements return the X and Y position of the start of the section, respectively.
X End **Y End** These measurements return the X and Y position of the end of the section, respectively.
Z Angle Returns the rotation of the section around the Z axis.

Data

Type	Description
Uniform Profile Point Cloud Profile	The profile (uniform or point cloud, respectively) that the tool extracts from the surface. Available to profile tools for profile measurement.

Surface Segmentation

The Segmentation tool separates surface data into "segments," based on the tool's parameters. Segments can be touching and overlapping to a certain degree. The Segmentation tool is especially useful in the food industry, for example to identify food items that are too small or too big, or items that are damaged.



For each segment, the tool returns the X and Y position of the center of its bounding box, its length and width, and its area, as well as several more global measurements, such as maximum / minimum width or length, and so on. For a complete list, see below.

The Segmentation tool can also be used as a second stage of processing after part detection. For example, part detection could be used to detect a tray (containing parts), and the Segmentation tool could then separate the parts within the tray. For information on part detection, see *Profile Part Detection* on page 373.

> The Segmentation tool cannot handle large overlaps.

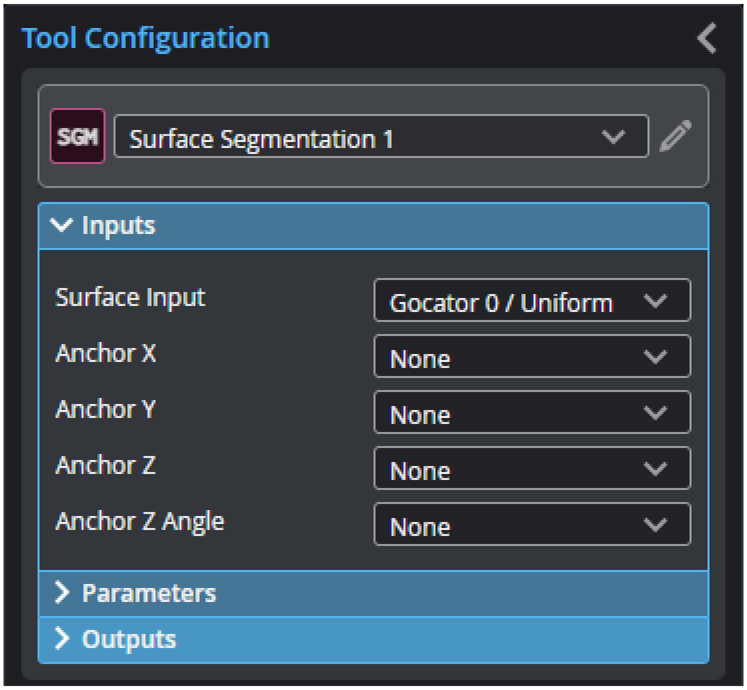
> The Segmentation tool does not perform template matching.

> To reduce processing time, consider using the decimation filter. For more information on this filter, see *Surface Filter* on page 552.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



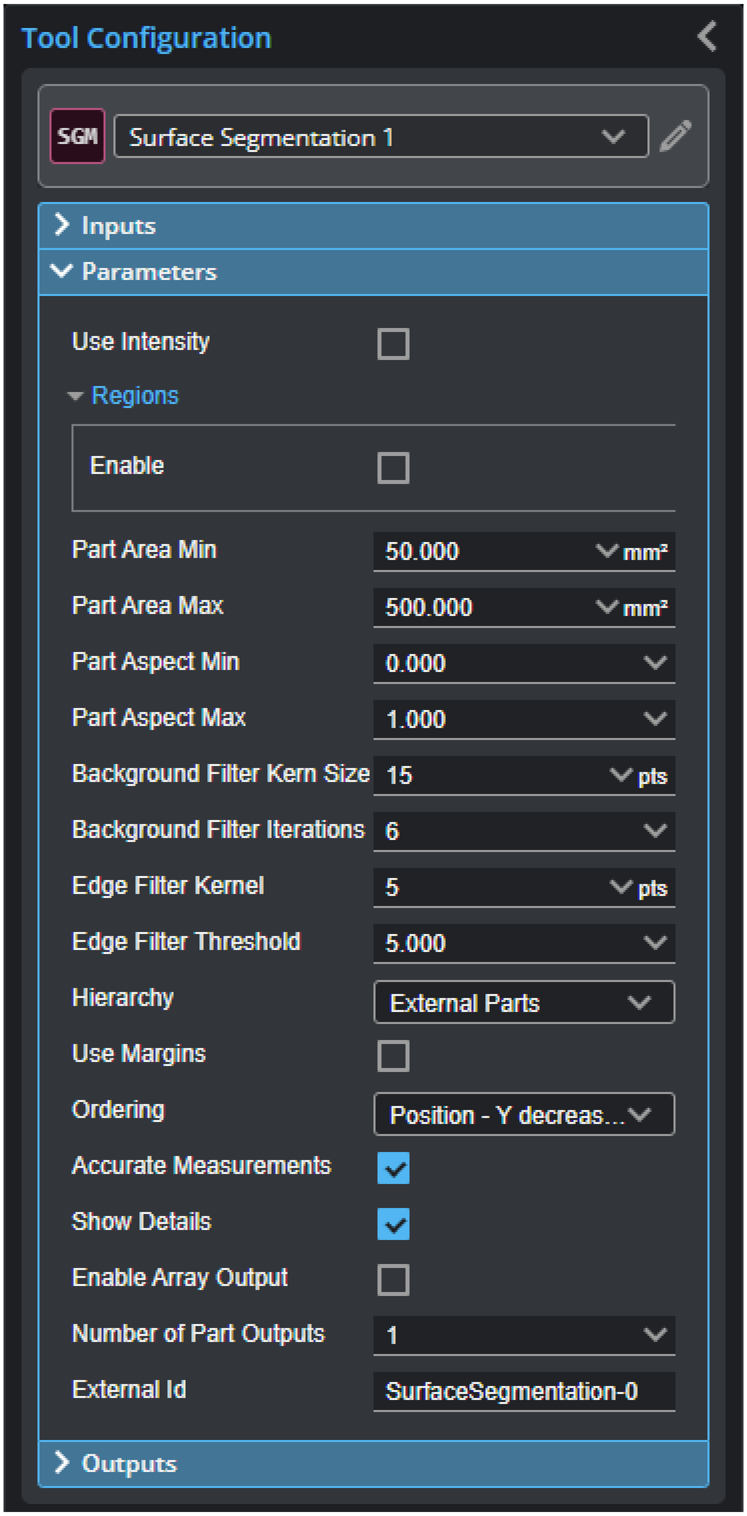
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

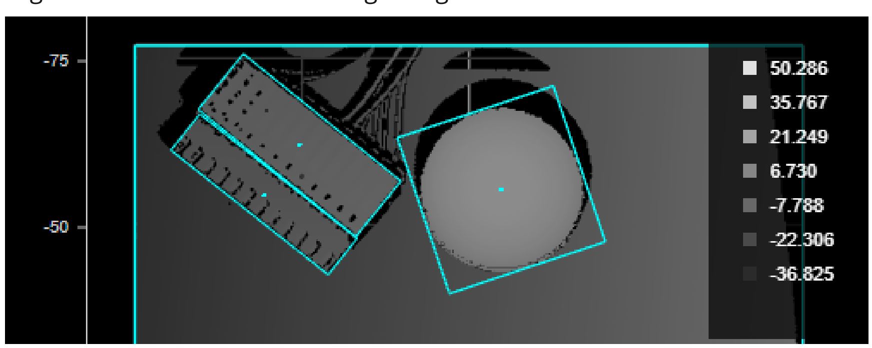
Parameters

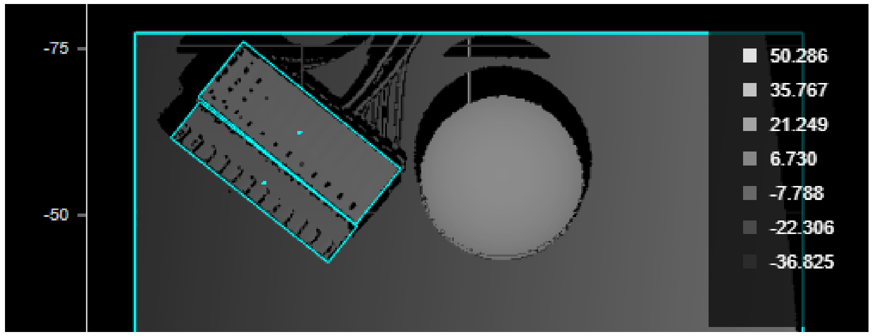
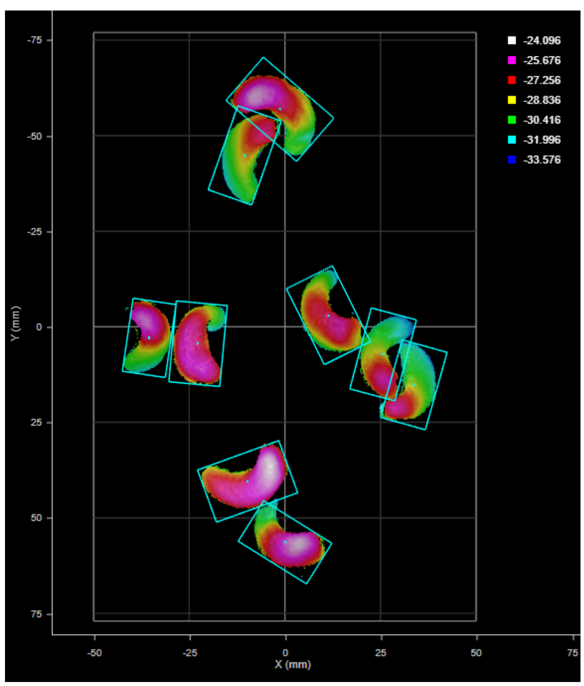
You configure the tool's parameters in the expandable **Parameters** section.

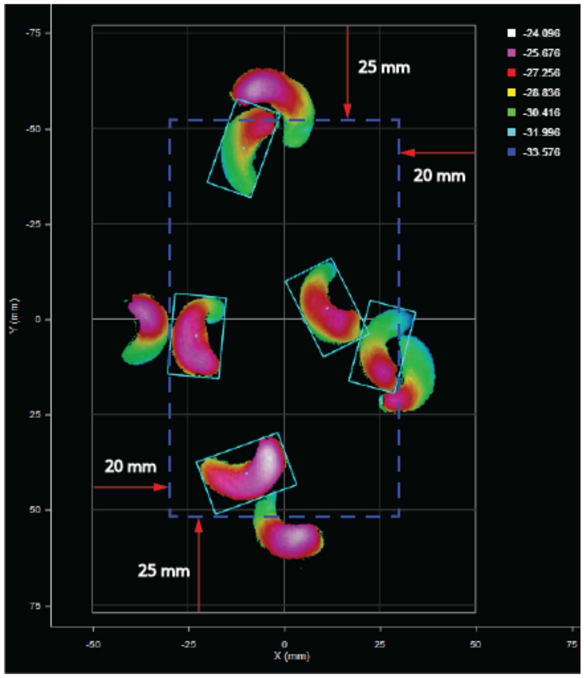


Parameters

Parameter	Description
Use Intensity	Causes the tool to use intensity. **Acquire intensity** must be enabled on the **Scan**

Parameter	Description
	page; otherwise, the tool will display an error.
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Part Area Min Part Area Max	The minimum and maximum areas in square millimeters for a part of the scan data to be identified as a segment.
Part Aspect Min Part Aspect Max	The minimum and maximum aspect ratios (minimum axis length in mm) / (maximum axis length in mm) of the best fit ellipse to the segment contour points for a segment to qualify to be added to the list of found segments.
Background Filter Kern Size Background Filter Iterations	These settings perform background separation. The greater each of these values is, the more separation will be achieved. You must find a balance that removes noise adequately without degrading the segment find quality.
Edge Filter Kernel	Use this value to adjust the “granularity” of the part edge detection.
Edge Filter Threshold	Controls the separation of the parts, increasing the gap between the parts so that they can be detected more easily.
Hierarchy	Use this setting to detect segments when they are surrounded by background data. Choose one of the following: **All Parts** or **External Objects**. **All Objects** This option lets you segment parts with surrounding background data. For example, in the following image, with **All Parts** selected, the sphere is correctly segmented from the surrounding background.  Note that this option may result in "over-segmentation": the tool may segment a part into two segments. **External Objects**

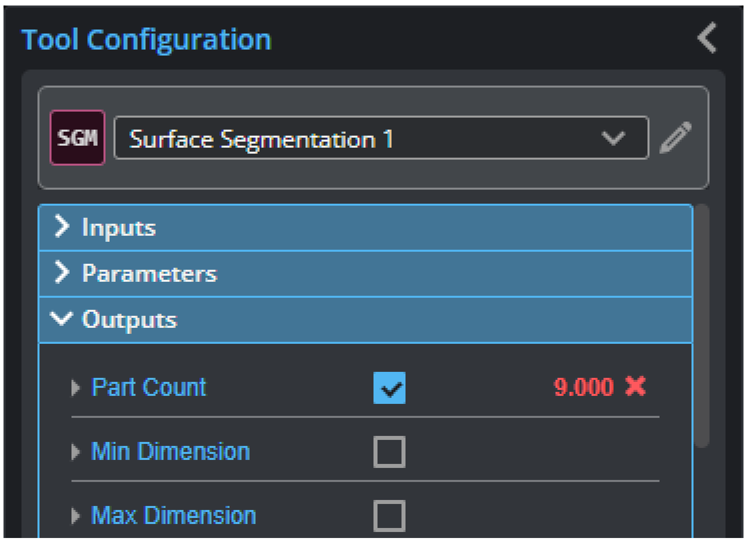
Parameter	Description
	In the following image, the sphere is not identified as a segment because of the surrounding background. It is treated as part of a large segment that includes all of the scan data.  **Holes** Use this option to get holes inside objects. **All Contours** Use this option to get all detected contours.
Use Margins Margins Left Right Top Bottom	When **Use Margins** is enabled, the tool discards parts whose center point is within the margins defined in the expandable Margins section. Margins extend inwards from the outer edges of the sensor's FOV (XY scan area) and are positive. In the following, with margins set to 0, nine parts are detected 

Parameter	Description
	In the following, the left and right margins are set to 20 mm, and the top and bottom margins are set to 25 mm. The sensor's FOV is 100 x 154 mm. The tool discards the four parts whose center points are within those margins. (Total part count is reduced to five.) 
Ordering	Orders the measurements, features, and surface data of the individual parts output by the tool. Choose one of the following: • Area - Large to small • Area - Small to large • Position - X increasing • Position - X decreasing • Position - Y increasing • Position - Y decreasing • Position - Z increasing • Position - Z decreasing
Accurate Measurements	Returns more accurate width, height, and area measurements, as well as a better Surface output, but the trade-off is greater processing time. If you only need center points or the number of parts, disable this parameter for faster processing..
Show Details	Toggles whether the tool displays the index and area of each individual part. Enabling this will increase processing time.
Enable Array Output	When **Enable Array Output** is disabled, the tool outputs each part as an individual Surface output. When **Enable Array Output** is enabled, the tool outputs parts in arrays. Be sure to enable the Part Surface in the **Outputs** section after enabling this parameter. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and*

Parameter	Description
	Aggregation on page 242.
Number of Part Outputs	Determines the number of parts the tool outputs as measurements, features (center points of parts), and surface data. Currently limited to 200 parts. If **Enable Array Output** is checked, this parameter is hidden and all parts are output in arrays.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

> When **Enable Array Output** is unchecked, individual measurement and feature outputs are produced for each part, for example, Max Z 1, Max Z 2, and so on, or Center 1, Center 2, and so on. When this parameter is checked, these measurements and features are output as arrays.

Measurements

Measurement
Part Count Returns the total number of segments identified, based on the tool's parameters.
Min Dimension **Max Dimension** The minimum and maximum dimensions among all of the identified segments, respectively.
Mean Width **Mean Length**

Measurement
The mean width and length of the segments, respectively.
Min Area **Max Area** The minimum and maximum area among all of the identified segments, respectively.
Sum Area The sum of the areas of the segments.
Mean Area The mean area of the segments.
Min Height **Max Height** **Mean Height** The minimum, maximum, and mean heights among all of the identified segments, respecitvely.
X Center {n} **Y Center {n}** The X and Y positions of the center of a part segmented from the surface.
Width {n} **Length {n}** The width and length of a part segmented from the surface. These are always the major and minor axis of a part, respectively.
Area {n} The area of a part segmented from the surface. The area is calculated using the contour of the part and resampling. For this reason, areas calculated using the Surface Volume tool will produce different measurements; for more information, see *Area* on page 691.

Features

Type	Description
Center {n}	The point representing the center of a segmented part.

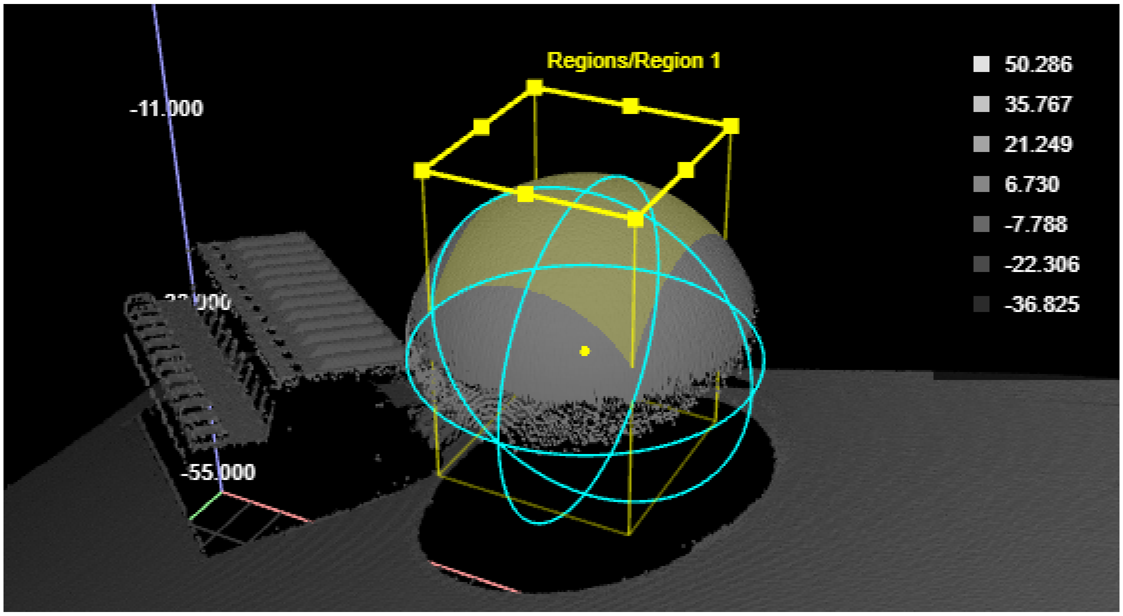
Data

Type	Description
Part Surface {n}	Surface data corresponding to each segmented part.
Diagnostic	Surface data you can use to evaluate the impact of the tool's kern size and iteration settings, which the tool uses to separate potential segments.

Surface Sphere

The Sphere tool lets you compute characteristics of a scanned sphere by specifying a region to inspect.

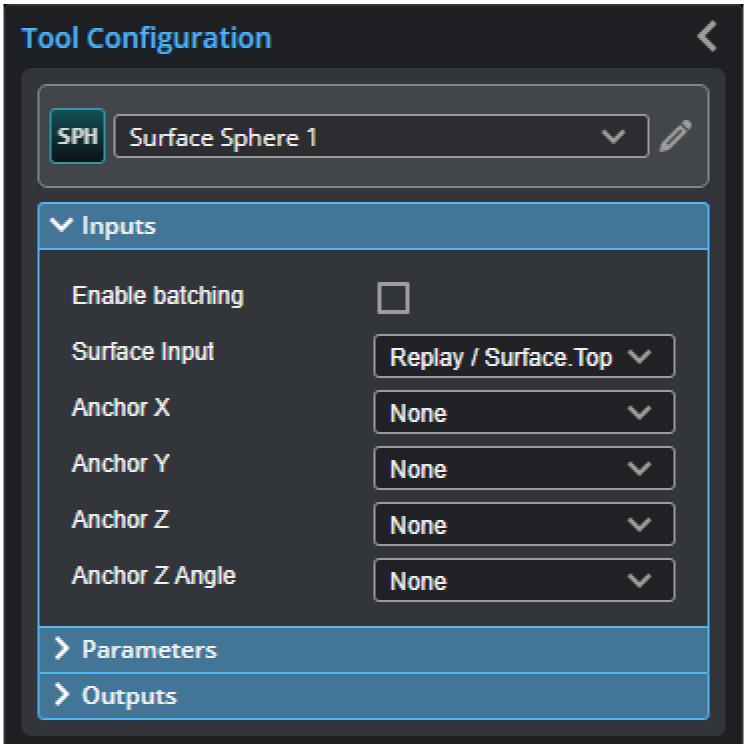
	For the tool to work properly, the tool's region must usually be enabled and properly placed to include only the sphere's scan data.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



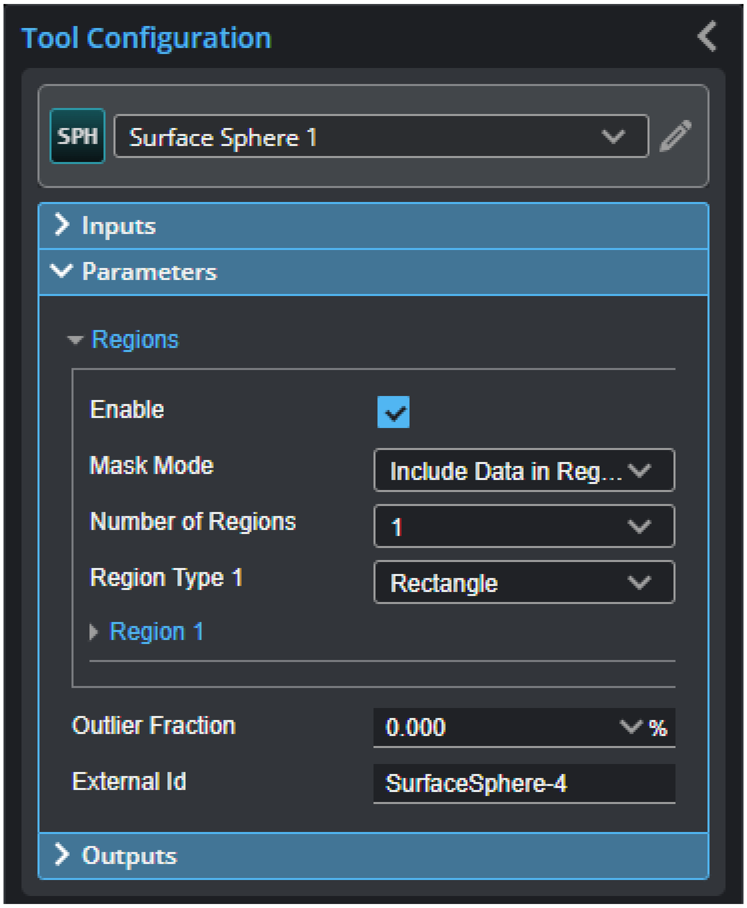
To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

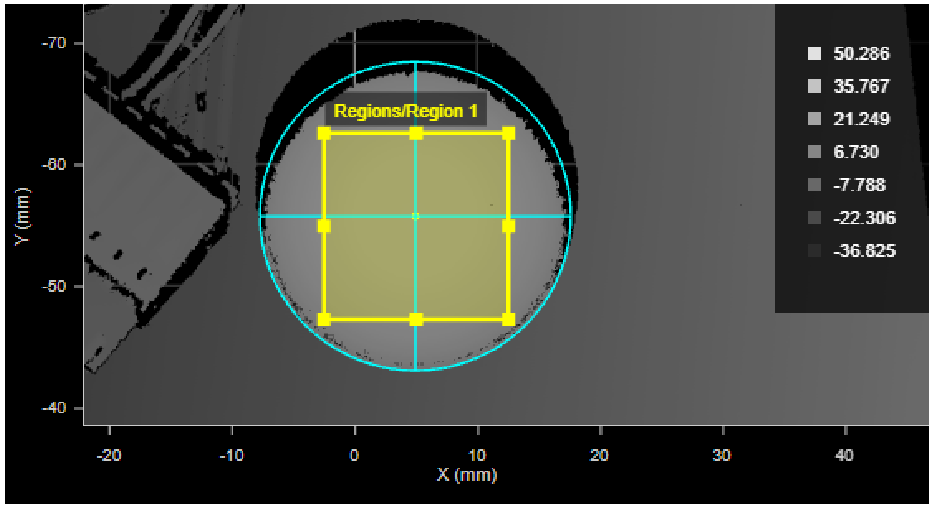
Parameters

You configure the tool's parameters in the expandable **Parameters** section.



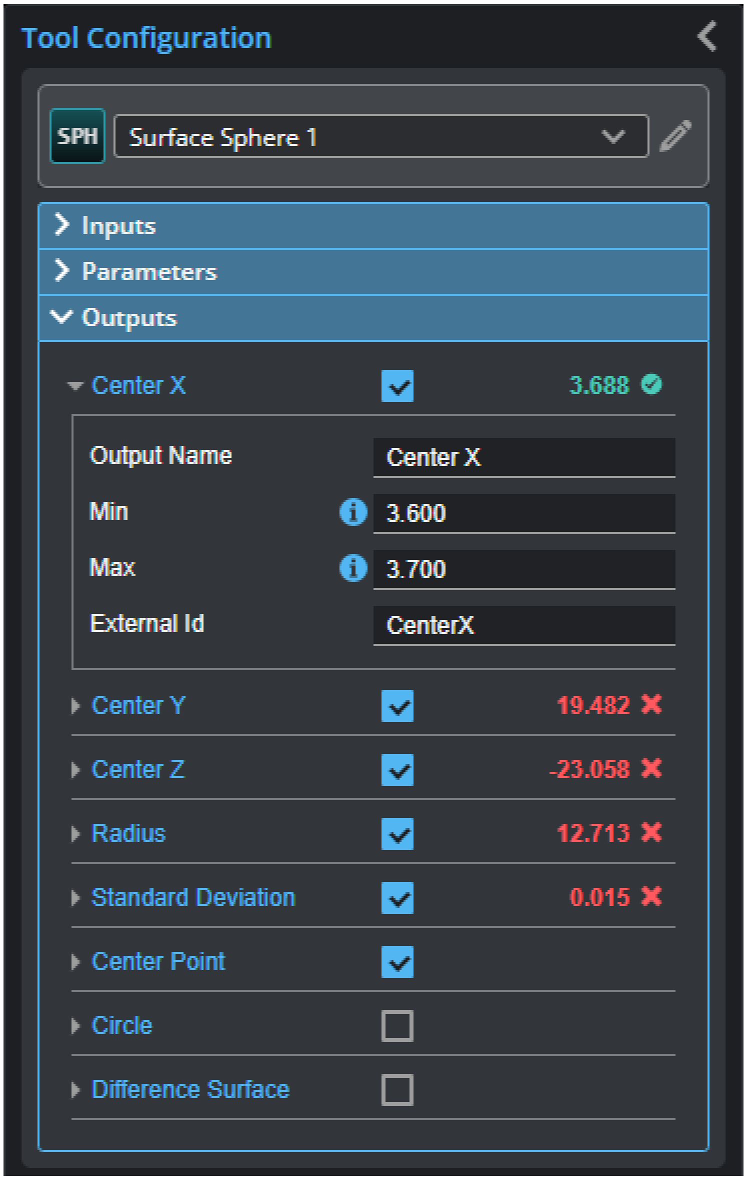
Parameters

Parameter	Description
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Number of Regions Region Type {n} Region {n}	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250. In order for the tool to correctly fit a sphere to the scan data, you must set the region so that it only contains data from the sphere on the target.

Parameter	Description
	
Outlier Fraction	The percentage of outlier points to exclude. Setting this to a small value can help the tool fit a better sphere when noise is present.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

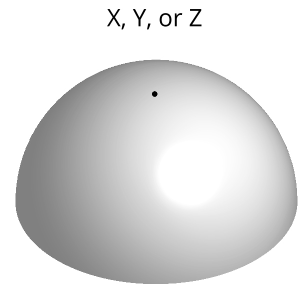
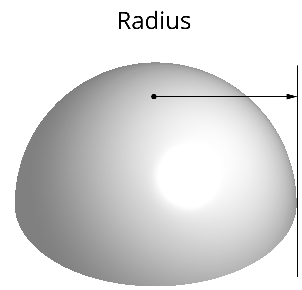
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Center X, **Center Y**, **Center Z** These measurements determine the X, Y, Z position of the center of the sphere, respectively.	
Radius Determines the radius of the sphere.	
Standard Deviation Determines the error of the points compared to the computed sphere. It is defined as the square root of the variance of the distance of every point to the computed sphere.	

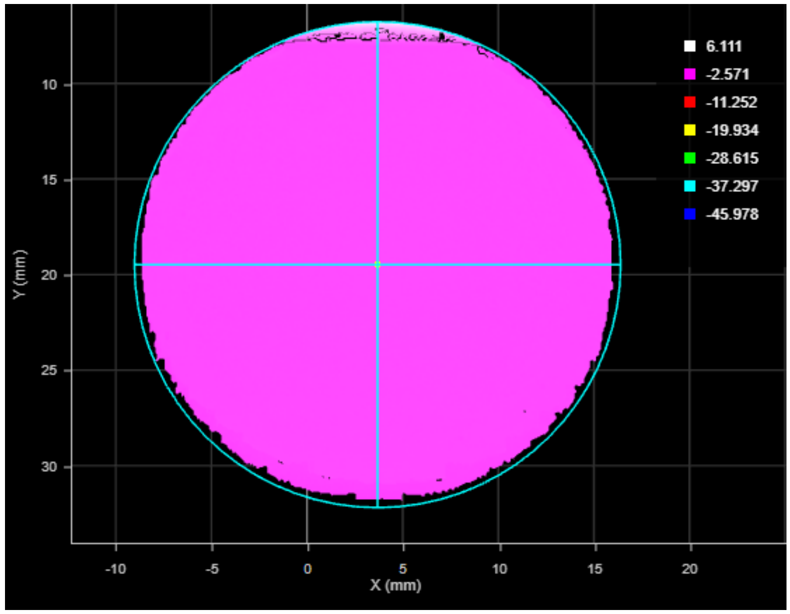
Features

Type	Description
Center Point	The center of the circle encompassing the widest part of the fitted sphere.
Circle	The circle encompassing the widest part of the fitted sphere.

For more information on geometric features, see *Geometric Features* on page 262.

Data

Type	Description
Difference Surface	Shows the fit error at each point in the height map.

Type	Description
	

Surface Stitch

The Stitch tool lets you combine up to 24 frames of scans into a single Surface scan. This lets you get a much larger scan volume with fewer sensors (either in a single sensor system or a multi-sensor system). For each scan, you can specify not only X, Y, and Z offsets (translations), but also X, Y, and Z angles (rotations), defining its relationship with the others. This means that when the sensor system is mounted to a robot, or if you are using, for example, an X-Y table, you can get a complete scan with fewer sensors. The resulting combined scan can then be used as input by any other Surface or Feature tool from its **Surface Input** drop-down.

The tool performs rotation first, and then translation.

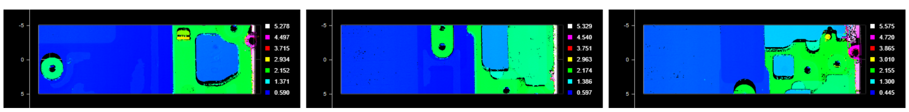
> The tool combines data simply by overwriting in sequence: it performs no averaging or blending. The tool also performs no fitting.

> Results are only as accurate as the motion system.

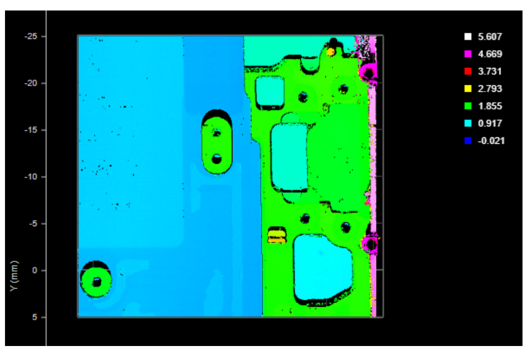
> Seams are often seen in combined data in stitching performed in anything other than along the Y axis.

The tool returns one measurement, which simply indicates the number of scans successfully added to the combined scan data.

The following shows three individual frames.

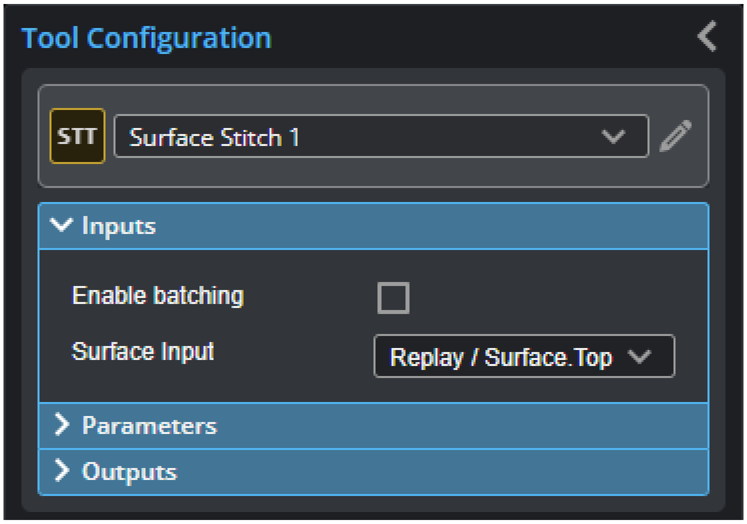


In the following, the tool has combined the frames into a single surface.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

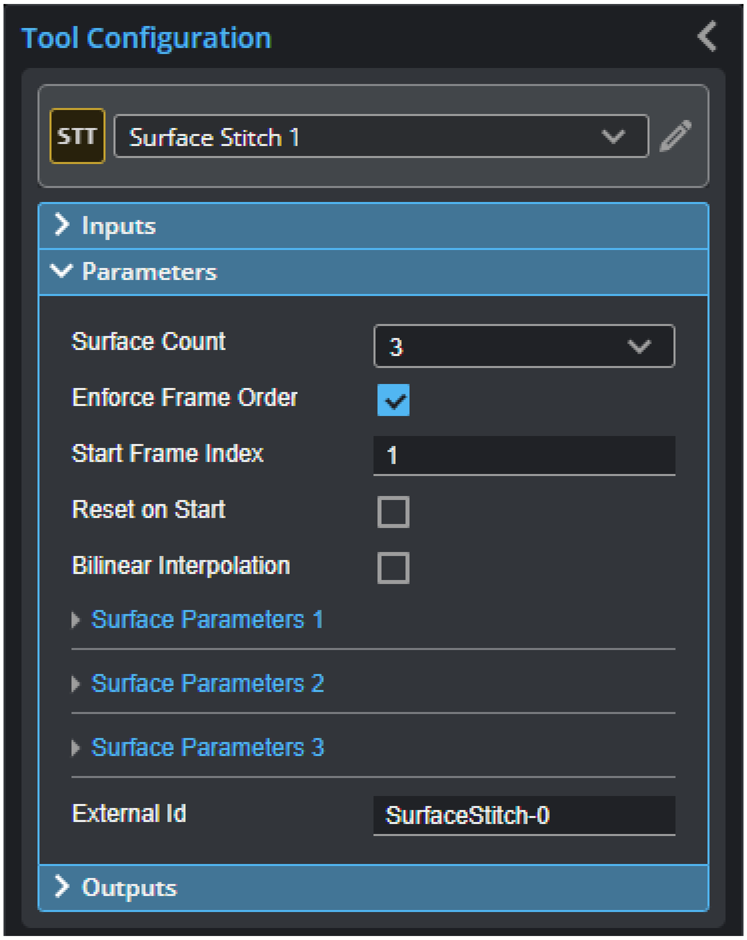


Inputs

Name	Description
Surface Input	The data the tool applies measurements to or processes.

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.

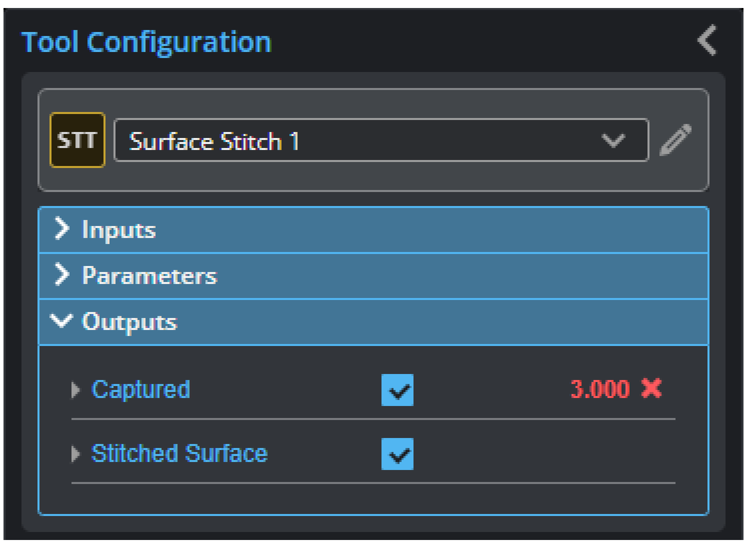


Parameters

Parameter	Description
Surface Count	The number of scans to combine into a single surface. For each, a "Surface Parameters" section is added. The tool accepts setting the number of scans to one: in this case it, behaves like a transform tool.
Enforce Frame Order	Restricts the stitching for specific frame indexes, starting at the frame indicated in **Start Frame Index**. If unchecked, an **Operation** drop-down is displayed. This setting is disabled if you attempt to stitch data from individual scans acquired using the Snapshot button (that is, all frame indexes are at 1).
Start Frame Index	The start frame index.
Operation	If **Enforce Frame Order** is disabled, the **Operation** drop-down is displayed. One of the following: • **Normal**: The tool automatically chooses this operation after you have chosen another operation. • **Reset buffers**: Resets the buffers used to stitch frames. • **Lock**: Lets you lock the current processing and outputs of the tool. Useful when you need to add another tool that will use this tool's output (for example,

Parameter	Description
	a Surface Section tool). If you do not lock the tool, as soon as you add the other tool, the output is cleared, which means you must re-execute the combined output again to configure the additional tool. Be sure to unlock the tool after you have configured any other tools.
Reset On Start	Clears buffers for the stitched surface when the sensor is started. Useful for situations where the sensor is started and stopped frequently (to capture a small number of frames), rather than starting the sensor and letting it run for a long period. Enable this parameter to prevent data from a previous capture session being stitched with data from the current capture session.
Bilinear Interpolation	Evaluates the height of each transformed point (through translation or rotation) based on its neighbors. More precise, but has an impact on performance.
Surface Parameters {n}	For each scan to be added to the combined surface scan, an expandable **Surface Parameters** section is added. The following settings are available: • X, Y, and Z Offset • X, Y, and Z Angle • Mirror: Mirrors input Surface {n}.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Captured Indicates the number of scans successfully added to the combined surface scan.

Only one of the following data types will contain data, depending on whether **Uniform Spacing** is enabled. For more information, see *Scan Modes and Intensity* on page 195.

Data

Type	Description
Stitched Surface Stitched Raw Surface	The stitched surface scan.

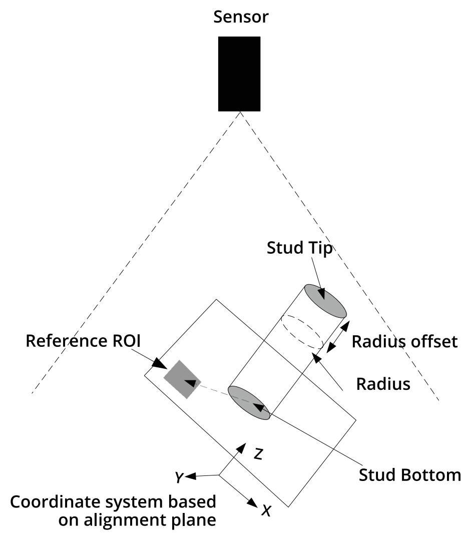
Surface Stud

The Stud tool measures the location and radius of a stud.

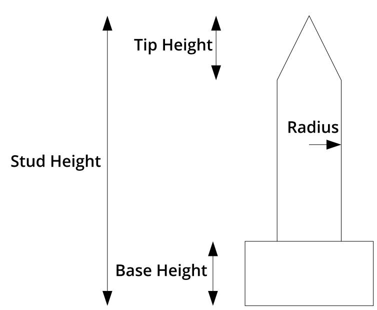
	The tool does not search for or detect the feature. The tool expects that the feature, conforming reasonably well to the defined parameters, is present and that it is on a sufficiently uniform background.

The tool uses a complex feature-locating algorithm to find a hold and then return measurements. For a detailed explanation of the algorithm, see *Stud Algorithm* on the next page.

The location of the stud is defined at either the stud tip or the stud base. The tip is the intersection of the stud axis and the top of the stud; the base is the intersection of the stud axis and the surrounding plane.



The stud shape is defined by the tip height and base height. The base and tip heights specify where the shaft with the nominal radius begins and ends.



Measurement Panel

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Stud Algorithm

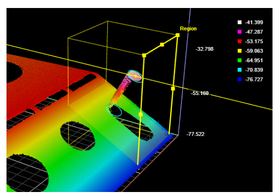
The Stud algorithm measures the stud in three steps: searching for the tip, finding the reference plane, and shaft fitting. Note that the tip and the side of the stud must be within the measurement region.

See the tool's parameters for an explanation of the options that affect the tool's algorithm.

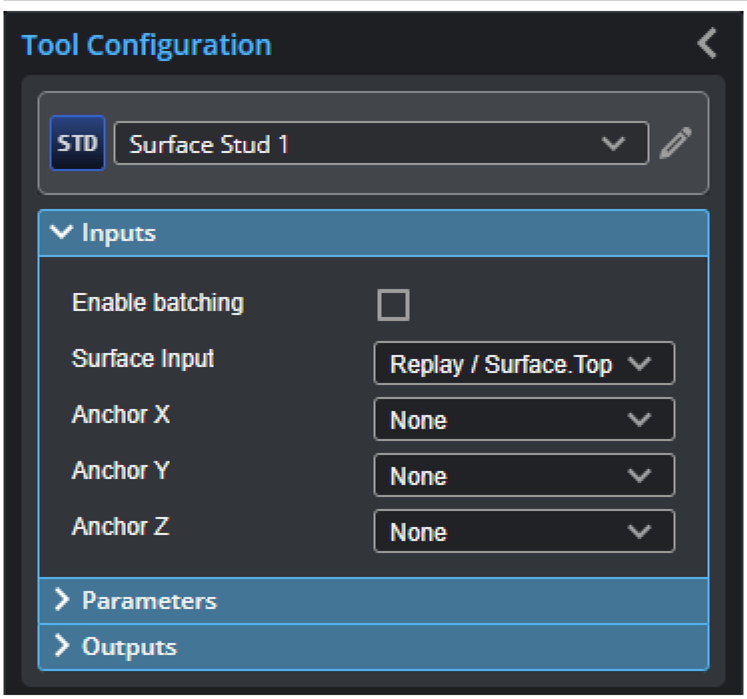
Searching for the tip - The algorithm looks for the approximate location of the tip. If Auto-Tilt is enabled, the algorithm uses the flat surface around the tip to estimate the orientations of the part. The approximate tip is the location of the highest (maximum Z) pixel after correction for the nominal tilt angle.

Finding the reference plane - The reference regions are positioned using the approximate tip, the nominal angle values, and the nominal stud length. Compared to the hole/opening, misplaced stud reference regions are more likely to cause a failure to produce any measurement.

Shaft fitting - The shaft region is determined based on the approximate tip position, the nominal angles, the reference plane position, and the stud nominal size parameters. Shaft fitting is successful if the algorithm can fit at least three circles with the stud diameter along the shaft. Fitting each circle requires sufficient data along the top portion the shaft. Because of occlusions, the bottom of the shaft is often not visible to the sensor and the algorithm is designed to handle this situation.



Inputs



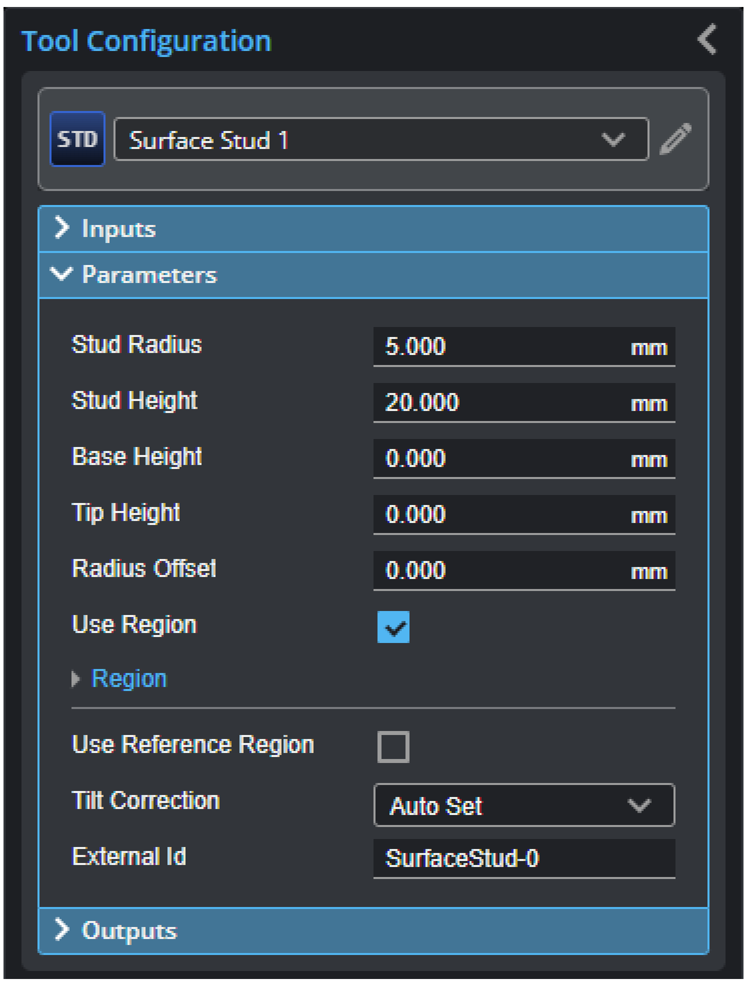
	To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Anchor X Anchor Y Anchor Z	The X, Y, or Z measurement of another tool that this tool uses as a positional anchor. Positional anchors are optional.

Parameters

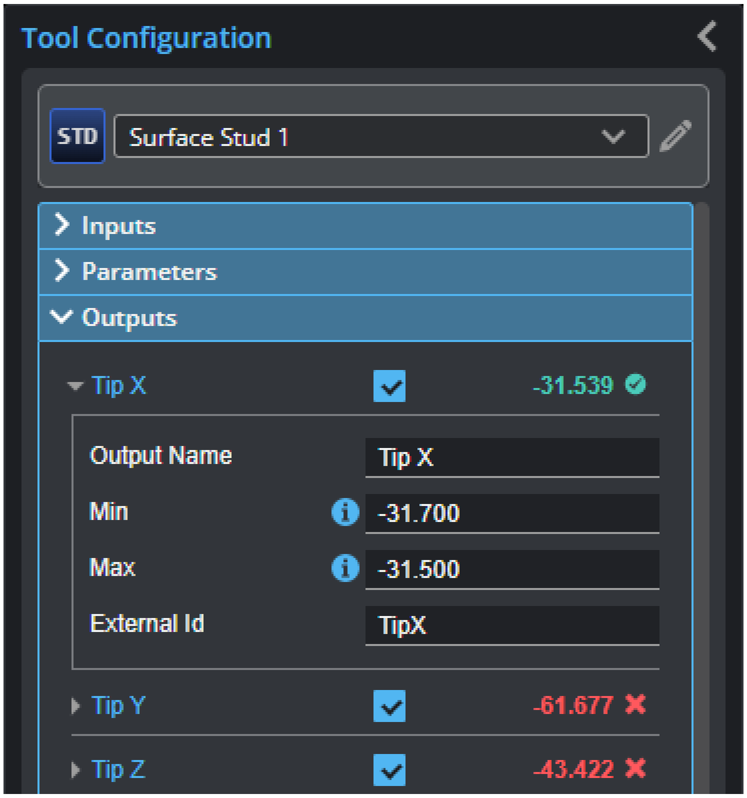
The following parameters are in the expandable **Parameters** section in the tool's configuration.



Parameters

Parameter	Description
Stud Radius	Expected radius of the stud.
Stud Height	Expected height/length of the stud.
Base Height	The height above the base surface that will be ignored when the (truncated) cone is fitted to the stud data.
Tip Height	The height from the top of the surface that will be ignored when the (truncated) cone is fitted to the stud data.
Radius Offset *(Radius measurement only)*	The distance from the tip of the stud from which the radius is measured.
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Use Reference Region	When enabled, displays the **Reference Type** setting.
Reference Type	The tool uses the reference regions to calculate the base plane of the stud. Reference regions are relative to the base of the stud.
Tilt Correction	Tilt of the target with respect to the alignment plane. **Autoset**: The tool automatically detects the tilt. The measurement region to cover more areas on the surface plane than other planes. **Custom**: You must enter the X and Y angles manually in the **X Angle** and **Y Angle** parameters (see below).
X Angle Y Angle	The X and Y angles you must specify when **Tilt Correction** is set to **Custom**. You can use the Surface Plane tool's X Angle and Y Angle measurements to get the angle of the surrounding surface, and then copy those measurement's values to the **X Angle** and **Y Angle** parameters of this tool. For more information, see Plane.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

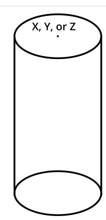
Outputs

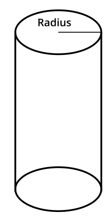


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Tip X Determines the X position of the stud tip.	
Tip Y Determines the Y position of the stud tip.	
Tip Z Determines the Z position of the stud tip.	
Shaft X Determines the X position of the stud shaft. The position is defined with the parameter Radius Offset.	
Shaft Y Determines the Y position of the stud shaft. The position is defined with the parameter Radius Offset.	
Shaft Z	

Measurement	Illustration
Determines the Z position of the stud shaft. The position is defined with the parameter Radius Offset.	
Base X Determines the X position of the stud base.	
Base Y Determines the Y position of the stud base.	
Base Z Determines the Z position of the stud base.	
Radius Determines the radius of the stud.	

Features

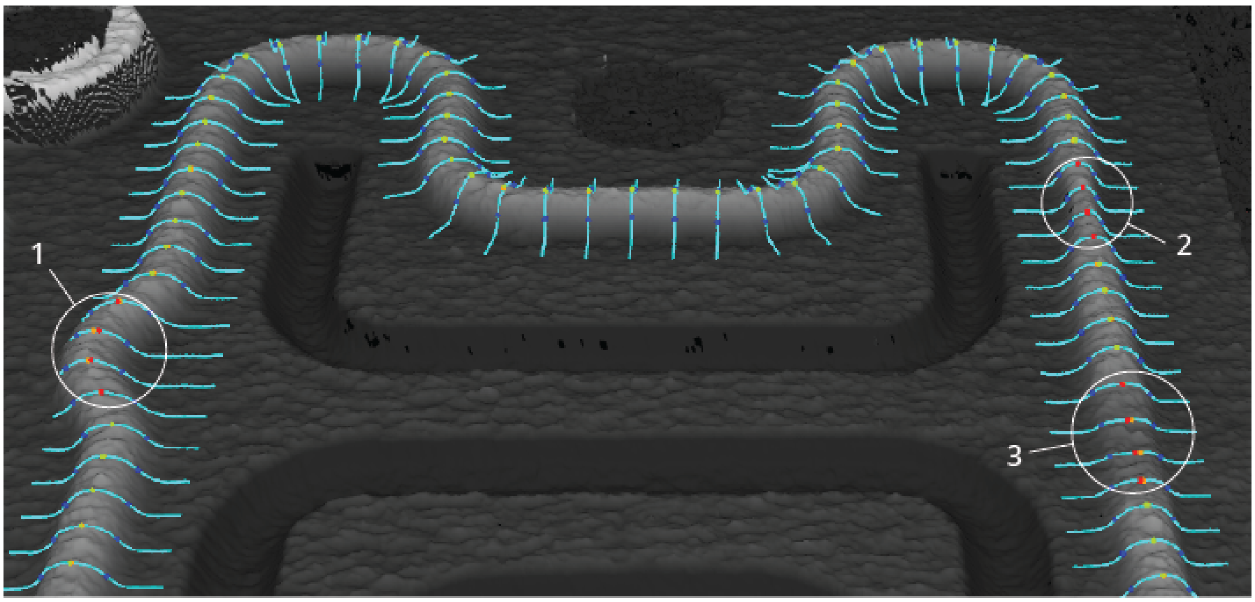
Type	Description
Tip Point	The center point of the stud tip.
Shaft Point	The center point of the stud shaft.
Base Point	The center point of the stud base.

For more information on geometric features, see *Geometric Features* on page 262.

Surface Track

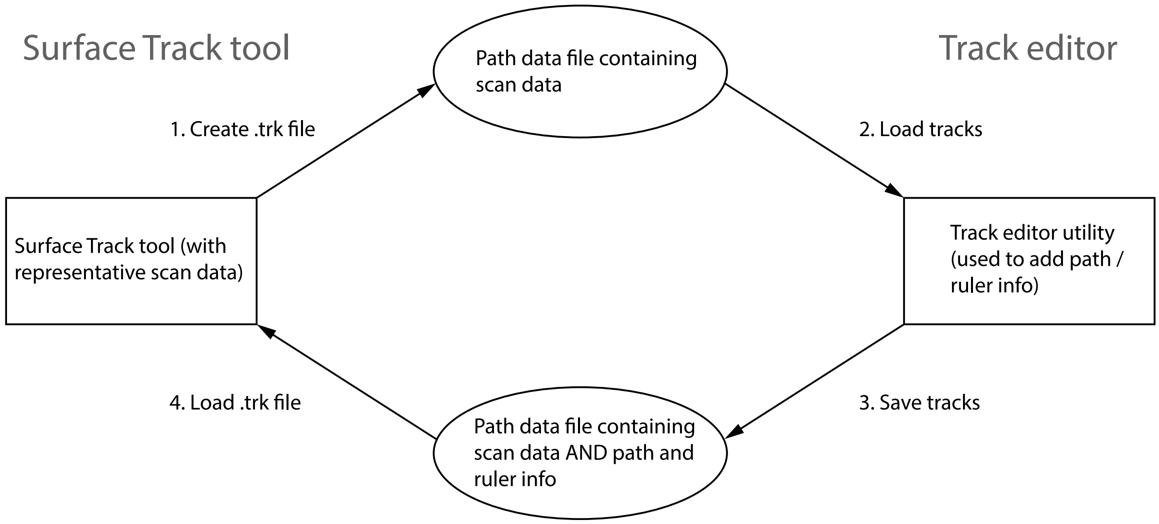
The Track tool lets you perform quality control and inspection along a path you define on representative scan data. The Track tool is especially useful for inspecting materials such as glue / sealant beads. The tool returns width and height measurements of the material, as well as OK and NG ("no good") counts, which let you monitor material overflow and breaks. A major advantage of the tool is that it removes the need to configure individual tools for each location along the path. You can use point and line geometric features to anchor the tool (for more information on geometric features, see *Geometric Features* on page 262).

> Gocator sensors have a limited amount of space for storing path files. For this reason, when working with large datasets, we recommend that you run the Track tool through a PC instance of GoPxL. For more information, see *Running GoPxL on a Windows PC* on page 828.



	Element	Description
1	Offset fails	The center points in this area of the track are too far off the defined path.
2	Width fails	The center points in this area of the track are too narrow.
3	Height failes	The center points in this area of the track are too low. (Note how the profiles are flatter than other profiles.)

You define the path along which the tool performs its measurements using a separate, PC-based utility (the "track editor"). The following shows the relationship between the Surface Track tool and the track editor.



For more information on the track editor, see *Track Editor* on page 968.

🗋	All instances of the Track tool on a sensor or in a PC instance of GoPxL share the same path file set in **File** (ending in .trk). For this reason, you must be careful when editing or removing path files shared by another instance of the tool.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Key Concepts

The following are important concepts for using both the track editor (see *Track Editor* on page 968) and the Surface Track tool itself:

Track: The material being measured, for example glue or sealant. The material can sit on a flat area on the target, or sit in a groove where the material touches one or both sides.

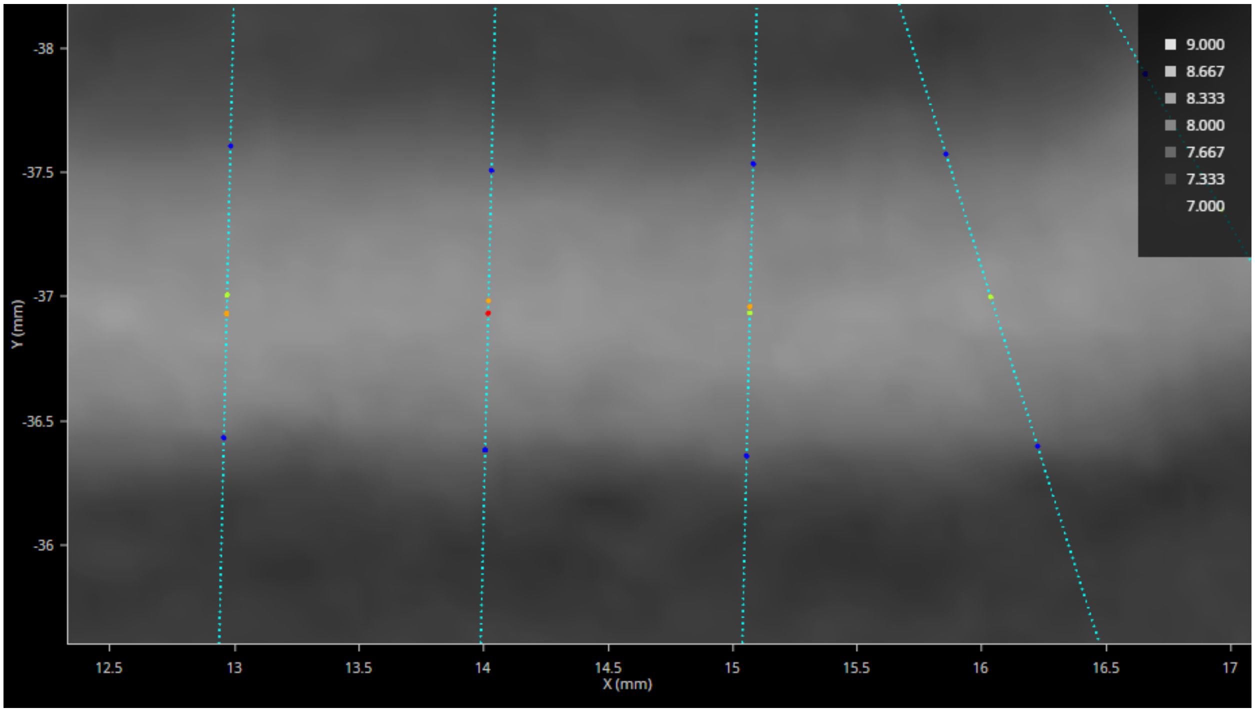
Path: The ideal center line of the track. You define the path in the track editor. You can define more than one path for use on scanned targets, but the Surface Track tool returns measurements for all paths.

Caliper / ruler: A caliper (in the tool) or ruler (in the track editor) is an area perpendicular to the path you define. You define the size (length and width) and spacing of the rulers in the track editor. The Surface Track tool extracts a profile from the surface data beneath a caliper and performs measurements based on the values you set in the tool's parameters.

Caliper / ruler profiles: The profiles extracted from the surface data under a caliper. The tool's measurements, which are configured using the tool's settings, are applied to these profiles.

Segment: One portion of the path, between points created by clicking on an image of scan data in the track editor. You can choose to configure rulers in segments independently, or choose to configure them in a batch mode.

The following shows a track with calipers. Colored dots provide important information (see below).



Track tool in data viewer, showing a track (lighter grey horizontal strip in the middle), calipers running perpendicular to the track (light blue dots). Dots of other colors provide additional information (see below).

When you enable **Show Measurement Results**, the tool displays dots on the calipers to provide the following information (see also the images below):

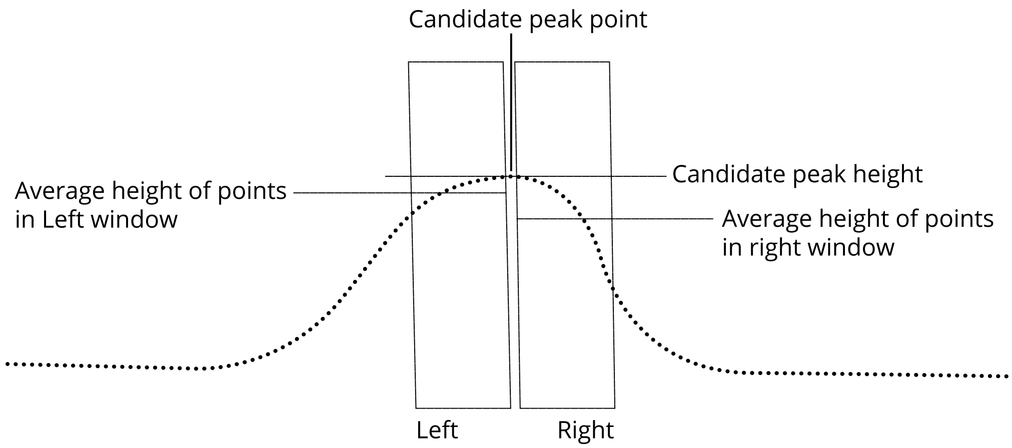
- Light blue dots: The data points in the caliper profile. When you enable **Show Path and Calipers**, the tool displays a white line centered on these dots to indicate the location of the caliper.
- Dark blue dotes: The detected sides of the track. These represent the width of the track under that caliper.
- Green dot: The center point (between the two sides of the track) on a caliper that *passes* all criteria set in the tool. These count toward the "OK Count" measurement.
- Red dots: The center point (between the two sides of the track) on a caliper that *fails* at least one of the criteria set in the tool. These count toward the "NG Count" measurement. Causes for failures include the width and the height of the track, as well as the area under the caliper profile.
- Orange dots: The peak (highest) point on the ruler. If the center point (green or red) is the same as the peak point, the tool only shows the center point.

Track Location

The tool attempts to locate the track using the profile data it extracts under each caliper, and does this by first locating the "peak" (the highest point on the caliper profile, based on certain criteria) and then locating the side points representing the "sides" of the track.

Peak Detection

The tool determines the peak point on a caliper profile by moving two windows—one to each side of the point being examined—and comparing the average height in those windows with the height of the point being examined. (The size of these windows is specified in **Center Window Size**.) If the height of the point being examined is greater than both the left and right average height by the value specified in **Center Threshold**, that point is considered a candidate peak point. The tool uses the candidate point with the *highest* height over both windows as the peak point.



Side Detection

After the tool has located the peak point, it locates the sides of the track starting from the peak point. You can choose between two methods for side detection: Maximum Gradient and Height Threshold.

Maximum Gradient:

Use this side detection method when the slope of the two sides show a clear drop-off. The following settings define the area in which the tool searches for a maximum gradient, which will determine the edge of the track.

Maximum Gradient Side Detection Parameters

Side Window Size	The size of the two adjacent windows the tool uses to determine the maximum slope on the left and right side of the track. Set this to roughly 3 to 5 times the smaller of the X and Y resolution of the sensor.
Max Track Width	The maximum width of the track over the caliper profile the tool searches for edge points. The tool uses this value to limit where the edge of the track might be detected.

Height Threshold:

Use this side detection method when the slope of two sides is very gradual. The tool finds the left and right edges by averaging the height of small fixed-size windows moving away from the peak point. Edge points are the left-most and right-most window locations where the average height is *below* a minimum height threshold.

Height Threshold Side Detection Parameters

Side Height Threshold	The minimum height that the average calculated in the fixed-width height threshold windows must be below.

Center Point Detection

The Track tool calculates the center point as the mid point between the left and right side points. This means that the center point may be different from the peak point.

Configuring the Surface Track Tool

To configure the tool, you must first acquire scan data of a representative target; preferably, the material on the target will fall within the expected tolerances. Next, you save the scan data from within the Track tool, and then load the scan data into the track editor. Then, after adding a path or paths, and configuring rulers on the data, you load the track data back into the Track tool. Finally, you configure the tool. For more information on key concepts you need to understand to configure the Track tool, see *Key Concepts* on page 662.

To configure the Track tool:

1. Scan a representative target, or load previously scanned data.

   For more information on loading previously scanned data, see *Working with Scan Data (Toolbar)* on page 112.

2. Add a Surface Track tool.

   For more information on adding a tool, see *Adding a Tool* on page 235.

3. In the Surface Track tool, choose **Create** from the **Operation** drop-down.

   The tool creates a file (for example, SurfaceTrack-0000.trk) containing scan data. Creating the file may take a few seconds.

4. Launch the track editor and configure the path or paths.

   For information on using the track editor, see *Track Editor* on page 968.

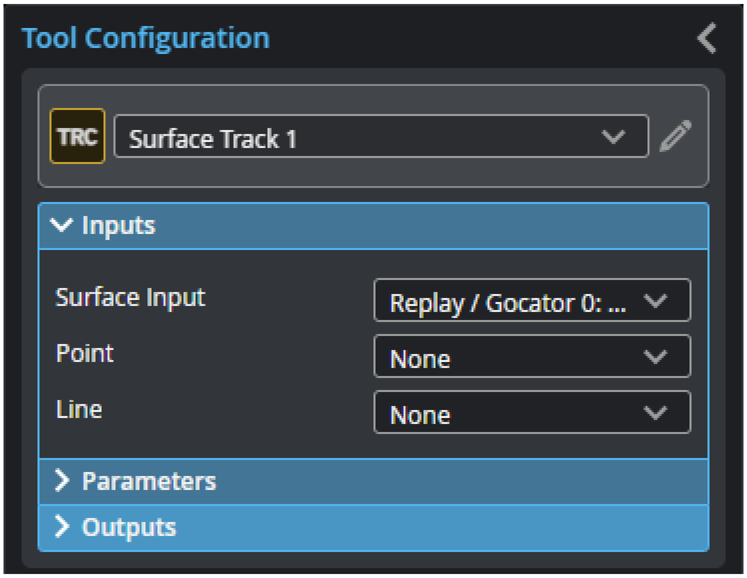
5. After you have finished editing the track data in the track editor, in the Surface Track tool, choose **Load** in the **Operation** drop-down to load the path data you just created.

6. Configure the Track tool as required.

   For information on the tool's measurements and settings, see the below.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

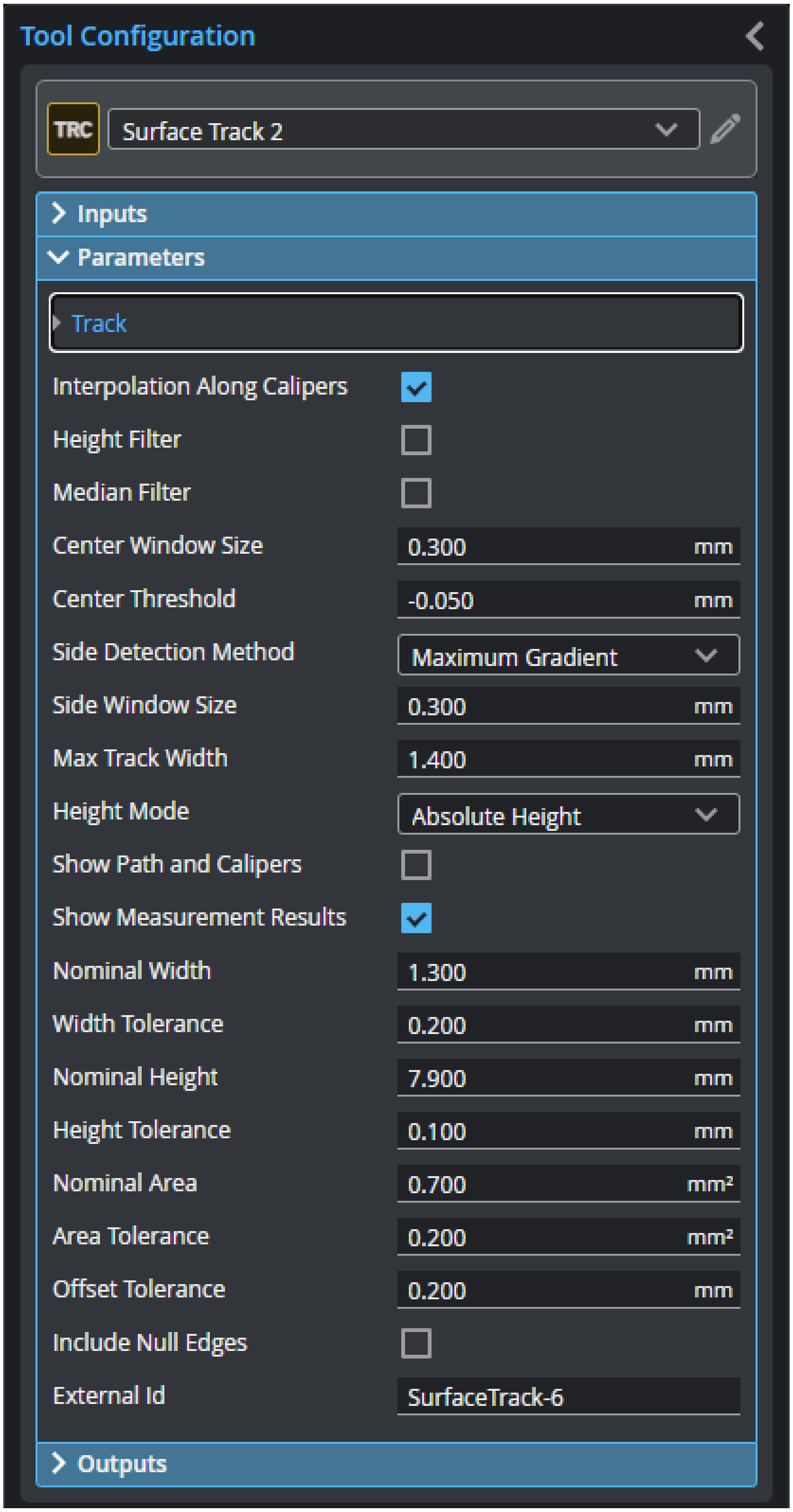


Inputs

Name	Description
Surface Input	The data the tool applies measurements to or processes.
Point Line	Point and line geometric features (produced by another tool) that you can select as anchors for translation and rotation transformations, respectively. Currently, you must select both in order for anchoring to work. For more information on geometric features, see *Geometric Features* on page 262.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.


Tool Configuration
TRC
Surface Track 2
Inputs
Parameters
Track
Interpolation Along Calipers
Height Filter
Median Filter
Center Window Size
0.300 mm
Center Threshold
-0.050 mm
Side Detection Method
Maximum Gradient
Side Window Size
0.300 mm
Max Track Width
1.400 mm
Height Mode
Absolute Height
Show Path and Calipers
Show Measurement Results
Nominal Width
1.300 mm
Width Tolerance
0.200 mm
Nominal Height
7.900 mm
Height Tolerance
0.100 mm
Nominal Area
0.700 mm²
Area Tolerance
0.200 mm²
Offset Tolerance
0.200 mm
Include Null Edges
External Id
SurfaceTrack-6
Outputs

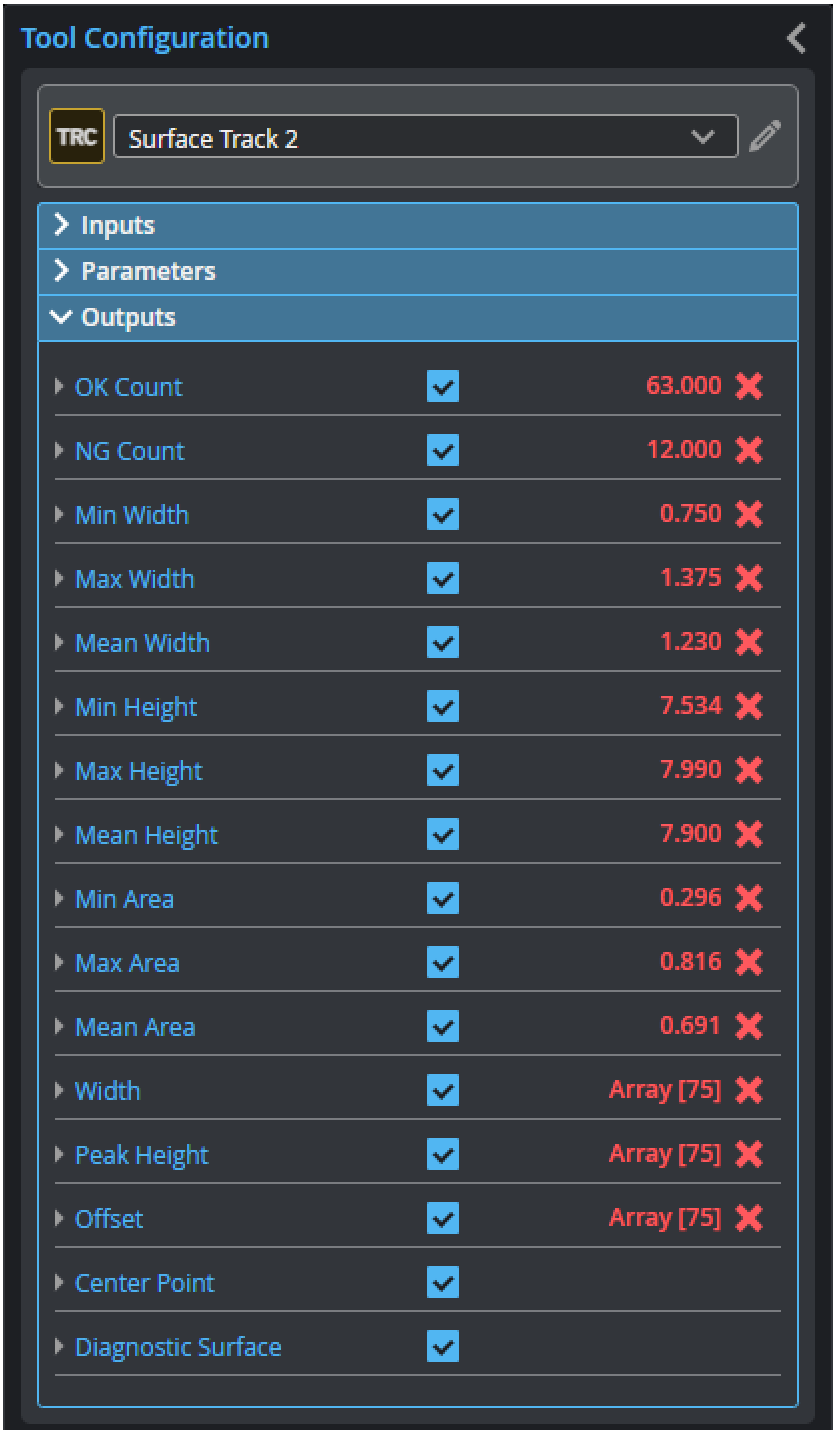
Main Parameters

Parameter	Description
Track	When expanded, displays file-related operations.
File	The track file that contains scan and path data. You add paths and calipers to the file using the track editor. For more information on the track editor, see *Track Editor* on page 968.
Operation	Provides operations related to the track file. One of the following: • **Normal**: Selected by the tool after you perform another file operation. • **Create**: Creates a new track file for use with the track editor. • **Load**: Loads the path file selected in **File**. • **Save**: Saves changes made in the scan data, as well as the geometric features used as anchors in the **Point Feature** and **Line Feature** settings, to the file selected in **File**. • **Delete**: Deletes the path file selected in **File**. • **Refresh**: Refreshes the list of files.
Interpolation Along Calipers	Enables linear interpolation on the profile extracted from the calipers to achieve sub-pixel accuracy in the width and height measurements.
Height Filter *Dependent settings* Threshold High Threshold Low	When **Height Filter** is enabled, use the **Threshold Low** and **Threshold High** settings to set a range to filter out noise or exclude other undesired data along the caliper profiles.
Median Filter *Dependent settings* Window Size	When **Median Filter** is enabled, specify the window the tool will use to smooth the height values of the points in the caliper profiles in the **Window Size** setting.
Center Window Size	The size of the left and right windows the tool moves along the caliper profile to detect whether the point centered between the two is the highest point along a caliper (the center point). Set this to roughly 50% of the typical width of the track as a starting point.
Center Threshold	The center point is determined by moving two side-by-side windows (left and right, **Center Window Size** setting) over each caliper profile. At each point, the height value between the two windows is compared to the average height of the left and right windows. If the center point height is greater, by the amount set in **Center Threshold**, than the average height in both the left and right windows, that point is considered a candidate center point. The candidate center point with the highest average height over both windows is used as the center point. It may be necessary to use a negative number in some cases, for example, when

Parameter	Description
	the top point slightly dips below its surroundings.
Side Detection Method *Dependent settings* Side Window Size Max Track Width Side Height Threshold	The method the tool uses to detect the two sides of the track. One of the following: **Maximum Gradient** or **Height Threshold**. For more information on side detection method settings, see *Side Detection* on page 664.
Height Mode	Determines how height values are interpreted in the tool's **Nominal Height** setting and what the returned height measurements represent. One of the following: **Absolute Height** - Height values are interpreted globally (all of the scan data). **Step Height** - Height values are relative to the surrounding area of the track.
Show Path and Calipers	Displays the path and calipers (as defined in the track editor) on the scan data.
Show Measurement Result	Shows dots on each caliper representing the results of the measurements on the profile extracted from the surface data under the caliper. For more information, see *Key Concepts* on page 662.
Nominal Width	The expected width of the track.
Width Tolerance	The tolerance applied to the nominal width.
Nominal Height	The expected height of the track. The **Height Mode** setting determines how the height is interpreted (absolute or step); for more information, see *Height Mode* above. This setting applies to the peak point, not the center point.
Height Tolerance	The tolerance applied to the nominal height. This setting applies to the peak point, not the center point.
Nominal Area	The expected cross-sectional area under the calipers on the track.
Area Tolerance	The tolerance applied to the nominal area.
Offset Tolerance	The maximum allowed distance between the center (highest) point on a ruler and the path. This setting applies to the center point.
Include Null Edges *Dependent settings* Null Fill Value	When enabled, displays a **Null Fill Value** parameter you can set to replace null values with the value.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

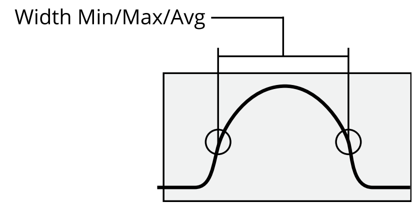
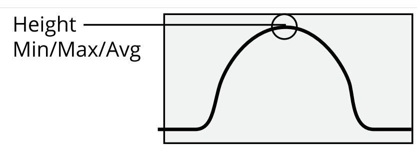
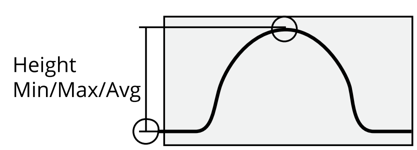
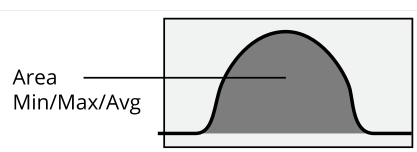
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
OK Count Returns the number of rulers along the path that pass all of the criteria set in the tool's parameters.	
NG Count Returns the number of rulers along the track path that fail the criteria set in the tool's parameters. (They are "no good.")	
Min Width **Max Width** **Mean Width** These measurements return the minimum, maximum, and mean width of the track.	 *Width measurements on a ruler profile. The Track tool's settings determine the locations of the "sides" of the track.*
Min Height **Max Height** **Mean Height** These measurements return the minimum, maximum, and mean height of the track at the center point. When **Height Mode** is set to **Absolute Height**, the height returned is the global height. When it is set to **Step Height**, the height is relative to the surface next to the track.	 *Height measurements on a ruler profile with Height Mode set to Absolute Height.*  *Height measurements on a ruler profile with Height Mode set to Step Height.*
Min Area **Max Area** **Mean Area** These measurements return the minimum, maximum, and mean area under the rulers.	 *Area measurements under a ruler profile.*
Width **Peak Height**	

Measurement	Illustration
Offset Arrays containing the widths, peak heights, and offsets of each ruler.	

Features

Type	Description
Center Point	An array of Point geometric feature representing the center point of each ruler.

For more information on geometric features, see *Geometric Features* on page 262.

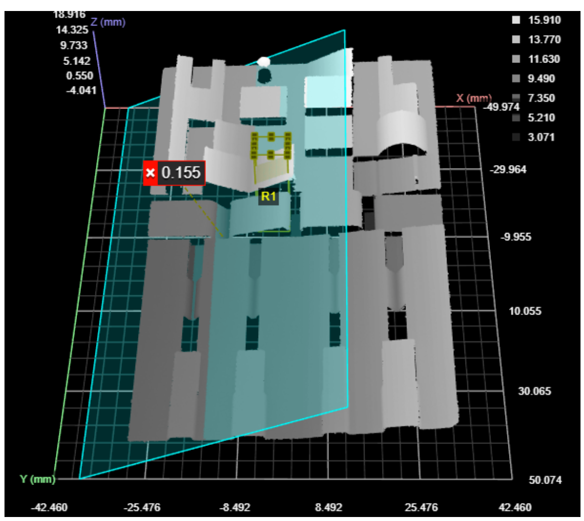
Data

Type	Description
Diagnostic Surface	Surface data created by combining the extracted profiles. Use for diagnostics.

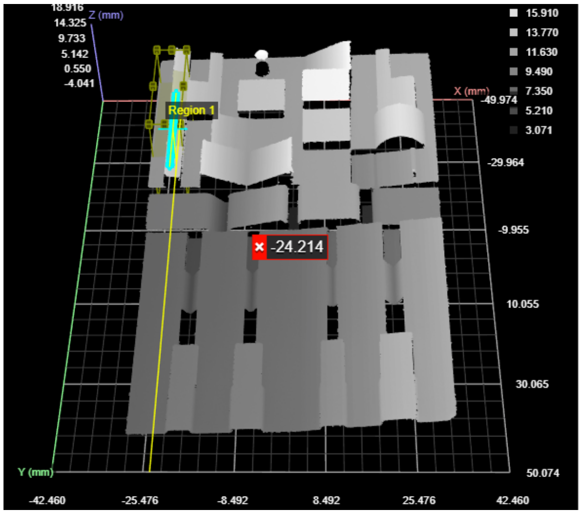
Surface Transform

The Surface Transform tool generates a new surface based on the coordinate system of geometric features the tool uses as input. The tool can take a zero-plane, line, and origin point to define this new coordinate system. You can then apply the built-in measurement tools or GDK tools to this new surface data. This could let you, for example, get the height of a feature relative to a slightly tilted or warped adjacent or surrounding reference surface, rather than the absolute height in the original scan volume relative to the sensor. The result is increased repeatability of your measurements.

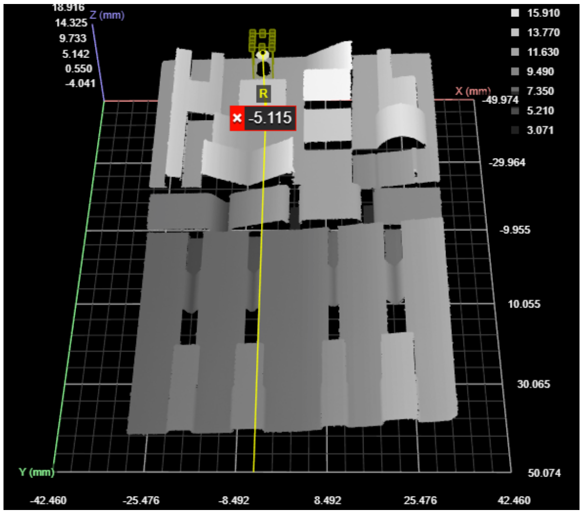
In *Combinations of geometric feature inputs and results* on page 675, the following geometric features are used by a Surface Transform tool in various combinations (a plane, a line, and a point).



A Surface Plane tool, with the region set to a small left-facing angled surface

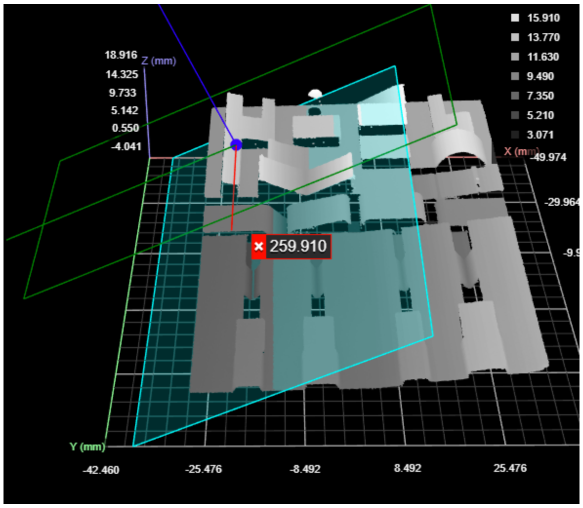


A Surface Edge tool, with the region set to the left edge of a raised surface (upper left of data viewer).



A Surface Position tool (maximum Z), with the region set to the raised point near the top of the data viewer.

Furthermore, in the sections below, two types of data are shown: the original (input) scan data and the transformed data. When the tool displays the original data, it overlays indicators of the new, transformed coordinate system on the data.



A Surface Transform tool using all three types of geometric feature inputs.
The data viewer is set to display the input surface data with an overlay of the transformed coordinate system.

In the data viewer, the following is displayed:

X, Y, and Z axes

The transformed axes are represented above by the red, green, and blue lines intersecting on the surface data above. Note how these are rotated with respect to the original coordinate system (the background grid, axes, and values along the axes).

Origin

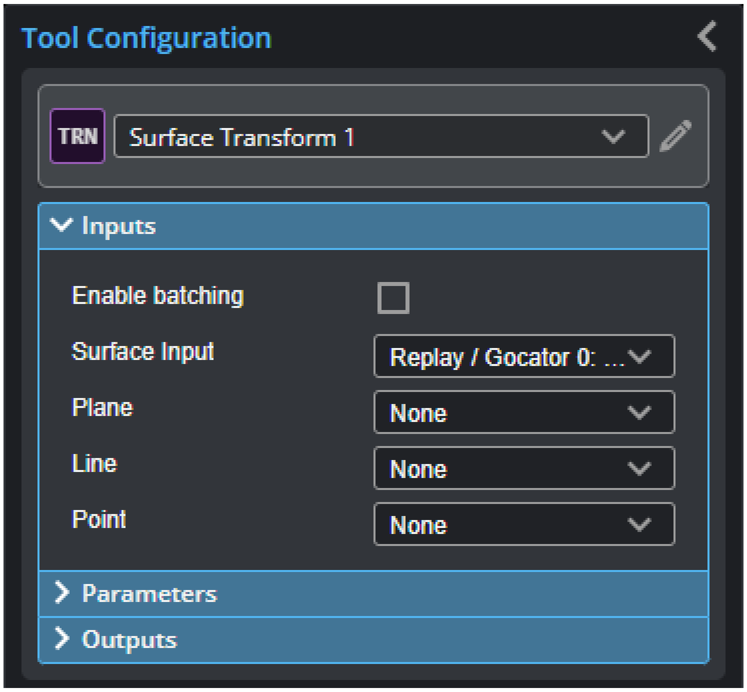
The new origin is represented by the dark blue dot at the intersection of the transformed axes.

Plane

The new plane is represented by the cyan rectangle.

	To switch between the original and transformed data, choose the output under the data viewer or in the tool's list of outputs.

Inputs



Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.
Plane Line Point	The Plane, Line, and Point geometric features the tool uses to perform transformations. The required geometric features depends on your application. For more information, see *Combinations of geometric feature inputs and results* below.

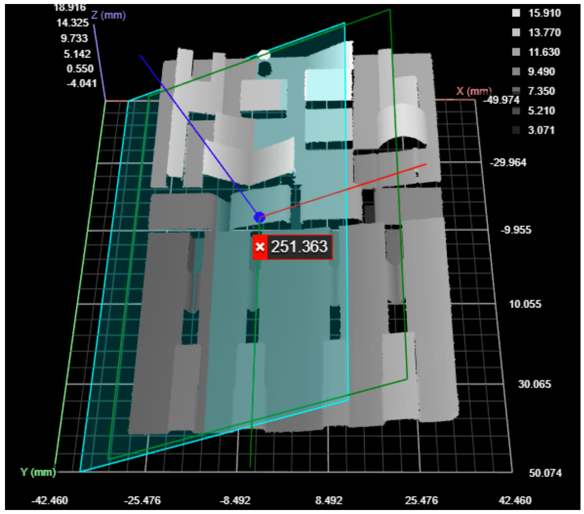
Combinations of geometric feature inputs and results

The Surface Transform tool accepts all combinations of input geometric features (plane, line, and point). For details and examples of each, see the following sections.

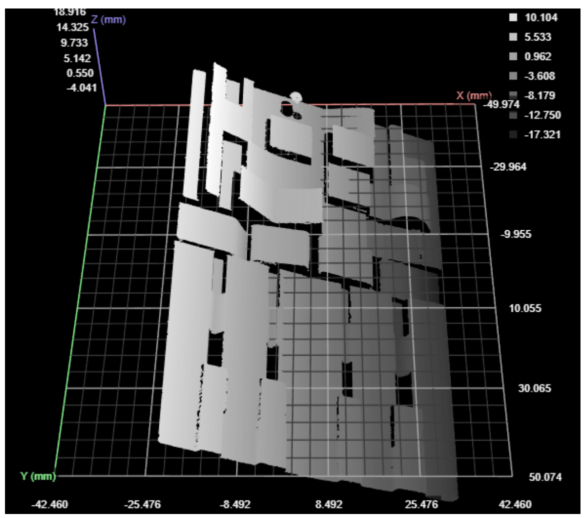
Plane

New Z=0 XY Plane	New X Axis	New Origin
Matches the input plane.	Parallel to the old X axis.	Old origin projected to plane.

Original data with overlay



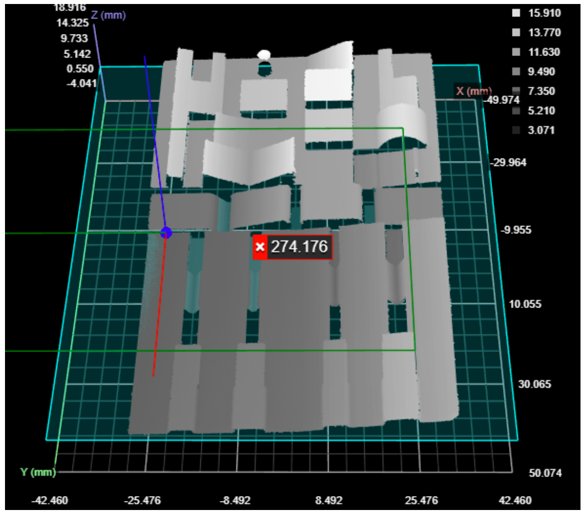
Transformed data



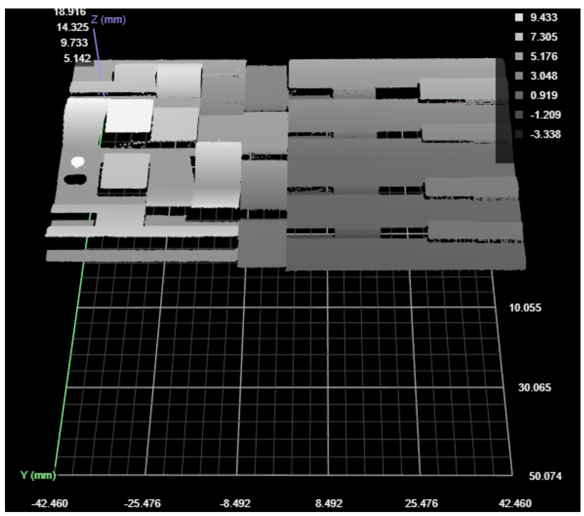
Line

New Z=0 XY Plane	New X Axis	New Origin
The new plane contains the line. The intersection of the new plane and the old plane is perpendicular to the input line.	Matches the line.	Old origin projected onto the line.

Original data with overlay



Transformed data

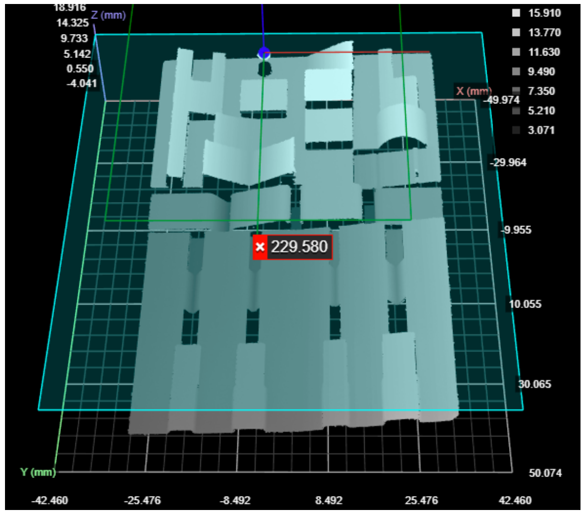


The direction of the X axis depends on the tool generating the line that Surface Transform takes as input. You may need to adjust the direction using the **Add Fixed Transform** settings.

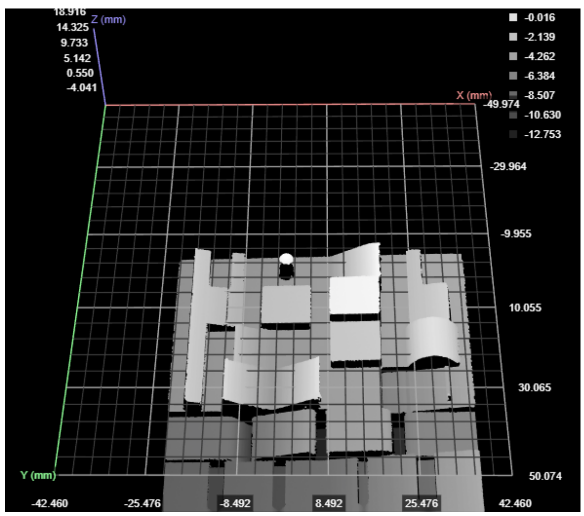
Point

New Z=0 XY Plane	New X Axis	New Origin
Through the input point, parallel to old Z=0 plane.	Parallel to the old axis.	The input point.

Original data with overlay



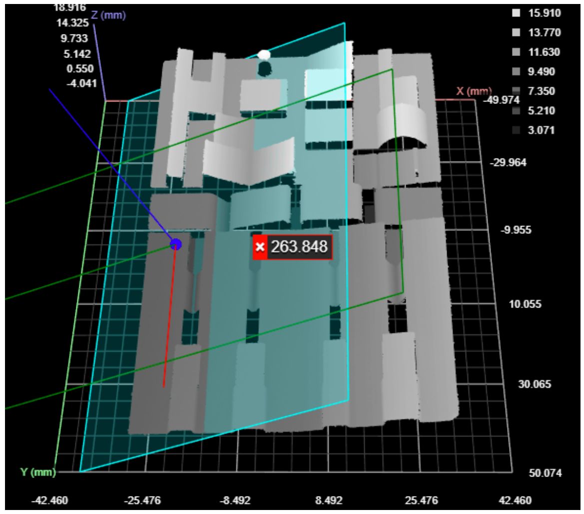
Transformed data



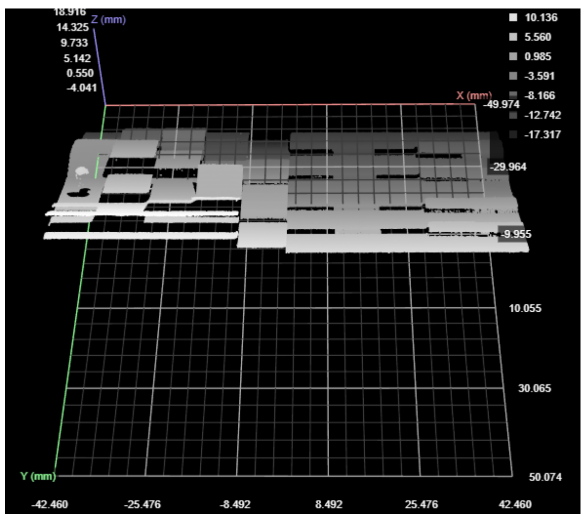
Plane + Line

New Z=0 XY Plane	New X Axis	New Origin
Matches the input plane.	Line projected onto the plane.	Old origin projected onto the projected line.

Original data with overlay



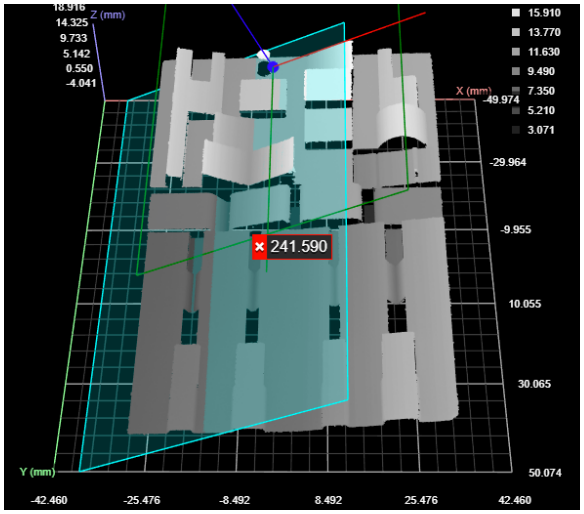
Transformed data



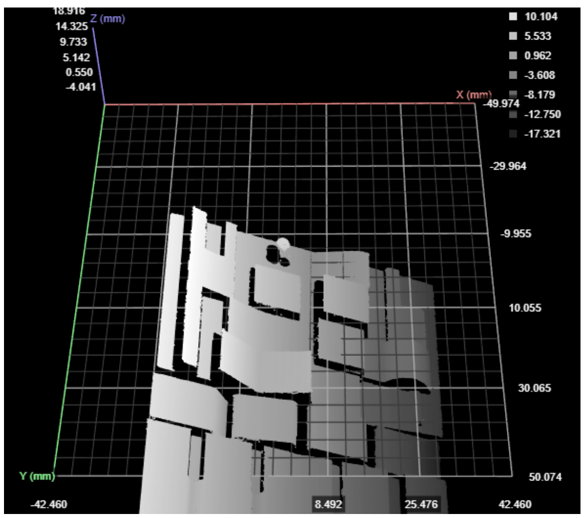
Plane + Point

New Z=0 XY Plane	New X Axis	New Origin
Matches the input plane.	Parallel to the old X axis.	At the input point, projected onto the plane.

Original data with overlay



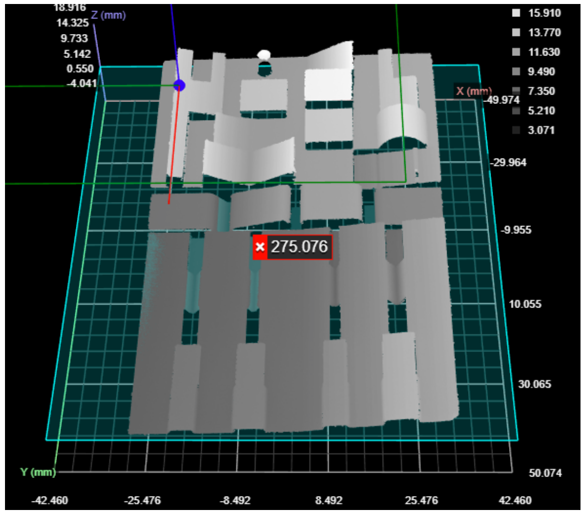
Transformed data



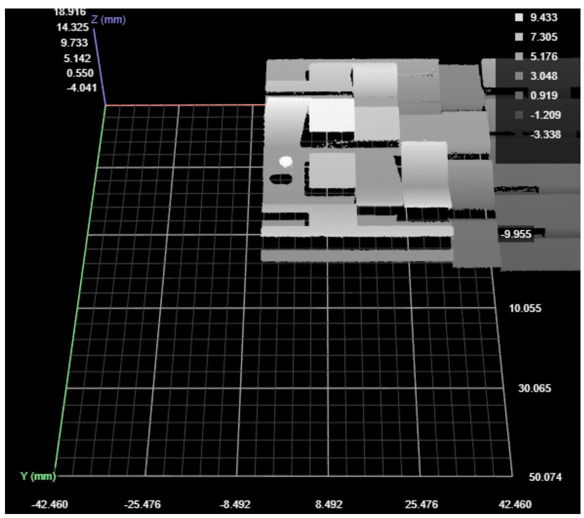
Line + Point

New Z=0 XY Plane	New X Axis	New Origin
The new plane contains the line. The intersection of the new plane and the old plane is perpendicular to the input line.	Matches the line.	The input point projected onto the line.

Original data with overlay



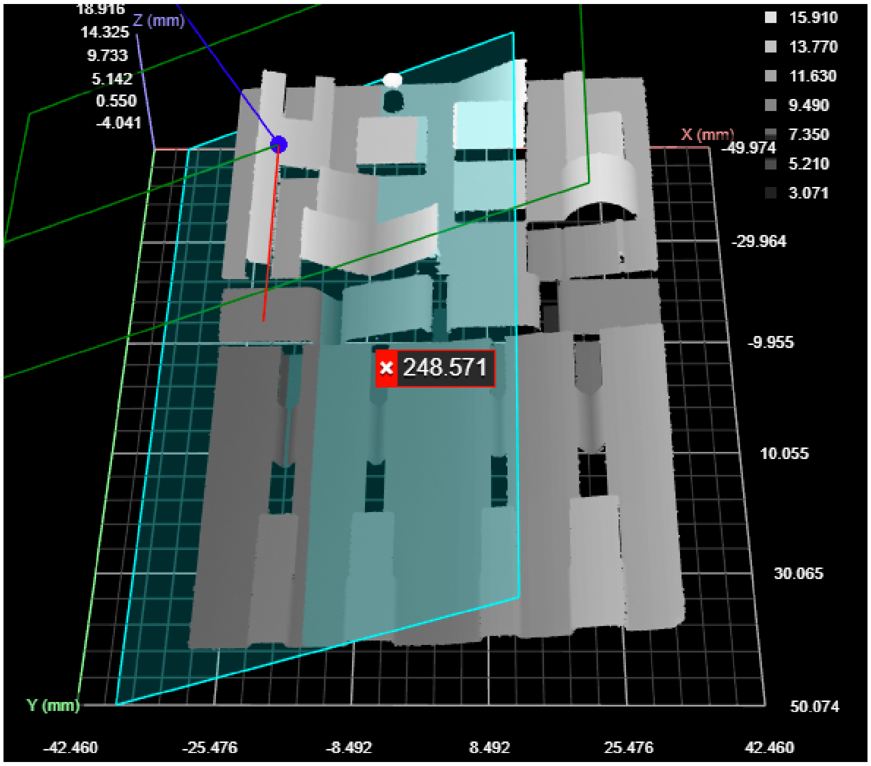
Transformed data



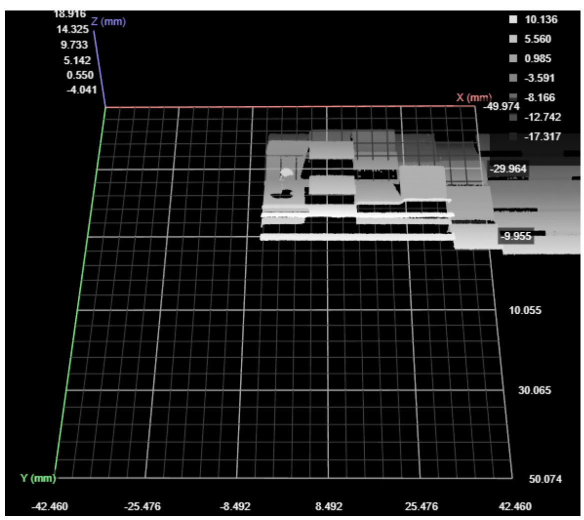
Plane + Line + Point

New Z=0 XY Plane	New X Axis	New Origin
Matches the input plane.	The input line projected onto the plane.	The input point projected onto the input line.

Original data with overlay



Transformed data



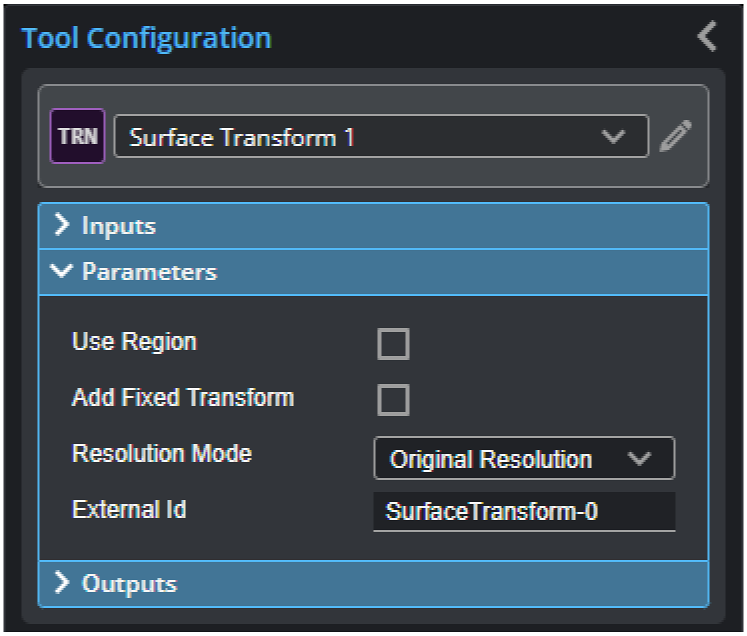
For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Scaling Modes

Line profile sensors have independent X and Y resolution settings: the former is set using the **Uniform spacing interval** setting (for more information, see *Uniform Spacing* on page 197), whereas the Y resolution is set using the **Spacing** setting in the **Trigger** panel (for more information, see *Triggers* on page 206. In many applications, the X resolution can be as much as 3-5 times higher than the Y resolution. Rotating scan data around Z greater than 45 degrees (for example, with the Transform tool) when there is a large difference between X and Y resolutions can result in significant data quality reduction. To avoid data quality reduction, choose one of the scaling modes that the tool offers (see below).

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.

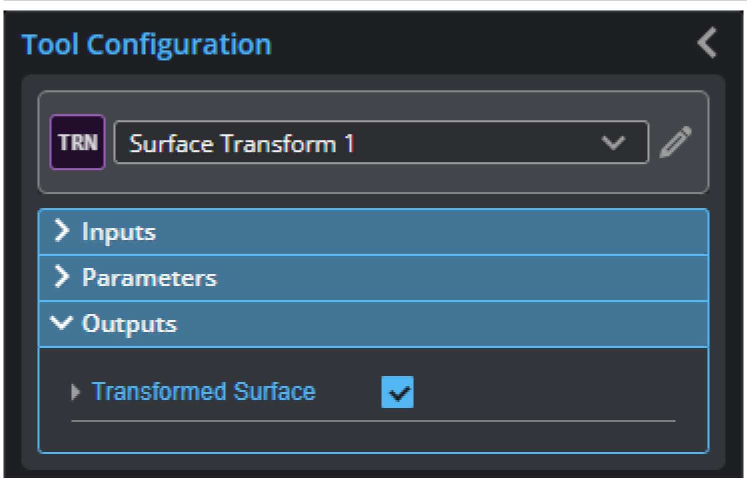


Parameters

Parameter	Description
Use Region	When enabled, displays Region parameters (see below). When disabled, the tool uses all data.
Region	The region to which the tool's measurements will apply. For more information, see *Regions* on page 250.
Add Fixed Transform	When enabled, displays X, Y, and Z offset and angle fields you can use to set additional transformations, which are applied after any transformations supplied by the input geometric features. Setting a fixed transformation can be useful if the geometric features the tool uses results in data rotated to an unusual orientation; you could, for example, rotate the data 90 or 180 degrees so that it is in the "expected" orientation, or shift it so that it's easier to work with.
Resolution Mode	Determines whether the tool scales the X or Y resolution so that they are the same (a 1:1 ratio), or leaves the X and Y resolutions as the original. One of the following: • **Optimal (uniform)**

Parameter	Description
	Brings the X/Y resolution ratio to 1:1 while preserving the pixel area. Best for random rotation around Z. Provides a balance between the highest and lowest possible resolutions, requiring an average amount of memory and processing time compared to the **High Oriented (uniform)** or **Low Oriented (uniform)** options. • **High Oriented (uniform)** Interpolates the lower resolution to match the higher resolution (between X and Y) in the input. Choose this option when increased resolution is preferred over speed and low memory usage. (This can result in a very high resolution output, creating a lot of data for subsequent tools to process. This can in turn result in slower processing.) • **Low Oriented (uniform)** Decimates the higher resolution to match the lower resolution (between X and Y) in the input. Choose this option when speed and low memory usage is preferred over resolution. (It can result in significant data quality reduction with large Z rotations if the X and Y resolutions of the input are very different.) • **Original Resolution** Keeps the original X and Y resolution of the scan. Use this option only when you expect little or no Z rotation. Otherwise, with X/Y resolution ratios that are not 1:1, large rotation around Z results in severe data quality reduction. For more information, see *Scaling Modes* on page 682.
Resolution Mode	

Outputs



Data

Type	Description
Transformed Surface	The transformed surface.

Surface Vibration Correction

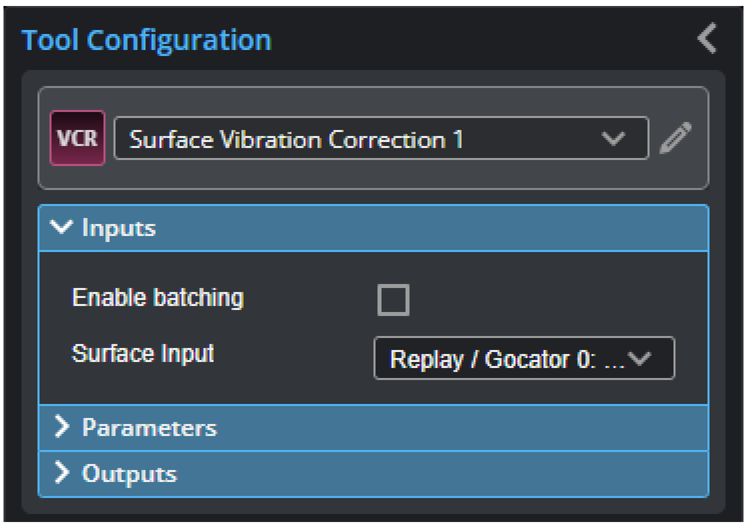
The Vibration Correction tool analyzes variation in surface data to remove high frequency noise in the data. The tool is useful for improving repeatability and accuracy of measurements when subtle vibrations in your transport system introduce height variations. The tool's intended use is to send corrected surface data to other tools.

> The Vibration Correction tool requires at least 64 lines of data in the surface data it receives as input to be able to output corrected surface data.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

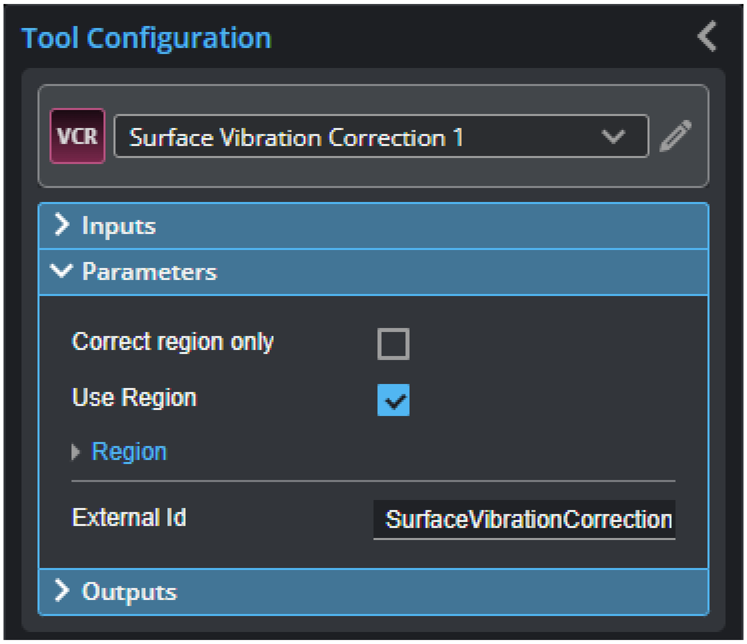


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Surface Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

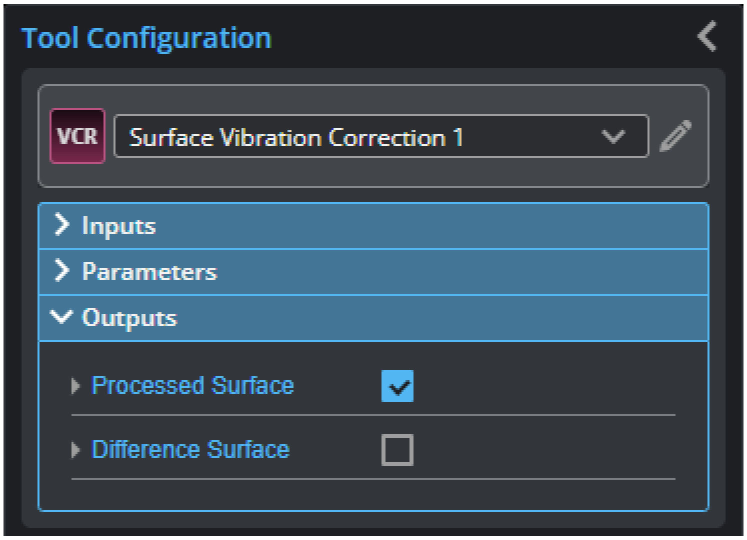


Parameters

Parameter	Description
Correct Region Only	If enabled, only the area under the region is corrected for vibration in the Surface data the tool outputs. This setting can be useful if vibration regularly occurs in a specific area of the scan data. When disabled, the tool applies correction to all of the scan data. This option is only displayed if **Use Region** is enabled.
Use Region	When enabled, lets you set a region and optionally choose to apply vibration correction only to that region (by enabling **Correct Region only**).
Region	The region whose data the tool will use to calculate the vibration correction. If **Correct Region Only** is enabled, the tool applies vibration correction *only* to the region. If **Correct Region Only** is disabled, the tool applies vibration correction to all of the scan data.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Data

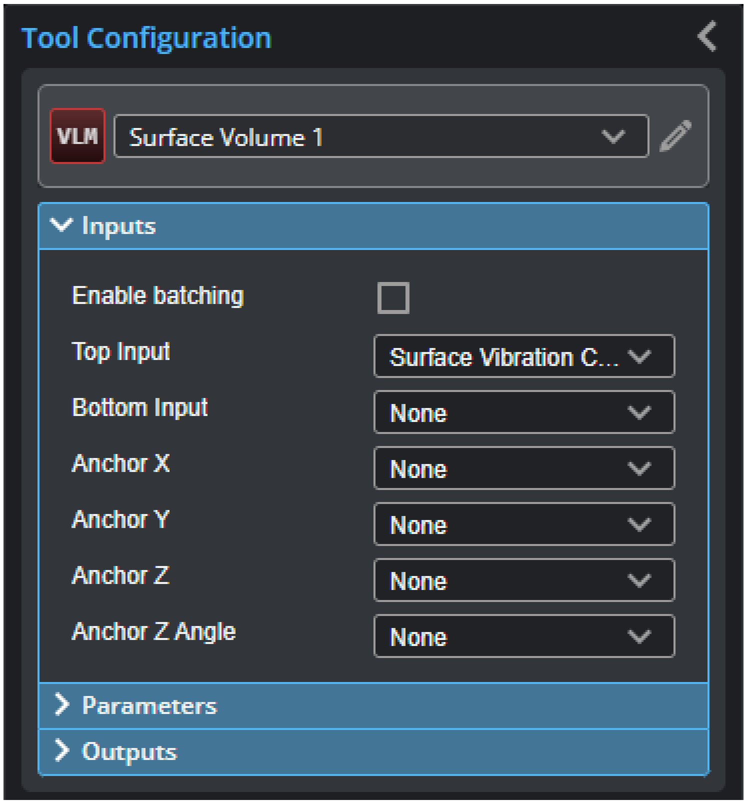
Type	Description
Processed Surface	Surface data corrected for vibration, available for use as input in other tools.
Difference Surface	Diagnostic Surface data showing the difference between the corrected surface and the original.

Surface Volume

The Volume tool returns the volume, area, and thickness of a part.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs



> To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

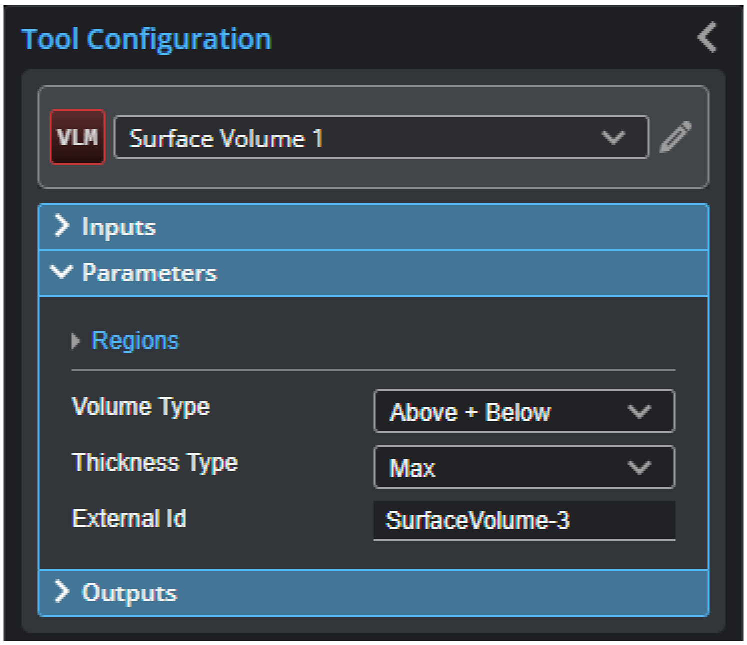
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each surface in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Top Input Bottom Input	The data the tool applies measurements to or processes. This tool can optionally take a second, bottom input. When **Enable Batching** is checked, both inputs take an array.
Anchor X	The X, Y, or Z measurement of another tool that this tool uses as a positional

Name	Description
Anchor Y Anchor Z	anchor. Positional anchors are optional.
Anchor Z Angle	The Z Angle measurement of another tool to use as a rotational anchor for this tool. Rotational anchors are optional.

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.



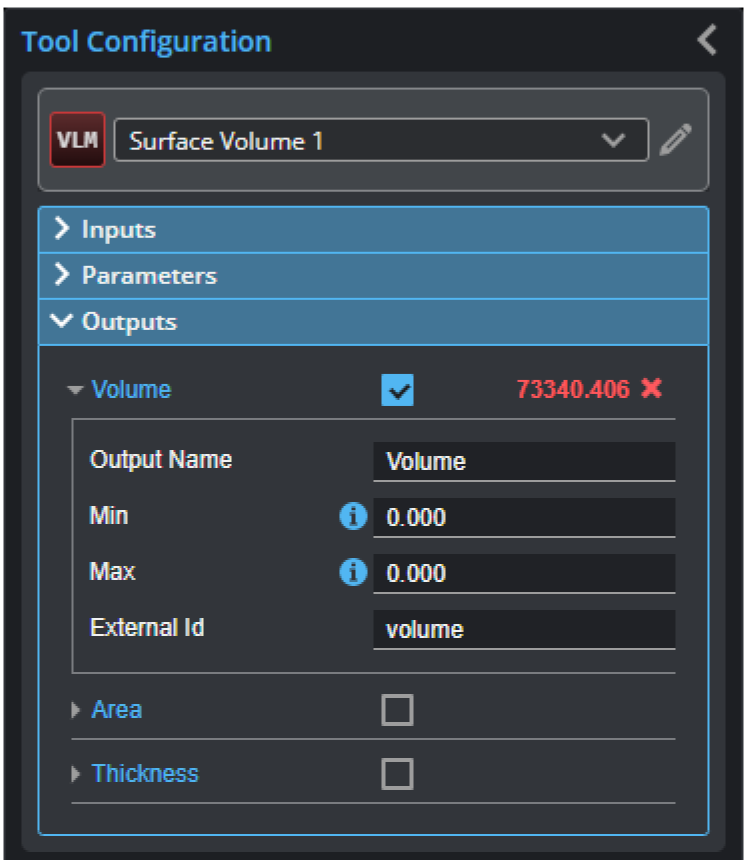
Parameters

Parameter	Description
Regions	When expanded, displays the region- and mask-related settings.
Enable	Enables regions and displays the region- and mask-related settings (see below).
Mask Mode Region Type 1 Region 1	When you enable regions (see above), the tool displays additional settings related to the region type. For details on the regions supported by this tool and their settings, see *Flexible Regions* on page 253. For general information on regions and the difference between standard and "flexible" regions, see *Regions* on page 250.
Volume Type	Used to compute different kinds of surface volumes. One of the following: • Above • Below • Above + Below (default) • Above - Below Above means the volume calculated using Surface data that is above the Z = 0 plane.

Parameter	Description
	Below means the volume calculated using Surface data that is below the Z = 0 plane. Note that Above and Below can't be negative, but Above - Below can be negative if Below > Above.
Thickness Type *(Thickness measurement only)*	Determines which feature in the Surface data the tool uses to calculate thickness. One of the following: • Max • Min • Average • Median • Centroid 2D (height of the centroid in the XY plane) • Centroid 3D (height of the centroid in the XYZ space).
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

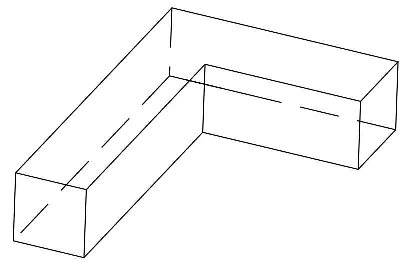
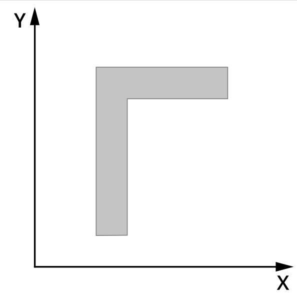
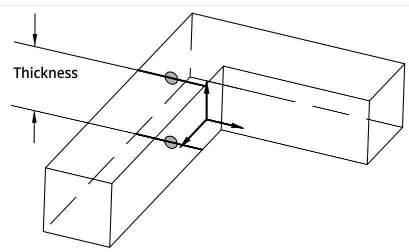
Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Volume Measures volume in XYZ space.	
Area Measures area in the XY plane. The area is the number of valid points multiplied by the X and Y resolution. Note that this is different compared to the area calculations produced by Surface Segmentation and Surface Blob; for more information, see the descriptions of the *Area {n}* measurements in *Surface Segmentation* on page 635 and *Surface Blob* on page 455.	
Thickness Measures thickness (height) of a part. Uncheck this measurement if you don't need it.	

Feature Measurement

The following sections describe the Feature tools.

Feature tools generate or measure geometric features; for more information on geometric features, see *Geometric Features* on page 262. The tools let you work with more complex geometry, letting you implement applications more quickly by reducing dependence on scripts and GDK tools to perform these kinds of measurements.

The "Create" tools (such as Feature Circle Create) generate geometric features from other tools or using values you provide.

The Feature Dimension tool lets you take measurements using geometric features.

The Feature Robot Pose tool is for use with robots.

Feature Circle Create

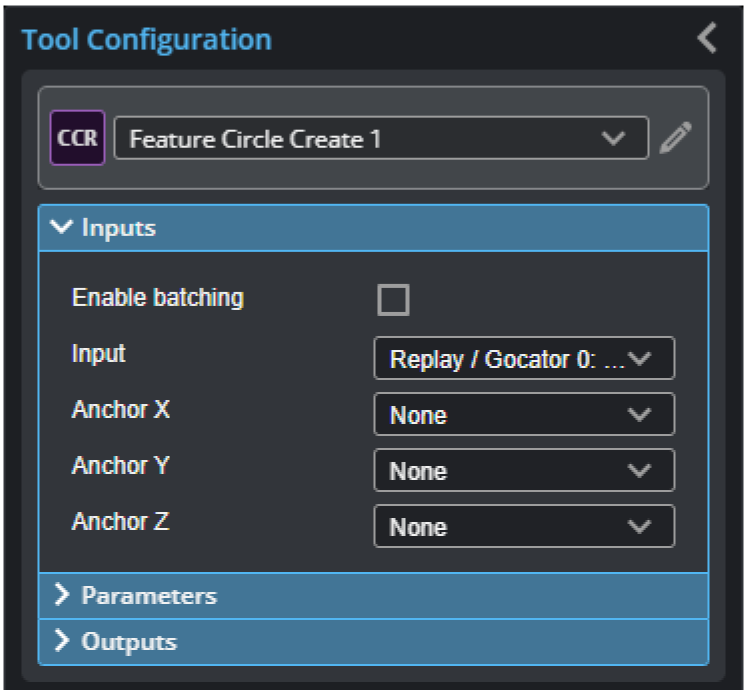
The Feature Circle Create tool lets you generate a Circle geometric features from geometric features generated by other tools. You can also generate a circle from an arbitrary position and radius. The following outputs are available (for more information, see *Output Types* on the next page):

- Constant circle
- Circle from points
- Circle from point and plane

For more information on geometric features, see *Geometric Features* on page 262.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

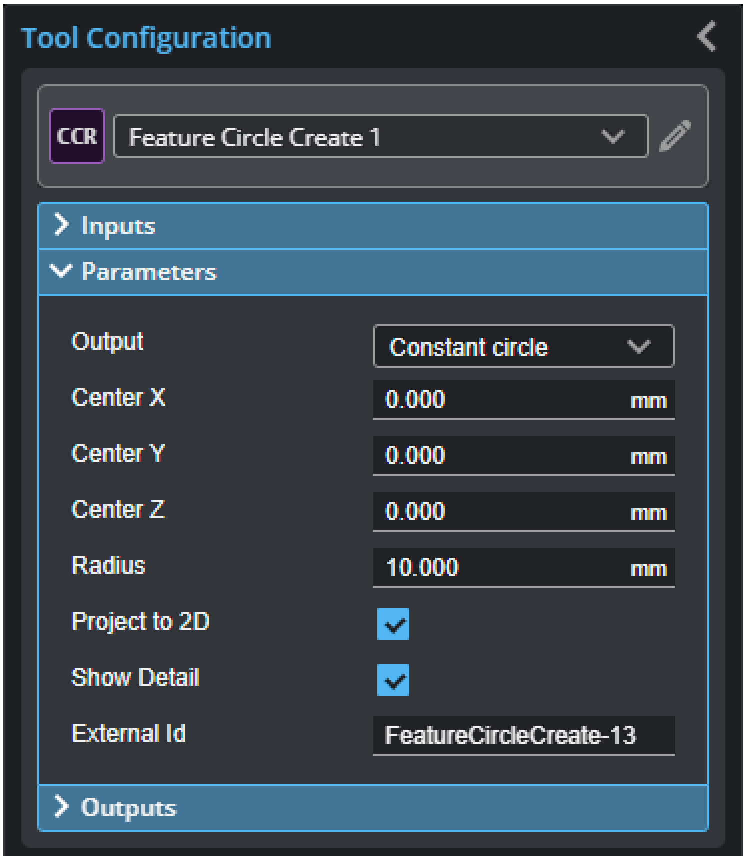


This tool can take an array as input. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.

The tool's geometric feature inputs depend on the output type you choose in the **Output** parameter. For information on the required inputs and parameters of each output type, see *Output Types* on the next page.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Output	The type of output the tool generates. Switching between the options changes the input types displayed in Input section and the parameters displayed in the Parameters section. For information on the output types and the required inputs and related parameters, see *Output Types* below.
Project To 2D	If checked, generates a circle in the XZ plane for Profile mode and in the XY plane for Surface mode. Otherwise, generates a 3D circle. Not shown when **Output** is set to “Circle from point and plane”.
Show Detail	Toggles the display of the input geometric features in the data viewer.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Output Types

The following sections describe the output types available in the tool's **Output** parameter, in addition to the required inputs and parameters.

Constant Circle

The **Constant circle** output type is useful if scan data from frame to frame is reliably fixed and you want to measure from an arbitrary circle to another feature. When you choose this output type, the tool displays the following parameters you can manually fill in to create a constant circle.

Disable **Project To 2D** to input an arbitrary normal vector (using the **Normal** parameters) to get a tilted 3D circle.

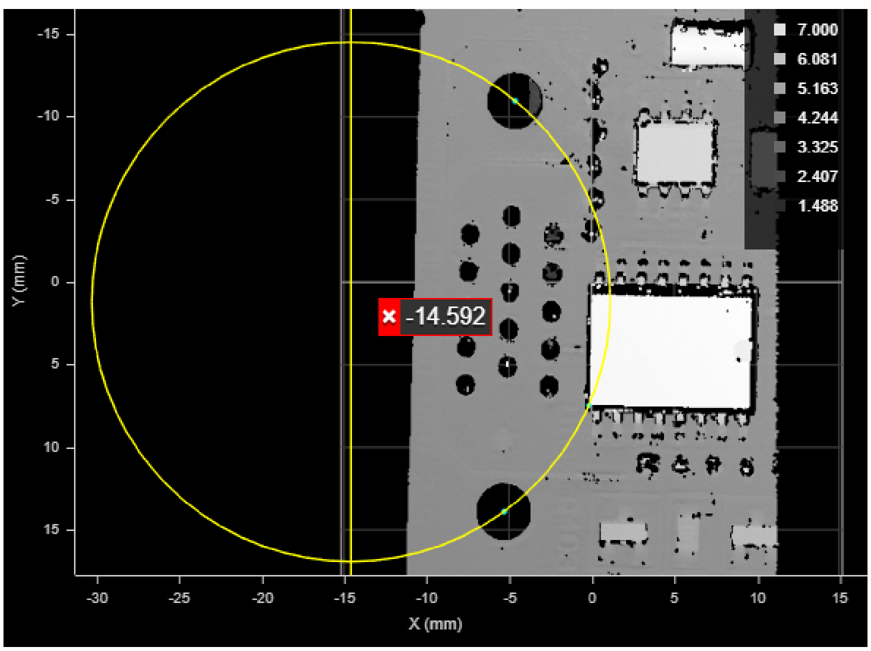
Constant Circle Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z offsets the tool will apply to the generated circle.	**Center X**, **Center Y**, **Center Z** The X, Y, and Z position, respectively, of the center of the constant circle. **Normal X**, **Normal Y**, **Normal Z** The X, Y, and Z for the normal vector. Only displayed when **Project To 2D** is disabled. **Radius** The radius of the constant circle.

Circle from Points

The **Circle from points** output type takes three point geometric features generated by another tool and fits a circle to those points. The circle is always on the XY plane.

When **Project To 2D** is enabled, the circle is projected to the XY plane. When **Project To 2D** is disabled, circle is tilted to fit the plane the three point inputs create.



Circle generated from the center points of the two holes and the corner of the chip (cyan points). (The corner is the intersect point resulting from the Feature Intersect tool, taking the left vertical and lower horizontal line edges of the chip as input.)

The X, Y, and Z measurements return the center of the circle. The X, Y, and Z Angle measurements return arbitrary values.

Circle from points Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z offsets the tool will apply to the generated circle. **Point 1**, **Point 2**, **Point 3** The three Point geometric features the tool uses as the center of the circle.	None.

Circle from point and plane

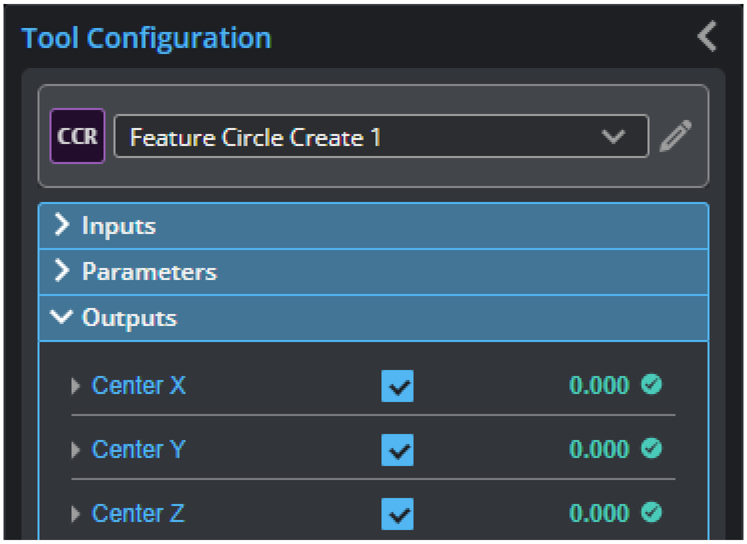
The **Circle from point and plane** output type generates a circle using Point and Plane geometric features and a user-defined radius. The circle is on the Plane received as input.

Circle from point and plane Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z offsets the tool will apply to the generated circle. **Point** The Point geometric features the tool uses as the center of the circle. **Plane** The Point geometric features the tool uses as the center of the circle.	**Radius** The radius of the circle.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

For information specific to the different output types, see *Output Types* on page 694.

Measurements

Measurement
Center X, **Center Y**, **Center Z** The X, Y, and Z positions of the center of the circle.
Radius The radius of the circle.
Z Offset The Z position of the intersection between the circle plane and the Z axis.
Distance The distance from the origin to the circle plane.
Normal X, **Normal Y**, **Normal Z** The X, Y, and Z normal vectores of the circle.

Features

Type	Description
Circle	The generated Circle geometric feature. For more information, see *Output Types* on page 694.

Feature Dimension

The Feature Dimension tool returns dimensional measurements from a point geometric feature to a reference point, line, or plane geometric feature. Geometric features are available as outputs from many tools, for example a Point geometric feature from Surface Position, or a line from Surface Edge. The feature creation tools such as Feature Circle Create, Feature Line Create, and so on, can also create geometric feature. For more information on geometric features, see *Geometric Features* on page 262.

Some examples:

- Measuring the distance between the center of a hole and an edge.
- Measuring the distance between the centers of two holes.
- Measuring the distance between a point and a plane.
- Measuring the distance between a point and the closest point on a circle.
- Obtaining the length of a stud by measuring the distance between its tip and base.

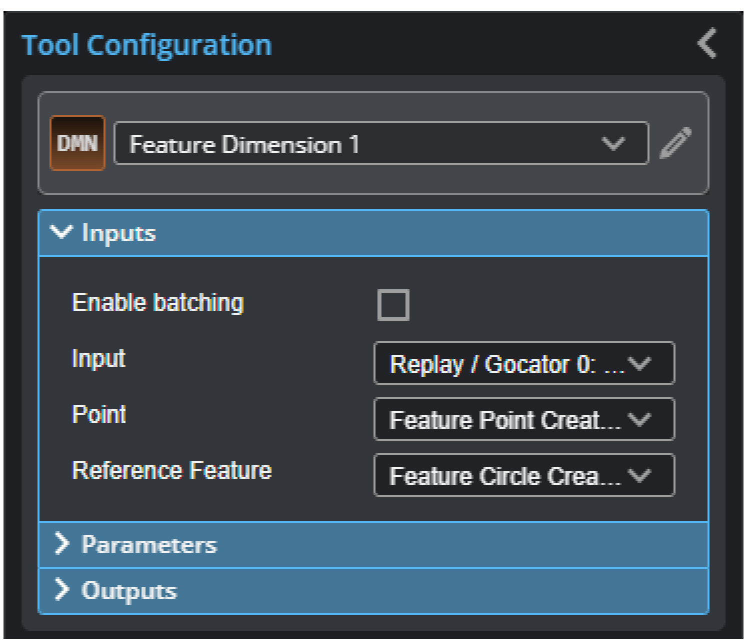
See *Adding a Tool* on page 235 for instructions on how to add measurement tools.

> In the following measurement descriptions, the first geometric feature is set in the **Point** drop-down. The second geometric feature is set in the **Reference Feature** drop-down.

> When **Reference Feature** is set to a feature other than a point, such as a circle or a line, measurements are between the point in **Point** and the *nearest point* on the reference feature (for example, the nearest point on a circle).

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

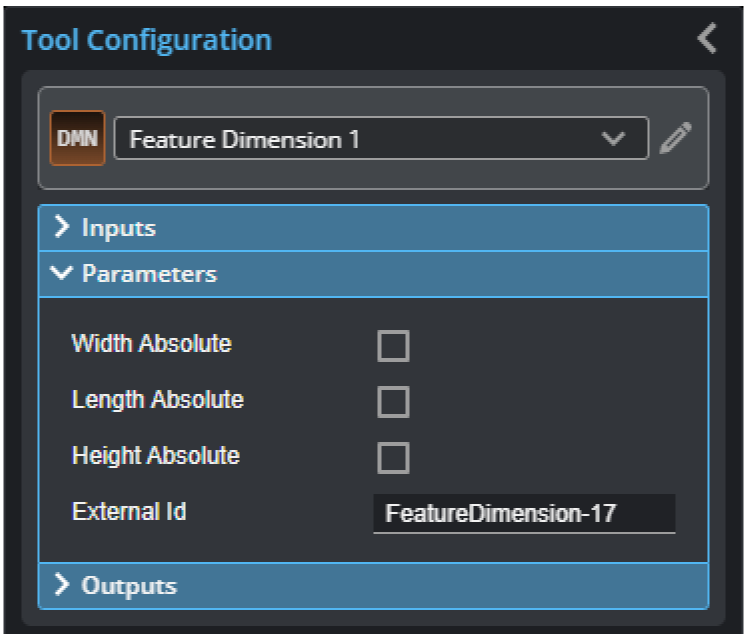


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each element in the array individually. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Input	The data the tool applies measurements to.
Point	A point geometric feature generated by another tool.
Reference Feature	A geometric feature generated by another tool. Dimensional measurements are calculated *from* the reference feature *to* the point in the **Point** input.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

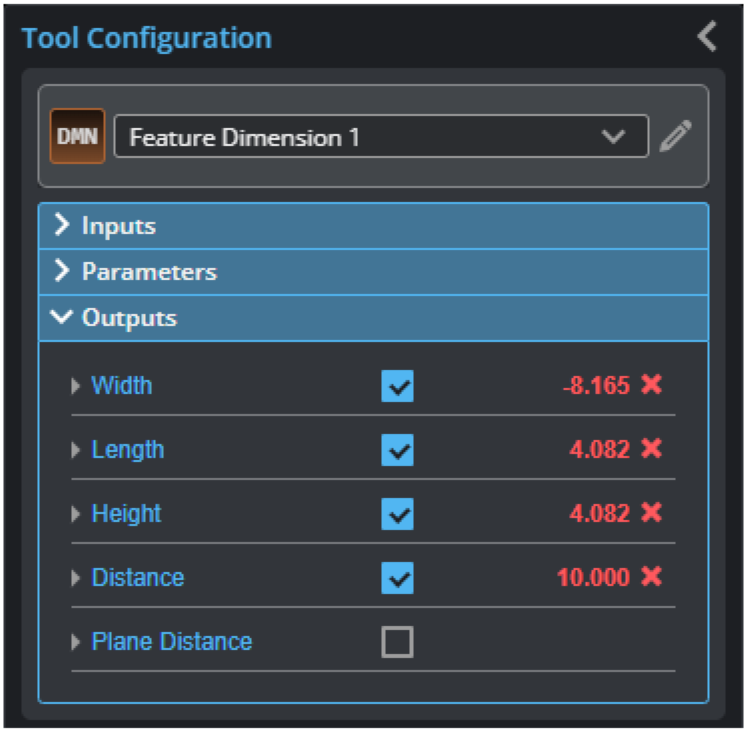


Parameters

Parameter	Description
Width Absolute Length Absolute Height Absolute	The tools returns measurements as absolute values.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

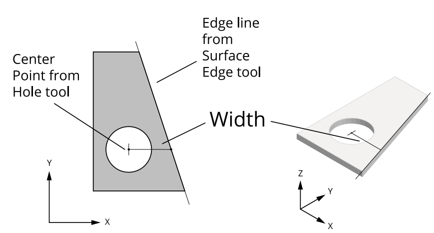
Most tools provide measurements, geometric features, or data as outputs.

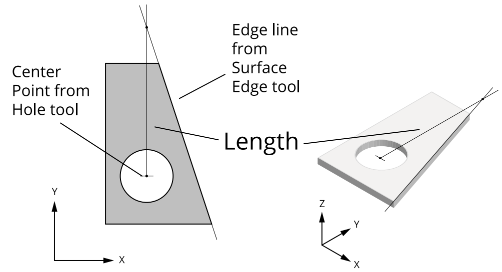
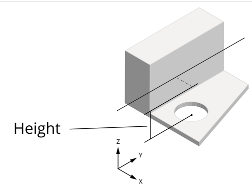
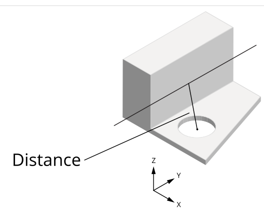
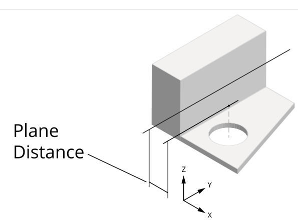


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement	Illustration
Width *Point-point*: The difference on the X axis between the points. *Point-line*: The difference on the X axis between the point and a point on the line. For profiles, the point on the line is at the same Z position as the first point. For surface data, the point on the line is at the same Y position. *Point-plane*: The difference on the X axis between a point and a point on the plane with the same Y and Z coordinates as the first point (or the intersection of the plane and a line from the first point, parallel to the X axis).	

Measurement	Illustration
Length *Point-point*: The difference on the Y axis between the points. *Point-line*: The difference on the Y axis between the point and, for profiles, the nearest point on the line; currently, always zero. For surface data, the point on the line is at the same X position as the first point. *Point-plane*: The difference on the Y axis between the point and a point on the plane with the same X and Z coordinates as the first point (or the intersection of the plane and a line from the first point, parallel to the Y axis).	
Height *Point-point*: The difference on the Z axis between the points. *Point-line*: The difference on the Z axis between the point and, for profiles, a point on the line at the same X position as the first point. For surface data, the point on the line is the one nearest to the first point. *Point-plane*: The difference on the Z axis between the point and a point on the plane with the same X and Y coordinates as the first point (or the intersection of the plane and a line from the first point, parallel to the Z axis).	
Distance *Point-point*: The direct, Euclidean distance between two point geometric features. *Point-line*: The direct, Euclidean distance between a point and the nearest point on the line. *Point-plane*: The direct, Euclidean distance between a point and the nearest point on the plane.	
Plane Distance *Point-point*: The distance between two point geometric features. For profile data, the points are projected onto the XZ plane (always the same as the Distance measurement). For surface data, the points are projected onto the XY plane. *Point-line*: The distance between a point and a line. For profile data, projected onto the XZ plane (always the same as the Distance measurement). For surface data, the distance is projected onto the XY plane.	

Measurement	Illustration
Point-plane: The distance between a point and a plane. For profiles, the distance is projected onto the XZ plane (always the same as the Distance measurement). For surface data, the distance is projected onto the XY plane.	

Feature Line Create

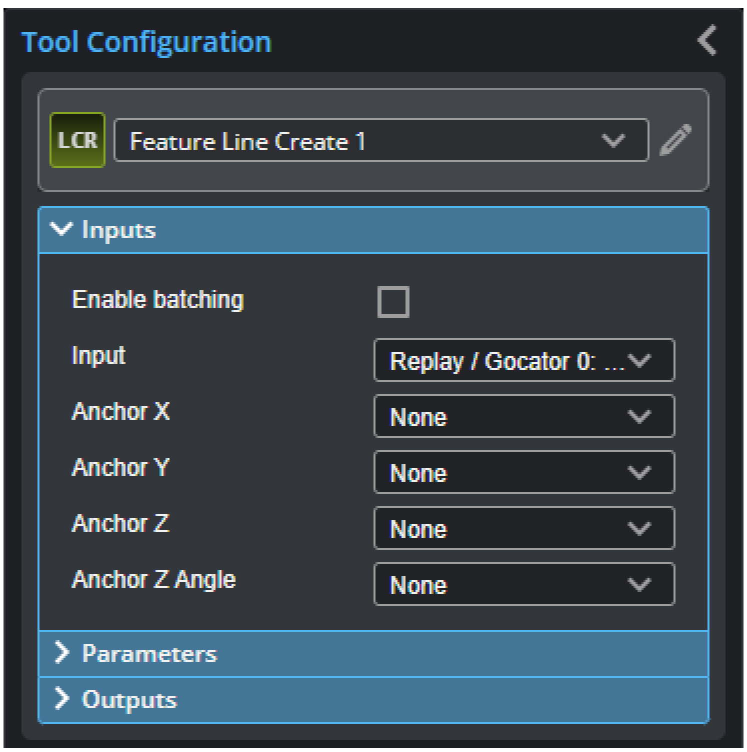
The Feature Line Create tool lets you generate a Line geometric features from geometric features generated by other tools. You can also generate a line from arbitrary positions you provide. The following outputs are available (for more information, see *Output Types* on the next page):

- Line
- Constant line
- Line from two points
- Perpendicular line from point and line
- Perpendicular line from point and plane
- Parallel line from point and line
- Intersect line of two planes
- Projected line on plane
- Bisect projected lines on plane
- Rotate line around point

For more information on geometric features, see *Geometric Features* on page 262.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

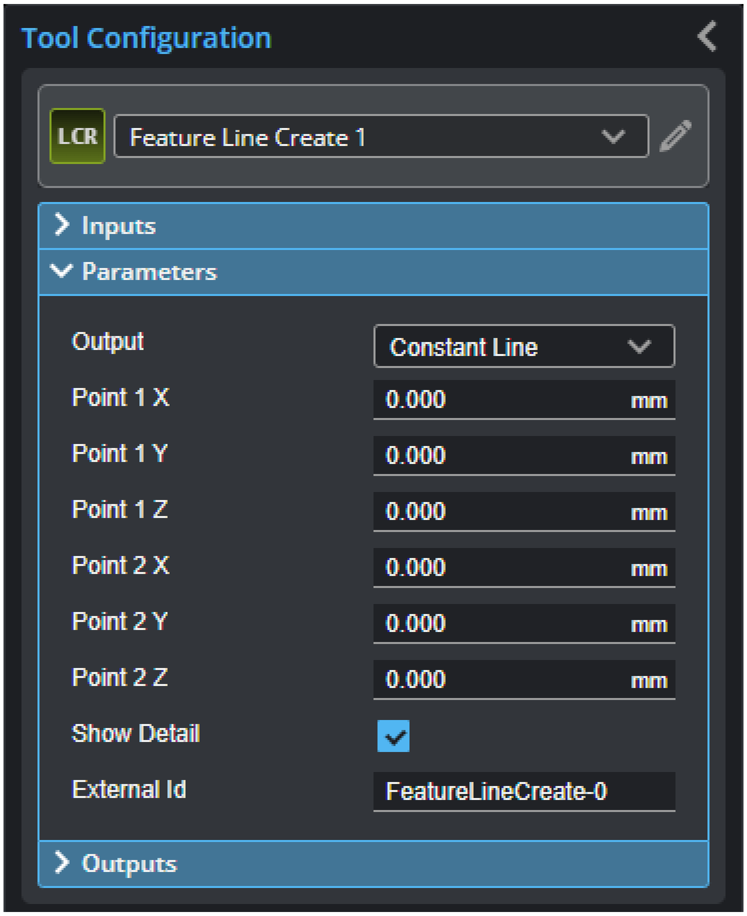


This tool can take an array as input. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.

The tool's inputs depend on the output type you choose in the **Output** parameter. For information on the required inputs and parameters of each output type, see *Output Types* on the next page.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Output	The type of output the tool generates. Switching between the options changes the input types displayed in Input section and the parameters displayed in the Parameters section. For information on the output types and the related parameters, see *Output Types* below.
Show Detail	Toggles the display of the input geometric features in the data viewer.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Output Types

The following sections describe the output types available in the tool's **Output** parameter, in addition to the required inputs and parameters.

Line

The **Line** output type takes a Line geometric feature as input. This output is useful if the tool takes a feature generated by another Feature Create tool as input, on which you want to perform measurements in a second Feature Create tool. This tool can also be useful if you have developed GDK tools that only generate geometric features (no measurements): you can use this tool to extract those measurements.

The X, Y, and Z measurements return the midpoint of the line. The Z Angle measurement returns the angle of the line around the Z axis. The X angle and Y angle measurements return arbitrary values.

Line Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors. **Line** The Line geometric feature the tool uses to create another Line geometric feature.	None.

Constant Line

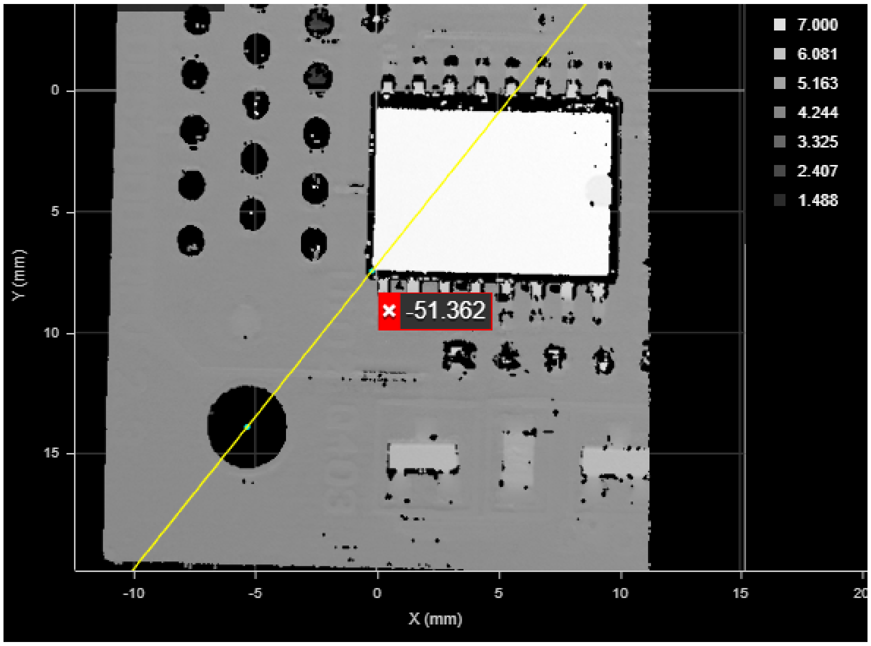
The **Constant line** output type displays parameters you can manually fill in to create a geometric feature. This output type is useful if the scan data from frame to frame is reliably fixed and you want to measure from an arbitrary line to another feature.

Line Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors.	**Point 1 X**, **Point 1 Y**, **Point 1 Z** The X, Y, and Z position, respectively, of the first end point of the constant line. **Point 2 X**, **Point 2 Y**, **Point 2 Z** The X, Y, and Z position, respectively, of the second end point of the constant line.

Line from Two Points

The **Line from two points** output type takes two Point geometric features as input.



A line between the center point of a hole and the corner of the chip.
(The corner is the intersect point resulting from the Feature Intersect tool,
taking the left vertical and lower horizontal line edges of the chip as input.)

The X, Y, and Z measurements return the midpoint of the line. The X, Y, and Z Angle measurements return the angle of the line.

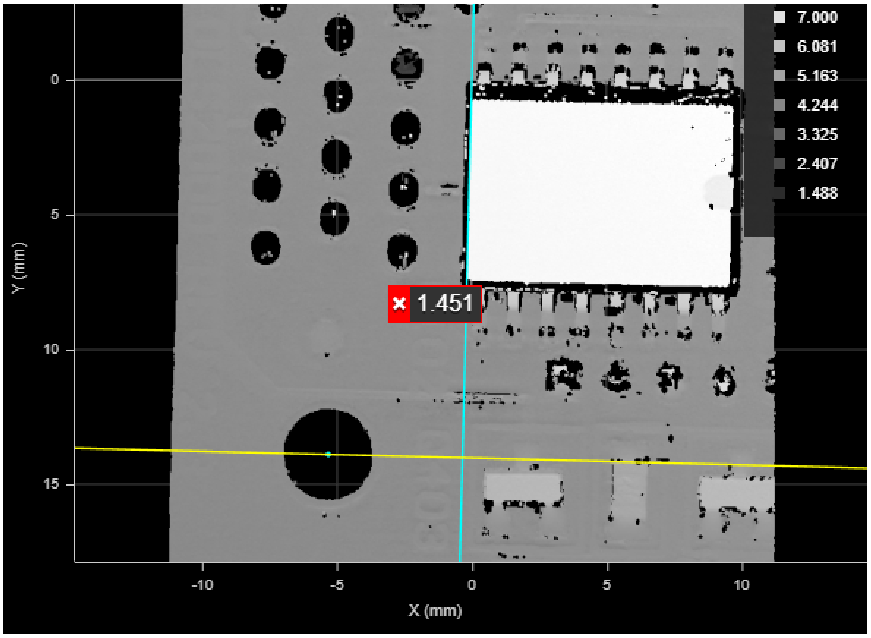
Line from two points Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors. **Point 1**, **Point 2** The two Point geometric features the tool uses as end points to create the line.	None.

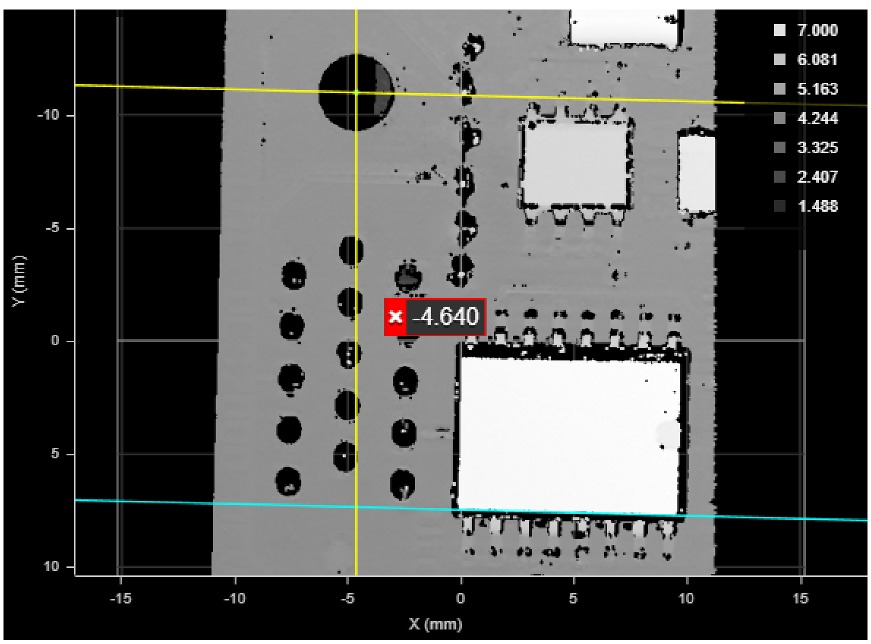
Perpendicular or Parallel Line from Point and Line

The **Perpendicular line from point and line** and **Parallel line from point and line** output types take a Point and a Line geometric feature as input to create another line.

In the following, the tool generates a roughly vertical line (yellow) perpendicular to the input line (cyan line along the left edge of the large integrated circuit), passing through the input point (cyan dot at the center of the hole).



In the following, the tool generates a roughly horizontal line (yellow) parallel to the input line (cyan line along the bottom edge of the large integrated circuit), passing through the input point (cyan dot at the center of the hole).



For both of these types of line output, the X, Y, and Z measurements return the position of the point.

For perpendicular line output, the X, Y, and Z angle measurements return the angles of the line.

For parallel line output, the Z angle measurement returns the angle of the line; the X and Y angle measurements both return arbitrary values.

Line from Point and Line Output Types - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors. **Point** The Point geometric feature the tool uses to create the line. **Line** The Line geometric feature the tool uses to create the line.	None.

Perpendicular Line from Point to Plane

The **Perpendicular line from point to plane** output type generates a perpendicular line from a point up to a plane.

Perpendicular line from point to plane Output Types - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors. **Point** The Point geometric feature the tool uses to create the line. **Plane** The Plane geometric feature the tool uses to create the line.	None.

Intersect Line of Two Planes

The **Intersect line of two planes** output type takes two plane geometric features as input and creates a line at their intersection.

The X, Y, and Z measurements return the midpoint. The X, Y, and Z Angle measurements return the angle of the line.

Intersect line of two planes Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input.	None.
Anchor X, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors.	
Plane 1, **Plane 2** The two Plane geometric feature the tool uses to create the line.	

Projected Line on Plane

The **Projected line on a plane** output type takes a Line and a Plane geometric feature to create a line projected onto a plane.

Projected line on a plane Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input.	**Projection Direction** One of the following: **Perpendicular to Plane** or **Along Z Axis**.
Anchor X, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors.	
Line The Line geometric feature the tool uses to create the line.	
Plane The Plane geometric feature the tool uses to create the line.	

Projected Center Line from Two Lines and a Plane

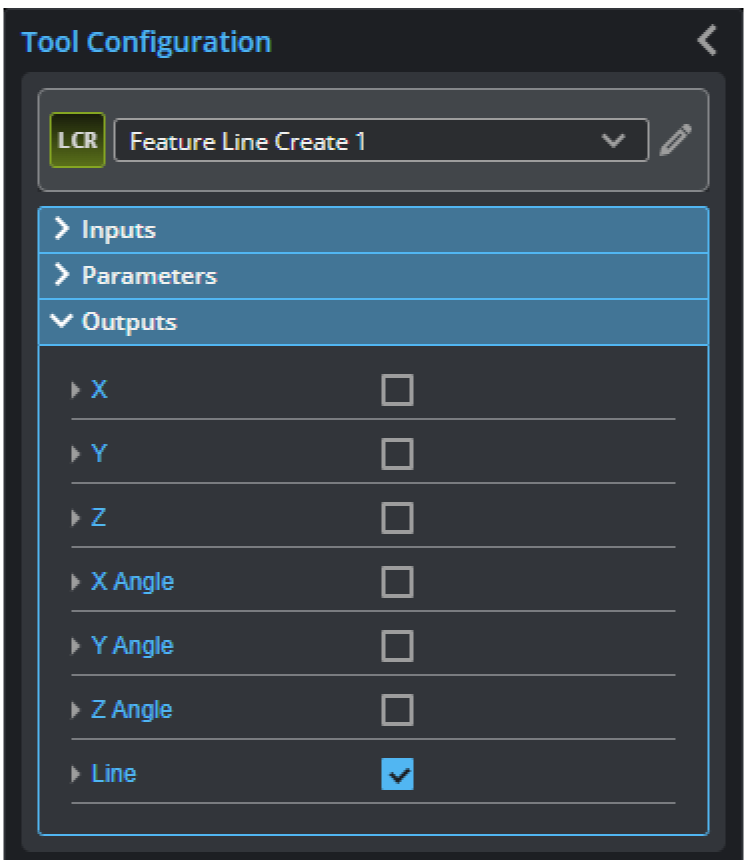
The **Projected Center Line from Two Lines and a Plane** output type takes creates a two Line and a Plane geometric feature as input and creates a line.

Bisect projected lines on plane Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z**, **Anchor Z Angle** The X, Y, and Z position measurements and the Z Angle measurement the tool uses as anchors. **Line** The Line geometric feature the tool uses to create the line. **Plane** The Plane geometric feature the tool uses to create the line.	None.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

For information specific to the different output types, see *Output Types* on page 704.

Measurements

Measurement
X, **Y**, **Z** The X, Y, and Z positions of the midpoint of the line described by the Point geometric features. For more information, see the sections above.
X Angle, **Y Angle**, **Z Angle** The X, Y, and Z angles of the line. For more information, see the sections above.

Features

Type	**Description**
Line	The generated Line geometric feature.

Feature Plane Create

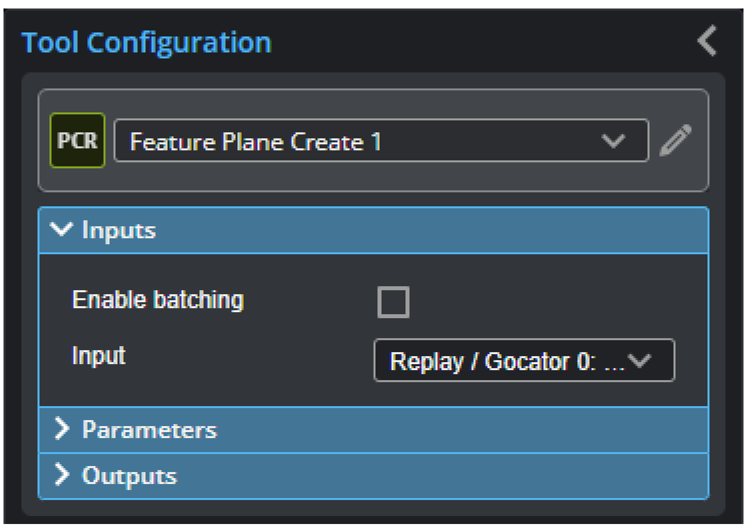
The Feature Plane Create tool lets you generate a Plane geometric features from geometric features generated by other tools. You can also generate a plane from arbitrary values you provide. The following outputs are available (for more information, see *Output Types* on the next page):

- Constant plane
- Plane from point and normal
- Plane from point and line
- Plane from three points
- Plane from circle
- Parallel plane from point and plane
- Perpendicular plane from point and plane
- Perpendicular plane from line and plane
- Bisect plane from two planes

For more information on geometric features, see *Geometric Features* on page 262.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

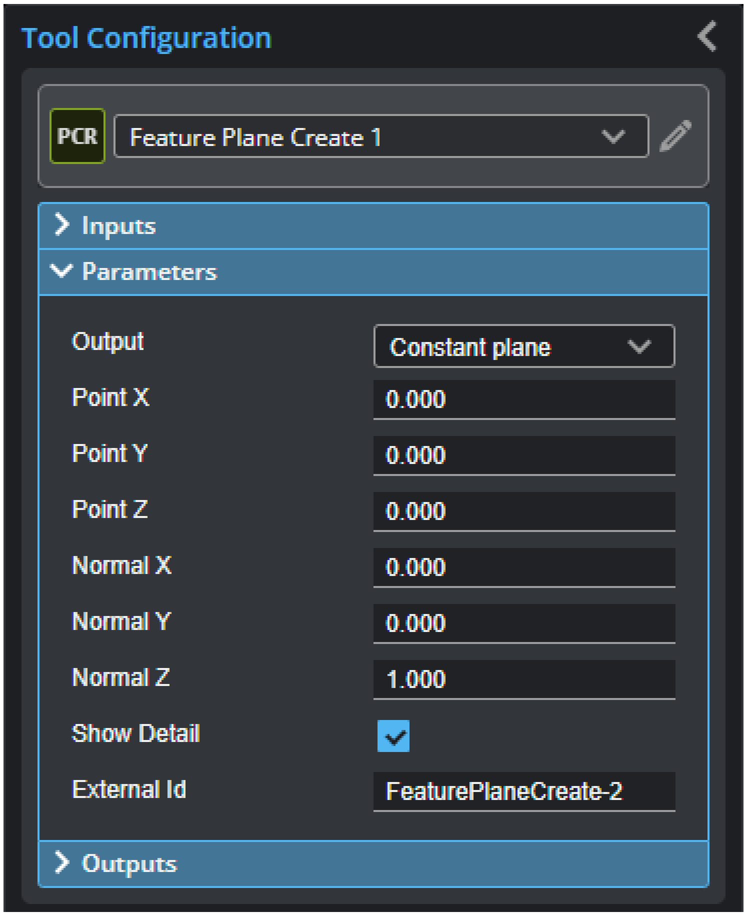


This tool can take an array as input. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.

The tool's inputs depend on the output type you choose in the **Output** parameter. For information on the required inputs and parameters of each output type, see *Output Types* on the next page.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Output	The type of output the tool generates. Switching between the options changes the input types displayed in Input section and the parameters displayed in the Parameters section. For information on the output types and the related parameters, see *Output Types* below.
Show Detail	Toggles the display of the input geometric features in the data viewer.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Output Types

The following sections describe the output types available in the tool's **Output** parameter, in addition to the required inputs and parameters.

Constant Plane

The **Constant plane** output type is useful if scan data from frame to frame is reliably fixed and you want to measure from an arbitrary plane to another feature. When you choose this output type, the tool displays the parameters listed below you can manually fill in to create a constant plane.

Constant Plane Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input.	**Point X**, **Point Y**, **Point Z** The X, Y, and Z points that define the plane. **Normal X**, **Normal Y**, **Normal Z** The X, Y, and Z normals, respectively, of the constant plane.

Plane from a Point and Normal

The tool uses the Line geometric feature as the normal of a plane, and positions the plane so that it intersects the Point geometric feature.

Plane from Point and Normal Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Point** The Point geometric feature with which the plane intersects. **Line** A Line geometric feature representing the normal that the tool uses to create the plane.	None.

Plane from a Point and Line

The tool uses Line and Point geometric features to create a plane.

Plane from Point and Line Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Point** The Point geometric feature with which the plane intersects. **Line** A Line geometric feature with which the plane intersects.	None.

Plane from Three Points

The **Plane from three points** output type creates a plane that intersects the three Point geometric features passed to the tool as input.

Plane from Three Points Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Point 1**, **Point 2**, **Point 3** The three Point geometric features the tool uses to create the plane.	None.

Plane from Circle

The **Plane from circle** output type creates a plane that intersects the Circle geometric feature passed to the tool as input.

Plane from Three Points Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Circle** The Circle geometric features the tool uses to create the plane.	None.

Parallel Plane from Point and Plane

The **Parallel plane from point and plane** output type creates a plane parallel to the provided Plane geometric feature that intersects the provided Point geometric feature.

Parallel Plane from Point and Plane Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Point** The Point geometric feature with which the created plane intersects. **Plane** The Plane geometric feature with which the created plane is parallel.	None.

Perpendicular Plane from Point and Plane

The **Perpendicular plane from point and plane** output type creates a plane perpendicular to the provided Plane geometric feature that intersects the two provided Point geometric features.

Perpendicular Plane from Point and Plane Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Point 1**, **Point 2** The Point geometric features with which the created plane intersects. **Plane** The Plane geometric feature with which the created plane is perpendicular.	None.

Perpendicular Plane from Line and Plane

The **Perpendicular plane from line and plane** output type creates a plane perpendicular to the provided Plane geometric feature, which intersects the provided Line geometric feature.

Perpendicular Plane from Line and Plane Output Type - Inputs and Parameters

Inputs	Type-Related Parameters
Input The Surface or Profile input. **Line** The Line geometric feature that bisects the created plane. **Plane** The Plane geometric feature with which the created plane is perpendicular.	None.

Bisect Plane from Two Planes

The **Bisect plane from two planes** output type creates a plane from the two provided Plane geometric features.

Bisect Plane from Two Planes Output Type - Inputs and Parameters

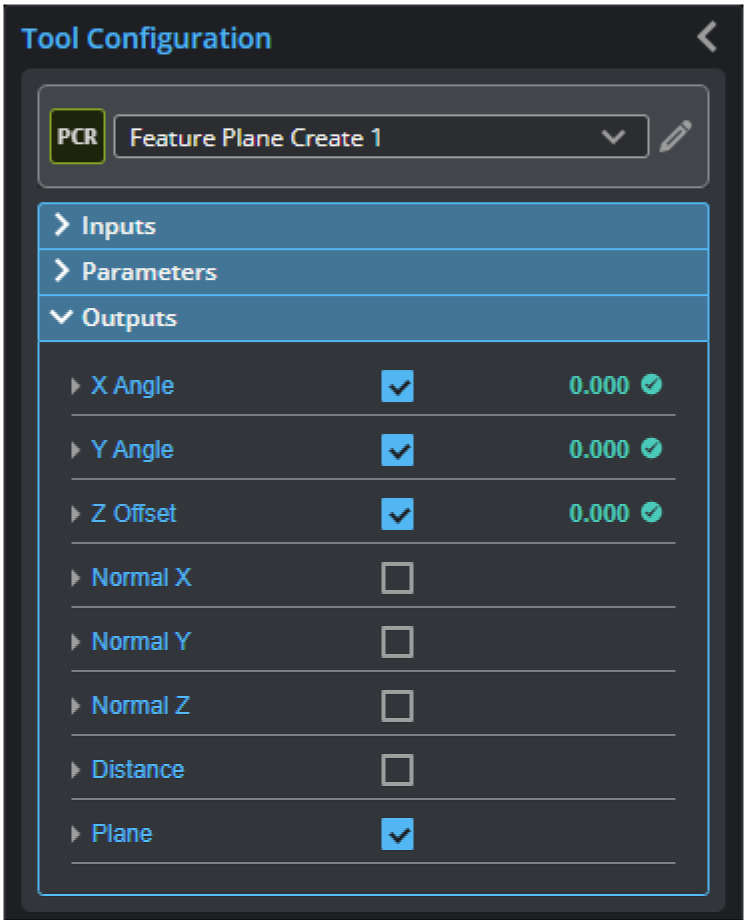
Inputs	Type-Related Parameters
Input The Surface or Profile input.	None.

Inputs	Type-Related Parameters
Plane 1, **Plane 2** The Plane geometric features the tool uses to create a bisecting plane	

See *Adding a Tool* on page 235 for instructions on how to add measurement tools.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

All outputs provide an external ID (available by expanding the output in the Outputs panel) for optional use in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

For information specific to the different output types, see *Output Types* on page 713.

Measurements

Measurement
X Angle, **Y Angle** Rotation angles of the plane, that is, angles such that rotating (0,0,1) around the Y axis by the Y Angle, and then around X axis by X Angle produces the normal.
Z Offset The Z value of intersection of the plane and the Z axis.
Normal X, **Normal Y**, **Normal Z** The normal X, Y, and Z vectors of the plane.
Distance The distance from the origin to the plane.

Features

Type	Description
Plane	The generated Plane geometric feature. For more information, see *Output Types* on page 713.

Feature Point Create

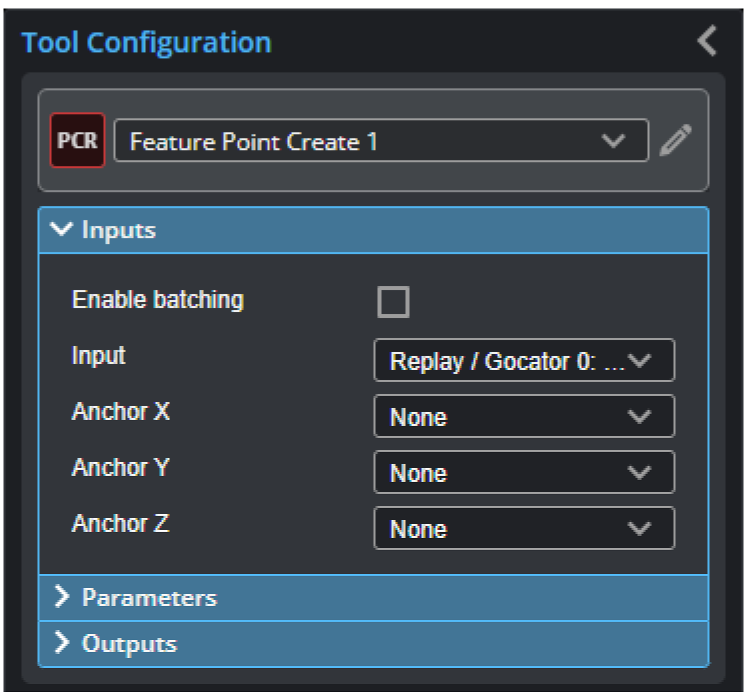
The Feature Point Create tool lets you generate a Point geometric features from geometric features generated by other tools. You can also generate a point from an arbitrary position you provide. The following outputs are available (for more information, see *Output Types* on the next page):

- Point
- Constant point
- Point from offset
- Point from three planes
- Point from line and circle
- Point from line and line
- Point from line and plane
- Projected point on line
- Projected point on plane

For more information on geometric features, see *Geometric Features* on page 262.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

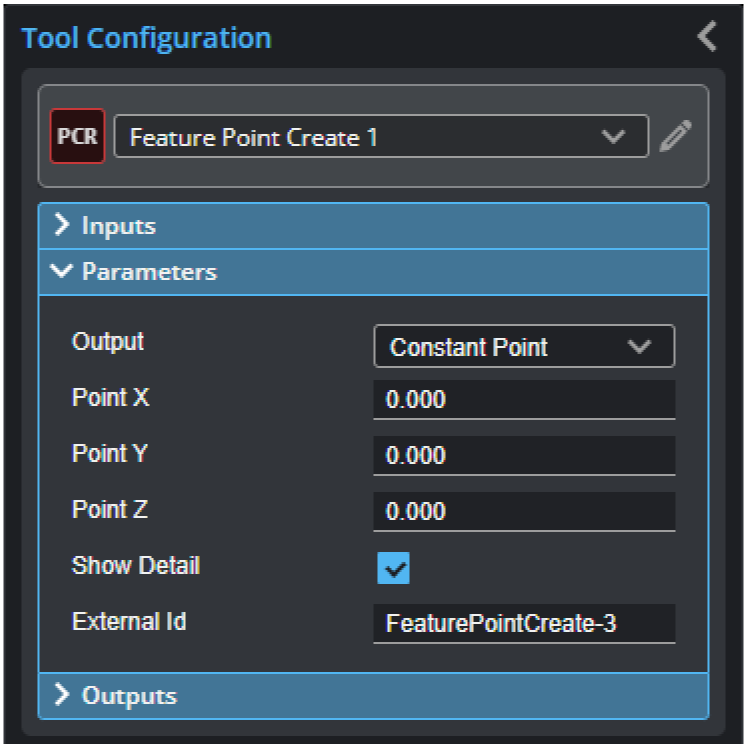


This tool can take an array as input. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.

The tool's inputs depend on the output type you choose in the **Output** parameter. For information on the required inputs and parameters of each output type, see *Output Types* on the next page.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Output	The type of output the tool generates. Switching between the options changes the input types displayed in Input section and the parameters displayed in the Parameters section. For information on the output types and the related parameters, see *Output Types* below.
Show Detail	Toggles the display of the input geometric features in the data viewer.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Output Types

The following sections describe the output types available in the tool's **Output** parameter, in addition to the required inputs and parameters.

Point

The **Point** output type takes a Point geometric feature as input. This output is useful if the tool takes a feature generated by another Feature Create tool as input, on which you want to perform measurements in a second Feature Create tool. This tool can be useful if you have developed GDK tools that only generate geometric features (no measurements): you can use this tool to extract those measurements.

Point Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors. **Point** The Point geometric feature the tool uses to create another Point geometric feature.	None.

Constant Point

The **Constant point** output type displays parameters you can manually fill in to create a geometric feature. This output type is useful if the scan data from frame to frame is reliably fixed and you want to measure from an arbitrary point to another feature.

Constant Point Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors.	**Point X**, **Point Y**, **Point Z** The X, Y, and Z position, respectively, of the first end point of the constant line.

Point from Offset

The **Point from offset** output type takes two Point geometric feature as input, and generates a point on the line connecting the two input Point features. You specify the offset direction and the distance along the line.

Point from Offset Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors. **Point 1**, **Point 2**	**Offset Direction** The direction along the line running through the two input Point geometric features that the point is offset. One of two values: Forward or Backward. A Forward offset direction offsets the point along the line moving from Point 1 to Point 2. A Backward offset direction offsets the point along the line moving from Point 2 to Point 1.

Inputs	Parameters
The Point geometric features the tool uses to create the line along which the offset Point geometric feature is placed.	**Offset Distance** The distance along the line running through the Point geometric features that the point is offset.

Point from Three Planes

The **Point from three planes** output type takes three Plane geometric features as input and generates a Point geometric feature at the intersection of those planes.

Point from Three Planes Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors. **Plane 1**, **Plane 2**, **Plane 3** The Plane geometric features the tool uses to generate the Point geometric feature. The point is at the intersection of the planes.	None.

Point from Line and Circle

The **Point from line and circle** output type takes a Line and a Circle geometric feature as input and generates a Point geometric feature where the line intersects the circle. The tool uses an intersection option to decide which intersection point to use.

Point from Line and Circle Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors. **Line** The Line geometric feature the tool uses to generate the Point geometric feature. **Circle**	**Intersection Option** The intersection between the line and the circle that the tool uses to generate the Point geometric feature. One of the following: Upper, Lower, or Middle. **Upper**: The uppermost intersection point in the data viewer. **Lower**: The lowermost intersection point in the data viewer. **Middle**: The midpoint between the two intersection

Inputs	Parameters
The Circle geometric features the tool uses to generate the Point geometric feature.	points.

Point from Line and Line

When the **Output** parameter is set to **Point from line and line**, the tool takes two Line geometric feature as input and generates a Point geometric feature where the lines intersect. The tool also returns an angle measurement for the angle between the lines.

Point from Line and Line Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input.	**Angle Range** The range returned by the XY Angle measurement. One of the following: "-180 to 180" or "0 to 360".
Anchor X, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors.	
Line 1, **Line 2** The Line geometric features the tool uses to generate the Point geometric feature.	

Point from Line and Plane

When the **Output** parameter is set to **Point from line and plane**, the tool takes a Line and a Plane geometric feature as input and generates a Point geometric feature where the lines intersect. The tool also returns an angle measurement for the angle between the lines.

Point from Line and Plane Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input.	**Angle Range** The range returned by the Angle measurement. One of the following: "-180 to 180" or "0 to 360".
Anchor X, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors.	
Line The Line geometric features the tool uses to generate the Point geometric feature.	
Plane The Plane geometric features the tool uses to generate	

Inputs	Parameters
the Point geometric feature.	

Projected Point on Line

The **Projected point on line** output type takes a Point and a Line geometric feature as input and generates a Point geometric feature.

Projected Point on Line Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors. **Point** The Point geometric features the tool uses to generate the Point geometric feature. **Line** The Line geometric features the tool uses to generate the Point geometric feature.	None.

Projected Point on Plane

The **Projected point on plane** output type takes a Point and a Plane geometric feature as input and generates a Point geometric feature.

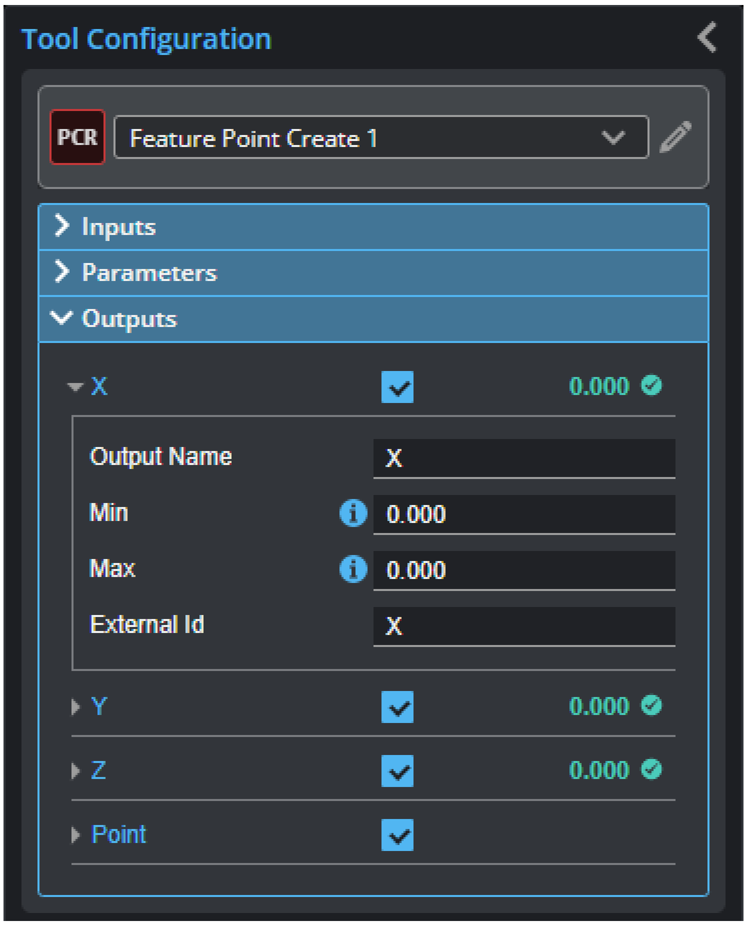
Projected Point on Plane Output Type - Inputs and Parameters

Inputs	Parameters
Input The Surface or Profile input. **Anchor X**, **Anchor Y**, **Anchor Z** The X, Y, and Z position measurements the tool uses as anchors. **Point** The Point geometric features the tool uses to generate the Point geometric feature.	**Projection Direction** The direction of the projection. One of the following: Perpendicular to Plane or Along Z Axis.

Inputs	Parameters
Plane The Plane geometric features the tool uses to generate the Point geometric feature.	

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

For information specific to the different output types, see *Output Types* on page 720.

Measurements

Measurement
X, Y, Z The X, Y, and Z positions of the Point geometric feature. For more information, see the sections above.

Measurement
XY Angle The angle measurement.

Features

Type	Description
Point	The generated Point geometric feature.

Feature Intersect

The Feature Intersect tool returns the intersection of a line or plane geometric feature and a second line or plane geometric feature. For line-line intersections, the lines are projected onto the Z = reference Z line plane for features extracted from a surface, and the intersection of the lines projected onto the Y = 0 plane for features extracted from a profile. The angle measurement between the two lines is also returned. The lines the tool takes as input are generated by other tools, such as Surface Edge or Surface Ellipse.

The Feature Intersect tool's positional measurements are particularly useful as anchor sources. For example, you can easily find a corner point on a part from two edges (produced by two Surface Edge tools) and using the X and Y positions as anchor sources.

When you use these positional anchors in combination with a Z Angle anchor from tools such as Surface Edge, you can achieve extremely robust, repeatable measurements.

For more information on anchoring, see *Measurement Anchoring* on page 264.

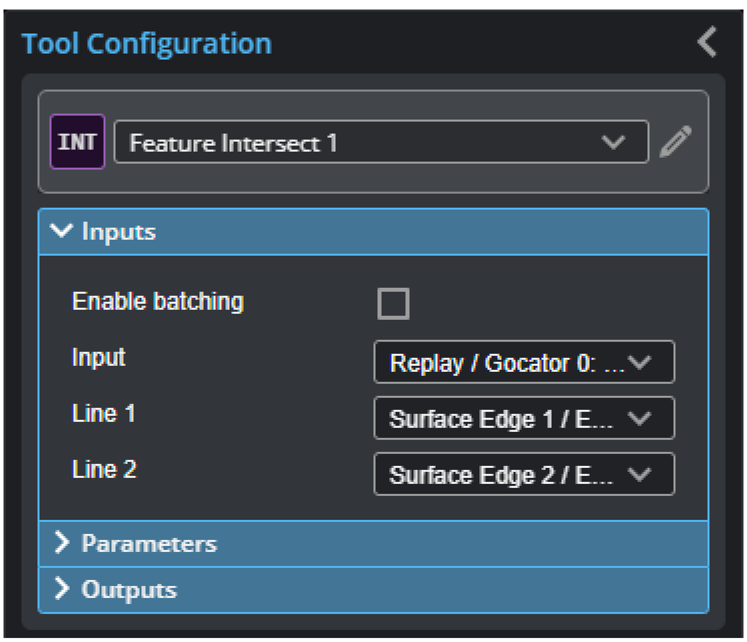
The Feature Intersect tool can also generate a point geometric feature representing the point of intersection of the lines that the Feature Dimension tool can use in measurements.

See *Adding a Tool* on page 235 for instructions on how to add measurement tools.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

The tool's inputs depend on the intersection type you choose in the **Intersection Type** parameter. For information on the required inputs and parameters of each output type, see *Feature Intersect* above.

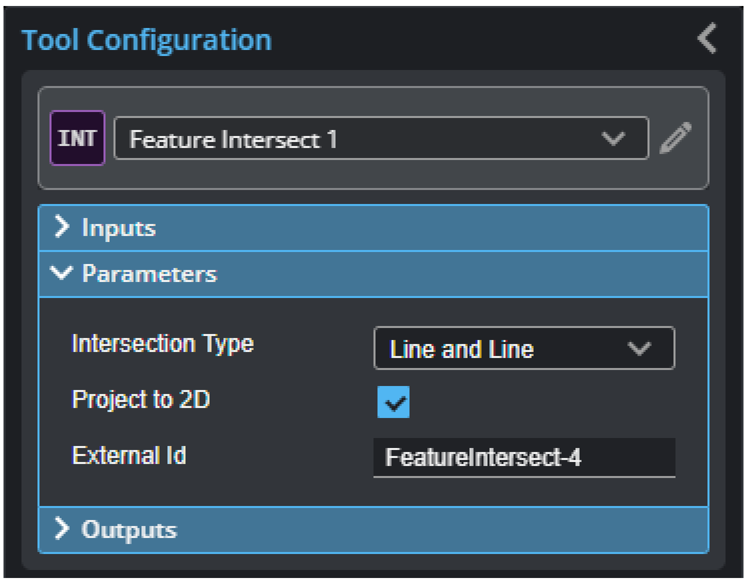


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each element in the array individually. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Input	The data the tool applies measurements to or processes.
Line or Plane	The geometric features the tool uses to calculate an intersect. A combination of Line or Plane geometric features, depending on the **Intersection Type** parameter.

Parameters

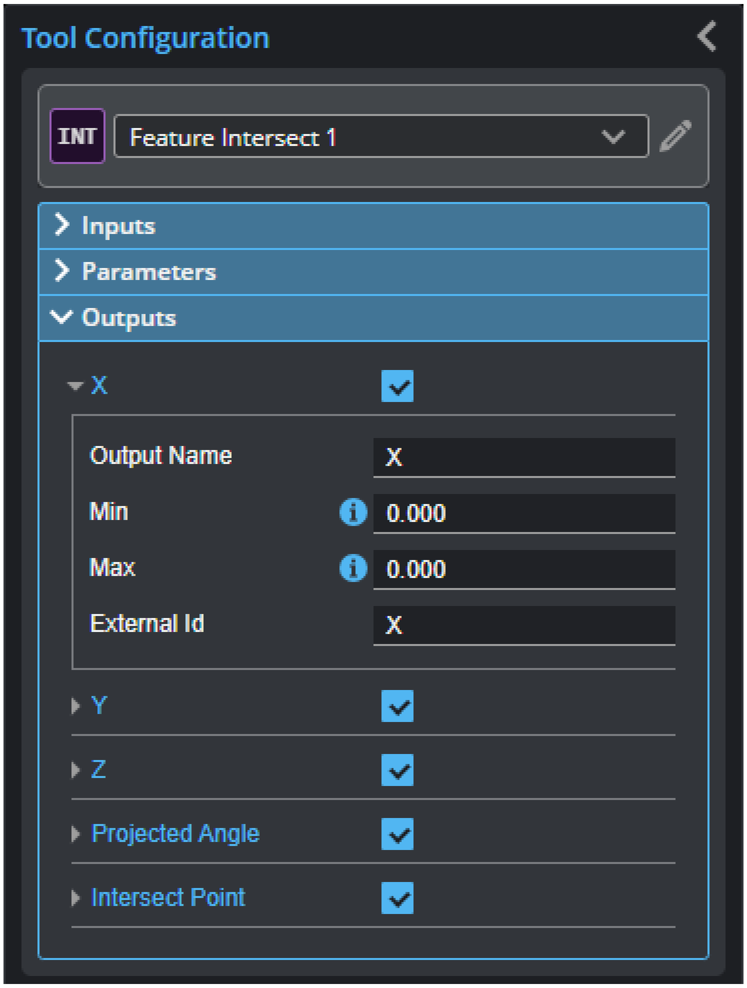
You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Parameter	Description
Intersection Type	One of the following: • Line and Line • Line and Plane • Plane and Plane The intersection type determines which inputs are available in the **Inputs**.
Project to 2D	Only shows when Intersection Type is “Line and Line”. If checked, the intersection angle is projected to the XZ plane for Profile mode and XY plane for Surface mode. Otherwise, outputs the 3D angle.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

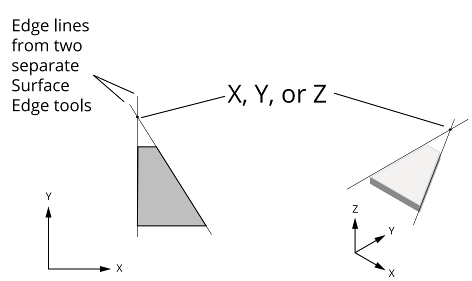


Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

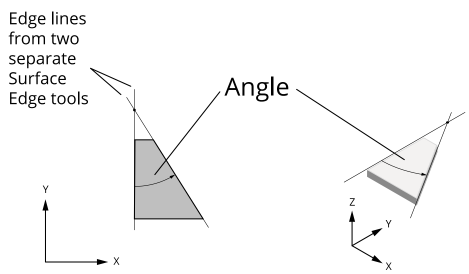
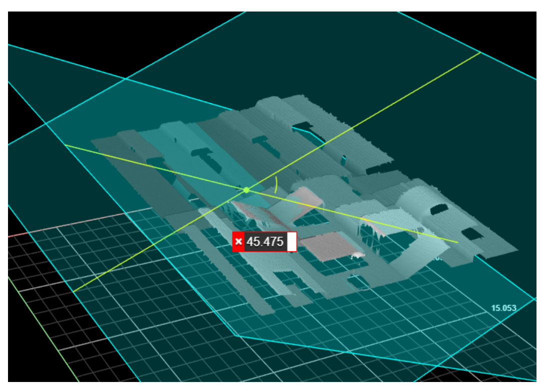
You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

<table>
<tr><th>Measurement</th><th>Illustration</th></tr>
<tr><td>X
Line-Line: The X position of the intersect point between the lines.
Line-Plane: The X position of the intersect point between the line and the plane.
Plane-Plane: The X position of the center of the line intersecting the planes.</td><td rowspan="2">



</td></tr>
<tr><td>Y
Line-Line: The Y position of the intersect point between</td></tr>
</table>

Measurement	Illustration
the lines. *Line-Plane*: The Y position of the intersect point between the line and the plane. *Plane-Plane*: The Y position of the center of the line intersecting the planes.	
Z *Line-Line*: The Z position of the intersect point between the lines. *Line-Plane*: The Z position of the intersect point between the line and the plane. *Plane-Plane*: The Z position of the center of the line intersecting the planes.	
Angle or **Projected Angle** (*Line-Line*: The angle between the lines, as measured from the line selected in **Reference Feature** to the line selected in **Line**. If **Project To 2D** is enabled, the output is Projected Angle. *Line-Plane*:The angle between the line and the perpendicular projection of the line onto the plane, as measured from the plane geometric feature selected in **Reference Feature** to the line selected in **Line**. *Plane-Plane*:The angle between the two planes, as measured from the plane geometric features selected in **Feature 1** and **Feature 2**.	 In the following image, the angle is measured between two planes (the small angled surfaces facing each other in the center of the image). 

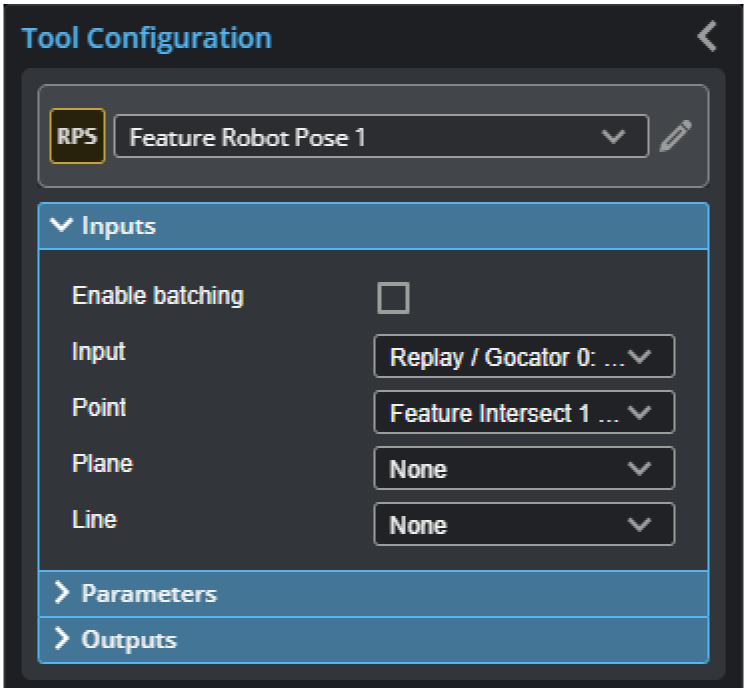
Features

Type	Description
Intersect Point	The intersect point of the two features.
Intersect Line	The line representing the intersection of two planes.

Feature Robot Pose

The Feature Robot Pose tool takes geometric features as input and outputs positional and rotational values. You can use these values in a robot system to control the robot.

Inputs



To use a measurement as an anchor, it must be enabled and properly configured in the tool providing the anchor. For more information on anchoring, see *Measurement Anchoring* on page 264.

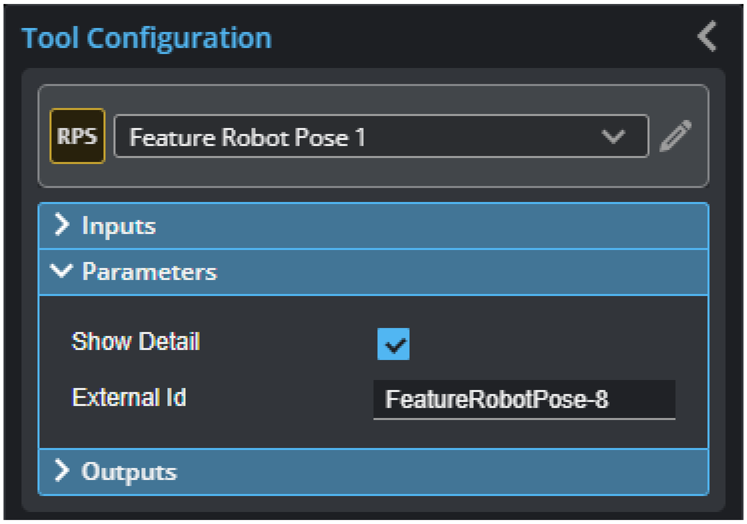
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each element in the array individually. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Input	The data the tool applies measurements to or processes.
Point	The Point geometric feature the tool extracts the X, Y, and Z measurements from. This input is required.
Plane	The Plane geometric feature the tool extracts the Roll and Pitch measurements from. This input is optional. If you omit it, the tools uses the XY plane for calculations.
Line	The Line geometric feature the tool extracts the Yaw measurement from. This input is optional. If you omit it, the tool uses the X axis for calculations.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.

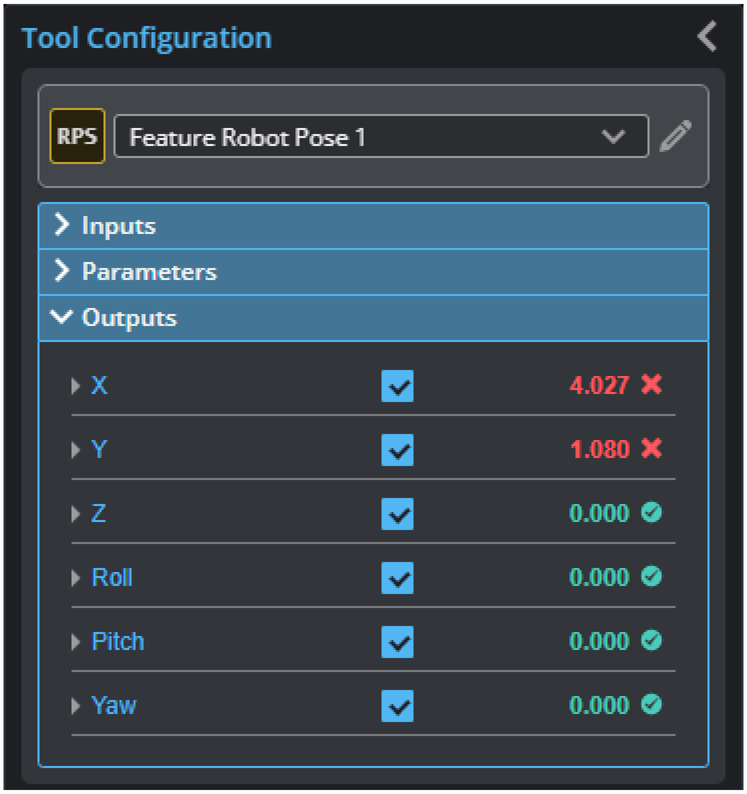


Parameters

Parameter	Description
Show Detail	Toggles the display of additional visualizations in the data viewer.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Measurements

Measurement	Illustration
X, **Y**, **Z** The X, Y, and Z positions of the Point geometric feature.	
Roll, **Pitch**, **Yaw** The rotational angles of the Plane and Line geometric feature.	

Data Export

The Data Export tool saves frames of scan data to local PC storage (by default, `C:\GoTools\DataExport`), letting you import and evaluate scan data in 3rd-party applications.

> This tool is only available in a PC instance of GoPxL. It is not available on a sensor or on GoMax NX.

You can optionally use a measurement as a condition for data export. For example, you could configure the tool to only export data if a Surface Position's Z measurement is a Fail (or a Pass).

The tool supports exporting the following formats. The available formats depend on the tool's input.

Output formats

Format	Surface	Profile	Image	Description
CSV (.csv)	✓	✓		CSV format. For more information, see *CSV File Format* on page 939.
PRO (.pro)		✓		MountainsMap format. Uniform data only.
Wavefront OBJ (.obj)	✓			Mesh with comma-separated vertices and facets in text format.
ODSCAD OMC (.omc)	✓			ODSCAD heightmap.
PCL PCD (.pcd)	✓			Point Cloud Data (PCD) file format for the Point Cloud Library (PCL).
PNG (.png)	✓		✓	**Surface** The heightmap as a PNG image. The image is either a 24-bit RGB image or a 16-bit grayscale image, depending on whether **16-bit Depth** is enabled (see below). **Image** An image from the sensor camera. The bit-depth depends on the camera.
STL ASCII (.stl)	✓			Mesh in the standard STL text format (can become very large).
STL Binary (.stl)	✓			Mesh in the binary STL format.
MountainsMap SUR (.sur)	✓			MountainsMap SUR heightmap.
TIFF (.tif)	✓		✓	**Surface** The heightmap as a TIFF image. The image is either a 24-bit RGB image or a 16-bit grayscale image, depending on whether **16-bit Depth** is enabled (see below).

Format	Surface	Profile	Image	Description
				Image An image from the sensor camera. The bit-depth depends on the camera.

The file naming convention is as follows:

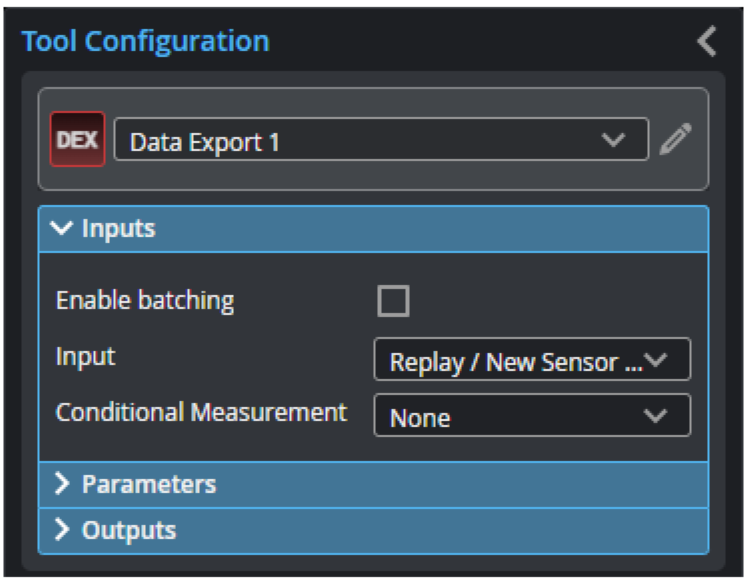
```
<frame>_<timestamp>_<counter>.<extension>
```

<counter> is optional: It is only present if a file with the same name is already in the output folder.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.



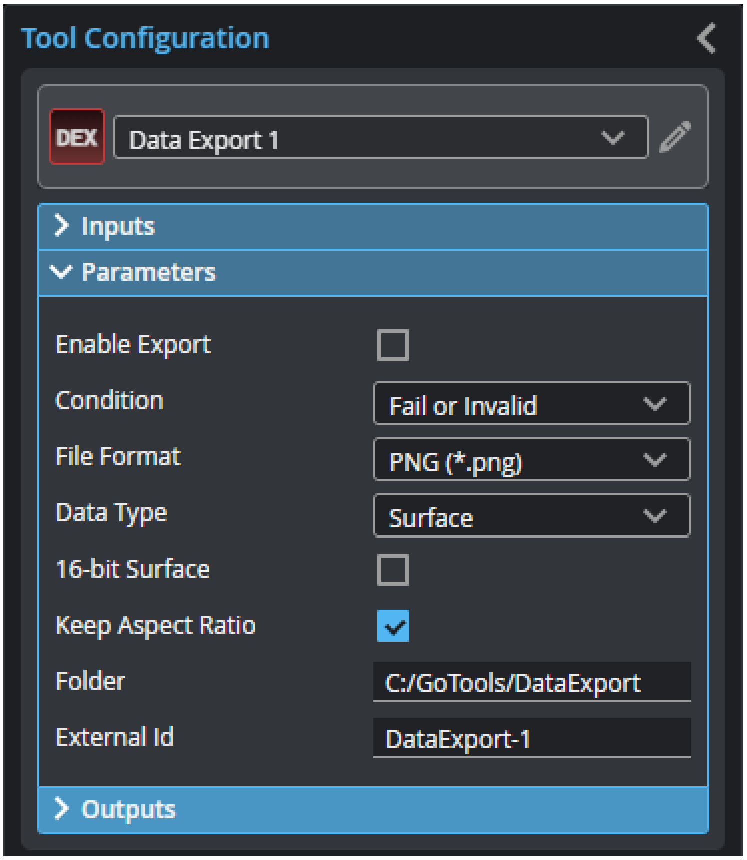
Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and exports the data for each item in the array individually. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Input	The data the tool exports.

Name	Description
	This tool can optionally take an array as input. For more information, see Arrays, Batching, and Aggregation.
Conditional Measurement	The measurement the tool uses to determine whether to export data. Can be None, in which case the tool exports all frames. In the **Parameters** section, set **Condition** to the condition type (see below).

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Name	Description
Enable Export	When this parameter is checked, the tool exports frames in the format you select in **File Format** to the folder specified in **Folder**.
Condition	If the **Conditional Measurement** input is set to a measurement, this determines which state of the measurement causes the tool to export data. One of the following: • Fail or Invalid • Invalid • Pass

Name	Description
	The setting is only displayed if the **Conditional Measurement** input is set to a measurement.
File Format	The format the tool uses to export data. For a list of the formats, see *Output formats* on page 734.
Data Type	The type of data the tool outputs, using the format selected in **File Format**. One of the following: **Surface** Outputs Surface data. **Intensity** If intensity data is available, the tool outputs this data. Otherwise, no output is produced. **Surface and Intensity** Outputs Surface data and, if it is available, intensity data. The tool appends "_surface" and "_intensity" to the file names of the outputs.
16-bit Surface	The bit depth of the heightmap output as an image. When disabled, the tool outputs a 24-bit RGB image. When enabled, the tool outputs a 16-bit grayscale image. Only available for PNG and TIFF outputs.
Keep Aspect Ratio	Preserves the aspect ration of the image outputs (PNG and TIFF) by resampling to equal XY spacing.
Folder	The folder in which the tool saves exported data. If the folder doesn't exist, the tool creates the folder. By default, `C:/GoTools/DataExport`.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

This tool provides no outputs other than saving exported data in `C:\GoTools\DataExport`.

Array Tools

Array tools let you create and work with arrays, or extract individual elements from an array.

> Arrays can contain any type of data supported in GoPxL (such as Surface or Profile data, or measurements). For more information on this and on arrays in general, see *Arrays, Batching, and Aggregation* on page 242.

The following Array tools are available:

- Array Decision: Takes an *array* of *measurements* and lets you apply different minimum/maximum thresholds to each measurement in the array.

  The tool is typically used in electronics pin inspection applications.

  For more information on this tool, see *Array Decision* on page 741.

- Array Index: Takes an *array* of data as input and extracts one or more pieces of data at the user-provided index. You can then perform measurements on the individual outputs, or perform measurements using more than one piece of data, such as measuring the thickness represented by two profiles or surfaces.

  Usage examples include separating the multiple layers produced by a confocal sensor into individual profiles or surfaces.

  For more information on this tool, see *Array Index* on page 744.

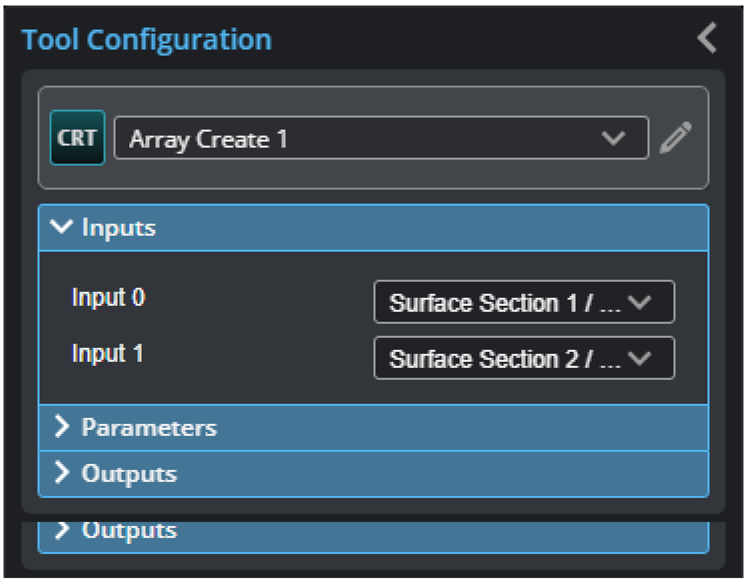
- Array Create: Takes *individual* inputs and creates an array containing those pieces of data as elements. For more information on this tool, see *Array Create* on the next page.

Array Create

The Array Create tool combines individual pieces of data (Profile, Surface, measurements, and so on) into an array. For example, a number of Surface scans could be combined into an array made up of those pieces of Surface data. Compatible measurement and processing tools can then take the array as input and perform batch operations on the elements in the array, such as measurements on each element's data or processing each element's data. For more information, see *Arrays, Batching, and Aggregation* on page 242.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

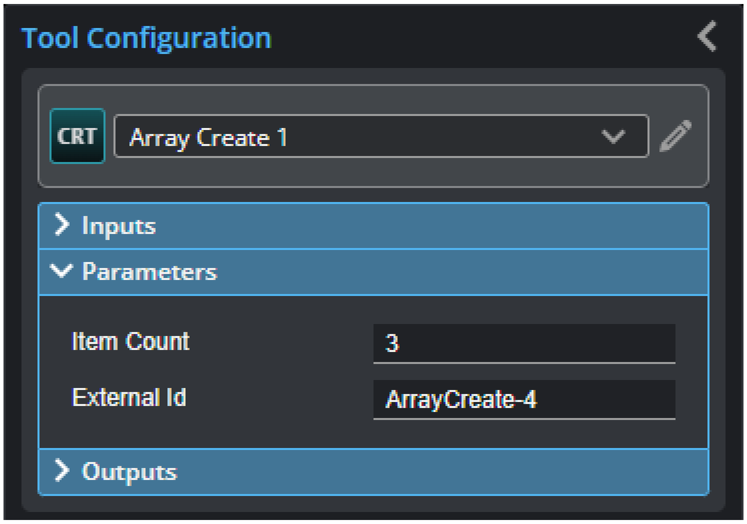


Inputs

Name	Description
Input {n}	An input that the tool will add to the output array. The number of inputs depends on the **Item Count** parameter. All inputs must be of the same type.

Parameters

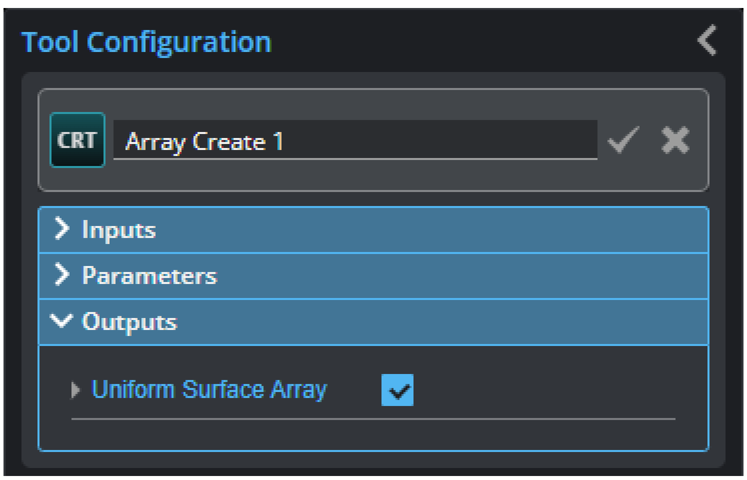
The following parameters are in the expandable **Parameters** section in the tool's configuration.



Parameters

Parameter	Description
Item Count	The number of items expected in the array.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



The type of the output depends on the type of the inputs.

Array Decision

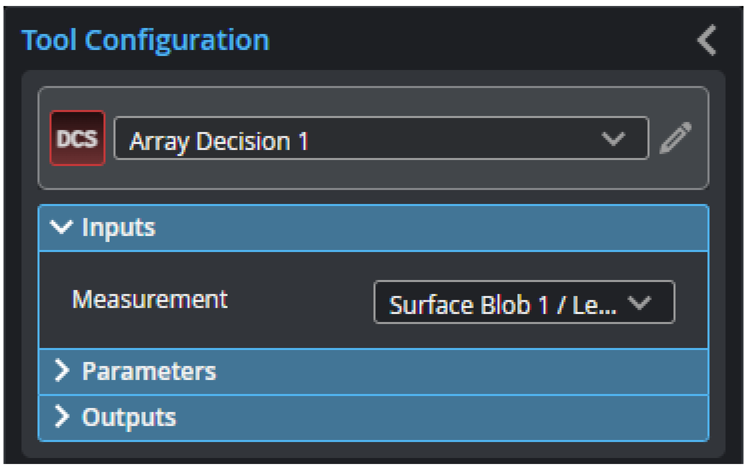
The Array Decision tool takes an array of measurements as input and applies individual minimum / maximum thresholds to one or more of the array elements. The tool returns results related to fails (number of fails, index and value of max/min fail, as well as arrays of failed measurements and their indices).

A common use case of Array Decision is to make pass/fail decisions for electronics pin inspection applications.

For general information on the Array tools, see *Array Tools* on page 738.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

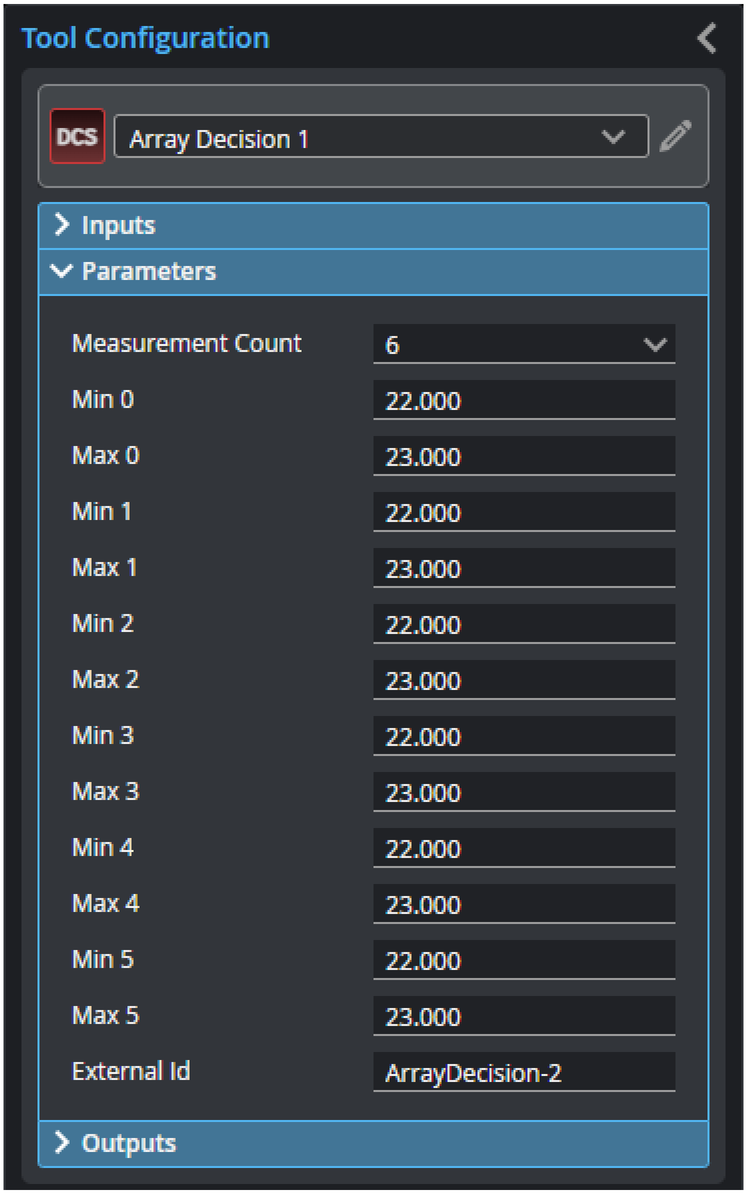


Inputs

Name	Description
Measurement	The array of measurements the tool takes as input.

Parameters

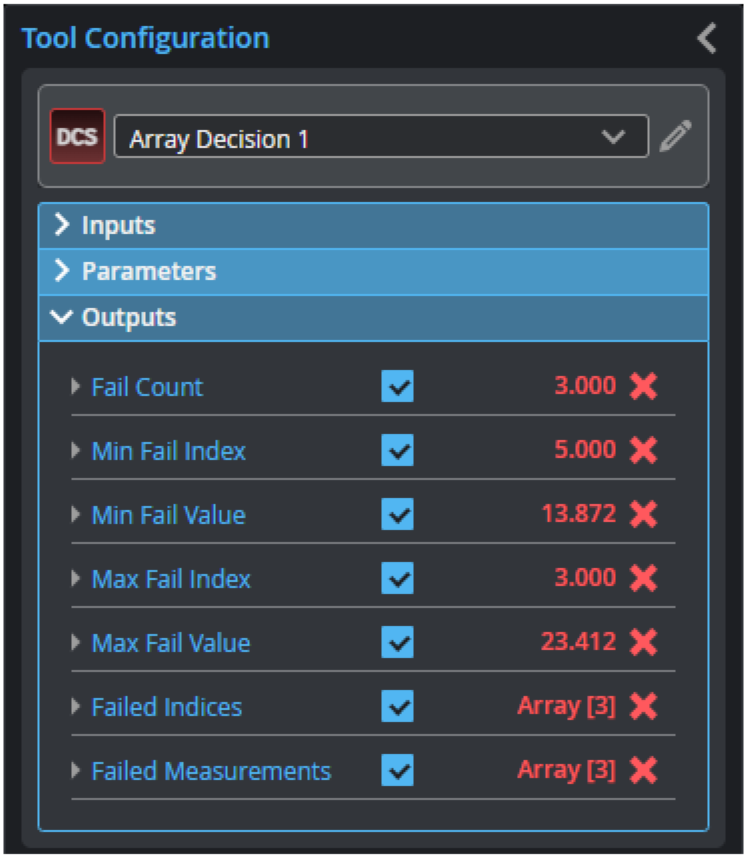
The following parameters are in the expandable **Parameters** section in the tool's configuration.



Parameters

Parameter	Description
Measurement Count	The maximum number of array elements expected in the input array. The tool displays Min{n} and Max{n} fields for each element.
Min {n} Max {n}	The minimum and maximum threshold for measurement element {n} in the input array of measurements.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

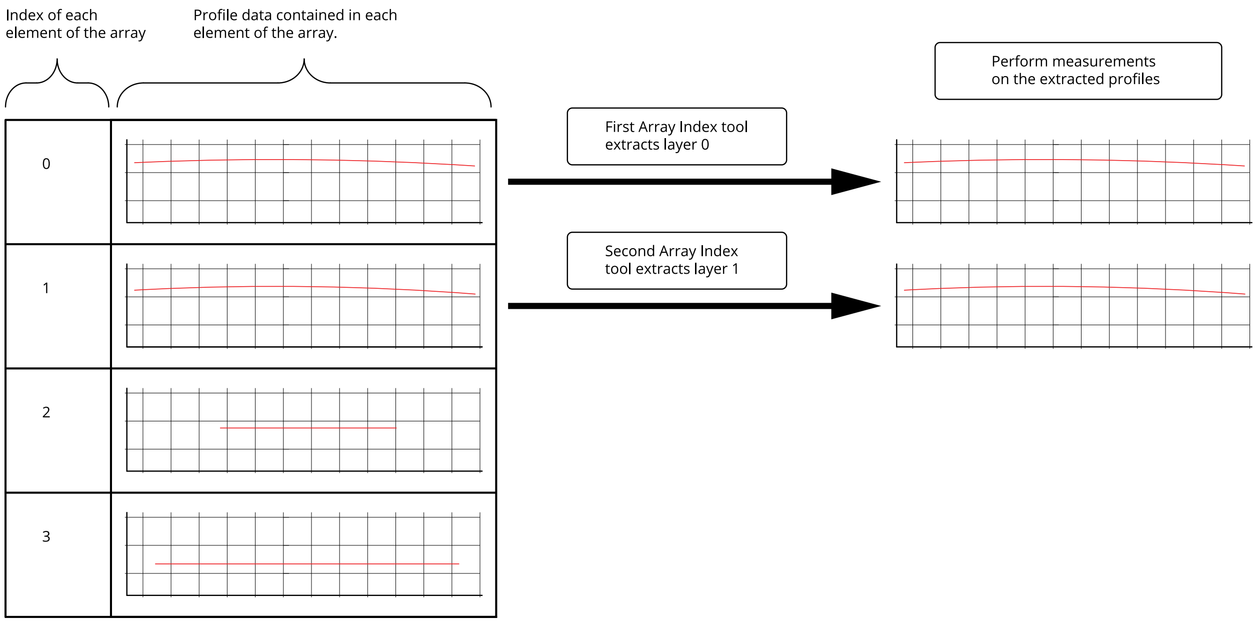


Measurements

Measurement
Fail Count The number of array values that are outside their Min/Max thresholds.
Min Fail Index The index of the array element with the greatest difference with respect to its Min threshold.
Min Fail Value The value of the array element at Min Fail Index.
Max Fail Index The index of the array element with the greatest difference with respect to its Max threshold.
Max Fail Value The value of the array element at Max Fail Index.
Failed Indices An array of the indices of the failed measurements.
Failed Measurements An array of the failed measurement values.

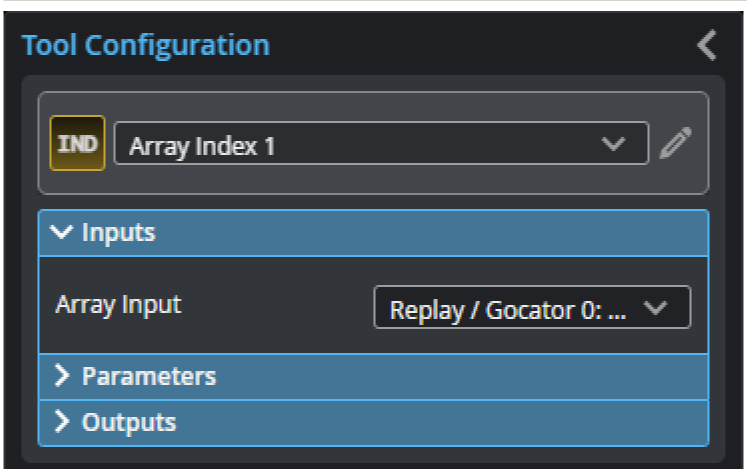
Array Index

The Array Index tool extracts a *single* piece of data (such as a profile or a surface) from the array it takes as input at the position (the index) you specify.



For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

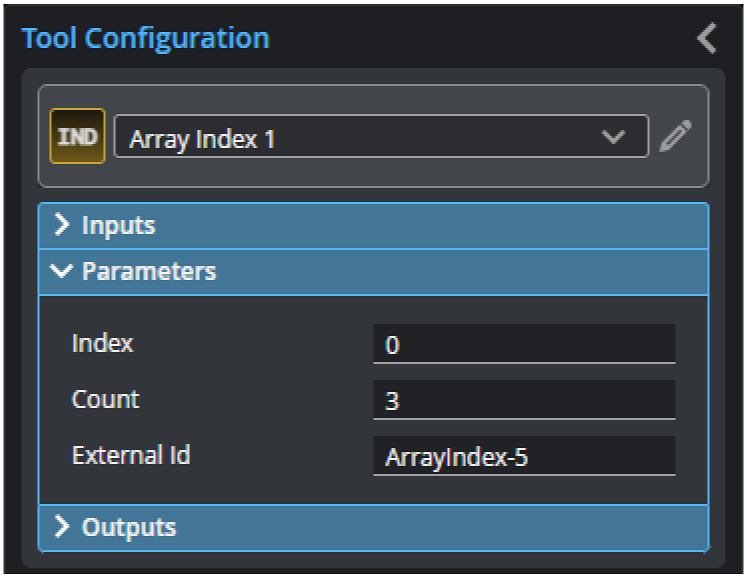


Inputs

Name	Description
Array Input	The data the tool applies measurements to or processes.

Parameters

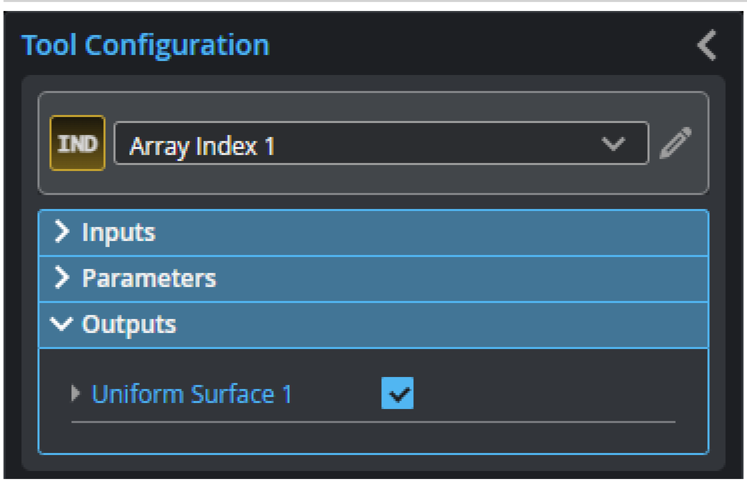
The following parameters are in the expandable **Parameters** section in the tool's configuration.



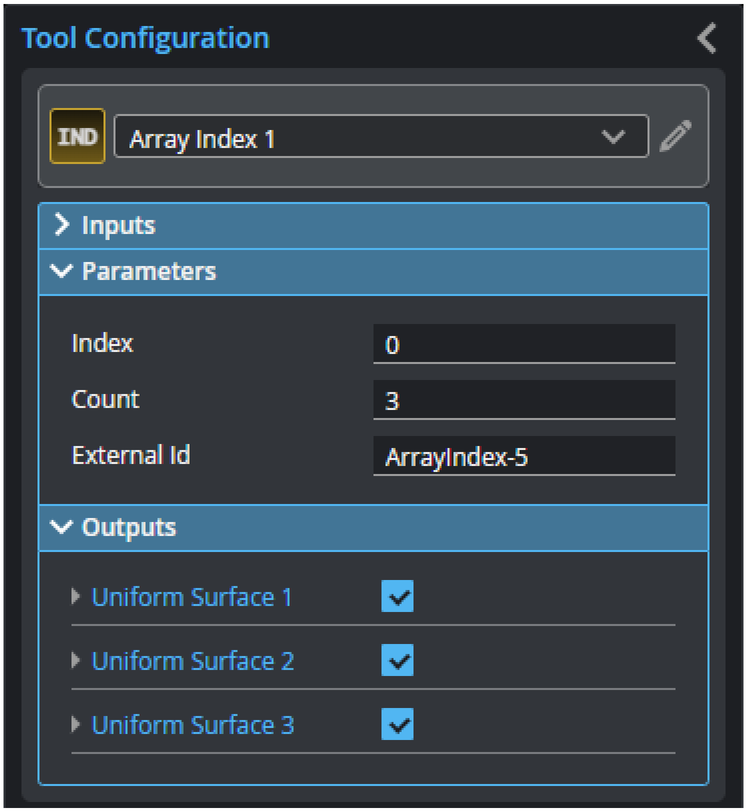
Parameters

Parameter	Description
Index	The position (the index) in the array at which the tool extracts and outputs. If **Count** is set to a value greater than 1, the index is the starting point for array element outputs.
Count	The number of individual outputs the tool produces, starting at the provided index (see above). This setting makes sending multiple outputs over protocols easier.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



If **Count** is set to a value greater than 1, the tool produces multiple outputs.



The type of the output depends on the type of the input array.

Mesh Measurement

The following sections describe Gocator's Mesh tools.

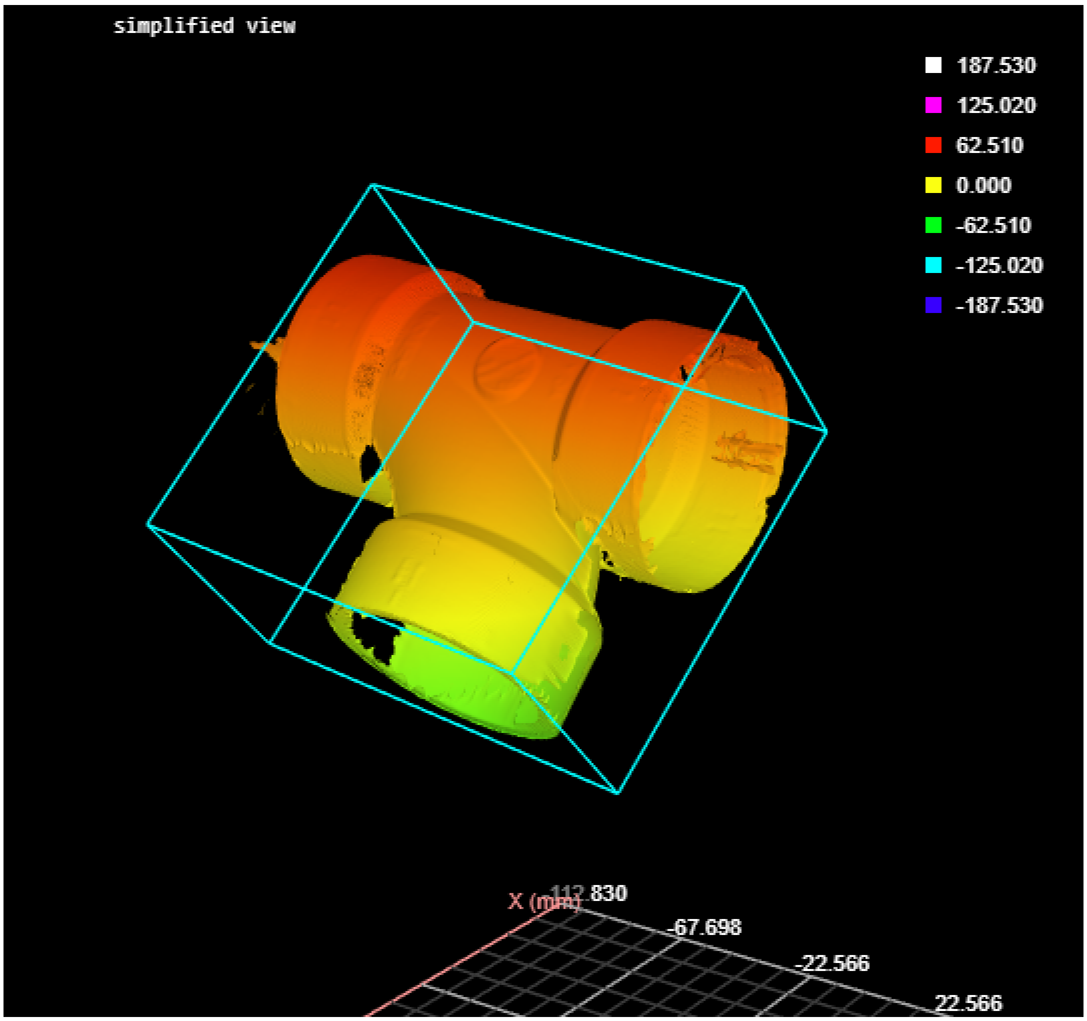
	The tools described in the following sections are *only* intended to be used with G2 sensors.

Mesh tools produce measurements on Mesh data output by the Surface Mesh tool, which stitches Surface data from multiple G2 sensors into a Mesh (for more information, see *Surface Mesh* on page 582). The Mesh Projection tool lets the sensor extract a surface from any angle of the Mesh data (using a plane returned by the Mesh Plane tool), after which it can apply any of the built-in or custom GDK-based Surface measurement tools to the extracted surface.

Mesh Bounding Box

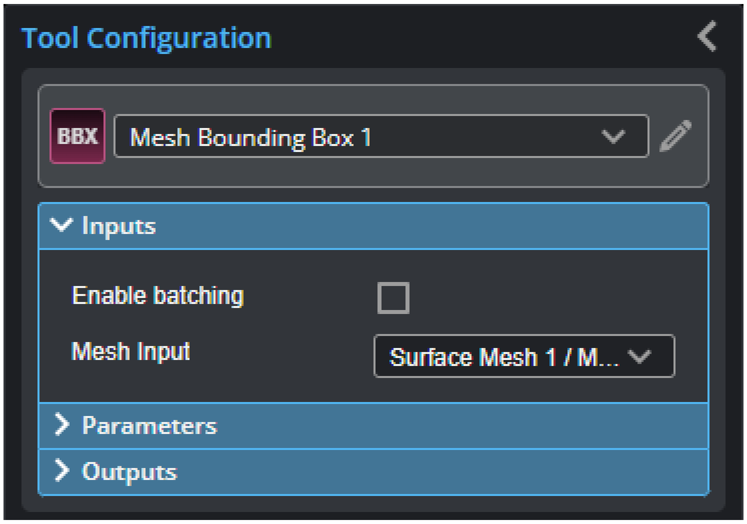
	This tool is *only* intended for use with G2 sensors.

The Mesh Bounding Box tool takes in Mesh scan data (produced by the Surface Mesh tool and some other Mesh tools) and returns measurements related to the bounding box encapsulating the scan data in the region of interest, such as the rotation of the bounding box, the dimensions of the bounding box, and its location. In addition to a Point geometric feature, the tool returns the Mesh data in the bounding box. You can apply one of the other Mesh tools to this data, or after extracting Surface data using Mesh Projection or Mesh Plane, you can apply any built-in or custom GDK-based tool to the extracted surface data.



Inputs

You configure the tool's inputs in the expandable **Inputs** section.

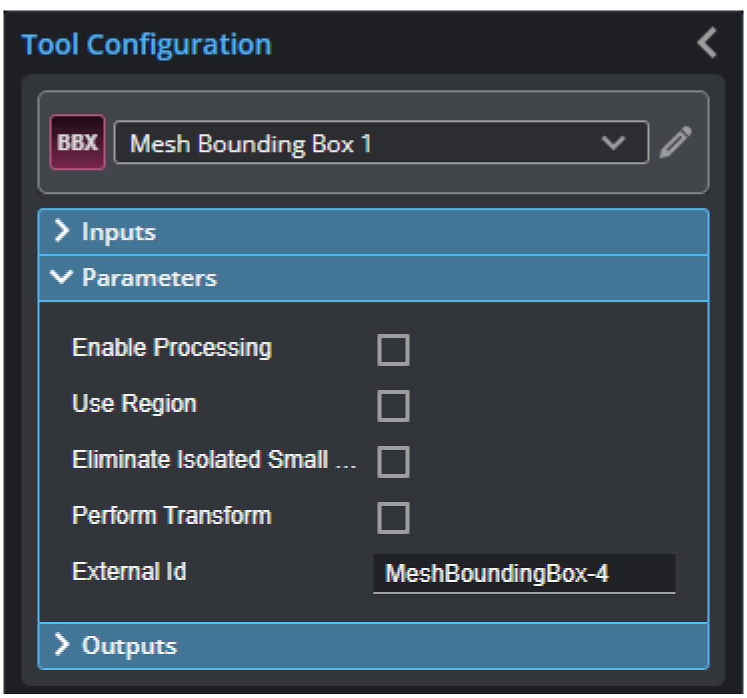


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each mesh in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Mesh Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

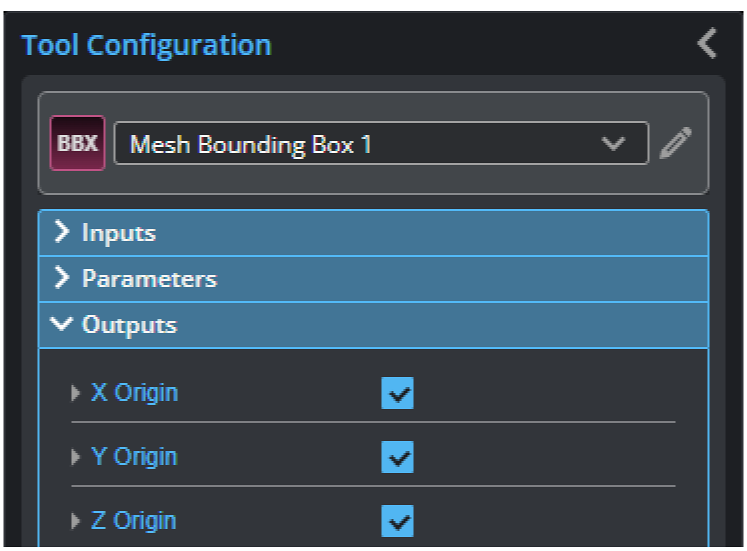


Parameters

Parameter	Description
Enable Processing	Causes the tool to start processing scan data from individual sensors. Make sure to properly configure the tool *before* enabling this option.
Use Region	Determines whether the tool limits the bounding box fit to Mesh data in a user-defined region to fit a bounding box. Enabling this option displays parameters you use to define the size and position of the region.
Eliminate Isolated Small Surface	Excludes small, unconnected data from the Mesh output.
Perform Transform Transform Mode	When **Perform Transform** is enabled, you can choose the which axes are the major, minor, and tertiary axes. The tool also centers the Mesh data at origin 0. This lets you align the part data however you want. **Transform Mode** is one of the following: • Minimal Alignment: The closest coordinate axes are arranged for alignment. • X > Y > Z Order • X > Z > Y Order • Y > X > Z Order • Y > Z > X Order • Z > X > Y Order • Z > Y > X Order
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

All outputs provide an external ID (available by expanding the output in the Outputs panel) for optional use in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Measurements

Measurement
X Origin **Y Origin** **Z Origin** These measurements return the X, Y, and Z position of the center of the fitted bounding box, respectively.
X Angle **Y Angle** **Z Angle** The angle of the fitted bounding box around the X, Y, and Z axis, respectively.
Width **Length** **Height** The width, length, and height of the fitted bounding box.

Features

Type	Description
Point	A point representing the center of the fitted bounding box.

For more information on geometric features, see *Geometric Features* on page 262.

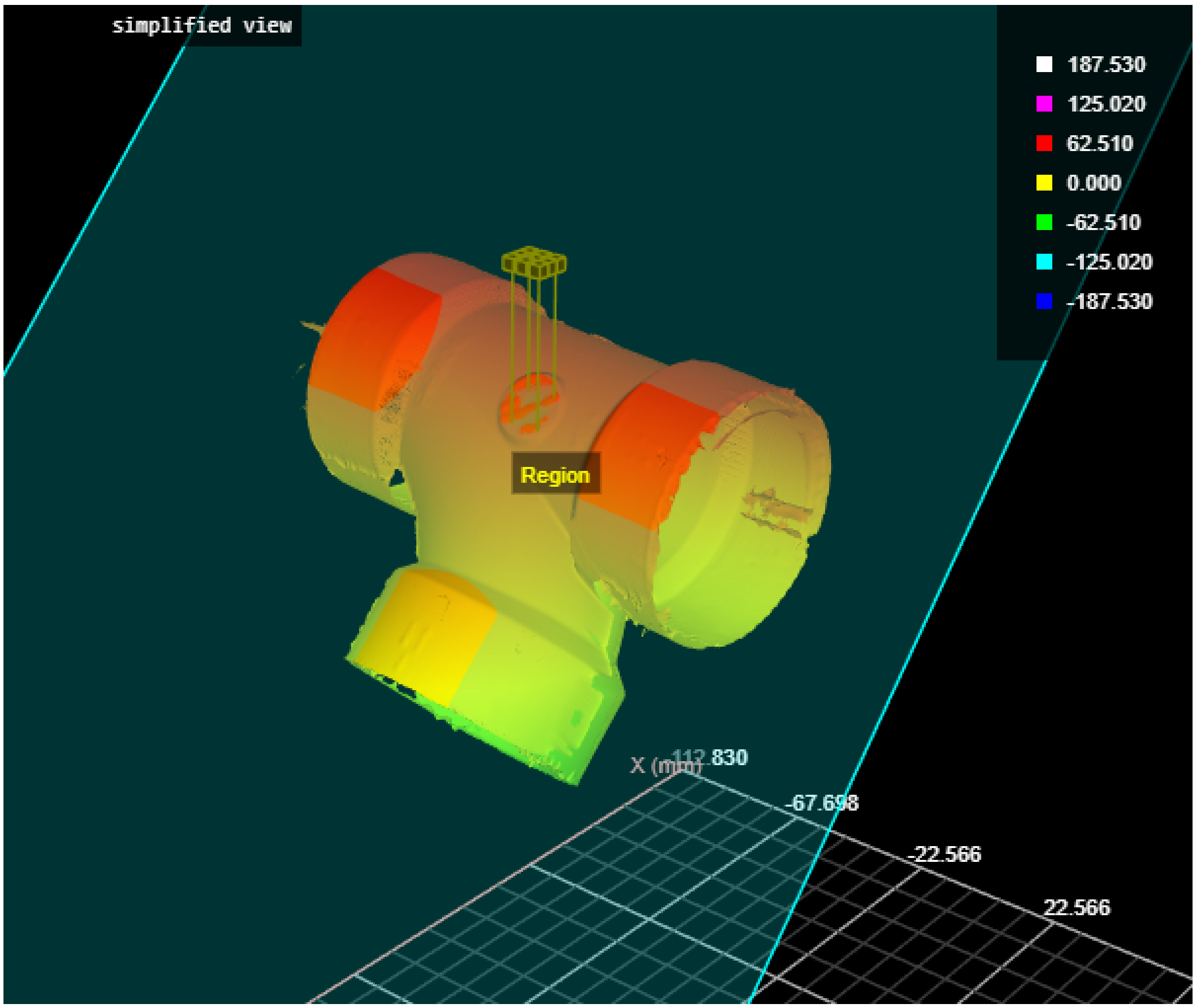
Data

Type	Description
Mesh	The Mesh data contained in the bounding box.

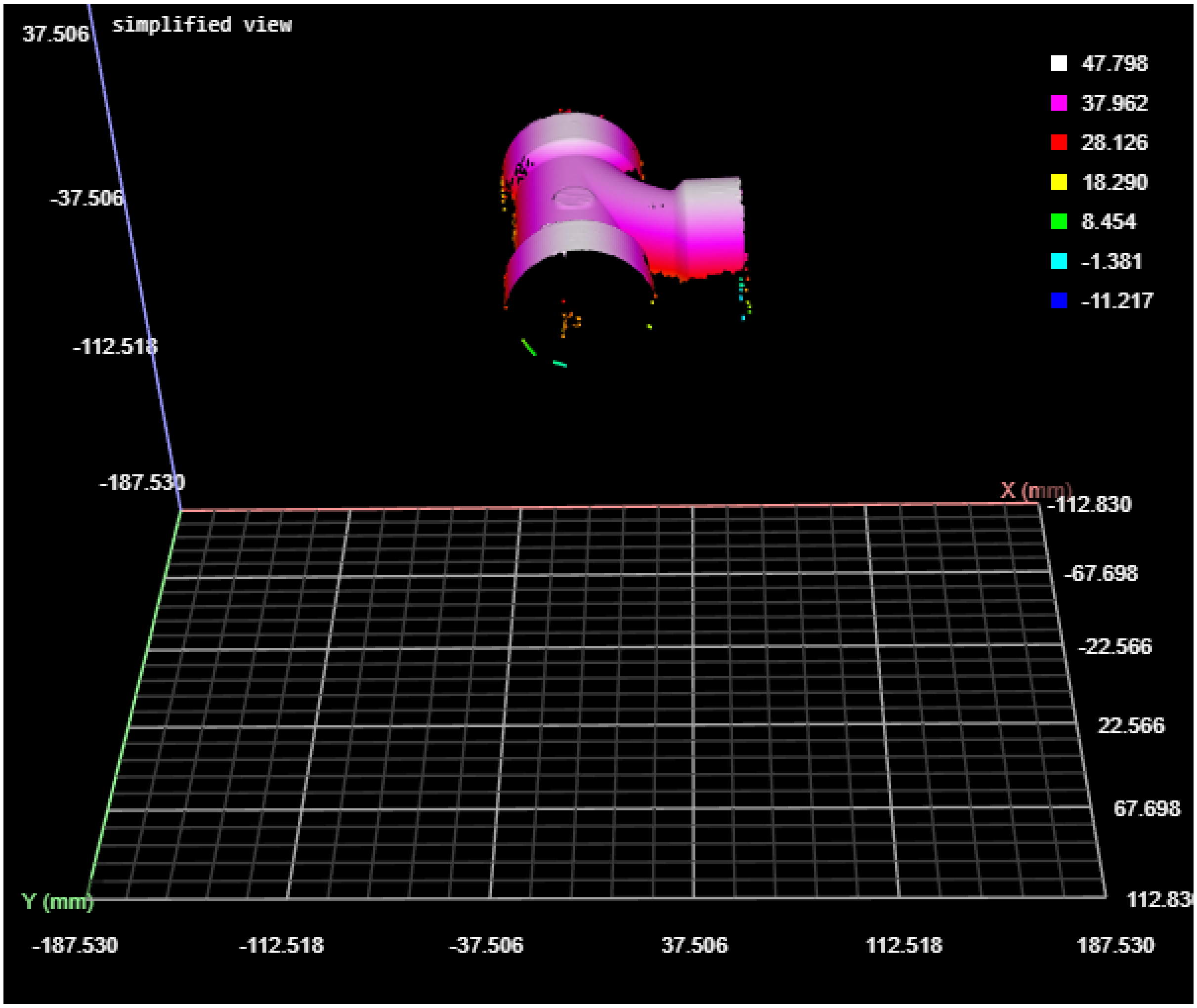
Mesh Plane

	This tool is *only* intended for use with G2 sensors.

The Mesh Plane tool takes in Mesh scan data (produced by the Surface Mesh tool and some other Mesh tools) and returns measurements on the plane fitted within the region of interest, such as deviations of the data points relative to the plane. The tool also returns a Plane geometric feature that can be used as input by the Mesh Projection tool (see *Mesh Projection* on page 757). Finally, the tool returns front and back Surface data extracted from the plane: you can apply any built-in or custom GDK-based tools to the resulting data. This means that with 360-degree scan data, you can, for example, apply measurements to the sides or bottoms of your target, rather than just the top.



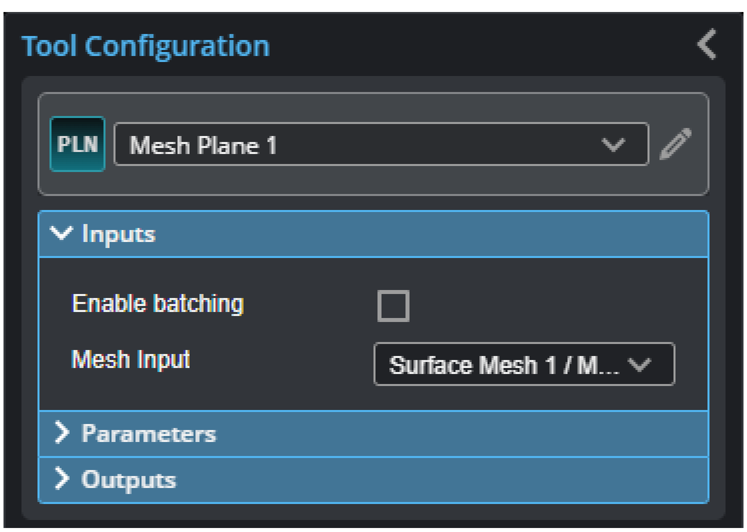
Mesh data with a region placed on a circular flat area. The plane fitted to the data in this region is shown in cyan.



The Front Surface data output is rotated by the plane's X, Y, and Z rotation.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

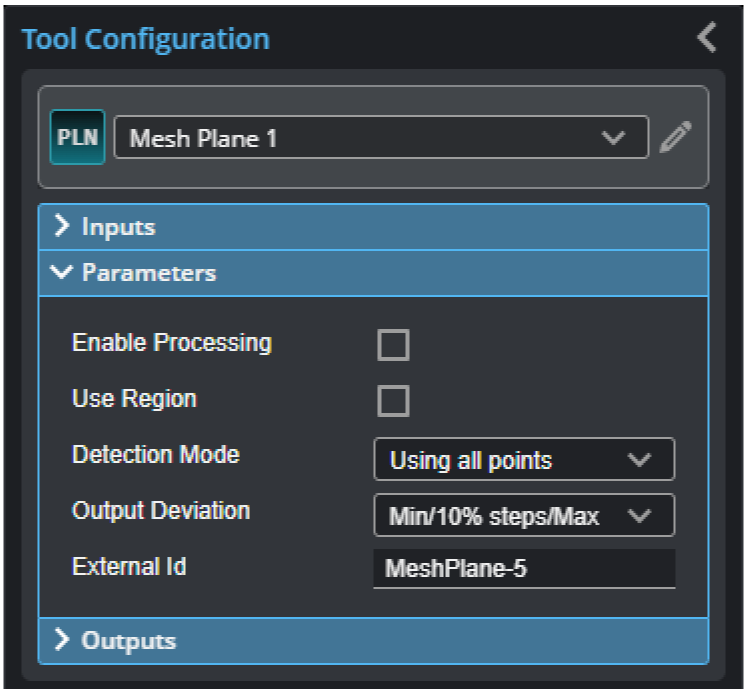


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each mesh in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Mesh Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



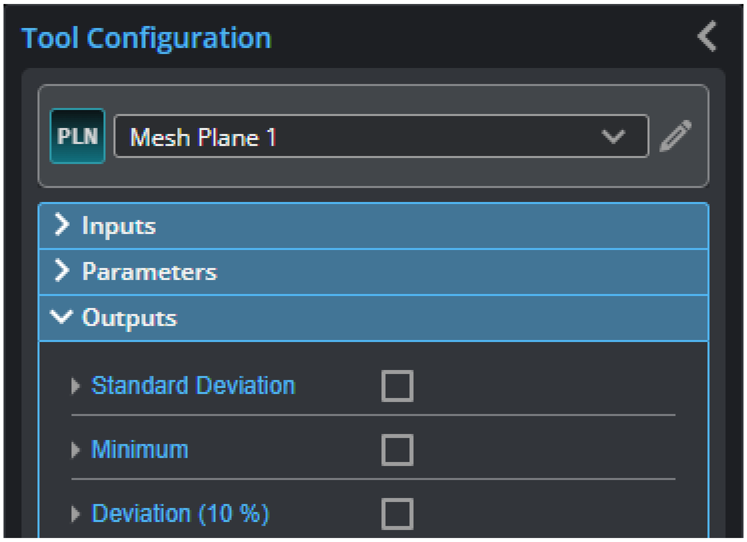
Parameters

Parameter	Description
Enable Processing	Enables processing.
Use Region	When enabled, displays region settings.
Region	Region settings.
Detection Mode	The plane detection mode. One of the following: **With Largest Area** **With Maximum Distance** **With Minimum Distance** Chooses the plane at the maximum or minimum distance in the region, respectively, from the 0 origin. Use these options when more than one plane fit is possible in the region. Works in conjunction with **Search Direction**.

Parameter	Description
	Using all points Uses all data points of the scan data in the region.
Search Direction	The search direction the tool will use to fit a plane. For example, when **Search Direction** is set to +Z, the tool starts searching from origin Z = 0 and moves along the positive Z axis. This parameter is only useful when **Plane Detection Mode** is set to one of the following: • With Largest Area • With Maximum Distance • With Minimum Distance The corresponding surface normals are taken into account in the processing so that the uninvolved points can be sorted out relatively quickly and safely. The fixed search angle is 45 degrees around the set direction. When **Search Direction** is set to **Input Direction**, the tool displays additional parameters: **Tilt Angle** and **Direction Angle**. **Tilt Angle** - The angle between the Z axis and the vector. **Direction Angle** - The vector is projected onto the XY plane and then rotated around the X axis. Specifically: X = sin(TiltAngle) * cos(DirectionAngle) Y = sin(TiltAngle) * sin(DirectionAngle) Z = cos(TiltAngle)
Output Deviation	Determines which deviations are output as measurements, which can be a combination of minimum and maximum, and a set of Deviation (x %) measurements (with the specified step between them). Can also be set so that no deviations are output. Use this to get a rough idea of the distribution of the deviation values (or a histogram of the deviations).
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Measurements

Measurement
Standard Deviation The standard deviation of the data points from the fitted plane.
Minimum **Maximum** The minimum and maximum error of the data points from the fitted plane, respectively.
Deviation (x%) Deviations of the data points from the fitted plane, sorted into stepped percentiles. You set number of steps using the **Output Deviation** parameter.

Features

Type	Description
Plane	A plane geometric feature.

For more information on geometric features, see *Geometric Features* on page 262.

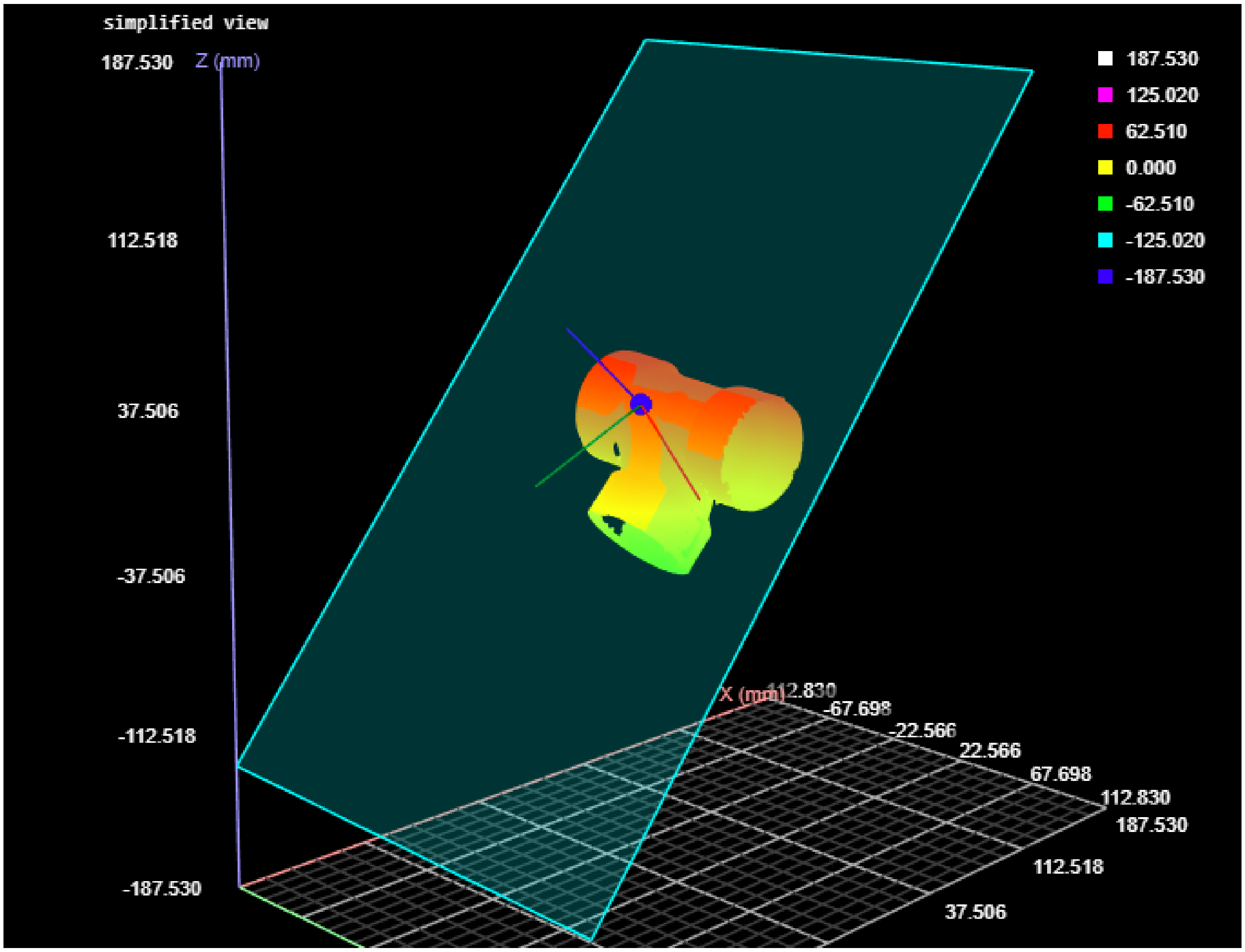
Data

Type	Description
Front Surface	Surface data representing the front of the meshed target.
Back Surface	Surface data representing the back of the meshed target.
Difference Surface	A Surface output that shows the fit error at each point in the height map.

Mesh Projection

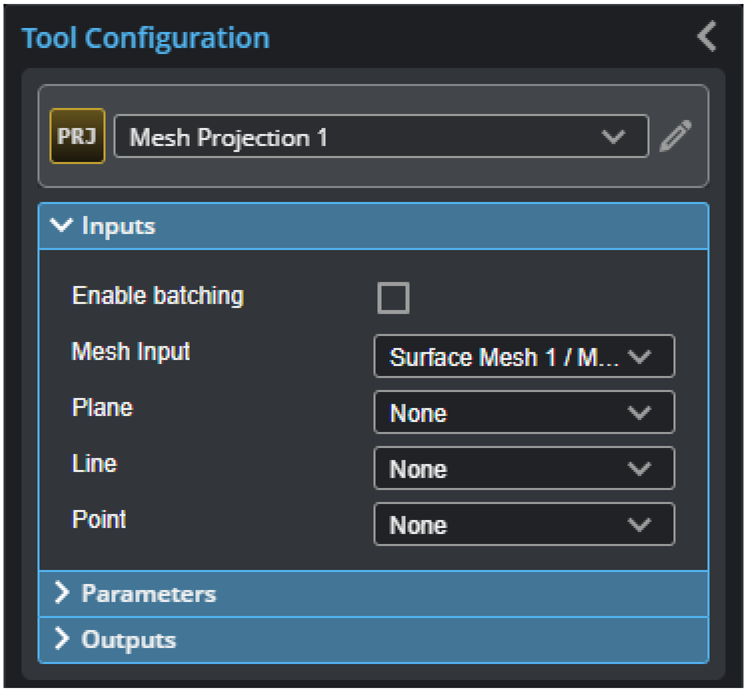
	This tool is *only* intended for use with G2 sensors.

The Mesh Projection tool takes in Mesh scan data (produced by the Surface Mesh tool) and extracts Surface data. The tool can optionally take plane, line, or point geometric features to perform transformations on the output surface data (if no geometric features are used as inputs, the surface parallel to the XY plane is output), or you can manually apply fixed transformation. You can then apply any built-in or custom GDK-based Surface tool to the resulting Surface data. This means that with 360-degree scan data, you can, for example, apply measurements to the sides or bottoms of your target, rather than just the top.



Inputs

You configure the tool's inputs in the expandable **Inputs** section.

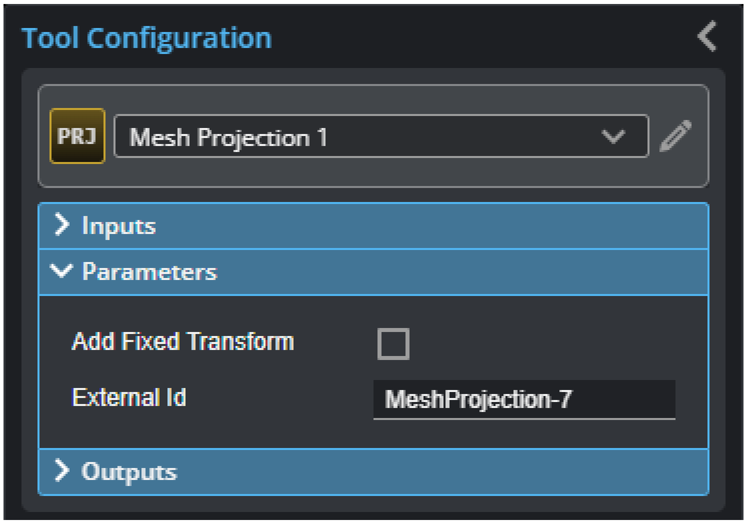


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each mesh in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with a maximum of two elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with a maximum of two elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. When **Enable Batching** is unchecked, the tool takes either a single surface, or an array of surfaces with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Mesh Input	The data the tool applies measurements to or processes.
Plane	The tool aligns the XY plane to the selected plane geometric feature in the output Surface data.
Line	The tool aligns the X axis to the selected line geometric feature in the output Surface data.
Point	The tool uses the selected point geometric feature the origin in the output Surface data.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.

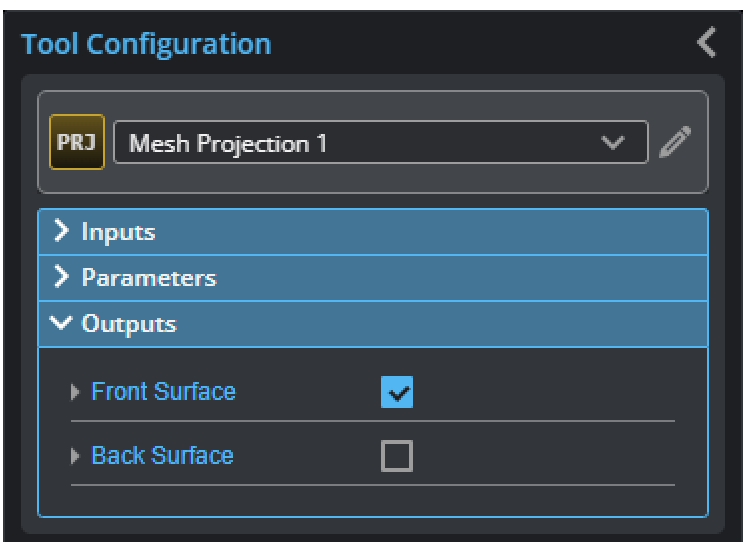


Parameters

Parameter	Description
Add Fixed Transform	When this parameter is enabled, you can provided fixed X, Y, and Z offsets, as well as X, Y, and Z angles, which the tool uses in the output Surface data.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Data

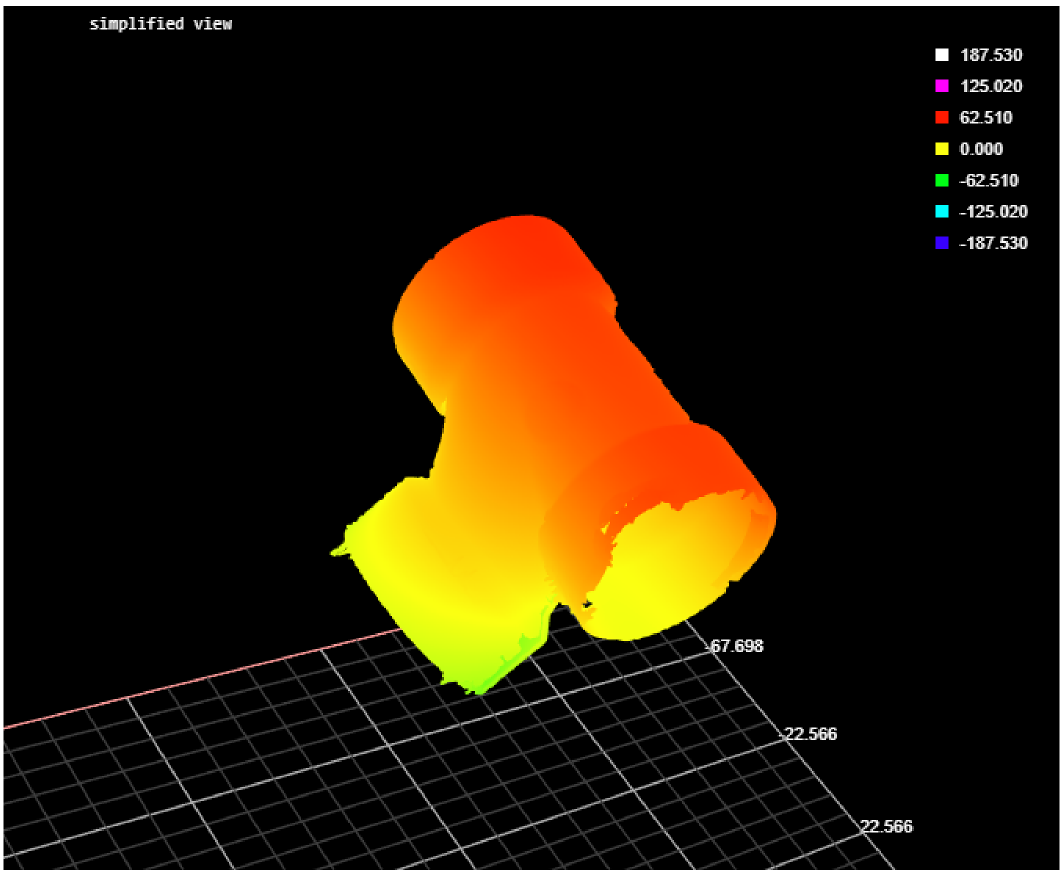
Type	Description
Front Surface	Surface data representing the front of the meshed target.
Back Surface	Surface data representing the back of the meshed target.

Mesh Template Matching

	This tool is *only* intended for use with G2 sensors.

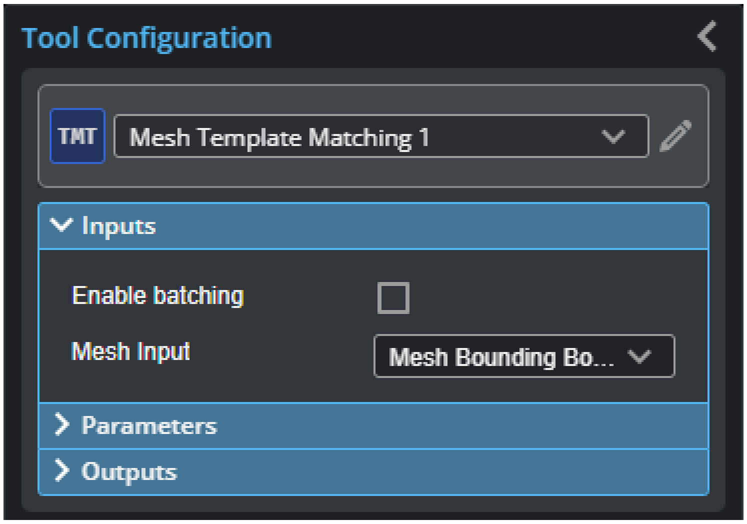
	This tool is not supported on A and B revision Gocator 2100 and 2300 sensors that are not accelerated (either by a PC-based application or by GoMax). The tool is supported in emulator scenarios.

The Mesh Template Matching tool takes in Mesh scan data (produced by the Surface Mesh tool) and a template you previously defined based on a "golden part" (itself created using the Mesh Template Matching tool). The tool returns measurements related to the position and orientation of the scan data relative to the template, such as offsets and rotations, as well as standard deviations between the scan data and the template. The tool can also output Mesh scan data.



Inputs

You configure the tool's inputs in the expandable **Inputs** section.

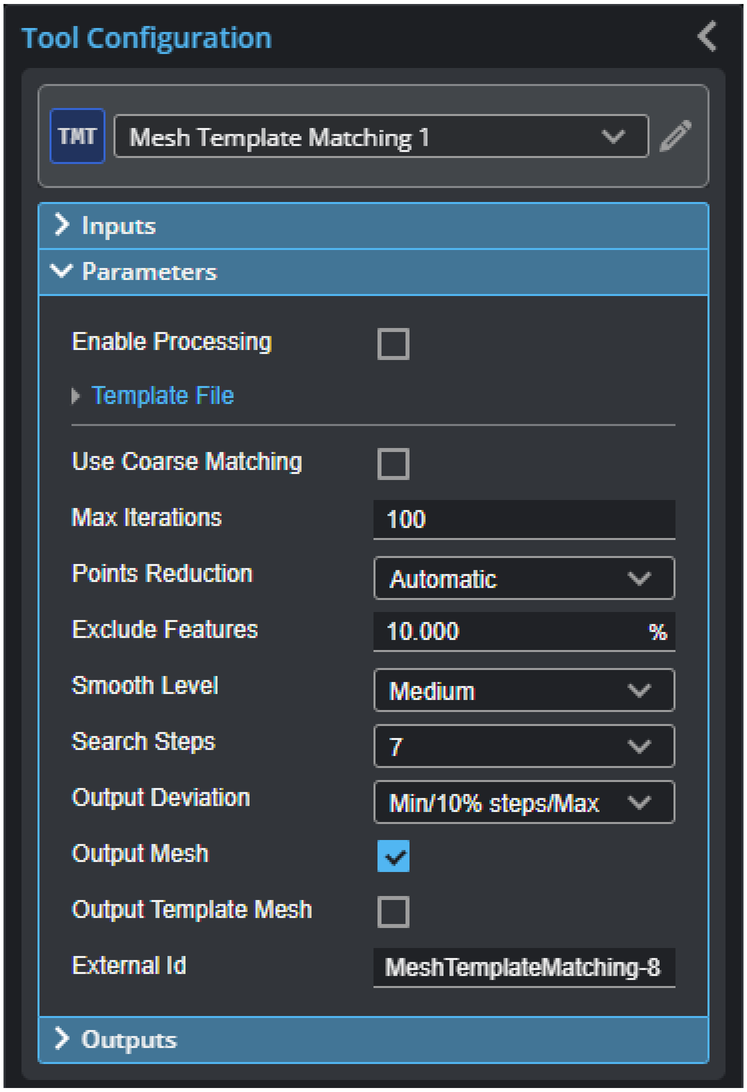


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each mesh in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Mesh Input	The data the tool applies measurements to or processes.

Parameters

You configure the tool's parameters in the expandable **Parameters** section.



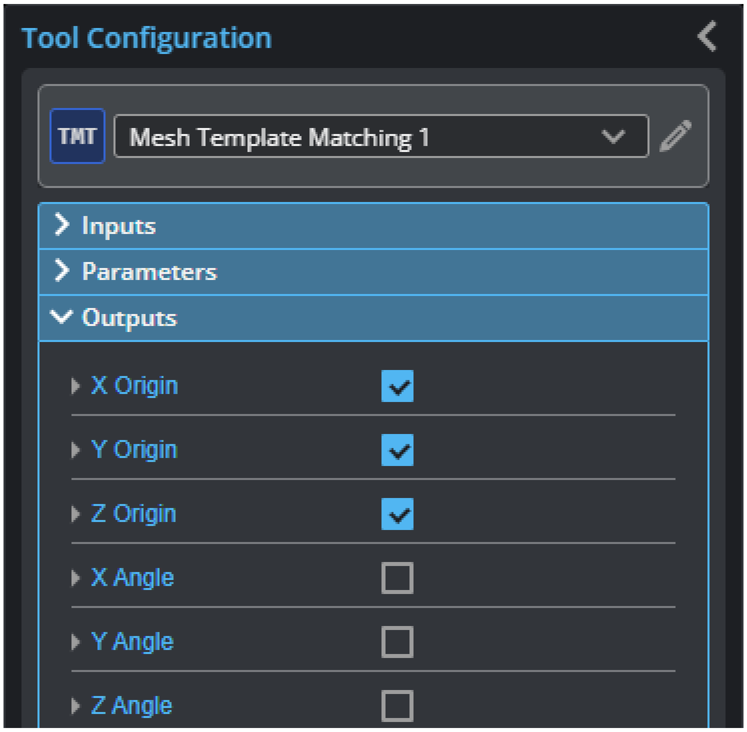
Parameters

Parameter	Description
Enable Processing	When this option is enabled, the tool compares the Mesh data to the loaded template.
Template File	Expanding section containing template-related parameters.
Operation Template Name Template File	The tool's operation mode. One of the following: **Normal**: When Enable Processing is enabled, the tool compares the Mesh scan data and the loaded template. **Load**: Displays a list of Mesh template files you can load (in the **Template File** drop-down). Supports the following three formats: • BCD: Internally used by the tool. To reduce the conversion processing time, a BCD file with the same name is automatically generated if the data is imported with a different format.

Parameter	Description
	• OBJ: A format for defining the 3D geometry for the surface of one or more objects. • STL: A file format commonly used for 3D printing and computer-aided design (CAD). **Save**: Saves the current frame of Mesh scan data as a template (in C:\GoTools\Mesh Template Matching\) in BCD format. Type the name of the template in the **File Name** field, and then check **Enable Processing** to save the file. A user message is displayed if the template is saved successfully. Note this option is only available if there is no template currently loaded in this tool. **Delete**: Deletes the template file you select in the **Template File** field. **Refresh**: Refreshes the template file list. **Remove**: Unload an existing template to make the Save option visible. So that user is aware that just the current input mesh data will be declared and saved as a template, when selecting the Save option.
Use Course Matching	Enable this parameter when the body surfaces to be matched are not symmetrical, and the original orientation and position deviate significantly from the measured data set.
Max Iterations	The maximum number of iterations the tool uses to match the Mesh scan data with the template. This parameter can be used to speed up the optimization process to some degree. Typically, leave this at the default value.
Points Reduction	Controls the number of points used in the matching process, which can improve processing time.
Exclude Features	Use this when there are high or low features on the part that should not be included in the matching. For example, at 10%, the tool excludes 10% of the points with maximum or minimum deviation from the matching process.
Smooth Level	The amount of smoothing the tool applies. LMI recommends leaving this setting at its default.
Search Steps	Determines the neighborhood level in which to search for connection point pairs.
Output Deviation	Determines which deviations are output as measurements, which can be a combination of minimum and maximum, and a set of Deviation (x %) measurements (with the specified step between them). Can also be set so that no deviations are output. Use this to get a rough idea of the distribution of the deviation values (or a histogram of the deviations).
Output Mesh	Outputs the transformed mesh.
Output Template Mesh	Outputs a mesh template.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

All outputs provide an external ID (available by expanding the output in the Outputs panel) for optional use in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Measurements

Measurement
X Origin **Y Origin** **Z Origin** **X Angle** **Y Angle** **Z Angle** These measurements represent the transform matrix that matches the input mesh against the template.
Standard Deviation The standard deviation of pairs of matching points.
Minimum **Maximum** The minimum and maximum deviation of pairs of matching points.
Deviation (x%) The deviations of pairs of matching points, sorted into stepped percentiles. You set the number of steps using the **Output Deviation** parameter.

Data

Type	Description
Mesh	The transformed Mesh. Only listed if the **Output Mesh** parameter is enabled.
Mesh Template	A template Mesh. Only listed if the **Output Template Mesh** parameter is enabled.

Measurement Formula Tool

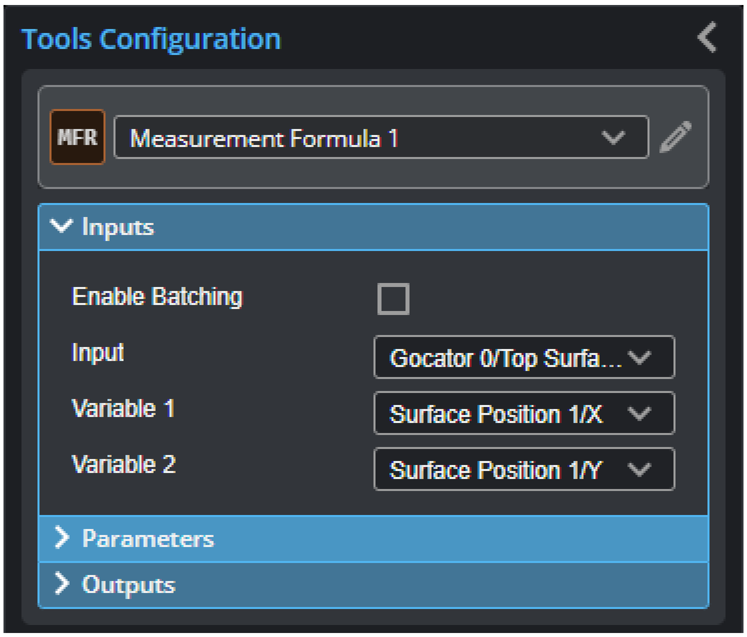
The Measurement Formula tool lets you create a mathematical expression containing aliases to one or more measurements from other measurement tools, up to a maximum of 20. The result of the calculation is returned in the tool's Result measurement.

The tool's default timeout is 5 seconds: if the tool fails to complete the calculation after 5 seconds, the result is set to Invalid.

The tool uses the C++ Mathematical Expression Toolkit Library (ExprTk). For full details on what it supports, see the toolkit's page (http://www.partow.net/programming/exprtk/).

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Inputs

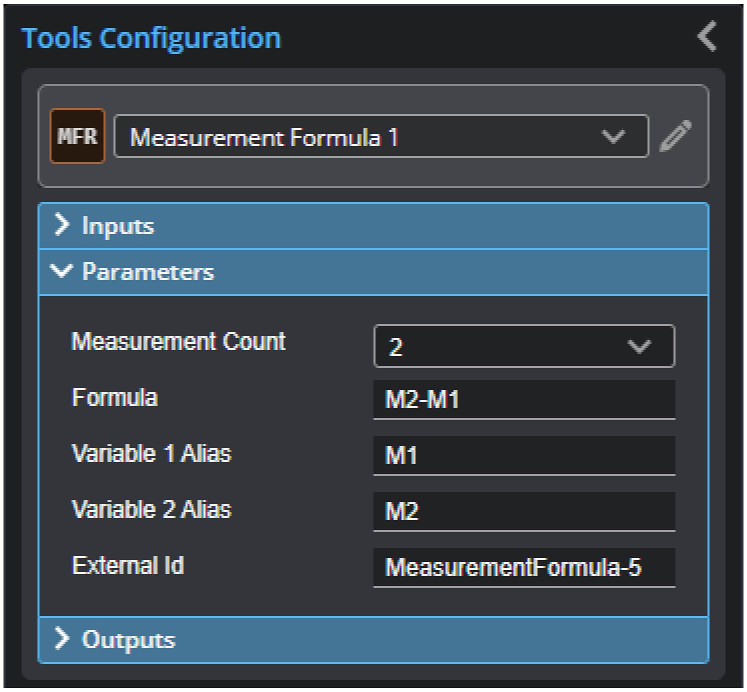


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each element in the array individually. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Input	Set this parameter to any available input. The selected input has no effect on the tool's calculations.
Variable {n}	A measurement from another tool that you use in the formula. The **Measurement Count** parameter controls the number of variable inputs.

Parameters

The following parameters are in the expandable **Parameters** section in the tool's configuration.



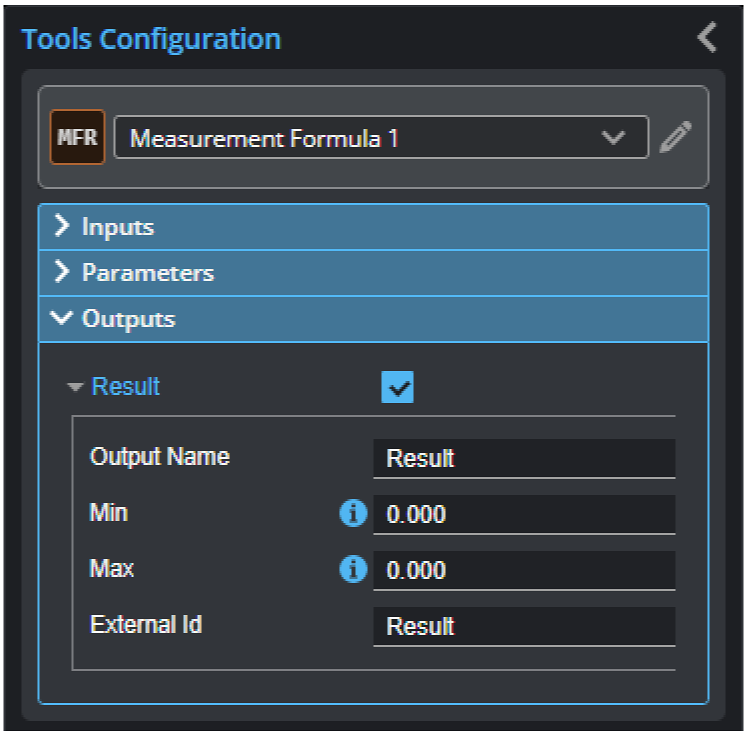
Parameters

Parameter	Description
Measurement Count	Sets the number of variable inputs in the **Inputs** section. For each, a **Variable {n}** input and a **Variable {n} Alias** parameter is added, and for use in the expression.

Formula	The mathematical expression.
	
	Use the variable aliases from the **Variable {n} Alias** parameters in the expression. For example:
	• M1*3+M2/2+4
	• tan(M1/180*pi)
	• max(M1, M2, M3)
	
	The tool supports complex mathematical formulas. For example, the following:
	$\sqrt{\frac{1-a\times e^{2t}+w^{\pi}}{sin(\frac{2x}{\pi})+cos(\frac{\pi}{y})}}$
	can be produced using the following formula:
	`sqrt((1-a*exp(2t)+w^(pi))/(sin(2x/pi)+cos(pi/y)))`
	
	The following shows how to perform a *while* loop:
	`y:=0; while(x<10){y:=y+x;x+=1;};y`

Parameter	Description
	Note that ":=" is the assignment operator, and "=" is the equality operator. The following shows how to use an if-else structure: `if(x>2){x:=x+2;}else{x:=x-2;};x`
Variable {n} Alias	The alias of Variable {n} (that is, a measurement set in the **Inputs** section). Use these aliases in **Formula**.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Result The result of the calculation in **Formula**.

Script

The Script tool executes a user-defined Python-based script. For a list of the tool's supported classes and methods, see *Functions and Classes* on page 774.

For examples, see *Examples* on page 783.

The tool supports all GoPxL data types as input and output:

- Surface and Profile data (including intensity)
- Measurements (values and decisions, and anchor sources for other tools)
- Geometric features
- Arrays of the supported types

> The Script tool does *not* support writing files when it is running on a sensor or GoMax NX device, and will display a permissions error if you attempt this. Writing to files is only supported when the tool is running on a PC instance of GoPxL. When writing to files, you should explicitly set the path to a writable location.

The tool does not support using `print()`, `sys.stdout`, or Python's `logging` module functions to send messages to GoPxL's system log. Instead, use one of the "log_" functions. For more information, see *Logging* on page 778.

Script tools are added, configured, and removed like other measurement tools.

> The Script tool uses Python version 3.6.

For information on adding, managing, and removing tools, as well as detailed descriptions of settings common to most tools, see *Tool Configuration* on page 240.

Included Python Modules

In addition to the standard Python library modules, the tool also includes the NumPy 1.19 module. NumPy is useful for operating on arrays.

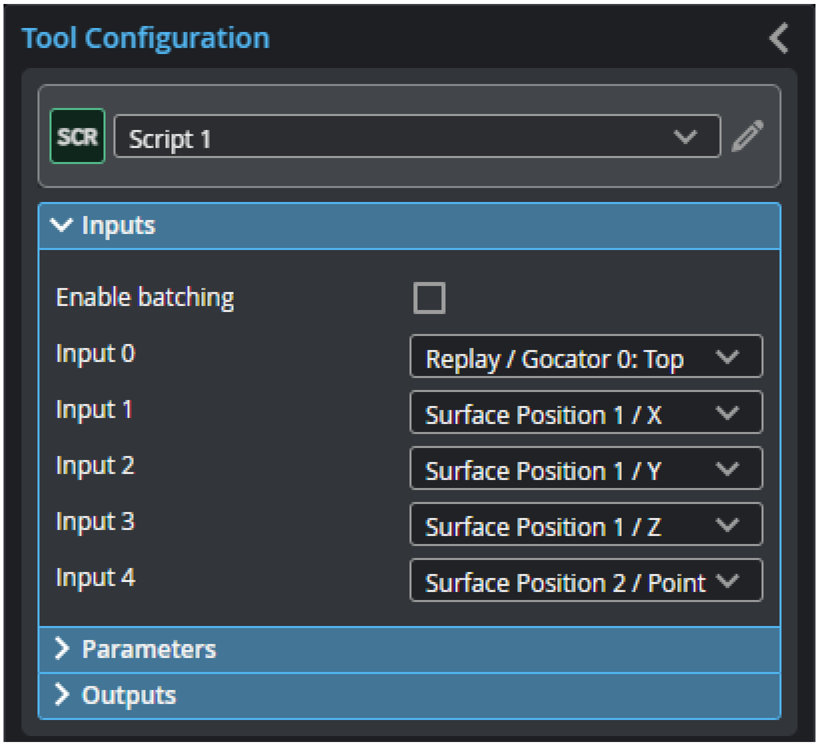
On Windows, you can install additional modules with `pip` by running the following command under the `GoPxL\res\python` folder:

```
python3.exe -m pip install <modulename>
```

Note that for some libraries you may need to specify a specific version to ensure compatibility with the version of Python included with GoPxL.

Inputs

You configure the tool's inputs in the expandable **Inputs** section.

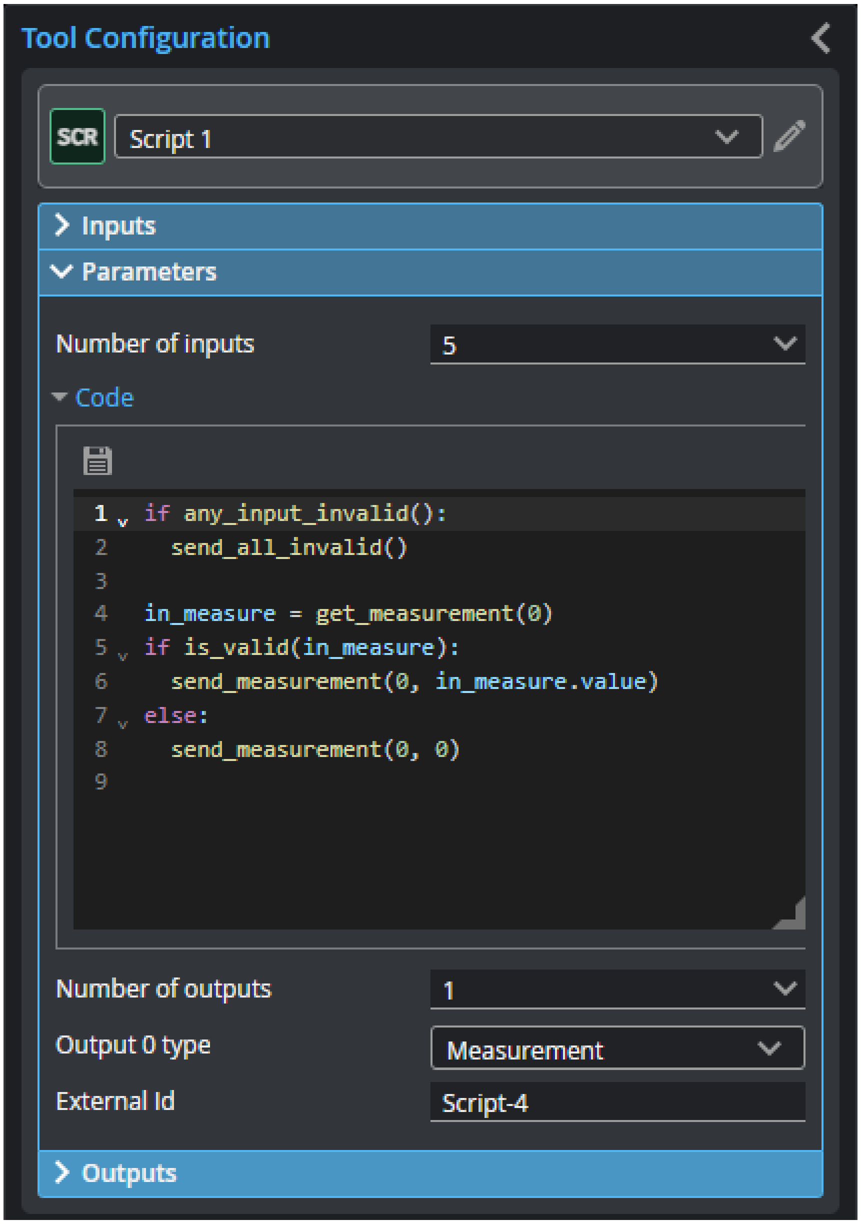


Inputs

Name	Description
Enable Batching	When **Enable Batching** is checked, the tool takes an array as input and processes each profile in the array individually. There is no limit to the size of the array, other than the processing limitations of the sensor. When **Enable Batching** is unchecked, the tool takes either a single profile, or an array of profiles with any number of elements. If the input is an array, the tool will aggregate the data in the array, and process that data as a whole. For more information on arrays, batching, and aggregating, see *Arrays, Batching, and Aggregation* on page 242.
Input {n}	Selects the input for the script where {n} is the index of the input. All input types are allowed. Input cannot be 'None'. In the `get_` functions' `index` parameters, use this index. The **Number of inputs** parameter (see below) determines the number of inputs. This tool can optionally take an array as input. For more information, see *Arrays, Batching, and Aggregation* on page 242.

Parameters

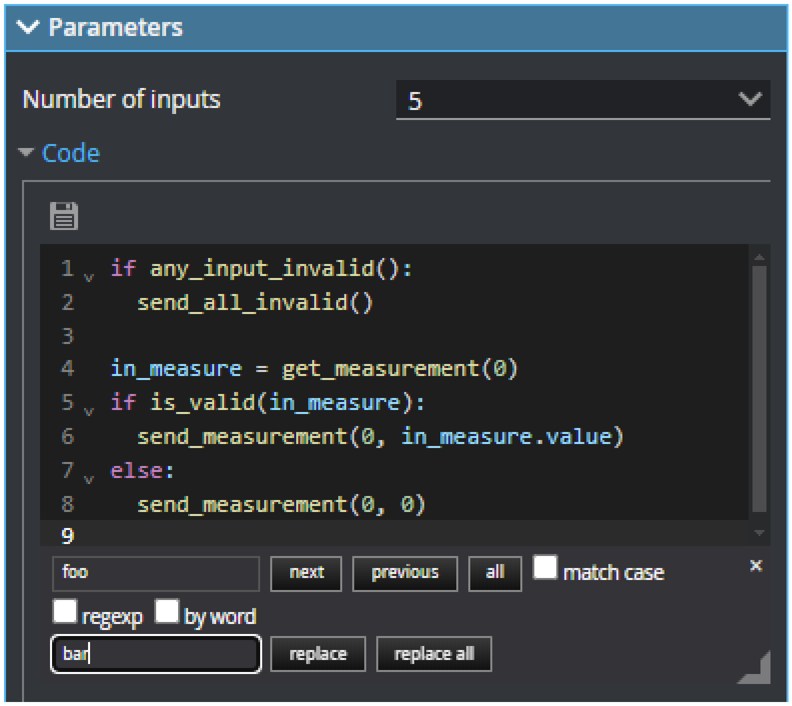
You configure the tool's parameters in the expandable **Parameters** section.



Parameters

Name	Description
Number of Inputs	Specifies the number of inputs needed in the script.
Code	The Python code the tool runs. Supports expanding and collapsing of code blocks. You can resize the editor vertically with the grabber to the lower left of the editor. You can resize the width of the editor by resizing the **Tool Configuration** panel. The following keyboard shortcuts are available in the editor. **Ctrl-S**: Saves the script. The tool checks the syntax. Use Cmd-S on macOS. **Ctrl-F**: Opens a search and replace panel. The search and replace panel supports regular expressions. Use Cmd-F on macOS. To close the panel, click the "x" to the

Name	Description
	upper right.



Ctrl-D: Highlights all occurrences of the word under the cursor. Each time you press Ctrl-D after that, the cursor moved to the next occurrence. Use Cmd-D on macOS.

Alt-ArrowLeft / Alt-ArrowRight: Move the cursor over the next element to the left or right, respectively.

Alt-ArrowUp / Alt-ArrowDown: Move the selected lines up or down one line, respectively. Combining this with Shift copies lines instead.

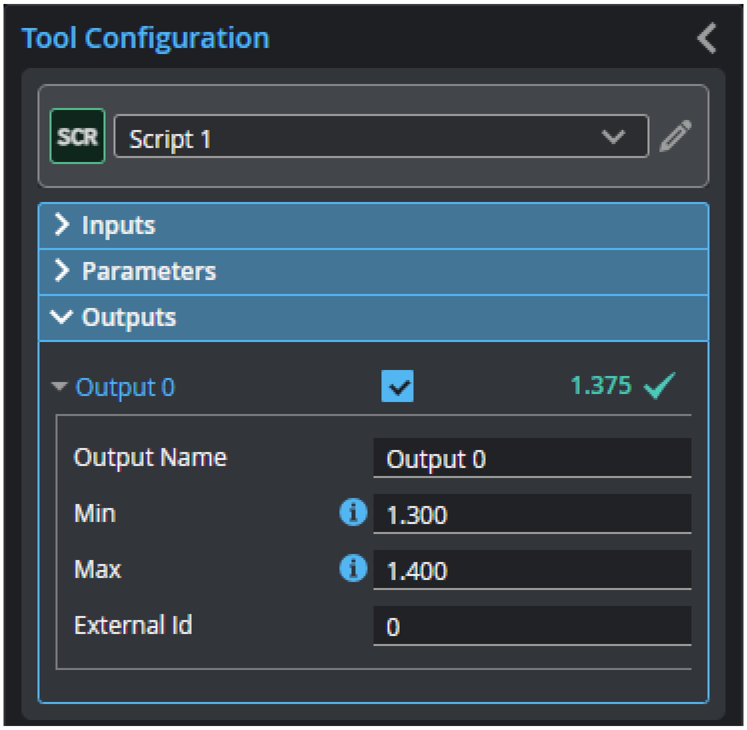
Ctrl-/: Toggles commenting. Use Cmd-/ on macOS.

Ctrl-[/ Ctrl-]: Decreases or increases the indentation of the selected line. Use Cmd-[or Cmd-] on macOS.

Name	Description
Number of outputs	Specifies the number of outputs the script produces.
Output type {n}	Selects the output type for each output, where {n} is the index of the output.
External ID	The external ID of the tool that appears in GoHMI Designer. For more information, see *GoHMI and GoHMI Designer* on page 844.

Outputs

Most tools provide measurements, geometric features, or data as outputs.



Outputs section with a measurement expanded to show user-configurable decision min/max fields and an external ID

You configure the **Min** and **Max** parameters by expanding the measurement in the **Outputs** section. In order for a measurement to return a Pass decision, the measurement must be between maximum and minimum values; the range is inclusive.

Measurements

Measurement
Output {n} Output from the Python script, where {n} is the index of the output. The number of outputs and the type of each output is determined by the **Number of outputs** and **Output type {n}** parameters, respectively.

Functions and Classes

Getting Input

The following Python functions are available for performing input operations and testing input validity.

The index of the input is given as an argument. If the input at the index is missing or it is the wrong type, None is returned. Functions will return an array if input is an array.

For information on the returned types, see *Data Structures* on page 778.

Input Functions

Function	Description
is_valid(obj)	Checks whether the object is valid or not. *Parameters*

Function	Description
	obj: Input message. *Returns* True if valid.
get_surface(index)	Gets a surface message or array of messages from the specified input. *Parameters* **index**: Input index. *Returns* SurfaceMsg, array of SurfaceMsg, or None.
get_profile(index)	Gets a profile message or array of messages from the specified input. *Parameters* **index**: Input index. *Returns* ProfileMsg, array of ProfileMsg, or None.
get_measurement(index)	Gets a measurement message or array of messages from the specified input. *Parameters* **index**: Input index. *Returns* MeasureMsg, array of MeasureMsg, or None.
get_point(index)	Gets a point message or array of messages from the specified input. *Parameters* **index**: Input index. *Returns* PointFeatureMsg, array of PointFeatureMsg, or None.
get_circle(index)	Gets a circle message or array of messages from the specified input. *Parameters* **index**: Input index. *Returns* CircleFeatureMsg, array of CircleFeatureMsg, or None.
get_plane(index)	Gets a plane message or array of messages from the specified input. *Parameters* **index**: Input index. *Returns* PlaneFeatureMsg, array of PlaneFeatureMsg, or None.
get_line(index)	Gets a line message or array of messages from the specified input. *Parameters* **index**: Input index. *Returns* LineFeatureMsg, array of LineFeatureMsg, or None.
get_image(index)	Gets an image message or array of messages from the specified input. *Parameters* **index**: Input index.

Function	Description
	Returns ImageMsg, array of ImageMsg, or None.

Sending Output

The following Python functions are available for performing output operations.

The index of the output is given as an argument.

For information on the output types, see *Data Structures* on page 778.

Output Functions

Function	Description
is_output_enabled(index)	Checks whether the output at the specified index is enabled. *Parameters* **index**: Output index. *Returns* True if the output is enabled.
send_surface(index, offset, scale, points, intensity=None, is_adjacent=None, header=None)	Sends a surface message or array of messages to the specified output. *Parameters* **index**: Output index. **offset**: Offset of the data or array of offsets. **scale**: Scale of the data or array of scales. **points**: Surface points data or array points data. Points for uniform data are in an array of size (height, width). Points for point cloud (non-uniform) data are in an array of size (height, width, 3). **intensity**: Surface intensity data of size (height, width), or array of intensity data. **is_adjacent**: Set to true if the surface data is adjacent. Can also be an array of adjacent info. **header**: Header part of the data message. Uses header from the first input if not set.
send_profile(index, offset, scale, points, intensity=None, slices=None, header=None)	Sends a profile message or array of messages to the specified output. *Parameters* **index**: Output index. **offset**: Offset of the data or array of offsets. **scale**: Scale of the data or array of scales. **points**: Profile point data or array of profile points. Data is in array of size (width) or (width,2). **intensity**: Profile intensity data of size (width), or array of profile intensities. **slices**: Not used. **header**: Header part of the data message. Uses header from the first input if not set.
send_measurement(index, value, label_	Sends a measurement message or array of messages to the specified output. Note that the decision can't be set with this function because the decision of a

Function	Description
position=None, header=None)	measurement is only set after output using the Min and Max thresholds on the outputs. *Parameters* **index**: Output index. **value**: Value of the data message, or array of values. Numpy.nan if measurement is invalid. **label_position**: Position of point in measurement message, or array of positions. **header**: Header part of the data message. Uses header from the first input if not set.
send_point(index, position, header=None)	Sends a point message or array of messages to the specified output. *Parameters* **index**: Output index. **position**: Position of feature point in data message, or array of positions. **header**: Header part of the data message. Uses header from the first input if not set.
send_line(index, position, direction, header=None)	Sends a line message or array of messages to the specified output. *Parameters* **index**: Output index. **position**: Point vector in the line, or array of point vectors. **direction**: Direction vector of the line, or array of direction vectors. **header**: Header part of the data message. Uses header from the first input if not set.
send_circle(index, position, normal, radius, header=None)	Sends a circle message or array of messages to the specified output. *Parameters* **index**: Output index. **position**: Center position of the circle, or array of center positions. **normal**: Normal vector of the circular plane, or array of normal vectors. **radius**: Radius of the circle, or array of radii. **header**: Header part of the data message. Uses header from the first input if not set.
send_plane(index, normal, origin_distance, header=None)	Sends a plane message or array of messages to the specified output. *Parameters* **index**: Output index. **normal**: Normal vector of the plane, or array of normal vectors. **origin_distance**: Shortest distance from origin to plane, or array of distances. **header**: Header part of the data message. Uses header from the first input if not set.
send_image(index, pixels, pixel_format= PixelFormat.MONO_8,	Sends an image message or array of messages to the specified output. *Parameters* **index**: Output index.

Function	Description
flipped_x=False, flipped_y=False, transposed=False, header=None)	**pixels**: Pixel data of image of size (height, width, pixel size), or array of image pixels. **pixel_format**: Pixel format of the image, or array of pixel formats. For more information, see *PixelFormat* on page 781. **flipped_x**: Set to `true` if the image data is flipped on the X axis, or array of flags. **flipped_y**: Set to `true` if the image data is flipped on the Y axis, or array of flags. **transposed**: Set to `true` if the image data is transposed, or array of flags. **header**: Header part of the data message. Uses header from the first input if not set.

Logging

Logging a message causes GoPxL to add a message to the log at the bottom of the user interface. There are three types of log messages.

Logging Functions

Function	Description
log_info(msg)	Logs a user informational message. *Parameters* **msg**: Message to be logged.
log_warning(msg)	Logs a user warning message. *Parameters* **msg**: Message to be logged.
log_error(msg)	Logs a user error message and displays a pop-up containing the message. When an error message is logged, all outputs are set to the Invalid state. *Parameters* **msg**: Message to be logged.

Memory

You can use a Script tool's `memory` dictionary for storing values during runtime. The dictionary is erased when scanning starts.

Data Structures

Data Structures

Function	Description
Region	2D Region box (Corner type). *Constructor* Region(x=None, z=None, width=None, height=None) *Parameters* **x**: X position of the center of the region. **z**: Z position of the center of the region. **width**: Width of the region.

Function	Description
	height: Height of the region.
Region3d	3D Region box (Corner type). *Constructor* Region3d(x=None, y=None, z=None, width=None, length=None, height=None, z_angle=None) *Parameters* **x**: X position of the center of the region. **y**: Y position of the center of the region. **z**: Z position of the center of the region. **width**: Width of the region. **height**: Height of the region. **length**: Length of the region. **z_angle**: Z Angle of the region.
Rect3d	3D rectangular region box. *Constructor* Rect3d(x=None, y=None, z=None, width=None, height=None, depth=None) *Parameters* **x**: X position of the center of the region. **y**: Y position of the center of the region. **z**: Z position of the center of the region. **width**: Width of the region. **height**: Height of the region. **depth**: Depth of the region.
Anchor3d	3D anchor (Corner-based). *Constructor* Anchor3d(x=None, y=None, z=None, z_angle=None) *Parameters* **x**: X position of the anchor. **y**: Y position of the anchor. **z**: Z position of the anchor. **z_angle**: Z Angle of the anchor.
Point	2D Point class. *Constructor* Point(x=None, y=None) *Parameters* **x**: X position of the Point. **y**: Y position of the Point.
Point3d	3D Point class. *Constructor* Point3d(x=None, y=None, z=None) *Parameters* **x**: X position of the Point. **y**: Y position of the Point.

Function	Description
	z: Z position of the Point.
Stamp	Represents acquisition metadata associated with a data message. Note that `width`, `height`, `x`, and `y` parameters refer to the dimensions of the camera image; for more information, see `points` in *SurfaceMsg* on page 782. *Constructor* Stamp(frame=None, time=None, signature=None, status=None, state=None, tag=None, encoder=None, encoder_snapshot=None, exposure=None, delay=None, width=None, height=None, y=None, x=None, x_subsampling=None, y_subsampling=None, temperature=None, subframe=None, reserved=None) *Parameters* **signature**: Equal to `kSTAMP_SIGNATURE` for camera messages; otherwise unused. **status**: Digital inputs, gate state, and other information at the time of message capture. **state**: Camera state id associated with message. **tag**: Optional tag that can be used for debugging or other special purposes. **frame**: Frame index of message data (counts up from zero). **time**: Timestamp corresponding to message data (μs). **encoder**: Encoder value corresponding to message data (encoder ticks). **encoderSnapshot**: Encoder value at time of most recent encoder snapshot signal (encoder ticks). **exposure**: Source image exposure (ns). **delay**: Delay before exposure (ns). **width**: Source camera image width, before subsampling. **height**: Source camera image height, before subsampling. **y**: Source camera image Y position. **x**: Source camera image X position. **x_subsampling**: X subsampling amount, expressed as power of 2. **y_subsampling**: Y subsampling amount, expressed as power of 2. **temperature**: Temperature corresponding to message data (centidegrees Celcius). **subframe**: Subframe status information. **reserved**: Reserved for future use.
Transform3d	3D transform matrix. *Constructor* Transform3d(xx, xy, xz, xt, yx, yy, yz, yt, zx, zy, zz, zt) *Parameters* **xx**, **xy**, **xz**, **xt**: X vector (first three) and origin. **yx**, **yy**, **yz**: Y vector and origin. **zx**, **zy**, **zz**: Z vector and origin.
MsgHeader	Header class of data message. *Constructor* MsgHeader(stamp=None, transform=None, bounding_box=None)

Function	Description
	Parameters **stamp**: Data stamp. **transform**: Transform matrix. **bounding_box**: Bounding box for the data.
PixelFormat	Enum class of pixel formats for image data message. **Mono formats** MONO_8 = <PixelFormat.MONO_8: 0x0001> MONO_10 = <PixelFormat.MONO_10: 0x0003> MONO_12 = <PixelFormat.MONO_12: 0x0005> MONO_14 = <PixelFormat.MONO_14: 0x0025> MONO_16 = <PixelFormat.MONO_16: 0x0007> **RGB formats** RGB_8 = <PixelFormat.RGB_8: 0x0014> BGR_8 = <PixelFormat.BGR_8: 0x0015> RGB_10 = <PixelFormat.RGB_10: 0x0018> BGR_10 = <PixelFormat.BGR_10: 0x0019> RGB_12 = <PixelFormat.RGB_12: 0x001A> BGR_12 = <PixelFormat.BGR_12: 0x001B> RGB_14 = <PixelFormat.RGB_14: 0x005E> BGR_14 = <PixelFormat.BGR_14: 0x004A> RGB_16 = <PixelFormat.RGB_16: 0x0033> BGR_16 = <PixelFormat.BGR_16: 0x004B> **YUV/YCBR formats** Y_CB_CR_8_CB_Y_CR = <PixelFormat.Y_CB_CR_8_CB_Y_CR: 0x003A> Y_CB_CR_411_8_CB_YY_CR_YY = <PixelFormat.Y_CB_CR_411_8_CB_YY_CR_YY: 0x003C> Y_CB_CR_422_8_CB_Y_CR_Y = <PixelFormat.Y_CB_CR_422_8_CB_Y_CR_Y: 0x0043> **Bayer formats** BAYER_GR8 = <PixelFormat.BAYER_GR8: 0x0008> BAYER_RG8 = <PixelFormat.BAYER_RG8: 0x0009> BAYER_GB8 = <PixelFormat.BAYER_GB8: 0x000A> BAYER_BG8 = <PixelFormat.BAYER_BG8: 0x000B> BAYER_GR10 = <PixelFormat.BAYER_GR10: 0x000C>

Function	Description
	BAYER_RG10 = <PixelFormat.BAYER_RG10: 0x000D> BAYER_GB10 = <PixelFormat.BAYER_GB10: 0x000E> BAYER_BG10 = <PixelFormat.BAYER_BG10: 0x000F>
ImageMsg	ImageMsg class for image data message. *Constructor* ImageMsg(header=None, pixel_format=None, flipped_x=False, flipped_y=False, transposed=False, pixels=None) *Parameters* **header**: Header part of the data message. **pixel_format**: Pixel format of the image. For more information, see *PixelFormat* on the previous page. **flipped_x**: Set to `true` if the image data is flipped on the X axis. **flipped_y**: Set to `true` if the image data is flipped on the Y axis. **transposed**: Set to `true` if the image data is transposed. **pixels**: Pixel data of image of size (height, width, pixel size).
SurfaceMsg	SurfaceMsg class for surface data message. *Constructor* SurfaceMsg(header=None, offset=None, scale=None, points=None, intensity=None, is_adjacent=None) *Parameters* **header**: Header part of the data message. **offset**: Offset of the data. **scale**: Scale of the data. **points**: Surface point data of size (height, width) or (height,width,3). **intensity**: Surface intensity data of size (height, width). **is_adjacent**: Set to `true` if the surface data is adjacent.
ProfileMsg	ProfileMsg class for profile data message. *Constructor* ProfileMsg(header=None, offset=None, scale=None, points=None, intensity=None) *Parameters* **header**: Header part of the data message. **offset**: Offset of the data. **scale**: Scale of the data. **points**: Profile point data of size (width) or (width,2) . **intensity**: Profile intensity data of size (width).
MeasureMsg	MeasureMsg Class for measurement data message. *Constructor* MeasureMsg(header=None, value=None, status=None, label_position=None) *Parameters* **header**: Header part of the data message. **value**: Value of the data message.

Function	Description
	status: Types from the MeasurementDecision enum class. For more information, see *MeasurementDecision* below. **label_position**: Position of point in measurement message.
MeasurementDecision	Enum class of decision type for measurement data message. FAIL= <MeasurementDecision.FAIL: 0> PASS= <MeasurementDecision.PASS: 1>
PointFeatureMsg	PoinFeaturetMsg class for point feature message. *Constructor* PointFeatureMsg(header=None, position=None) *Parameters* **header**: Header part of the data message. **position**: Position of feature point in data message.
LineFeatureMsg	LineFeatureMsg class for line feature message. *Constructor* LineFeatureMsg(header=None, position=None, direction=None) *Parameters* **header**: Header part of the data message. **position**: Point vector in the line. **direction**: Direction vector of the line.
CircleFeatureMsg	CircleFeatureMsg class for circular feature message. *Constructor* CircleFeatureMsg(header=None, position=None, normal=None, radius=None) *Parameters* **header**: Header part of the data message. **position**: Center position of the circle. **normal**: Normal vector of the circular plane. **radius**: Radius of the circle.
PlaneFeatureMsg	PlaneFeatureMsg class for planar feature message. *Constructor* PlaneFeatureMsg(header=None, normal=None, origin_distance=None) *Parameters* **header**: Header part of the data message. **normal**: Normal vector of the plane. **origin_distance**: Shortest distance from origin to plane.

Examples

Measurement input and output

The following example shows how to read a measurement value input and output a new measurement value that is equal to the input multiplied by a factor of two.

```
# Get the measurement object at input 0
measurement = get_measurement(0)

# Extract the value from the measurement
measurement_value = measurement.value

# Calculate a new measurement value
measurement_output = 2 * measurement_value

# Send the calculated value as a new measurement to output 0
send_measurement(0, measurement_output)
```

The following example shows how to read measurement value inputs and write them to a CSV file. (This example is only intended for use on a PC instance of GoPxL, as writing to files on-sensor or on a GoMax device is blocked.)

```
import csv

# Get measurement objects
measurement_1 = get_measurement(0)
measurement_2 = get_measurement(1)

# Extract values from measurement objects
measurement_1_value = measurement_1.value
measurement_2_value = measurement_2.value

# Path to the output CSV file
output_path = r'C:\GoTools\Script\measurement.csv'

# Write measurements to CSV file
with open(output_path, 'a') as f:
    writer = csv.writer(f)
    writer.writerow([measurement_1_value, measurement_2_value])
```

Math operations

You can access mathematical operations through the standard Python math module.

```
# Import the math module to access mathematical functions
import math

# Get the measurement object at input 0
measurement = get_measurement(0)

# Extract the value from the measurement
measurement_value = measurement.value

# Calculate the square root of the absolute value using the Math module
sqrt_value = math.sqrt(math.fabs(measurement_value))

# Send the square root as a new measurement to output 0
send_measurement(0, sqrt_value)
```

Accessing valid/invalid and decision pass/fail

You can use the `is_valid()` function of the `MeasureMsg` class to check whether the value is valid or invalid. An invalid measurement is represented by the value `numpy.nan`. Other input classes also have `is_valid()` functions, which return `true` if the object data is not null.

The pass/fail criteria for the input `MeasureMsg` can be checked using the `decision` member variable.

```
# Check if any inputs are invalid
if any_input_invalid():
  send_all_invalid()

# Get the input measurement
measurement0 = get_measurement(0)

# Access validity from a measurement
if is_valid(measurement0):
  log_info("The input measurement is Valid")
else:
  log_info("The input measurement is Invalid")

# Access decision from a measurement
if measurement0.decision == MeasurementDecision.PASS:
  log_info("The input measurement has a Pass decision")
else:
  log_info("The input measurement has a Failed decision")
```

Stamp access

```
profile = get_profile(0)

stamp = profile.header.stamp

log_info('Stamp = {}'.format(vars(stamp)))

send_measurement(0, stamp.frame)
send_measurement(1, stamp.time)
send_measurement(2, stamp.encoder)
```

Arrays

The following code reads an array input and outputs an array.

```
# Get an array of measurement objects at input 0
measurements = get_measurement(0)

# Extract the values from each measurement object
```

```
measurement_values = []
for measurement in measurements:
    measurement_values.append(measurement.value)

# Calculate new values
measurements_output = [value * 2 for value in measurement_values]

# Send the calculated values as an array to output 0
send_measurement(0, measurements_output)
```

Inverting a Surface in Z with null handling

The following code inverts a Surface in Z with null handling. It also shifts the output Surface in X by the bounding box width in order to not overlap the original.

```
surf = get_surface(0)

flipped_points = surf.points.copy()
flipped_offset = surf.offset

flipped_points[flipped_points != -32768] *= -1
flipped_offset.x += surf.header.bounding_box.width

send_surface(0, flipped_offset, surf.scale, flipped_points, surf.intensity)
```

Minimum and maximum Z value of a uniform Surface with null handling

```
surf = get_surface(0)

s = surf.points[surf.points != -32768]

max_z = numpy.max(s) * surf.scale.z + surf.offset.z
min_z = numpy.min(s) * surf.scale.z + surf.offset.z

send_measurement(0, min_z)
send_measurement(1, max_z)
```

Grid of coordinates for anchoring

The following script shows how you can perform the same inspection using a single tool at many locations on a grid. It generates X and Y measurement arrays that you use as anchor inputs in another tool (the anchored tool).

In the anchored tool, you must check **Enable batching** so that the tool generates an array of outputs.

```
x = numpy.linspace(-15, 15, 10) # 10 points spanning 0..30 mm range
y = numpy.linspace(-15, 15, 10) # 10 points spanning 0..30 mm range

xv, yv = numpy.meshgrid(x, y)

send_measurement(0, xv.flatten())
send_measurement(1, yv.flatten())
```

Image processing (subtracting two images)

```
im0 = get_image(0)
im1 = get_image(1)

delta = im0.pixels.astype(numpy.float32) - im1.pixels.astype(numpy.float32)
stdev = numpy.std(delta)

new_pixels = numpy.clip(delta / (2*stdev) * 128 + 128, 0, 255).astype(numpy.uint8)

send_image(0, new_pixels, PixelFormat.RGB_8)
send_measurement(1, stdev)
```

Image processing (forward to HTTP server)

```
# Script (using built in urlib package)
import urllib.request
import json
import numpy as np

# Assuming get_image(0).pixels returns a numpy array
array = get_image(0).pixels
array_bytes = array.tobytes()

# Encode shape information as JSON in the headers
headers = {'Content-Type': 'application/octet-stream', 'X-Array-Shape': json.dumps
(array.shape)}
url = 'http://127.0.0.1:5000/receive_array'

# Encode array bytes data and prepare request
req = urllib.request.Request(url, data=array_bytes, headers=headers)
urllib.request.urlopen(req)


# SERVER (server.py)

from flask import Flask, request, Response
import numpy as np
import cv2
import json
```

```python
app = Flask(__name__)

@app.route('/receive_array', methods=['POST'])
def receive_array():
    # Get the content of the request
    data = request.data

    # Extract shape information from headers
    shape_info = json.loads(request.headers.get('X-Array-Shape'))
    height, width, channels = shape_info

    # Extract image bytes
    image_array = np.frombuffer(data, dtype=np.uint8)
    image_array = image_array.reshape((height, width, channels))

    # Display image
    #cv2.imshow('Received Image', image_array)
    #cv2.waitKey(0)
    #cv2.destroyAllWindows()

    # Print size of received image
    print("Received image size:", image_array.shape)

    return Response(status=200)

if __name__ == '__main__':
    app.run(debug=False, threaded=True, port=5000)
```

The following is a sample client to demonstrate the functionality of the HTTP server forwarding snippet above.

```python
# SAMPLE CONSOLE CLIENT (client.py)
# A standalone demonstration of the functionality used by the image HTTP
# forwarding script.

import requests
import numpy as np
import json

# Assuming you have a function to generate numpy arrays as images
def generate_numpy_array():
    # Generating a sample numpy array
    array = np.random.rand(100, 100, 3) * 255  # Random 100x100x3 array of floats between
0 and 255
    return array.astype(np.uint8)

def send_array_to_server(array):
    url = 'http://127.0.0.1:5000/receive_array'
    array_bytes = array.tobytes()

    # Encode shape information as JSON in the headers
    headers = {'Content-Type': 'application/octet-stream', 'Array-Shape': json.dumps
(array.shape)}

    response = requests.post(url, data=array_bytes, headers=headers)
```

```
# Generate and send arrays to the server
while True:
    array = generate_numpy_array()
    send_array_to_server(array)
```

Profile processing (using a profile from disk as a correction to new profiles from sensor)

```
# Save profile template to disk
import csv

profile = get_profile(0)
points = profile.points

template_path = r'C:\GoTools\Script\profile.csv'
numpy.savetxt(template_path, points, delimiter=',', fmt='%d')

send_measurement(0, 1)

# Load and correct
import csv

profile = get_profile(0)
points = profile.points

# load template
template_path = r'C:\GoTools\Script\profile_template.csv'

if 'template' not in memory:
    template_points = numpy.genfromtxt(template_path, delimiter=',', dtype=numpy.int16)
    memory['template'] = template_points
    log_info("Loaded template")
else:
    template_points = memory['template']

# subtract while handling null values
mask = (template_points != -32768) & (points != -32768)
result = numpy.full_like(points, -32768)
numpy.subtract(points, template_points, out=result, where=mask, dtype=numpy.int16)

send_profile(0, profile.offset, profile.scale, result)

# optionally send original profile and template as outputs
send_profile(1, profile.offset, profile.scale, points)
send_profile(2, profile.offset, profile.scale, template_points)

# optionally save all three into a new CSV file

output_path = r'C:\GoTools\Script\profile_log.csv'
combined_array = numpy.vstack((points, template_points, result)).T

with open(output_path, 'w', newline='') as csv_file:
    writer = csv.writer(csv_file)
    writer.writerow(['Profile', 'Template', 'Difference'])
    writer.writerows(combined_array)
```

Getting function signatures

Instead of looking up functions in the manual to see arguments to be passed in, you can use the `signature` function in the `inspect` module to get the signature of any function. For example, the code below displays an error message: (`index, value, label_position=None, header=None`).

```
from inspect import signature

log_error(str(signature(send_measurement)))
```

FTP - Writing numpy or CSV files to disk, and sending them to an FTP server

```
import numpy as np
import csv
import os
from ftplib import FTP

from datetime import datetime

surface = get_surface(0)

xRes = surface.scale.x
yRes = surface.scale.y
zRes = surface.scale.z
xOff = surface.offset.x
yOff = surface.offset.y
zOff = surface.offset.z
length = surface.points.shape[0]
width = surface.points.shape[1]
size = width*length

# scale Z points to engineering units
Z = surface.points.copy()
Z = Z.astype(np.double)
Z.setflags(write=1)
Z[Z==-32768] = np.nan
Z = Z*zRes+zOff
Z = np.round(Z, 10)

# 1D array example:
# generate X array
Xarr = (np.asarray(range(width), dtype=np.double) * xRes) + xOff
Xarr = np.tile(Xarr, length)
# generate Y array
Yarr = (np.arange(length, dtype=np.double)* yRes) + yOff
Yarr = np.repeat(Yarr, repeats=width)
# flatten and generate Z array
Zarr = Z
Zarr = Zarr.flatten()
Zarr = (Zarr * zRes) + zOff
# stack arrays
data_3DXYZ = np.stack((Xarr,Yarr,Zarr), axis = 1)
```

```
data_3DXYZ = data_3DXYZ.round()

# generate local path
now = datetime.now()
unique_filename = now.strftime("%Y-%m-%d_%H-%M-%S_XYZ.npy")
documents_path = os.path.join(os.environ['USERPROFILE'], 'Documents')
writePath = os.path.join(documents_path,unique_filename)

# save numpy file, default example
np.save(writePath,data_3DXYZ)

# save csv file
#with open(writePath,'w',newline='') as csvfile:
#    writer = csv.writer(csvfile,delimiter=',')
#    writer.writerow(["X","Y","Z"])
#    writer.writerows(data_3DXYZ)


# 2D array example:
'''
# Custom header elements for post processing
header_title_elements = ["xRes", "yRes", "zRes",
                         "xOff", "yOff", "zOff",
                         "length","width","size"]
header_title_string = ','.join(header_title_elements)

header_elements = [str(xRes), str(yRes), str(zRes),
                   str(xOff), str(yOff),str(zOff),
                   str(length), str(width), str(size)]
header_data_string = ','.join(header_elements)

# generate local path
now = datetime.now()
unique_filename = now.strftime("%Y-%m-%d_%H-%M-%S_XYZ.csv")
documents_path = os.path.join(os.environ['USERPROFILE'], 'Documents')
writePath = os.path.join(documents_path,unique_filename)

# Write the headers to the file
#with open(writePath, 'w') as f:
#    f.write(header_title_string + '\n')  # First line of headers (titles)
#    f.write(header_data_string + '\n')   # Second line of headers (actual values)

#with open(writePath, 'ab') as f:  # Open as binary to avoid issues across different
Python versions
#    np.savetxt(f, Z, delimiter=',', fmt='%s')
'''

#Send file over FTP

# FTP server details
ftp_host = '127.0.0.1'
ftp_username = 'tester'
ftp_password = 'password'
file_path = writePath  # The local path to your file
remote_path = '/' + unique_filename  # The remote path where you want to upload the file

# Establish FTP connection and login
if 'FTP' not in memory:
    ftp = FTP(ftp_host)
```

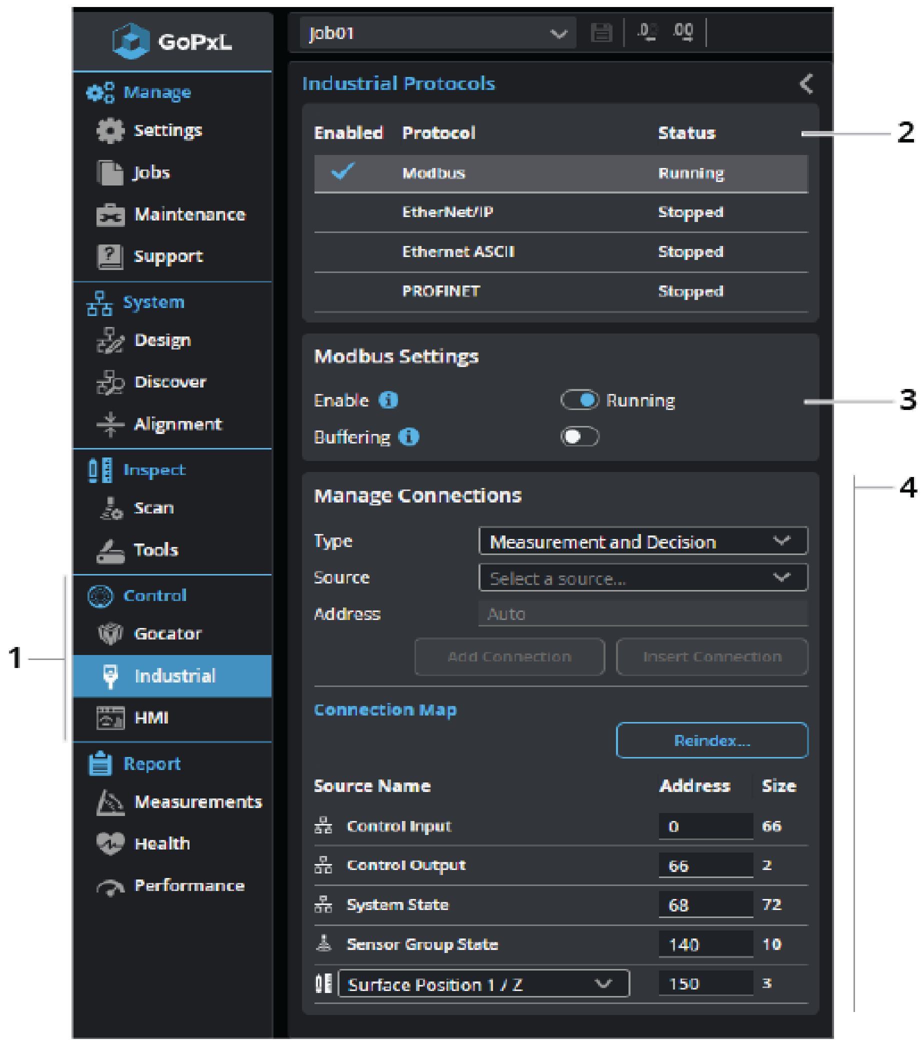
```
    ftp.login(ftp_username, ftp_password)
    memory['FTP'] = ftp
ftp = memory['FTP']

try:
    # Open the file in binary read mode
    with open(file_path, 'rb') as file:
        ftp.storbinary(f'STOR {remote_path}', file)
except Exception as e:
    log_info(f"An error occurred: {e}")
```

Configuring Control

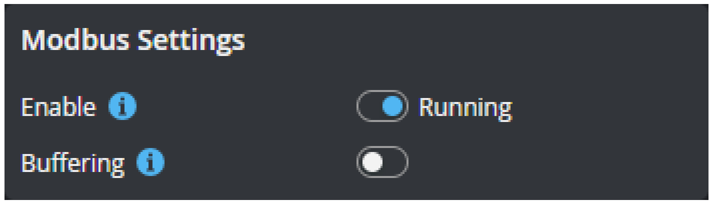
External devices such as PLCs (programmable logic controllers) can control Gocator sensors using various communication protocols and hardware interfaces. Sensors can also send scan data, measurement results, and measurement decisions to these devices. PLCs can in turn control other devices such as ejection chutes.

You can also use the Gocator communication protocol with software written using the GoPxL SDK and REST API to control a sensor and receive data.

The following sections describe the **Control** category and the available protocols and interfaces.

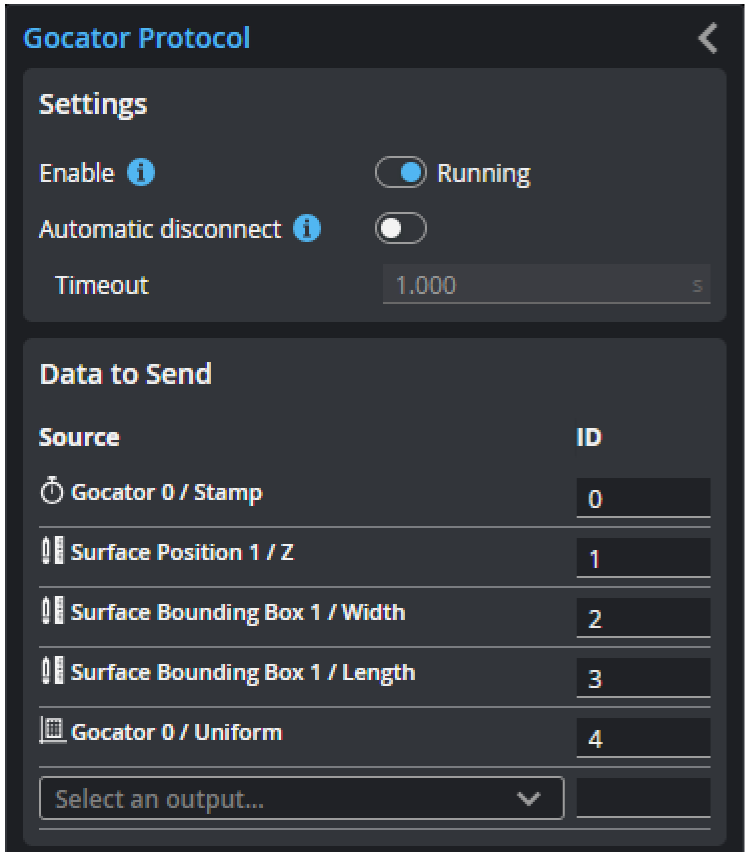


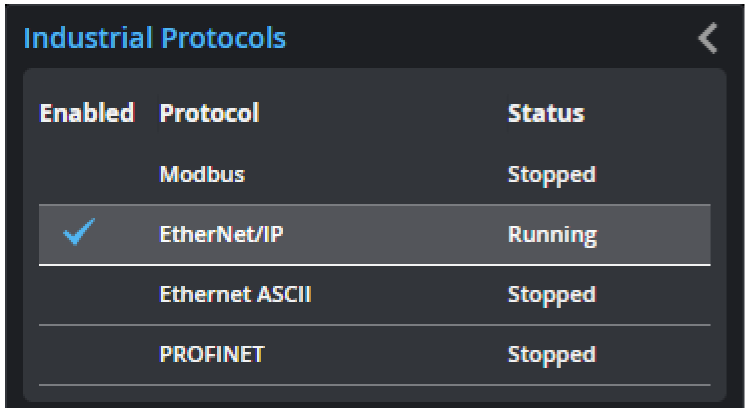
	Element	Description
1	Control pages	Control category and pages corresponding to the different control and data output methods available in GoPxL. For information on the available control methods, see *Control interfaces* on the next page.

	Element	Description
2	Protocol selector	When you are on the Industrial page, displays the available industrial protocols and whether a protocol is enabled. Click a protocol in the table and toggle **Enable** to Running in the Settings section below..
3	Settings	Lets you enable, disable, and configure protocols. To enable a service, click the **Enable** toggle to Running. 
4	Manage Connections	Lets you add and remove the connections (such as stamps, measurements and decisions) to the connections map (industrial protocols) and outputs (Gocator protocol).

Control interfaces

Interface	Description
Gocator	The **Control** > **Gocator** page lets you enable and configure the Gocator communication protocol. You use this protocol in conjunction with the GoPxL SDK and REST API. The protocol uses TCP messages to control the sensor or sensors and to transmit data and measurement results to a client computer. You can select which measurements and what type of scan data to send (image; range, profile, or surface, depending on sensor model; or intensity).

Interface	Description
	 For information on configuring the Gocator communication protocol in GoPxL, see *Gocator Communication Protocol* on page 797. For information on the SDK and REST API, see *GoPxL SDK and REST API* on page 871.
Industrial	The Industrial pages lets you enable and configure the following protocols over an Ethernet connection: • Modbus • EtherNet/IP • Ethernet ASCII • PROFINET

<table>
<tr><th>Interface</th><th>Description</th></tr>
<tr><td></td><td>


Modbus

The Modbus protocol lets you communicate with a sensor from a PLC that supports Modbus.

For more information on configuring the Modbus protocol in GoPxL, see *Modbus Protocol* on page 799.

For the Modbus protocol specification, see *Modbus Protocol* on page 897.

EtherNet/IP

The EtherNet/IP protocol lets you communicate with a sensor from a PLC that supports EtherNet/IP.

For more information on configuring the EtherNet/IP protocol in the GoPxL UI, see *EtherNet/IP Protocol* on page 803.

For the EtherNet/IP protocol specification, see *EtherNet/IP Protocol* on page 885.

Ethernet ASCII

The Ethernet ASCII protocol lets you communicate with a sensor using an ASCII format.

For more information on configuring the ASCII protocol over Ethernet in the GoPxL interface, see *Ethernet ASCII Protocol* on page 808.

For a list of the commands, see *Ethernet ASCII Protocol* on page 902.

PROFINET

The PROFINET protocol lets you communicate with a sensor from a PLC that supports PROFINET.

For more information on configuring the PROFINET protocol in the GoPxL UI, see *PROFINET Protocol* on page 813.

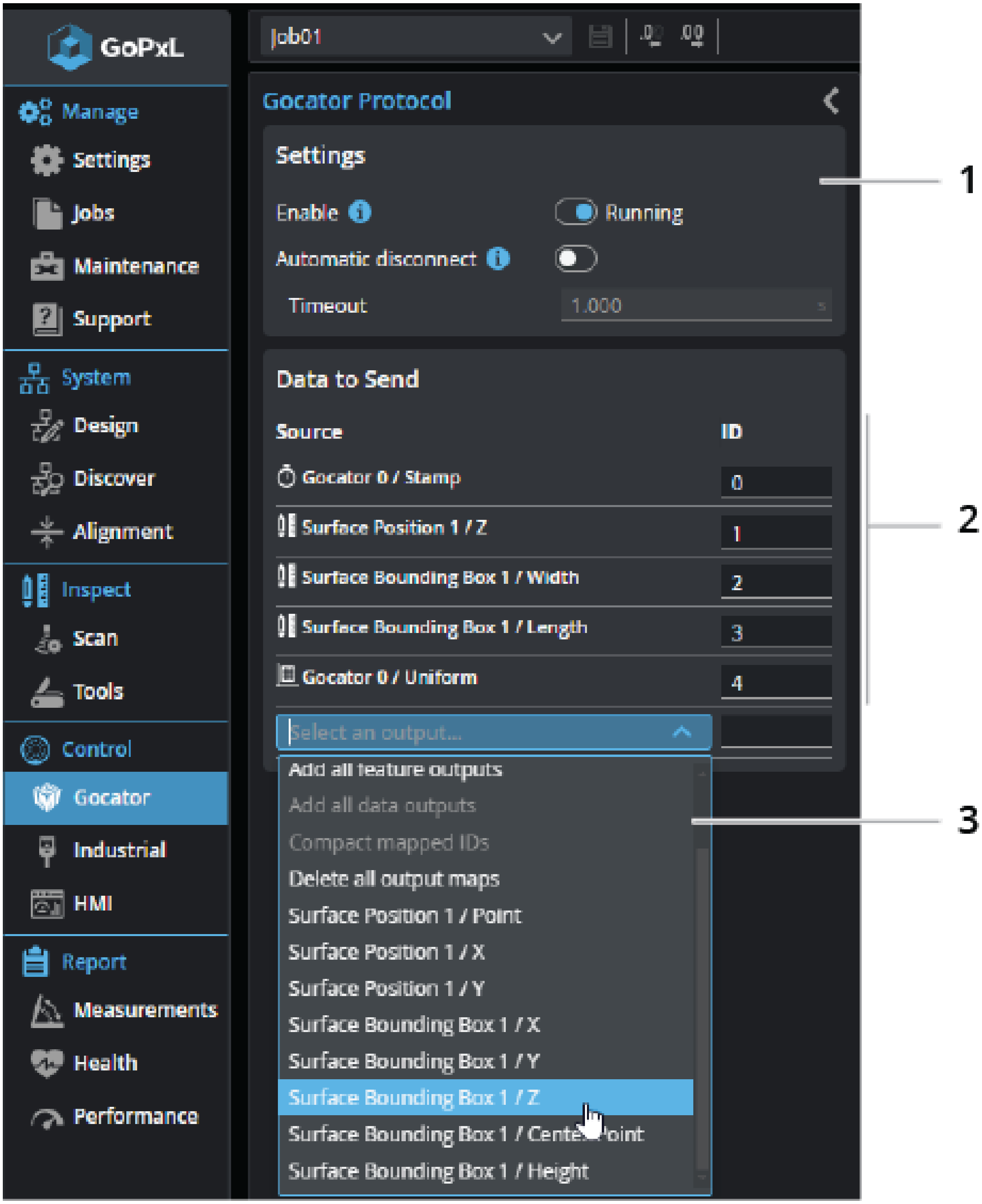
For the PROFINET protocol specification, see *PROFINET Protocol* on page 892.
</td></tr>
<tr><td>GoHMI</td><td>GoHMI and the GoHMI Designer let you create a custom user interface connected</td></tr>
</table>

Interface	Description
	to Gocator outputs that users can access from a browser on a PC, a touchscreen, or a mobile device. For more information, see *GoHMI and GoHMI Designer* on page 844.

Gocator Communication Protocol

The Gocator communication protocol uses TCP messages to control sensors and to transmit data and measurement results to a client computer. You can select which measurements and what type of scan data to send (image; range, profile, or surface, depending on sensor model; or intensity).

All of the tasks that you can accomplish in the web interface can be accomplished programmatically by sending and receiving the protocol control commands, with the SDK or the REST API. For more information on the SDK and the REST API, see *GoPxL SDK and REST API* on page 871.



	Element	Description
1	Protocol settings	Lets you enable the protocol and configure automatic disconnect. For information on the settings available here, see *Configuration area parameters* on

	Element	Description
		the next page.
2	Added outputs	Outputs that have been added from the output selection list. You can manually change the ID.
3	Output selection list	You use the output selection list to choose the outputs you want to send over the protocol. The list also provides shortcut commands to add or remove outputs. For example, the **Add all measurement outputs** shortcut adds all available measurement outputs.

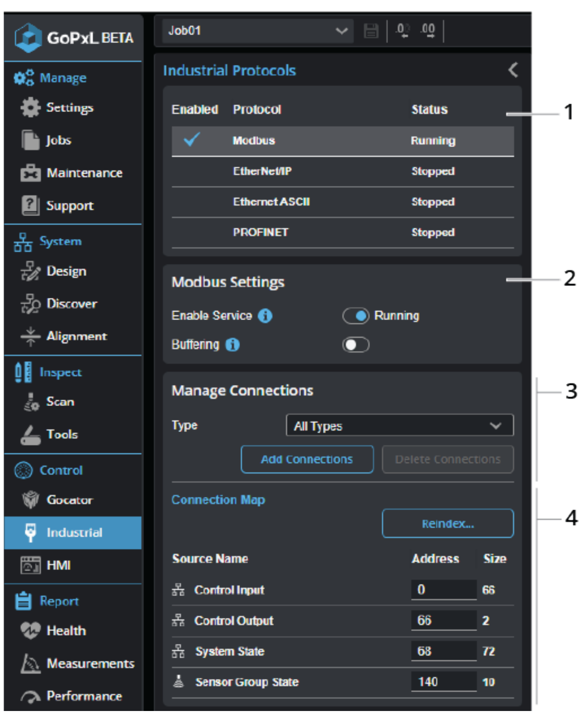
Configuration area parameters

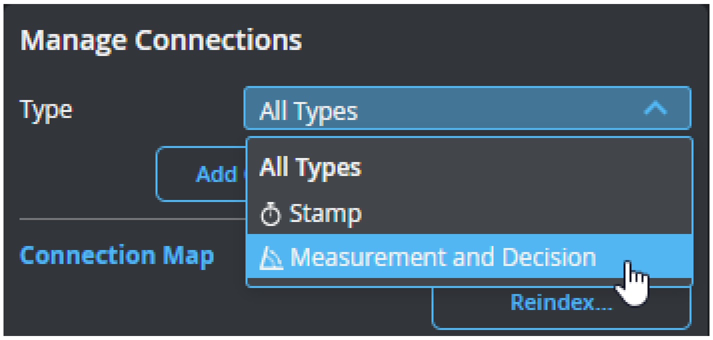
Parameter	Description
Enable	Toggles whether the Gocator communication protocol is enabled.
Automatic disconnect	When enabled, automatically disconnects from the client after the timeout value in **Timeout** if the sensor is unable to send data.
Timeout	Sets the timeout value when **Automatic disconnect** is enabled.

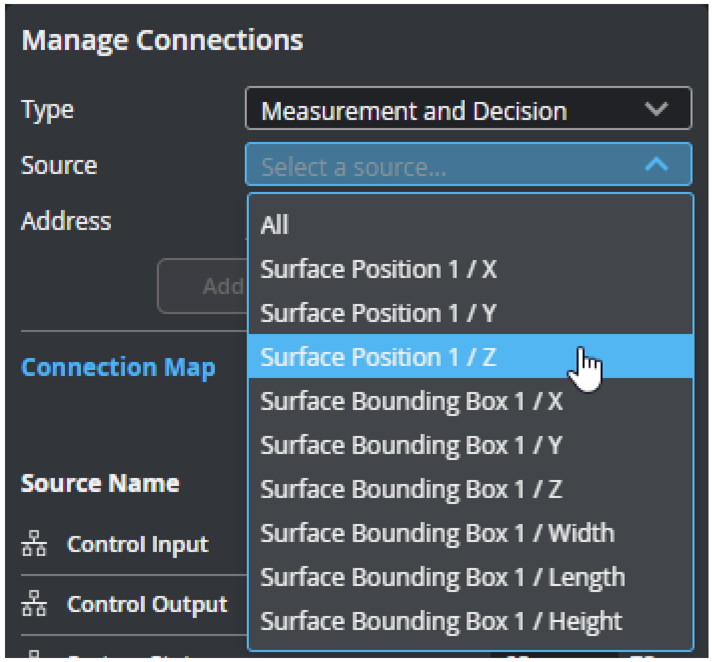
Modbus Protocol

You can use a Modbus programmable logic controller (PLC) over Ethernet to operate a sensor and receive measurement values. Modbus only supports a subset of the tasks that can be accomplished in the web interface (for example, starting and stopping sensors, and switching jobs). Scan data can't be sent to the PLC.

You configure the protocol on the **Control** > **Industrial** page. After configuring the Modbus protocol in GoPxL (see below), use the information in *Modbus Protocol* on page 897 to configure your PLC.



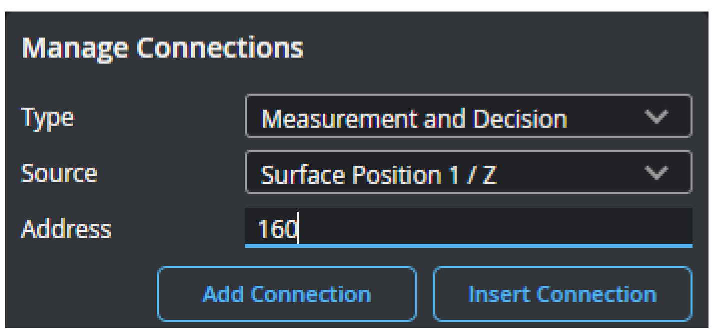
	Element	Description
1	Protocol selector	Click a row in the table to choose which protocol to configure.
2	Settings	Lets you enable the Modbus protocol and buffering. **Enable** Toggle this to enable and disable the selected protocol. **Buffering** Used to enable/disable buffering of scan outputs. On line profile sensors, buffering should be enabled in Surface mode when **Surface Generation Type** is set to Continuous mode and if multiple objects may be detected within a time frame shorter than the polling rate of the PLC. If buffering is enabled, the PLC must send the 'Buffer Advance' command to advance the queue before reading the measurement results. For more information, see *Control Input* on page 899. For information on continuous surface generation and its settings, see *Surface Generation* on page 199.
3	Manage Connections	Lets you add, insert, and delete sources (such as stamps or measurements) in the connection map. Added connections appear in the connection map at the bottom of the panel. **Type** Use **Type** to optionally filter the connection types available to be added or inserted.  When **Type** is set to a type other than All Types, sources of that type are listed for selection in the **Source** drop-down. Choose All to add all sources of that type.

Element	Description
	

Add Connections

Adds one or more connections to the connection map. If **Type** is set to All Types, all available sources are added. If **Type** is set to a specific type, the source selected in **Source** is added (or all sources of that type if All is selected in **Source**).

By default, connections are added directly after the last source in the connections map. You can optionally set the address at which a source is added using the **Address** parameter.



Insert Connections

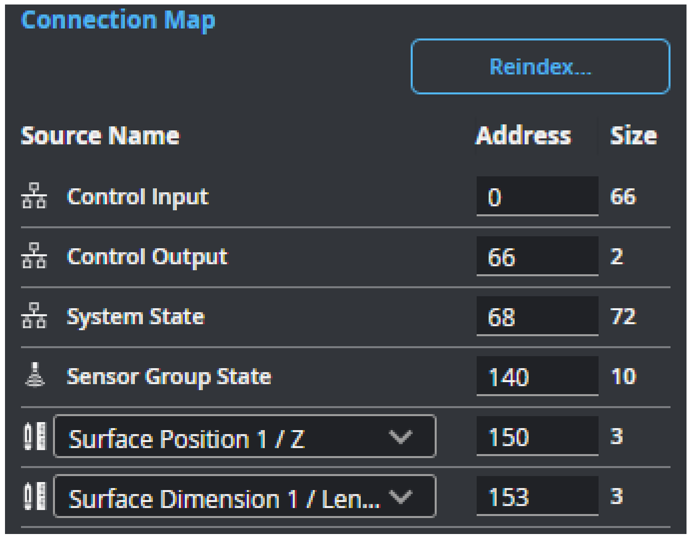
Set **Address** to the address of a previously added source and click **Insert Connection** to insert the selected source into the connection map at that address and shift others down. Inserting a source at an address not currently used adds it at that address.

<table>
<tr><th></th><th>Element</th><th>Description</th></tr>
<tr><td></td><td></td><td>

<u>Delete Connections</u>

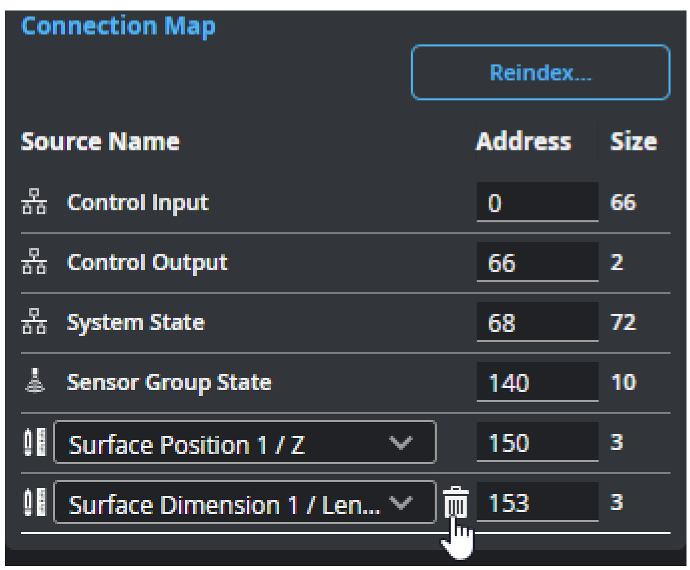
Removes all user-added connections.
</td></tr>
<tr><td>4</td><td>Connection map</td><td>

Lists the added connections. In the following, two measurements have been added.

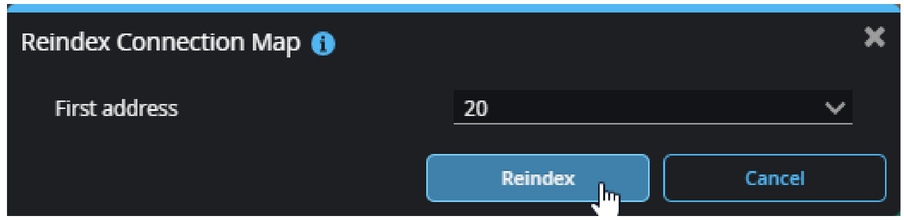


You can change the address of a source by editing it and pressing Tab or clicking elsewhere in the interface. The sources are sorted automatically after you change an address.

To remove a source from the connections map, hover over the source and click the source's trashcan icon.



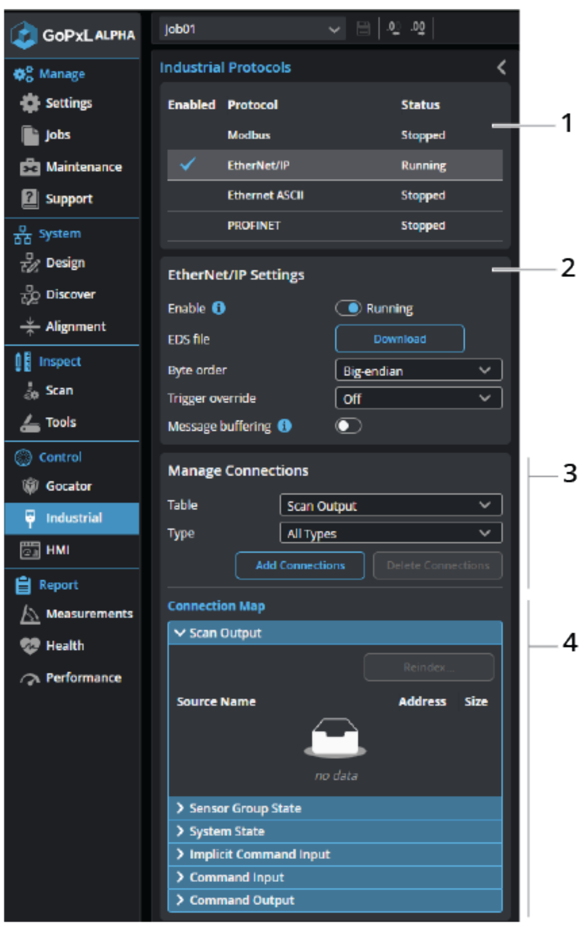
Use **Reindex...** to remove gaps of unused addresses between connections. You
</td></tr>
</table>

Element	Description
	can also use **Reindex...** to set the value of the first index. 

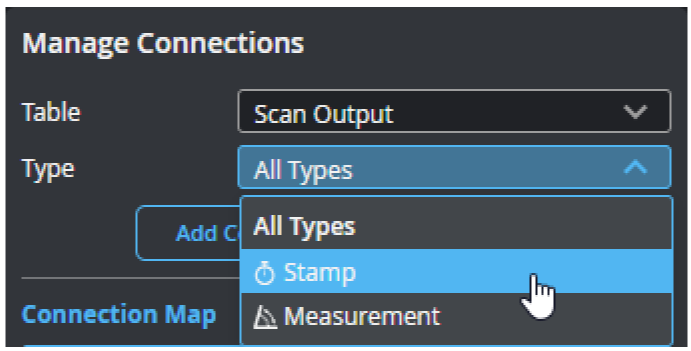
EtherNet/IP Protocol

You can use an EtherNet/IP programmable logic controller (PLC) over Ethernet to operate a sensor and receive measurement values. EtherNet/IP only supports a subset of the tasks that can be accomplished in the web interface (for example, starting and stopping sensors, and switching jobs). Scan data can't be sent to the PLC.

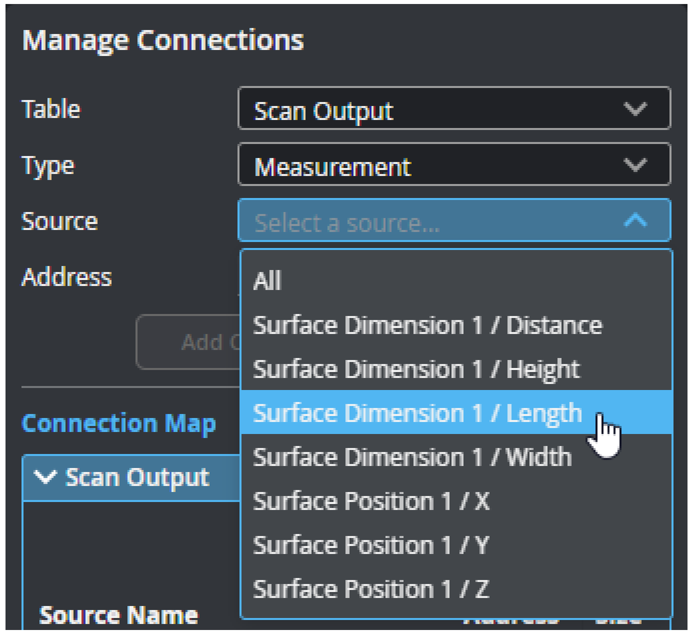
You configure the protocol on the **Control** > **Industrial** page. After configuring the EtherNet/IP protocol in GoPxL (see below), use the information in *EtherNet/IP Protocol* on page 885 to configure your PLC.



	Element	Description
1	Protocol selector	Click a row in the table to choose which protocol to configure.
2	Settings	Lets you enable and configure the EtherNet/IP protocol. **Enable** Toggle this to enable and disable the selected protocol. **EDS file** Click **Download** to save an EDS file to the client computer for later installation on the PLC. **Byte order** One of the following: **Big-endian**: The most significant byte (the "big end") of the data is placed at the byte with the lowest address. The rest of the data is placed in order of decreasing significance in the next three bytes of memory. **Little-endian**: The least significant byte (the "little end") of the data is placed at the byte with the lowest address. The rest of the data is placed in order of increasing significance in the next three bytes in memory. Most Allen-Bradley PLCs default to Little Endian addressing formats, but you should verify this when configuring the PLC. **Trigger override** One of the following: **Off**, **Cyclic**, and **Change of State**. When you set up the PLC to communicate with a Gocator using change of state implicit messaging, an event task must be created on the PLC to rapidly check whether the sensor is running; if the frame count increases, data is copied to an array. The event task period must allow the event task to be executed at a higher rate than Gocator frame rate. **Message buffering** Used to enable and disable buffering of scan outputs. You should enable buffering if you are using part detection and if multiple objects may be detected within a period shorter than the polling rate of the PLC. If buffering is enabled with the EtherNet/IP protocol, the buffer is automatically advanced when the Sample State assembly object is read (see *EtherNet/IP Protocol* on page 885). Message buffering only applies to tool outputs sent over explicit messaging. For more information on part detection, see Continuous in *Surface Generation* on page 199.

	Element	Description
3	Manage Connections	Lets you add, insert, and delete sources (such as stamps or measurements) in the connection map. Added connections appear in the connection map at the bottom of the panel. **Table** When GoPxL is running on a sensor, **Table** only contains one value (Scan Output) that represents a register assembly. When GoPxL is running on a PC or GoMax NX, **Table** contains an additional entry (Sensor Group State). Use this parameter to choose which register a source is added to. **Type** Use **Type** to optionally filter the connection types available to be added or inserted.  When **Type** is set to a type other than All Types, sources of that type are listed for selection in the **Source** drop-down. Choose All to add all sources of that type.

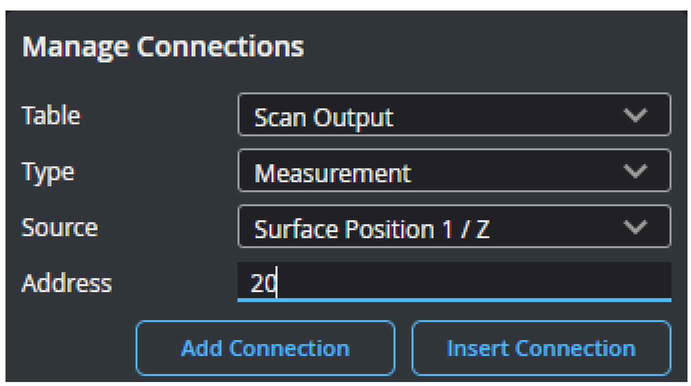
Element	Description



Add Connections

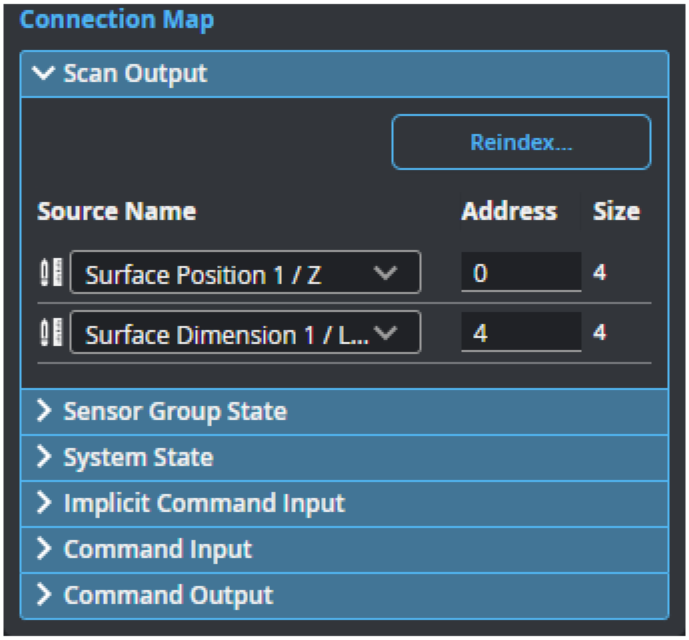
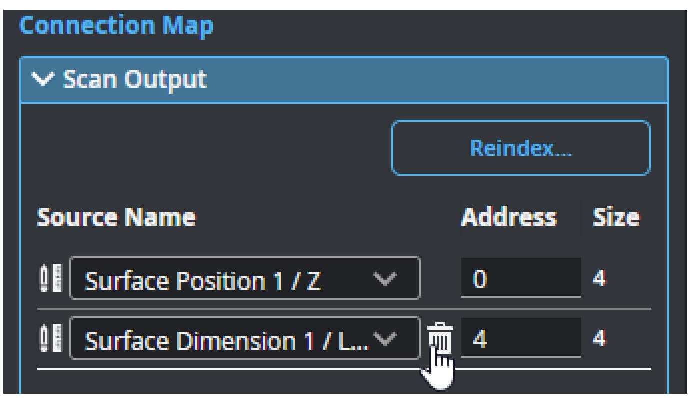
Adds one or more connections to the connection map. If **Type** is set to All Types, all available sources are added. If **Type** is set to a specific type, the source selected in **Source** is added (or all sources of that type if All is selected).

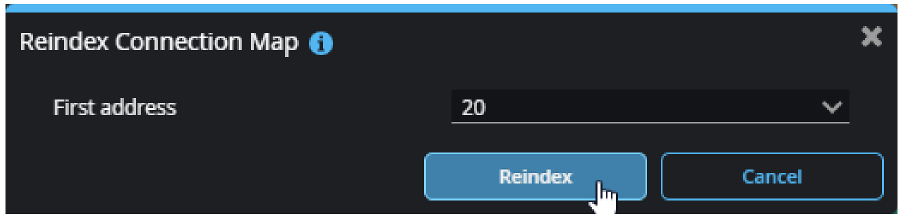
By default, connections are added directly after the last source in the connections map. You can optionally set the address at which a source is added using the **Address** parameter.



Insert Connections

Set **Address** to the address of a previously added source and click **Insert Connection** to insert the selected source into the connection map at that address and shift others down. Inserting a source at an address not currently used adds it at that address.

	Element	Description
		Delete Connections Removes all user-added connections.
4	Connection map	Lists the added connections. In the following, two measurements have been added.  You can change the address of a source by editing it and pressing Tab or clicking elsewhere in the interface. The sources are resorted after you change an address. To remove a source from the connections map, hover over the source and click the source's trashcan icon.  Use **Reindex...** to remove gaps of unused addresses between connections. You can also use **Reindex...** to set the value of the first index.

Element	Description
	

Ethernet ASCII Protocol

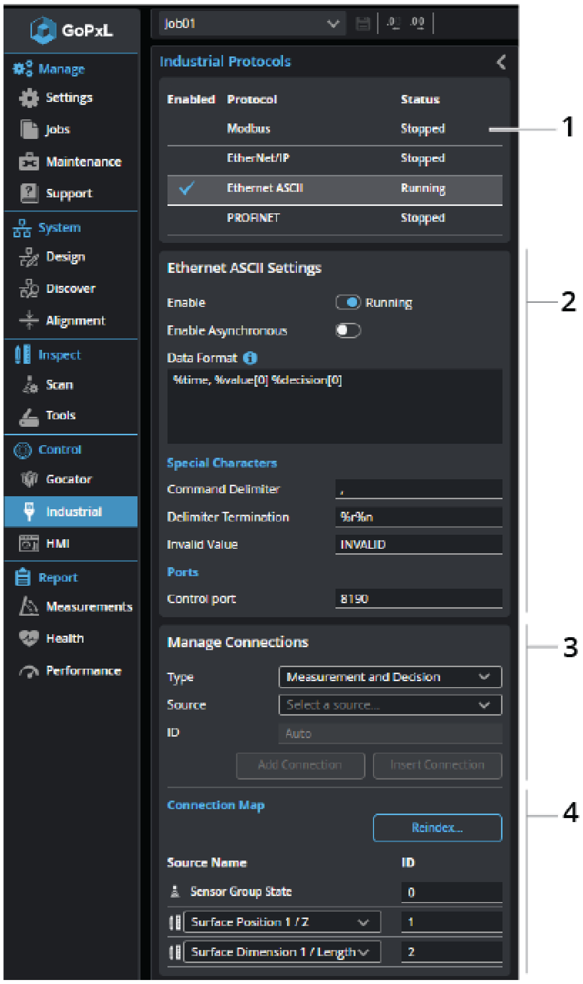
You can use a Programmable Logic Controller (PLC) that supports the Ethernet ASCII protocol to operate a sensor and receive values and decisions (together called results).

Two modes are available: asynchronous and polling.

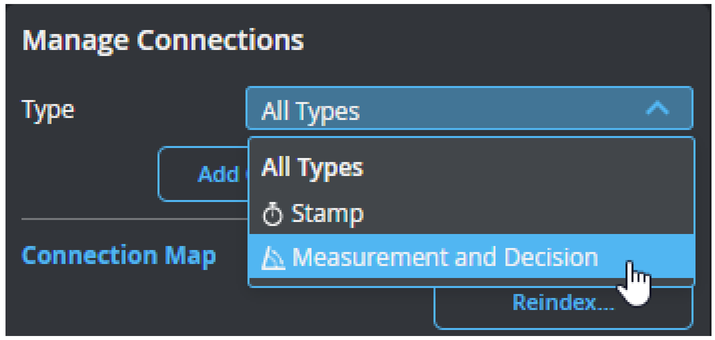
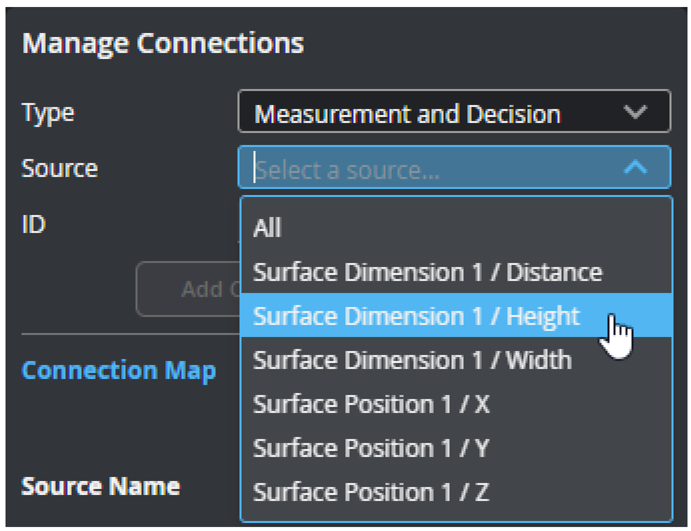
When the sensor is in asynchronous mode, results are sent automatically, as soon as they are available. No commands are required.

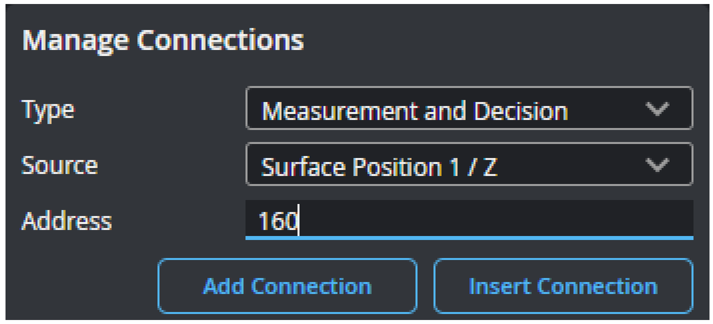
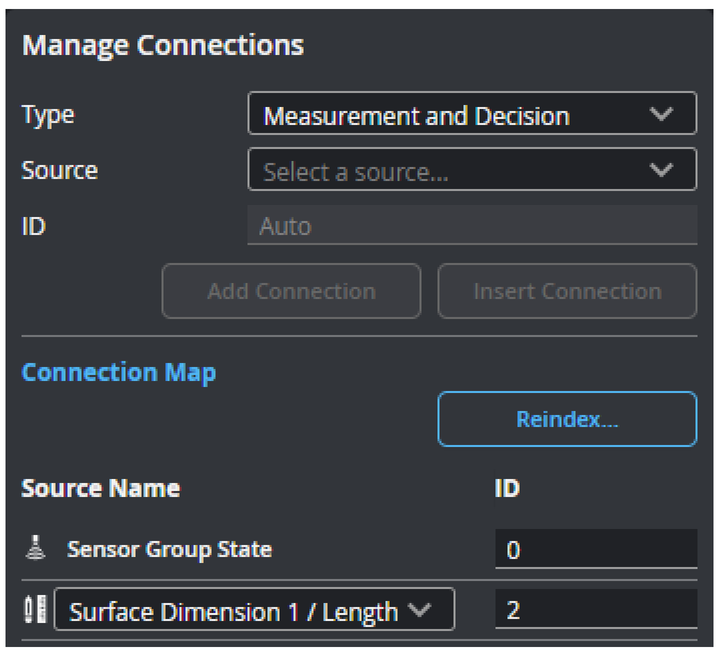
When the sensor is in polling mode, you use commands on the PLC to operate the sensor and to retrieve results. Ethernet ASCII protocol in polling mode only supports a subset of the tasks that can be accomplished in the web interface, and only measurement results can be transmitted to the PLC (scan data is not sent).

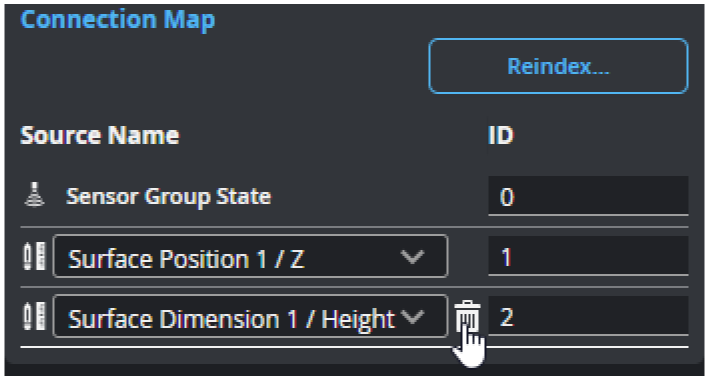
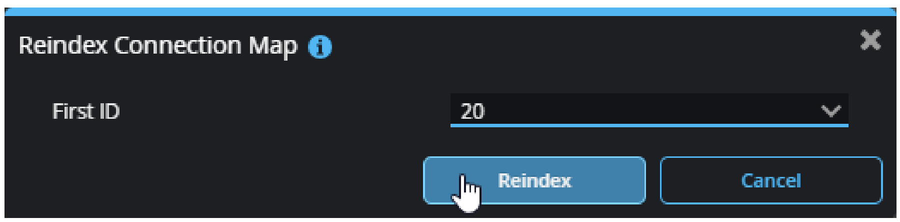
You configure the protocol on the **Control** > **Industrial** page. After configuring the ASCII protocol parameters in GoPxL (see below), see *Ethernet ASCII Protocol* on page 902 for information on the available polling commands.



	Element	Description
1	Protocol selector	Click a row in the table to choose which protocol to configure.
2	Settings	Lets you enable and configure the Ethernet ASCII protocol. For descriptions of the parameters, see *Configuration parameters* on page 812.
3	Manage Connections	Lets you add, insert, and delete sources (such as stamps or measurements) in the connection map. Added connections appear in the connection map at the bottom of the panel. **Type** Use **Type** to optionally filter the connection types available to be added or inserted.

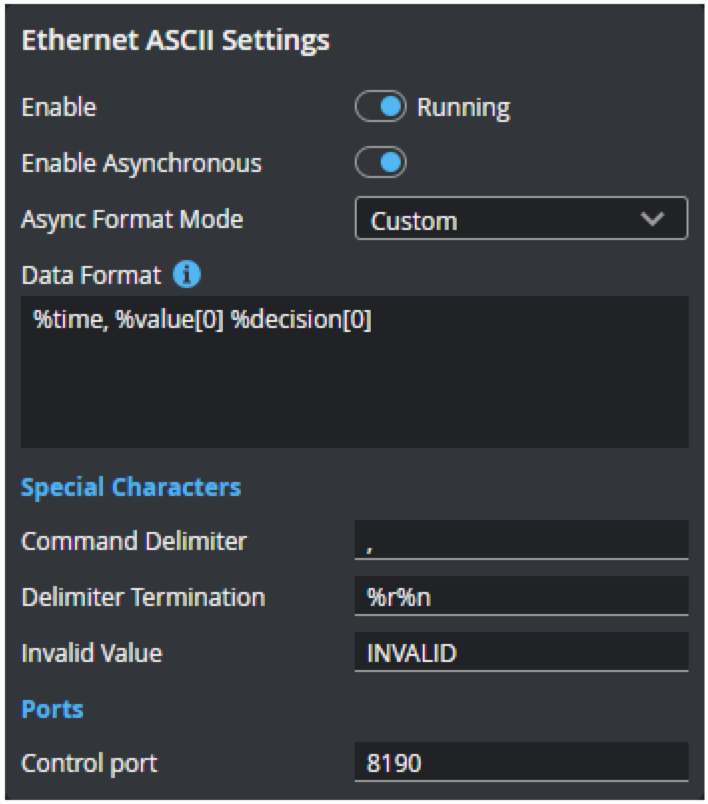
Element	Description
	 When **Type** is set to a type other than All Types, sources of that type are listed for selection in the **Source** drop-down. Choose All to add all sources of that type.  **Add Connections** Adds one or more connections to the connection map. If **Type** is set to All Types, all available sources are added. If **Type** is set to a specific type, the source selected in **Source** is added (or all sources of that type if All is selected). By default, connections are added directly after the last source in the connections map. You can optionally set the address at which a source is added using the **Address** parameter.

	Element	Description
		 **Insert Connections** Set **Address** to the address of a previously added source and click **Insert Connection** to insert the selected source into the connection map at that address and shift others down. Inserting a source at an address not currently used adds it at that address. **Delete Connections** Removes all user-added connections.
4	Connection map	Lists the added connections. In the following, two measurements have been added.  You can change the ID of a source by editing it and pressing Tab or clicking elsewhere in the interface. The sources are resorted after you change an address. To remove a source from the connections map, hover over the source and click

Element	Description
	the source's trashcan icon.  Use **Reindex...** to remove gaps of unused addresses between connections. You can also use **Reindex...** to set the value of the first index. 

Configuration parameters

Parameter	Description
Enable	Toggles whether the protocol is enabled.
Enable Asynchronous	Toggles asynchronous operation. In asynchronous mode, the data results are transmitted when they are available. Set the **Async Format Mode** and **Async Data Format** parameters as required. (These parameters are only displayed when asynchronous mode is enabled.) When this option is disabled, the protocol runs in polling (or synchronous) mode, and you must send commands on the data channel to request the latest result. For a list of the polling commands, see *Commands and Formats* on page 903.
Async Format Mode	The formats used when asynchronous mode is enabled.
Data Format	One of the following Standard, Standard with Stamp, and Custom. Setting this parameter to Custom lets you edit the format in **Data Format**. Click the info icon for a list of replacement patterns. This format is used when you use the Result command without providing any arguments; for more information, see *Commands and Formats* on page 903. Otherwise, the built-in format is used. C language *printf*-style formatting is also supported: for example, %sprintf[%09d, %value[0]]. This allows fixed length formatting for easier input parsing in PLC and robot controller logic.

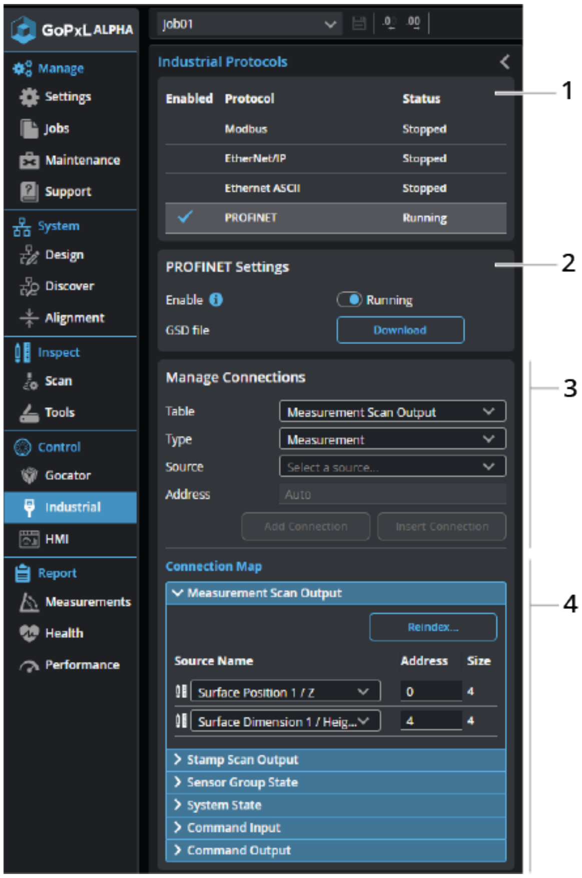
Parameter	Description
	
Command Delimiter	Sets the command delimiter character.
Delimiter Termination	Sets the command termination character.
Invalid Value	Sets the invalid value characters.
Control Port	Sets the control port. This is the port to which you must connect.

PROFINET Protocol

You can use a PROFINET Programmable Logic Controller (PLC) over Ethernet to operate a sensor and receive measurement values. PROFINET only supports a subset of the tasks that can be accomplished in the web interface (for example, starting, stopping, and aligning sensors, and switching Jobs). Scan data can't be sent to the PLC.

> GoPxL supports PROFINET on-sensor and on GoMax NX. PROFINET is not currently supported on PC instances of GoPxL. This functionality is planned for the near future.

You configure the protocol on the **Control** > **Industrial** page. After configuring the PROFINET protocol in GoPxL (see below), use the information in *PROFINET Protocol* on page 892 to configure your PLC.



	Element	Description
1	Protocol selector	Click a row in the table to choose which protocol to configure.
2	Settings	Lets you enable and configure the PROFINET protocol. **Enable** Toggle this to enable and disable the selected protocol. **GSD File** Click **Download** to save a GSD file to the client computer for later installation on the PLC.
3	Manage Connections	Lets you add, insert, and delete sources (such as stamps or measurements) in the connection map. Added connections appear in the connection map at the bottom of the panel. **Table** When GoPxL is running on a sensor, **Table** only contains two options (Measurement Scan Output and Stamp Scan Output), which represent register assemblies. When GoPxL is running on GoMax NX, **Table** contains an additional

<table>
<tr><th></th><th>Element</th><th>Description</th></tr>
<tr><td></td><td></td><td>
entry (Sensor Group State). Use this parameter to choose which register a source is added to.

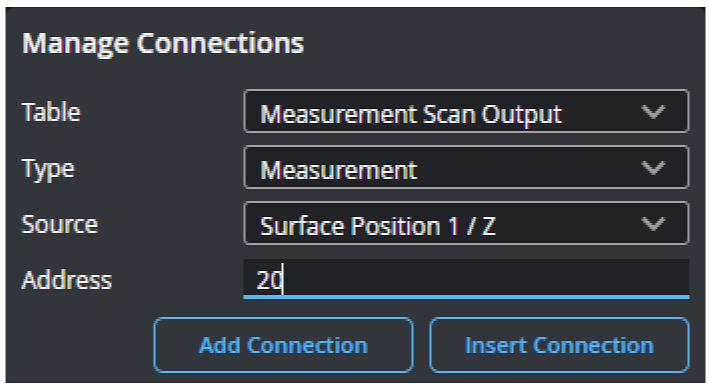
<u>**Type**</u>

The types of connection available to be added or inserted.

<u>**Add Connections**</u>

Adds one or more connections to the connection map. If **Type** is set to All Types, all available sources are added. If **Type** is set to a specific type, the source selected in **Source** is added (or all sources of that type if All is selected).

By default, connections are added directly after the last source in the connections map. You can optionally set the address at which a source is added using the **Address** parameter.

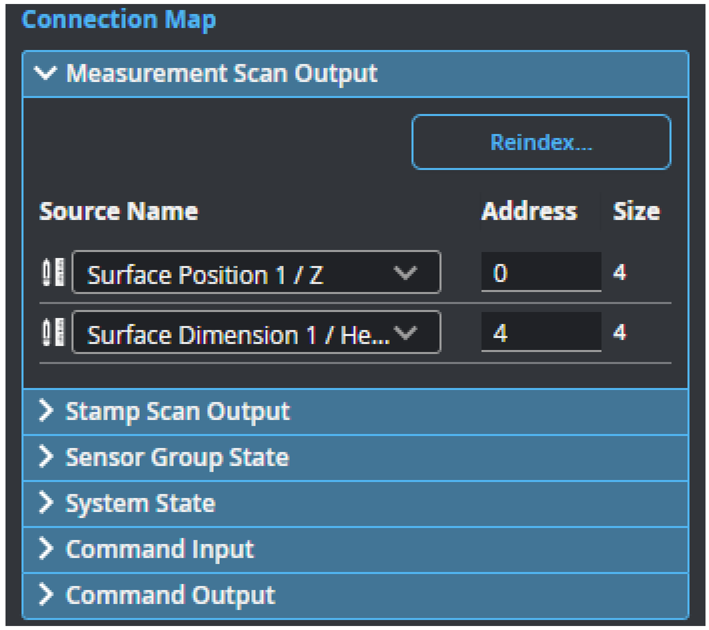


<u>**Insert Connections**</u>

Set **Address** to the address of a previously added source and click **Insert Connection** to insert the selected source into the connection map at that address and shift others down. Inserting a source at an address not currently used adds it at that address.

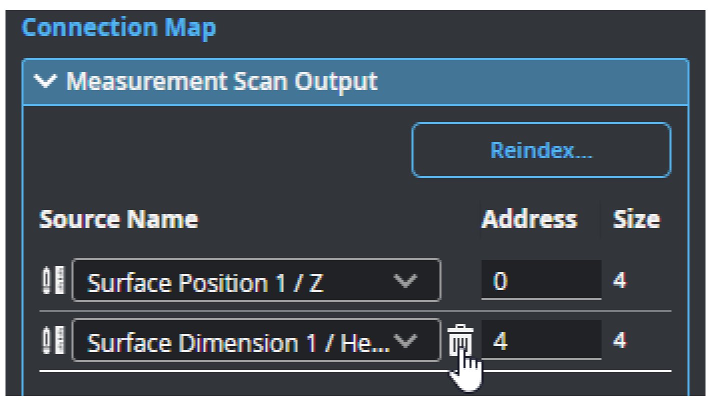
<u>**Delete Connections**</u>

Removes all user-added connections.
</td></tr>
<tr><td>4</td><td>Connection map</td><td>Lists the added connections. In the following, two measurements have been added.</td></tr>
</table>

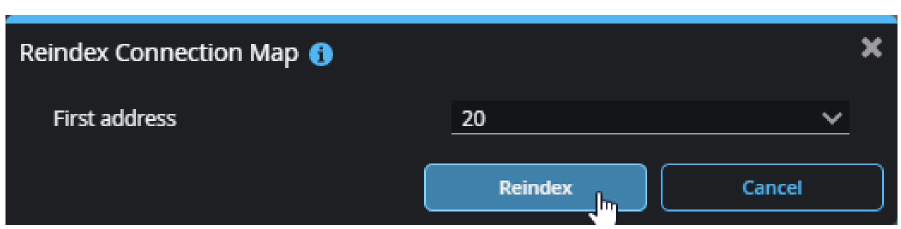
<table>
<tr><th>Element</th><th>Description</th></tr>
<tr><td></td><td>


You can change the address of a source by editing it and pressing Tab or clicking elsewhere in the interface. The sources are resorted after you change an address.

To remove a source from the connections map, hover over the source and click the source's trashcan icon.



Use **Reindex...** to remove gaps of unused addresses between connections. You can also use **Reindex...** to set the value of the first index.

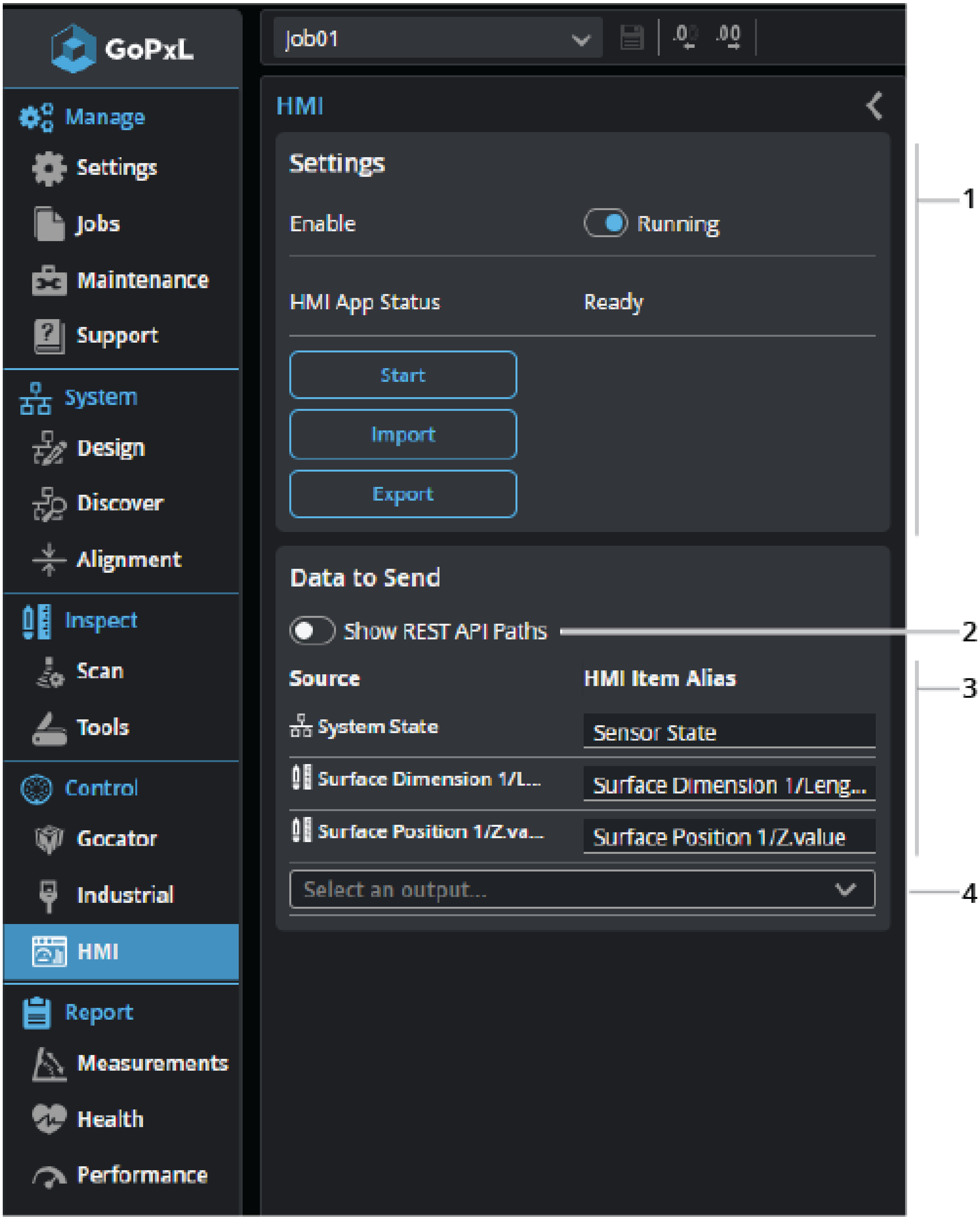


</td></tr>
</table>

HMI

The HMI panel lets you enable GoHMI, choose which outputs you want to send to an HMI application, and start communication with the HMI.

You configure the HMI on the **Control** > **HMI** page. For details, see *GoHMI and GoHMI Designer* on page 844.



When you start HMI on a PC instance of GoPxL, a helper application (WebIQ Server) launches.

<table>
<tr><th></th><th>Element</th><th>Description</th></tr>
<tr><td>1</td><td>HMI</td><td>Lets you enable the HMI service. After you have enabled the service, the following functions are available:

Start / Stop

Starts / stops the HMI application. After starting the application, a **View live** button is available, which lets you open the HMI in a browser tab.

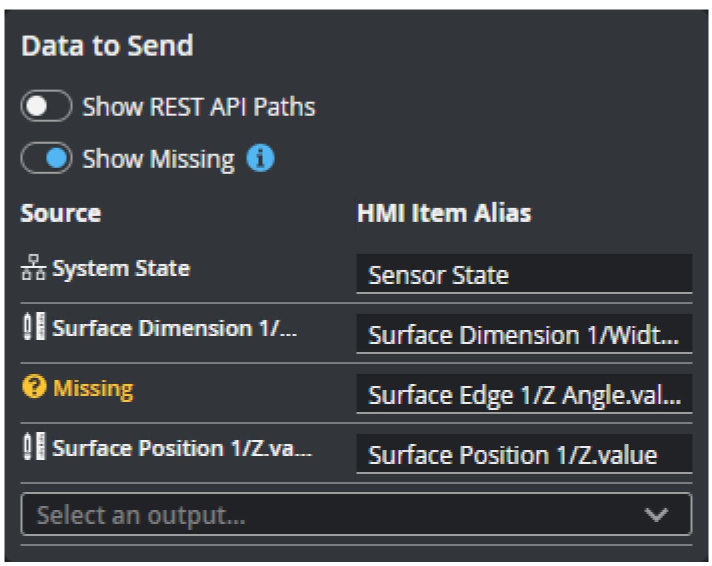
View live

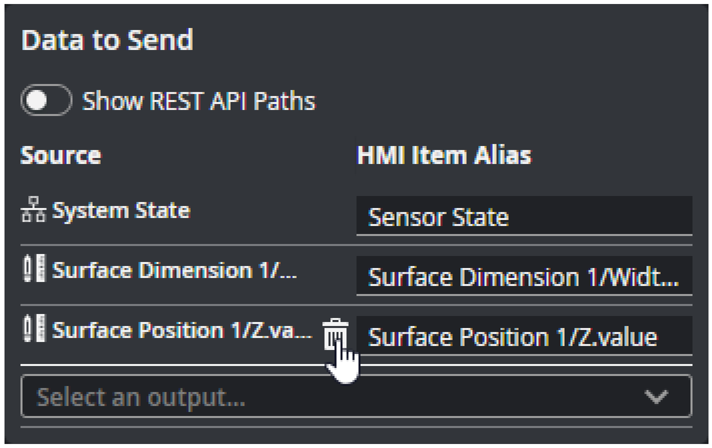
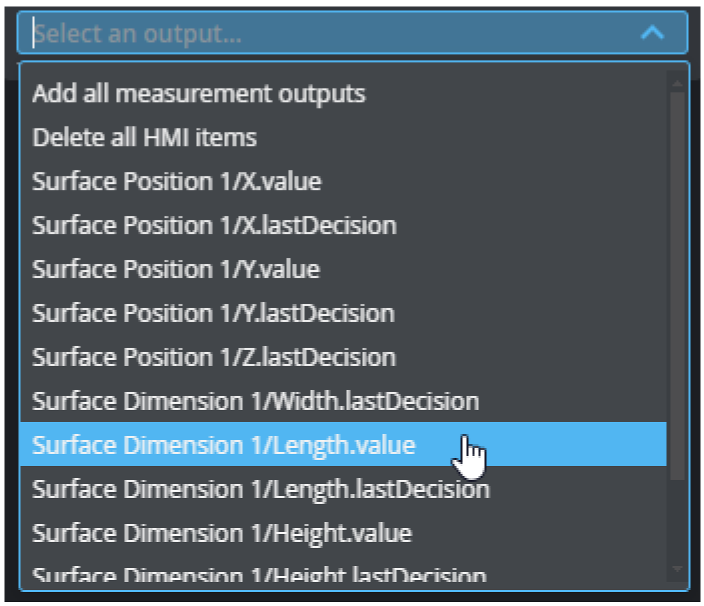
Opens the HMI in a browser tab.

Import / Export

Lets you import and export HMIs so that you can transfer them to a different platform, for example, going from a PC to a GoMax device.</td></tr>
<tr><td>2</td><td>Show REST API Paths</td><td>Toggles between showing the full internal path to an output and an alias, which is simplified. In GoHMI Designer, you will see the REST API path; toggling this on can help make it easier to find which item is which.</td></tr>
<tr><td>3</td><td>Show Missing</td><td>If the source for a connection is missing (that is, the tool providing the source has been removed in the job, or an output has been disabled), this option is displayed. Enabling it lists connections whose source can't be found.

You can delete the missing connection (by hovering over the row in **Data to Send** and clicking the trashcan icon) or enable the source if it has just been disabled.


</td></tr>
<tr><td>3</td><td>Tool & Sensor Outputs</td><td>The list of outputs (scan data and tool outputs) that GoPxL will send to an HMI. You can remove an output from the list by hovering over it and clicking the trashcan icon.</td></tr>
</table>

	Element	Description
		
4	Output selector	Lists the available outputs. Click an output in the drop-down to add it to the outputs GoPxL will send to an HMI. 

Reporting

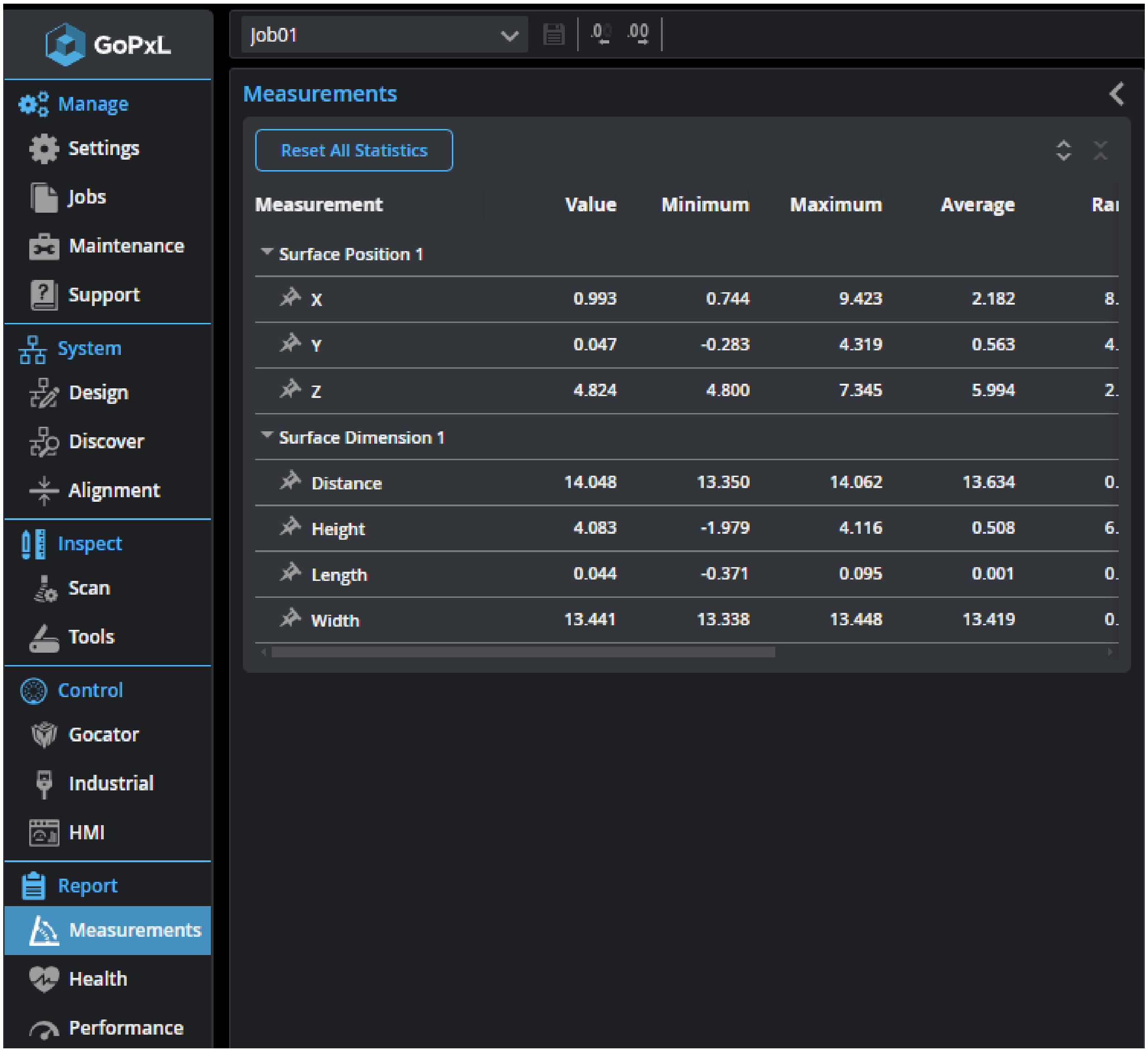
The **Report** category summarizes sensor and system health information and provides tool and measurement statistics. It also provides tool performance statistics.

Available report pages

Report	Description
Measurements	The **Measurements** page provides statistics for each enabled measurement. For more information, see *Measurements* below.
Health	The **Health** page lets you monitor system and sensor health indicators, as well as various counters, such as processing drops. For more information, see *Health* on page 823.
Performance	The **Performance** page provides performance statistics. For more information, see *Performance* on page 825.

Measurements

The **Report** > **Measurements** page displays statistics for each measurement enabled in the **Inspect** > **Tools** page, grouped by the tool that contains the measurement. Values are reset when the sensor is started. To reset statistics manually, click the **Reset All Statistics** button.



For each measurement, the following information is available:

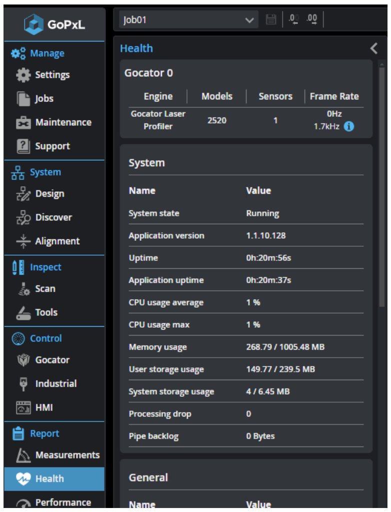
Measurement statistics

Name	Description
Value	The most recent measurement value.
Minimum	The minimum measurement value that has been observed.
Maximum	The maximum measurement value that has been observed.
Average	The average of all measurement values collected since the sensor was started.
Range	The difference between Max and Min.
Std Dev	The standard deviation of all measurement values collected since the sensor was started.

Name	Description
Pass Count	The number of pass decisions the measurement has generated.
Fail Count	The number of fail decisions the measurement has generated.
Invalid Count	The number of frames that returned no valid measurement value.

Health

In the **Health** panel, you can monitor system, sensor group, and sensor health indicators, as well as various counters, such as processing drops.



The following tables list the information available in the Health panel.

> When a sensor is being accelerated (by a PC instance of GoPxL or by GoMax NX), the value reported in the System section comes from the accelerator. However, only the following metrics are appreciably different when accelerated: Uptime, CPU Usage, Memory Usage, and System/User Storage Usage.

System health

Name	Description
System state	Current system state.
Application version	System software version.
Uptime	Time since the system was power-cycled or reset. (If the sensor is accelerated by a PC, the PC's uptime.)
Application uptime	Time since GoPxL started. (On a PC, indicates how long the PC has been running.)
CPU usage average	Average system CPU usage for the system.
CPU usage max	Maximum system CPU usage for the system.
Memory usage	Sensor memory (MB used / MB total available). When a sensor is accelerated, this displays N/A.
User storage usage	Sensor flash storage (MB used / MB total available). When a sensor is accelerated,

Name	Description
	this displays N/A.
System storage usage	Non-volatile storage available (MB used / MB total available).
Processing drop	The sum of various indicators related to processing drops including drops due to insufficient CPU and buffer overflows.
Pipe backlog	Number of bytes queued for processing.

General health

Name	Description
Encoder value	Current encoder value (ticks).
Encoder frequency	Current encoder frequency (Hz).
Processing latency	Last delay from camera exposure start to when the results are ready for output.
Processing latency (Max)	Peak delay from camera exposure start to when the results are ready for output.
Scan count	Number of scans performed since sensor state last changed to Running.
Scanner drops	Number of sensor group (scanner) drops.
Current speed	Current speed of the sensor.
Trigger drops	Count of camera frames dropped due to excessive trigger speed.

For each sensor in a system, GoPxL displays a table listing the health indicators for that sensor.

Sensor {n} health

Name	Description
Master connected	Whether the sensor is connected to a Master sensor networking hub.
Laser safety	Whether Laser Safety is enabled. With laser-based sensors, laser safety must be enabled in order to scan.
Sensor uptime	Time since the sensor was power-cycled or reset.
CPU usage average	Average system CPU usage for the sensor.
CPU usage max	Maximum system CPU usage for the sensor.
Ethernet receive rate	Rate at which data is being received by the sensor over Ethernet (Mbps).
Ethernet send rate	Rate at which data is being sent to the sensor over Ethernet (Mbps).
Ethernet link speed	Speed of the Ethernet link.
Memory usage	Sensor memory usage (MB used / MB total available).
User storage usage	Sensor flash storage usage (MB used / MB total available). When
System storage usage	Non-volatile storage available (MB used / MB total available).
Max spot count	Maximum available spots, across all cameras. (G2)
Total exposure time	Total exposure time of laser light (G2), projector (G3), or light source (G5)
Internal temperature	Internal sensor temperature (degrees Celsius).

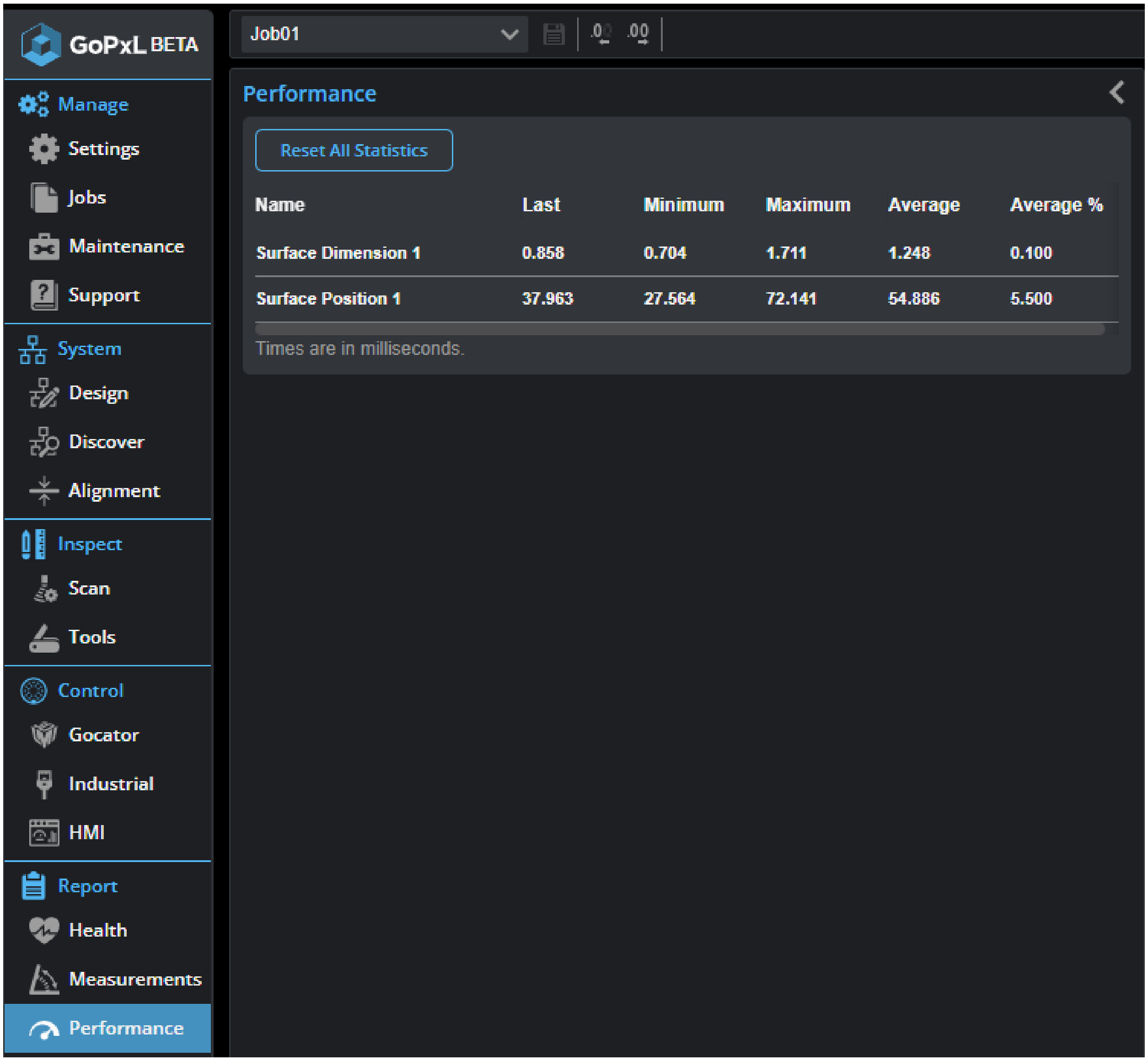
Name	Description
Laser driver temperature	Laser driver control temperature (degrees Celcius). (G2)
Light driver temperature	Light driver control temperature (degrees Celcius). (G5)
LED PCB temperature	LED PCB temperature (degrees Celcius). (G5)
External thermistor temp.	External thermistor temperature (degrees Celcius). (G5)
Projector illumination temp	Projector temperature (G3).
Trigger drops	Count of camera frames dropped due to excessive trigger speed.
Camera buffer drops	Count of camera buffer drops due to excessive trigger speed. For G3, the buffer drops are shown for each camera.

Control

Name	Description
Gocator control drop	The number of Gocator protocol drops.

Performance

The **Report** > **Performance** page displays performance statistics (execution time in milliseconds) for each tool added in the **Inspect** page.



For each tool, the following information is available:

Performance Statistics

Name	Description
Last	The latest execution time of the tool.
Minimum	The minimum execution time of the tool.
Maximum	The maximum execution time of the tool.
Average	The average execution time of the tool.
Average %	The average percentage of the execution time of the tool.

Accelerating Sensors

For performance-critical applications, you can run a sensor through a PC instance of GoPxL or through a GoMax NX device. For more information on running GoPxL through a PC, see *Running GoPxL on a Windows PC* on page 828.

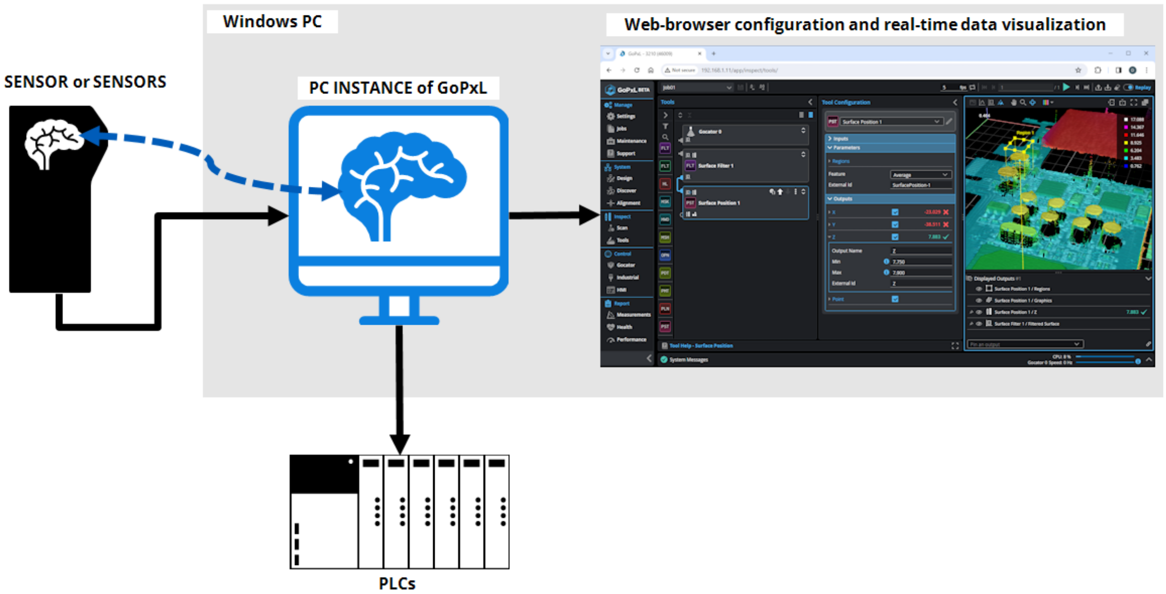
Running GoPxL on a Windows PC

GoPxL is the software you use to configure Gocator sensors. Typically, you connect directly to a sensor's IP address via a web browser, and use the GoPxL software running on the sensor to configure and run the sensor. (By default, a sensor's IP address is 192.168.1.10. If you can't find a sensor, see *GoPxL Discovery Tool* on page 935.)

You can also run GoPxL on a Windows PC. We call GoPxL running on a PC a "PC (or local) instance of GoPxL." If your Windows account has Administrator rights, you can install a PC instance of GoPxL as a windows "service." A PC instance installed (and running) as a service can run a sensor without the need for a user to be logged into Windows.

A PC instance of GoPxL lets you do the following:

- Accelerate sensors: For applications where performance is critical, you should run a sensor through a PC instance of GoPxL. After you connect a sensor to GoPxL running on a PC, the instance automatically starts accelerating it, handling the processing of the scan data, rather than the sensor. This improves system performance, potentially letting you increase scan rates or add more tools. For more information, see *Accelerating a Sensor* on page 835.



- Load previously recorded scan data to examine it or to configure measurement tools offline, without a sensor, using that data. For more information, see *Loading Scan Data* on page 839.

For information on the ports a PC instance uses (for example, in order to ensure ports are not blocked over your network), see *Required Ports* on page 58.

If your Windows account has Administrator rights, you can install a PC instance of GoPxL as a windows "service." A PC instance installed (and running) as a service can run a sensor without the need for a user to be logged into Windows. For more information, see *Instance Settings* on page 833

The user interface of GoPxL running on a sensor and GoPxL running on a PC is almost identical. When running a sensor through a PC instance of GoPxL, you can't upgrade or restart the sensor. You must remove the sensor from the PC instance and then connect to the sensor directly to upgrade the sensor.

🗋	Job files you create in a PC instance of GoPxL are separate from any jobs that are on the sensor. Be sure to back them up if ever you need to run them directly on the sensor.

System Requirements

The following are the system requirements for using GoPxL on a Windows PC:

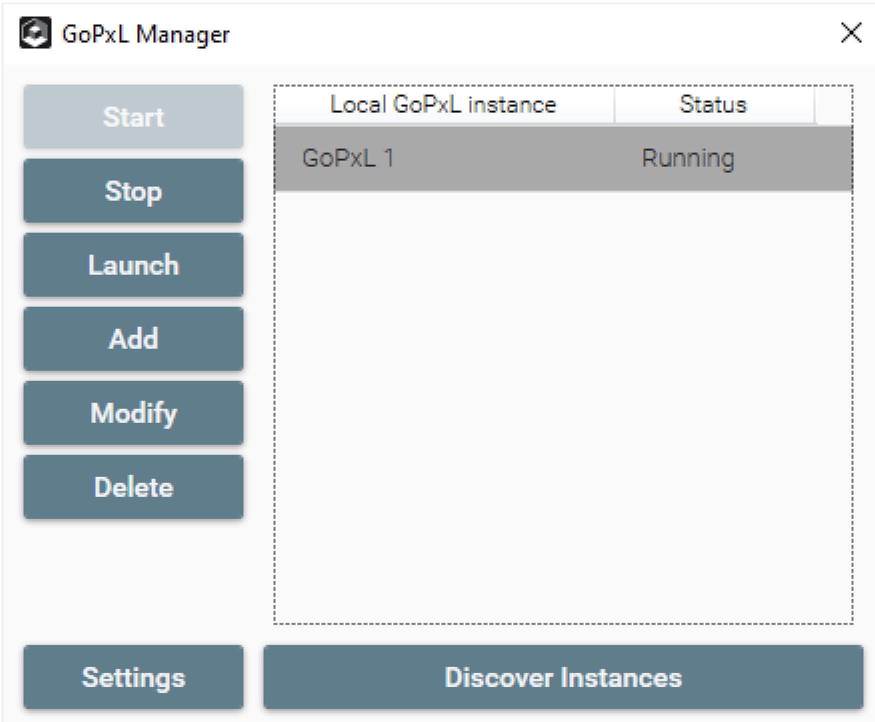
- Processor: x86-64 (Core i3 or equivalent)
- RAM: 8 GB
- Hard drive: 64 GB
- Operating system: Windows 10 (32-bit and other OS are not supported)
- Display: 1366 x 768

To accelerate more sensors or to run the system at higher speeds, use a computer with more system resources.

Note that acceleration of line profile sensors does not currently use a computer's graphics card.

Launching GoPxL on a Windows PC

The first time you launch GoPxL for Windows, a GoPxL instance is automatically created and started, and the GoPxL interface opens in your default browser. You use the GoPxL Manager application (which is started when you start GoPxL.exe) to add, remove, and configure instances. The following assumes you have connected a sensor to a network and that you can access its web interface.



The GoPxL Manager application. An instance is automatically added the first time you launch GoPxL.

To launch a PC instance of GoPxL

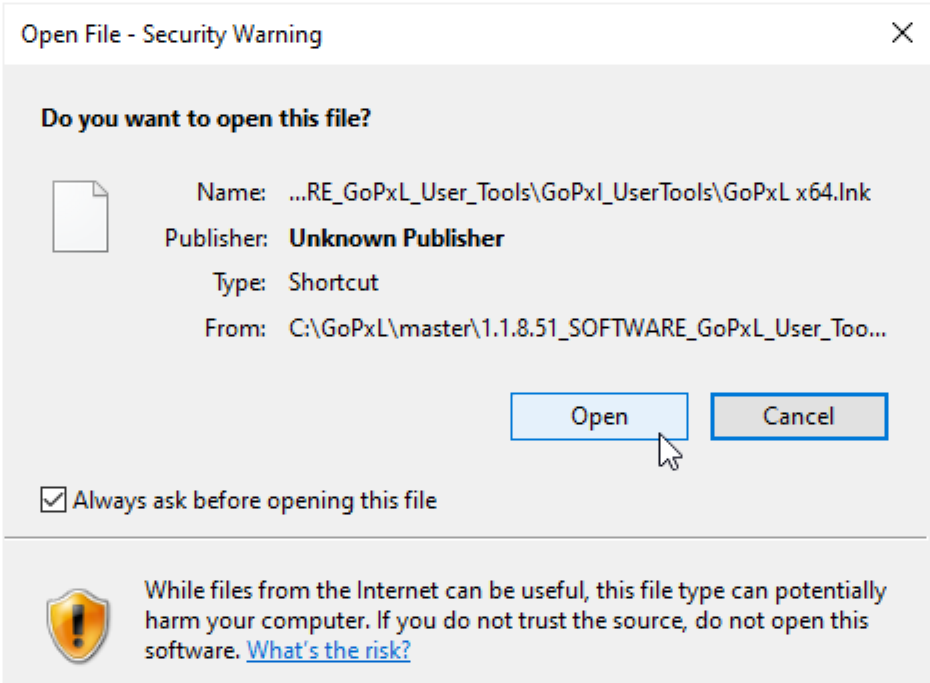
1. On the PC connected to the sensor, download and unzip the GoPxL Utilities package (`14631-x.x.x.x_SOFTWARE_GoPxL_Utilities.zip`).

   The version of the package must match the version of GoPxL on the sensor.

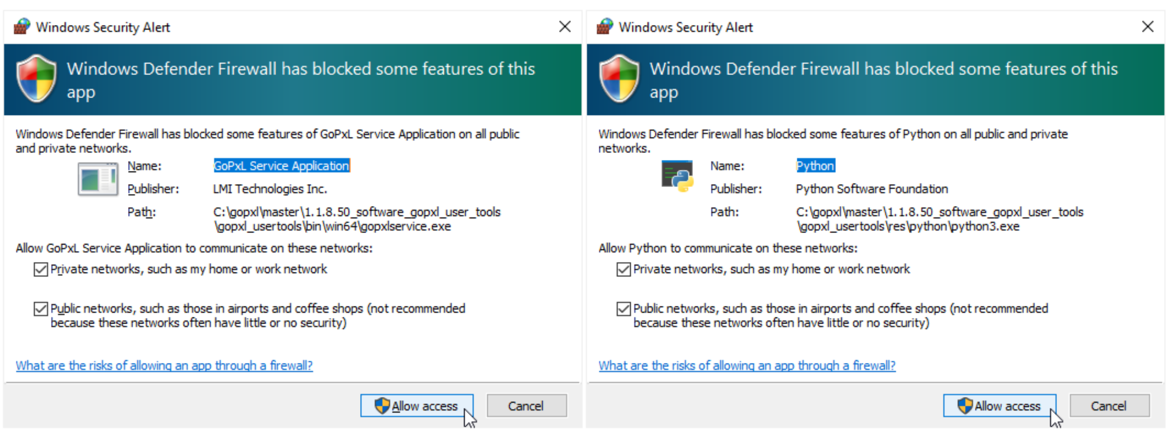
2. From the `GoPxL` folder, launch `GoPxL x64`.

   GoPxL Manager creates a local, PC-based instance of GoPxL and launches it in a new tab in your default browser, using IP address 127.0.0.1.

   Windows may display a security warning asking if you want to launch GoPxL. Click **Open** to continue.



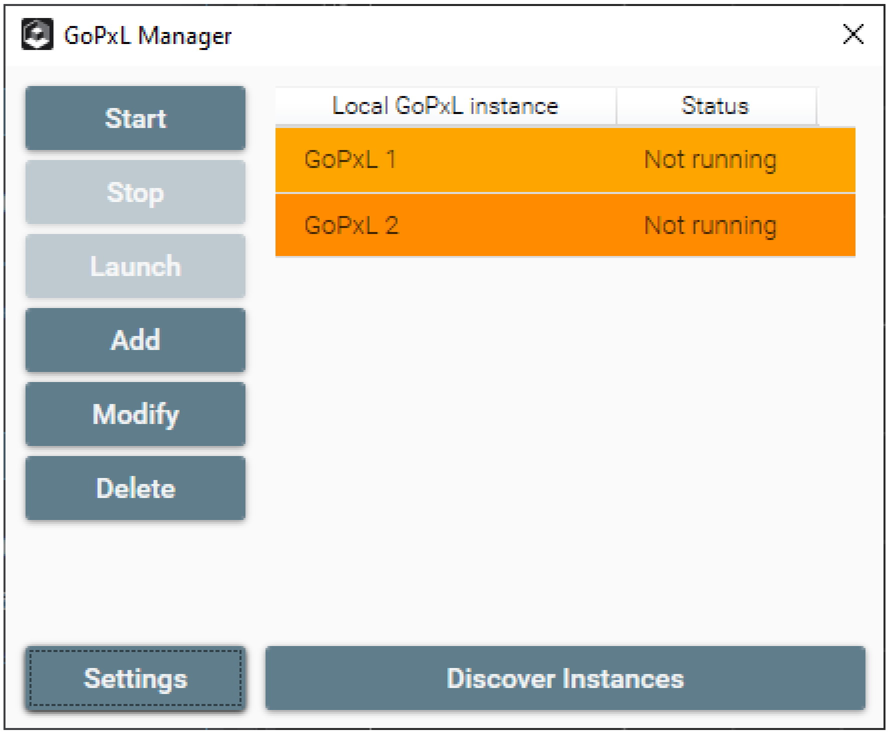
Additionally, your firewall may ask if you want to allow the GoPxL Service Application and Python to communicate over your network. To avoid issues, LMI recommends allowing communication over both Public and Private networks. Because the notifications appear one on top of the other, make sure to check the appropriate boxes and carefully click **Allow access** for both alerts.



At this point, you can do one of the following:

- Add one or more sensors to the PC instance to accelerate the sensors. For information on adding a sensor to a sensor group, see *Accelerating a Sensor* on page 835.
- Open scan data someone has previously recorded (in a .gprec, .gpbak, or .gpsup file) and examine the data or use it to configure measurement tools without taking your sensor system offline. For more information, see *Loading Scan Data* on page 839. (For information on the file formats used with GoPxL, see *File Formats* on page 842.)

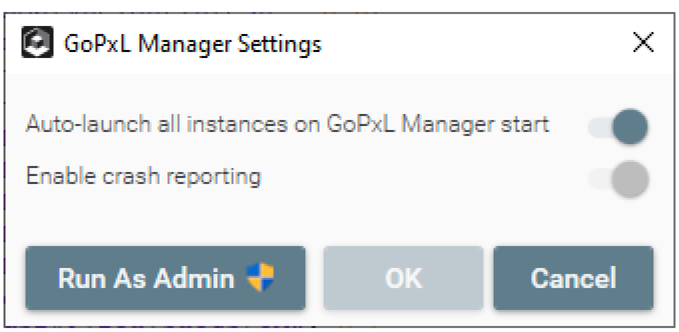
All ports used by PC instances must be unique. In some situations, it is possible to accidentally set the ports of two or more instances to the same port. This causes a conflict between those instances, and is indicated by highlighting in GoPxL Manager.



To correct this, select one of the instances, click **Modify**, and change the duplicated port to a different one. You can also delete any instances that use duplicate ports.

Application Settings

The following application settings are available:



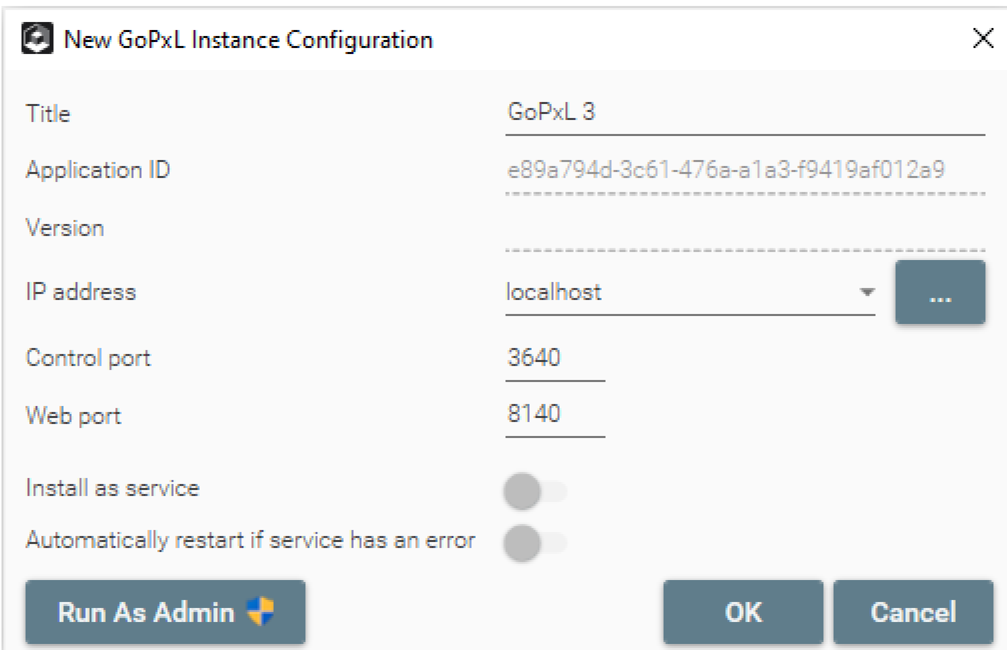
Parameters

Parameter	Description
Auto-launch all instances on GoPxL Manager start	When enabled, all instances not installed as a service will automatically start when GoPxL Manager starts. (Instances installed as a service will start when the computer is started.)

Parameter	Description
Enable crash reporting	If a PC instance of GoPxL crashes, a report will be generated. This can be useful if you need to troubleshoot problems with LMI support. LMI recommends leaving crash reporting enabled. To enable or disable this setting, you must be running GoPxL Manager as an administrator. To do this, click **Run As Admin**. The application will restart and let you make the change. After making the change, you should close GoPxL Manager and start it again. If you choose to disable crash reporting, an icon on the **Settings** button will indicate that

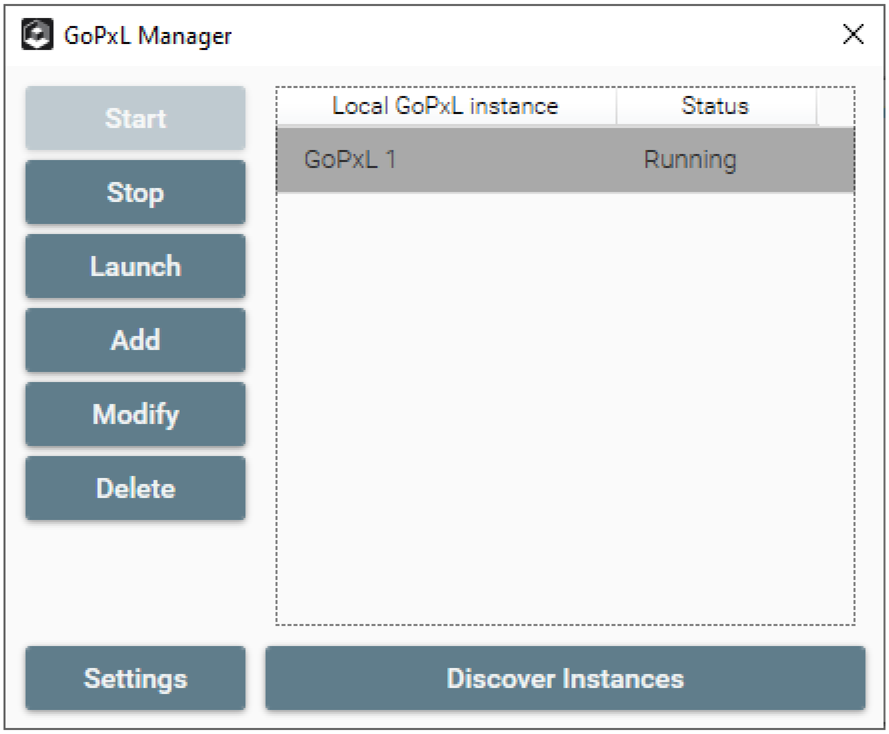
Instance Settings

When adding or modifying an instance, you can change various settings (see *Instance Configuration* on the next page).



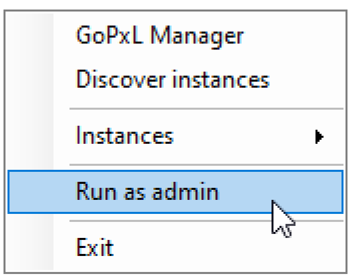
To add an instance, click **Add** and configure it if needed, and then click OK.

To modify an instance, select an instance in the list of instances and click **Modify**. To make changes to an instance, it must be stopped; select the instance and click **Stop** if needed.



GoPxL Manager with one instance running.

To change the settings of a service instance, and to delete, start, or stop a service instance, you must be running GoPxL Manager as an administrator. You can do this by clicking **Run As Admin** in the Instance Configuration dialog (see above), which restarts GoPxL Manager with administrator rights. You can also run right-click its icon in the Windows taskbar and click **Run as admin**.



Instance Configuration

Parameter	Description
Title	The name of the instance in GoPxL Manager.
Application ID	The ID of the GoPxL instance. Not editable.
Version	The version of the GoPxL instance. Not editable.
IP address	The IP address of the instance. You can select an available address on your PC from the dropdown, or manually specify the address.
Control port Web port	The main ports the instance uses. These must be unique for each instance.

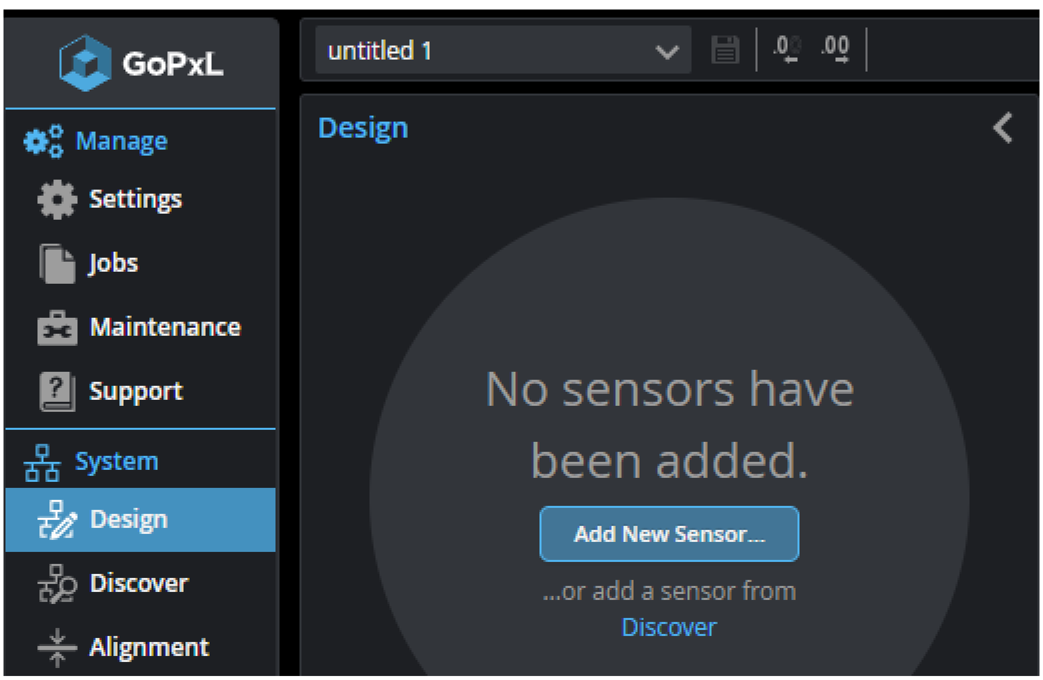
Parameter	Description
	Web port The port you use when connecting to the instance, for example, to `http://127.0.0.1:8120/`. **Control port** The port you use with the GoPxL SDK and REST API (see *GoPxL SDK and REST API* on page 871).
Install as a service	When this setting is enabled, the instance is installed as a service. A PC instance installed (and running) as a service can run a sensor without the need for a user to be logged into Windows.
Automatically restart if service has an error	If an instance of GoPxL is configured to be a service and it has an error, it will restart automatically.

Accelerating a Sensor

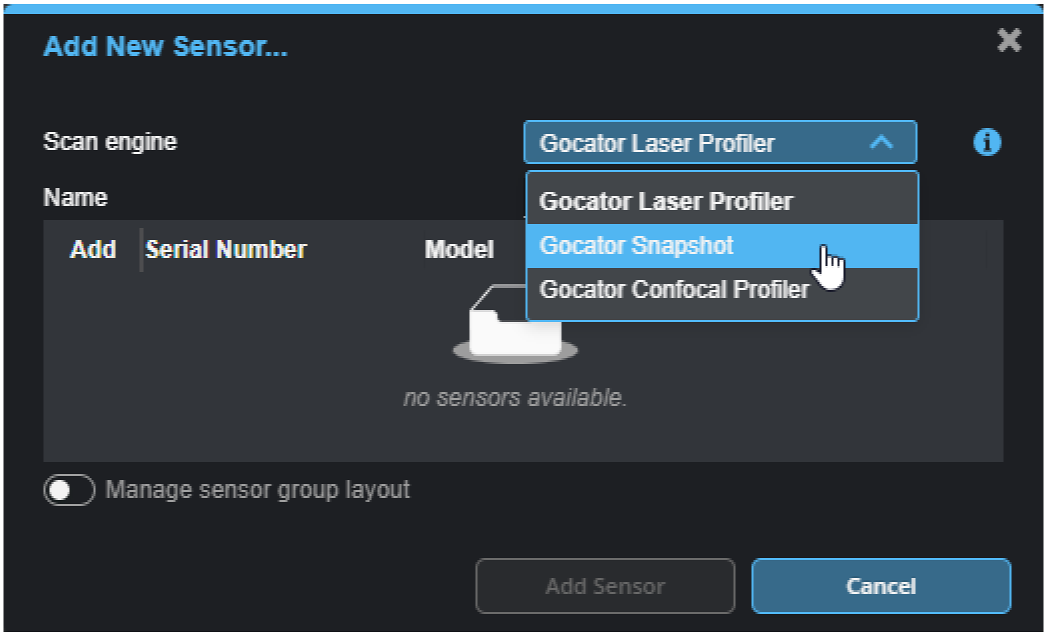
You accelerate a sensor (or group of sensors) by connecting it to a PC instance of GoPxL.

To accelerate a sensor

1. In the browser, go to the **System** > **Design** page and click **Add New Sensor...**

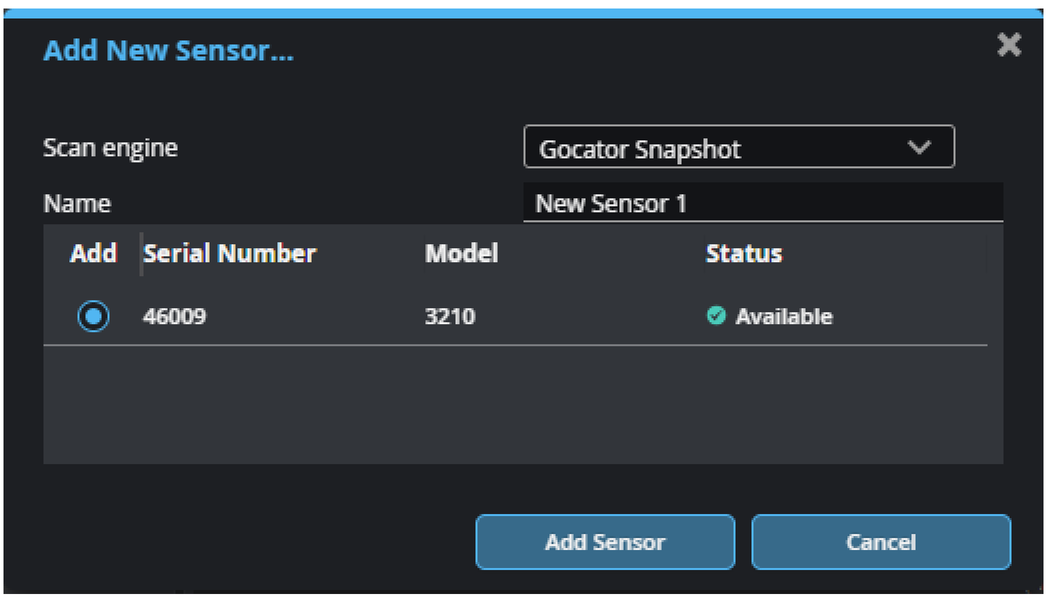


2. In **Add New Sensor**, choose the scan engine appropriate for your device

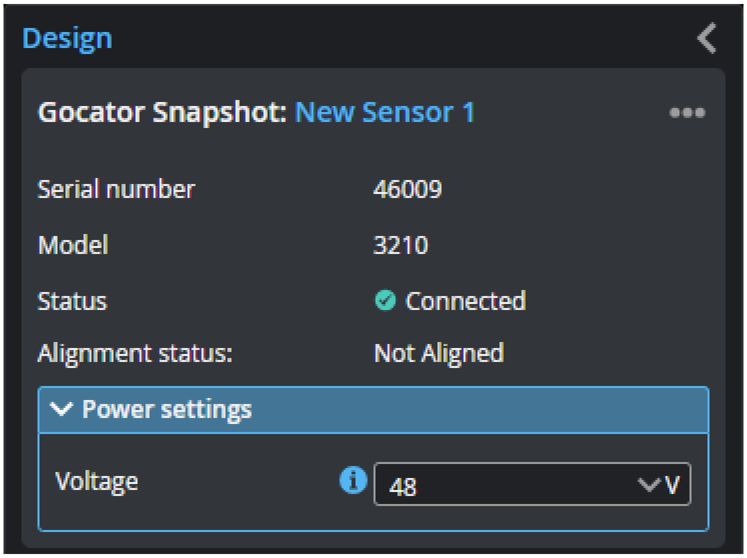


3. Click next to the device you want in the **Add** column, and click **Add Sensor**.

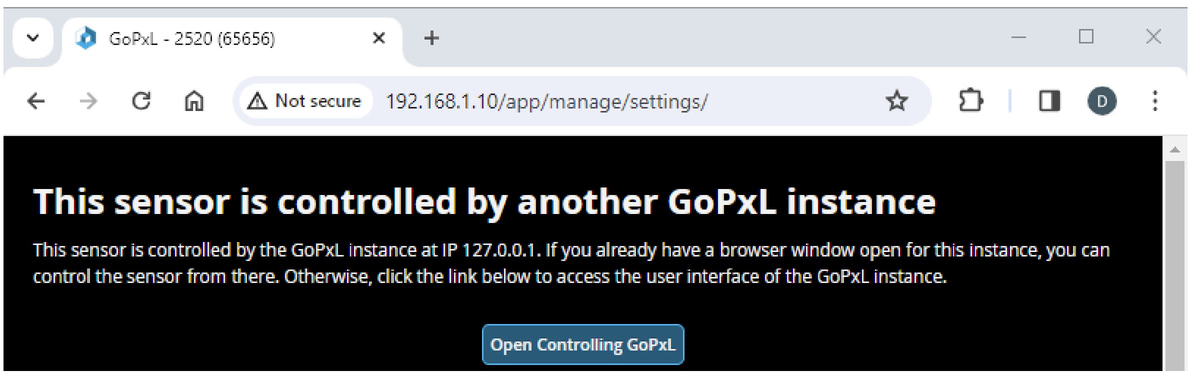
   You can optionally set the name of the sensor.



The physical sensor is now connected to the PC instance of GoPxL and is automatically accelerated.



You access the sensor via the IP address of the PC instance, through which you control and configure it.

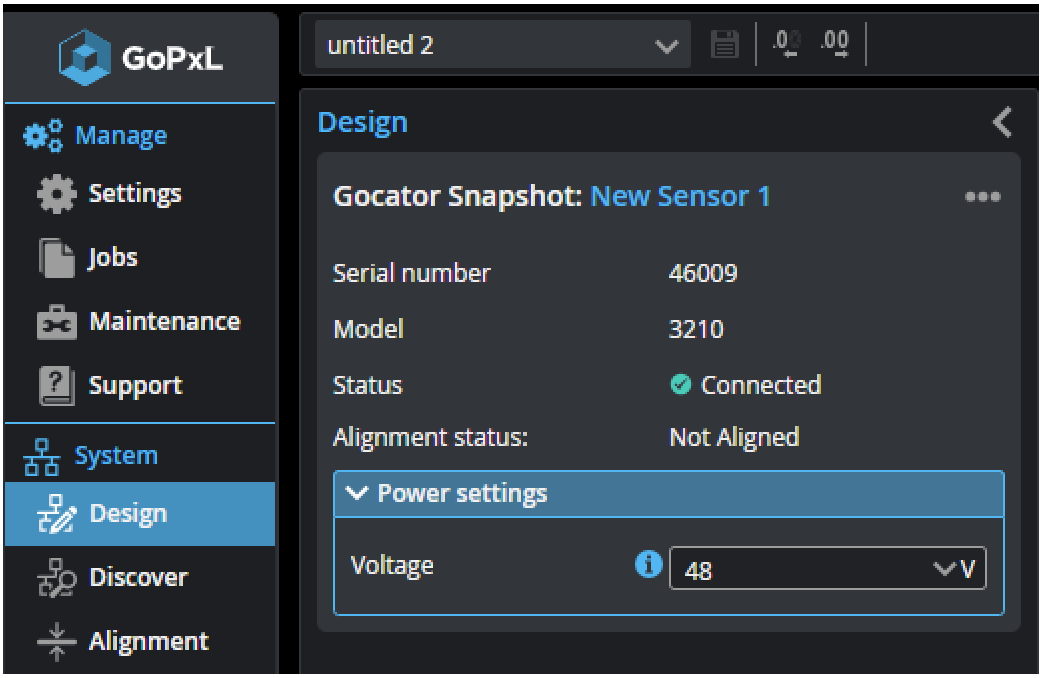


Stopping Acceleration

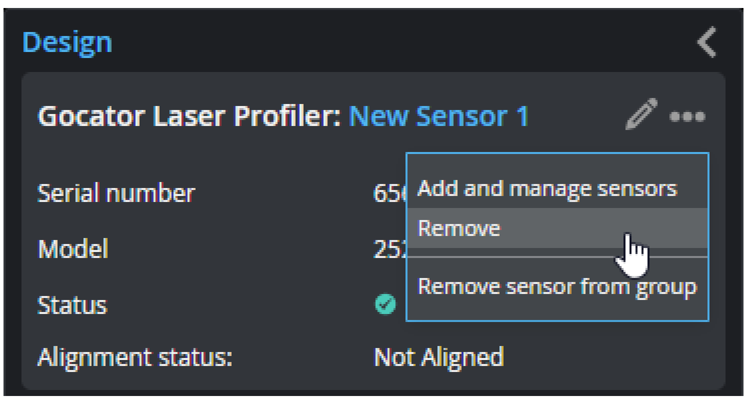
To stop accelerating a sensor, you remove it from the PC instance of GoPxL it's connected to. You also need to remove a sensor from a PC instance to upgrade the sensor's firmware.

To remove a sensor from a PC instance

1. In the GoPxL interface of the PC instance, go to the **System** > **Design** page.



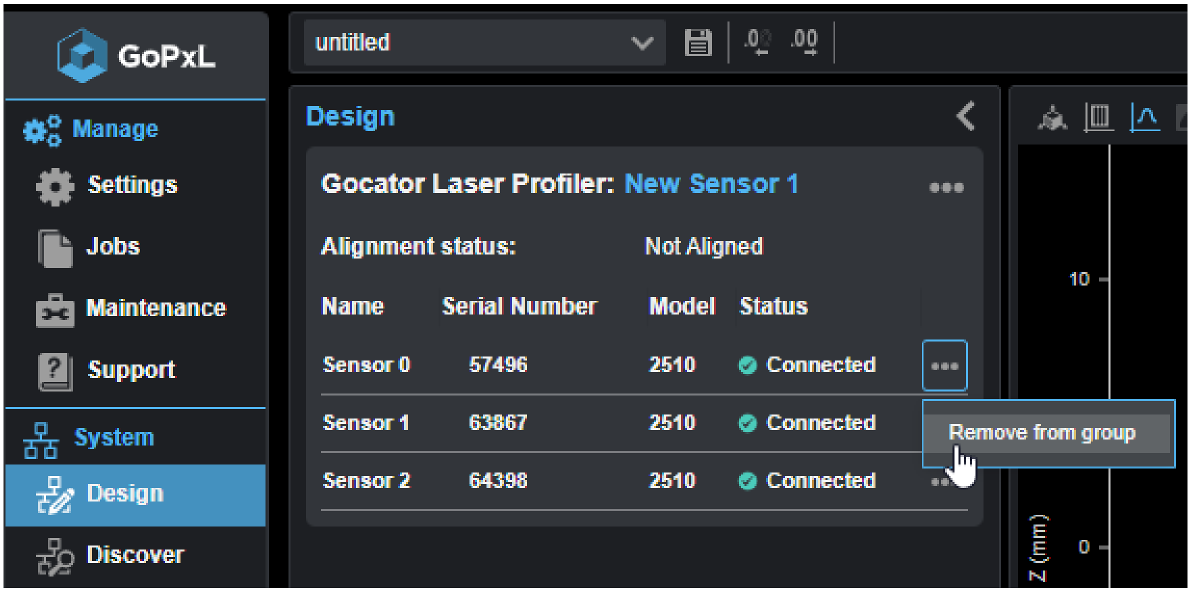
2. With single G2 sensors, click the three dots menu (•••) and click **Remove** or **Remove sensor from group**.



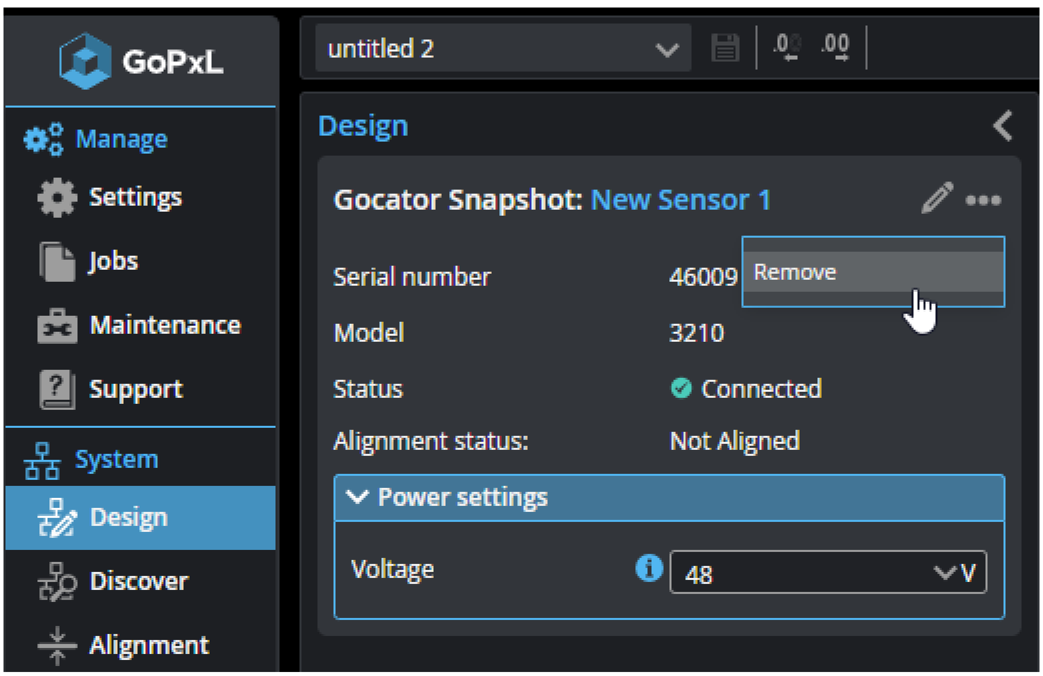
Remove: Choose this to remove the sensor and the sensor group containing it. If you have started configuring settings on the **Inspect** > **Scan** page, choose the next option instead.

Remove sensor from the group: Choose this if you have started configuring settings on the **Inspect** > **Scan** page and just need to add a different sensor to the sensor group.

3. With multi-sensor G2 systems, click the three dots menu (•••) in the row containing the sensor you want to remove, and click **Remove from group**.



4. With G3 sensors, click the three dots menu (•••), and choose **Remove**.



The sensor is removed from the PC instance.

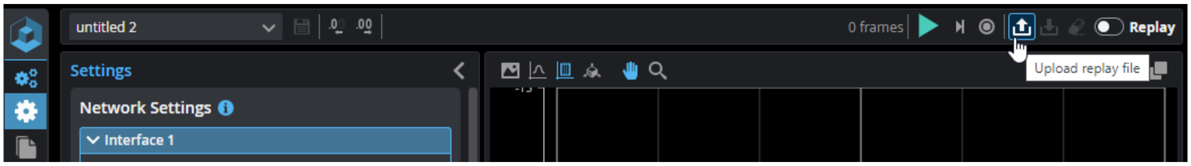
Loading Scan Data

To examine previously recorded scan data offline, you load a .gprec file into the PC instance of GoPxL.

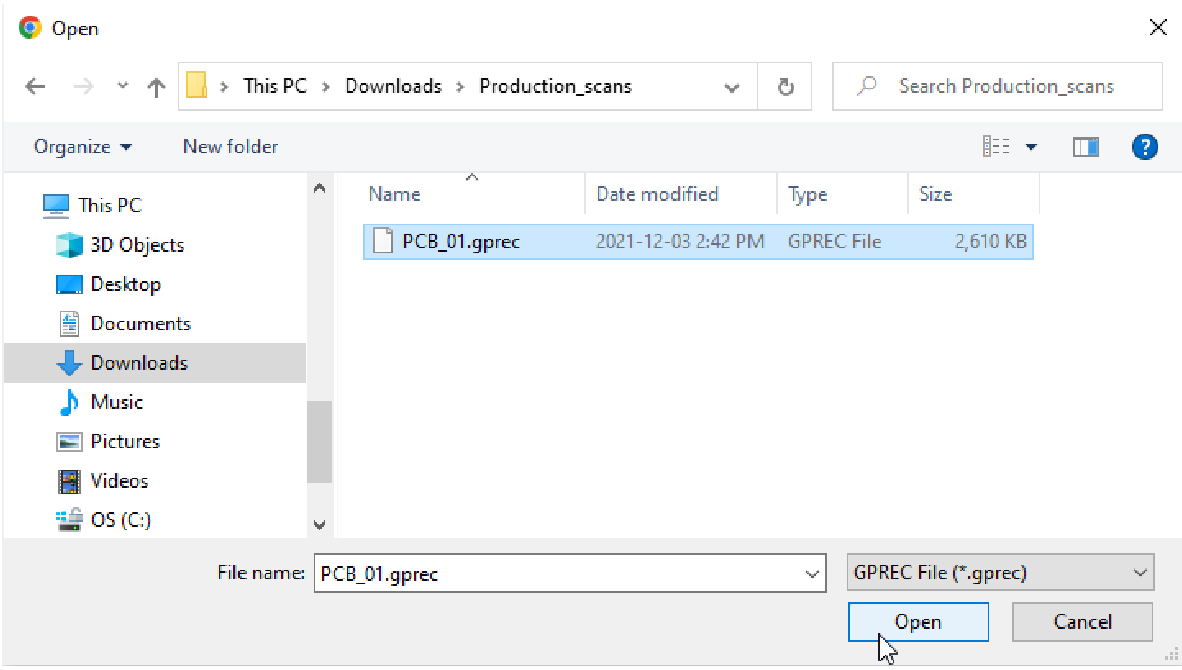
	You can also load a backup file or a support file to examine previously recorded scan data.

To load recorded data (.gprec)

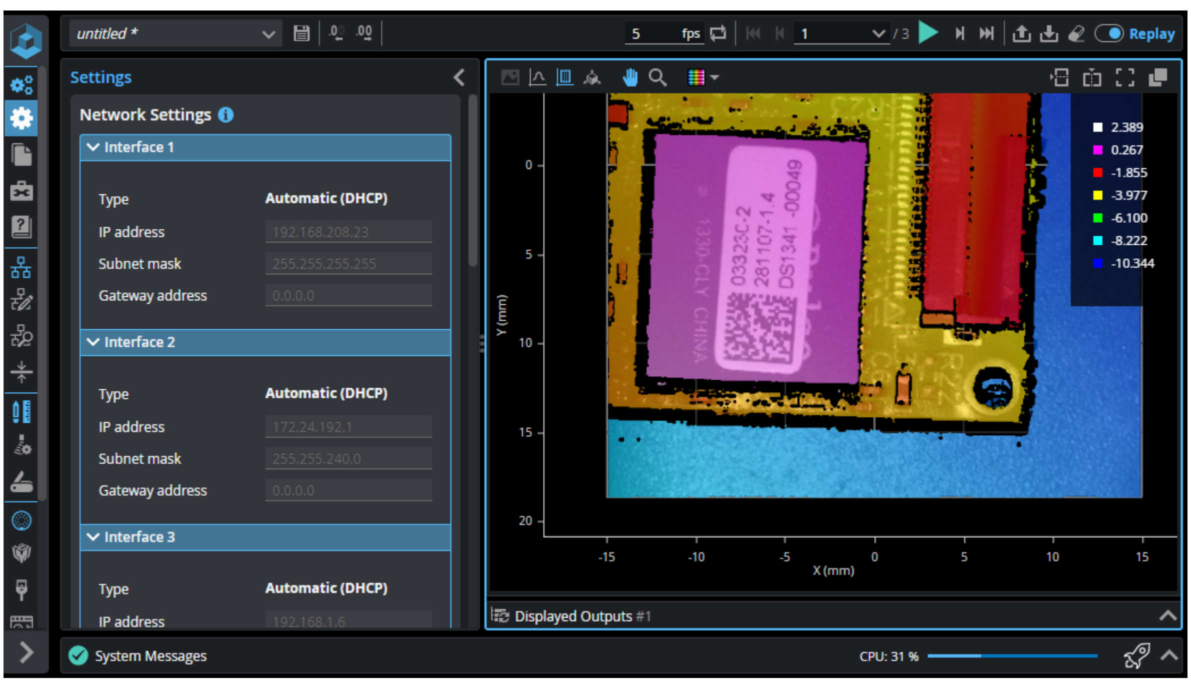
1. In the GoPxL interface, above the data viewer, click the Upload replay file button.



2. In the **Open** dialog, locate and select the recording (.gprec) you want to load, and click **Open**.



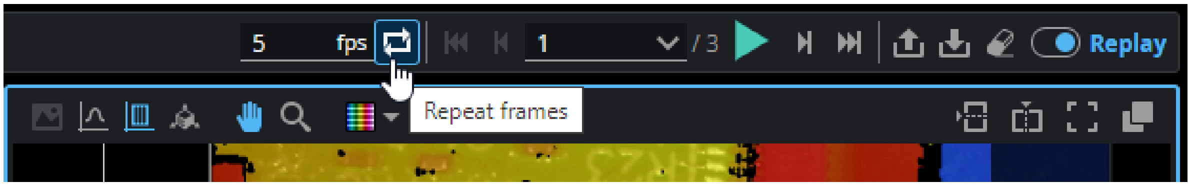
The recording loads in GoPxL, and GoPxL automatically switches to Replay mode.



You can now examine the recordings and configure tools using the data.

3. Use the replay controls above the data viewer to move through the frames, go to a specific frame, toggle and configure the speed of repeat frames mode (continuous loop), and so on.

   Hover over the replay controls for tooltips describing their function.

   Using repeat frames mode with a slow frame speed is useful for adjusting measurement tool settings and seeing the impact on scan data, without having to click Start or step through the frames of scan data manually. Repeat frames mode is also useful for demos.

File Formats

The following formats are used with GoPxL, or can be converted using the replay converter.

File formats

Format	GoPxL extension	Gocator firmware 6.x and earlier	Content	Description
Job	.gpjob	.job	• Scan settings • Tool settings • Control/output settings	Quickly change a set of settings by switching from one job to another. For more information, see *Jobs* on page 119.
Recording	.gprec	.rec	• Current job settings (see above) • Recorded data Note that sensor settings are not necessarily visible after loading a recording.	Quickly save and load recorded data for development and diagnostic purposes. For more information on recording and on loading recorded data, see *Working with Scan Data (Toolbar)* on page 112. You can use the replay converter, provided in the utilities package, to convert .gprec files to and from various formats. For more information, see *Replay Converter Tool* on page 938.
Backup	.gpbak	.bak	• All jobs • Layout and transformation • Current recorded data • Global settings	Allows restoring a sensor to a previous state in case undesired changes were made. You back up and restore sensors on the **Maintenance** page. For more information, see *Backup and Restore* on page 122.
Support	.gpsup	.gs	• Everything the Backup file contains • Additional diagnostic information	For supporting development and diagnosing issues. You download and upload support files on the **Support** page. For more information, see *Support* on page 124.

The following formats can be converted using the replay converter. For more information, see *Replay Converter Tool* on page 938.

File formats

Format	GoPxL extension	Gocator firmware 6.x and earlier	Content	Description
CSV	n/a	.csv	Scan data in CSV format	First download replay data (.gprec) in GoPxL, and then use the Replay Converter tool to convert to the CSV format, for use with third-party tools.
.pcd / .sur / .srf / .pro	n/a	n/a	Scan data in various formats	First download replay data in GoPxL, and then use the Replay Converter tool to convert to the desired format, for use with third-party tools.

You can also use the Data Export tool to export scan data in various formats. For more information, see *Data Export* on page 734.

GoHMI and GoHMI Designer

GoHMI lets you create flexible human-machine interfaces (HMIs), which lets you provide a custom GUI containing system and inspection metrics. Your HMI can also be touch-enabled. You create and edit HMIs in GoHMI Designer. You can design different layouts for different devices so that, for example, your HMI will adapt to the screen it is being displayed on (a PC monitor, a laptop, a tablet, or a phone), as well as its orientation on the screen.

Before using the designer, you *must* enable GoHMI on GoPxL; for more information, see *Enabling and Configuring GoHMI in GoPxL* on page 849.

🗋	GoHMI and GoHMI Designer only support a single HMI app, called "GoHMI."

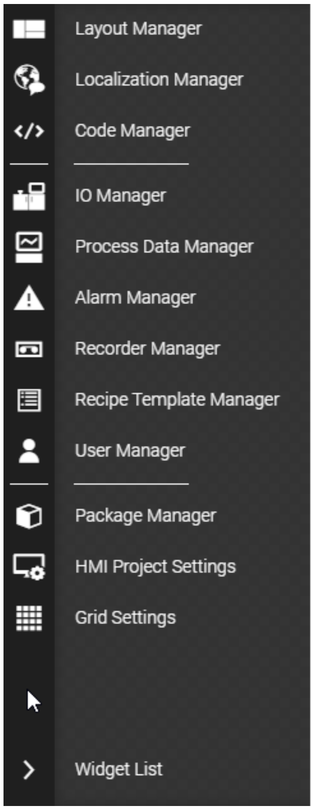
LMI provides a default HMI app with GoHMI that you use as a starting point or simply to explore the capabilities of GoHMI. When you launch GoHMI Designer, the default app loads in the designer.

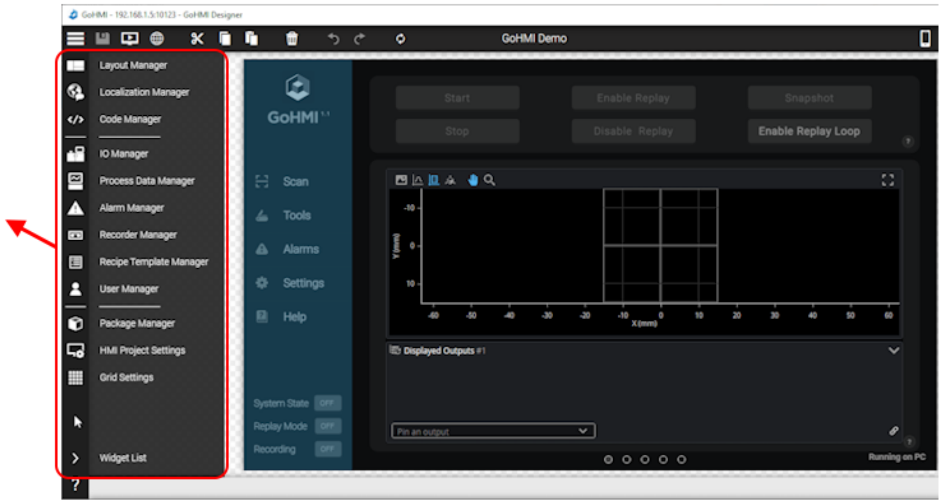
GoHMI Designer is available in the GoPxL Utilities package (`14631-x.x.x.x_SOFTWARE_GoPxL_Utilities.zip`), available on LMI's Product Downloads page (https://lmi3d.com/product-downloads/). After you have downloaded the package, unzip it and make note of the folder's location; the shortcut to the designer is in the `GoPxL` folder.

🗋	GoHMI Designer requires the Microsoft Visual C++ Redistributable package. This package is usually already installed on most PCs. If you are unable to launch HMI Designer, you may need to install this package, which can be downloaded at https://docs.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170.

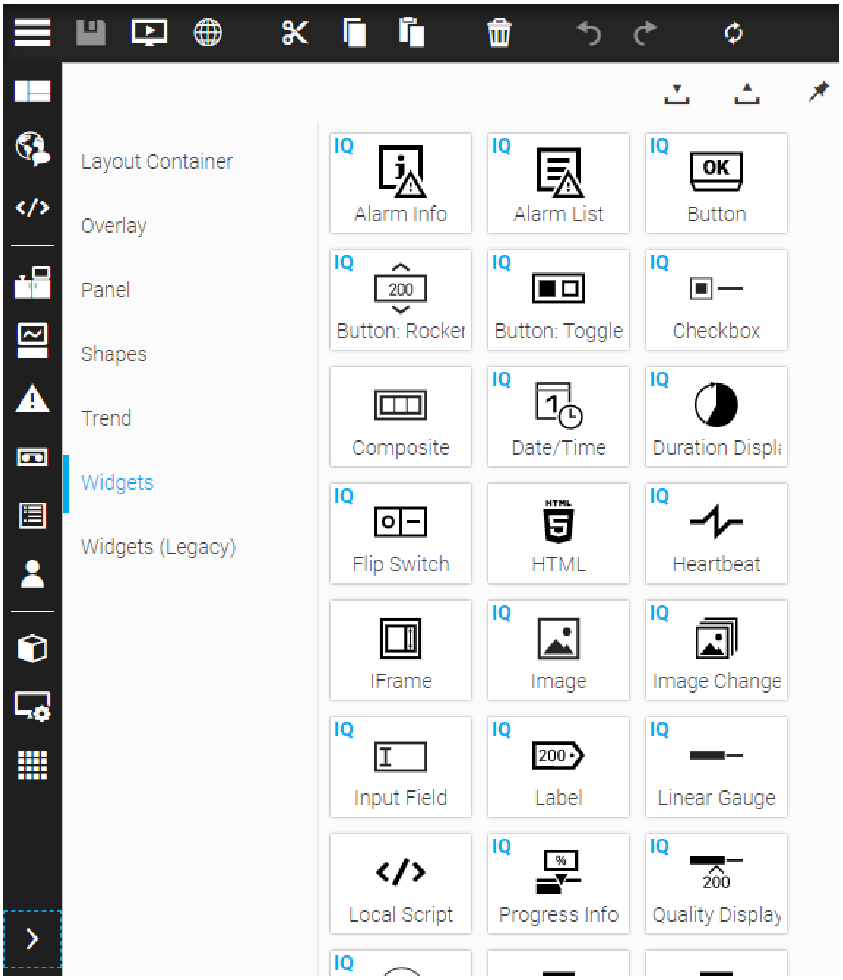
⚠	If you decide to run your HMI application on a sensor, instead of on a PC instance or a GoMax NX device, you *should not* use the HMI data trending functionality. Data trending writes data repeatedly to storage, which can shorten the lifespan of the sensor's flash memory. Also, if the sensor's storage becomes full, the sensor may not be able to start. If you need data trending, you should run the HMI on a PC or on GoMax.

GoHMI Designer organizes the available design functionalities and settings into "managers" and settings pages. After you have opened an HMI app in GoHMI Designer, they are listed in a collapsible panel along the left side of the application. Move your mouse over the left side of the designer to expand it. Click on a manager or settings item to switch to it.





The side panel also lets you access the list of widgets (user interface controls such as buttons) and layout containers you use to create your HMI. Click on "Widget List" to open the list. The widgets are grouped into categories. Below, the "Widgets" category is open.



GoHMI application layout development involves the concept of containers. Containers contain widgets or other containers (that is, containers can be nested). Containers can display their contents horizontally (for example, a row of buttons and fields) or vertically (for example, a stack of buttons).

By nesting containers and adjusting their alignment, you have complete control over your HMI app's layout and design.

The following provides a high level view of what you need to do to start creating an HMI. The first steps below refer to general acquisition and measurement tool setup, which is described elsewhere in the manual. The remaining steps are described in further details below.

1. Install a sensor or group of sensors.

   For more information, see the installation and networking topics under *Getting Started* on page 24.

2. Configure the sensor acquisition settings.

   For information on configuring acquisition settings, see *Scan - Configuring Acquisition* on page 195.

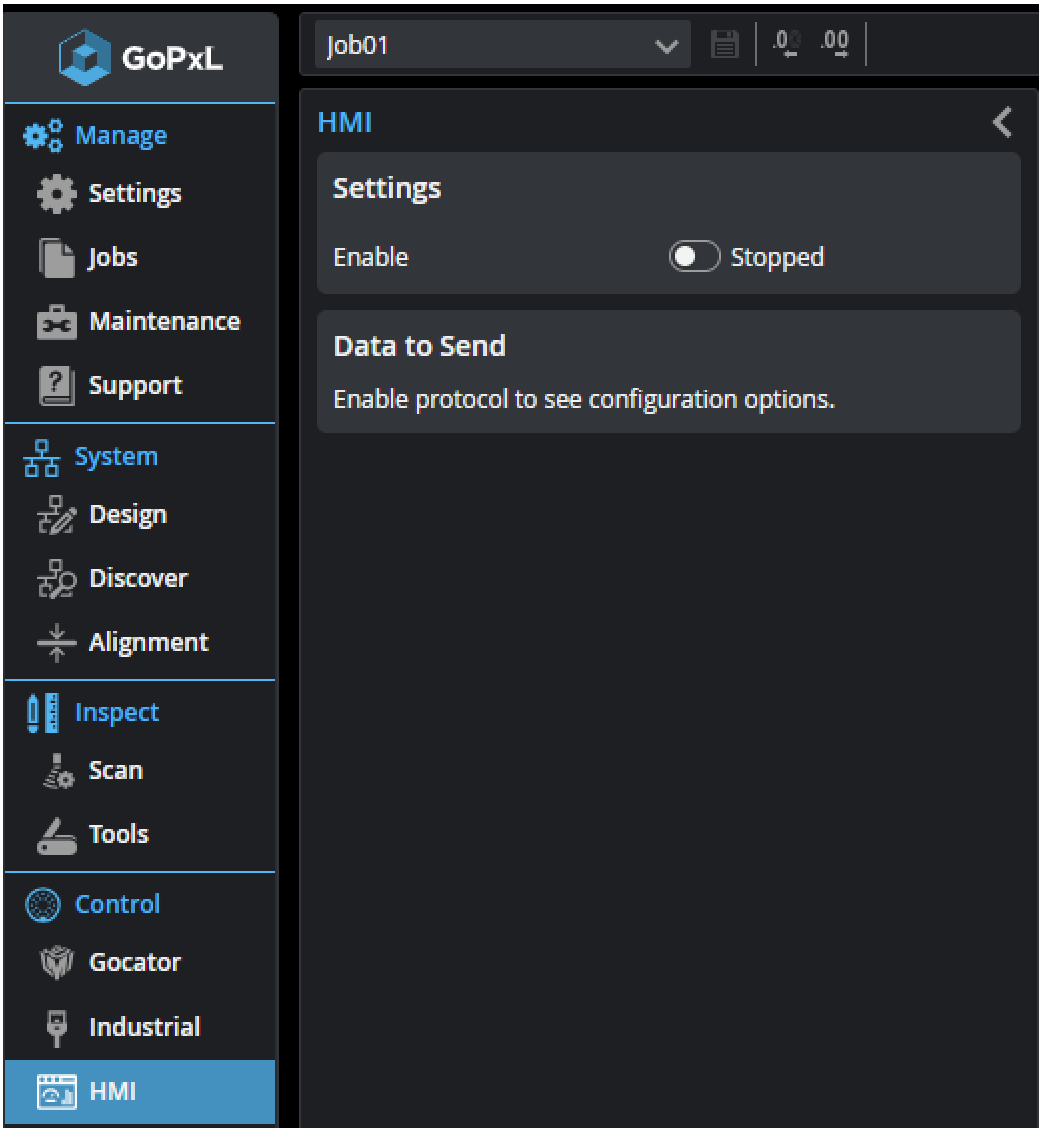
3. Add and configure measurement and processing tools.

   For more information, see *Tools - Measurement and Processing* on page 232 or the topics for specific tools.

4. (Optional) If your application requires the Gocator communication protocol or any of the industrial protocols, enable and configure these.

   For more information on the protocols available in GoPxL, see *Protocols (PLCs and other hardware)* on page 884.

5. Enable the HMI control settings in the **Control** > **HMI** page using the toggle and configure GoHMI.

   For more information, see *Enabling and Configuring GoHMI in GoPxL* on page 849.

6. Launch GoHMI Designer.

   For more information, see *Launching GoHMI Designer* on page 851.

The following lists the browser requirements of GoHMI apps:

- Chrome 80 (released 04.02.2020)
- Firefox 74 (released 10.03.2020)
- Edge 80 (released 07.02.2020)

- Safari 14 (released 16.09.2019)
- Opera 67 (released 25.02.2020)

Overview of Managers and Settings Pages

For information on the editors (called "managers") you use in GoHMI Designer, see the following table.

Overview of managers and settings pages

Task	Manager	Description
Creating HMIs with Multiple Views	• Layout Manager	Creating views in your HMI using panels in GoHMI Designer.
Laying Out Your HMI	• Layout Manager	Styling your HMI in GoHMI Designer.
Using Widgets	• Layout Manager	Using the widgets provided in GoHMI Designer.
Styling the HMI	• Layout Manager	Laying out views and widgets in GoHMI Designer.
Using UI Actions	• Layout Manager	Defining UI-actions (such as mouse clicks on buttons in the HMI) to implement simple behaviors in GoHMI Designer.
Localizing the HMI	• Localization Manager	Localizing your HMI in GoHMI Designer.
Using the Integrated Code Editor	• Code Manager	Using scripts to add capabilities and features to your web HMI or to modify the appearance of widgets.
Processing Data Interface and Data Items	• IO Manager • Process Data Manager	Configuring the data interface and connecting your HMI to a sensor system using GoHMI Designer.
Simulating Process Values	• Process Data Manager	Simulating process values in GoHMI Designer to evaluate or to demonstrate your web HMI without a connected sensor system.
Using Alarms and Alarm Lists	• Alarm Manager	Defining alarms and messages, store them as historical alarms, and display the corresponding messages in the HMI.
Using Recorder and Trend Display	• Recorder Manager	Recording process variables and displaying them in a trend diagram in the HMI. The integrated data recorder supports both live trends and trends from historical data.
Using Recipes and Recipe Widgets	• Recipe Template Manager	Generating recipe templates and useful metadata such as "article numbers," "customer description," and so on, which serve only for documentation. In applications, recipe process values can be applied or captured.
Managing Users and Access Groups	• User Manager	Creating users and user groups and how to assign access rights.
Managing Packages	• Package Manager	Extending GoHMI Designer with additional packages for widgets, themes, and starter apps.
Adjusting App Settings	• HMI Project Settings	Changing application settings such as project settings

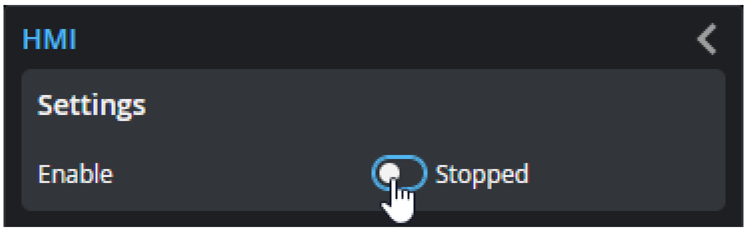
		in GoHMI Designer.
Adjusting Grid Settings	• Grid Settings	Adjusting the snap-to grid size in the designer.

Enabling and Configuring GoHMI in GoPxL

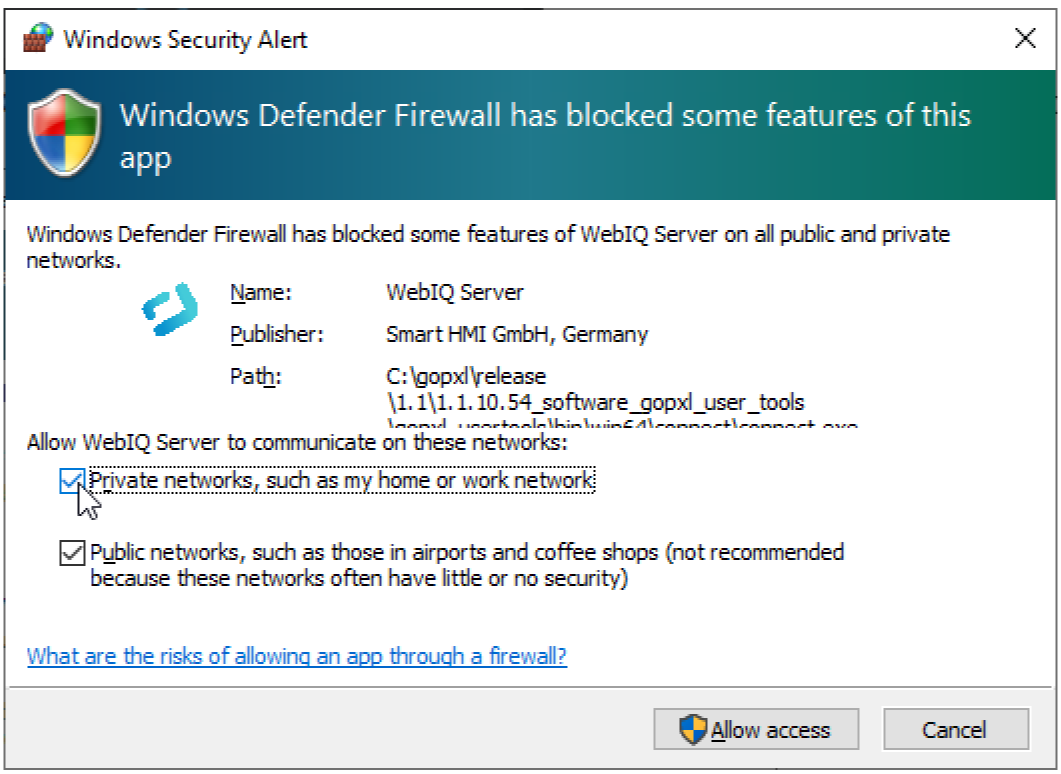
This section describes how to enable and configure GoHMI in the GoPxL interface. Before continuing, make sure you have created a sensor system, configured acquisition, and added and enabled the tools and measurements you want to display in the HMI.

To enable and configure GoHMI

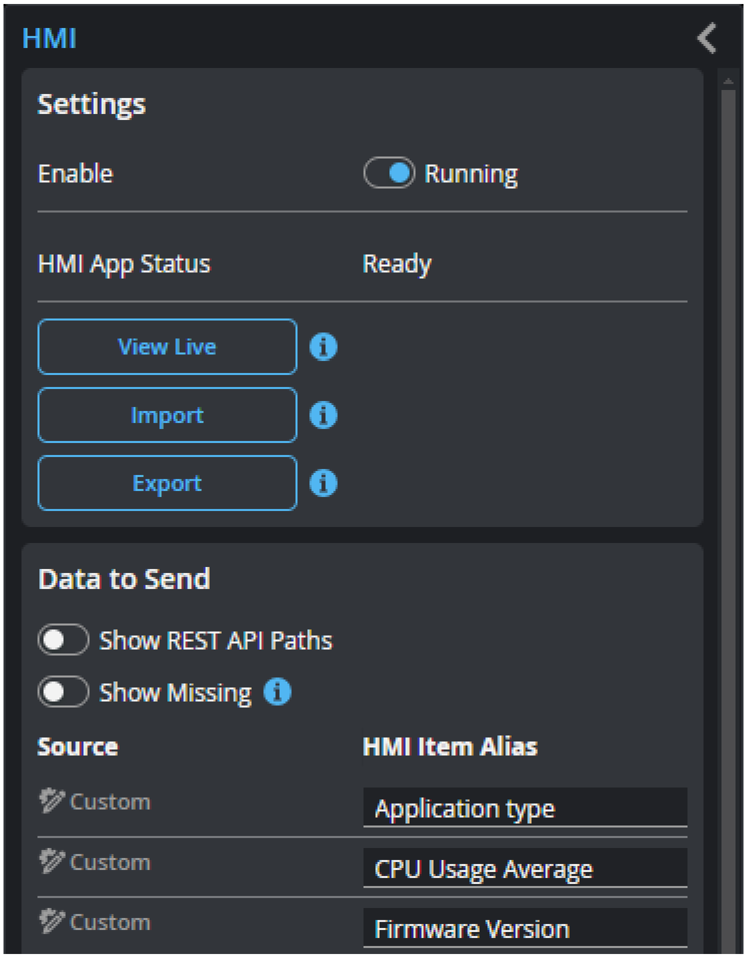
1. On the **Control** page, in the **HMI** panel, enable the HMI service.



If you are running GoPxL on a PC, a Windows Security Alert referring to WebIQ (the service on which GoHMI Designer runs) may appear. Enable both networks.



The HMI service launches, and the HMI panel displays the following:



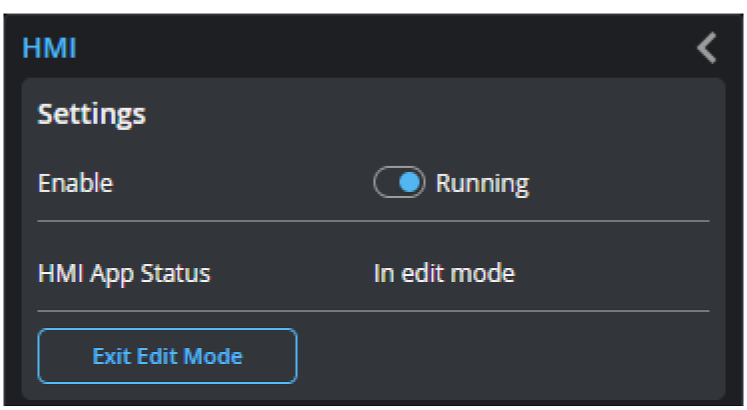
The **Settings** panel provides three buttons. The **Data to Send** panel below that lists the data items, including measurement values and decisions.

When the application is running, GoPxL displays a **View live** button that launches the application, in a new tab. You can make note of the URL of the HMI in this tab for future use.

Import lets you import an HMI application.

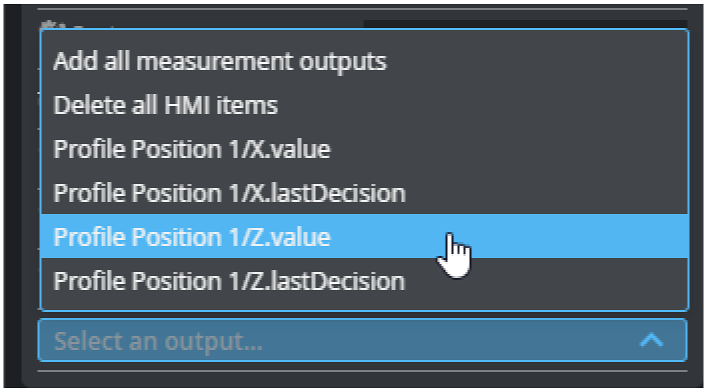
Export lets you export the HMI application currently available on the device (or in the PC instance).

2. If you see **In edit mode** as the HMI status instead of **Ready**, you must clear the workspace in GoHMI Designer.

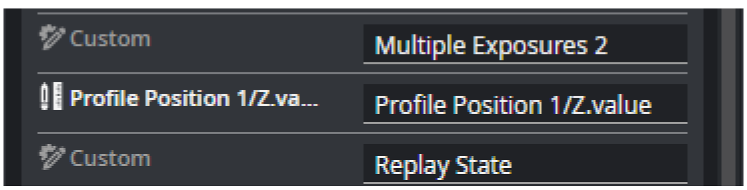


The status **In edit mode** means that the HMI app is currently loaded into the GoHMI Designer workspace. If you know you have finished making changes in GoHMI Designer, and have closed it without publishing it (to the sensor or PC instance of GoPxL), you can "force" exiting edit mode by clicking **Exit Edit Mode**. Note that doing this removes any unpublished changes in GoHMI Designer. For information on properly closing the designer and publishing changes, see *Exiting, and Saving and Publishing Changes* on page 861.

3. In the **Data to Send** panel below, expand the output drop-down and click an output you want to send to the HMI.



After a few seconds, the output appears in the list of outputs that GoPxL will send to the HMI.



You can toggle between showing HMI item aliases and HMI item paths of the outputs.

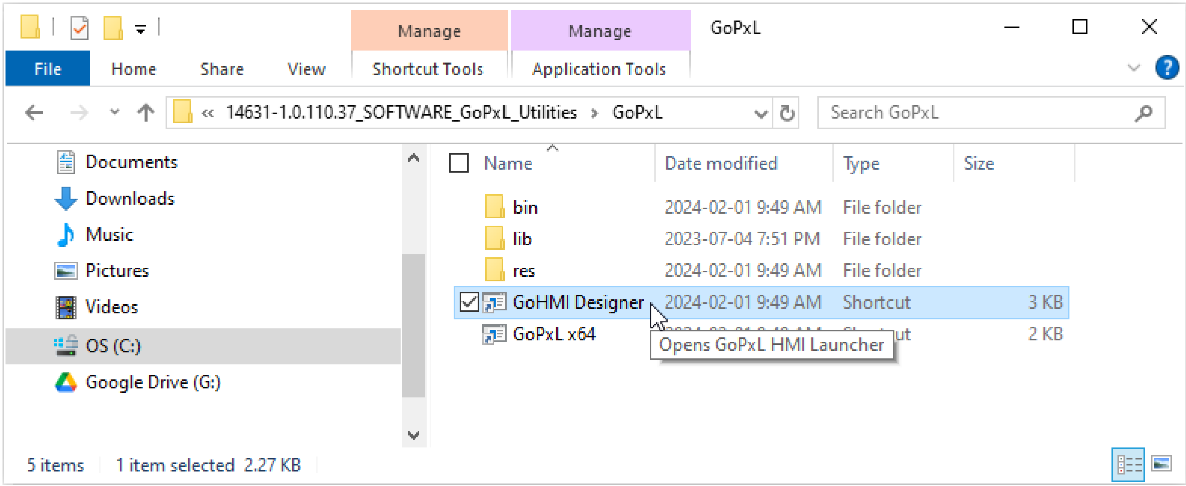
If you don't see the output you want to add, check that you have enabled it in the measurement tool, on the **Inspect** > **Tools** page.

Launching GoHMI Designer

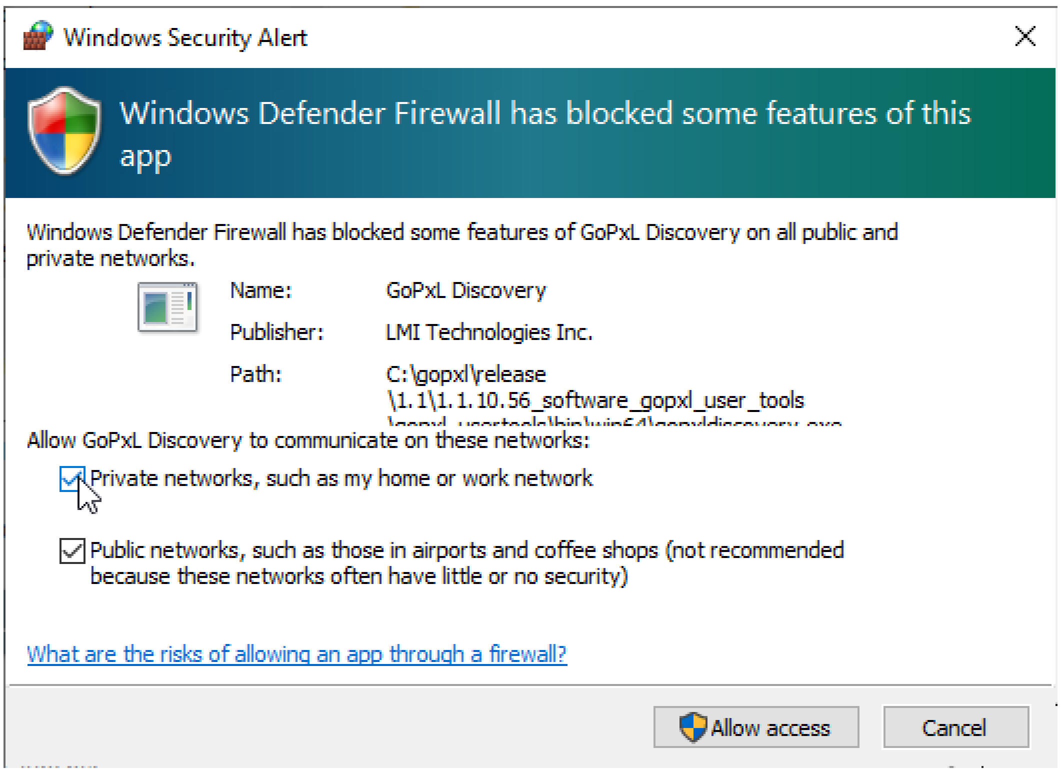
After you have enabled the HMI service in GoPxL and configured it by adding the outputs you want to send to the HMI, you can launch GoHMI Designer to create or edit an HMI.

To launch GoHMI Designer

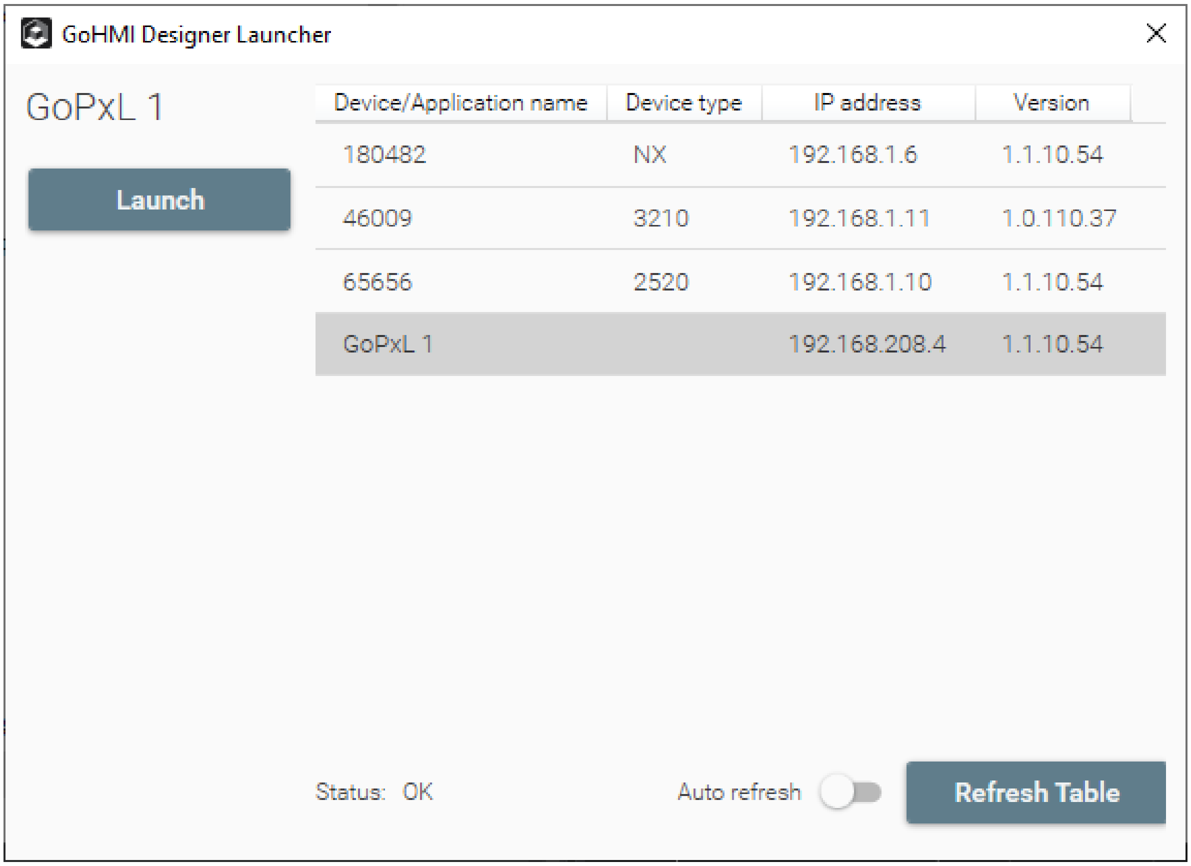
1. Go to where you extracted the GoPxL tools, and double-click the GoHMI Designer shortcut.



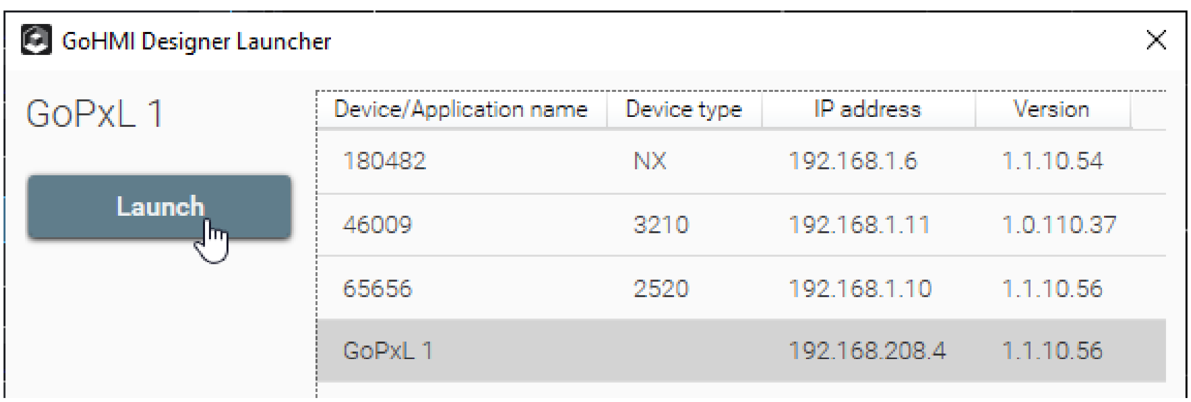
2. (Optional) If a Windows Security Alert opens, enable both networks and click **Allow access**.



GoHMI Designer Launcher opens and lists the available instances of GoPxL. In the following, we see two sensors, a PC instance of GoPxL called "GoPxL 1," and a GoMax NX device.



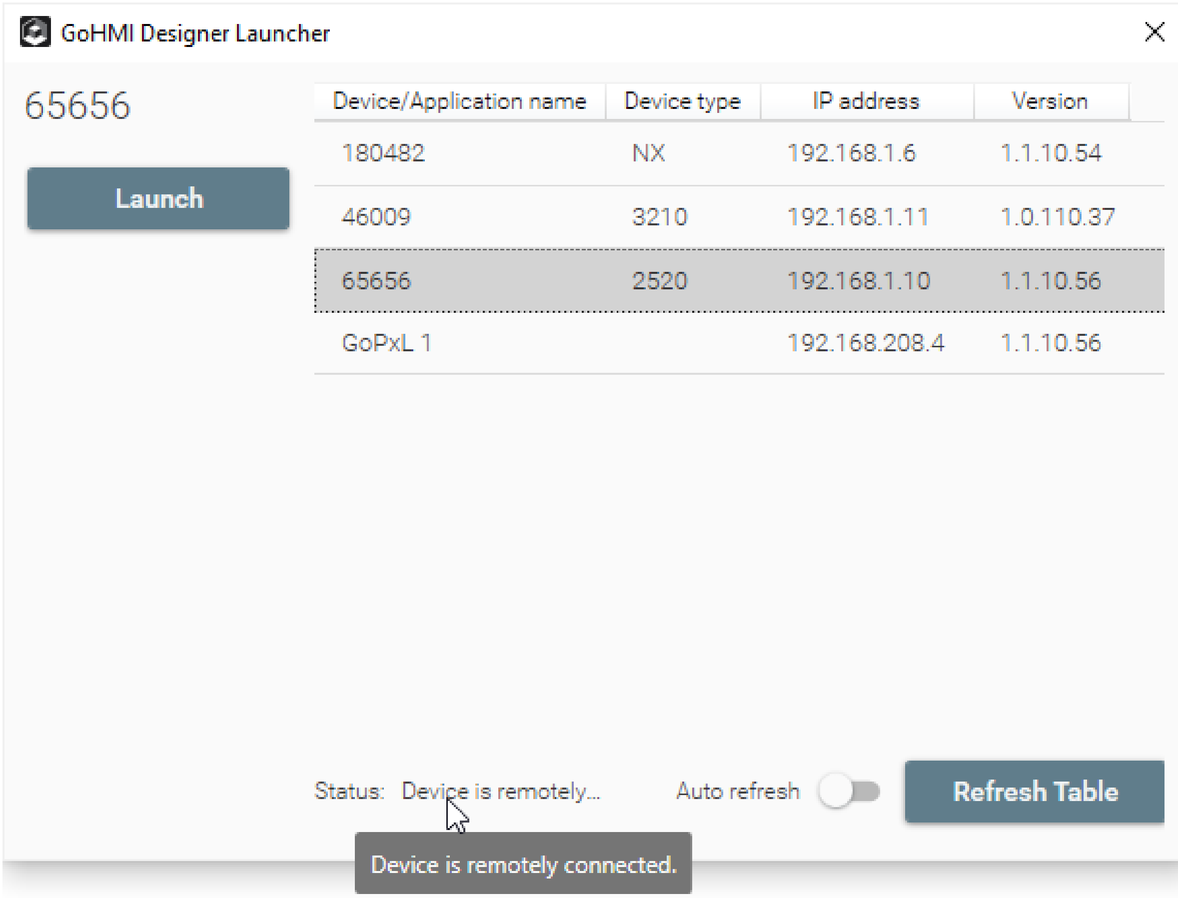
3. In the designer, select the row of the GoPxL instance for which you want to launch GoHMI Designer and click **Launch**.



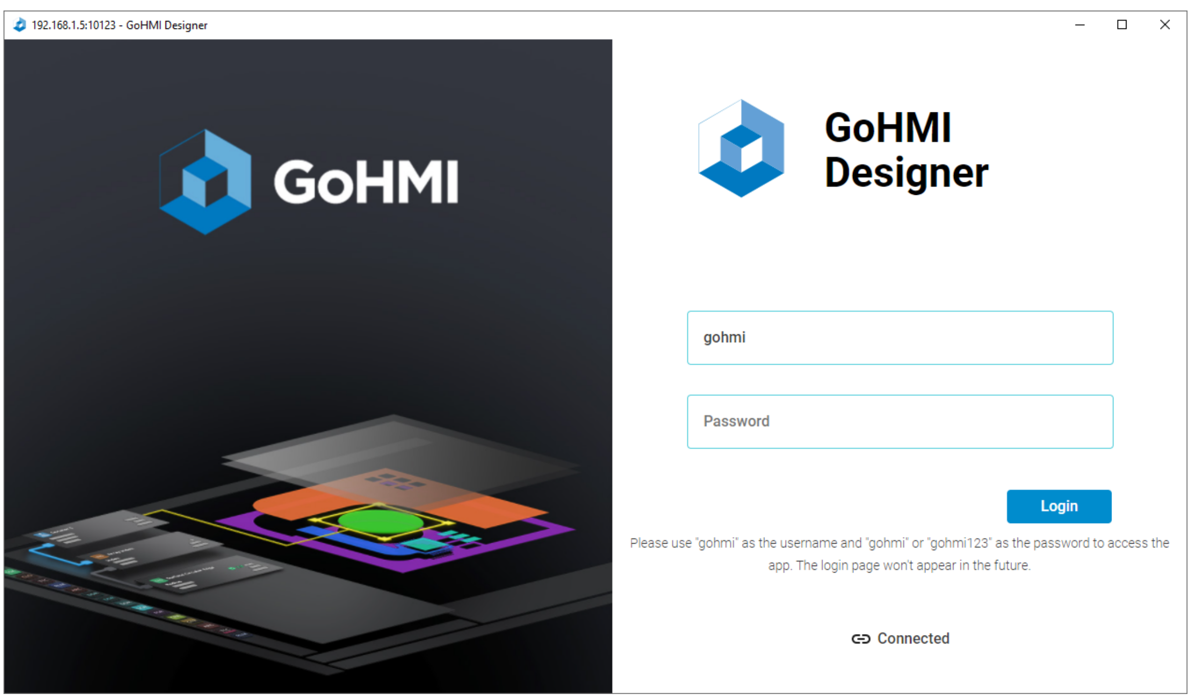
You may need to refresh the table to get the most up-to-date list by clicking **Refresh Table**. You can toggle **Auto Refresh** to have the table refresh automatically.

Note that you can only launch one instance of GoHMI Designer at a time.

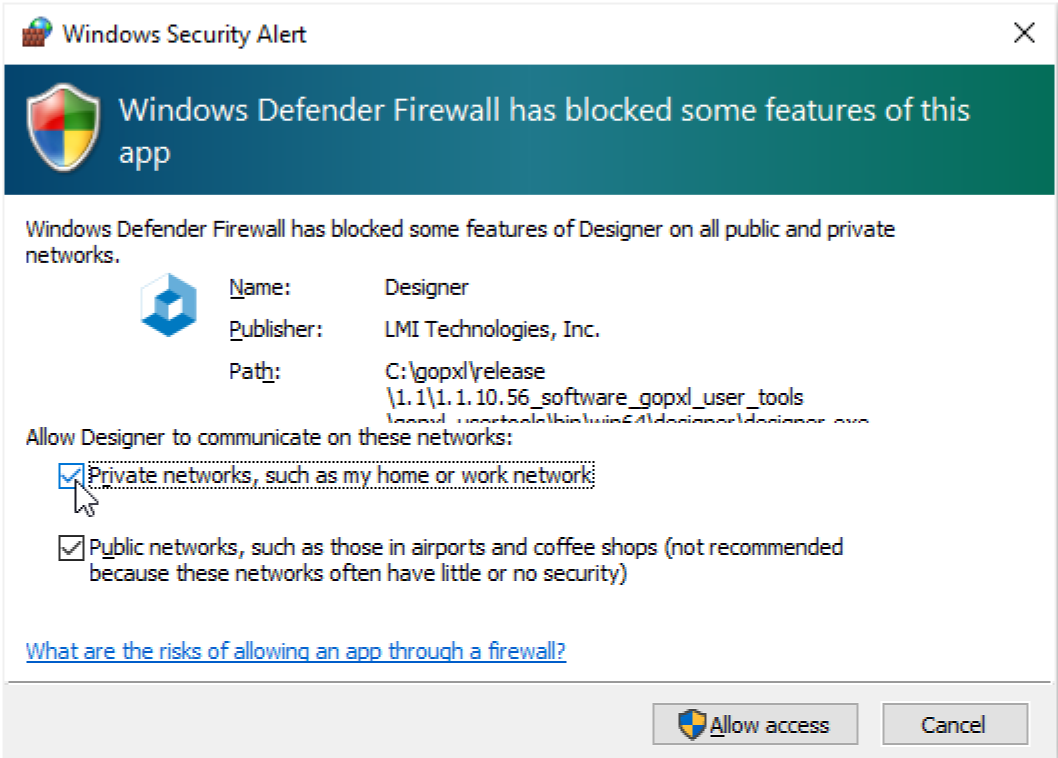
If a sensor is currently running through a PC instance of GoPxL, its status is "Device is remotely connected." In this case, you can only launch GoHMI Designer from the PC instance through which the sensor is running.



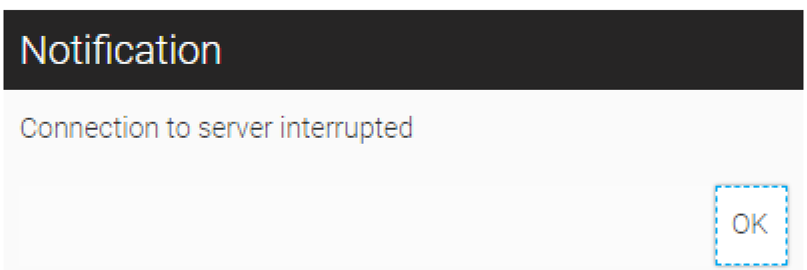
The launcher closes and the GoHMI Designer login screen appears.



A Windows Security Alert may appear. If so, allow the designer to communicate over both networks.



If you see a notification warning that the connection to the server has been interrupted, make sure you have enabled the HMI service in GoPxL. After enabling the service, click OK in the notification. (For more information, see *Enabling and Configuring GoHMI in GoPxL* on page 849.)



4. Log in to the designer.

   By default, use "gohmi" or "gohmi123" for the user name and "gohmi" for the password.

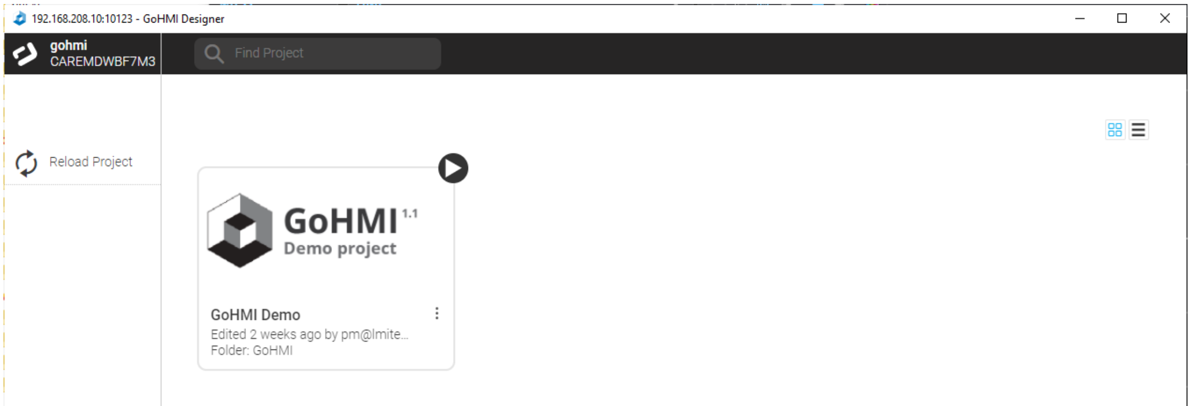
Opening, Editing, and Saving a Project

This section describes how to open the a GoHMI project (called GoHMI), modify it, and save it.

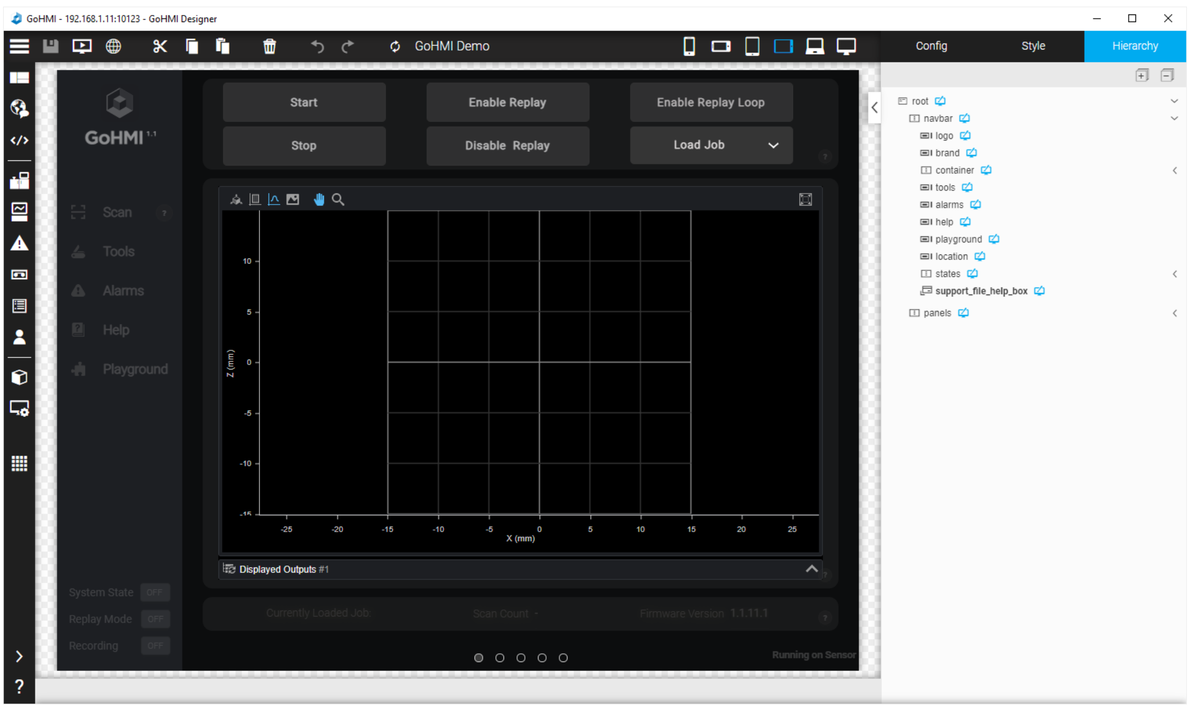
GoHMI Designer only supports a single HMI application at a time.

To open a project

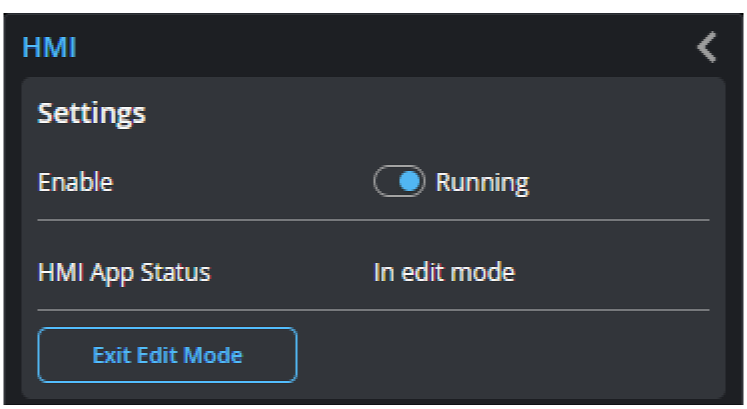
1. In the GoHMI Designer main screen, hover the mouse over the GoHMI project and click it.



The project opens in the Layout Manager.



Note that after the HMI loads in GoHMI Designer, in the GoPxL interface the HMI app status becomes "In edit mode."



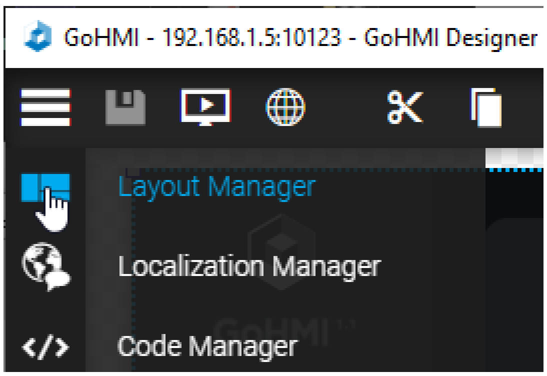
Adding a Widget

This section describes how to add a widget to the demo app. To keep things simple, we will simply describe how to add a label to the demo HMI app, change the text it displays, and modify the color of the text.

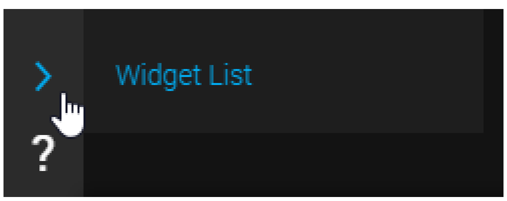
	Before modifying the demo HMI, you should export it for safe-keeping. For more information, see *Importing and Exporting an HMI* on page 863.

To add a widget

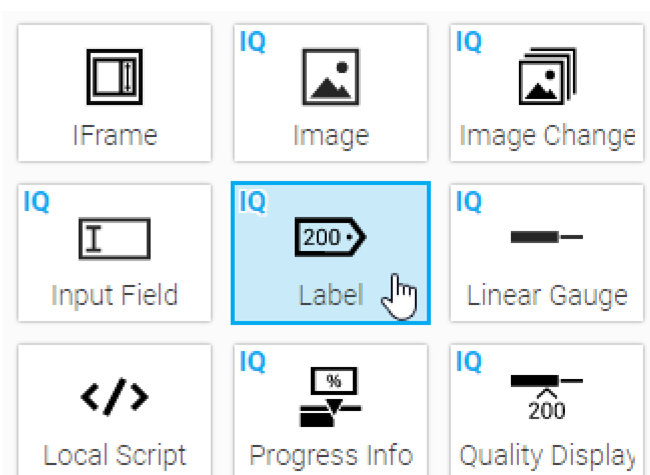
1. Move the mouse over side bar on the left and click Layout Manger to make sure you're in the Layout Manager.



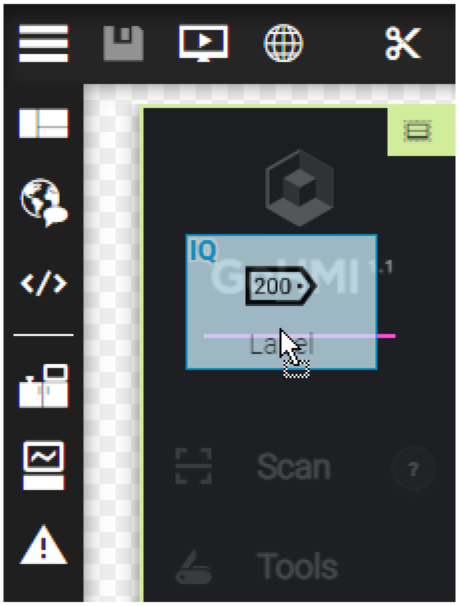
2. While still hovering over the side bar, move down to Widget List and click it.



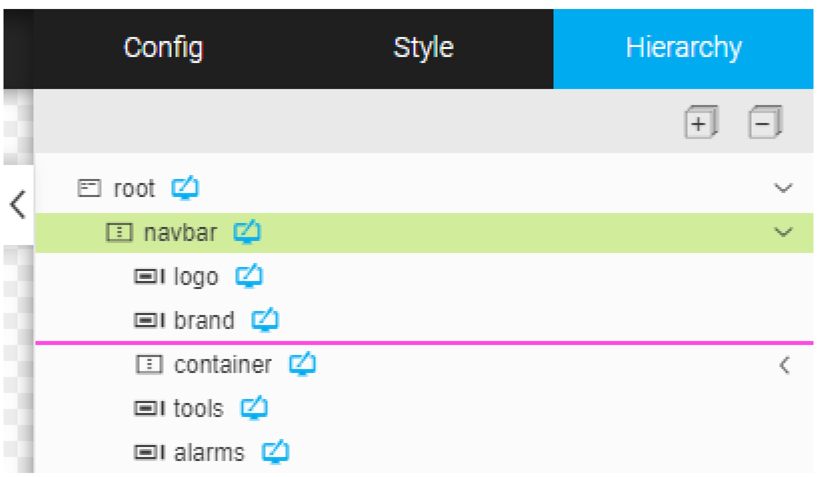
3. In the Widget List panel that opens, click and drag the Label widget toward the HMI in the Layout Manager.



As you move the widget over the HMI, notice that the container to which the widget would be added if you dropped it is indicated by a green rectangle. A pink line indicates where in the container it would be dropped.

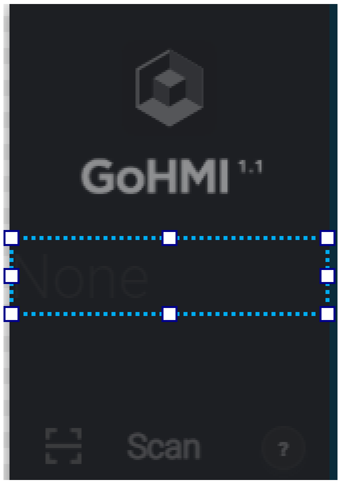


Note that as you move the widget over the HMI, its position is also shown in the hierarchy of containers and widgets in the **Hierarchy** panel to the right. You can also drag widgets onto this hierarchy to position them between widgets.

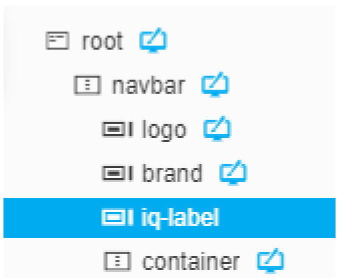


4. Drop the widget, for example after the GoHMI "brand" item.

   The widget is added to the HMI, and initially shows "(no value)". The default color of the text will make it difficult to read.

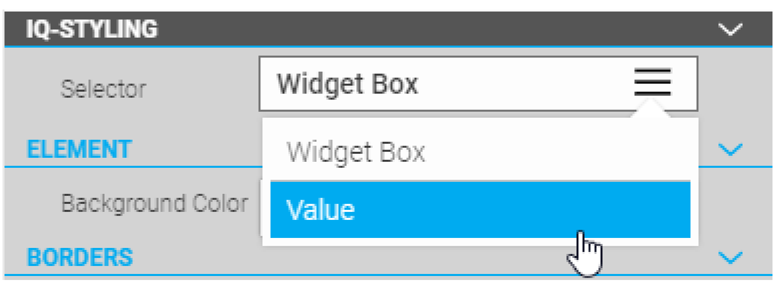


The Label widget now appears in the hierarchy as "iq-label," between the "brand" and "container" items.

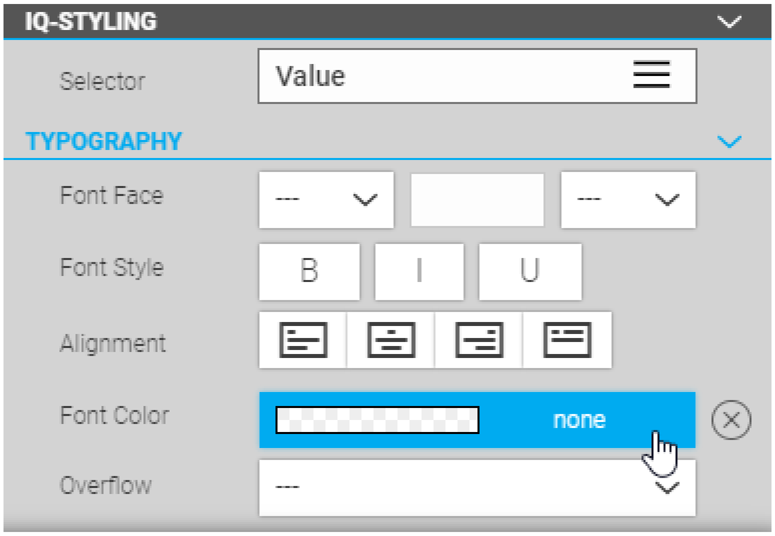


Understanding where widgets will "land" involves understanding container types and their alignment settings.

To style widgets, you will typically use a stylesheet, which is beyond the scope of this document. However, you can manually style a widget by selecting the widget in the **Hierarchy** panel, switching to the **Style** panel, scrolling down to the **IQ-Styling** section, and applying the desired style. For this example, you must change **Selector** from Widget Box to Value.

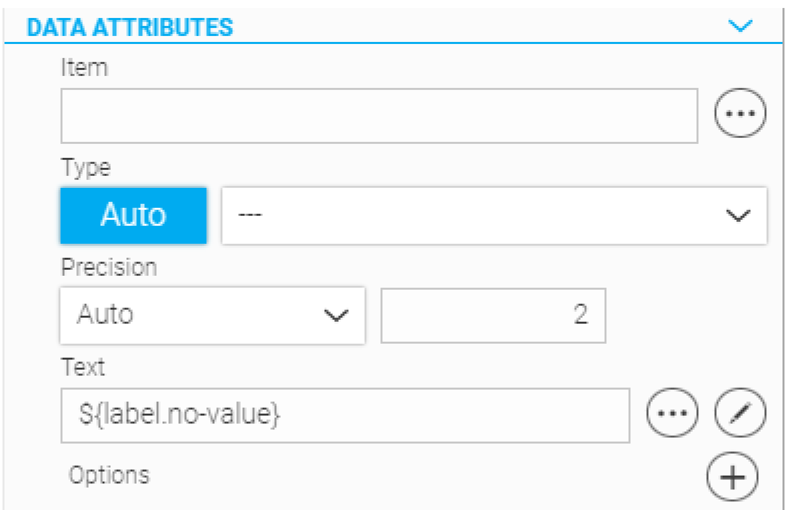


Click the **Font Color** selector.



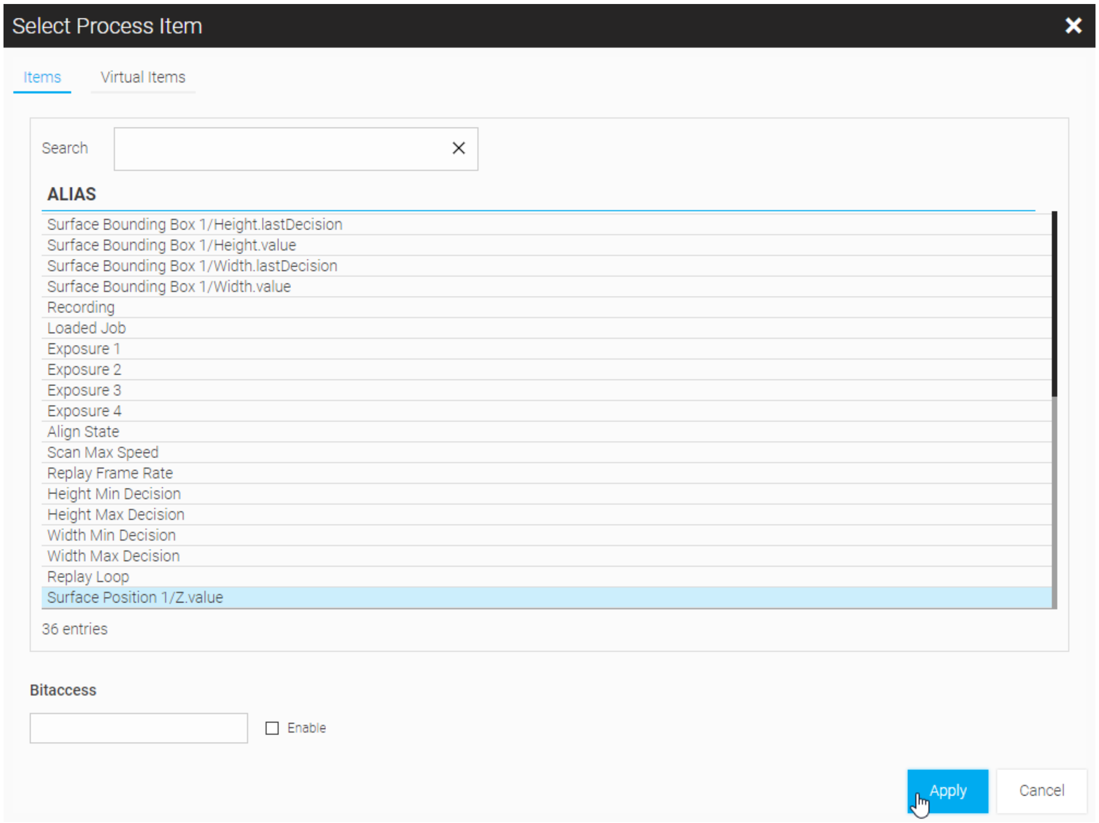
In the selector, replace "000000" with "ffffff", and click outside the selector to apply the change.

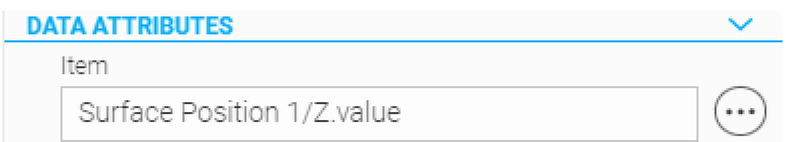
To set the value of the label, you must switch to the **Config** tab, and set it in the **Data Attributes** section. If you need to set the label to static text, edit the **Text** field. You can set the value directly here (by replacing the variable in the field with text), for example, with "My label".



You can also set up a variable in the Localization Manager and set the widget's text using that variable.

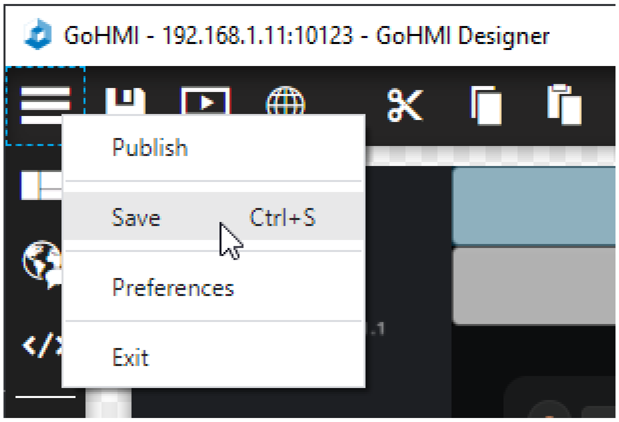
You can also set the value of the label to something the HMI is receiving from the sensor, such as a measurement value. To do this, in the **Data Attributes** section, click the three dots next to the **Item** field, choose the item in the **Select Process Item** dialog, and click **Apply**.



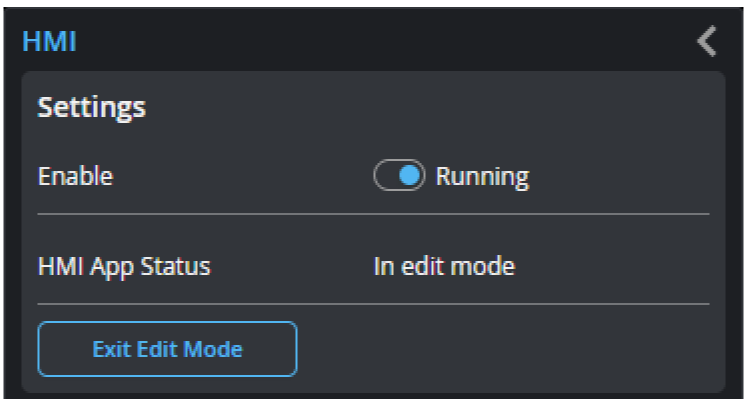


Exiting, and Saving and Publishing Changes

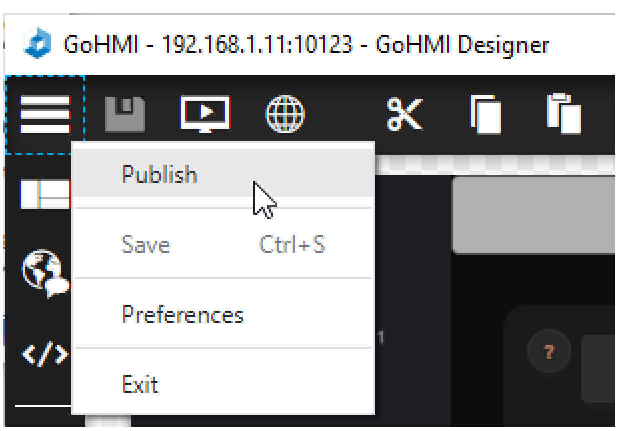
While you are working on an HMI, you should save it regularly. Save changes from the top left menu, by pressing Ctrl+S, or by clicking the Save icon () at the top of the interface. When you save changes, they are not written to the sensor or PC instance of GoPxL. They are only saved in the designer's workspace. If you close the designer by using the Close icon () at the upper right of the designer, the current state of the project is kept in the designer, and when you open the designer again, you can continue from where you stopped.



Note that when you close the designer using the Close icon (×), the HMI remains in "edit mode," and you can't run the HMI, or import / export it from GoPxL. To do this, you must publish the HMI (see below).

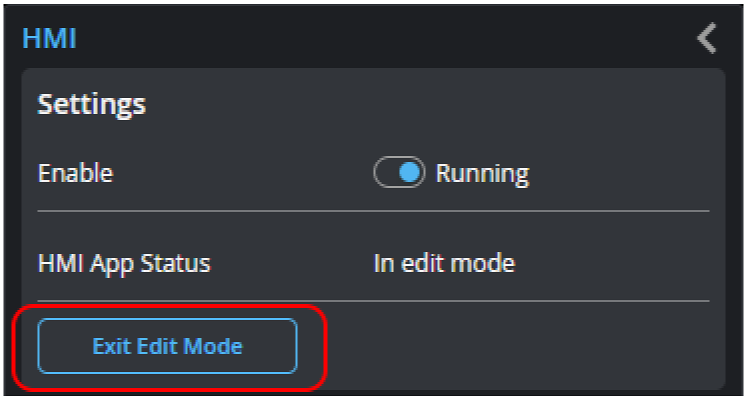


After you have finished working on an HMI in GoHMI Designer, you must publish it. Publishing an HMI copies the project loaded in the designer to the device or the PC instance of GoPxL. (These are called the "runtime.")



In some situations, such as when you have closed the designer without publishing unneeded changes, you can "force" exiting edit mode by clicking **Exit Edit Mode** in GoPxL, in the **Control**

> **HMI** page. Note that doing this removes any unpublished changes in GoHMI Designer, so make sure it is safe to do this.



Changing Mapped Outputs in GoPxL

To add or remove measurements at a later date in the list of outputs that are sent to the HMI application, the **HMI App Status** in GoPxL must say "Ready". If **HMI App Status** shows "In edit mode" and if you have unpublished changes in the designer that you want to keep, follow the steps in *Launching GoHMI Designer* on page 851 and then, in GoHMI Designer, publish the changes. For more information on publishing changes, see *Exiting, and Saving and Publishing Changes* on page 861. Otherwise, you can force GoPxL to exit edit mode by clicking **Exit Edit Mode**, but you will lose any unpublished changes.

Importing and Exporting an HMI

You can import an HMI app from the GoPxL interface. After importing an HMI app, you can then edit it in GoHMI Designer. After you have imported an HMI you can also start it from the GoPxL interface by clicking **Start** and then **View live**.

⚠	Importing an HMI app overwrites the HMI app currently in GoHMI Designer. Consider exporting the HMI app from the GoPxL interface on a regular basis.

To back up an HMI you have created or to provide it to someone else, you can export it.

	Imported and exported HMI apps are zip files.

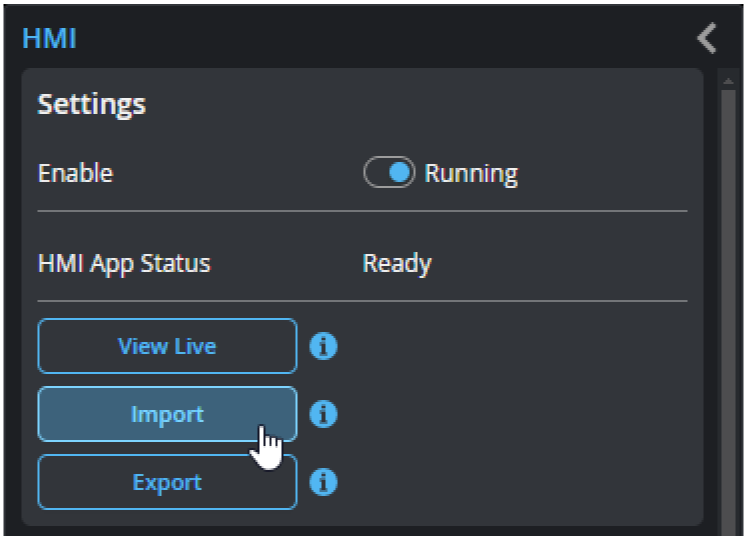
To import an HMI app

1. Make sure you have backed up the current HMI or that it is OK to overwrite it.

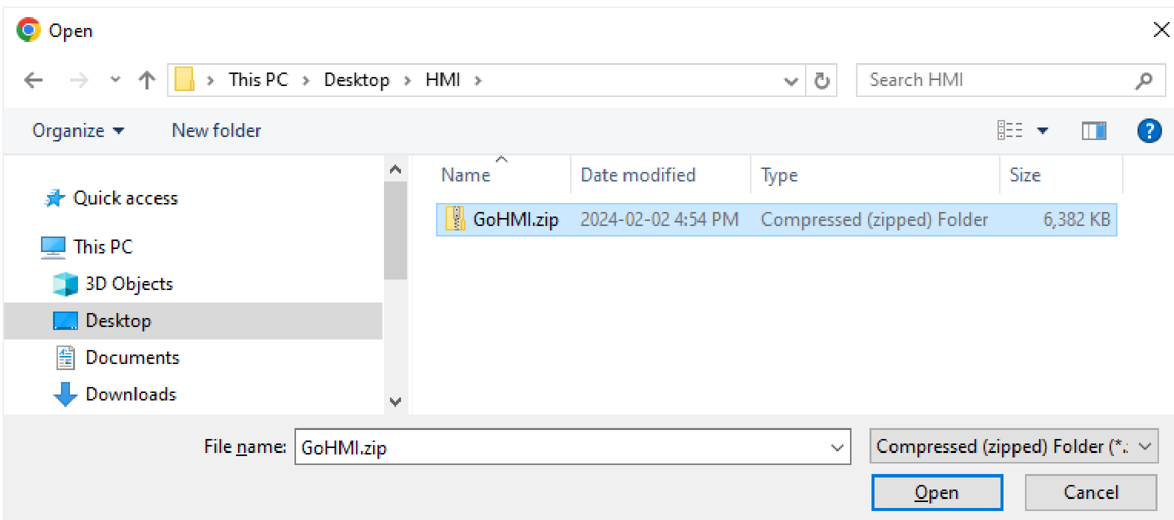
   Importing an HMI app overwrites the app in GoHMI Designer.

   For information on exporting an HMI, see *To export an HMI app* on the next page.

2. In the GoPxL interface, click **Import**.



3. In the Open dialog that displays, go to the location of the zipped HMI app project and click **Open**.



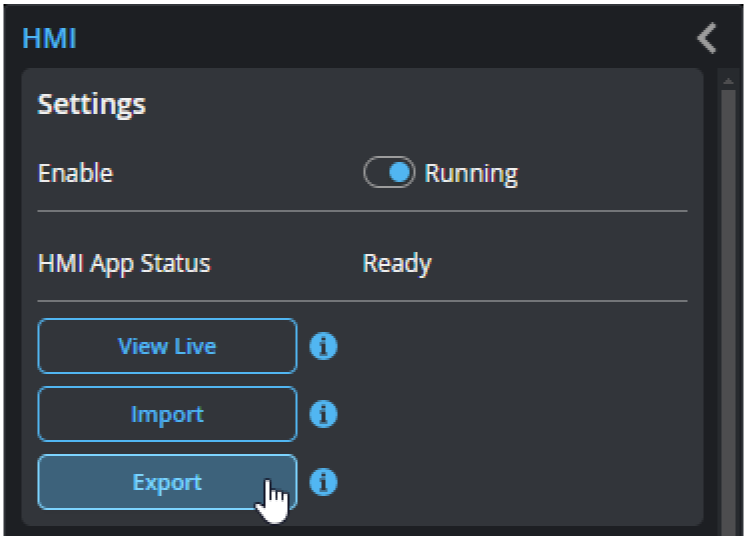
GoPxL imports the HMI app.

The app is now available for editing in GoHMI Designer (see See *Opening, Editing, and Saving a Project* on page 855).

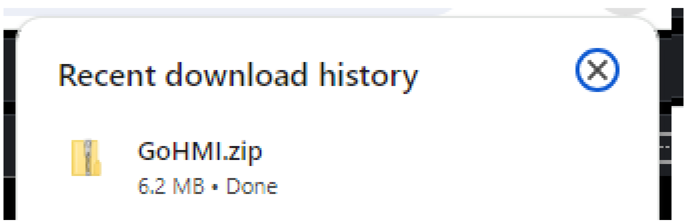
If you want to back up or send your HMI app to someone, you export it in GoPxL interface.

To export an HMI app

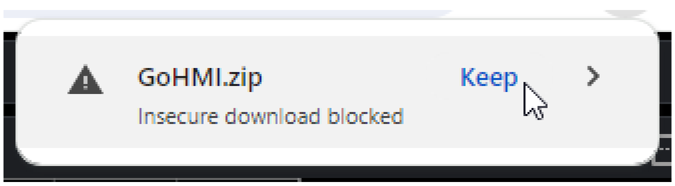
- In the GoPxL interface, click **Export**.



GoPxL downloads the HMI app and saves it as a zip file (GoHMI.zip).



You may need to tell your browser to accept the download (Chrome).



Using a Data Viewer in an HMI

You add a data viewer to an HMI by adding an IFrame widget in GoHMI Designer, and then copying a link from the GoPxL web interface (in the Displayed Outputs panel) and pasting it into the IFrame. The link tells the HMI the IP address of the sensor, as well as which sensor output and measurements it should display.

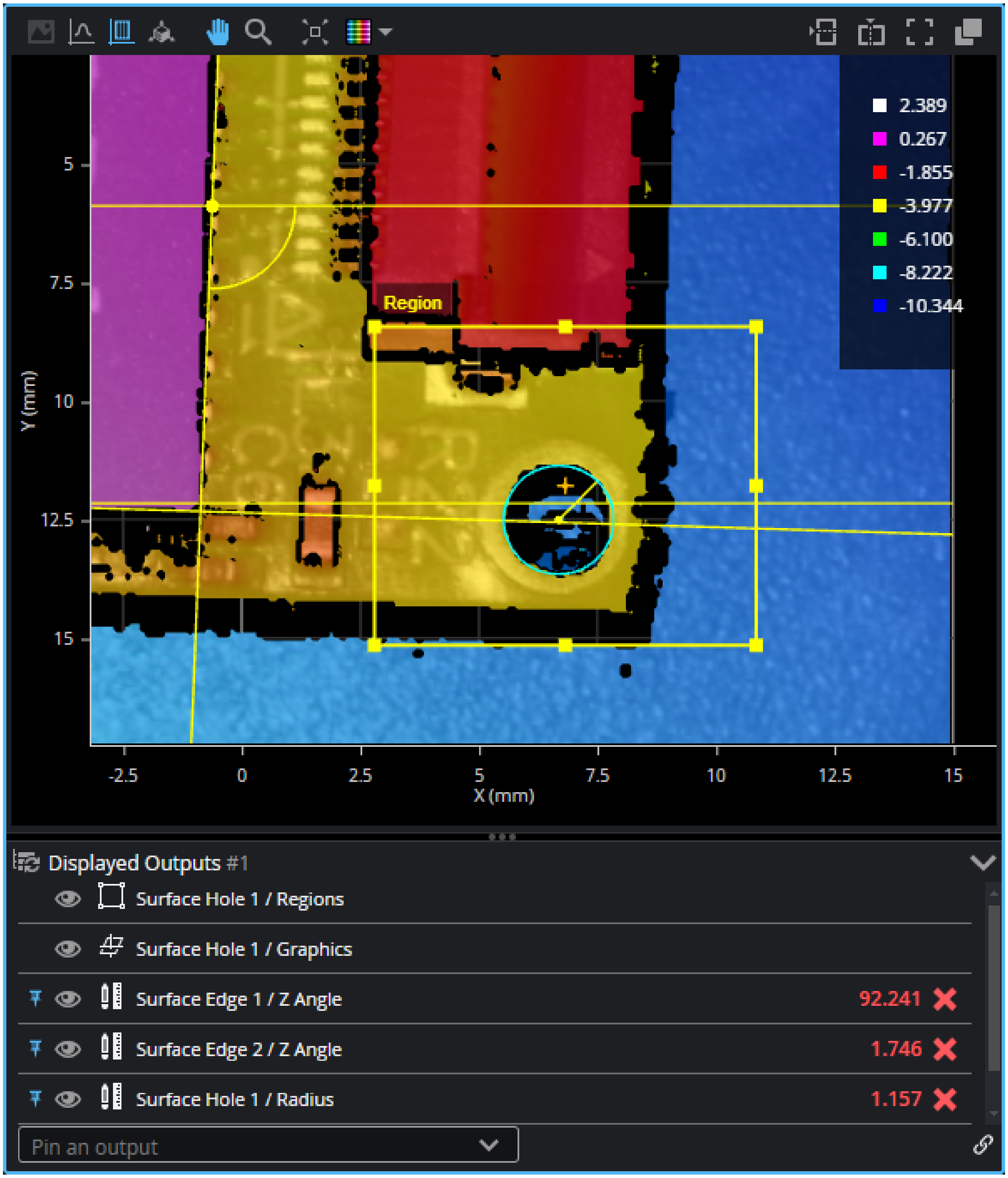
> The default HMI app includes an IFrame widget configured to display the data viewer. The IP address used in the link is 127.0.0.1:8100, which is valid for a PC instance of GoPxL (if one is running at that address and port).
>
> If you want to try out the default app with your sensor, follow the steps below to get the link from your sensor, and update the IFrame's **Source URL** field with that link (see steps 1 to 3, and step 8).

To add a data viewer to an HMI

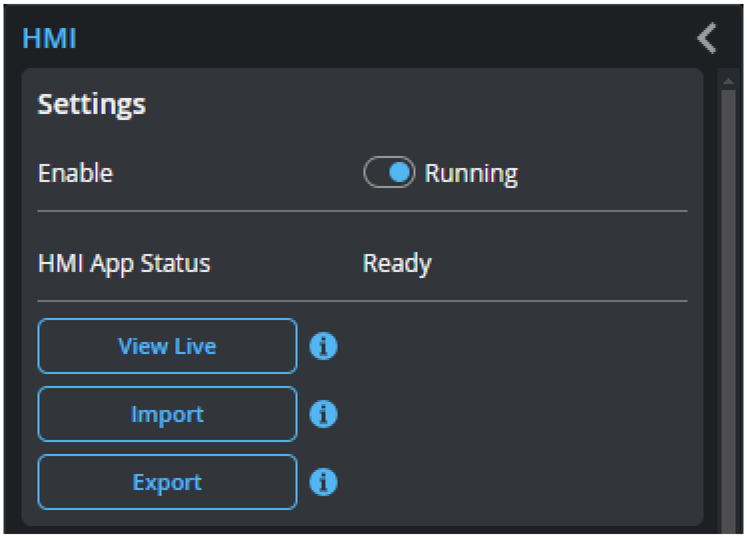
1. In GoPxL, set up a data viewer as required.

   For example, you can pin measurements so that they appear in the data viewer.

2. Under the data viewer window, make sure the Displayed Outputs pane is expanded.



3. Click the link icon () in the lower right of the Displayed Outputs pane to copy the path for the data viewer.
4. Make sure GoHMI is enabled and configured in the **Control** > **HMI** page.

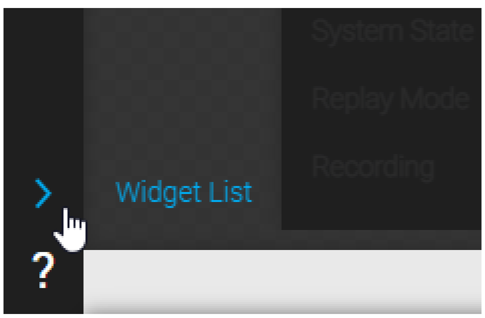


For more information, see *HMI* on page 817.

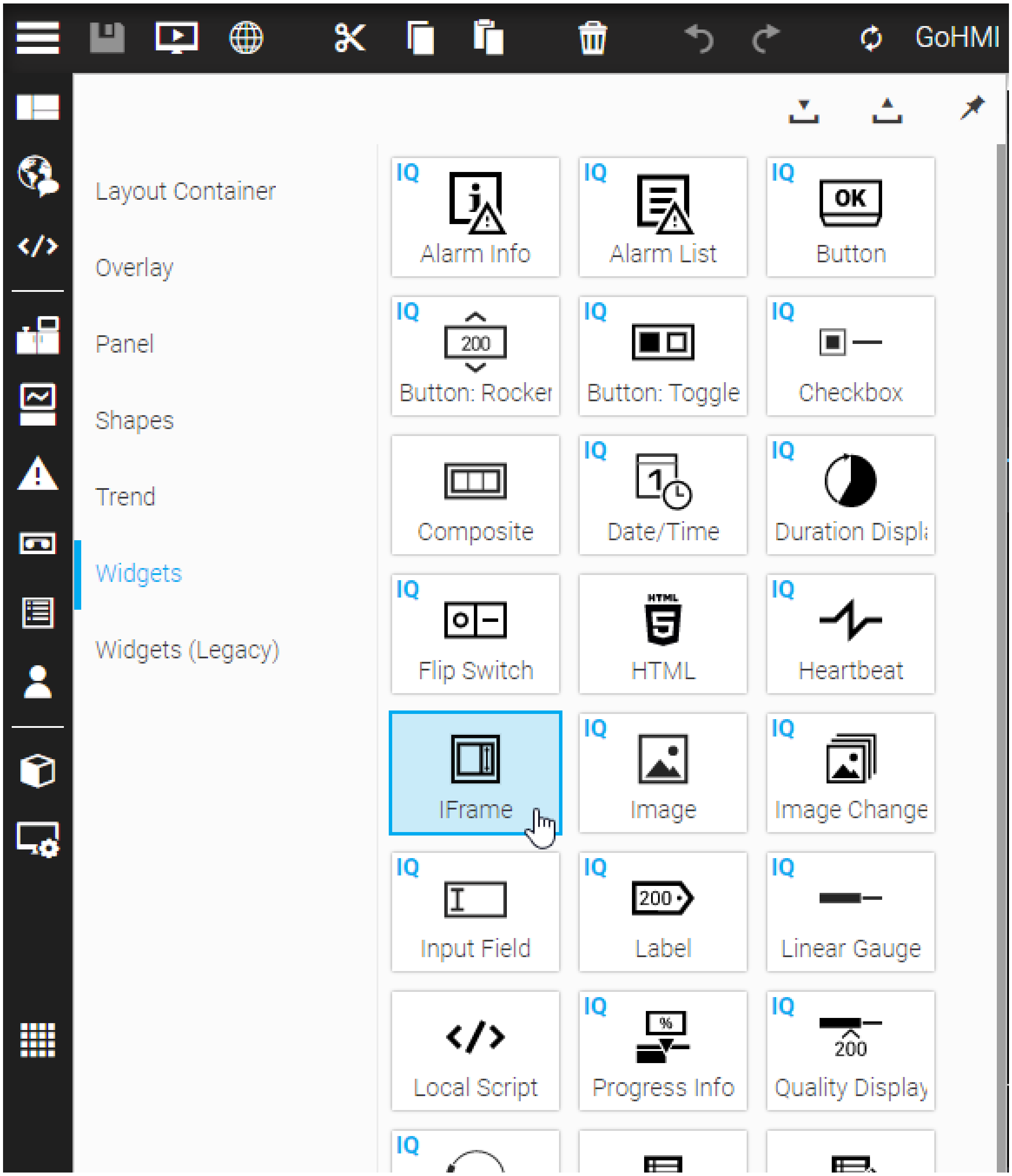
5. Launch GoHMI Designer.

   For information on launching GoHMI Designer, see *Launching GoHMI Designer* on page 851.

6. In GoHMI Designer, in the Layout Manager, expand the widget list.

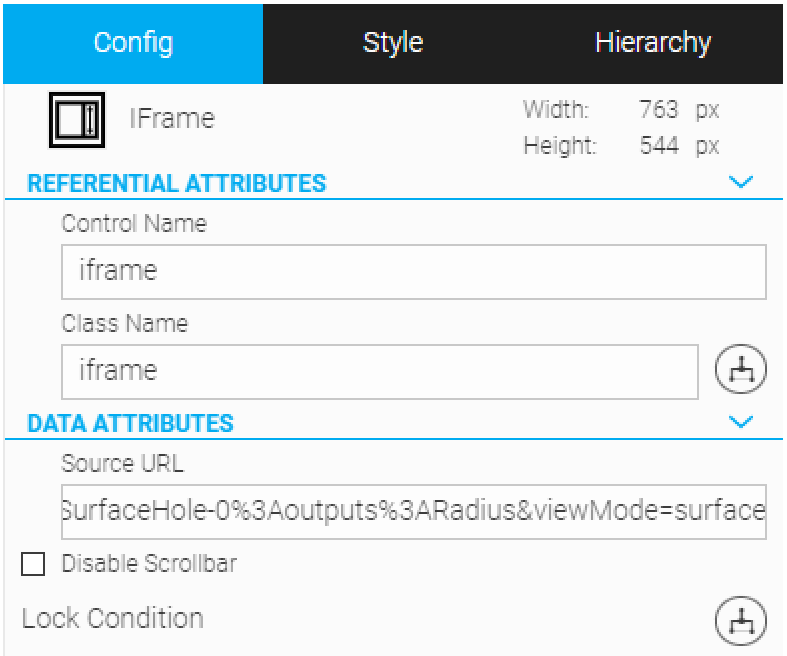


7. From the widget list, drag an IFrame widget into the HMI.



Resize the IFrame widget if necessary.

8. In the **Config** tab, paste the URL you copied in step 1 into **Data Attributes** > **Source URL**.

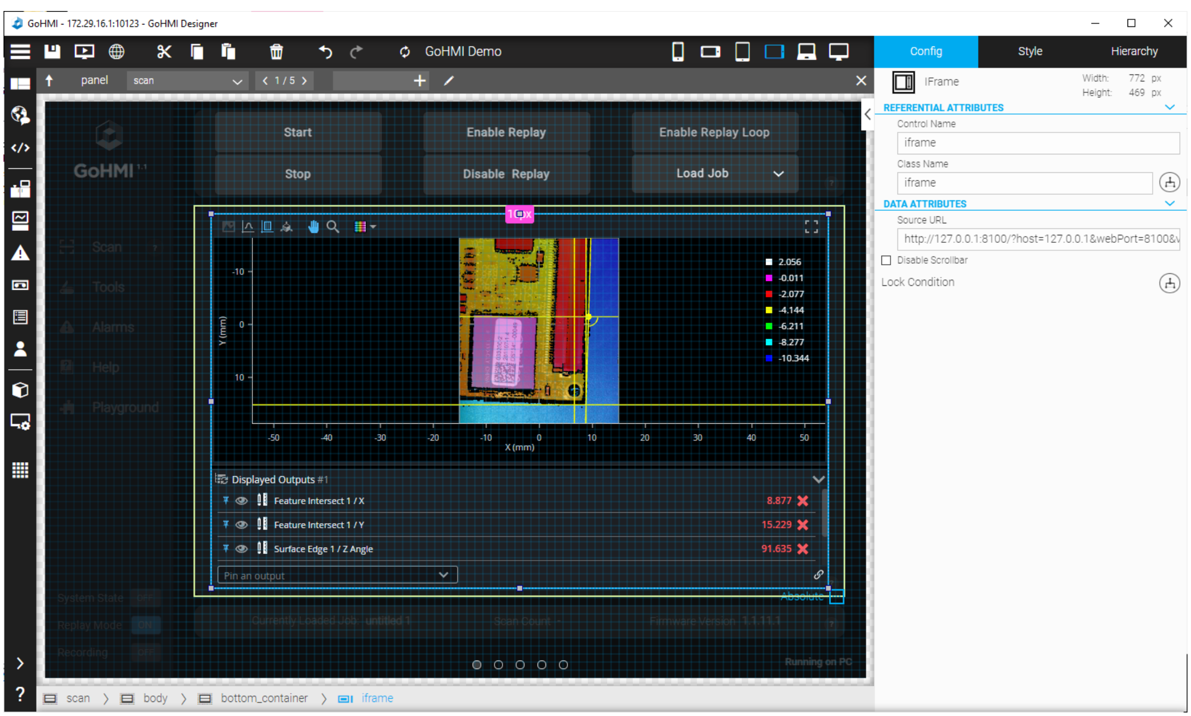


The URL will look something like this:

```
http://127.0.0.1:8100/?host=127.0.0.1&webPort=8100&visualizerOnly=true&displayedOutputs=Surface.Top,tools%3AFeatureIntersect-5%3Aoutputs%3AX,tools%3AFeatureIntersect-5%3Aoutputs%3AY,tools%3ASurfaceEdge-2%3Aoutputs%3AZAngle,tools%3ASurfaceHole-1%3Aoutputs%3AX&viewMode=surface
```

After a few seconds, the data viewer from the connected instance of GoPxL appears in the IFrame widget.

The following shows what the default HMI app looks like after updating the IFrame widget's source URL.



Limiting Flash Memory Write Operations

Several operations and REST API / SDK functions write to the sensor's flash memory. The lifetime of the flash memory is limited by the number of write cycles. Therefore it is important to avoid frequent write operation to the sensor's flash memory when you design your system with the SDK or GoHMI.

	Power loss during flash memory write operation will also cause sensors to enter rescue mode.

GoPxL SDK and REST API

The GoPxL Software Development Kit (GoPxL SDK) includes open-source software libraries and documentation that you can use to programmatically access and control GoPxL devices.

> The GoPxL SDK and REST API use the term "scanner" instead of "sensor group." The two are interchangeable.

> The GoPxL SDK uses mm, mm^2, mm^3, hertz, microseconds, and degrees as standard units unless otherwise mentioned.

The following sections describe the Software Development Kit (GoPxL SDK).

Installation Guide

Location of GoPxL SDK and Documentation

The GoPxL SDK, and the SDK and REST API references are available in the `14630-1.1.x.x_SOFTWARE_GoPxL_SDK.zip` package.

The zip file contains both a C++ and a .NET SDK organized as follows:

- `GoPxL_SDK_Cpp\`
- `GoPxL_SDK_dotNet\`

Visual Studio Solution Files and Linux Makefile

For Windows, Visual Studio solution files are included for both the C++ and .NET SDKs.

- C++: `GoPxL_SDK_Cpp\Gocator`
- .NET: `GoPxL_SDK_dotNet\Gocator`

For Linux, first ensure that you have the g++ compiler package and then run the following command from the `GoPxL_SDK_Cpp\Gocator\GoPxLSdk` directory:

```
make -f GoPxLSdk-Linux_X64.mk
```

Note that only the C++ SDK is supported on Linux.

Building the SDK

To compile in Visual Studio, you may need to retarget the solution to the installed Windows SDK version. To retarget a solution, open the solution in Visual Studio, right-click the solution in the Solution Explorer, choose **Retarget solution** in the context menu. In the dialog box that appears, select the desired version from the drop-down list and click **OK**.

The compiled Linux output libraries can be found in the `GoPxL_SDK/lib/linux_x64` directory.

System Requirements

The GoPxL SDK can be compiled using the following environment.

For Windows:

- Visual Studio 2017 or later
- Windows SDK (included with Visual Studio installation)

For Linux:

- GCC X64 compiler, version 7.5.0 (GccX64_7.5.0-p3)
- GNU Make

Accessing Documentation

The full SDK class references and REST API resources are available at the following locations:

GoPxL SDK Class Reference

C++: `GoPxL_SDK_Cpp\doc\GoPxLSdk\GoPxLSdk.html`

.NET: `GoPxL_SDK_dotNet\doc\GoPxLSdk\GoPxLSdk.html`

REST API Resources

C++: `GoPxL_SDK_Cpp\doc\rest-api-doc.html`

.NET: `GoPxL_SDK_dotNet\doc\rest-api-doc.html`

Sample Projects

Sample projects showing how to perform various operations are provided, each one demonstrating a specific area. Examples are located in the following folders:

- C++: `GoPxL_SDK_Cpp\Gocator\GoPxLSdk\samples`
- .NET: `GoPxL_SDK_dotNet\Gocator\GoPxLSdkNet\samples`

Header Files

When including header files from the GoPxL SDK, reference the GoPxLSDK directory as the source directory. For example:

`#include <GoPxLSdk/GoRestClient.h>`.

Additionally, note that the SDK header files reference files from the kApi directory. For example:

`#include <kApi/kApiDef.h>`.

Getting Started with the SDK

GoPxL offers a simple programming API interface for connecting, configuring, and acquiring data from sensors and the GoMax vision accelerator.

> Example code shown below is for C++.

All GoPxL SDK programs should have the sequence described below, at a minimum.

Connecting a Device

The following sequence assumes you know the IP address of the device. If you do not know the IP address, or if the address is dynamic, an application can discover the devices that are available on the network using the GoPxL SDK discovery capability. For more information, see *Discover Devices* on the next page.

1. Initialize the SDK library

   For C++ SDK programs only, construct the SDK library by calling `GoApiLib_Construct`. You must call `GoApiLib_Construct` before any API functions are used. (.NET SDK programs don't require this.)

```
// Constructs GoPxL API core framework.
GoApiLib_Construct(&goApiLib)
```

   After completing your operations with the SDK library, don't forget to release the resources using `kDestroyRef()`.

```
// Releases resources and resets the object handle to kNULL
kDestroyRef(&goApiLib)
```

2. Instantiate a `GoSystem` object and connect the device.

```
// Define device IP address and control port
constexpr const char* DEVICE_IP_ADDRESS = "192.168.1.10";
constexpr const uint32_t CONTROL_PORT = GO_PXL_SDK_DEFAULT_CONTROL_PORT; // Defaults is 3600


// Parse device IP address
kIpAddress deviceIpAddress;
kIpAddress_Parse(&deviceIpAddress, DEVICE_IP_ADDRESS);

// Create a GoSystem object
GoSystem system;

// Set up connection
system.SetAddress(deviceIpAddress);
system.SetControlPort(CONTROL_PORT);

// Connect to the device
system.Connect();
```

GoSystem Class Member Functions

The `GoSystem` class represents a single Gocator system. A Gocator system consists of one or more sensors/devices. Below are the member functions of the GoSystem class

```
// SetAddress: Sets the IP address used for connecting to the device.
void SetAddress(const kIpAddress& ipAddress);

// Address: Gets the IP address used for connecting to the device.
kIpAddress Address() const;

// SetControlPort: Sets the control port used for connecting to the device.
void SetControlPort(k32u port);

// ControlPort: Gets the control port used for connecting to the device.
k32u ControlPort() const;

// Connect: Connects to the device.
void Connect();

// Disconnect: Disconnects from the device.
void Disconnect();

// IsConnected: Checks if the device is connected.
bool IsConnected();

// Start: Starts the device.
void Start();

// Stop: Stops the device.
void Stop();

// RunningState: Gets the current state of the device.
State RunningState();

// Client: Returns a reference to the REST API client for performing operations.
GoRestClient& Client();

// SensorPath: Retrieves the REST API resource path of a sensor based on its serial
number.
ResourcePath SensorPath(SerialNum serialNum);
```

Discover Devices

If a device IP address is not known (for example, because it is assigned through DHCP), you can discover the devices available on the network using the GoPxL SDK discovery function:

1. Include the `GoDiscoveryClient` header file.
2. Create an instance of `GoDiscoveryClient` and call the `BlockingDiscover` function with a specified timeout.
3. Access the list of discovered instances.

```
#include "GoDiscoveryClient.h"
auto discovery = std::make_unique<GoDiscoveryClient>();
```

```
discovery->BlockingDiscover(3000);
auto& instances = discovery->InstanceList();
```

4. After discovering the instances, you can loop through the instances to get IP addresses and other instance-related information.

```
// Print instance IP address and GDP port.
for (const auto& instance : instances) {
    std::cout << "IP Address: " << kIpAddress_AsString(instance.GetIpAddress()) << std::endl;
    std::cout << "GDP Port: " << instance.GetGdpPort() << std::endl;
}
```

For more information, see the *Discover* sample.

Configure Device Settings

You configure GoPxL device settings, which are represented as JSON resources, through JSON-based REST API commands. The `GoRestClient` class provides access to read and write settings. You can use methods like Call, Read, Update, Create, and Delete, among others, to interact with the device settings.

Start by instantiating a `GoSystem` object. Through this object, you can directly access the `GoRestClient` instance via `system.Client()`. Then, to perform `Read()` or other operations, simply continue chaining the methods, for example, `system.Client().Read()`.

These methods follow a similar format:

```
"command /path/to/callable/resource {'argument':'value'}"
```

For example the following command:

```
call /jobs/commands/rename { "sourceName" : "job1", "destName" : "renamedJob1"
}
```

can be implemented as follows:

```
// Rename job.
auto payload = GoJson(R"({
    "sourceName" : "SDK-demo",
    "destName" : "SDK-demo-job"
})");

system.Client().Call("/jobs/commands/rename", payload);
```

For the full list of resources available, see the *GoPxL 1.1 - REST API Resources and User Guide*:

C++: `GoPxL_SDK_Cpp\doc\rest-api-doc.html`

.NET: `GoPxL_SDK_dotNet\doc\rest-api-doc.html`

It's important to understand that GoPxL device settings are divided into two general types:

- sensor-level: These settings only apply to individual sensors.
- scanner-level: These settings apply to all of the sensors in a group of sensors (a "scanner").

To help explain the functions, some sections below include definitions of the API resources.

Accessing Sensor Resources

The most common use case is modifying settings that apply to individual sensors, such as exposure or active area.

If you know a device's serial number, you can access the corresponding resource using the `SensorPath` function. You can then modify the settings under the sensor's resources.

```
// Update single exposure value.
auto sensorPath = system.SensorPath(sensorSerialNumber);
auto payload = GoJson(R"({
    "parameters" : {
        "exposureSettings" : {
            "exposureMode" : 0,
            "singleExposure" : 400
        }
    }
})");

system.Client().Update(sensorPath, payload);
```

Accessing Scanner Resources

You access system-level settings (those that apply to a group of sensors) through the `scanner` resources.

If you know the engine ID and scanner ID (available in the REST API reference documentation), you can use the following to access the sensor group and modify the trigger setting:

```
// Access and modify trigger settings for the scanner.
auto scannerPath = "/scan/engines/LMILaserLineProfiler/scanners/scanner-0";
auto payload = GoJson(R"({
    "parameters" : {
        "triggerSettings" : {
            "source" : 3
        }
    }
})");

system.Client().Update(scannerPath, payload);
```

Configuring scanner resources through the SDK is less common. These resources are related to physical setup. Normally, you configure these settings through the user interface during the initial system installation and setup, and the settings don’t need to change for different production runs.

Accessing an Array Element

Some resources are returned as an array. One example is the `visibleSensors` resource, which can have multiple elements.

The following returns an array of sensors under the `/scan/visibleSensors` resource path.

```
sensors = system.Client().Read("/scan/visibleSensors/").GetResponse().Content().At("sensors");

// Get the first element in the array
auto sensor = sensors.Begin();
auto sensorSerialNumber = sensor.Value().At("/serialNumber").Get<std::string>();

// Alternatively, iterate through each sensor in the array
for (auto sensor = sensors.Begin(); sensor != sensors.End(); ++sensor)
    auto sensorSerialNumber = sensor.At("/serialNumber").Get<std::string>();
}
```

Receive Device Data

GoPxL streams out acquired data (such as Profile and Surface data, and measurements) using the GoPxL Data Protocol (GDP). To receive the data, you use a `GoGdpClient` instance, which you first need to connect to the device. You can use `GoPxLSdk::MessageType` to identify the type of data received and the associated stamp information. Stamps contain encoder and trigger timing information, current time, and frame index, and are grouped into datasets with their corresponding data.

Data can be received on the same thread used for controlling the sensors (that is, synchronously) or on a different thread (that is, asynchronously). Note that data is received as the `GoDataSet` type. For more information on `GoDataSet`, see *GDP Data Types* on page 880.

For synchronous data retrieval, use the `ReceiveData()` method:

1. Instantiate a GoPxL Data Protocol client and connect to a device.

```
#include <GoPxLSdk/GoGdpClient.h>

constexpr k32u GDP_PORT = GO_PXL_SDK_DEFAULT_GDP_SERVER_PORT; // Default port is 3601

// Instantiate a GoGdpClient object
auto gdpClient = std::make_unique<GoGdpClient>();

// Connect the GoGdpClient to the device using sensor IP and GDP port
gdpClient->Connect(SENSOR_IP, GDP_PORT);
```

2. Receive data.

```
// Establish a synchronous data transfer with a 3-second timeout.
gdpClient->ReceiveDataSync(3000);

```

```
// Retrieve the received data set.
const GoDataSet& receivedDataSet = gdpClient->DataSet();
```

Then loop through each message (a unit of data received from the GoPxL sensor) to handle the data based on its type (Profile, Surface, Measurements, and so on) using the `MessageType` enum. For example:

```
for (size_t i = 0; i < receivedDataSet.Count(); ++i)
{
    switch (receivedDataSet.GdpMsgAt(i).Type())
    {
        case GoPxLSdk::MessageType::STAMP:
            // Handle stamp message
            ...

        case GoPxLSdk::MessageType::UNIFORM_SURFACE:
            // Handle uniform surface data
            ...
    }
}
```

For more information on synchronous data retrieval, see the `ReceiveProfile` and `ReceiveSurface` samples.

For asynchronous data retrieval, you use the following steps:

1. Instantiate a GoPxL Data Protocol client and connect to a device.
2. Start the data reception thread

   Call `GoGdpClient::ReceiveDataAsync(func)` to initiate data reception asynchronously. This starts a background thread (`receiveThread`) to receive and deserialize messages continuously.
3. Handle received data.

   Upon receiving individual data messages, they are reassembled into a `GoDataSet` object. The data reception continues until you call `GoGdpClient::Close()`.
4. Process the data.

   The received data sets are added to a message queue (`Go::MsgQueue<GoDataSet> dataQueue`). A separate thread (`dataThread`) removes the data sets from the queue for processing.
5. Access the data

   Define a callback function to handle the received data sets.

The following is an example of retrieving data asynchronously:

```
void onData(const GoDataSet& receivedDataSet)
{
    for (size_t i = 0; i < receivedDataSet.Count(); i++)
    {
        switch (receivedDataSet.GdpMsgAt(i).Type())
        {
            case GoPxLSdk::MessageType::STAMP:
            {
                const GoGdpStamp& stampMsg = (const
```

```
GoGdpStamp&)receivedDataSet.GdpMsgAt(i);
                // Process stamp message
               ...
            }
            // Handle other message types as needed
        }
    }
}

// Connect to GDP and start receiving data asynchronously
gdpClient->Connect(SENSOR_IP, GDP_PORT);
gdpClient->ReceiveDataAsync(onData);
```

For details, see the `ReceiveAsync` sample.

The following lists the `GoGdpClient` class methods you can use to interact with the GoPxL Data Protocol (GDP) server.

```
// Constructs GoGdpClient.
GoGdpClient();

// Destructs GoGdpClient.
~GoGdpClient();

// Connects to the GdpServer.
void Connect(kIpAddress ipAddr, k32u port);

// Closes the TCP connection.
void Close();

// Receives data synchronously from the GDP server.
void ReceiveDataSync(k64u receiveTimeoutInMilliseconds);

// Receives data asynchronously from the GDP server.
void ReceiveDataAsync(std::function<void(const GoDataSet& receivedDataSet)>& func);

// Gets the pointer to the received data set over the Goctor Data Protocol.
const GoDataSet& DataSet() const;
```

Receive Health and Metrics Information

Health information, along with various metrics, is transmitted using the REST API. To handle received metrics data, set up a callback function as shown below:

```
// Callback function for handling received metrics data.
static void onMetrics(const GoStreamResponse&)
{
    // Extract payload & content from the received data
    auto content = notification.Payload();
  int appUptime = content.Get<int>("/appUpTime");
  ...
}

```

```
// Set up the callback function for receiving metrics data asynchronously.
system.Client().SetStreamHandler(onMetrics);

// Start the stream to receive metrics data from the specified path
system.Client().StartStream("/system/metrics");

// Stop the stream and disconnect from receiving metrics data, call StopStream
system.Client().StopStream("/system/metrics");
```

For more information, see the `ReceiveHealth` sample.

Accelerated Devices

Control of devices accelerated by GoMax NX or a PC instance of GoPxL uses the same function API. The only difference is that you must use the IP address of the GoMax NX device or the PC running GoPxL when connecting to the device (that is, in `system->Connect()`).

GDP Data Types

Value Types

The GoPxL SDK is built on a set of basic data structures, utilities, and functions, which are contained in the *kApi* library.

The following basic value types are used by the *kApi* library.

Value Data Types

Type	Description
k8u	8-bit unsigned integer
k16u	16-bit unsigned integer
k16s	16-bit signed integer
k32u	32-bit unsigned integer
k32s	32-bit signed integer
k64s	64-bit signed integer
k64u	64-bit unsigned integer
k64f	64-bit floating number
kBool	Boolean, value can be kTRUE or kFALSE
kStatus	For the definitions of this type, in the folder containing the GoPxL SDK, see Platform\kApi\kApi\kApiDef.h.
kIpAddress	IP address

Output Types

Data outputs are encapsulated as data messages in the GoPxL SDK. The following message types are available in the SDK.

Output Types

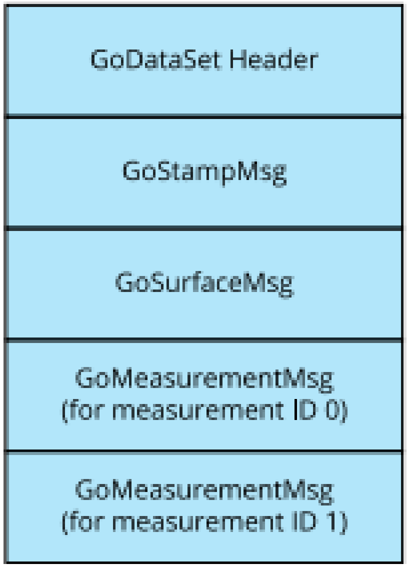
Type	Description
GoGdpBoundingBox	Bounding Box results output based on part matching results
GoGdpFeaturePoint	Feature point output
GoGdpFeaturePlane	Feature plane output
GoGdpFeatureLine	Feature line output
GoGdpFeatureCircle	Feature circle output
GoGdpImage	Video image
GoGdpMeasurement	Measurement output
GoGdpMesh	Mesh data output
GoGdpNull	Contains no valid data. Received when a measurement tool's data output doesn't generate data when triggered. This messages allow user to synchronize message from different outputs based on number of message received.
GoGdpProfilePointCloud	Profile point cloud data (non-uniform spacing)
GoGdpProfileUniform	Uniform spacing / resampled profile data
GoGdpRendering	Graphics objects associated with a tool outputs*
GoGdpSpots	Detected spots when overlay exposure is enabled*
GoGdpStamp	Acquisition stamp
GoGdpSurfacePointCloud	Surface point cloud data (non-uniform spacing)
GoGdpSurfaceUniform	Uniform spacing / resampled surface data

*Not commonly used in a production system.

All of the messages above are extended from `GoGdpMsg`. For more information, see *GoDataSet* below.

GoDataSet

Data is passed to the data handler in a `GoDataSet` object. The `GoDataSet` object is a container that can hold any type of data, including scan data (sections or surfaces), measurements, or results from various operations. Data inside the `GoDataSet` object is represented as messages. The following illustrates the content of a `GoDataSet` object of a Surface mode setup with two measurements.



The following lists the available GoDataSet types:

```
enum struct MessageType : k16u {
    SIGNAL                  = 1,    // Signals that data on a stream is invalidated.
    NULL_TYPE               = 10,   // Null data type that contains a status code.
    STAMP                   = 11,   // Stamp information.
    UNIFORM_PROFILE         = 12,   // Uniform profile data.
    PROFILE_POINT_CLOUD     = 13,   // Raw profile data.
    UNIFORM_SURFACE         = 14,   // Raw surface data.
    SURFACE_POINT_CLOUD     = 15,   // Raw surface data.
    IMAGE                   = 16,   // Image data.
    SPOTS                   = 17,   // Spot data associated with the image.
    MESH                    = 18,   // Mesh data.
    MEASUREMENT             = 19,   // Measurement data.
    RENDERING               = 70,   // Graphical data associated with an output.
    POINT_FEATURE           = 71,   // Point feature data.
    LINE_FEATURE            = 72,   // Line feature data.
    PLANE_FEATURE           = 73,   // Plane feature data.
    CIRCLE_FEATURE          = 74,   // Circle feature data.
    HEALTH                  = 100   // Health data.
};
```

Measurement Values and Decisions

There are two outputs for each measurement: a value and a decision. A measurement value is returned as a 64-bit number.

The measurement decision specifies the state of the measurement value as follows:

Measurement Decisions

Decision	Description
1	The measurement value is between the maximum and minimum decision values (inclusive). This is a pass decision.

Decision	Description
0	The measurement value is outside the maximum and minimum. This is a fail decision.

Limiting Flash Memory Write Operations

Several operations and REST API / SDK functions write to the sensor's flash memory. The lifetime of the flash memory is limited by the number of write cycles. Therefore it is important to avoid frequent write operation to the sensor's flash memory when you design your system with the SDK or GoHMI.

	Power loss during flash memory write operation will also cause sensors to enter rescue mode.

Integrations

GoPxL currently provides three types of integrations: Gocator (for use with the GoPxL SDK and API), industrial protocols, and GoHMI.

For information on the Gocator protocol, see *Gocator Protocol* below.

For information on the protocols supported by GoPxL, see *Protocols (PLCs and other hardware)* below.

For information on GoHMI, see *GoHMI and GoHMI Designer* on page 844.

Protocols (PLCs and other hardware)

GoPxL supports various protocols for communicating with sensors over Ethernet (TCP/IP).

The following protocols are available:

- Gocator
- Modbus
- EtherNet/IP
- PROFINET
- ASCII

Gocator Protocol

When the Gocator Communication Protocol is enabled in the GoPxL interface, you can use the GoPxL SDK and REST API to send commands from and receive data from SDK-based applications.

For information on configuring the Gocator protocol in GoPxL on the **Control** page, see *Gocator Communication Protocol* on page 797.

For information on the GoPxL SDK and the REST API, see *GoPxL SDK and REST API* on page 871.

EtherNet/IP Protocol

EtherNet/IP is an industrial protocol that allows bidirectional data transfer with PLCs that support the protocol. It encapsulates the object-oriented Common Industrial Protocol (CIP). EtherNet/IP communication enables the client to:

- Start and stop sensors
- Load job files
- Align sensors
- Receive sensor states, stamps, and measurement results

> Implicit messaging is always enabled.

This section describes the EtherNet/IP message and data formats. The commands described in the sections below are those specific to the Gocator protocol and not the complete EtherNet/IP reference command set.

GoPxL supports EtherNet/IP communication to provide system, sensor group, and scan output data to a PLC. Sensors also accept incoming commands. After you enable EtherNet/IP in the GoPxL web interface, the sensor starts listening on the standard EtherNet/IP port 44818 for one or more EtherNet/IP clients. GoPxL operates as an EtherNet/IP server, and a PLC that connects to it acts as the EtherNet/IP client. All communication is initiated by the PLC, with GoPxL responding to the EtherNet/IP requests.

While the EtherNet/IP protocol includes several types of requests, the two basic operations are reading and writing EtherNet/IP objects. Commands are invoked by the PLC sending *set* requests. Data is requested by the PLC by sending *get* requests for EtherNet/IP objects.

EtherNet/IP supports both explicit messaging and implicit messaging. Data can be read as Big-Endian format or Little-Endian format based on the configuration selected in web interface.

Connections Map

The EtherNet/IP protocol supports the following objects. Except for mandatory objects, all objects are mapped to a register assembly. Sensors support all mandatory objects for explicit messaging, including assembly objects. Implicit messaging supports only assembly objects.

Mandatory Objects

Mandatory objects are present in the EtherNet/IP service irrespective of the configured measurement objects.

Identity object (Class 0X01)

Attribute	Name	Type	Value	Description	Access
1	Vendor ID	UINT	1256	ODVA-provided vendor ID	Get
2	Device Type	UINT	43	Device type	Get
3	Product Code	UINT	1	Product code	Get

Attribute	Name	Type	Value	Description	Access
4	Revision	Structure of			Get
		USINT	1	Major revision number	
		USINT	3	Minor revision number	
6	Serial number	UDINT	32-bit value	Sensor serial number	Get
7	Product Name	SHORT STRING 32	"Gocator"	Gocator product name	Get

The TCP/IP Object contains read-only network configuration attributes such as IP Address. TCP/IP configuration via EtherNet/IP is not supported. See Volume 2, Chapter 5-3 of the CIP Specification for a complete listing of TCP/IP object attributes.

TCP/IP object (Class 0XF5)

Attribute	Name	Type	Value	Description	Access
1	Status	UDINT		TCP interface status (dynamic)	Get
2	Configuration Capability	UINT		(dynamic)	Get
3	Configuration Control	UINT		Product code (dynamic)	Get
4	Physical Link Object	Structure (See description)		See 5.3.3.2.4 of CIP Specification Volume 2: Path size (UINT) Path (Padded EPATH)	Get
5	Interface Configuration	Structure (See description)		See 5.3.3.2.5 of CIP Specification Volume 2: IP address (UDINT) Network mask (UDINT) Gateway address (UDINT) Name server (UDINT) Secondary name (UDINT) Domain name (UDINT)	Get

The Ethernet Link Object contains read-only attributes such as MAC Address (Attribute 3). See Volume 2, Chapter 5-4 of the CIP Specification for a complete listing of Ethernet Link object attributes.

Ethernet Link Object (Class 0XF6)

Attribute	Name	Type	Value	Description	Access
1	Interface Speed	UDINT	1000	Ethernet interface data rate (mbps)	Get
2	Interface Flags	UDINT		See 5.4.3.2.1 of CIP Specification Volume 2: Bit 0: Link Status 0 – Inactive 1 - Active	Get

Attribute	Name	Type	Value	Description	Access
				Bit 1: Duplex 0 – Half Duplex 1 – Full Duplex	
3	Physical Address	Array of 6 USINTs		MAC address (for example: 00 16 20 00 2E 42)	Get

Assembly Objects

All assembly objects are modeled with class 4 and attribute 3. Only the instance ID changes between assembly objects.

A register assembly contains register blocks. This is a one-to-one mapping, which means each EtherNet/IP assembly object will have a corresponding control framework register assembly. These objects are added to GoPxL irrespective of the selection of register assembly, but the data is added to this EtherNet/IP object based on the selection of register assembly.

Assembly Objects

Object name	Register assembly	Supported Block Type Internal Identifier	Service	Instance ID
Command input object	Control Input register assembly	"ControlInput"	SetAttributeSingle (0x10)	0x70
Command output object	Control Output register assembly	"ControlOutput"	GetAttributeSingle (0x0E)	0x71
Sensor Group State object	Sensor Group State register assembly	"Scanner"	GetAttributeSingle (0x0E)	0x72
System State object	System State register assembly	"System"	GetAttributeSingle (0x0E)	0x73
Sample State object	Sample State register assembly	"MeasurementValue" "Stamp"	GetAttributeSingle (0x0E)	0x74
Implicit Messaging output object	Sample State register assembly	"MeasurementValue" "Stamp"	GetAttributeSingle (0x0E)	0x65
Implicit Messaging input object	Implicit Control Input register assembly	"ControlInput"	SetAttributeSingle (0x10)	0x64

The EtherNet/IP protocol supports various blocks types. Each block type can be either an input block or an output block. Input blocks contain write-only objects and are used by the EtherNet/IP client to send commands such as starting or stopping a scan. Output blocks contain read-only objects and are used to store various types information, such as control, system, and sensor group state data, as well as scan output data.

Supported Block Types

The block types supported by the EtherNet/IP protocol are listed in the table below. The size field indicates the number of registers that each block type requires.

Supported Block Types

Block Type Internal Identifier	Read or Write only?	Size (number of EtherNet/IP registers)	Description
"ControlInput"	WO	67	Command message set by the EtherNet/IP client.
"ControlOutput"	RO	3	Result data corresponding to a command that was written to the Control Input block.
"MeasurementValue"	RO	4	Measurement value data of a scan output measurement data message.
"Scanner"	RO	18	Information about a specific sensor group.
"Stamp"	RO	36	Stamp message information.
"System"	RO	76	Information about the LMI device.

Control Input

The input of the Control Input block contains the fields described below:

Control Input Fields

Field	Type	Address offset	Description
Command Sequence Number	16u	0	The command sequence number is set by the external client to uniquely identify a command request. This number is sent back in the Control Output. This is an optional field but is recommended that Ethernet IP clients use this to help the clients correlate the Command Status in the Control Output with a specific command request in Control Input.
Command Identifier	8u	2	Identifies the requested action from the external client. Supported actions: 0 = No Command 1 = Start Scanning 2 = Stop Scanning 3 = Align Sensor Group 4 = Clear Alignment 5 = Load Job File 6 = Software Trigger
Command Arguments	Array[8u]	3	Arguments for commands that require arguments. For the Load Job File command, the command arguments will contain the null terminated job file name. The job file name must not include the job file extension. The job extension is automatically appended by the Ethernet IP server. This field takes up a maximum of 64 registers. Each byte contains a single character.

Control Output

The output of the Control Output block contains the fields described below:

Control Output Fields

Field	Type	Address offset	Description
Command Sequence Number	16u	0	The command sequence number received from the client in the Control Input.
Command Status	8u	2	Status of the command, referenced by the Command Sequence Number. 0 = No Command Received 1 = In Progress 2 = Success 3 = Fail

System State

The output of the System State block contains the fields described below:

System State Fields

Field	Type	Address offset	Description
System Uptime	64u	0	Amount of time the application has been running, in seconds.
System State	8u	8	Whether system is running or not. 0 = System is stopped 1 = System is running
Current Job File Name Length	8u	9	Number of characters in the current job file name.
Current Job File Name	Array[8u]	10	Name of the job file currently loaded into the system, including extension. Maximum length of job file name is 64 characters. Each register contains a single character. Job file name is NOT null terminated.
Buffer Count	8u	74	The current amount of buffered scan output data when buffering is enabled.
Buffer Overflow	8u	75	A value of 0 indicates that no overflow has occurred. A value of 1 indicates that overflow occurred and data is being lost.

Sensor Group State

The output of the Sensor Group block contains the fields described below:

Sensor Group State Fields

Field	Type	Address offset	Description
Current Encoder	64s	0	The current encoder position.

Field	Type	Address offset	Description
Position			
Current Sync Time	64u	8	Current time of the clock used to stamp all scan output messages, in microseconds.
Alignment State	8u	16	Indicator of whether all devices under the sensor group are aligned (that is, devices in the sensor group all have a non-identity transform matrix). 0 = Not Aligned 1 = Aligned 2 = Aligning
Laser Enabled	8u	17	Whether one or more devices under the sensor group has its laser enabled to be able to turn on. 0 = All devices' lasers are disabled and cannot turn on 1 = One or more devices' lasers can be turned on

Stamp

The output of the Stamp block contains the fields described below:

Stamp Fields

Field	Type	Address offset	Description
Z Position	64s	0	Encoder position at time of last index pulse of last frame.
Encoder Position	64s	8	Encoder position of last frame when the image data was scanned/taken.
Time	64u	16	The timetick of the last frame. A timetick is a 64-bit positive integer that is guaranteed to increase monotonically starting from zero. It is guaranteed to be unique for every scan from a given sensor group. To convert this value to microseconds, use the following: `time in microseconds = timetick / 1.024`
Frame Index	64u	24	The frame number of the last frame.
Digital Input States	8u	32	Digital input states of last frame. Bit 0: State of digital input 1 on master or sensor Bit 1: State of digital input 2 on master or sensor
Buffer	8u	33	Unused
Buffer	8u	34	Unused
Buffer	8u	35	Unused

Measurement Value

The output of the Measurement Value block contains the fields described below.

Note that this block no longer contains a Decision field. This functionality will be added in the future.

Measurement Fields

Field	Type	Address offset	Description
Value	32s	0	Measurement value. This value is the actual measurement value multiplied by 1000. If the measurement value is invalid, the 32-bit value is set to 0x80000000.

PROFINET Protocol

PROFINET is an Industrial Ethernet network protocol that allows controllers such as PLCs to communicate with sensors. Sensors are PROFINET IO devices with Conformance Class A.

GoPxL supports PROFINET communication to provide system state, sensor group state, stamp, and measurement output data to a PLC, as well as accept incoming commands. PROFINET defines the entire data exchange between controllers (called "IO-Controllers") and the devices (called "IO-Devices"). GoPxL operates as a PROFINET IO-Device, and a PLC that connects to it acts as the PROFINET IO-Controller. All communication is initiated by the PLC, with GoPxL responding to the PROFINET requests.

> GoPxL supports PROFINET on-sensor and on GoMax NX. PROFINET is not currently supported on PC instances of GoPxL. This functionality is planned for the near future.

While the PROFINET protocol includes several types of requests, the two basic operations are reading and writing PROFINET objects. Commands are invoked by the PLC sending *set* requests. Data is requested by the PLC by sending *get* requests for PROFINET objects.

Connections Map

The PROFINET protocol supports the following objects.

Assembly Objects

A register assembly is a container that contains register blocks. This is a one-to-one mapping, which means each PROFINET module will have a corresponding control framework register assembly. These modules are added to the GoPxL irrespective of the selection of register assembly, but the data is added to this PROFINET module based on the selection of register assembly.

Assembly Objects

PROFINET Module Name	Register assembly	Supported Register Block Types	Service	Instance ID	Slot Number
Command input module	Control Input register assembly	"ControlInput"	Write	0x30	1
Command output module	Control Output register assembly	"ControlOutput"	Read	0x31	2
System State module	System State register assembly	"System"	Read	0x32	3
Stamp module	Sample State register assembly	"Stamp"	Read	0x33	4
Measurement module	Sample State register assembly	"MeasurementValue"	Read	0x34	5
Sensor Group State module	Sensor Group State register assembly	"Scanner"	Read	0x35	6

The PROFINET protocol supports various block types. Each block type can be either an input block or an output block. Input blocks contain write-only objects and are used by the PROFINET IO-Controller to send commands such as starting or stopping a scan. Output blocks contain read-only objects and are used to store various types information, such as control, system, and sensor group state data as well as scan output data (stamp and measurements).

Supported Block Types

The block types supported by the PROFINET protocol are listed in the table below. The size field indicates the number of registers that each block type requires.

Supported Block Types

Block Type	Read or Write only?	Size (number of PROFINET registers)	Description
"ControlInput"	WO	67	Command message set by the PROFINET IO-Controller.
"ControlOutput"	RO	3	Result data corresponding to a command that was written to the Control Input block.
"MeasurementValue"	RO	4	Measurement data of a scan output measurement data message.
"Scanner"	RO	18	Information about a specific sensor group.
"Stamp"	RO	36	Stamp message information.
"System"	RO	76	Information about the LMI device.

Control Input

The input of the Control Input Block contains the fields described below:

Control Input Fields

Field	Type	Address offset	Description
Command Sequence Number	16u	0	The command sequence number is set by the external IO-Controller to uniquely identify a command request. This number is sent back in the Control Output. This is an optional field but is recommended that PROFINET IO-Controllers use this to help the IO-Controllers correlate the Command Status in the Control Output with a specific command request in Control Input.
Command Identifier	8u	2	Identifies the requested action from the external client. Supported actions: 0 = No Command 1 = Start Scanning 2 = Stop Scanning 3 = Align Sensor Group 4 = Clear Alignment 5 = Load Job File 6 = Software Trigger
Command Arguments	Array[8u]	3	Arguments for commands that require arguments.

Field	Type	Address offset	Description
			For the Load Job File command, the command arguments will contain the null terminated job file name. The job file name must not include the job file extension. The job extension is automatically appended by the PROFINET IO-Device. This field takes up a maximum of 64 registers. Each byte contains a single character.

Control Output

The output of the Control Output Block contains the fields described below:

Control Output Fields

Field	Type	Address offset	Description
Command Sequence Number	16u	0	The command sequence number received from the IO-Controller in the Control Input.
Command Status	8u	2	Status of the command, referenced by the Command Sequence Number. 0 = No Command Received 1 = In Progress 2 = Success 3 = Fail

System State

The output of the System State Block contains the fields described below:

System State Fields

Field	Type	Address offset	Description
System Uptime	64u	0	Amount of time the application has been running, in seconds.
System State	8u	8	Whether system is running or not. 0 = System is stopped 1 = System is running
Current Job File Name Length	8u	9	Number of characters in the current job file name.
Current Job File Name	Array[8u]	10	Name of the job file currently loaded into the system, including extension. Maximum length of job file name is 64 characters. Each register contains a single character. Job file name is NOT null terminated.
Buffer Count	8u	74	The current amount of buffered scan output data when buffering is enabled.
Buffer Overflow	8u	75	A value of 0 indicates that no overflow has occurred.

Field	Type	Address offset	Description
			A value of 1 indicates that overflow occurred and data is being lost.

Sensor Group State

The output of the Scanner (Sensor Group) Block contains the fields described below:

Sensor Group State Fields

Field	Type	Address offset	Description
Current Encoder Position	64s	0	The current encoder position.
Current Sync Time	64u	8	Current time of the clock used to stamp all scan output messages, in microseconds.
Alignment State	8u	16	Indicates whether all devices under the sensor group are aligned (that is, all devices in the sensor group have a non-identity transform matrix). 0 = Not Aligned 1 = Aligned 2 = Aligning
Laser Enabled	8u	17	Whether one or more devices under a sensor group has its laser enabled to be able to turn on. 0 = All devices' lasers are disabled and cannot turn on 1 = One or more devices' lasers can be turned on

Stamp

The output of the Stamp Block contains the fields described below:

Stamp Fields

Field	Type	Address offset	Description
Z Position	64s	0	Encoder position at time of last index pulse of last frame.
Encoder Position	64s	8	Encoder position of last frame when the image data was scanned/taken.
Time	64u	16	The timetick of the last frame. A timetick is a 64-bit positive integer that is guaranteed to increase monotonically starting from zero. It is guaranteed to be unique for every scan from a given sensor group. To convert this value to microseconds, use the following: `time in microseconds = timetick / 1.024`
Frame Index	64u	24	The frame number of the last frame.
Digital Input States	8u	32	Digital input states of last frame. Bit 0: State of digital input 1 on master or sensor

Field	Type	Address offset	Description
			Bit 1: State of digital input 2 on master or sensor
Buffer	8u	33	Unused
Buffer	8u	34	Unused
Buffer	8u	35	Unused

Measurement

The output of the Measurement Block contains the fields described below:

Measurement Fields

Field	Type	Address offset	Description
Value	32s	0	Measurement value. This value is the actual measurement value multiplied by 1000. If the measurement value is invalid, the 32-bit value is set to 0x80000000.

Modbus Protocol

GoPxL supports the Modbus TCP communication to provide system, sensor group, and scan output data to a PLC, as well as accept incoming commands. After enabling Modbus in the GoPxL web interface (for more information, see *Modbus Protocol* on page 799), the sensor starts listening on port 502 for one or more Modbus clients. GoPxL operates as a Modbus server (slave), and a PLC that connects to it acts as the Modbus client (master). All communication is initiated by the PLC, with GoPxL responding to the Modbus requests.

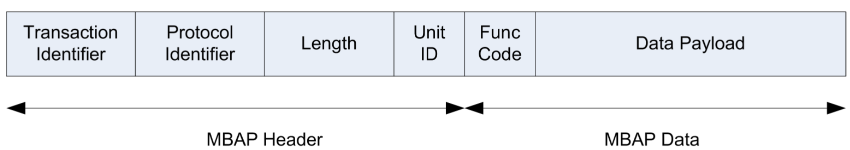
While the Modbus protocol includes several types of requests, the two basic operations are reading and writing registers. Commands are invoked by the PLC sending write-register requests. Data is requested by the PLC by sending read-register requests.

There are 16-bits per register in the Modbus protocol. 32-bit and 64-bit values are stored in consecutive registers in Big-Endian format.

If buffering is enabled with the Modbus protocol (when part detection is used and if multiple objects may be detected within a time frame shorter than the polling rate of the PLC), the PLC must send the Buffer Advance command to advance the queue before reading the measurement results (see *Control Input* on page 899). For more information on part detection, see *Profile Part Detection* on page 373.

Messages

All Modbus TCP messages consist of an MBAP header (Modbus Application Protocol), a function code, and a data payload.



The MBAP header contains the following fields:

Modbus Application Protocol Header

Field	Length (Bytes)	Description
Transaction ID	2	Used for transaction pairing. The Modbus Client sets the value and the Server (the sensor) copies the value into its responses.
Protocol ID	2	Always set to 0.
Length	2	Byte count of the rest of the message, including the Unit identifier and data fields.
Unit ID	1	Used for intra-system routing purpose. The Modbus Client sets the value and the Server (the sensor) copies the value into its responses.

Modbus Application Protocol Specification describes the standard function codes in detail. Gocator supports the following function codes:

Modbus Function Code

Function Code	Name	Data Size (bits)	Description
3	Read Holding Registers	16	Read multiple data values from the sensor.
4	Read Input Registers	16	Read multiple data values from the sensor.
6	Write Single Register	16	Send a command or parameter to the sensor.
16	Write Multiple Registers	16	Send a command and parameters to the sensor.

The data payload contains the registers that can be accessed by Modbus TCP messages. If a message accesses registers that are invalid, a reply with an exception is returned. Modbus Application Protocol Specification defines the exceptions and describes the data payload format for each function code.

The sensor data includes 16-bit, 32-bit, and 64-bit data. All data are sent in big endian format, with the 32-bit and 64-bit data spread out into two and four consecutive registers.

32-bit Data Format

Register	Name	Bit Position
0	32-bit Word 1	31 .. 16
1	32-bit Word 0	15 .. 0

64-bit Data Format

Register	Name	Bit Position
0	64-bit Word 3	63 .. 48
1	64-bit Word 2	47 .. 32
2	64-bit Word 1	31 .. 16
3	64-bit Word 0	15 .. 0

Connections Map

The Modbus protocol supports various blocks types. Each block type can be either an input block or an output block. Input blocks contain write-only registers and are used by the Modbus client to send commands such as starting or stopping a scan. Output blocks contain read-only registers and are used to store various types information, such as control, system, and sensor group state data as well as scan output data.

The starting address location of each block within the Modbus register map may be assigned using the GoPxL web interface.

The register address of a specific field within a block is equal to the block's address plus the field's address offset. For example, the command identifier in the Control Input block has an address offset equal to 1. If the Control Input block is located at register address 100, the command identifier value is located at register address 101.

Supported Block Types

The block types supported by the Modbus protocol are listed in the table below. The size field indicates the number of registers that each block type requires.

Supported Block Types

Supported Block Type	Read or Write only?	Size (number of Modbus registers)	Description
"ControlInput"	WO	66	Command message set by the Modbus client.
"ControlOutput"	RO	2	Result data corresponding to a command that was written to the Control Input block.
"Measurement"	RO	3	Measurement data of a scan output measurement data message.
"Scanner"	RO	10	Information about a specific sensor group.
"Stamp"	RO	17	Stamp message information.
"System"	RO	72	Information about the LMI device.

Control Input

The input of the Control Input block contains the fields described below:

Control Input Fields

Field	Type	Address offset	Description
Command Sequence Number	16u	0	The command sequence number is set by the external client to uniquely identify a command request. This number is sent back in the Control Output. This is an optional field but is recommended that Modbus clients use this to help the clients correlate the Command Status in the Control Output with a specific command request in Control Input.
Command Identifier	16u	1	Identifies the requested action from the external client. Supported actions: 0 = No Command 1 = Start Scanning 2 = Stop Scanning 3 = Align Sensor Group 4 = Clear Alignment 5 = Load Job File 6 = Software Trigger 7 = Buffer Advance
Command Arguments	16u	2	Arguments for commands that require arguments. For the Load Job File command, the command arguments will contain the *NULL terminated* job file name. The job file name must *not* include the job file extension. The job

Field	Type	Address offset	Description
			extension is automatically appended by the Modbus server. This field takes up a maximum of 64 registers.
Reserved	16u	66	Not used. This field takes up 57 registers.

Control Output

The output of the Control Output block contains the fields described below:

Control Output Fields

Field	Type	Address offset	Description
Command Sequence Number	16u	0	The command sequence number received from the client in the Control Input.
Command Status	16u	1	Status of the command, referenced by the Command Sequence Number. 0 = No Command Received 1 = In Progress 2 = Success 3 = Fail This field takes up 1 register.

System State

The output of the System State block contains the fields described below:

System State Fields

Field	Type	Address offset	Description
System Uptime	64u	0	Amount of time the application has been running, in seconds.
System State	16u	4	Whether system is running or not. 0 = System is stopped 1 = System is running
Current Job File Name Length	16u	5	Number of characters in the current job file name.
Current Job File Name	16u	6	Name of the job file currently loaded into the system, including extension. Maximum length of job file name is 64 characters. Each register contains a single character. Job file name is *not* null terminated.
Buffer Count	16u	70	The current amount of buffered scan output data when buffering is enabled.
Buffer Overflow	16u	71	A value of 0 indicates that no overflow has occurred. A value of 1 indicates that overflow occurred and data is being lost.

Field	Type	Address offset	Description
			This field takes up 1 register.

Sensor Group State

The output of the Sensor Group State block contains the fields described below:

Sensor Group State Fields

Field	Type	Address offset	Description
Current Encoder Position	64s	0	The current encoder position.
Current Sync Time	64u	4	Current time of the clock used to stamp all scan output messages, in microseconds.
Alignment State	16u	8	Indicates whether all devices under the sensor group are aligned (that is, devices in the sensor group all have a non-identity transform matrix). 0 = Not Aligned 1 = Aligned 2 = Aligning
Laser Enabled	16u	9	Whether one or more devices under a scanner has its laser enabled to be able to turn on. 0 = All devices' lasers are disabled and cannot turn on 1 = One or more devices' lasers can be turned on This field takes up 1 register.

Stamp

The output of the Stamp block contains the fields described below:

Stamp Fields

Field	Type	Address offset	Description
Z Position	64s	0	Encoder position at time of last index pulse of last frame.
Encoder Position	64s	4	Encoder position of last frame when the image data was scanned/taken.
Time	64u	8	The timetick of the last frame. A timetick is a 64-bit positive integer that is guaranteed to increase monotonically starting from zero. It is guaranteed to be unique for every scan from a given sensor group. To convert this value to microseconds, use the following: `time in microseconds = timetick / 1.024`
Frame Index	64u	12	The frame number of the last frame. This field takes up 4 registers.
Digital Input States	16u	16	Digital input states of last frame. Bit 0: State of digital input 1 on master or sensor

Field	Type	Address offset	Description
			Bit 1: State of digital input 2 on master or sensor

Measurement

The output of the Measurement Block contains the fields described below:

Measurement Fields

Field	Type	Address offset	Description
Value	32s	0	Measurement value. This value is the actual measurement value multiplied by 1000. If the measurement value is invalid, the first register of the measurement value is set to 0x80000000. This means the first register of the measurement value is set to 0x8000 and the second register value is 0x0000.
Decision	16u	2	Measurement decision. Bit 0: 0 = Fail 1 = Pass Bit 1: 0 = Decision value is valid 1 = Decision value is invalid This field takes up 1 register.

Ethernet ASCII Protocol

You can use a programmable logic controller (PLC) that supports the Ethernet ASCII protocol to operate a sensor and receive values and decisions. To use the ASCII protocol, it must be enabled and configured in the active job. By default, the control port used to connect to the sensor is 8190. For information on enabling Ethernet ASCII, as well as changing the control port and other protocol configurations in GoPxL, see *Ethernet ASCII Protocol* on page 808.

Two modes are available: asynchronous and polling.

When the sensor is in asynchronous mode, results are sent automatically, as soon as they are available. No commands are required, but the client can still send a command to the sensor to receive output data or to control the device.

When the sensor is in polling mode, you use commands to operate the sensor and to retrieve results. Ethernet ASCII protocol in polling mode only supports a subset of the tasks that can be accomplished in the web interface, and only measurement results can be transmitted to the PLC (scan data is not sent).

This section describes the following aspects of the Ethernet ASCII protocol:

- the polling operation commands you can use to control a sensor and retrieve results from it
- result formats

The protocol communicates using ASCII strings. The output result format from the sensor is user-configurable.

Commands and Formats

Commands are sent from the client to the sensor. The commands are not case sensitive. However, the `readprop` command's metrics path argument *is* case sensitive.

The command format is:

<COMMAND><DELIMITER><PARAMETER><TERMINATION>

If a command has more than one parameter, each parameter is separated by the delimiter.

Similarly, the reply has the following format:

<STATUS><DELIMITER><OPTIONAL RESULTS><DELIMITER>

The status can either be "OK" or "ERROR". The optional results can be relevant data for the command if successful, or a text based error message if the operation failed. If there is more than one data item, each item is separated by the delimiter.

You can change the delimiter and termination characters in the **Special Character** settings in the GoPxL interface. For more information, see *Ethernet ASCII Protocol* on page 808.

> Measurement IDs come from the **ID** field in the connection map. Be sure to add measurements to the connection map and make note of the IDs of the measurements you need to add to the data format.

Ethernet ASCII client commands

Ethernet ASCII client commands

Command	Arguments	Description
start	-	Starts the sensor.
stop	-	Stops the sensor.
trigger	-	Software Trigger.
loadjob	Job name	Load the specified job file.
result	-	Return the measurement data specified in the custom format string. For a description of the measurement value (prefixed with the letter "V") and decision (prefixed with the letter "D") format, see *Measurement Output Format* on page 909.
	One or more measurement ids, separated by command delimiter.	Return the measurement data for each measurement id in the following format string: M[ID],V[VALUE],D[DECISION],... Example: result,0,1 Output: OK,M0,V-5000,D0,M1,V5000,D1
measurement	One or more	Return the measurement data for each measurement id in the

Command	Arguments	Description
	measurement ids, separated by command delimiter.	following format string: M[ID],V[VALUE],D[DECISION],... For a description of the measurement value (prefixed with the letter "V") and decision (prefixed with the letter "D") format, see *Measurement Output Format* on page 909. Example: measurement,0,1 Output: OK,M0,V-5000,D0,M1,V5000,D1
value	One or more measurement ids, separated by command delimiter.	Return the measurement value for each measurement id in the following format string: M[ID],V[VALUE],... For a description of the measurement value (prefixed with the letter "V") format, see *Measurement Output Format* on page 909. Example: value,0,1 Output: OK,M0,V-5000,M1,V5000
decision	One or more measurement ids, separated by command delimiter.	Return the measurement decision for each measurement id in the following format string: M[ID],D[DECISION],... For a description of the measurement decision (prefixed with the letter "D") format, see *Measurement Output Format* on page 909. Example: decision,0,1 Output: OK,M0,D0,M1,D1
stamp	-	Return the time, encoder and frame index stamp values for the first stamp id found. Order of values returned is: Time, Encoder, Frame. For the "time" argument, this is the timetick of the last frame. A timetick is a 64-bit positive integer that is guaranteed to increase monotonically starting from zero. It is guaranteed to be unique for every scan from a given sensor group. To convert this value to microseconds, use the following: `time in microseconds = timetick / 1.024` Example: stamp Output: OK,Time,381497381349,Encoder,0,Frame,18,
	One or more of these arguments in any order, separated by the command delimiter: "time" "encoder" "frame"	Return the specified stamp value(s) in the order of the provided arguments. Example: stamp,time,frame Output: OK,381497381349,18

Command	Arguments	Description
	A stamp ID.	Return the time, encoder, and frame index stamp values for the passed stamp ID. Example: stamp,1
time	One or more stamp ids, separated by command delimiter.	Return the time stamp value for each stamp id. The timetick of the last frame. A timetick is a 64-bit positive integer that is guaranteed to increase monotonically starting from zero. It is guaranteed to be unique for every scan from a given sensor group. To convert this value to microseconds, use the following: `time in microseconds = timetick / 1.024` Example: time,1 Output: OK,381497381349
encoder	One or more stamp ids, separated by command delimiter.	Return the encoder stamp value for each stamp id. Example: encoder,1 Output: OK,0
frame	One or more stamp ids, separated by command delimiter.	Return the frame index stamp value for each stamp id. Example: frame,1 Output: OK,18
align	-	Performs the alignment operation.
clearalign	-	Clears sensor alignment.
readprop	One or more string representing a resource whose property to return, separated by command delimiter.	Return the resource JSON object specified by the resource path. It is possible to use JSON pointer to retrieve a specific value within the JSON document. Example #1: readprop,/system Output: OK,{"_links":{"self":{"href":"/system"}},"_schema":{"properties":{"autoLiveSimEnabled":{"sortOrder":200,"title":"autoLiveSimEnabled","type":"boolean"},"autoStart":{"sortOrder":100,"title":"autoStart","type":"boolean"},"runState":{"enum":[0,1,2],"enumText":["Ready","Running","Conflict"],"readOnly":true,"sortOrder":0,"title":"runState","type":"integer"}},"sortOrder":0,"title":"","type":"object"},"autoLiveSimEnabled":true,"autoStart":true,"runState":1} Example #2: readprop,/system#/runState,/system#/autoStart Output: OK,1, true

The following lists the possible errors for each command.

Ethernet ASCII error handling

Command	Arguments	Description
start	ERROR,Could not start the sensor	
stop	ERROR,Could not stop the sensor	
trigger	ERROR,Could not trigger	
loadjob	ERROR,Job name required.	Job name has to be provided as a command argument.
	ERROR,Failed to load job <job name>.job	
result	ERROR,There is no measurement data to output. Please confirm that the sensor is running	Could not find any measurements. Make sure protocol's connection map is not empty and the sensor is running.
	ERROR,Invalid parameter. Please verify your input	Provided command argument was invalid. It must be a number.
	ERROR,Specified measurement ID not found. Please verify your input	Could not find a measurement data for specified ID.
	ERROR,The custom format string is not valid. Please verify your input	Custom format string is invalid.
measurement	ERROR,One or more measurement id must be provided	Measurement ID has to be provided as an argument.
	ERROR,There is no measurement data to output. Please confirm that the sensor is running	Could not find any measurements. Make sure protocol's connection map is not empty and the sensor is running.
	ERROR,Invalid parameter. Please verify your input	Provided command argument was invalid. It must be a number.
	ERROR,Specified measurement ID not found. Please verify	Could not find a measurement data for specified ID.

Command	Arguments	Description
	your input	
value	ERROR,One or more measurement id must be provided	Measurement ID must be provided as an argument.
	ERROR,Invalid parameter. Please verify your input	Provided command argument was invalid. It must be a number.
	ERROR,Measurement with ID not found.	Could not find a measurement data for specified ID.
decision	ERROR,One or more measurement id must be provided	Measurement ID must be provided as an argument.
	ERROR,Invalid parameter. Please verify your input	Provided command argument was invalid. It must be a number.
	ERROR,Measurement with ID not found.	Could not find a measurement data for specified ID.
stamp	ERROR,Invalid stamp command format	Invalid command arguments.
	ERROR,One or more stamp id must be provided	Stamp ID must be provided as an argument.
	ERROR,Invalid parameter. Please verify your input	Provided stamp ID was invalid. It must be a number.
	ERROR,Stamp with ID not found	Could not find a stamp data for specified ID.
time	ERROR,One or more stamp id must be provided	Stamp ID must be provided as an argument.
	ERROR,Invalid parameter. Please verify your input	Provided stamp ID was invalid. It must be a number.
	ERROR,Stamp with ID not found	Could not find a measurement data for specified ID.
encoder	ERROR,One or more stamp id must be provided	Stamp ID must be provided as an argument.
	ERROR,Invalid parameter. Please	Provided stamp ID was invalid. It must be a number.

Command	Arguments	Description
	verify your input	
	ERROR,Stamp with ID not found	Could not find a measurement data for specified ID.
frame	ERROR,One or more stamp id must be provided	Stamp ID must be provided as an argument.
	ERROR,Invalid parameter. Please verify your input	Provided stamp ID was invalid. It must be a number.
	ERROR,Stamp with ID not found	Could not find a measurement data for specified ID.
align	ERROR,Could not align	
clearalign	ERROR,Could not clear alignment	
readprop	ERROR,String representing a resource must be provided	Resource path must be provided as an argument.
	ERROR,Could not read property	Resource may not exist.

Data Output Format Mode

Polling Mode

Polled data output is displayed using the custom format. You can modify this format in the GoPxL interface. For more information, see *Ethernet ASCII Protocol* on page 808. Because polling mode is always available, even when asynchronous output is enabled, you can always modify the custom format.

Format Mode	Description
Custom	This format can be changed/edited by the user. The default format is: %time, %value[0] %decision[0]

Asynchronous Mode

When asynchronous output mode is enabled, the data output can be in one of three formats:

Format Mode	Description
Standard	This is a pre-defined format that cannot be changed: M[id],V%value[id],D%decision[id],...
Standard with Stamp	The format is the Standard format pre-pended with the time and encoder information.

Format Mode	Description
	This is a pre-defined format that cannot be changed: T%time,E%encoder,M[id],V%value[id],D%decision[id],..
Custom	This format is the same as the format used by polling mode. You can change this in the GoPxL interface. For more information, see *Ethernet ASCII Protocol* on page 808. Default format: %time, %value[0] %decision[0]

Measurement Output Format

A measurement contains the fields described below.

Field	Type	Address offset	Description
Value	32s	0	Measurement value. This value is the actual measurement value multiplied by 1000. If the measurement value is invalid, the string "INVALID" is the output for the measurement value.
Decision	8u	4	Measurement decision. Bit 0: 0 = Fail 1 = Pass Bit 1: 0 = Decision value is valid 1 = Decision value is invalid

GenICam GenTL Driver

For known issues when using GoPxL with GenTL, see the GoPxL release notes.

GenICam is an industry standard for controlling and acquiring data from an imaging device. Gocator sensors support GenICam through a GenTL Producer driver.

The included GenTL driver allows GenICam-compliant third-party software applications such as Halcon and Common Vision Blox to acquire and process 3D data and intensity generated from the sensor.

Currently, the GenTL driver can't perform an alignment if you haven't saved a job first.

The following sensor scan modes are supported:

- Video
- Profile (with **Uniform Spacing** disabled). In this mode, the point cloud profiles are resampled and accumulated into a surface.
- Surface (with **Uniform Spacing** enabled)

For more information on scan modes and uniform spacing, see *Scan Modes and Intensity* on page 195.

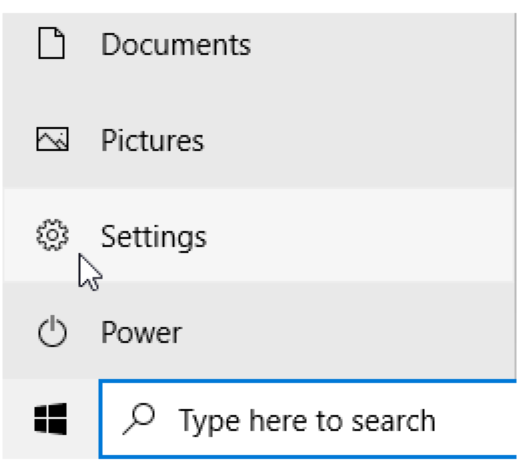
To use these third-party software applications, you *must* configure a system variable so the software can access the GenTL driver. For instructions, see *To configure system variables to use the driver in Windows 10*, below.

To get the utilities package containing the driver (14631-x.x.x.x_SOFTWARE_GoPxL_Utilities.zip), go to https://lmi3d.com/product-downloads/, choose your product using the drop-down menus, and download the zip package.

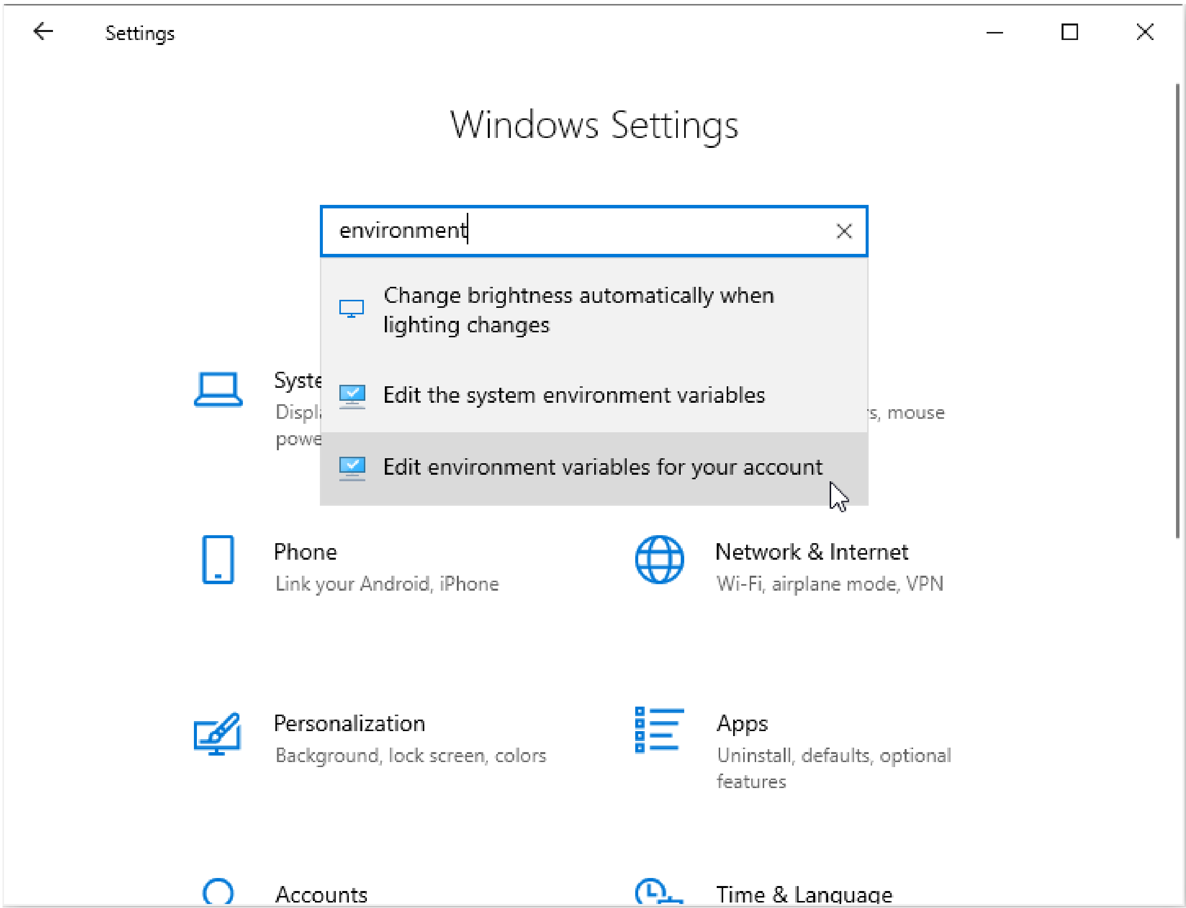
After downloading the package and unzipping the file to a location you will remember, you will find the driver in Integration\GenTL\x64 (you can move the GenTL folder to a more convenient location).

To configure system variables to use the driver in Windows 10

1. Click the Start menu and launch Settings.

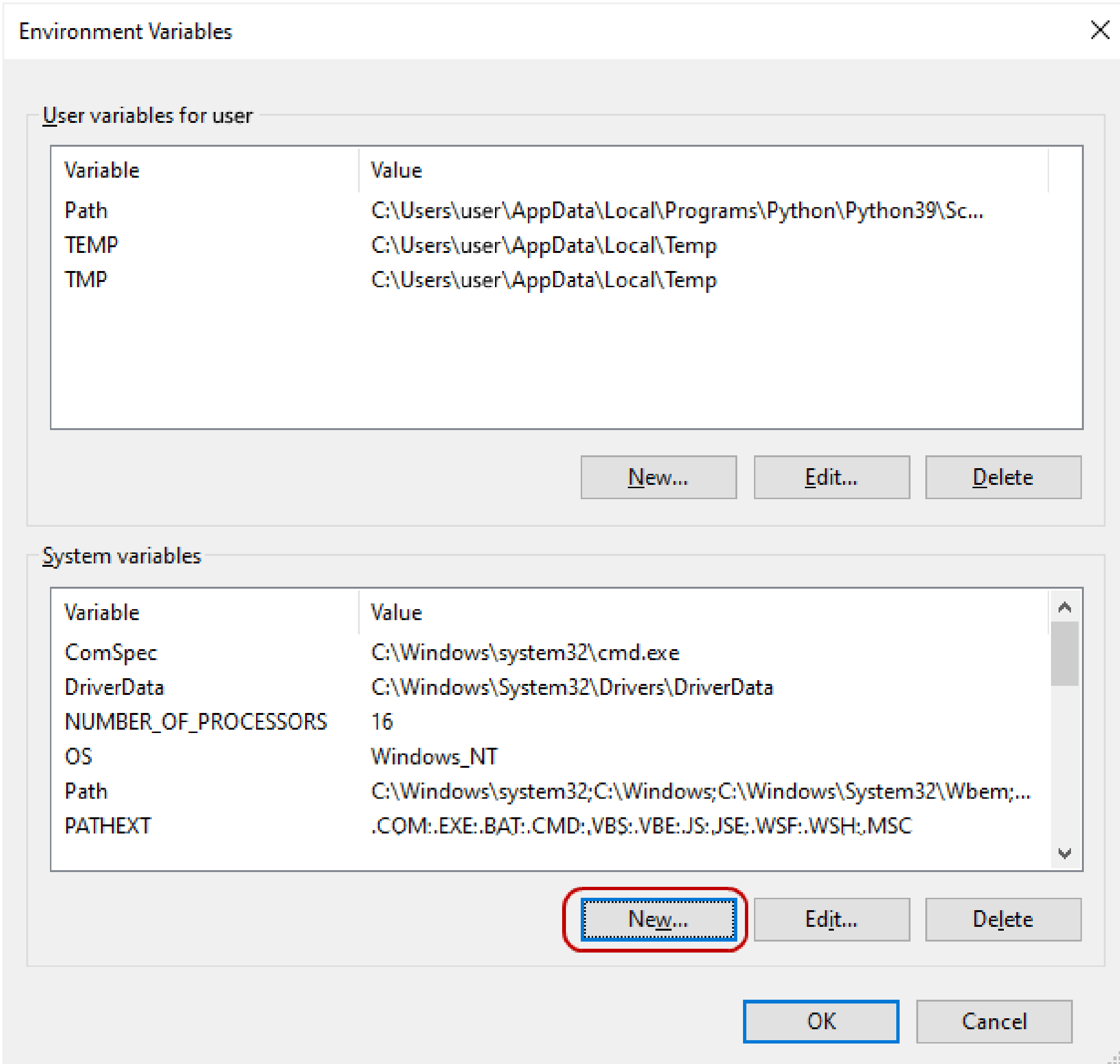


2. In the Settings application search field, type "environment".



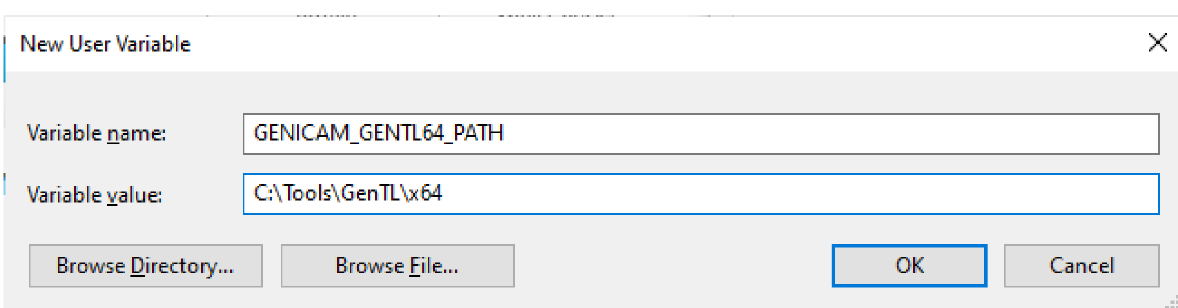
The Environment Variables dialog opens.

3. In the Environment Variables dialog, under the System variables list, click **New**.



4. In the **New System Variable** dialog, enter the following information, depending on your system:

Variable name	Variable value
GENICAM_GENTL64_PATH	The full path to the GenTL\x64 folder.



5. Click OK in the dialogs until they are all closed.

To work with the GenTL driver, you must enable the Gocator protocol and the desired outputs on the **Gocator** page. For more information on configuring the Gocator protocol, see *Gocator Communication Protocol* on page 797.

If you need intensity data, check **Acquire Intensity** in the **Scan Mode** panel on the **Scan** page and enable the intensity output on the **Gocator** page. For more information on enabling intensity, see *Scan Modes and Intensity* on page 195.

The GenTL driver packs the output, intensity, and stamps (e.g., time stamp, encoder index, etc.) into either a 16-bit RGB image or a 16-bit grey scale image. You can select the format in the Go2GenTL.xml setting file.

The width and height of the 16-bit RGB or grey scale image is calculated from the maximum number of columns and rows needed to accommodate the sensor's field of view and the maximum part length.

16-bit RGB Image

When the 16-bit RGB format is used, the height map, intensity, and stamps are stored in the red, green, and blue channel respectively.

Channel	Details
Red	Height map information. The width and height of the image represent the dimensions in the X and Y axis. Together with the pixel value, each red pixel presents a 3D point in the real-world coordinates. The following formula can be used to calculate the real-world coordinates (X, Y, Z) from pixel coordinates (Px, Py, Pz): X = X offset + Px * X resolution Y = Y offset + Py * Y resolution Z = Z offset + Pz * Z resolution Refer to the blue channel on how to retrieve the offset and resolution values. If Pz is 0 if the data is invalid. The Z offset is fixed to -32768 * Z resolution. Z is zero if Pz is 32768.
Green	Intensity information. Same as the red channel, the width and height of the image represent the dimension in the X and the Y axis. Together with the pixel value, each blue pixel represents an intensity value in the real-world coordinates. The following formula can be used to calculate the real-world coordinates (X, Y, Z) from pixel coordinates (Px, Py, Pz): X = X offset + Px * X resolution Y = Y offset + Py * Y resolution Z = 16-bit intensity value The intensity value is 0 if the intensity image is not available. Gocator outputs 8-bit intensity values. The values stored in the 16-bit RGB image is multiplied by 256. To obtain the original values, divide the intensity values by 256.

Channel	Details
	Refer to the blue channel on how to retrieve the offset and resolution values.
Blue	Stamp information. Stamps are 64-bit auxiliary information related to the height map and intensity content. The next table explains how the stamps are packed into the blue pixel channel

The following table shows how the stamp information is packed into the blue channel. A stamp is a 64-bit value packed into four consecutive 16-bit blue pixels, with the first byte position storing the most significant byte.

Stamp Information from GenTL driver

Stamp Index	Blue Pixel Position	Details
0	0..3	Version
1	4..7	Frame Count
2	8..11	Timestamp (μs)
3	12..15	Encoder value (ticks)
4	16..19	Encoder index (ticks) This is the encoder value when the last index is triggered
5	20..23	Digital input states
6	24..27	X offset (nm)
7	28..31	X resolution(nm)
8	32..35	Y offset (nm)
9	36..39	Y resolution (nm)
10	40..43	Z offset (nm)
11	44..47	Z resolution (nm)
12	48..51	Height map Width (in pixels)
13	52..55	Height map length (in pixels)
14	56..59	Specify if the intensity is enabled

16-bit Grey Scale Image

When the 16-bit grey scale format is used, the height map, intensity, and stamps are stored sequentially in the grey scale image.

The last row of the image contains the stamp information.

Rows	Details
0 .. (max part height - 1)	Height map information. The width and height of the image represent the dimensions in the X and Y axis. Together with the pixel value, each pixel presents a 3D point in the real-world coordinates. The following formula can be used to calculate the real-world coordinates (X, Y, Z) from pixel coordinates (Px, Py, Pz):

Rows	Details
	X = X offset + Px * X resolution Y = Y offset + Py * Y resolution Z = Z offset + Pz * Z resolution Refer to the blue channel on how to retrieve the offset and resolution values. If Pz is 0 if the data is invalid. The Z offset is fixed to -32768 * Z Resolution. Z is zero if Pz is 32768.
(max part height) .. 2* (max part height) If intensity is enabled	Intensity information. The width and height of the image represent the dimension in the X and the Y axis. Together with the pixel value, each blue pixel represents an intensity value in the real-world coordinates. The following formula can be used to calculate the real-world coordinates (X, Y, Z) from pixel coordinates (Px, Py, Pz): The following formula assumes Py is relative to the first row of the intensity information, not the first row of the whole 16-bit grey scale image. X = X offset + Px * X resolution Y = Y offset + Py * Y resolution Z = 16-bit intensity value This intensity value is 0 if the intensity image is not available. Gocator outputs 8-bit intensity values. The values stored in the 16-bit Grey scale image is multiplied by 256. To obtain the original values, divide the intensity values by 256. Refer to the stamps on how to retrieve the offset and resolution values.
The last row of the 16-bit grey scale image	Stamp information. Stamps are 64-bit auxiliary information related to the height map and intensity content. The next table explains how the stamps are packed into the blue pixel channel

The following table shows how the stamp information is packed into the last row. A stamp is a 64-bit value packed into four consecutive 16-bit pixels, with the first byte position storing the most significant byte.

Stamp Information from GenTL driver

Stamp Index	Column Position	Details
0	0..3	Version
1	4..7	Frame Count
2	8..11	Timestamp (µs)
3	12..15	Encoder value (ticks)
4	16..19	Encoder index (ticks) This is the encoder value when the last index is triggered
5	20..23	Digital input states
6	24..27	X offset (nm)
7	28..31	X resolution(nm)

Stamp Index	Column Position	Details
8	32..35	Y offset (nm)
9	36..39	Y resolution (nm)
10	40..43	Z offset (nm)
11	44..47	Z resolution (nm)
12	48..51	Height map Width (in pixels)
13	52..55	Height map length (in pixels)
14	56..59	Specify if intensity is enabled or not

Registers

GenTL registers are multiples of 32 bits. The registers are used to control the operation of the GenTL driver, send commands to the sensors, or to report the current sensor information.

Register Map Overview

Register Address	Name	Read/Write	Length (bytes)	Description
260	WidthReg	RO	4	Specify the width of the returned images. The part height map is truncated if it is wider than the specified width.
264	HeightReg	RO	4	Specify the height of the returned images (i.e., length of the part). The part height map is truncated if it is longer than the specified length.
292	ResampleMode	RO	4	Enable the resampling logic in the GenTL driver 0 – Disable resampling 1 – Enable resampling When resampling is enabled, the GenTL driver will resample the height map so that the pixel spacing is the same in the X and Y axis.
296	EncoderValue0	RO	4	Report the current encoder value (least significant 32-bit). The current encoder value is latched from the sensor when this register is read.
300	EncoderValue1	RO	4	Report the current encoder value (most significant 32-bit). The encoder value is latched when EncoderValue0 register is read. User should read EncoderValue0 before reading EncoderValue1.
304	Configuration File	RW	16	Read the name of sensor live configuration file or switch (write) the sensor configuration file. The configuration name is NULL terminated and includes the extension ".job". Writing to this

Register Address	Name	Read/Write	Length (bytes)	Description
				register causes the sensor to switch to the specified configuration.
320	Transformation X offset	RO	4	Return the sensor transformation X offset
324	Transformation Z offset	RO	4	Return the sensor transformation Z offset
328	Transformation Angle	RO	4	Return the sensor transformation angle
332	Transformation Orientation	RO	4	Return the sensor transformation orientation
336	Clearance distance	RO	4	Return the sensor clearance distance

XML Settings File

The settings file, Go2GenTL.xml, resides in the same directory as the Gocator GenTL driver. Users can set the resample mode and output format by changing the setting in this file.

Element	Type	Description
ResampleMode	32u	Settings to disable or enable resampling mode: 0 – Disable 1 – Enable When resampling mode is enabled, the GenTL driver will resample the height map so that the pixel spacing is the same in the X and Y axis. The default value is 1.
DataFormat	32u	Settings to choose 16-bit RGB or 16-bit grey scale image output: 0 – 16-bit RGB Image 1 – 16-bit grey scale Image The default value is 0.
OverrideImageWidth OverrideImageLength	32u	If Halcon is consuming the output of a multi-sensor system or of a tool such as Surface Stitch, that is, where multiple frames are combined, you should set OverrideImageWidth and OverrideImageLength to the maximum expected size. OverrideImageWidth = image width / X resolution OverrideImageLength = image length / Y resolution Otherwise, leave the default of 0.
RawPartDetectionHeightThreshold RawPartDetectionMaxLength	Float	If the Gocator sensor is in Profile mode, the GenTL driver converts the profiles the sensor produces into an image that Halcon can process, using these settings. RawPartDetectionHeightThreshold: Detects the start and end of a

Element	Type	Description
		part when scanning. When data points in the profile go over this threshold, the driver starts concatenating the profiles. As soon as no data points are above this threshold, the driver stops concatenating profiles. RawPartDetectionMaxLength: The maximum length of a part. Either condition (data points going back below the threshold or maximum length reached) will stop profile concatenation for the part.
DataCapacity	64u	The amount of memory, in bytes, used for buffered data. Adjust this depending on the needs of the GenTL consumer.

Interfacing with Halcon

Halcon is a comprehensive software package for machine vision applications with an integrated development environment. A sensor can use the included GenTL driver to stream Surface scan data and intensity data into Halcon in real-time. Halcon is supported in GoPxL 1.0 and higher. LMI has tested using Halcon 23.05.

> The current GenTL driver does not support scanning in profile mode.

For information on setting up the GenTL driver, see *GenICam GenTL Driver* on page 910.

This section describes how to configure Halcon to acquire data from the Gocator sensor. You should be familiar with the sensor's Surface mode. Before continuing, make sure Halcon is installed.

> Halcon sets the size of the buffer for acquisition only when the driver is initialized. For this reason, if you increase the size of the sensor output after driver initialization, the scan data Halcon receives will be cropped.

> If Halcon is consuming the output of a multi-sensor system or of a tool such as Surface Stitch, that is, where multiple frames are combined, you should set OverrideImageWidth and OverrideImageLength to the maximum expected size. For more information, see *XML Settings File* on the previous page.

Setting Up Halcon

Before using Halcon with a sensor, you must set up Halcon.

To set up Halcon:

1. Connect a sensor to the PC running Halcon.

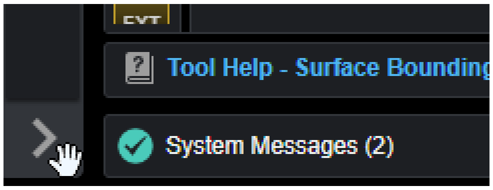
   You will need a Master hub to connect the sensor to the PC. For more information, see *Installation* on page 36 and *Network and Sensor Setup* on page 50.

2. Connect to GoPxL running either on a sensor or a PC using a web browser.

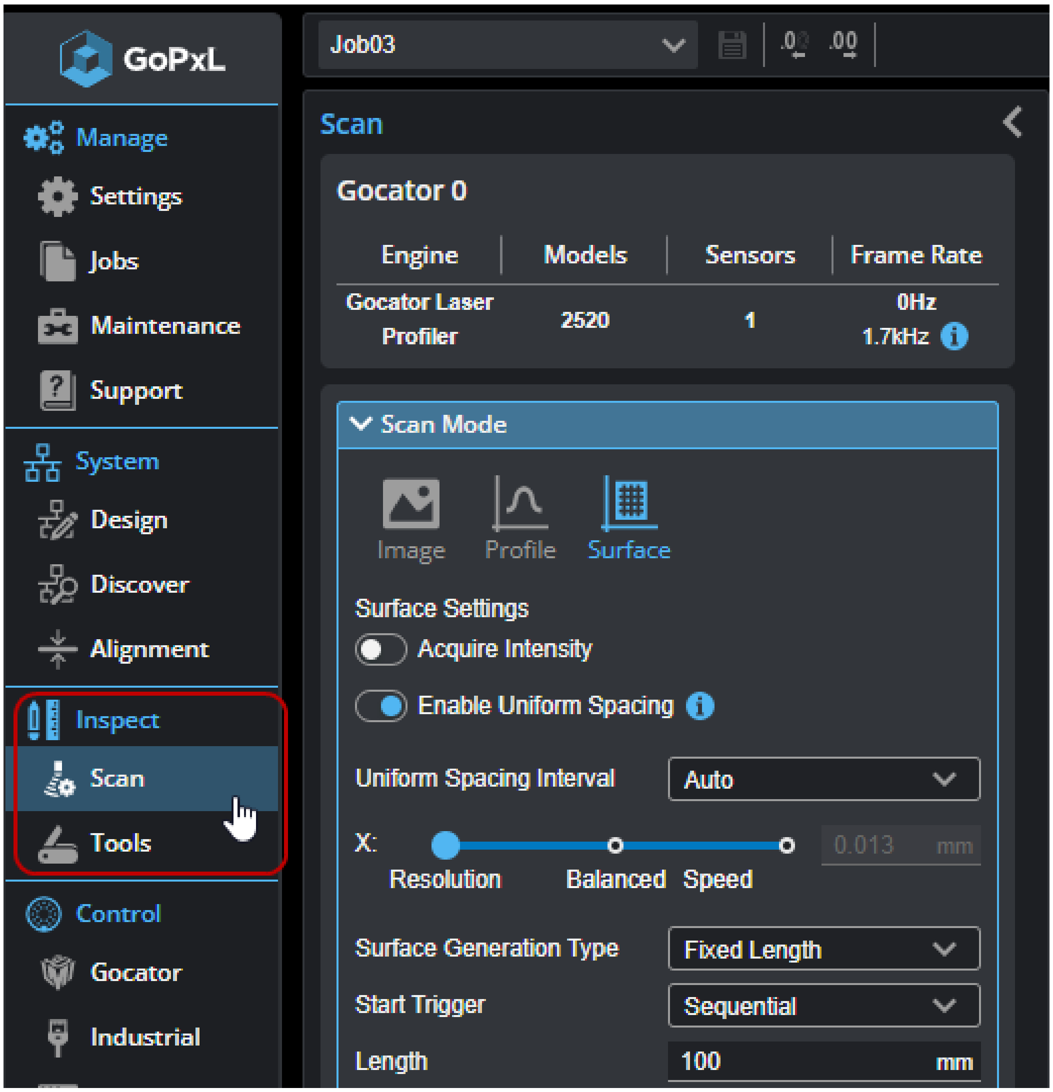
   By default, a sensor's IP is 192.168.1.10. If you have changed the sensor's IP, use that. If you can't find your sensor, see *GoPxL Discovery Tool* on page 935.

   For more information on running GoPxL on a PC, see *Running GoPxL on a Windows PC* on page 828.

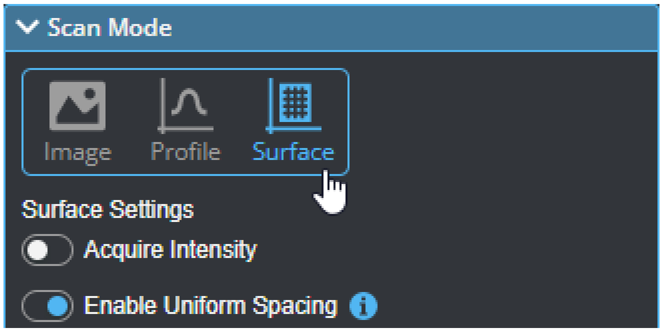
3. In the GoPxL interface, expand the navigation bar on the left side of the interface.



4. Click the **Scan** page icon in the navigation bar on the left.

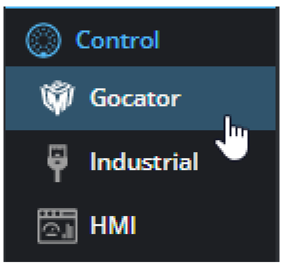


5. On the **Scan** page, click the **Surface** icon to switch to Surface mode.

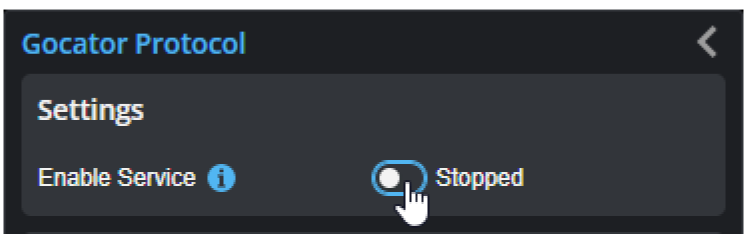


6. (Optional) If you need intensity data, enable **Acquire Intensity**.
7. Configure the sensor to produce the desired surface data.

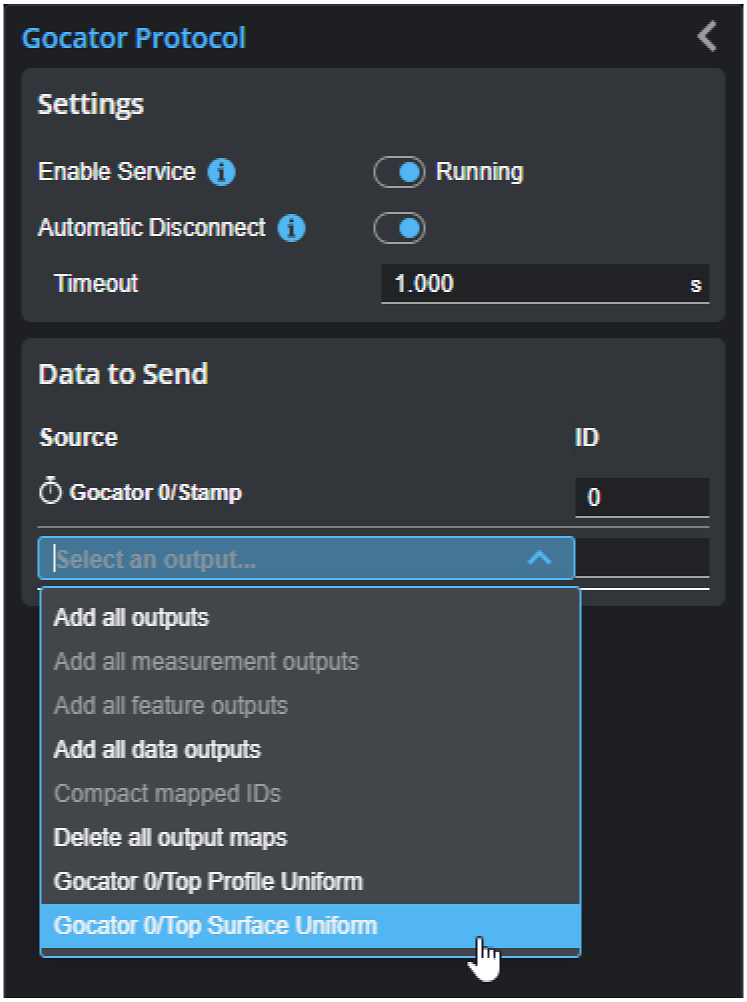
   For more information on configuring sensors, see *Scan - Configuring Acquisition* on page 195.
8. Click the **Gocator** page icon in the navigation bar.



9. On the **Gocator** page, enable the Gocator protocol.

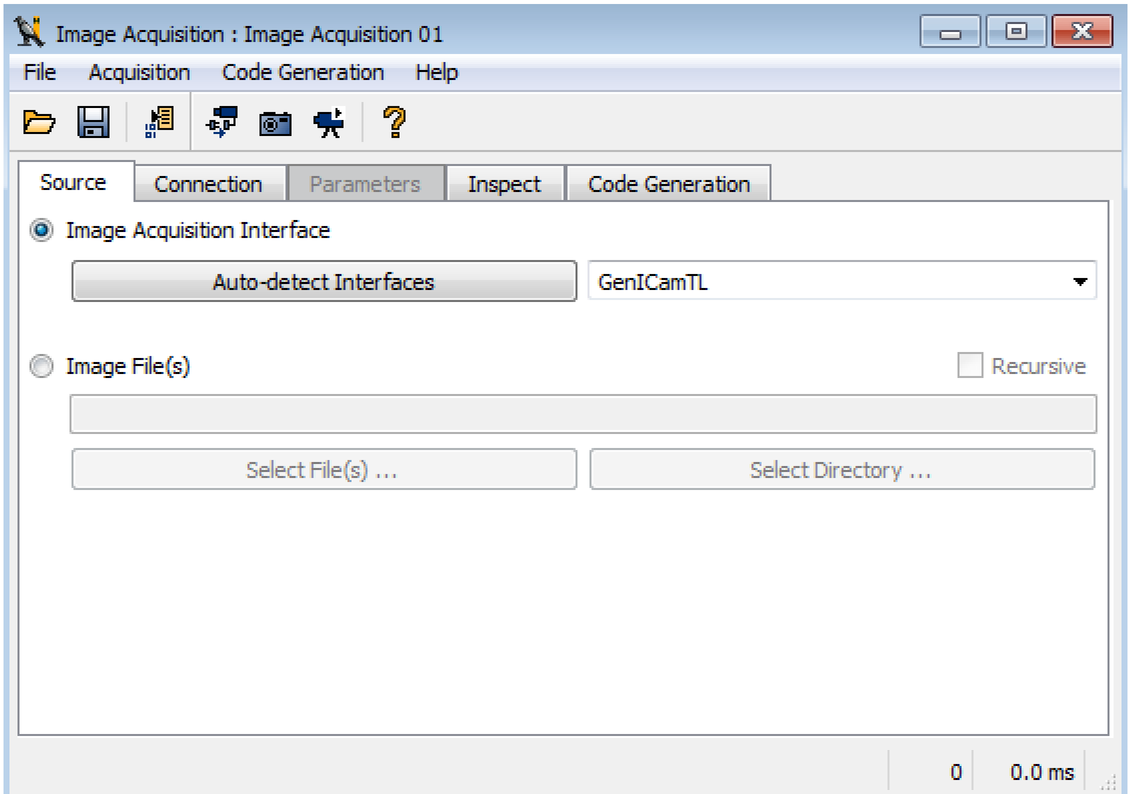


10. In the **Data to Send** section, choose the surface you want to send to Halcon.



For more information on configuring Gocator output, *Gocator Communication Protocol* on page 797.

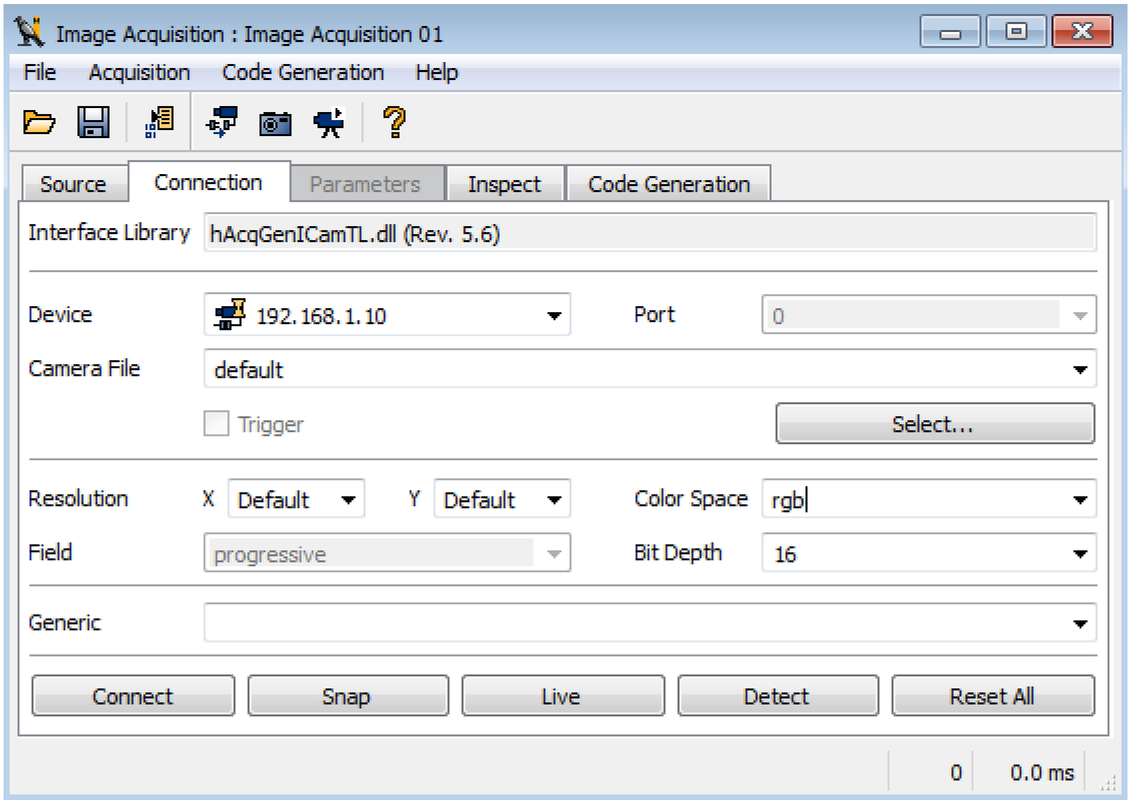
11. Make sure the sensor is running.
12. On the PC, launch Halcon.
13. In Halcon, in the **Assistants** menu, click **Open New Image Acquisition**.
14. In the dialog that opens, in the **Source** tab, check the **Image Acquisition Interface** option and choose `GenICamTL` in the drop-down.



The driver uses the Gocator protocol discovery messages to search for available Gocator sensors. Discovery messages can be blocked by a PC's firewall. You should therefore turn off the firewall and try again if the sensor can't be detected.

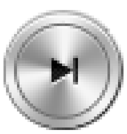
15. Switch to the **Connection** tab.

    If Halcon detects a sensor, the sensor's IP will be listed next to **Device**.

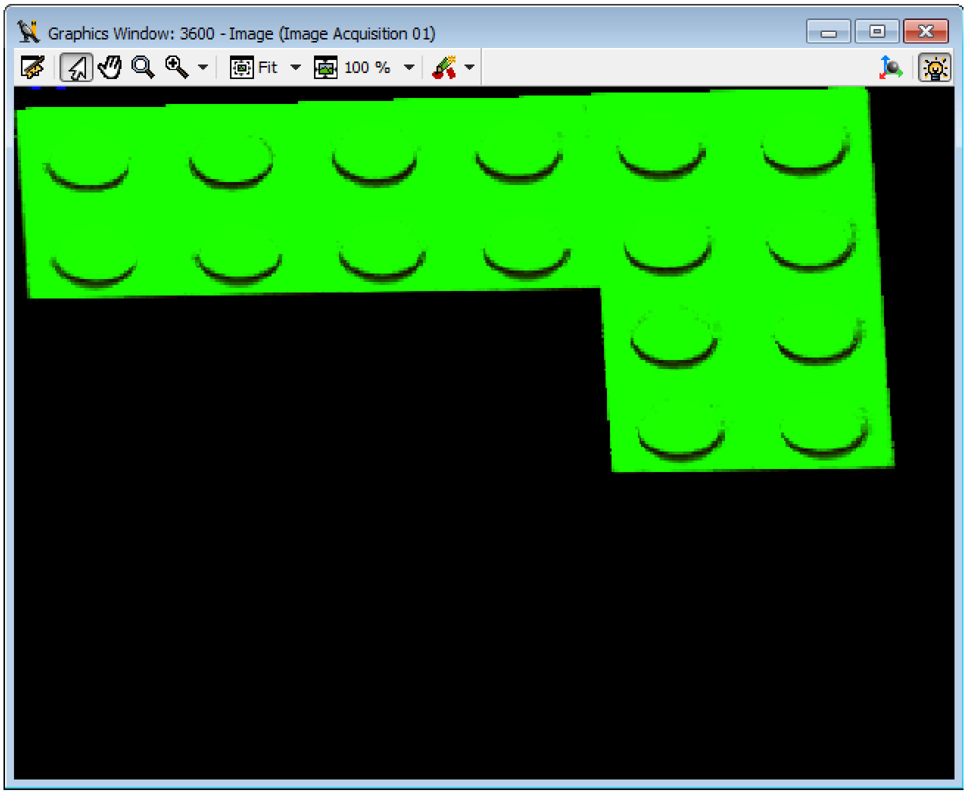


16. In the **Connection** tab, set **Color Space** to `RGB` and **Bit Depth** to `16`.

17. In the sensor's web interface, click the Snapshot button to trigger the output of a surface.



The output displays in the Halcon **Graphics Window**.



Halcon is now configured for use with the sensor.

Halcon Procedures

In Halcon, you can use various internal procedures (functions) to decompose the RGB image and to control registers that the GenTL driver opens.

You can import the procedures into your own code by selecting File > Insert Program > Insert Procedures and then choosing the example code `Continuous_Acq.hdev` under the `Examples/Halcon` directory.

The following table describes each of these procedures.

Halcon Procedures

Procedures	Description
set_framegrabber_param	Generic Halcon function to set parameters on the sensor. Can be used to set sensor-specific settings and execute specific commands. For a complete list of settings that can be changed, see below. In the generic form: `set_framegrabber_param( AcqHandle, 'Name', 'Value')` *Parameters (Input)* **AcqHandle**: Acquisition handle created by `open_framegrabber`. **Name**: The name of the parameter to set on the sensor. **Value**: The parameter value to set on the sensor.

Procedures	Description
	For full details on using this function, see *XmlSetting and XmlCommand Parameters* on page 928.
get_ framegrabber_ param	Generic Halcon function to get parameters on the sensor. Used in conjunction with set_ framegrabber_param to set the path to the setting, after which the setting's value is retrieved with get_framegrabber_param. In the generic form: `get_framegrabber_param( AcqHandle, 'Name', 'Value')` *Parameters* **AcqHandle** (input): Acquisition handle created by `open_framegrabber`. **Name** (input): The name of the parameter to set on the sensor. **Value** (output): The variable where parameter's value will be stored. For full details on using this function, see *XmlSetting and XmlCommand Parameters* on page 928.
Go2GenTL_ ParseData	The GenTL driver packs the height map, intensity and stamp information into a 16-bit RGB image. The function is used to extract data from the RGB image. For details on how the information is packed in the data, see the sections under *GenICam GenTL Driver* on page 910. The function accepts the image acquired from `grab_image_async`, and returns the height map, intensity and stamps. ***Parameters*** **Image** (input): RGB Image acquired by using `grab_image_async`. **HeightMap** (output): The height map image. **Intensity** (output): The intensity image. **Stamps** (output): The stamps structure. **Index** (input): The index of the scan to extract. **NumScans** (output): The number of scans in the acquired image. Each output is returned as decimal value. ***Example*** `Go2GenTL_ParseData(Image, HeightMap, Intensity, Stamps, 0, NumScans)`
Go2GenTL_ ParseStamps	The stamp information is encoded in the image and can be retrieved using the Go2GenTL_ ParseData function. The function Go2GenTL_ParseStamps is used to extract the stamp information from the stamp structure. For details on how the information is packed in the data, see the sections under *GenICam GenTL Driver* on page 910. The function accepts the stamp structure acquired from the Go2GenTL_ParseData function, and returns the various values encoded in the stamps. ***Parameters (Input)*** **Stamps**: The stamps structure acquired by using Go2GenTL_ParseData. ***Parameters (Output)*** **FrameCount**: The number of frames.

Procedures	Description
	Timestamp: The timestamp. **Encoder**: The encoder position. **EncoderIndex**: The last index of the encoder. **Inputs**: The digital input states. **xOffset**: The X offset in millimeters. **xResolution**: The X resolution in millimeters. **yOffset**: The Y offset in millimeters. **yResolution**: The Y resolution in millimeters. **zOffset**: The Z offset in millimeters. **zResolution**: The Y resolition in millimeters. **zOffset**: The Z offset in millimeters. **zResolution**: The Z resolution in millimeters. **Width**: The width (number of columns) of the image that contains the part. The part width can be less than the image width requested by the user. **Height**: The height or length (number of rows) of the image that contains the part. The part length can be less than the image height requested by the user. **HasIntensity**: Specifies if the intensity image is available. The intensity image is available if **Acquire Intensity** is enabled in the sensor's web interface. **NumScans**: The number of scans in the acquired image. **SyncTimestamp**: The PTP time stamp. **MeasurementIds**: Array of every measurement's ID. **MeasurementValues**: Array of every measurement's value. **MeasurementDecisions**: Array of every measurement's decision. **MeasurementDecisionCodes**: Array of every measurement's code. Each output is returned as decimal value. ***Example*** `Go2GenTL_ParseStamps(Stamps, FrameCount, Timestamp, EncoderPosition, EncoderIndex, Inputs, xOffset, xResolution, yOffset, yResolution, zOffset, zResolution, Width, Length, HasIntensity, NumScans, SyncTimestamp, MeasurementIds, MeasurementValues, MeasurementDecisions, MeasurementDecisionCodes)`
Go2GenTL_ ResampleMode	Returns the resample mode. ***Parameters (Input)*** **AcqHandle**: Acquisition handle created by `open_framegrabber`.

Procedures	Description
	Parameters (Output) **ResampleMode**: No - Resample is disabled. Yes - Resample is enabled. When resampling is enabled, the GenTL driver resamples the height map so that the pixel spacing is the same on the X and Y axis. ***Example*** `Go2GenTL_ResampleMode (AcqHandle, ResampleMode)` To set the resample mode, you must directly modify Go2GenTL.xml, which is in the same directory as the sensor GenTL driver (Go2GenTL.cti).
Go2GenTL_ ConfigFileName	Returns the current live sensor job file name. *Parameters (Input)* **AcqHandle**: Acquisition handle created by `open_framegrabber`. *Parameters (Output)* **ConfigFile**: The name of the job file. The file name includes the extension .job. *Example* `Go2GenTL_ConfigFileName (AcqHandle, ConfigFile)`
Go2GenTL_ SetConfigFileName	Sets the sensor live configuration. *Parameters (Input)* **AcqHandle**: Acquisition handle created by `open_framegrabber`. **ConfigFile**: The name of the job file. The file name should include the extension .job. *Example* `Go2GenTL_SetConfigFileName (AcqHandle, 'test2.cfg')`
Go2GenTL_ Encoder	Returns the current encoder value. When this function is called, the GenTL driver retrieves the latest encoder value from the sensor. The value is returned as a two-element tuple. The first element is the least significant 32-bit value, and the second element is the most significant 32-bit value. *Parameters (Input)* **AcqHandle**: Acquisition handle created by `open_framegrabber`. *Parameters (Output)* **EncoderValue**: The current encoder value. *Example* `Go2GenTL_Encoder(AcqHandle, EncoderValue)`
Go2GenTL_ ImageSize	Returns the size of the image returned by the GenTL driver. *Parameters (Input)* **AcqHandle**: Acquisition handle created by `open_framegrabber`. *Parameters (Output)* **Width**: The width of the image. **Height**: The height of the image. *Example* `Go2GenTL_ImageSize(AcqHandle, Width, Height)`

Procedures	Description
	To set the image size, you must directly modify Go2GenTL.xml, which is in the same directory as the sensor GenTL driver (Go2GenTL.cti).
Go2GenTL_ CoordinateXYZ	Returns the real-world coordinates (X, Y, Z) of the part given the row and column position in the height map. The values of the offset and resolution input parameters can be retrieved using Go2GenTL_ ParseData. *Parameters (Input)* **HeightMap**: The height map image. **Row**: The row in the height map. **Column**: The column in the height map. **xOffset**: The X offset in millimeters. **xResolution**: The X resolution in millimeters. **yOffset**: The Y offset in millimeters. **yResoluion**: The Y resolution in millimeters. **zOffset**: The Z offset in millimeters. **zResolution**: The Z resolution in millimeters. *Parameters (Output)* **coordinateXYZ**: The real-world coordinates.
Go2GenTL_ Exposure	Returns the current exposure. *Parameters (Input)* **AcqHandle**: Acquisition handle created by `open_framegrabber`. *Parameters (Output)* **Exposure**: The current exposure value (in µs). The value is returned as an integer. Decimals are truncated. *Example* `Go2GenTL_Exposure(AcqHandle, exposure)`
Go2GenTL_ SetExposure	Sets the current exposure. *Parameters (Input)* **AcqHandle**: Acquisition handle created by `open_framegrabber`. **Exposure**: The current exposure value (in µs), as an integer. *Example* `Go2GenTL_SetExposure(AcqHandle, exposure)`

XmlSetting and XmlCommand Parameters

Sensor Settings

You use the `set_framegrabber_param` and `get_framegrabber_param` functions to set and retrieve the values of advanced parameters on sensors, and additionally modify special settings and send commands at the driver level. (Examples are shown below and in *Sample Program for Various GenTL Settings* on page 931.)

When you set the second parameter of `set_framegrabber_param` to "XmlSetting", you can use the function to do the following:

- Set the path to a setting you wish to set *or* get.
- Set the value of a setting.

- Set an attribute of a setting.

When you set the second parameter of `set_framegrabber_param` to "XmlCommand", you can send commands and change driver-level settings at the driver level.

Finally, when you retrieve the value of a sensor setting, you set the second parameter of `get_framegrabber_param` to "XmlSettingReturn".

For example, to get the value for a setting, use `set_framegrabber_param` repeatedly to set the XML path to the setting, by chunks of up to 30 characters, and retrieve the value into the passed variable in `set_framegrabber_param`. The following retrieves the value for the setting "Setup/Filters/XSmoothing/Window":

```
set_framegrabber_param(AcqHandle, 'XmlSetting', 'Setup/Filters/')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'XSmoothing/')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'Window')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', var)
```

To set a value, use `set_framegrabber_param` repeatedly to set the XML path to the setting, by chunks of up to 32 characters, add "=<value>", and end with an empty string.

The following sets the value of "Setup/Filters/XSmoothing/Window" to 5:

```
set_framegrabber_param(AcqHandle, 'XmlSetting', 'Setup/Filters/')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'XSmoothing/')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'Window=5')
set_framegrabber_param(AcqHandle, 'XmlSetting', '')
```

You can also optionally add the "\n" character to the end of the last line of the set operation:

```
set_framegrabber_param(AcqHandle, 'XmlSetting', 'Setup/Filters/')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'XSmoothing/')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'Window=5\n')
```

To set or get an attribute, use the same format, but use a space to delimit the attribute ('Setup/Filters/XSmoothing/Window min="0.044"').

For a set or a get, if an index or a parameter is required, use parentheses:

```
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/ResetGocator(1)')
set_framegrabber_param(AcqHandle, 'XmlSetting', '')
```

GenTL Driver Settings and Commands

You can change settings or send commands at the driver level. These options always start with GenTL/, in contrast with the Setup/ node for configuring the sensor.

For example, to stop a sensor programmatically, you can do the following:

```
set_framegrabber_param(AcqHandle, 'XmlCommand', 'GenTL/StopGocator')
set_framegrabber_param(AcqHandle, 'XmlCommand', '')
```

You can also optionally add the "\n" character to the end of the last line of the set operation:

```
set_framegrabber_param(AcqHandle, 'XmlCommand', 'GenTL/StopGocator\n')
```

The following driver-level settings are available:

Driver-level settings

Setting	Comment
"GenTL/Buffers"	Read-Write. Gets/Sets the number of buffers to use in the acquisition pipeline.
"GenTL/Encoder"	Read-only. Gets the current encoder value.
"GenTL/IsResponsive"	Read-only. Gets the responsiveness of the sensor.
"GenTL/HasChanges"	Read-only. Indicates whether the sensor needs a GenTL/Refresh command (see See *Driver-level commands* on the next page).
"GenTL/Version"	Read-only. Returns the version of the GenTL driver.
"GenTL/StartStopControl"	Read-Write. Enables or disables the user defined control. Setting this value to 1 allows the user to start and stop manually the sensor, using the "GenTL/StartGocator" and "GenTL/StopGocator" commands.
"GenTL/ScheduledStart" "GenTL/SystemScheduledStart" "GenTL/SensorScheduledStart"	Read-only. Schedule a start when the stamp (in the current domain, can be time or encoder mode) is reached. Applies for all three "GenTL/XScheduledStart" commands.
"GenTL/State" "GenTL/State(id)"	Read-only. By default, gets the state of the main sensor. If an ID is provided, the state of the connected sensor is returned. Use ID=0 for the main sensor. Any buddy starts at ID=1. For a system with a main and 2 buddies, the indices can be from 0 to 2.
"GenTL/BuddyState" "GenTL/BuddyState(id)"	Read-only. By default, gets the state of the first buddy sensor. If an ID is provided, the state of the connected sensor is returned. Use ID=0 for the first buddy sensor. For a system with a main and 2 buddies, the indices can be 0 or 1.
"GenTL/BuddyAt" "GenTL/BuddyAt(id)"	Read-only. By default, gets the serial number of the first buddy sensor. If an ID is provided, the serial number of the connected sensor is returned. Use ID=0 for the first buddy sensor. For a system with a main and 2 buddies, the indices can be 0 or 1.
"GenTL/BuddyCount"	Read-only. Returns the number of buddy sensors. For a system with a main and 2 buddies, the function will return 2.

Setting	Comment
"GenTL/Main"	Read-only. Returns the serial number of the main sensor.
"GenTL/AlignResult"	Read-only. Returns the alignment result of the last "GenTL/Align" command (see See *Driver-level commands* below).
"GenTL/LastResult"	Read-only. Returns the last result for any GenTL XmlCommand.
"GenTL/TransformX(id)" "GenTL/TransformY(id)" "GenTL/TransformZ(id)" "GenTL/TransformXAngle(id)" "GenTL/TransformYAngle(id)" "GenTL/TransformZAngle(id)"	Read-Write. Gets or sets the transformation value for a given transform of the specificed sensor ID.
"GenTL/TransformEncoderResolution"	Read-Write. Gets/Sets the encoder resolution for a sensor.

The following driver-level commands are available:

Driver-level commands

Setting	Comment
"GenTL/FlushOldBuffers"	Flushes the GenTL pipeline buffers. Can be used to ensure only a new surface captured, instead of getting the previously captured surfaces.
"GenTL/FlushGocatorBuffers"	Flushes the Gocator pipeline buffers. Can be used to ensure only a new surface captured, instead of getting the previously captured surfaces.
"GenTL/StartGocator"	In an enabled StartStopControl mode, starts the Gocator programmatically.
"GenTL/StopGocator"	In an enabled StartStopControl mode, stops the Gocator programmatically.
"GenTL/ResetGocator" "GenTL/ResetGocator(wait)"	By default, this call resets the Gocator without waiting. If the wait parameter is set to 1, the command will wait for the Gocator to be reset, and reconnects to it automatically if possible.
"GenTL/Align(timeout)"	Performs an alignment on the sensor, waiting for the given timeout in seconds.
"GenTL/ClearAlignment"	Clears the current alignment on the sensor.
"GenTL/ResetEncoder"	Resets the encoder. Note that this only works if the encoder is directly connected to the sensor, and may not work on encoders connected to a Master.
"GenTL/Reconnect"	Attempts to reconnect to the sensor, after it was disconnected or reset.
"GenTL/Refresh"	Updates the client state to reflect any changes that have occurred in the sensor network.
"GenTL/ClearData"	Clears the data on message queue of the Gocator.
"GenTL/SoftwareTrigger"	Triggers the Gocator for an acquisition, if it is setup to use Software Triggering.

Sample Program for Various GenTL Settings

```
* Connect to the sensor. This code is auto generated by the Image Acquisition dialog box.
User can manually override the directory path by editing the line below.
* Set image type to 'rgb' if the format is 16-bit RGB. 'gray' if the format is 16-bit mono
open_framegrabber ('GenICamTL', 0, 0, 0, 0, 0, 0, 'progressive', 16, 'rgb', -
1, 'false', 'default', '127.0.0.1:3190', 0, -1, AcqHandle)
```

```
* Example calls for configuring the sensor
* Go2GenTL_ResampleMode (AcqHandle, ResampleMode)
* Go2GenTL_ImageSize (AcqHandle, ImageWidth, ImageHeight)
* Go2GenTL_Encoder (AcqHandle, EncoderValue)
Go2GenTL_Exposure (AcqHandle, currentExposure)
Go2GenTL_SetExposure (AcqHandle, currentExposure + 20)
set_framegrabber_param (AcqHandle, 'grab_timeout', 100000)
* set_framegrabber_param (AcqHandle, 'XmlCommand', 'GenTL/ClearData\n')

* Note: Configuration name has to be less than 16 characters
* Go2GenTL_SetConfigFileName (AcqHandle, 'newExposure.job')
* Go2GenTL_ConfigFileName (AcqHandle, ConfigFile1)

* Other examples
* *Get fixed length
* set_framegrabber_param(AcqHandle, 'XmlSetting', 'Setup/SurfaceGener')
* set_framegrabber_param(AcqHandle, 'XmlSetting', 'ation/FixedLength/')
* set_framegrabber_param(AcqHandle, 'XmlSetting', 'Length')
* get_framegrabber_param(AcqHandle, 'XmlSettingReturn', var1)
* *Set fixed length
* set_framegrabber_param(AcqHandle, 'XmlSetting', 'Setup/SurfaceGener')
* set_framegrabber_param(AcqHandle, 'XmlSetting', 'ation/FixedLength/')
* set_framegrabber_param(AcqHandle, 'XmlSetting', 'Length=400')
* set_framegrabber_param(AcqHandle, 'XmlSetting', '')

* Version
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/Version')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', version)

* Transform
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformX(1)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', xOffset)
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformX(1)=0.01\n')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformX(1)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', xOffset)

set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformYAngle(1)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', yAngle)
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformYAngle(1)=0.01\n')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformYAngle(1)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', yAngle)

* Encoder Resolution
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformEncoderResolution')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', encoderRes)
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformEncoder')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'Resolution=0.002\n')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/TransformEncoderResolution')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', encoderRes)

* Buddy List
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/BuddyCount')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', buddyCount)
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/BuddyAt(0)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', buddy0)
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/BuddyAt(1)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', buddy1)
```

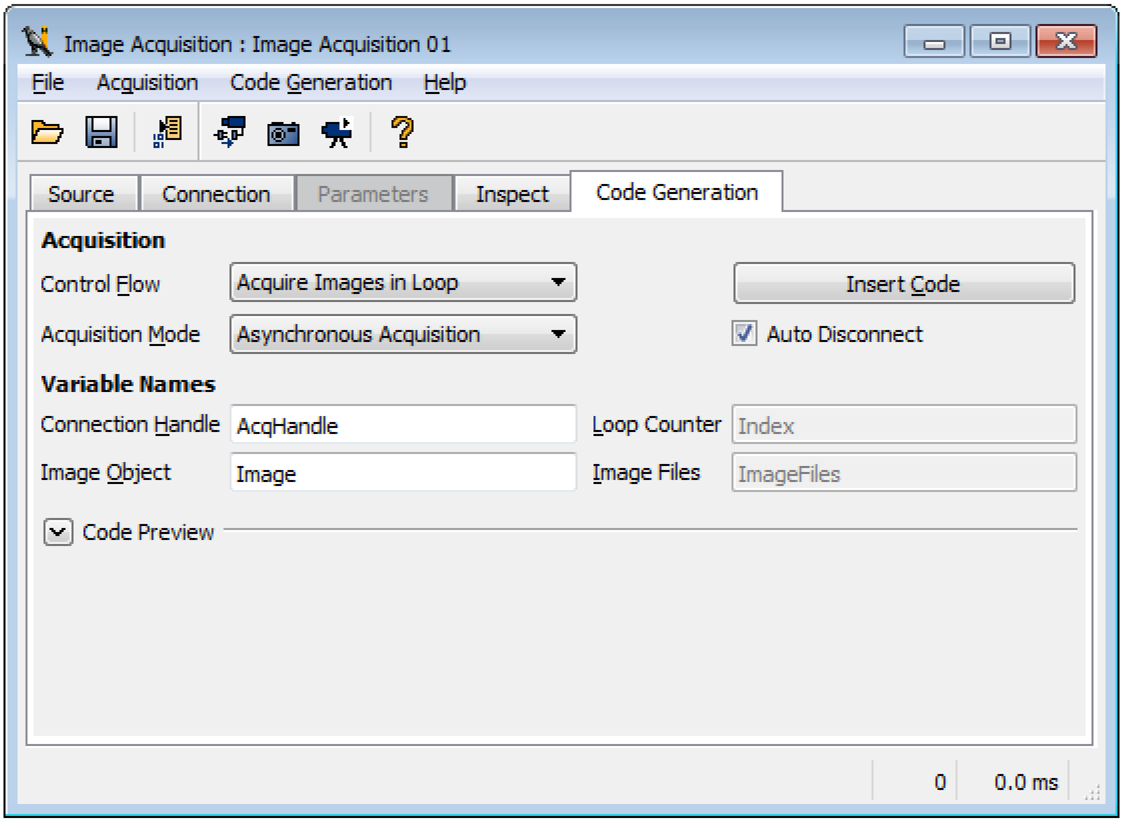
```
* State
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/Refresh\n')

set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/State')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', state0)
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/State(1)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', state1)

set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/BuddyState')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', bstate0)
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/BuddyState(1)')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', bstate1)

* Alignment
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/ClearAlignment')
set_framegrabber_param(AcqHandle, 'XmlSetting', '')

set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/AlignResult')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', alignResult)

set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/Align(5)')
set_framegrabber_param(AcqHandle, 'XmlSetting', '')

set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/AlignResult')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', alignResult)

* Reset
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/ResetGocator(1)\n')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/LastResult')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', lastResult)

* Responsiveness
set_framegrabber_param(AcqHandle, 'XmlSetting', 'GenTL/')
set_framegrabber_param(AcqHandle, 'XmlSetting', 'IsResponsive')
get_framegrabber_param(AcqHandle, 'XmlSettingReturn', isResponsive)


close_framegrabber (AcqHandle)
```

Generating Halcon Acquisition Code

Halcon lets you insert acquisition code into your code in the IDE.

To generate acquisition code:

1. In Halcon, in the **Assistants** menu, click **Open New Image Acquisition**.
2. In the dialog that opens, in the **Code Generation** tab, set **Acquisition Mode** to **Asynchronous Acquisition**.



3. Under **Acquisition**, click **Insert Code** to generate the code that will open the acquisition device.

> To handle cases when the `grab_image` function times out while waiting for data, add a `try-catch` statement around the `grab_image` function code.

After the example code is generated, you should add a `catch` instruction to bypass the acquisition timeout event, and use the Go2GenTL_ParseData function to extract information from the returned image.

An example, `Continuous_Acq.hdev`, is included in the `Examples/Halcon` directory.

Utilities

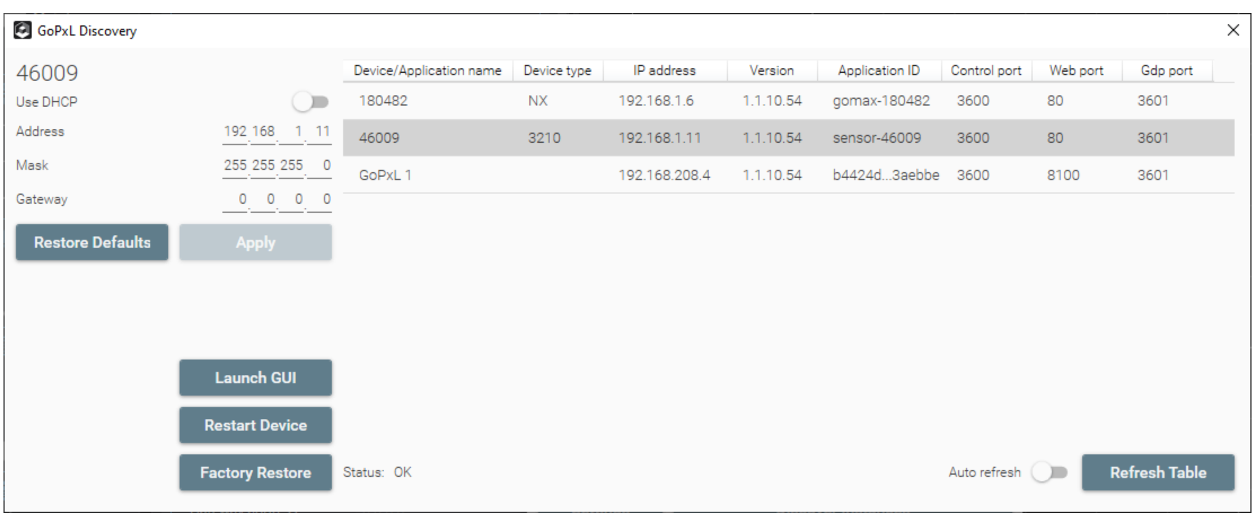
The following sections describe some of the utilities tools provided with a Gocator sensor, as well as the CSV format that a sensor can export. For information on the integrations available with a sensor, see *Integrations* on page 884.

You can get the tools in the utilities package (`14631-x.x.x.x_SOFTWARE_GoPxL_Utilities.zip`) from the downloads area of the LMI Technologies website: https://lmi3d.com/product-downloads/.

- Alignment CAD (G2): CAD files for pyramid plate and double-sided pyramid alignment targets. For use with Surface Align Wide and Surface Align Ring on multi-sensor G2 systems, respectively. You should only use these if your application requires a high degree of accuracy. For more information, see *Aligning Sensors to 6 Degrees of Freedom* on page 157.
- Discovery Tool: Used to find sensors on a network. See *GoPxL Discovery Tool* below.
- Replay Converter Tool: Lets you convert between .gprec files and .srf/.sur/.pcd/.pro files. It also lets you convert .gprec files to CSV files. For more information, see *Replay Converter Tool* on page 938.
- Track Editor: Used with the Surface Track tool. For more information, see *Surface Track* on page 661.
- Pattern Editor: Used to edit patterns created in the Surface Pattern Matching tool. For more information, see *Pattern Editor* on page 951.

GoPxL Discovery Tool

The GoPxL Discovery tool lets you perform various operations with sensors running GoPxL, PC instances of GoPxL, and GoMax NX units.



You can do the following with a sensor running GoPxL:

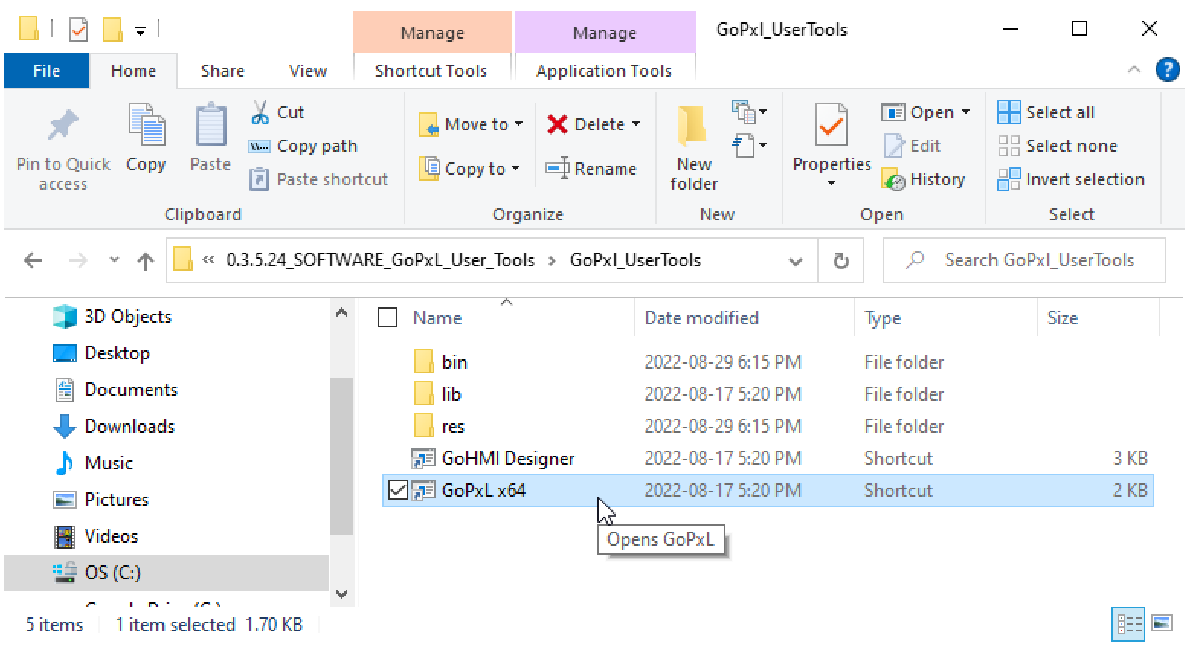
- Find a sensor if you have forgotten its IP address.
- View port information.
- Set the network configuration (IP address, mask, and so on) of a sensor, and also restore these values to the default.
- Launch the GUI of a sensor in your browser.
- Restart the sensor.
- Perform a factory restore on a sensor.

You can do the following with a PC instance of GoPxL or a GoMax NX:

- Find its IP address.
- View port information.
- Launch the GoPxL of the PC instance or GoMax NX in your browser.

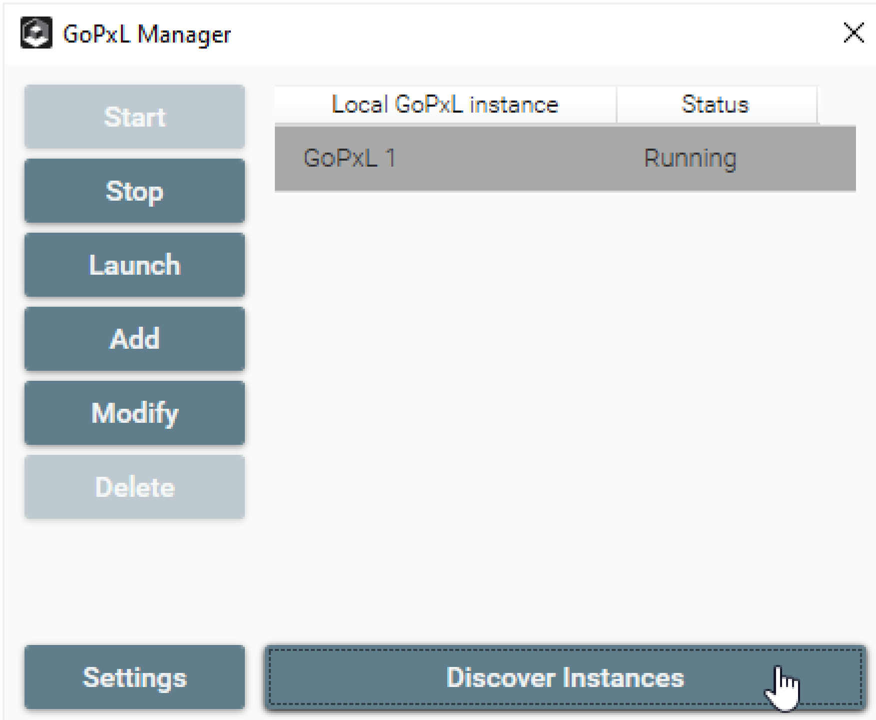
To run the GoPxL Discovery tool

1. If it isn't already running, start the Windows version of GoPxL.



   GoPxL Manager opens.

2. In the manager tool, click **Discover Instances**.



Your firewall may ask if you want to allow the GoPxL Service Application and Python to communicate over your network. To avoid issues, LMI recommends allowing communication over both Public and Private networks. Because the notifications appear one on top of the other, make sure to check the appropriate boxes and carefully click **Allow access** for both alerts.

The GoPxL Discovery tool opens.

When the GoPxL Discovery tool opens, it lists the sensors running GoPxL and the PC instances of GoPxL that are currently visible on the network. Sensors are identified in the **Device/Application Name** column by their serial number, and in the **Device Type** column by their model. If a sensor is running through a PC instance of GoPxL, its status (at the bottom of the tool) is "Device is remotely connected", and you can't perform any operations on it from the tool. Otherwise, the sensor's status is "OK".

GoPxL instances are identified in the **Device/Application Name** column by their name.

The tool provides important network-related information, namely, IP addresses and Web, Control, and GDP ports. The Web port is the port used to connect to the GoPxL interface in a browser. For sensors, the default port is 80; you don't need to specify this in the URL when connecting directly to a sensor. The Control port and the GDP ports are used by the SDK and the REST API. (For an introduction to these, see *GoPxL SDK and REST API* on page 871.)

If a sensor or GoPxL instance is missing, click the **Refresh Table** button. Optionally, you can also enable the **Auto refresh** toggle; the table refreshes .

To perform operations on a sensor or a GoPxL instance, click its row in the tool and then perform the operation.

Replay Converter Tool

The Replay Converter is a console application that lets you do the following conversions with Profile and Surface scan data:

- .gprec <=> .srf/.sur/.pcd (Valid for Surface input data)
- .gprec <=> .pro (Valid for Profile input data)
- .gprec => .csv (Valid for Profile/Surface input data)

Intensity information is included when converting to and from .pcd and .srf files. Intensity is not supported when converting to and from the .sur format.

You can get the tool in the utilities package (`14631-x.x.x.x_SOFTWARE_GoPxL_Utilities.zip`) from the downloads area of the LMI Technologies website: https://lmi3d.com/product-downloads/.

After downloading the utility package, unzip the file and run ReplayConverter.exe (under Tools\ReplayConverter). For example, `ReplayConverter.exe -i myData.gprec -f 0 -o myOutput.sur`.

The tool provides the following parameters:

Parameters

Name	Description
--help	Show the help.
-i --input	Specify the file name of the input file.
-f --frame	Specify the *zero-based index* of the frame to be exported in the recording data. For example, to export the first frame in the scan data, use "-i 0"; for the second frame,

Name	Description
	use "-i 1". (Default is 0.)
-w --width	Specify the width of the exported surface. (Only valid for importing .pcd files.)
-h --height	Specify the height of the exported surface. (Only valid for importing .pcd files.)
-s --swap	Whether to swap the X and Z coordinates. (Only valid for importing .pcd files.)
-z --zoom	Specify the zooming factor of the exported surface. (Only valid for importing .pcd files.)
-r --remove	Remove the specific point. (Only valid for importing .pcd files.)
-a --all	Export all the frames if the import is recording data.
-o --output	Specify the file name of the output file. If the imported file isn't recording data and the exported file name isn't specified, the utilities uses the imported file name as the exported file name.

CSV File Format

An exported CSV file contains a series of sections. Each section begins with a row containing the name of the section, and ends with a row containing the string "End". An empty line separates each section.

Each section usually contains one or more subsections. Each subsection has a header row containing a list of field names, followed by one or more rows of data. There is usually no empty line between the subsections.

Other structures within sections are possible.

> The "Part" section is unrelated to part detection. It represents Surface data when **Enable uniform spacing** is checked.

Info

This section contains basic system information. It has one header row and one value row. The fields are described below:

Info Fields

Field	Description
CSV Version	Version of the CSV file format.
Sensor Count	Number of sensors in the system.
Trigger Mode	Trigger source: 0 – Time

Field	Description
	1 – Encoder 2 – Digital input 3 – Software
Trigger Rate	Frame rate for time trigger (Hz).
Trigger Delay Domain	Output delay domain: 0 – Time (µs) 1 – Encoder (mm)
Trigger Delay	Output delay (µs or mm, depending on delay domain defined above).
Operation Mode	The scan mode.
XResolution	System X resolution (mm).
YResolution	System Y resolution (mm).
ZResolution	System Z resolution (mm).
YSpeed	Y Speed (mm/s).
Layout	Sensor orientation: 0 – Normal (single-sensor system) / Wide (dual-sensor system) 1 – Opposite 2 – Reverse 3 – Grid

DeviceInfo

This section contains information about each device in the system. There is one header row, and one value row per device.

> Device and firmware information (the ID, Model, and Version fields) is not currently output.

DeviceInfo Fields

Field	Description
ID	Device serial number
Model	Device part number
Version	Firmware version
Exposure Mode	Exposure mode: 0 – Single exposure 1 – Multiple exposures 2 – Dynamic exposure
Exposure 0 through Exposure 4	Multiple exposures
Exposure Min	Dynamic exposure min
Exposure Max	Dynamic exposure max
FOV X	Active area X

Field	Description
FOV Y	Active area Y
FOV Z	Active area Z
FOV Width	Active area width
FOV Height	Active area length (Y). (Note difference in terminology.)
FOV Depth	Active area height (Z). (Note difference in terminology.)
Transform X	Transform X offset (mm)
Transform Y	Transform Y offset (mm)
Transform Z	Transform Z offset (mm)
Transform X Angle	Transform X Angle (degrees)
Transform Y Angle	Transform Y angle (degrees)
Transform Z Angle	Transform Z angle (degrees)

RecordingFilter

Recording filters are not currently supported.

Profile

This section describes uniform (or resampled) profile data, which is produced when the sensor is in Profile mode and uniform spacing is enabled. It has two sub-sections: attributes and data.

The attribute section has only one row of data.

Attribute Section Fields

Field	Description
Frame Count	Total number of frames
Column Count	Number of columns
X Offset	X offset (mm)
Y Offset	Y offset (mm)
Z Offset	Z offset (mm)

The data section has one or more rows of data per frame (for example, range and intensity).

Data Section Fields

Field	Description
Frame	Frame index
Source	Source (for example, 0 for Top)
Time	Stamp time
Encoder	Stamp encoder
Z Encoder	Stamp encoder Z
Inputs	Stamp inputs
Exposure	Stamp exposure (μs)

Field	Description
Y	Y value (mm)
Axis	Axis: Z (range) or I (Intensity)
(x values)	Each column in the sub-header is a resampled X position Each column in the data is the range (mm) or intensity (count)

RawProfile

This section describes point cloud profile data (or unresampled / raw data), which is produced when the sensor is in Profile mode and uniform spacing is disabled. It has two sub-sections: attributes and data.

The attribute section has only one row of data.

Attribute Section Fields

Field	Description
Frame Count	Total number of frames
Column Count	Number of columns
X Offset	X offset (mm)
Y Offset	Y offset (mm)
Z Offset	Z offset (mm)

The data section has one or more rows of data per frame (for example, range and intensity).

Data Section Fields

Field	Description
Frame	Frame index
Source	Source (for example, 0 for Top)
Time	Stamp time
Encoder	Stamp encoder
Z Encoder	Stamp encoder Z
Inputs	Stamp inputs
Exposure	Stamp exposure (µs)
Y	Y value (mm)
Axis	Axis: X, Z, or I (Intensity)
(x values)	Each column in header is an index. Each column in data is the X/Z value (mm) or intensity (count)

Part

This section describes uniform Surface data, which is produced when the sensor is in Surface mode and uniform spacing is enabled.

> Only the data for the frame currently selected in the UI is exported when you export part data to a CSV file.

The section has two sub-sections: attributes and data.

The attribute section has only one row of data.

Attribute Section Fields

Field	Description
Frame	Frame index
Source	Source (for example, 0 for Top)
Time	Stamp time
Encoder	Stamp encoder
Z Encoder	Stamp encoder Z
Inputs	Stamp inputs
Row Count	Number of rows
Column Count	Number of columns
X Offset	X offset (mm)
Y Offset	Y offset (mm)
Z Offset	Z offset (mm)

The data section contains the data of a single surface scan. Each data row corresponds to one Y position. The first row contains the X values, and the first column contains the Y values. The region inside contains the range values (mm) for the corresponding row and column.

SurfacePointCloud

This section describes point cloud data (unresampled surface data), which is produced when the sensor is in Surface mode and uniform spacing is disabled. It has two sub-sections: attributes and data.

The attribute section has only one row of data.

Attribute Section Fields

Field	Description
Frame	Total number of frames
Source	Source (for example, 0 for Top)
Time	Stamp time
Encoder	Stamp encoder
Z Encoder	Stamp encoder Z
Inputs	Stamp inputs
Row Count	Number of rows
Column Count	Number of columns
X Offset	X offset (mm)
Y Offset	Y offset (mm)
Z Offset	Z offset (mm)

The data section contains the data of a single surface scan. The first row (header) can be ignored.

The data are (x,y,z) tuples expanded into a flat list of values, for example:

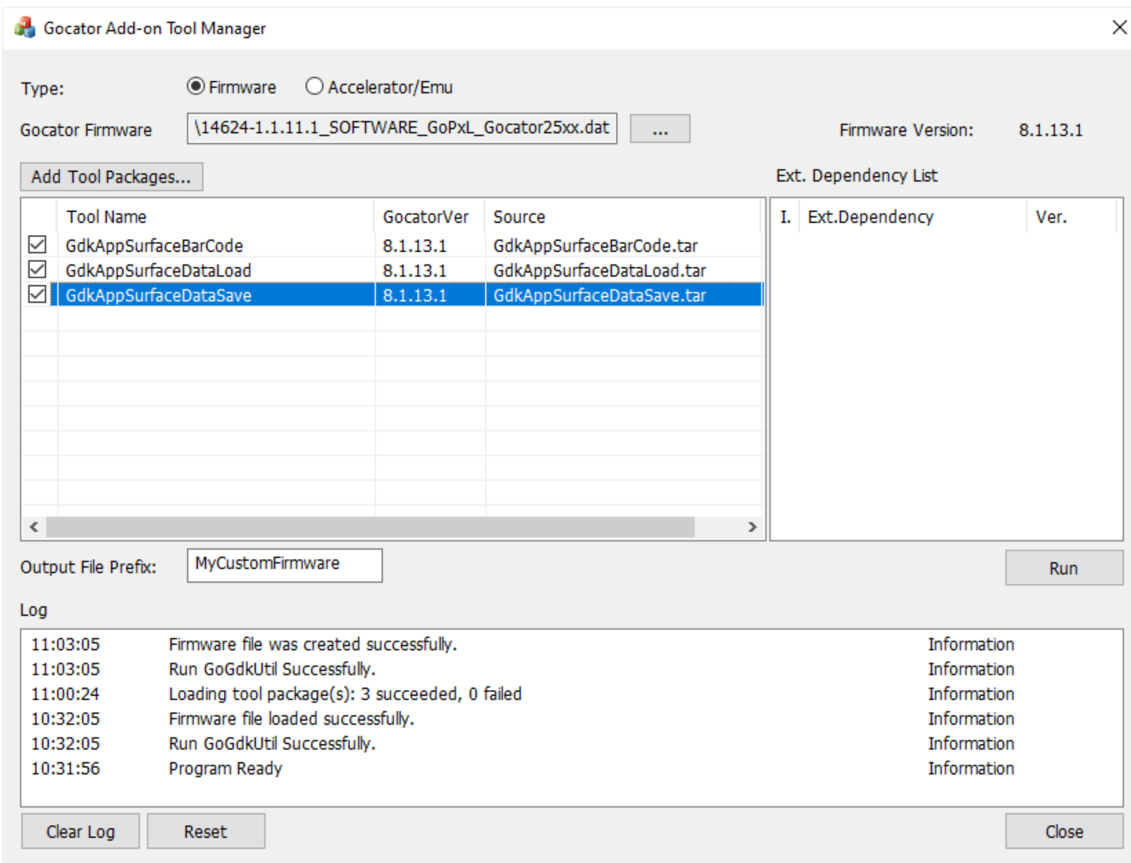
```
p0x, p0y, p0z, p1x, p1y, p1z, ..., pnx, pny, pnz
p(n+1)x, p(n+1)y, p(n+1)z, ...
...
```

Because the data has multiple rows and columns, it forms a rectangular grid of (x,y,z) tuples. The rows and columns do not correspond exactly to X and Y values, but suggest adjacency. i.e. positions with consecutive row indices or column indices are generally adjacent to each other in (x,y,z) coordinates.

The values are provided in mm, with the resolution and offset already calculated in the values.

Beta Add-on Tool Manager

The Beta Add-on Tool Manager lets you quickly and easily add LMI-provided beta tools to GoPxL. After you add tools to a firmware file, upload the firmware to a compatible sensor; for more information, see *Software Upgrade* on page 122. If you have added tools to GoPxL for PC, launch it to use them; for more information, see *Running GoPxL on a Windows PC* on page 828.



The tool manager is available in the 14632-x.x.x.x_SOFTWARE_GoPxL_Tools_AddOn_Beta.zip package, available on the LMI Product Downloads page (https://lmi3d.com/product-downloads/), under the *Beta Software Releases* section.

To get the package:

1. Go to https://lmi3d.com/product-downloads/ and log in to your account.
2. Select Gocator in the brand drop-down.
3. Choose the package that corresponds to the firmware you wish to run the tool manager on and download it.

Remember that the tools available in the package are *beta* tools. LMI does not recommend using them in production settings.

The general workflow with the Gocator Add-on Tool Manager is as follows:

1. Load a .dat Gocator firmware file (or point the manager to the "GoPxL" folder in an unzipped Utilities package).
2. Load one or more beta tool packages (in .tar archives).
3. Enable the tools you want to add to the loaded firmware.
4. Run the tool manager on the firmware and tool packages.

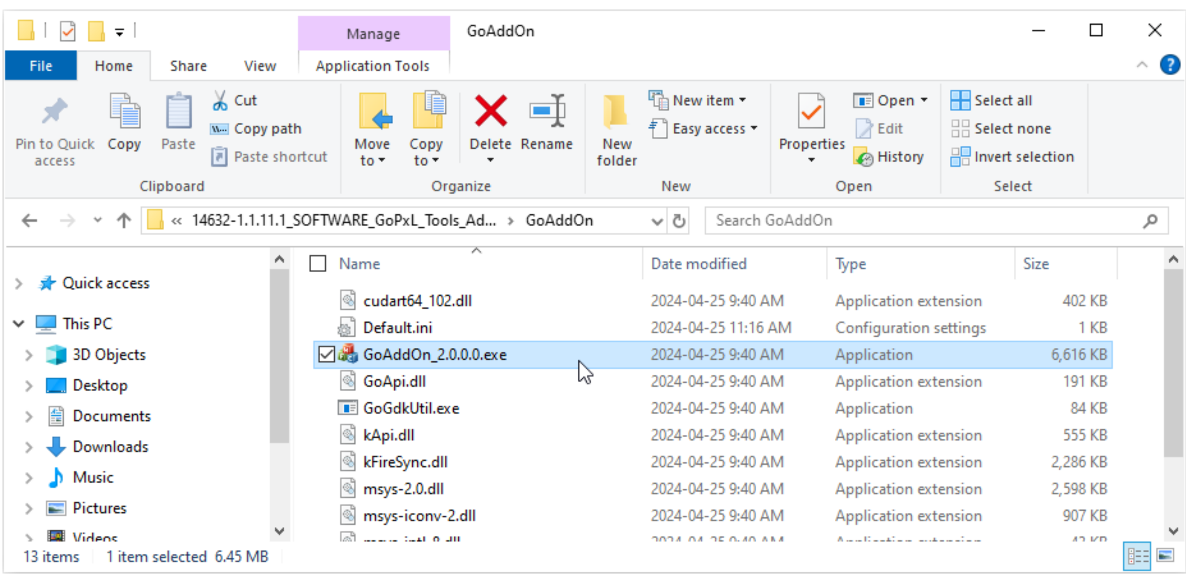
The result is a modified firmware file or GoPxL for PC that contains the tools you selected.

At any point before creating the new firmware with the tool manager, you can click **Reset** to remove the loaded firmware and tool packages, and start over.

Adding Beta Tools

To add beta tools to a firmware or to GoPxL for PC:

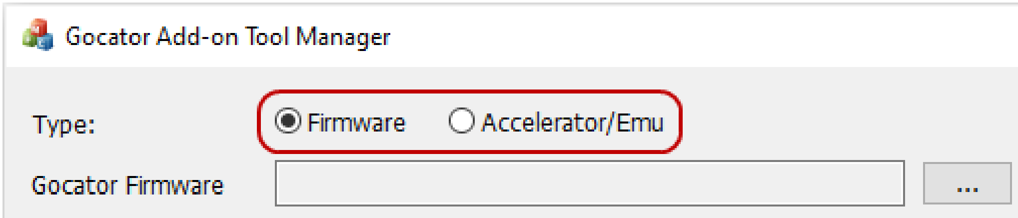
1. If you haven't already done so, download and unzip the package containing the Gocator Add-on Tool Manager to a convenient location on your computer.
2. Launch the tool manager (GoAddOn_x.x.x.x.exe) from the \GoAddOn subfolder.



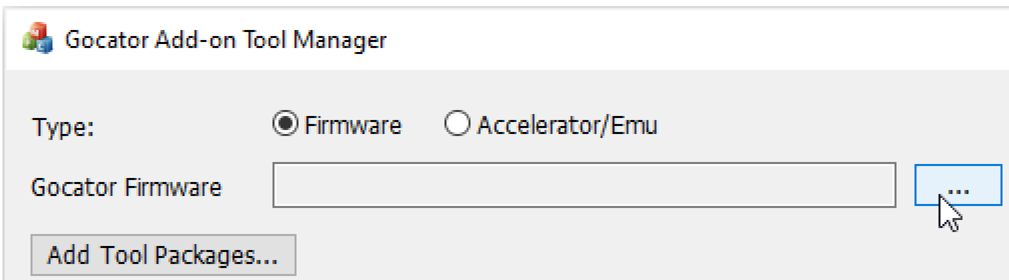
3. In the tool manager, set the type.

   To create a custom firmware you will upload to a sensor, set the type to **Firmware**.

   To create a custom Windows version of GoPxL, set the type to **Accelerator/Emu**.

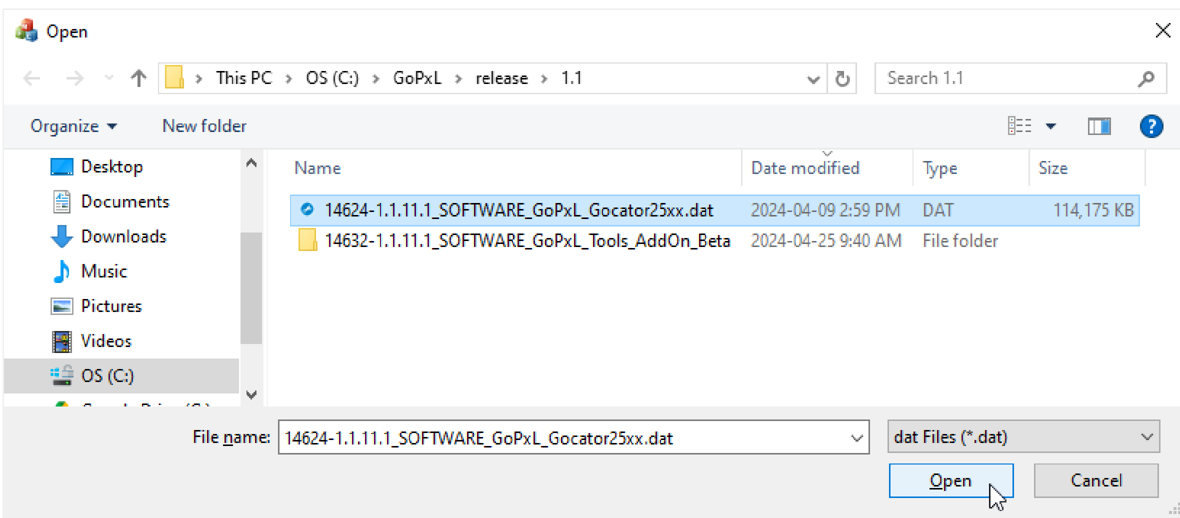


4. Below the type, click the ... button to choose the firmware or GoPxL folder.

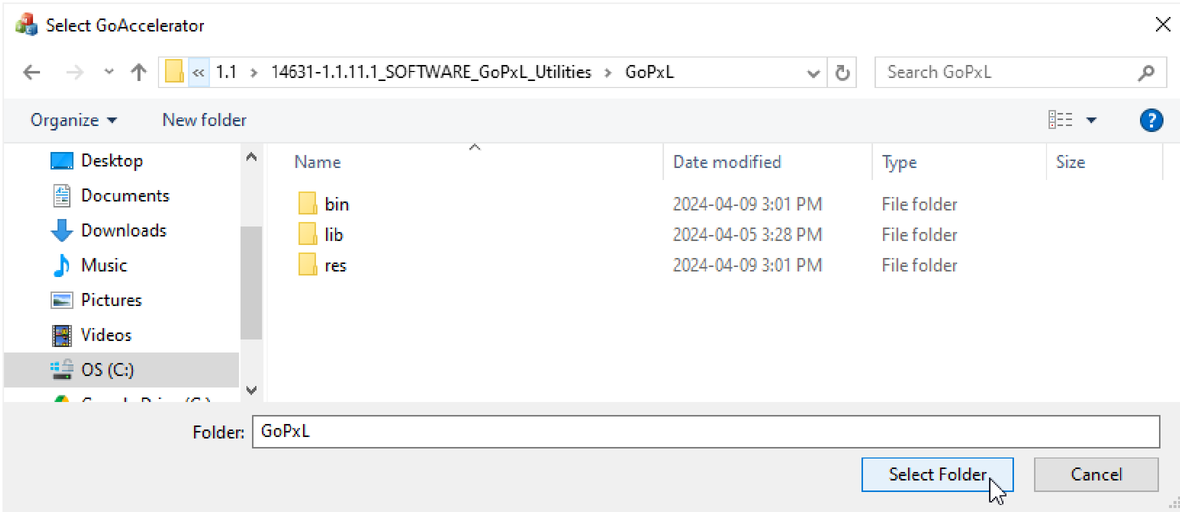


5. In the Open dialog that displays, navigate to the location of the .dat firmware file or the folder you want to use.

   Choosing a firmware:

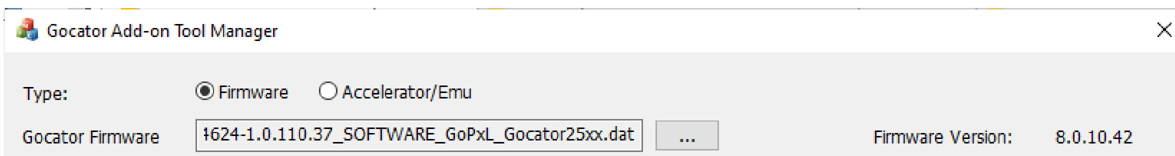


   Choosing a folder containing GoPxL for Windows:



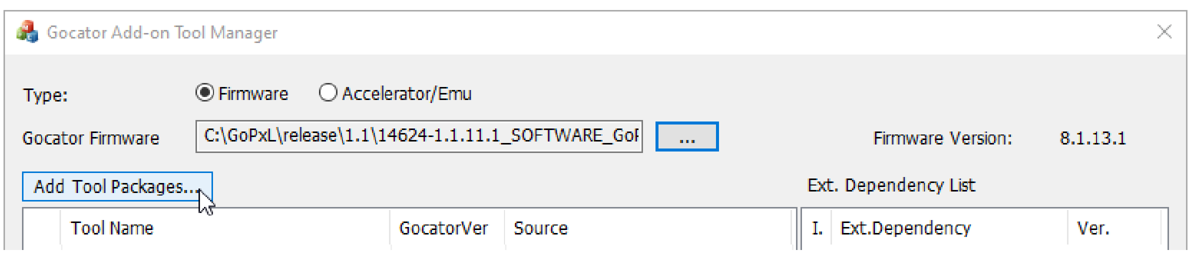
Note that you choose the `GoPxL` folder.

The tool manager loads GoPxL.

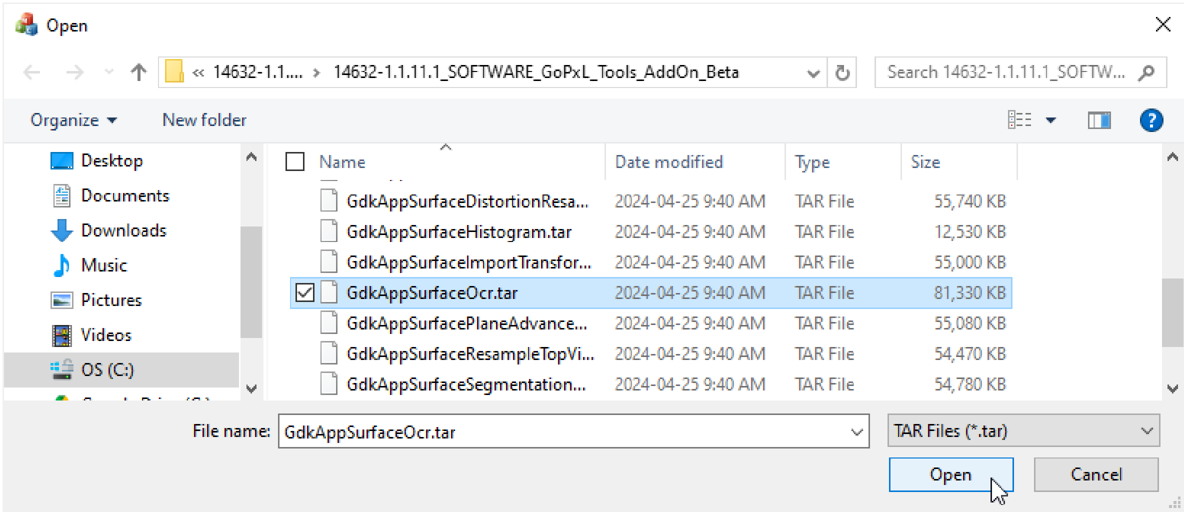


If the version of GoPxL was previously modified using the tool manager, the tools will be listed in the tool manager. For information on removing tools, see *Removing Beta Tools* on page 949.

6. Click the **Add Tool Packages...** button above the tool list.



7. In the Open dialog, navigate to the folder containing the \GoAddOn folder, select a .tar tool package, and click **Open**.

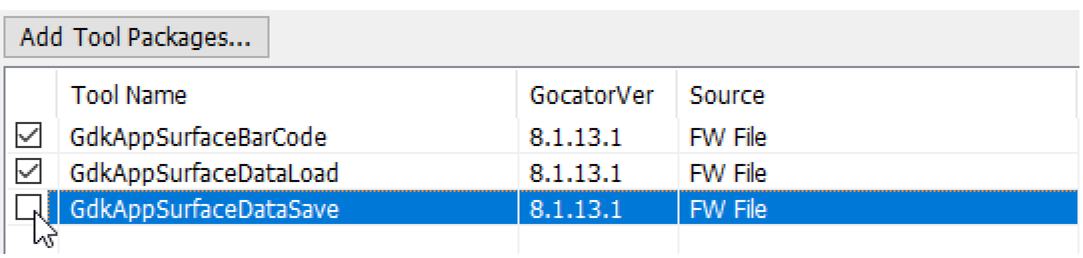


The tool manager adds the package to the tool list.

Note that you can add multiple packages at once from the Open dialog.

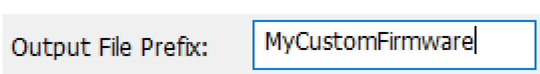
> The version of a tool package must match the version of the loaded firmware or folder. If a package's version does not match the firmware's, it is highlighted with grey in the tool list and can't be selected and included in the firmware. (The version of a tool package comes from the version of the GDK used to create it. It is not currently possible for users to create their own tool packages.).

8. When you have finished adding the packages, in the tool list, check the checkbox next the tools you want to add to the loaded firmware or folder.



> Be sure to check at least one tool. If a tool is not checked, it will not be added to GoPxL.

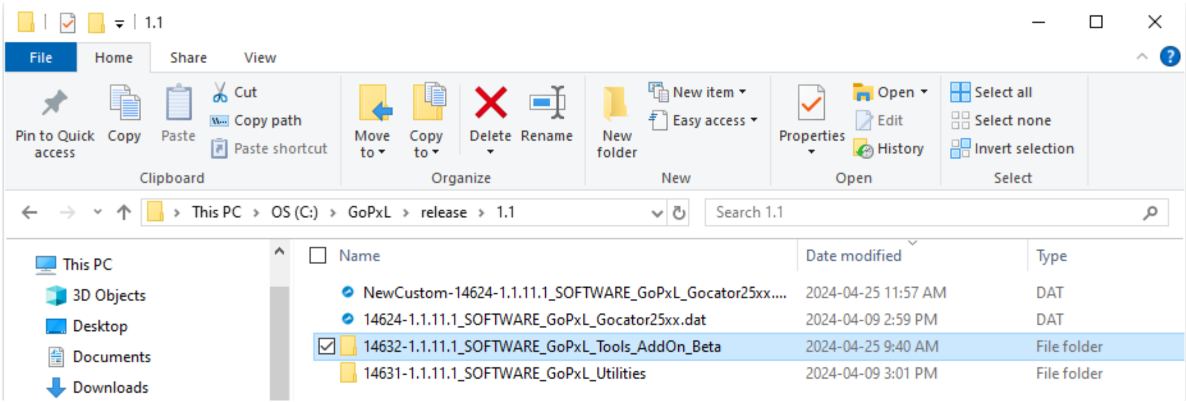
9. (Optional) If you are adding tools to a firmware, in the **Output File Prefix** field, change the default "NewCustom" to something that will help you remember what the new firmware is for.



This field is hidden when you are adding tools to GoPxL for Windows folder.

10. Click **Run**.

If you are adding tools to a firmware, the tool manager creates a new firmware file that contains the beta tools you selected, using the prefix you provided for the filename. The new firmware is created in the same location as the original firmware you loaded.



> The tool manager will overwrite an existing firmware without warning.

If you are adding tools to a GoPxL for PC folder, the tool manager updates it with the added tools.

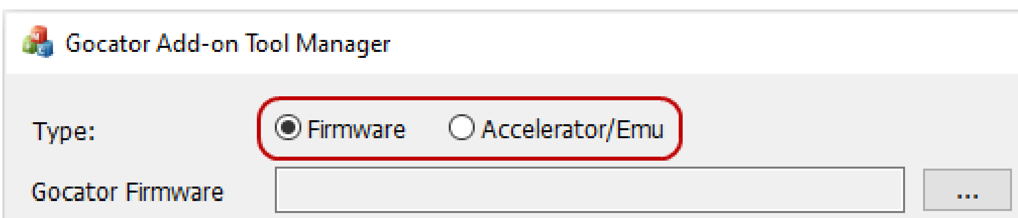
After you have successfully created the new firmware or PC version of GoPxL, you can upload it to any compatible sensor (for more information, see *Software Upgrade* on page 122) or run it (for more information, see *Running GoPxL on a Windows PC* on page 828).

Removing Beta Tools

To remove a tool you previously added to a firmware or from GoPxL for PC:

1. Follow steps 1 to 5 in *To add beta tools to a firmware or to GoPxL for PC:* on page 945

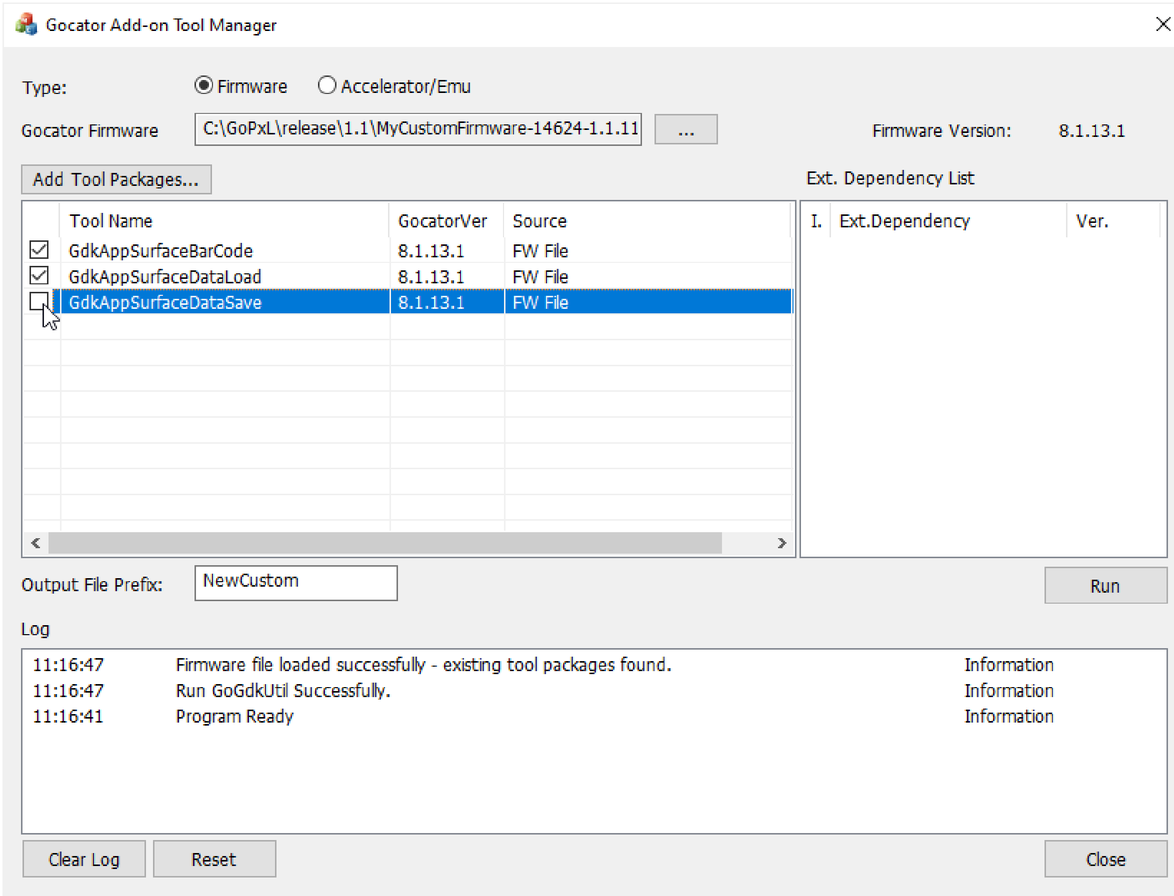
   The type should match the type used when the tool was added.

   You must navigate to the location of the previously modified GoPxL (firmware or PC version).

   After loading GoPxL, the tool list is automatically populated.

2. In the tool list, deselect the tools you want to remove.



3. Click **Run**.

   The Gocator Add-on Tool Manager saves a new version of GoPxL with the deselected tools removed.

	The tool manager overwrites existing firmware files with no warning.

After you have successfully created the new firmware or PC version of GoPxL, you can upload it to any compatible sensor (for more information, see *Software Upgrade* on page 122) or run it (for more information, see *Running GoPxL on a Windows PC* on page 828).

Pattern Editor

The pattern editor lets you modify patterns created in the Surface Pattern Matching tool (for more information on the tool, see *Surface Pattern Matching* on page 599). Although the patterns created in the Surface Pattern Matching tool will often result in good matches with your targets, you can use the pattern editor to improve the models, specifically by doing the following:

- Remove unwanted contours the Surface Pattern Matching tool has detected on edges in the scan data.
- Re-detect contours from the scan data using higher or lower levels of input image resolution (taken from the scan data) or contrast levels, compared to what the Surface Pattern Matching tool does internally.
- Identify certain contours as being required for a match to occur.
- Identify certain contours as being used to determine the position of a matched instance.

	The Pattern Editor contains a **Pattern** setting (Surface or Image). Leave this set to Surface. The Image setting is reserved for future use.

The editor uses two folders `C:\GoTools\` on the PC it is running on.

`GoPxLSurfacePatternMatchingSensor`

When you choose a sensor or PC instance of GoPxL in the **Source** drop-down and click **Connect**, the editor empties this folder and copies any model files on the source into the folder. For this reason, if you have previously worked on model from this folder, make sure you have uploaded any changed models to the sensor or PC instance. The models in this folder are listed in the Model Files until you switch **Source** to PC Work Directory, or empty it by clicking **Clear Sensor Work Directory**.

The folder is referred to as the "Sensor Work Directory" in the editor.

`GoPxLSurfacePatternMatching`

This is called the "PC Work Directory" in the editor. You typically use this folder to transfer model files between sensors or PC instances of GoPxL. Also, because it is only emptied if you manually do it by clicking **Clear PC Work Directory**, it may be better to work from this folder. To get models in this folder, you first connect to a source and then click **Copy all files to the PC work directory**.

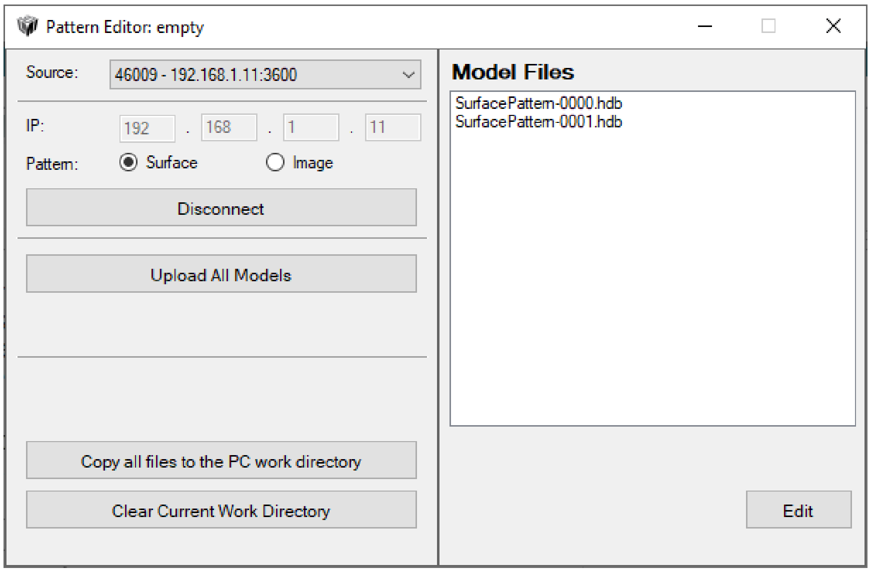
To see the files in this folder, switch **Source** to "PC Work Directory".

The pattern editor is available in the GoPxL Utilities package (`14631-x.x.x.x_SOFTWARE_GoPxL_Utilities.zip`, in the `Tools\Pattern Editor` folder. You can find the package on LMI's Product Downloads page (https://lmi3d.com/product-downloads/).

Launching the Pattern Editor

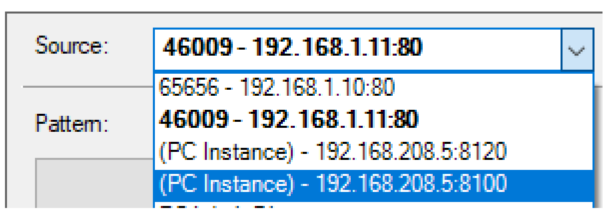
The first time you run the pattern editor, it may take longer to launch, as it will register certain DLLs required by the application.

The pattern editor can work with model files that come from an unaccelerated sensor or from a PC instance of GoPxL (which may or may not be accelerating a sensor). In all cases, the editor works with copies of the files in working folders in the local PC file system.



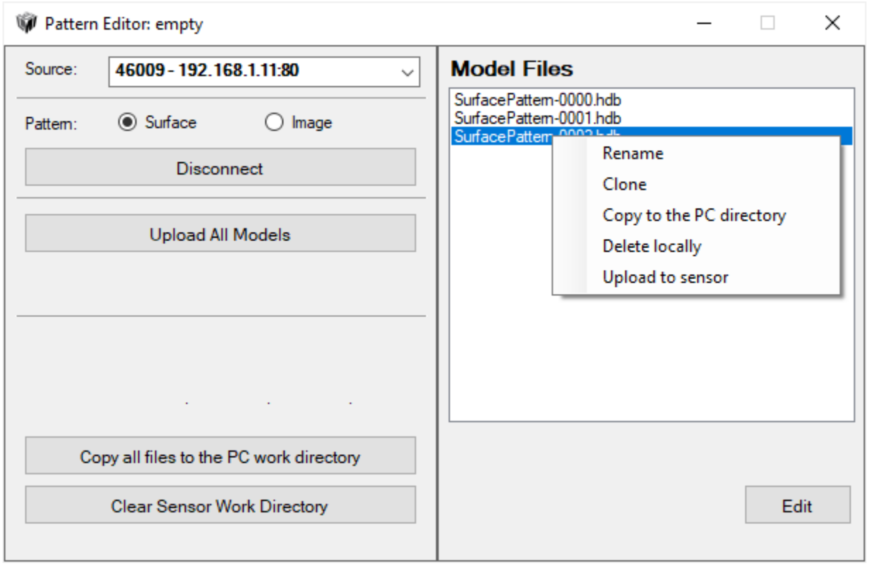
The Pattern Editor application

Parameters

Name	Description
Source	The source from which the editor fetches model files.  The currently connected sensor is displayed in bold in the **Source** list. The list contains the following types of entries: **Sensor** Sensors are listed with their serial number followed by their IP address. **PC Instances**

Name	Description
	Instances of GoPxL running on the PC are identified by "(PC Instance)", followed by their IP address. **PC Work Directory** This is a local directory (`C:\GoTools\GoPxLSurfacePatternMatching`) that you typically use to transfer model files between sensors and PC instances of GoPxL.
Pattern	Leave this set to Surface.
Connect / Disconnect	Connects to the source in **Source**. Note that connecting to a sensor deletes all models in the sensor working folder (`C:\GoTools\SurfacePatternMatchingSensor`) and then downloads the models currently on the sensor to that folder. For this reason, if you have modified models with the Pattern Editor, make sure to upload them before disconnecting and reconnecting. To fetch any models you have created on-sensor after connecting, disconnect using the **Disconnect** button, and then reconnect.
Upload All Models	Uploads the models from the `GoPxLSurfacePatternMatchingSensor` local folder to the connected source (to a sensor or a PC instance of GoPxL).
Edit	Opens the selected model in the model editor; you can also double-click a model in the list to edit it. For more information, see *Overview of the Editor* on the next page.
Copy all files to the PC work directory Copy all files to the sensor work directory	Copies files between the PC Work Directory and the Sensor Work Directory. The name of this button depends on whether **Source** is set to **PC Work Directory**, or to a sensor or PC instance of GoPxL.
Clear Sensor Work Directory Clear PC Work Directory	Removes all models from the local working folder. The name of this button depends on whether **Source** is set to **PC Work Directory**, or to a sensor or PC instance of GoPxL. When you clear the sensor work directory, you are only deleting the files locally, not on the sensor or PC instance.

By right-clicking on a model in the list of models, you can perform the operations listed below.



Rename: Renames the model.

Clone: Makes a copy of the selected model using the name you provide.

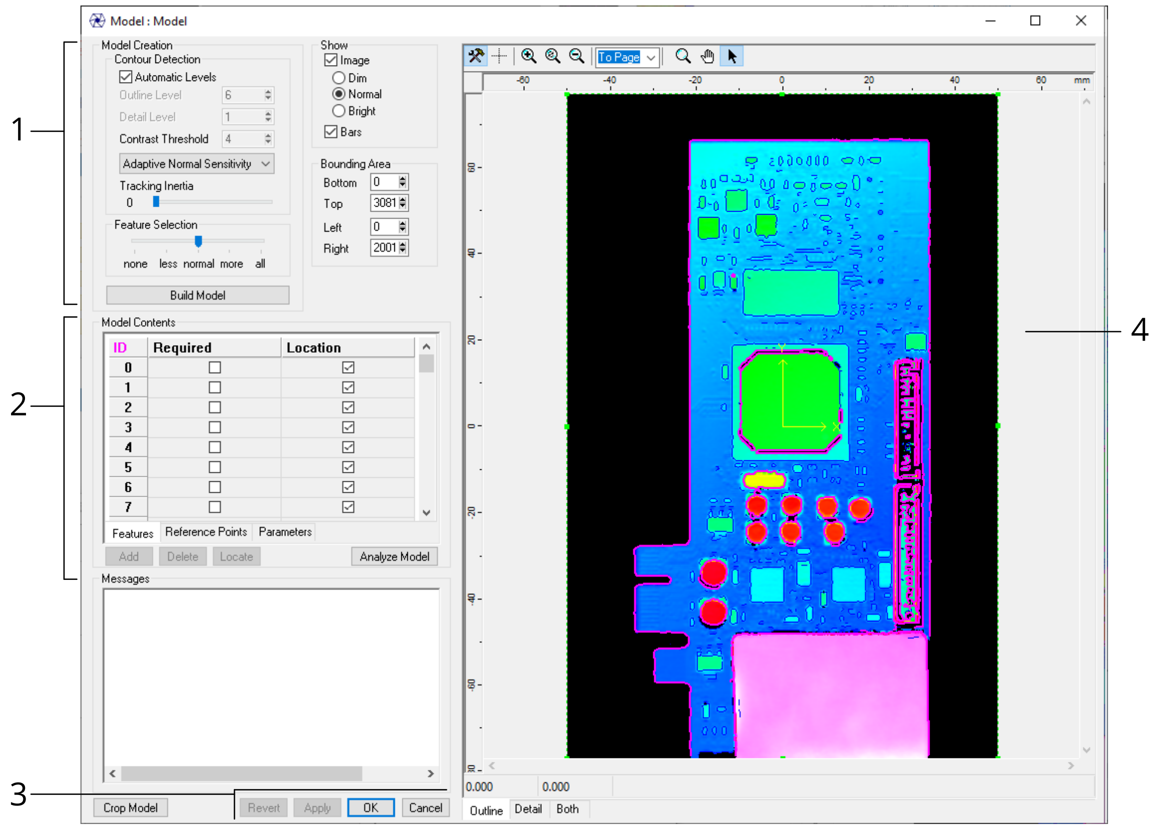
Copy to the PC directory / **Copy to the sensor directory**: Copies the model to a local directory.

Delete locally: Deletes the model from the `GoPxLSurfacePatternMatchingSensor` folder. Not displayed when **Source** is set to "PC Work Directory".

Upload to sensor: Uploads the model to the connected sensor.

Overview of the Editor

After clicking Edit in the pattern editor helper application, the selected model opens in the editor window.



	Element	Description
1	Model Creation pane	Settings related to contour detection and feature selection. After configuring these settings, or resizing the model's bounding box (green dotted line), you must rebuild the model using the **Build Model** button.
2	Model Contents	The list of the features in a model (contours used in recognition and location of an instance). Note that some model contents (reference points and some settings in the **Parameters** tab) are not currently supported by the Surface Pattern Matching tool.
3	Save and discard buttons	Used to apply changes to a model, revert to the model's original state when it was loaded, and so on.
4	Outline, Detail, and Both tabs	The editor tabs that show the Outline and Detail levels of the model. The **Both** tab shows both levels together, but you can't edit models on this tab.

Models

Models are made up of features selected from the source contours detected either by the Surface Pattern Matching tool or by the model editor itself (if you rebuild the model using the **Build Model** button). The features are used to identify and locate instances in the scan data.

Contours, and the features selected from the contours for use in recognizing and locating an instance, work on two “coarseness” levels: the Outline level and the Detail level.

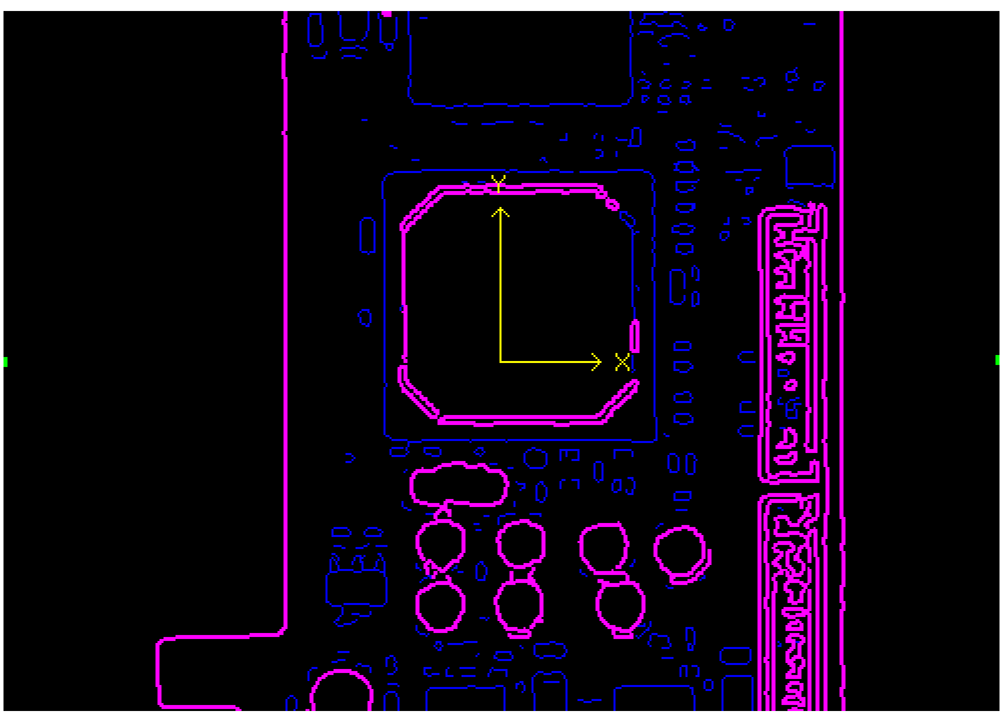
Outline: Used to quickly identify potential instances of a pattern in scan data. The Outline level is the "coarse" level of contours / features. The features at this level can be less stable, as they are not used to calculate the location of the instance. For example, a label whose position might change from frame to frame or a hole whose size might change from frame to frame could be kept at the Outline level.

Detail: Used to confirm whether an instance is in fact valid and to refine its location. The Detail level is the "fine" level of contours / features. The features at this level must be more stable and rigid with respect to one another. For this reason, given Surface scan data, include features that are all on the same plane to ensure that their positions will not be unstable due to parallax or other scanning issues. Furthermore, features on a part that might vary in size from frame to frame, or change position (such as a label), should be excluded.

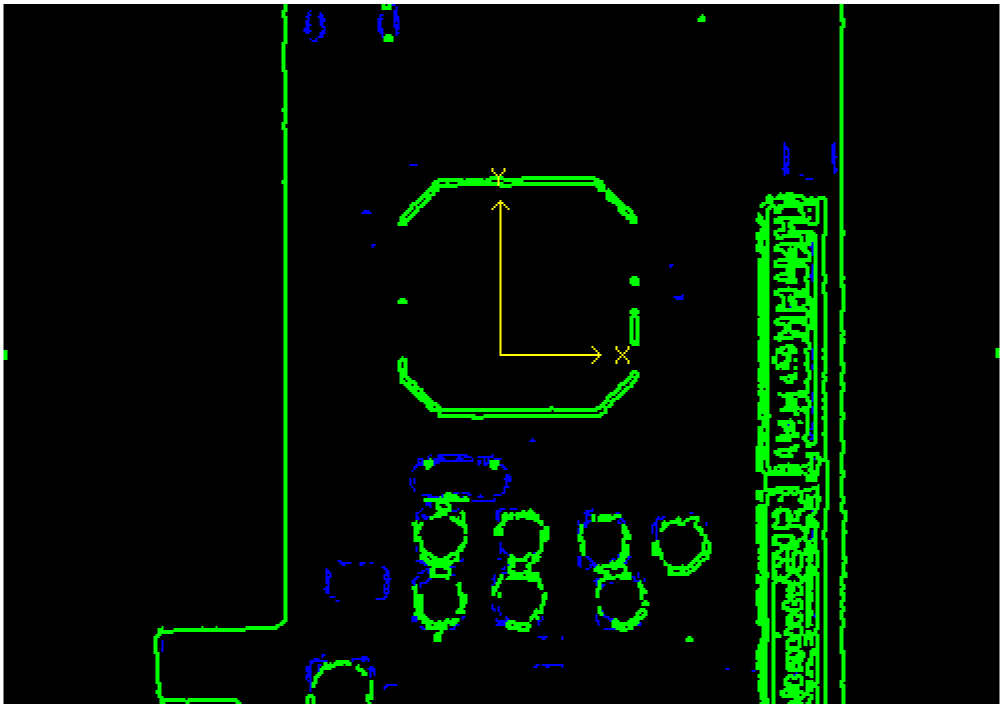
That said, the Outline and Detail levels will often be similar in terms of which features are included.

You can edit (add and remove) features at these levels separately, in the Outline and Detail panels in the main editor window. For more information on adding and removing features, see *Adding and Removing Features Manually* on the next page.

In the editor, “unused” contours (those not selected to take part in instance recognition or locating) are indicated with dark blue paths. Features (contours selected to take part in instance recognition or locating) are indicated with either magenta paths (at the Outline level) or with green paths (at the Detail level); the features in a model are listed in the **Model Contents** pane.



Dark blue unused contours and magenta features at the Outline level.



Dark blue unused contours and green features at the Detail level.

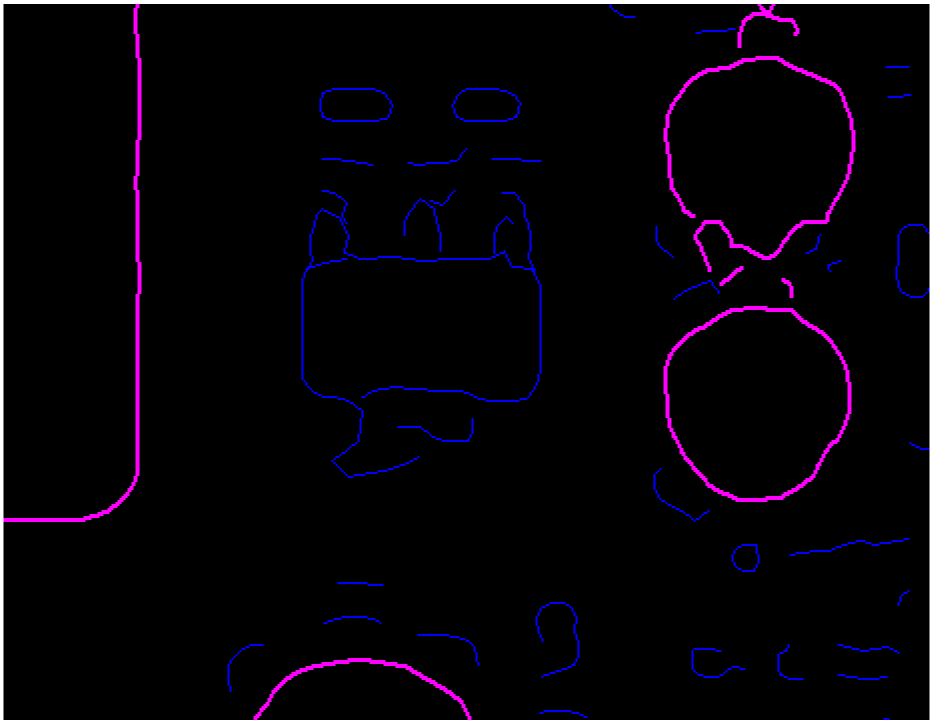
In the Surface Pattern Matching tool, only the Detail level of features is displayed.

Adding and Removing Features Manually

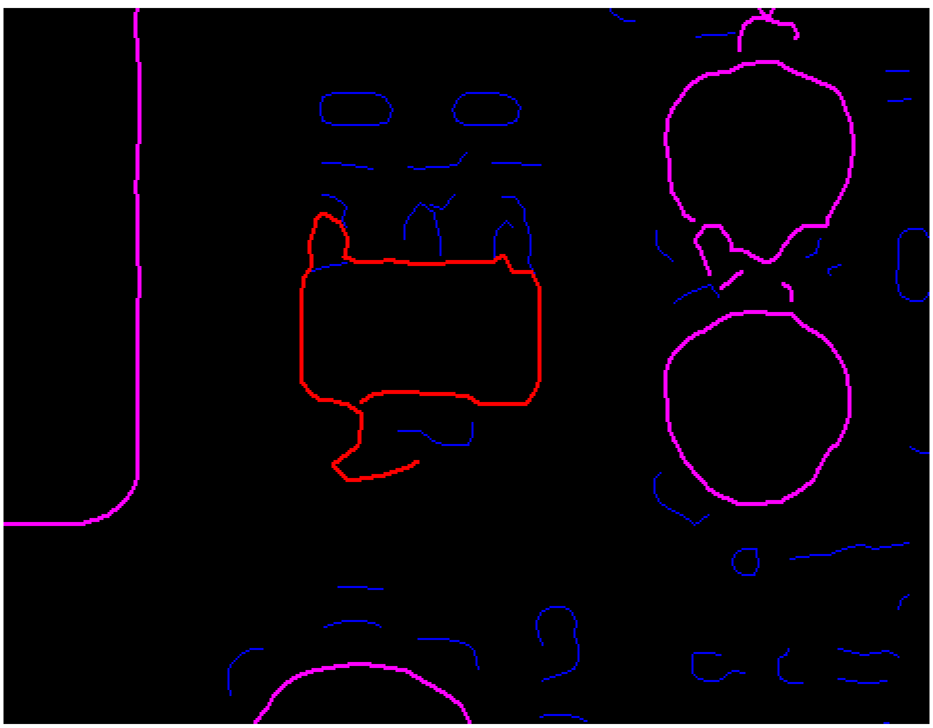
You can manually add features to a model from the source contours, or remove features currently in a model, at both the Outline and Detail levels. This can be useful if the model produced by the Surface Pattern Matching tool includes features related to parts of targets that could change or be present/absent from frame to frame. You should only include features that are constant from frame to frame.

Adding and removing features works in the same way in the Outline and Detail tabs.

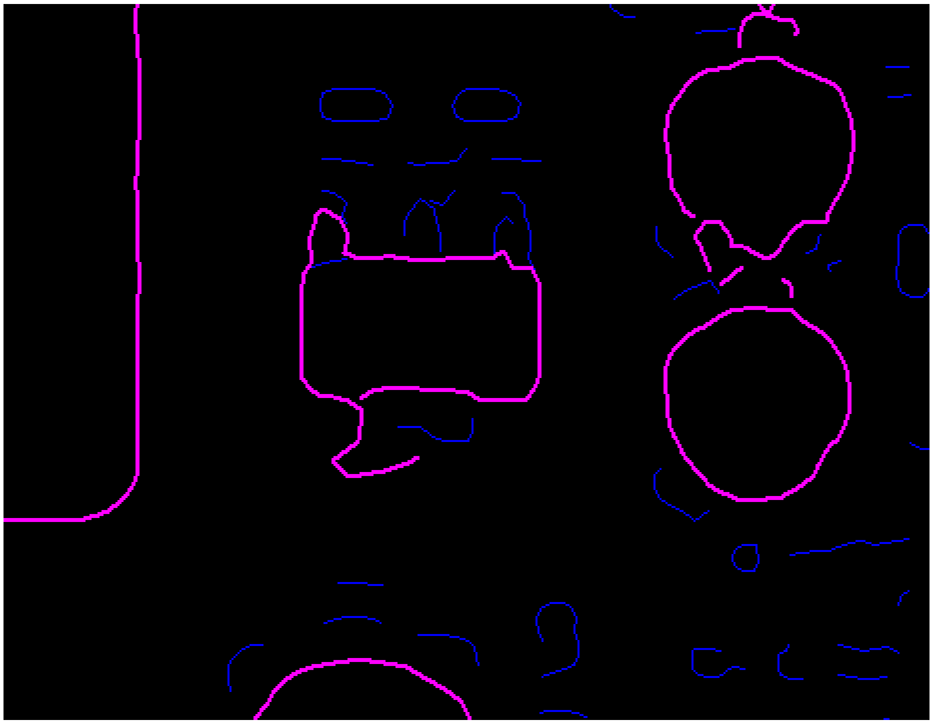
To add a feature from a source contour, double-click a dark blue contour in either the Outline or Detail tab and click Add or press the Insert key on your keyboard.



Dark blue unused contour (contours already added as features in the model are magenta).

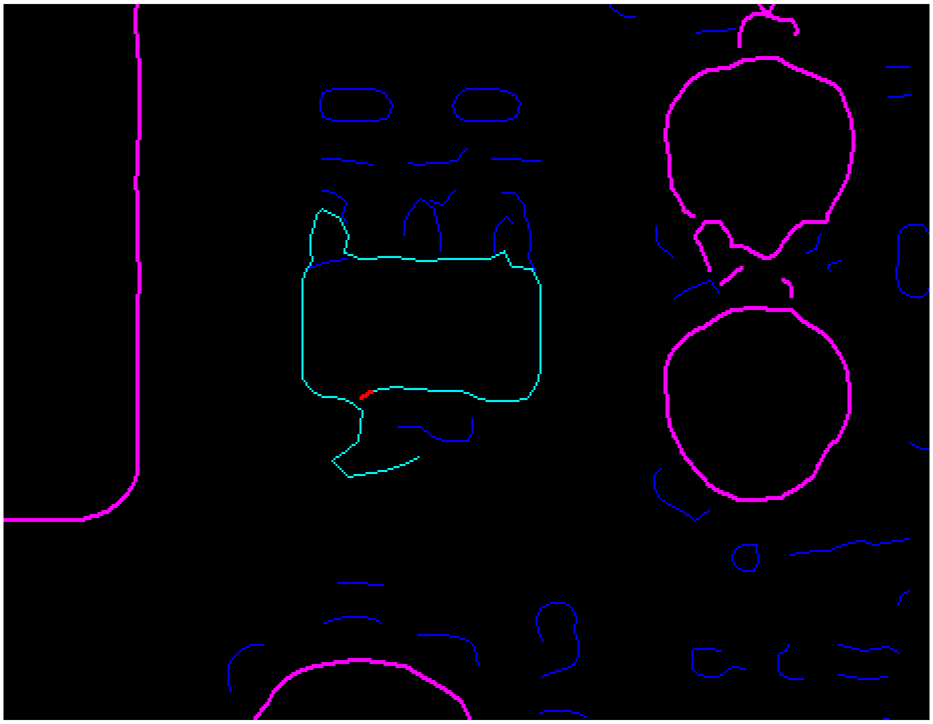


Contour selected by double-clicking it.



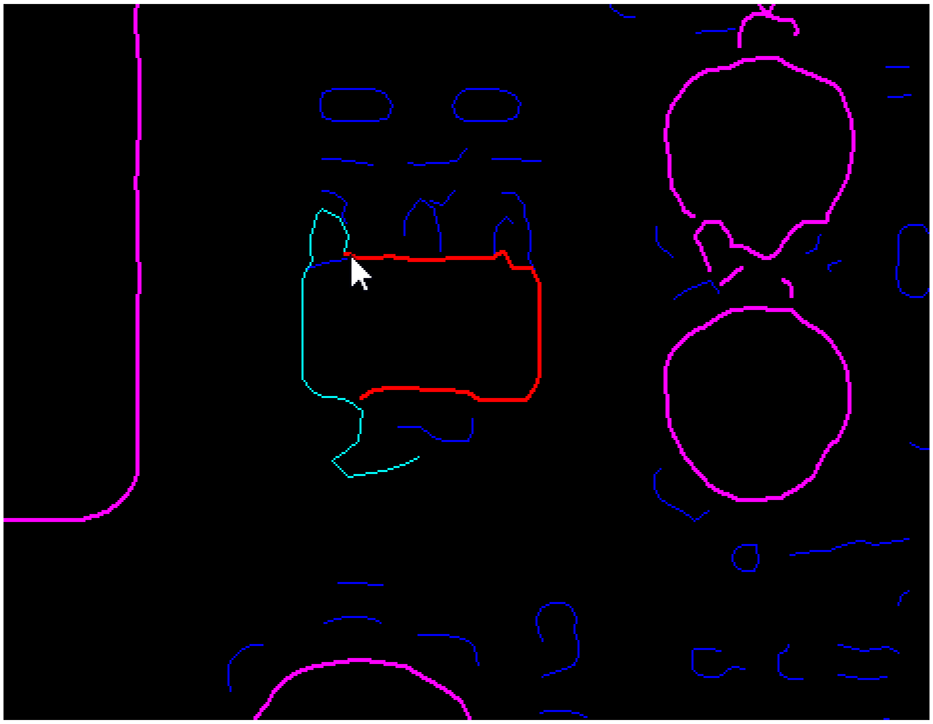
Contour added as a feature in the model (magenta).

If you single-click a dark blue unused contour, it turns cyan and lets you select segments of the contour.

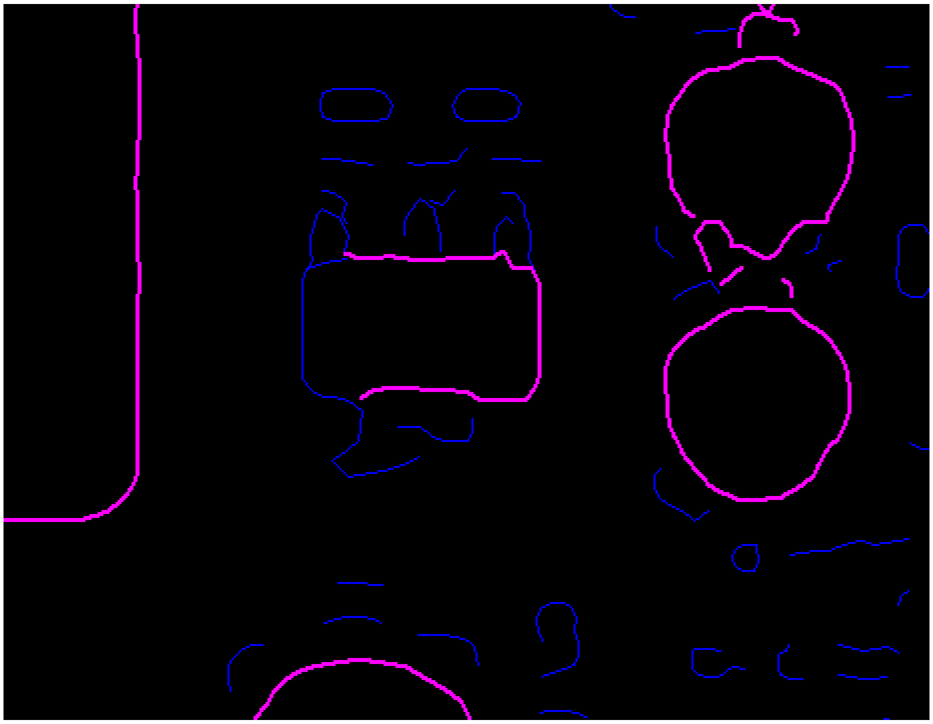


Contour highlighted in cyan with a selected segment (red).

Pressing the Ctrl key on your keyboard and clicking another segment selects a portion of the cyan path.



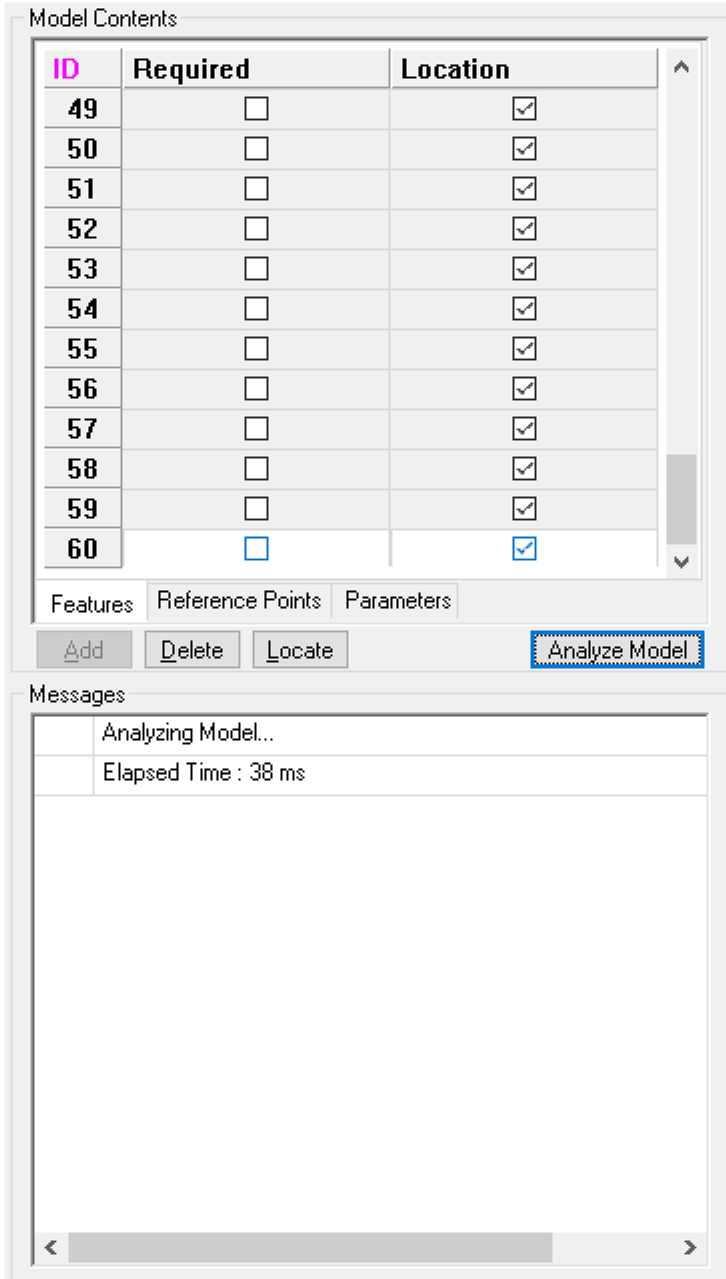
Clicking **Add** or pressing the Insert key on your keyboard adds the segment of the contour as a feature to the model.



After adding a feature, it is added to the list of features on the **Feature** tab in the **Model Contents** panel. Do not press the build button after adding or removing features, or you will lose the modifications that were just made. You must however save the changes; for more information, see *Saving and Discarding Changes* on page 967.

To remove a feature, click a magenta or green path in the editor to select it and click **Delete** or press the Delete key on your keyboard. After removing a feature, it is removed from the list of features on the Feature tab in the Model Contents panel. You do not need to build the model, but must save the changes; for more information, see *Saving and Discarding Changes* on page 967.

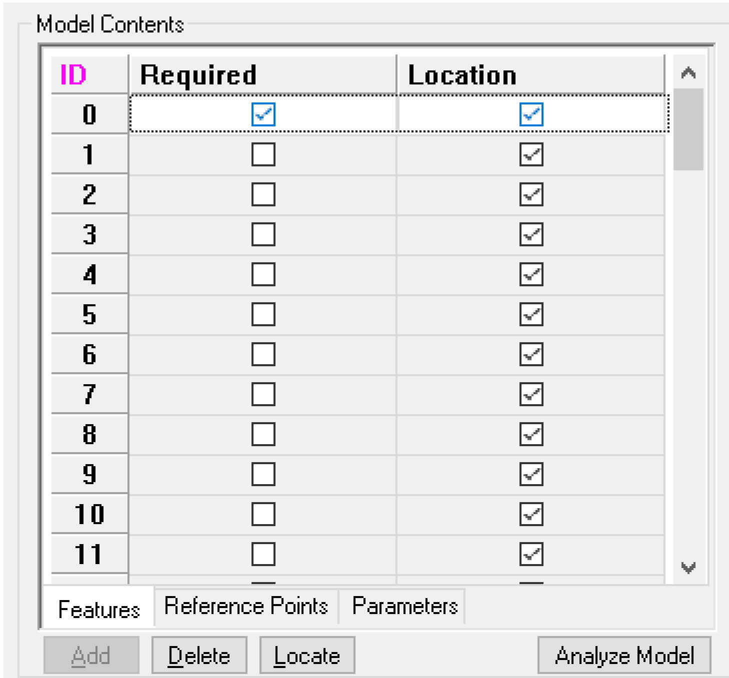
After adding a feature to a model or removing a feature from a model, you should analyze the model by clicking **Analyze Model**. Make note of any errors in the **Messages** panel.



No error messages after clicking Analyze Model.

Setting Required and Locating Features

In the list of features in the **Model Contents** pane, you can indicate that a feature is "required" or that it is used to calculate the location of an instance by checking the appropriate checkbox next to the feature.



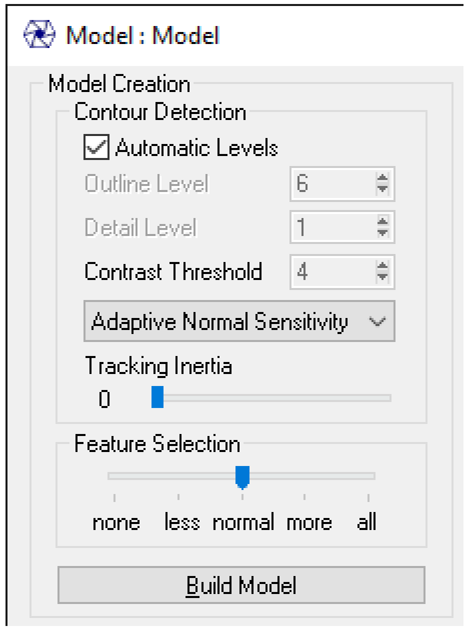
When Required is checked for a feature, it *must* be found by the Surface Pattern Matching tool in order for an instance to be identified.

When Location is checked for a feature, the Surface Pattern Matching tool uses the feature to calculate the location of instances. If a feature’s location is not checked, it is only used for instance recognition. An example of the latter is a tag or label glued to an object. Although the label's contours (it's shape or what is written on it) might be unique enough to help recognize an instance, it's position *on* the object (that is, relative to the other features) might vary in its position from frame to frame. For this reason, it might be useful for instance recognition, but not for determining the location of the object.

Model Creation Settings and Rebuilding

The Surface Pattern Matching tool uses internally fixed settings to detect contours in the scan data and then select features from those source contours. In the pattern editor, you can increase or decrease the contour detection levels, change the contrast threshold, and so on, and then rebuild the model. This can be useful if the Surface Pattern Matching tool’s internal settings do no produce the right amount of source contours and subsequently features.

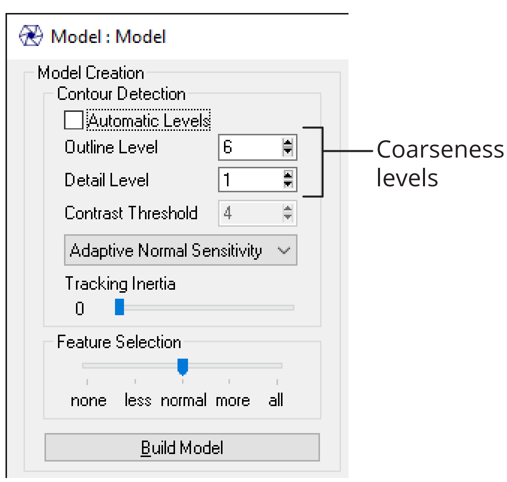
The settings described here are found in the **Model Creation** section of the model editor.



After making changes to any of these settings, you must rebuild the model by clicking **Build Model**, and then save the changes. You should also click **Analyze Model** after rebuilding a model. Pay special attention to messages in the **Messages** pane at the bottom of the editor to make sure there are no errors. For more information on saving changes, see *Saving and Discarding Changes* on page 967.

Coarseness Levels

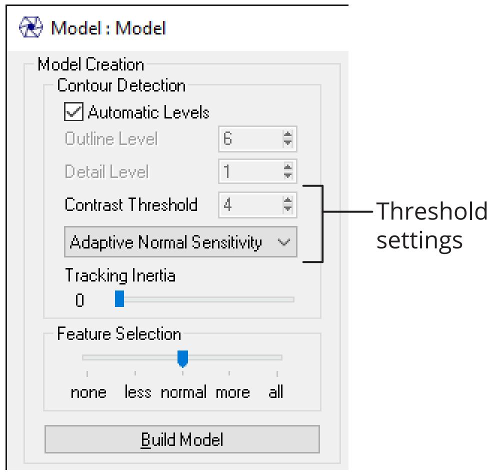
By default, the pattern editor uses automatically determined contour coarseness values (at both the Outline and Detail levels) to detect contours in the scan data image. If you uncheck **Automatic Levels**, you can change the **Outline Level** and **Detail Level** values to generate more or fewer contours, from which you can then select features that more reliably represent your target.



The **Outline Level** and **Detail Level** values range from 1 to 16. At the lowest value, contours are detected in a full-resolution version of the image based on the scan data, which results in more contours from which to choose features. At higher values, contours are detected in a reduced-resolution version of the image based on the scan data: the resolution is reduced by the setting's value, which results in fewer contours being detected. Note that **Detail Level** must be less than or equal to **Outline Level**.

Thresholds

You can adjust the level of sensitivity the pattern editor uses to detect contours in the scan data image.



By default, the sensitivity is set to Adaptive Normal Sensitivity, but you can set it to one of the following adaptive sensitivity levels, or to a fixed threshold value (see below).

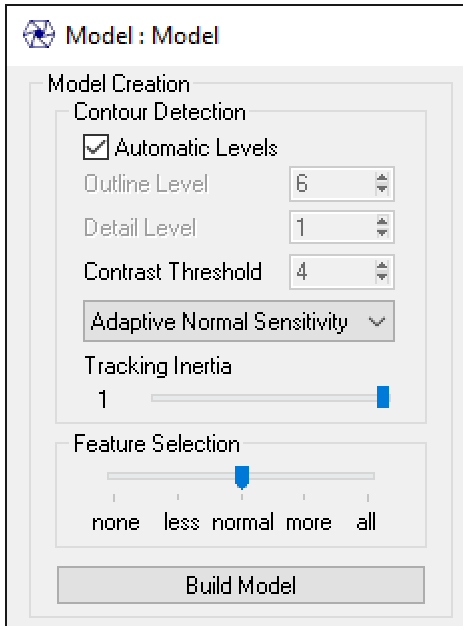
Adaptive High Sensitivity: Results in more low-contrast contours, but also noise.

Adaptive Low Sensitivity: Results in strongly defined contours and eliminates noise, but may miss important contour segments.

If you set the dropdown to Fixed Value, you can then set a fixed threshold in **Contrast Threshold**. The **Contrast Threshold** value corresponds to the minimum step required to detect corners. A lower value generates more contours when you rebuild the model, but may also result in noise.

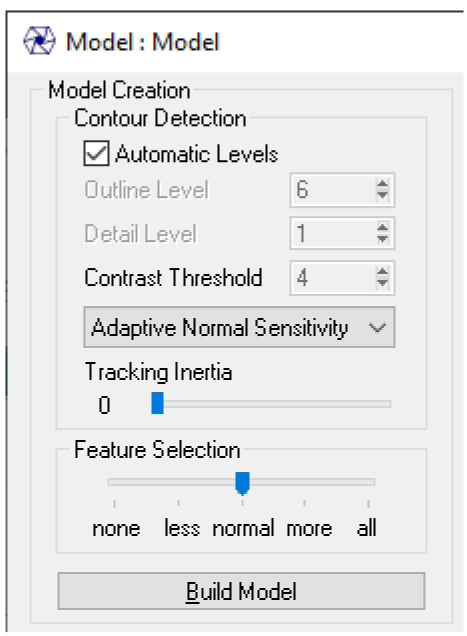
Tracking Inertia

Setting the Tracking Inertia slider to 1 closes small gaps in the source contours, connecting contours that might otherwise be broken into smaller sections.



Feature Selection

This setting ranges from **none** to **all**, which determines which features the pattern editor selects from the detected contours and adds to the model when you rebuild it. You should use **none** (which adds no features to the model) if you want to manually add features to the model from the detected contours. The **normal** setting tries to add the most appropriate features to the model; use this setting with simple to moderately complex parts. The **all** setting adds all detected contours as features to the model; only use this with very complex parts, such as electronic parts.



Saving and Discarding Changes

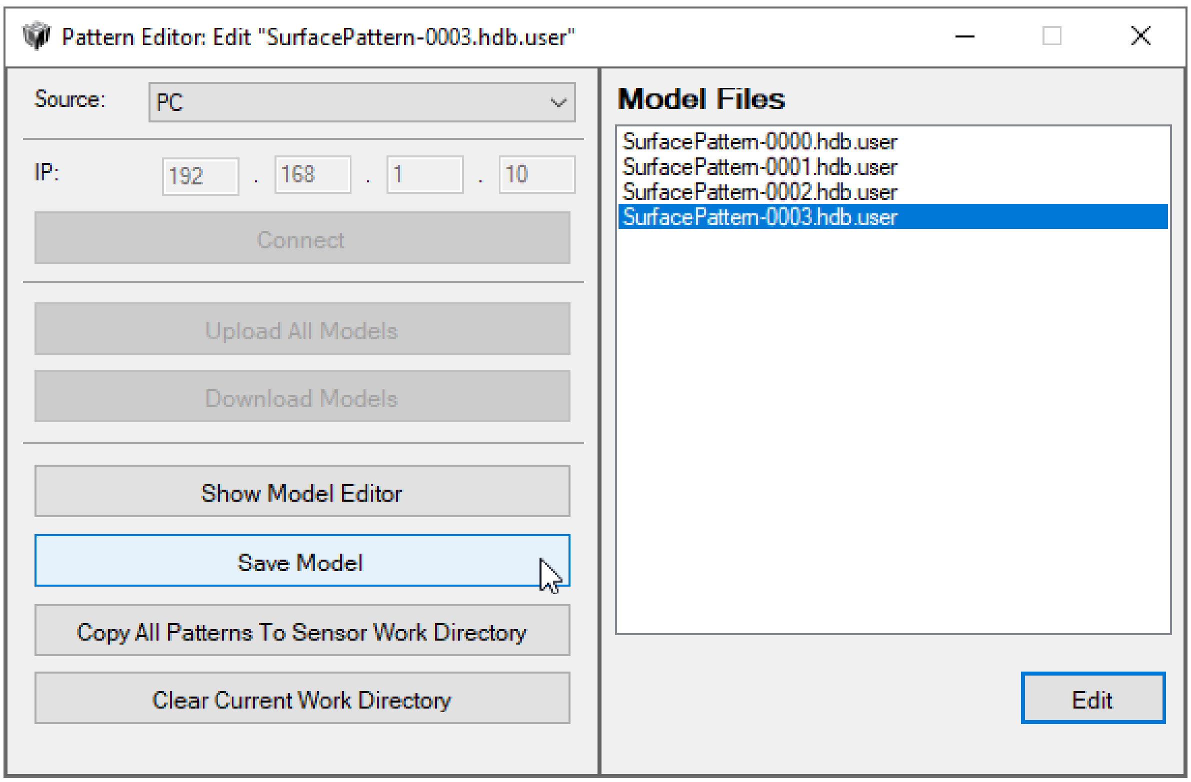
After making changes to a model (either adding or removing features, or re-detecting contours by clicking **Build Model**), you *must* do the following:

1. In the model editor, at the bottom of the window, click **Apply** or **OK**.

   Clicking **Apply** leaves the model editor open. Do this if you want to continue working on a model (for example, if you want to test the model in Gocator before closing the model).

   Clicking **OK** closes the model editor.

2. In the pattern editor helper application, click the model you were working on, and click **Save Model**.



   The model is saved to the working folder.

After making changes to a model, you must re-load the pattern in the instance of Surface Pattern Matching to see the changes.

Before saving changes (either by clicking **Apply** or **OK**), you can revert the model to its initial state by clicking **Revert**.

Miscellaneous

Reference points, which you can create in the pattern editor on the **Reference Points** tab in the **Model Contents** panel, are not currently supported by the Surface Pattern Matching tool.

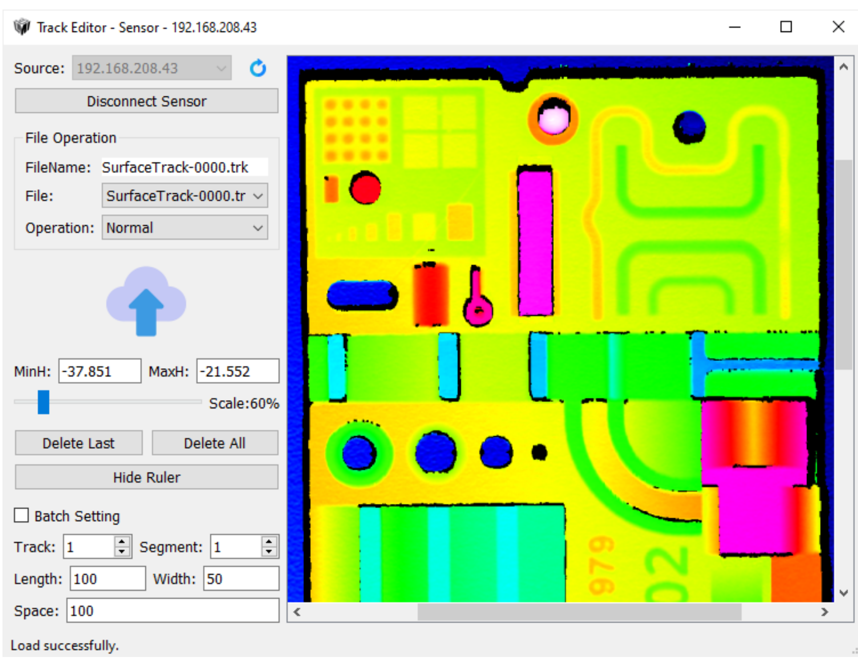
The Surface Pattern Matching tool does not currently support the custom shading area (on the **Parameters** tab in the **Model Contents** panel).

You can change the region of the model by resizing the green bounding box (or setting its dimensions in the **Bounding Area** section in the model editor) and rebuilding the model. After resizing and rebuilding the model to the desired region, you can use the **Crop Model** button to reduce the size of the final model surface, and reduce the pattern file size.

In the Show section in the model editor, you can hide the scan data to see only the contours and features by unchecking **Image**. Note that the Dim / Normal / Bright options below **Image** only apply to intensity data.

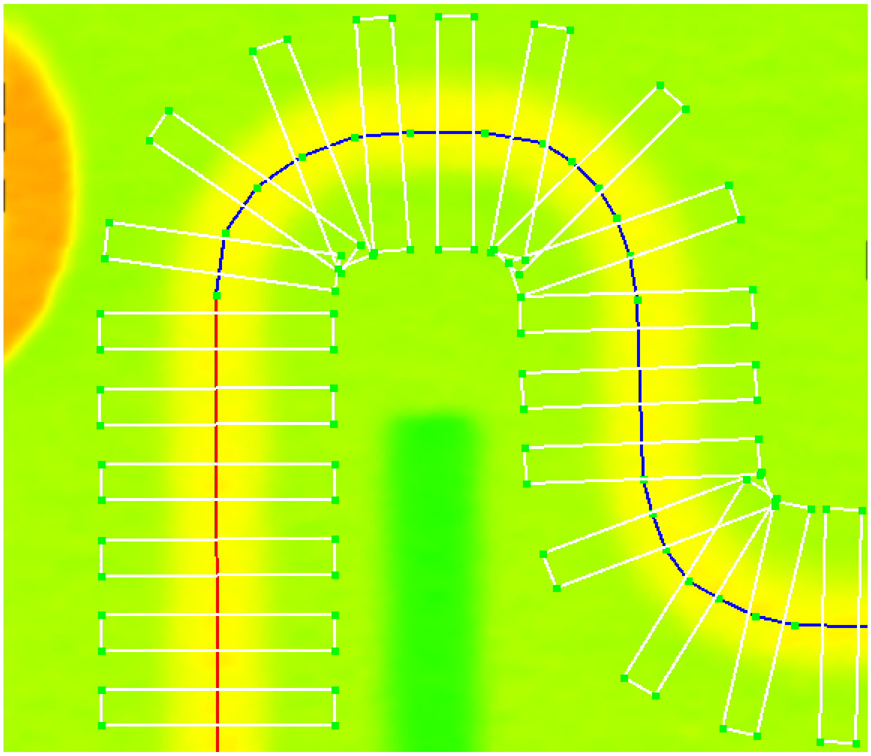
Track Editor

You use the track editor in conjunction with the Surface Track tool. The track editor lets you configure "path" and "ruler" information on a frame of scan data from a sensor or GoMax NX device, a PC instance of GoPxL, or a GoPxL backup file. The Surface Track tool uses this information to inspect targets along the defined path. For more information on paths, rulers, and other concepts related to the Surface Track tool, see *Surface Track* on page 661.



The track editor

In the track editor, you define one or more paths, and configure rulers along these paths.

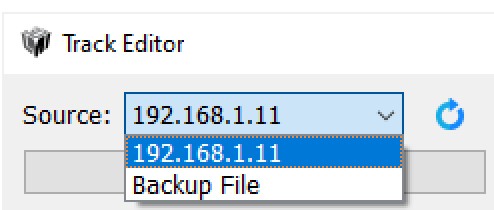


Closeup of the track editor window, showing a track of material on a surface (yellow on green), path points (green dots), a path (blue segments; red segment for the currently selected segment), and rulers (white rectangles).

	The following assumes that you have already scanned a representative target and created a track file from within the Track tool. For more information, see the first steps of *To configure the Track tool:* on page 665.

To load and work with scan/track data:

1. In the track editor, in the **Source** drop-down, choose one of the following:



- **An IP address**: Choose this option to edit a track file on a sensor or a GoMax NX device, or in a PC instance of GoPxL.
- **Backup File**: Choose this option to edit a track file in a backup file you have created from a sensor or a PC instance of GoPxL and saved in the PC's file system. For more information on backup files, see *Backup and Restore* on page 122.

	If you only see "Backup File" but expect to see an IP address, make sure the device or PC instance is running and is connected to your network, and then click the Reset button ( ).

2. If you have chosen an IP address, click **Connect Sensor**.

The track editor loads the data from the device or PC instance.

3. If you have chosen a backup file, click **Load Backup**, navigate to the location of the backup file, select it, and click **Open**.

   The track editor loads the data.

4. Under **File Operation**, in **File**, select the track file you have created in the Surface Track tool, and under **Operation**, choose Load.

> If you only see "External" under **File**, it means no track file has been defined on the device / PC instance or in the backup file. You can use "External" to load a .trk file from the PC's file system. Otherwise, create a track file in the Surface Track tool, and then disconnect and reconnect the source.

   The editor loads the specified track file.

   For a complete list of the file operations, see *File operations* below.

5. Do one or more of the following:
   - Move the slider to the left or right to zoom in or out in the editor's viewer.
   - Move the data in the track editor's window using the scrollbars or the mouse wheel.
   - Set **MinH** and **MaxH** and then reload the track data to assign a narrower height range to the height map colors. This may help make the track clearer in the editor. This does not change the height map data.

File operations

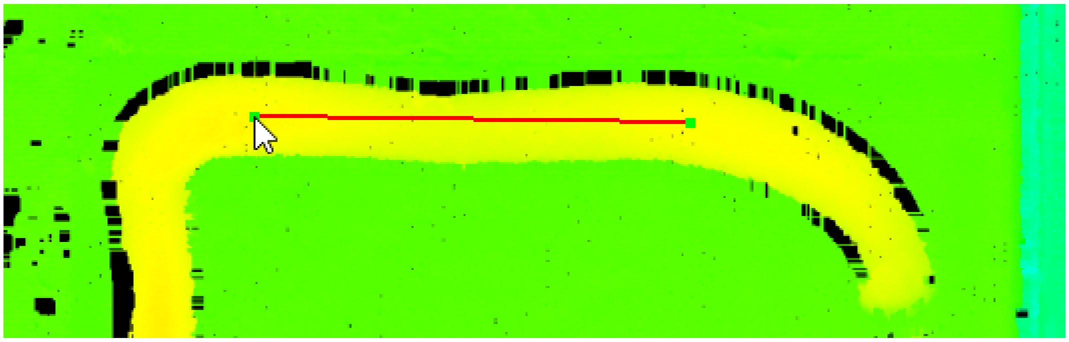
Operation	Description
Normal	The utility defaults to this option after performing another operation.
Load	Loads the track file selected in the **File** list. If you choose "External" in the **File** list, choosing Load displays a dialog that lets you load a file on the PC.
Save	Saves the currently loaded track file (as shown in **FileName**).
Save As	Saves the currently loaded track file (as shown in **FileName**) as a new file.
Delete	Deletes the file selected in the **File** list.
Refresh	Refreshes the **File** list.

After you have loaded the data, you must add a path and configure its rulers.

To add and configure a path and its rulers:

1. In the track editor, click on the middle of the track somewhere in the scan data, move the mouse pointer to another location, and click again.

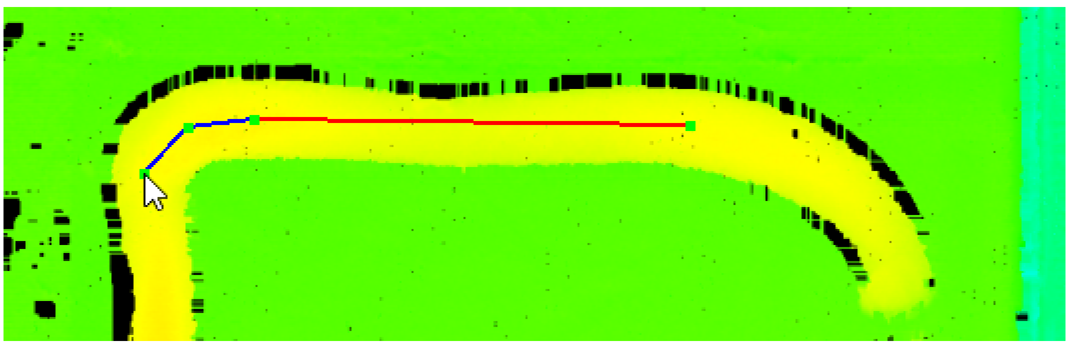
   A red segment between the first two green path points appears in the editor window.



You can move path points using the mouse at any time to adjust the path. You can also delete the last point by clicking **Delete Last**. To delete all path points, click **Delete All**.

2. Continue clicking along the track to add more path points, building up the path.

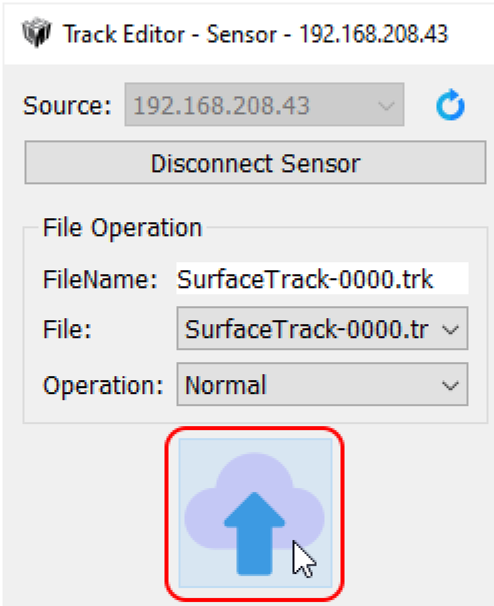
   When adding points on corners, add more points to follow the track more precisely.



3. Continue clicking until you complete the path along the track.
4. Modify the ruler settings.

   For more information, see *Ruler Settings* on the next page.
5. Under **Operation**, click **Save** to save the path information to the data.

   For a complete list of the file operations, see *File operations* on the previous page.
6. Click the Sync button.

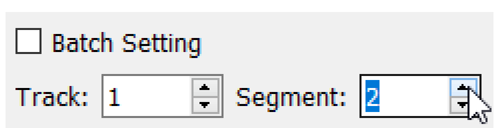


The changes are synchronized to the source (device or backup file).

Ruler Settings

After you have finished adding a path, you can configure the rulers on the path (the dimensions and the spacing of the rulers). You can choose to apply dimensions/spacing to rulers in all segments at the same time by checking **Batch Setting**.

Otherwise, you must move through the individual path segments by clicking the spinner control in the **Segment** field and setting the ruler dimensions for each segment.



The following table lists the ruler settings available in the track editor:

Track editor: ruler settings

Setting	Description
Length	The dimension of the ruler perpendicular to the path. Be sure to use a value large enough to cover the track from one side to another and to include enough surface on each side of the track (the surface to which the material is applied) for the Track tool to properly detect the track.
Width	The dimension of the ruler along the path.
Space	The space between rulers on the path. Because you will typically place path points closer together around corners, you may need to use smaller spacing around corners.

Specifications

The following sections describe the specifications of Gocator sensors and connectors, as well as Master hubs.

Sensors

The following sections provide the specifications of Gocator sensors.

Gocator 2300 Series

The Gocator 2300 series consists of the following models.

Unless otherwise specified, the wavelength of all models is 660 nanometers.

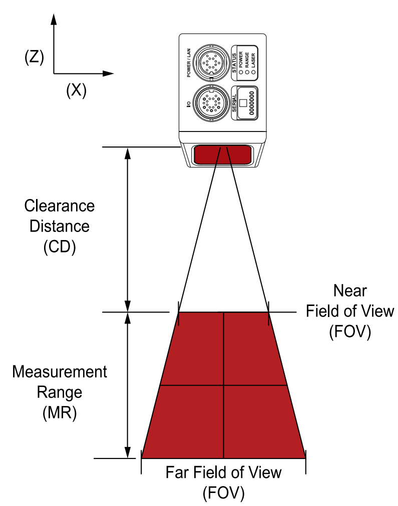
The following models are not typically industry-specific and can be used for various applications.

MODEL	2320	2330	2340	2350	2370	2375	2380
Data Points / Profile	1280	1280	1280	1280	1280	1280	1280
Linearity Z (+/- % of MR)	0.01	0.01	0.01	0.01	0.04	0.03	0.04
Resolution Z (mm)	0.0018 - 0.0030	0.006 - 0.014	0.013 - 0.037	0.019 - 0.060	0.055 - 0.200	0.175 - 0.925	0.092 - 0.488
Resolution X (mm) (Profile Data Interval)	0.014 - 0.021	0.044 - 0.075	0.095 - 0.170	0.150 - 0.300	0.275 - 0.550	0.255 - 0.790	0.375 - 1.100
Repeatability Z (µm)	0.4	0.8	1.2	2	8	12	12
Clearance Distance (CD) (mm)	40	90	190	300	400	650	350
Measurement Range (MR) (mm)	25	80	210	400	500	1350	800
Field of View (FOV) (mm)	18 - 26	47 - 85	96 - 194	158 - 365	308 - 687	324 - 1010	390 - 1260
Laser Classes	2, 3R	2, 3R; 3B	2, 3R; 3B	2, 3R; 3B	2, 3R	2, 3R	2, 3R
Input Voltage (Power)	+24 to +48 VDC (13 W); Ripple +/- 10%						
Dimensions (mm)	35x120x149.5	49x75x142	49x75x197	49x75x272	49x75x272	49x75x272	49x75x272
Weight (kg)	0.8	0.74	0.94	1.3	1.3	1.3	1.3

The following models are typically used in road applications.

MODEL	2342	2375
Data Points / Profile	1280	1280
Linearity Z (+/- % of MR)	0.01	0.05
Resolution Z (mm)	0.015 - 0.040	0.154 - 0.560
Resolution X (mm)(Profile Data Interval)	0.095 - 0.170	0.27 - 0.80
Repeatability Z (μm)	1.2	n/a
Clearance Distance (CD) (mm)	190	650
Measurement Range (MR) (mm)	210	1350
Field of View (FOV) (mm)	64 - 140	345 - 1028
Laser Classes	3R, 3B	3B (NIR, 808 nm)
Input Voltage (Power)	+24 to +48 VDC (13 W); Ripple +/- 10%	+48 VDC (20 W); Ripple +/- 10%
Dimensions (mm)	49x75x197	49x75x272
Weight (kg)	0.94	1.3

The following diagram illustrates some of the terms used in the table above.



Optical models, laser classes, and packages can be customized. Contact LMI for more details.

Gocator 2300 series sensors provide 1280 data points per profile. Gocator 2100 series sensors provide 640 data points per profile.

Specifications stated are based on standard laser classes. Linearity Z, Resolution Z, and Repeatability Z may vary for other laser classes.

All specification measurements are performed on LMI's standard calibration target (a diffuse, painted white surface).

Linearity Z is the worst case difference in average height measured, compared to the actual position over the measurement range.

Resolution Z is the maximum variability of height measurements across multiple frames, with 95% confidence.

Resolution X is the distance between data points along the laser line.

Repeatability Z is measured with a flat target at the middle of the measurement range. It is the 95% confidence variation of the average height over 4096 frames. Height values are averaged over the full FOV.

See *Resolution and Linearity* on page 62 for more information.

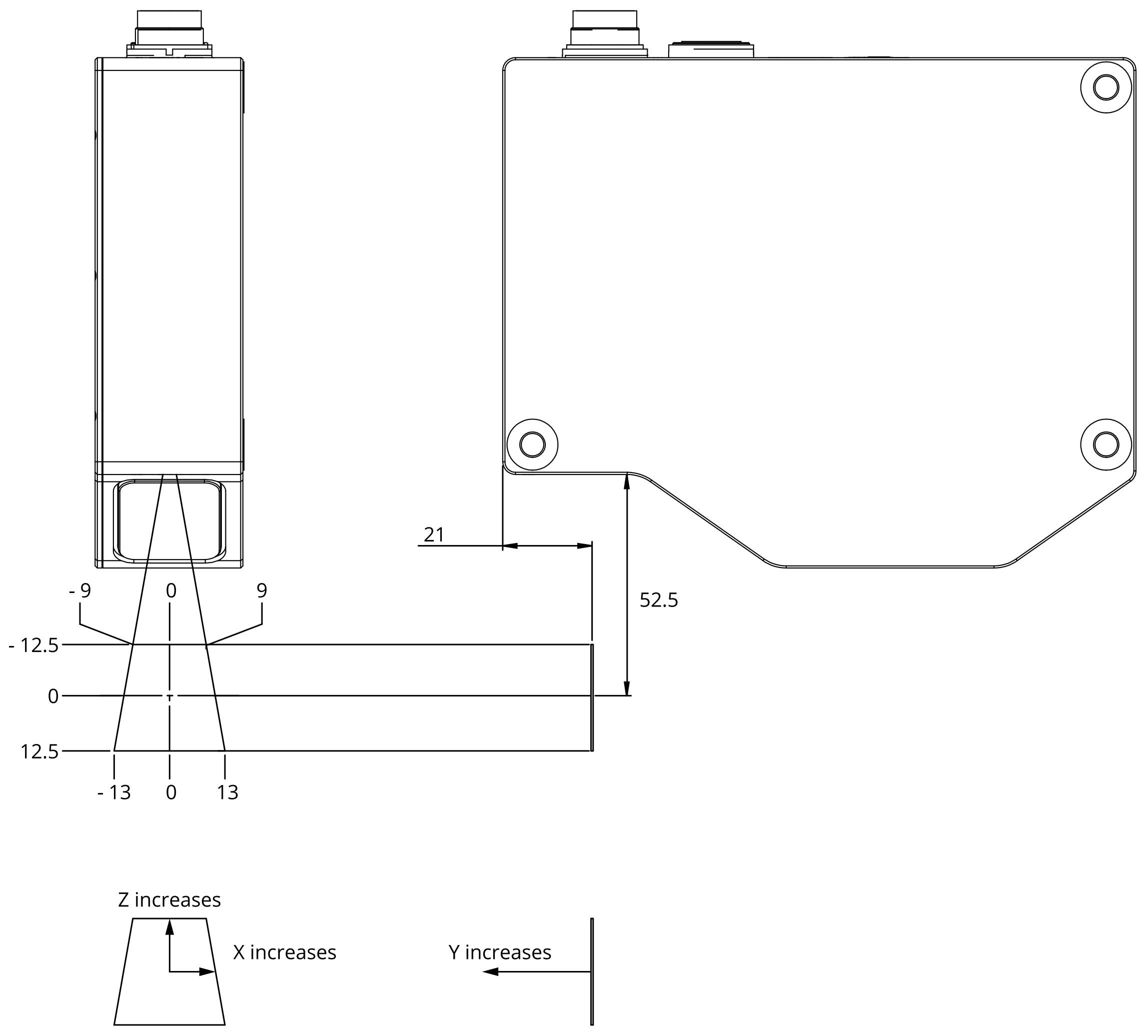
ALL 2x00 SERIES MODELS	
Scan Rate	Approx. 170Hz to 5000 Hz
Interface	Gigabit Ethernet
Inputs	Differential Encoder, Laser Safety Enable, Trigger
Outputs	2x Digital output, RS-485 Serial (115 kBaud), 1x Analog Output (4 - 20 mA)
Housing	Gasketed aluminum enclosure, IP67
Operating Temp.	0 to 50° C
Storage Temp.	-30 to 70° C

Mechanical dimensions, CD/FOV/MR, and the envelope for each sensor model are illustrated on the following pages.

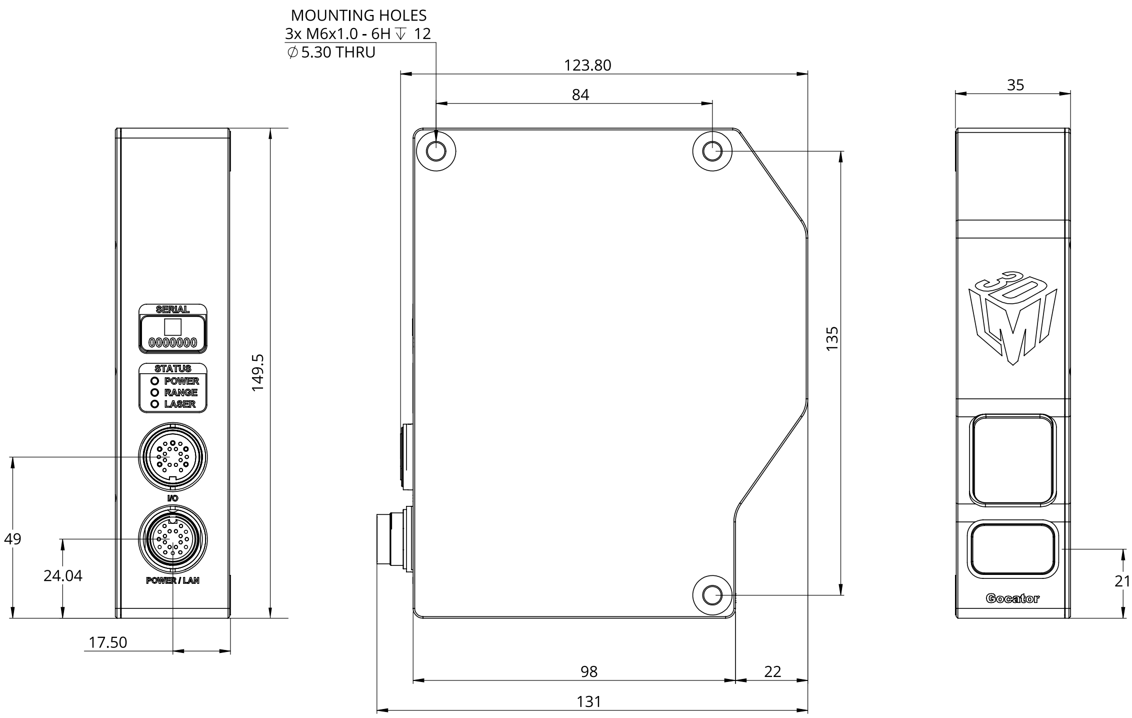
In the following drawings, the physical height dimensions of measurement ranges are reversed compared to the logical height values returned by sensors. For an illustration of the orientation of coordinate systems in relation to physical sensors, see the sections below for each sensor model.

Gocator 2320

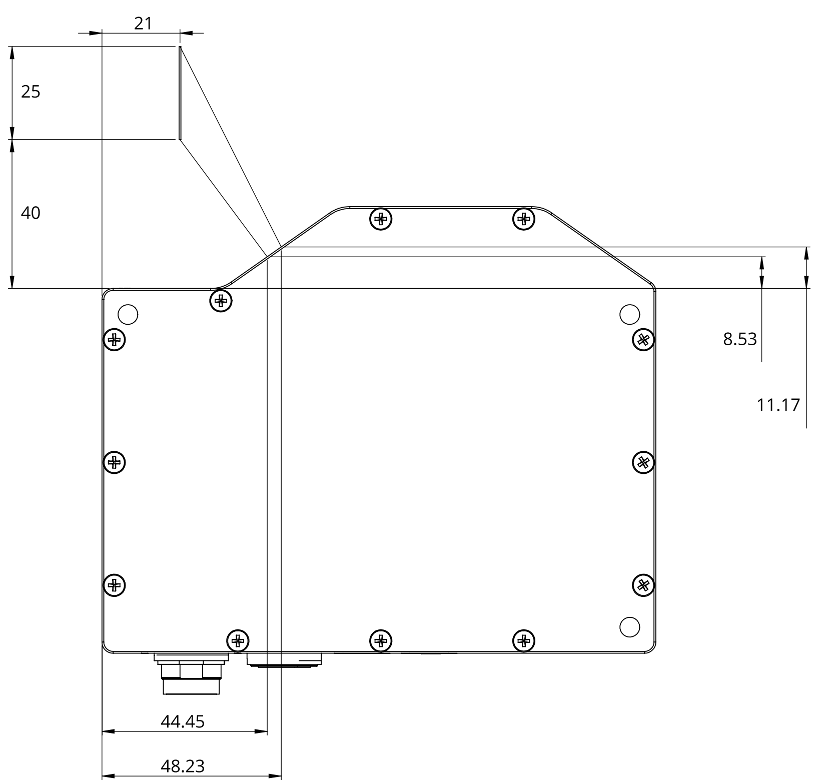
Field of View / Measurement Range / Coordinate System Orientation



Dimensions

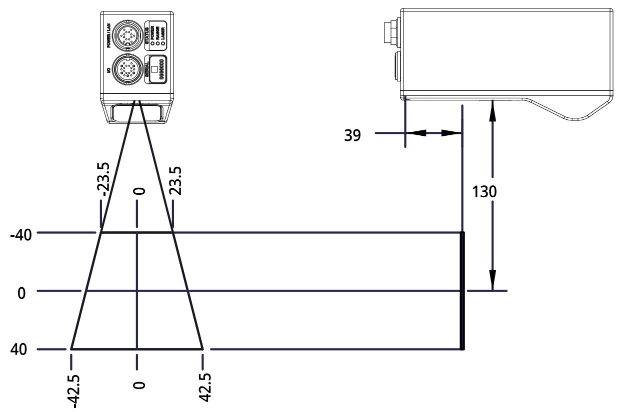


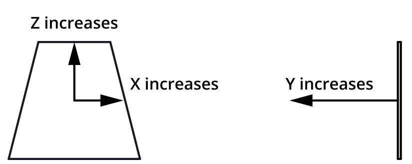
Envelope



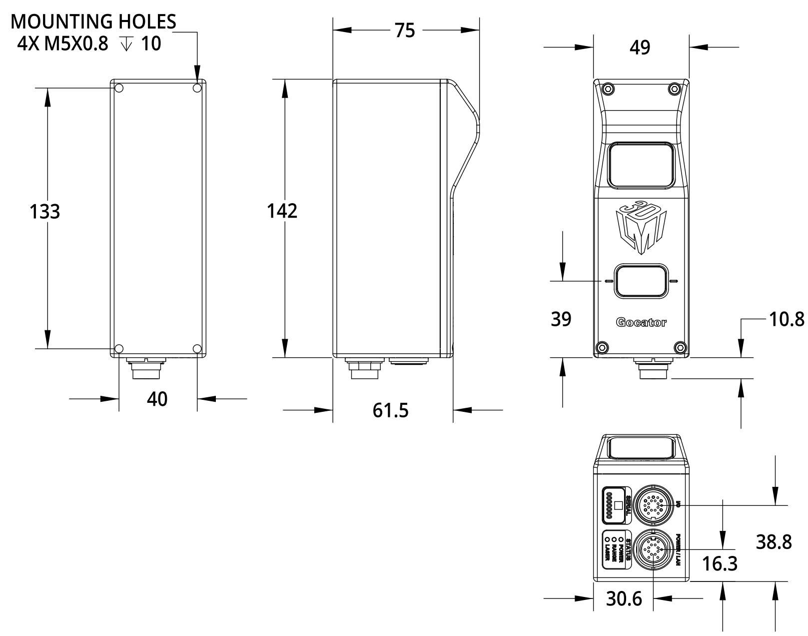
Gocator 2330

Field of View / Measurement Range / Coordinate System Orientation

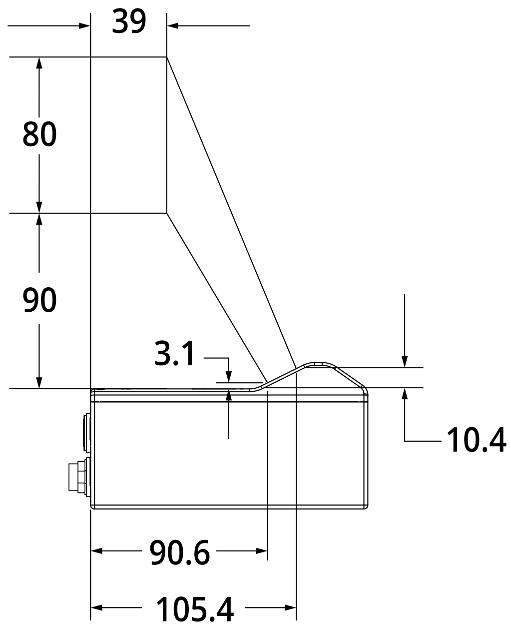




Dimensions

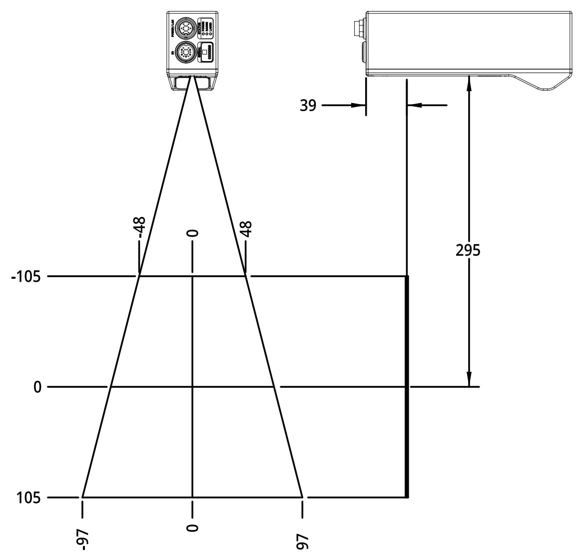


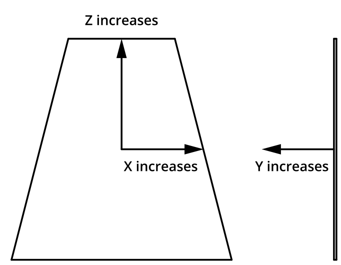
Envelope



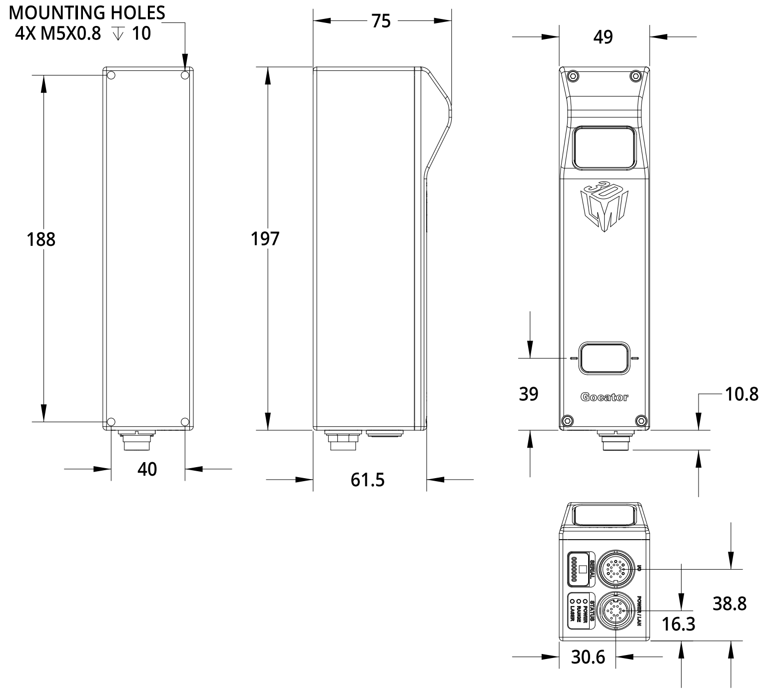
Gocator 2340

Field of View / Measurement Range / Coordinate System Orientation

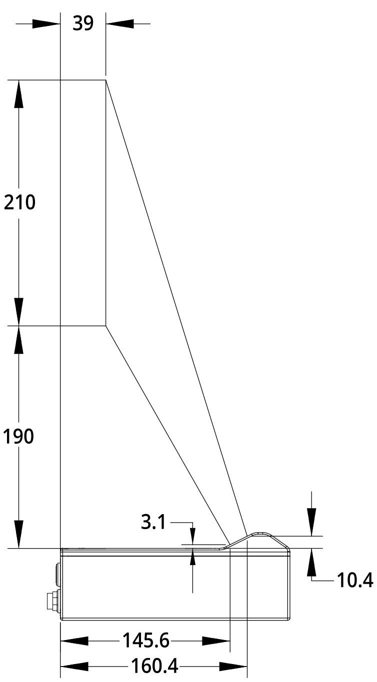




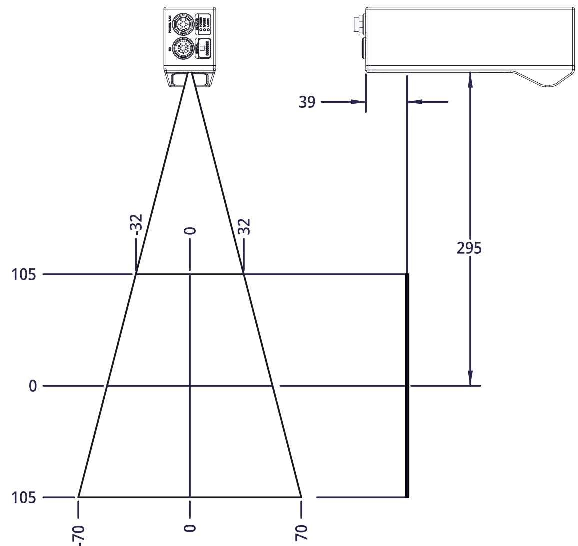
Dimensions

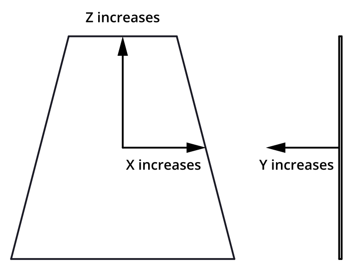


Envelope

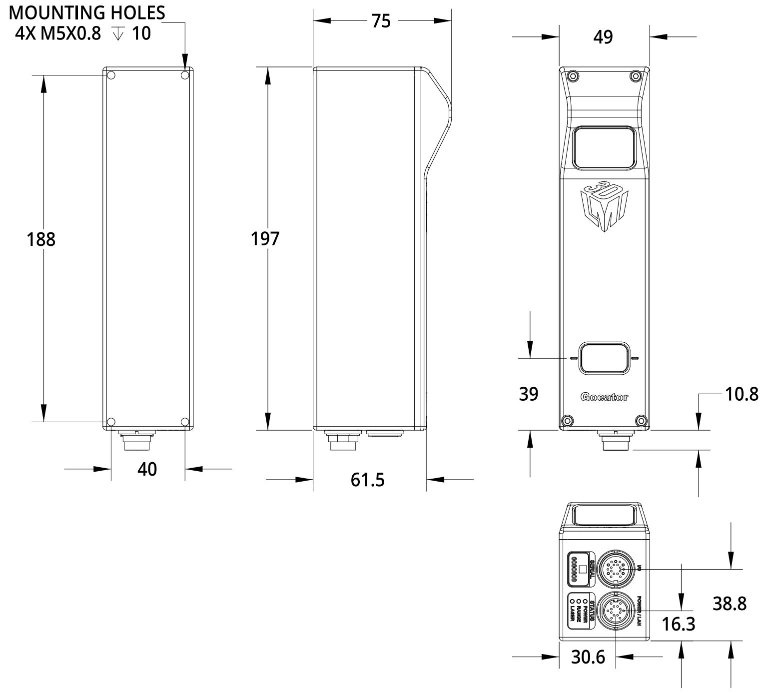


Field of View / Measurement Range / Coordinate System Orientation

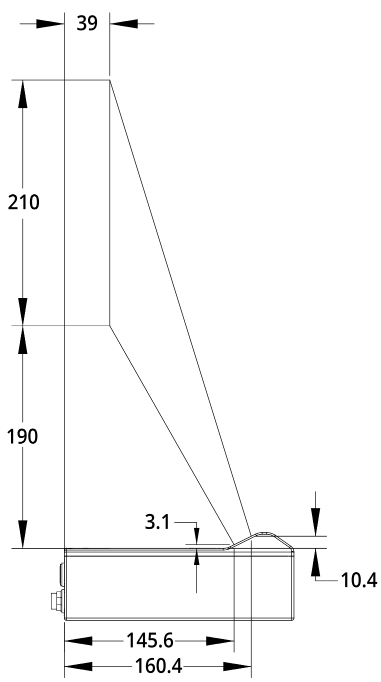




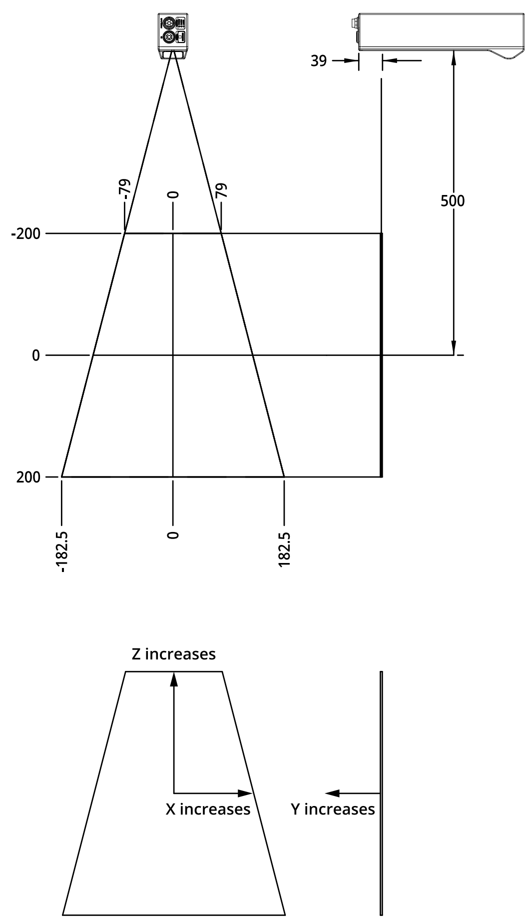
Dimensions



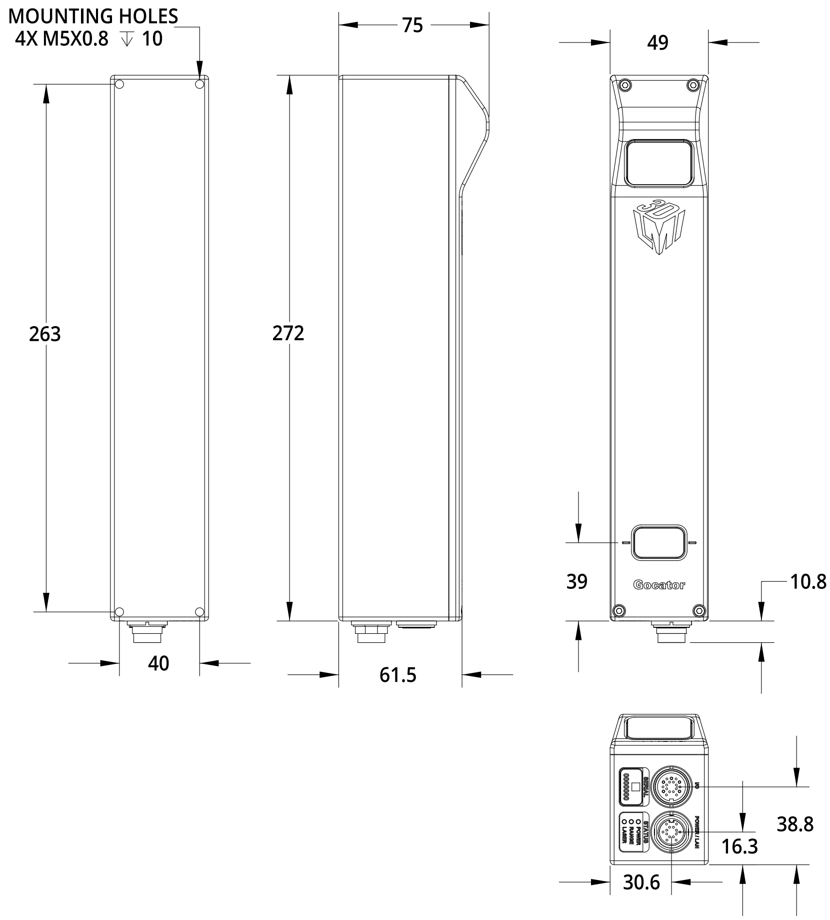
Envelope



Field of View / Measurement Range / Coordinate System Orientation

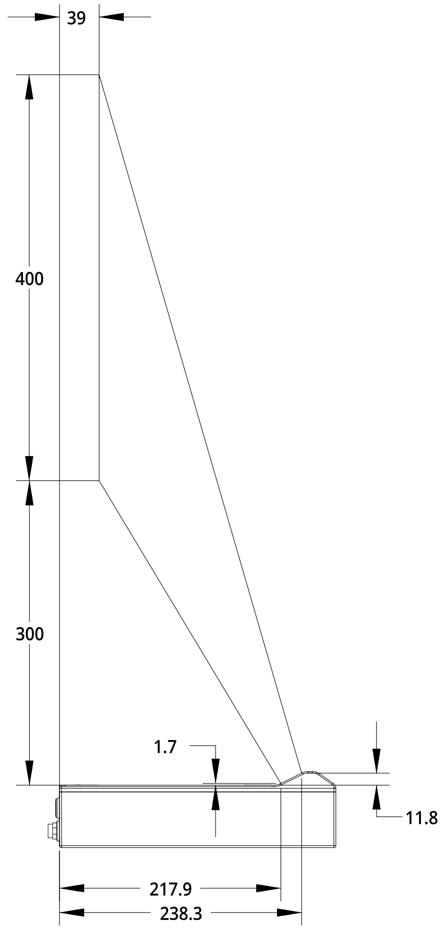


Dimensions



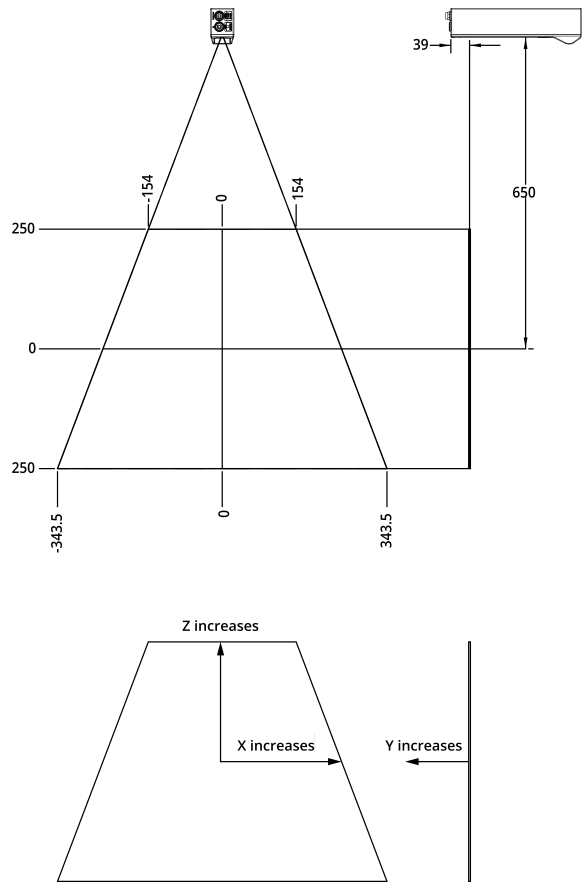
MOUNTING HOLES
4X M5X0.8 ↧ 10
263
40
75
272
61.5
49
39
10.8
Gocator
38.8
16.3
30.6

Envelope

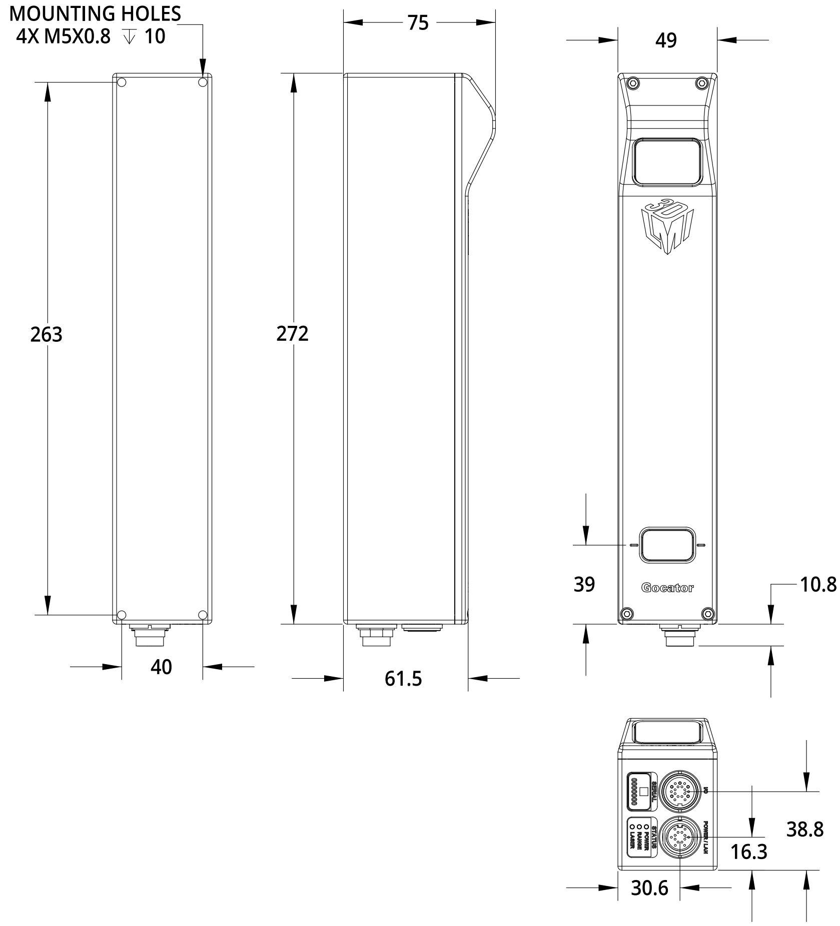


Gocator 2370

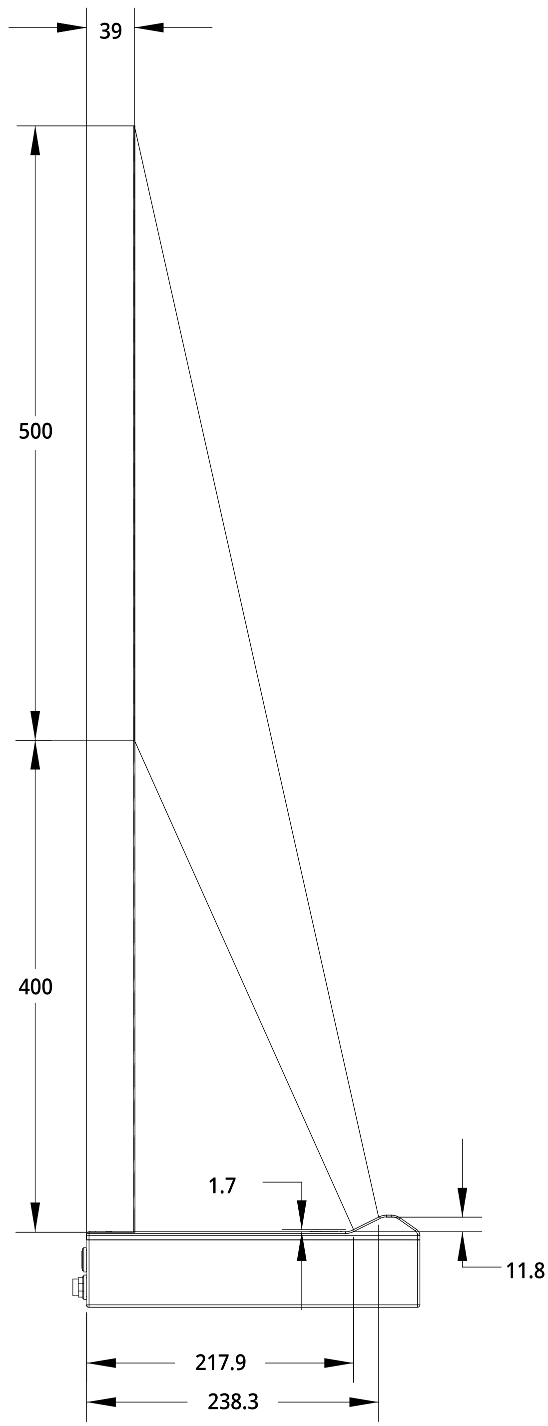
Field of View / Measurement Range / Coordinate System Orientation



Dimensions

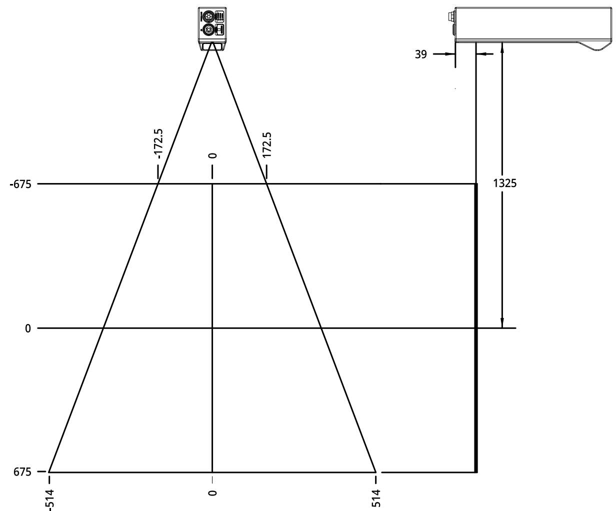

MOUNTING HOLES
4X M5X0.8 ⟂ 10
263
40
75
272
61.5
49
39
Gocator
10.8
38.8
16.3
30.6

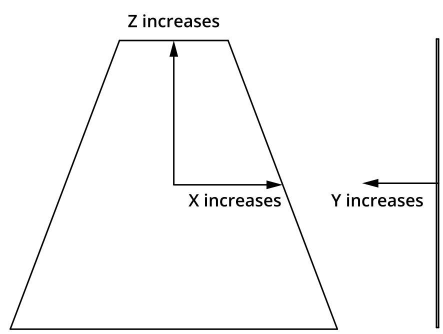
Envelope



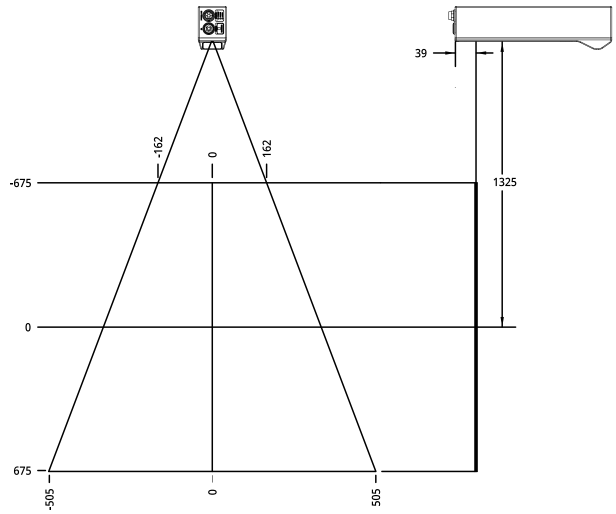
Field of View / Measurement Range / Coordinate System Orientation

Class 3B

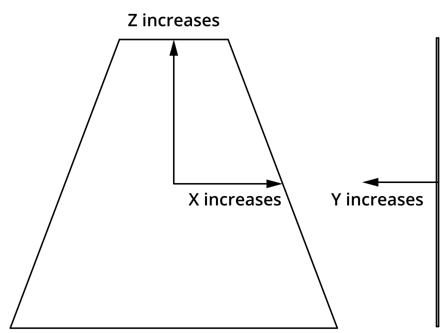




Class 2 & 3R

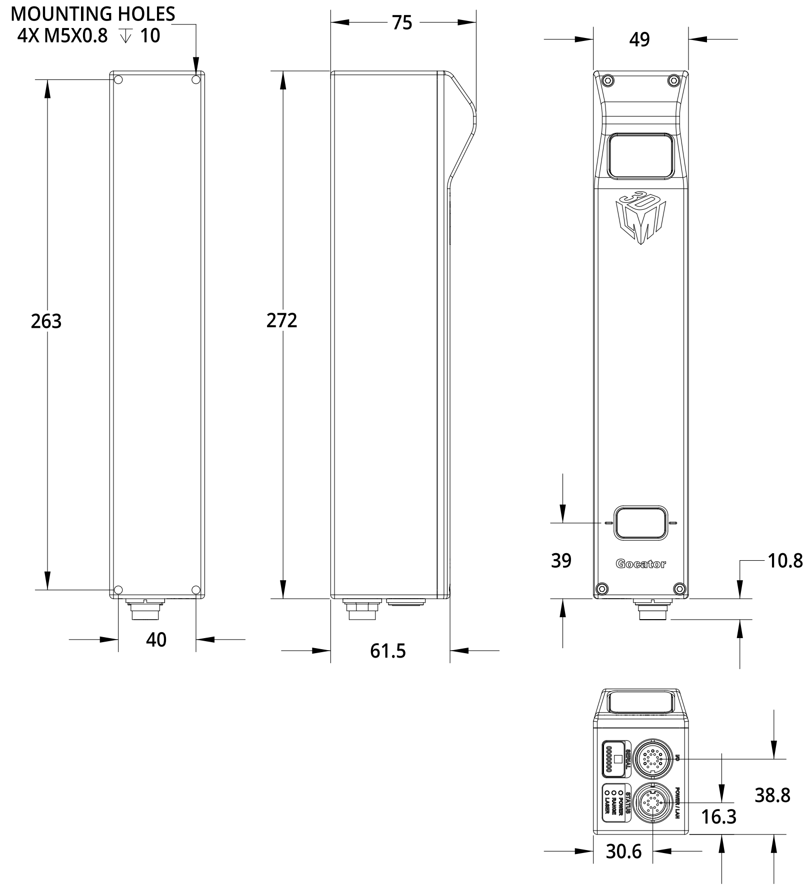


39
-162
0
162
-675
1325
0
675
-505
0
505



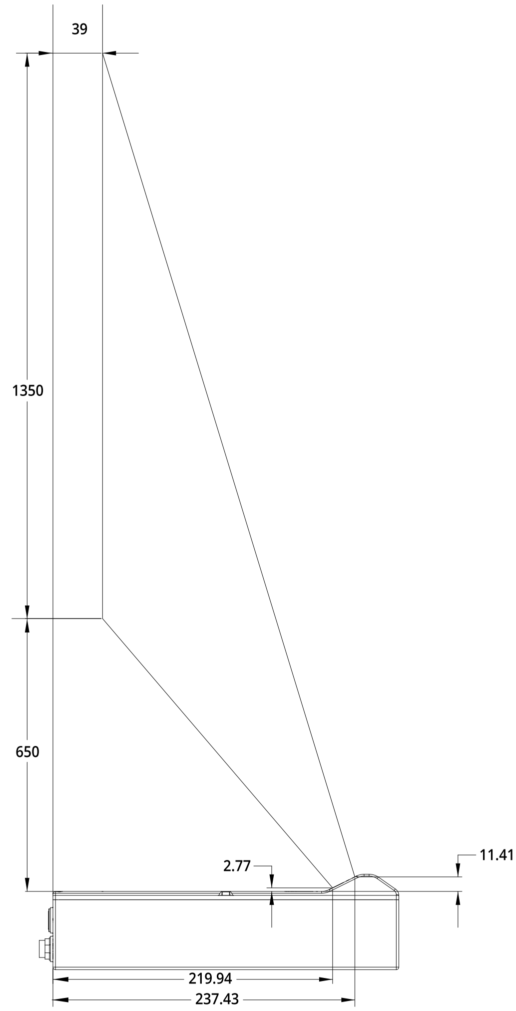
Z increases
X increases
Y increases

Dimensions

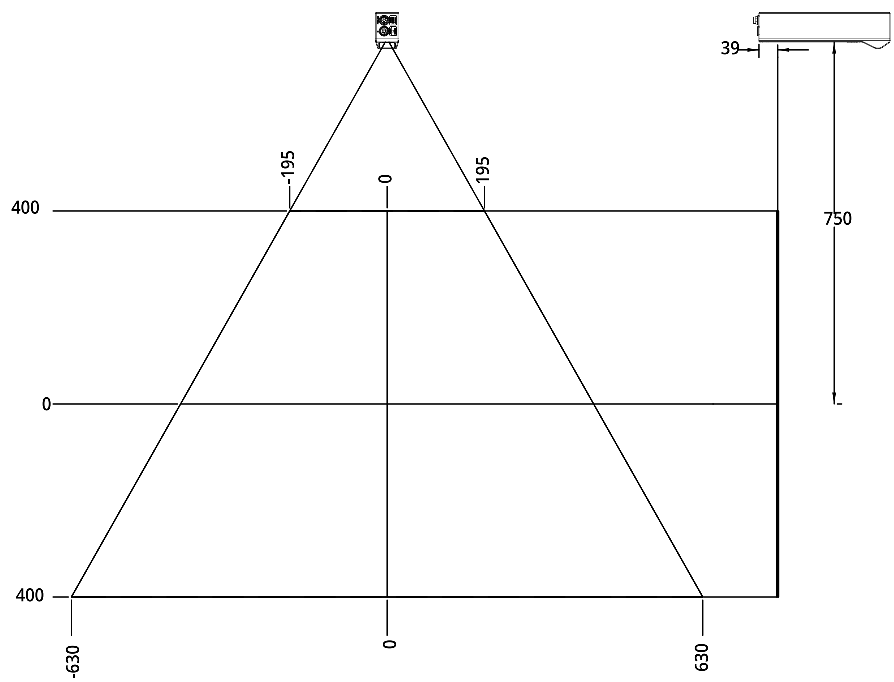


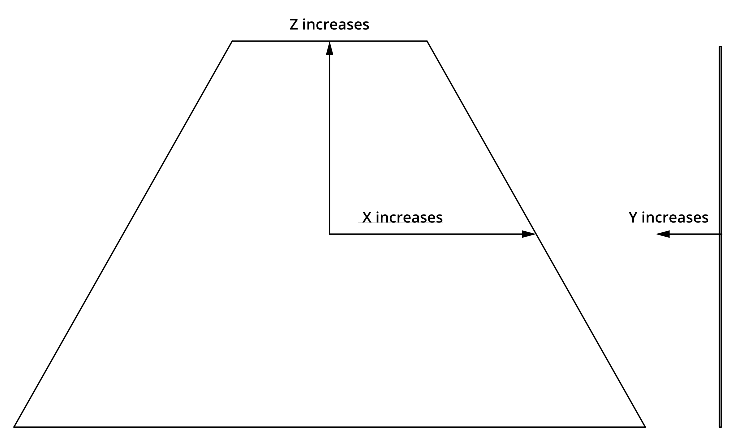
MOUNTING HOLES
4X M5X0.8 ⟂ 10
75
49
263
272
39
10.8
40
61.5
38.8
16.3
30.6

Envelope

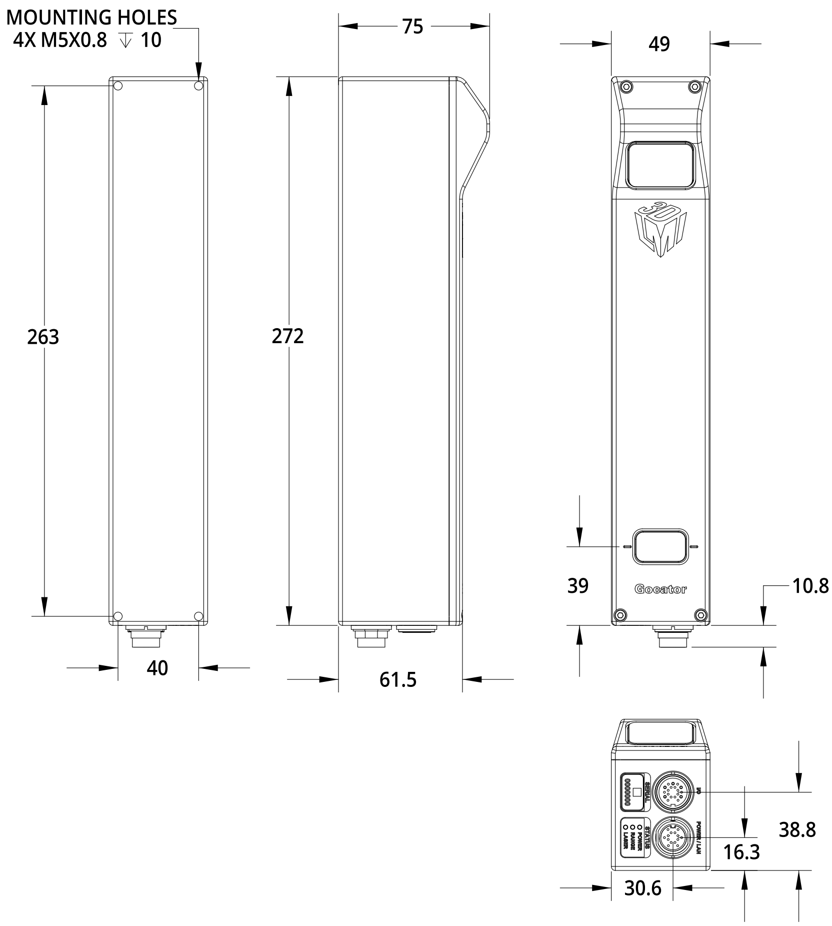


Field of View / Measurement Range / Coordinate System Orientation

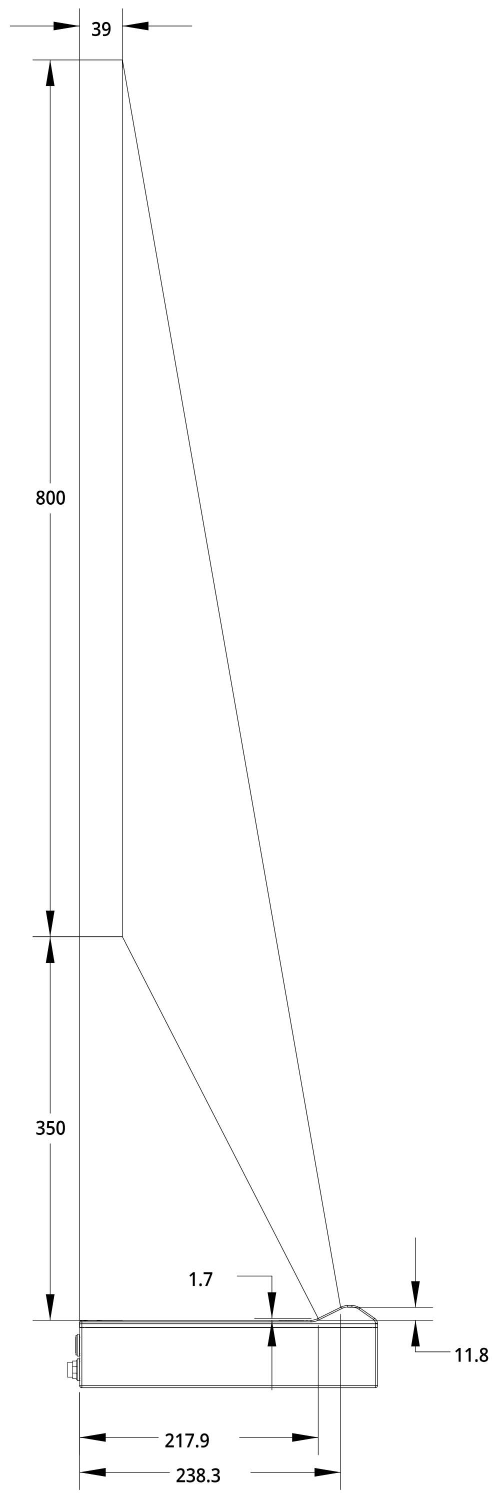




Dimensions


MOUNTING HOLES
4X M5X0.8 ↧ 10
75
49
263
272
39
10.8
Gocator
40
61.5
38.8
16.3
30.6

Envelope

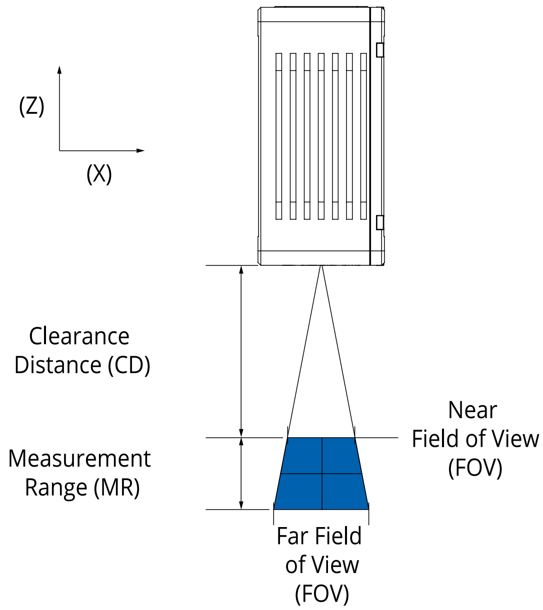


Gocator 2400 Series

The Gocator 2400 series consists of the following models:

MODEL	2410	2420	2430	2440	2450	2490
Scan Rate (Hz)	200 Hz, up to 5 kHz	200 Hz, up to 5 kHz	320 - 5000	310 - 5000	270 - 5000	370 Hz (full view) or 800 Hz (configured for 1m x 2m field of view) to 5000 Hz
Data Points / Profile	1710	1940	1500	1500	1800	1920
Linearity Z (+/- % of MR)	0.015	0.006	0.01	0.01	0.01	0.04
Resolution X (mm) (Profile Data Interval)	0.0058 - 0.0062	0.014¶ - 0.0165	0.037 - 0.057	0.09 - 0.13	0.100 - 0.255	0.25 - 1.1
Repeatability Z (µm)	0.2	0.4	0.8	1.2	2.0	12
Clearance Distance (CD) (mm)	19	60	75	183	270	350
Measurement Range (MR) (mm)	6	25	80	210	550	1525
Field of View (FOV) (mm)	10 - 10	27 - 32	47 - 85	96 - 194	145 - 425	390 - 2000
Laser Classes	2M, 3R (blue, 405 nm)		2, 3R, 3B (red, 660 nm; blue, 405 nm)		2, 3R, 3B (blue, 405 nm)	2, 3R (red, 660 nm)
Dimensions (mm)	44x90x145	44x90x145	44x90x155	44x90x190	44x90x240	49x85x272
Weight (kg)	0.88	0.88	1.0	1.2	1.2	1.5

The following diagram illustrates some of the terms used in the table above.



Optical models, laser classes, and packages can be customized. Contact LMI for more details.

Specifications stated are based on standard laser classes. Linearity Z, Resolution Z, and Repeatability Z may vary for other laser classes.

All specification measurements are performed on LMI's standard calibration target (a diffuse, painted white surface).

Linearity Z is the worst case difference in average height measured, compared to the actual position over the measurement range.

Resolution Z is the maximum variability of height measurements across multiple frames, with 95% confidence.

Resolution X is the distance between data points along the laser line.

Repeatability Z is measured with a flat target at the middle of the measurement range. It is the 95% confidence variation of the average height over 4096 frames. Height values are averaged over the full FOV.

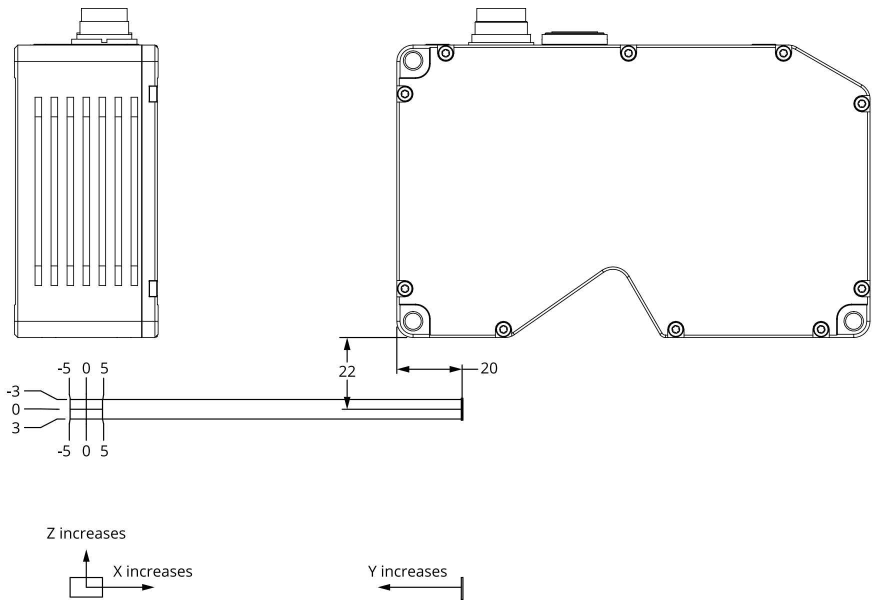
See *Resolution and Linearity* on page 62 for more information.

ALL 2400 SERIES MODELS	
Interface	Gigabit Ethernet
Inputs	Differential Encoder, Laser Safety Enable, Trigger
Outputs	2x Digital output, RS-485 Serial (115 kBaud), 1x Analog Output (4 - 20 mA)
Housing	Gasketed aluminum enclosure, IP67
Input Voltage (Power)	+24 to +48 VDC (9 W; 13 Watts for Gocator 2490); Ripple +/- 10%
Operating Temp.	0 to 50° C
Storage Temp.	-30 to 70° C

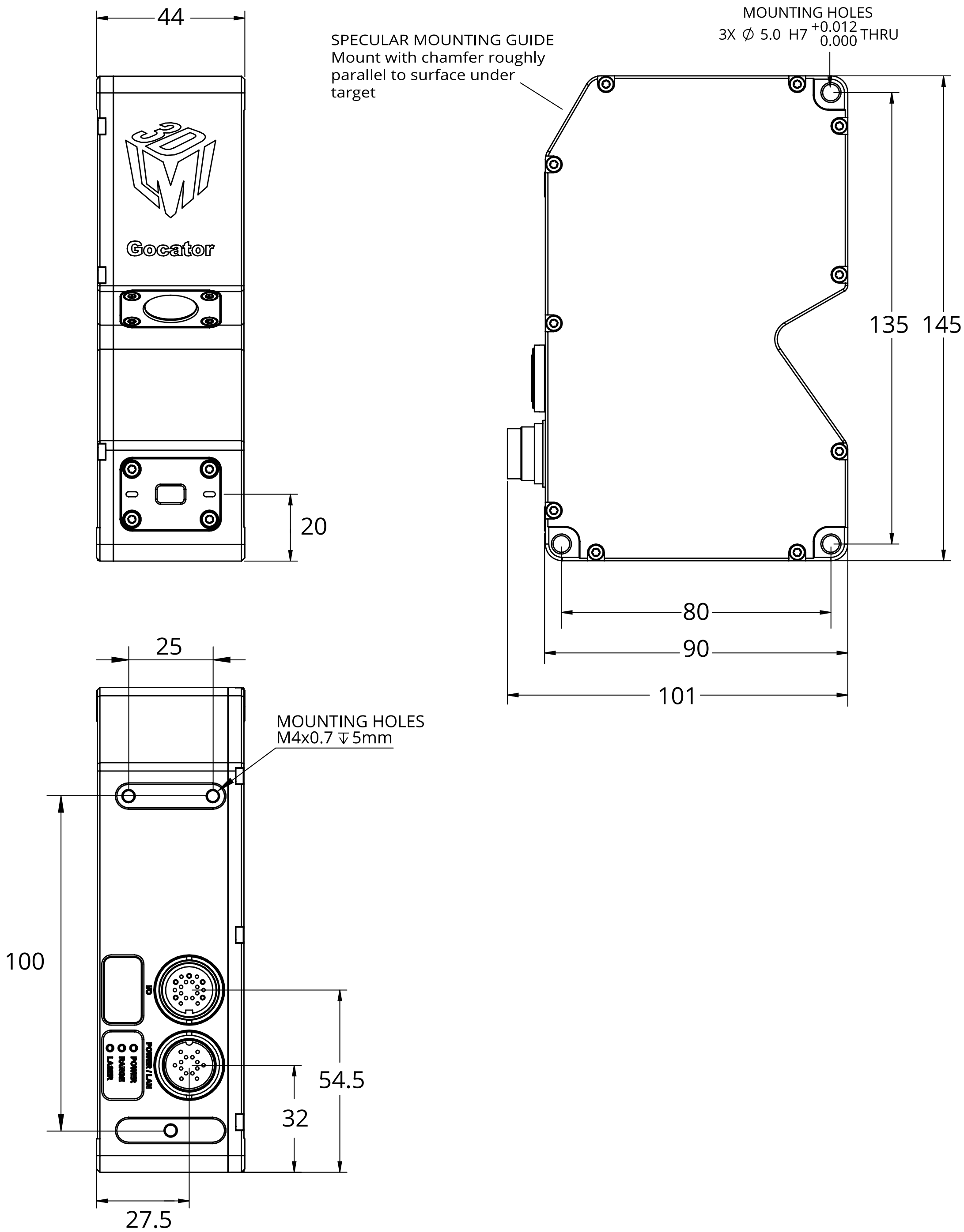
Mechanical dimensions, CD/FOV/MR, and the envelope for each sensor model are illustrated on the following pages.

Gocator 2410

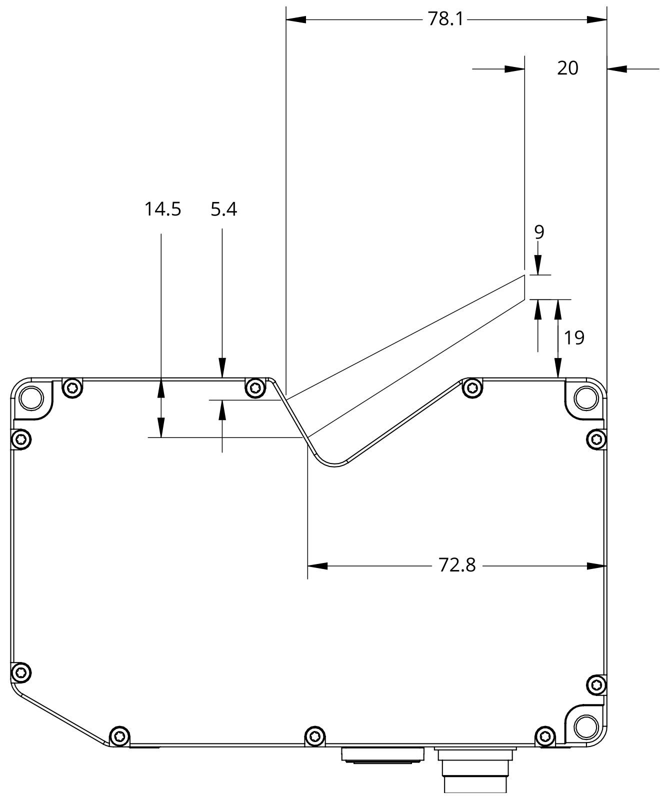
Field of View / Measurement Range / Coordinate System Orientation



Dimensions

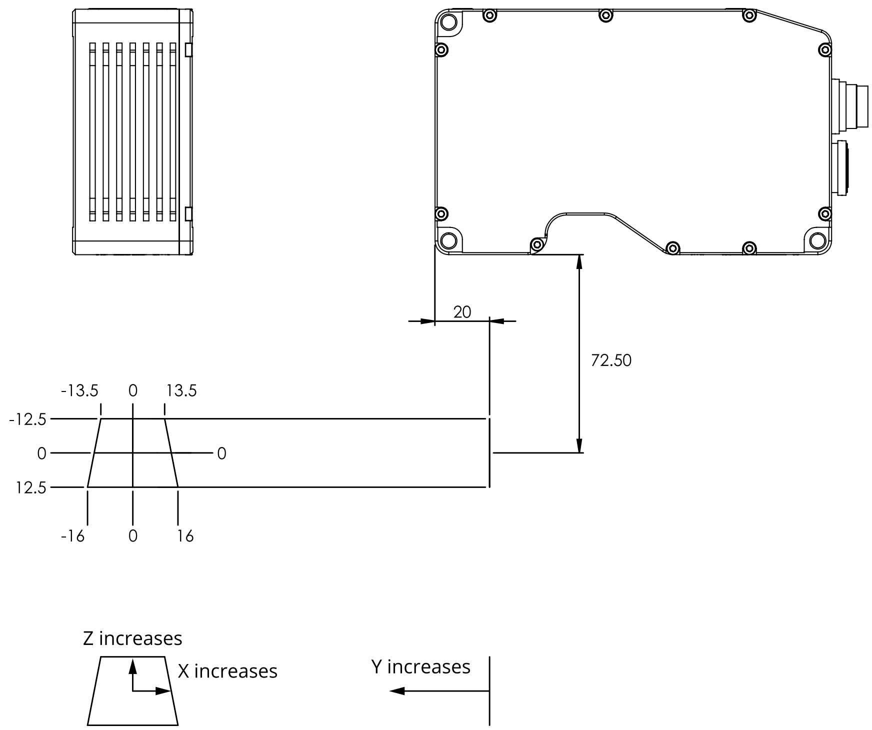

44
Gocator
20
SPECULAR MOUNTING GUIDE
Mount with chamfer roughly
parallel to surface under
target
MOUNTING HOLES
3X ⌀ 5.0 H7 +0.012 0.000 THRU
135
145
80
90
101
25
MOUNTING HOLES
M4x0.7 ↧5mm
100
IO
POWER
RANGE
LASER
POWER / LAN
54.5
32
27.5

Envelope

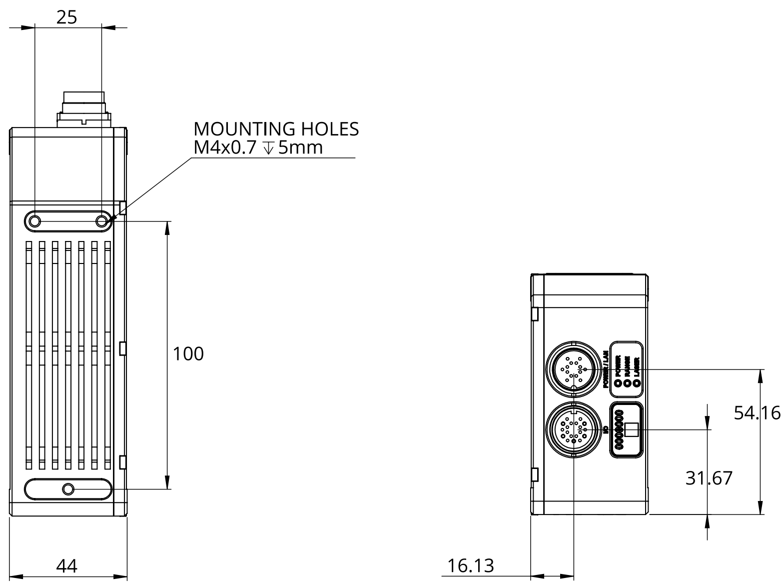


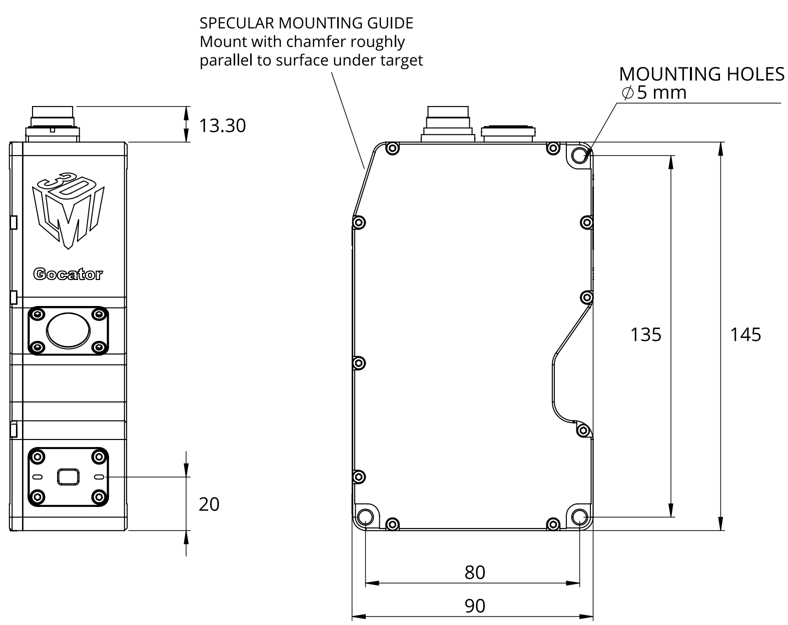
Gocator 2420

Field of View / Measurement Range / Coordinate System Orientation

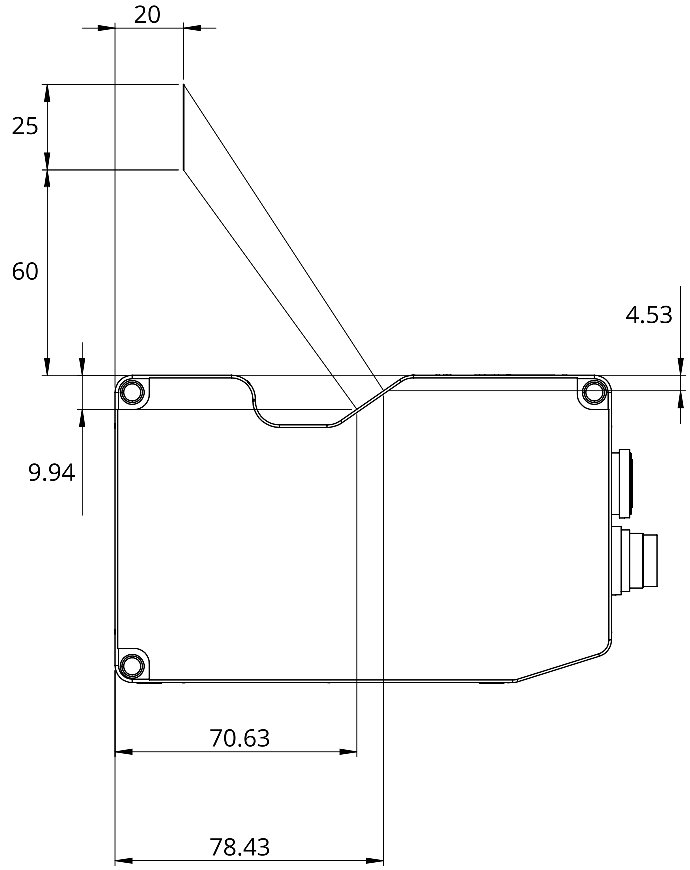


Dimensions


25
MOUNTING HOLES
M4x0.7 ↧5mm
100
44
54.16
31.67
16.13

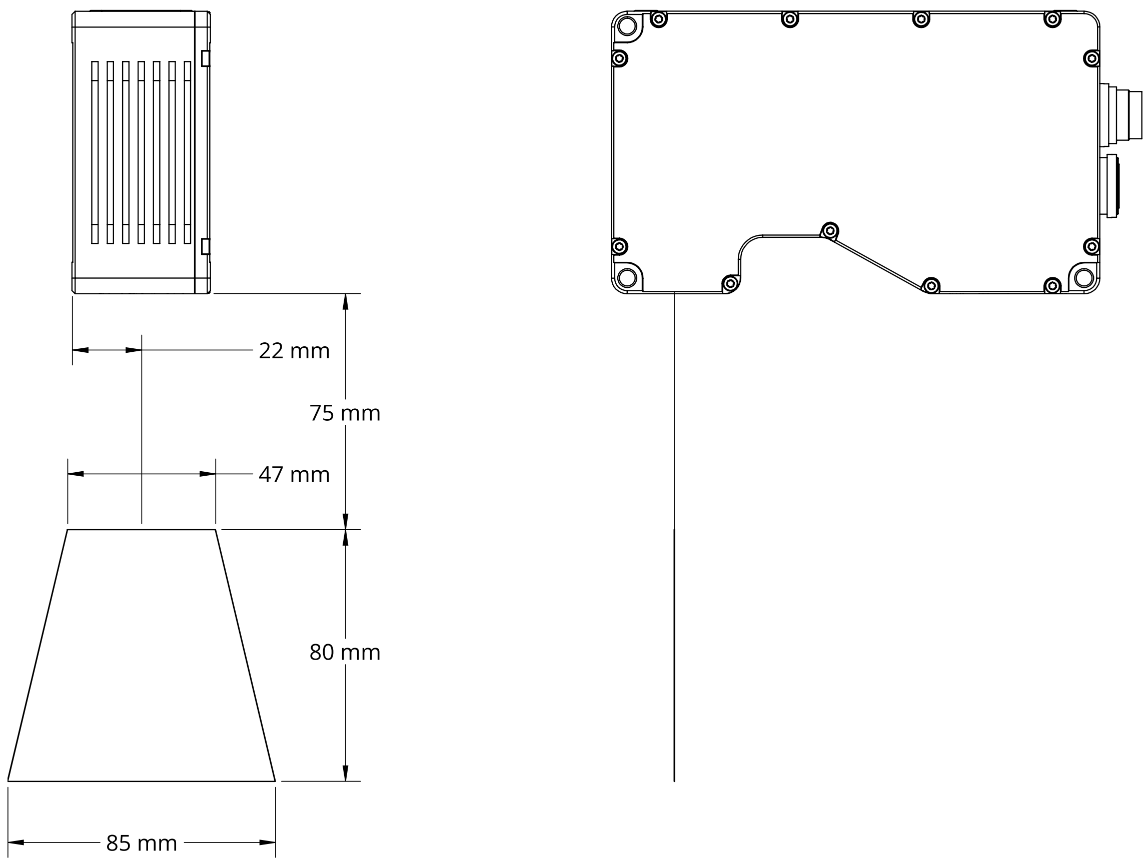

SPECULAR MOUNTING GUIDE
Mount with chamfer roughly
parallel to surface under target
MOUNTING HOLES
⌀5 mm
13.30
Gocator
135
145
20
80
90

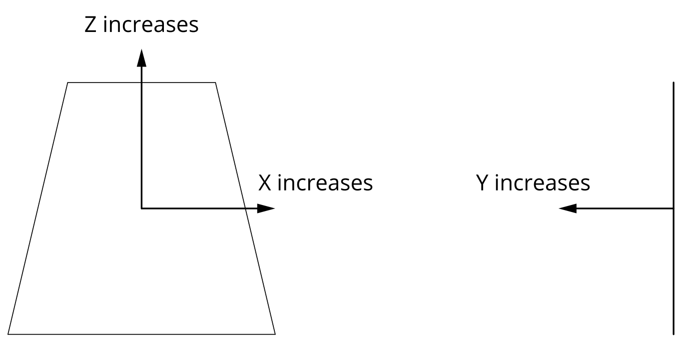
Envelope



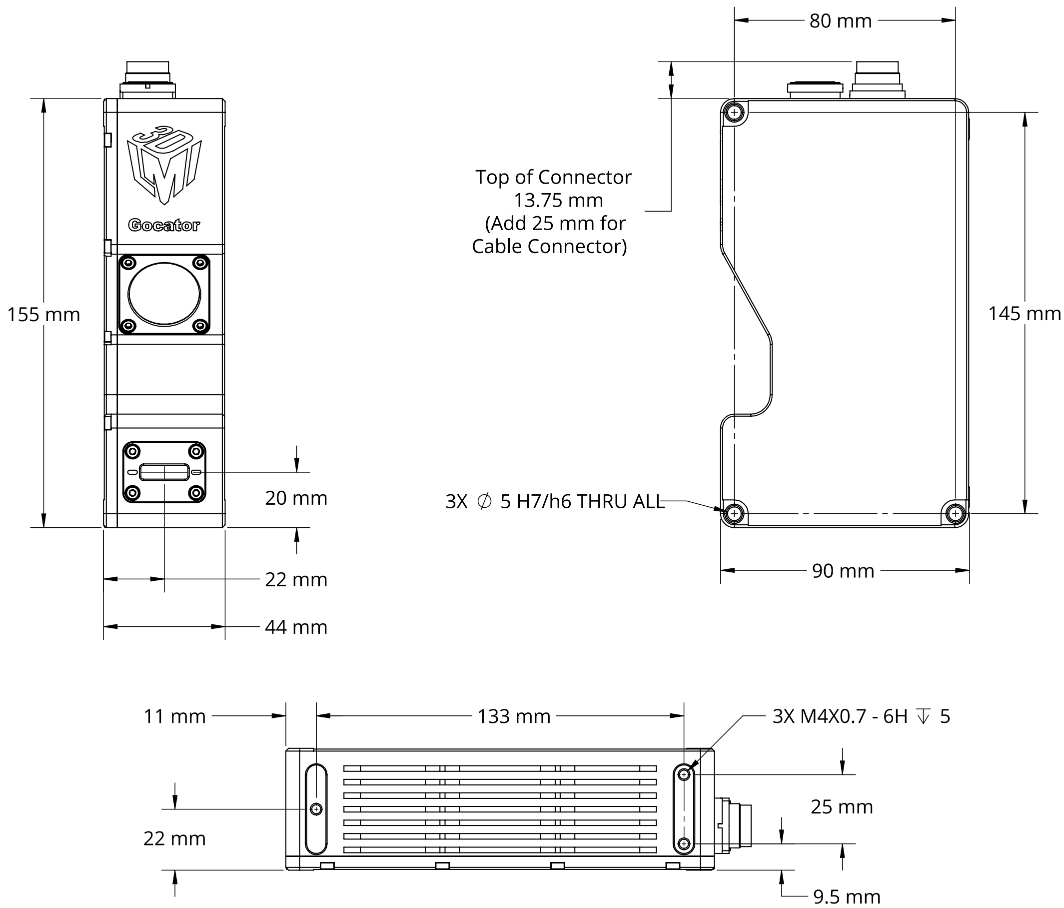
Gocator 2430

Field of View / Measurement Range / Coordinate System Orientation

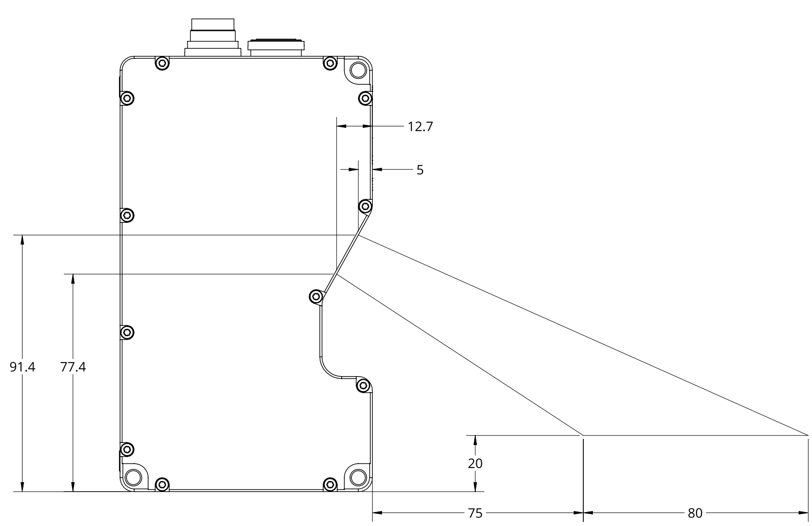




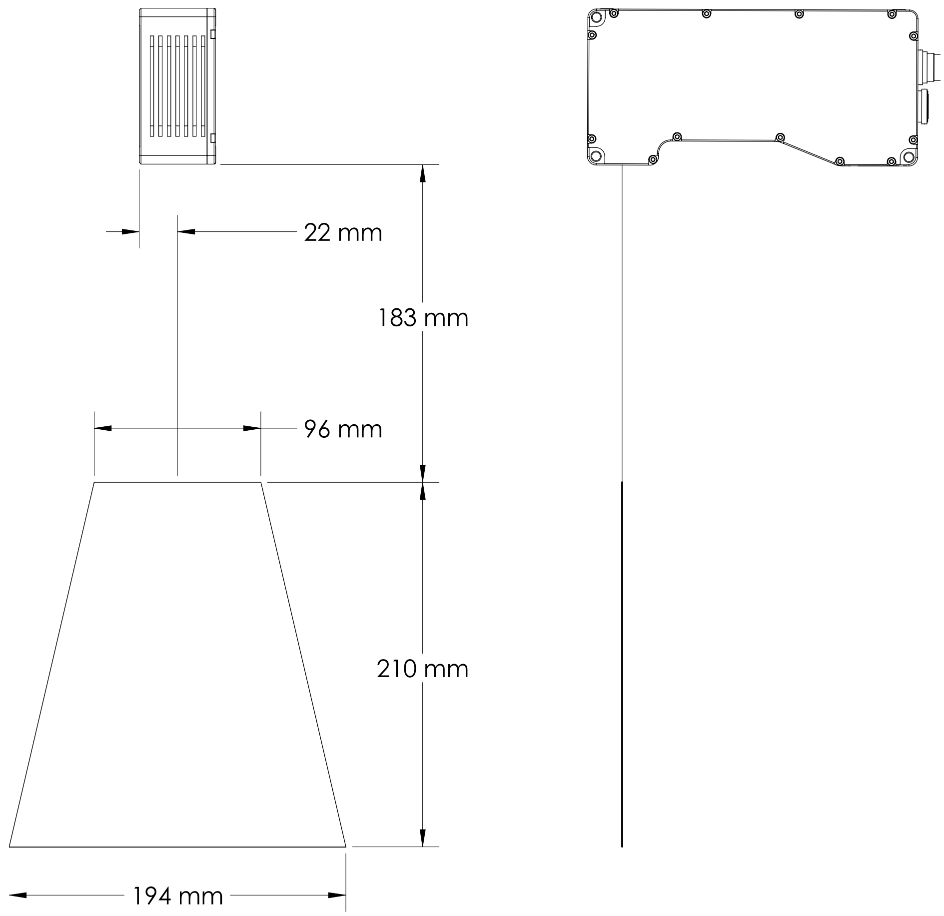
Dimensions

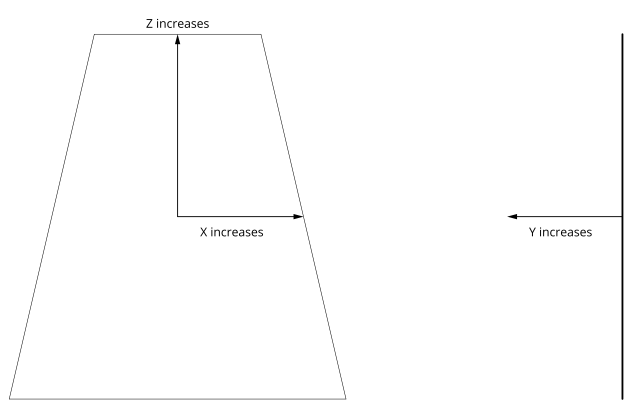


Envelope

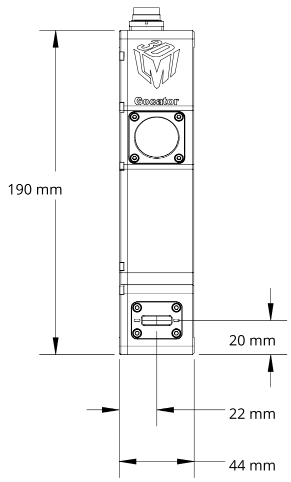


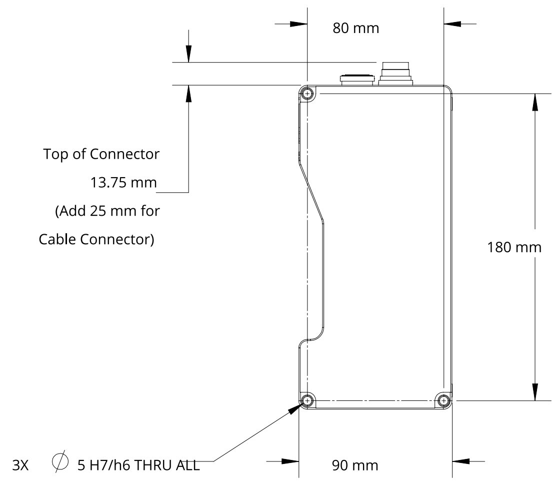
Field of View / Measurement Range / Coordinate System Orientation

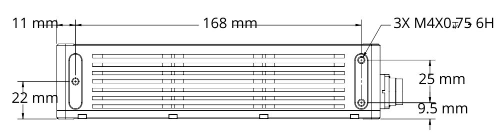




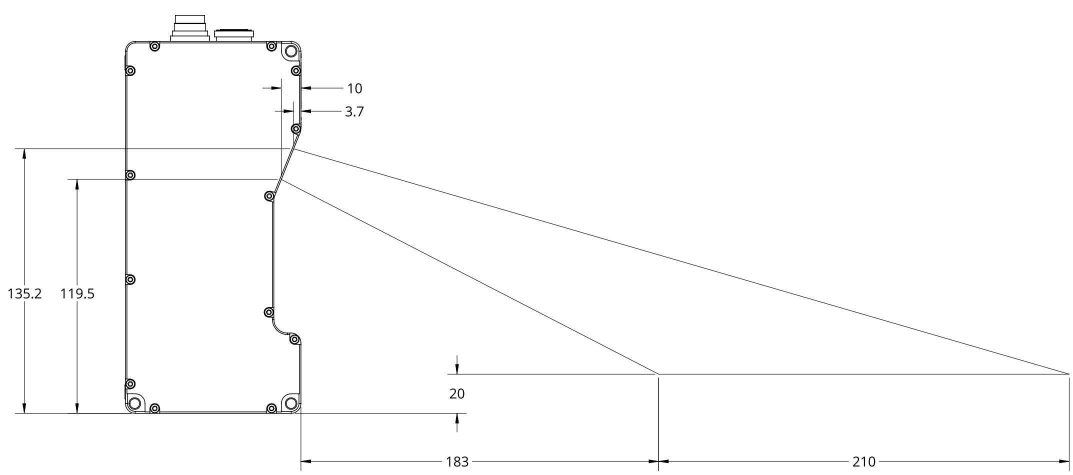
Dimensions





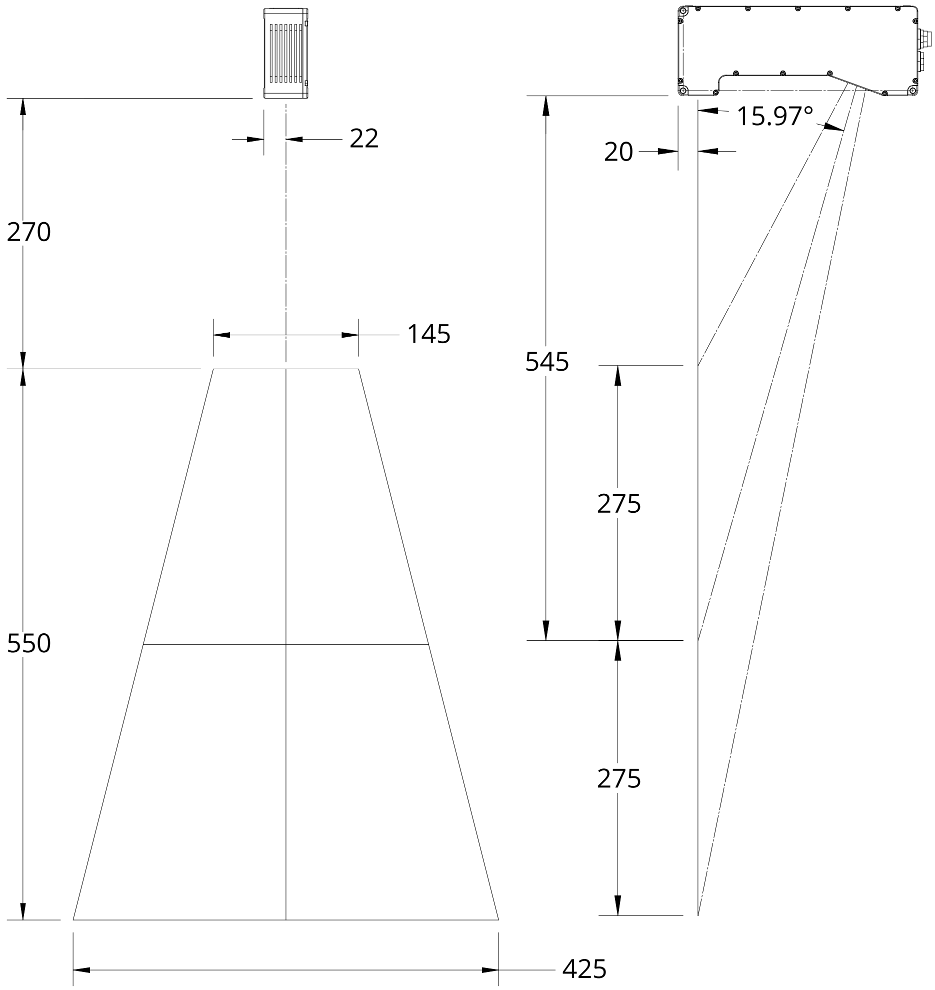


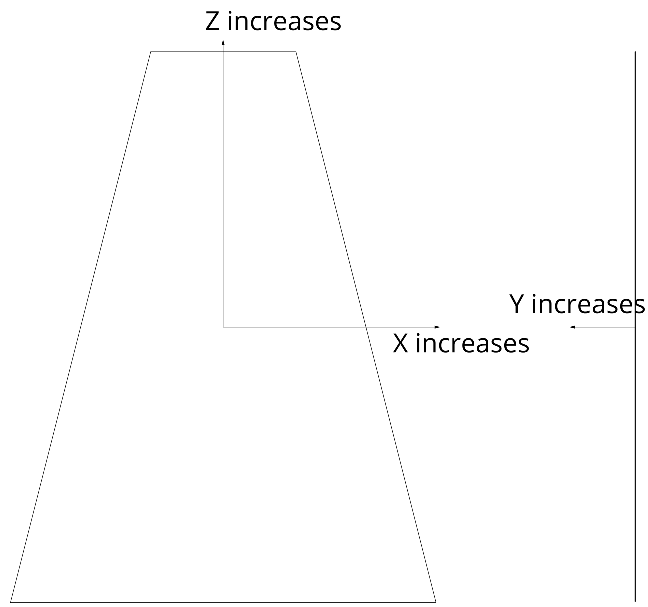
Envelope



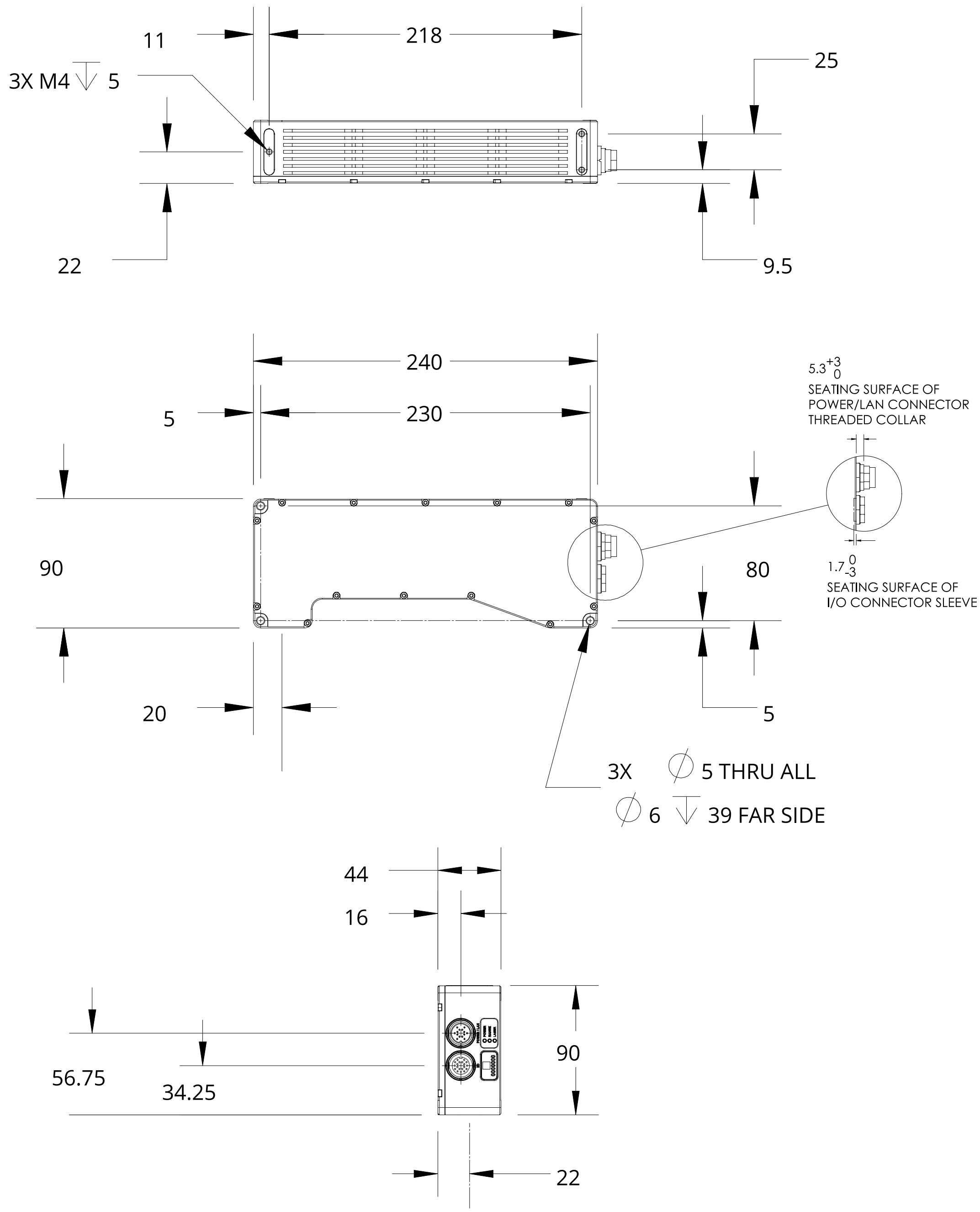
Gocator 2450

Field of View / Measurement Range / Coordinate System Orientation

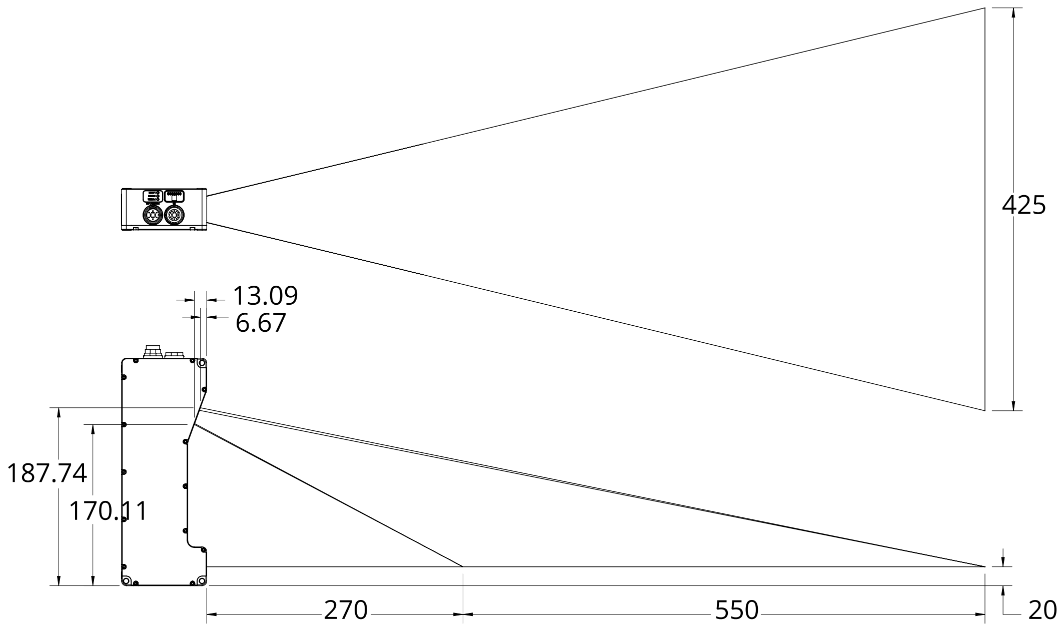

22
270
145
550
425
20
15.97°
545
275
275


Z increases
Y increases
X increases

Dimensions

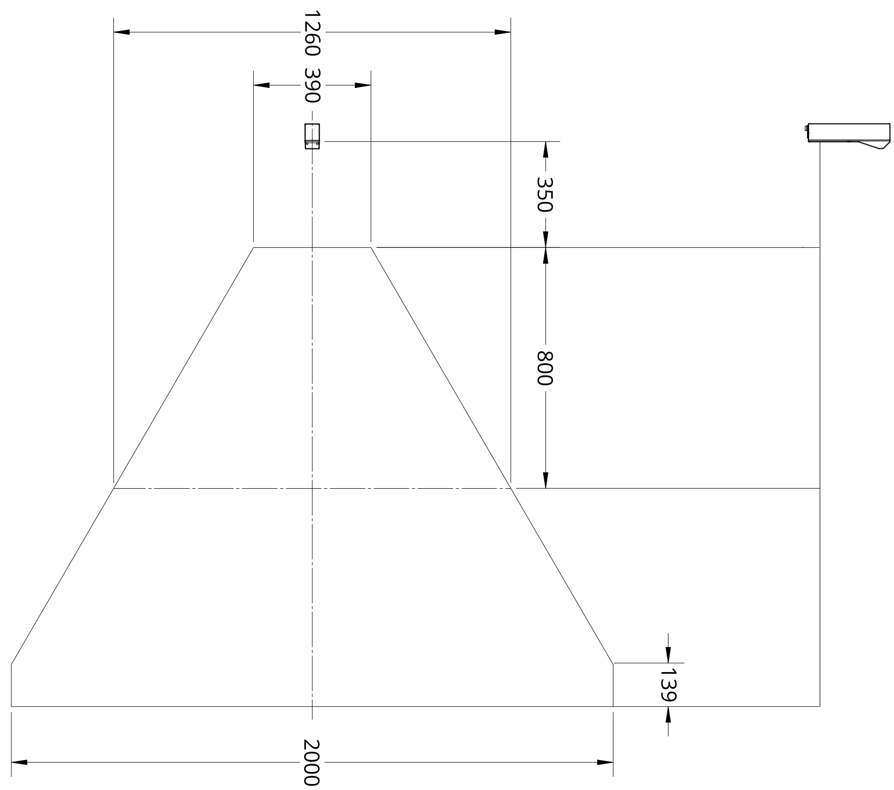

11
218
3X M4 5
25
22
9.5
240
5
230
5.3 +3 0
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
90
80
1.7 0 -3
SEATING SURFACE OF
I/O CONNECTOR SLEEVE
20
5
3X ∅ 5 THRU ALL
∅ 6 39 FAR SIDE
44
16
56.75
34.25
90
22

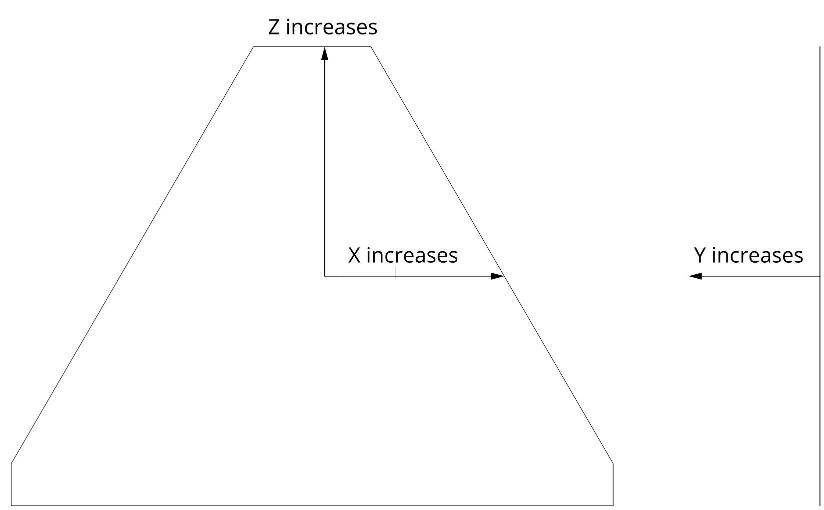
Envelope



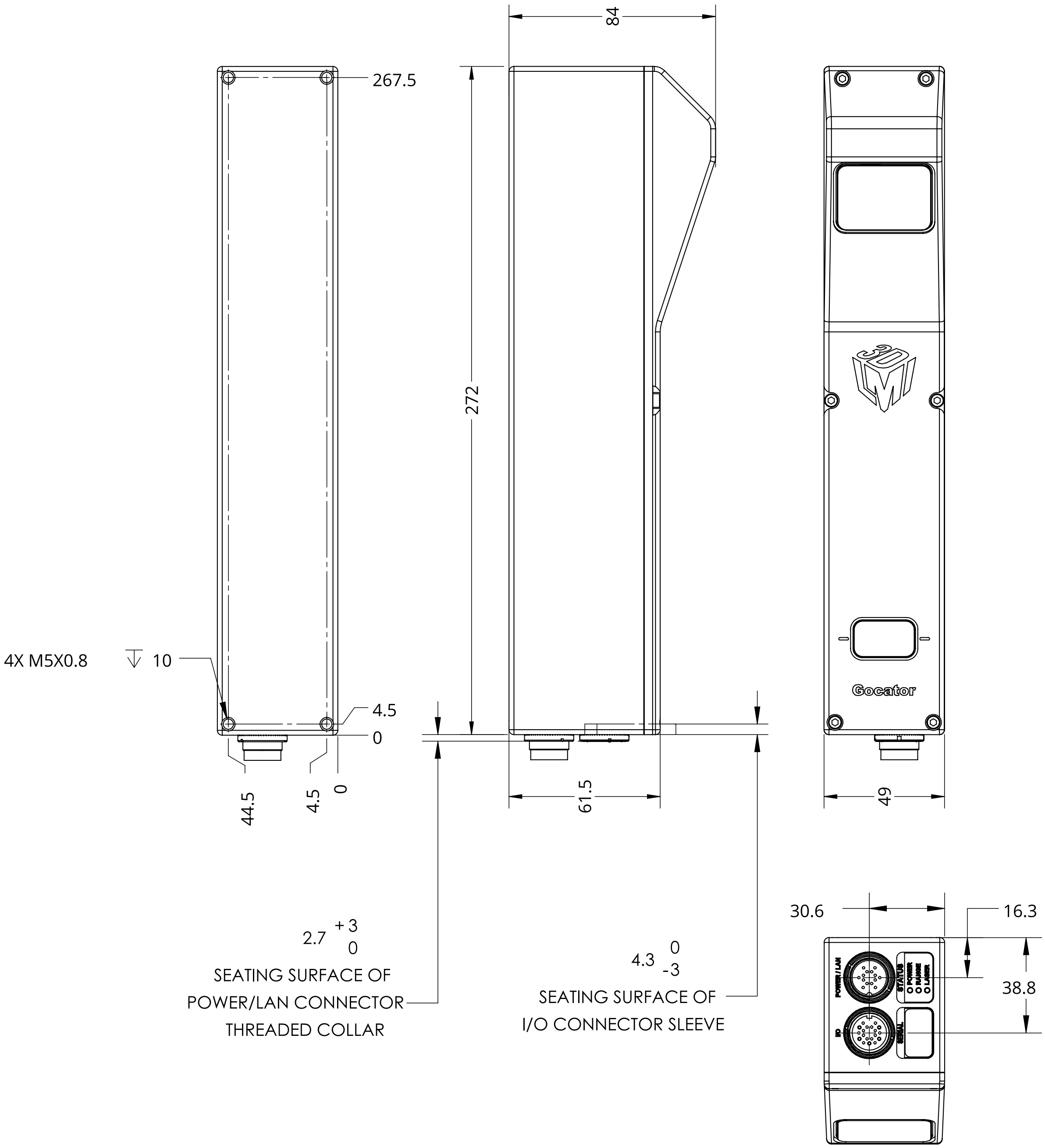
Gocator 2490

Field of View / Measurement Range / Coordinate System Orientation



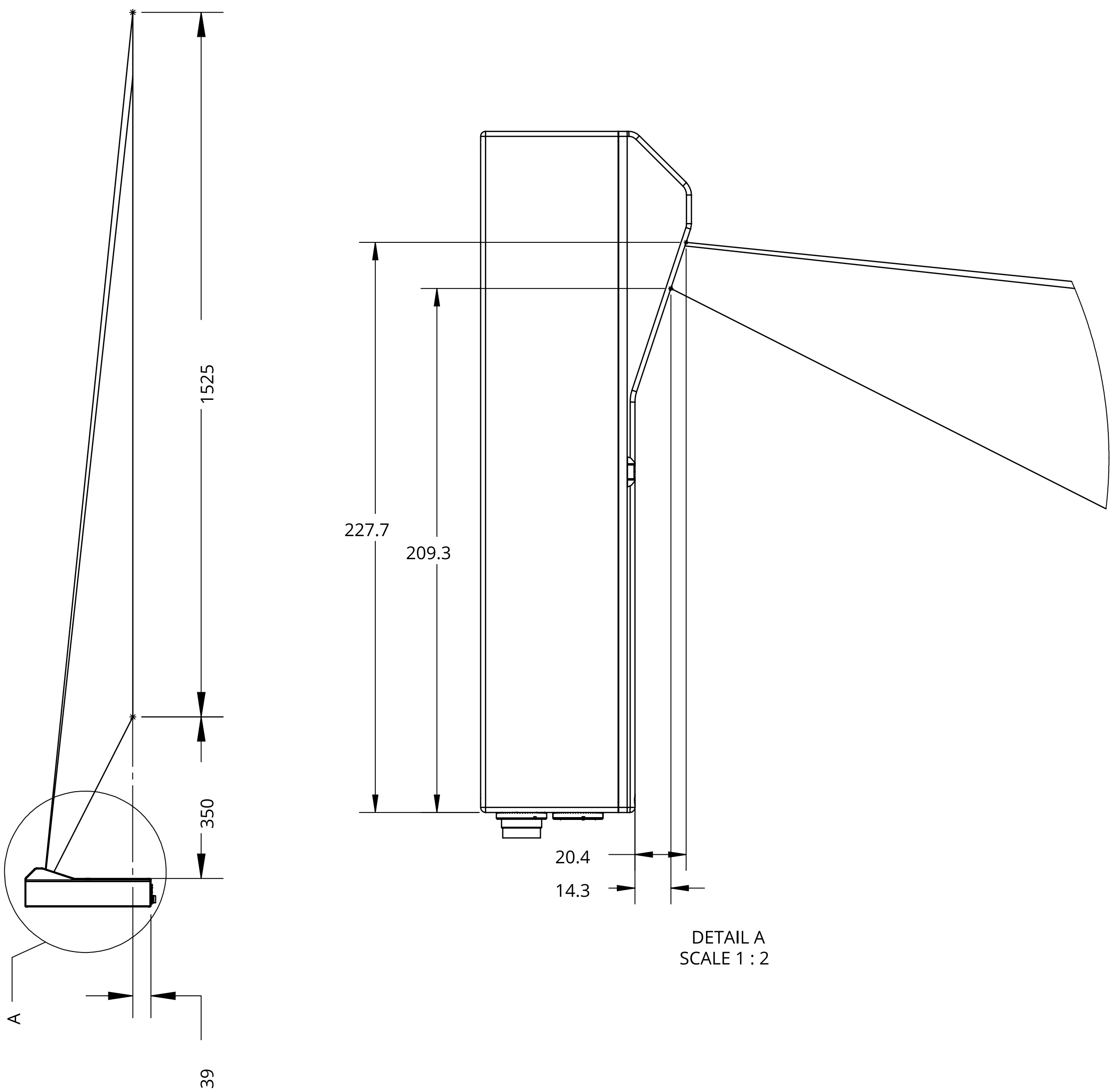


Dimensions



84
267.5
272
4X M5X0.8
10
4.5
0
44.5
4.5
0
61.5
49
2.7 +3 0
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
4.3 0 -3
SEATING SURFACE OF
I/O CONNECTOR SLEEVE
30.6
16.3
38.8
Gocator

Envelope

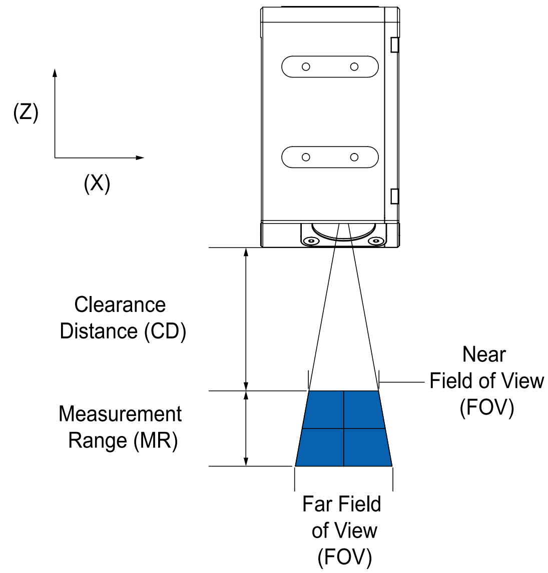


Gocator 2500 Series

The Gocator 2500 series consists of the following models:

MODEL	2510	2512	2520	2522	2530	2540	2550
Data Points / Profile	1920	1920	1920	1920	1920	1920	1920
Resolution X (µm) (Profile Data Interval)	8.0	8.0	13.0 - 17.0	13.0 - 17.0	28.0 - 54.0	64-160	80-270
Linearity Z (+/- % of MR)	0.015	0.015	0.006	0.006	0.01	0.05	0.06
Repeatability Z (µm)	0.2	0.2	0.4	0.4	0.5	1.2	2.0
Clearance Distance (CD) (mm)	17.0	17.0	47.5	17.75	40	152	216
Measurement Range (MR) (mm)	6	6	25	25	80	295	595
Field of View (FOV) (mm)	13.0 - 14.5 (diffuse)	13.0 - 14.5 (diffuse & specular)	25 - 32.5 (diffuse)	25.0 - 32.5 (diffuse) 25.0 (specular)	48 - 100 (diffuse)	120-292	154-518
Scan Rate	2.4 kHz to 10 kHz		1.6 kHz to 10 kHz		2 kHz to 10 kHz	1.8 kHz (full FOV) up to 10 kHz	1.8 kHz (full FOV) up to 10 kHz
Laser Classes	2 (blue, 405 nm)	2 (blue, 405 nm)	2 (blue, 405 nm)	2 (blue, 405 nm)	2, (blue, 405 nm)	2, 3R, 3B (blue, 405 nm)	2, 3R, 3B 2 (blue, 405 nm)
Dimensions (mm)	46x80x110	46x80x110	46x80x110	46x110x110	46x80x110	55x105x195	55x105x195
Weight (kg)	0.65	0.65	0.65	0.65	0.65	1.48	1.48

The following diagram illustrates some of the terms used in the table above.



Unlike other Gocator line profile sensors, the clearance distance for Gocator 2500 sensors is measured from the lowest point of the sensor's housing, not the laser window.

Specifications stated are based on standard laser classes. Linearity Z and Repeatability Z may vary for other laser classes.

All specification measurements are performed on LMI's standard calibration target (a diffuse, painted white surface).

Linearity Z is the worst case difference in average height measured, compared to the actual position over the measurement range.

Resolution X is the distance between data points along the laser line.

Repeatability Z is measured with a flat target at the middle of the measurement range. It is the 95% confidence variation of the average height over 4096 frames. Height values are averaged over the full FOV.

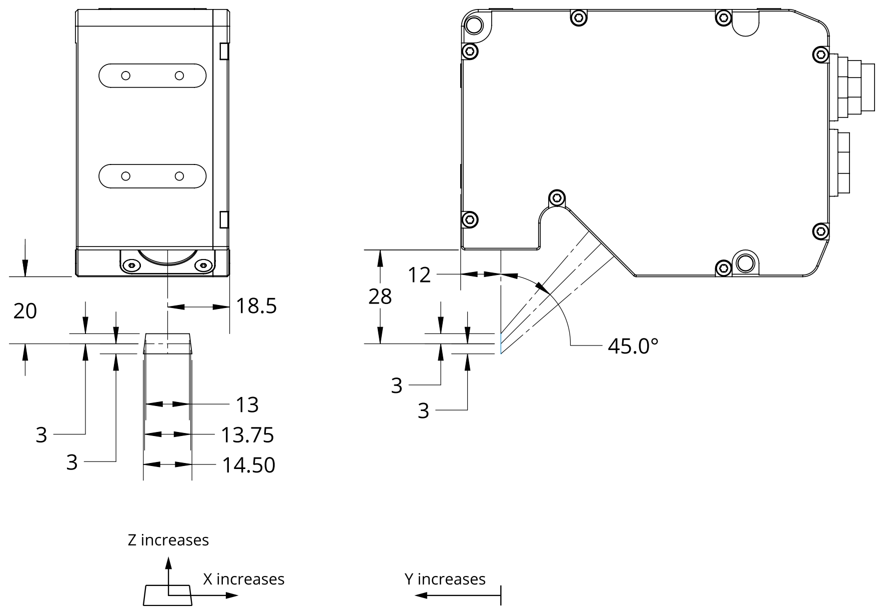
See *Resolution and Linearity* on page 62 for more information.

ALL 2500 SERIES MODELS	
Interface	Gigabit Ethernet
Inputs	Differential Encoder, Laser Safety Enable, Trigger
Outputs	2x Digital output, RS-485 Serial (115 kBaud)
Housing	Gasketed aluminum enclosure, IP67
Input Voltage (Power)	+24 to +48 VDC (15 W); Ripple +/- 10%
Operating Temp.	0 to 40 °C; for Gocator 2540 and 2550, 0 to 45 °C
Storage Temp.	-30 to 70 °C

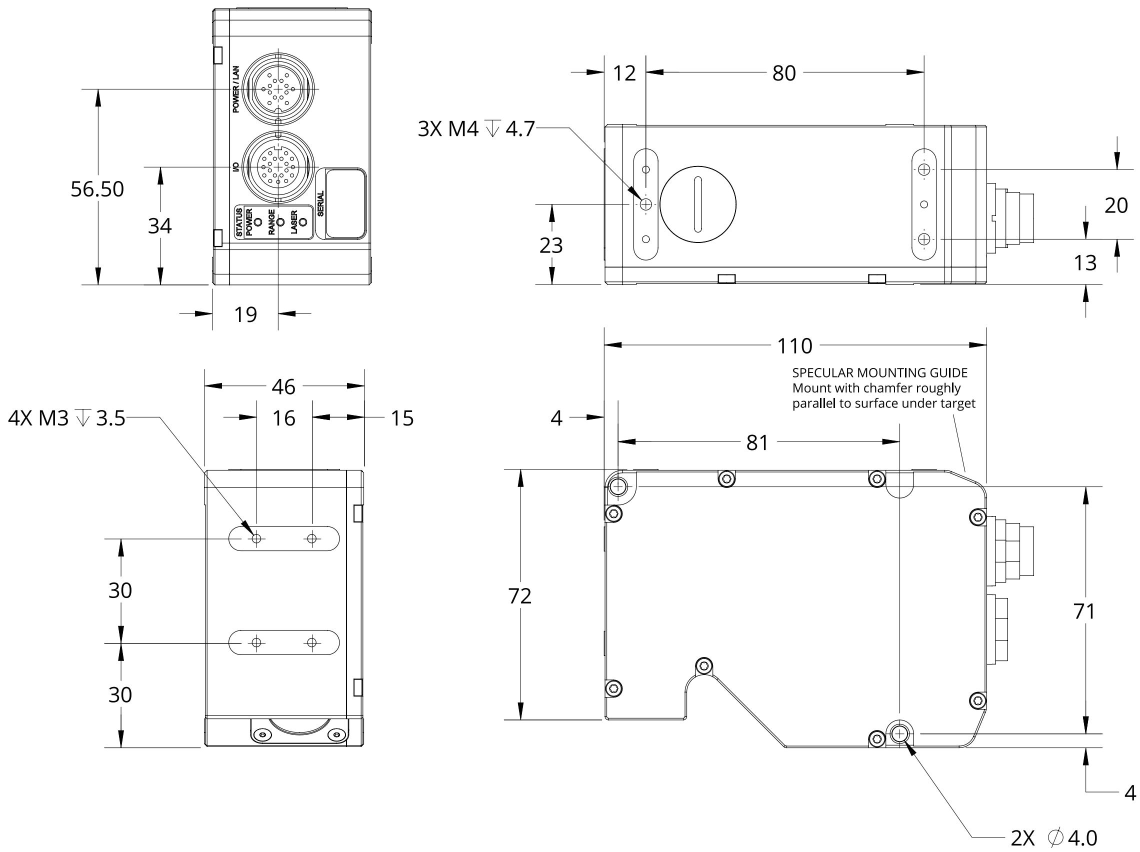
Mechanical dimensions, CD/FOV/MR, and the envelope for each sensor model are illustrated on the following pages.

Gocator 2510/2512

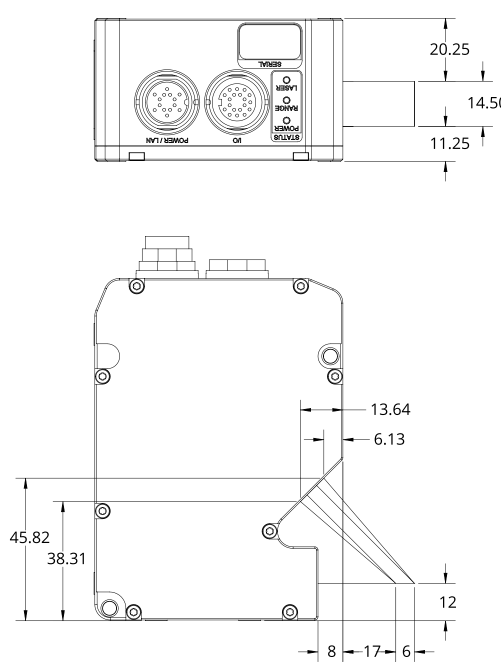
Field of View / Measurement Range / Coordinate System Orientation



Dimensions

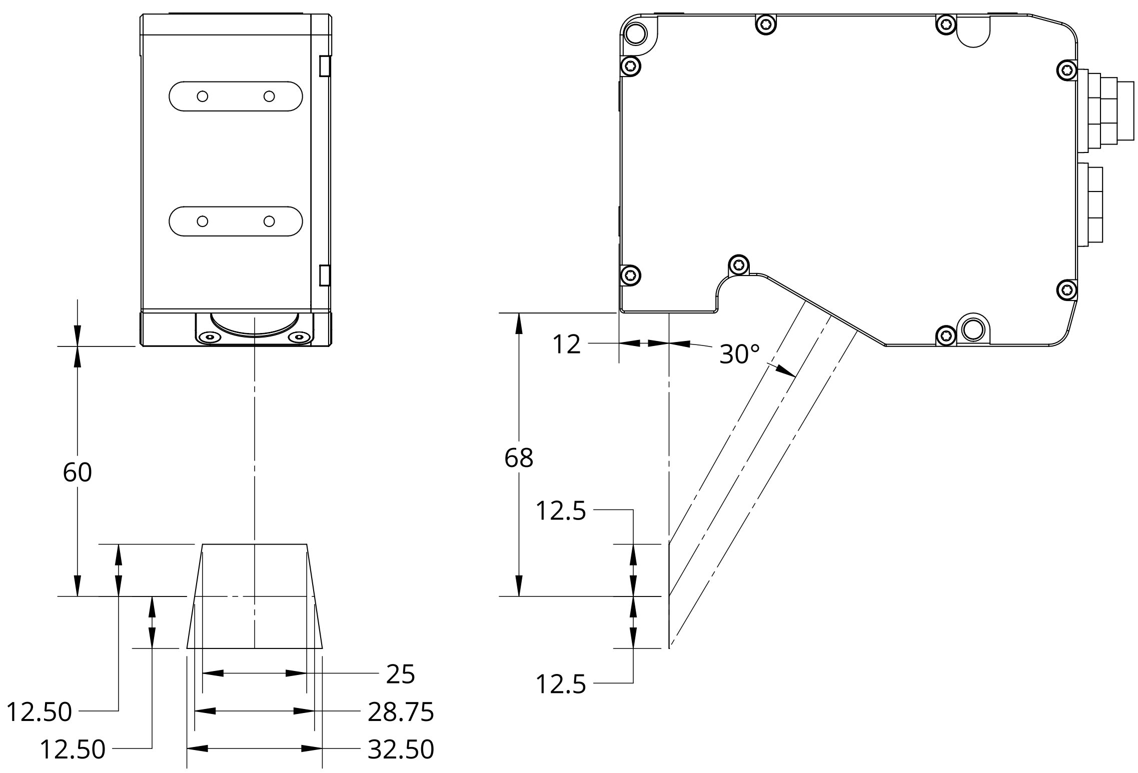


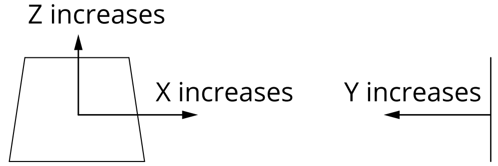
Envelope



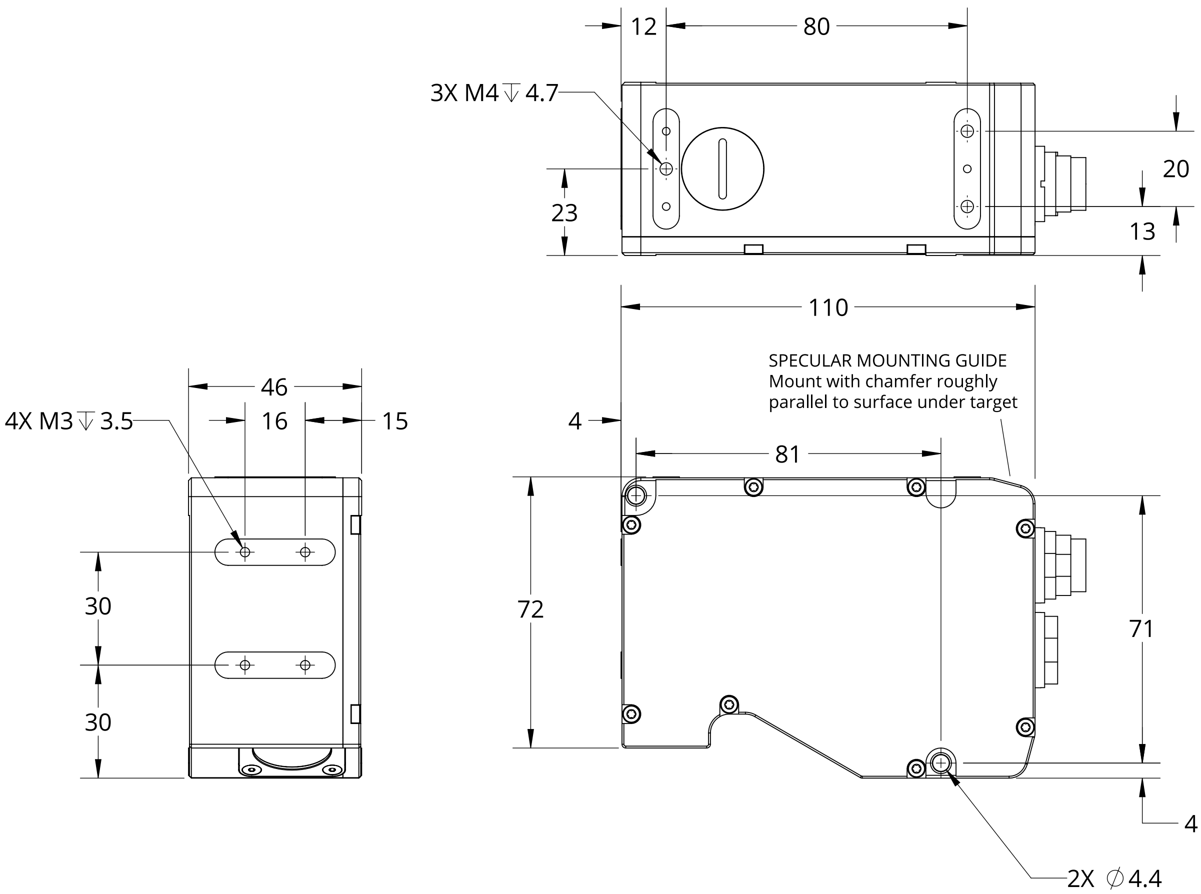
Gocator 2520

Field of View / Measurement Range / Coordinate System Orientation

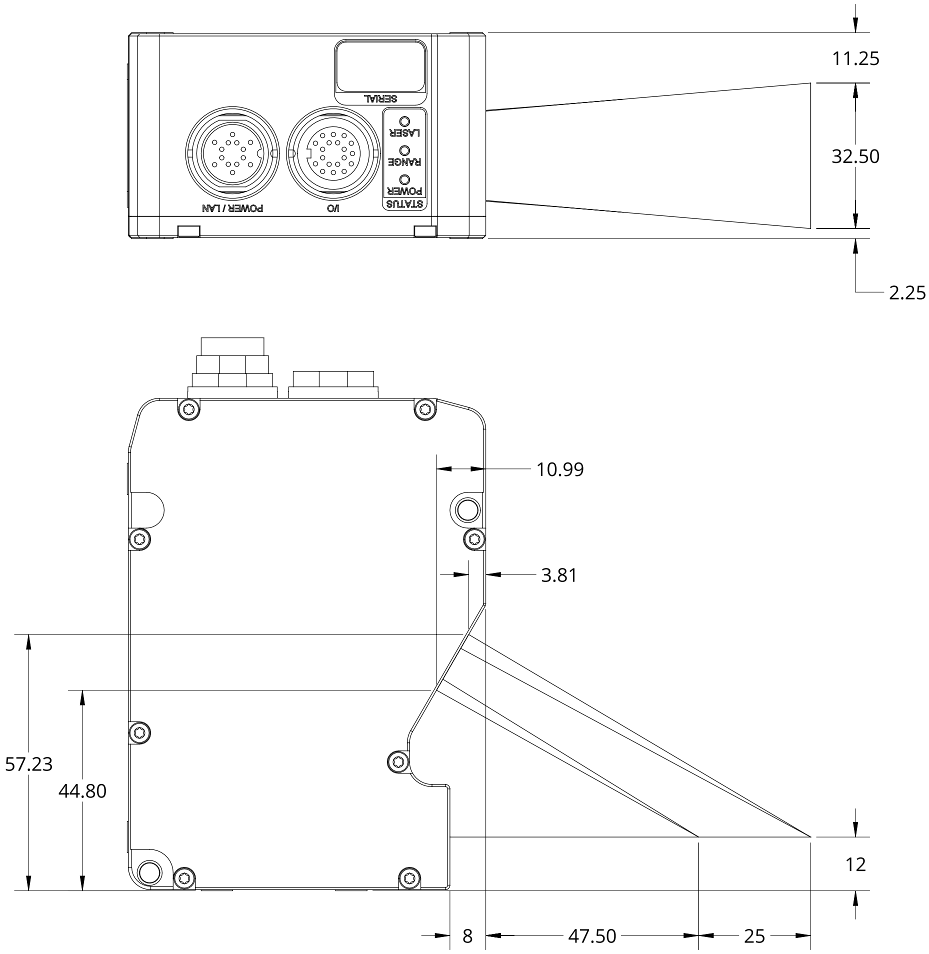




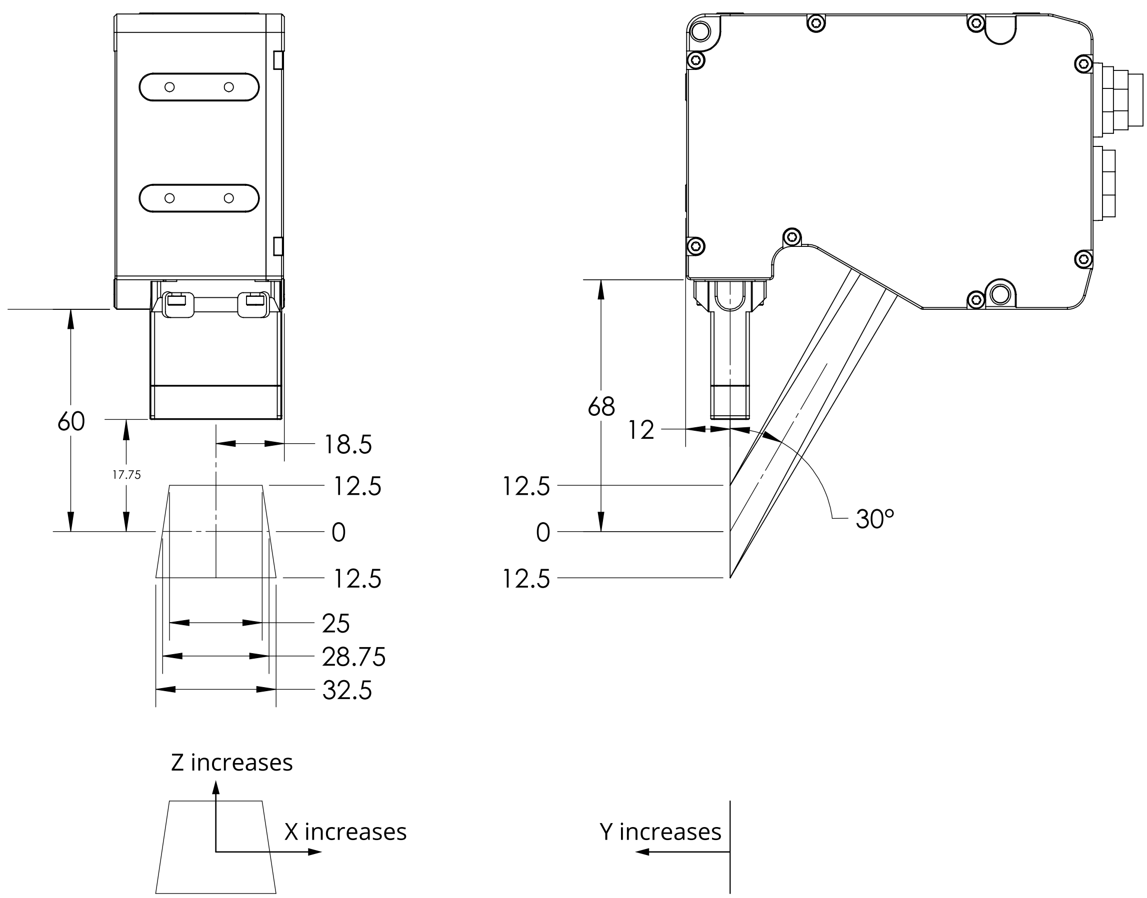
Dimensions


12
80
3X M4 ⟂ 4.7
23
20
13
110
SPECULAR MOUNTING GUIDE
Mount with chamfer roughly
parallel to surface under target
46
4X M3 ⟂ 3.5
16
15
4
81
30
30
72
71
4
2X ∅4.4

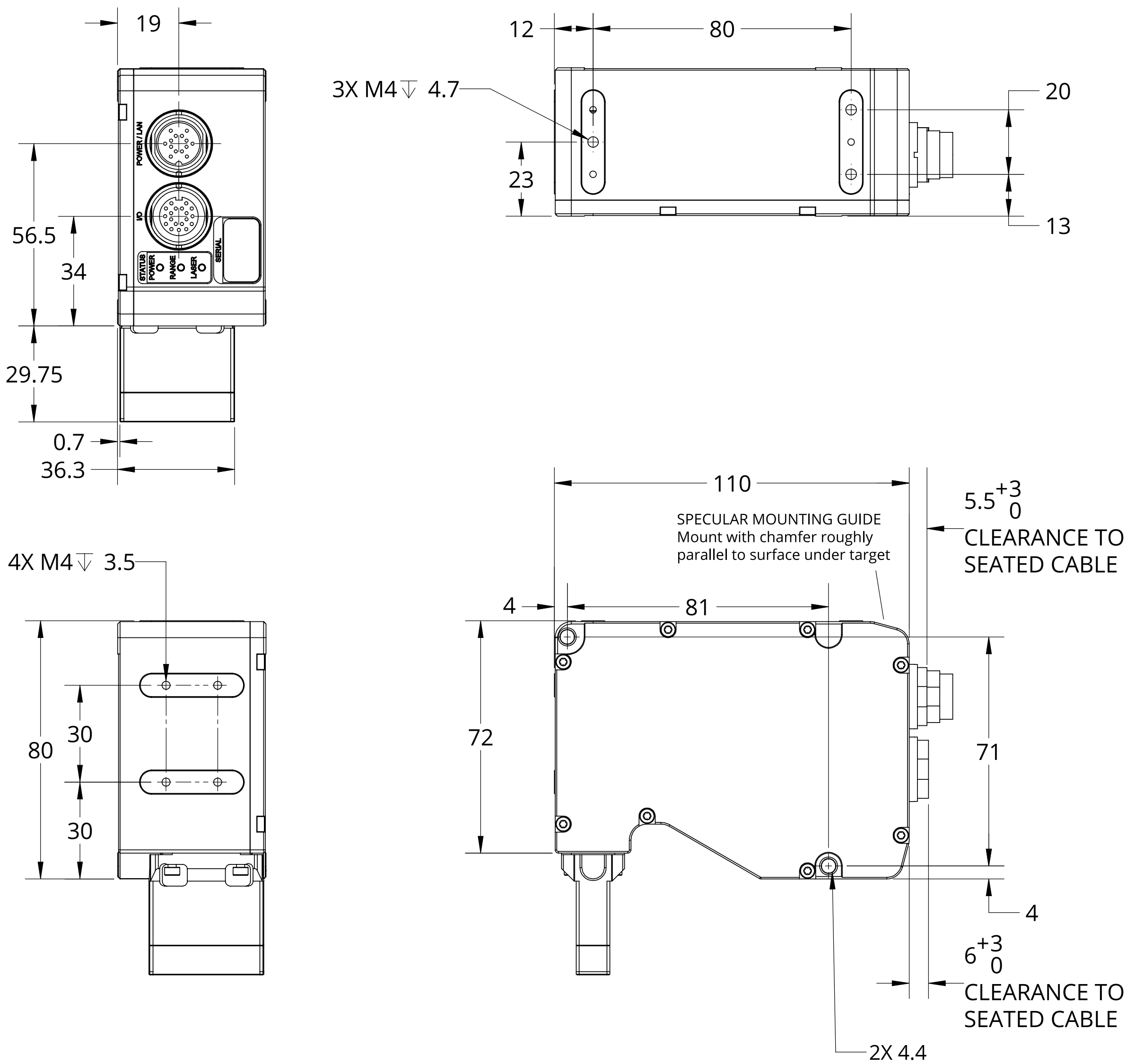
Envelope


11.25
32.50
2.25
SERIAL
LASER
RANGE
POWER
STATUS
POWER / LAN
I/O
10.99
3.81
57.23
44.80
12
8
47.50
25

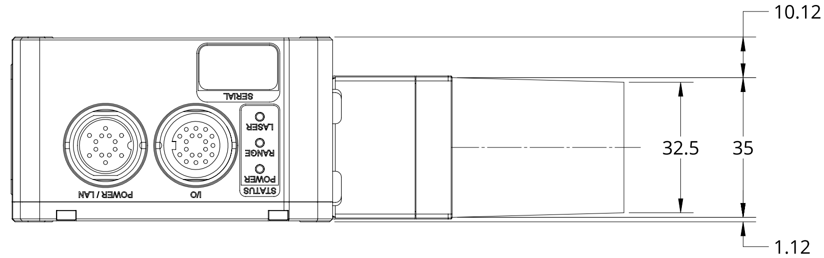
Field of View / Measurement Range / Coordinate System Orientation



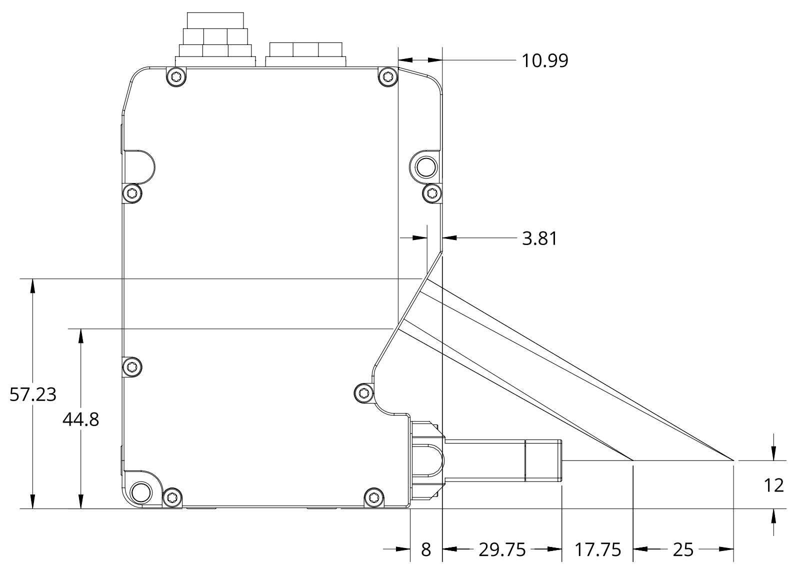
Dimensions


19
POWER / LAN
I/O
STATUS
POWER
RANGE
LASER
SERIAL
56.5
34
29.75
0.7
36.3
12
80
3X M4 ↧ 4.7
23
20
13
4X M4 ↧ 3.5
80
30
30
110
5.5 +3 0
CLEARANCE TO SEATED CABLE
SPECULAR MOUNTING GUIDE
Mount with chamfer roughly parallel to surface under target
4
81
72
71
4
6 +3 0
CLEARANCE TO SEATED CABLE
2X 4.4

Envelope



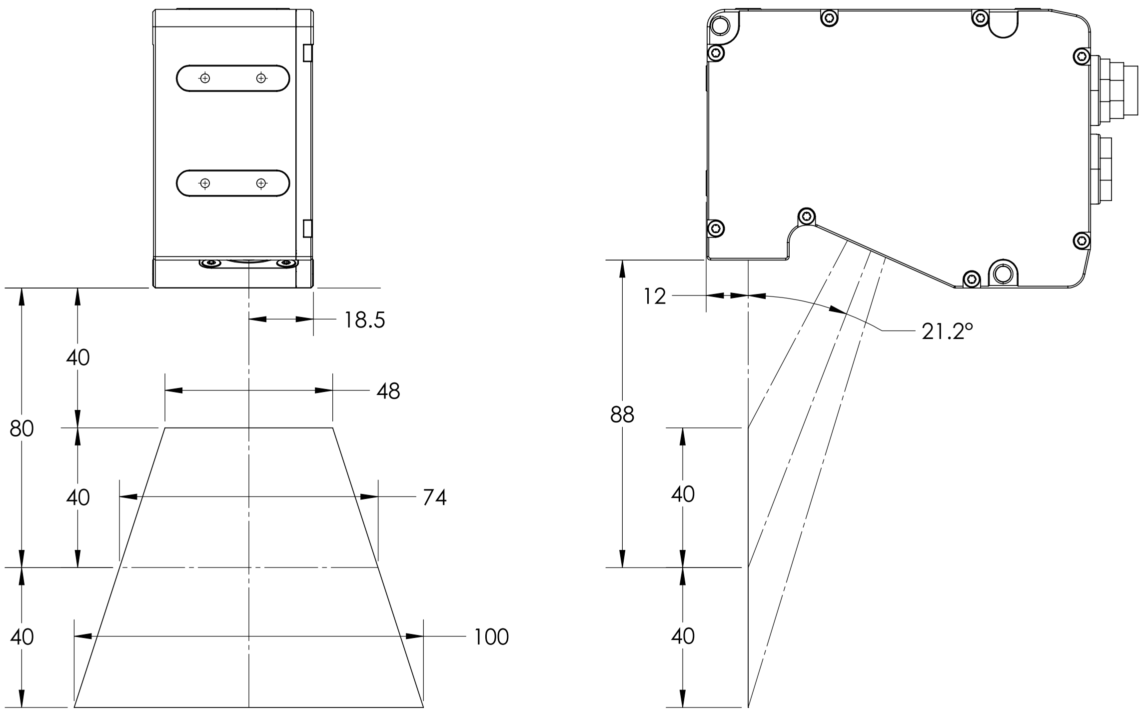
10.12
SERIAL
LASER
RANGE
POWER
STATUS
POWER / LAN
I/O
32.5
35
1.12

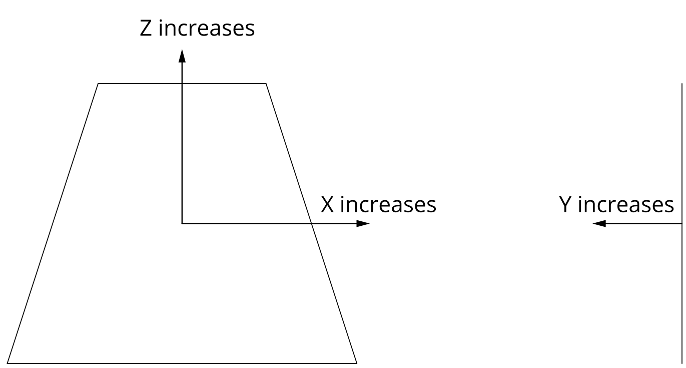


10.99
3.81
57.23
44.8
12
8
29.75
17.75
25

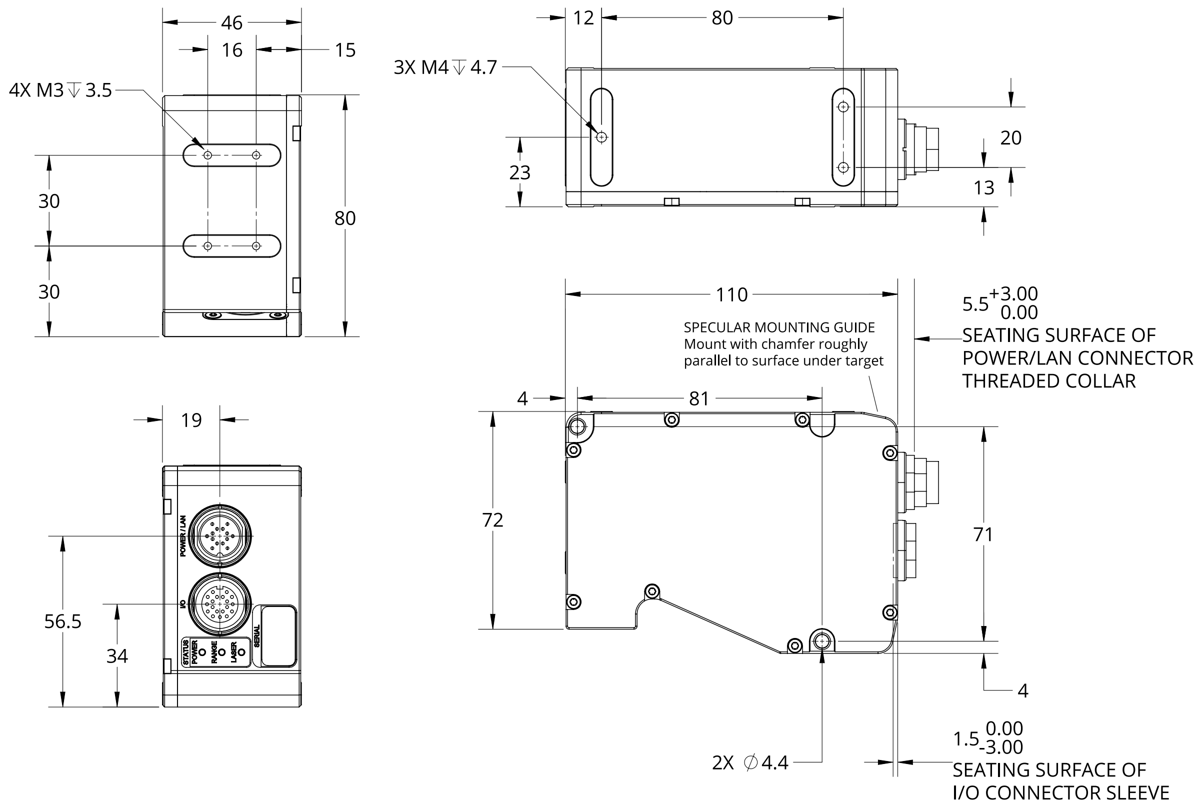
Gocator 2530

Field of View / Measurement Range / Coordinate System Orientation



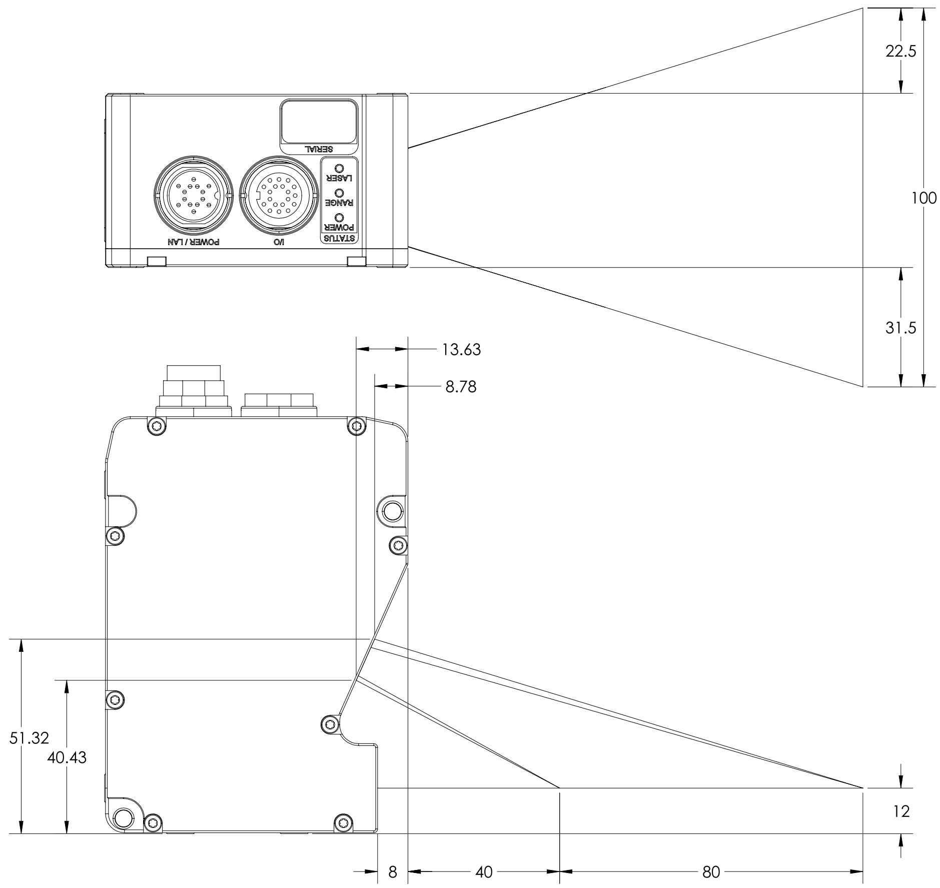


Dimensions

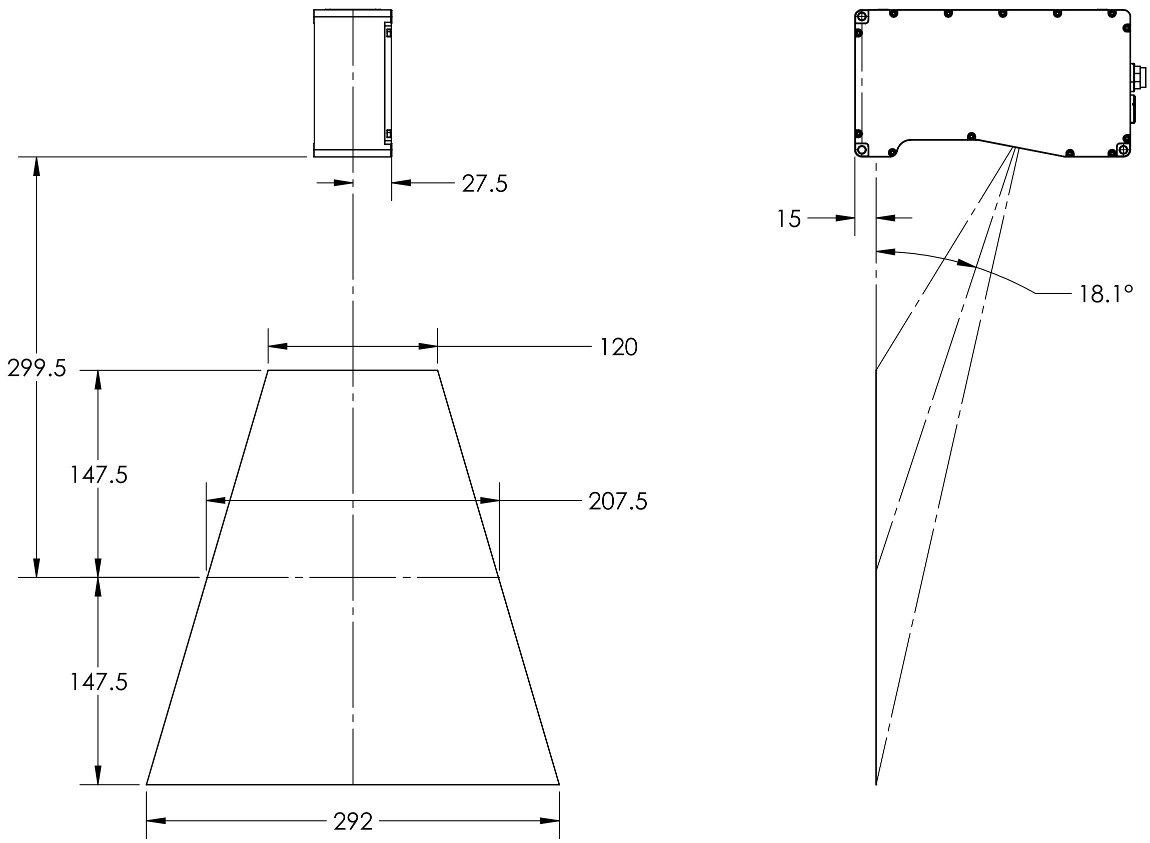


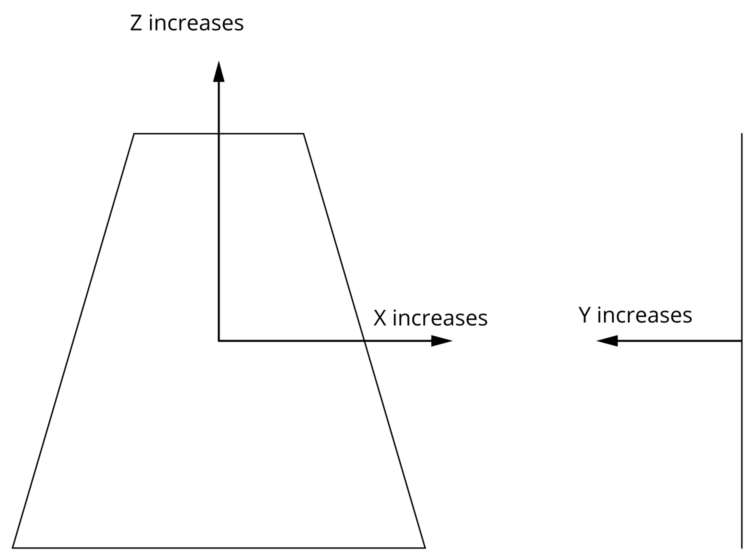
46
16
15
4X M3 ⟂ 3.5
30
30
80
12
80
3X M4 ⟂ 4.7
23
20
13
110
5.5 +3.00 / 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
SPECULAR MOUNTING GUIDE
Mount with chamfer roughly
parallel to surface under target
4
81
19
72
71
56.5
34
4
2X ∅4.4
1.5 0.00 / -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE
POWER / LAN
I/O
STATUS
POWER
RANGE
LASER
SERIAL

Envelope

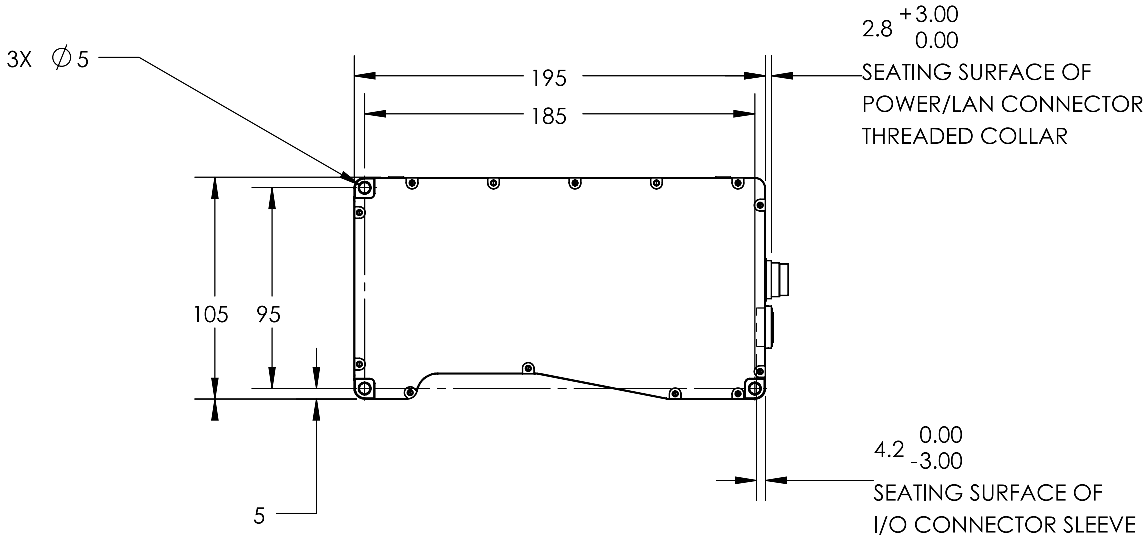

22.5
SERIAL
LASER
RANGE
POWER
STATUS
POWER / LAN
I/O
100
31.5
13.63
8.78
51.32
40.43
12
8
40
80

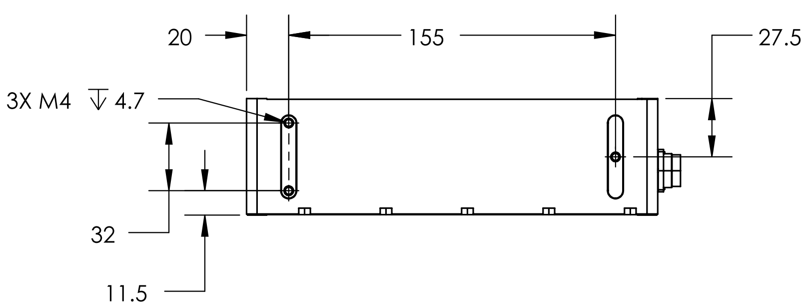
Field of View / Measurement Range / Coordinate System Orientation

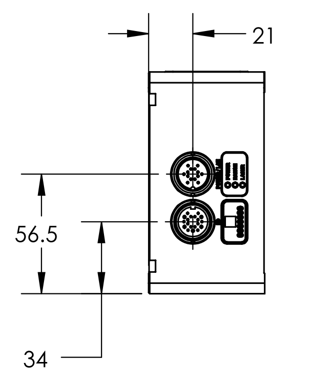




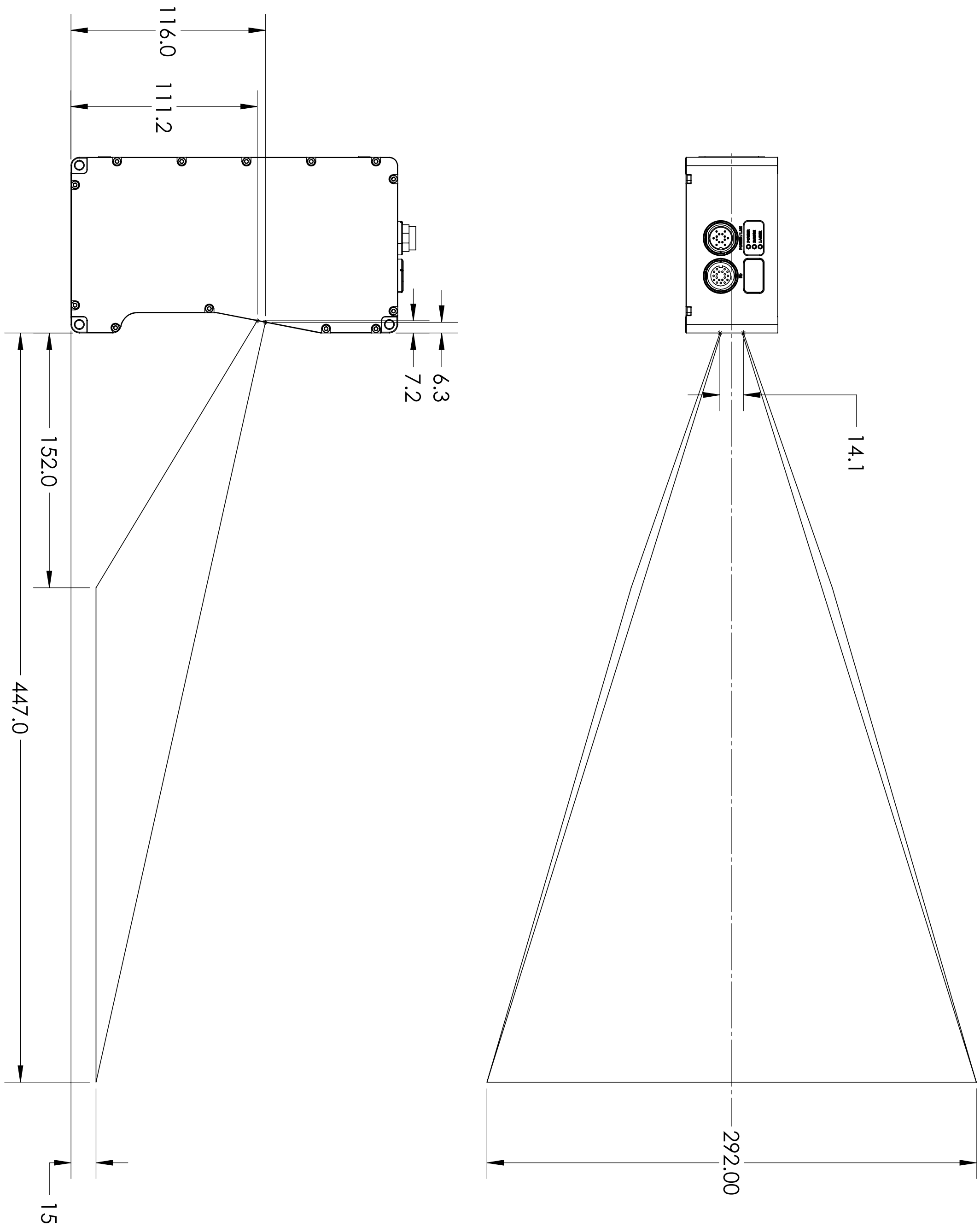
Dimensions


3X ∅5
195
185
2.8 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
105
95
5
4.2 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE

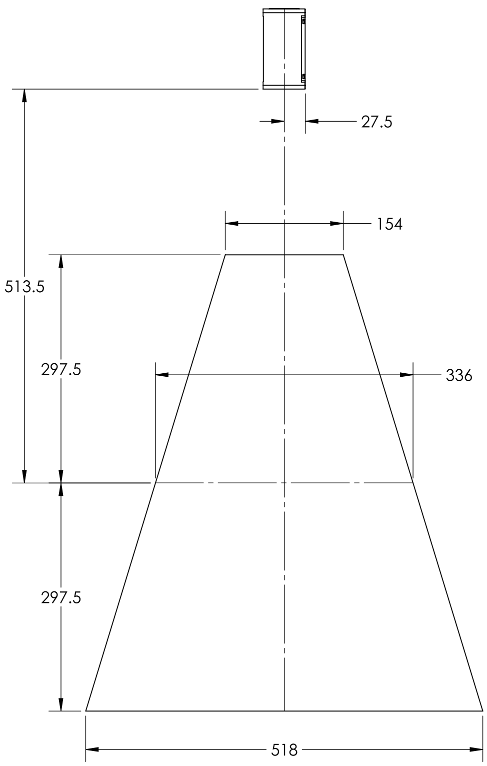

20
155
27.5
3X M4 ↧ 4.7
32
11.5

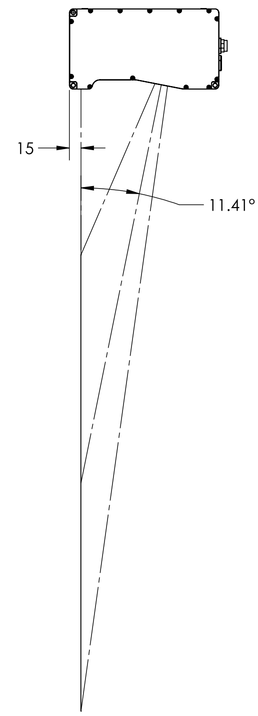

21
56.5
34

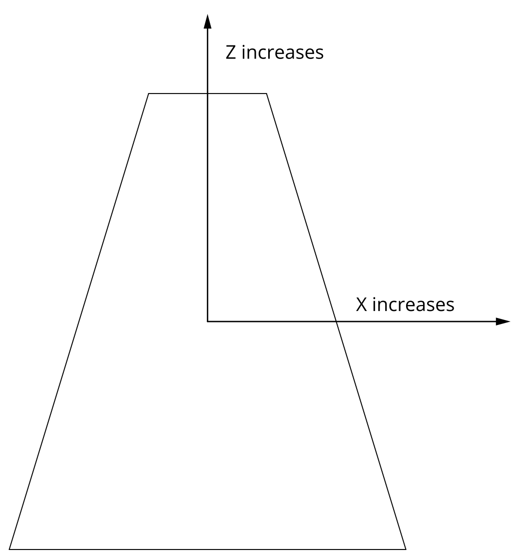
Envelope

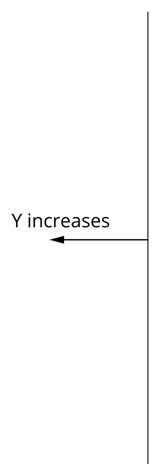


Field of View / Measurement Range / Coordinate System Orientation

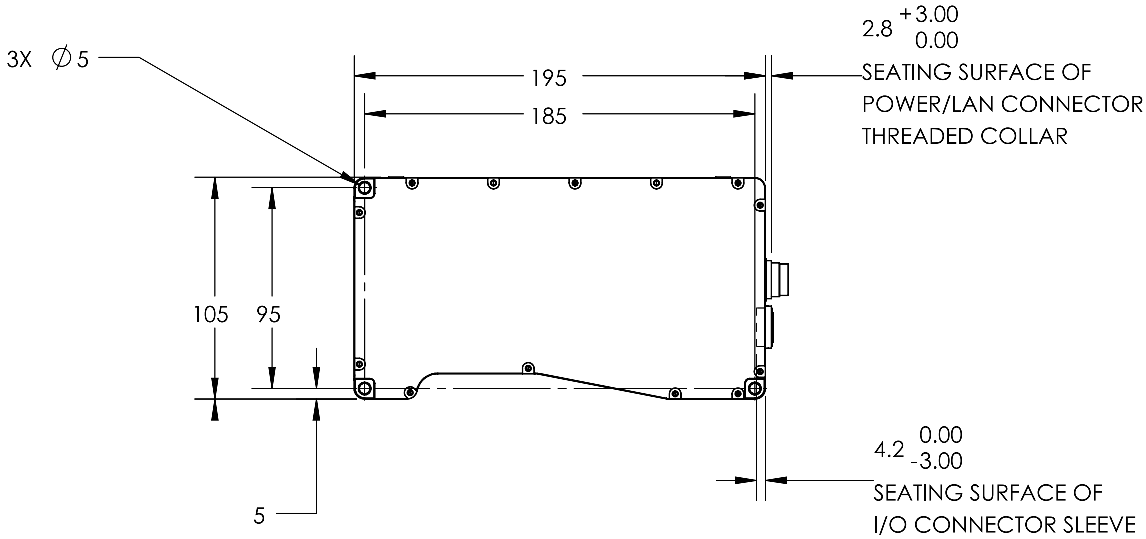


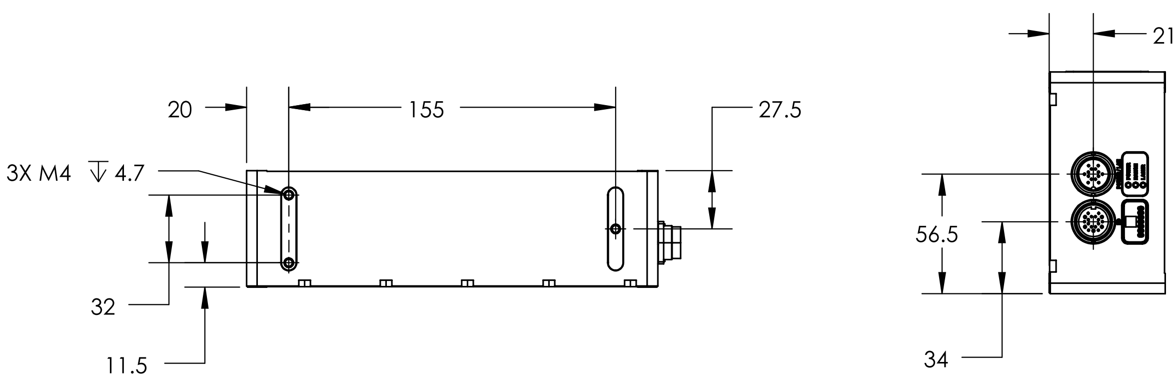




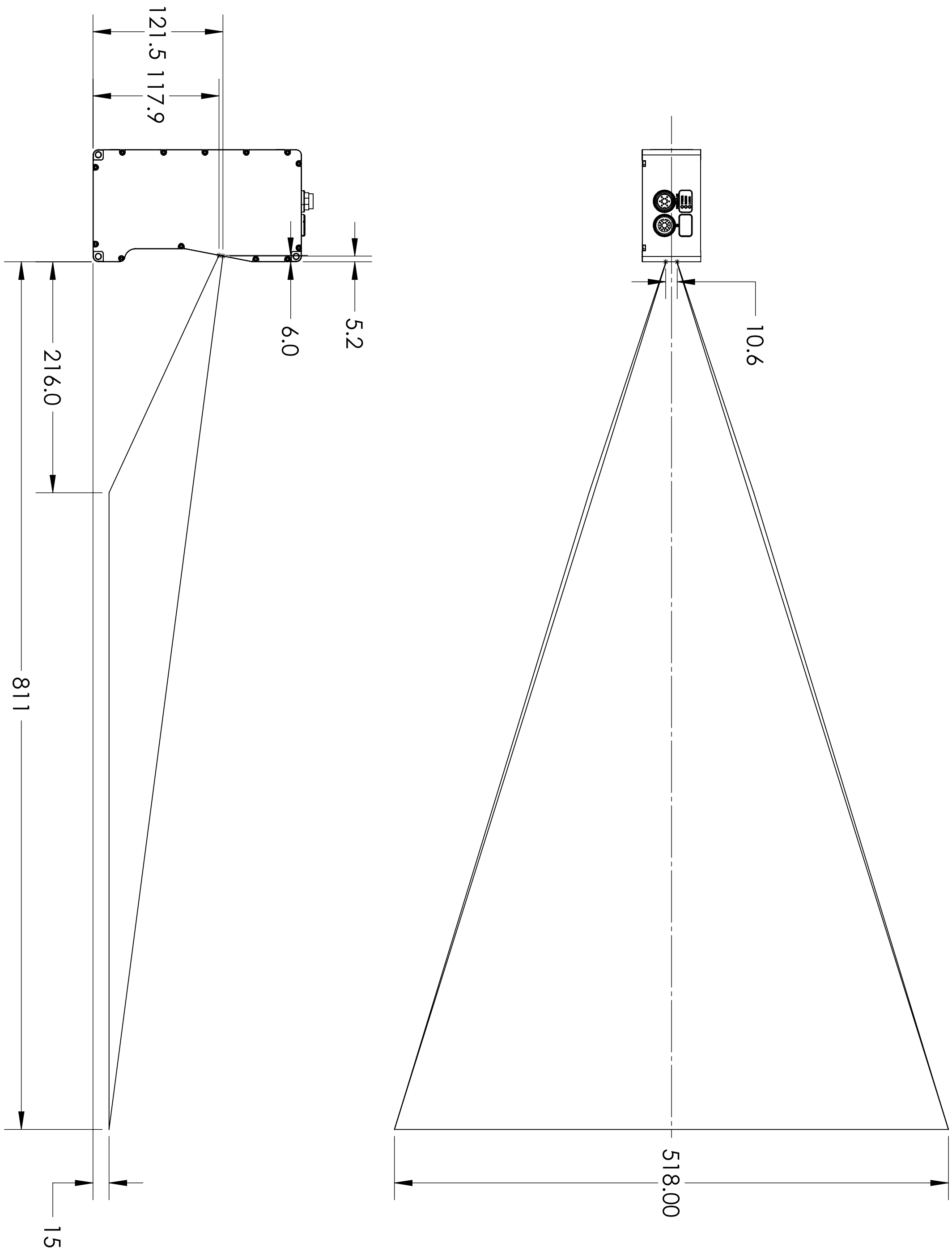


Dimensions


3X ∅ 5
195
185
2.8 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
105
95
5
4.2 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE


20
155
27.5
3X M4 ⊽ 4.7
32
11.5
21
56.5
34

Envelope



Gocator 2600 Series

The Gocator 2600 series consists of the following models:

MODEL	2610	2618	2630	2640	2650	2670	2690
Data Points / Profile	4192	4192	4192	4192	4192	4192	3700
Scan Rate (Hz)*	1100 - 9000	700 - 10000	600 - 9000	600 - 9000	600 - 9000	600 - 9000	900 - 10000
Resolution X (µm) (Profile Data Interval)	2.5	5.0 - 5.4	18 - 33	28 - 46	47 - 104	67 - 197	124 - 550
Linearity Z (+/- % of MR) **	0.015	0.015	0.03	0.04	0.04	0.05	0.08
Repeatability Z (µm) **	0.22	0.38	0.30	1.00	2.70	10.00	12.00
Clearance Distance (CD) (mm)	19.4	44.5	110	170	330	495	325
Measurement Range (MR) (mm)	5.0	12	130	190	475	1060	1550
Field of View (FOV) (mm)	10.2 - 10.8	20 - 23	71 - 135	105 - 198	190 - 430	272 - 817	385 - 2000
Laser Classes	2, 3R, 3B (blue, 405 nm)	2, 3R, 3B (blue, 405 nm)	2, 3R, 3B (blue, 405 nm)	2, 3R, 3B (blue, 405 nm)	2, 3R, 3B (blue, 405 nm)	2, 3R, 3B (blue, 405 nm)	2, 3R (red, 660 nm)
Dimensions (mm)	50 x 116 x 125	46 x 80 x 110	55 x 105 x 165	55 x 105 x 195	55 x 105 x 280	55 x 105 x 280	55 x 105 x 280
Weight (kg)	0.9	0.65	1.34	1.48	2.12	2.12	2.12

* Speed ranges are from default configuration (full field of view and full measurement range) to high speed configuration (reduced field-of view and measurement range, uniform spacing disabled, optimized data spacing and output).

** These results are achieved with LMI standard target and optimized sensor configuration.

The following diagram illustrates some of the terms used in the table above.

Specifications stated are based on standard laser classes. Linearity Z and Repeatability Z may vary for other laser classes.

All specification measurements are performed on LMI’s standard calibration target (a diffuse, painted white surface).

Linearity Z is the worst case difference in average height measured, compared to the actual position over the measurement range.

Resolution X is the distance between data points along the laser line.

Repeatability Z is measured with a flat target at the middle of the measurement range. It is the 95% confidence variation of the average height over 4096 frames. Height values are averaged over the full FOV.

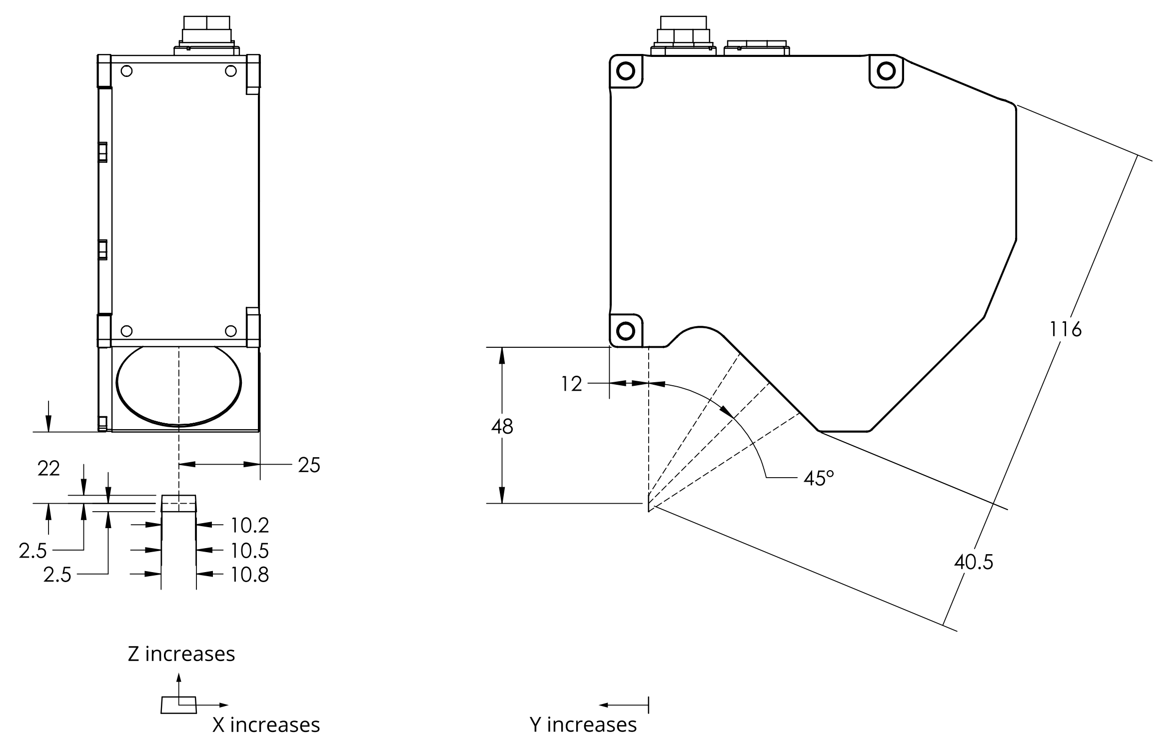
See *Resolution and Linearity* on page 62 for more information.

ALL 2600 SERIES MODELS	
Interface	Gigabit Ethernet
Inputs	Differential Encoder, Laser Safety Enable, Trigger
Outputs	2x Digital output, RS-485 Serial (115 kBaud)
Factory Communication	PROFINET, Modbus, EtherNet/IP, ASCII, Gocator
Input Voltage (Power)	+24 to +48 VDC (15 W); Ripple +/- 10%
Housing	Gasketed metal enclosure, IP67
Operating Temp.	0 to 50 °C
Storage Temp.	-30 to 70 °C
Vibration Resistance	10 to 55 Hz, 1.5 mm double amplitude in X, Y, and Z directions, 2 hours per direction
Shock Resistance	15 g, half sine wave, 11 ms, positive and negative for X, Y, and Z directions

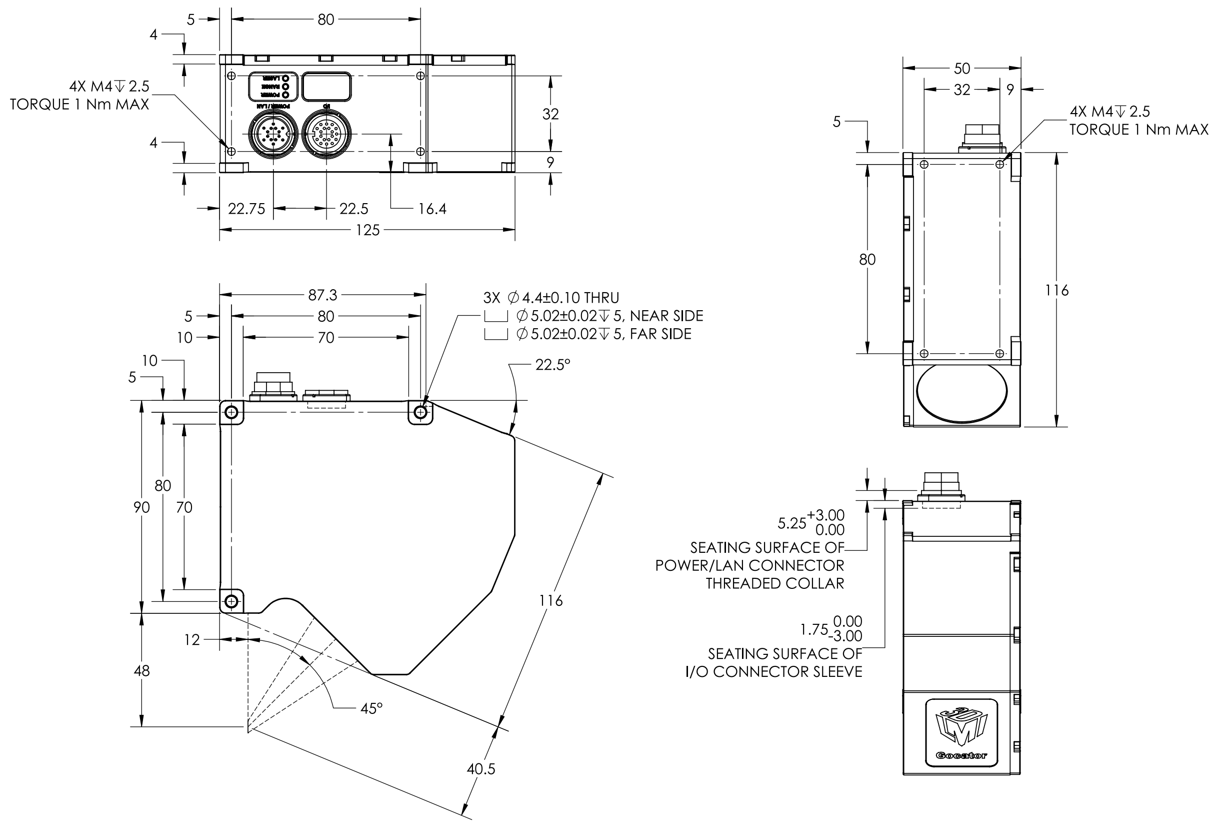
Mechanical dimensions, CD/FOV/MR, and the envelope for each sensor model are illustrated on the following pages.

Gocator 2610

Field of View / Measurement Range / Coordinate System Orientation

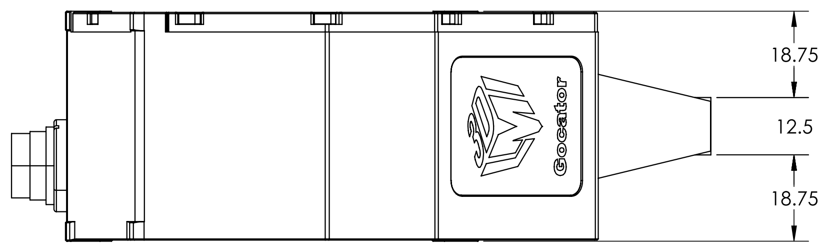


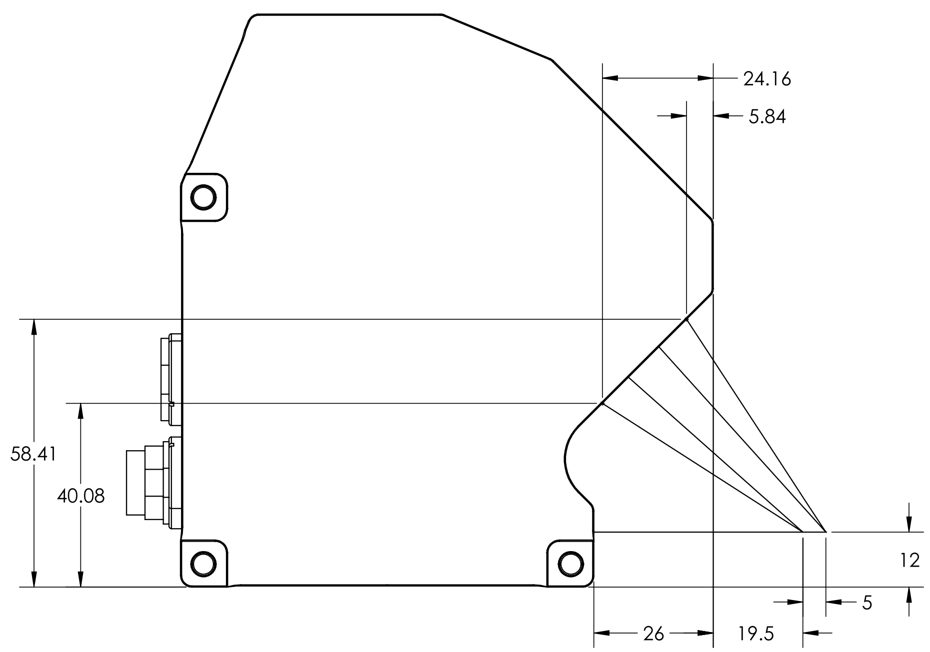
Dimensions



4X M4 ↧ 2.5
TORQUE 1 Nm MAX
3X ∅4.4±0.10 THRU
∅5.02±0.02 ↧ 5, NEAR SIDE
∅5.02±0.02 ↧ 5, FAR SIDE
22.5°
45°
5.25 +3.00 0.00
SEATING SURFACE OF POWER/LAN CONNECTOR THREADED COLLAR
1.75 0.00 -3.00
SEATING SURFACE OF I/O CONNECTOR SLEEVE

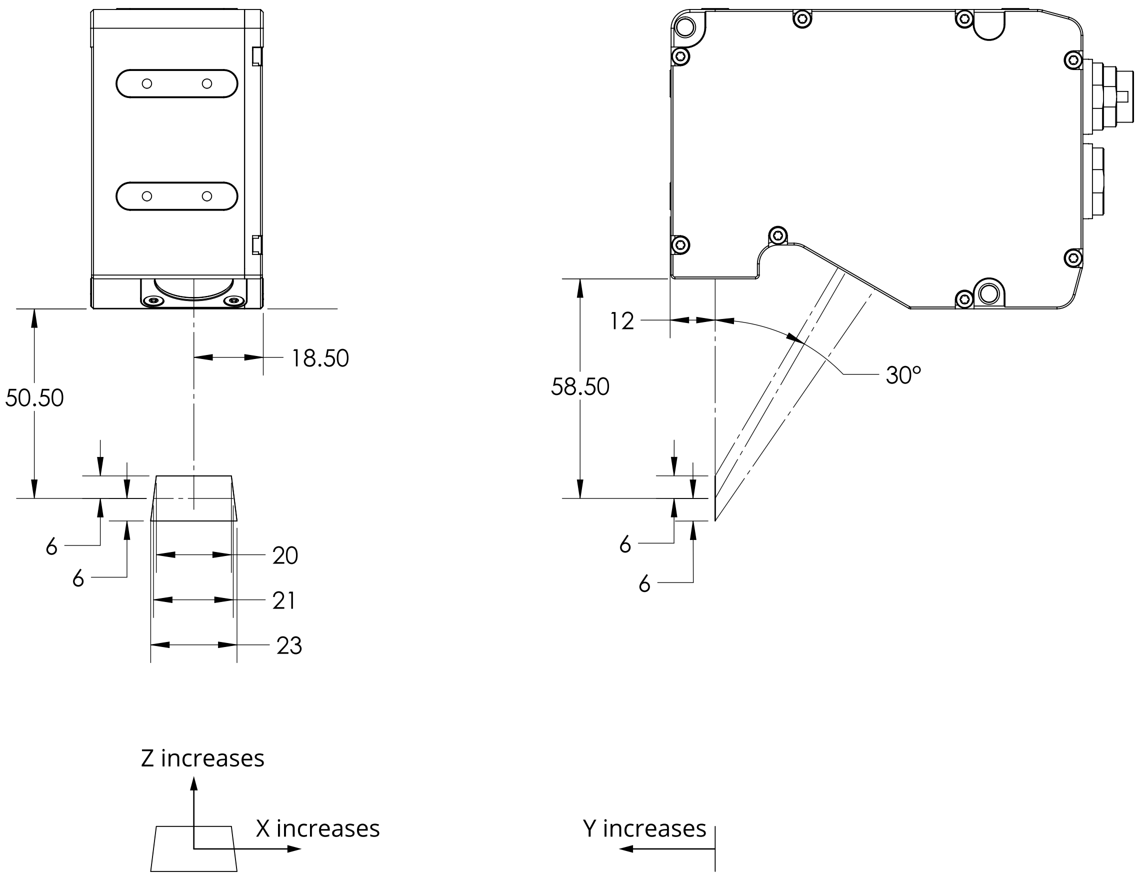
Envelope



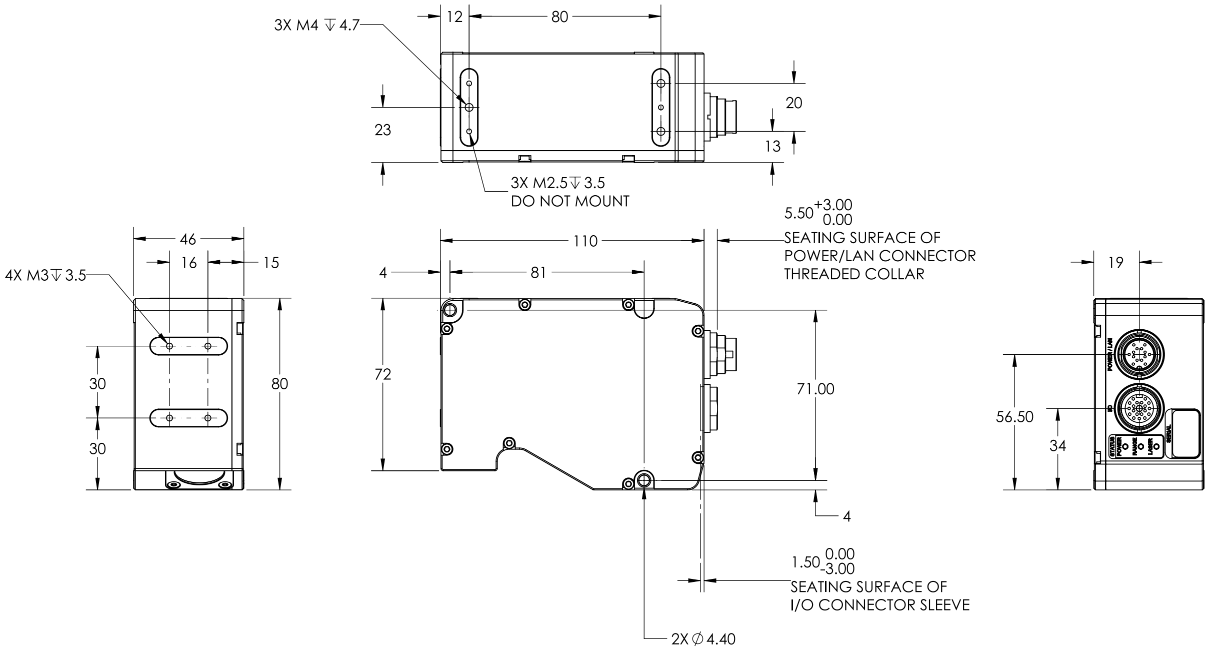


Gocator 2618

Field of View / Measurement Range / Coordinate System Orientation

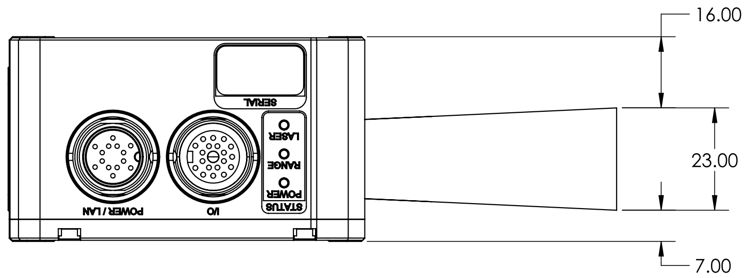


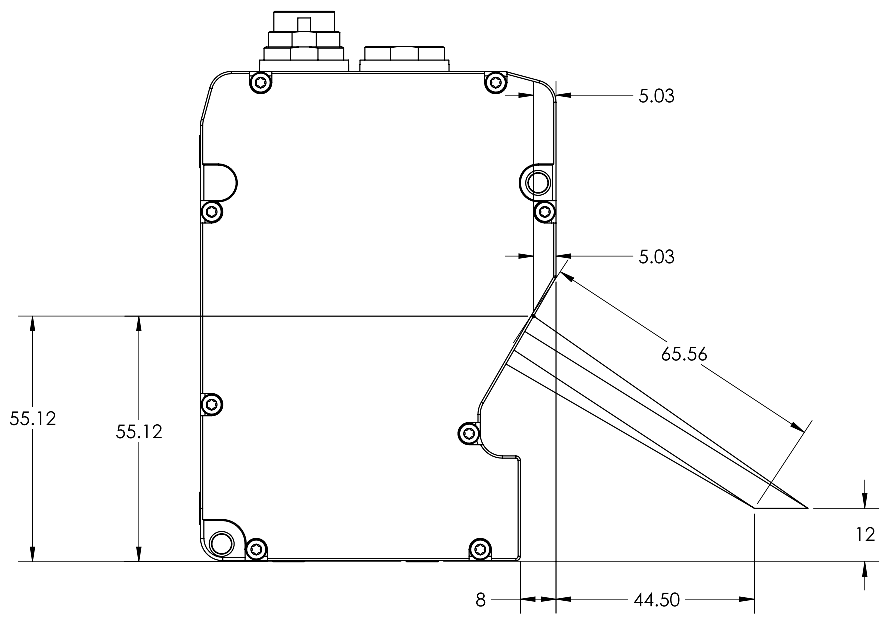
Dimensions



3X M4 ↧ 4.7
12
80
20
23
13
3X M2.5↧ 3.5
DO NOT MOUNT
5.50 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
46
16
15
4X M3↧ 3.5
110
4
81
19
30
80
72
71.00
56.50
34
30
4
1.50 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE
2X ∅ 4.40

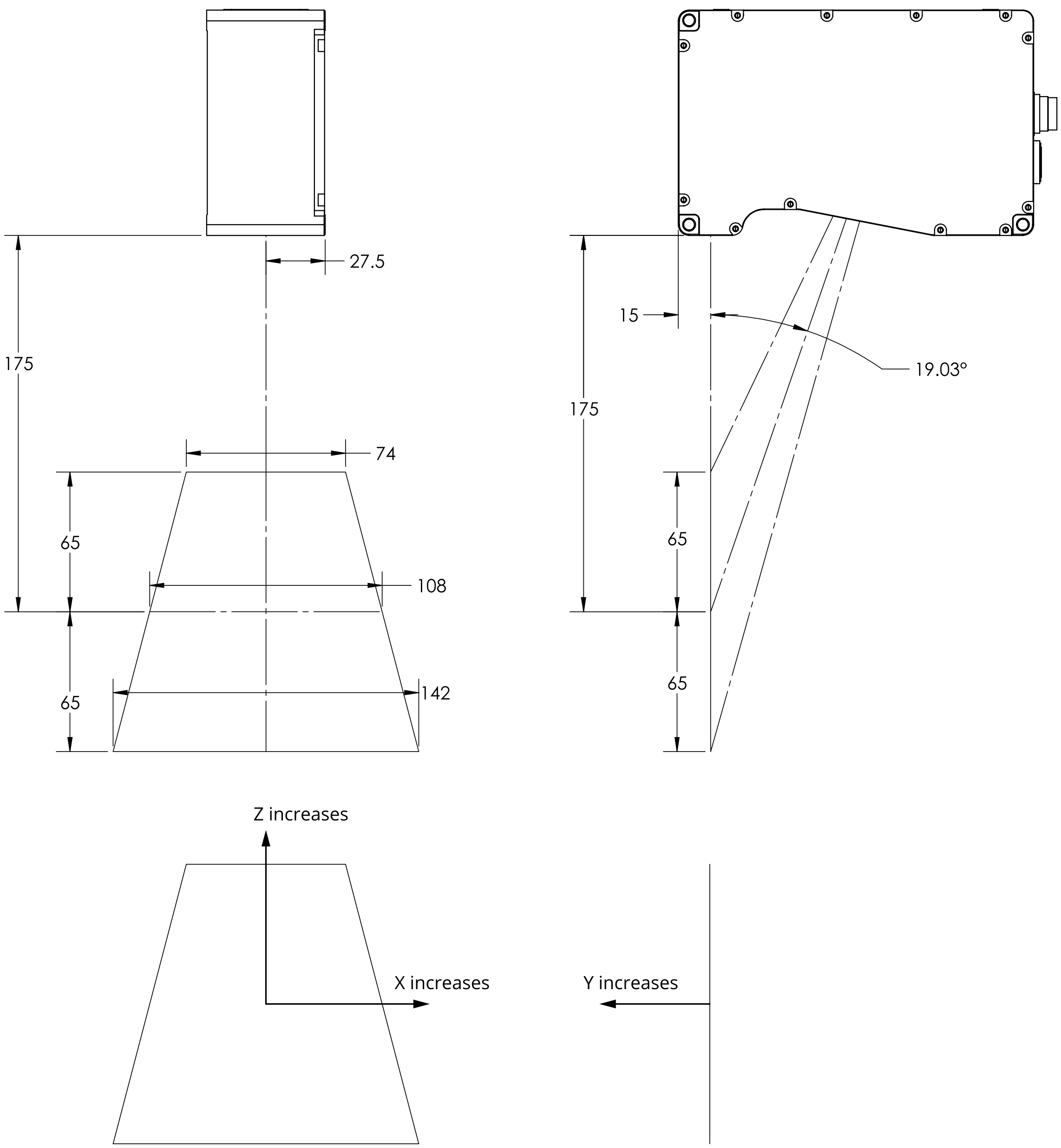
Envelope


16.00
SERIAL
LASER
RANGE
POWER
STATUS
POWER / LAN
I/O
23.00
7.00

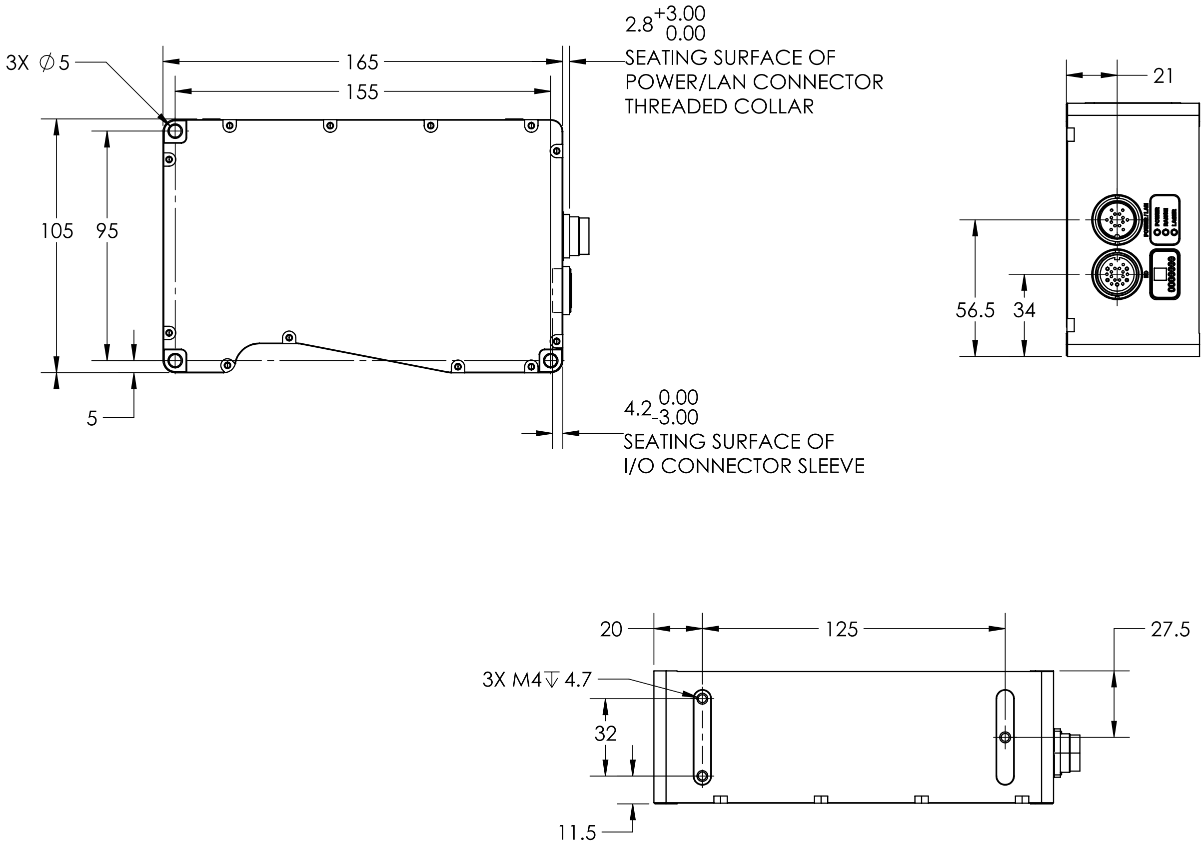

5.03
5.03
65.56
55.12
55.12
12
8
44.50

Gocator 2630

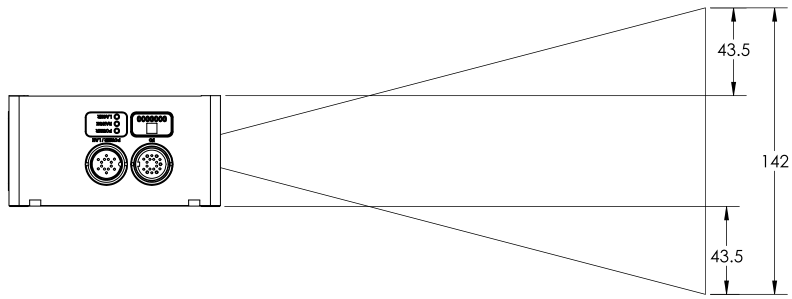
Field of View / Measurement Range / Coordinate System Orientation

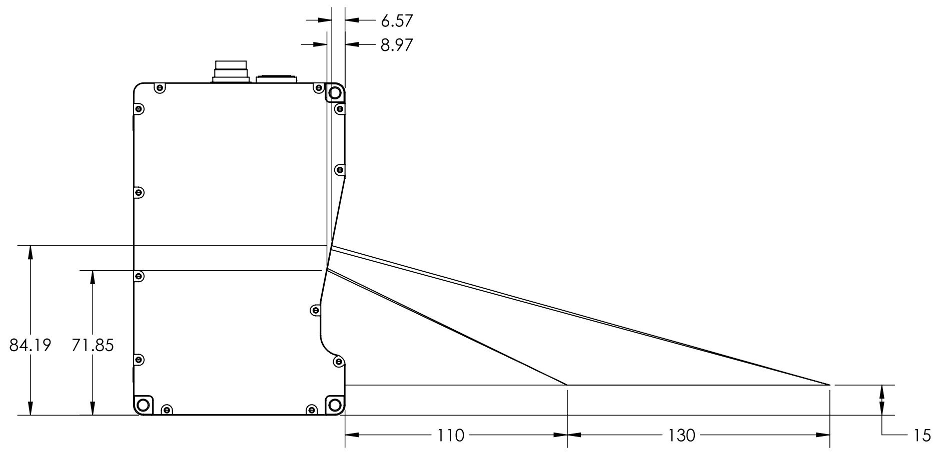


Dimensions

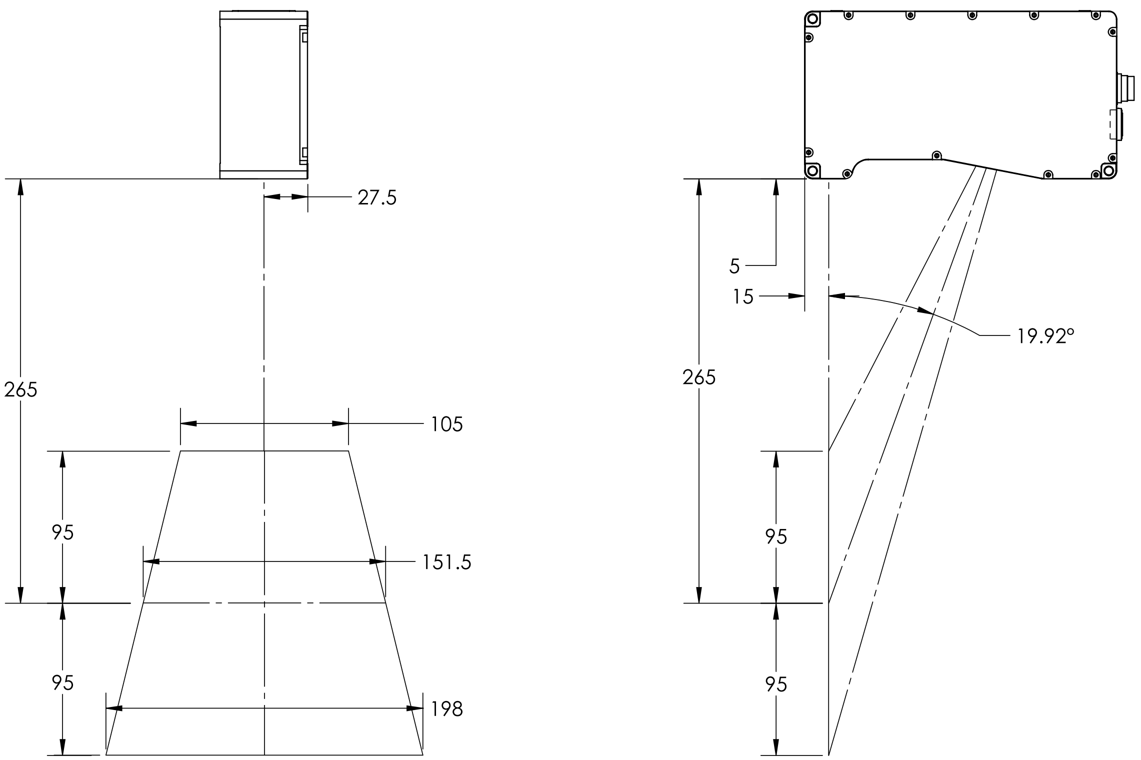

2.8 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
3X ∅5
165
155
105
95
5
4.2 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE
21
56.5
34
20
125
27.5
3X M4 ↧ 4.7
32
11.5

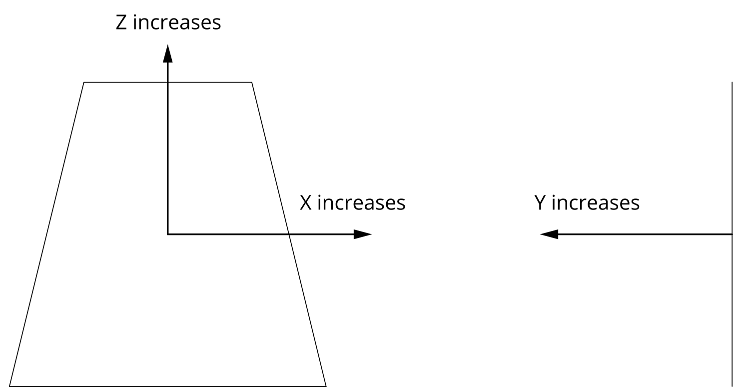
Envelope



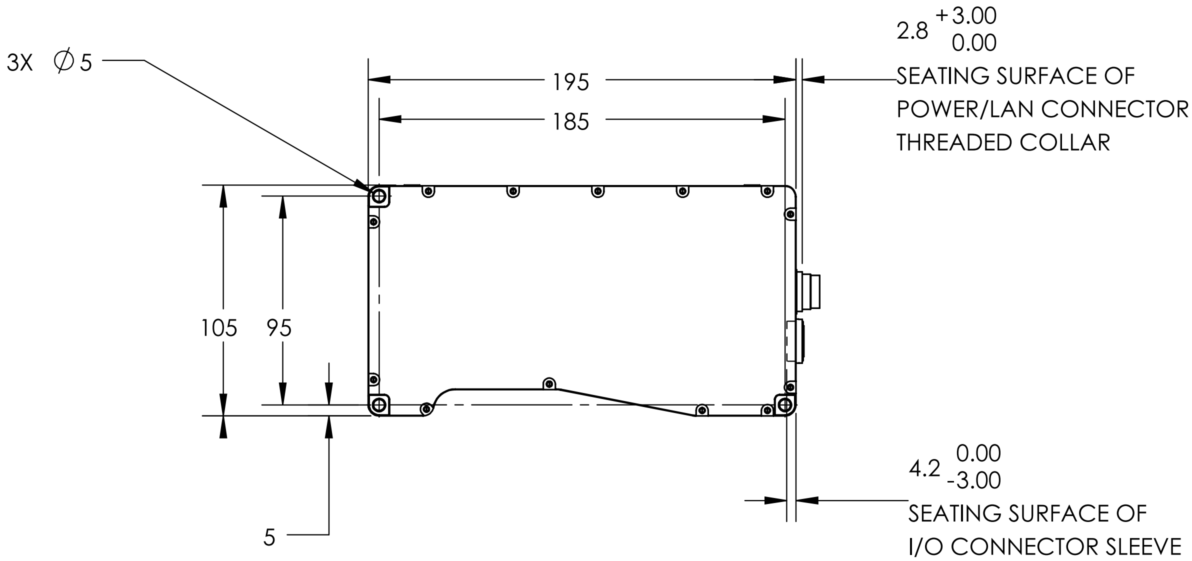


Field of View / Measurement Range / Coordinate System Orientation

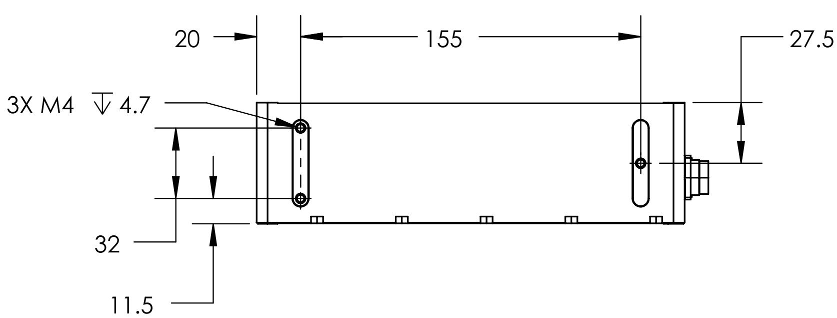




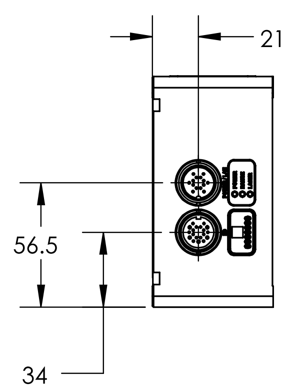
Dimensions



3X ∅5
195
185
2.8 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
105
95
5
4.2 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE

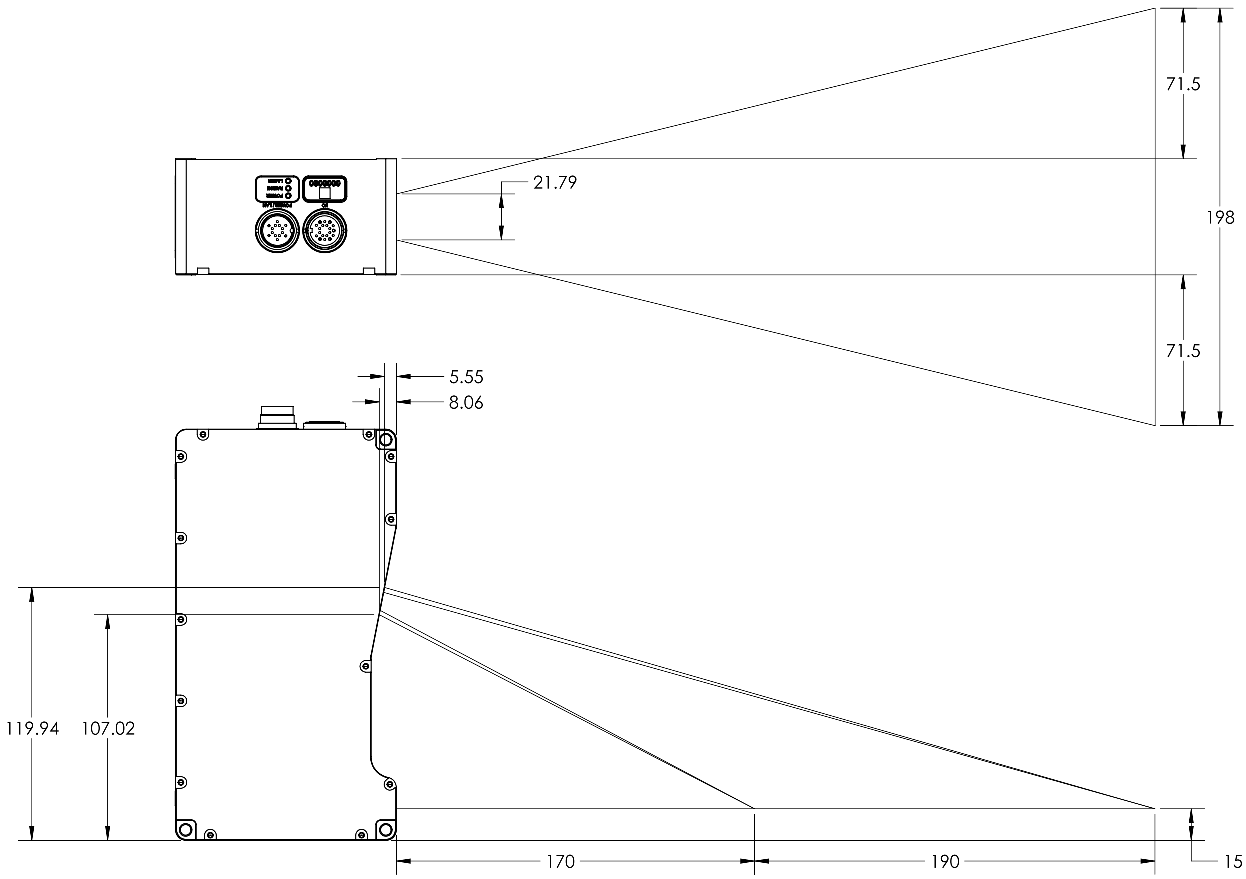


20
155
27.5
3X M4 ↧ 4.7
32
11.5



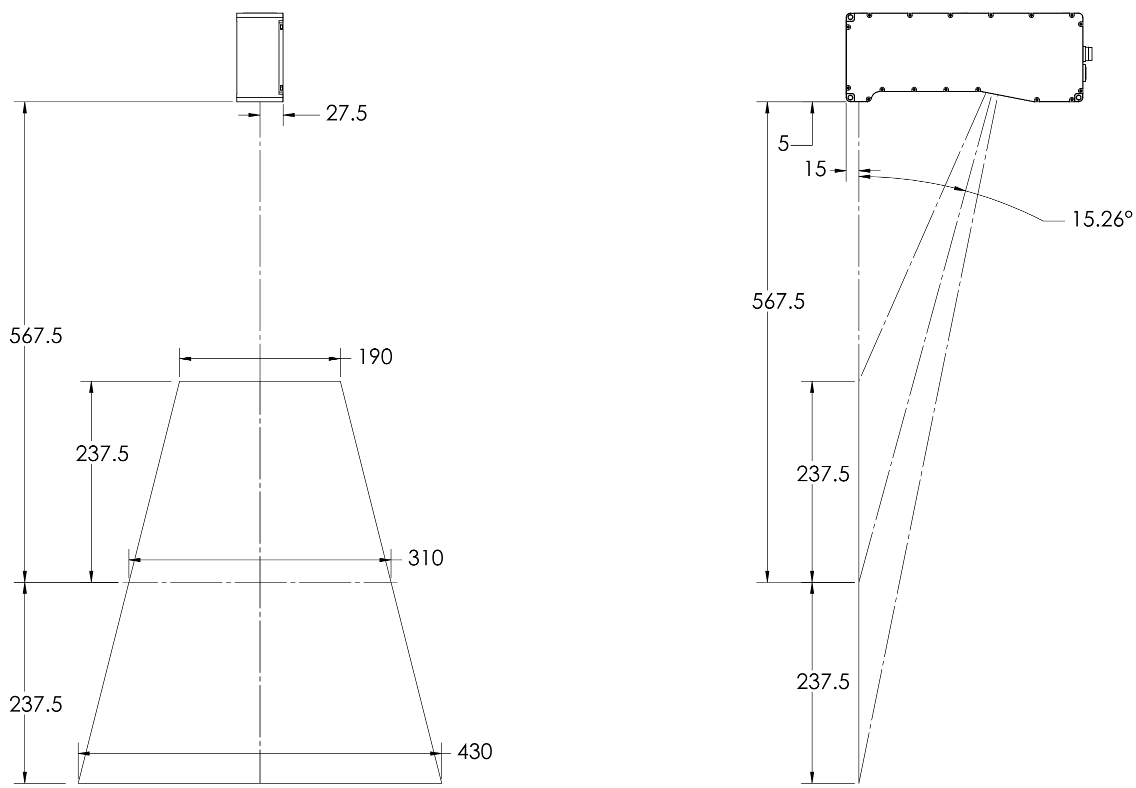
21
56.5
34

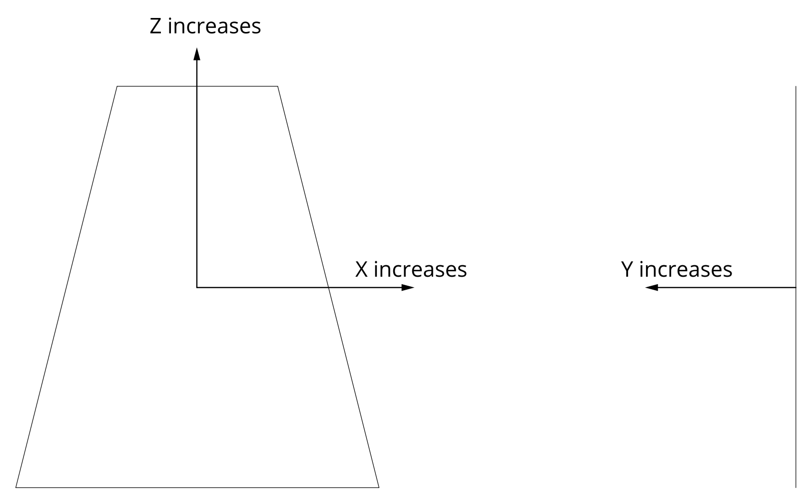
Envelope



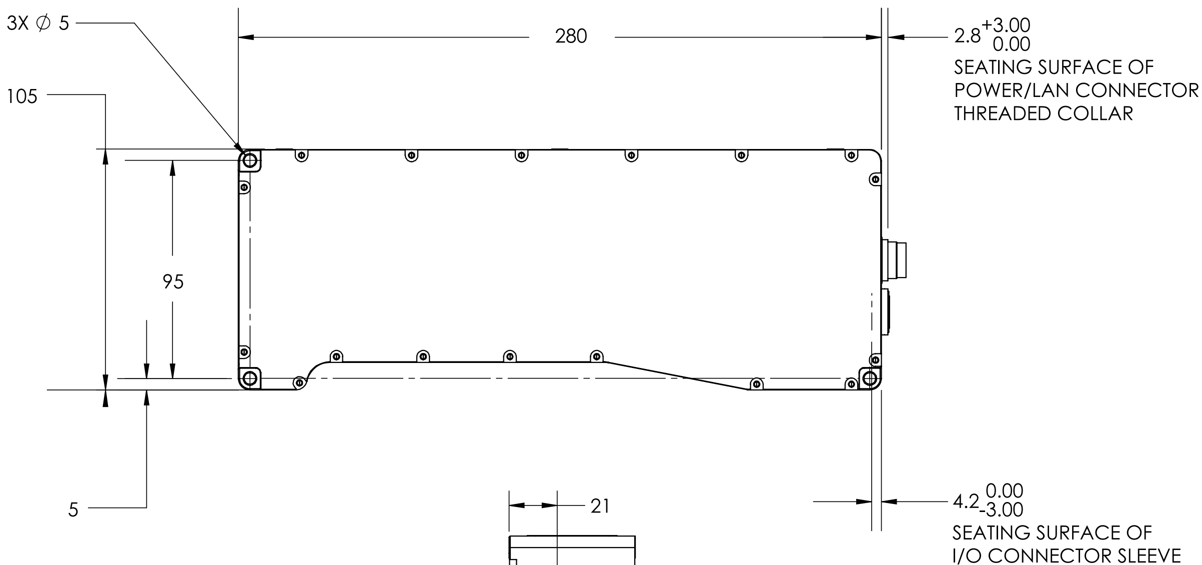
Gocator 2650

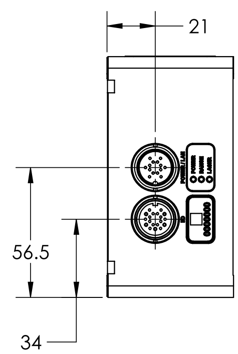
Field of View / Measurement Range / Coordinate System Orientation

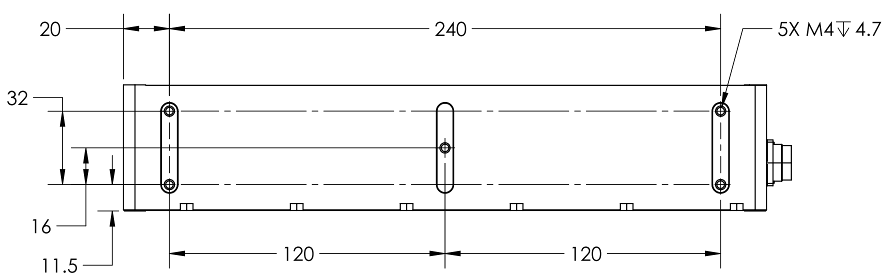




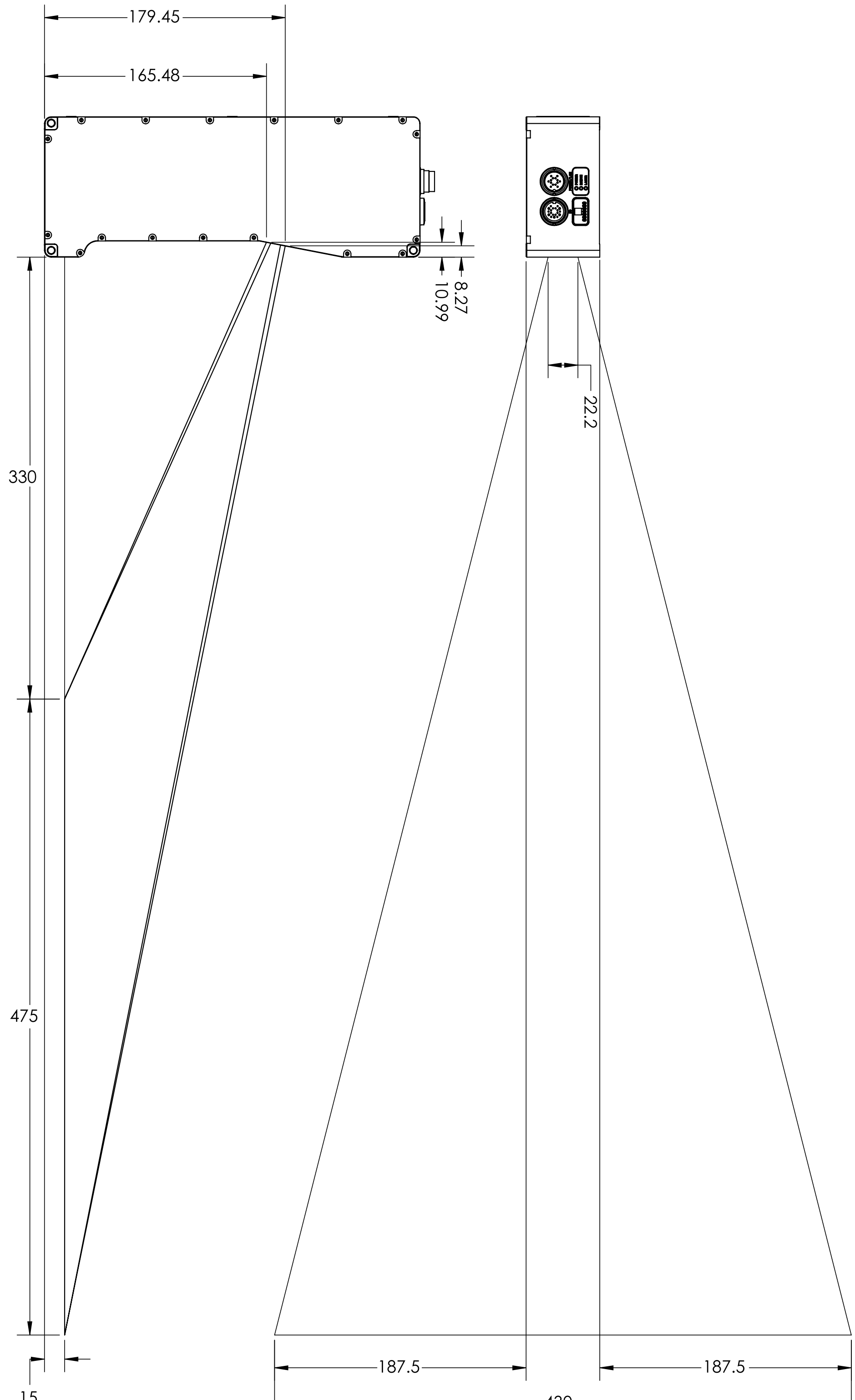
Dimensions


3X ∅ 5
280
2.8 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
105
95
5
4.2 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE


21
56.5
34

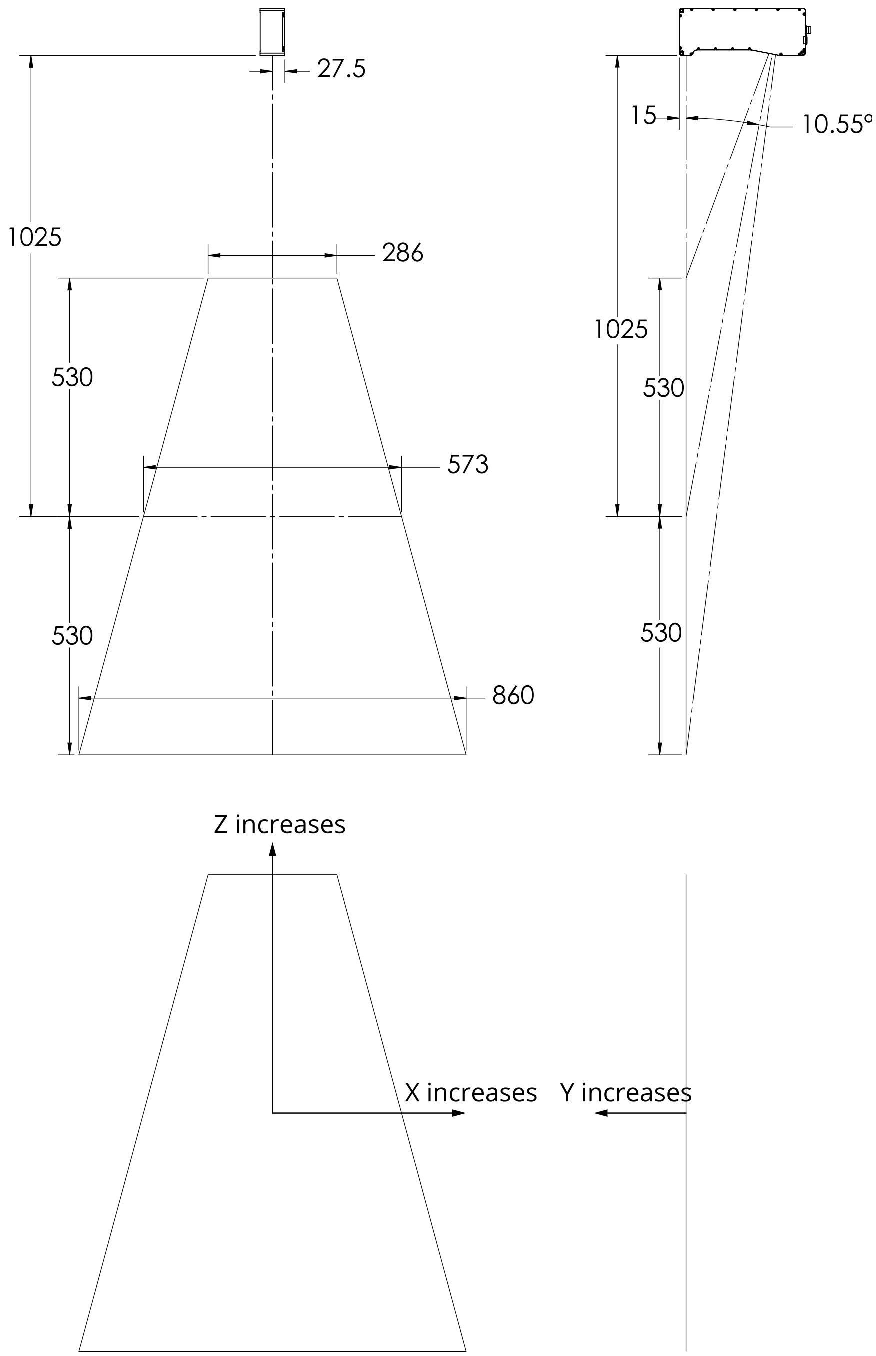

20
240
5X M4 ↧ 4.7
32
16
11.5
120
120

Envelope

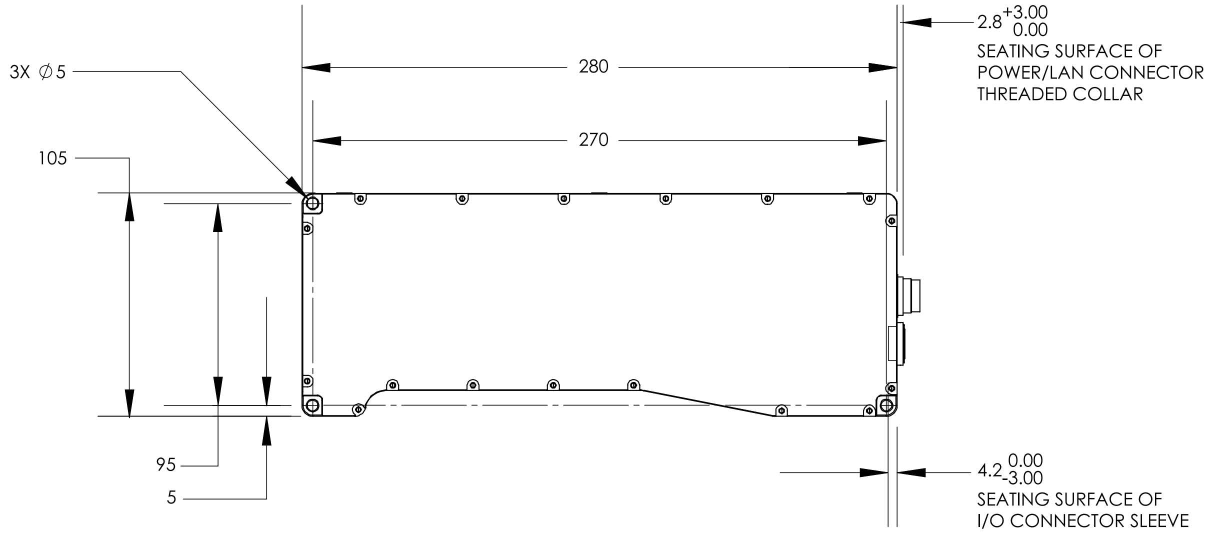


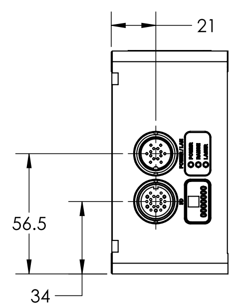
Gocator 2670

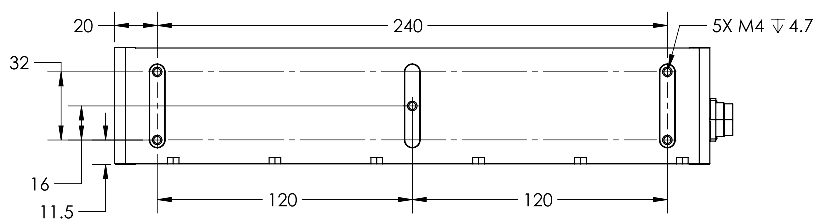
Field of View / Measurement Range / Coordinate System Orientation



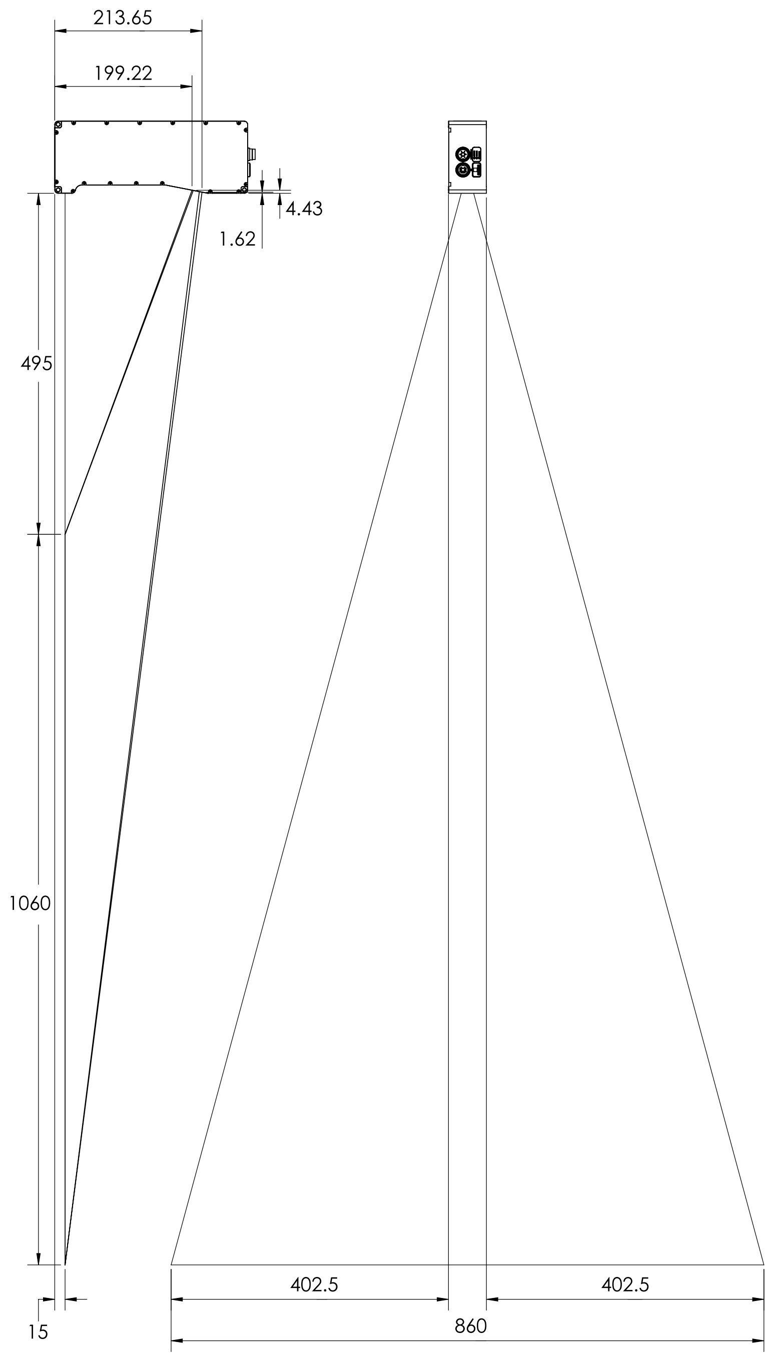
Dimensions


3X ∅5
280
270
2.8 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
105
95
5
4.2 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE


21
56.5
34

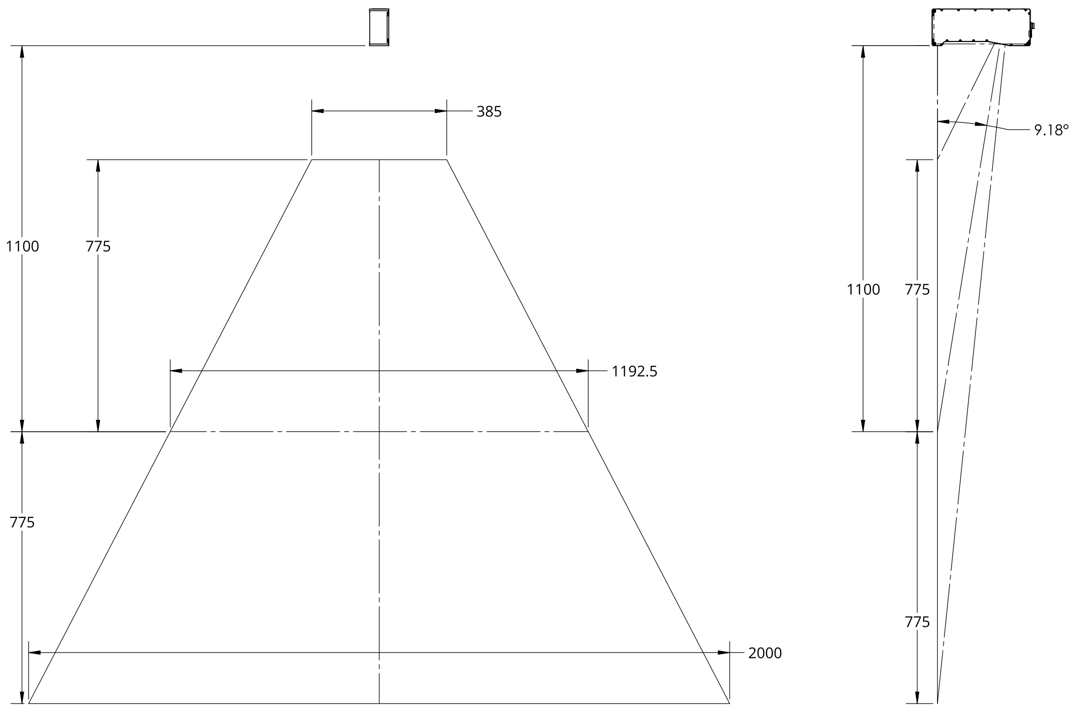

20
240
5X M4 ⏷ 4.7
32
16
11.5
120
120

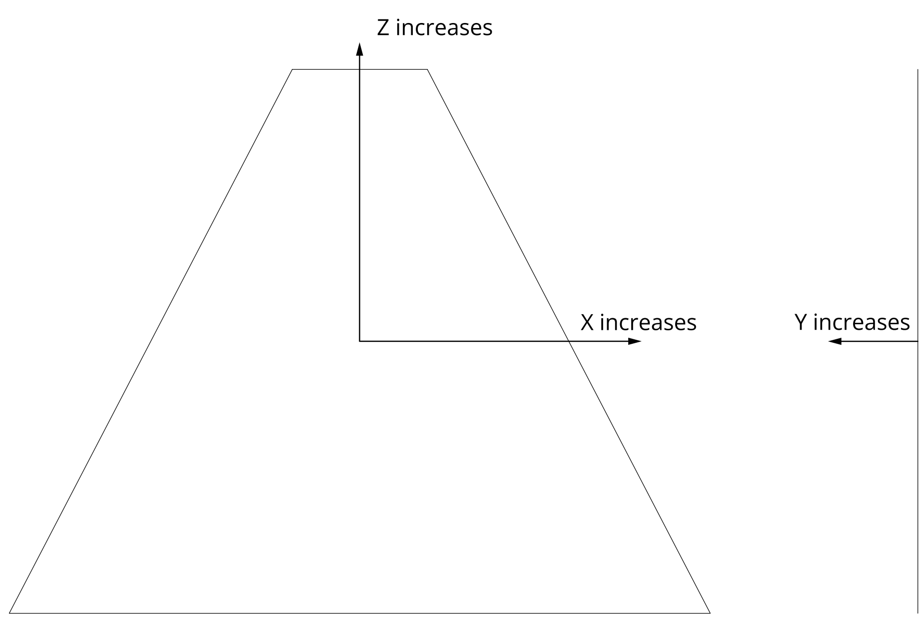
Envelope



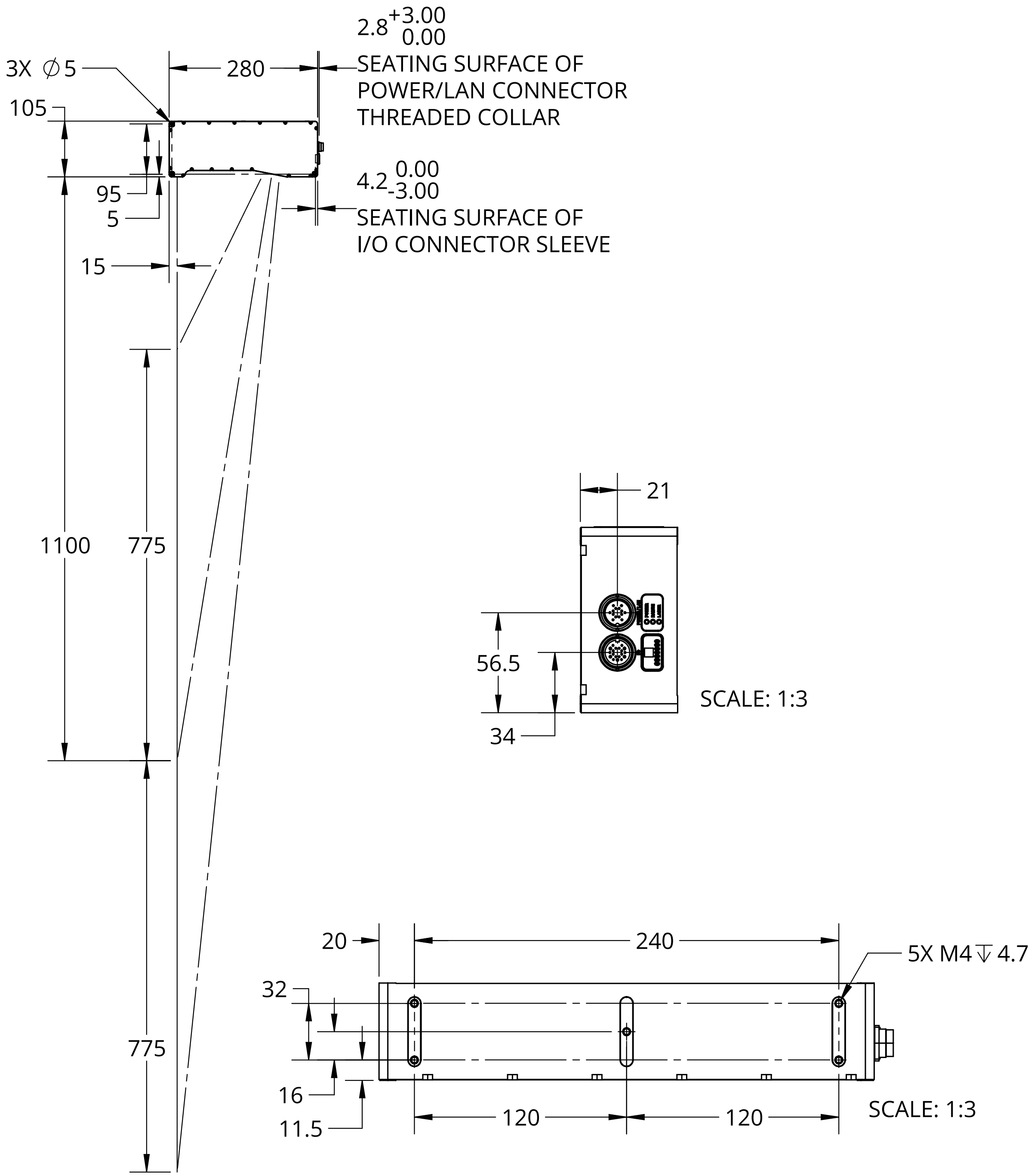
Gocator 2690

Field of View / Measurement Range / Coordinate System Orientation



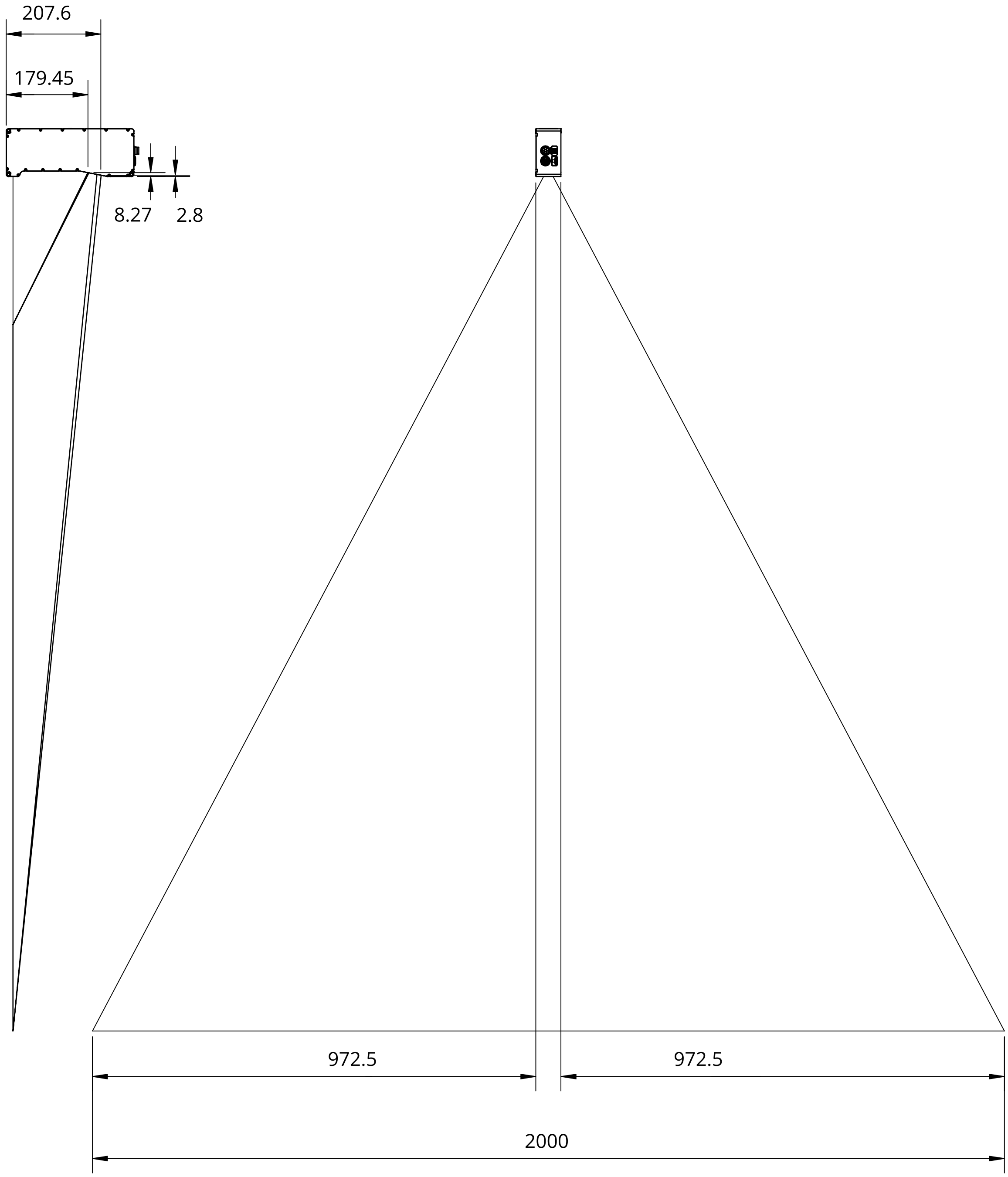


Dimensions



2.8 +3.00 0.00
SEATING SURFACE OF
POWER/LAN CONNECTOR
THREADED COLLAR
3X ∅5
280
105
95
5
4.2 0.00 -3.00
SEATING SURFACE OF
I/O CONNECTOR SLEEVE
15
1100
775
775
21
56.5
34
SCALE: 1:3
20
240
5X M4 ↧ 4.7
32
16
11.5
120
120
SCALE: 1:3

Envelope



Sensor Connectors

The following sections provide the specifications of the connectors on Gocator sensors.

Gocator Power/LAN Connector

The Power/LAN connector is a 14 pin, M16 style connector that provides power input, laser safety input, and Ethernet.

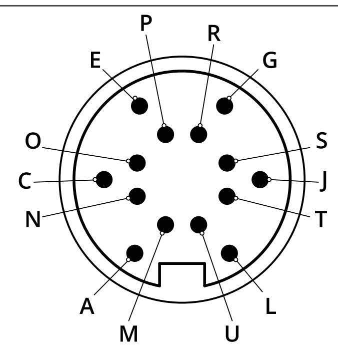
	This connector is rated IP67 only when a cable is connected or when a protective cap is used.

	Some sensors require a minimum input voltage of 48 VDC. Verify the accepted input voltage for your sensor in the sensor's specifications; for specifications, see *Sensors* on page 975.

This section defines the electrical specifications for Power/LAN Connector pins found on Gocator sensors, organized by function.

Gocator Power/LAN Connector Pins

Function	Pin	Lead Color on Standard Cordsets	Lead Color on High Flex Cordsets
GND_24-48V*	L	White/Orange & Black	Orange/Red
GND_24-48V*	L	Orange/Black	Orange/Black
DC_24-48V*	A	White/Green & Black	Green/Red
DC_24-48V*	A	Green/Black	Green/Black
Safety-	G	White/Blue & Black	Blue/Black
Safety+	J	Blue/Black	Blue/Red
Sync+**	E	White/Brown & Black	Brown/Red
Sync-**	C	Brown/Black	Brown/Black
Ethernet MX1+	M	White/Orange	White/Orange
Ethernet MX1-	N	Orange	Orange
Ethernet MX2+	O	White/Green	White/Green
Ethernet MX2-	P	Green	Green
Ethernet MX3-	S	White/Blue	White/Blue
Ethernet MX3+	R	Blue	Blue
Ethernet MX4+	T	White/Brown	White/Brown
Ethernet MX4-	U	Brown	Brown



*View: Looking into the connector **on** the sensor*

*Both wires must be connected to the power supply. The leads are connected to the same connector pin, and both are required to support the maximum possible current draw.

**The Sync leads are not connected in the open wire versions of the Power/LAN cordsets.

Grounding Shield

The grounding shield should be mounted to the earth ground.

Power

Apply positive voltage to DC_24-48V.

It is not necessary to power down a sensor's power source such as a Master before unplugging the sensor from the Master. (Sensors can be "hot-swapped.")

Some sensors require a minimum input voltage of 48 VDC. Verify the accepted input voltage for your sensor in the sensor's specifications; for specifications, see *Sensors* on page 975.

Power requirements

Function	Pins	Min	Max
DC_24-48V	A	24 V (Some models require a minimum of 48 V.)	48 V
GND_24-48VDC	L	0 V	0 V

Laser Safety Input

The Safety_in+ signal should be connected to a voltage source in the range listed below. The Safety_in- signal should be connected to the ground/common of the source supplying the Safety_in+.

You should not use the Laser Safety input to start and stop the laser. Instead, use the digital input for triggering the laser; for more information, see *Digital Input* on page 1061.

Laser safety requirements

Function	Pins	Min	Max
Safety_in+	J	24 V	48 V
Safety_in-	G	0 V	0 V

⚠ Confirm the wiring of Safety_in- before starting the sensor. Wiring DC_24-48V into Safety_in- may damage the sensor.

Gocator I/O Connector

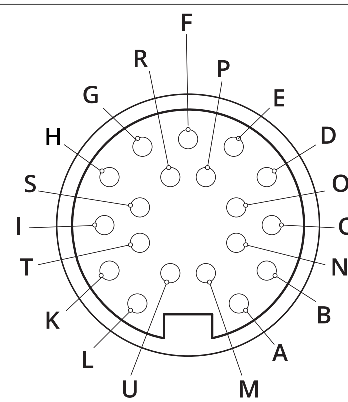
The Gocator I/O connector is a 19 pin, M16 style connector that provides encoder, digital input, digital outputs, serial output, and analog output signals.

This connector is rated IP67 only when a cable is connected or when a protective cap is used.

This section defines the electrical specifications for I/O connector pins, organized by function.

Gocator I/O Connector Pins

Function	Pin	Lead Color on Standard Cordset	Lead Color on High Flex Cordset
Trigger_in+	D	Grey	Blue / Red
Trigger_in-	H	Pink	Blue / Black
Out_1+ (Digital Output 0)	N	Red	Brown / Red
Out_1- (Digital Output 0)	O	Blue	Brown / Black
Out_2+ (Digital Output 1)	S	Tan	Green / Red
Out_2- (Digital Output 1)	T	Orange	Green / Black
Encoder_A+	M	White / Brown & Black	Pink / Red
Encoder_A-	U	Brown / Black	Pink / Black
Encoder_B+	I	Black	Yellow / Red
Encoder_B-	K	Violet	Yellow / Black
Encoder_Z+	A	White / Green & Black	White / Red
Encoder_Z-	L	Green / Black	White / Black
Serial_out+	B	White	Purple / Red
Serial_out-	C	Brown	Purple / Black
	E	Blue / Black	Red
	G	White / Blue & Black	Black
Analog_out+ (Reserved on Gocator 2500 series sensors)	P	Green	Gray / Red
Analog_out- (Reserved on Gocator 2500 series sensors)	F	Yellow & Maroon / White	Gray / Black & Orange / Black
Reserved	R	Maroon (not connected)	Orange / Red (not connected)



*View: Looking into the connector **on** the sensor*

Grounding Shield

The grounding shield should be mounted to the earth ground.

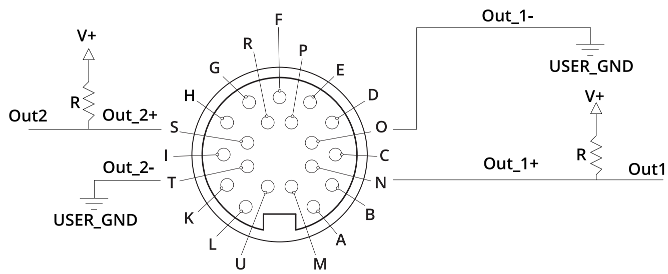
Digital Outputs

Each sensor has two optically isolated outputs. Both outputs are open collector and open emitter, which allows a variety of power sources to be connected and a variety of signal configurations.

> Digital outputs cannot be used when taking scans using the Snapshot button, which takes a single scan and is typically used to test measurement tool settings. Digital outputs can only be used when a sensor is running, taking a continuous series of scans.

Out_1 (Collector – Pin N and Emitter – Pin O) and Out_2 (Collector – Pin S and Emitter – Pin T) are independent and therefore V+ and GND are not required to be the same.

Function	Pins	Max Collector Current	Max Collector–Emitter Voltage	Min Pulse Width
Out_1	N, O	40 mA	70 V	20 µs
Out_2	S, T	40 mA	70 V	20 µs

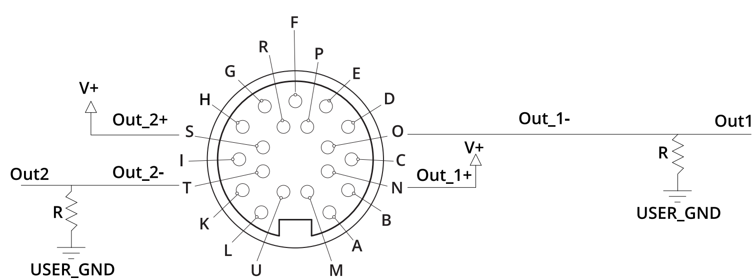


The resistors shown above are calculated by R = (V+) / 2.5 mA.

The size of the resistors is determined by power = (V+)^2 / R.

Inverting Outputs

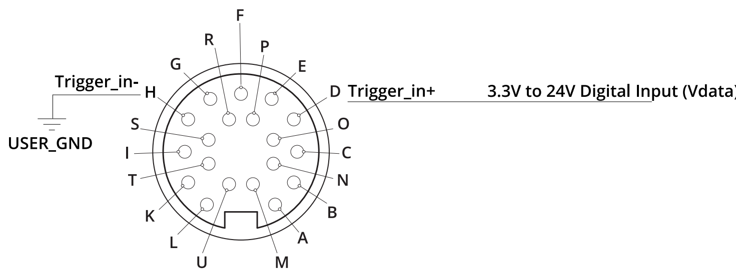
To invert an output, connect a resistor between ground and Out_1- or Out_2- and connect Out_1+ or Out_2+ to the supply voltage. Take the output at Out_1- or Out_2-. For resistor selection, see above.



Digital Input

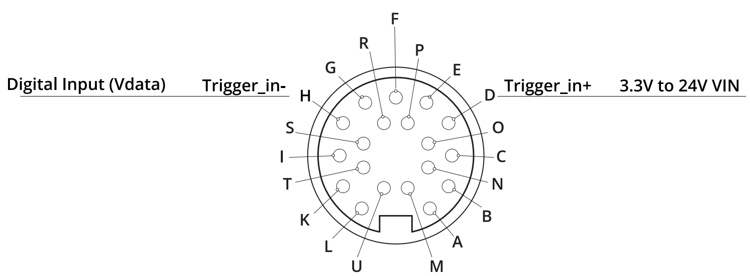
Every sensor has a single optically isolated input. To use this input without an external resistor, supply 3.3 - 24 V to the positive pin and GND to the negative.

	Do not use the Laser Safety input to start and stop the laser. Instead, use the Digital input to do this.



Active High

If the supplied voltage is greater than 24 V, connect an external resistor in series to the positive. The resistor value should be R = [(Vin-1.2V)/10mA]-680.



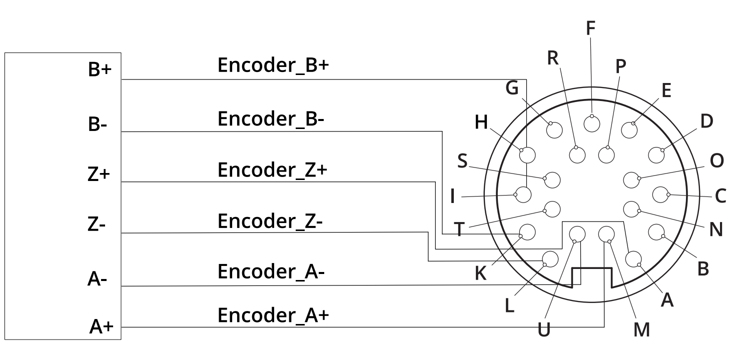
Active Low

To assert the signal, the digital input voltage should be set to draw a current of 3 mA to 40 mA from the positive pin. The current that passes through the positive pin is I = (Vin – 1.2 – Vdata) / 680. To reduce noise sensitivity, we recommend leaving a 20% margin for current variation (i.e., uses a digital input voltage that draws 4mA to 25mA).

Function	Pins	Min Voltage	Max Voltage	Min Current	Max Current	Min Pulse Width
Trigger_in	D, H	3.3 V	24 V	3 mA	40 mA	20 µs

Encoder Input

Encoder input is provided by an external encoder and consists of three RS-485 signals. These signals are connected to Encoder_A, Encoder_B, and Encoder_Z.



Function	Pins	Common Mode Voltage		Differential Threshold Voltage			Max Data Rate
		Min	Max	Min	Typ	Max	
Encoder_A	M, U	-7 V	12 V	-200 mV	-125 mV	-50 mV	1 MHz
Encoder_B	I, K	-7 V	12 V	-200 mV	-125 mV	-50 mV	1 MHz
Encoder_Z	A, L	-7 V	12 V	-200 mV	-125 mV	-50 mV	1 MHz

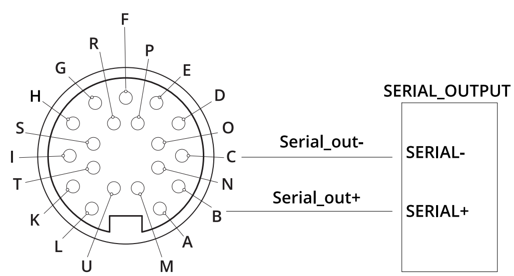
Gocator supports differential RS485 or RS422 signalling, as well as TTL. Both + and - signals must be connected.

Encoders are normally specified in *pulses* per revolution, where each pulse is made up of the four quadrature *signals* (A+ / A- / B+ / B-). Because the sensor reads each of the four quadrature signals, you should choose an encoder accordingly, given the resolution required for your application.

Serial Output

Serial RS-485 output is connected to Serial_out as shown below.

Function	Pins
Serial_out	B, C

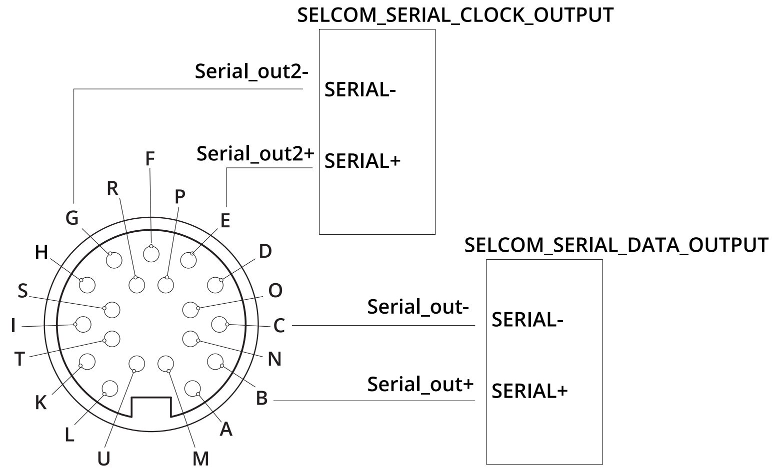


Selcom Serial Output

Serial RS-485 output is connected to Serial_out and Serial_out2 as shown below.

Gocator 2500 series sensors do not support the Selcom serial protocol.

Function	Pins
Serial_out (data)	B, C
Serial_out2 (clock)	E, G



Analog Output

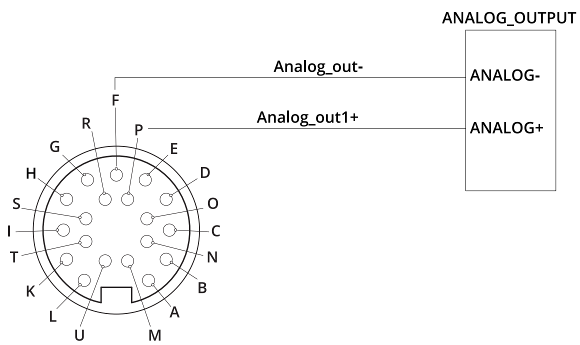
The Sensor I/O Connector defines one analog output interface: Analog_out.

> Gocator 2500 series sensors do not support analog output.

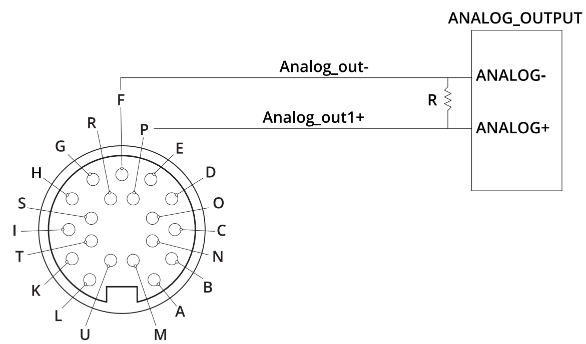
> Gocator 2600 series sensors do not support analog output.

> You do not need to supply an external power source.

Function	Pins	Current Range
Analog_out	P, F	4 – 20 mA

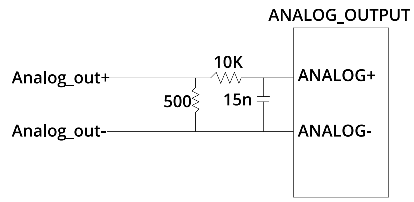


Current Mode



Voltage Mode

To configure for voltage output, connect a 500 Ohm ¼ Watt resistor between Analog_out+ and Analog_ out- and measure the voltage across the resistor. To reduce the noise in the output, we recommend using an RC filter as shown below.


ANALOG_OUTPUT
10K
Analog_out+
ANALOG+
500
15n
Analog_out-
ANALOG-

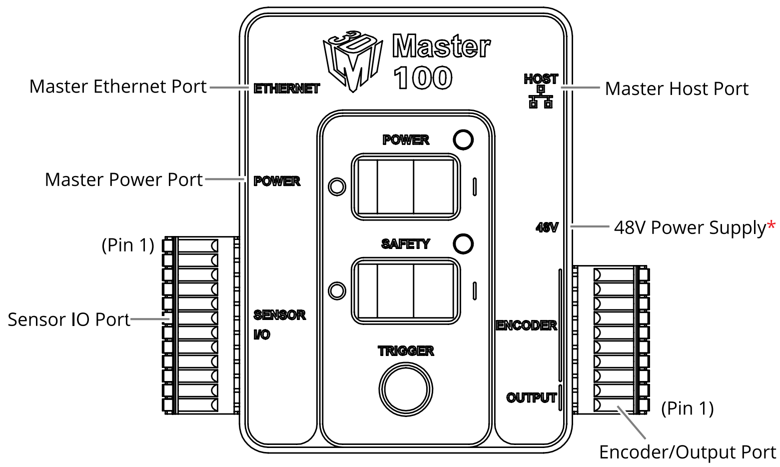
Master Network Controllers

The following sections provide the specifications of Master network controllers.

For information on maximum external input trigger rates, see *Maximum Input Trigger Rate* on page 213.

Master 100

The Master 100 accepts connections for power, safety, and encoder, and provides digital output.



*Contact LMI for information regarding this type of power supply.

Connect the Master Power port to the Gocator's Power/LAN connector using the Gocator Power/LAN to Master cordset. Connect power RJ45 end of the cordset to the Master Power port. The Ethernet RJ45 end of the cordset can be connected directly to the Ethernet switch, or connect to the Master Ethernet port. If the Master Ethernet port is used, connect the Master Host port to the Ethernet switch with a CAT5e Ethernet cable.

To use encoder and digital output, wire the Master's Gocator Sensor I/O port to the Gocator IO connector using the Gocator I/O cordset.

Sensor I/O Port Pins

I/O Pin	Master Pin	Conductor Color
Encoder_A+	1	White/Brown & Black
Encoder_A-	2	Brown/Black
Encoder_Z+	3	White/Green & Black
Encoder_Z-	4	Green/Black
Trigger_in+	5	Grey
Trigger_in-	6	Pink
Out_1-	7	Blue
Out_1+	8	Red

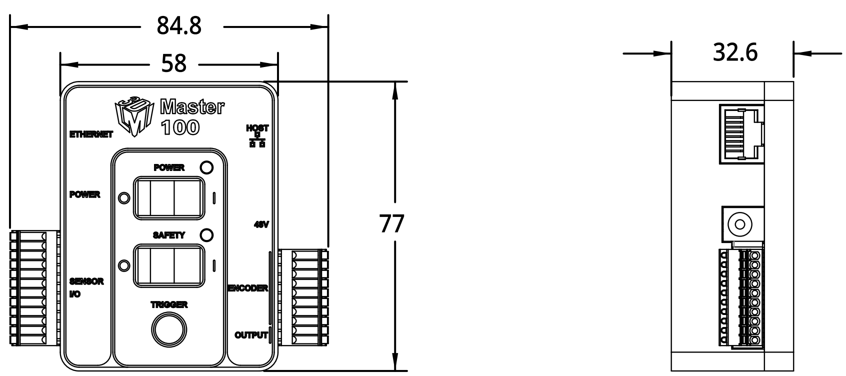
I/O Pin	Master Pin	Conductor Color
Encoder_B+	11	Black
Encoder_B-	12	Violet

The rest of the wires in the I/O cordset are not used.

Encoder/Output Port Pins

Function	Pin
Output_1+ (Digital Output 0)	1
Output_1- (Digital Output 0)	2
Encoder_Z+	3
Encoder_Z-	4
Encoder_A+	5
Encoder_A-	6
Encoder_B+	7
Encoder_B-	8
Encoder_GND	9
Encoder_5V	10

Master 100 Dimensions

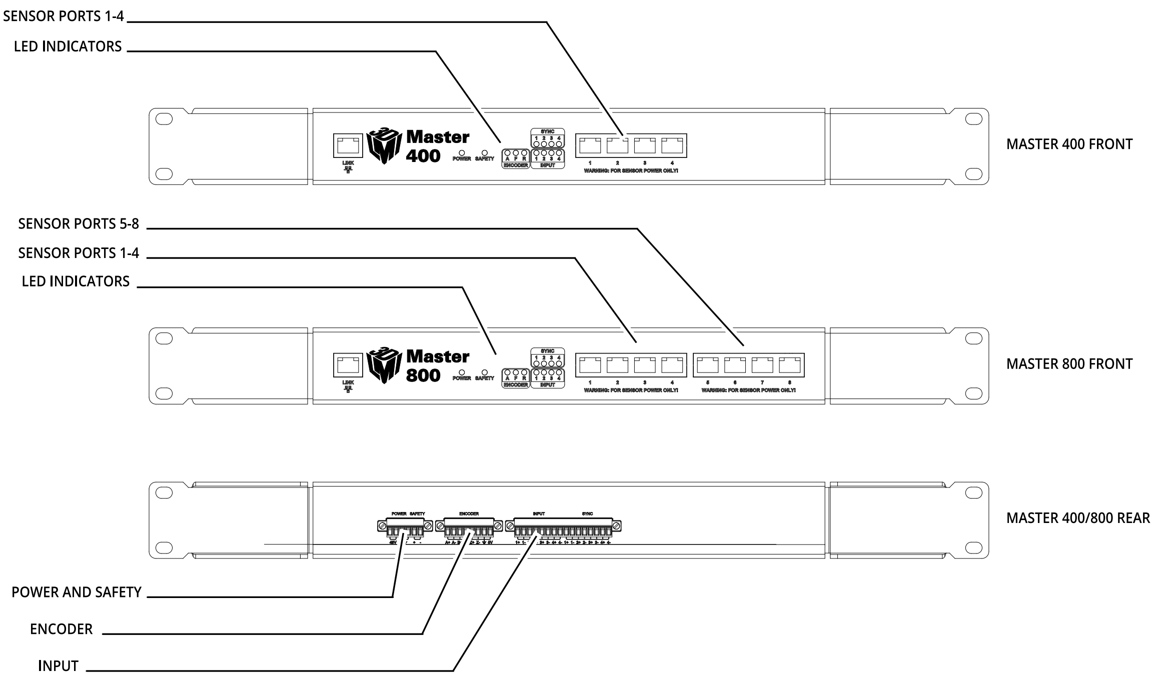


Master 400/800

Master network controllers provide sensor power and laser safety, and broadcast system-wide synchronization information (i.e., time, encoder count, encoder index, and digital I/O states) to all devices on a sensor network.

	It is not necessary to power down a sensor's power source such as a Master before unplugging the sensor from the Master. (Sensors can be "hot-swapped.")

	The Phoenix connectors on Master 400/800/1200/2400 are not compatible with the connectors on Master 810/2410. For this reason, if you are switching models in your network, you must rewire the connections to the Master.



Power and Safety

Power and Safety (6 pin connector)

Function	Pin
+48VDC	1
+48VDC	2
GND (24-48VDC)	3
GND (24-48VDC)	4
Safety Control+	5
Safety Control–	6

The 6-pin Power and Safety connector's specifications are as follows:

- Manufacturer PN: 1847097
- Phoenix Contact MC 1,5/ 6-STF-3,5

> The power supply must be isolated from AC ground. This means that AC ground and DC ground are not connected.

> The Safety Control requires a voltage differential of 24 VDC to 48 VDC across the pin to enable the laser.

Encoder

Encoder (8 pin connector)

Function	Pin
Encoder_A+	1
Encoder_A-	2
Encoder_B+	3
Encoder_B-	4
Encoder_Z+	5
Encoder_Z-	6
GND	7
+5VDC	8

The 8-pin Encoder connector's specifications are as follows:

- Manufacturer PN: 1847181
- Phoenix Contact MC 1,5/ 8-STF-3,5

Input

Input and Sync (16 pin connector)

Function	Pin
Input 1	1
Input 1 GND	2
Reserved	3
Reserved	4
Reserved	5
Reserved	6
Reserved	7
Reserved	8
Reserved	9
Reserved	10

Function	Pin
Reserved	11
Reserved	12
Reserved	13
Reserved	14
Reserved	15
Reserved	16

The 16-pin Input and Sync connector's specifications are as follows:

- Manufacturer PN: 1847262
- Phoenix Contact MC 1,5/ 16-STF-3,5

The Input and Sync connector does not need to be wired up for proper operation.

Master 400/800 Electrical Specifications

Electrical Specifications

Specification	Value
Power Supply Voltage	+48 VDC
Power Supply Current (Max.)	10 A
Power Draw (Min.)	5.76 W
Safety Input Voltage Range	+24 VDC to +48 VDC
Encoder Signal Voltage	Differential (5 VDC)
Digital Input Voltage Range	Logical LOW: 0 to +0.1 VDC Logical HIGH: +3.3 to +24 VDC

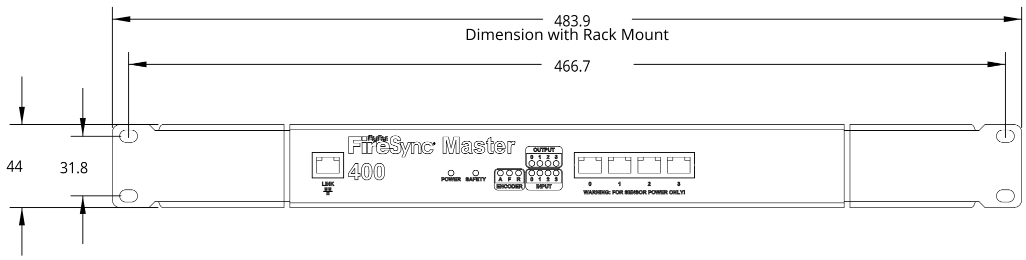
⚠ When using a Master hub, the chassis must be well grounded.

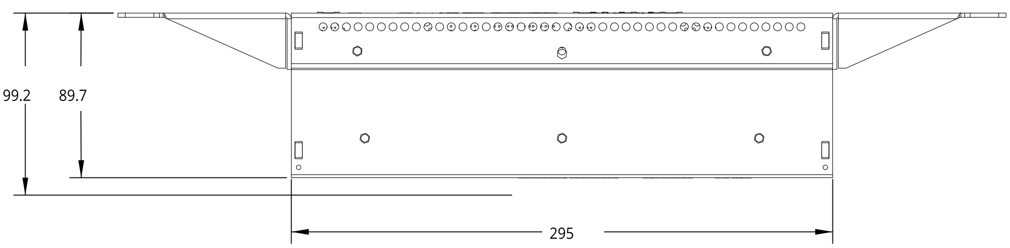
The power supply must be isolated from AC ground. This means that AC ground and DC ground are not connected.

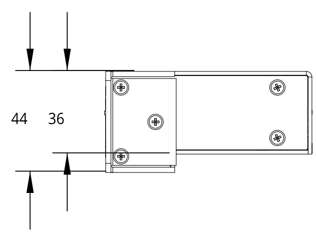
The Power Draw specification is based on a Master with no sensors attached. Every sensor has its own power requirements that need to be considered when calculating total system power requirements..

Master 400/800 Dimensions

The dimensions of Master 400 and Master 800 are the same.







Master 810/2410

Master network controllers provide sensor power and laser safety, and broadcast system-wide synchronization information (i.e., time, encoder count, encoder index, and digital I/O states) to all devices on a sensor network.

> It is not necessary to power down a sensor's power source such as a Master before unplugging the sensor from the Master. (Sensors can be "hot-swapped.")

> Some sensors require a minimum input voltage of 48 VDC. Verify the accepted input voltage for your sensor in the sensor's specifications; for specifications, see *Sensors* on page 975.

The following table summarizes Master 810 and 2410:

Master 810 and 2410	
Input Voltage (Power)	+24-48 VDC (2 Watts)[1]
Total Power	Master 810 / 2410 input power + (sensor input power x number of sensors)
Capacity	Master 810: up to 8 sensors Master 2410: up to 24 sensors
I/O	4 digital inputs[2] Single-Ended Active LOW: 0 to +0.8 VDC Single-Ended Active HIGH: +3.3 to +24 VDC Differential LOW: 0.8 to -24 VDC Differential HIGH: +3.3 to +24 VDC 10-pin Phoenix For more information, see *Electrical Specifications* on page 1076.
Encoder	Differential (5 VDC, 12 VDC) Single-ended (5 VDC, 12 VDC)[3] For more information, see *Electrical Specifications* on page 1076.
LED Indicators	Safety, power, encoder, input. For more information, see *LED Indicators* on the next page.
Cable	Dual CAT5e cable for power / safety / synchronization / data
Weight (kg)	Master 810: 0.6 Master 2410: 0.9
Storage Temperature	-30 to 70 °C
Operation Temperature	0 to 50 °C

Notes

1. Refer to sensor datasheets for additional power required by sensors.
2. Gocator only supports one digital input.
3. Supports open collector, pull-up resistor, line driver, push-pull, and TTL.

The following table describes the meanings of the encoder and sensor port LED indicators:

LED Indicators

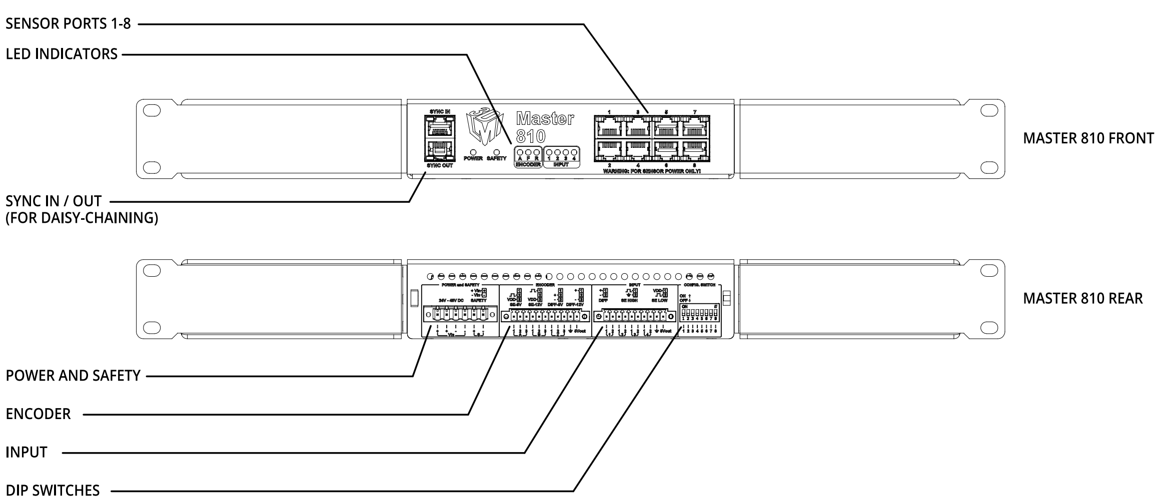
Indicator	Description
Power	Device is on.
Safety	Indicates the status of the Laser Safety circuitry. The “On” state indicates that all sensor light sources are active.
Encoder A	Reserved
Encoder F	**On continuously**: Forward motion with no indexing is detected. **Blinking**: Forward motion with indexing is detected.
Encoder R	**On continuously**: Reverse motion with no indexing is detected. **Blinking**: Reverse motion with indexing is detected.
Input 1-4	Digital input ports 1-4 active.
SYNC IN and SYNC OUT Ports (Green and Orange LEDs)	Reserved.
Sensor Port Green LED	Indicates that a sensor is connected to the port and is powered up.
Sensor Port Orange LED	Not used.

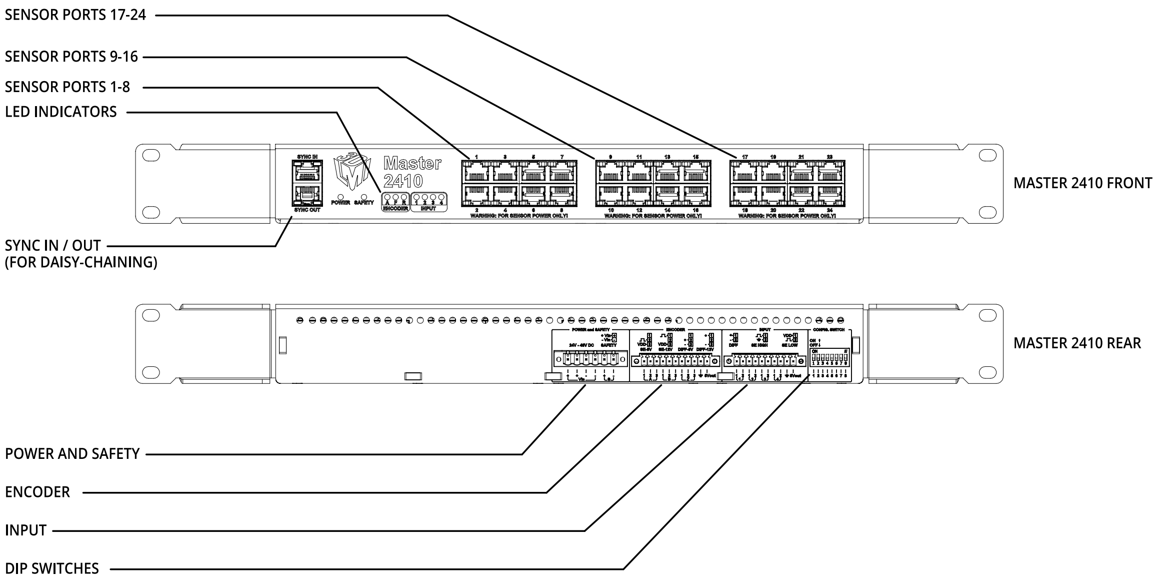
Master 810 and 2410 can be mounted to DIN rails using the provided adapters (for more information, see *Installing DIN Rail Clips: Master 810 or 2410* on page 45). The units are also provided with removable adapters for 1U rack mounting; the mounting holes for this option are compatible with older Master models (400/800/1200/2400).

> The Phoenix connectors on Master 400/800/1200/2400 are not compatible with the connectors on Master 810/2410. For this reason, if you are switching models in your network, you must rewire the connections to the Master.

Master 2410 can currently be used with encoders with a maximum quadrature frequency of 300 kHz.

Master 810 can be configured to work with a maximum encoder quadrature frequency of 6.5 MHz. For more information, see *Configuring Master 810* on page 47.





Power and Safety

Power and Safety (6 pin connector)

Function	Pin
Power In+	1
Power In+	2
Power In-	3
Power In-	4
Safety Control+	5
Safety Control–	6

The 6-pin Power and Safety connector's specifications are as follows:

- CONNECTOR, 6 Position Terminal Block Plug, Female Sockets 0.200" (5.08mm) 180° Free Hanging (In-Line)
- Manufacturer PN: 1912223
- Supplier Part Number 277-11017-ND
- Manufacturer: Phoenix Contact

> The power supply must be isolated from AC ground. This means that AC ground and DC ground are not connected.

> The Safety Control requires a voltage differential of 24 VDC to 48 VDC across the pin to enable the laser.

Encoder

Encoder (11 pin connector)

Function	Pin
Encoder_A_Pin_1	1
Encoder_A_Pin_2	2
Encoder_A_Pin_3	3
Encoder_B_Pin_1	4
Encoder_B_Pin_2	5
Encoder_B_Pin_3	6
Encoder_Z_Pin_1	7
Encoder_Z_Pin_2	8
Encoder_Z_Pin_3	9
GND (output for powering external devices)	10
+5VDC (output for powering external devices)	11

For Encoder connection wiring options, see *Encoder* on page 1077.

The 11-pin Encoder connector's specifications are as follows:

- CONNECTOR, 11 Position Terminal Block Plug, Female Sockets 0.138" (3.50mm) 180° Free Hanging (In-Line)
- Manufacturer PN: 1847217
- Supplier Part Number 277-8897-ND
- Manufacturer: Phoenix Contact

Input

On earlier revisions of Master 810 and Master 2410, the inputs are labeled 0-3.

Input (10 pin connector)

Function	Pin
Input 1 Pin 1	1
Input 1 Pin 2	2
Reserved	3
Reserved	4
Reserved	5
Reserved	6
Reserved	7
Reserved	8
GND (output for powering other devices)	9
+5VDC (output for powering other devices)	10

For Input connection wiring options, see *Input* on page 1079.

The following are the 10-pin Input connector's specifications:

- CONNECTOR, 10 Position Terminal Block Plug, Female Sockets 0.138" (3.50mm) 180° Free Hanging (In-Line)
- Manufacturer PN: 1847204
- Supplier Part Number 277-6350-ND
- Manufacturer: Phoenix Contact

The Input connector does not need to be wired up for proper operation.

Electrical Specifications

Some sensors require a minimum input voltage of 48 VDC. Verify the accepted input voltage for your sensor in the sensor's specifications; for specifications, see*Sensors* on page 975

Electrical Specifications

Specification	Value
Power Supply Voltage	+24 VDC to +48 VDC
Power Supply Current (Max.)*	Master 810: 9 A Master 2410: 25 A * Fully loaded with 1 A per sensor port.
Power Draw (Min.)	Master 810: 1.7 W Master 2410: 4.8 W
Safety Input Voltage Range	+24 VDC to +48 VDC
Encoder Signal Voltage	Single-Ended Active LOW: 0 to +0.8 VDC Single-Ended Active HIGH: +3.3 to +24 VDC Differential LOW: 0.8 to -24 VDC Differential HIGH: +3.3 to +24 VDC For more information, see *Encoder* on the next page.
Digital Input Voltage Range	Single-Ended Active LOW: 0 to +0.8 VDC Single-Ended Active HIGH: +3.3 to +24 VDC Differential LOW: 0.8 to -24 VDC Differential HIGH: +3.3 to +24 VDC For more information, see *Input* on page 1079. If the input voltage is above 24 V, use an external resistor, using the following formula: R = [(Vin - 1.2V) / 10mA] - 680

When using a Master hub, the chassis must be well grounded.

🗋	The power supply must be isolated from AC ground. This means that AC ground and DC ground are not connected.

🗋	24 VDC power supply is only supported if all connected sensors support an input voltage of 24 VDC.

🗋	The Power Draw specification is based on a Master with no sensors attached. Every sensor has its own power requirements that need to be considered when calculating total system power requirements..

Encoder

Master 810 and 2410 support the following types of encoder signals: Single-Ended (5 to 12 VDC, 12 to 24 VDC) and Differential (5 to 12 VDC, 12 to 24 VDC).

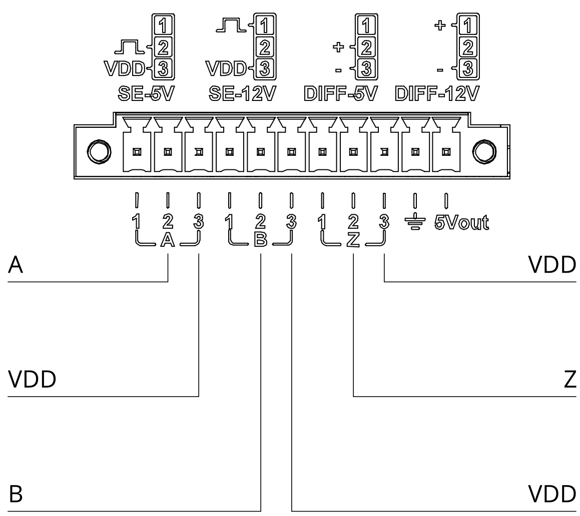
For 5 to 12 VDC operation, pins 2 and 3 of each channel are used.

🗋	The 5-volt encoder input supports up to 12 volts for compatibility with earlier Master network controllers. However, we strongly recommend connecting 12-volt output encoders to the appropriate 12-volt input to attain maximum tolerance.

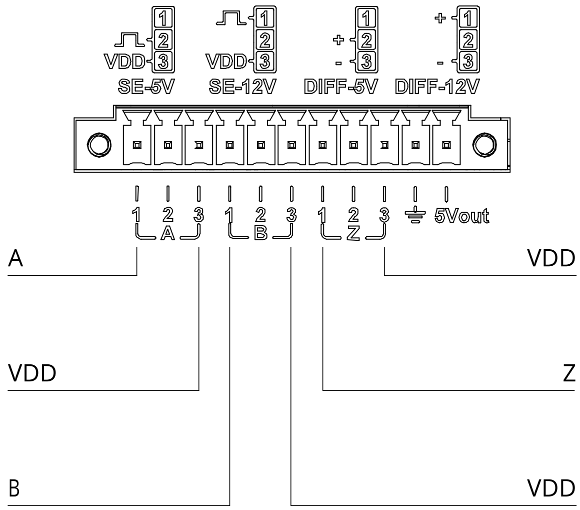
For 12 to 24 VDC operation, pins 1 and 3 of each channel are used.

🗋	Although Master 810 and 2410 are currently labeled as only supporting up to 12 VDC, they can support up to 24 VDC.

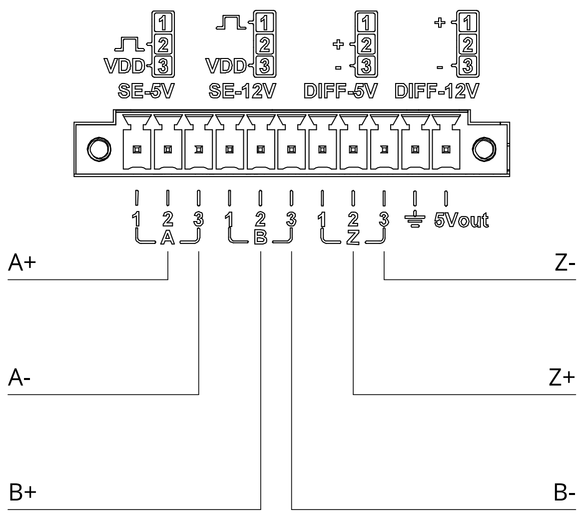
To determine how to wire a Master to an encoder, see the illustrations below.



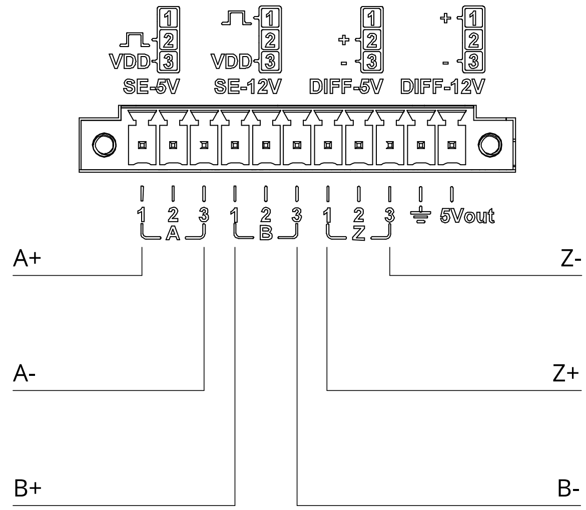
Single-Ended 5 VDC



Single-Ended 12 VDC



Differential 5 VDC



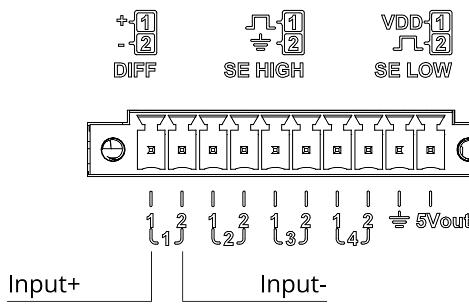
Differential 12 VDC

Input

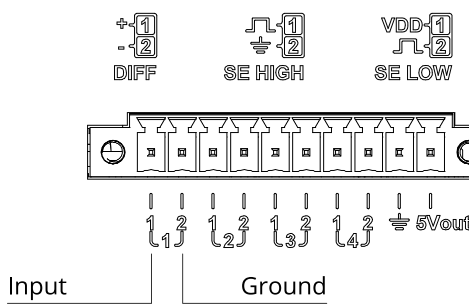
Master 810 and 2410 support the following types of input: Differential, Single-Ended High, and Single-Ended Low.

Currently, Gocator only supports Input 0.

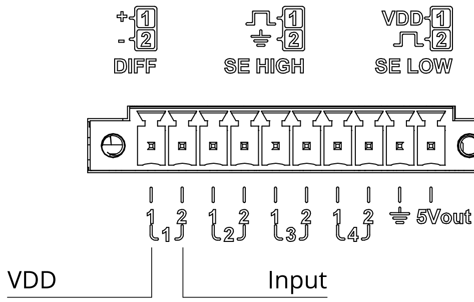
For digital input voltage ranges, see the table below.



Differential



Single-Ended Active High



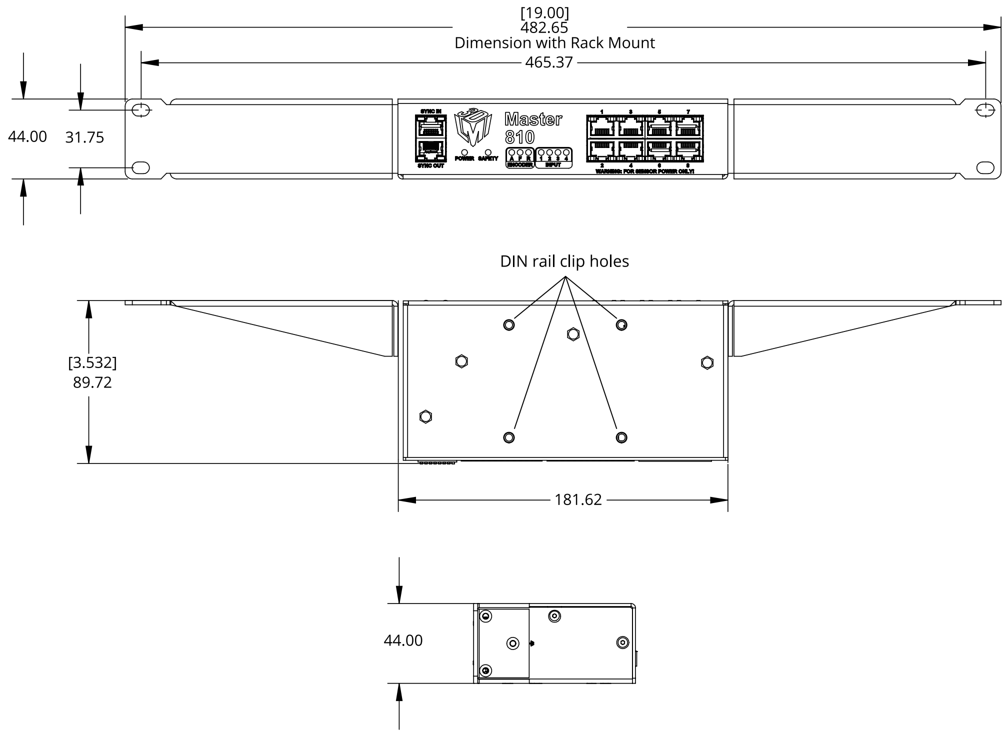
Single-Ended Active Low

Digital Input Voltage Ranges

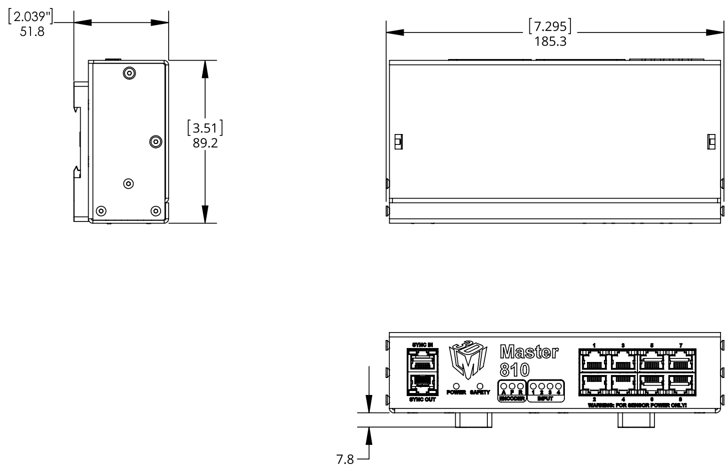
	Input Status	Min (VDC)	Max (VDC)
Single-ended Active High	Off	0	+0.8
	On	+3.3	+24
Single-ended Active Low	Off	(V_{DD} - 0.8)	V_{DD}
	On	0	(V_{DD} - 3.3)
Differential	Off	-24	+0.8
	On	+3.3	+24

Master 810 Dimensions

With 1U rack mount brackets:



With DIN rail mount clips:



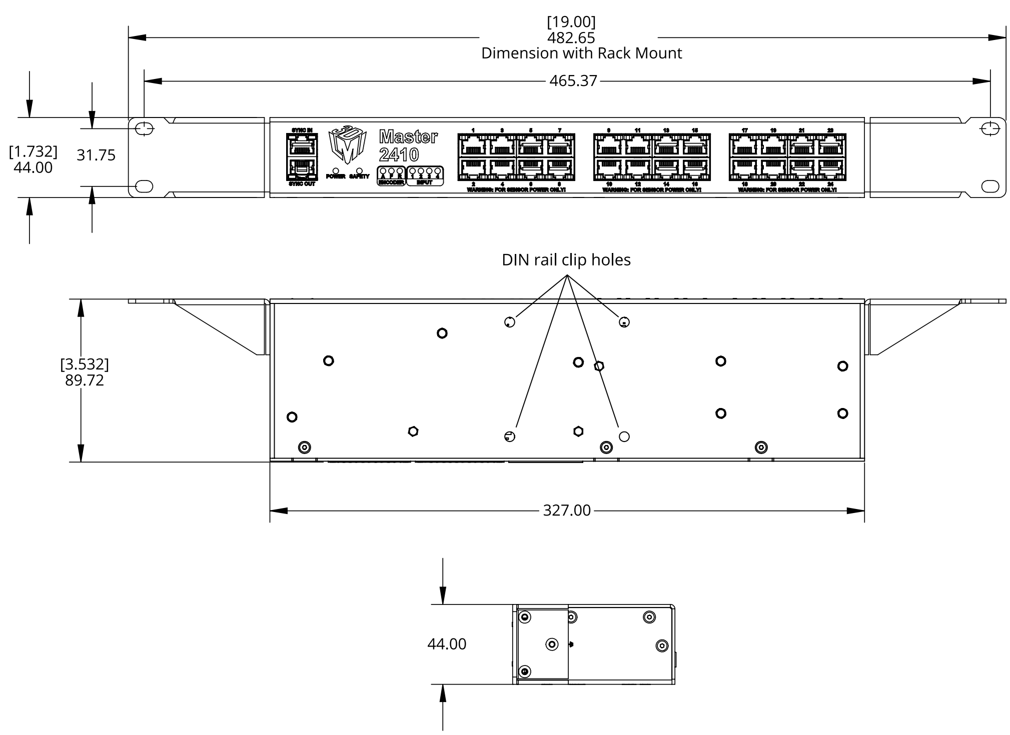
Older revisions of Master 810 and 2410 network controllers use a different configuration for the DIN rail clip holes.

For information on installing DIN rail clips, see *Installing DIN Rail Clips: Master 810 or 2410* on page 45.

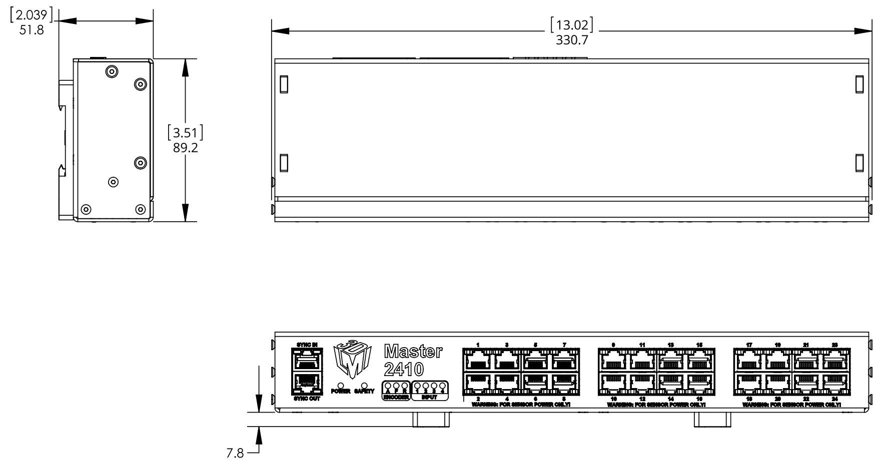
The CAD model of the DIN rail clip is available at https://www.winford.com/products/cad/dinm12-rc.igs.

Master 2410 Dimensions

With 1U rack mount brackets:



With DIN rail mount clips:



Older revisions of Master 810 and 2410 network controllers use a different configuration for the DIN rail clip holes.

For information on installing DIN rail clips, see *Installing DIN Rail Clips: Master 810 or 2410* on page 45.

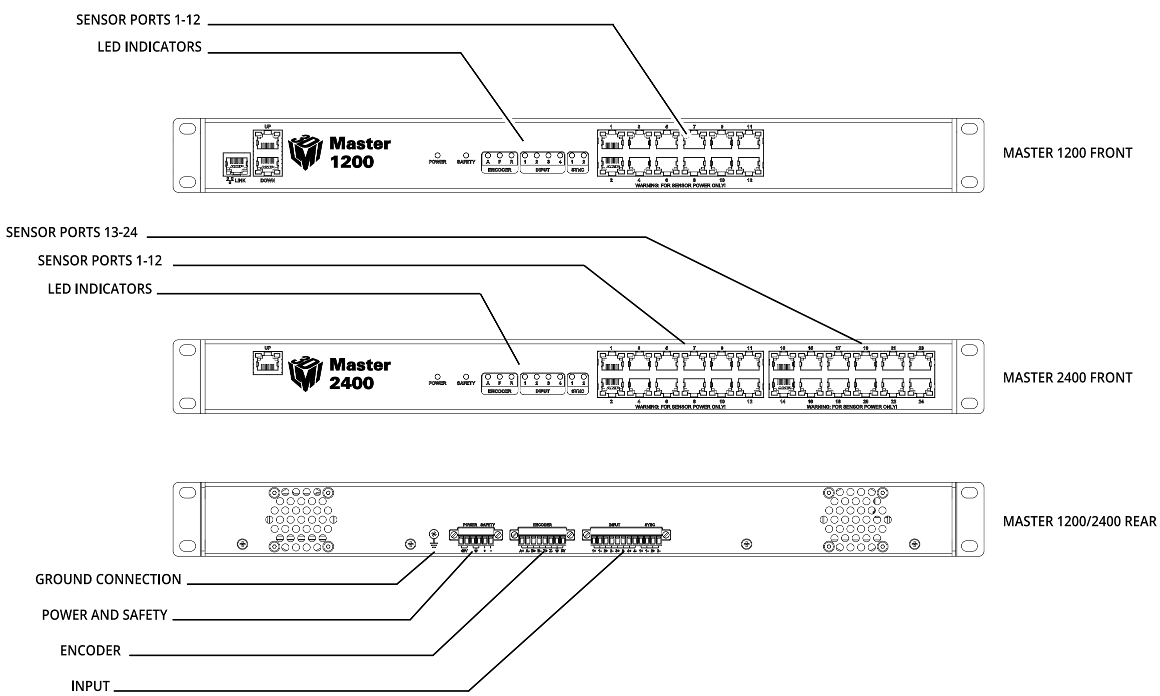
The CAD model of the DIN rail clip is available at https://www.winford.com/products/cad/dinm12-rc.igs.

Master 1200/2400

Master network controllers provide sensor power and laser safety, and broadcast system-wide synchronization information (i.e., time, encoder count, encoder index, and digital I/O states) to all devices on a sensor network.

🗋	It is not necessary to power down a sensor's power source such as a Master before unplugging the sensor from the Master. (Sensors can be "hot-swapped.")

🗋	The Phoenix connectors on Master 400/800/1200/2400 are not compatible with the connectors on Master 810/2410. For this reason, if you are switching models in your network, you must rewire the connections to the Master.



Power and Safety

Power and Safety (6 pin connector)

Function	Pin
+48VDC	1
+48VDC	2
GND (24-48VDC)	3
GND (24-48VDC)	4
Safety Control+	5
Safety Control–	6

The 6-pin Power and Safety connector's specifications are as follows:

- Manufacturer PN: 1847097
- Phoenix Contact MC 1,5/ 6-STF-3,5

> The power supply must be isolated from AC ground. This means that AC ground and DC ground are not connected.

> The Safety Control requires a voltage differential of 24 VDC to 48 VDC across the pin to enable the laser.

Encoder

Encoder (8 pin connector)

Function	Pin
Encoder_A+	1
Encoder_A-	2
Encoder_B+	3
Encoder_B-	4
Encoder_Z+	5
Encoder_Z-	6
GND	7
+5VDC	8

The 8-pin Encoder connector's specifications are as follows:

- Manufacturer PN: 1847181
- Phoenix Contact MC 1,5/ 8-STF-3,5

Input

Input and Sync (12 pin connector)

Function	Pin
Input 1	1
Input 1 GND	2
Reserved	3
Reserved	4
Reserved	5
Reserved	6
Reserved	7
Reserved	8
Reserved	9
Reserved	10

Function	Pin
Reserved	11
Reserved	12

The 12-pin Input and Sync connector's specifications are as follows:

- Manufacturer PN: 1847220
- Phoenix Contact MC 1,5/ 12-STF-3,5

The Input and Sync connector does not need to be wired up for proper operation.

Master 1200/2400 Electrical Specifications

Electrical Specifications

Specification	Value
Power Supply Voltage	+48 VDC
Power Supply Current (Max.)	10 A
Power Draw (Min.)	5.76 W
Safety Input Voltage Range	+24 VDC to +48 VDC
Encoder Signal Voltage	Differential (5 VDC)
Digital Input Voltage Range	Logical LOW: 0 to +0.1 VDC Logical HIGH: +3.5 to +6.5 VDC

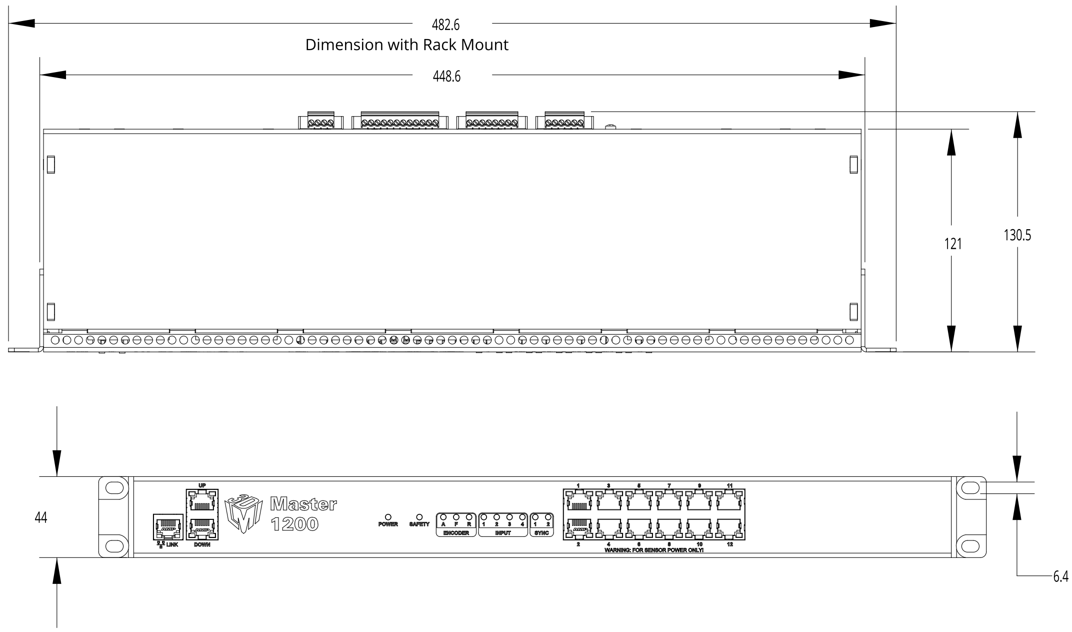
When using a Master hub, the chassis must be well grounded.

The power supply must be isolated from AC ground. This means that AC ground and DC ground are not connected.

The Power Draw specification is based on a Master with no sensors attached. Every sensor has its own power requirements that need to be considered when calculating total system power requirements..

Master 1200/2400 Dimensions

The dimensions of Master 1200 and Master 2400 are the same.


482.6
Dimension with Rack Mount
448.6
121
130.5
44
6.4
Master
1200

Accessories

Masters

Description	Part Number
Master 100 - for single sensor (development only)	30705
Master 810 - for networking up to 8 sensors	301114
Master 2410 - for networking up to 24 sensors	301115

High Flex Gocator Cordsets - Straight Connectors

Description	Part Number
1.2m I/O cordset, open wire end	301175-1.2m
2m I/O cordset, open wire end	301175-2m
5m I/O cordset, open wire end	301175-5m
10m I/O cordset, open wire end	301175-10m
15m I/O cordset, open wire end	301175-15m
20m I/O cordset, open wire end	301175-20m
25m I/O cordset, open wire end	301175-25m
Custom length (< 25m) I/O cordset, open wire end	301175
2m Power and Ethernet cordset, 1x open wire end, 1x RJ45 end	301176-2m
5m Power and Ethernet cordset, 1x open wire end, 1x RJ45 end	301176-5m
10m Power and Ethernet cordset, 1x open wire end, 1x RJ45 end	301176-10m
15m Power and Ethernet cordset, 1x open wire end, 1x RJ45 end	301176-15m
20m Power and Ethernet cordset, 1x open wire end, 1x RJ45 end	301176-20m
25m Power and Ethernet cordset, 1x open wire end, 1x RJ45 end	301176-25m
Custom length (< 25m) Power and Ethernet cordset, 1x open wire end, 1x RJ45 end	301176
1.2m Power and Ethernet to Master cordset, 2x RJ45 ends	301165-1.2m
2m Power and Ethernet to Master cordset, 2x RJ45 ends	301165-2m
5m Power and Ethernet to Master cordset, 2x RJ45 ends	301165-5m
10m Power and Ethernet to Master cordset, 2x RJ45 ends	301165-10m
15m Power and Ethernet to Master cordset, 2x RJ45 ends	301165-15m
20m Power and Ethernet to Master cordset, 2x RJ45 ends	301165-20m
25m Power and Ethernet to Master cordset, 2x RJ45 ends	301165-25m
Custom length (< 25m) Power and Ethernet to Master cordset, 2x RJ45 ends	301165

High Flex Gocator Cordsets - 90-degree Connectors

Description	Part Number
2m I/O cordset, 90-deg, open wire end	301172-2m
5m I/O cordset, 90-deg, open wire end	301172-5m
10m I/O cordset, 90-deg, open wire end	301172-10m
15m I/O cordset, 90-deg, open wire end	301172-15m
20m I/O cordset, 90-deg, open wire end	301172-20m
25m I/O cordset, 90-deg, open wire end	301172-25m
Custom length (< 25m) I/O cordset, 90-deg, open wire end	301172
2m Power and Ethernet cordset, 90-deg, 1x open wire end, 1x RJ45 end	301171-2m
5m Power and Ethernet cordset, 90-deg, 1x open wire end, 1x RJ45 end	301171-5m
10m Power and Ethernet cordset, 90-deg, 1x open wire end, 1x RJ45 end	301171-10m
15m Power and Ethernet cordset, 90-deg, 1x open wire end, 1x RJ45 end	301171-15m
20m Power and Ethernet cordset, 90-deg, 1x open wire end, 1x RJ45 end	301171-20m
25m Power and Ethernet cordset, 90-deg, 1x open wire end, 1x RJ45 end	301171-25m
Custom length (<25m) Power and Ethernet cordset, 90-deg, 1x open wire end, 1x RJ45 end	301171
2m Power and Ethernet to Master cordset, 90-deg, 2x RJ45 ends	301173-2m
5m Power and Ethernet to Master cordset, 90-deg, 2x RJ45 ends	301173-5m
10m Power and Ethernet to Master cordset, 90-deg, 2x RJ45 ends	301173-10m
15m Power and Ethernet to Master cordset, 90-deg, 2x RJ45 ends	301173-15m
20m Power and Ethernet to Master cordset, 90-deg, 2x RJ45 ends	301173-20m
25m Power and Ethernet to Master cordset, 90-deg, 2x RJ45 ends	301173-25m

Notes related to cordsets

For information on cordset bend radius limits, see *Cordset Bend Radius Limits* on page 41.

Custom cordset lengths between 25 m and 60 m (maximum) and connector orientations are available upon request. Prices depend on length and orientation requested. Extension cordsets with one male and one female Gocator connector are also available on request.

90-degree connectors can be adjusted one of four different orientations by opening the connector.

Calibration Disks

Description	Part Number
Calibration Disk, 40mm	30727
Calibration Disk, 100mm	30728

Troubleshooting

Review the guidance in this chapter if you are experiencing difficulty with a sensor system.

If the problem that you are experiencing is not described in this section, see *Return Policy* on page 1091.

Mechanical/Environmental
The sensor is warm.

- It is normal for a sensor to be warm when powered on. A sensor is typically 15° C warmer than the ambient temperature.

Connection
When attempting to connect to the sensor with a web browser, the sensor is not found (page does not load).

- Verify that the sensor is powered on and connected to the client computer network. The Power Indicator LED should illuminate when the sensor is powered.
- Check that the client computer's network settings are properly configured.
- Use the Sensor Recovery tool to verify that the sensor has the correct network settings. See *GoPxL Discovery Tool* on page 935 for more information.

When attempting to log in, the password is not accepted.

- Use the Sensor Recovery tool. See *GoPxL Discovery Tool* on page 935 for steps to reset the password.

Data Acquisition
The sensor emits laser light, but the Range Indicator LED does not illuminate and/or points are not displayed in the Data Viewer.

- Verify that the measurement target is within the sensor's field of view and measurement range. See *Specifications* on page 973 to review the measurement specifications for your sensor model.
- Check that the exposure time is set to a reasonable level. See *Exposure* on page 219 for more information on configuring exposure time.

Performance
The sensor CPU level is near 100%.

- Consider reducing the speed. If you are using a time or encoder trigger source, see *Triggers* on page 206 for information on reducing the speed. If you are using an external input or software trigger, consider reducing the rate at which you apply triggers.
- Consider reducing the resolution.
  For more information on configuring resolution, see *Uniform Spacing* on page 197 and *Triggers* on page 206.
- Review the measurements that you have programmed and eliminate any unnecessary measurements.

Return Policy

Return Policy

Before returning the product for repair (warranty or non-warranty) a Return Material Authorization (RMA) number must be obtained from LMI. Please call LMI to obtain this RMA number.

Carefully package the sensor in its original shipping materials (or equivalent) and ship the sensor prepaid to your designated LMI location. Please ensure that the RMA number is clearly written on the outside of the package. Inside the return shipment, include the address you wish the shipment returned to, the name, email and telephone number of a technical contact (should we need to discuss this repair), and details of the nature of the malfunction. For non-warranty repairs, a purchase order for the repair charges must accompany the returning sensor.

LMI Technologies Inc. is not responsible for damages to a sensor that are the result of improper packaging or damage during transit by the courier.

Software Licenses

CLI11

Website:

https://github.com/CLIUtils/CLI11

License:

xxhash

Website:

https://github.com/Cyan4973/xxHash

License:

xxHash Library

Copyright (c) 2012-present, Yann Collet

All rights reserved.

BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

----

xxhsum command line interface

Copyright (c) 2013-present, Yann Collet

All rights reserved.

GPL v2 License

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

JSON for C++

Website:

https://github.com/nlohmann/json

License:

OpENer

Website:

https://github.com/EIPStackGroup/OpENer

License:

Neither the name of Rockwell Automation, ODVA, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission from the respective owners.

The Communications Stack Software for EtherNet/IP, or any portion thereof, with or without modifications, may be incorporated into products for sale. However, the software does not, by itself, convey any right to make, have made, use, import, offer to sell, sell, lease, market, or otherwise distribute or dispose of any products that implement this software, which products might be covered by valid patents or copyrights of ODVA, Inc., its members or other licensors nor does this software result in any license to use the EtherNet/IP mark owned by ODVA. To make, have made, use, import, offer to sell, sell, lease, market, or otherwise distribute or dispose of any products that implement this software, and to use the EtherNet/IP mark, one must obtain the necessary license from ODVA through its Terms of Usage Agreement for the EtherNet/IP technology, available through the ODVA web site at www.odva.org. This license requirement applies equally (a) to devices that completely implement ODVA's Final Specification for EtherNet/IP (“Network Devices”), (b) to components of such Network Devices to the extent they implement portions of the Final Specification for EtherNet/IP, and (c) to enabling technology products, such as any other EtherNet/IP or other network protocol stack designed for use in Network Devices to the extent they implement portions of the Final Specification for EtherNet/IP. Persons or entities who are not already licensed for the EtherNet/IP technology must contact ODVA for a Terms of Usage Agreement.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

picoc

Website:

https://github.com/jpoirier/picoc

License:

Copyright (c) 2009-2011, Zik Saleeba

Copyright (c) 2015, Joseph Poirier

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Zik Saleeba nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

tar (binary only)

License:

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modifiedit, and giving a relevant date.

- b) The work must carry prominent notices stating that it isreleased under this License and any conditions added under section7. This requirement modifies the requirement in section 4 to “keep intact all notices”.

- c) You must license the entire work, as a whole, under thisLicense to anyone who comes into possession of a copy. ThisLicense will therefore apply, along with any applicable section 7additional terms, to the whole of the work, and all its parts,regardless of how they are packaged. This License gives nopermission to license the work in any other way, but it does notinvalidate such permission if you have separately received it.

- d) If the work has interactive user interfaces, each must displayAppropriate Legal Notices; however, if the Program has interactiveinterfaces that do not display Appropriate Legal Notices, yourwork need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product(including a physical distribution medium), accompanied by theCorresponding Source fixed on a durable physical mediumcustomarily used for software interchange.

- b) Convey the object code in, or embodied in, a physical product(including a physical distribution medium), accompanied by awritten offer, valid for at least three years and valid for aslong as you offer spare parts or customer support for that productmodel, to give anyone who possesses the object code either (1) acopy of the Corresponding Source for all the software in theproduct that is covered by this License, on a durable physicalmedium customarily used for software interchange, for a price nomore than your reasonable cost of physically performing thisconveying of source, or (2) access to copy theCorresponding Source from a network server at no charge.

- c) Convey individual copies of the object code with a copy of thewritten offer to provide the Corresponding Source. Thisalternative is allowed only occasionally and noncommercially, andonly if you received the object code with such an offer, in accordwith subsection 6b.

- d) Convey the object code by offering access from a designatedplace (gratis or for a charge), and offer equivalent access to theCorresponding Source in the same way through the same place at nofurther charge. You need not require recipients to copy theCorresponding Source along with the object code. If the place tocopy the object code is a network server, the Corresponding Sourcemay be on a different server (operated by you or a third party)that supports equivalent copying facilities, provided you maintainclear directions next to the object code saying where to find theCorresponding Source. Regardless of what server hosts theCorresponding Source, you remain obligated to ensure that it isavailable for as long as needed to satisfy these requirements.

- e) Convey the object code using peer-to-peer transmission, providedyou inform other peers where the object code and CorrespondingSource of the work are being offered to the general public at nocharge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from theterms of sections 15 and 16 of this License; or

- b) Requiring preservation of specified reasonable legal notices orauthor attributions in that material or in the Appropriate LegalNotices displayed by works containing it; or

- c) Prohibiting misrepresentation of the origin of that material, orrequiring that modified versions of such material be marked inreasonable ways as different from the original version; or

- d) Limiting the use for publicity purposes of names of licensors orauthors of the material; or

- e) Declining to grant rights under trademark law for use of sometrade names, trademarks, or service marks; or

- f) Requiring indemnification of licensors and authors of thatmaterial by anyone who conveys the material (or modified versions ofit) with contractual assumptions of liability to the recipient, forany liability that these contractual assumptions directly impose onthose licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

ant-design

Website:

https://ant.design/

array-move

Website:

https://github.com/sindresorhus/array-move

License:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

lodash

Website:

https://lodash.com/

License:

Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

mobx

Website:

https://mobx.js.org/README.html

ramda

Website:

https://ramdajs.com/

License:

The MIT License (MIT)

Copyright (c) 2013-2019 Scott Sauyet and Michael Hurley

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

rc-menu

Website:

https://github.com/react-component/menu

License:

The MIT License (MIT)

Copyright (c) 2014-present yiminghe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

react

Website:

https://reactjs.org/

License:

MIT License

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

react-dnd

Website:

https://github.com/react-dnd/react-dnd/

License:

The MIT License (MIT)

Copyright (c) 2015 Dan Abramov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

react-router

Website:

https://github.com/ReactTraining/react-router

License:

MIT License

Copyright (c) React Training 2016-2018

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

rxjs

Website:

https://github.com/ReactiveX/rxjs

License:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License.

You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Support

For help with a component or product, please submit a technical support request at http://support.lmi3d.com/.

If you are unable to submit a support request or prefer to contact LMI by phone or email, use the contact information below.

	Response times for phone or email support requests are longer than requests submitted through LMI's support request system.

North America

Phone	Vancouver +1 833 Gocator (+1 833 462 2867) Detroit +1 248 298 2839
Fax	+1 604 516 8368
Email	support@lmi3d.com

Europe

Phone	+31 45 850 7000
Fax	+31 45 574 2500
Email	support@lmi3d.com

For more information on safety and laser classifications, please contact:

U.S. Food and Drug Administration
Center for Devices and Radiological Health
WO66-G609
10903 New Hampshire Avenue
Silver Spring MD 20993-0002
USA

Contact

Americas	EMEAR	ASIA PACIFIC
LMI Technologies (Head Office)	LMI Technologies GmbH	LMI (Shanghai) Trading Co., Ltd.
Burnaby, Canada	Berlin, Germany	Shanghai, China
+1 604 636 1011	+49 (0)3328 9360 0	+86 21 5441 0711

LMI Technologies has sales offices and distributors worldwide. All contact information is listed at lmi3D.com/contact/locations.