VOS1000/2000/5000

Universal 2-D Vision Sensor

Manual

Your automation, our passion.

PEPPERL+FUCHS

With regard to the supply of products, the current issue of the following document is applicable: The General Terms of Delivery for Products and Services of the Electrical Industry, published by the Central Association of the Electrical Industry (Zentralverband Elektrotechnik und Elektroindustrie (ZVEI) e.V.) in its most recent version as well as the supplementary clause: "Expanded reservation of proprietorship"

Worldwide

Pepperl+Fuchs Group

Lilienthalstr. 200

68307 Mannheim

Germany

Phone: +49 621 776 - 0

E-mail: info@de.pepperl-fuchs.com

North American Headquarters

Pepperl+Fuchs Inc.

1600 Enterprise Parkway

Twinsburg, Ohio 44087

USA

Phone: +1 330 425-3555

E-mail: sales@us.pepperl-fuchs.com

Asia Headquarters

Pepperl+Fuchs Pte. Ltd.

P+F Building

18 Ayer Rajah Crescent

Singapore 139942

Phone: +65 6779-9091

E-mail: sales@sg.pepperl-fuchs.com

https://www.pepperl-fuchs.com

2021-03

2021-03

1 Introduction

1.1 Content of this Document

This document contains information required to use the product in the relevant phases of the product life cycle. This may include information on the following:

- Product identification
- Delivery, transport, and storage
- Mounting and installation
- Commissioning and operation
- Maintenance and repair
- Troubleshooting
- Dismounting
- Disposal

Note

For full information on the product, refer to the further documentation on the Internet at www.pepperl-fuchs.com.

The documentation comprises the following parts:

- This document
- Datasheet

In addition, the documentation may comprise the following parts, if applicable:

- EU-type examination certificate
- EU declaration of conformity
- Attestation of conformity
- Certificates
- Control drawings
- Instruction manual
- Other documents

1.2 Target Group, Personnel

Responsibility for planning, assembly, commissioning, operation, maintenance, and dismounting lies with the plant operator.

Only appropriately trained and qualified personnel may carry out mounting, installation, commissioning, operation, maintenance, and dismounting of the product. The personnel must have read and understood the instruction manual and the further documentation.

Prior to using the product make yourself familiar with it. Read the document carefully.

1.3 Symbols Used

This document contains symbols for the identification of warning messages and of informative messages.

Warning Messages

You will find warning messages, whenever dangers may arise from your actions. It is mandatory that you observe these warning messages for your personal safety and in order to avoid property damage.

Depending on the risk level, the warning messages are displayed in descending order as follows:

Danger!

This symbol indicates an imminent danger.

Non-observance will result in personal injury or death.

Warning!

This symbol indicates a possible fault or danger.

Non-observance may cause personal injury or serious property damage.

Caution!

This symbol indicates a possible fault.

Non-observance could interrupt the device and any connected systems and plants, or result in their complete failure.

Informative Symbols

Note

This symbol brings important information to your attention.

Action

This symbol indicates a paragraph with instructions. You are prompted to perform an action or a sequence of actions.

2 Product Description

2.1 Introduction

Vision sensors VOS1000, VOS2000, and VOS5000 (hereinafter referred to as sensors) are image-processing sensors for noncontact detection and testing of objects and for reading 1-D and 2-D codes in industrial environments.

To use the sensor to test an object, you must configure the sensor using the free Vision Configuration Tool software. The software must be installed on a PC. The software allows you to adapt existing jobs or simulate offline (i.e., without the sensor being connected).

During the configuration process, you create one or more test tasks, called jobs. A job consists of:

- A taught-in reference image for each object to be tested
- The vision tools that check for one or more features in the defined image areas of the object
- The function blocks assigned to the digital outputs, e.g., output 1 signals the result "Pass" and output 2 signals "Fail"

You can save up to 32 jobs on a single sensor. Several vision tools can be parameterized and executed simultaneously within a job. Digital inputs can be used to switch between jobs during operation.

The measurement results from the camera, such as code reading, can be preprocessed in the sensor using an integrated scripting language and output digitally to a data interface such as TCP/IP, PROFINET, and EtherNet/IP.

2.2 Functional Description

The available vision tools cover a very wide range of applications, such as:

Detection

The simple characteristic checks or position checks based on connected pixel contours or edges. In addition, it is possible to count previously learned objects or to specifically detect which object is currently in front of the sensor.

Positioning and Guidance

The exact positioning of an object using a reference contour in the image or the guidance of a robot arm. In addition, locators can be defined to perform characteristic checks on objects in any position.

Identification and Character Reading (OCR)

The code reading of common 1-D codes or 2-D codes. A variety of common code symbologies are supported.

In addition, there are optimized identification vision tools to detect direct product marks such as lasered or needled codes.

Fonts can be learned using Optical Character Recognition (OCR) tools, to support the automated reading of printed fonts such as date codes on labels.

Optical Gauging

Optical gauging tasks can be performed once the camera image is calibrated. Component tolerances such as object widths, circle diameters, gaps, threads or other complex shapes can be reliably determined and monitored.

Designs and Versions

There are two housing designs:

VOS1000/2000: F226 as a compact camera either with internal illumination, optics, or as a C-mount version. VOS1000 = 640 x 480 pixels, VOS2000 = 1280 x 960 pixels

For more complex measuring tasks, the VOS5000 is available under F227 as a five megapixel camera (2560 x 2048 pixels) with a C-mount and external illumination.

Special Features

- 100% automatic inspection
- Integrated optics, illumination, and vision tools
- Remote access via software, image display
- Automatic image transfer to an FTP server
- Job changeover via hardware, teach-in input
- Very high degree of flexibility using various vision tools
- Industry I/Os and PLC protocols integrated
- Password protection
- Preprocessing of measurement results through integrated scripting language
- Software on the camera. Integrated analyzer with flexible output string formatting

2021-03

2.3 Versions

The sensor types are offered in two versions. Each version offers a different housing design. VOS1000/2000 ("F226" housing design) as a compact camera with internal illumination or as a C-mount version.

The VOS5000 is available under the F227 housing design for more complex measuring tasks. This version is a five megapixel camera with a C-mount lens and a connection for external lighting.

"F226" Housing Design

M12 lens

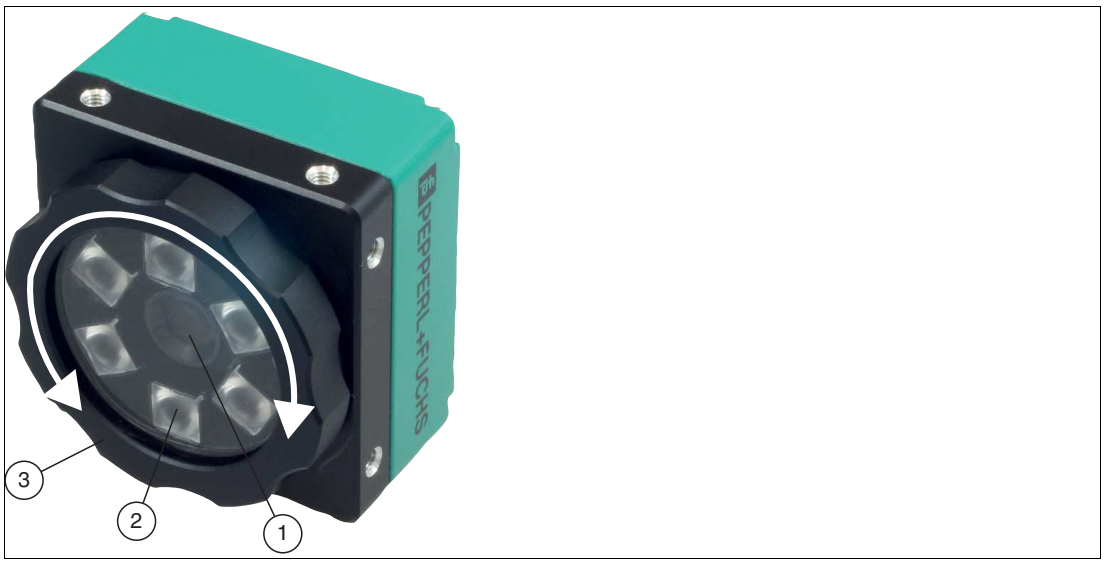


Figure 2.1 M12 lens (1), cover (3), and LED ring light (1)

The VOS1000-F226R-8MM-S and VOS2000-F226R-8MM-S sensors are equipped with an M12 lens and a focal length of 8 mm. Greater focal lengths (6, 12, and 16 mm) are available on request.

The M12 option comes with a special IP67 cover that allows you to adjust the focus by turning the lens cap without removing the cover.

The camera with the M12 lens also has an LED ring light, which is mounted inside the cover. The ring light is available in red. White and blue ring lights are available on request.

C-mount lens

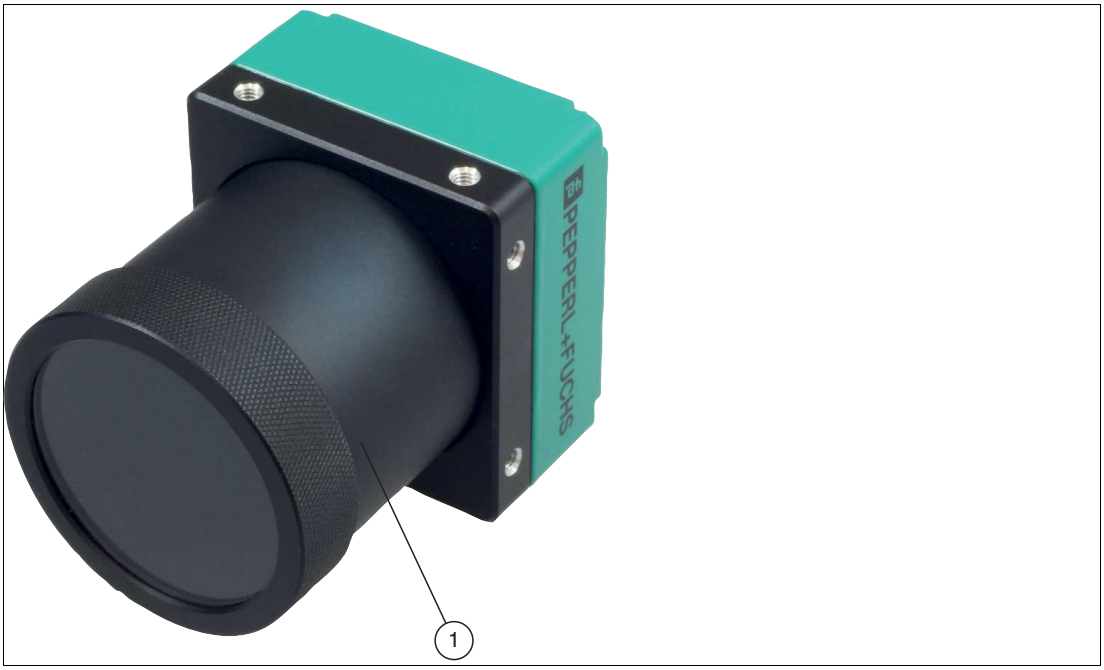


Figure 2.2 C-mount lens (1)

The VOS2000-F226-C-S sensor has a C-mount lens. The C-mount lens does not feature internal illumination. Optional external lighting can be used.

"F227" Housing Design

C-mount lens

Figure 2.3 C-mount lens incl. lighting

The VOS5000-F227-C-S sensor has a C-mount lens. The C-mount lens does not feature internal illumination. Optional external lighting can be used.

3 Delivery, Transport and Storage

3.1 Unpacking

Check the product for damage while unpacking. If the product should be damaged, inform the post office or parcel service and notify the supplier.

Retain the original packaging in case the device must be stored or shipped again at a later date.

Should you have any questions, please contact Pepperl+Fuchs.

3.2 Transport

The following guidelines must be observed and followed to ensure safe transport:

Note

Damage to the device due to improper transport!

- Do not remove packaging prior to assembly.
- Note the symbols on the packaging.
- Pack the device for transport such that it is protected from impact and humidity. Use the original packaging, which provides optimal protection.

3.3 Storage

When storing the device, pack it to protect it from shock and contamination. The original packaging provides optimal protection for the device. Store the device in a dry environment protected from humidity and potential harmful debris.

4 Mounting

4.1 Sensing Range

The size of the field of view depends on the operating distance, the lens used, and the optical resolution. Depending on the measuring task, the appropriate lens must be selected. Note the sensing range of the respective lens when determining lens placement.

Note

Definition

- **Operating distance**: This is the distance between the optical surface of the sensor and the test object.
- **Field of view**: This is the image surface visible to a sensor at a given operating distance. If you need a larger field of view, you must increase the distance between the sensor and the object being tested. As the field of view increases in size, the optical resolution decreases. This affects the maximum achievable accuracy.

[statictext not found!] - Key:

Pepperl+Fuchs LensHelp

For support when planning your sensing range, you can download the **Pepperl+Fuchs LensHelp** application from the download page of our website http://www.pepperl-fuchs.com. Simply enter the product name or item number in the **Product/Keyword** search box and click **Search**.

Select your product from the list of search results. In the list of product information, click **Software**.

The figures below show fields of view at different operating distances for each lens focal length. The values in the table below serve as a reference point. The actual values may differ on account of the lens tolerances.

4.1.1 VOS1000 and VOS2000 Sensing Range

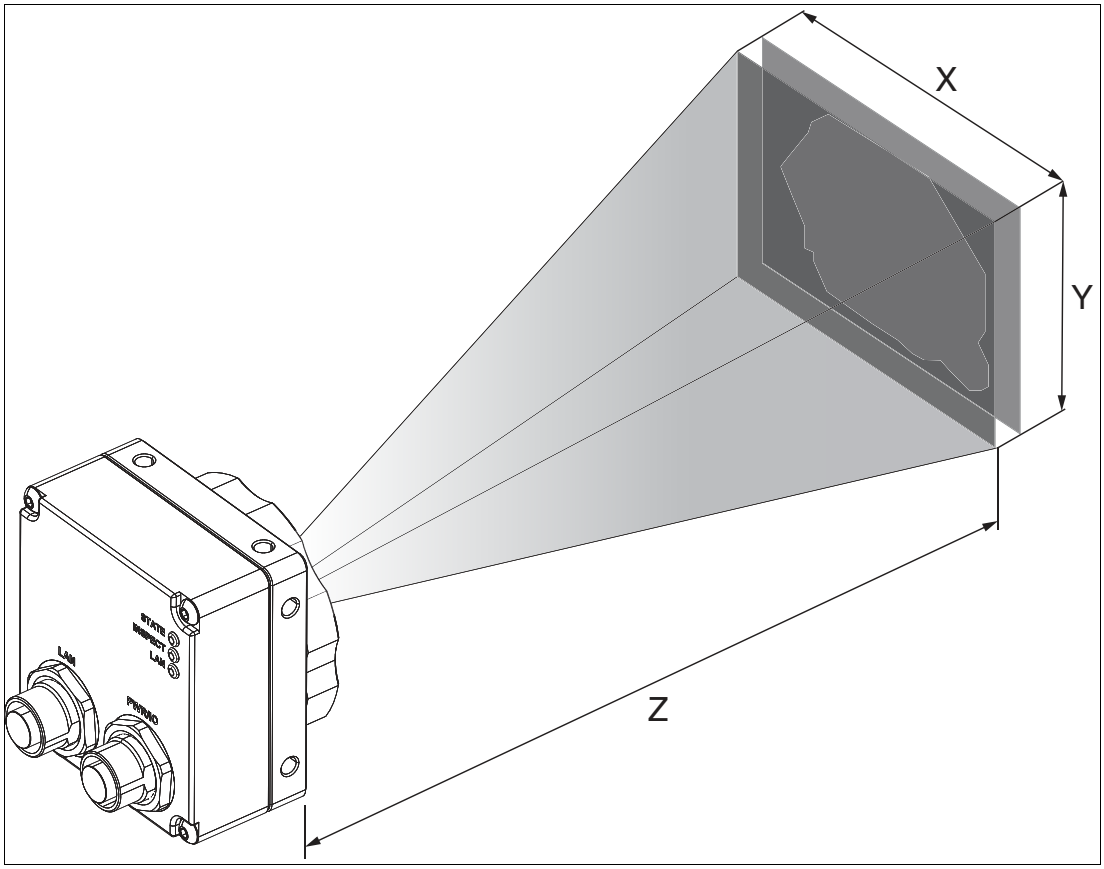
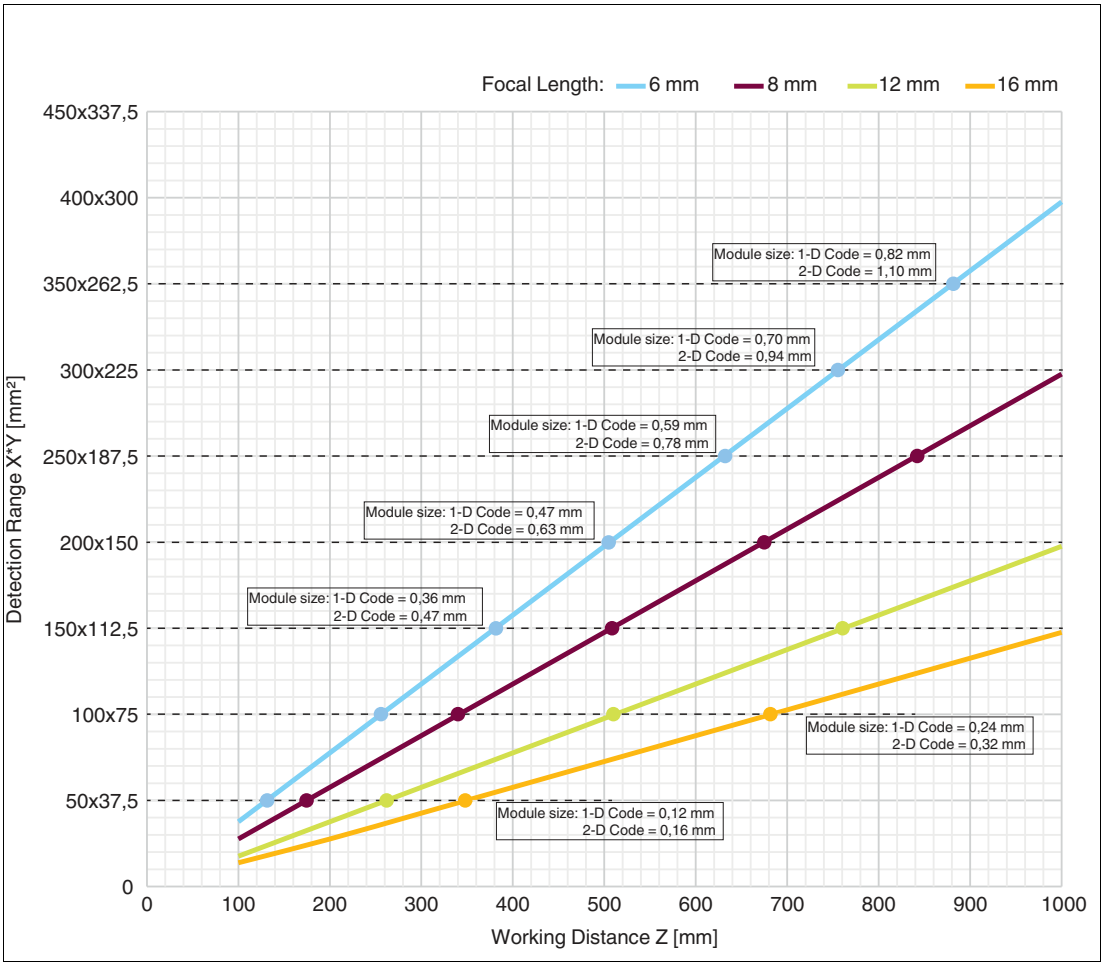


Figure 4.1 Sensing range (schematic diagram)

The following table shows the sensing range at different operating distances for each lens length. The values in the table below serve as a reference point. The actual values may differ on account of the lens tolerances.

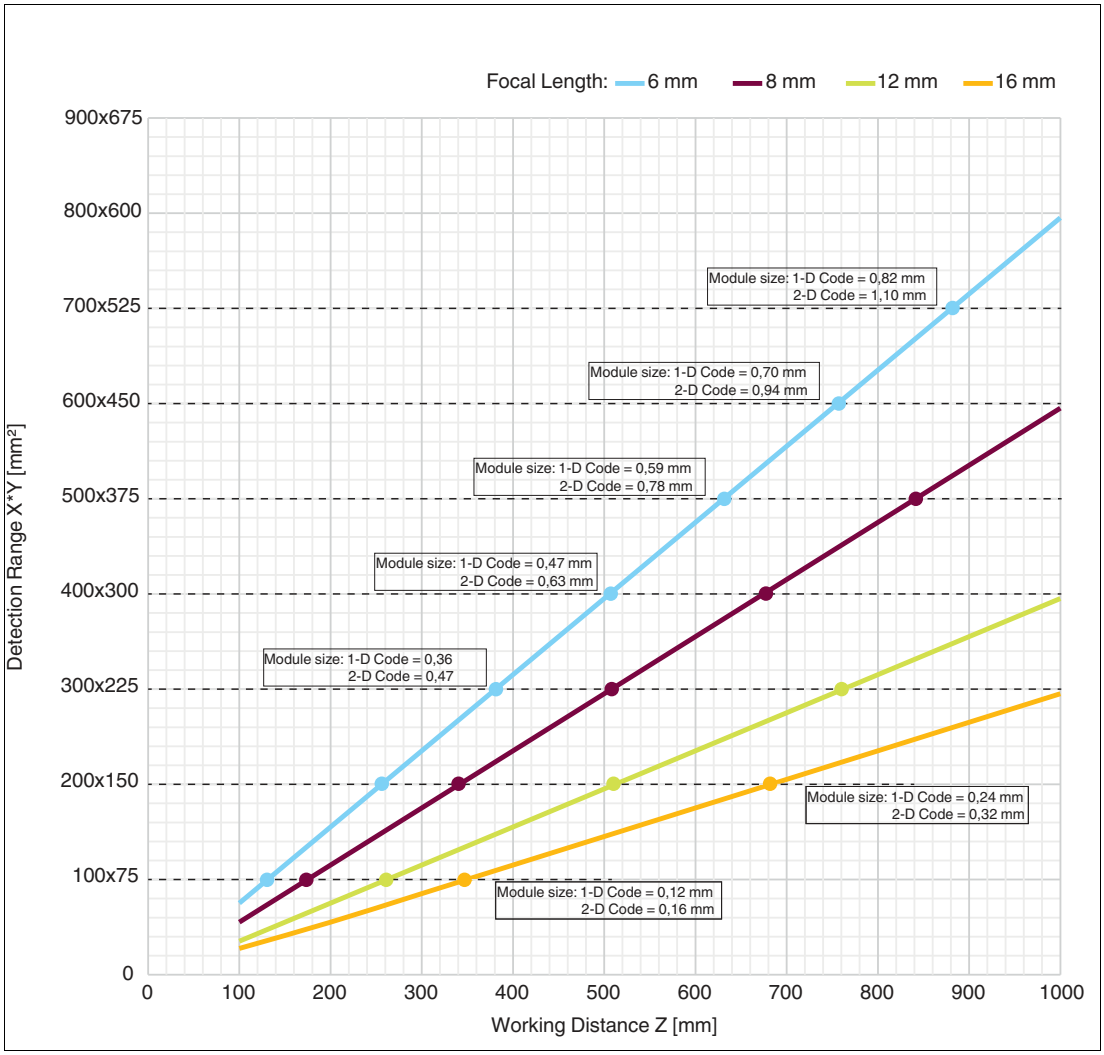
VOS1000 Read Field Diagram



Operating distance Z [mm]	VOS1000 sensing range							
	6 mm lens		8 mm lens		12 mm lens		16 mm lens	
	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]
100	37.6	28.2	27.6	20.7	17.6	13.2	16.65	12.49
200	77.6	58.2	57.6	43.2	37.6	28.2	27.6	20.7
300	117.6	88.2	87.6	65.7	57.6	43.2	42.6	31.95
400	157.6	118.2	117.6	88.2	77.6	58.2	57.6	43.2
500	197.6	148.2	147.6	110.7	97.6	73.2	72.6	54.45
600	237.6	178.2	177.6	133.2	117.6	88.2	87.6	65.7
700	277.6	208.2	207.6	155.7	137.6	103.2	102.6	76.95
800	317.6	238.2	237.6	178.2	157.6	118.2	117.6	88.2
900	357.6	268.2	267.6	200.7	177.6	133.2	132.6	99.45
1000	397.6	298.2	297.6	223.2	197.6	148.2	147.6	110.7

2021-03

VOS2000 Read Field Diagram



Operating distance Z [mm]	VOS2000 sensing range							
	6 mm lens		8 mm lens		12 mm lens		16 mm lens	
	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]
100	75.2	56.4	55.2	41.4	35.2	26.4	33.3	24.98
200	155.2	116.4	115.2	86.4	75.2	56.4	55.2	41.4
300	235.2	176.4	175.2	131.4	115.2	86.4	85.2	63.9
400	315.2	236.4	235.2	176.4	155.2	116.4	115.2	86.4
500	395.2	296.4	295.2	221.4	195.2	146.4	145.2	108.9
600	475.2	356.4	355.2	266.4	235.2	176.4	175.2	131.4
700	555.2	416.4	415.2	311.4	275.2	206.4	205.2	153.9
800	635.2	476.4	475.2	356.4	315.2	236.4	235.2	176.4
900	715.2	536.4	535.2	401.4	355.2	266.4	265.2	198.9
1000	795.2	596.4	595.2	446.4	395.2	296.4	295.2	221.4

4.1.2 VOS2000 C-Mount Sensing Range

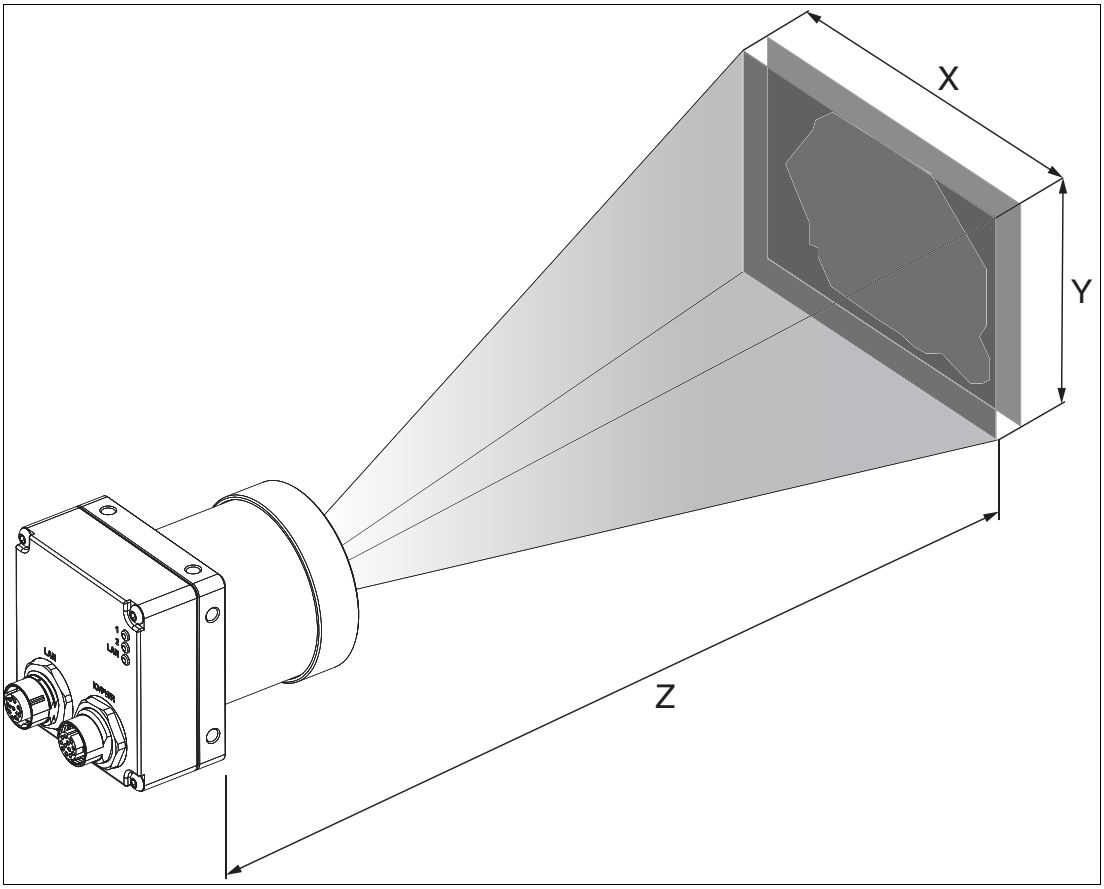
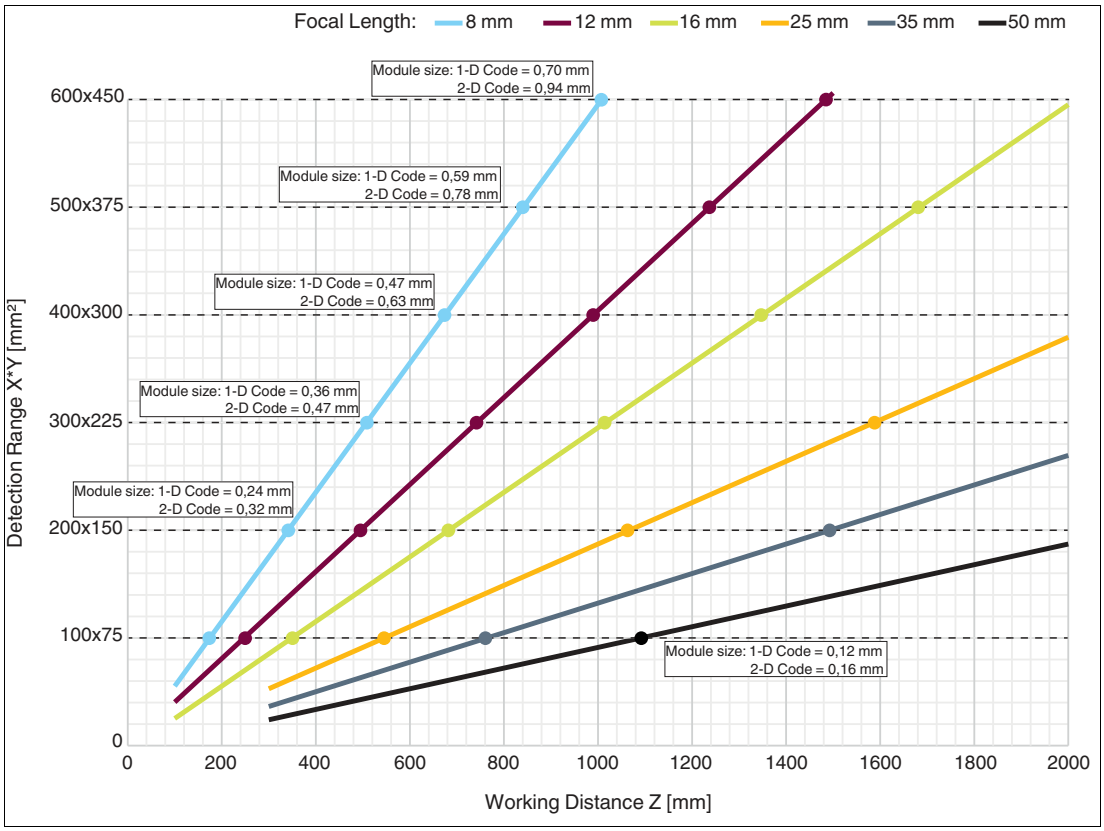


Figure 4.2 Sensing range (schematic diagram)

The following table shows the sensing range at different operating distances for each lens length. The values in the table below serve as a reference point. The actual values may differ on account of the lens tolerances.

VOS2000 C-Mount Read Field Diagram



Operating distance Z [mm]	VOS2000 C-Mount—Sensing Range					
	8 mm lens		12 mm lens		16 mm lens	
	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]
100	55.20	41.40	40.40	30.30	25.2	18.9
200	115.2	86.4	80.81	60.60	55.2	41.4
300	175.2	131.4	121.21	90.91	85.2	63.9
400	235.2	176.4	161.61	121.21	115.2	86.4
500	295.2	221.4	202.01	151.51	145.2	108.9
600	355.2	266.4	242.42	181.81	175.2	131.4
700	415.2	311.4	282.82	212.11	205.2	153.9
800	475.2	356.4	323.22	242.42	235.2	176.4
900	535.2	401.4	363.62	272.72	265.2	198.9
1000	595.2	446.4	404.03	303.02	295.2	221.4
1250	-	-	505.03	378.78	370.2	277.65
1500	-	-	606.04	454.53	445.2	333.9
1750	-	-	-	-	520.2	390.15
2000	-	-	-	-	595.2	446.4

Operating distance Z [mm]	VOS2000 C-Mount—Sensing Range					
	25 mm lens		35 mm lens		50 mm lens	
	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]
100	-	-	-	-	-	-
200	-	-	-	-	-	-
300	52.8	39.6	36.34	27.26	24	18
400	72	54	50.06	37.54	33.6	25.2
500	91.2	68.4	63.77	47.83	43.2	32.4
600	110.4	82.8	77.49	58.11	52.8	39.6
700	129.6	97.2	91.2	68.4	62.4	46.8
800	148.8	111.6	104.91	78.69	72	54
900	168	126	118.63	88.97	81.6	61.2
1000	187.2	140.4	132.34	99.26	91.2	68.4
1250	235.2	176.4	166.63	124.97	115.2	86.4
1500	283.2	212.4	200.91	150.69	139.2	104.4
1750	331.2	248.4	235.2	176.4	163.2	122.4
2000	379.2	284.4	269.49	202.11	187.2	140.4

2021-03

4.1.3 VOS5000 Sensing Range

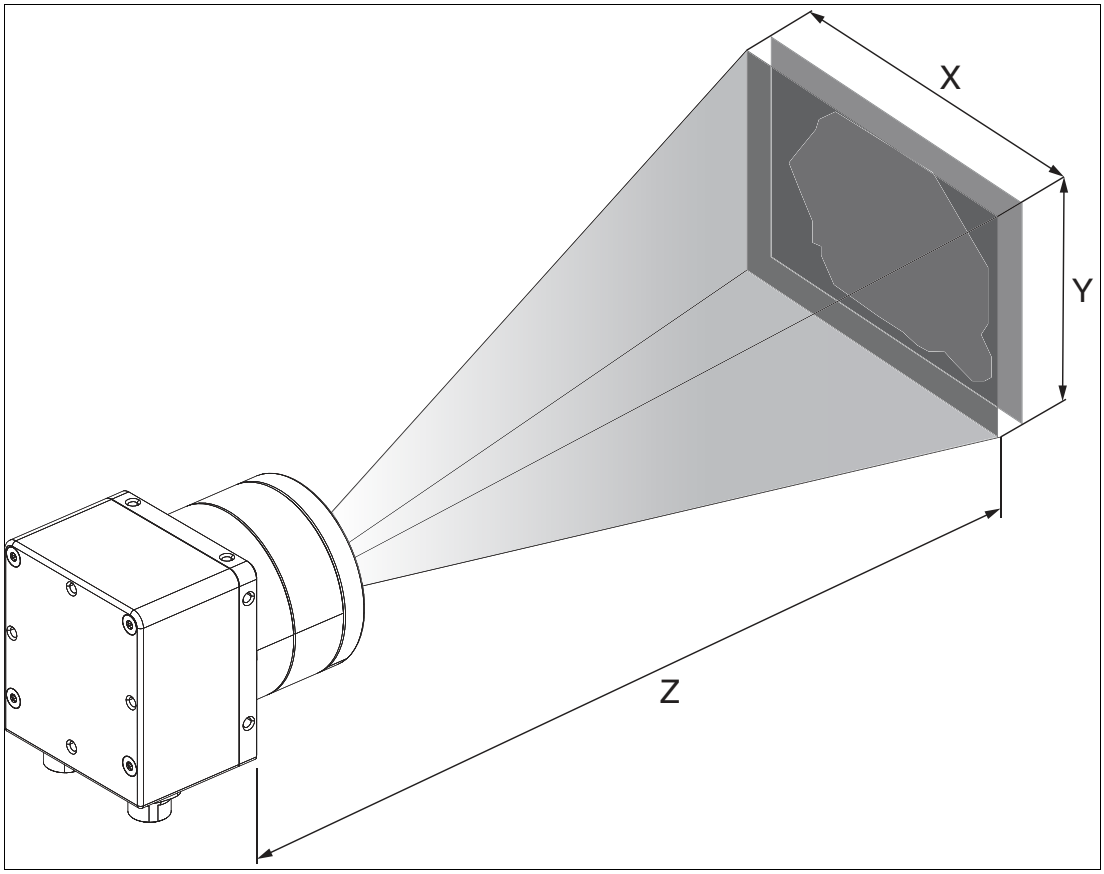
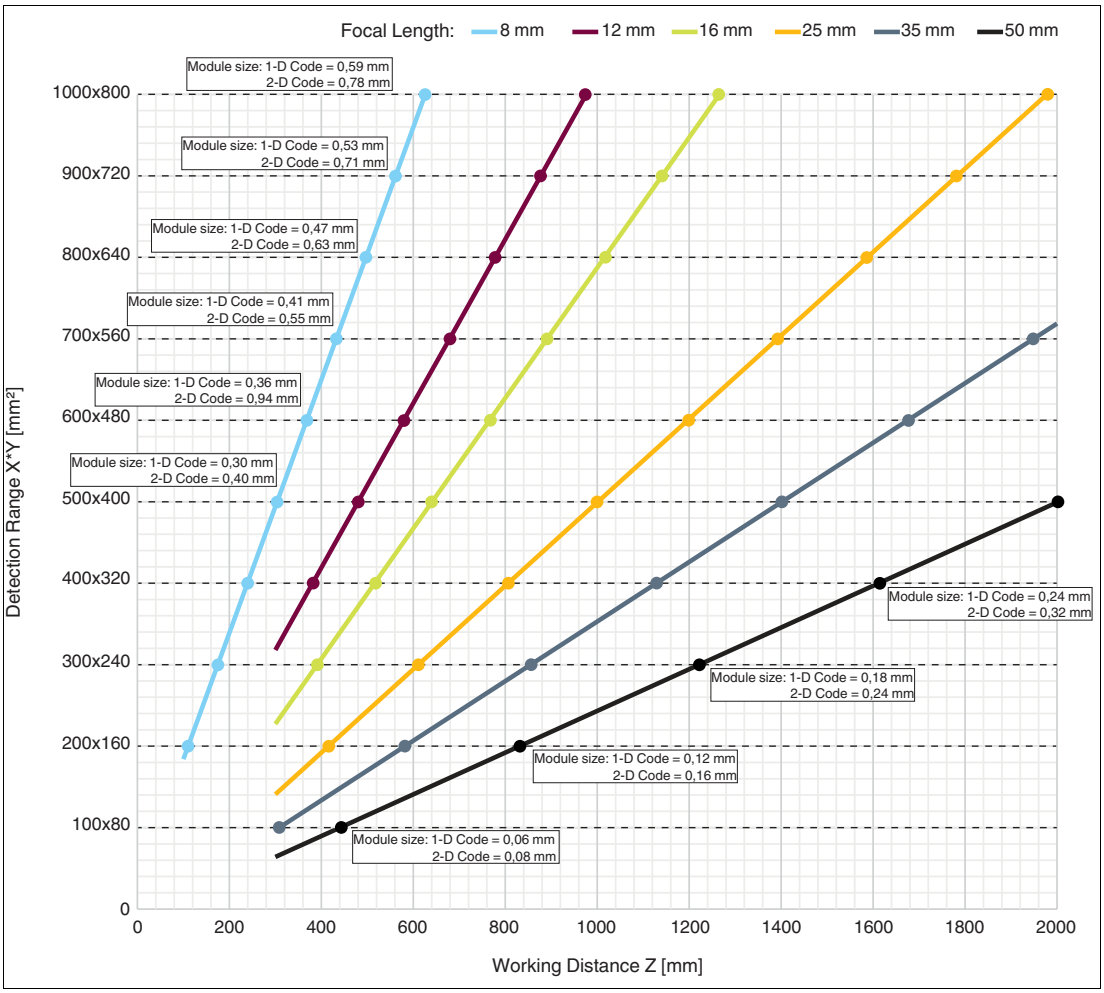


Figure 4.3 Sensing range (schematic diagram)

The following table shows the sensing range at different operating distances for each lens length. The values in the table below serve as a reference point. The actual values may differ on account of the lens tolerances.

VOS5000 Read Field Diagram



Operating distance Z [mm]	VOS500 sensing range					
	8 mm lens		12 mm lens		16 mm lens	
	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]
100	184	138	-	-	-	-
200	339	254	-	-	-	-
300	494	371	318	239	227.2	181.76
400	649	487	419	314	307.2	245.76
500	804	603	520	390	387.2	309.76
600	960	720	621	465	467.2	373.76
700	1115	836	721	541	547.2	437.76
800	1270	952	822	617	627.2	501.76
900	1425	1069	923	692	707.2	565.76
1000	1580	1185	1024	768	787.2	629.76
1250	-	-	-	-	987.2	789.76
1500	2356	1767	1528	1146	1187.2	949.76
1750	-	-	-	-	1387.2	1109.76
2000	3132	2349	2032	1524	1587.2	1269.76

2021-03

Operating distance Z [mm]	VOS500 sensing range					
	25 mm lens		35 mm lens		50 mm lens	
	X [mm]	Y [mm]	X [mm]	Y [mm]	X [mm]	Y [mm]
100	-	-	-	-	-	-
200	-	-	-	-	-	-
300	140.8	112.64	96.91	77.53	64	51.2
400	192	153.6	133.49	106.79	89.6	71.68
500	243.2	194.56	170.06	136.05	115.2	92.16
600	294.4	235.52	206.63	165.3	140.8	112.64
700	345.6	276.48	243.2	194.56	166.4	133.12
800	396.8	317.44	279.77	223.82	192	153.6
900	448	358.4	316.34	253.07	217.6	174.08
1000	499.2	399.36	352.91	282.33	243.2	194.56
1250	627.2	501.76	444.34	355.47	307.2	245.76
1500	755.2	604.16	535.77	428.62	371.2	296.96
1750	883.2	706.56	627.2	501.76	435.2	348.16
2000	1011.2	808.96	718.63	574.9	499.2	399.36

4.1.4 Module Size in Codes

When setting the operating distance between the sensor and the code, refer to the read field diagrams above. When setting the operating distance, the **"module size"** must be known, i.e., the width of the narrowest bar (for 1-D codes) or the side length of the square for Data Matrix codes (2-D codes). If the operating distance selected is too large, reading is either unreliable or not possible.

The values specified in the read field diagrams are maximum values for a specific module size. The operating distance must therefore be less than or equal to the specified distance for a certain module size.

1-D Code

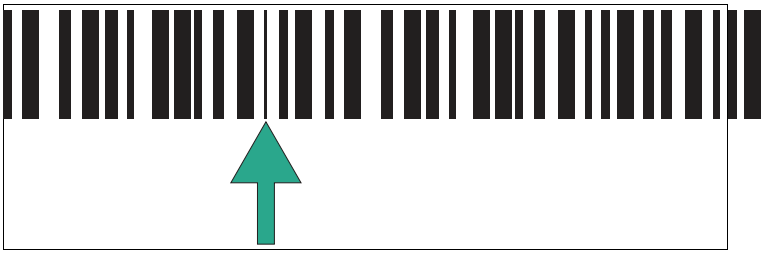


Figure 4.4 Module size = minimum bar width

In a one-dimensional barcode, a module refers to the narrowest element (bar or gap) in a code.

2-D Code

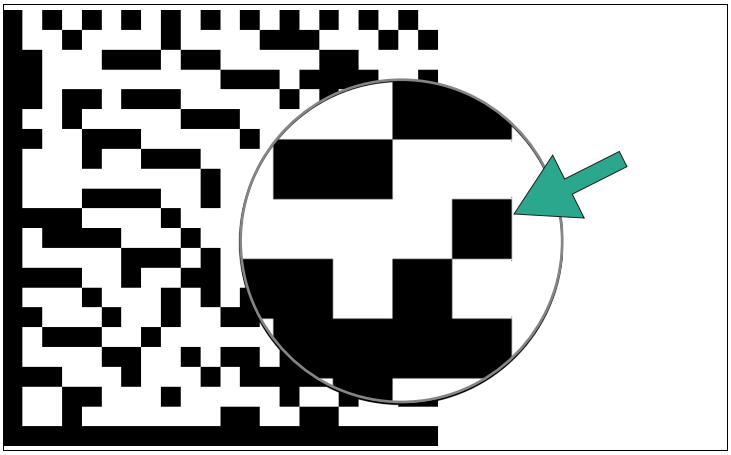


Figure 4.5 Module size = edge length of a square/bit

In Data Matrix codes, a module is a bit (black or white). The module size (also module width) is the edge length of a bit in millimeters or micrometers. Data Matrix codes are scalable, i.e., the module size can be adapted to the application.

4.2 Replacing M12 Lenses

Figure 4.6 M12 lens set "VOS-LE-M12-KIT"

The VOS1000-F226R-*MM-S and VOS2000-F226R-*MM-S sensors are equipped with M12 lenses. These lenses can be replaced as required. The M12 lens set "VOS-LE-M12-KIT" contains four M12 lenses with focal lengths of 6 mm, 8 mm, 12 mm, and 16 mm.

The focus can be adjusted by turning the lens cap.

The process for replacing an M12 lens is described below:

Removing M12 Lenses

Before you can remove the M12 lens, you must remove some parts of the sensor. Use extreme caution. Disconnect the power before working on the sensor.

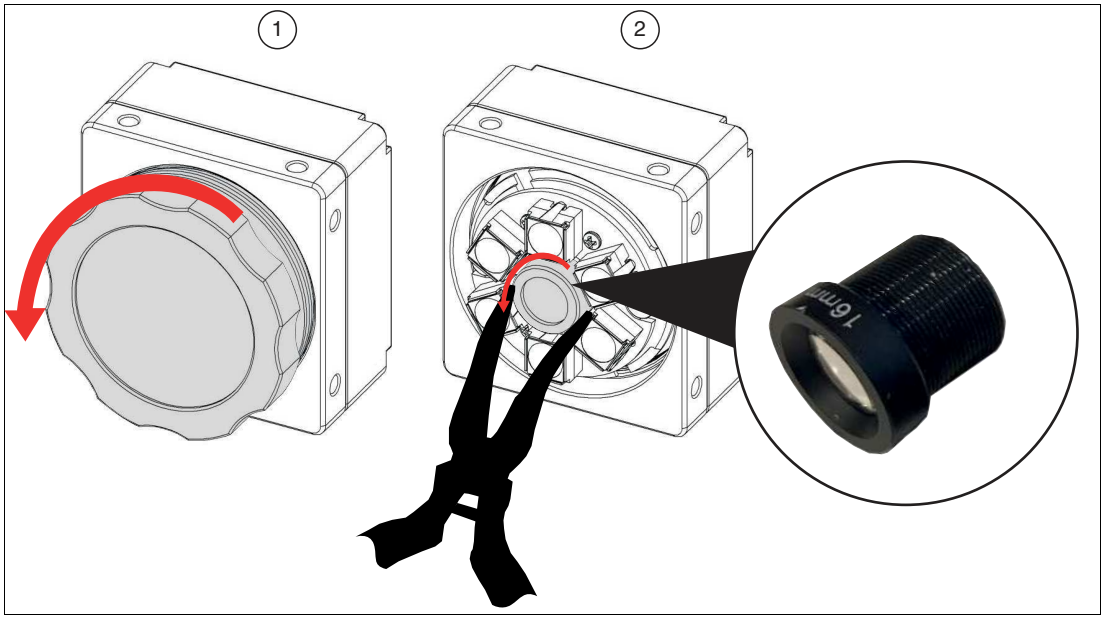


Figure 4.7 Replacing M12 lenses

1. Remove the lens cap (1) from the sensor by turning the lens cap (1) counter-clockwise until it comes off the sensor.
2. Remove the M12 lens (2) from the sensor by turning the M12 lens (2) counter-clockwise until it comes off the sensor.

Note

We recommend using curved needle-nose pliers to unscrew the M12 lens. The lenses are sensitive, so we recommend using heat-shrinkable tubes for the needle-nose pliers to avoid damaging the housing or the lenses.

Mounting M12 Lenses

Before you can remove the M12 lens, you must remove some parts of the sensor. Use extreme caution. Disconnect the power before working on the sensor.

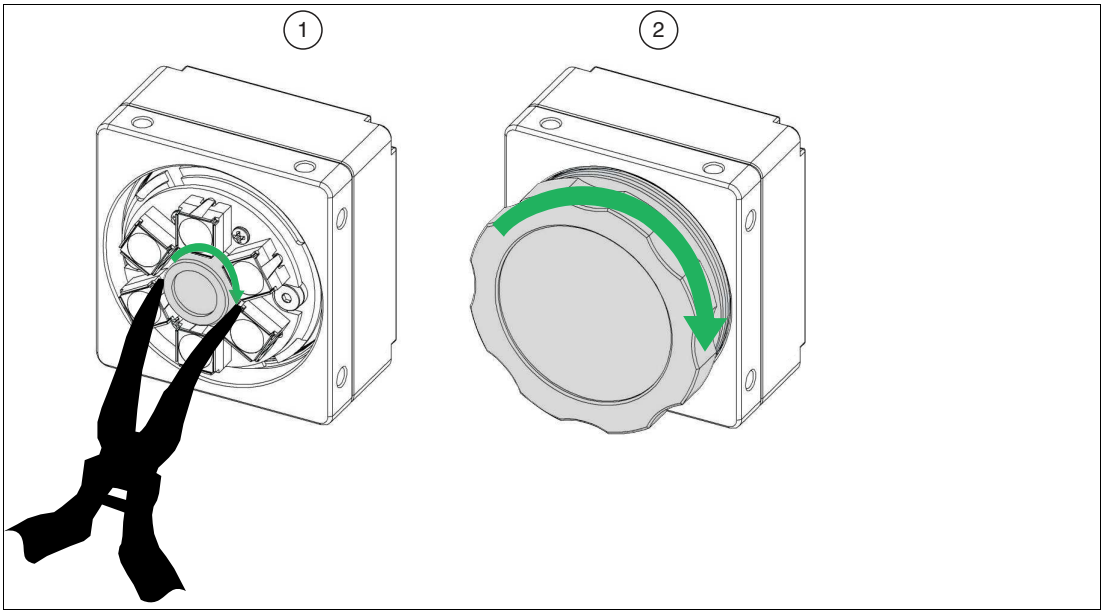


Figure 4.8 Replacing M12 lenses

1. Turn the new M12 lens (3) by hand clockwise into the sensor until it is firmly seated.
2. Screw the lens cap (4) by hand clockwise onto the sensor.

Note

We recommend using curved needle-nose pliers to screw the M12 lens in. The lenses are sensitive, so we recommend using heat-shrinkable tubes for the needle-nose pliers to avoid damaging the housing or the lenses.

4.3 Mounting the Sensor

Note the following factors when mounting the sensor to ensure reliable sensing mode:

- Sensing range between sensor and test object:
  - VOS1000 and VOS2000: see chapter 4.1.1
  - VOS2000 C-Mount: see chapter 4.1.2
  - VOS5000: see chapter 4.1.3
- Module size in codes, see chapter 4.1.4.
- Environmental influences: If the sensor lens becomes dirty or scratched after setup, the sensor can no longer function reliably. For this reason, select the mounting location such that contamination of any kind is largely avoided, or ensure regular cleaning. The specified ambient conditions in the datasheet must be observed.
- Prevention of ambient light such as sunlight and hall lighting.

Caution!

Damage to the equipment caused by improper mounting!

Device components can be damaged if the permissible screw-in depth and the maximum permissible tightening torque are exceeded.

Observe the maximum permissible screw-in depth to avoid damaging the device or mounting incorrectly.

Never exceed the maximum permissible tightening speed of the fixing screws.

Note

The final mechanical orientation can only take place after the electrical connections have been made because the captured images can only be displayed on the PC using the configuration software.

Once the sensor is mounted, ensure that there is still sufficient space to connect the cable to the sensor.

Securing and Aligning the Sensor

When positioning the sensor, ensure that the camera's field of vision is not obscured by other objects.

If you want to test an object with reflective spots, we recommend mounting the sensor at an angle of 10 to 20 degrees with respect to the vertical axis, to prevent reflections.

VOS1000/2000

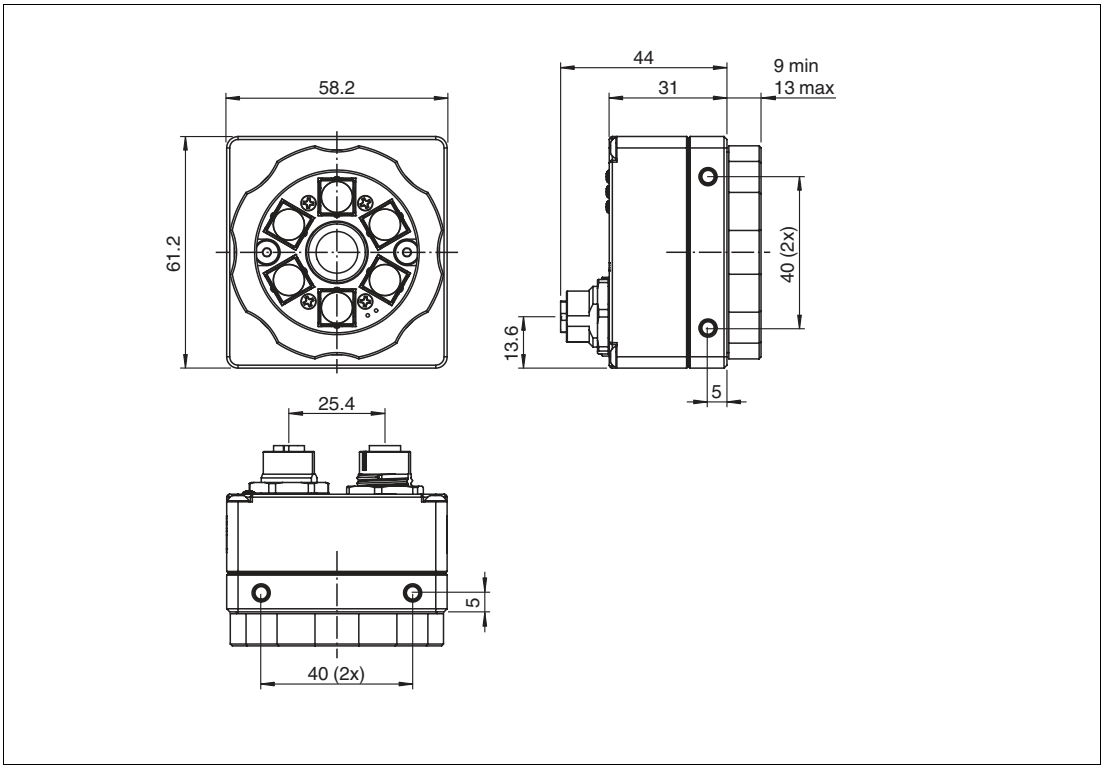
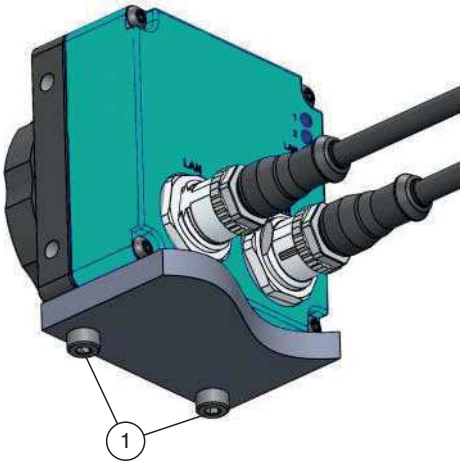
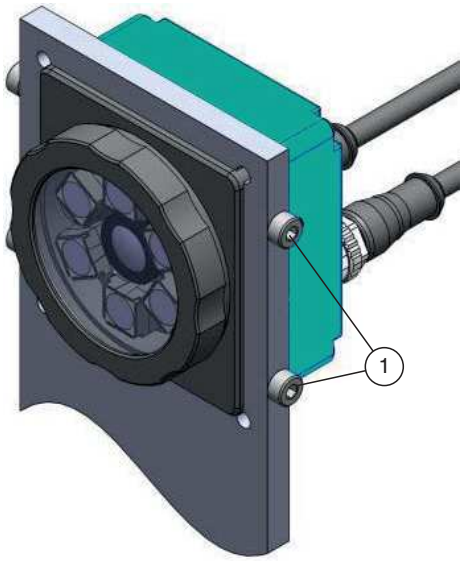


Figure 4.9 Dimensional drawing

Mounting Illustration	Information on Mounting
	Mounting on the underside of the device with M4 screws (1): You can use the two mounting holes with internal thread to mount the housing on the underside of the device. The maximum screw-in depth of the M4 screws is 7 mm.
	Side mounting with M4 screws (1): You can use the two mounting holes with internal thread on both sides of the sensor to mount the housing by its right side, left side, or by both sides. The maximum screw-in depth of the M4 screws is 7 mm.

VOS5000

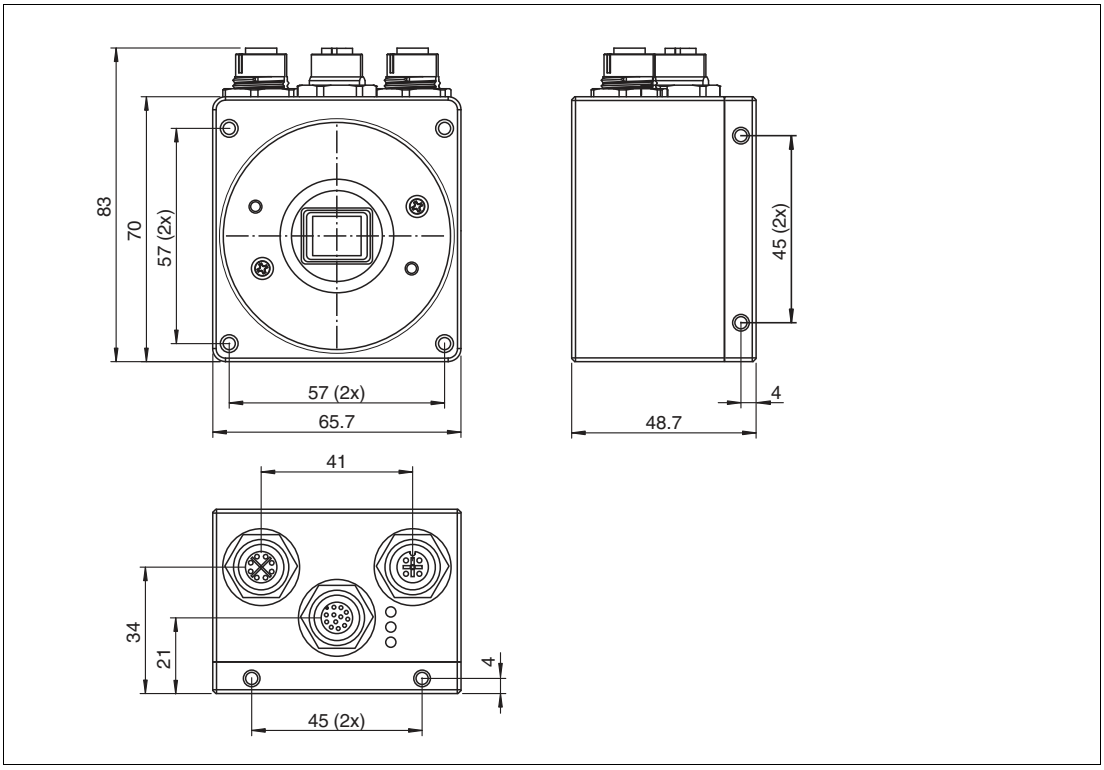
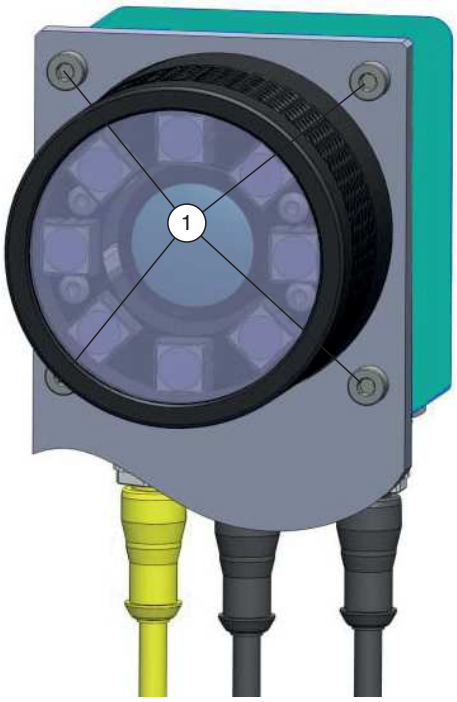


Figure 4.10 Dimensional drawing

Mounting Illustration	Information on Mounting
	Mounting on the top of the device with M4 screws (1): You can use the two mounting holes with internal thread to mount the housing on the top of the device. The maximum screw-in depth of the M4 screws is 7 mm.
	Mounting on the rear of the device with M4 screws (1): You can use the four mounting holes with internal thread to mount the housing on the rear of the device. The maximum screw-in depth of the M4 screws is 7 mm.
	Mounting on the front of the device with M4 screws (1): You can use the four mounting holes with internal thread to mount the housing on the front of the device. The maximum screw-in depth of the M4 screws is 7 mm.

Adjusting Image Sharpness / Focus Ring

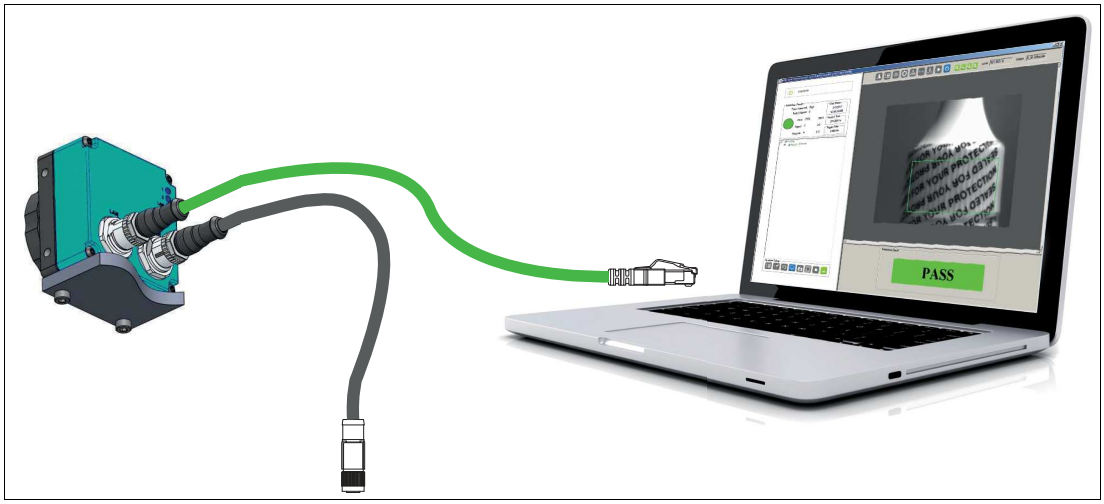
Turn the focus ring on the sensor to adjust the sharpness of the image. Turning the ring clockwise will focus objects that are further away. Turn the ring counter-clockwise to bring closer objects into focus.

5 Installation

5.1 Connecting the Sensor

Connecting VOS1000/2000

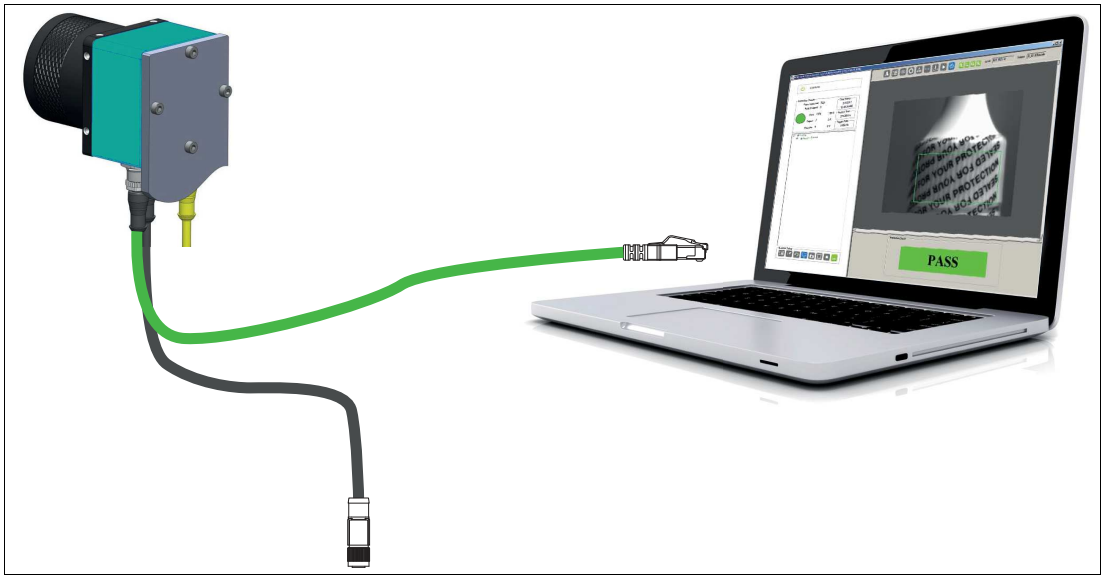
At the back of the sensor there are two connections: LAN and PWR/I/O.
To prepare the sensor for operation, proceed as follows:



1. Connect the Ethernet cable to the LAN connection on the sensor. Connect the other end of the Ethernet cable to the PC, PLC, or factory LAN (the PC is only required for setup).
2. Connect the connection module to the Power I/O connection on the sensor.
3. Connect the power supply to the connection module in the off state.

Connecting VOS5000

There are three connections on the base of the sensor: LAN, PWR/I/O, and LAMP.
To prepare the sensor for operation, proceed as follows:



1. Connect the Ethernet cable to the LAN connection on the sensor. Connect the other end of the Ethernet cable to the PC, PLC, or factory LAN (the PC is only required for setup).
2. Connect the connection module to the Power I/O connection on the sensor.
3. Connect the power supply to the connection module in the off state.

   For more information on electrical connection, see the following section.

2021-03

5.2 Electrical Connection

5.2.1 VOS1000 and VOS2000

There are two connections and three LEDs on the back of the sensor.

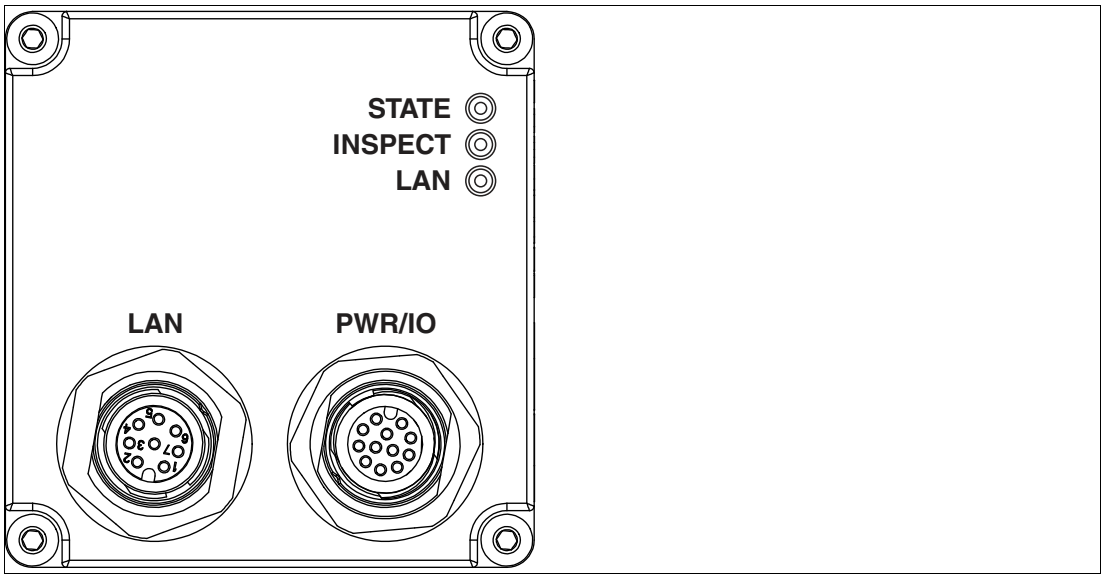


Figure 5.1 Back of the sensor with connections and LEDs

Sensor connections and LEDs

Connection/LED	Designation	Function
Connection	LAN	10/100 BaseT Ethernet connection. Provides the primary interface for configuring the sensor, developing the application, and monitoring the results.
	PWR/IO	Enables access to the sensor I/O: 3 IN, 3 OUT, RS-232. Also provides a PWR input (12 – 30 V). Note: The voltage supply to both connections is at the same potential. The voltage supply can be connected via the PWR/IO or LAN plug connection. Never connect both at the same time!
LED	STATE	• Lights up blue: sensor started, not configured (no job file) • Lights up green: job loaded, ready for execution • Flashes green: job loaded and executed, detection in progress • Lights up red: sensor fault
	INSPECT	• Flashes blue: starts (should stop after 20 seconds) • Lights up green: inspection confirmation • Lights up blue: test cycle (result of runtime decision) • Lights up red: inspection error (result of runtime decision)
	LAN	• Lights up blue: warm restart or restart • Red/green/yellow: network activity

Note

The sensor does not support the IEEE 802.3af Power over Ethernet (PoE) standard and should not be directly connected to a PoE-supported router.

LAN and PWR/IO Connection Assignment

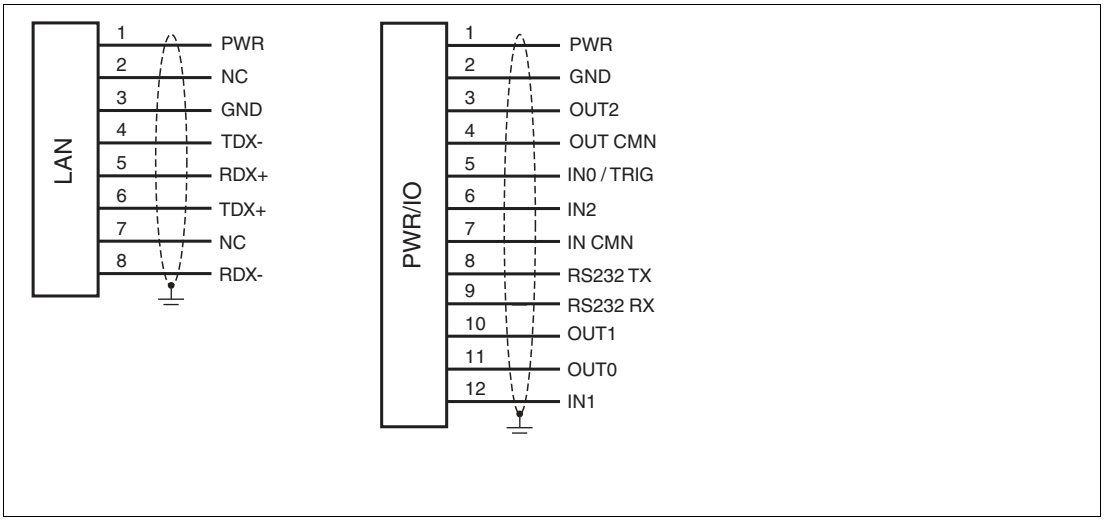
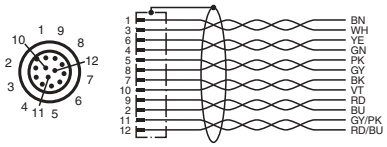
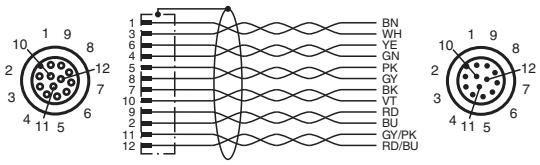
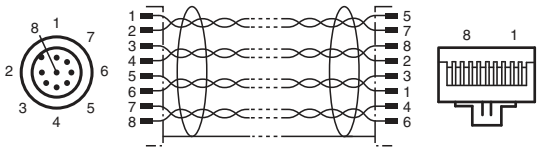


Figure 5.2 Power source diagram

Cable Overview

Cable	Description	Power Source
V112S-G-BK5M-PUR-ABG	Single-ended male cordset with M12 cable A-coded, 12-pin, PUR cable, black, shielded, open end	
V112-G-BK3M-PUR-ABG-V112-G	Cordset, M12 socket to M12 plug, A-coded, 12-pin, PUR cable, black, shielded	
V19S-G-GN5M-PUR-ABG-V45-G	Bus cable, Ethernet, M12 plug, A-coded, 8-pin to RJ45 Ethernet-coded, 8-pin, PUR cable, green, Cat6A, shielded	

5.2.2 VOS5000

There are three connections and three LEDs on the base of the sensor.

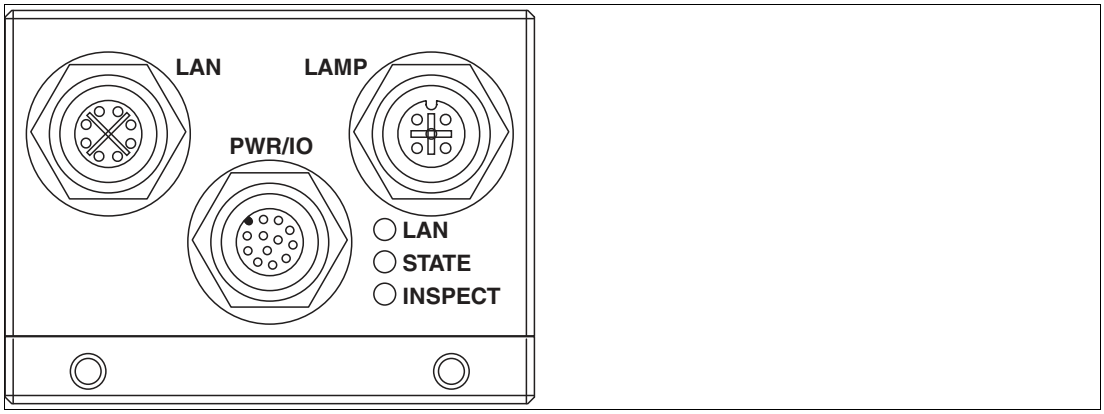


Figure 5.3 Base with connections and LEDs

Sensor connections and LEDs

Connection/LED	Designation	Function
Connection	LAN	10/100 BaseT Ethernet connection. Provides the primary interface for configuring the sensor, developing the application, and monitoring the results.
	PWR/IO	Enables access to the sensor I/O: 3 IN, 3 OUT, RS-232. Also provides a PWR input (12 – 30 V). Note: The voltage supply to both connections is at the same potential. The voltage supply can be connected via the PWR/IO or LAN plug connection. Never connect both at the same time!
	LAMP	Provides PWR and flash control for a local LED light source.
LED	LAN	• Lights up blue: warm restart • Red/green/yellow: network activity
	STATE	• Lights up blue: sensor started, not configured (no solution file) • Lights up green: solution loaded, ready for execution • Flashes green: solution loaded and executed, acquisition in progress • Lights up red: sensor fault
	INSPECT	• Flashes blue: starts (should stop after 20 seconds) • Lights up green: inspection confirmation (result of runtime decision) • Lights up blue: test cycle (result of runtime decision) • Lights up red: inspection error (result of runtime decision)

LAN and PWR/IO Connection Assignment

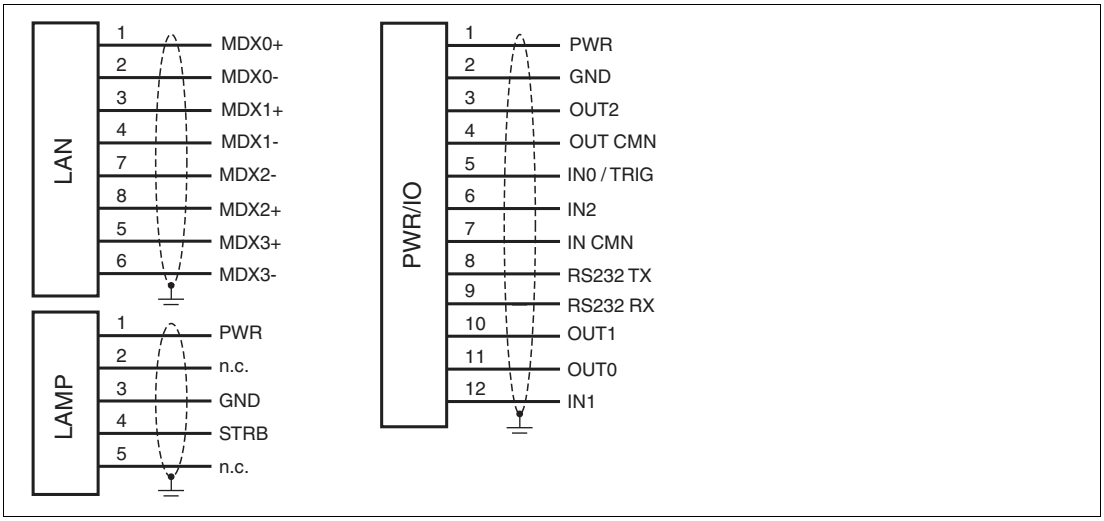
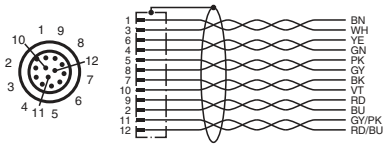
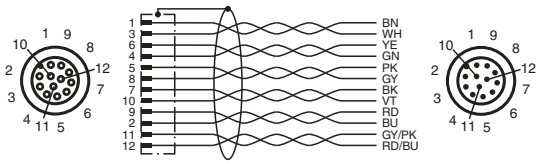
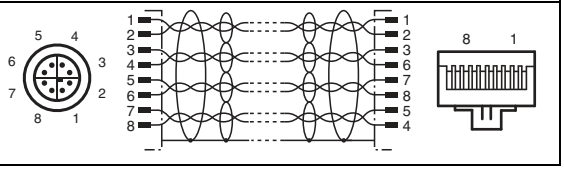


Figure 5.4 Power source diagram

Cable Overview

Cable	Description	Power Source
V112S-G-BK5M-PUR-ABG	Single-ended male cordset, M12, A-coded, 12-pin, PUR cable, black, shielded	
V112-G-BK3M-PUR-ABG-V112-G	Cordset, M12 socket to M12 plug, A-coded, 12-pin, PUR cable, black, shielded	
V19SX-G-GN5M-PUR-ABG-V45-G	Bus cable, Ethernet, M12 plug, X-coded, 8-pin to RJ45 Ethernet-coded, 8-pin, PUR cable, green, Cat6A, shielded	

6 Commissioning

6.1 Downloading and Installing Software

To set up the sensor, you need the Vision Configuration Tool application. The Vision Configuration Tool image processing application provides a range of image processing functions that meet various automated test requirements. There are two ways to download the application. Download directly from the Pepperl+Fuchs homepage or alternatively from the sensor supplied with the integrated Vision Configuration Tool application. The two ways to install the application are described below.

Installing the Vision Configuration Tool

This section describes how to obtain the installation file from the Pepperl+Fuchs homepage and then install it.

1. Go to the Pepperl+Fuchs homepage at http://www.pepperl-fuchs.com and enter the product name or item number in the search function. You can find the Vision Configuration Tool in the **Software** section of the product detail page for the device.
2. Save the installation file locally.
3. Start the exe file.

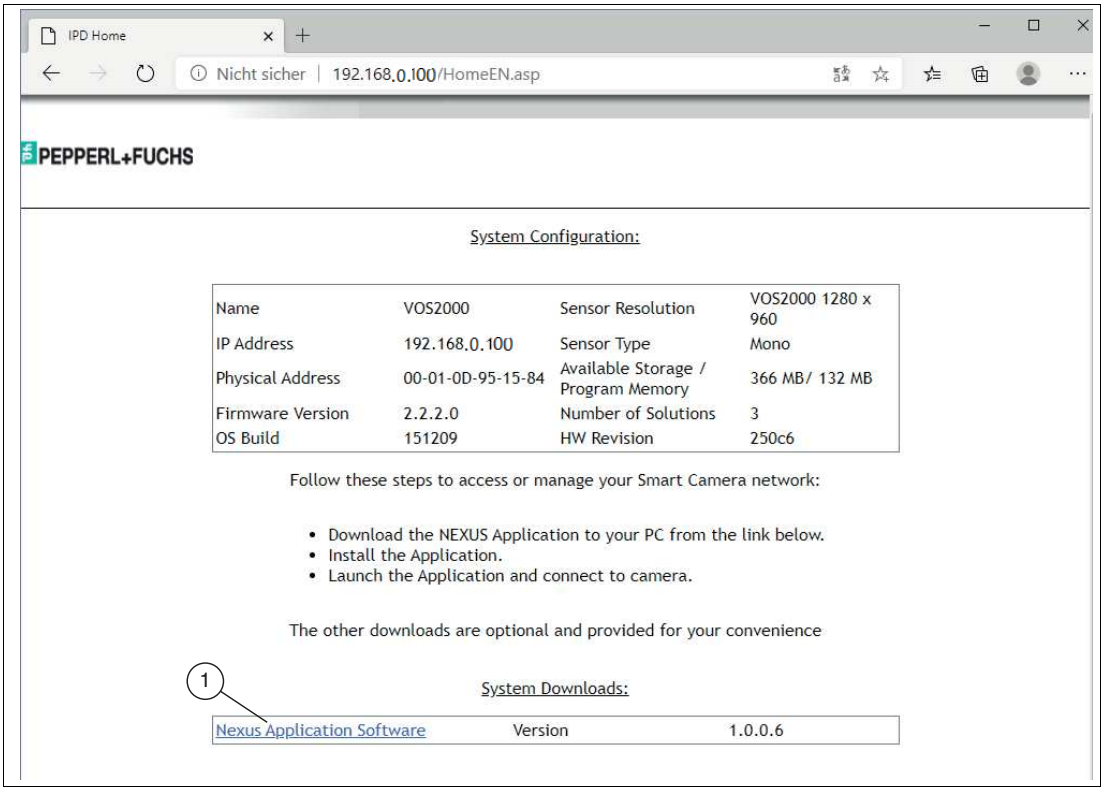
Figure 6.1 exe file

4. Follow the instructions for the installation process.

↳ After installation, several image processing functions are available in the Windows Start menu under **"Start > All programs > Pepperl+Fuchs Vision Configuration Tool"**.

Downloading and Installing the Vision Configuration Tool from the Sensor

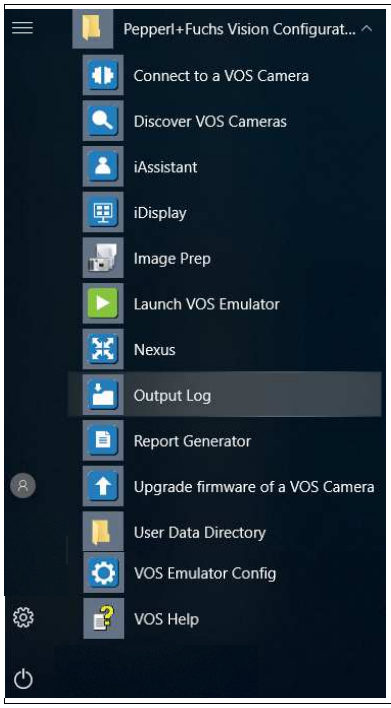
Alternatively, you can download the Vision Configuration Tool image processing application directly from the sensor. The sensor is supplied with the integrated Vision Configuration Tool image processing application. The application interface is accessed via the Ethernet connection to a PC running Microsoft® Internet Explorer 6 or higher.



1. Open a web browser and type the sensor address 192.168.0.100 in the address bar and press Enter to confirm.

   ↳ The homepage of the sensor web browser is displayed in the browser.
2. Click on the link for "Nexus Application Software" (1) and save the file locally.
3. After the download is complete, you can close the web browser.
4. Open the Downloads folder on your PC.
5. Right-click the exe file and select "Run as administrator."
6. After installation, several programs are available in the Windows Start menu under **"Start > All programs > Pepperl+Fuchs Vision Configuration Tool"**.

Programs



Program	Description
Connect to a VOS Camera	Clicking the "Connect" button opens the main Vision Configuration Tool (see chapter 7) user interface for the selected sensor. This user interface allows you to develop, load, store, or monitor applications (inspections).
Discover VOS Cameras	iDiscover is used to set the IP address of the sensor or to change the device name. You can use this link to launch Discover VOS Cameras or you can use the Nexus application to change the address of a sensor. All sensors are supplied with the same address and name. You must change the address if there is more than one sensor in the network. Changing the name is optional, but is recommended to avoid confusion.
iAssistant	iAssistant is used to clean (delete) outdated files on the PC. Nexus can manage multiple versions of the sensor firmware. If you encounter problems, you may need to clean up. You can use this link to launch iAssistant or you can use the Nexus application to launch iAssistant.
iDisplay	The iDisplay application supports the display of multiple sensors. You can view multiple active sensors in the Nexus user interface.
Image Prep	Convert JPG to BMP.
Launch VOS Emulator.	Starts the emulator.
Nexus	Launches Nexus to access the sensor and to manage the sensor. See chapter 6.3.
Output Log	This application provides the option to save results from connected sensors to CSV files on the PC.
Report Generator	Generate a solution report.
Upgrade firmware of a VOS Camera	This program provides the option to update sensors to a different firmware version.
User Data Directory	Opens the user directory

Program	Description
VOS Emulator Config	Launches the configuration window for the emulator. In the VOS emulator, you can test or demonstrate applications. For this, you need a test photo that you have taken with the sensor. The resolution of the test photo should match the resolution of the selected sensor. You can upload and test the test photo in the emulator. See chapter 6.2.
VOS Help	VOS online help

6.2 VOS Emulator

You can use the emulator to launch and evaluate test applications.

Setting the Emulator

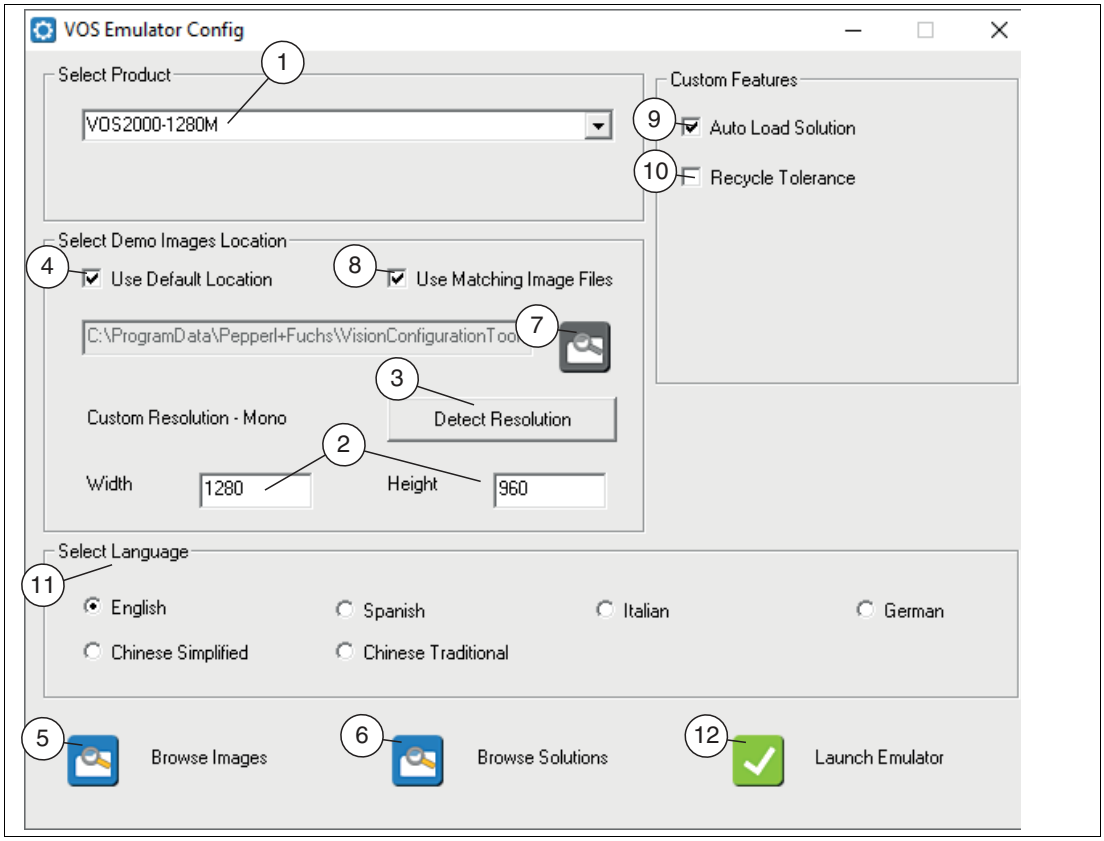


Figure 6.2 Emulator menu

1. Start the Emulator Configuration program from the Windows Start menu under **"Start > All Programs > Pepperl+Fuchs Vision Configuration Tool > VOS Emulator Config."**
2. Select your sensor from the "Select Product" drop-down list (1).

   ↳ The resolution (width and height in pixels) (2) of the selected sensor is displayed under "Custom Resolution." You can determine the resolution of the sensor using "Detect Resolution" (3).
3. Select the image source. The emulator application provides some test images. These are available if the "Use Default Location" (4) check box is checked. The application data directories can be accessed via "Browse Images" (5) or "Browse Solutions" (6). You can use other images by clearing this "Use Default Location" check box (4) and switching to another directory (7).

Note

Images must be monochrome bitmaps with the resolution of the selected sensor.

You can use images with a different resolution by clearing the "Use Matching Image Files" check box (8). Use this option with care, since jobs that were created without a matching image resolution are not transferred to the sensor.

4. Set the user-defined functions.

Note

- Auto Load Solution (9): If you check this check box, a default job is loaded at startup and the application test is started.
- Recycle Tolerance (10): If you select this check box, the test will output the state "Pass", "Recycle", or "Fail", depending on the tolerances set.

5. Select a language under "Select Language" (11).
6. Click "Launch Emulator" (12).

↳ The application test is opened in the Vision Configuration Tool.

6.3 Nexus

Nexus is the main application used to connect, set up, and manage the sensors. The application can be downloaded and installed from the Pepperl+Fuchs homepage or installed with the emulator.

The following functions are supported by Nexus:

- Finding sensors in the network
- Configuring network parameters
- Updating firmware
- Using a mix of sensors and firmware versions
- Connecting sensors for setup, processing, and runtime monitoring

Overview of the User Interface

The user interface of the Nexus application is shown below. When you first launch the application, it automatically detects connected sensors in the network (left image) and shows what it finds in the camera window (right image).

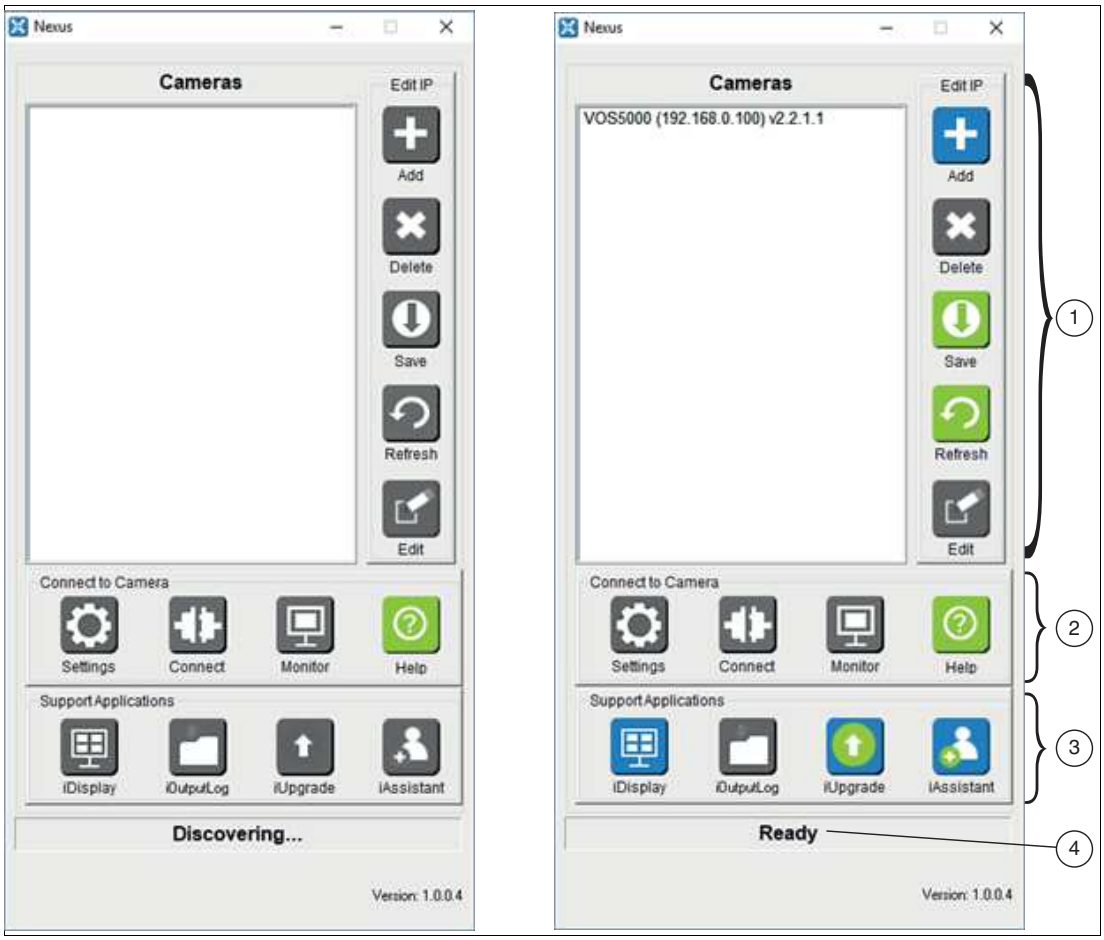


Figure 6.3 Nexus user interface

1. IP settings, configuration, and detection controls
2. Sensor settings and connection controls
3. Multisensor display and support applications
4. Activity status

User interface action buttons are gray during sensor detection, and some remain gray until a sensor is selected. Click on a sensor to activate its control buttons. Clicking a green or red button has an immediate effect; clicking a blue button requires further action.

Starting the Nexus Application

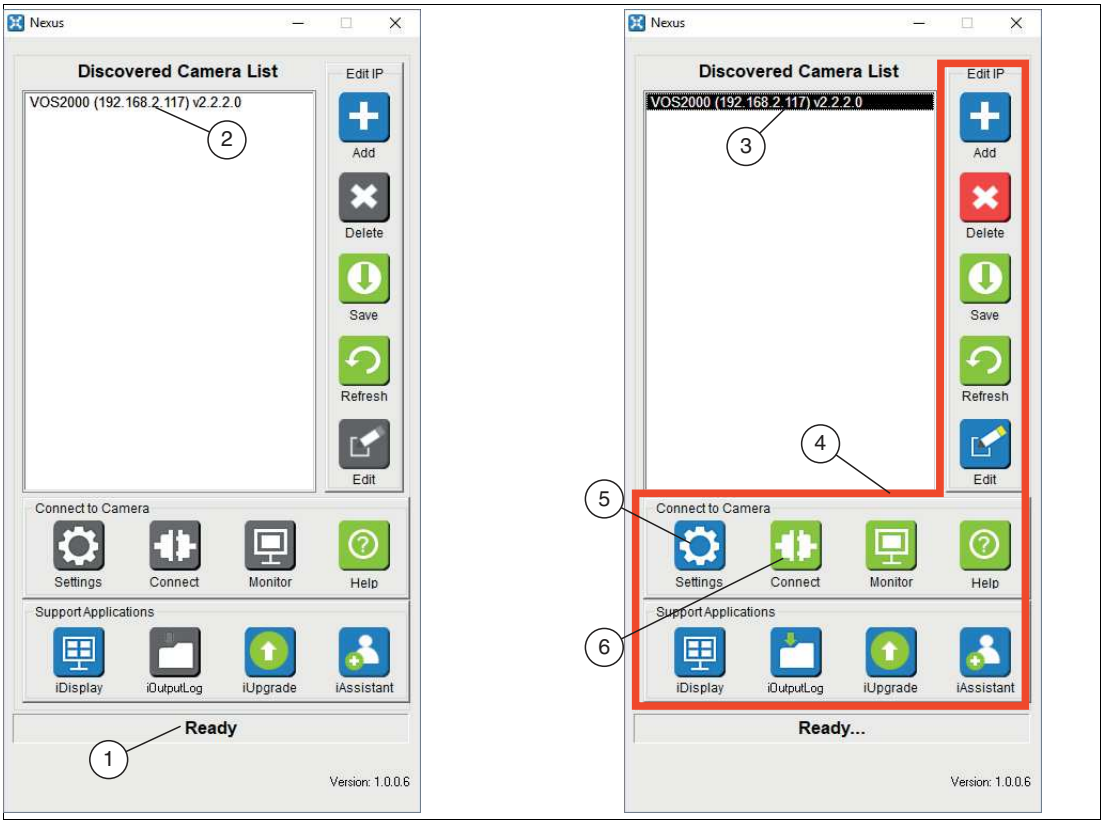


Figure 6.4 Nexus user interface

1. Open **Nexus** on your PC.

   ↳ Nexus scans the network for connected sensors. The status field (1) shows the progress. As soon as the status changes to "Ready", all connected sensors are displayed in the "Cameras" list (2).

   Note

   If an incompatible sensor is detected, all action buttons remain gray until the associated IP settings have been corrected, see "Changing the IP Address" on page 46.

2. Click on a sensor in the "Cameras" list (3).

   ↳ All buttons (4) on the user interface become active.

3. Click "Settings" (5) to adjust the camera settings.

4. Click "Connect" (6) to open the **"VOS Configuration Tool"** application and begin job programming.

Overview of IP Settings

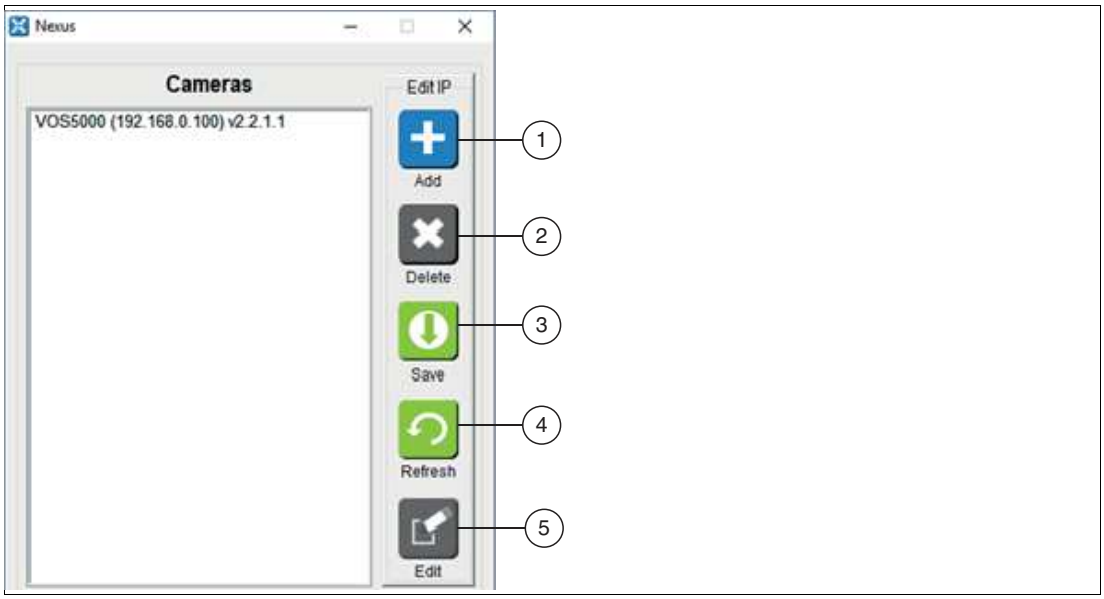


Figure 6.5 IP settings

1. Manually add sensors that were not found before the automatic search.
2. Delete the selected sensor. This sensor is not displayed when the configuration is saved.
3. Save the sensor configuration list as shown.
4. Perform a manual search.
5. Edit IP settings to correct network incompatibilities.

Note

Note regarding item 4

The camera window is updated when you perform a new manual search. Your saved configuration file is retained. The next time you start Nexus, it will only show the sensors you want to see.

If you want to delete the saved configuration file, click "Save", and the dialog box offers an option to delete it. Similarly, the dialog box allows you to overwrite the saved file.

When you update a sensor in your saved configuration list, you must manually start a new search to update the list. A new configuration file is saved. If you do not do this, you will have trouble connecting to the sensors if the update version files are not stored on the system.

Changing the IP Address

When Nexus is launched, it scans the network connections of the system in search of compatible sensors. If sensors with an incompatible IP address are found, they are highlighted in red (1). Sensors hidden behind a router may not be detected, but can be added manually using the IP addresses of the hidden sensors. To set the IP address, proceed as follows:

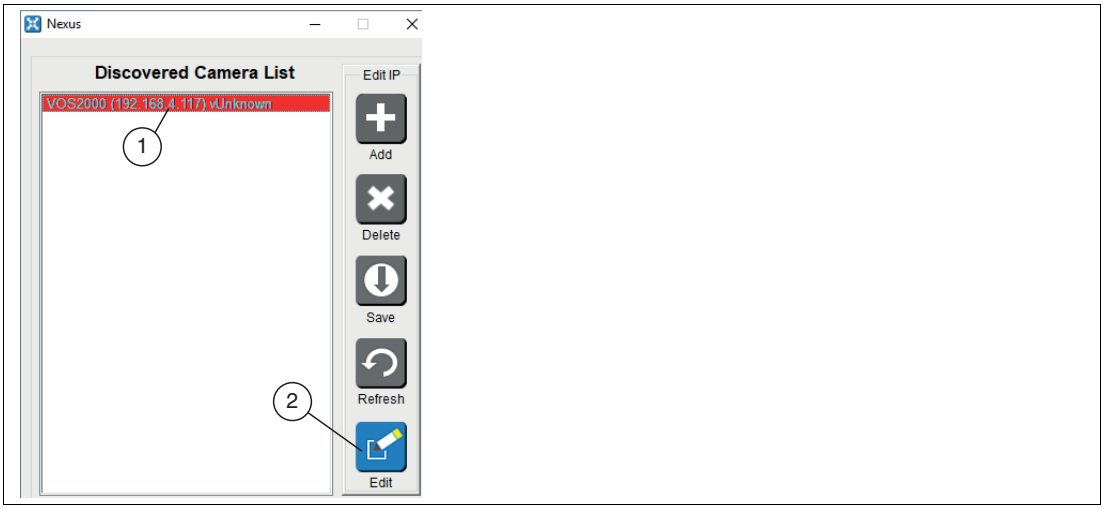


Figure 6.6 Unknown IP address

1. Select a sensor from the list (1) and click "Edit" (2).

Note

If an incompatible sensor is detected, all action buttons remain gray until the associated IP settings have been corrected. When you close the editing window, a search for sensors is started.

↳ The window for the "Resolve Camera Settings" IP settings opens.

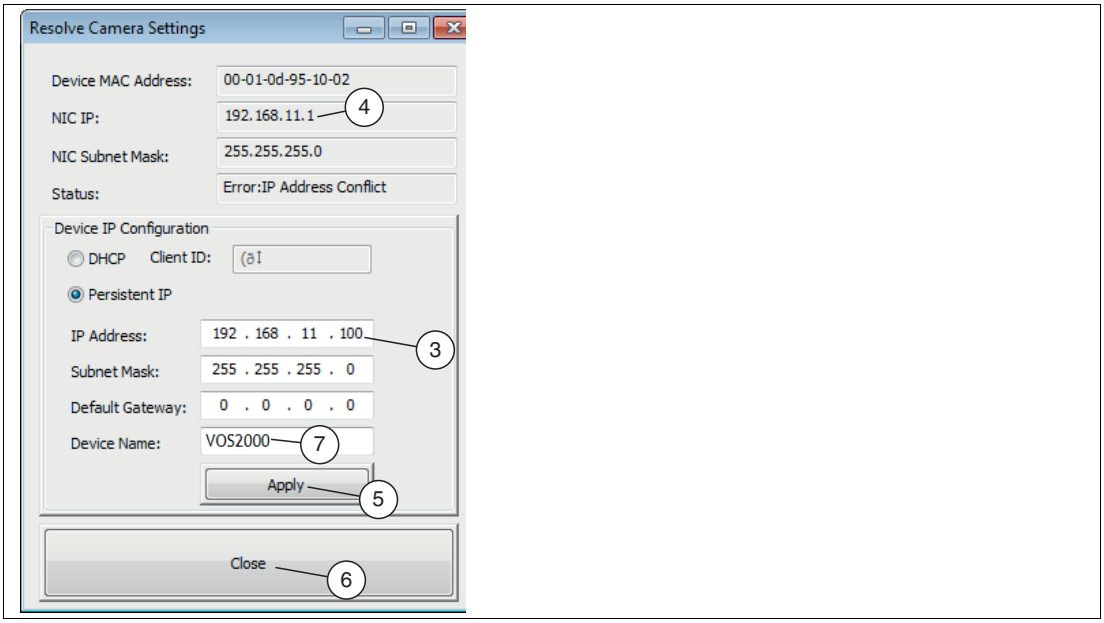


Figure 6.7 Changing the IP address

2. Change the IP address (3) of the sensor to use a free IP on the subnet of your network card (4). Then click on "Apply" (5) and "Close" (6) to save the changes.

Note

You can also change the standard device name (7) in this settings window.

↳ Nexus scans your network for connected sensors.

Note

Once the sensor is detected, you can configure which sensor should appear in your network when Nexus is launched. This is useful if you have manually added sensors through a router, or if many sensors in your network are directly or indirectly visible to multiple users. Saving a custom configuration prevents the automatic search when Nexus is launched.

Overview of Application Settings

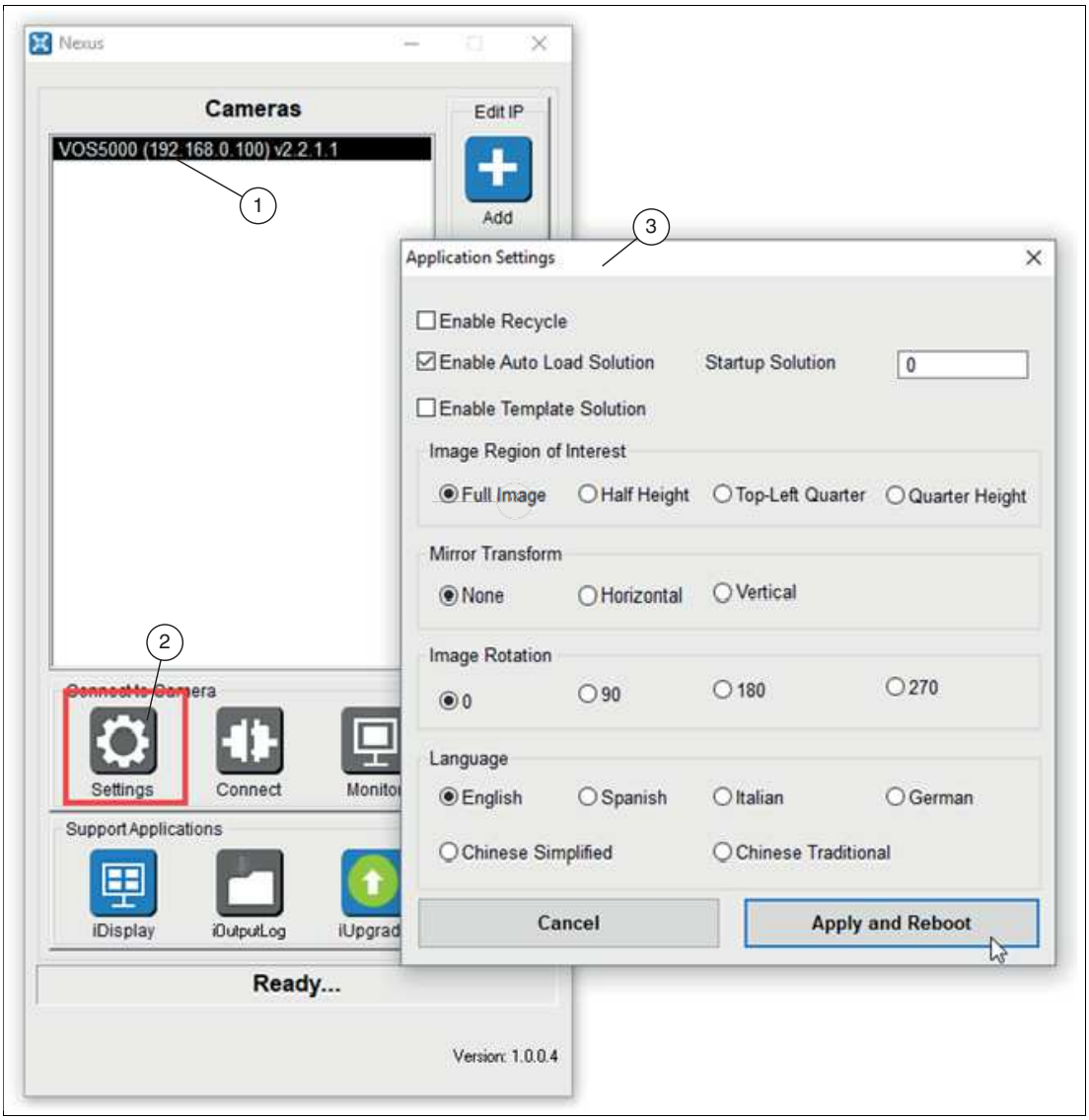


Figure 6.8 Changing sensor settings

1. List of connected sensors
2. "Settings" menu
3. "Application Settings" window

Changing Application Settings

1. In the Nexus application window, click on the desired sensor in the list and then click on "Settings."

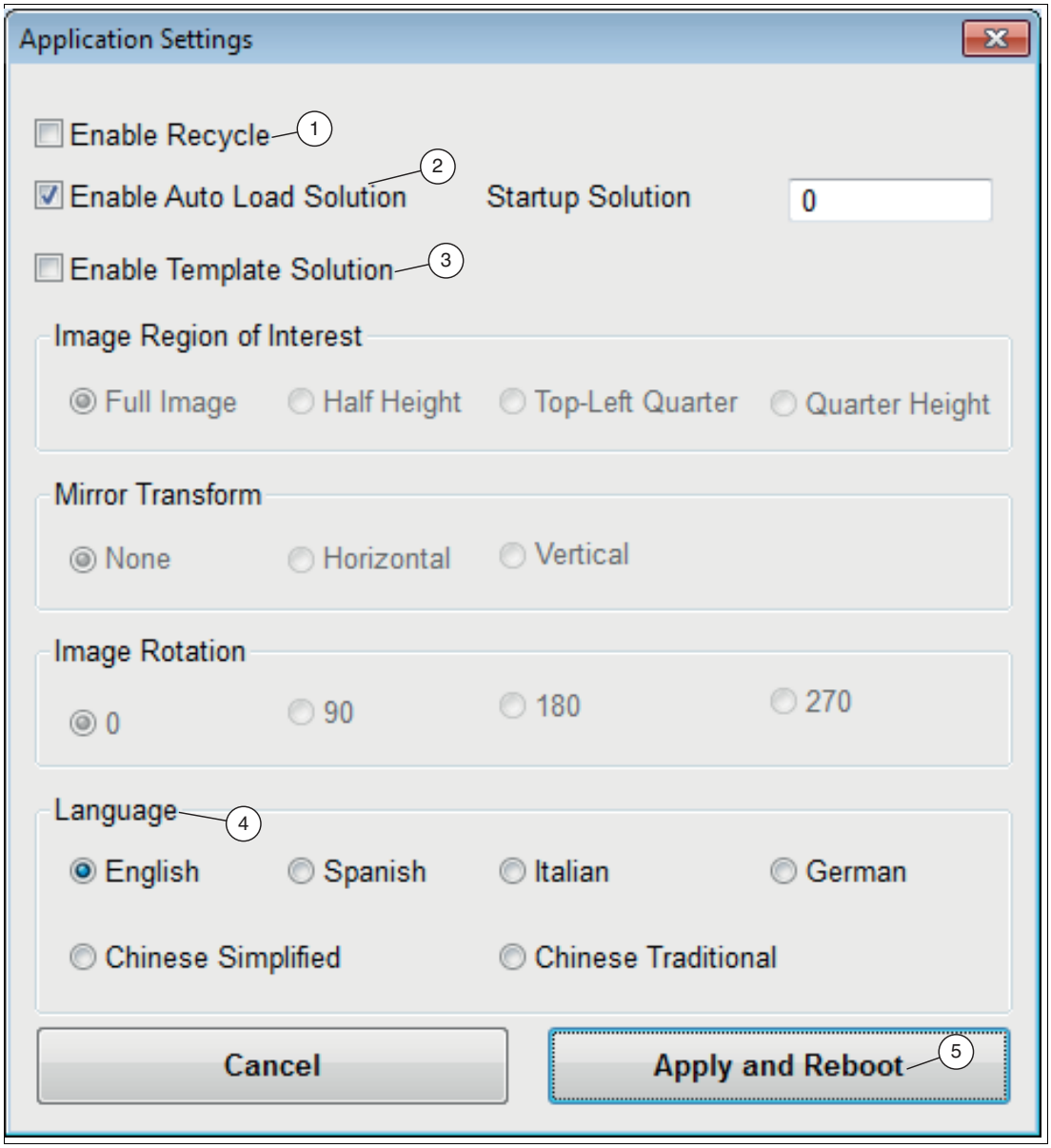


Figure 6.9 Application Settings

↳ The "Application Settings" window opens.

Note

1. You can enable or disable "Recycle" tolerances for the Vision Configuration Tool during measurements. If you clear the check box, the "Recycle" tolerances in the Vision Configuration Tool are disabled. The tolerances "perfect" and "pass/fail" remain enabled. See "Setting the Tolerance Values" on page 77.
2. Enable a previously programmed job file (Solution) when the sensor is turned on or restarted.
3. Select this option to load a previously saved job file template. The job file acts as a template for your new measurement. All settings of the saved job file (variables, scripts, and tools) are the starting point for the next measurement process.
4. You can change the display language in the Vision Configuration Tool application software.

5. When you have completed your settings, click on "Apply and Reboot" (5). The sensor is restarted.

Establishing a Sensor Connection

To connect to a sensor for setup or monitoring, proceed as follows:

Select your sensor from the list (1) and click either "Connect" button (2) or "Monitor" (3).

Figure 6.10 Establishing a sensor connection

↳ The user interface of the selected sensor is started. If there is no application running on the sensor, the monitoring application will not start. If you use multiple VOS sensors, the user interface associated with the selected sensor type is started.

Note

- Clicking the "Connect" button opens the main Vision Configuration Tool (see chapter 7) user interface for the selected sensor. This user interface allows you to develop, load, store, or monitor applications (inspections).
- We recommend that you do not run the main user interface during operation if your application has demanding processing or speed requirements.
- You can open connections to multiple sensors by selecting each sensor from the list and clicking "Connect" or "Monitor" one at a time. Note, however, that this increases the load on the connection.
- The iDisplay application provides a basic user interface that displays images and results. It is identical to the main user interface for setup; however, most control buttons are disabled. You can only change the update rate of the display and choose between text or graphical inspection status.

Multisensor display and support applications

Nexus provides the main application for real-time viewing, logging of inspection results, and updating of sensor firmware.

iDisplay

The iDisplay application supports the display of multiple sensors. The update rate of each sensor depends on the test speed and the processing effort of the running application. Sensors give priority to processing over display or logging.

Selecting Sensors

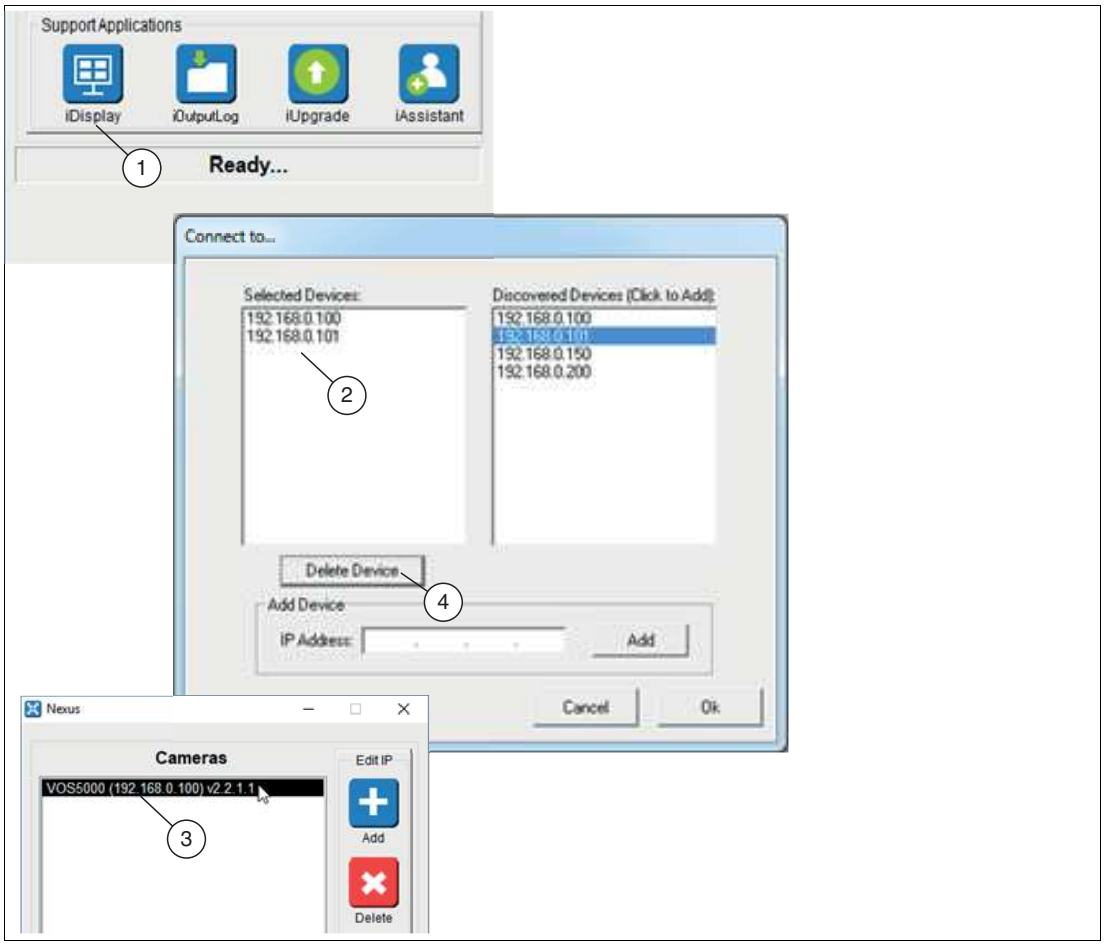


Figure 6.11 iDisplay

1. Click "iDisplay" (1).

   ↳ The application starts. When the application is first launched, a "Connect to..." configuration menu appears, in which you must select which sensors (2) are to be included for display.

2. Click on the desired sensor (2). Use the "Delete Device" button (4) to remove sensors from the selection.

   ↳ The sensor is added to the list (3).

Note

The sensor configuration is stored on the respective PC to speed up successive application launches.

iOutputLog

This application provides the option to save results from connected sensors to CSV files on the PC. The CSV files can be viewed using an editor or copied to another system and processed further there.

Creating a CSV File

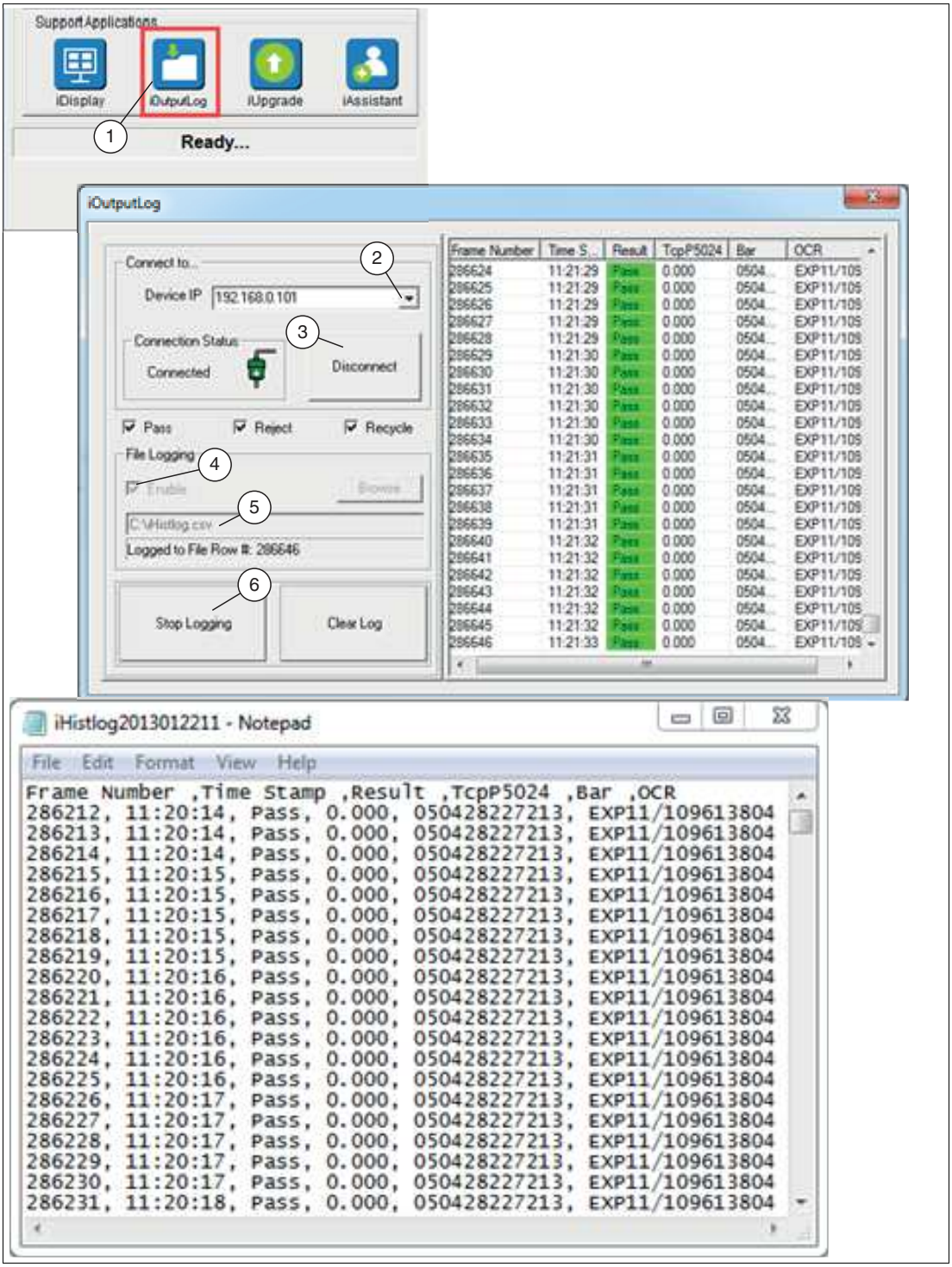


Figure 6.12 iOutputLog

2021-03

1. Click "iOutputLog" (1) to access the interface for setting up logging.
2. Select the device IP (2) from which you want to log the results.
3. Click "Connect" (3).
4. Enable "File Logging (4)" and enter a file name (5). The file name should be specific to the sensor you are connected to (e.g., VOS1000.csv).
5. Click "Start Logging" (6).

   ↳ Data recording is started.

Note

You can set up multiple logging operations one after the other. To do this, follow the steps above to create CSV files.

iUpgrade

This program provides the option to update sensors to a different firmware version. This may be necessary to access new functions, troubleshoot faults, or generally facilitate servicing. However, we do not recommend upgrading if the latest firmware meets your expectations. Although we try to ensure solution compatibility, problems may be experienced upgrading to versions with newer functions.

Note

After each update, make sure that your settings are working properly.

Note

Please execute any safety-critical updates recommended by Pepperl+Fuchs.

Updating the Firmware Version

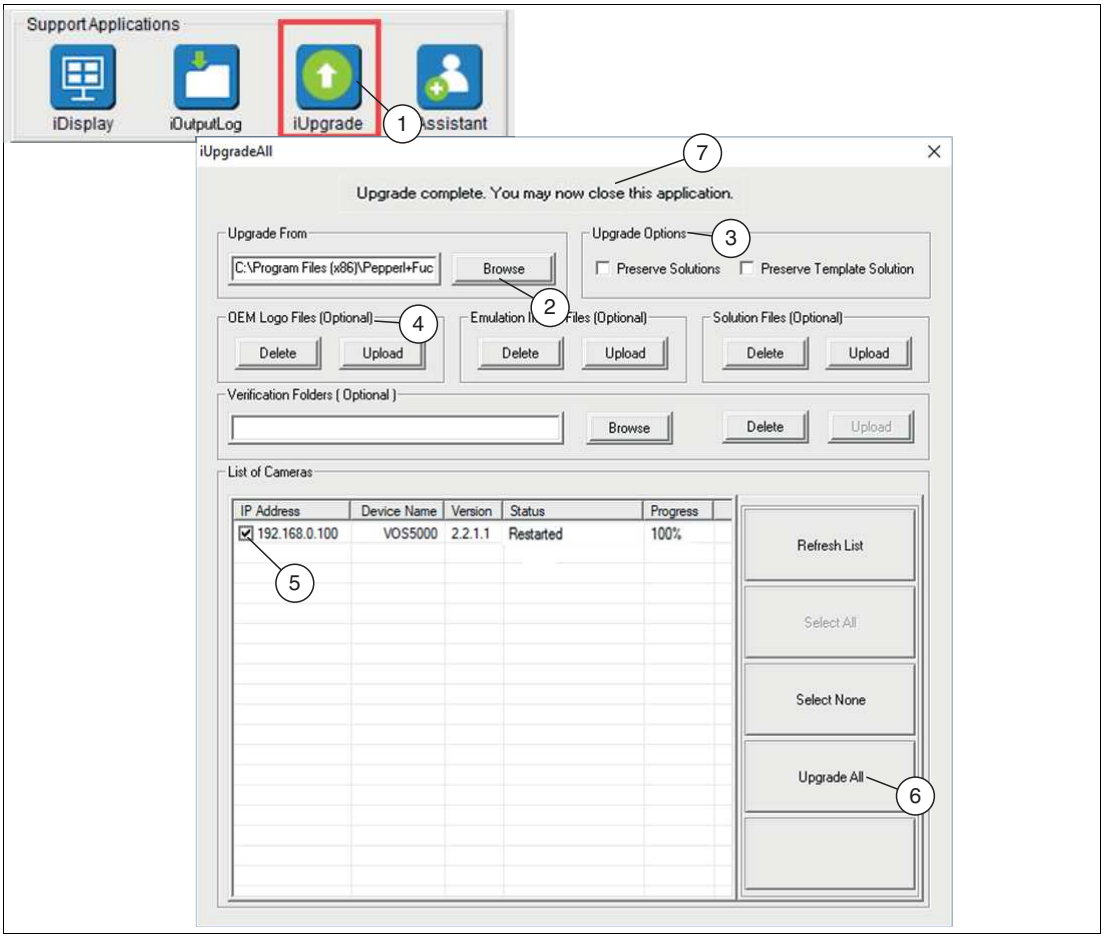


Figure 6.13 iUpgrade

1. Open the upgrade user interface using "iUpgrade" (1).
2. Click "Browse" (2) to access the firmware upgrade file from your PC.

3. Select one of the two boxes (3) to ensure that existing job files of the sensors to be updated are retained.
4. Use "Upload" (4) to install a custom logo on the Vision Configuration Tool screen.

Note

The logo must be a 24 bit bitmap file with a size of 200 x 52 pixels.

5. Select the sensors you want to update from the drop-down list (5).
6. Click "Upgrade All" (6) to update all selected sensors.
7. After the successful update, close the window by clicking the "X" in the top right corner.

Note

The iUpgrade program is not integrated in the Nexus application. As a result, Nexus does not know when an update is complete. When you exit the program, Nexus waits 45 seconds for the updated sensor to complete its restart and then updates the device list.

iAssistant

Nexus manages the software version differences between the sensors connected to the network. In some cases, such as when the network configuration expands over time, Nexus automatically retrieves the files needed to connect to the sensor.

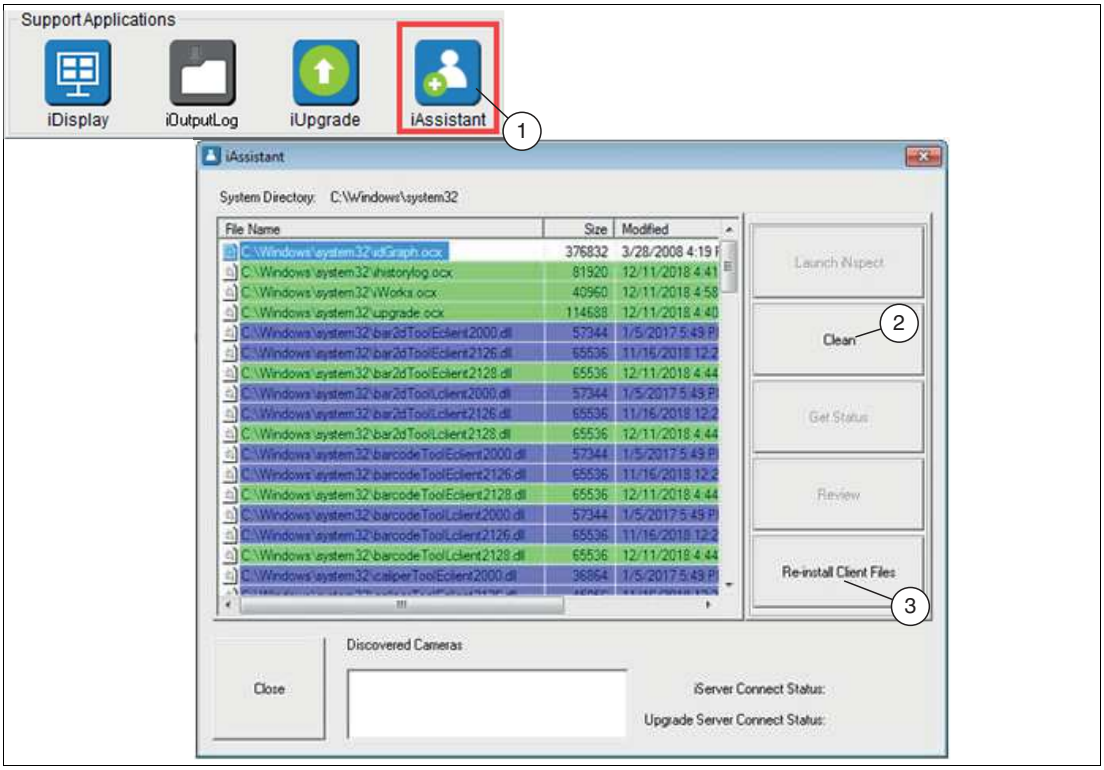


Figure 6.14 iAssistant

iAssistant (1) scans and reports device files on the PC that were installed when sensors were connected. In some cases, device files must be removed if there are incompatibilities between sensor software versions. This would be indicated by an "incompatible version" message when attempting to connect to the sensor. In this case, the system directory can be deleted from the files of the incompatible versions by clicking "Delete" (2) in the iAssistant user interface. New compatible files are installed when you reconnect to the sensor or by clicking "Re-install Client Files" (3).

- Unmarked: Indicates non-critical or unversioned files.
- Highlighted in green: Indicates that the files are the latest version found by Nexus. No action is required. Click "Close."
- Highlighted in blue: Indicates that Nexus has downloaded some files from a previous version to ensure compatibility. These are usually tool-specific files. No action is required. Click "Close."
- Highlighted in red: Indicates that important files are out of date. The Vision Configuration Tool may not work properly. In this case, the following steps must be performed:
  - Click "Clean" (2) to remove all files.
  - Then click "Re-install Client Files" (3).
  - Wait for the list to populate again before closing it.
  - Click "Close."

2021-03

Note

If you receive an error message indicating that not all files are deleted, check that your web browser and all sensor-related applications are closed. If you continue to receive errors, restart both your PC and the sensor and run iAssistant again.

7 Vision Configuration Tool

Once you have installed and powered the sensor, you can use the Vision Configuration Tool to access the sensor from a PC via a network connection, allowing you to then program and control the sensor.

Selecting the Sensor and Starting the Vision Configuration Tool

You will need a PC running the Nexus application. The sensor is connected via a local network or by direct connection via an Ethernet cable.

1. Start the Nexus software on your PC.

   ↳ A search is carried out for connected sensors and these are displayed on the Nexus user interface under "Cameras."

2. Select your sensor from the "Cameras" list and click on the "Connect" button.

   ↳ The Vision Configuration Tool starts.

Introduction to the Vision Configuration Tool User Interface

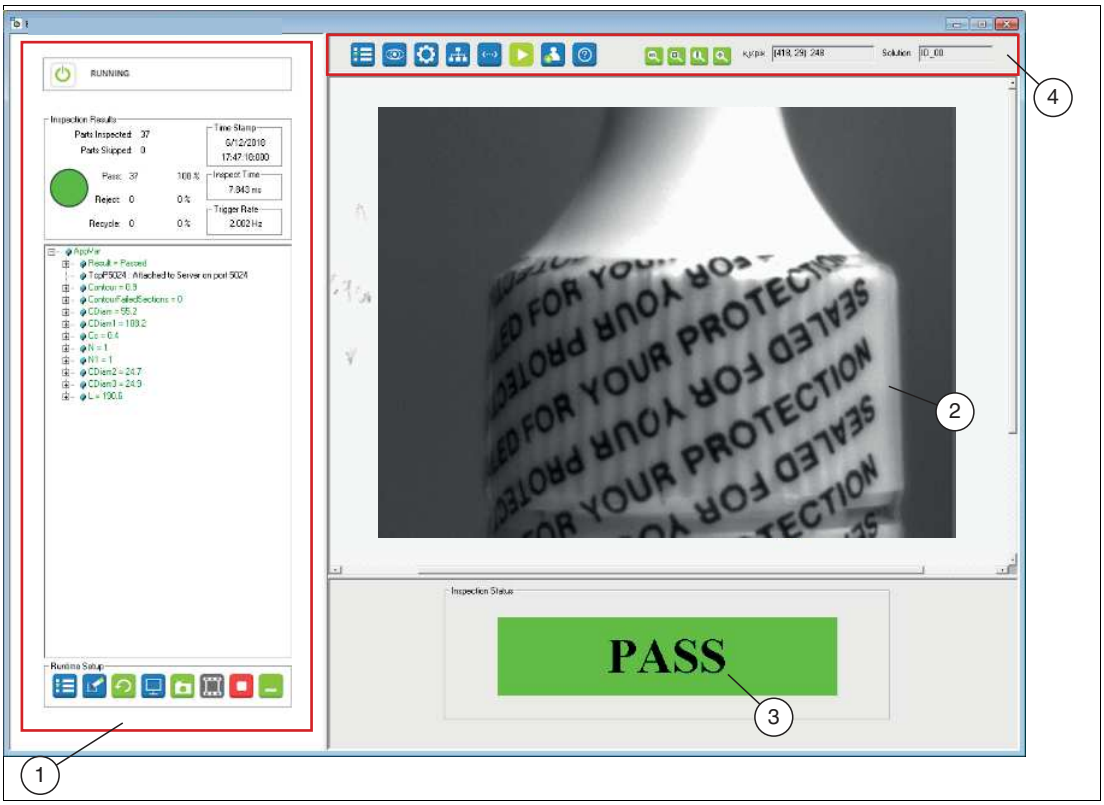


Figure 7.1 Vision Configuration Tool

The user interface of the Vision Configuration Tool consists of four areas. The left-hand side of the window (1) is used for instruction and setup. The large right-hand side of the window (2) is used to display the image and drawing tools. The lower right window range (3) is used for configuration and status display. The upper right window range (4) is the navigation bar. You can change the size of the windows.

Note

When setting up a sensor, the first five functions in the navigation bar—from left to right—are set. The process begins with creating a job, followed by setting the sensor, selecting the appropriate tools and finally setting up the connection and editing the script. The test sequence can then be started.

Navigation Bar

Button	Description
	**Create Job**: create a new job, load a saved job, import a job. Save, export, or delete a job file.
	**Sensor setup**: Select the trigger source, activate the flash control, exposure, triggering.
	**Select and apply tool**: create control tools. Set good/bad tolerances for the check.
	**Set up connections**: create and configure a PLC connection, TCP/IP, or RS-232 data stream settings. Set up I/O and image logging.
	**Edit script**: write customized programming scripts to expand solution functionality.
	**Run job**: opens the dialog box for the test sequence. Shows the test image with tools, good/bad status, statistics, and variables.
	**User management**: create and manage user accounts and passwords. Block users from editing functions.
	**Help**: opens the online help.
	**Zoom out**: reduces the size of the image.
	**Drag to fit**: zooms in or out to fill the current image display area.
	**Reset zoom**: restores the original image size.
	**Zoom in**: increases the size of the image.
	**Mouse pointer**: displays the mouse cursor position in X and Y coordinates and the pixel value.
	**Job ID**: displays the ID number and name of the currently loaded job.
	Closes the Vision Configuration Tool application window. The sensor continues to run.

7.1 Job Settings

Job files are configured and stored on the sensor memory. You can store up to 32 job files in the sensor memory. The actual number of jobs you can store depends on the image size and the application complexity (number of measurements, scripting, communication, etc.). You can export and import job files on your PC.

Overview of Job Settings

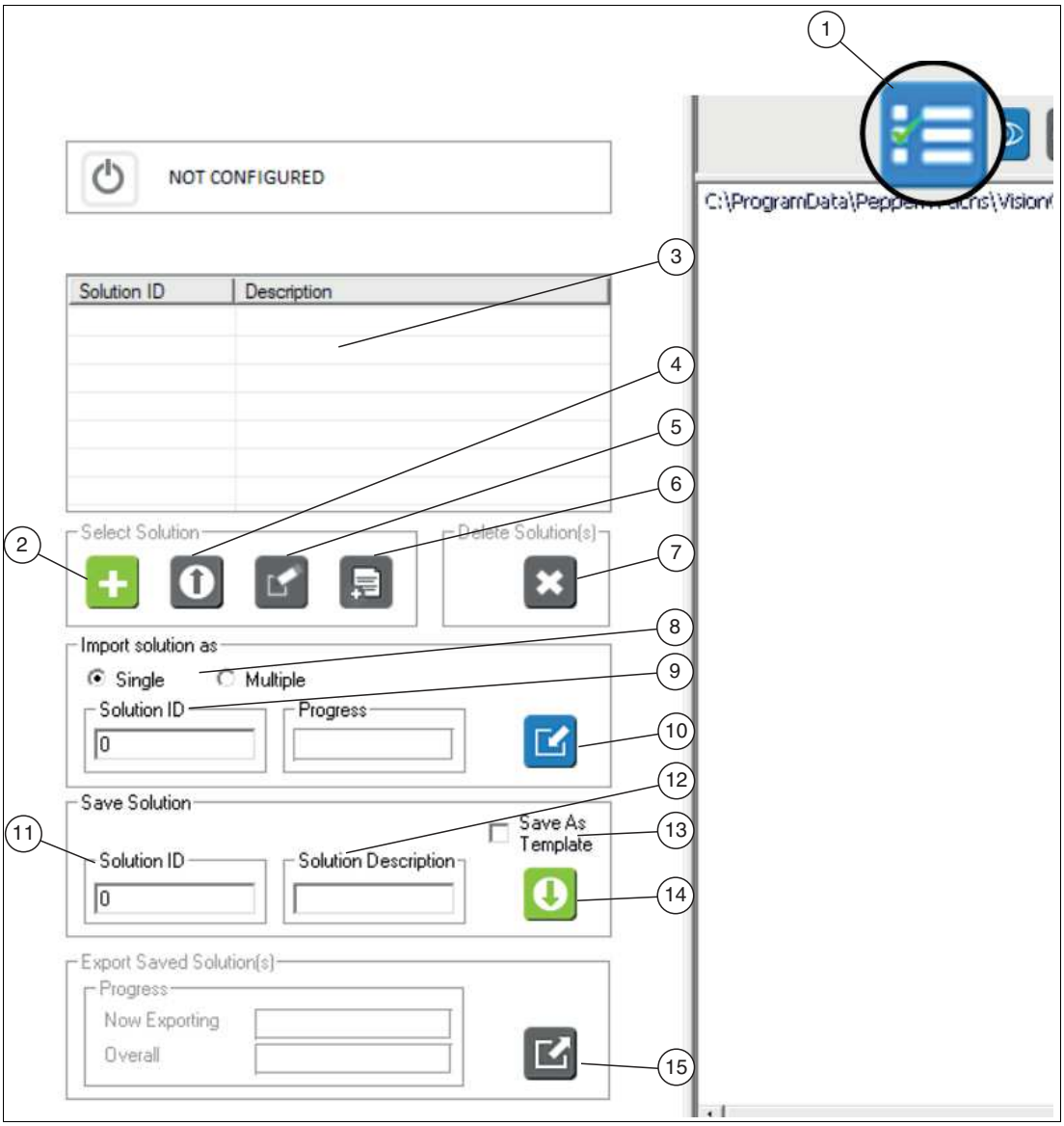


Figure 7.2 Job settings

Item	Function	Description
1	Solution Setup	Job menu
2	Start New Solution	Loads a saved sample job or resets the sensor to the default state or "unconfigured" state. All parameters are reset, except for the sensor settings. Saved jobs are not deleted from the sensor memory. Note: The "sample job" must be enabled in the "Application Settings" menu, which is launched from the Nexus application (see "Changing Application Settings" on page 49).
3	List of Solutions	A job will appear in this list once created and saved. In the "Solution ID" column, there are check boxes next to the ID number. Use these check boxes to select one or more jobs.
4	Load Solution	Loads the selected job from the list as the current or in-progress job.
5	Load for Edit / Disable Switching	Disables all mechanisms for switching job files (except this operating panel). This is required for editing jobs that use automatic job switching based on variables or inputs. The sensor will not switch a job while you are editing it. You can re-enable switching by saving the edited job file.
6	Generate Report	Generates an HTML file that documents the sensor type, measurement tools and tolerances, variables, and scripting in the one job selected in the list of saved jobs. The job must be saved before the report is generated.
7	Delete	Use the list to select one or more jobs. Click "Delete" to delete all selected jobs.
8	Import Solution(s)	• Single: If you select the "Single" option, you can enter a number to assign to the imported job. • Multiple: If you select the "Multiple" option, you can navigate to a directory and select more than one job file. You cannot assign new job numbers. Only file names with the pattern "Solution##.bin" are displayed. Use the "Control" and "Shift" keys to generate multiple selected files.
9	Solution ID	Use this field to enter a number from 0 to 31 for a single job when the "Single" option is checked. This number becomes an index in the sensor memory. Enter a Job ID number before clicking "Load Solution."
10	Import Solution	This function loads previously saved jobs from the connected PC to the sensor memory. A Windows menu for opening files will appear. Click a single job or use the "Control" key to select multiple jobs.
11	Solution ID	Enter a number from 0 to 31 to be the job ID. This becomes an index in the sensor memory.
12	Solution Description	Give each job a description that you will recognize.
13	Save As Template	Saves the current job as "Template.bin." This file is loaded when you click "Start New Solution" if "Template Solution Enable" is enabled in the "Application Settings" menu, which is launched from the Nexus application (see "Changing Application Settings" on page 49).
14	Save Solution	Use "Save Solution" to save your job to the sensor memory.
15	Export	The "Export" function allows you to save one or more jobs from the sensor memory on your PC. You must first save your job to the sensor memory before you can export it. Use the check boxes in the job list to select which jobs are exported to the PC.

Starting a New Job

The following steps show how to create one or more jobs.

1. Click the "Solution Setup" button (1) in the navigation bar.

   ↳ The left side of the editing window displays the controls to select an existing job file or start a new job. If there are no saved job files, only the "Start New Solution" button (2) is active.

2. Click the "Start New Solution" button (2).

   ↳ If an existing job is running, this action deletes or overrides the current job. The software deletes any job that is in progress and launches the Vision Configuration Tool with the default settings. The sensor address, trigger, and exposure settings are not changed.

   ↳ When you click "Start New Solution", some of the buttons in the navigation bar are disabled. You must complete the sensor setup before the other buttons can be used. If you load a previously saved job, all buttons are enabled.

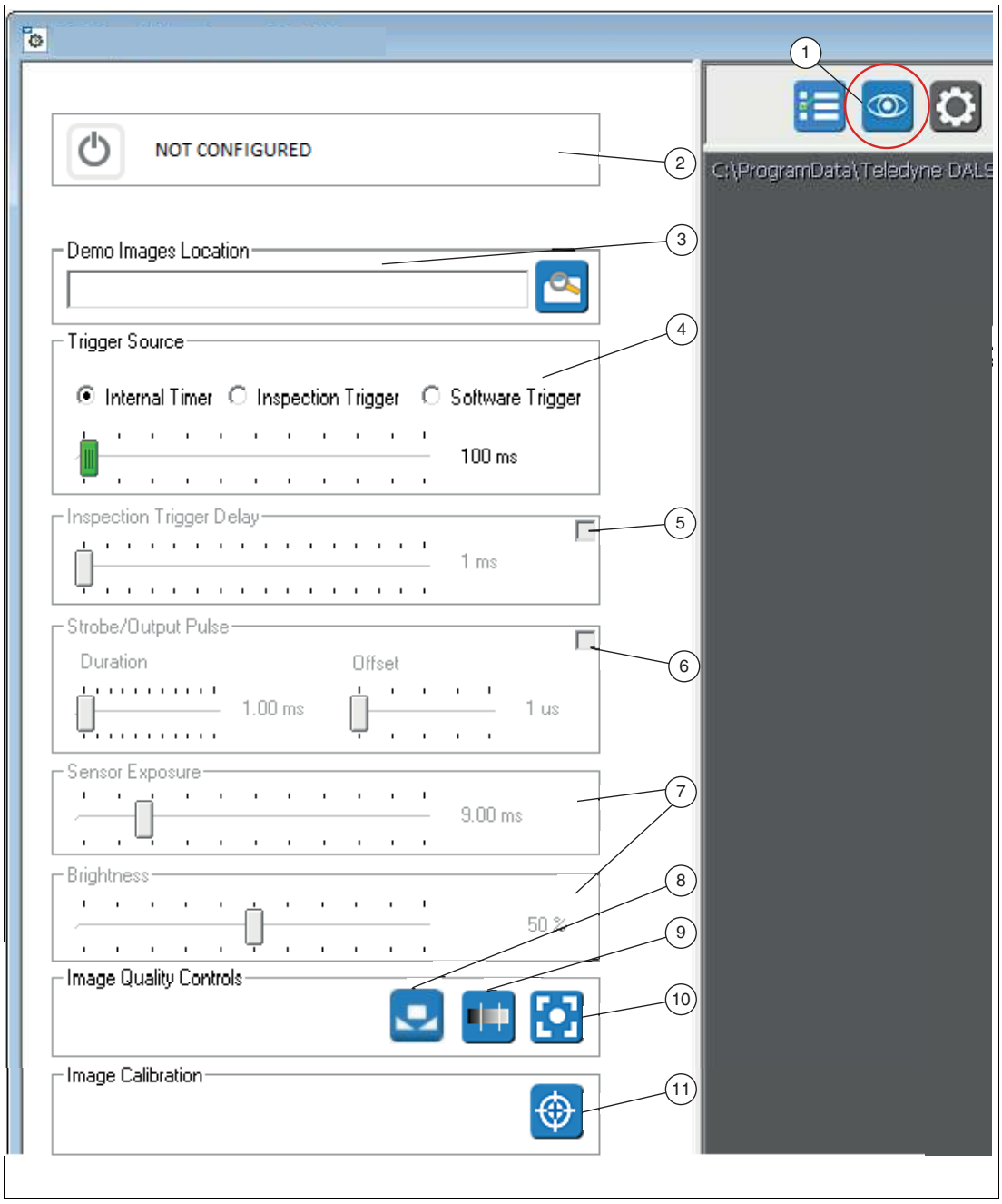
Note

Editing scripts and measurement tools changes the operating mode and disables the job change. Editing scripts can interrupt normal communication with PLCs and peripherals. Save your job and then reload it if you experience problems. Loading a job resets the operating mode unless you have selected the "Load for Edit/Disable Switching" option.

7.2 Sensor Settings

Once you have connected the sensor, you can use the Vision Configuration Tool to access the sensor from a PC via a network connection, allowing you to program and control the sensor.

Setting up a Sensor



1. Sensor setup
2. Current status or execution mode
3. Location of the image file in emulator or demo mode
4. Triggers
   - Internal timer: The slider adjusts the time between images
   - Test trigger: an external signal
   - Software trigger: a software command or a script function

5. Inspection trigger delay: Click the box to enable the slider. You can use the slider to set a delay between the time the object is under the trigger sensor and the time the object is under the camera.
6. Flash light/output pulse: Click the box to enable the sliders to define a pulse output to control a light source.
7. Sensor exposure and brightness: Use these sliders to adjust the exposure and brightness to enhance the image.
8. Image quality controls: White balance or color balance of the image.
9. Image quality controls: View the dynamic range (brightness range) of the image.
10. Image quality controls: View the sharpness quality of the image.
11. Calibrate the measurement.

Click the "Sensor Setup" button (1) in the navigation bar to access the "Set up sensor" window.

↳ The instruction and setup window changes to display the sensor menu. Here you can set the trigger and sensor image settings. If you are using a conveyor belt or other moving parts, adjust the trigger delay, exposure, and brightness with moving parts.

Note

Images that appear dark to the human eye still contain all image edge information.

Triggers

- **Using the internal timer**
  If you are using the internal timer and are processing a complex application, you may need to increase the trigger source slider to avoid skipping parts. In some cases, it is advisable to start with longer intervals during setup, such as 100 ms, and then reduce it when the application is complete.
  First, save your job with the internal timer set to the longer time interval, then go back and try to set the time to a smaller interval. Note the skipped part count on the display. If your time interval is too small, parts or images may be skipped.
  Run a few trials using bad parts to see if your time will be significantly extended if the part or feature is not found, especially in the case of matching, 2-D codes, barcodes, and OCR tools. You may want to change the time interval in these tools. Skipped parts are considered errors. An error pulse is output.
- **Using the external trigger**
  If using a presence detection sensor and your part does not appear in the image area, you may need to use the test trigger delay. This allows you to compensate for small differences between the time the part is under the presence detection sensor and the time when the part is actually in front of the camera lens. If the image of a non-moving part is sharp and the image of a moving part is blurry, you must use the exposure control of the sensor to reduce the exposure time. This should improve image quality.
- **Using the exposure control**
  If the image of a non-moving part is sharp and the image of a moving part is blurry, use the sensor exposure control to reduce the exposure time. This can improve the image quality. The sensor exposure setting indicates the approximate exposure time. The slider range changes according to the programmable area of the camera. For quick-moving parts, you must have a very low exposure time and a very bright light source. The time between the external trigger (trigger rate) or the internal timer setting (frame rate) must be greater than the exposure time plus the image capture time (image time) plus the trigger delay. The exposure setting overrides the internal timer setting if this condition is not met.
  (Trigger rate or internal timer) > (exposure + image time + trigger delay)

Using the Dynamic Range

The dynamic range is expressed in number of exposure levels (light values) and shows the range between the brightest and darkest points of an image.

Figure 7.3 Dynamic Range

In the "Set up sensor" menu, you can reach the configuration window by clicking on the "Dynamic range" button. The green bars indicate the range of pixel values in your image. The goal is to set t0 for the lens opening, brightness, and exposure to give you the widest dynamic range. Click on the "Dynamic range" button again to return to the "Set up sensor" menu.

Using the Focus Quality

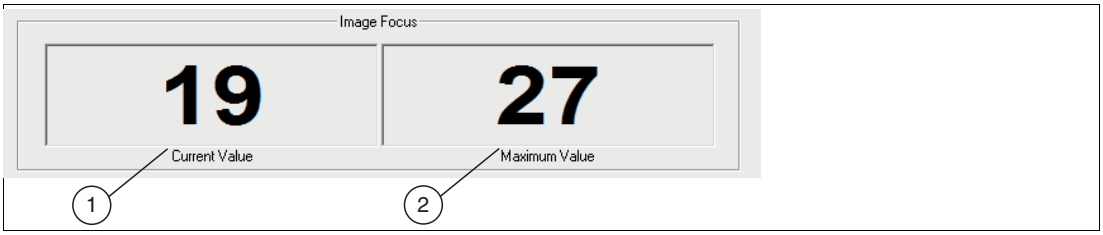


Figure 7.4 Focus quality

From the "Set up sensor" menu, click on the "Image Focus" button to access the configuration window.

The number on the left is the current focus quality (1) and the number on the right is the last maximum quality achieved (2). Adjust the lens to obtain the highest value.

Click on the "Image Focus" button again to return to the "Set up sensor" menu.

Note

Numbers above 300 indicate overexposure (mono or color) or excessive infrared (IR) light (color only).

Using the Calibration Tools

From the "Set up sensor" menu, click the "Calibration tools" button to access the configuration window.

Calibration is fundamentally not required. Two types of calibration are available: spatial calibration and color calibration.

Note

The calibration window works with a frozen snapshot image. Click the "Take photo" or "Trigger snapshot" button to take a new photo. If you need to adjust the focus, click the "Back to sensor setup" button in the setup window to get a live image.

Lens Correction

Lens correction compensates for perspective distortion caused by camera angle or lens distortion. You can perform a lens correction and orientation at the same time. The lens correction is applied to images prior to orientation.

Performing a Lens Correction

1. Place a grid of dots and/or lines or a checkerboard pattern of alternating black and white squares in front of the camera. Note that the distance corresponds to the distance of the test object.

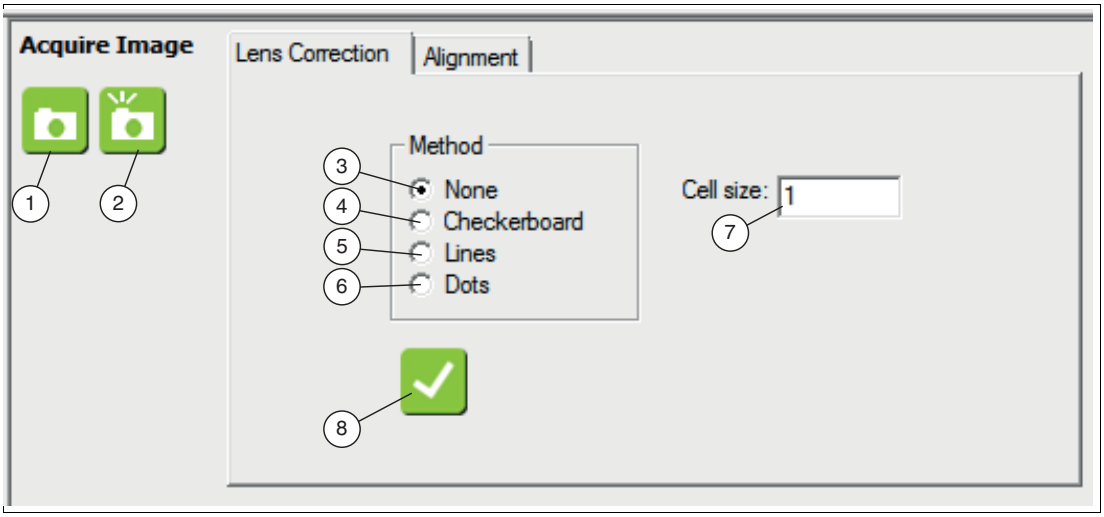


Figure 7.5 Lens Correction

2. Click the "Take photo" (1) or "Trigger snapshot" (2) button to take a photo.
3. Click in the selection field next to the correct acquisition type.
   - None (3) — no correction or cancel a correction.
   - Checkerboard (4) — place a checkerboard in front of the camera.
   - Lines (5) — place a line grid in front of the camera.
   - Dots (6) — place a grid of circular dots in front of the camera.
4. Enter the cell size (7) required for your calibration type.
5. Click the "Perform calibration" button (8) to apply your scale changes. Wait for the software to detect the grid and draw a line grid on the image.

2021-03

6. When you are finished, click on the "Back to sensor setup" button in the navigation bar.

Alignment Calibration

Note

Alignment calibration is only required in conjunction with certain vision tools. Before performing alignment calibration, determine which vision tools (see chapter 7.3) you will be using.

Alignment calibration allows you to set the values and units specified in the measurements.

You can change the scale of the measurement using the "Scaling" button (ruler symbol) in the tool settings. Resetting the scale for the entire coordinate system returns to the pixel measurements. This undoes the spatial calibration.

Performing an Orientation

Note that all tolerance values in the tool settings are unlocked when you set the orientation.

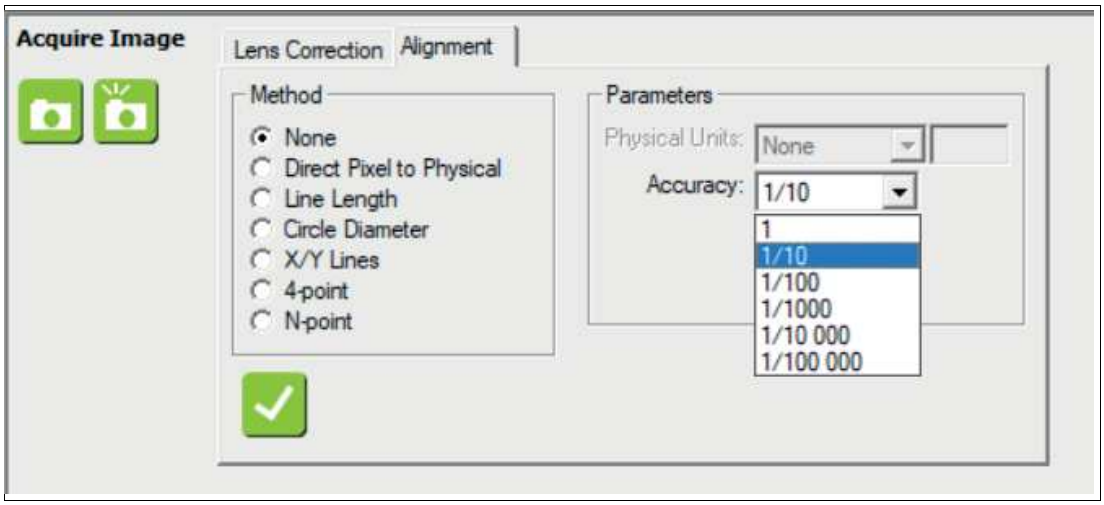


Figure 7.6 Orientation

1. Place your object in front of the camera.
2. Click the "Take photo" or "Trigger snapshot" button to take a photo.
3. Click on the selection field next to the right calibration method. Adjust the size of the lines or circles that appear in the image.

   ↳ The number of input fields under "Physical Units" changes with the selection of the calibration method. Select the physical units you want to see in the image display.
4. Under "Accuracy", select the number of decimal places to display and report.
5. Enter the scale multipliers or cell size required for your calibration type.
6. Click the "Perform calibration" button to apply your scale changes. For the "Distorted by..." options, wait until the software detects the grid and draws a line grid on the image. If you select "4-point", wait for the software to draw a large rectangle that connects the 4 dots.
7. When you are finished, click the "Back to sensor setup" button in the settings window.

Alignment Methods

Method	Description
None	No scale change, no units are displayed. The numbers given are associated with pixels as units.
Direct Pixel to Physical	Enter X and Y multipliers (horizontal and vertical) and select a physical unit and accuracy.
Line Length	A line appears in the image area. Use the handles at each end to change the length of this line. Place the line on a reference target or target. Enter the length or size of the target object and select a physical unit and accuracy. The same scale or multiplier is applied horizontally and vertically.
Circle Diameter	A large circle appears in the image area. Use the two handles to resize this circle. Place this circle on a reference target or target. Enter the diameter of the circle or target object and select a physical unit and accuracy. The same scale or multiplier is applied horizontally and vertically.
X/Y Lines	Two lines appear in the image area: one horizontal and one vertical. Resize the two lines by using the handles at both ends. Place each line on a reference target or target. Enter the length or size of each target. Select a physical unit and accuracy.
4-point	Drag the four points that appear in the image to the four corners of a rectangular or square object that has perspective distortion. Alternatively, drag the four points to active points you created using the appropriate tools. You can zoom in or out to refine your placement. Initially, the four points appear at the outer corners of the image area. If your image is larger than the currently visible area, you will need to zoom out to see the four dots.
N-point	Left-click in the image to create a point. Right-click on the point to display a coordinate editing menu. Enter world coordinates for the point. Click the point and wait a few seconds. The image does not change.

Alignment Parameters

Parameter	Description
Physical Units	Select a unit from the drop-down list. This unit is displayed on the image during capture.
Accuracy	Select the displayed number of decimal places for measurement results.

2021-03

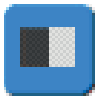
7.3 Vision Tools

This section describes how to use vision tools to solve test tasks. There are several ways to solve an image processing application. The benefit of vision tools is that you can solve complex test tasks using simple techniques. To provide you with the foundations for getting started, sample applications are described in the following sections.

The following table provides an overview of the vision tools.

Overview of All Vision Tools

Vision Tool	Symbol	Description
Match Tool		This function is used to learn and match features, check, and tool alignment. In addition, this function supports 360° orientation. See chapter 7.3.1.
Count Tool		This function is used to count or locate features. See chapter 7.3.2.
Edge Count Tool		This function is used to look for or count edge transitions, and for position determination and/or alignment. See chapter 7.3.3.
Intensity Tool		This function is used to calculate statistics and detect the presence or absence of an element. See chapter 7.3.4.
Caliper Tool		This function is used to measure distances. It also supports irregular shapes. See chapter 7.3.5.
Point Tool		This function is used to detect a single edge point for position measurement or as a position encoder for the orientation of other tools. See chapter 7.3.6.
Tip Tool		This function is used to find the "tip" of an object. See chapter 7.3.7.
Pencil Tool		This function is used to set a reference line from which further measurements are made. See chapter 7.3.8.
Distance Tool		This function is used to measure the distance between two edges. See chapter 7.3.9.
Rake Tool		This function is used to measure multiple distances and for average determination between two edges. See chapter 7.3.10.
Contour Tool		This function is used to find edges. See chapter 7.3.11.
Angle Tool		This function is used to measure an angle between two edges. See chapter 7.3.12.
Arc Tool		This function is used to measure the radius of a curve. See chapter 7.3.13.

Vision Tool	Symbol	Description
Circle Tool		This function is used to measure the diameter of a circle. See chapter 7.3.14.
Concentric Tool		This function is used to measure the center width and wall width. See chapter 7.3.15.
Graphics Tool		This function is used to add text, draw shapes, and to label and highlight tool results and functions in the image.
Preprocessing Tool		This function is used to improve specific properties of an image, e.g., edge or contrast enhancement, and convert a grayscale image into a "digital" image. See chapter 7.3.16.
1-D Code Tool [Barcode Tool]		This function is used to read 1-D codes. See chapter 7.3.17.
2-D Code Tool		This function is used to read 2-D codes. See chapter 7.3.18.
Text Recognition [OCR]		This function is used for optical text recognition. See chapter 7.3.19.
Verify Tool		The Verify Tool compares the taught-in template with an object in the image and reports how different the object is from the template. See chapter 7.3.20.
Color Meter Tool		This function is used to identify and measure colors. See chapter 7.3.21.

Using Vision Tools

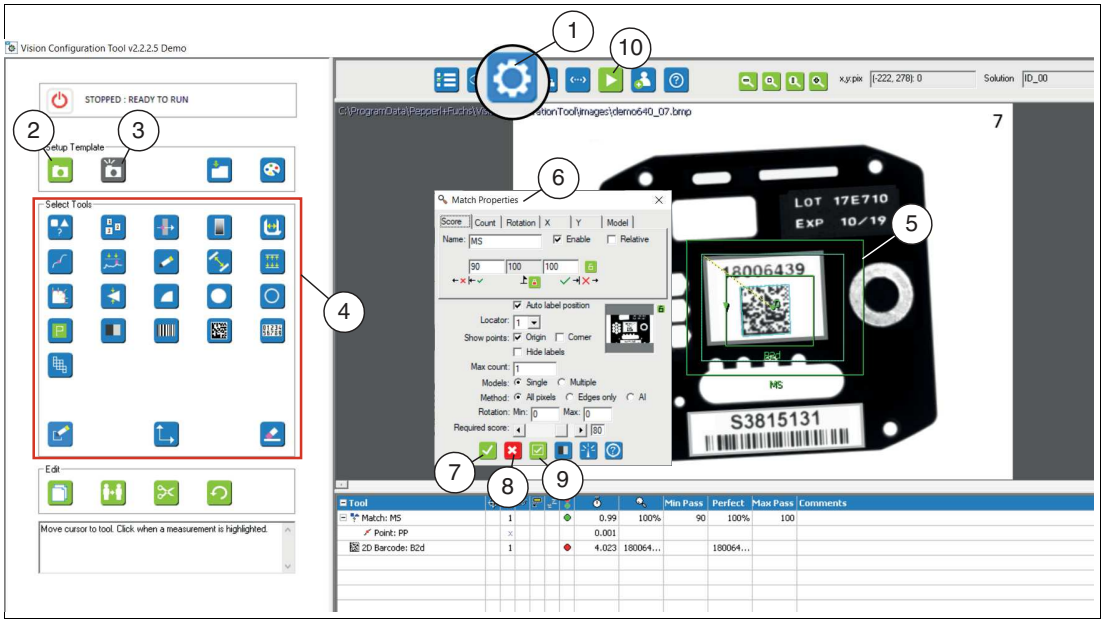


Figure 7.7 Vision Tools main menu

1. Open the "Vision Tools" menu in the toolbar (1)
2. Take a picture of the object (2) or (3). This will be your reference image of the object being tested.

Note

"Take a photo" (2) captures an instant image. Use this function when working with stationary objects.

"Trigger snapshot" (3) waits for a trigger signal or a software trigger. Use this function when working with moving objects.

3. Select the vision tools required for your check from the tool selection area (4).
4. Draw the search field in the image area (5).

Note

Moving Objects

If your test object moves, you must first create a locator point. Several tools are available for this purpose (e.g., the Match Tool: see "Creating a Locator Point" on page 84). The following tools can be used to create locators: Count Tool, Point Tool, Edge Count Tool, and Pencil Tool. Refer to the description in the online help of the Vision Configuration Tool.

5. Open the Properties menu of the tool (6) by right-clicking on the search area outline in the image area.

↳ In the Properties menu, you can set tolerances for good/bad or good/perfect/bad. The Properties menu also displays parameters for refining the tool operation or for enabling additional measurements. Some tools have additional measurements that can be enabled.

Note

Help

Each tool has a help page that describes all parameters and options.

6. Click the "OK" button (7).

   ↳ The changes are applied and the editing field closes.

7. Click the "Cancel" button (8).

   ↳ The editing field closes without any changes being made.

8. Click the "Apply" button (9).

   ↳ The changes are applied without closing the editing field.

9. Click "Run" (10) in the navigation bar when you are finished with the tool settings.

   ↳ You can check how well your tools are functioning and the required test times.

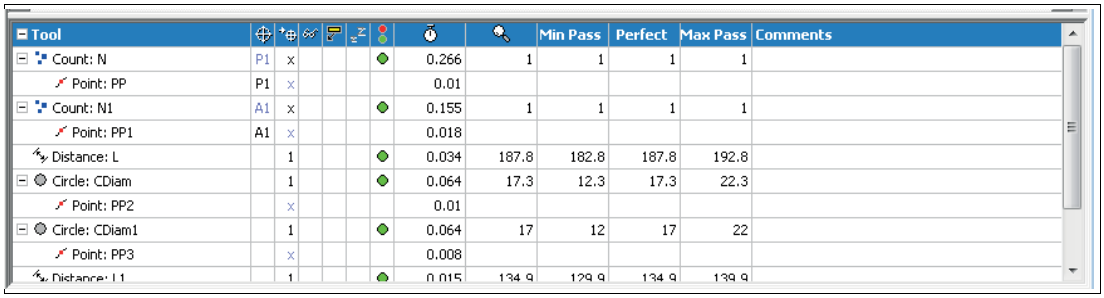
10. Click "Connections" in the navigation bar to set up communication or inputs and outputs.

11. Click "Scripts" in the navigation bar to add additional instructions or functions to your application (see chapter 7.5).

2021-03

Tool Status Table

When you apply tools in the image area, these tools appear in the tool list in the configuration and status area. You can right-click a tool's name to open the "Properties" editing menu for that tool. The columns and headings are explained in the online help. You can change the width of the columns by dragging the dividing lines in the table header. The comment box does not appear in other operating windows.



Function	Description
Tool	Controls for tool names: Right-click a tool's name or click the tool's symbol in the list to open the "Properties" edit field for that tool. Click the [-] to close the list of dependencies for a tool, or click the [+] to expand it. Click the "Tool" heading to show or hide the list. Header controls: Click the "Tool" heading to reverse the order of the list. Click the [-] to collapse the list or the [+] to expand it. Right-click the "Tool" heading to open a tool list property field in which you can filter tools by type or name. The text box is case sensitive.
 Locator Definition	Locator definition displays which tools are used to create locators. The tool and the active points are specified. "P" stands for position locator and "A" for rotation locator.
 Locator Assignment	Locator assignment indicates to which locator the tool is assigned.
 Visibility	Hides or shows tools. Click in this column to hide a tool in the image area. A dash appears and the tool is grayed out in the list to indicate that the tool is hidden. Click a second time to show/hide the tool again. Click the column header to show or hide all tools. This can be useful if you have tools that overlap and you have problems choosing the right tool. In the "Run" operating panel, a hidden tool does not appear in the image area. The measured value appears in the list of values.
 Toggle Labels	Hides or show tool labels. Click this symbol in the column header to show/hide labels for all tools. An "x" in this column indicates that a label is hidden. This can be useful if you have a lot of tools or a lot of found objects and the labels overlap the display.
 Enable/Disable	Enable or disable tools. Click in this column to disable a tool. The tool is hidden in the image area and grayed out in the tool list, and a small X in this column indicates that a tool is disabled. In the "Run" operating panel, a disabled tool is not displayed in the image area and its measured value is always zero. (You can also use a script to disable or enable a tool).
 Pass/Fail	A red circle indicates that the measurement has failed. A green circle indicates that the measurement has passed.

Function	Description
 Execute time	The estimated execution time, in milliseconds, for the individual tool.
 Measured value	The current measured value.
Min Pass	Minimum tolerance for Recycle/Pass. "Recycle" if the measured value is less than this value. "Pass" if the measured value is greater than or equal to this value.
Perfect	The set "perfect" value.
Max Pass	Maximum tolerance for Pass/Recycle. "Pass" if the measured value is less than or equal to this value. "Recycle" if the measured value is greater than this value.
Comments	You can enter comments here. The comments only appear in the tool list. The comments do not appear on other screens.

Setting the Tolerance Values

For each tool you can set tolerance values for your measurement results. Depending on the measurement result, the tool (Vision Tool) can assume the status "Pass", "Recycle", or "Fail." In the script, this result can be evaluated under the variable #VisionToolVariable.Result.

The following tolerance settings are available, depending on the particular tool:

Input Fields

Symbol	Designation	Function
	Minimum Recycle	The minimum recycle tolerance represents the smallest extended measurement tolerance. This value is located between the smallest "Recycle" value and the smallest value at which the measurement is passed (minimum pass). The "Recycle" value is reused in the next measurement. Note: The recycle tolerance is enabled or disabled in the "Application Settings" menu in the Nexus application (see "Changing Application Settings" on page 49).
	Minimum Pass	Smallest value to pass the measurement.
	Perfect	This exact value needs to be measured to pass the measurement.
	Maximum Pass	Largest value to pass the measurement.
	Maximum Recycle	The maximum recycle tolerance represents the largest extended measurement tolerance. This value is located between the largest "Recycle" value and the largest value at which the measurement is passed (maximum pass). The "Recycle" value is reused in the next measurement. Note: The recycle tolerance is enabled or disabled in the "Application Settings" menu in the Nexus application (see "Changing Application Settings" on page 49).
	Fail or Reject too small	Measurement failed, the measured value is too small.
	Fail or Reject too big	Measurement failed, the measured value is too large.
	Locking	The padlock is used to lock or unlock values, so that they do not change when you move or resize your tool's search field or customize the tool properties. A red padlock indicates that the values are locked. A green padlock indicates that the values are not locked. Some measurements are locked by default when you apply a vision tool.
	Ruler	The "Ruler" button allows you to define a new unit (scale) for the current measurement or for the entire coordinate system used.

Example

A requirement may contain information indicating that the expected measured value is 100 ± 5 and that a test object can be recycled or post-processed if the measured value is 100 ± 7.

Relative Measured Values

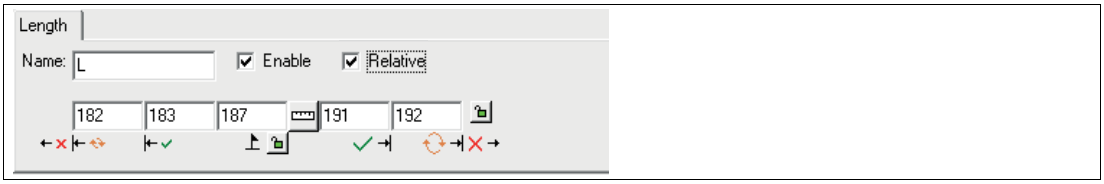


Figure 7.8 Relative measured values

Vision tools that measure a numerical value have a check box labeled "Relative." If this box is selected, the difference to perfect is reported, not the actual measured values. For example: The set value of perfect is 187 and the measured value is 189. The difference to perfect is 2. If the "Relative" check box is cleared, the measured value 189 is reported. If the "Relative" check box is selected, 2 is reported.

Note

General Tool Information (Vision Tools)

- Increasing the size of the search field increases the test time.
- While drawing a tool, you can cancel by right-clicking. This is useful if you have selected the wrong tool.
- Adding a preprocessing filter may increase the test time. See chapter 7.3.16.
- If the Point, Distance, or Rake Tool is selected, permitted edges are highlighted when you move the mouse over the image area. You can right-click any highlighted edge (or the edge of the image area) or any position in the image (but not on a tool) to change the edge threshold value. Increasing the threshold value can help to ignore background features or reduce the complexity of the image. A smaller number is more sensitive and includes more edges. A larger number is less sensitive and includes fewer borders.

7.3.1 Match Tool

The Match Tool is used to locate one or more planes of a particular pattern. The pattern can include all pixels in an area or just the edges. The Match Tool compares the selected area with the same area in the taught-in image and returns a percentage evaluation of the quality of the match. The match is useful for determining the presence/absence of objects or for creating a locator point.

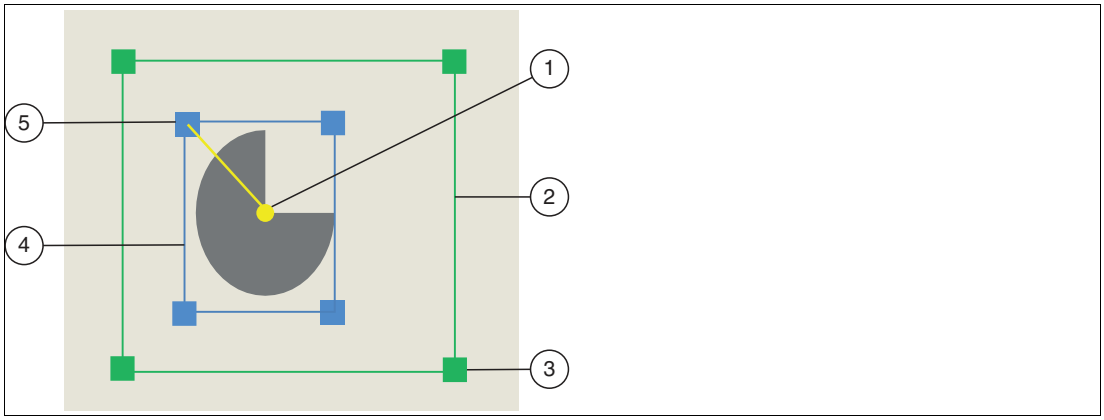


Figure 7.9 Match Tool

1. Origin of the object
2. Search area
3. Resizing the search area
4. Object area
5. Resizing the object area

Match Tool Information	
Search area shapes	Rectangular
Tool preparation	Two corner points to form a rectangle
Search search area	Resizable to cover the partial movement area
Pattern search area	Resizable to define functions within the search search area
Supported algorithm	Correlation (template match) or edge
Number of matches	User defined
Match results	Defines the accuracy of the match (up to 100 %)
Rotation	User-defined (affects the speed)
Accuracy	Partial pixels
Speed optimization	Image undersampling for large features
Measurements	Result, number (number of matches), and rotation

Using the Match Tool

1. Click the "Match Tool" button in the tool selection area.

   ↳ Move the mouse over the image in the work area. There should be no highlighted borders.

2. Left-click on your starting point. Release the left mouse button and move the mouse to the area that represents your search area. Left-click again to draw a rectangular search area on the image.

   ↳ The selected field is labeled (MS, MS1, etc.). The inner rectangle is the object area. The outer rectangle is the search area.

3. Resize the search area by clicking and dragging one of the corners of the search area. Use the curved arrow to rotate the search area. A small arrow on the outer edge of the search field indicates the direction.

4. Right-click on an edge of the rectangular search area.

   ↳ The properties window of the Match Tool will open.

Match Tool Properties Window

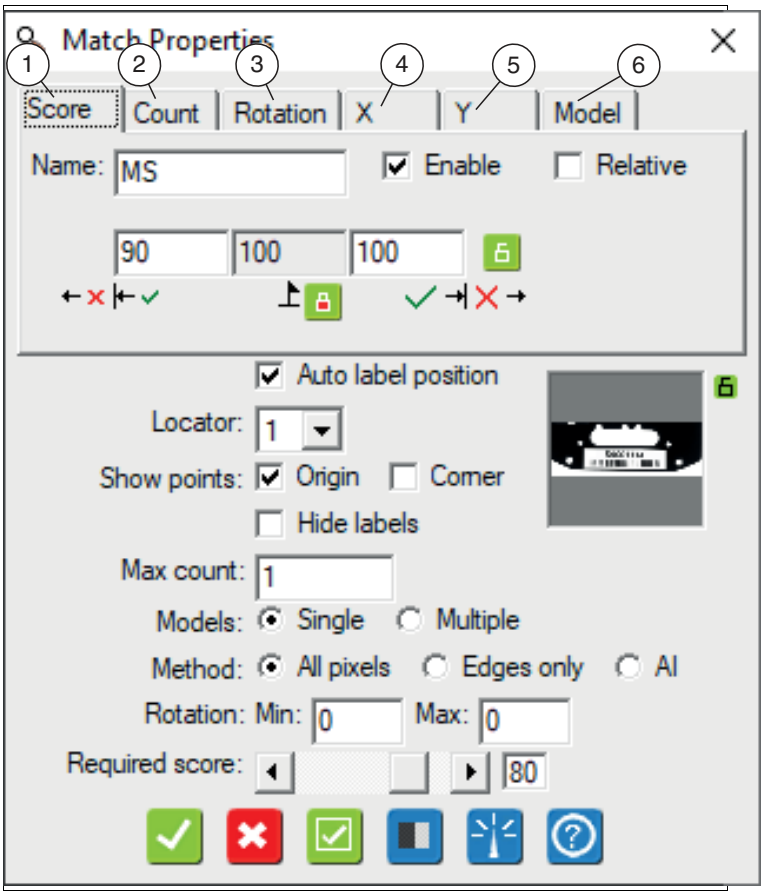


Figure 7.10 Properties window

Output Variables

Item	Tab	Description
1	Score	In this menu, you can set the match quality (in percent) for each object found. You can use the tolerances to take into account known faults in the object. If you are teaching in a faultless object, the value for perfect and the two fields next to it should also be 100. You must lock the values for Tolerances and Perfect so that they do not change when you close or reopen the Properties box. If the number of objects is greater than one, this measurement generates a series of measured scores for all objects. All individual scores can be displayed on the operator monitor and are available in the Script editor.
2	Count	In this menu, you can set the number of matches found.
3	Rotation	In this menu, you can set the rotation angle for each object. The rotation angle is specified with the accuracy set in the "Sensor Settings" menu under Calibration. If the number of objects is greater than one, this measurement creates a directory of rotation measured values for all objects found. All individual rotation values can be displayed on the operator monitor and are available in the Script editor.
4	X	In this menu, you can set the horizontal position for the center of each object. All position values can be displayed on the operator monitor and are available in the Script editor.
5	Y	In this menu, you can set the vertical position for the center of each object. All position values can be displayed on the operator monitor and are available in the Script editor.
6	Model	In this menu, you can set the name (string) of the found object.

Output Variable Settings

Setting	Description
Name	You can change the variable name of the tool or the measurement. The variable name is displayed in the image, in the tool list, and in the measurement results. The name is used as a variable in scripts.
Enable	Enables the data output of the relevant output variable.
Relative	See "Relative Measured Values" on page 78
Tolerance	See "Setting the Tolerance Values" on page 77

Parameters

Parameter	Description
Auto label position	Sets the name field next to the area in the display. Clear the check box to move the name field to another position, or drag the name field to the image with the property field closed.
Locator	This function causes the tool to follow the test object as it moves. Select a locator (by number) that you want to follow. To detach the tool from the locator or to hold the tool in a fixed position in the image area, select X. You can define up to four locators.
Show points	This function displays a point in the center (origin) or corner of the object found in the region and allows you to use the point(s) in other measurements. You can also use the point tool to drag the "center" to another position. This can be useful in some applications such as robot guidance.
Hide labels	Use this check box to hide the number of "labels" on each match object found.
Max count	This field specifies the number of matches expected in the search area. This allows the tools to find multiple matches in a search area or to end the search when the maximum number is reached.
Models	Select single or multiple models. The pattern template area is disabled if you select multiple models.
Method	Here you can choose which type of template or type of pattern match is used. • All pixels—means that the pixel intensities match. • Edges—finds only the edges in the search area and ignores the pixel intensity values. (See example at the end of the section.) • All—both types of pattern matches are used.
Rotation	The range (in degrees) for the rotation of the match object. • Min—enter a negative number for counter-clockwise rotation. • Max—enter a positive number for clockwise rotation. Note: You can enter rotation limits of min= -180 and max= 180 or min= 0 and max= 360.
Required score	Required score—sets a minimum number or a threshold for the score of each hit. Potential hits that do not meet this minimum are rejected or not recognized.

Buttons

Button	Description
OK	The changes are applied and the editing window is closed.
Cancel	The editing window closes without any changes being made.
Apply	The changes are applied without closing the editing window.
Preprocess	This function allows you to add image filtering to the tool. The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.
Advanced	These settings are optional. They allow more control over the object and the runtime. You can zoom in and out on areas and hide them. Hiding areas allows you to cut them from your search area by using a brush to turn those areas red. This function increases the test time.
Help	This opens the help window. Here you will find useful operational information.

Example: Edges

You can use edge matching or geometric matching for image templates with clearly recognizable contours. The "Edge" option is available on the properties page of the Match Tool. This method of pattern-finding is more stable and accurate than area-based matching. However, its execution speed is slower. Keep the search area shapes for patterns and areas as small as possible for optimal performance.

To locate a particular feature on the Viking's helmet, the following example uses an edge-based pattern comparison. This feature can be quickly found in any orientation as long as the part rotates within the defined search area. To align a Count Tool, which counts the number of characters on the part, the center of the match is used. The Count Tool search area rotates according to the output of the match locator.

Figure 7.11 Example: edge-based object comparison

Creating a Locator Point

The Match Tool is suitable for creating a locator because it has a search field. You can customize the search field to compensate for the movement of the test object. You can change the Match Tool properties to search only for edges or for all pixel intensities. You can also change the rotation limits of the Match Tool.

1. Click the "Match Tool" button in the tool selection area.

   ↳ Move the mouse over the image in the work area. There should be no highlighted borders.
2. Draw a search field around the test object. Set the search area to the largest expected movement range of the test object.
3. Right-click on an edge of the rectangular search field.

   ↳ This opens the editing window for the Match Tool.
4. Next to the "Show points" function, select "center."
5. Create a second search field in the image area. Set the search area to the largest expected movement range of the test object.
6. Right-click on an edge of the rectangular search field.

   ↳ This opens the editing window for the Match Tool.
7. Next to the "Show points" function, select "center."
8. Move the mouse over the center of the **first** search area. When the point turns red, right-click to open the point properties.

   ↳ In the "Act as Locator" area, you can define the position locator and the rotation locator.
9. Click the radio button next to "Pos" (for position). Click "OK" to accept changes and close the "Point Properties" menu.

   ↳ A "crosshairs" symbol appears in the image area, centered on the point.

   Note

   The position locator alone can track or follow horizontal and vertical motion. The position locator alone cannot track or follow rotations. The position locator alone may be sufficient if your part is mechanically limited to not rotating, such as a square or rectangular part that slides between two close-fitting rails. If your part rotates slightly, you will need a rotation locator.
10. Move the mouse over the center of the **second** search area. When the point turns red, right-click to open the point properties.

    ↳ In the "Act as Locator" area, you can define the position locator and the rotation locator.
11. Click the radio button next to "Rot" (for rotation). Click "OK" to accept changes and close the "Point Properties" menu.

    ↳ A "crosshairs" symbol appears in the image area, centered on the point.
12. Run a test run to check that the locator point has been detected and is within the search area. Where necessary, customizations must be made.

2021-03

7.3.2 Count Tool

The Count Tool searches for light objects on a dark background or dark objects on a light background. The tool can locate the center and corner points of surfaces that can be used as localization points, position tracking, or reference points for other tools.

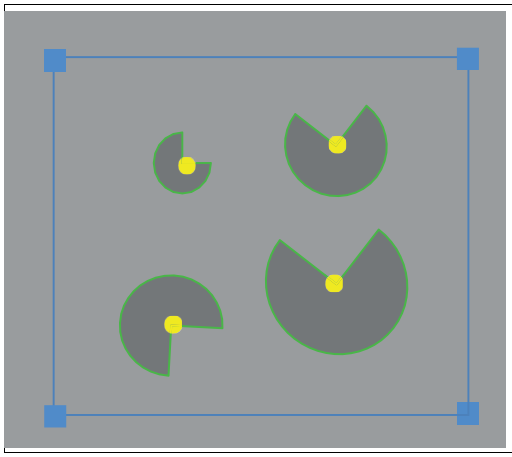


Figure 7.12 Rectangular counting in the search area shows found blobs with their centers
The Count Tool provides interactive filtering to extract blobs of interest

Count Tool Information	
Supports search area shapes	Rectangle, polygon, circle, annulus, ½ annulus
Tool preparation	Variable click point for creating a search area shape
Measurements	Count (total) area, secondary area, main area (each)
Points	Center corner of each bounding box
Surface selection	Dark, light, or optional
Background	Uniform variable (light change over the search area)
Threshold	Interactive slider
Surfaces filter	Area, secondary area, main area, X, Y, first, or all

Using the Count Tool

1. Click the "Count Tool" button in the tool selection area.

   ↳ Move the mouse over the image in the work area.

2. Left-click on your starting point. Release the left mouse button and move the mouse to the area that represents your search area. Left-click again to draw a rectangular search area on the image.

   ↳ The selected field is labeled (N, N1, etc.). Inside the search area, the areas found are outlined in yellow.

3. Resize the search area by clicking and dragging one of the corners of the search area.

4. Right-click on an edge of the rectangular search area.

   ↳ The properties window for the Count Tool will open.

Count Tool properties window

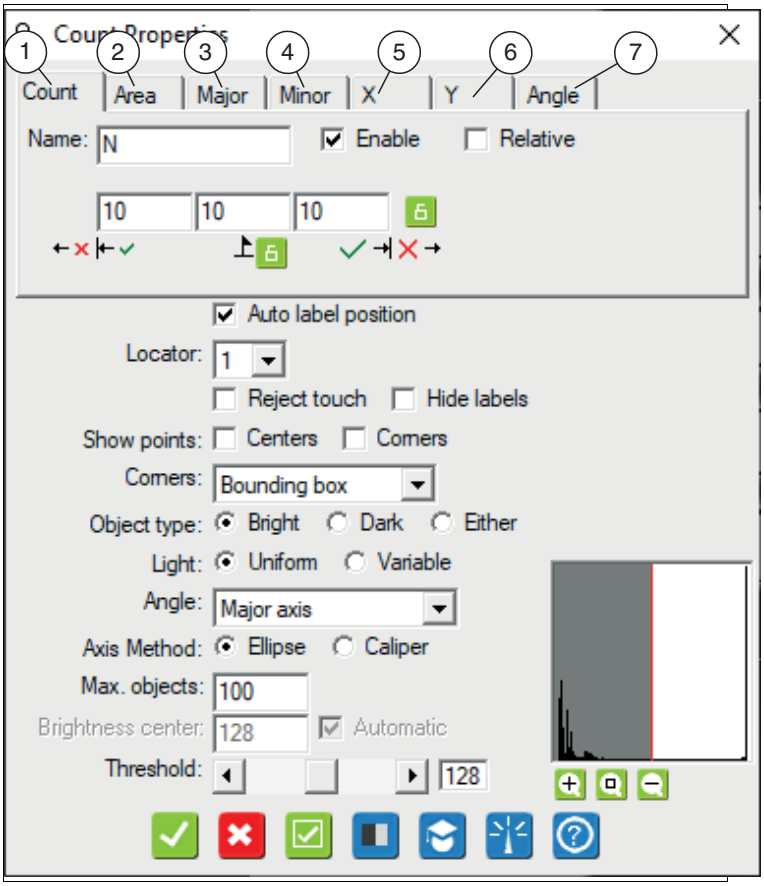


Figure 7.13 Properties window

Output Variables

Item	Tab	Description
1	Count	This menu is used to set the number of found areas in the search field.
2	Area	This menu is used to set the measurement of the size (surface area) of each object. This measurement creates a series of measured surface area values for all objects. All area values can be displayed on the operator monitor and are available in the Script editor.
3	Major	This menu is used to set the measurement of the major axis of each object. This measurement generates a series of measured values. All values can be displayed on the operator monitor and are available in the Script editor.
4	Minor	This menu is used to set the measurement of the minor axis of each object. This measurement generates a series of measured values. All values can be displayed on the operator monitor and are available in the Script editor.
5	X	In this menu, you can set the horizontal position for the center of each object. All values can be displayed on the operator monitor and are available in the Script editor.
6	Y	In this menu, you can set the vertical position for the center of each object. All values can be displayed on the operator monitor and are available in the Script editor.
7	Angle	The angle (of the major axis or the longest edge) of the horizontal, for each object. This measurement generates a series of measured values. All individual values can be displayed on the operator monitor and are available in the Script editor. The count angle displayed in the operator monitor is an average of the angles found for all objects. The count angle is useful for individual objects.

Output Variable Settings

Setting	Description
Name	You can change the variable name of the tool or the measurement. The variable name is displayed in the image, in the tool list, and in the measurement results. The name is used as a variable in scripts.
Enable	Enables the data output of the relevant output variable.
Relative	See "Relative Measured Values" on page 78
Tolerance	See "Setting the Tolerance Values" on page 77

Parameters

Parameter	Description
Auto label position	Sets the name field next to the area in the display. Clear the check box to move the name field to another position, or drag the name field to the image with the property field closed.
Locator	This function causes the tool to follow the test object as it moves. Select a locator (by number) that you want to follow. To detach the tool from the locator or to hold the tool in a fixed position in the image area, select X. You can define up to four locators.
Reject touch	This function refers to objects that touch the edge of your search field.
Hide labels	Use this check box to hide the number of "labels" on each object found.
Show points	This function displays a point in the center (origin) or corner of the object found in the search field.
Corners	Use this function to change the behavior when the "Show points" selection is set to "Corners." Bounding box = uses the corners of the bounding box that surrounds the blob object. Major axis = use the end points of the calculated major axis.
Object type	Light means that light objects on a dark background are identified and counted. Dark means that dark objects on a light background are identified and counted. Both means finding both light and dark objects.
Light	The type of incident light on the object or light change in the search area. Variable means that the light can vary across the object (from left to right or from top to bottom). Selecting "Variable" switches on the "local adaptive threshold", but is less sensitive to background fluctuations. Selecting "Uniform" switches on the "adaptive threshold", which is applied uniformly to the entire search field.
Angle	Use this function to set the measurement and display of the angle of the major axis or the longest straight line in the object. The selection depends on the shape of the objects you are looking for. The major axis is calculated by estimating an "equivalent elliptical" object with the moments of inertia as the actual object. This is a quick calculation that provides a good estimate of the major axis for objects close to an ellipse or rectangular shape. This does not work for square or near-square objects. The longest straight line selects one side of the square and specifies the angle. This works well for squares, but is not suitable for other shapes.
Axis Method	Use this function to select a method for calculating the major axis of the object. Ellipse: This is the default method; it calculates an "equivalent ellipse" for the object. Caliper: This calculates a caliper diameter on the basis of the object's widest point. This method is more accurate, but requires more computing time and is therefore slower.
Max objects	The maximum number of objects found or accepted in the search field.

Parameter	Description
Brightness center	Use this function to assign a "center" intensity threshold if the object type is one of the two and the "Automatic" check box is cleared. The brightness center is used when the object type is one of the two. If the "Automatic" check box is selected, the center is calculated from the content of the search field. This is the usual process for detecting and counting objects or faults. Assigning a threshold value for the center means that the Count Tool counts one object. If the search field is either completely light or completely dark and = 0 if the region is filled with an intensity close to the "center" value. Use the threshold value to increase or decrease the area around this "center" value.
Threshold	A lower threshold value is more sensitive to faults or light changes. The field on the right shows the histogram or intensity graph (in black) for the pixels along the search field. The shaded area indicates the threshold value level. You can change the threshold value using the slider or by dragging the boundary between light and shade in the field. The buttons below the intensity graph are used for vertical zoom. Note: The slider and the boundary can move in the same direction or in opposite directions, depending on the value of the object type parameter.

Buttons

Button	Description
OK	The changes are applied and the editing window is closed.
Cancel	The editing window closes without any changes being made.
Apply	The changes are applied without closing the editing window.
Preprocess	This function allows you to add image filtering to the tool. The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.
Learn	The "Learn" function automatically adjusts the minimum or maximum area by learning from a series of "good" images. This function is optional.
Advanced	These settings are optional. They allow more control over the object and the runtime. You can zoom in and out on areas and hide them. Hiding areas allows you to cut them from your search area by using a brush to turn those areas red. This function increases the test time.
Help	This opens the help window. Here you will find useful operational information.

Sample Applications

The Count Tool is flexible and can be used in multiple applications.

The following sample application uses the Count Tool to check the number and size of holes in a metal plate. One or more search areas can be used. The image on the right shows how filtering can be applied to find surface areas of a certain size.

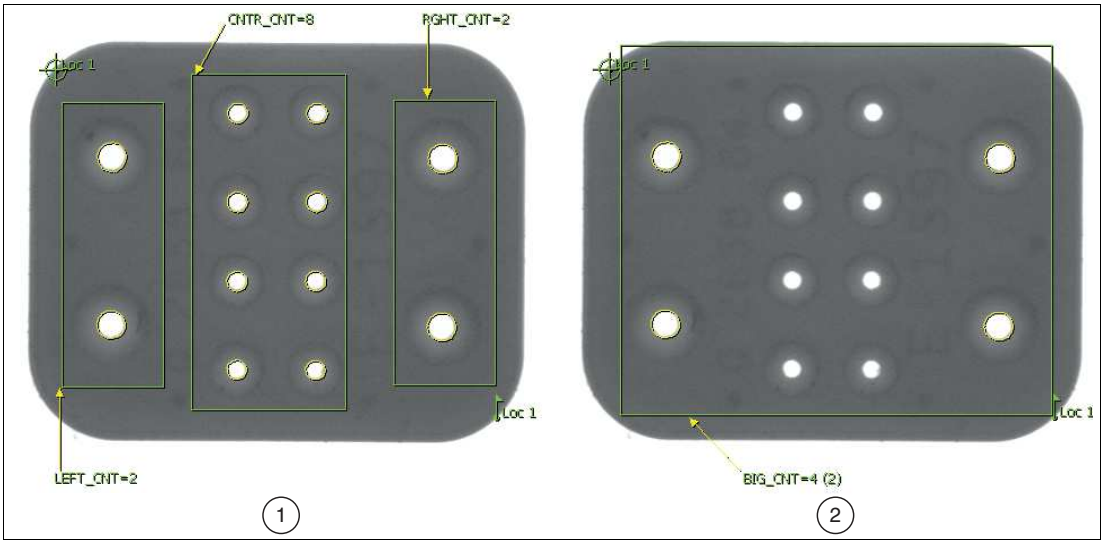


Figure 7.14 Example: checking holes in a metal plate

1. Checking holes on a metal plate
2. Filtering by hole size

In the following application, the Count Tool is used to determine the area of an object as a dimension of the connected pixels. This is useful for checking objects of any shape.

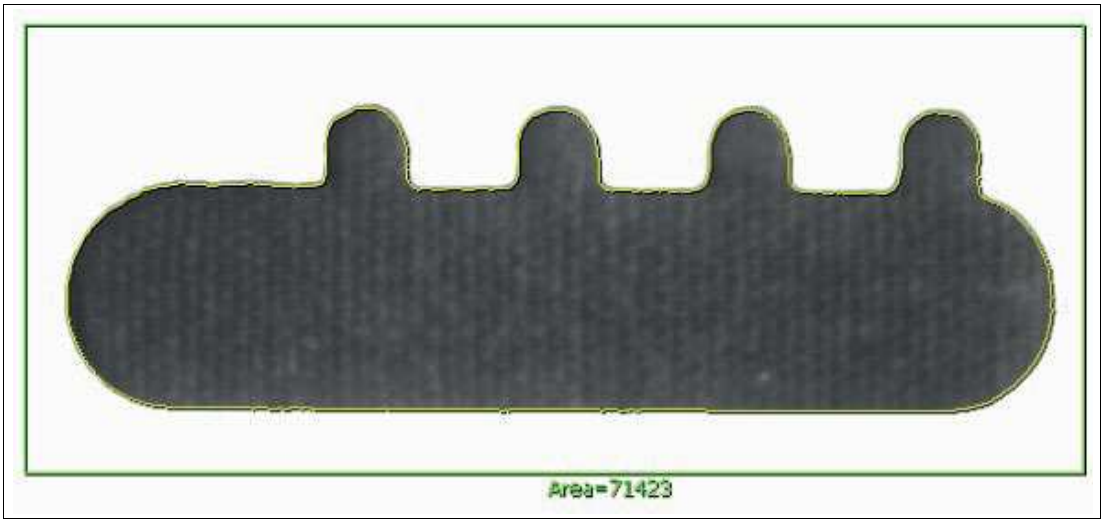


Figure 7.15 Example: determining dimension of connected pixels

The example below shows how the Count Tool can be combined with an Edge Tool to create a 360° positioner for orientation. The Count Tool is used to determine the X-Y position of the inner black surface (1). Since the black surface is symmetrical, it cannot provide the rotation information required for orientating a pattern comparison tool used on the outside to check the edge. However, this information can be determined using an Edge Tool that has a circular search area (2) and is aligned with the position locator. The Edge Tool detects the grayscale transition of the "L" sign (3) embossed on the part, regardless of its orientation. The point of this edge transition can be used as the rotation locator of the positioner. When the position and rotation locators are defined, the adjustment tool can be positioned exactly above the expected position of the notch.

2021-03

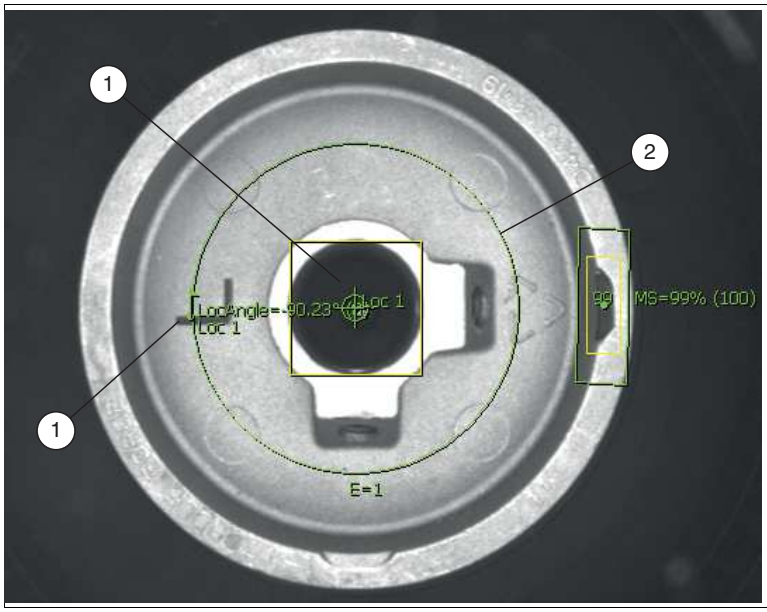


Figure 7.16 360° positioner

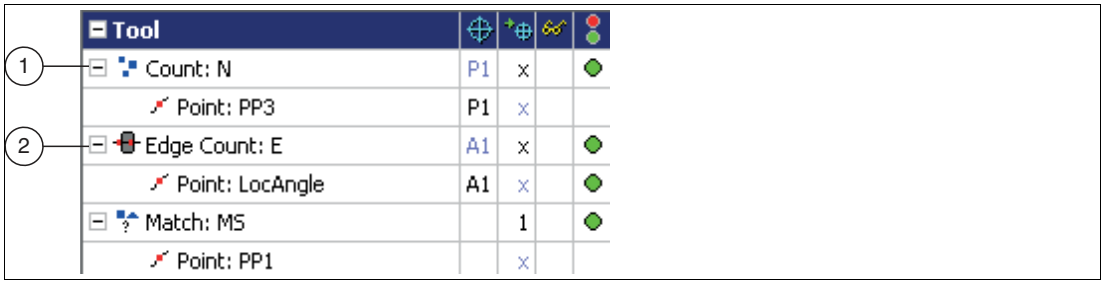


Figure 7.17 Tool combination

1. P1 = position indicator 1
2. A1 = rotation indicator 1

7.3.3 Edge Count Tool

The Edge Count Tool is used to count the number of edges that a line passes over. It can also be used to measure the distance between each pair of edge points along the path. The Edge Count Tool is suitable for various applications, including detecting the presence/absence of features and precision measurement.

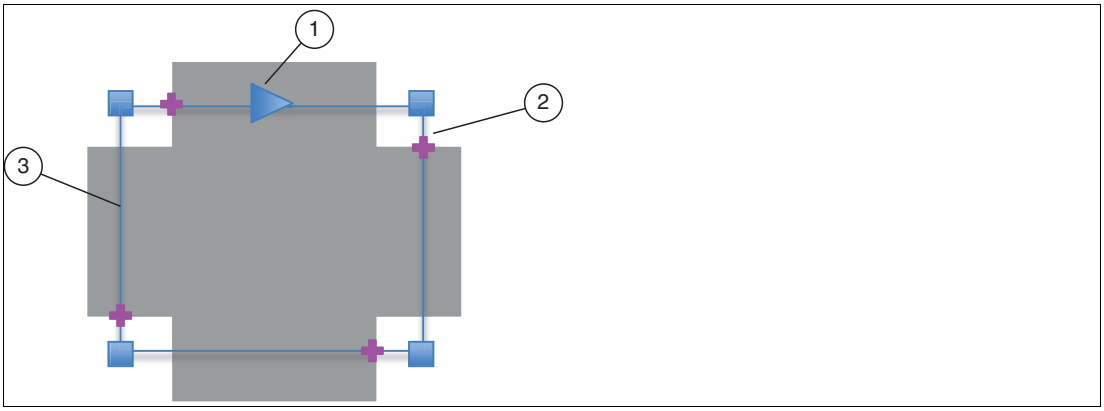


Figure 7.18 Edge counting

1. Default edge scanning direction
2. Light to dark border point
3. Number of edges of rectangular search area

Edge Count Tool Information	
Search area shapes	Line, rectangle, polyline, circle, arc
Tool preparation	Variable click points to create the search area shape
Measurements	Number, average, minimum point spacing, maximum point spacing
Scan direction	Selectable, with or against arrow
Selection of edge	Thickest or first edge
Polarity of the edge	Light to dark, dark to light, optional
Edge transition	Automatic with sensitivity control
Edge points	Passive (measurement only) or active

In the example below, the tool looks for the mounting alignment tabs for an assembly on a cell phone keyboard. This is possible using line search areas that are aligned to a positioning aid on the part (not shown). This example shows the localization of dark to light edges.

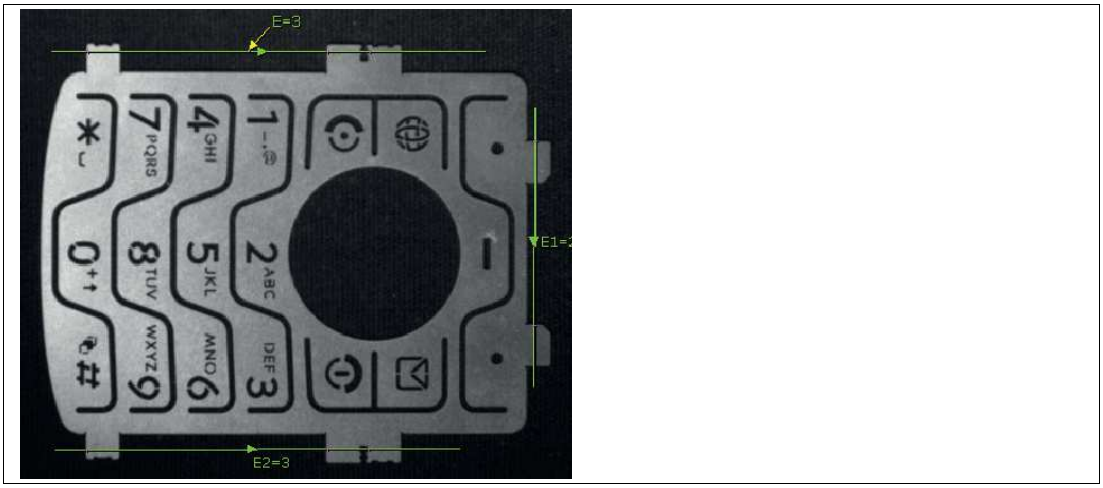


Figure 7.19 Sample application: cell phone keyboard

Similarly, the Edge Tool can be used to search for unrequired edges. For this application, an Edge Tool is positioned over the cap. If no edges are detected, the cap is correctly attached.

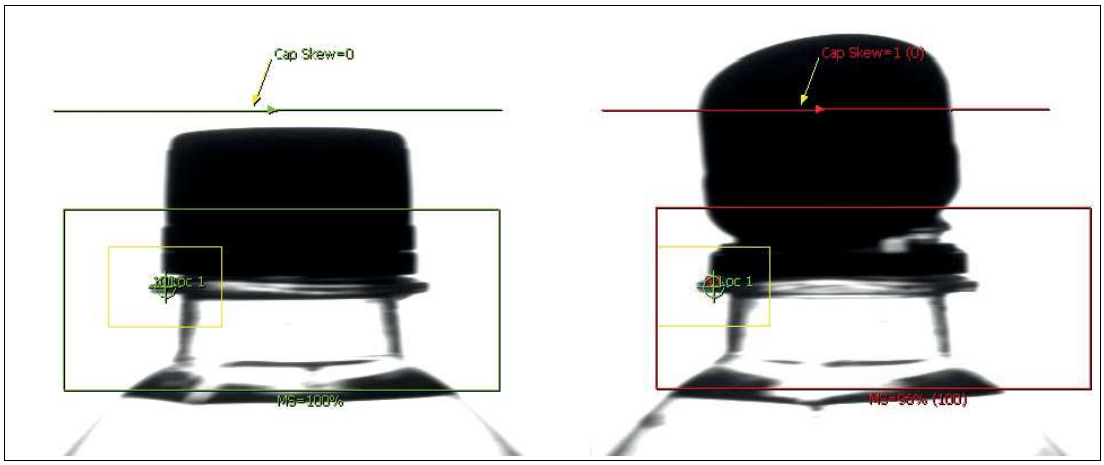


Figure 7.20 Sample application: bottle cap

The Edge Count Tool can be used to measure the distance between each pair of edge points along the line. The following example measures the distance between each edge on a roll of electrically punched pins to check for deformations.

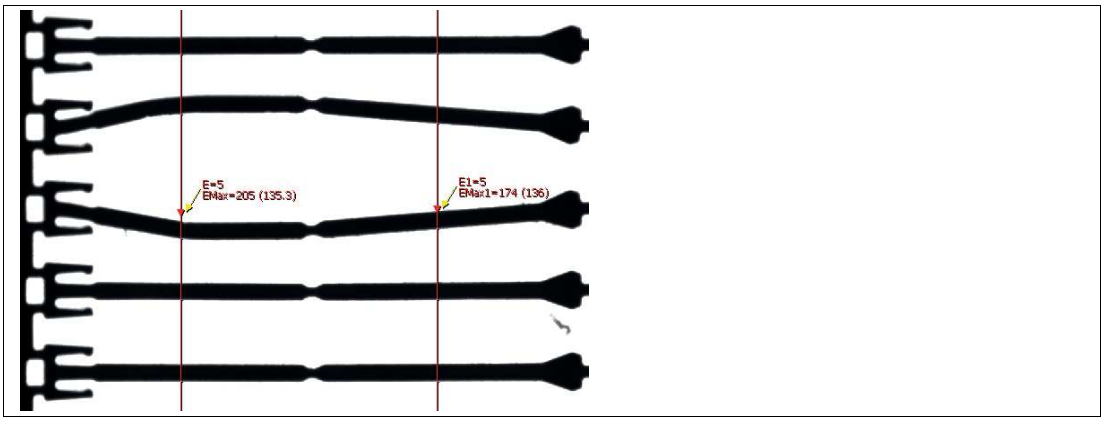


Figure 7.21 Sample application: roll of electrically punched pins

In the following application, the pins on a PCB connector are counted to ensure that the assembly is correct.

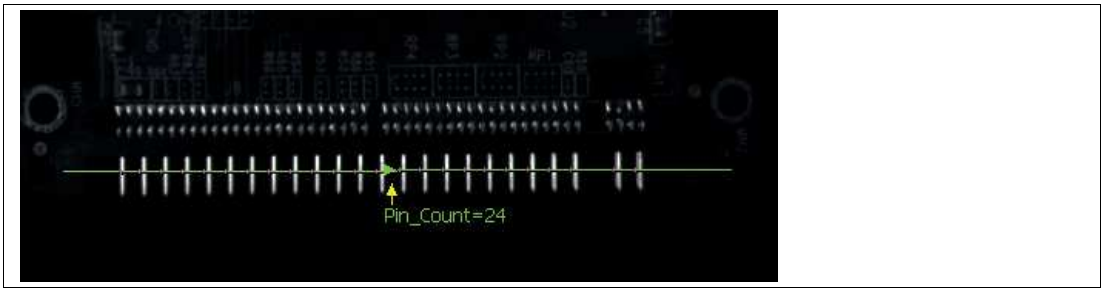


Figure 7.22 Sample application: PCB connectors

7.3.4 Intensity Tool

The Intensity Tool measures the grayscale values or brightness values of the pixels in the search field. The values can be output via static functions such as average brightness value or other static functions.

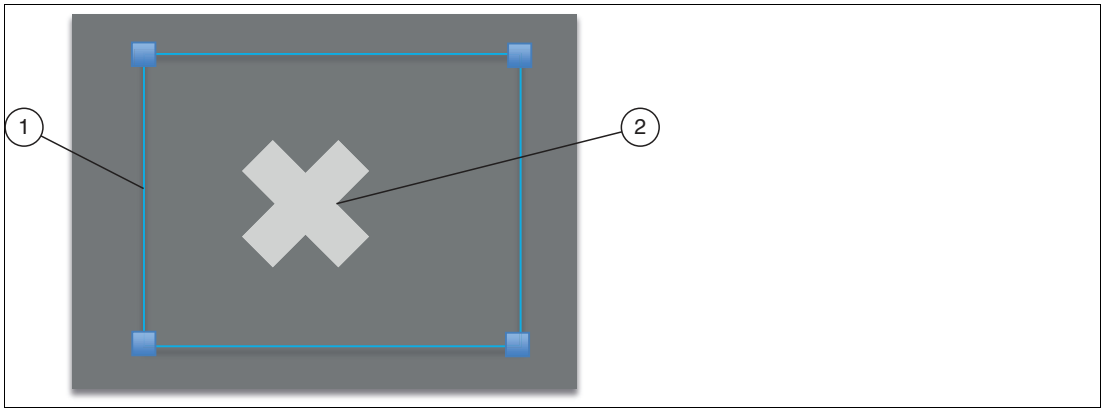


Figure 7.23 Intensity Tool

1. Intensity rectangular search area
2. Detected feature

Intensity Tool Information	
Search area shapes	Rectangle, polygon, circle, annulus, ½ annulus
Tool preparation	Variable click point for creating a search area shape
Measurements	Average, minimum, maximum, standard deviation, mode, mean value of intensities
Limit values	Minimum and maximum brightness values

The Intensity Tool returns the average pixel value of all grayscale values within the search area. It also displays the minimum and maximum values, the standard deviation, the mode, and the mean value. The min/max values are often used to look for very dark or very light faults, such as pin holes in a surface. The min/max count limiters are useful in some situations where you want to hide the min/max values from the average result.

In the example below, the differences in brightness between the average grayscale values are used to determine whether a pill is present in the pack. The grayscale values vary by 50 shades of gray with controlled lighting, which is sufficient for a reliable result.

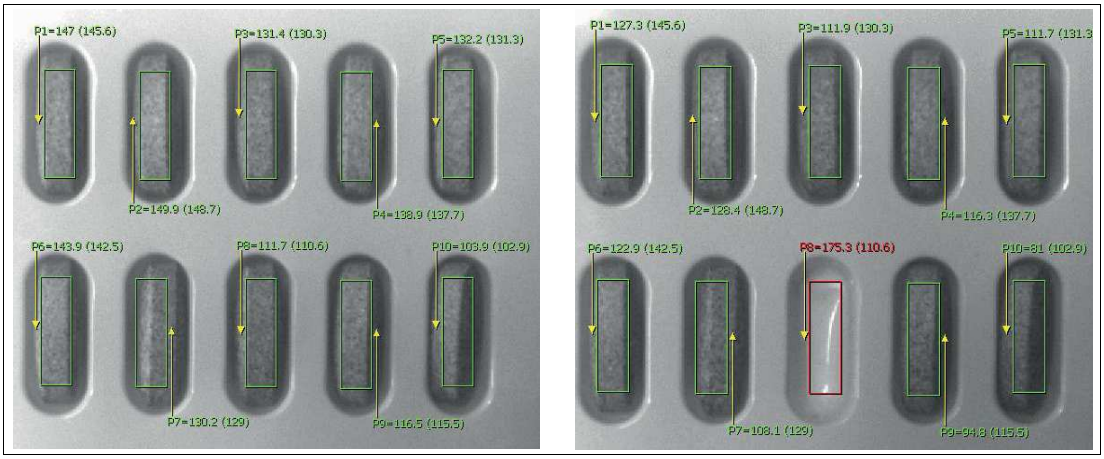


Figure 7.24 Example: pill pack

Similarly, the following example shows how the Intensity Tool is used to check whether the mounting tabs on a plastic part are closed. In this case, one of the tabs is damaged and the part is rejected before it moves to the next stage of the process.

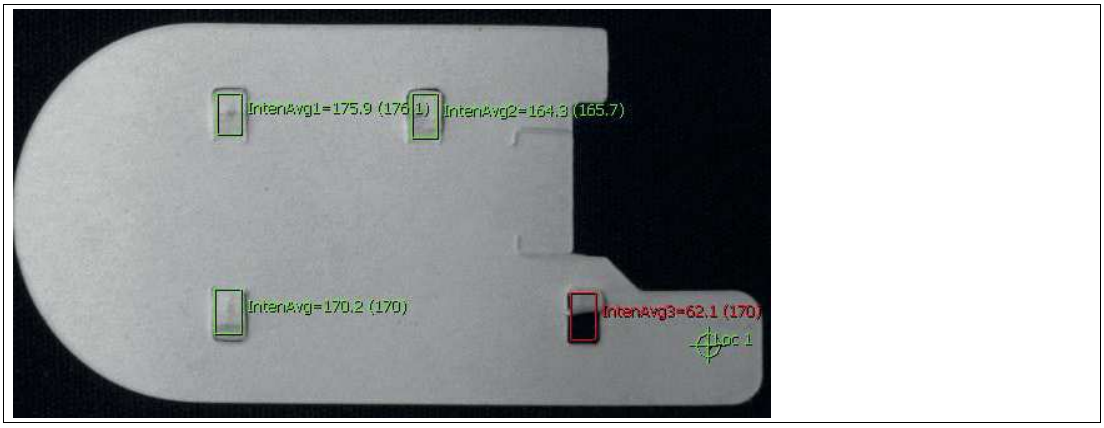


Figure 7.25 Example: mounting tabs on a plastic part

Another example shows how the Intensity Tool is used to check a toothbrush head. This application looks for major faults where one side or the center of the brush is missing.

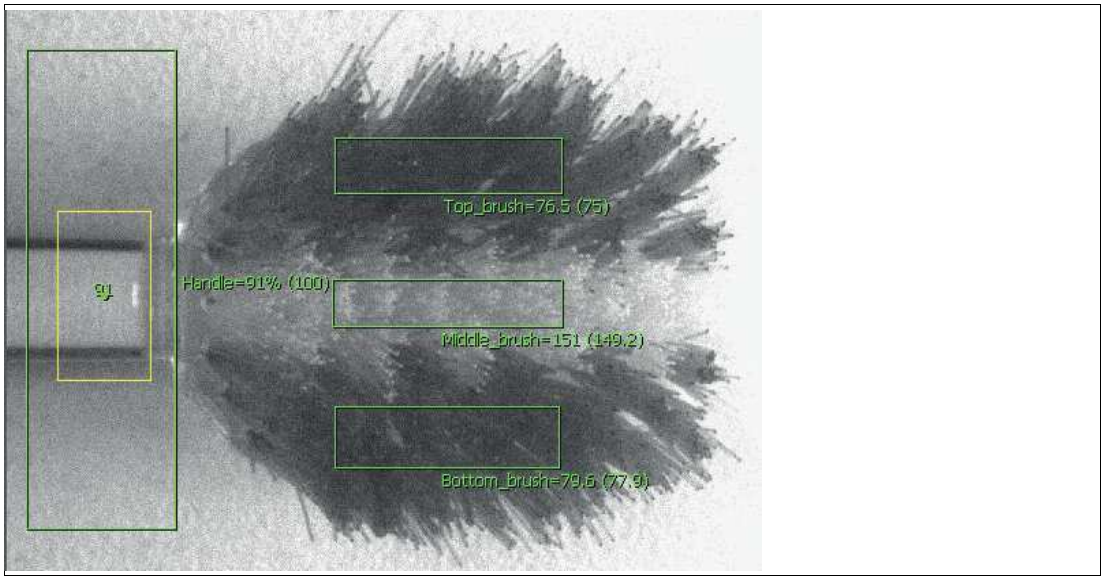


Figure 7.26 Example: toothbrush head

The following example shows how a bright spot in the center of a molded plastic part can be eliminated by using the maximum number limiter when calculating the average intensity. In this case, the nine brightest pixels associated with the bright spot are removed by setting the maximum count limit to ten.

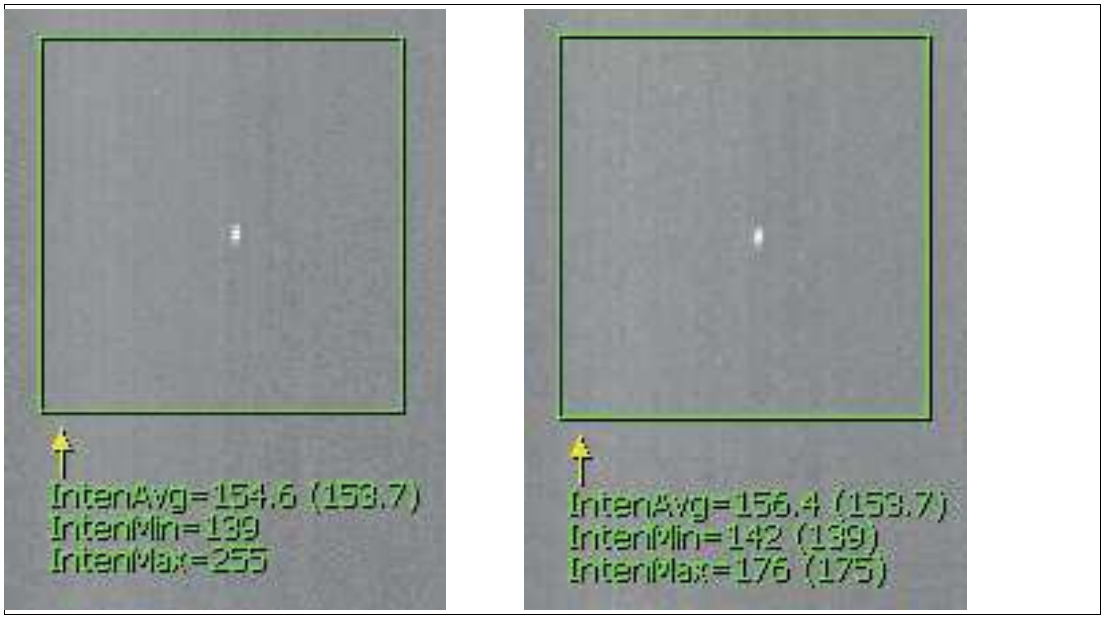


Figure 7.27 Example: molded plastic part

7.3.5 Caliper Tool

The caliper is used to measure the distance between two ends—just like a real caliper does. The caliper allows you to measure the width of a complex shape by scanning from the outside to the inside, or the width of a gap by scanning from the inside to the outside, or by doing a combination of both.

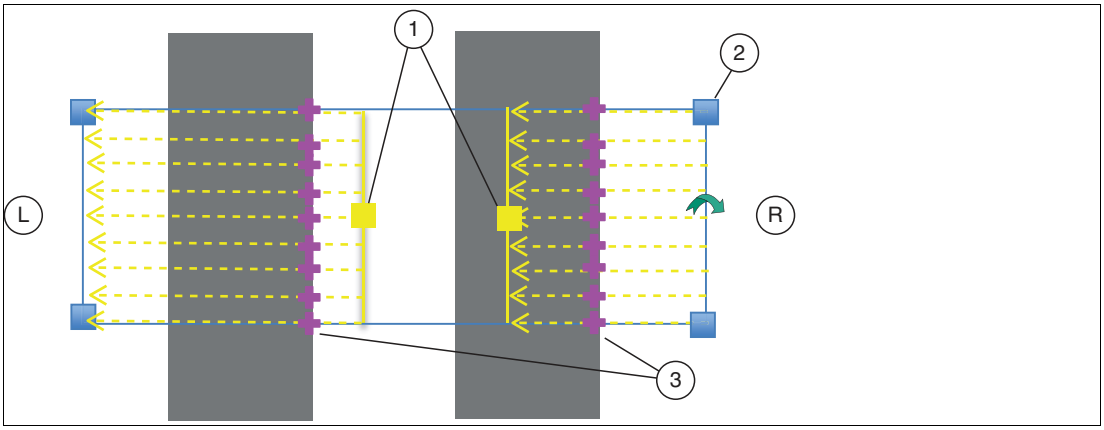


Figure 7.28 Caliper Tool

1. Caliper scan limiters
2. Caliper search area
3. Caliper edge points

L Left-hand scanning of the caliper: from inside to outside

R Right-hand scanning of the caliper: from outside to inside

Caliper Tool Information	
Search area shapes	Rectangle with movable scan limits
Tool preparation	Three click points: top-left, top-right, bottom
Measurements	Length
Scan direction	From inside to outside or from outside to inside
Selection of edge	Thickest or first edge
Polarity of the edge	Light to dark, dark to light, optional
Edge transition	Automatic with sensitivity control
Accuracy	2 to 50 rake points per side
Outlier support	No

The scan limiters define the size of the scan search area on both sides of the caliper. Each search area can be resized by moving the yellow square. If you move one of the scan limits to one edge of the caliper search area, the tool measures on one side.

For illustration purposes, the following example shows the caliper being used to measure the inner minimum and outer maximum distance between two arcs.

Left-hand and right-hand limiters are placed inside the arcs and each side is configured to scan from the inside to the outside and to look for light to dark border transitions.

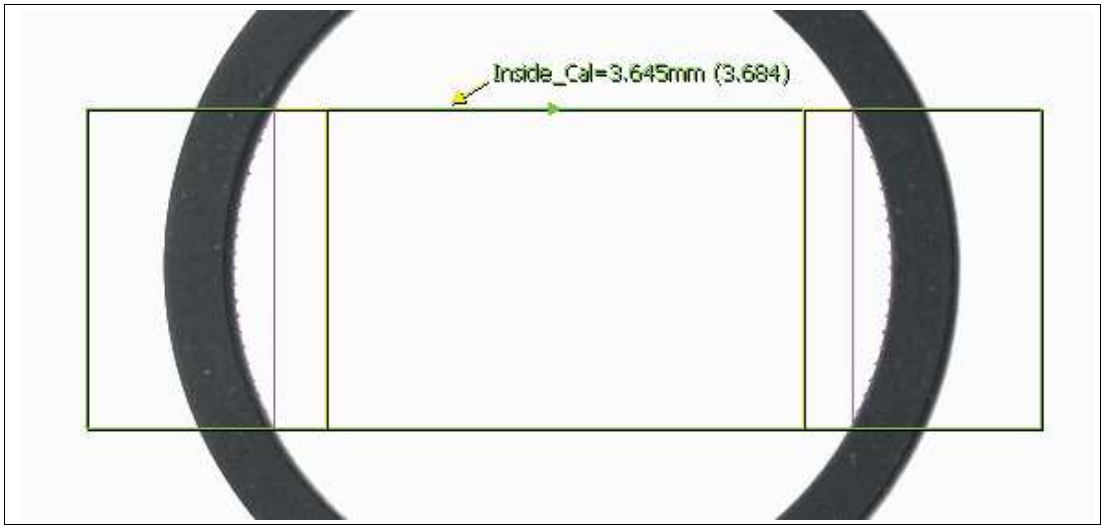


Figure 7.29 Limiters inside the arcs

To measure the outside caliper, the scan direction and the edge type are reversed, as seen in the figure below.

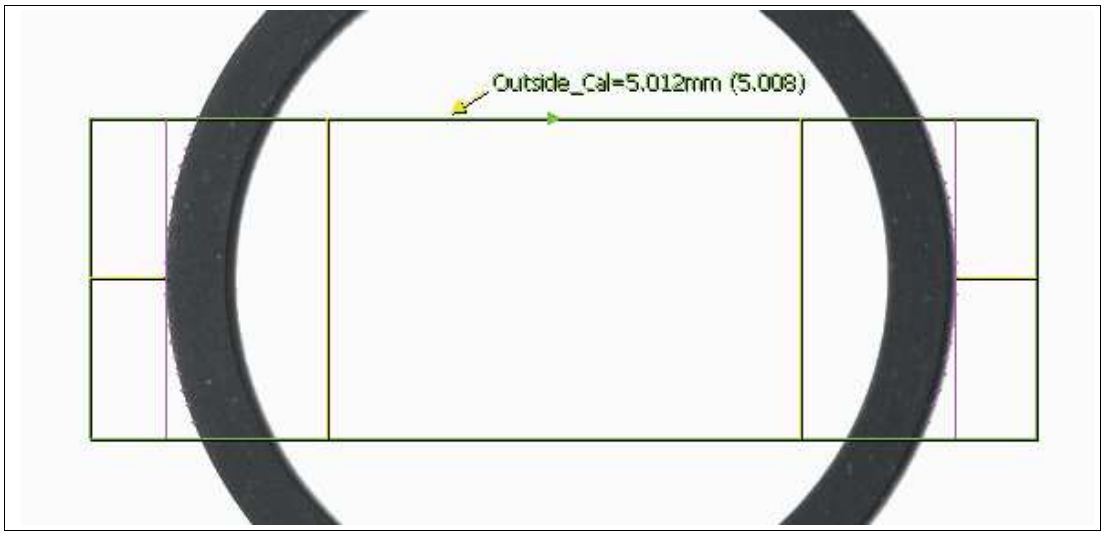


Figure 7.30 Limiters outside the arcs

To measure the distance between the left-hand and right-hand outer points on a thread, a Caliper Tool is used in the following example.

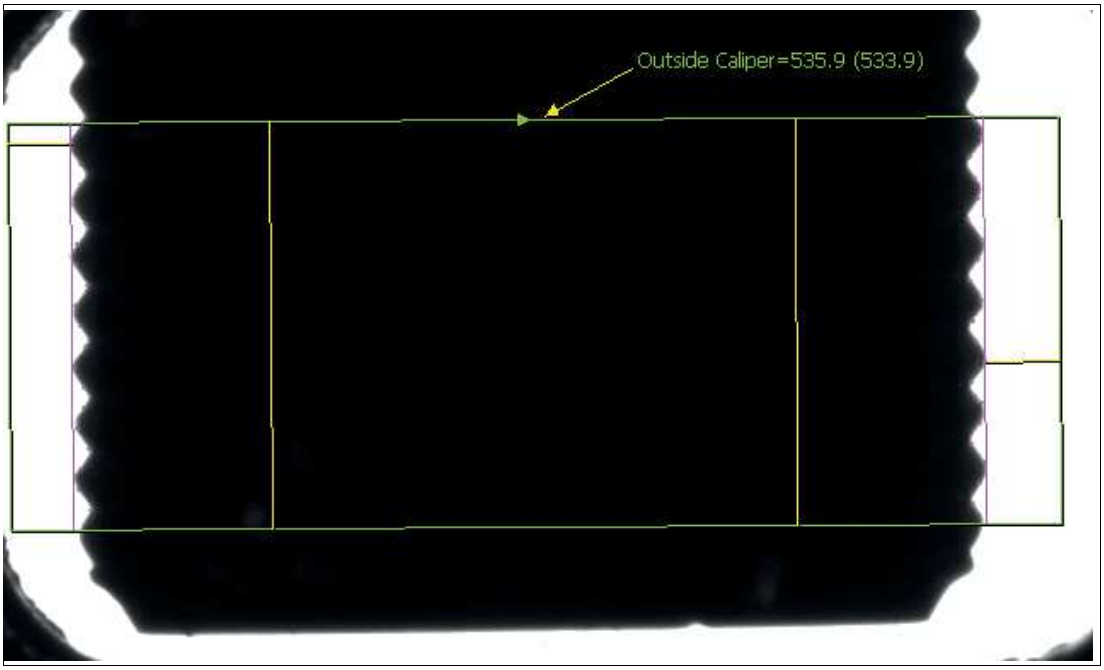


Figure 7.31 Measuring the outer points on a thread

In the following example, the Caliper Tool is applied to a preprocessed image of a cylindrical plastic container. This application is intended to ensure that the ends of the container are installed and undamaged (which is shown as a distorted end).

7.3.6 Point Tool

The Point Tool is used to locate the exact location of a point along an edge. The tool matches a line to the edge and reports the intersection of the line with a vertical line passing through the user's click point. The Point Tool finds edge points used in the line fitting algorithm by searching along invisible lines within its search area. Additional edge points can be added to improve accuracy. The search area can be resized such that it almost corresponds to the edge.

Point tools are often used as position sensors to facilitate the use of other tools. A single point defined as the "position locator" provides XY position information for each tool to which it refers. Adding a second "rotation locator" point (which refers to the first) also provides the orientation angle.

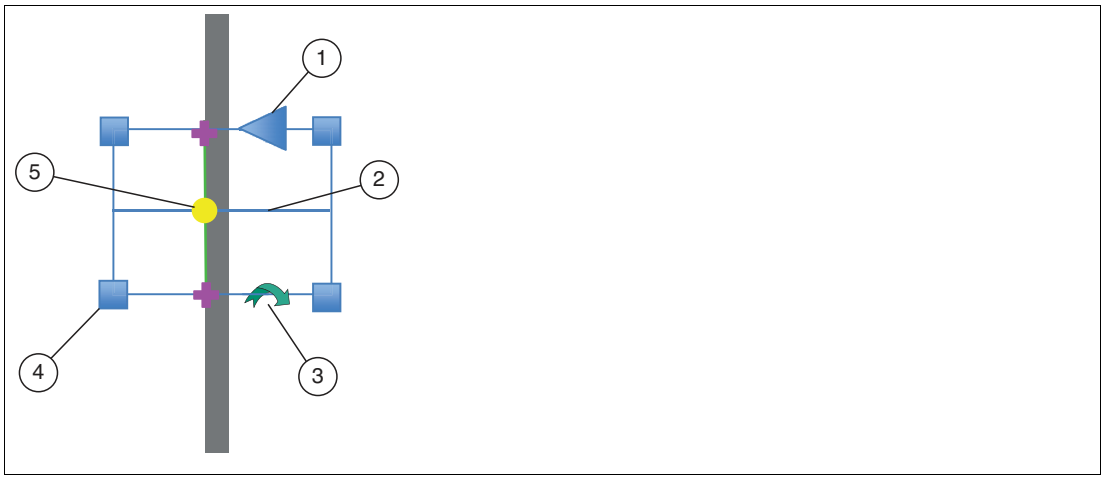


Figure 7.32 Point on the best fitting line with two-point accuracy

1. Default edge scanning direction
2. Click point line
3. Rotating tool for the search area
4. Resizing the search area
5. Point

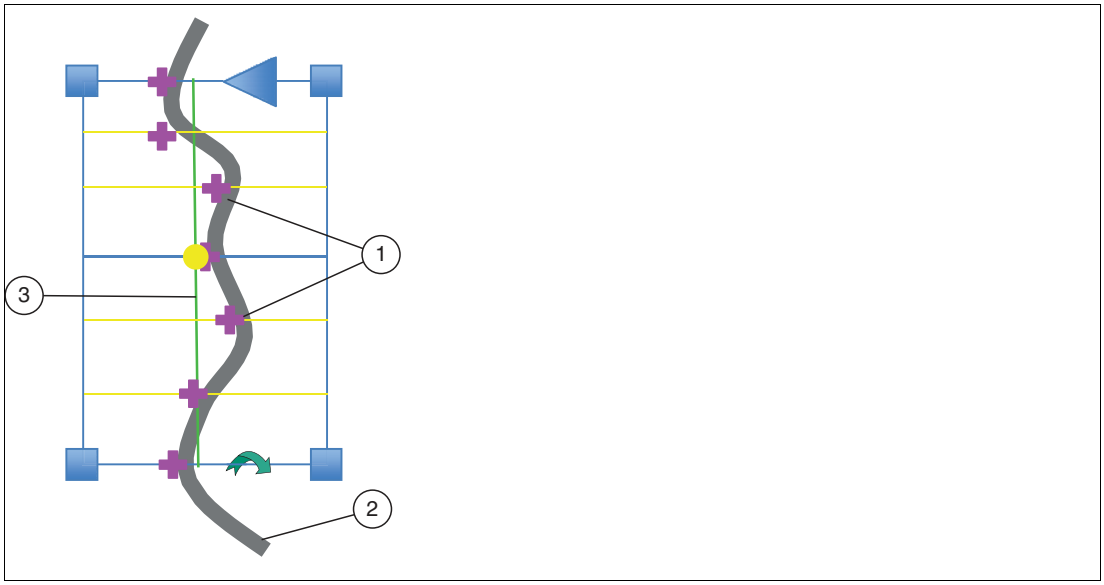


Figure 7.33 Point on the best fitting line with seven-point accuracy

1. Line fitting points
2. Edge
3. Line fitting

Point Tool Information	
Search area shapes	Point (edge search area at each end)
Tool preparation	Click any edge
Measurements	Position deviation, XY coordinates
Scan direction	Selectable, with or against arrow
Selection of edge	Thickest or first edge
Polarity of the edge	Light to dark, dark to light, optional
Edge transition	Automatically with sensitivity slider
# Rake lines	2-50 points (three default values)

Creating a Point Tool on an Object

1. Click the "Point Tool" button in the tool selection area.
2. Move the mouse over the image in the work area.

   ↳ You see the borders in the image highlighted as you move over them. You see that the points in the search area turn red as you move over them.
3. Right-click on an edge in the image.

Figure 7.34 Point Tool

↳ The Point Tool (1) is created on the selected edge.

4. Move the mouse over the point.

   ↳ A search area appears around the point.
5. Resize the search area by clicking and dragging one of the corners of the search area. Use the curved arrow to rotate the search area. A small arrow on the outer edge of the search field indicates the direction.

Editing the Point Tool

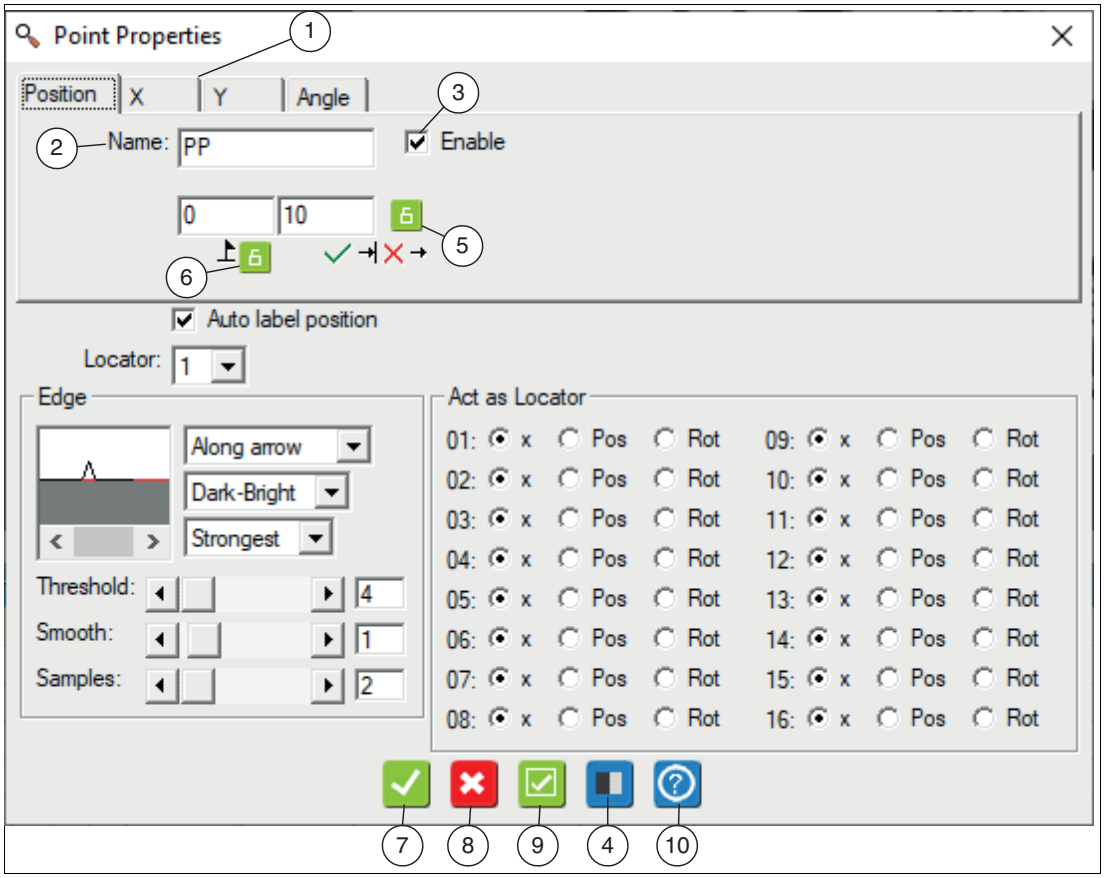


Figure 7.35 Point Tool editing window

1. Right-click on the Point Tool in the image.

   ↳ This opens the editing field for the point properties. By default, no measurement is enabled. In the editing window, you can rename the selected point (2), enable the measurement (3), and manage the tabs (1).

2. Click the "Preprocess" button (4) to add an image filter to the tool.

   Note

   The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.

3. Click the "padlock" button next to the tolerance fields (5) or perfect value (6) to lock or unlock the tolerance values or perfect value.

   ↳ The values do not change when you move or resize the area, or make other changes.

4. Click the "OK" button (7).

   ↳ The changes are applied and the editing window is closed.

5. Click the "Cancel" button (8).

   ↳ The editing window closes without any changes being made.

6. Click the "Apply" button (9).

   ↳ The changes are applied without closing the editing window.

7. Click the "Help" button (10).

   ↳ This opens the help window. Here you will find useful operational information.

Output Variables

Tab	Measurement
Item	Variation in either horizontal or vertical position. All units are given in pixels.
X	Horizontal position, in pixels.
Y	Vertical position, in pixels.
Angle	Rotation of the taught-in position, only if this point is a rotation locator point. You must define both the position and rotation locators for the locator point. Rotation is always positive clockwise. For all other points, the angle is always 0.

Creating a Locator Point

You have the option of using the Point Tool to define a "locator point." This provides XY position information for each tool to which it refers. In some applications, it may be necessary to define a locator point if the test objects are not always in the same position in front of the camera.
You need two locator points to track the rotation of a tool, with or without horizontal and vertical movement. You need a locator point to track horizontal and vertical movements without rotation.

1. Expand the search field of the tool that refers to the reference point to capture all expected motion.
2. Click on the "Point Tool" button in the tool selection area. In the image area, hover over an active point and right-click to open the Point Tool properties.

   ↳ In the "Act as Locator" area, you can define the position locator and the rotation locator.
3. Click the radio button next to "Pos" (for position). Click the "OK" button to accept changes and close the "Point Properties" menu.

   ↳ A "crosshairs" symbol appears in the image area, centered on the point.
4. Right-click on a second active point in the image area. Click the radio button next to "Rot" (for rotation). Click the "OK" button to accept changes and close the "Point Properties" menu.

   ↳ A flag icon attached to this point will appear in the image area.

Note

The two locator points (for position and rotation) should be separated by as large a distance as possible and should stay in the camera's field of view.

5. Check whether the locator point follows the test object or whether you need to make customizations.

7.3.7 Tip Tool

The Tip Tool is a version of the Point Tool. It determines the end points (tip) of a corner or edge. Like the Point Tool, the Tip Tool looks for edge transitions along the lines within the search area. It then arranges the point to determine the tip. The size of the search field can be adjusted, so that it approximates the edge. Increasing the accuracy adds more lines to the pattern.

In most cases, the Tip Tool is used to create a point (2) for other tools. It shows the distance dimension of the tip (3) relative to the outer edge of the search area.

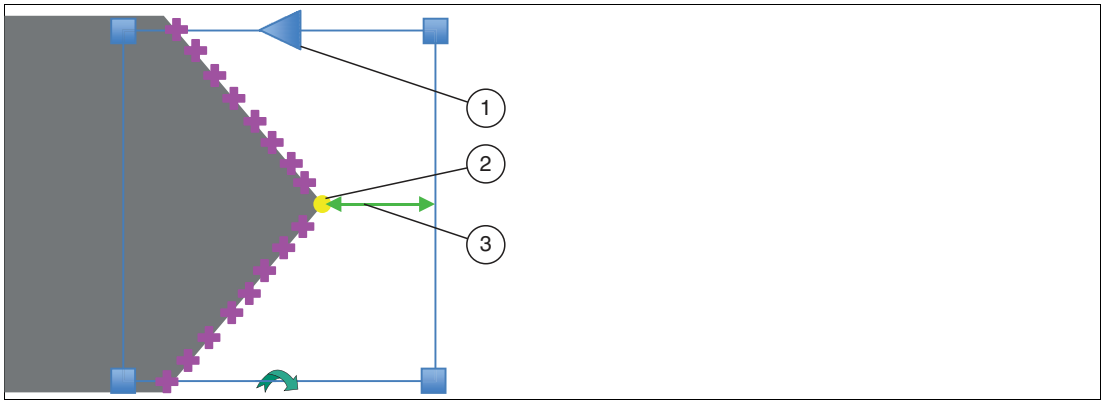


Figure 7.36 Tip Tool

Tip Tool Information	
Search area shapes	Rectangular
Tool preparation	One click on any edge point
Accuracy	2 to 50 rake points within the search area
Scan direction	Selectable, with or against arrow
Edge polarity	Light to dark, dark to light, optional
Selection of edges	Thickest or first edge
Edge transition	Automatic with sensitivity control

The Tip Tool is used to locate the position of a tip for a distance measurement, as shown below.

Note

A distance Tool or Point Tool is not suitable for a tip or corner point, since the rake points that define the best fitting line are on both edges of the tip.

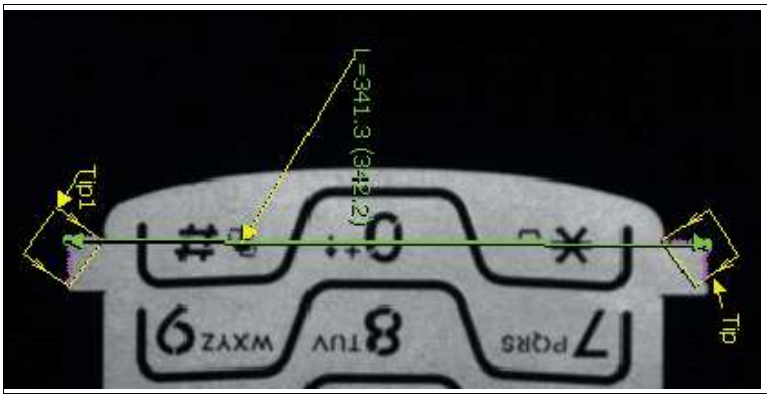


Figure 7.37 Example 1: Tip Tool for distance measurement

In the example below, the tool is looking for the gap length between the tip point on the right and the surface on the left.

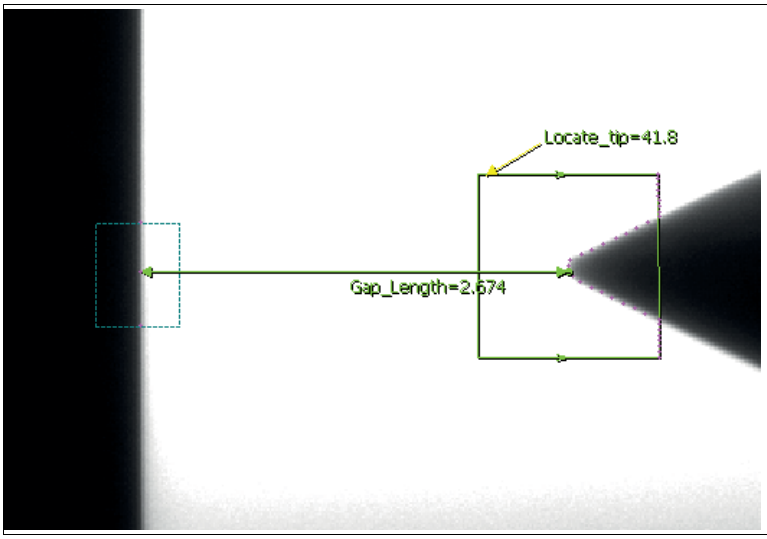


Figure 7.38 Example 2: Tip Tool for distance measurement

7.3.8 Pencil Tool

The Pencil Tool is used to set a reference line from which further measurements are made. For example, a distance (3) or angle tool can use a Pencil Tool as a locator. A Pencil Tool can be created along an image edge or where there are no edges, i.e., a "floating" reference.

If you create a pencil line (1) along an image edge, the tool snaps to the first click point (2) and traces the edge as you move the mouse. The second click point defines the end of the pencil line (1). In this case, the pencil line creates a two-point tool (4) along the edge that defines the rotation or angle of the line. If the last click point is not on an edge, the tool creates the two-point tool along the edge on which the pencil line was started. Once created, the point tools (4) define the locators for the Pencil Tool. They can be moved and resized by selecting the Point Tool in the toolbox. Locator points will not be created if you create a pencil line that does not correspond to an edge.

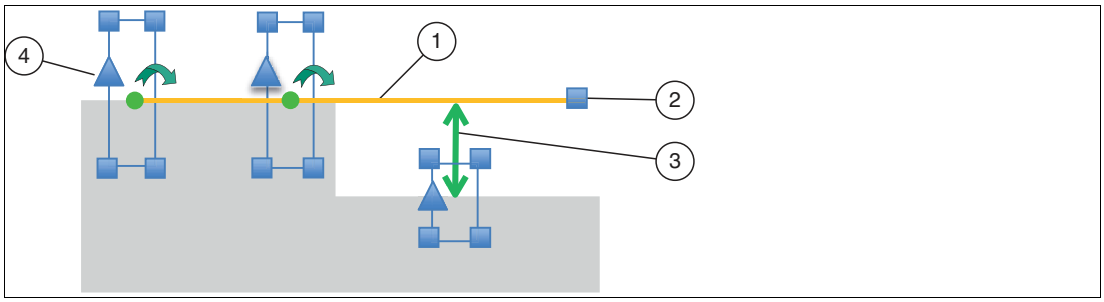


Figure 7.39 Pencil Tool

Pencil Tool Information	
Search area shapes	Straight line
Tool preparation	Two click points on start point and end point
Pencil rotation	Relative to the starting point and the Y axis
Scan direction	Adjustable for locator points, with or against arrow
Edge polarity	Light to dark, dark to light, optional
Selection of edge	Thickest or first edge, for locator points
Edge transition	Automatically with slider for sensitivity

Note

You can attach additional point tools to a pencil line. When the tool is selected, click it and drag a new Point Tool as the new locator on the line. The pencil tool line is a line that best fits between all the point tools you have defined.

The following example shows how two pencil lines can be used to locate the top and right border of a workpiece to align (position) other tools. The intersection of the lines is used for positioning and one of the points on the right-hand pencil line is used as a rotary encoder for angle determination. The points on each pencil line are determined by the first dark-to-light edge in their respective search fields. The search fields of the points are expanded to allow rotation of the part.

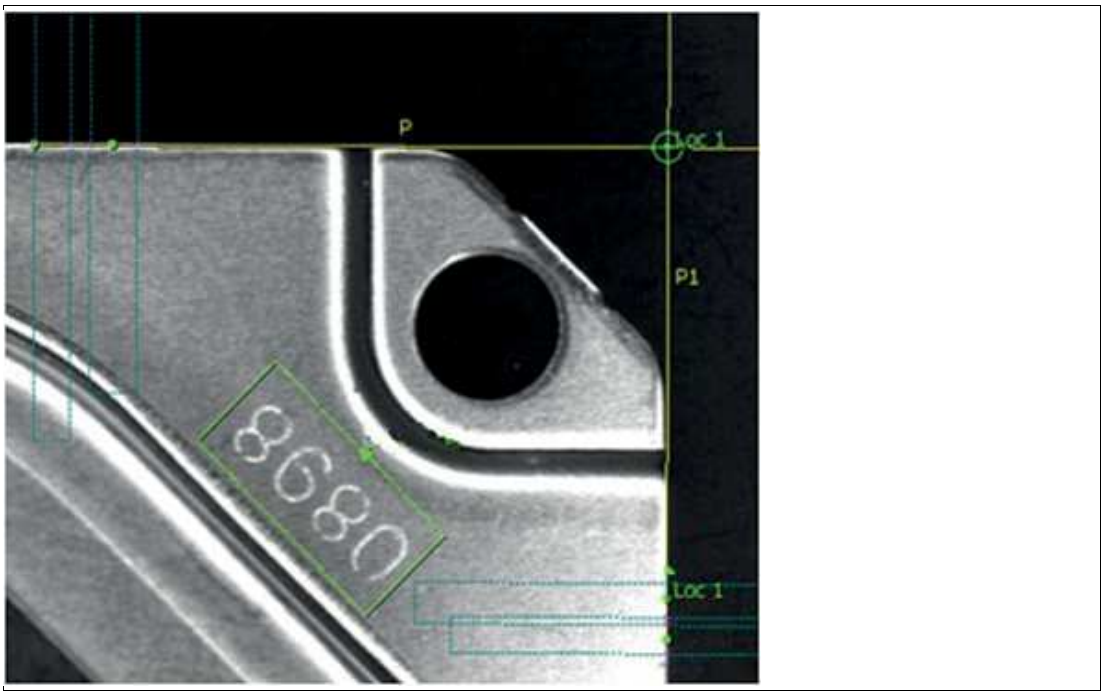


Figure 7.40 Example: locating the border of a workpiece

The left-hand image (1) in the example below shows a pencil line that is used as a floating reference for distance measurement. The right-hand image (2) shows how multiple points are used to construct pencil lines to perform an average distance measurement.

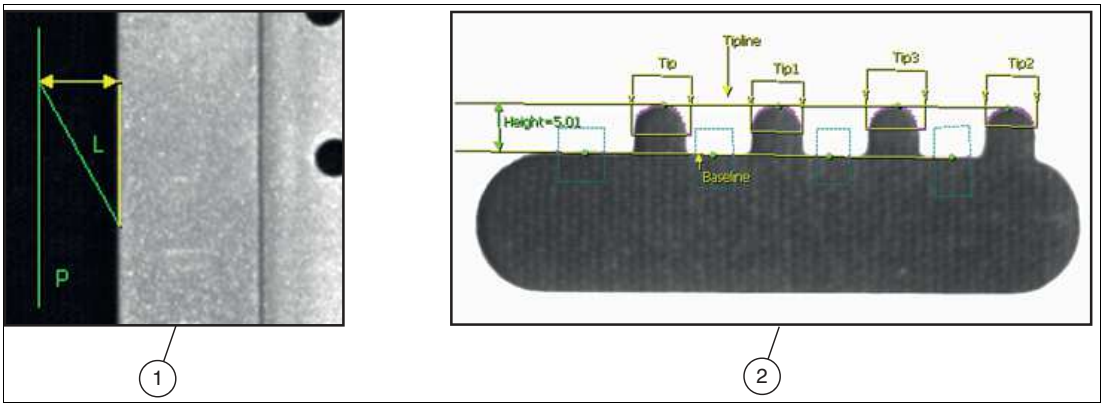


Figure 7.41 Example: floating reference for distance measurement

Rotation measurement using the Pencil Tool can be used to specify the orientation of a workpiece in the X-Y plane. This can be used to guide a robot when performing loading or mounting corrections. The example below uses a Count Tool to find the center of the circular cutout, and uses an Edge Tool to find the key, which may be positioned in any orientation around the center. A Pencil Tool between the center and the current edge point on the key outputs the angle relative to the base line.

Note

In this case, the Pencil Tool determines a positive angle clockwise in relation to the left-hand X-Y origin. The base line is defined as the X-axis of the primary point, i.e., the point that defines the starting point of the pencil.

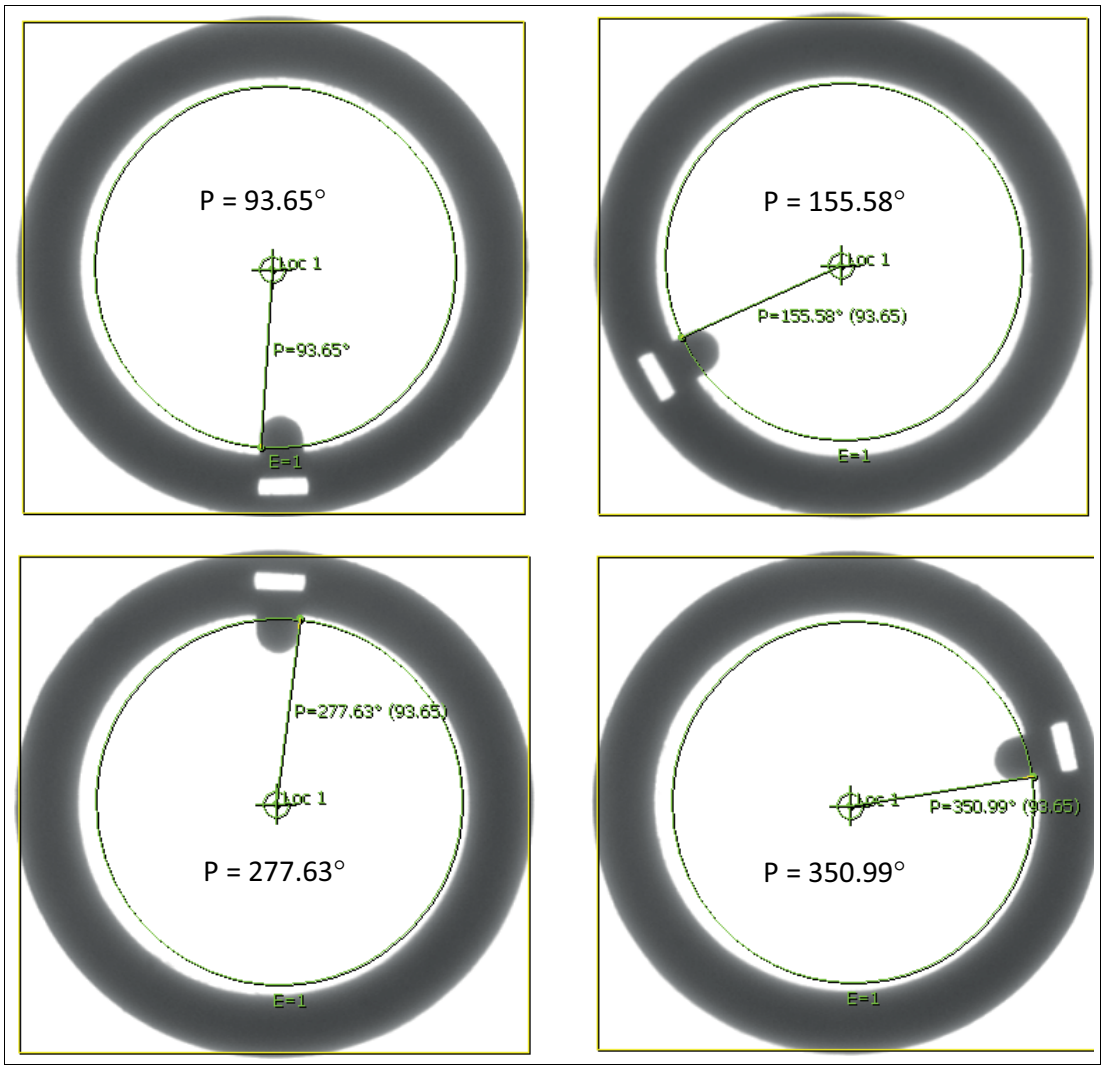


Figure 7.42 Example: rotation measurement

7.3.9 Distance Tool

The Distance Tool is used to measure the distance between two features. Such features include, e.g., image edges (image borders), search area edges, and points generated by other tools. For a high level of measurement accuracy, both edges within the search areas should be straight at both ends.

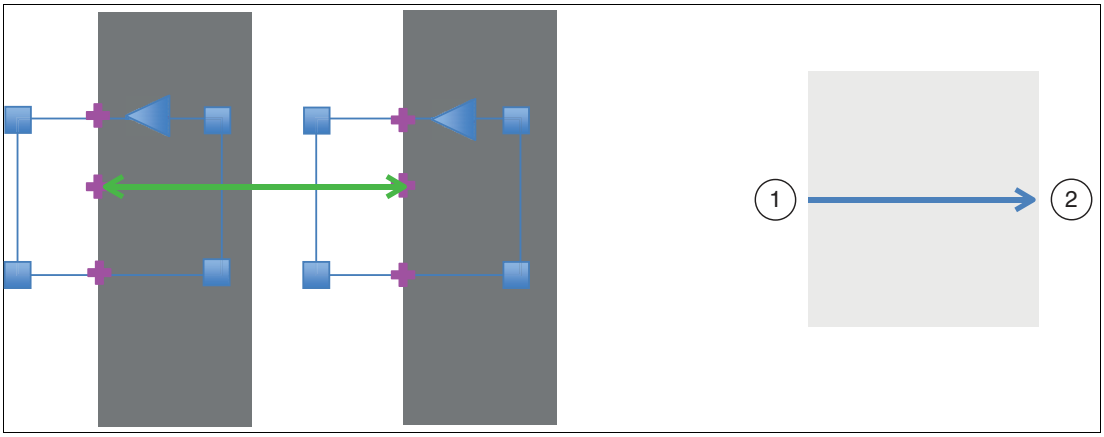


Figure 7.43 Distance Tool

Distance Tool Information	
Search area shapes	Rectangle (border search area at each end)
Tool preparation	Two click points on start and end features.
Measurement	Distance between two edges
Scan direction	Selectable, with or against arrow
Edge polarity	Light to dark, dark to light, optional
Selection of edge	Thickest or first edge
Edge transition	Automatically with slider for sensitivity

When it is anchored to an edge of the image, the Distance Tool always looks for a dark-light transition. When a distance Tool is created on an image, this scan direction is determined automatically. If the direction is incorrect or needs to be changed because you have moved the locator field to another edge, you can use the rotation control point to change the scan direction.

The figure below shows the distance between the selected edge areas in pixel sizes, measured according to the camera sensor resolution. The software supports scaling of the pixels in real measured variables.

Note

Scaling to a real measured variable can be defined either locally for the vision tool or globally for all measurement tools. The measuring distance to the object must be fixed in advance. If the measuring distance changes, the number of pixels per mm changes.

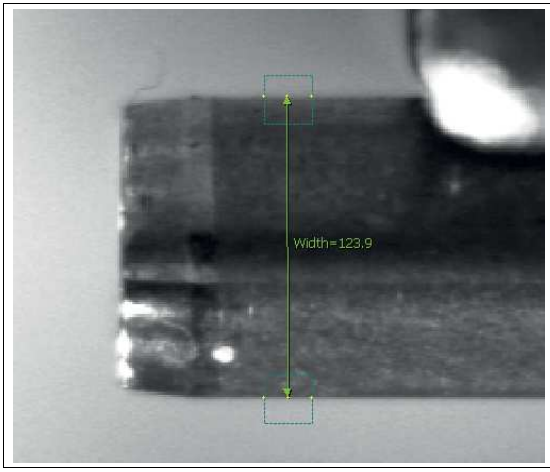


Figure 7.44 Distance between selected edge areas

If the two edges to be measured are straight and parallel, the Distance Tool can measure the vertical distance between them.

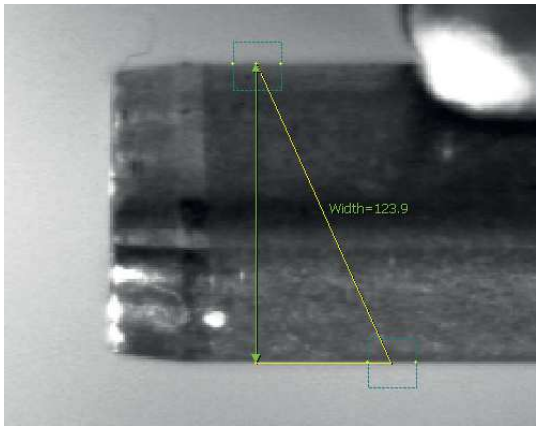


Figure 7.45 Straight and parallel edges

The ends of a Distance Tool can be anchored to a pencil tool, a point tool, or one of the four image boundaries.

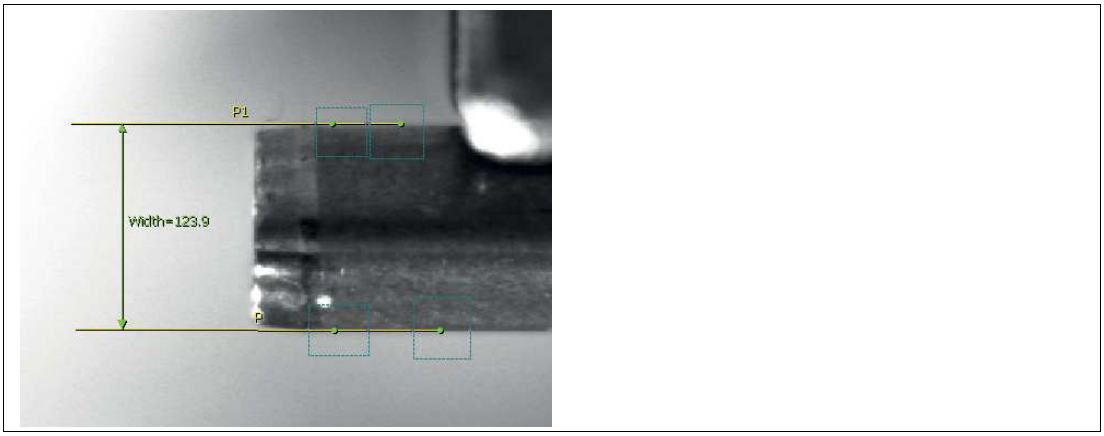


Figure 7.46 The example shows the Distance Tool attached to pencil lines.

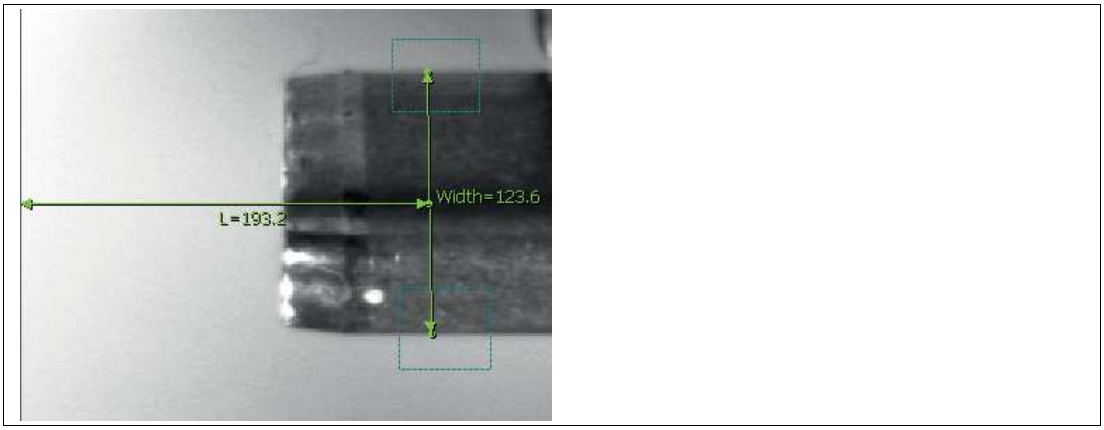


Figure 7.47 The image shows a Distance Tool, which is positioned at the center of another distance dimension and at the edge of the image.

In the following example, the Count Tool determines the centers of the inner holes, which are used by a Distance Tool to measure the distance between the centers.

Figure 7.48 Distance between centers of holes

The Distance Tool connects the centers of the inner holes and determines the axial distance between the two centers. The Pencil Tool connects the centers of the inner holes and determines the angle.

7.3.10 Rake Tool

The Rake tool is an option of the distance tool. It creates multiple planes of the distance tool and reports the average, minimum, and maximum distance of the rake points. It also reports the standard deviation.

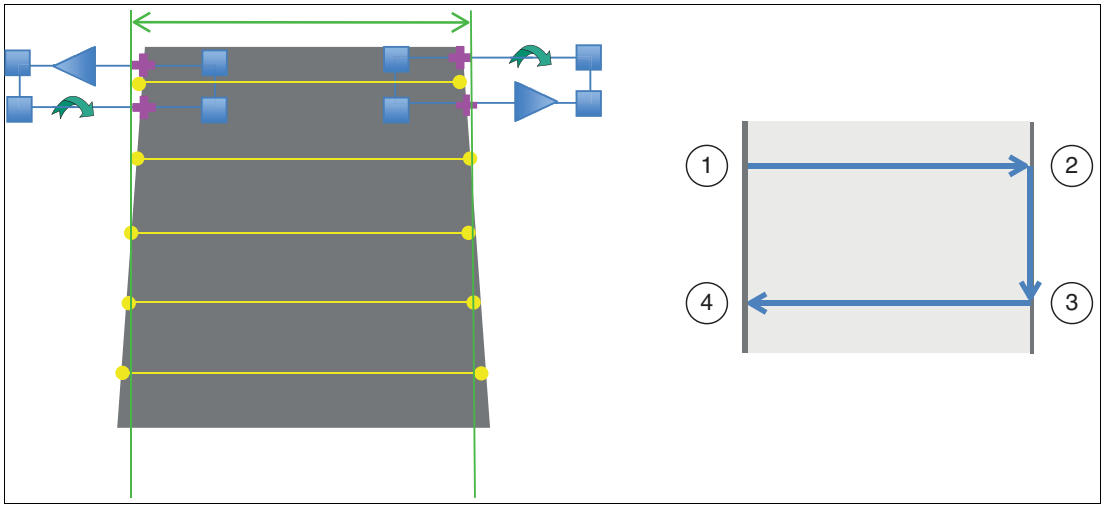


Figure 7.49 Rake tool

Distance Tool Information	
Search area shapes	Rectangle (border search area at each end)
Tool preparation	Four click points at the beginning and end
Measurement	Average, minimum, maximum, standard deviation of the distance between two edges
Scan direction	Selectable, with or against arrow
Edge polarity	Light to dark, dark to light, optional
Selection of edge	Thickest or first edge
Edge transition	Automatically with slider for sensitivity
Accuracy	Number of rake lines (up to 64)

When the rake tool is anchored to an edge of the image, it always looks for a dark-light transition. When a rake tool is created on an image, this scan direction is determined automatically. If the direction is incorrect or must be changed because you have moved the locator field to another edge, you can use the rotation control point to change the scan direction.

The rake tool performs multiple distance measurements at even intervals between two edges. It is useful for measuring how even the distances between the edges are.

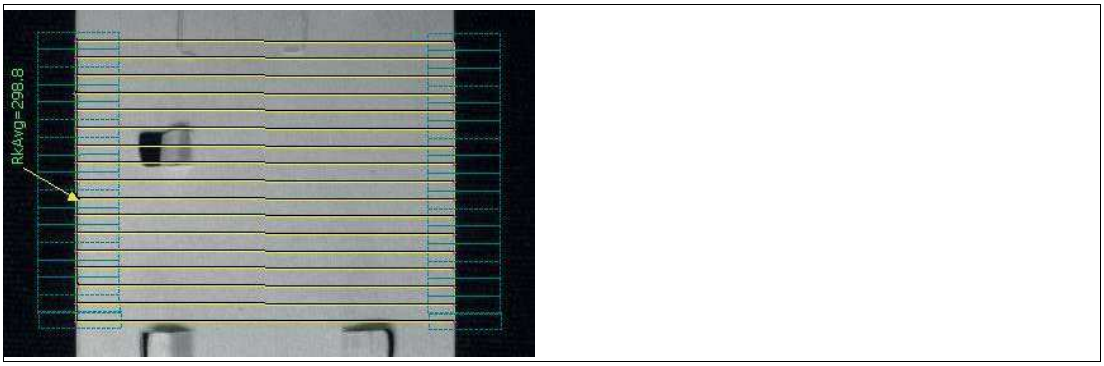


Figure 7.50 Distance measurements at even intervals

The rake tool can use the boundaries of an image or a pencil line as a locator. Individual lengths from the rake can be represented in equations. The figure below shows the left side of the rake tool is aligned with the edge of the image area.

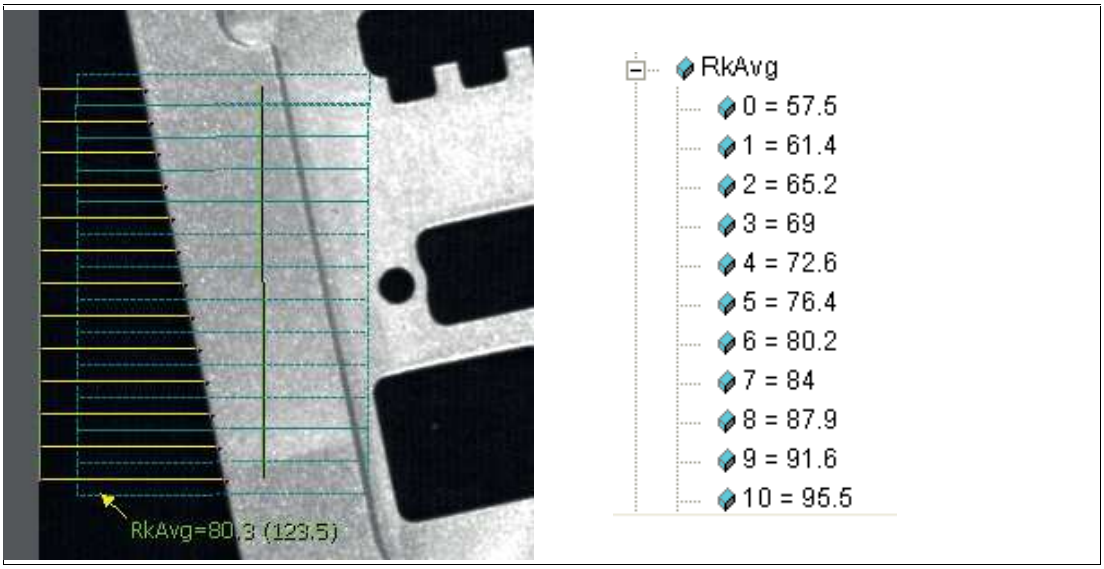


Figure 7.51 Boundaries of an image or a pencil line as a locator

7.3.11 Contour Tool

The Contour tool is used to measure the properties of any shaped contour (continuous edge). The tool tracks an edge during setup and detects deviations from the runtime by measuring the edge point positions (2) along the particular chip line (1). Missing edges indicate an interruption.

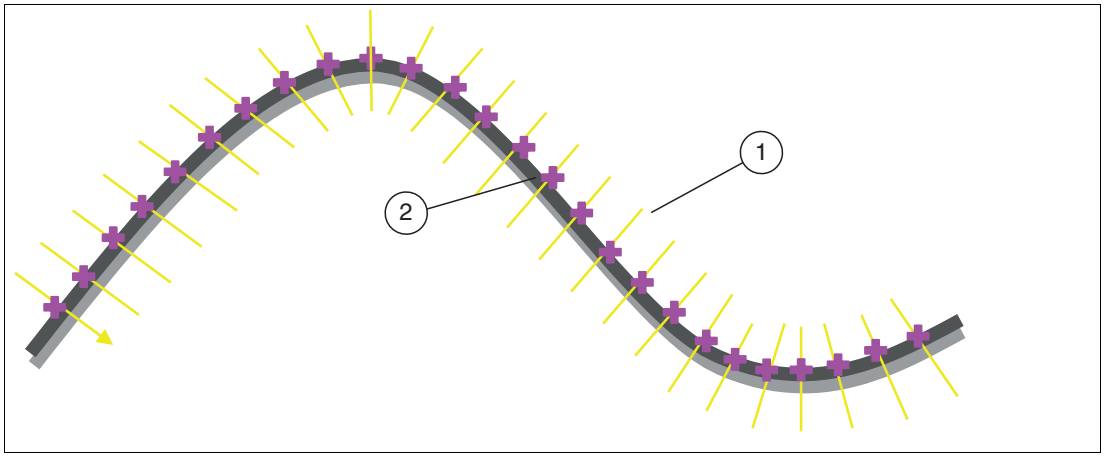


Figure 7.52 Contour tool

Contour Tool Information	
Search area shapes	Automatic edge tracking
Tool preparation	Two clicks on start and end contour points
Measurement	Position deviation, number of failed parts
Scan direction	Selectable, with or against arrow
Edge polarity	Light to dark, dark to light, optional
Selection of edge	Thickest or first edge
Edge transition	Automatically with slider for sensitivity

The following plastic parts are typical applications in which the contour of the workpiece must be checked after the casting process. This allows faults to be identified on the mold.

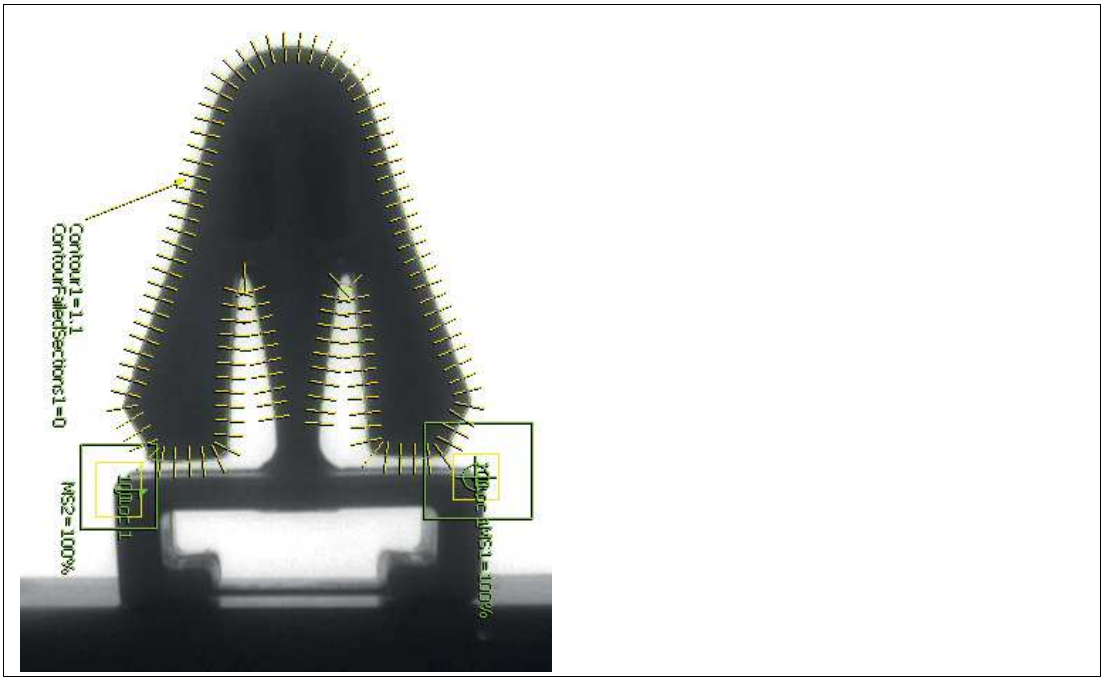


Figure 7.53 Example: identifying faults on a mold

The figure below shows faulty sections of a workpiece in which the edge points are not within the user tolerance due to an unwanted notch in the part. In some cases, a certain degree of tolerance may be permitted until a limit of successive "faulty sections" is reached.

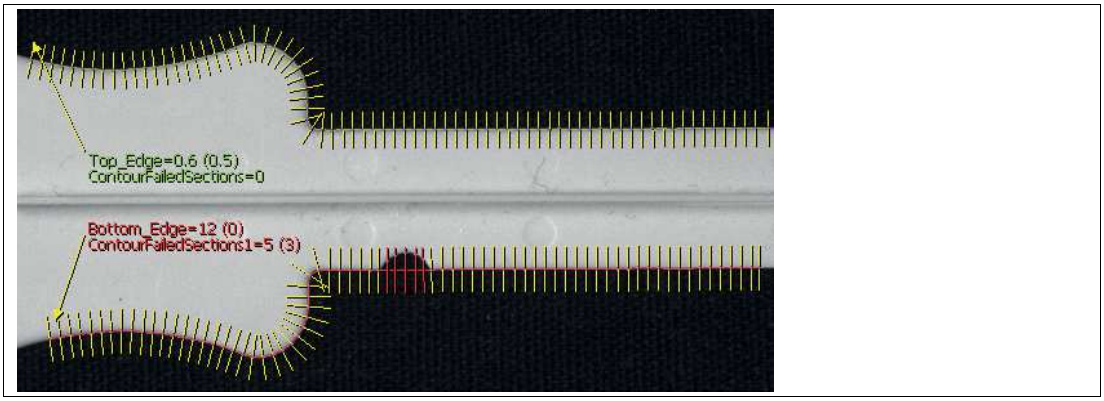


Figure 7.54 Example: unwanted notches in the workpiece

7.3.12 Angle Tool

The Angle tool is used to measure the angle between two objects. Supported objects include image borders, image edges, and pencil lines. For an accurate measurement, both edges within the search areas should be straight at both ends.

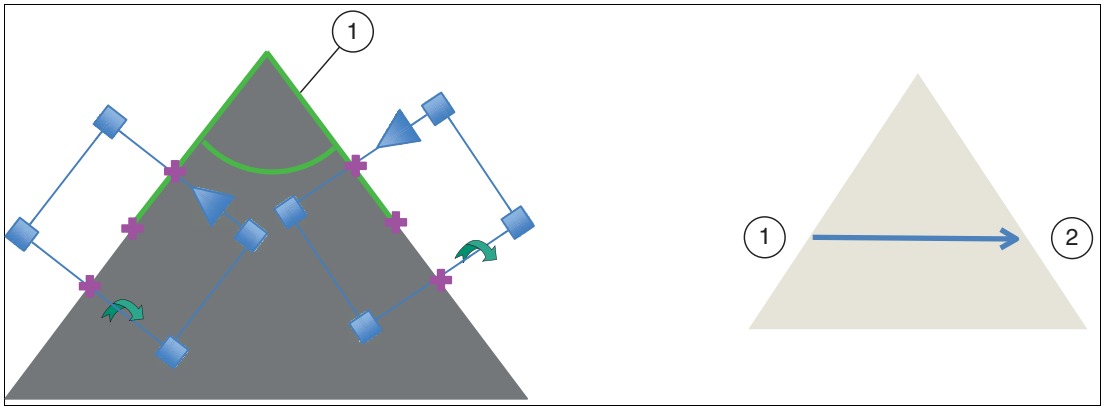


Figure 7.55 Angle tool

Circle Tool Information	
Search area shapes	Rectangle (border search area at each end)
Tool preparation	One click point at each angle edge
Measurement	Angle (in degrees) between two edges
Scan direction	With or against the arrow, in the search field
Edge polarity	Light to dark, dark to light, optional
Selection of edge	Thickest or first edge
Edge transition	Automatically with slider for sensitivity
Resolution	2 to 50 edge points within the search area

The Angle tool is used to measure the cutting accuracy of each corner of a metal plate. An angle of 90° (± 0.5°) is expected for each corner. The following example shows that the bottom left corner (1) is out of tolerance.

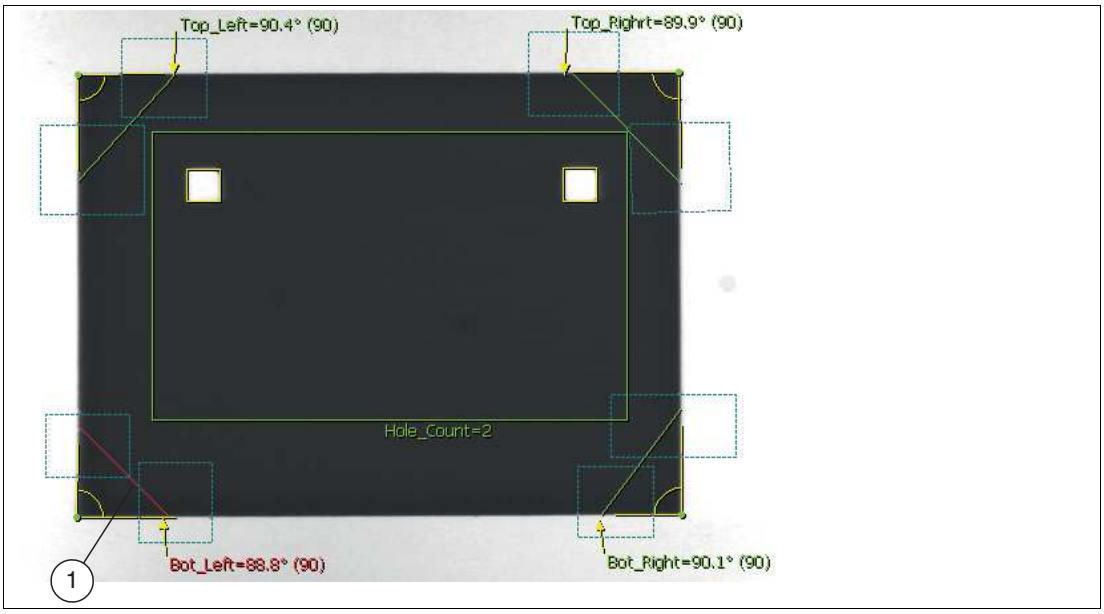


Figure 7.56 Example: cutting accuracy at each corner of a metal plate

The Angle tool is also useful for locating the intersection (corner) of two edges. This point can be used as a reference point in applications where the defining edges are straight and the part movements are small.

The figure below shows the bottom right corner point (1), which is used as a reference point for orientating other tools.

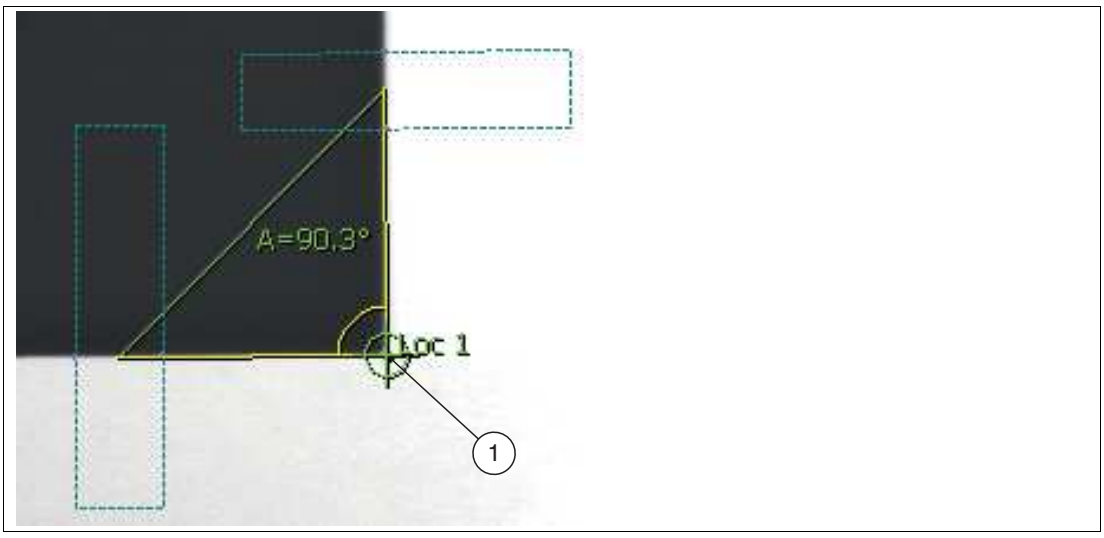


Figure 7.57 Reference point for orientating other tools

7.3.13 Arc Tool

The Arc Tool is a version of the Circle Tool that is used to measure the radius (1) and center position (2) of an arc (3). Arcs with a larger angle provide more accurate results. Select arcs of at least 90°.

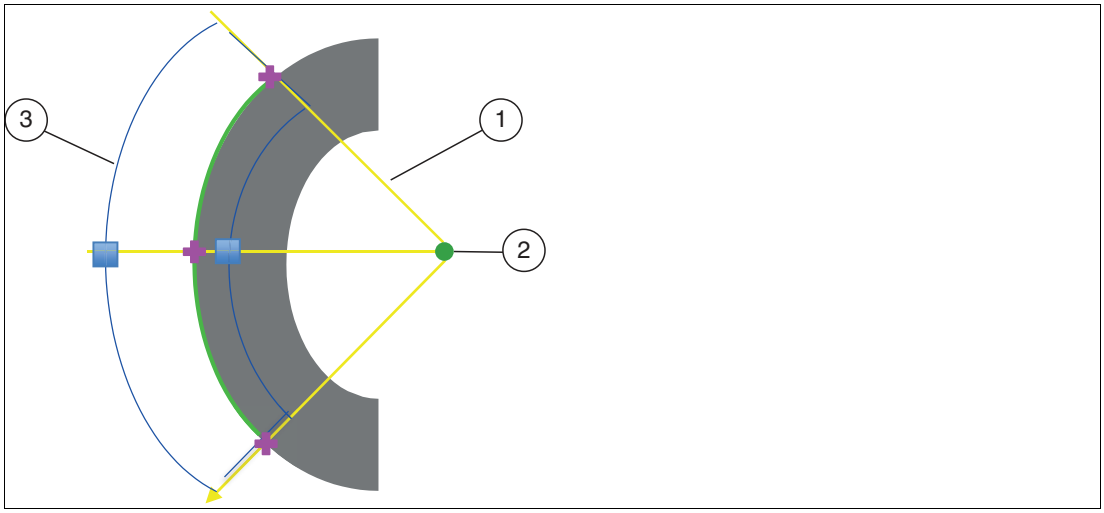


Figure 7.58 Arc Tool

Arc Tool Information	
Search area shapes	Arc (half annulus)
Tool preparation	Three click points at any point along the edge of the circle
Measurement	Diameter, roundness; minimum, maximum, standard deviation of the radius
Number of spokes	Maximum 720
Scan direction	Selectable, from inside to outside or from outside to inside
Edge polarity	Light to dark, dark to light, optional
Selection of edge	Thickest or first edge
Edge transition	Automatically with slider for sensitivity
Outlier support	Yes

As with the Circle Tool, the outlier control can be used in the Arc Tool to eliminate small faults along the arc edge as shown below:

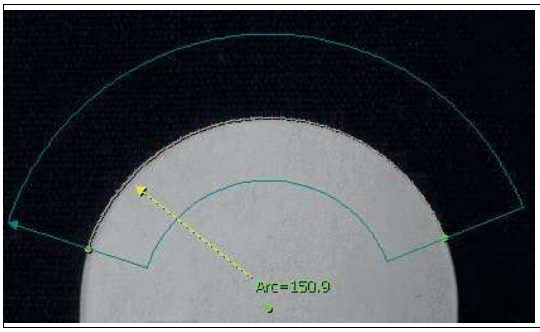


Figure 7.59 Arc = 150.9°

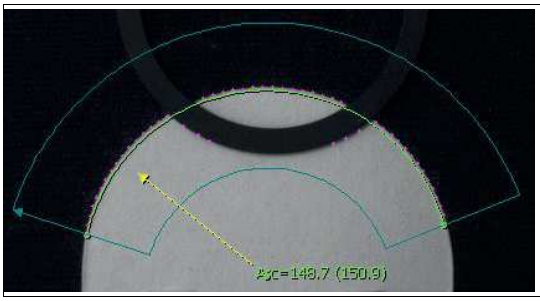


Figure 7.60 Arc = 148.7° without outlier control

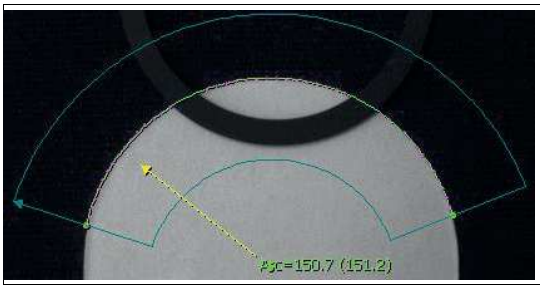


Figure 7.61 Arc = 150.7° with outlier control

7.3.14 Circle Tool

The circle tool is used to determine the diameter and the center position of a circle. The circle tool looks for edges (1) along the spokes (2) within an adjustable, circular search area (4). The more spokes used, the better the accuracy of the measurement. This results in a decreasing processing speed. The tool creates a "best fit" circle (3) from the edge points (1) to determine the diameter.

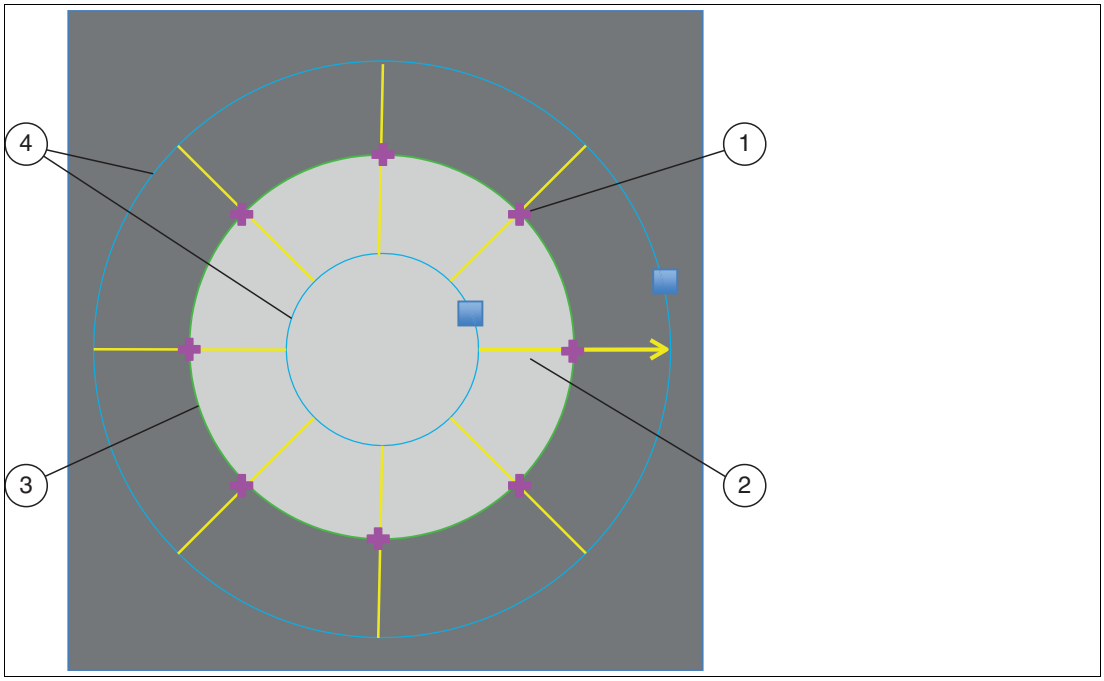


Figure 7.62 Circle tool

Circle Tool Information	
Search area shapes	Ring
Tool preparation	Three click points at any point along the edge of the circle
Measurement	Diameter, roundness; minimum, maximum, default diameter
Number of spokes	Maximum 720
Scan direction	Selectable, from inside to outside or from outside to inside
Edge polarity	Light to dark, dark to light, optional
Selection of edge	Thickest or first edge
Edge transition	Automatically with slider for sensitivity
Outlier support	Yes

The circle tool can measure an incomplete or damaged circle. This is done with the aid of outlier suppression. In the bottom left of the figure below, the cut-out section (2) causes some edge points (the purple points [3]) to be included in the calculations of the circle, which results in the diameter and the center (the green point [4]) being moved to a minimal degree. By using a 20% outlier suppression parameter (bottom right image [1]), both the diameter and the center are now correct.

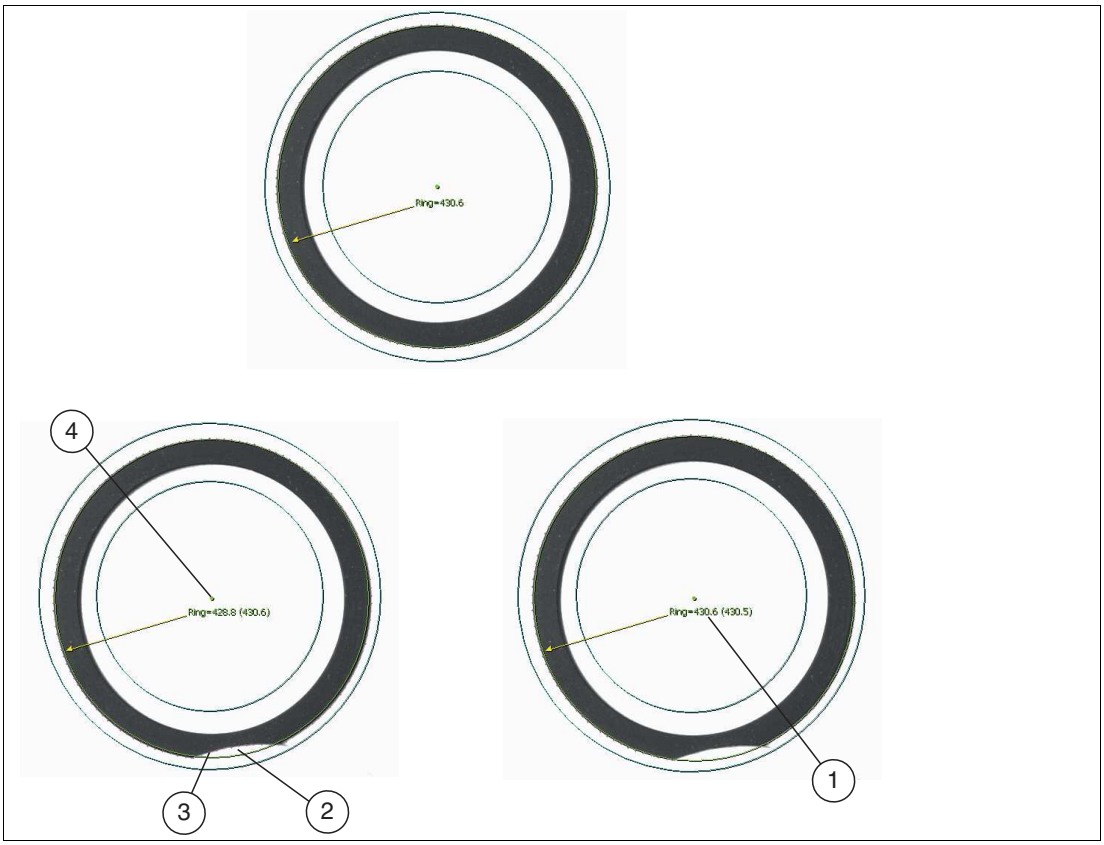


Figure 7.63 Example: Incomplete or damaged ring with outlier suppression

7.3.15 Concentric Circle Tool

The concentric circle tool is used to measure the concentricity of two circles. The degree of concentricity is the maximum distance from one of the two circles to the average position of the two circle centers (1 and 2). The concentric circle tool allows you to measure (at regular intervals) the widths of the ring formed by the two circles.

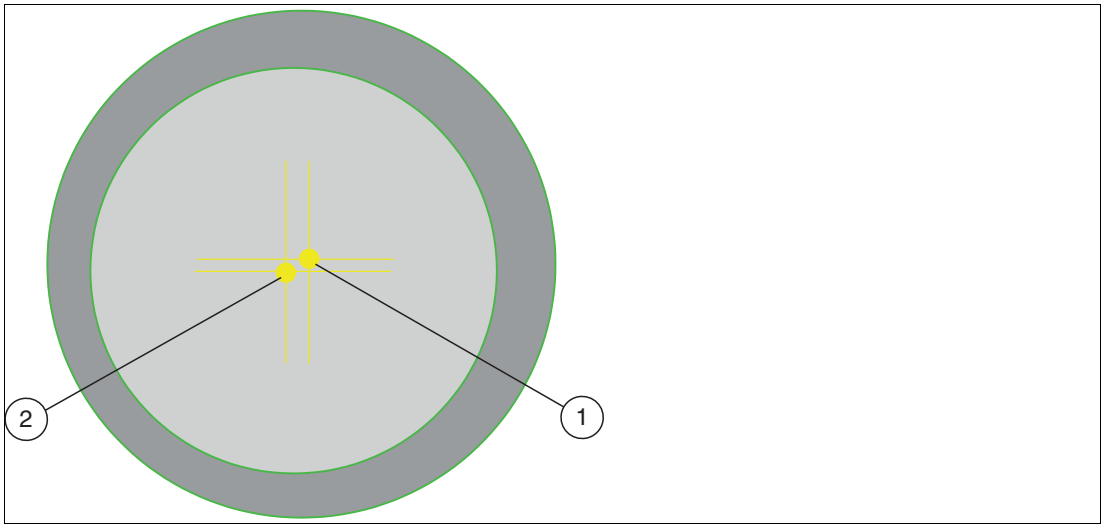


Figure 7.64 Concentric circle tool

Information About the Concentric Circle Tool	
Search area shapes	References of two defined circle tools
Tool preparation	Two mouse clicks, one on each circle outline
Measurement	Average, minimum, maximum, and standard deviation between the positions of the circle centers; Width of the area between the two circles

In the following examples, the concentric circle tool is used to check whether the inner and outer walls are correctly aligned. The image on the right shows the exact course of a paint can in the direction of a filling station. If the outer wall is damaged, the concentric measurement will fail and the can will be rejected.

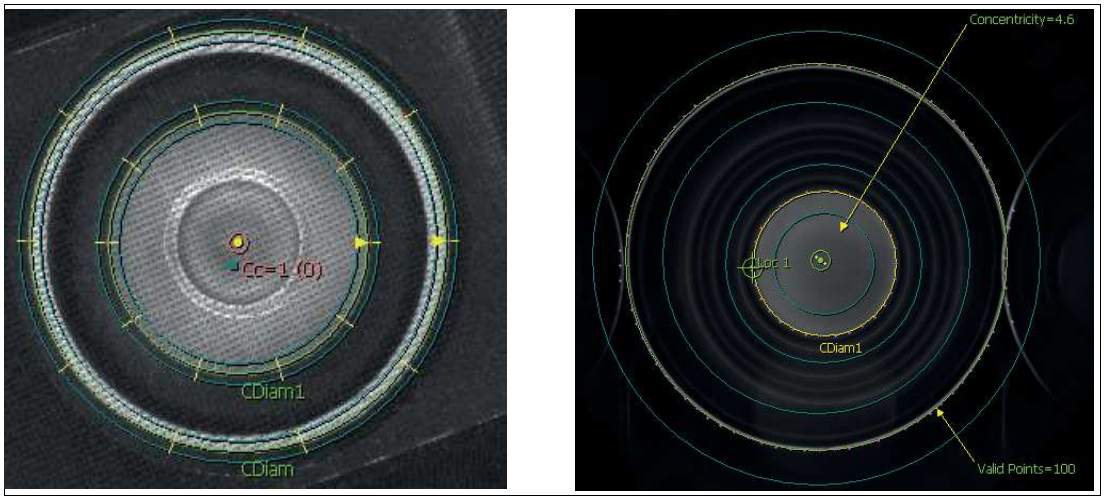


Figure 7.65 Example: concentric circles

7.3.16 Preprocessing Filter (Preprocess)

To improve specific properties of an image, preprocessing filters can be used before a tool is applied. By applying one or more preprocessing steps, features that are difficult or impossible to measure can be simplified.

Use the selection lists to select up to five different preprocessing filters for the selected search field. Multiple preprocessing filters are applied in sequence, from left to right. Some preprocessing filters have parameters that you can change. The fields below the selection list have names and default values. The fields remain blank if there are no modifiable parameters.

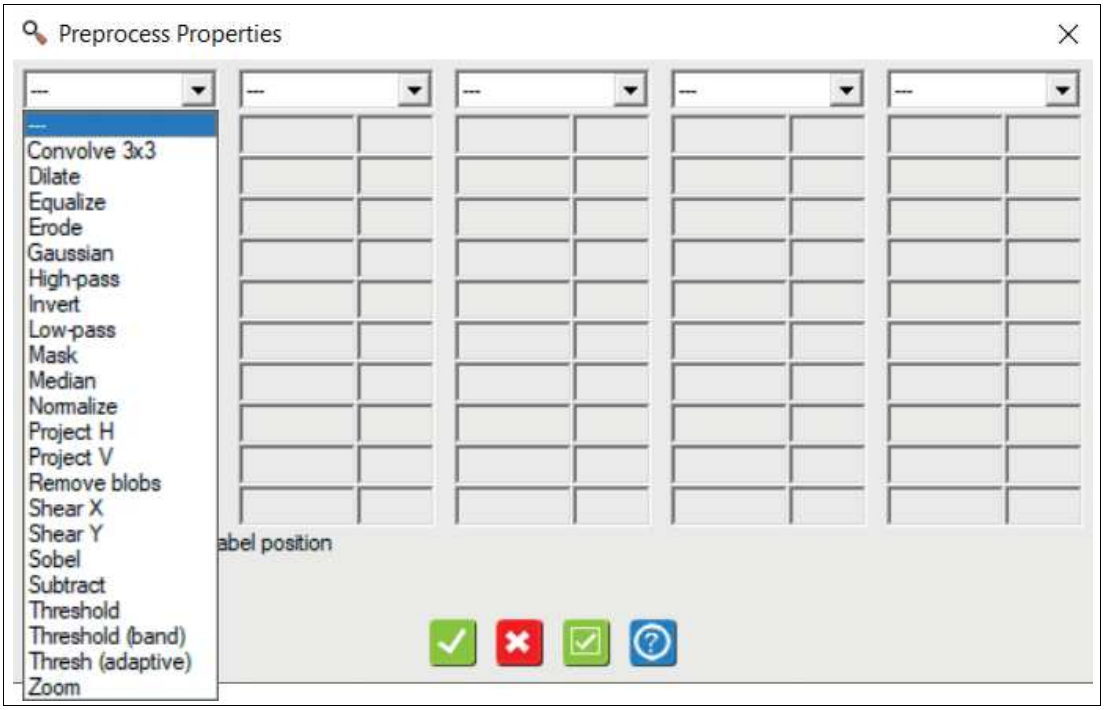


Figure 7.66 Screenshot—Preprocess filter

Note

Avoid excessive use of preprocess filters. First, try adjusting the image using the lighting and the optics (lens). Preprocessing modifies the original image data. It also extends the test time.

Convolve 3x3

The Convolve operator is the basic neighboring operator on which other operators such as low pass and high pass are based. The Convolve 3x3 operator can be programmed to perform various neighboring functions. The following is an example of using convolve functions to perform separate horizontal and vertical edge enhancement.

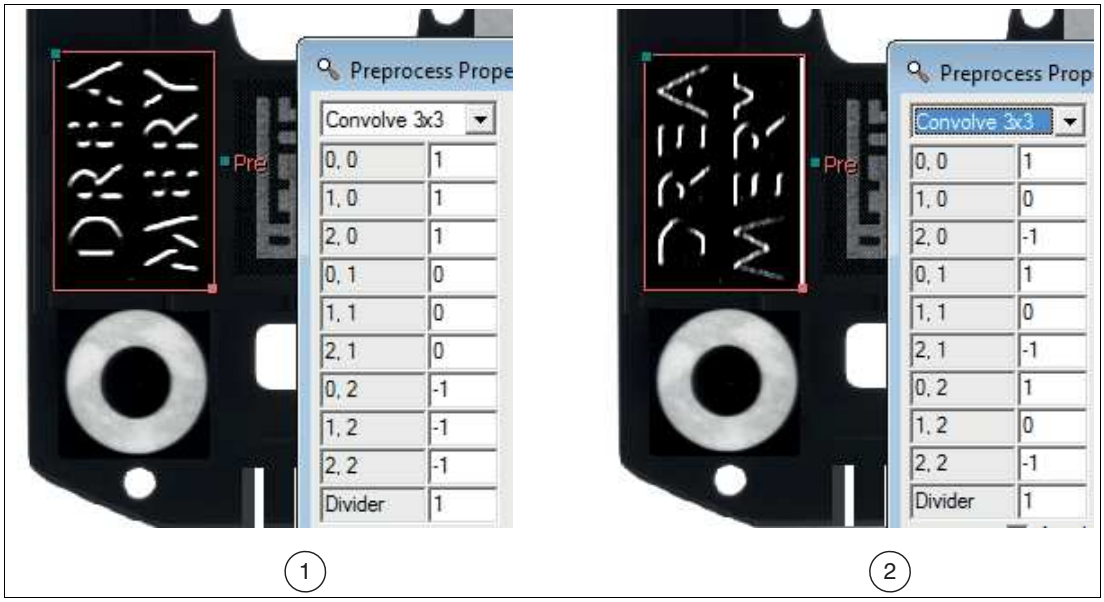


Figure 7.67 Convolve operator

1. Horizontal edges, light over dark
2. Vertical edges, light to dark

Dilate Function

The Dilate operator expands the lighter pixels. It is useful for combining small elements that are lighter than their surroundings.

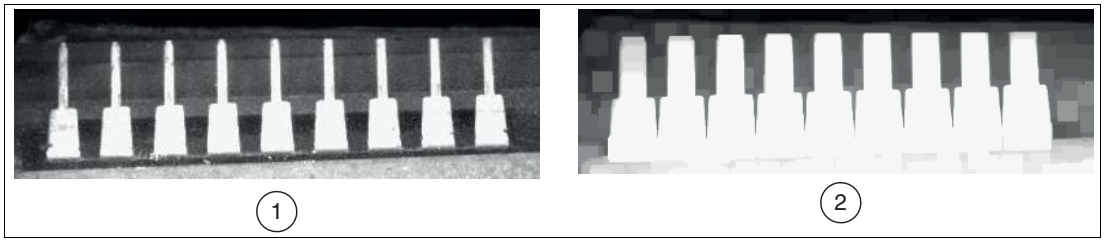


Figure 7.68 Dilate function

1. Original image
2. Expanded image (all pins grouped)

Equalize

This function extends the image intensity to cover the entire range from 0 to 255. The darkest pixels in the area are set to 0. The lightest pixels in the area are set to 255. Other pixels are set based on statistical standardization of the intensity histogram.

Erode

Erode is the opposite of the Dilate function. It expands the darker pixels. It is useful for resolving features that are connected by a relatively thin element.

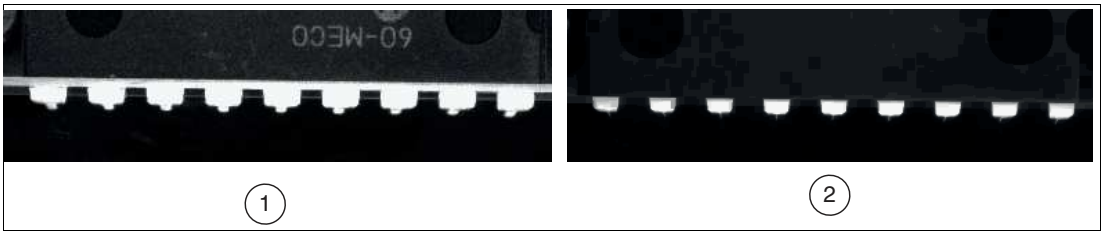


Figure 7.69 Eroding

1. Original image
2. Eroded image (pins are separated and smaller)

To obtain the size of the features after an eroding operation, a dilation operation can be performed:

Figure 7.70 Additional dilation operation

Gaussian Filter

This function uses a Gaussian filter to smooth image intensities. Gaussian filtering blurs details and removes structures with high spatial frequency from images. The Gaussian filter is used to reduce noise or to smooth edges.

High-Pass

A high-pass eliminates the effect of constantly changing intensity on an image. The function is so-called because it only allows for high-frequency fluctuations.

In the following example, it is difficult to detect all printed characters against the background when using a Count Tool due to the curvature of the teapot and the uneven lighting. Changing the sensitivity of the Count Tool merely allows you to discern different parts of the teapot.

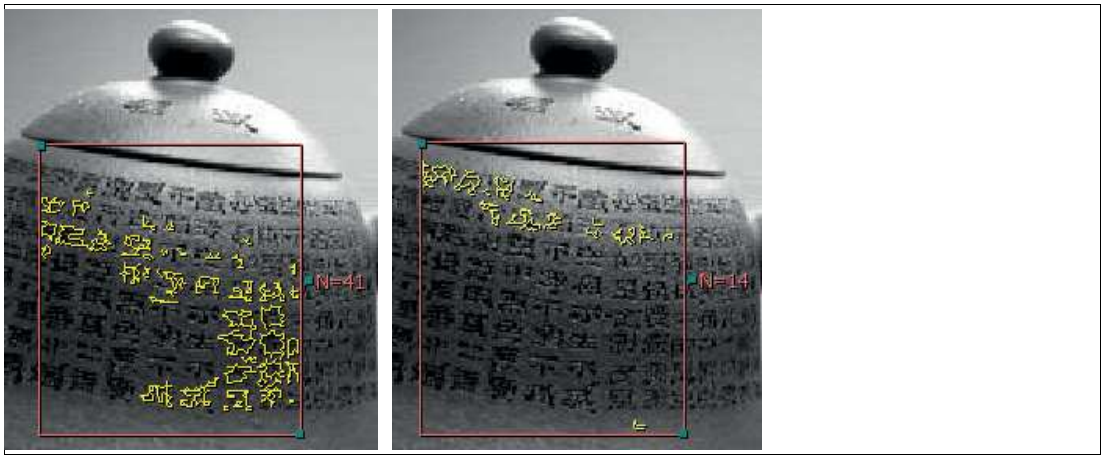


Figure 7.71 Example: Count Tool

The following example shows how to use the high-pass function to filter out all characters.

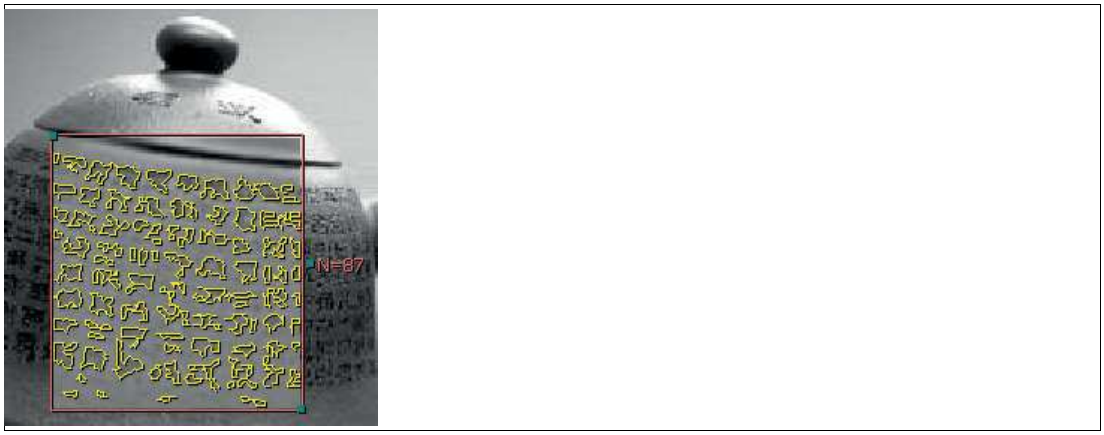


Figure 7.72 Example: high-pass

Invert

Reverses black and white (to produce a "negative" image). Each pixel is replaced by 255 minus the value of the pixel. (255 - input pixels) = new pixel

Low-Pass

A low-pass function has a smoothing or blurring effect on an image. The function is so-called because it only allows for low-frequency fluctuations. The low-pass function is useful for removing closely spaced irregularities in the intensity caused by electronic noise in the camera or by texture fluctuations on the actual test object.

In the following example, a caliper tool is used to determine the distance between the top and bottom of a teapot lid. When you apply the tool to the image on the left, the caliper fails due to the imprint on the side (i.e., it detects high-frequency edges irrelevant to the measurement). For the algorithm, the imprint appears as randomly distributed noise. Applying the low-pass filter function results in the image on the right. The caliper tool can detect the edges to be measured.

Note

Make sure you take into account the offset from the actual position caused by the filter. This offset would be consistent from pattern to pattern.

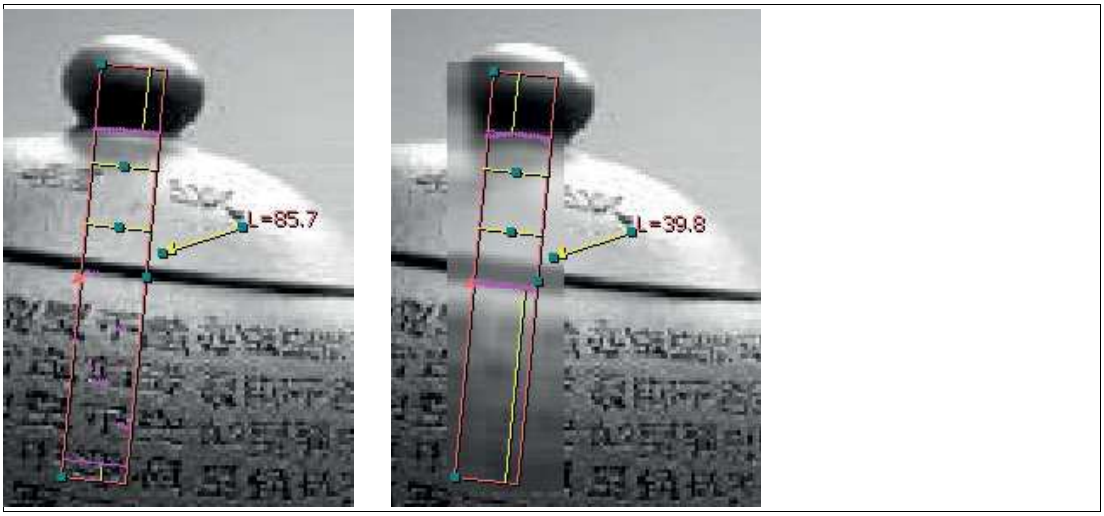


Figure 7.73 Example: low-pass

Mask

Draw a pixel search field. Use mask on a digital image (threshold result).

- **Fill value**: A pixel value for the drawn mask.
- **Reverse**:
  - 0 (false) draws the white areas (pixel value not equal to 0) with the fill value.
  - 1 (correct) draws the black areas (pixel value equal to 0) with the fill value.

Median

The median setting replaces a pixel with the median value of its neighbors. The median setting is similar to the low-pass function in that it smoothes an image. The median setting is better suited for removing peak noise than the low-pass function.

Original image

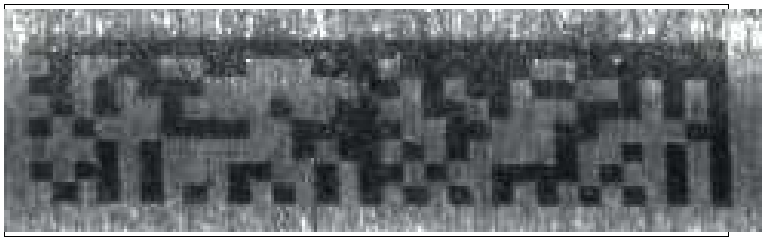


Figure 7.74 The original image

Low pass

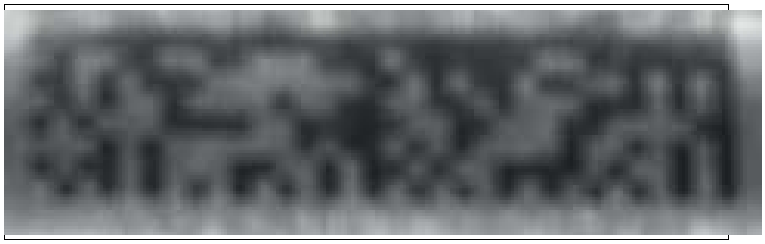


Figure 7.75 Low pass (the background texture is suppressed, but the edges become blurred)

Median

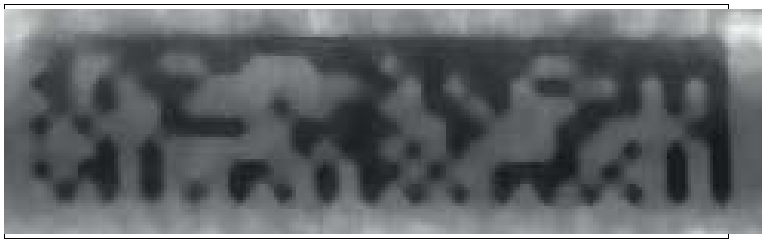


Figure 7.76 Median (the edges are sharper)

Normalize

This function removes the min% and max% values and adjusts (scales) the pixel intensities to cover the entire range from 0 to 255. The min% and max% parameters are a percentage of the lowest and highest values to be ignored in the calculation.

Project H

The project H operator replaces each pixel with the average pixel value of the horizontal line on which the pixel is located. This can be useful for ignoring minor deviations in the vertical direction.

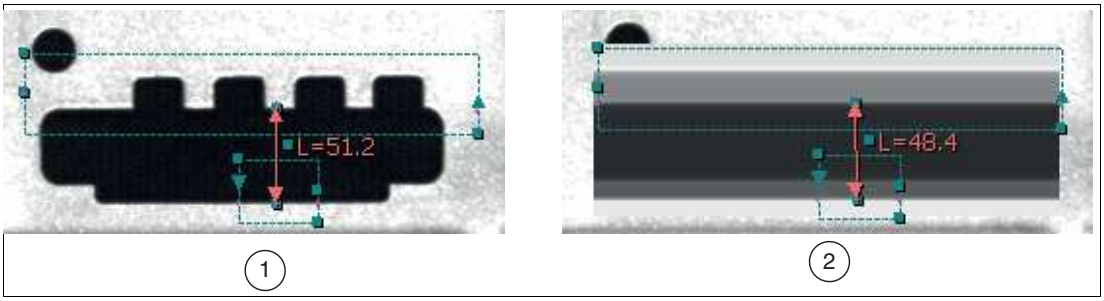


Figure 7.77 Project H

1. Original image
2. Project H image (small vertical projections are ignored)

Project V

Project V is similar to project H, except that the projection is in the vertical direction.

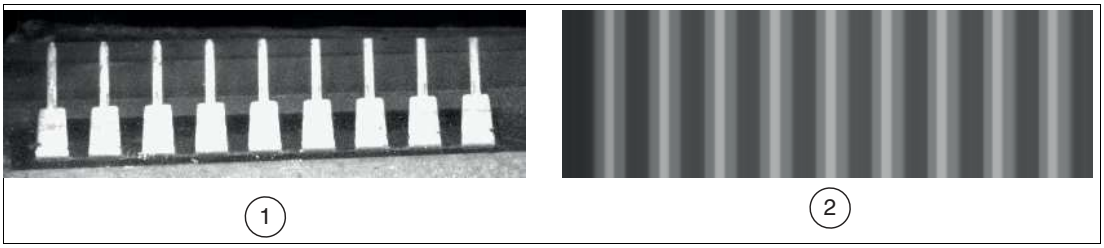


Figure 7.78

1. Original image
2. Project V image (all pins grouped)

Removes Blobs

Removes objects in the search area.

- **White blobs** = 1 (true) remove white blobs, 0 (false) remove dark blobs.
- **8 ways** = 1 (true) 8-way connection, 0 (false) 4-way connection.
- **Rm edge blobs** = 1 (true) removes all blobs that touch the search area contour, 0 (false) does not automatically remove blobs that touch the search area contour.
- **Limit value type** = 1 (inside) removes objects that are within the size limits, 0 (outside) removes objects that are outside the size limits.
- **Min/max** = all minimum and maximum size limits are specified in pixels.

Shear X

Shear horizontally. Positive angle shears clockwise (obliquely to the right). Parameters: shear angle in degrees, interpolation method to supplement new/missing pixels, background mode, how the background is to be filled: 0 = automatic, 1 = use background value, 2 = none, background value a pixel intensity. See interpolation method and background filling.

Shear Y

Shear vertically. Positive angle shears clockwise (obliquely downward). Parameters: shear angle in degrees, interpolation method to supplement new/missing pixels, background mode, how the background is to be filled: 0 = automatic, 1 = use background value, 2 = none, background value a pixel intensity. See interpolation method and background filling.

Sobel

The Sobel operator is used to improve the horizontal and vertical edges.

Figure 7.79 Sobel operator

1. Original image
2. Improved image

Subtract

Subtracts the values of the pixels in the captured runtime image from the values of the pixels in the template image. Since the possible range is -255 to +255, we add +128 to each pixel. All resulting values below 0 are set to 0. All resulting values above 255 are set to 255.

Threshold

The Threshold operator converts a grayscale image to a digital image. Pixels above a threshold value become white and pixels below the threshold value become black.

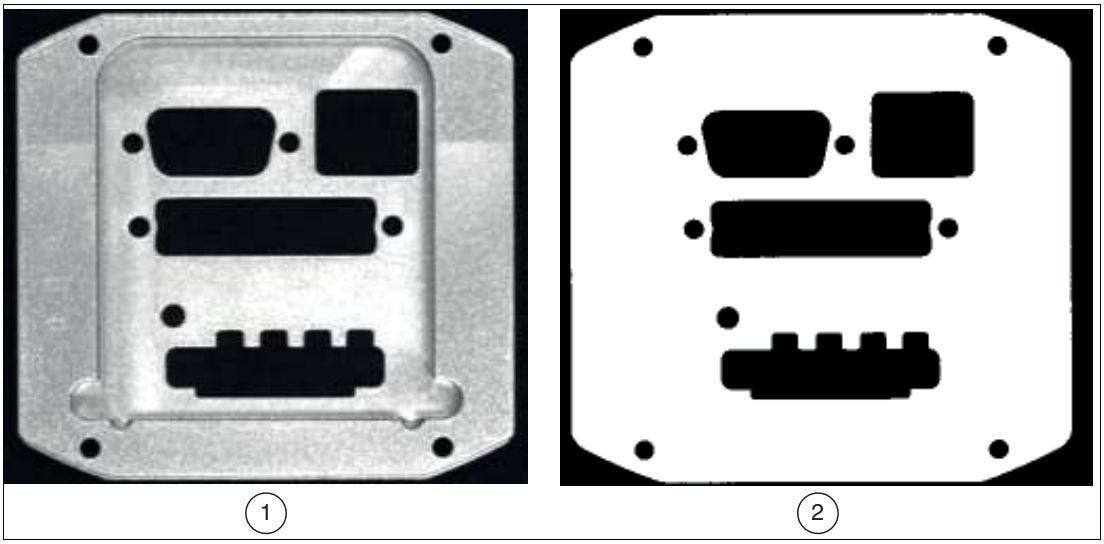


Figure 7.80 Grayscale image into a digital image

1. Original image
2. Threshold image (weak deviations are removed)

Threshold (Band)

Generates a binary program code consisting of ones and zeros. All pixels with values less than Low and greater than High are changed to 0 (black). All pixels with values equal to Low, equal to High, and all pixels with values between Low and High are changed to 255 (white).

Adaptive Threshold

The Adaptive Threshold operator is the same as a high-pass operator followed by a threshold operator.

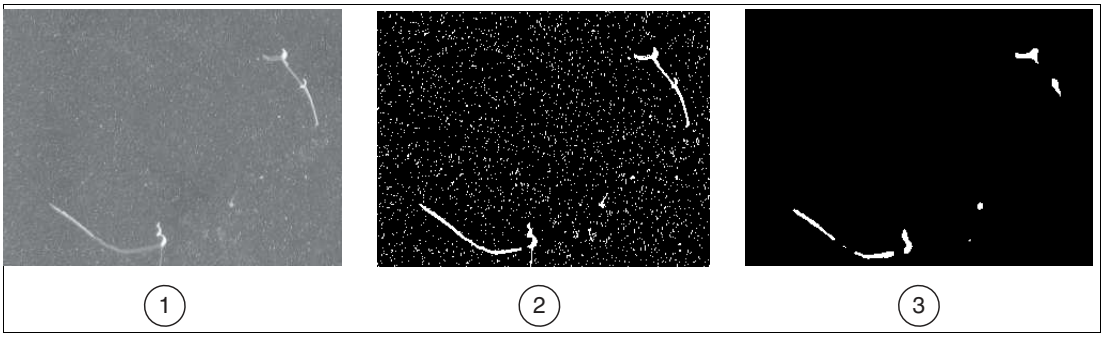


Figure 7.81 Adaptive Threshold operator

1. Original image
2. Adaptive threshold
3. After median filter

Zoom

Enlarges ("zoom in") or shrinks ("zoom out") the image in the area by the X-scale multipliers (0.001 to 1000) and Y-scale multipliers (0.001 to 1000).

7.3.17 1-D Code (Barcode Tool)

The 1-D code tool is used to read the most common barcodes. For codes that are detected, refer to the tables below in this section.

Multiple codes can be detected in a search field (multiple code reading) with optional output of the code position in the image. Alternatively, you can span multiple search fields in the image to read codes at predefined positions. Enable the metric output to monitor the print quality of the code.

Figure 7.82 1-D code tool

Reading Codes

To enable reliable implementation of your application, the codes must have good image quality, contrast, and the right resolution to be read.

Observe the following instructions for reading codes with the sensor.

- Select the operating distance between the sensor and the code using the read field diagrams above. To do so, the **"module size"** must be known, i.e., the width of the narrowest bar (for 1-D codes) or the side length of the square for Data Matrix codes (2-D codes). If the operating distance selected is too large, reading is either unreliable or not possible.
- The values specified in the read field diagrams are maximum values for a specific module size. For a specific module size, the operating distance must be less than or equal to the specified distance.
- Adjust the operating distance and the focus so that the bars or squares in the image are clearly distinguishable. If the bars or squares blur together, the images cannot be read correctly.
- Ensure that the code to be read is completely within the field of view.
- Ensure uniform lighting of the code to be read. Inhomogeneous code lighting may lead to incorrect readings.
- During parameterization, observe the information in the following description for the 1-D code tool.
- Test the settings and make sure that your settings enable process-safe reading. We recommend trying out multiple codes when testing your application. Consider possible changes in lighting conditions as well as the situation in the test.

Using the 1-D Code Tool

1. Click the "1-D Code Tool" button in the tool selection area.

   ↳ Move the mouse over the image in the work area.

2. Left-click on your starting point. Release the left mouse button and move the mouse to the area that represents your search area. Left-click again to draw a rectangular search area on the image.

   ↳ The selected field is labeled (Bar, Bar1, etc.).

3. Resize the search area by clicking and dragging one of the corners of the search area. The maximum size of the search area for 1-D codes is 2048 x 2048 pixels

Note

The default orientation or code direction is left to right (zero degrees) for horizontal codes. Codes at other angles (e.g., 90°, 180°, 270°) may not be detected immediately by the default settings. As described below, you can use the 1-D Code Properties tab to change the "Orientation" to automatic or to a specific angle.

4. Right-click on an edge of the rectangular search area.

   ↳ The properties window for the 1-D code tool will open.

1-D Code Tool Properties Window

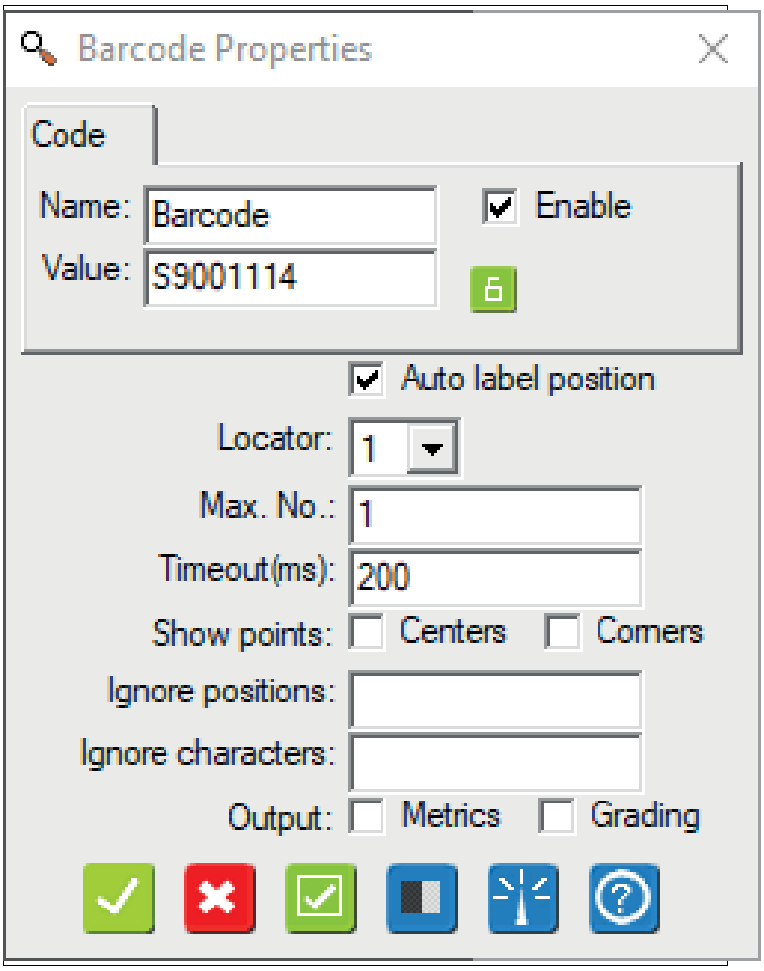


Figure 7.83 Properties window

2021-03

PEPPERL+FUCHS

Output Variable

Tab	Description
Code	The tool will operate differently depending on the mode set. • Read code: The code is read and the content can be output. • Compare code: The code is read and a check is carried out to see whether the content matches the specified content. You can enter a fixed value under "Value."

Output Variable Settings

Setting	Description
Name	You can change the variable name of the tool or the measurement. The variable name is displayed in the image, in the tool list, and in the measurement results. The name is used as a variable in scripts.
Enable	Enables the data output of the relevant output variable.
Value	In this field, enter the character string to be compared with the content of the current barcode. The result of the tool is "passed" when the code is read and the read character string matches the set character string. Otherwise, the result is "failed." Leave an asterisk in this field if the 1-D code changes. The measurement reports any read value. It only fails if no value can be read. "?" = placeholder for a single character "*" = placeholder for multi-digit characters. The "padlock" allows you to lock the value, so that it does not change while you move the search field. The value should be unlocked while you change the parameter settings.

Parameters

Parameter	Description
Auto label position	Sets the name field next to the area in the display. Clear the check box to move the name field to another position, or drag the name field to the image with the property field closed.
Locator (anchor point)	This function causes the search field of the tool to follow the test object as it moves. Select a locator (by number) that you want to follow. To detach the tool from the locator or to hold the tool in a fixed position in the image area, select X. You can define up to four locators.
Max. No.	Number of codes expected in the search field. Increase this number if you have more than one code in your search field. You can define a search field with multiple codes or multiple search fields each with one code. The maximum number is 255. Larger numbers result in an error message.
Timeout	This function sets a timeout (in milliseconds) for complicated, incorrect, or non-existent codes. Note: This timeout refers to the decoding or code reading algorithm, but does not include the preprocessing time.
Show points	This function displays active points and allows you to use them as an anchor point for other tools or to output the point coordinates in the image.
Ignore positions	This function skips or ignores character positions in the character string. You can ignore unimportant, unrelated, or changing positions. Enter position numbers separated by spaces. Use 1 for the first character position (not 0), use 2 for the second character, and so on. Use -1 for the last character, use -2 for the second-last character.

Parameter	Description
Ignore characters	This function skips or ignores characters that are partially added to some barcodes. Example: If the first position and the $ characters are ignored, a decoded string of 1234 will have a result value of 1234.
Extended outputs (Output)	Output of the code quality according to ISO. The output enables additional output data. Metric data: Indicates the code symbology detected and the number of read codes. Quality Assessment (Grading): Provides information about readability and print quality.

Buttons

Button	Description
OK	The changes are applied and the editing window is closed.
Cancel	The editing window closes without any changes being made.
Apply	The changes are applied without closing the editing window.
Preprocess	This function allows you to add image filtering to the tool. The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.
Advanced	These settings are optional. They allow more control over the object and the runtime. You can zoom in and out on areas and hide them. Hiding areas allows you to cut them from your search area by using a brush to turn those areas red. This function increases the test time.
Help	This opens the help window. Here you will find useful operational information.

Editing the 1-D Code Tool

1. Right-click on an edge of the rectangular search area.

   ↳ This opens the editing field for the 1-D code tool.

2. Click the "Preprocessing filter" button to add an image filter to the tool.

   Note

   The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.

3. Click the "padlock" button next to the value to store a fixed value.
4. Click the "OK" button.

   ↳ The changes are applied and the editing field closes.

5. Click the "Cancel" button.

   ↳ The editing field closes without any changes being made.

6. Click the "Apply" button.

   ↳ The changes are applied without closing the editing field.

7. Click the "Help" button.

   ↳ This opens the help window. Here you will find useful operational information.

Advanced Settings

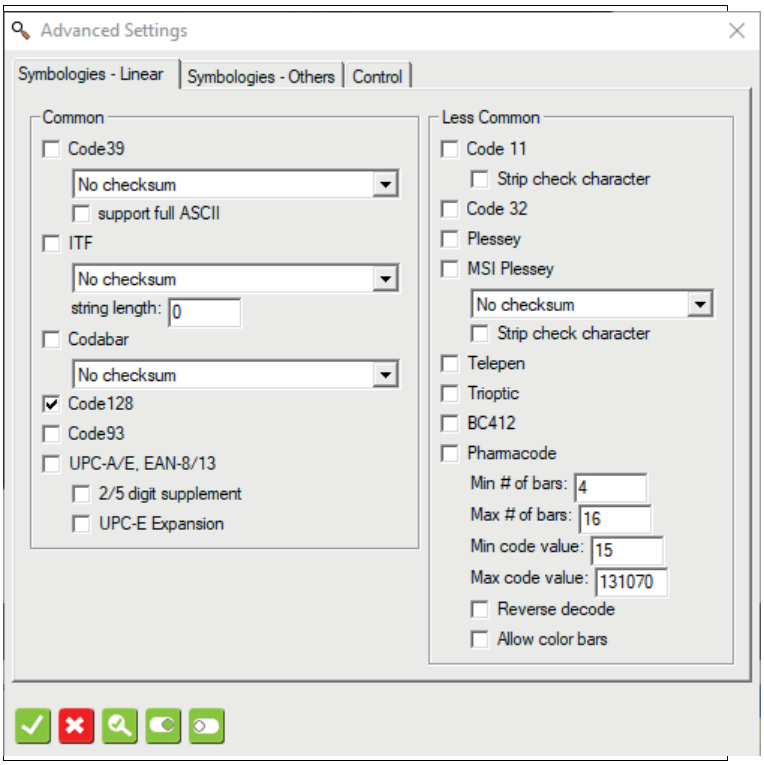
Figure 7.84 Menu bar

The buttons are the same for all tabs

Item	Button	Description
1	OK	Changes are applied and the editing field closed.
2	Cancel	The editing window closes without any changes being made.
3	Identify	The code type is automatically identified and selected in the search field.
4	Select all	All code types or symbologies are enabled on both tabs.
5	Deselect all	All code types are disabled on both tabs.

Tab Description

Symbologies - Linear

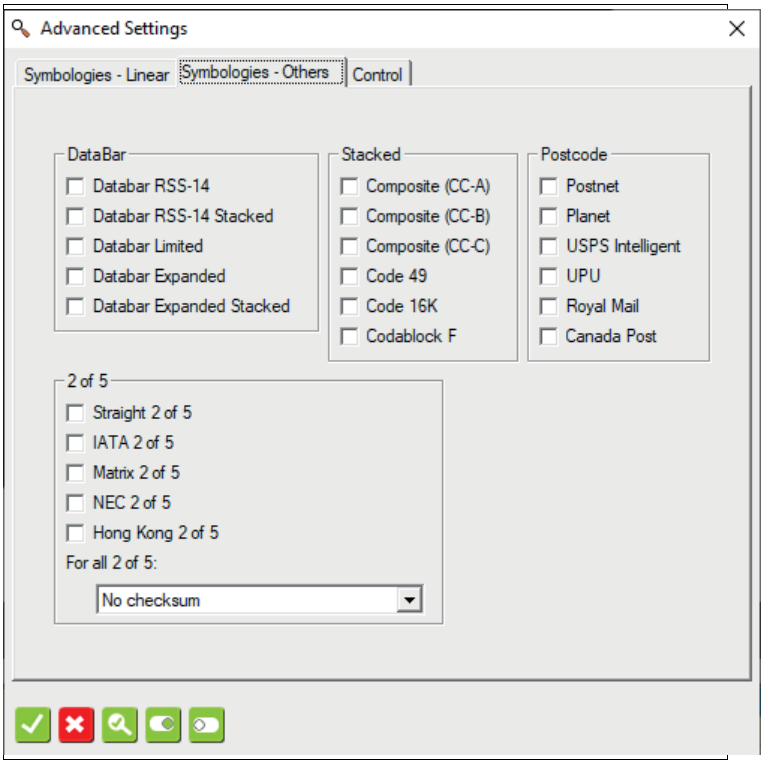


Menu	Description
Code 39	Enables Code 39 codes to be decoded. The following checksum options are available for selection: • **No checksum**: Disables checksum calculation. • **Use checksum, output check character**: Calculates the checksum and includes the check character in the output of the read character string. • **Use checksum, strip check character**: Calculates the checksum and does **not** include the check character in the output of the read character string. **Support full ASCII**: Enable or disable full ASCII character support for Code 39. If disabled, the output is numeric.
ITF	Enables ITF (Interleaved 2 of 5) to be decoded The following checksum options are available for selection: • **No checksum**: Disables checksum calculation. • **Use checksum, output check character**: Calculates the checksum and includes the check character in the output of the read character string. • **Use checksum, strip check character**: Calculates the checksum and does **not** include the check character in the output of the read character string. **String length**: Sets a minimum string length and controls the check of the quiet zone on ITF. • **0**: The default quiet zone check is used and no check is performed on the code length. • **1**: A short quiet zone is allowed and no length check is performed. • **event number**: The default quiet zone is used and the string to be decoded must be at least N-long to be decoded. • **odd number larger than 1**: A short quiet zone is allowed and the decoding string must be at least N-1 long to be decoded.

2021-03

Menu	Description
Codabar	Enables Codabar codes to be decoded. The following checksum options are available for selection: • **No checksum**: Disables checksum calculation. • **Use checksum, output check character**: Calculates the checksum and includes the check character in the output of the read character string. • **Use checksum, strip check character**: Calculates the checksum and does **not** include the check character in the output of the read character string.
Code 128	Enables Code 128 codes to be decoded.
Code 93	Enables Code 93 codes to be decoded.
UPC-A/E, EAN-8/13	Enables UPC-A, UPC-E, EAN-8, and EAN-13 codes to be decoded. **2/5 digit supplement**: Enables 2 or 5 digit additional code for UPC and EAN codes. **UPC-E Expansion**: Enables UPC-E expansion.
Code 11	Enables Code 11 codes to be decoded. **Strip check character**: Clears the checksum character from Code 11.
Code 32	Enables Code 32 codes to be decoded.
Plessey	Enables Plessey codes to be decoded.
MSI Plessey	Enables MSI Plessey codes to be decoded. The following checksum options are available for selection: • **No checksum**: Disables checksum calculation. • **Use Mod 10 checksum** • **Use Mod 10/10 checksum** • **Use Mod 11/10 checksum** **Strip check character**: Clears the checksum character from MSI Plessey codes.
Telepen	Enables Telepen codes to be decoded.
Trioptic	Enables Tripotic codes to be decoded.
BC412	Enables BC412 codes to be decoded.
Pharmacode	Enables Pharmacode codes to be decoded. **Min # bars** Sets the minimum number of bars in the Pharma Code. **Max # bars** Sets the maximum number of bars in the Pharma Code. **Min code value** Sets the minimum numeric code value. **Max code value** Sets the maximum numeric code value. **Reverse decode** Enables decoding of reversed images or reverse decoding. **Allow color bars** Allows color codes to be decoded.

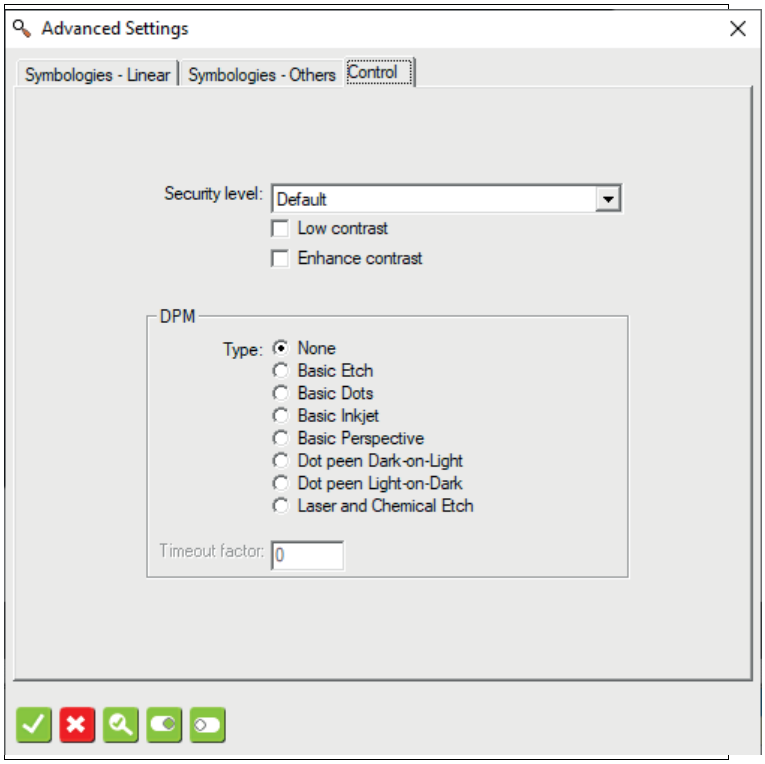
Symbologies - Others



Select the types of codes you want to read or decode. You can make multiple selections if your code types vary. This may extend the inspection time.

Menu	Description
DataBar	Select which types of DataBar codes to decode.
Stacked	Select which types of stacked codes to decode.
Postcode	Select which types of postal codes to decode.
2 of 5	Select which types of codes to decode. The following checksum options are available for selection: • **No checksum**: Disables checksum calculation. • **Use checksum, output check character**: Calculates the checksum and includes the check character in the output of the read character string. • **Use checksum, strip check character**: Calculates the checksum and does **not** include the check character in the output of the read character string.

Control



Menu	Description
Security level	Selects the read or decode aggressiveness to be used. A higher aggressiveness means "more aggressive read or decode attempts." If the print quality is poor or the resolution is low (code too small), the number of misread or incorrectly decoded codes may increase and the execution time may be extended. Reducing aggressiveness reduces the number of misreadings or incorrect decodings of poor-quality or low-resolution codes. Applies to: Code 128, Code 39, UPC/EAN, ITF, Codabar, Code 93.
Low contrast	Enables codes with low contrast or reverse contrast to be decoded. This may increase the execution time. Applies to: Code 128, Code 39, UPC/EAN, ITF, Codabar, Code 93.

Menu	Description
Enhance contrast	Enables you to improve the image to decode any poorly printed codes. This may increase the execution time. Cannot be used with Dot peen Dark-on-Light, Dot peen Light-on-Dark, or Laser and Chemical Etch.
DPM Type	Selects image conditioning options specifically for direct part marking (DPM). Note: For all DPM options, the maximum search field size is 614,000 pixels. • **None**: Does not allow DPM-specific processing. • **Basic Etch**: Optimized for simple laser or chemical etching. • **Basic Dots**: Point-jet and inkjet images • **Basic Inkjet**: Basic inkjet images and poor-quality inkjet images. • **Basic Perspective**: Reads laser markings and normal codes of good quality that are centered in the image but have strong perspective distortion. This mode is faster but less robust than "Basic Etch" for reading poor-quality codes. • **Dot peen Dark-on-Light**: This is the most robust method for reading dark dot peen codes on a light background. • **Dot peen Light-on-Dark**: This is the most robust method for reading light dot peen codes on a dark background. • **Laser and Chemical Etch**: This is the most robust method for reading laser or chemical etching point codings. **Timeout factor**: The timeout factor sets a timeout for difficult, deteriorated, or non-existent DPM codes. There is no timeout if this field = 0. The actual timeout for the device is ten times the number entered here in milliseconds. Note: This timeout does not apply to None, Basic Etch, or Basic Dots. This timeout applies only to Basic Inkjet, Basic Perspective, Dot peen Dark-on-Light, Dot peen Light-on-Dark, and Laser and Chemical Etch.

7.3.18 2-D Code (2-D Code Tool)

This tool is used to read 2-D codes.

Multiple codes can be detected in a search field (multiple code reading) with optional output of the code position in the image. Alternatively, you can span multiple search fields in the image to read codes at predefined positions. Enable the metric output to monitor the print quality of the code.

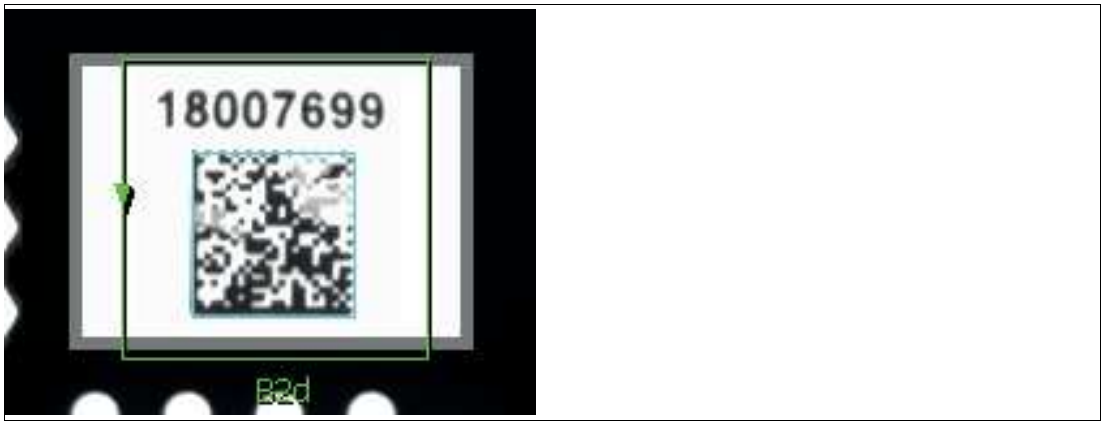


Figure 7.85 2-D code tool

Reading Codes

To enable reliable implementation of your application, the codes must have good image quality, contrast, and the right resolution to be read.

Observe the following instructions for reading codes with the sensor.

- Select the operating distance between the sensor and the code using the read field diagrams above. To do so, the **"module size"** must be known, i.e., the width of the narrowest bar (for 1-D codes) or the side length of the square for Data Matrix codes (2-D codes). If the operating distance selected is too large, reading is either unreliable or not possible.
- The values specified in the read field diagrams are maximum values for a specific module size. For a specific module size, the operating distance must be less than or equal to the specified distance.
- Adjust the operating distance and the focus so that the bars or squares in the image are clearly distinguishable. If the bars or squares blur together, the images cannot be read correctly.
- Ensure that the code to be read is completely within the field of view.
- Ensure uniform lighting of the code to be read. Inhomogeneous code lighting may lead to incorrect readings.
- During parameterization, observe the information in the following description for the 1-D code tool.
- Test the settings and make sure that your settings enable process-safe reading. We recommend trying out multiple codes when testing your application. Consider possible changes in lighting conditions as well as the situation in the test.

Using the 2-D Code Tool

1. Click the "2-D Code Tool" button in the tool selection area.
2. Select the shape of the "Region" from the options that appear below the tool selection window.

Note

- Use a rectangle for codes that are close to 0°, 90°, 180°, and 270° rotations in your template image.
- Use a polygon to draw a rectangle at any angle in the template image.
- Use an arc area (area between two arcs or curves) to read a circle of printed code.

↳ Move the mouse over the image in the work area.

3. Left-click on your starting point. Release the left mouse button and move the mouse to the area that represents your search area. Left-click again to draw a search area on the image.

↳ The selected field is labeled (B2d).

4. Resize the search area by clicking and dragging one of the corners of the search area.
5. Right-click on an edge of the rectangular search area.

↳ The properties window for the 2-D code tool will open.

2-D Code Tool Properties Window

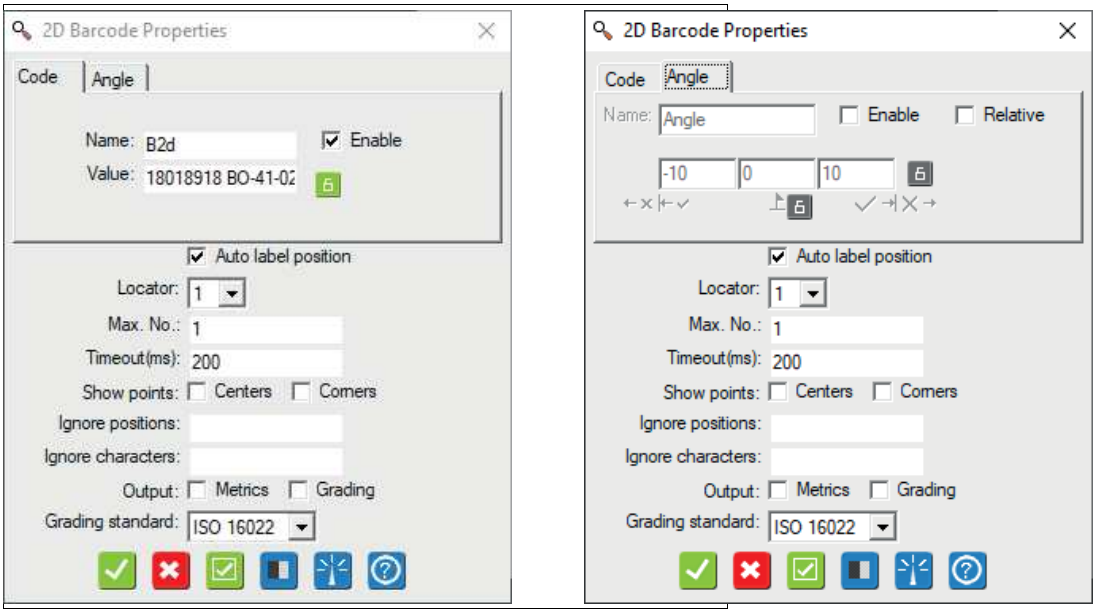


Figure 7.86 Code properties window

Output Variables

Tab	Description
Code	The tool will operate differently depending on the mode set. • Read code: The code is read and the content can be output. • Compare code: The code is read and a check is carried out to see whether the content matches the specified content. You can enter a fixed value under "Value."
Angle	The angle of the code defined by a line drawn from the top left corner to the top right corner in relation to the horizontal or the top edge of the image area. Clockwise is positive.

Output Variable Settings

Setting	Description
Name	You can change the variable name of the tool or the measurement. The variable name is displayed in the image, in the tool list, and in the measurement results. The name is used as a variable in scripts.
Enable	Enables the data output of the relevant output variable.
Relative	See "Relative Measured Values" on page 78
Tolerance	See "Setting the Tolerance Values" on page 77
Value	In this field, enter the character string to be compared with the content of the current code. The result of the tool is "passed" when the code is read and the read character string matches the set character string. Otherwise, the result is "failed." Leave an asterisk in this field if the 2-D code changes. The measurement reports any read value. It only fails if no value can be read. "?" = placeholder for a single character "*" = placeholder for multi-digit characters. The "padlock" allows you to lock the value, so that it does not change while you move the search field. The value should be unlocked while you change the parameter settings.

Parameters

Parameter	Description
Auto label position	Sets the name field next to the area in the display. Clear the check box to move the name field to another position, or drag the name field to the image with the property field closed.
Locator (anchor point)	This function causes the search field of the tool to follow the test object as it moves. Select a locator (by number) that you want to follow. To detach the tool from the locator or to hold the tool in a fixed position in the image area, select X. You can define up to four locators.
Max. No.	Number of codes expected in the search field. Increase this number if you have more than one code in your search field. You can define a search field with multiple codes or multiple search fields each with one code. The maximum number is 64. Larger numbers result in an error message.
Timeout	This function sets a timeout (in milliseconds) for complicated, incorrect, or non-existent codes. Note: This timeout refers to the decoding or code reading algorithm, but does not include the preprocessing time.
Show points	This function displays active points and allows you to use them in other tools. • Centers: Centers the center of the boundary box surrounding the code. • Corners: The four corner points of the boundary box that surround the code are placed in the corners.

Parameter	Description
Ignore positions	This function skips or ignores character positions in the character string. You can ignore unimportant, unrelated, or changing positions. Enter position numbers separated by spaces. Use 1 for the first character position (not 0), use 2 for the second character, and so on. Use -1 for the last character, use -2 for the second-last character.
Ignore characters	This function skips or ignores characters that are partially added to some codes. Example: If the first position and the $ characters are ignored, a decoded string of 1234 will have a result value of 1234.
Extended outputs (Output)	The output enables additional output data. Metric data: Indicates the code symbology detected and the number of read codes. Quality Assessment (Grading): Provides information about readability and print quality.
Grading standard	The selection list allows you to specify which quality standard is used for the classification calculation. The code quality can be assessed based on the following standards: ISO/IEC 15415, ISO/IEC 16022, AIM DPM, AS 9132. The code is evaluated according to the criteria of the standards with the existing lighting. The assessment is therefore only based on the standards, since this vision sensor does not have standardized lighting.

Buttons

Button	Description
OK	The changes are applied and the editing window is closed.
Cancel	The editing window closes without any changes being made.
Apply	The changes are applied without closing the editing window.
Preprocess	This function allows you to add image filtering to the tool. The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.
Advanced	These settings are optional. They allow more control over the object and the runtime. You can zoom in and out on areas and hide them. Hiding areas allows you to cut them from your search area by using a brush to turn those areas red. This function increases the test time.
Help	This opens the help window. Here you will find useful operational information.

Editing the 2-D Code Tool

1. Right-click on an edge of the rectangular search area.

   ↳ This opens the editing field for the 2-D code tool.

2. Click the "Preprocessing filter" button to add an image filter to the tool.

   Note

   The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.

3. Click the "padlock" button next to the value to store a fixed value.
4. Click the "OK" button.

   ↳ The changes are applied and the editing field closes.

5. Click the "Cancel" button.

   ↳ The editing field closes without any changes being made.

6. Click the "Apply" button.

   ↳ The changes are applied without closing the editing field.

7. Click the "Help" button.

   ↳ This opens the help window. Here you will find useful operational information.

Advanced Settings

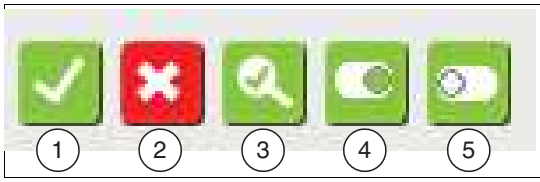


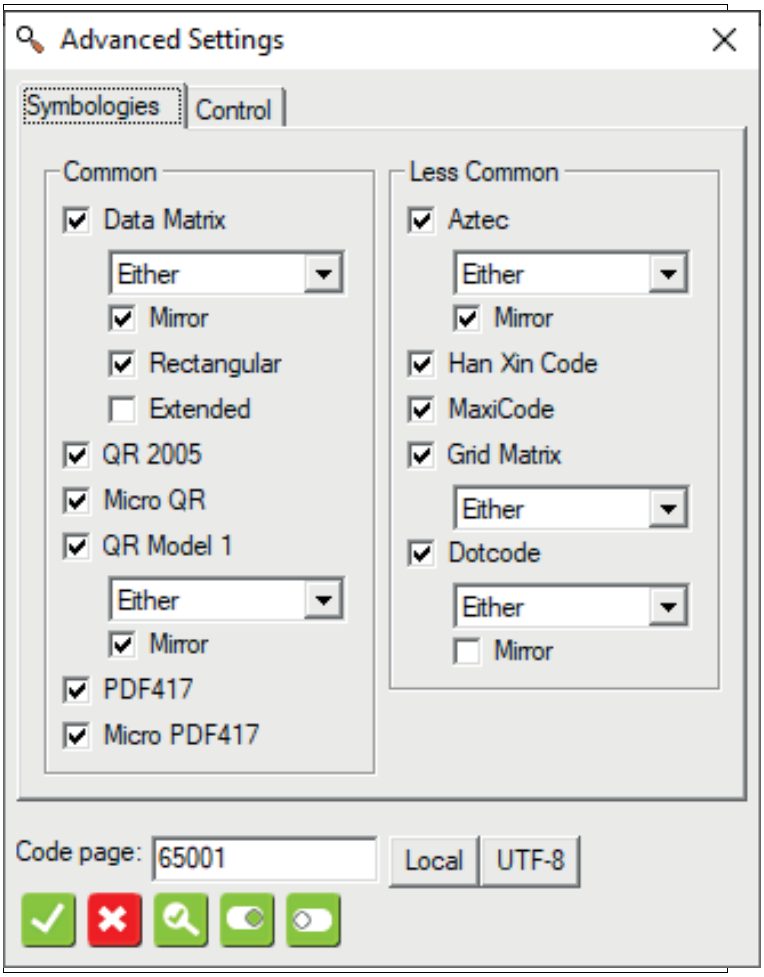
Figure 7.87 Menu bar

The buttons are the same for all tabs

Item	Button	Description
1	OK	Changes are applied and the editing field closed.
2	Cancel	The editing window closes without any changes being made.
3	Identify	The code type is automatically identified and selected in the search field.
4	Select all	All code types or symbologies are enabled on both tabs.
5	Deselect all	All code types are disabled on both tabs.

Tab Description

Symbologies

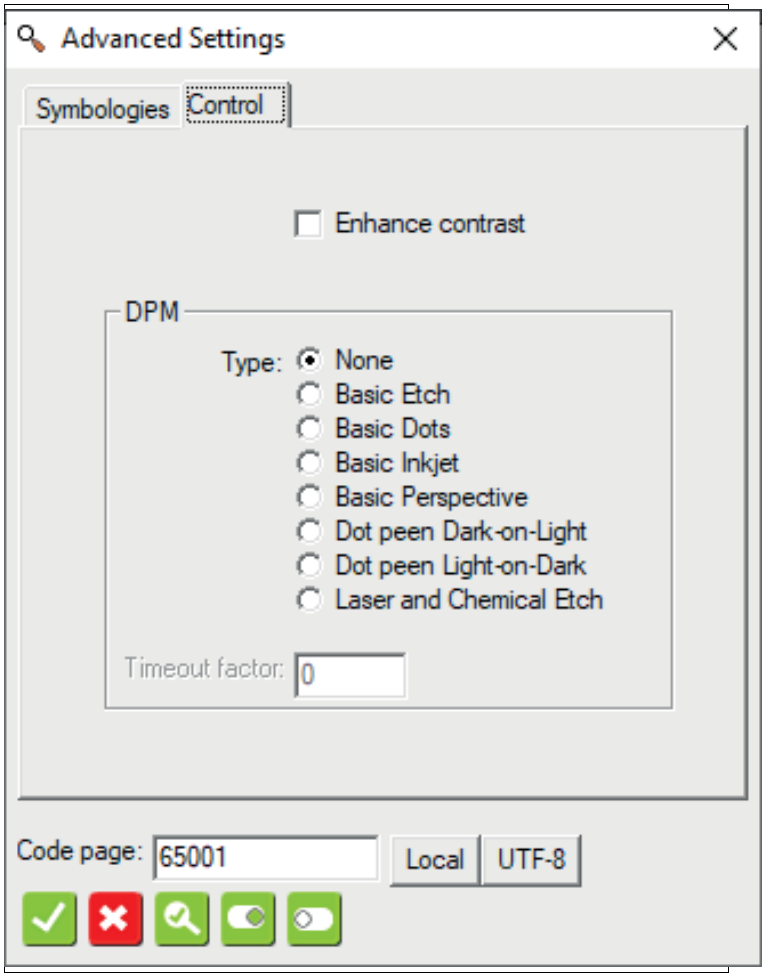


Menu	Description
Data Matrix	Enables the Data Matrix codes to be decoded. The polarity options from the drop-down list are Dark-on-Light (dark codes on a light background), Light-on-Dark (light codes on a dark background), or allow both. • **Mirror**: Enables both normal and mirrored images of codes to be decoded. If this box is not checked, only normal images will be decoded. • **Rectangular**: Enables standard rectangular codes to be decoded. • **Extended**: Enables extended rectangular codes to be decoded.
QR 2005	Enables the QR 2005 code to be decoded.
Micro QR	Enables the QR Model 1 code to be decoded.
QR Model 1	Enables the Data Matrix codes to be decoded. The polarity options from the drop-down list are Dark-on-Light (dark codes on a light background), Light-on-Dark (light codes on a dark background), or allow both. • **Mirror**: Enables both normal and mirrored images of codes to be decoded. If this box is not checked, only normal images will be decoded.
PDF417	Enables the PDF 417 code to be decoded.
Micro PDF417	Enables the Micro PDF 417 code to be decoded.
Aztec	Enables the Aztec code to be decoded. • **Mirror**: Enables both normal and mirrored images of codes to be decoded. If this box is not checked, only normal images will be decoded.

2021-03

Menu	Description
Han Xin Code	Enables the Han Xin code to be decoded.
MaxiCode	Enables the MaxiCode to be decoded.
Grid Matrix	Enables the Grid Matrix code to be decoded. The polarity options from the drop-down list are Dark-on-Light (dark codes on a light background), Light-on-Dark (light codes on a dark background), or allow both.
DotCode	Enables the DotCode to be decoded. The polarity options from the drop-down list are Dark-on-Light (dark codes on a light background), Light-on-Dark (light codes on a dark background), or allow both. • **Mirror**: Enables both normal and mirrored images of codes to be decoded. If this box is not checked, only normal images will be decoded.

Control



Menu	Description
Enhance contrast	Enables you to improve the image to decode any poorly printed codes. This may increase the execution time. Cannot be used with Dot peen Dark-on-Light, Dot peen Light-on-Dark, or Laser and Chemical Etch.
DPM Type	Selects image conditioning options specifically for direct part marking (DPM). Note: For all DPM options, the maximum search field size is 614,000 pixels. • **None**: Does not allow DPM-specific processing. • **Basic Etch**: Optimized for simple laser or chemical etching. • **Basic Dots**: Point-jet and inkjet images • **Basic Inkjet**: Basic inkjet images and poor-quality inkjet images. • **Basic Perspective**: Reads laser markings and normal codes of good quality that are centered in the image but have strong perspective distortion. This mode is faster but less robust than "Basic Etch" for reading poor-quality codes. • **Dot peen Dark-on-Light**: This is the most robust method for reading dark dot peen codes on a light background. • **Dot peen Light-on-Dark**: This is the most robust method for reading light dot peen codes on a dark background. • **Laser and Chemical Etch**: This is the most robust method for reading laser or chemical etching point codings. **Timeout factor**: The timeout factor sets a timeout for difficult, deteriorated, or non-existent DPM codes. There is no timeout if this field = 0. The actual timeout for the device is ten times the number entered here in milliseconds. Note: This timeout does not apply to None, Basic Etch, or Basic Dots. This timeout applies only to Basic Inkjet, Basic Perspective, Dot peen Dark-on-Light, Dot peen Light-on-Dark, and Laser and Chemical Etch.

7.3.19 OCR

The OCR Tool is used for optical character recognition (e.g., letters and numbers). The OCR Tool can also be used to read logos and ID marks.

Figure 7.88 OCR Tool

Using the OCR Tool

1. Click the "OCR Tool" button in the tool selection area.
2. Select the shape of the "Region" from the options that appear below the tool selection window.

Note

- Use a rectangle for texts at 0°, 90°, 180°, and 270° rotations.
- Use polyline for text at any angle (not in 90 degree increments).
- Use an annulus (area between two circles) or an arc area (area between two arcs or curves) to wrap printed text around a circle.

↳ Move the mouse over the image in the work area.

3. Left-click on your starting point. Release the left mouse button and move the mouse to the area that represents your search area. Left-click again to draw a search area on the image.

↳ The selected field is labeled (OCR1...).

4. Resize the search area by clicking and dragging one of the corners of the search area.
5. Right-click on an edge of the rectangular search area.

↳ The properties window of the OCR Tool will open.

OCR Tool Properties Window

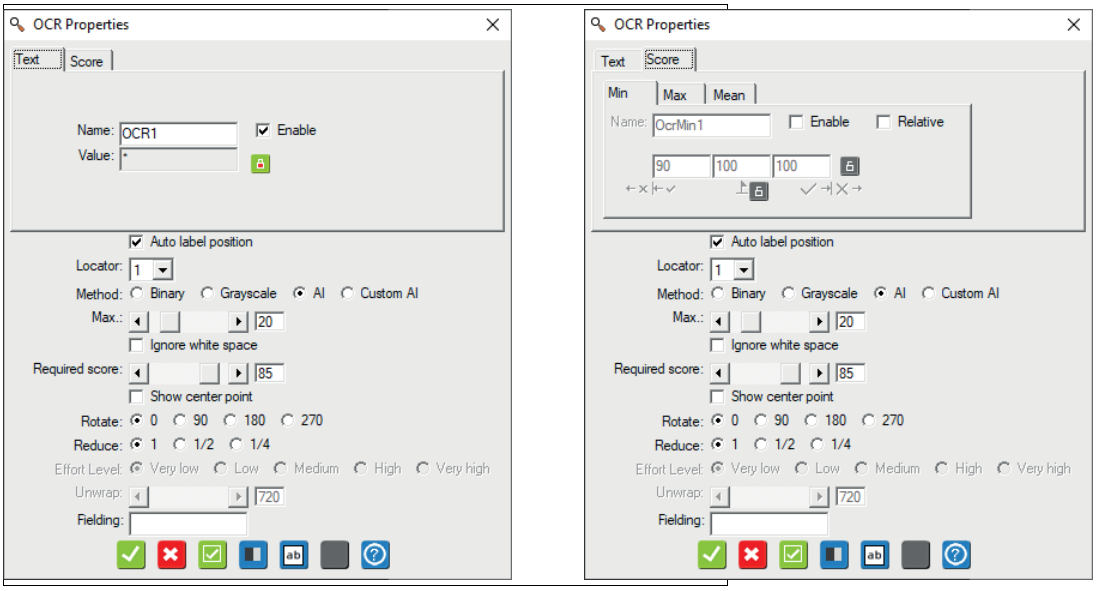


Figure 7.89 Properties window—Text and Score

Output Variables

Tab	Description
Text	In this menu, the read text value is output as a string variable. You can compare the read-in data string with a fixed text value that is entered in the "Value" field.
Score	• **Min**: lowest match result value for all characters. • **Max**: highest match result value for all characters. • **Mean**: average match for all characters. This measurement creates a series of measured result values for all identified characters. All individual scores can be displayed on the operator monitor and are available in the Script editor.

Output Variable Settings

Setting	Description
Name	You can change the variable name of the tool or the measurement. The variable name is displayed in the image, in the tool list, and in the measurement results. The name is used as a variable in scripts.
Enable	Enables the data output of the relevant output variable.
Relative	See "Relative Measured Values" on page 78
Tolerance	See "Setting the Tolerance Values" on page 77
Value	If you always want to compare the same text, enter the expected text in "Value." The measurement fails if it does not match the specified text. Leave an asterisk (*) in this field if the text changes. The asterisk is a placeholder. The measurement reports any read value. It only fails if no value is read. The "padlock2 button allows you to lock the entered value, so that it does not change when you move the search field. The value must be unlocked if you change the parameter settings.

2021-03

Parameters

Parameter	Description
Auto label position	Sets the name field next to the area in the display. Clear the check box to move the name field to another position, or drag the name field to the image with the property field closed.
Locator (anchor point)	This function causes the search field of the tool to follow the test object as it moves. Select a locator (by number) that you want to follow. To detach the tool from the locator or to hold the tool in a fixed position in the image area, select X. You can define up to four locators.
Method	This parameter is used to set the algorithm with which the characters are offset from the background and characters are identified at runtime. Do not change this value when you start teaching in the characters. • **Digital**: A threshold value is used to create a "digital" image. This separates the characters and the background. The "blob analysis" is used to identify character patterns. The taught-in characters are used for matching. • **Grayscale**: Grayscale correlation matching is used to identify characters. The taught-in characters are used for matching. • **AI**: A threshold value is used to create a "digital" image. AI uses a library of previously taught-in characters that contain capital letters and punctuation. This library does not contain lower case letters. If the characters are not detected correctly, another evaluation method must be used.
Max.	Maximum number of characters. Limiting the number of read characters may increase the speed.
Ignore white space	By ignoring spaces between characters, blank spaces are hidden as separators. This is especially useful for vertical data strings or polyline strings.
Required score	Required score sets a minimum quantity or threshold for the number of hits for each result. Results that do not meet this minimum are rejected or not recognized.
Show center point	This function creates a point in the center of each detected character.
Rotation	Character orientation (left to right, top to bottom, etc.). Positive direction of rotation is counter-clockwise. Use this function to change the direction in which characters are detected or read.
Reduce	To consume less data and work more quickly, reduce the number of partial samples during teaching in and testing when working with large characters. Do not use partial samples for characters with small or fine details.
Effort Level	Use a simplified or advanced read algorithm. High settings are slower, but more accurate, and function better with deteriorated or low-contrast characters. Low settings are quicker, but less thorough.

Parameter	Description
Unwrap	The reading or sensing distance around the area between two circles is limited or increased. 360 is a complete revolution. 720 is two complete revolutions. If you use a number greater than 360, the area at the beginning of the roll-out scan is repeated and contains characters or a character string that can be split by a 360 degree scan. This parameter is important if the part moves or rotates. The OCR Tool examines the pattern rectangle (the unwrapped annulus) to find an area that represents a full rotation (360), measures the spaces between them, and begins with the character that follows the largest gap in the part. It is assumed that the data string(s) do not fill the entire circle without gaps and that the gap at the beginning/end of the data strings is larger than all the spaces between characters or data strings.
Fielding	The field boundary positions in the character string are defined as specific character types. This allows the OCR Tool to be accelerated by reducing the number of characters in comparison with the characters found. • **?**: any character • **#**: numeric characters only • **a**: lower case letters only • **A**: upper case letters only • **$**: letters only, upper and/or lower case

Buttons

Button	Description
OK	The changes are applied and the editing window is closed.
Cancel	The editing window closes without any changes being made.
Apply	The changes are applied without closing the editing window.
Preprocess	This function allows you to add image filtering to the tool. The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.
Font Editor	You can use the Font Editor to import fonts or individual characters, teach in new characters, and save taught-in fonts or characters.
Help	This opens the help window. Here you will find useful operational information.

Editing the OCR Tool

1. Right-click on an edge of the rectangular search area.

   ↳ This opens the editing field for the OCR Tool.

2. Click the "Preprocessing filter" button to add an image filter to the tool.

   Note

   The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.

3. Click the "padlock" button next to the value to store a fixed value.
4. Click the "Font Editor" button to edit characters or fonts.
5. Click the "OK" button.

   ↳ The changes are applied and the editing field closes.

6. Click the "Cancel" button.

   ↳ The editing field closes without any changes being made.

7. Click the "Apply" button.

   ↳ The changes are applied without closing the editing field.

8. Click the "Help" button.

   ↳ This opens the help window. Here you will find useful operational information.

Font Editor

You can use the Font Editor to import fonts or individual characters, teach in new characters, and save taught-in fonts or characters.

Note

The "Rotate" parameter from the OCR text properties changes the direction in which the characters appear in the "Add" tab of the Font Editor. Select the character orientation before starting to adjust settings in the Font Editor. For the area between two annuluses, the display shows 720° (the default "roll out") or two full cycles to prevent missing characters where the pattern begins or where the annulus is "split." For the ring area, the ring is scanned with 0° counter-clockwise (the base line is the outer circle), with 180° clockwise (the base line is the inner circle). If you change the "Rotation", reverse the read direction and mirror the characters in the annulus.

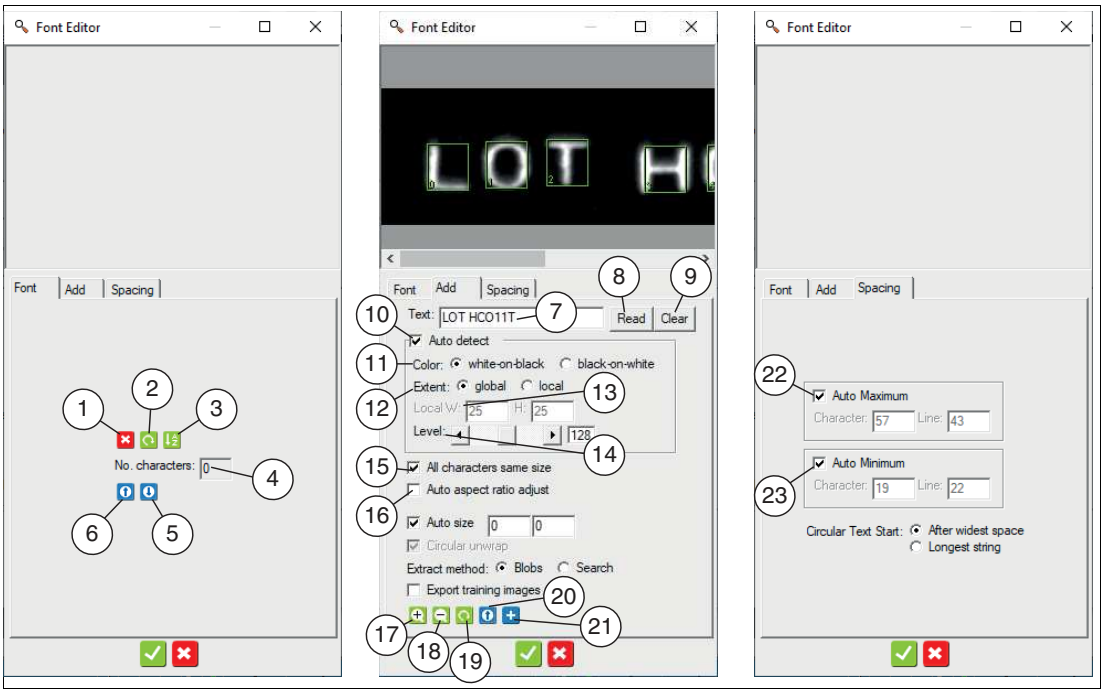


Figure 7.90 Font Editor

Tab	Item	Designation	Description
Font	1	Delete	Deletes the selected characters.
	2	Rotate	Rotates selected characters 90 degrees clock-wise.
	3	Sort	Sorts characters alphabetically by name.
	4	No. characters	Total number of taught-in characters.
	5	Load and save	Saves all taught-in characters in a character set file.
	6	Open file	Opens a previously saved character set file.

Tab	Item	Designation	Description
Add	7	Text	Designation of the detected characters.
	8	Read	Reads or identifies all characters in boxes on the image.
	9	Clear	Deletes the text field and image area.
	10	Auto detect	Enables automatic detection of characters in the image area.
	11	Color	Select white-on-black for light characters on a dark background or black-on-white for dark characters on a light background.
	12	Extent	Type of filter to be used to separate the characters from the background. "Global" applies a limit value to the entire image area. "Local" uses a smaller adaptive filter.
	13	Local W, H	Adjusts the size of a local filter.
	14	Level	Adjustable level for separating characters from the background.
	15	All characters same size	Clear this box if the characters differ in size (width or height). Check this box if all characters have the same height and width. All the boxes in the image are limited to the same size.
	16	Auto aspect ratio adjust	Changes the height or width (if necessary) to follow the AI library of taught-in characters more closely. This may improve the results in some examples.
	17	Zoom in	Enlarges the image area.
	18	Zoom out	Reduces the size of the image area.
	19	Rotate image	Rotates the image 90 degrees clockwise.
	20	Load image	Opens an image file that is used to teach in/add additional characters.
	21	Save characters	Add all characters identified in the image field and text box to the taught-in font.
Spacing	22	Auto Maximum	The software specifies maximum distance between characters and lines. Clear the check box to override the software values and set your own distance. • **Character**: maximum distance between characters. • **Line**: maximum distance between lines.
	23	Auto Minimum	The software specifies a minimum distance between characters and lines. Clear the check box to override the software values and set your own distance. • **Character**: minimum distance between characters. • **Line**: minimum distance between lines.

Teaching in Characters

There are two ways to teach in characters: automatic detection of multiple characters and the manual method. Use the manual method if you are not satisfied with the results of the automatic detection.

Teaching in Characters Automatically

The following steps describe which settings you need to configure to detect characters automatically.

1. In the "Add" tab, select the "Auto detect" box. This function is enabled by default.
2. In the "Add" tab, clear or select the "All characters same size" check box if all characters are the same size. This box will be grayed out when you start teaching in.
3. Select the appropriate color: "white-on-black" for light characters on a dark background or "black-on-white" for dark characters on a light background.
4. Select the extent of the filter. Use the "global" selection if the ambient lighting is relatively constant. A uniform limit value is applied to the entire environment. Use the "local" selection if the ambient lighting fluctuates or changes. The local adaptive filter uses "Local W and H" to define a local filter size and calculate limit values.
5. Change the "Level" setting to adjust the delimitation of the characters from the background. To do this, use the slider bars to move or resize boxes.

   ↳ The characters are read again when you adjust the "Level" value. The software identifies all characters because the AI function is activated during teach-in.
6. Correct any misidentified characters and add the corrected characters to your character set.
7. Enter all characters in the "Text" input field in the "Add" tab, including repetitions or duplicates. The number of characters in the "Text" input field must match the number of characters identified in the image.
8. Click the "+" (Add to font) button at the bottom of the menu to add all characters to the taught-in character set (even if all characters have been correctly identified by AI).

   ↳ The menu switches to the "Font" tab.

Teaching in Characters Manually

The following steps describe which settings you need to configure to detect characters manually.

1. Clear the "Auto detect" box in the "Add" tab. This function is enabled by default.
2. In the "Add" tab, clear or select the "All characters same size" check box if all characters are the same size. This box will be grayed out when you start teaching in.
3. Left-click once on any corner of the first character. Release the left mouse button and move the mouse to the location that represents your search area for the character. Left-click again to draw a rectangle around your character.
4. Resize the rectangle by clicking and dragging one of the corners of the rectangle. Adjust the size of the rectangle so that it fits closely to the entire character and contains as little background as possible.
5. Draw boxes around each character you want to teach in. You can learn one character at a time, or draw multiple boxes and add them all at once.
6. Use "Clear" and "Read" in the "Add" tab to allow the "AI" software to attempt to identify characters in the rectangles you have created. Correct any incorrect characters.
7. For each character around which you have drawn a border (including repeats), type a character label or name in the "Text" field. Usually, the character itself is the designation. The number of characters in the "Text" field must match the number of characters identified in the image.
8. Click the "+" (Add to font) button at the bottom of the menu to add all characters to the taught-in character set (even if all characters have been correctly identified by AI).

   ↳ The menu switches to the "Font" tab.

7.3.20 Verify Tool

The Verify tool compares the taught-in template with an object in the image and reports how much the object differs from the template. You can teach in the Verify tool a series of samples to identify a selection of suitable variations.

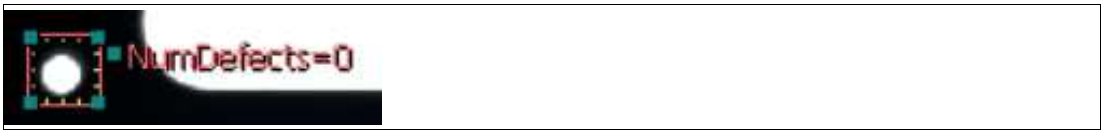


Figure 7.91 Verify tool

Using the Verify Tool

1. Click on the "Verify Tool" button in the tool selection area.
2. Left-click on your starting point. Release the left mouse button and move the mouse to the area that represents your search area. Left-click again to draw a search area on the image.

   ↳ The selected search area is labeled (NumDefects, NumDefects1, etc.). Yellow marks will appear on the inside edges of the rectangular search area. These marks define a grid that splits the rectangle into square "cells."
3. Resize the search area by clicking and dragging one of the corners of the search area.
4. Left-click in the search field of your object, hold down the left mouse button, and drag to highlight as many cells as you want to delete.

   ↳ The cells marked with a yellow X are not included in the taught-in image and are not included in the error measurement.
5. Hold down the right mouse button and drag it across the yellow X to insert the cells back into the object's search field.
6. Right-click on an edge of the rectangular search area.

   ↳ The properties window for the Verify tool will open.

Verify Tool Properties Window

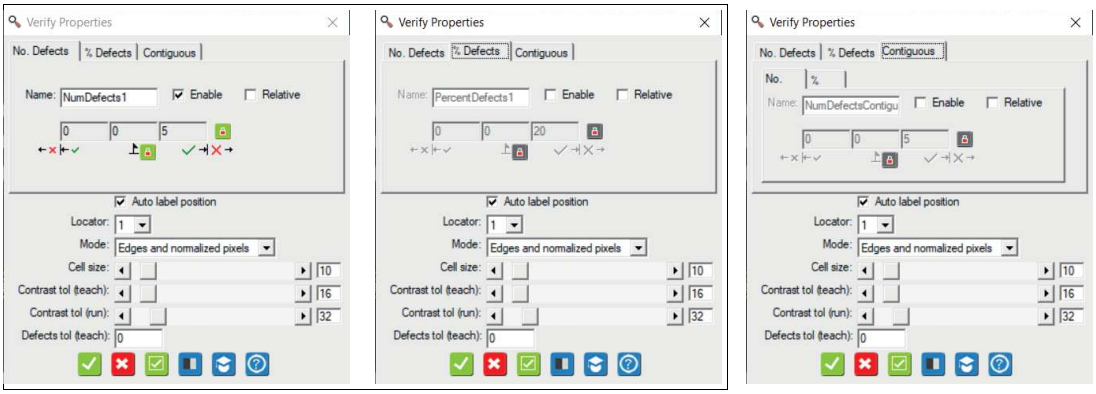


Figure 7.92 Properties window

Output Variables

Tab	Description
No. Defects	The total amount or number of cells that did not match the images identified as good. 0 = perfect match, no defects.
% Defects	The percentage of cells that did not correspond to the images identified as good. 0 = perfect match, no defects.
Contiguous: No.	The number of cells that make up the largest single defect. "Contiguous" cells are connected at one or more sides. Cells that only touch each other at the corners are not counted.
Contiguous: %	The number of cells representing the largest single defect, expressed as a percentage of the total amount of cells in the object's search field. Cells that only touch each other at the corners are not counted.

Output Variable Settings

Setting	Description
Name	You can change the variable name of the tool or the measurement. The variable name is displayed in the image, in the tool list, and in the measurement results. The name is used as a variable in scripts.
Enable	Enables the data output of the relevant output variable.
Relative	See "Relative Measured Values" on page 78
Tolerance	See "Setting the Tolerance Values" on page 77

Parameters

Parameter	Description
Auto label position	Sets the name field next to the area in the display. Clear the check box to move the name field to another position, or drag the name field to the image with the property field closed.
Locator	This function causes the tool to follow the test object as it moves. Select a locator (by number) that you want to follow. To detach the tool from the locator or to hold the tool in a fixed position in the image area, select X. You can define up to four locators.

Parameter	Description
Mode	Select a mode to determine which information or adjustment methods are used (for the operating time). • **Edges**: Uses the standard deviation of the pixels in each cell. This mode is largely independent on lighting changes. This mode is effective when the lighting of the part to be tested changes. • **Raw pixels**: Uses the average of the pixels in each cell. This mode is useful when looking for small but widespread changes in the reflectance of a part (e.g., caused by traces of fire on the part). This mode is very sensitive to changes in lighting. You therefore need to use a light source that remains stable over time. • **Normalized pixels**: Uses the average of pixels in each cell, minus the average across the entire search area, divided by the standard deviation of pixels in the search area [(cellMean - ROIMean) / ROIStdDev]. This mode is less sensitive to changes in lighting and less sensitive to changes in reflectance. • **Edges and normalized pixels**: Uses the maximum value (or a greater difference from the edges and the normalized pixels). In this mode, certain changes in reflectance are detected, e.g., when printed material begins to "fade" and the structure and texture of a part changes. When you change the mode, you do not need to teach in again. The data for all three operating modes is stored during the teach-in process.
Cell size	Size of the square cells in pixels. Smaller cells give more detail, but require a little more computing time. The range is 4 to 100, i.e., the individual cells are 4x4 pixels up to 100x100 pixels.
Contrast tol(teach)	Permissible contrast difference in grayscale during the teach-in step. Changes below this limit are ignored. Changes above this limit are applied.
Contrast tol (run)	Permissible contrast difference in grayscale during runtime or measurement. Changes below this limit are ignored. Changes above this limit are defective.
Defects tol (teach)	Permissible percentage of defective area during the teach-in process. If the difference is greater than this value, a new image is taught in and saved. A higher setting reduces the number of images taught in, which in turn reduces the amount of memory required.

Buttons

Button	Description
OK	The changes are applied and the editing window is closed.
Cancel	The editing window closes without any changes being made.
Apply	The changes are applied without closing the editing window.
Preprocess	This function allows you to add image filtering to the tool. The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.
Learn	In this menu, the current image is added to the database.
Help	This opens the help window. Here you will find useful operational information.

Editing the Verify Tool

1. Right-click on an edge of the rectangular search area.

   ↳ This opens the editing field for the verify tool.

2. Click the "Preprocessing filter" button to add an image filter to the tool.

   Note

   The preprocessing filter is optional. You should add a preprocessing filter before adjusting or locking the other parameters. A preprocessing filter will extend the runtime of the test.

3. Click the "padlock" button next to the value to store a fixed value.
4. Click the "Learn" button to open the learning menu.
5. Click the "OK" button.

   ↳ The changes are applied and the editing field closes.

6. Click the "Cancel" button.

   ↳ The editing field closes without any changes being made.

7. Click the "Apply" button.

   ↳ The changes are applied without closing the editing field.

8. Click the "Help" button.

   ↳ This opens the help window. Here you will find useful operational information.

"Learn" Menu

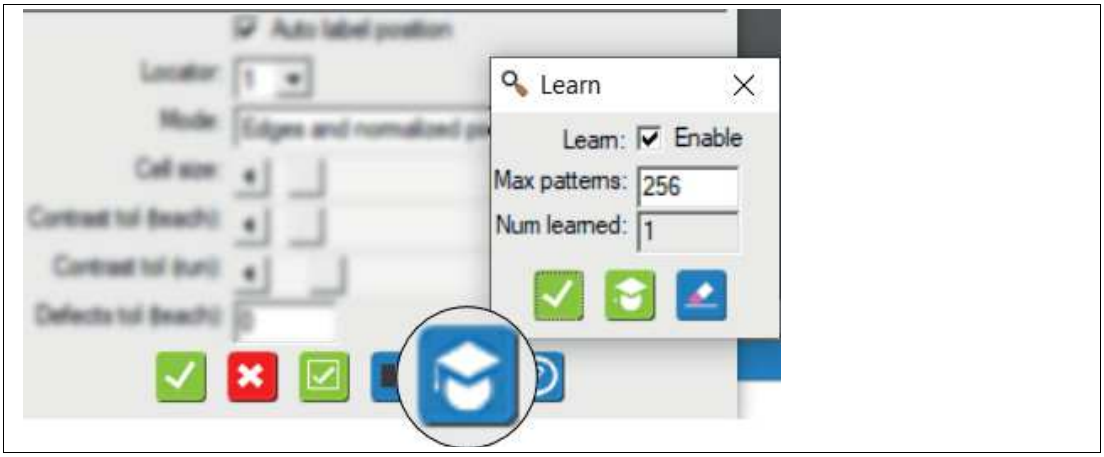


Figure 7.93 Learn

Button	Description
Learn	Selecting this box enables learning examples; clearing the box disables learning. Remember to return to this menu and disable learn mode before you perform your test. If you do not disable learn mode, all test images will be added to the "good examples" database and no errors will be reported.
Max patterns	This defines the maximum number of "good" images that are saved in the database of taught-in "good examples."
Num learned	This shows how many images are saved in the database of the taught-in "good examples."
Delete	The "Delete" button in the "Learn" menu deletes all images in the "good examples" database. Note: If you had to re-adjust the size, position, or number of cells, you may have incorrect examples in the database. You should delete them and add the current image to the database.

7.3.21 Color Meter Tool

The color meter tool is mainly used to determine the color of an area. It uses the assigned color classes of the Vision Configuration Tool application. It is often used to perform spot checks on an assembly or package for correct colors. A typical example would be checking the correct (color-coded) fuses installed in a vehicle fuse box.

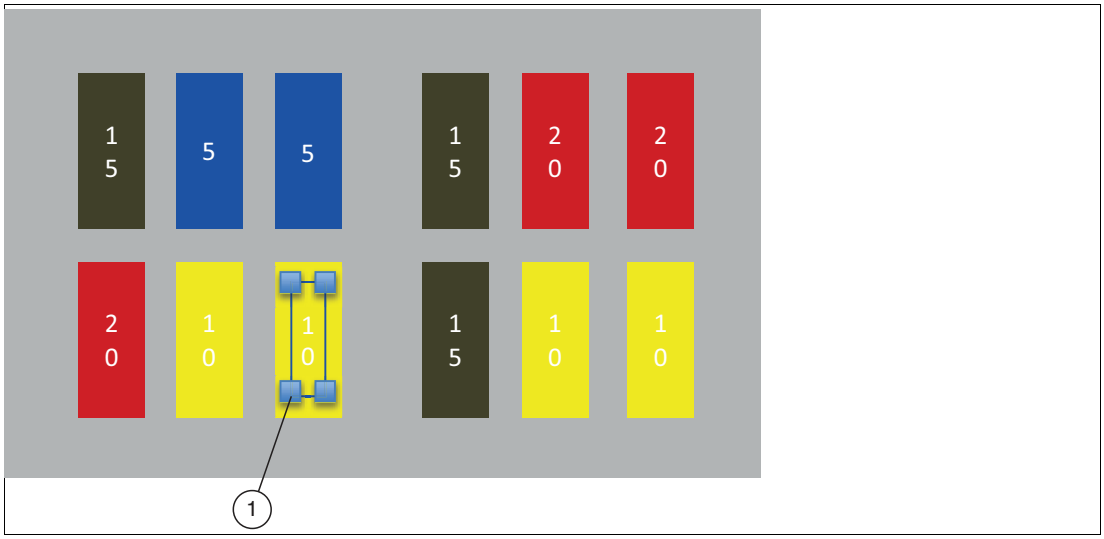


Figure 7.94 Example: vehicle fuse box

Color Meter Tool Information	
Search area shapes	Rectangular
Tool preparation	Two click points to form a rectangle
Measurements	The three main colors in the search area, each with a % + RGB average value.

Instead of specifying the exact RGB value of a color, which can vary significantly depending on lighting and sensor properties, the Vision Configuration Tool groups the colors into a small number of classes. To illustrate this, look at the first image below. We can group the colors in this image into five classes: Red, Green, Blue, Yellow and Background (all others). Each class can contain multiple shades of the same color, so that each pixel in the image is assigned to a class.

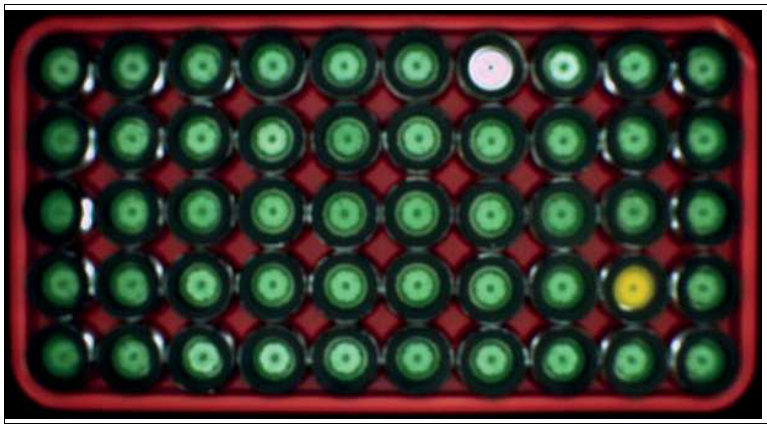


Figure 7.95 Example 1: color classes

When the image is split into individual sections, the color meter tool can display the three main colors representing the search area, along with the density of each individual color.

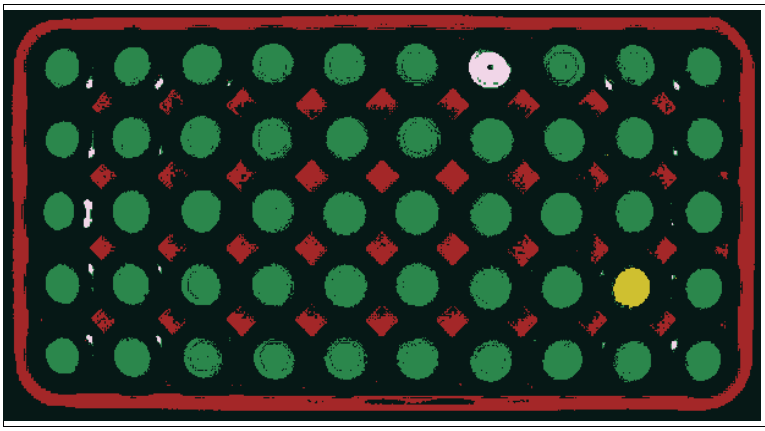


Figure 7.96 Example 2: color classes

The following example checks the correct mounting of a medical instrument. In this application, a combination of colorimeter and Edge Count Tools is used.

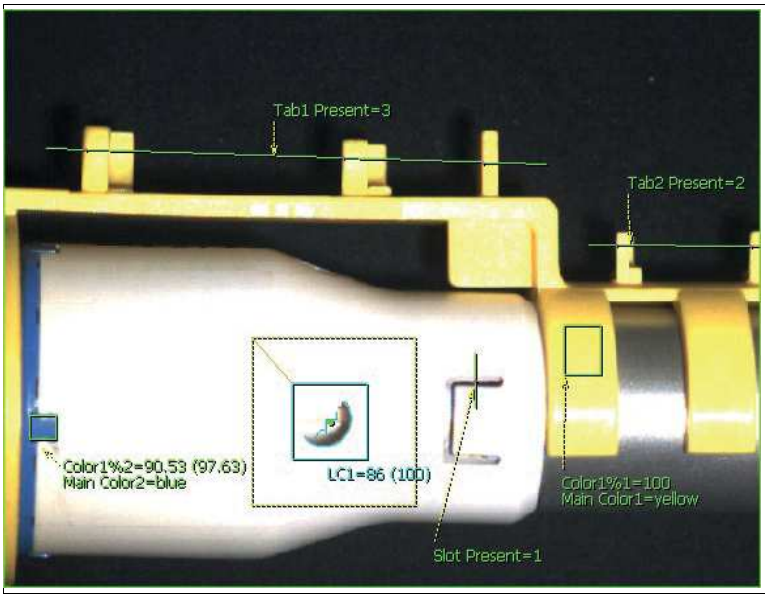


Figure 7.97 Example: medical instrument

In the following application, the color on some plastic parts is checked to ensure proper packaging. In this case, the blue part is expected to be white and the check fails.

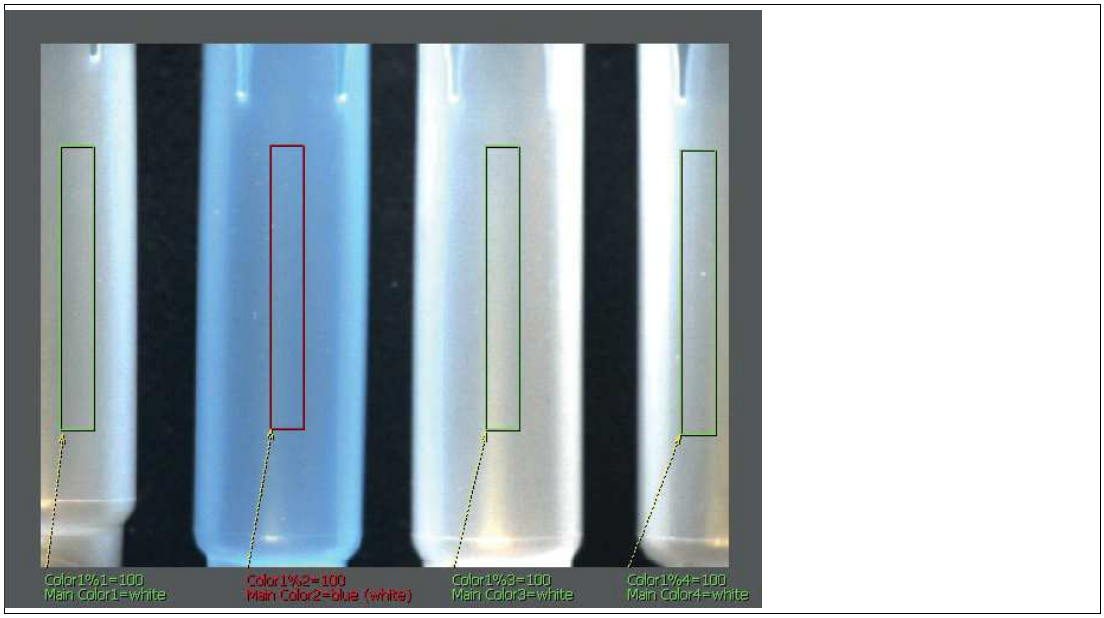


Figure 7.98 Example: color of the plastic parts

Teaching in Colors

Conventional color classifiers work using area-based models. These classifiers calculate a cube or a sphere based on taught-in colors. They allow only a few taught-in colors, which means that these colors must be selected carefully. The color classifier in the Vision Configuration Tool takes away the issue of too many taught-in colors. You can drag the mouse around and teach in dozens or hundreds of pixels in a single step. This means that you can easily define multiple shades of the same color or put completely different colors in the same group (i.e., apple = green or red, orange = orange, lemon = yellow, etc.).

Once the teach-in phase is complete, the Vision Configuration Tool assigns each pixel in the image to a class. If there are five colors in the image and you only teach in three of them, the Vision Configuration Tool assigns the two non-taught-in colors to the "Background" class. The more colors you teach in the easier it becomes to segment pixels.

If you make an error when teaching in and accidentally teach in the wrong color, you can repeat the process and teach in the correct color. For example, you selected a green pixel and accidentally taught it in as red. Instead of reversing the error, select the same pixel again and teach it in as green. In this way, you can teach in many sample pixels in a short time.

Example of Teaching in a Color

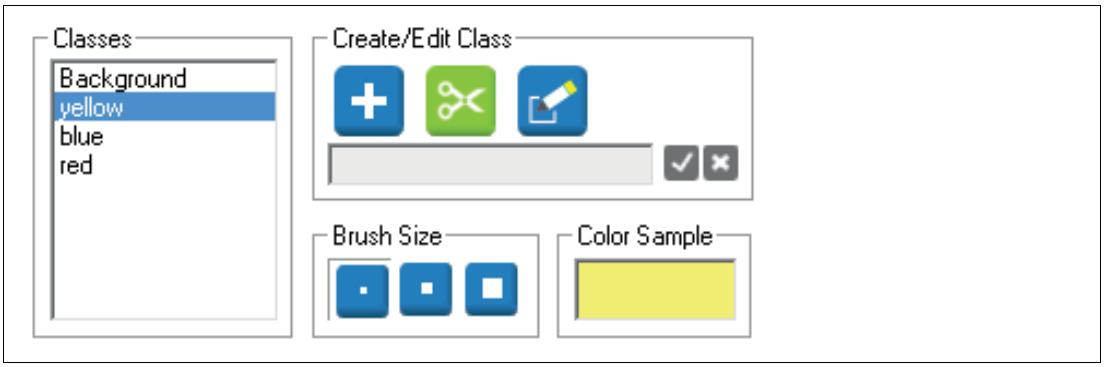


Figure 7.99 Vision Configuration Tool

We define three colors to classify the pills (1). The colors around the pills are taught in as the "Background" class.

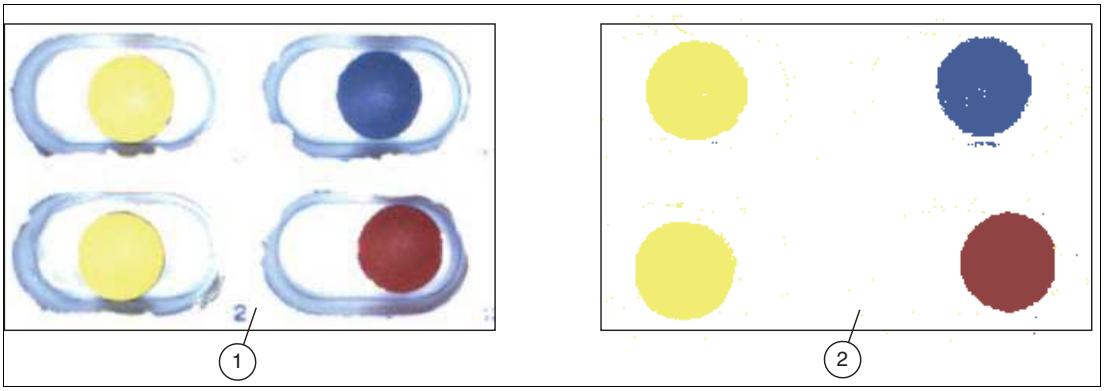


Figure 7.100 Example: pills

1. Original image
2. Taught-in color classes

7.4 Interface Configuration

This section describes how to configure the sensor to communicate with other devices.

Overview of Interface Menu

The interface menu is the place to set up the connection to external devices (e.g., PLC or storage devices) for control/status communication.

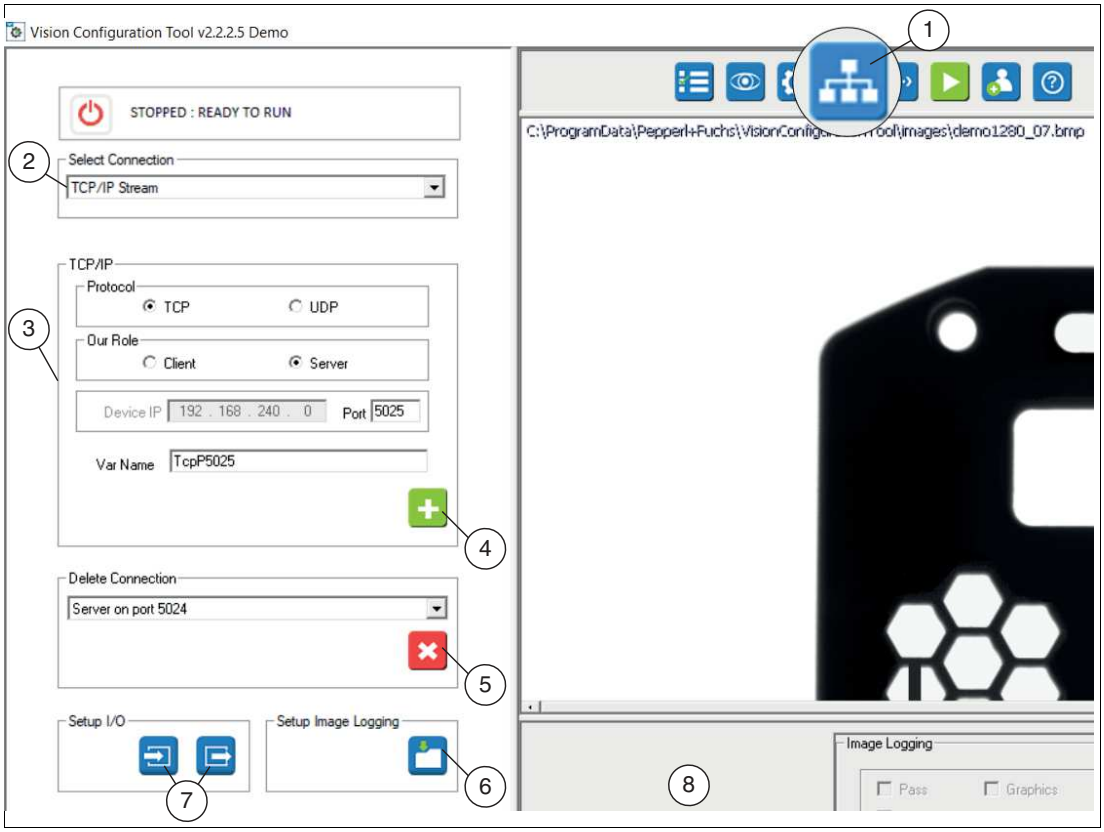


Figure 7.101 Example of interface menu for TCP/IP Stream

Item	Menu	Description
1	Set up communication	The "Set up communication" button takes you to the main menu for interface configuration.
2	Select connection method	Use the drop-down window to select your interface type.
3	Set up connection method	Use this section of the menu to set up the connection of the selected interface type. The menu changes depending on the PLC type selected. Various functions are available for selection.
4	Add configuration	Add new configuration
5	Delete configuration	Delete the selected configuration
6	Set up image logging	Enable/disable history log and image logging
7	Configuration I/O	You can use these two buttons to set up the inputs and/or outputs.
8	Configuration and status window	The settings of the selected menu are displayed in the configuration and status window.

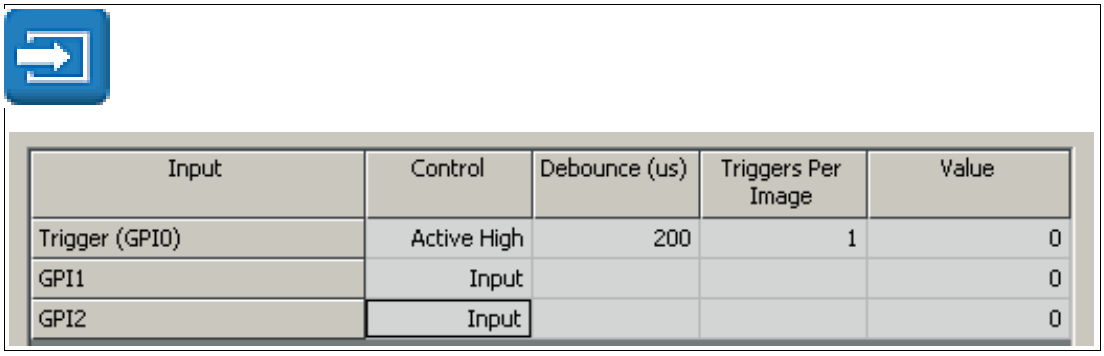
Sensor Name and Address

Sensors are supplied with a default name "VOSxxxx" and an address (192.168.0.100). If you are connecting more than one sensor in the same network or on the same PLC, change the IP address so that each sensor has its own address. You should also change the name so that each sensor has its own name. This is important and even critical for some PLCs and network configurations to avoid confusion. Use the Nexus application to change the name and address of the sensor (see "Changing the IP Address" on page 46).

2021-03

7.4.1 Inputs

When you click the "Inputs" button, the settings menu for inputs opens in the configuration and status window. In this menu, you can adjust the following settings:



Input	Control	Debounce (us)	Triggers Per Image	Value
Trigger (GPI0)	Active High	200	1	0
GPI1	Input			0
GPI2	Input			0

Figure 7.102 Input menu

Function	Description
Trigger (GPI0)	You can set the polarity (Active High or Active Low) in the "Control" column, enter a debounce period, and set a trigger divider "Triggers Per Image" for the sensor trigger input.
GPI1 & GPI2	You can use a drop-down list to select the function of the two remaining inputs. The "Value" column shows the current status of all inputs (1 for high or 0 for low).

Example of Job Change via Inputs

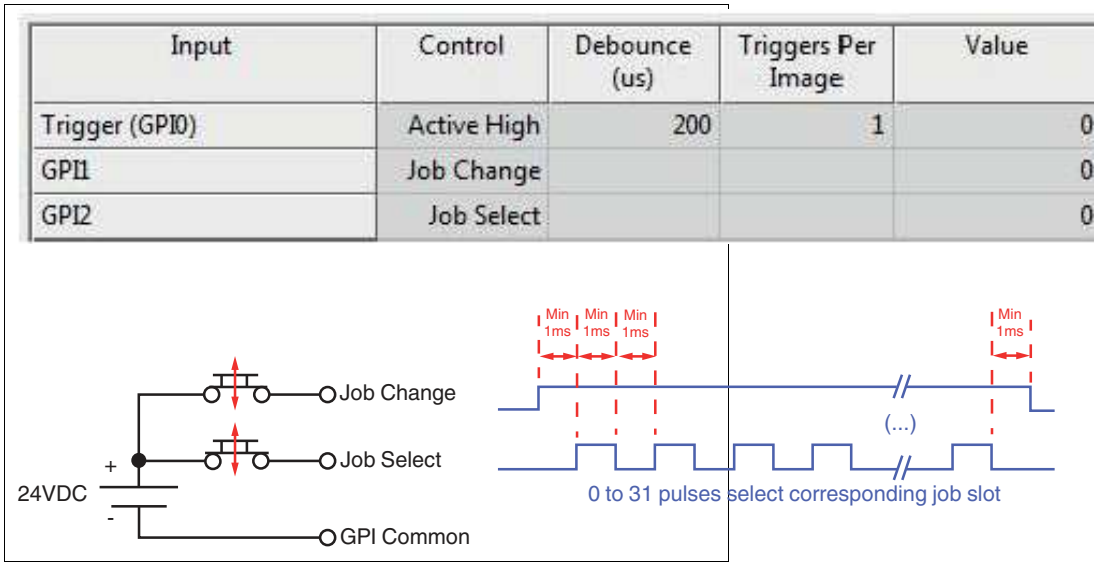


Figure 7.103 Sample input menu

You can save up to 32 job files to your VOS device that can be loaded as and when needed. The job files can be changed over in different ways. Via the "Select Solution" menu selection, via the "ChangeSolution(#)" script function, or via the inputs as described below.

To use the sensor inputs for the job change, select the "Job Change" and "Job Select" functions in the configuration window of the input control panel as shown in the figure.

A PLC sets a HIGH signal on the "Job Change" line GPI1 and sends a series of pulses on the "Job Select" line GPI2 corresponding to the job number; e.g., 12 pulses for job file 12. The "Job Select" pulse sequence requires a setup and delay time of one millisecond between the edge transitions. If a job change is detected, the sensor cancels the current test process (if it is running) and switches over immediately.

7.4.2 Outputs

When you click the "Outputs" button, the settings menu for outputs opens in the configuration and status window. In this menu, you can adjust the following settings:

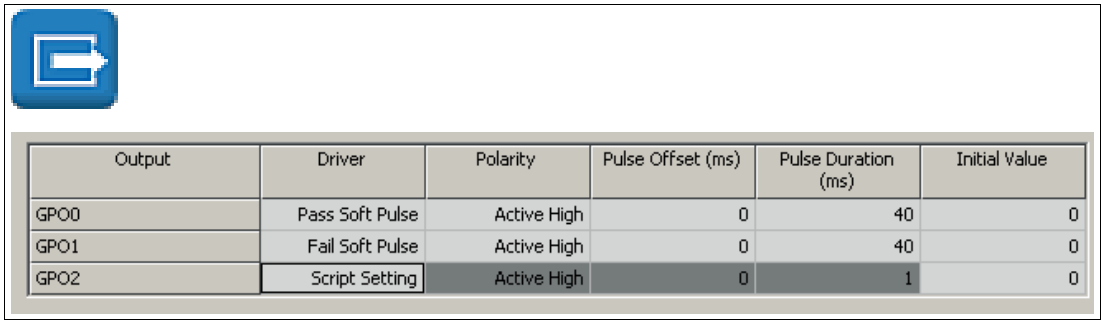


Figure 7.104 Output menu

Function	Description	
Driver	Use the drop-down list to select what is used to control the output:	
	Script Setting	The output is controlled by equations or instructions in the script. This value prevents the other fields in the row from being displayed.
	Pass Pulse	An "active pulse" is output if the result is "Pass."
	Recycle Pulse	An "active pulse" is output if the result is "Recycle."
	Fail Pulse	An "active pulse" is output if the result is "Fail." Skipped parts are counted as "Fail."
	NOTE: The "Pass" pulse, the "Recycle" pulse, and the "Fail" pulse are hardware-controlled pulses. The offset and the clock start with the image capture trigger. This is a "deterministic" time measure. If a decision result is not available when the pulse shift time is reached, a "Fail" pulse is output by default. Minimum pulse offset = exposure time + capture time + inspection time Runtime overrange (e.g., displaying or saving images) can also affect the request for the minimum pulse offset. You should investigate or calibrate this time based on your specific system usage and use of other devices in the same network.	
	Strobe Pulse	A pulse is output after a trigger input. The pulse uses the settings from the sensor settings menu
	Pass Soft Pulse	An active (polarity value) pulse occurs when the result of the job is "Pass." This is a software-controlled output. The offset and time begin with the availability of the decision result. This is a "non-deterministic" time sequence.
	Fail Soft Pulse	An active (polarity value) pulse occurs when the result of the job is "Fail." This is a software-controlled output. The offset and time begin with the availability of the decision result. This is a "non-deterministic" time sequence. Skipped parts are also counted as "Fail."
Polarity	Use the drop-down list to select "Active High" or "Active Low" for the output pulse.	
Pulse Offset	Enter a value for the output pulse delay to synchronize it with external devices.	
Pulse Duration	Enter a value for the output pulse width.	
Initial Value	Enter a value for the state or value of the output when the job is loaded.	

7.4.3 Image Logging

When you click "Setup Image Logging", the settings menu for image logging opens in the configuration and status window. You can use the "Image Logging" function to store a limited number of images and data in the sensor memory. The "Image File Logging" function can be used to save sensor images to the PC.

Setting the History and Image Logging

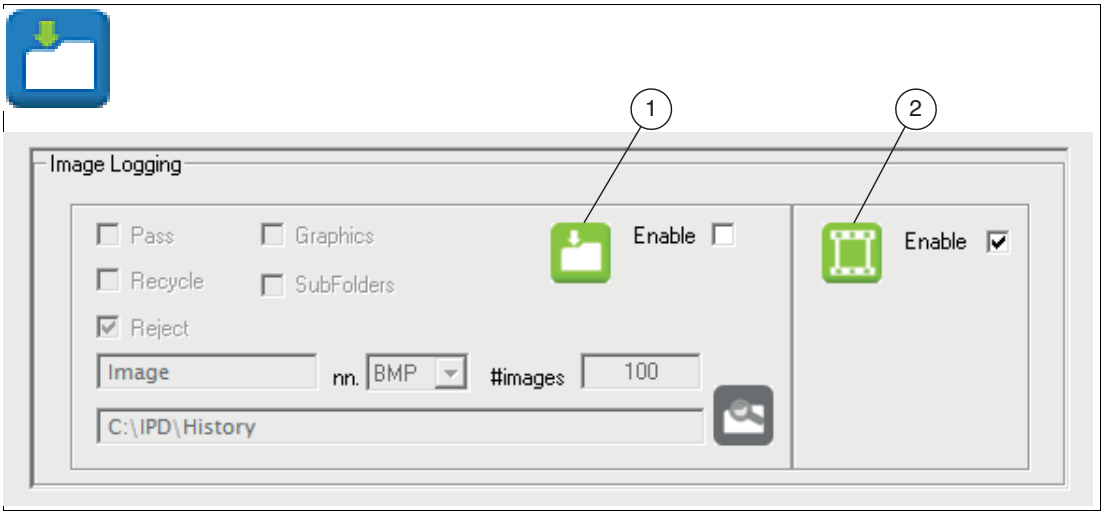


Figure 7.105 Image Logging menu

Saving Sensor Images to the PC (Image File Logging)

The number field (#images) is the maximum number of images. Images are stored under file names with consecutive numbers (image0.bmp, image1.bmp). The counter continues to run on this number and will overwrite older images until you disable image logging.

1. Select the "Enable" check box (1).

Note

To stop saving images, return to this operating panel, clear the check box next to "Enable" (1), and close the operating panel.

2. Enter the maximum number of images to be saved in "#images."
3. Choose a category: "Pass", "Recycle", or "Reject."
4. If you select more than one category, select "SubFolders" to create subdirectories for "Pass", "Recycle", and "Reject."
5. Check the "Graphics" check box to save measurement graphics. If the box is cleared, only camera images will be saved.
6. Change the file name if necessary (the default name is "Image").
7. Select the image format: BMP or JPG.
8. Specify the target directory.

2021-03

Saving Images and Results in the Sensor

Select the "Enable" check box (2) for the "Image Logging" function to save images and/or data in the sensor.

Note

There are three categories: Pass, Recycle, and Reject. Each category saves the last 20 images at a resolution of 640 x 480 pixels, two or four images at a resolution of 1280 x 960 pixels. Clear the "Enable" check box to prevent images or data from being saved in the sensor. This will shorten the test periods. The history log appears in the "Run solution" window when you click "History Recall" in the execution section (see chapter 7.6).

7.4.4 RS-232

The serial connection or "RS-232 stream" is used for general communication such as messages and data.

Figure 7.106 RS-232

RS-232 Example

You can create data strings using the Free Edit editor and the String Formatting editor (see chapter 7.5.1). You can use the following functions in script instructions to read and write on the serial interface:

PutPortString GetPortChar GetPortString WriteFormatString

The "PutPortString" function sends the exact string, without evaluating variables. The "WriteFormatString" function evaluates variables.

Serial I/O Functions

Function	Description
GetPortChar( )	Returns a new character input from the COM serial interface, if any, otherwise returns a value of zero.
GetPortString(endingChar)	Returns a new data string entered by the COM serial interface, if any, otherwise an empty data string is returned immediately. **endingChar**: Specifies the received character to indicate the end of a received character string. This character is not included in the returned character string.
PutPortString(string)	Sends a string to the COM serial interface. Does not perform evaluations for embedded variables.

7.4.5 TCP/IP Stream

The TCP/IP stream is used for general communication with devices, e.g., for outputting messages or data.

Note

Windows Firewall should be switched off on systems that communicate via the networkor should be setaccordingly. See chapter 8.1.

TCP/IP Interface Example

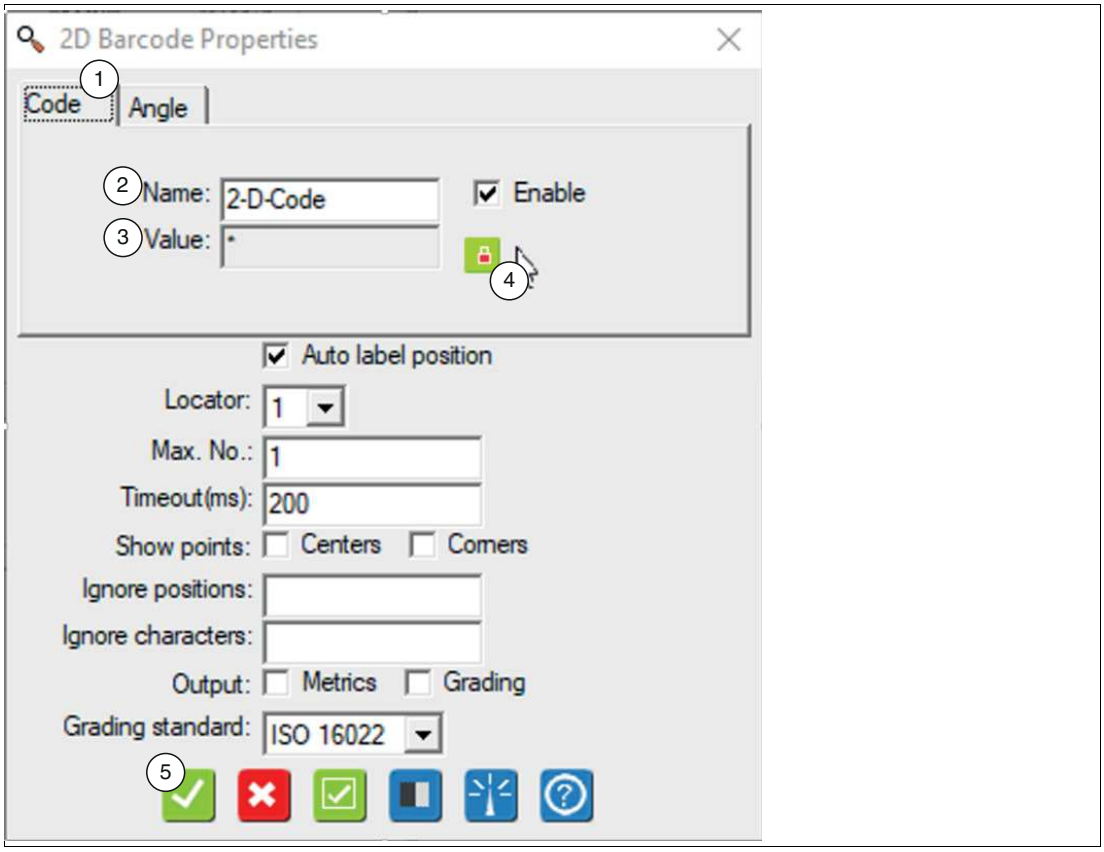


Figure 7.107 Properties window

1. Create a new job. See chapter 7.1.
2. Set the output variable "Name" (2) (example: 2-D code tool) in the properties window under the "Code" tab.
3. Place a "*" symbol as a placeholder for multi-digit characters in the input field of the output variable "Value" (3).
4. Click the padlock symbol (4) to lock the variable.
5. Click the "Accept" button (5).

↳ The changes are applied and the properties window closes.

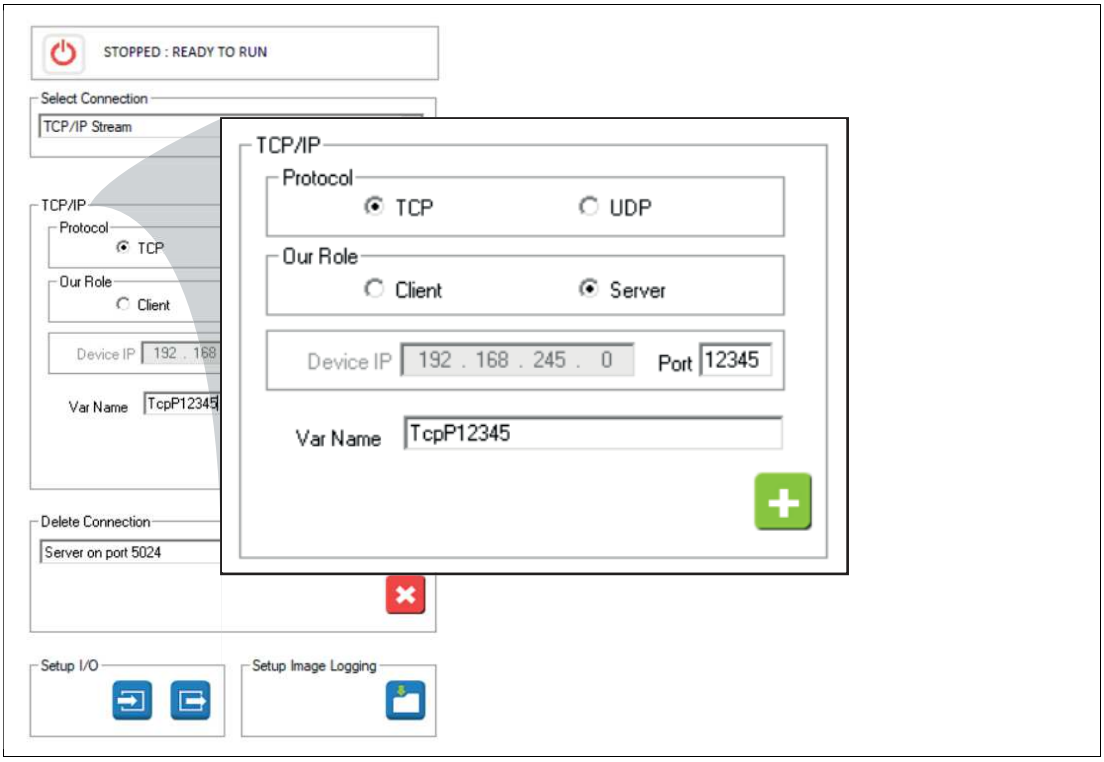


Figure 7.108 Interface parameters and a variable name

6. Set the interface parameters and a variable name for the interface. See table "Parameter Description" on page 180.

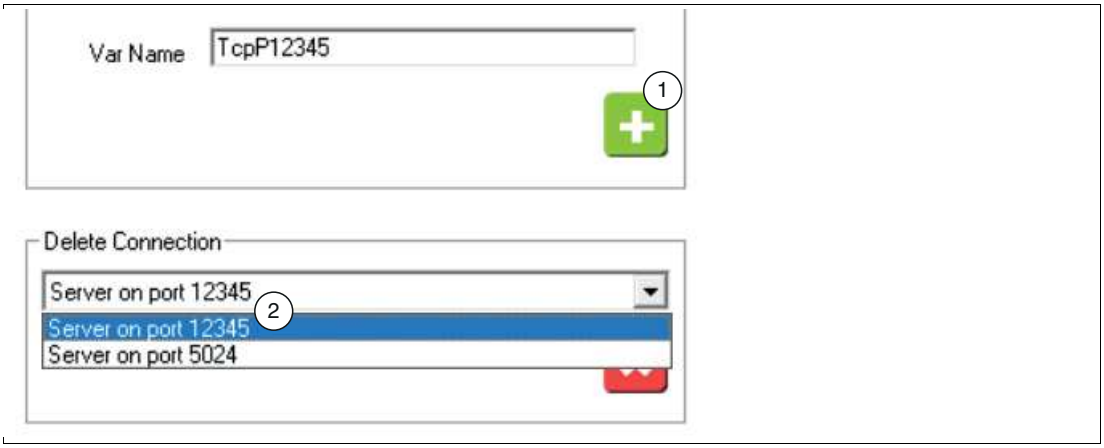


Figure 7.109

↳ The TCP/IP interface is available in the variable tree once adding it using the "+" symbol (1).

7. To set a script to output the code via the TCP/IP interface, open the script editor (see chapter 7.5) in the navigation bar.

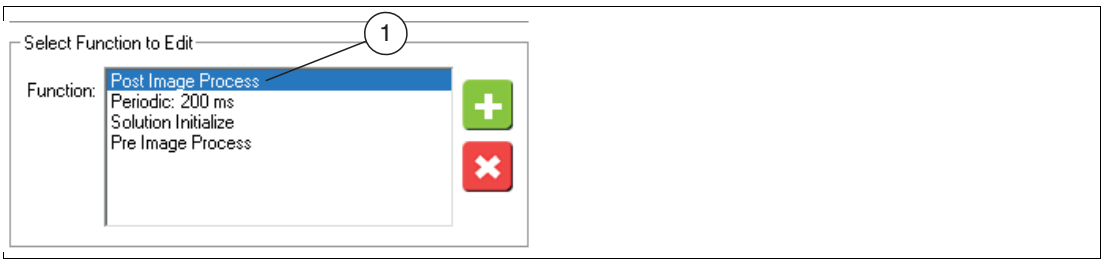


Figure 7.110 Selecting function blocks

2021-03

PEPPERL+FUCHS

8. Select the "Post Image Process" function block from the list (1).

   ↳ The selected function block is retrieved after processing the captured image.

9. Create a script to write the output variable of the vision tool to the interface variable (in this example: TcpP12345). To do this, enter your script function (1) in the input window.

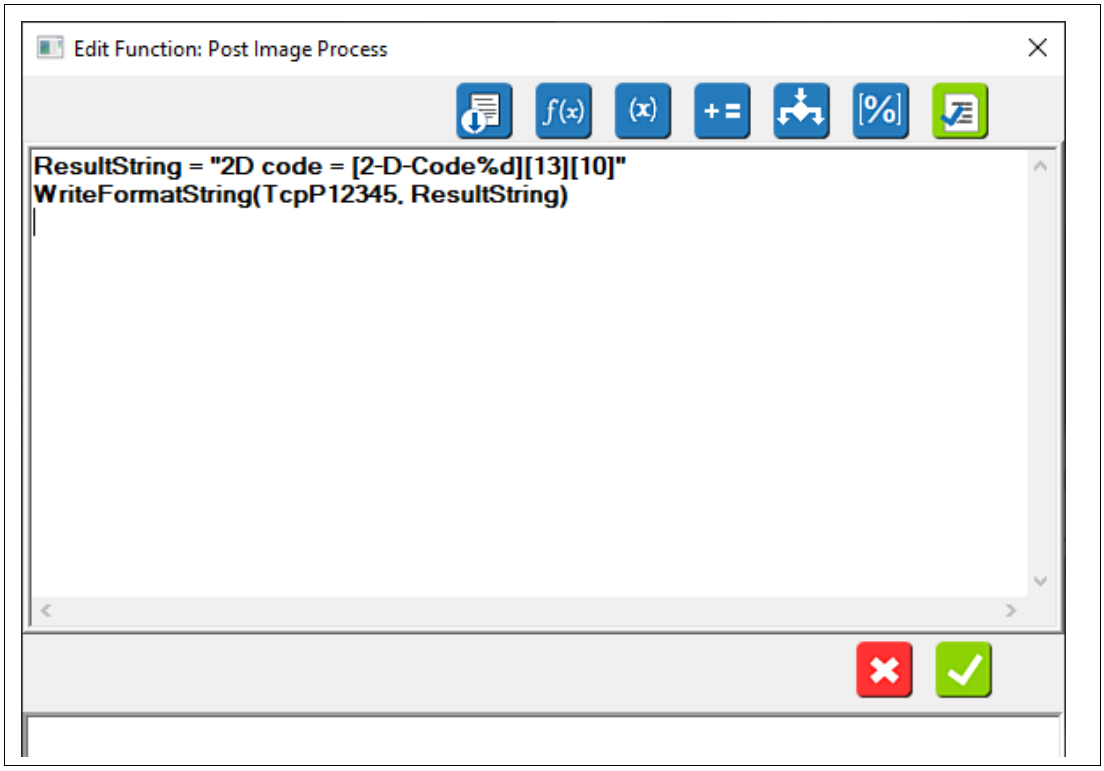


Figure 7.111 Script function

Example

```
ResultString = "2D code = [2-D-Code%d][13][10]"
WriteFormatString(TcpP12345, ResultString)
// WriteFormatString() function.
// Control character: [13] = Carriage Return, [10] Linefeed
```

10. Click "OK" (2) to confirm your entry.

Figure 7.112 Application test

11. Test your application by clicking the "Run solution" button in the navigation bar to start an "application test" (1). See chapter 7.6.

Parameter Description

Parameter	Description
Protocol	Select TCP or UDP protocol
Client	The VOS software sends data to a network server.
Server	Other computers request data from the VOS software. The requests have no significant impact on the inspection throughput.
Device IP	The IP address of the server to which the data is sent.
Port	If VOS is a server, assign a port number. If VOS acts as client, enter the port number used by the server. Port 5024 is the default assignment and is available to assign your own format and your own conditions. The following port numbers are reserved and cannot be used: all port numbers lower than and including 1024, 5005 to 5023. You should use port number 5024 or higher.
VarName	This field displays a default variable name (e.g., "TCPIP5024"). You can use the suggested name or change it to a more meaningful name.

TCP/IP Example

In this example, port 5025 is used for communication. The sensor uses the shape detection tool (Match Tool) to find a part and sends the image pixels, X,Y coordinates, and angle to a PC.

Function: **Post Image Process**	
WriteFormatString(TCPP5025 , "\n\rX=[PX%3.2f], Y=[PY%3.2f], Theta=[MR%3.1f], Result = [Result.0%d]\n\r")	// X, Y, angle, and result

The sensor receives commands to trigger and modify the jobs. The first character in the string is the trigger. The second character is a job number. The current job is 0. The "T2" character string changes to job 2 and triggers the camera. The job test instruction differs from job to job. This test is: "not job 0" (the current job) and "a number less than 9" (valid job numbers 0 to 8).

```
Function: Periodic: 200 ms
ReadBuffer= ReadString( TCPP5024 , 13 )                                  // 13 is the end character
if(ReadBuffer!= "")                                                      // if NOT an empty data string
                    CommandString= ReadBuffer
                    Counter = Counter + 1                                // examine one character at a time
                    CommandCharacter= Substring(Command-String, 0, 1)    // trigger command
                    ProgramNumber= Substring(Command-String, 1, 1)       // job number
                    if(INT(ProgramNumber) > 0                            // Check job number
                                        ChangeSolu-tionID(Pro-gramNumber)
                                        endif
                                        if(Command-Character= "T")
                                        trigger()
                                        endif
endif
```

Note

The check in job change should be adjusted for each job file. The job should be changed if the "ProgramNumber" is not equal to the current ID number of the job. In the sample script shown above, the current job is 0 and the change request is valid if ProgramNumber is greater than zero. For example, you can use the following in Job02: if(ProgramNumber != 2) means "!=" "not equal to".

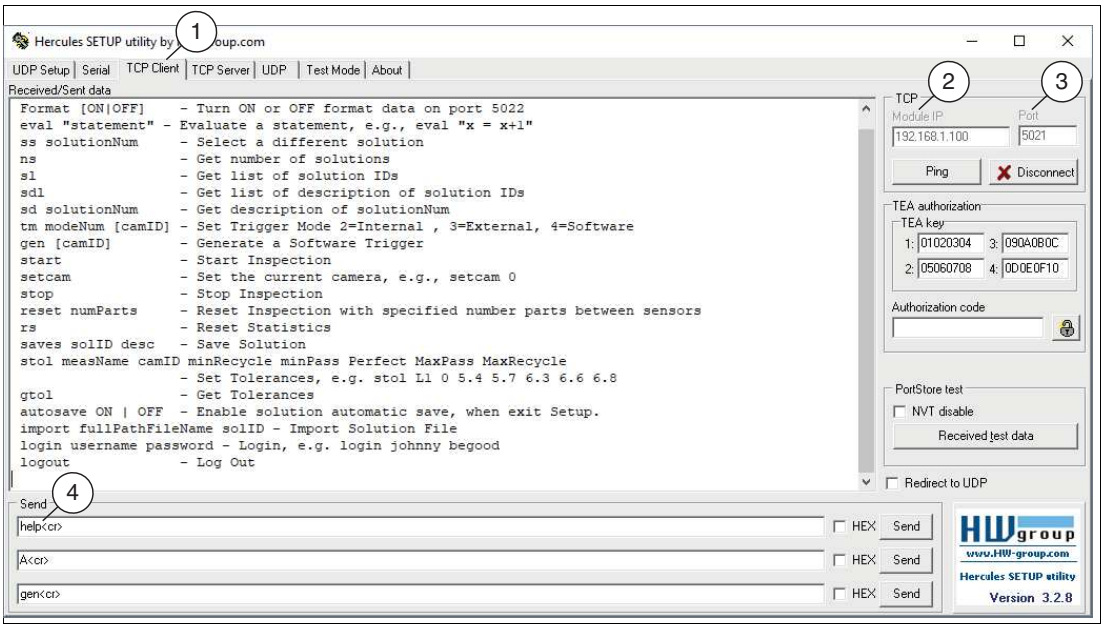
You can limit the test to valid job numbers. We do not recommend doing this, since each time you add a new job eachjob file must be edited.

Standard Interface Commands

You can issue commands to the sensor over the TCP/IP network to select the jobs in progress or to temporarily change the trigger mode. The spare port 5021 is intended for receiving commands.

Displaying Interface Commands

To display interface commands, you need a port monitor. The following example uses the "Hercules SETUP utility" port monitor from HW Group.



1. Open the "Hercules SETUP utility" program and select the "TCP Client" tab (1).
2. Under "TCP", enter the IP address (2) of the sensor and enter the port number 5021 (3).
3. Type "help" (4) followed by Enter or Return to display a list of available commands.

Interface Commands

Command	Description
Format [ON|OFF]	Enables or disables the preformatted output at port 5022 (the history log or iOutputLog). A job must be active.
eval "statement"	Evaluates an instruction once. This can change the values of arguments defined in the equation assignment control panel or create new arguments. Example: eval x returns the current value of x. eval x=5 changes the value from x to 5. If the job contains an instruction that uses x, the behavior may change due to the new value.
ss##	Changes the ID of the job in progress The new job must be saved on the sensor. The job in progress is not changed if the new ID is not a valid job.
ns	Returns the number of jobs in the sensor memory.
sl	Returns a list of the job ID numbers used (separated by commas).
sdl	Returns a list of job descriptions (separated by commas).
sd ##	Returns a description for the specified job ID number.
tm #	Changes the trigger mode. This is a manual or temporary change. The saved job settings are not changed. When you navigate to the "Sensor" menu, the trigger settingsare reset to the original settings of the job. 2 = internal timer, 3 = external sensor trigger, 4 = software trigger. 0 and 1 are not valid. Note: In software trigger mode, the sensor can react to the external trigger and the software trigger. If there is no external trigger, the image capture waits for the software trigger. If there is an external trigger, the image capture responds to the trigger and does not wait for your software trigger.
gen	Creates a software trigger. This is used after setting the temporary trigger mode to **tm 4**.
start	Start or restart the test.
stop	Stop or pause the test.
reset #	Reset the production linewith a specific number of parts between the two sensors.
rs	Reset pass/recycle/fail counter statistics.
saves ## Desc	Saves the current job using the specified ID number and description character string.
gtol	Returns the tolerances for all image captures in the following sequence: measName minRecycle minPass Perfect MaxPass MaxRecycle. See the two examples in **stol**.
stol <values>	Set tolerances for the specified measurement. The number of arguments changes as Recycle is enabled/disabled. The camID is always 0 for the sensor. If Recycle is enabled: **stol measName camID minRecycle minPass perfect MaxPass MaxRecycle** For example: stol L1 0 5.4 5.7 6.3 6.6 6.8 then gtol returns the current tolerance values: measName minRecycle minPass Perfect MaxPass MaxRecycle L1 5.4 5.7 6.3 6.6 6.8 If Recycle is disabled: **stol measName camID minPass 0 Perfect 0 MaxPass** For example: stol L1 0 5.4 0 6.3 0 6.8 then gtol returns the current tolerance values: measName minPass ignored Perfect ignored MaxPass L1 5.4 0.0 6.3 0.0 6.8
autosave [ON|OFF]	Enables or disables saving the job when you exit the VOS software.

Command	Description
export FullPathFile-Name ##	Exports the identified job number under the specified path and name.
import FullPathFile-Name ##	Imports the specified job file and saves it under the specified ID number.
login username password	Log in to a password-protected account (if the administrator has enabled passwords and accounts).
logout	Log out

7.4.6 EtherNet/IP

Setting up the Connection

Menu	Description
Register Type	Select a register type here from the drop-down list for the variable that is attached to the PLC. For example, "int - signed 16."
VariableName	A default variable name will appear (e.g., "EIPint") once the register type has been selected. You can use the suggested name or change it to a more meaningful name.
Reg Base Addr	This field is used to assign a basic index to the register and the variable. The default base address "0" is usually the appropriate option. The attached variable is used as an array. Therefore, any base index specified here will be added to your array index to form the total offset. If you specify a "Reg Base Addr" of "100" for an attached "EIPint" variable, a reference to "EIPint[80]" would reference a position with an offset of 180. (Reg Base Addr 100 + Array Offset 80).

EtherNet/IP Example

This example describes how to add scripts to communicate with the PLC.

1. Click "Edit scripts" in the navigation bar.
2. Click "Post Image Process" in the settings window (left).
3. Click "Edit" at the bottom, below the image area.
4. Create equations or instructions for the control panel. Add your variables. Your variable name appears in the list of variables.

Example

Function: **Post Image Process** //send results to the SPS input register.

EIPdint[0] = IntenAvg

EIPdint[1] = L

EIPdint[2] = Result.0

// Check is complete

inspBusy = 0

5. Click "Check Syntax" to check for errors.
6. Click "Save" to save and to close the "Free Edit" window.

Note

You usually want the sensor to switch jobs and trigger a signal from the PLC:

7. Click "Periodic: 200 ms" in the "Edit scripts" settings field.
8. Click "Edit" at the bottom, below the image area.
9. Add instructions that indicate to the control panel what job is being executed. Read a job change request from the output registers of the control panel. (See example below).
10. Add instructions that trigger the sensor, delete, and then reactivate the trigger.

Example

Send results to the PLC input register:

Function: **Periodic: 200 ms**

```
//Send the current jobs to the PLC
EIPdint[3] = GetSolutionID()
EIPdint[4] = Global.FrameCount
//Query any job change request from the PLC output register
solReq = EIPdint[1]
if(solReq > 0) //this test instruction is different in each job
    ChangeSolution(solReq)
endif
//
//Query any trigger requests from PLC output registers
trigReq = EIPdint[0]
//Trigger only on leading edge of the register transition
if((trigReq = 1) AND (trigArmed = 1))
    trigArmed = 0 //prevents multiple triggering.
    inspBusy = 1 //the check is performed
    trigger()
endif
//Restart trigger if PLC register is 0
if(trigReq = 0) trigArmed = 1
//Record the status of the PLC connection
plcStat = IsConnected(EIPdint)
//
//Sends feedback number
hb = hb + 1
if(hb > 999) hb = 1
EIPdint[5] = hb
```

11. Click "Check Syntax" to check for errors.
12. Click "Save" to save and to close the "Free Edit" window.
13. Save the job, reload the job, and then execute the job.

Configuring the Control Panel

This example shows the RSLogix5000 control panel.

1. Open the PLC programming environment.

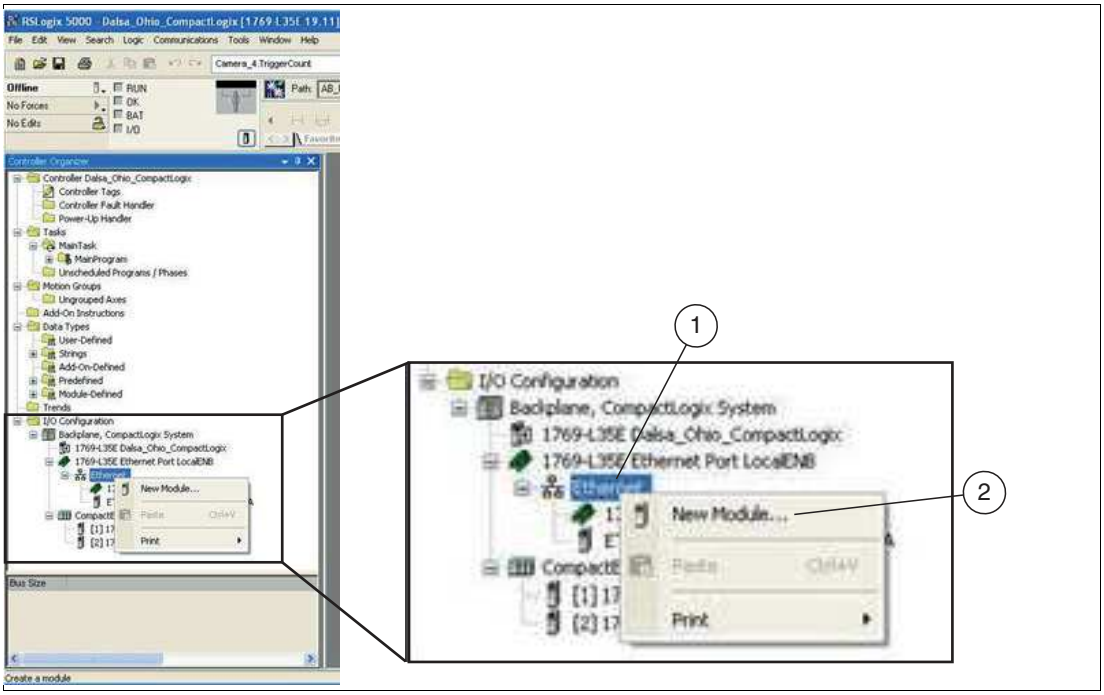


Figure 7.113 Ethernet—selecting a new module

2. Right-click on the Ethernet connection "Ethernet" (1) and select "New Module..." (2).

↳ This opens the "Select Module" selection window.

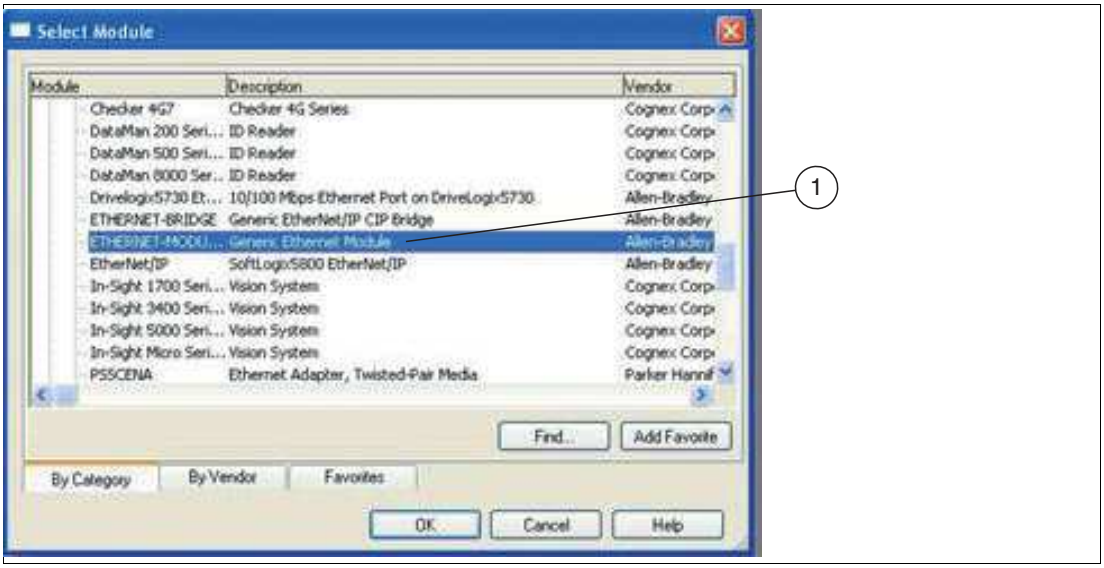


Figure 7.114 "Select Module"

3. In the "Select Module" selection window, select the "Generic Ethernet Module" (1) entry.

2021-03

PEPPERL+FUCHS

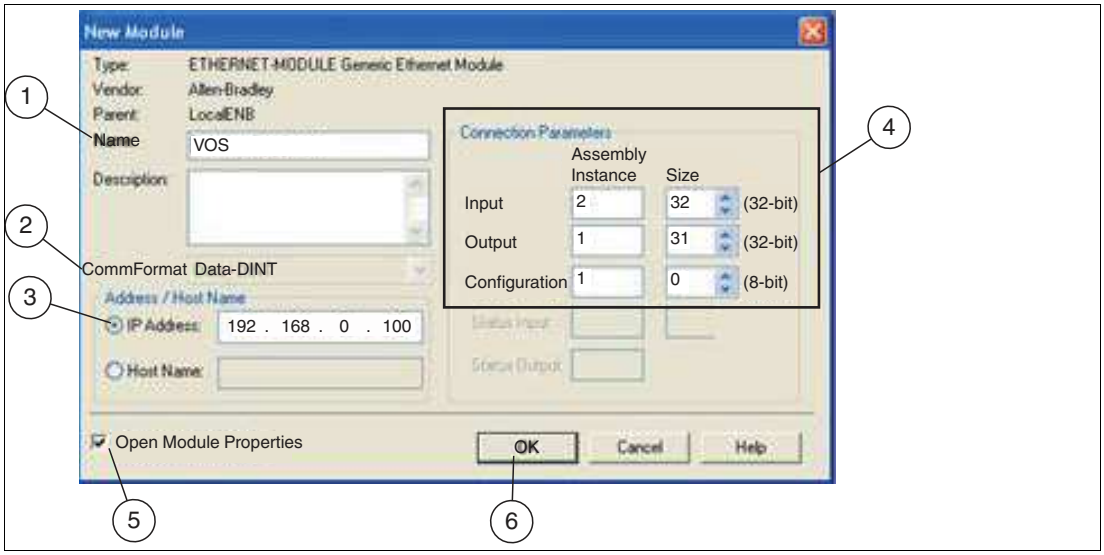


Figure 7.115 Module connection properties

4. In the "New Module" menu, give the sensor a name (1) such as VOS. Keep the CommFormat as Data-DINT (2). Enter the IP address (3) of your sensor.
5. Enter the input, output, and configuration data for the "Assembly Instance" and "Size" (4) as shown:
   - Input: 2, 32
   - Output: 1, 31
   - Configuration: 1, 0
6. Make sure the "Open Module Properties" (5) check box is selected and click "OK" (6).

Note

If more than 30 register values need to be read and written, you could use the following settings:

- Input: 2, 125
- Output: 2, 124,
- Configuration 1, 0

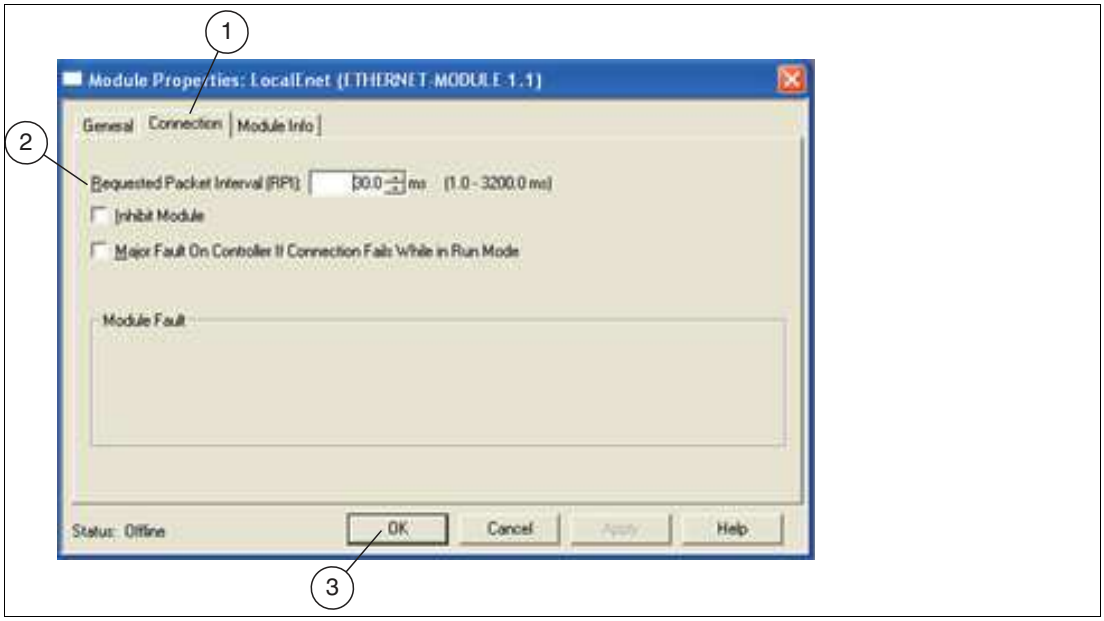


Figure 7.116 Module properties

7. In the "Module Properties" menu, click on the "Connection" tab (1). Enter a "Requested Packet Interval (RPI)" (2) that is not less than 30 ms. This is the time interval during which the Logix processing unit requests new data from the sensor. Depending on network traffic, this number must be significantly higher (150 ms).
8. Click "OK" (3) to confirm your entries.
9. Save your settings and load them to the PLC.

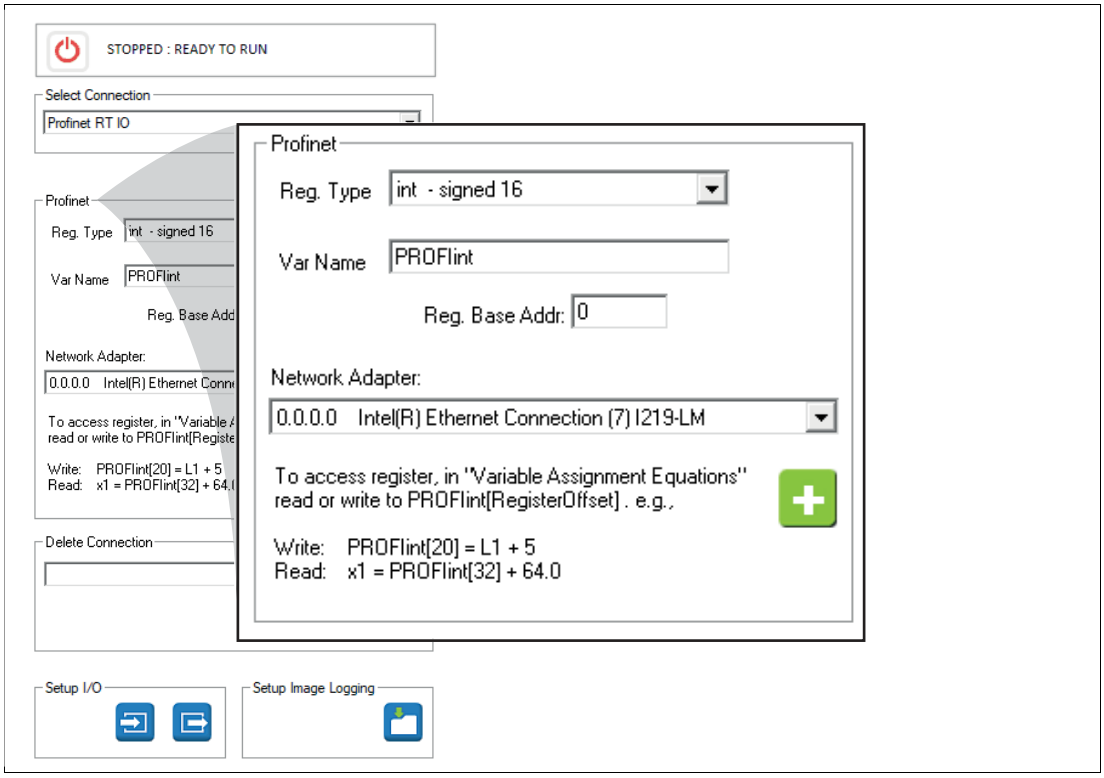
↳ The program is compiled before downloading to the PLC. You can also compile the program manually before downloading it.

Note

The sensor writes to EIP[0] and maps to the PLC position I:DATA[1] (I for input). This does not happen during a read operation. EIP[0] is mapped to the PLC position O:DATA[0].

7.4.7 PROFINET

The sensor is compatible with PLCs that use the PROFINET UDP or PROFINET RT protocol.



Setting up the Connection

Menu	Description
Reg. Type	Select a register type from the list. This defines the data type or data size.
VariableName	A default variable name is displayed in this field. The default name is based on the data type (e.g., "PROFIint"). You can use the suggested name or change it to a more meaningful name.
Reg. Base Addr.	This field is used to enter a register base address if your system requires it.

PROFINET Example

This example describes how to add scripts to communicate with the PLC.

1. Click "Edit scripts" in the navigation bar.
2. Click "Post Image Process" in the settings window (left).
3. Click "Edit" at the bottom, below the image area.
4. Create equations or instructions for the control panel. Add your variables. Your variable name appears in the list of variables.

Example

Function: **Post Image Process**

//Send results to the PLC input register.

PROFIdint[0] = IntenAvg

PROFIdint[1] = L

PROFIdint[2] = Result.0

5. Click "Check Syntax" to check for errors.

Note

When the sensor writes to the attached PROFIdint variable, it updates the input module on the PROFINET control panel (I address range 256 ... 509). PROFIdint is a 32 bit signed value. The sensor page references therefore index an array of the dint size.

PROFIdint[0] = x, writes the value that appears at position %ID256 on the control panel,

PROFIdint[1] = x, writes the value that appears at position %ID260 on the control panel, which is the next double-size index.

6. Click "Save" to save and to close the "Free Edit" window.

Note

You usually want the sensor to switch jobs and trigger a signal from the PLC:

7. Click "Periodic: 200 ms" in the "Edit scripts" settings field.
8. Click "Edit" at the bottom, below the image area.
9. Add instructions that indicate to the control panel what job is being executed. Read a job change request from the output registers of the control panel. (See example below).
10. Add instructions that trigger the sensor, delete, and then reactivate the trigger.

Example

Send results to the PLC input register:

Function: **Periodic: 200 ms**

```
//Send the current jobs to the PLC
PROFIdint[3] = GetSolutionID()
PROFIdint[4] = Global.FrameCount
//Query any job change request from the PLC output register
solReq = PROFIdint[1]
if(solReq > 0) //this test instruction is different in each job
    ChangeSolution(solReq)
endif
//Query any trigger requests from PLC output registers
trigReq = PROFIdint[0]
//Trigger only on leading edge of the register transition
if((trigReq = 1) AND (trigArmed = 1))
    trigArmed = 0 //prevents multiple triggering.
    trigger()
endif
//Restart trigger if PLC register is 0
if(trigReq = 0) trigArmed = 1
```

11. Click "Check Syntax" to check for errors.
12. Click "Save" to save and to close the "Free Edit" window.
13. Save the job, reload the job, and then execute the job.

Note

When the sensor writes to the attached PROFIdint variable, it reads the output module on the PROFINET control panel (Q address range 256 ... 509).

profiCmd= PROFIdint[0], reads from memory %QD256 to the control panel,

profiCmd= PROFIdint[1], reads from memory %QD260 to the control panel,

…….

profiCmd= PROFIdint[62], reads from memory %QD504 to the control panel,

Commissioning with Siemens TIA Portal

Creating a Project

To create a project, proceed as follows:

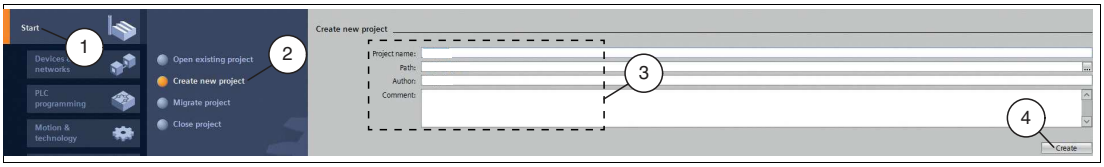


Figure 7.117 Creating a project

1. Launch the TIA Portal.
2. In the portal view, select **Start** (1) > **Create new project** (2).
3. Define fields for the project (3), e.g., by entering a name for the project in the **Project name** field.
4. Use the **Create** button (4) to confirm your entry.

Integrating the Control Panel

To integrate the control panel, proceed as follows:

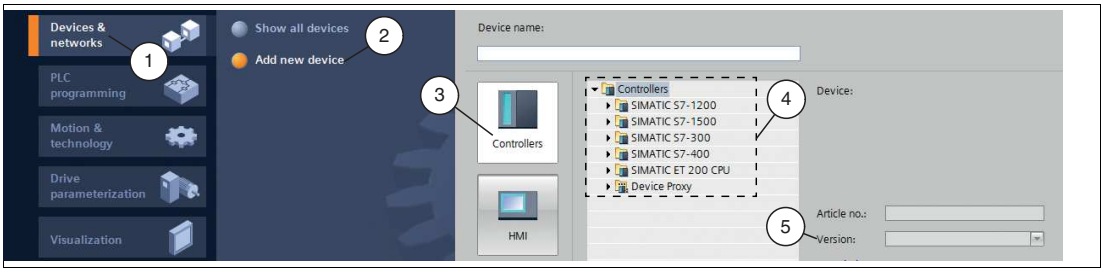


Figure 7.118 Integrating the Control Panel

1. In the portal view, select **Device & networks** (1) > **Add new device** (2).
2. Click the **Controllers** button (3).
3. Select your control panel from the hardware catalog (4). Make sure that you select the right firmware version of the control panel in the **Version** field (5).
4. Double-click on **Add** to add the control panel to the project.

↳ The project view opens.

Note

Now configure the control panel as required. For example, define settings for the PROFINET interface or startup/cycle behavior. Additional information can be found in the Siemens AG user documentation.

Installing the GSDML File

You require a GSDML file to operate the sensor. The GSDML file can be downloaded from our website: www.pepperl-fuchs.com. Enter the product name or item number in the Product/Keyword search box and click on Search. Select your product from the list of search results and click on the Software tab in the product information list. A list of all available downloads is displayed. To install the GSDML file, proceed as follows:

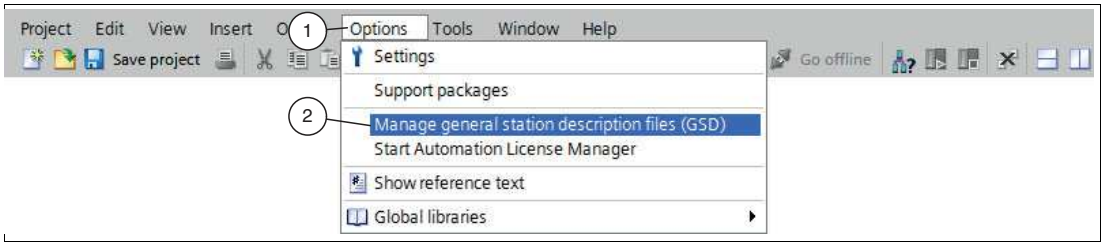


Figure 7.119 GSDML file

1. Select **Options** (1) > **Manage general station description files (GSD)** (2).

   ↳ The **Manage general station description files** window opens.

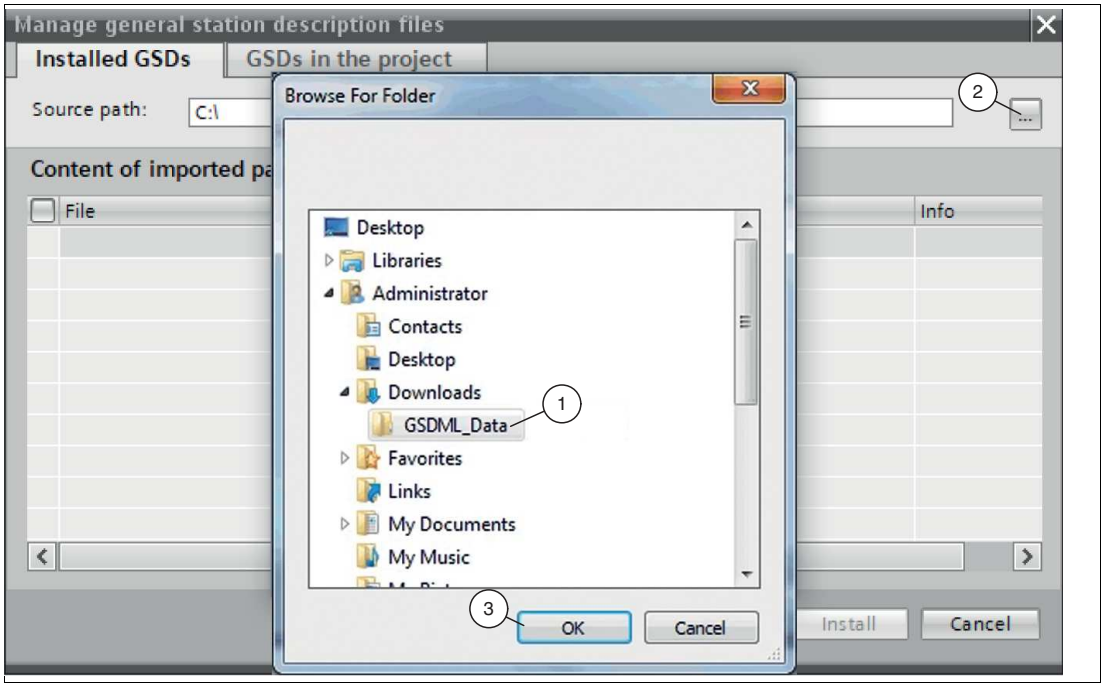


Figure 7.120 Searching for a GSDML file

2. Click the **Browse** button (2) to search for the GSDML file on your computer.
3. Select the folder containing the GSDML file (1).
4. Click **OK** (3) to confirm your selection.

   ↳ All the GSDML files in the selected folder are displayed in the list.
5. Select the relevant GSDML file by checking the box to the left of the file name.
6. Click **Install**.

   ↳ The installation process starts automatically.

   ↳ Once the file is installed successfully, the system issues a notification that installation was successful. Close this window. The device data is added to the hardware catalog.

Note

Ensure that the project is in offline mode. Otherwise, it will not be possible to replicate the hardware components in the configuration software.

Integrating the Sensor

To integrate the sensor and connect it to the control panel, proceed as follows:

1. In the project tree, right-click on the control panel and select **Go to network view**.

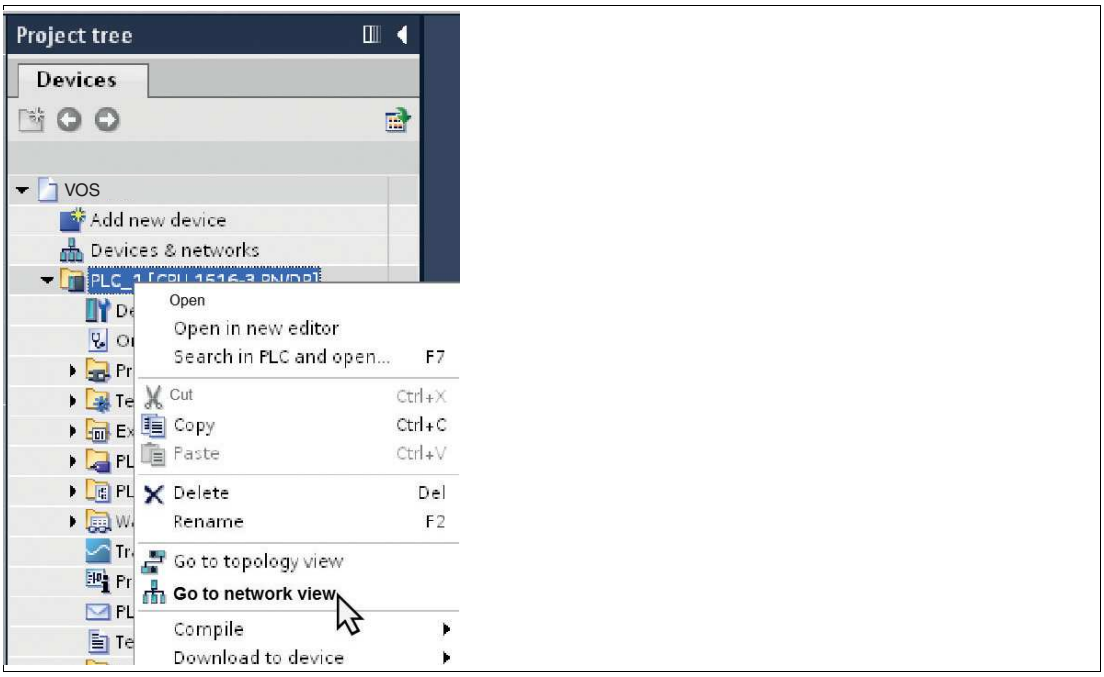


Figure 7.121 Selecting the network view

↳ The **Network view** opens in the work area.

2. Open the hardware catalog and navigate through the tree structure (1) to your sensor.
3. Select your sensor from the hardware catalog (1) and drag and drop it into the network view (4).

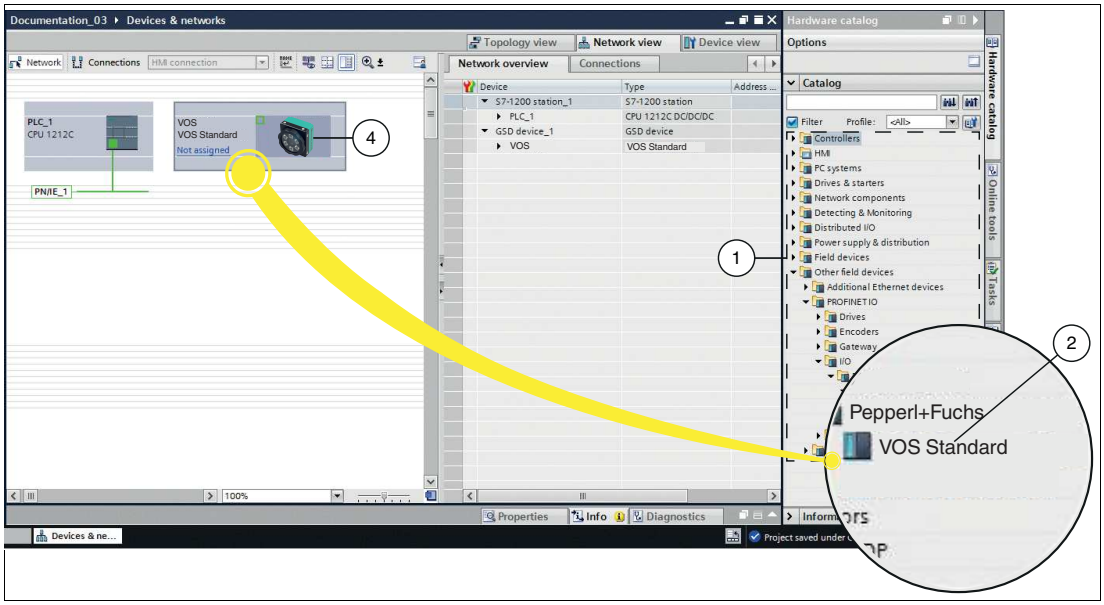


Figure 7.122 Integrating the sensor

↳ The sensor is displayed in the network view window (4).

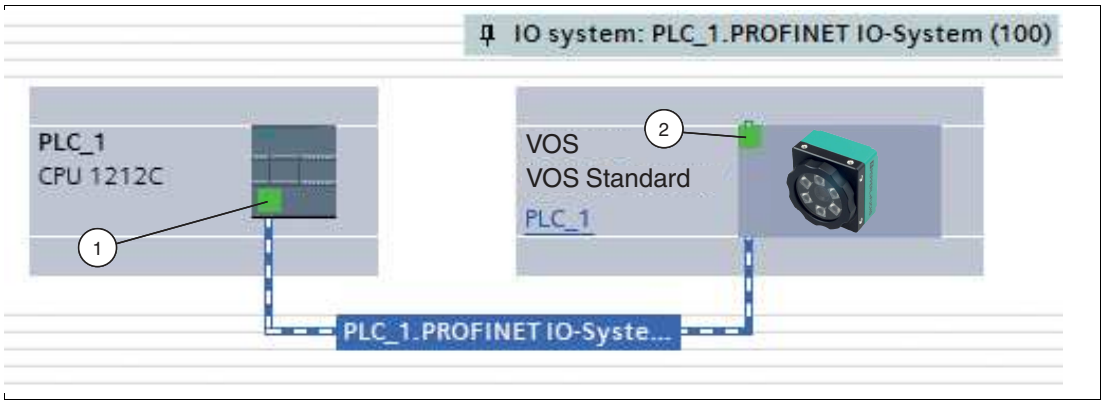


Figure 7.123 Connecting the sensor to the control panel

4. To connect the sensor to the control panel, proceed as follows:

1. Click the green PROFINET interface (1) for the control panel and do not let go of the mouse button.
2. Drag the line to the PROFINET interface (2) on the gateway.
3. Release the mouse button.

↳ The sensor is now connected to the control panel.

Setting the Sensor Name and IP Address

For the sensor to be addressed as a participant in the PROFINET network, the following parameters are required:

1. In the **Devices and networks** window, switch to the **Device view** tab.
2. Select your sensor from the drop-down list. In this example, the sensor is called "VOS."

   ↳ The sensor properties appear in the inspection window.
3. Open the **Properties** (1) > **General** (2) tabs.
4. Click the **Ethernet addresses** subtree node (3).

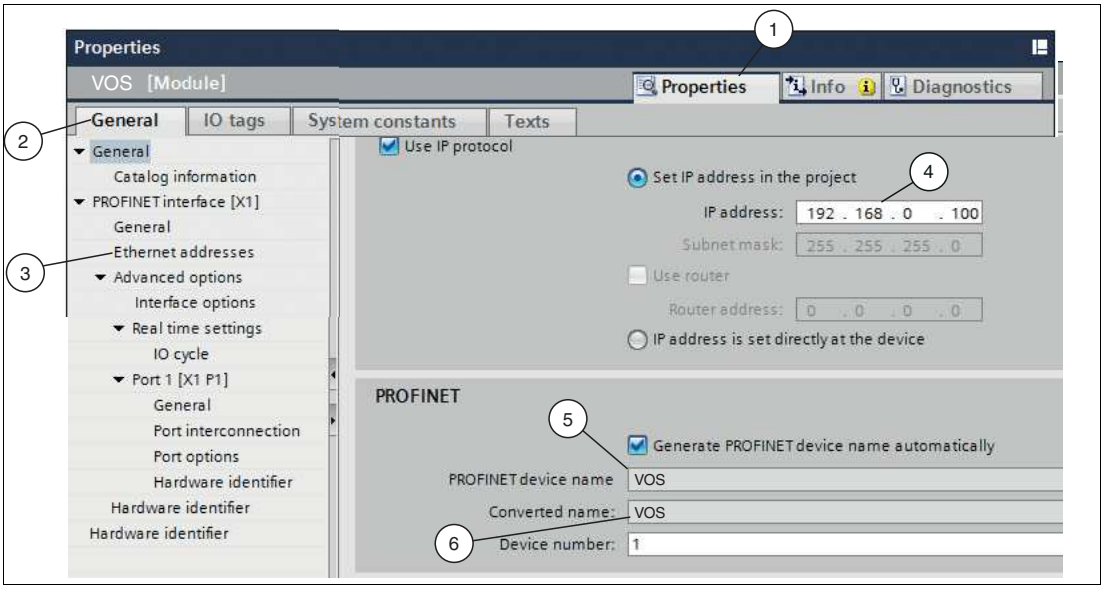


Figure 7.124 Sensor properties

5. If you have selected **Set IP address in the project**, define the IP address in the project. To do this, enter the address of your sensor in the **IP address** (4) field.

Note

Sensor name

In the PROFINET system, each device has a unique device name (symbolic name). In the configuration program, the device name is automatically assigned using the device name from the GSDML file and a sequential number. If using a sensor, the "PROFINET device name" (5) and "Converted name" (6) fields are identical. You cannot change the device name in the view shown above. The "PROFINET device name" (5) sensor must match the name in the "Converted name" field (6), so that PROFINET can connect correctly. If necessary, adapt the device name, see "Changing the IP Address" on page 46.

Setting the Watchdog Time

1. Check the "Watchdog time" (3). Go to the "General" (1) tab and select "Advanced options > Real time settings" (2).
2. If necessary, reset the "Update time" (5) and the "Watchdog time" (3). You may need to experiment with the values to determine which watchdog and cycle time works best for your program and your environment. Larger values may be required for a more complex program, a slow trigger rate, or a slow product line.

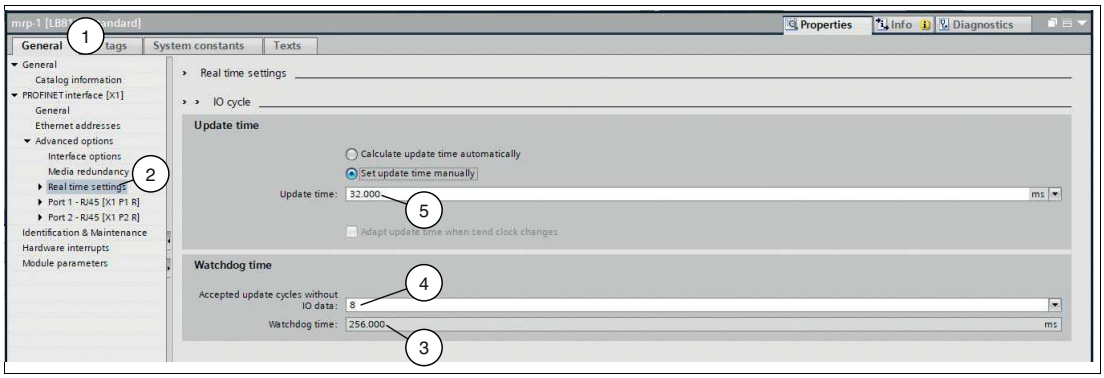


Figure 7.125 Watchdog time

Note

The **"Watchdog time"** (3) is calculated from the **"Update time"** (5) multiplied by **"Accepted update cycles without IO data"** (4).

If, as in this example, eight cycles are accepted without IO data, the resulting watchdog time is 256 ms.

7.5 Script Settings

Creating scripts allows you to define or modify system control and integration functions. The Script Tool differs from the rest of the user interface in that it supports the use of functions and more traditional programming constructs.

Script Editor

In the Script editor, you can preprocess the variables or data outputs of the vision tools and send them to a communication interface or activate/control the switching outputs according to a script logic. You can define commands to control the camera via an external interface. This allows you to perform tasks such as job changeovers or triggering image captures.

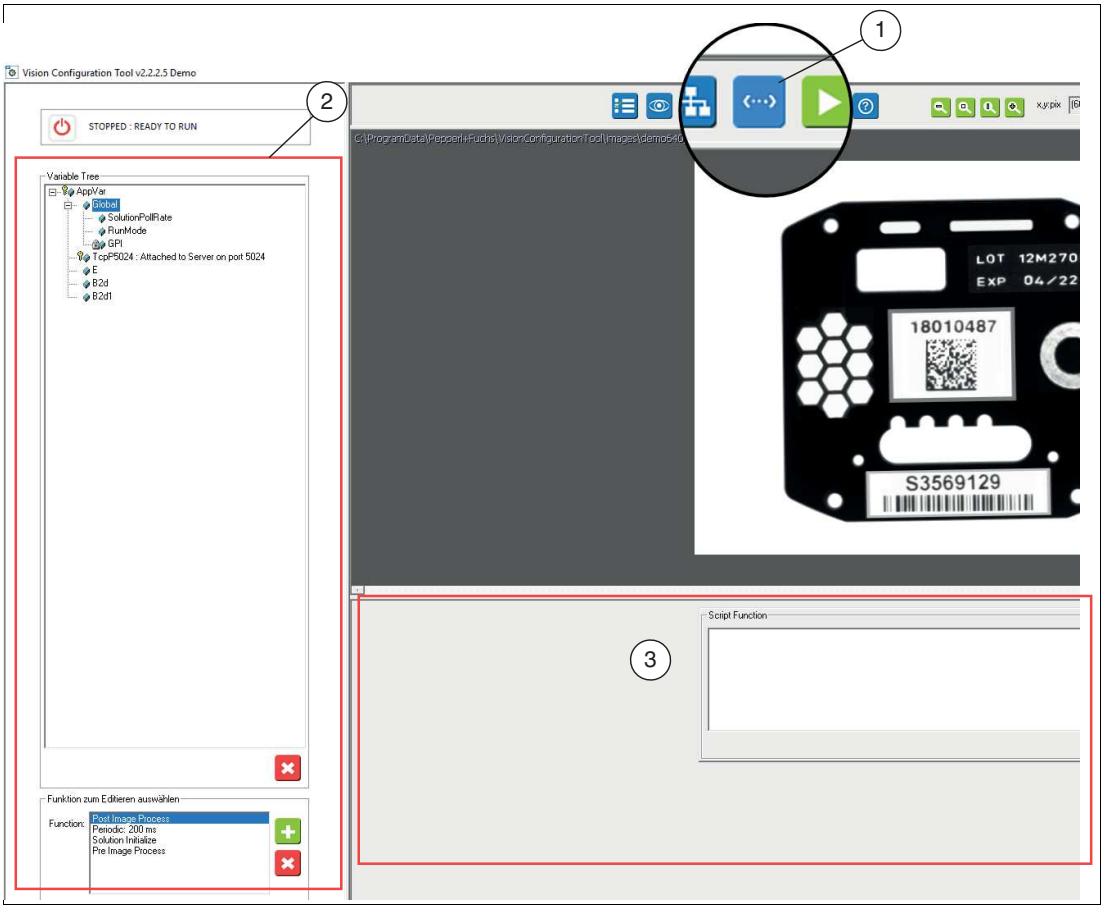


Figure 7.126 Script editor

Opening the Script Editor

Click the "Script" button (1) in the navigation bar.

↳ This opens the Scripts menu.

Note

There are three sections within the script window. The operating panel (2) on the left contains a tree list of variables and a function list. The Script Function manager (3) is located on the right-hand side, below the image window.

7.5.1 Formatting Data Strings

You can use the String Editor to create an output string for the COM interface or TCP/IP interface from the existing variables. The Script Editor automatically generates a corresponding function script.

You have the option to write your own data strings. Use the "Format string" button to open the String Editor window.

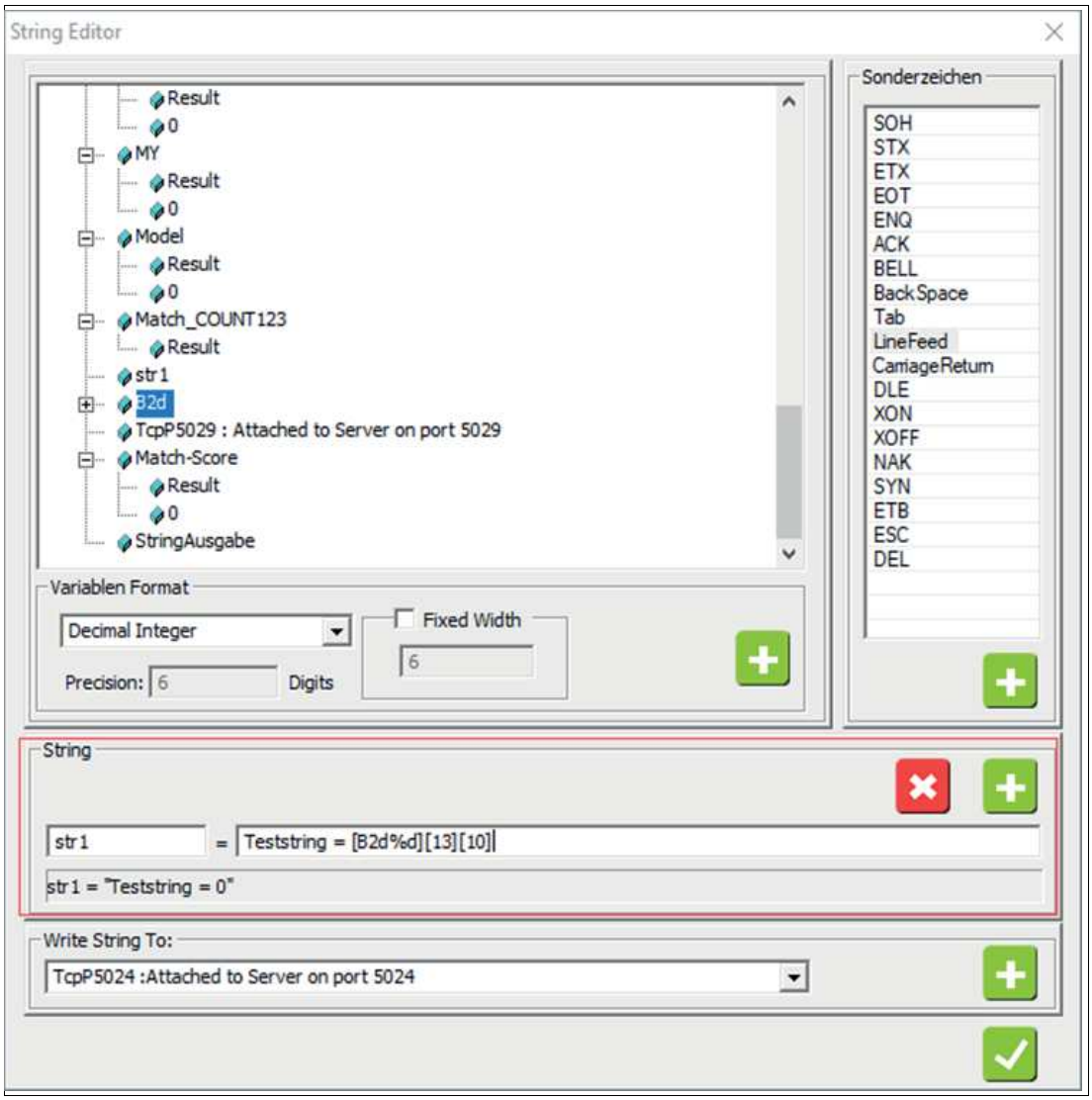


Figure 7.127 String Editor

A variable name for the output string "str1" is defined in the String field.

Fixed ASCII characters are entered in the output string and certain result variables of the camera in square brackets [B2d%d], each with a defined data format (e.g., variable name%d = decimal integer), and control characters such as Carriage Return [13] and Line Feed [10].

Use the plus symbol to assign the string to a previously defined output port e.g., TCP5024. The corresponding string is created directly in the Script Editor as code and can be used.

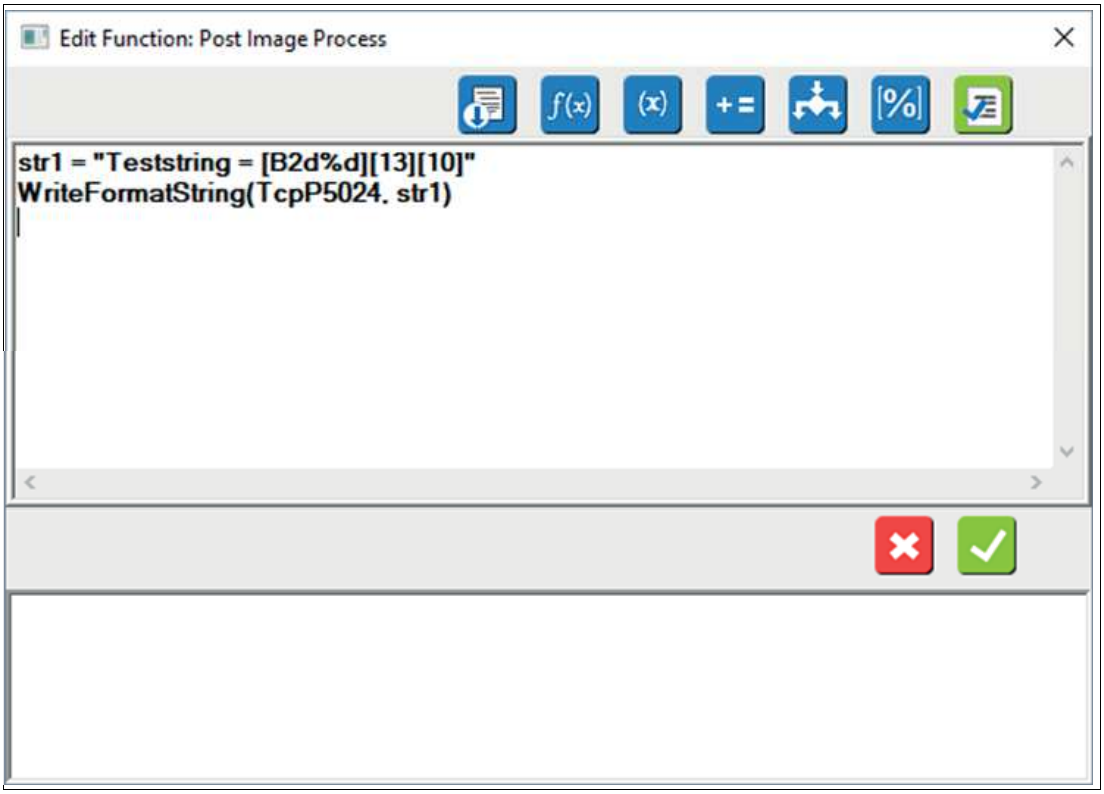


Figure 7.128 Edit Function

7.5.2 Variables

The Variable Tree contains all available variables of the device. This includes the available global result variables (e.g., PassCount), variables of the previously defined vision tools (Boolean, string, integer, etc.) and the previously set interfaces.

There are essentially three types of variables that can be used in a script:

1. **Global device variables:** variables connected to external hardware, such as I/O.
2. **Global control variables:** variables that enable system-wide status or control.
3. **Local user variables:** variables associated with user-defined tools or functions.

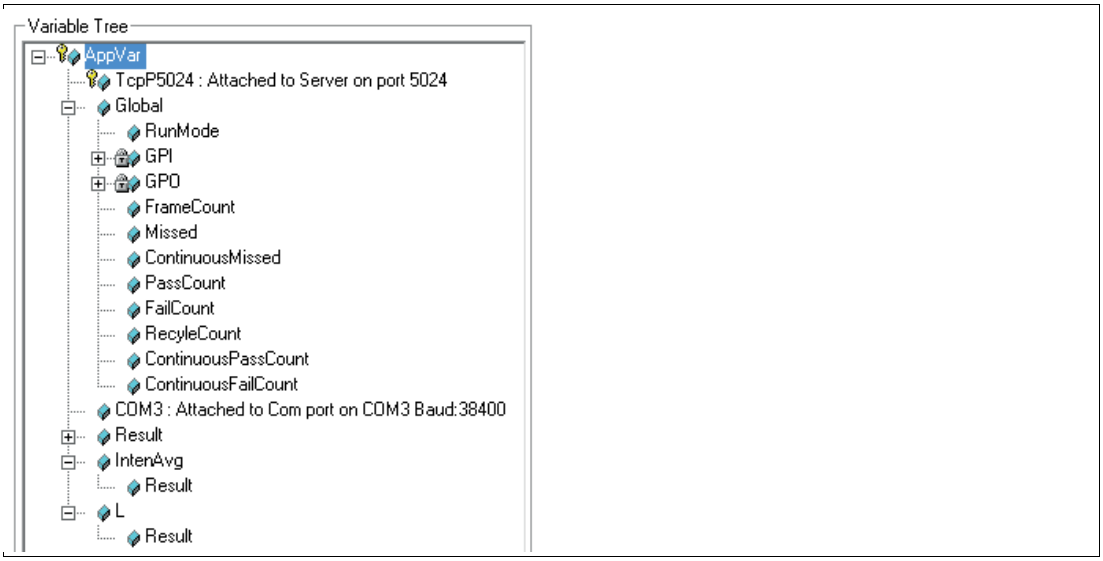


Figure 7.129 Variable Tree

Variable Basics and Nomenclature

Use square brackets for variable names, especially names with spaces. Variable names inserted or dragged into a field are enclosed in parentheses. All of your equations and variables are saved in the job file and are retained when the job file is reloaded. Variables added by the user belong to the current job. When you load another job, your user variable set is replaced by the set belonging to this new job. There are many predefined variables with special functions for use in scripts. You can also create your own variables. When you reference a variable, this variable is automatically created or instantiated. A separate creation or declaration step is not required. The following table lists predefined variables.

Variable Definitions

Variable	Description
Result.0	This variable is read-only. This is the value of "Result" before it is output. This allows results to be evaluated by equations before the output is sent to the monitoring unit, the decision I/O, and other communication systems. Result.0 returns three values: 1 = "Pass", 2 = "Recycle", 3 = "Fail."
Result	This variable is read-only. The variable returns the result of all measurements (and all devices, if more than one is connected). This result is sent to the monitoring unit, the decision I/O, and other communication systems (e.g., PLC, Ethernet, serial interface, etc.) The result returns three values: 1 = "Pass", 2 = "Recycle", 3 = "Fail." Each measurement has one result (L1.result).
Global.GPI[#]	This is an input for general applications. The device treats and evaluates all inputs like a stationary logic input.
Global.GPO[#]	This is a universal output. By default, the outputs are kept at "high" or "low" until the next result is available. (The pass/recycle/fail decision outputs are pulsed.) You can use the pulse function or the "delayed event" function to create a pulse output.
Global.RunMode	This is the current operating state or operating mode. 0 = running, 1 = stopped.
Global.FrameCount	This is the number of images that have been captured since job files or programs named "Solutions" were loaded, or since the statistics were reset ("Reset Statistics" button on the user interface).
Global.Missed	Number of failed captures. These are also called skipped captures.
Global.ContinuousMissed	Number of successively failed captures.
Global.PassCount	Number of successfully captured objects.
Global.FailCount	Number of failed objects.
Global.RecycleCount	This is the value of the "Recycle" counter or the number of recycled objects.
Global.ContinuousPassCount	The number of objects that were captured in a series or in succession.
Global.ContinuousFailCount	The number of captures that failed in a series or in succession.
Global.SolutionPollRate	If this variable is set to a non-zero value, the Job Change and Job Select inputs are queried at a rate equal to the value of Global.SolutionPollRate in Hertz. For example, if Global.SolutionPollRate = 5, the inputs are queried with 5 z or 200 ms.

Special Global Variables

Some special variables are not visible in the variable tree or in the free editor. These variables are as follows:

Variable	Description
ShowPrepro-cessed	Enables (=1) or disables (=0) the display of preprocessing in the search field at runtime.
RelearnIndex	Uses one of the general input values to trigger the customization or relearning of certain tools (Match, Locator). RelearnIndex=1 performs relearning (in the next captured image) when GPI[1] is enabled. This instruction must be added in the "Solution Initialize" function (job initialization). Create a new relearn variable to specify a tool to be relearned. Example: To assign the MS1 Match Tool to be relearned, MS1.Relearn=1. The MS1 Match Tool is taught in again (in the following image) when the input assigned by RelearnIndex GPI[1] is confirmed. You must connect the selected input such that it only reads the value 1 if relearning is required or if it uses the RelearnOnZero variable.
RelearnOnZero	RelearnOnZero =1 forces relearn if the defined relearn input (RelearnIndex) is 0 and not 1. This instruction must be added in the "Solution Initialize" function.
Prog	Prog (= permanent variables) is the bracket for variables to be retained through job changeovers. Use "Prog" to create variables that retain their current value and are not deleted when a job file is loaded. Permanent variables are deleted when the device is switched off. Permanent variables are defined with a prog prefix and can be stored in a job file (i.e., Prog.myvariable).

Note

Deleting variables

Select a variable in the Variable Tree and click the "Delete" button to remove it. You cannot restore a variable once deleted. This affects the output signals, the decision table, and the operating panel. Deleting a measurement tool variable (e.g., L1) does not delete the tool, but the measurement is no longer available for scripting.

7.5.3 Function Blocks

Function blocks consist of functions, equations, or instructions that affect an output or result. Most function blocks can be shared or retrieved by other function blocks (such as subprograms). Some function blocks are executed in sequence (higher-level function blocks), whereas others are based on a user-defined event, such as a time interval or a transition to a global input. Higher-level function blocks are a special class of function blocks that are executed in a predefined sequence. They can retrieve other function blocks, but cannot be retrieved by other function blocks.

Scripting instructions must be created for communication with a PLC.

Scripting instructions allow you to define the result output and the data format. These can be calculations based on measurement results, displayed messages for the user, or variables that are read from a PLC or written to a PLC.

The function menu lists the four most commonly used functions to retrieve scripts:

Predefined Function Blocks

Function	Description
Post Image Process	This function block is executed after the tools are applied to the image and before the results are sent.
Periodic 200 ms	This function block runs or is repeated every 200 milliseconds. This is useful for checking inputs, status, or PLC communication.
Solution Initialize	This function block runs once when the job file is loaded (from the memory) before it starts.
Pre Image Process	This function block runs after an image has been captured and before tools are applied.

Creating or Editing Scripting Instructions

1. Select one of the four function blocks (1) from the list or click the "+" new function button (2) to add another function.

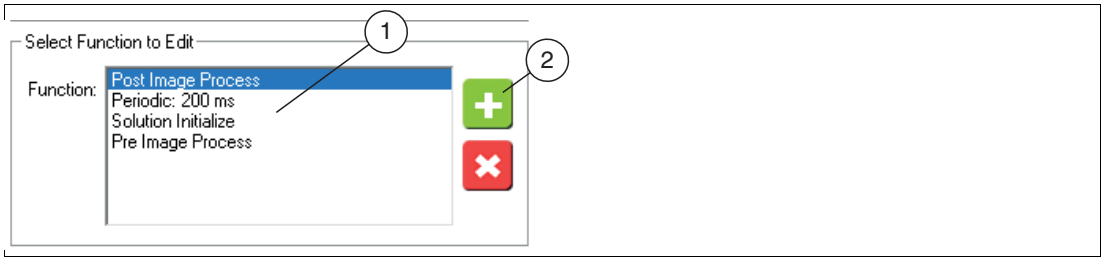


Figure 7.130 Selecting function blocks

↳ If the function block already contains instructions, they appear in the "Script Function" field.

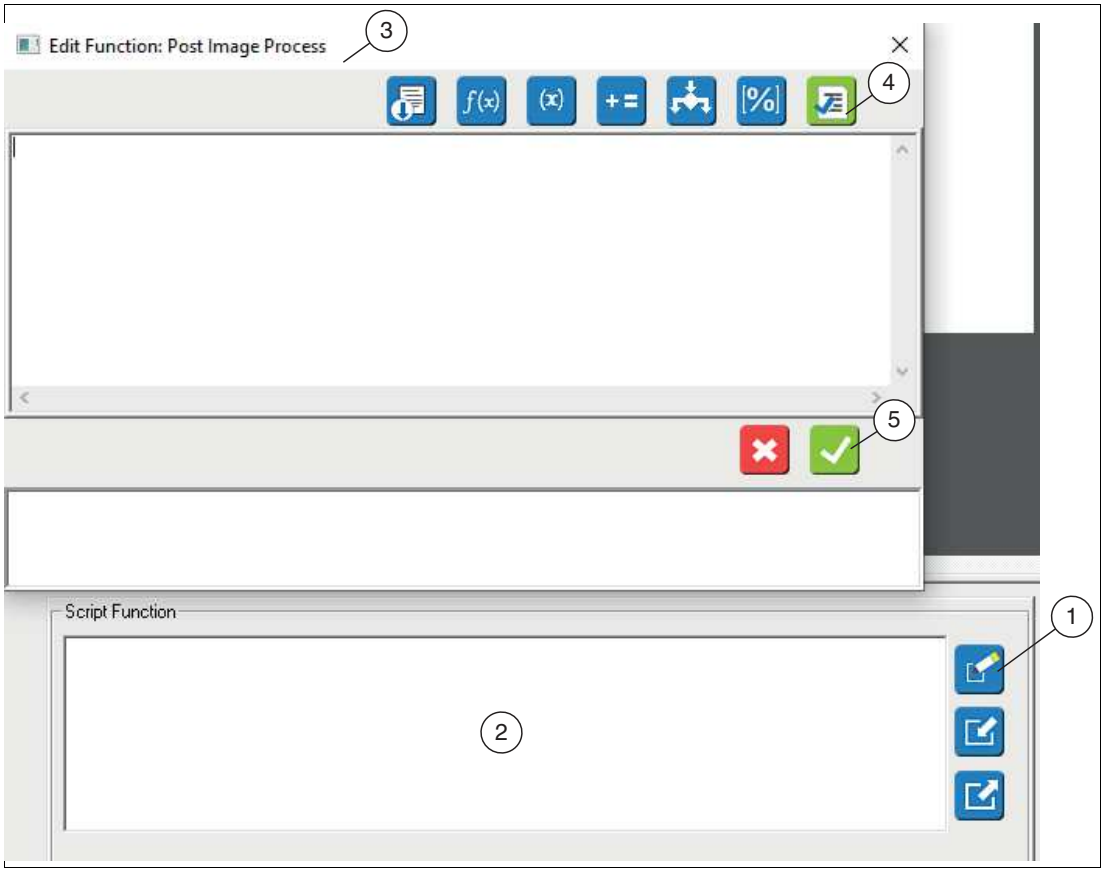


Figure 7.131 Script function

2. Click the "Free Edit" button (1) or in the text box (2).

↳ This opens the "Edit Function:" input window (3).

3. Enter your instruction or function in the input window.

Note

- Instructions are made in algebraic form, for example: `a = b+c`
- Function blocks are retrieved in the form:
  `z = aFunctionName(param1, param2)`

4. Click the "Check Syntax" button (4) to check your inputs. Runtime errors associated with the application of the job are not checked.
5. Click "OK" (5) to confirm your entry.

Function Elements in the Script Editor

Function	Description
	Insert a template script. The selection options are trigger or switching solutions from a PLC variable. The variables that you have created are listed. You can create new variables by selecting a PLC type.
	Adds a function. Opens a menu that lists functions by category.
	Adds a variable. Opens a menu that lists variables by category.
	Adds an operator. Opens a menu with a list of operators.
	Opens a menu of program flow elements.
	Opens the data string editor window.
	Syntax check. Messages appear at the bottom of the window range. You can double-click a message to highlight the code at the top of the window range.

Adding a New Function Block

The four most commonly used function blocks are listed by default. They do not require any additional configuration. The following optional function blocks can be added later.

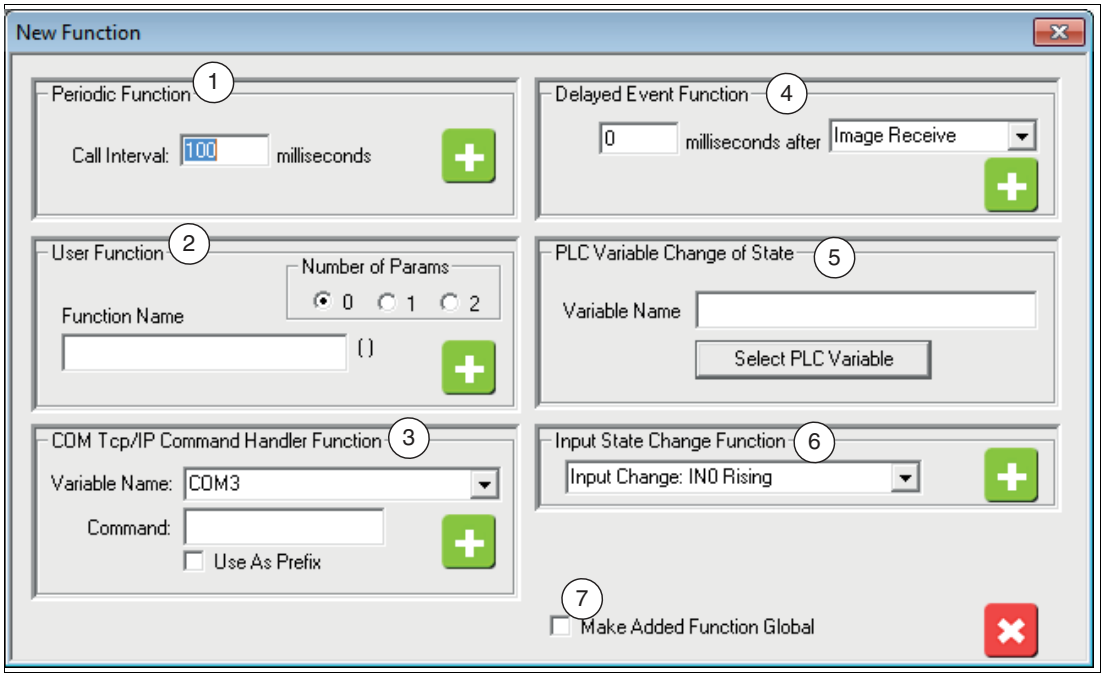


Figure 7.132 Adding new function blocks

Item	Function	Description
1	Periodic	A periodic function block is retrieved at a predefined fixed time interval.
2	User	A user-defined function block can be retrieved by other script functions. You can define the number of transferred parameters: one, two, or none. You cannot rename the parameters. Note: Each user-defined function block must contain the return instruction ("return()" instruction).
3	COM Tcp/IP Command Handler Function	This function block is retrieved when a command is received on a COM port or a TCP/IP connection.
4	Delayed Event	A delayed event function block is called with a delay to a specific event. There are three possible events to choose from. Immediately after image capture (Image Receive), immediately after calculating the capture results (Image Process), or based on a software trigger (Software Trigger).
5	PLC Variable Change of State	This function block is retrieved when a PLC variable or register changes the value or state. Note: This applies only to EtherNet/IP and/or if the device is a MODBUS slave (not a MODBUS master).

Item	Function	Description
6	Input State Change	A hardware trigger function is retrieved based on a change in an input signal. Applications are: trigger image capture via inputs, job changeover.
7	Make Added Function Global	If this check box is selected when you create a new function block, it is made available to all new applications that you create. Edits are shared across all applications. You can delete a "global" function block only in the application in which it was originally created. Note: This does not apply retroactively to existing function blocks or existing applications.

7.5.4 Script Functions

Camera Functions

Function	Description
AutoSaveEnable(enable)	Switches automatic job saving on or off. When autosave is switched on, the job is automatically saved when a user exits the top-level (main level) settings window. The job changeover is automatically enabled. Enable = 0 — turn off autosave. 1 — enable autosave.
ChangeSolution(requestedSolutionID)	Loads a job file. If the "requestedSolutionID" does not exist, the current job file continues to run without fault detection.
ChangeStartupSolution(startupSolutionID)	Changes the job number that is automatically loaded at startup or restart. This change is retained during a system restart until it is changed again. If there is no "startupSolutionID", the system starts with job 00 if this exists.
Copy(source, dest, numElements)	Copies numElements from source (an element field) to dest (an element field). The copy function can be used to update multiple PLC registers in a single transaction.
Delay(milliseconds)	Generates a delay or wait time in a script. This is more efficient than a loop over a timestamp. Adding a delay increases the execution time. Do not add a delay that is greater than your average capture time or the time between triggers. Use Delay() only in a periodic function; it is ignored in all other function groups.
FormatTime(timeVal)	Converts a time value in milliseconds since January 1, 1601 to a character string representing the current date and time. Example: time1 = GetTime()
GetColor(colorID, pixelValue)	Returns the specified primary color value. colorID — indicates the desired primary color: 0 = blue, 1 = green, 2 = red. pixelValue — value for a color pixel, e.g., a value returned by GetPixel. Example: centerPix= GetPixel(0, 320, 240) centerRed= GetColor(2, centerPix)
GetInspectTime()	Returns the inspection time of the previous execution. When it is retrieved in the "Post Image Process" function, this is the current execution.
GetMaxInspectTime()	Returns the maximum inspection time since the last reset.
GetMinInspectTime()	Returns the minimum inspection time since the last reset.
GetPixel(camID, x, y)	Returns the value for the pixel specified by camID, x, and y. x: the x coordinate. 0 = the left-most column. y: the y coordinate. 0 = top line. Example: centerPix= GetPixel(0, 320, 240)
GetSolutionID()	Returns the current job ID number.

Function	Description
GetTime()	Returns a value representing the current date and current time. The value returned is the number of milliseconds since January 1, 1601 (in the local time zone). See "FormatTime" function. Note: "GetTime" is not accurate enough to accurately calculate your cycle time to milliseconds. Use the "TimeMillisec" function for these calculations.
GetTimeString()	Returns a data string value representing the current date and current time (local time zone). Example: "now = GetTimeString()" sets "now" to a string value of "9/7/2009 16:25:28:429." dateString= FormatTime(time1) sets "dateString" to a string value of "9/7/2009 15:25:28:429."
GetUserName()	Returns the user name for the currently logged in user.
GetVersion()	Outputs the firmware version of the sensor (the Vision Configuration Tool version).
ImageProcess(DllFileName, functionName, numVars, VarArray)	Retrieve a "functionName" function exported from the "DLL DllFileName." numVars — the number of elements in the "VarArray" field. VarArray — parameter field with a size of up to 20 elements. Parameters can be used as In and/or Out.
Print(string, endOfFile)	Sends a character string of data to the default printer. Does not evaluate embedded variables. endOfFile - 0 = data string is stored temporarily until a subsequent print retrieval is made with "endOfFile" = 1. 1 = All saved text from previous retrievals to print and the transferred character string are sent immediately to the printer. Note: Use this function carefully because every retrieval with "endOfFile=1" causes at least one page to be printed.
SetDisplayStatus(statusMsg, color)	Specifies the message that appears in the "Inspection Status" field (in the configuration and status window) linked to the monitor window. This overrides the display of "Pass" or "Fail." statusMsg — the character string that is displayed. For multiple lines, add the \n character to display a new line. The body of the message is automatically large enough to be as large as possible and still appear in the "Inspection Status" field. String formatting information for variables of the [Var%FormatData] form is supported. msg1="[L1%0.2f]" means that the value of L1 is displayed with two digits to the right of the decimal point. Color — the string name of the color in which the message text is displayed. Possible values are: "black", "red", "green", "yellow", "blue", "magenta", "cyan", "white", "dark red", "dark green", "dark yellow", "dark blue", "dark magenta", "dark cyan", "light gray 1", "yellow green", "sky blue", "cream", "light gray 2", "medium gray."

2021-03

Function	Description
StartInspect( )	Starts the image inspection.
SetAppButton(buttonNumber, buttonName)	Creates a user-defined button in the Run or Monitor operating panel. Up to four buttons can be added. A "User" function group appears with an "X button clicked" for each defined button. encodeMethod • 0 = without compression. • 1 = JPEG compression (default for color). • 2 = proprietary high-speed compression (default for mono).
SetImageEncode(encodeMethod)	Changes the image encoding used to compress images sent to a connected end device.
StartInspect( )	Starts image inspection.
StopInspect( )	Stops image inspection. Note: Use this function in a periodic function. The function must not be used in image post-processing.
SwitchingIsEnabled( )	Returns 1 if job switching is enabled. Returns 0 if job switching is not enabled.
TimeMillisec( )	Returns the current time in milliseconds.
return(FunctionReturnValue)	Returns the specified value "FunctionReturn-Value" from a user-defined function. Example: Return((p1 + p2) /2).

Statistical and Vision-Tools-Related Functions

Function	Description
GetMean(measurementVar)	Returns the arithmetic mean for the specified measurement. Example: L1Mean = GetMean(L1)
GetStdDev(measurementVar)	Returns the standard deviation for the specified measurement. Example: L1StdDev = GetStdDev(L1)
GetMin(measurementVar)	Returns the minimum value that occurred for the specified measurement. Example: L1Min = GetMin(L1)
GetMax(measurementVar)	Returns the maximum value that occurred for the specified measurement. Example: L1Max = GetMax(L1)
ResetVarStats(measurementVar)	Resets the measurement statistics (min, max, mean, std dev) for the specified measurement. All earlier data samples for the measurement will be deleted for statistical calculations. Example: ResetVarStats(L1)
GetToolType(measurementVar)	Returns a number indicating the type of measurement tool for the specified variable. Example: L1type = GetToolType(L1) variable L1type returns the value 6.
GetNthToolType(varIndex, CamID)	Returns a number indicating the type of measurement tool for the specified variable index and camera ID. Returns zero if no tool exists for "varIndex" and "camID." varIndex — 0 to (number of tools minus 1) that exist for the specified camera.

Function	Description
GetToolName(varIndex, camID)	Returns the simple name (no camera prefix) of the measurement tool, for the specified variable index and camera ID. varIndex — 0 to (number of tools minus 1) that exist for the specified camera.
RequestRelearn(measurementVar)	Causes the measurementVar to be re-taught in on the next image. Example: RequestRelearn(Bar)
SetTolerances(measurementVar, toleranceArrayIn)	Specifies the five tolerance "pivot points" for the specified measurement variable. measurementVar — variable name. toleranceArrayIn — five-element field of tolerance values. (as shown below, under GetTolerances).
GetTolerances(measurementVar, toleranceArrayOut)	Obtains the five tolerance "pivot points" for the specified measurement variable. measurementVar — variable name. toleranceArrayIn — five-element field of tolerance values. (as shown below, under GetTolerances). • Index 0: minimum recycling value or minimum value for "Pass" • Index 1: minimum value for "Pass" or 0 • Index 2: perfect value • Index 3: maximum value for "Pass" or 0 • Index 4: maximum recycling value or maximum value for "Pass"
SetMatchString(measurementVar, perfectMatch)	Sets a "measurementVar" data string a new "PerfectMatch" value. Used only for measurements that return a data string (barcode, 2-D code, OCR) to determine a new "perfect" value.
SetNthTolerances(varIndex, camID, toleranceArrayIn)	Sets the five tolerance "pivot points" for the specified variable index and the camera ID. varIndex — 0 to (number of tools minus 1) that exist for the specified camera. toleranceArrayIn — five-element field of tolerance values (as shown above in GetTolerances).
GetNthTolerances(varIndex, camID, toleranceArrayIn)	Sets the five tolerance "pivot points" for the specified variable index and the camera ID. varIndex — 0 to (number of tools minus 1) that exist for the specified camera. toleranceArrayIn — five-element field of tolerance values (as shown above in GetTolerances).
GetToolValue(toolName)	Outputs the measured value for the tool. The tool name ("L1") is transferred. Example: value = GetToolValue("L1")
GetToolResult(toolName)	Returns the result value for the tool. The tool name ("L1") is transferred. Example: value = GetToolResult("L1")
GetVarDimension(varName)	Returns the number of child variables of the "varName" variable.

2021-03

Function	Description
SetParam(numArgs, argList)	Changes the position of an existing region (ROI) using a list of data arguments. • numArgs — the number of data arguments in "argList" • argList — a list of values for setting the ROI parameters.
SetToolText(toolName, toolText)	Sets or changes the "toolText" that appears in the "toolName" Graphics Tool (Express only).
SetToolPenColor(toolName, red, green, blue)	Sets or changes the outline of the "toolName" Graphics Tool to red, green, or blue. Express only.
SetToolFill(toolName, red, green, blue)	Set or change the fill color of the "toolName" Graphics Tool to red, green, or blue. Express only.
WriteVar(narName, value)	Writes a value to a script variable. • varName — the name of the variable to be written to. • Value — the value to be written to the variable. Example: WriteVar("InputThreshold", 4.5)
Sort(keyVarName, ascend, followVarNames)	Sorts the field specified by "keyVarNam" in ascending order. • ascending - 0 = sort in descending order. 1 = sort in ascending order. • followVarNames — an optional list of the names of field variables that are sorted in the same reassignment as "keyVarName." Example: If the index value 0 is moved from keyVarName to index 5, the same reassignment occurs for all "followVarNames" fields.
ReadVar(varName)	Reads the value of a script variable. varName — the name of a script variable.
DeleteVar(varName)	Delete the "varName" variable. This is useful for removing a variable that is left over from a previous script or function.
GetNumElements(varName)	Returns the actual dimension or number of elements for a "varName" measured variable that generates a field of values.
EnableFormat(varName, enable)	Enables or disables the output of the "varName" measured value in the output log or script output. enable - 0 = does not output "varName." 1 = allows output of "varName."

Mathematical Functions

Function	Description
sin(radians)	The result is the sine of the angular dimension. The argument must be specified in the angular dimension.
cos(radians)	The result is the cosine of the angular dimension. The argument must be specified in the angular dimension.
tan(radians)	The result is the tangent of the angular dimension. The argument must be specified in the angular dimension.

Function	Description
asin(x)	The result is the arc sine of x in the range -p/2 to p/2 angular dimension, where -1 <= x <= 1.
acos(x)	The result is the arc cosine of x in the range -p/2 to p/2 angular dimension, where -1 <= x <= 1.
atan(x)	The result is the arc tangent of x in the range -p/2 to p/2 angular dimension.
atan2(y, x)	The result is the arc tangent of y/x in the range -p to p angular dimension.
exp(x)	The result is the exponential value of x.
log(x)	The result is the natural logarithm of x.
sqrt(x)	The result is the square root of x.
pow(x, y)	The result is set x to the power of y.

Data String Functions and Character Functions

Function	Description
Find(substring, inString)	Finds the first occurrence of a substring in the inString input and returns the zero-based index digit of the first matching character. Returns -1 if no match was found. Spaces are counted. Example: idx = find("00", "SM WRA 0057 4321") returns 7, or sets idx= 7.
Substring(string, startIndex, length)	Forms a substring from the input string, starting with the start index (zero-based) of length characters. If length = 0, all characters up to the end of the data string are included in the substring. Example: s2 = substring("SM WRA 0057 4321", 9, 0) returns string "57 4321" in s2.
StrLen(string)	Outputs the number of characters in a character string.
GetChar(string, index)	Outputs the character that is in the data string at index (zero-based).
SetChar(string, index, char)	Sets the character in string, which is at the index (zero-based), to "char."
int(string)	Converts the entered character string (of numbers) to an integer value. Example: x = int("33") sets x = 33
float(string)	Converts the entered character string (of numbers) to a floating-point value. Example: x = float("57,499") sets x = 57,499
char(int)	Converts the entered integer to a character.

Function	Description
string(int)	Converts the input number to a character string, e.g., an integer int(base 10), or a floating-point number. A floating-point number uses the same formatting as "WriteFormatString." Example: LengthStr = string("[L%0.3f]")
FormatString(stringForm)	Outputs a formatted character string by using the format in the "stringForm" argument. "stringForm" can be created using the String editor. Example: The statement str1 = FormatString("[IntenAvg%.4f]") results in a string "55,7015" (returned in str1).

Note

For additional functions, refer to the Vision Configuration Tool online help section.

7.5.5 Script Examples

The following examples show how to use the Script Tool. Each example includes a brief description of the application and associated code excerpts from the corresponding functions. The examples cover basic script concepts that apply to standard applications.

Using Script Command to Switch Switching Outputs

pulse(activeVal, offsetMillisec, durationMillisec) — generates a pulse output.

activeVal - 1 = active high pulse, 0 = active low pulse.

offsetMillisec — Offset or delay in milliseconds from the time this specification is executed.

durationMillisec — Pulse duration in milliseconds.

This function is executed in the **Post Image Processing** function.

GPIO Settings

Purpose/Task	Script	Description

After global result variable (result), output a pulse signal via GPO[0] or GPO[1].	if (Result=1)
	Global.GPO[0] = pulse(1,0,20)
	else
	Global.GPO[1] = pulse(1,0,20)
	endif()
		The equation outputs an active high pulse of 20 ms to GPO [0] (with no delay) if the job result is equal to 1 (Result=1).
		The equation outputs an active high pulse of 20 ms to GPO [1] (with no delay) if the job result fails.

After Match Tool result variable (MS1.Result), output a pulse signal via GPO[1].	If (MS1.Result = 3) Global.GPO[1] =pulse(1,0,400)	The equation outputs a 400 ms active high pulse at the GPO [1] (with no delay) if the MS1 Match Tool fails (result=3).
After Match Tool result variable (MS1.Result), output a pulse signal via GPO[1].	If (MS1.Result != 1) Global.GPO [1] = (pulse(1,0,400)	The equation generates the same pulse if the MS1 Match Tool is not "Pass" (result = 2 or 3).
After Match Tool result: hit result outputting a pulse signal via GPO[1]	If (MS1 < 90) Global.GPO [1] = pulse(1,5,50)	MS1 is the measured value of the match. You can use the measured value in statements instead of the result of a measurement as shown below. Output a 50 ms active high pulse at the GPO [1] (5 ms delay) if the MS1 value of the match is less than 90.
After length measurement result <400, output a pulse signal via GPO[1].	If (L1 < 400) Global.GPO [1] = pulse(1,5,50)	Similarly, if distance measurement L1 is less than 400, a 50 ms active high pulse is output to GPO [1] (5 ms delay).
After Match Tool result variable (MS1.Result), permanently turning a GPO[1] on or off	If(1) Global.GPO [1] = (MS1.Result != 1) Replacing If(1) with "If ( )" produces the same result.	Set the GPO [1] to logical 1 if the MS1 Match Tool does not exist; set the GPO [1] to logical 0 if the MS1 Match Tool does exist. Note: If you set an output to a level, it remains so until you change it. For example, you could set an output to "1" as above and reset it to "0" before you process the next image with the "Pre Image Process" function.

2021-03

FTP File Transfer

There are several ways to transfer images. This example shows how to store images on an FTP server.

WriteImageFile(fileName, camID): Retrieved only from the "Post Image Process" function, transfers the current image from the sensor specified by **camID** to the specified **fileName** (0 for the sensor).

fileName: Full path of the file to be saved.

In this example, no user name or password is required. The name of the image file is "img#.bmp."

Function: **Post Image Process**:

x = x+1

fn = "ftp://ftp.xyz.com/images/img" + x + ".bmp"

WriteImageFile(fn, 0)

Text Customization on the User Interface

You have the option of using scripting to display results or messages in the "Display Status" window on the user interface.

If(Result= 1) // if the global result variable is PASS=1

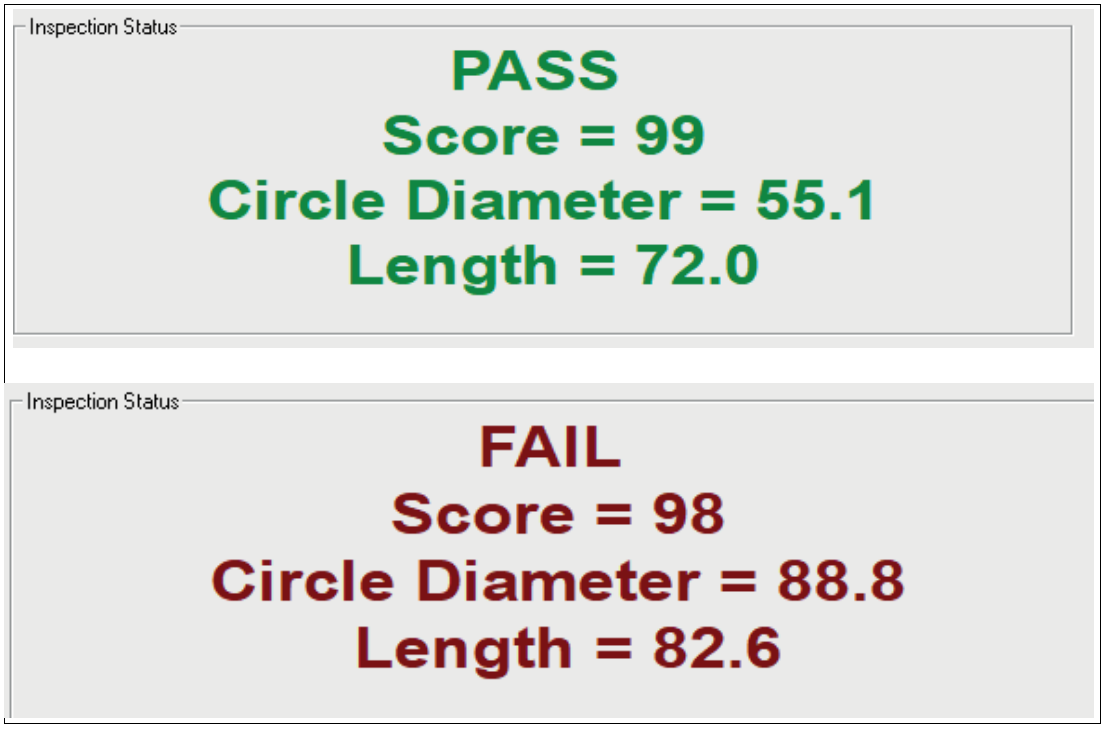
SetDisplayStatus("PASS \n Score = [MS%d] \n Circle Diameter = [CDiam%.1f] \n Length = [L%.1f]", "darkgreen") // Set the user interface to display the tool results in green

else

str1 = "FAIL, Score = [MS%d]. Circle Diameter = [CDiam%.1f], Length = [L%.1f] [13][10]" // Set the user interface to display the tool results in red

SetDisplayStatus("FAIL \n Score = [MS%d] \n Circle Diameter = [CDiam%.1f] \n Length = [L%.1f]", "darkred")

endif



The string name of the color in which the status indicator appears. Possible values are: "black", "red", "green", "yellow", "blue", "magenta", "cyan", "white", "dark red", "dark green", "dark yellow", "dark blue", "dark magenta", "dark cyan", "light gray 1", "yellow green", "sky blue", "cream", "light gray 2", "medium gray."

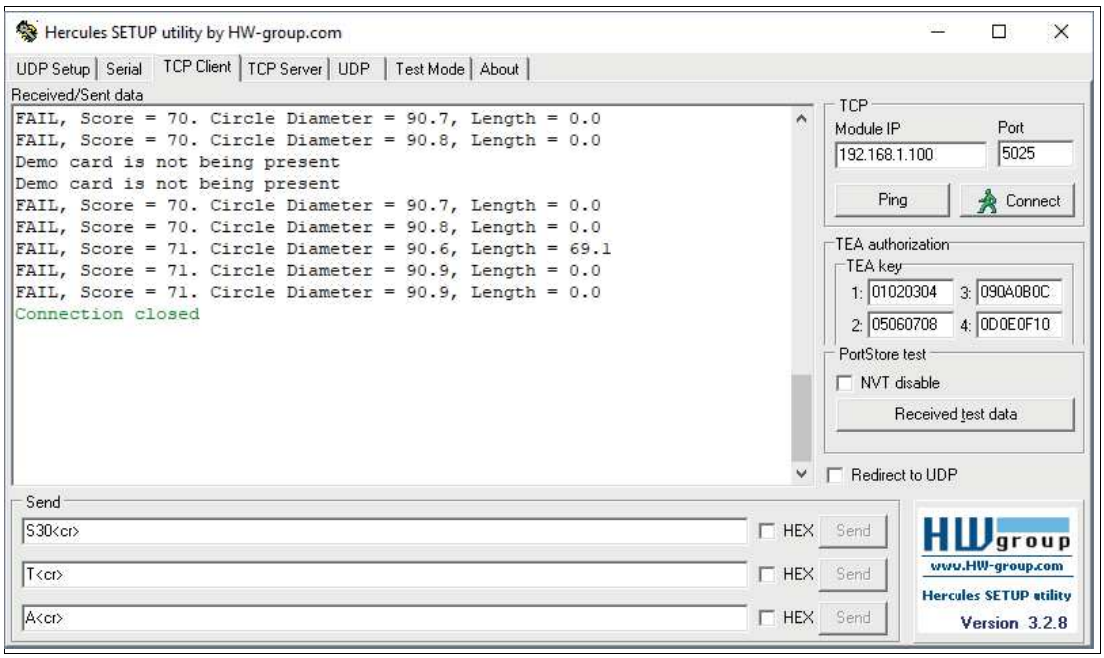
Enabling and Disabling Vision Tools

You can disable the "Factory Settings" in the tool table (see "Tool Status Table" on page 75). The tool is hidden in the image and the results are disabled. All results are displayed with = 0. The tool table disables the tool for all image captures. You can also disable a tool in Scripts. The tool is enabled or disabled on the basis of a variable or input. Use the tool name with the "Disable" switch. Disable=1 hides the tool and disables its results. Disable=0 displays the tool and enables its results. The tool remains disabled if you do not enable it again. If you are disabling a tool on the basis of a variable, we recommend the "Pre Image Process" function. If you want to disable a tool on the basis of an input, you can periodically use the function used to monitor the input.

```
If(GPI[3]=1)
   MS.Disable=1 // Disable and hide Match Tool if input = 1.
else
   MS.Disable=0 // Enable & display Match Tool if input = 0.
endif
```

TCP/IP

TCP/IP: Use of an IF instruction to output a specific data string



if(MS.Result = 1) // If the result of the Match Tool is positive (e.g., object detected)

if(Result = 1) // If the global result variable is PASS = 1 (2 = Recycle, 3 = FAIL)

str1 = "PASS, Score = [MS%d]. Circle Diameter = [CDiam%.1f], Length = [L%.1f] [13][10]" // Passed string

else

str1 = "FAIL, result = [MS%d]. Circle Diameter = [CDiam%.1f], Length = [L%.1f] [13][10]" // Failed string

endif

else

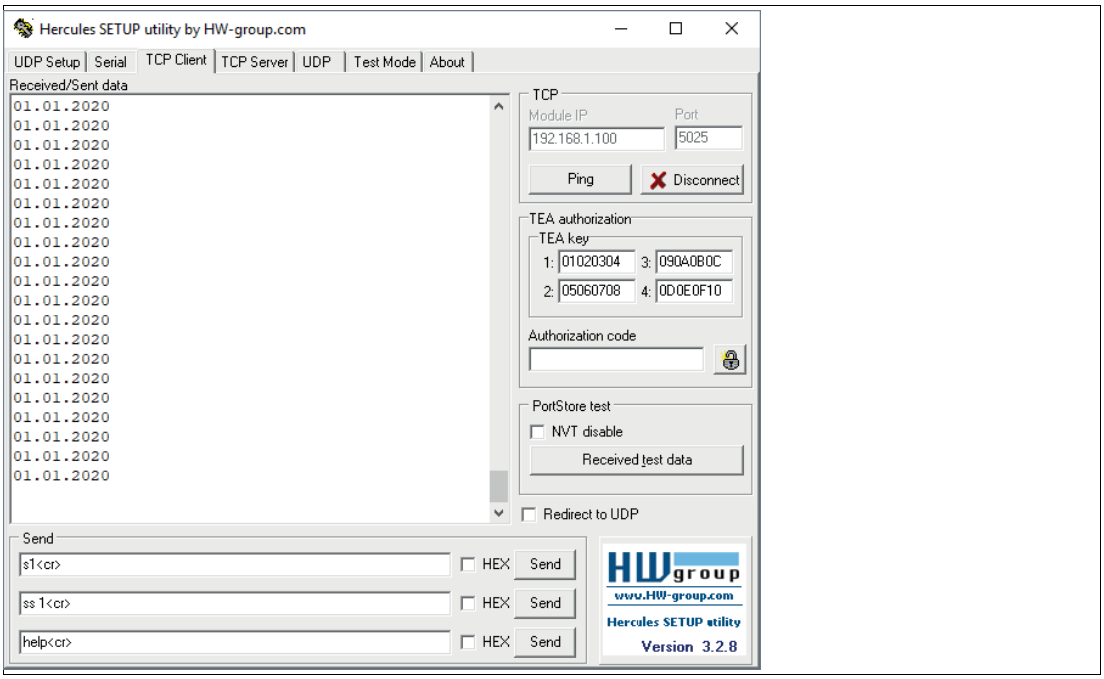
str1 = "Demo card is not being present [13][10]" // Object not detected

endif

WriteFormatString(TcpP5025, str1) // Write str1 variable to TCP port 5025 (port number must be preconfigured)

TCP/IP: Use a substring and form a character string consisting of multiple strings

In this example, separators are added to the date code.



OCR2 =12012020 // The OCR date code is read in without separators

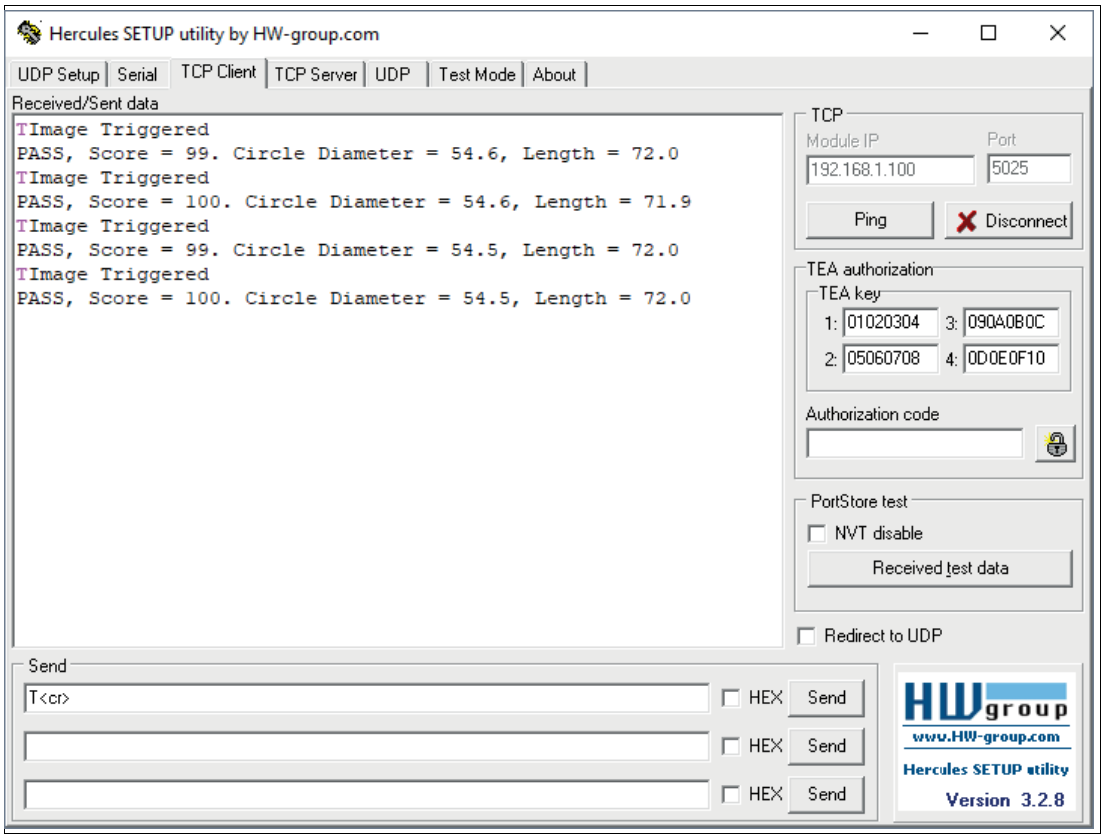
DateCode=substring(OCR2 ,0 ,2) + "." +substring(OCR2 ,2 ,2) + "." +substring(OCR2 ,4 ,0) // The date code is generated with separators

//Substring (data string, start index, length) forms a substring from the input string, starting at the start index (zero-based). If "length = 0", all characters up to the end of the data string are included in the substring.

str1="[DateCode%s][13][10]"

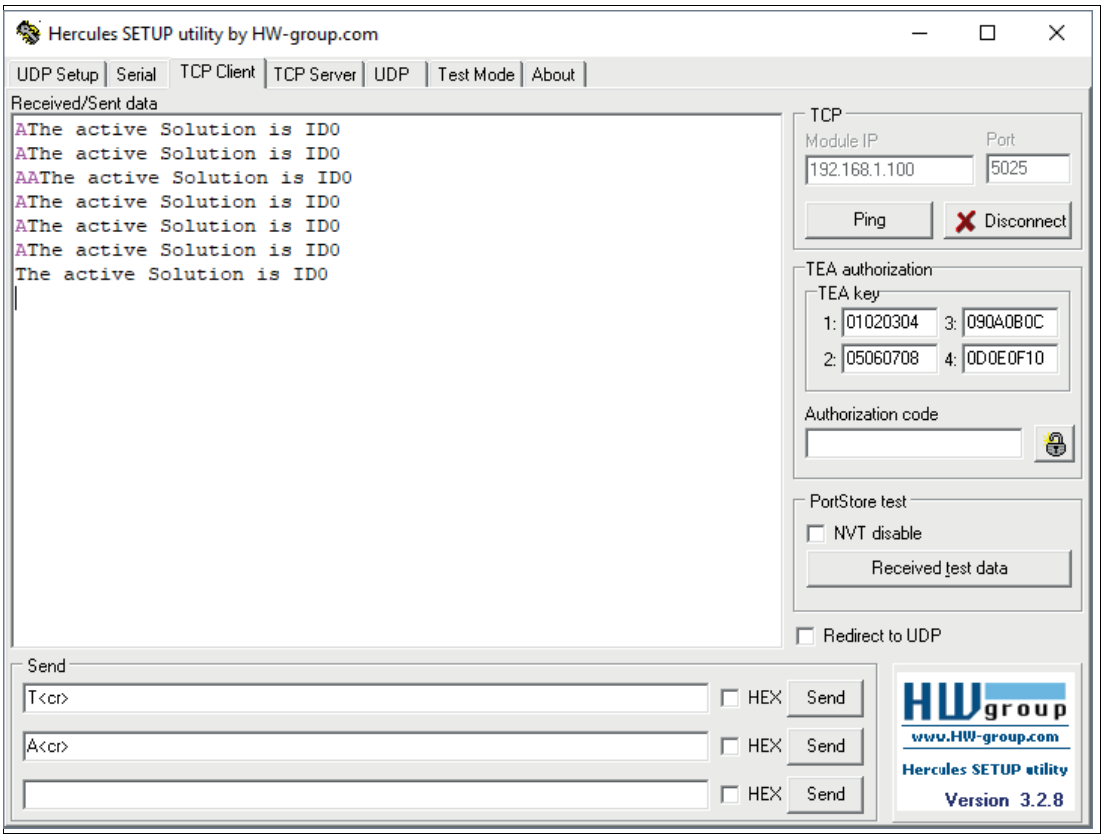
WriteFormatString(TcpP5025, str1)

TCP/IP: Triggering Image Capture—"T" Command




```
ReadBuffer = ReadString( TcpP5025 , 13 ) // 13 (<cr>) is the end character
if(ReadBuffer!= "") // if NOT an empty data string
  CommandString= ReadBuffer
  Counter = Counter + 1
  CommandCharacter =Substring(CommandString, 0, 1) // Trigger command
   if(CommandCharacter= "T") // "T" trigger command
     trigger()
     WriteFormatString (TcpP5025, "Image triggered[10][13]") // Optional feedback for triggers
    endif
endif
```

TCP/IP: Retrieving Job ID—"A" Command

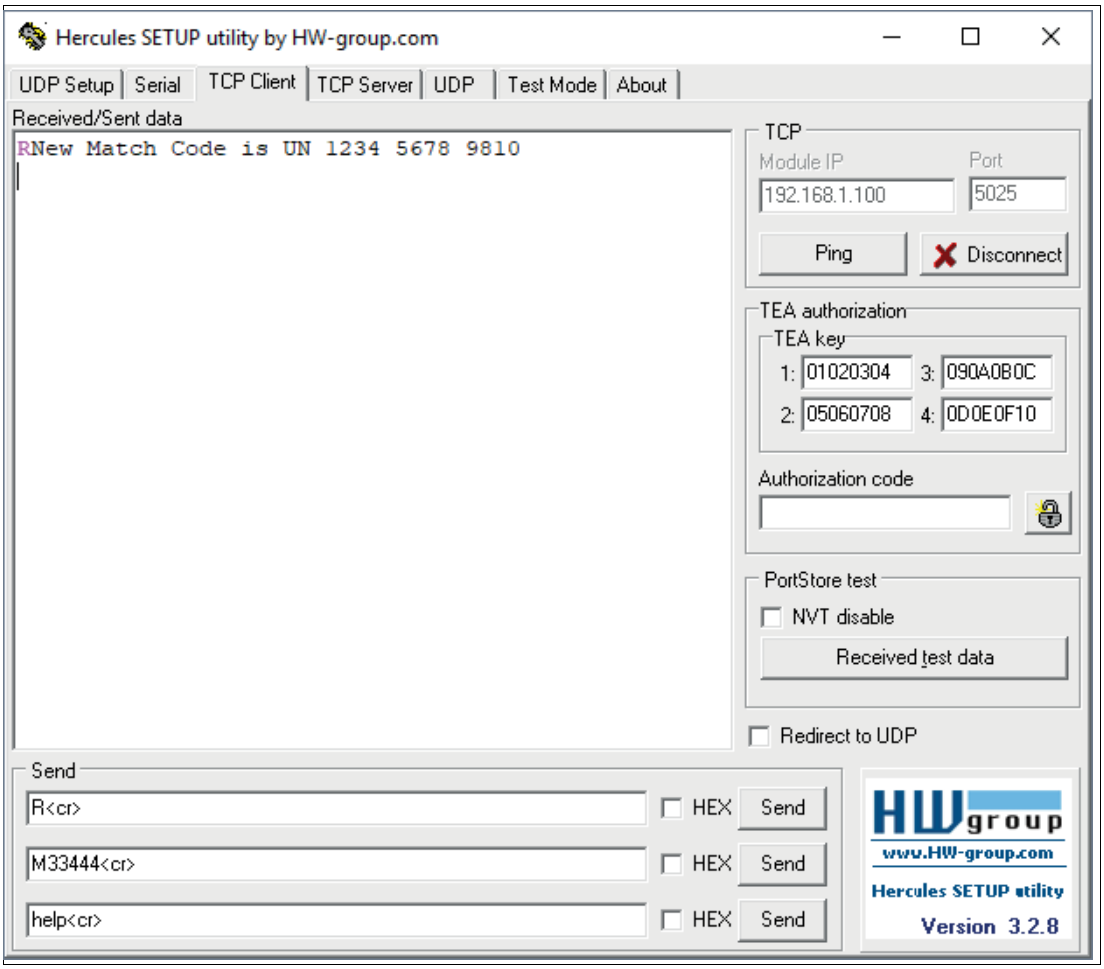



```
ReadBuffer = ReadString( TcpP5025 , 13 ) // 13 (<cr>) is the end character
if(ReadBuffer!= "") // if NOT an empty data string
  CommandString= ReadBuffer
  Counter = Counter + 1
  CommandCharacter =Substring(CommandString, 0, 1) // Trigger command
   if(CommandCharacter= "A") // Letter "A" outputs the active job
    ActiveSolution = GetSolutionID( )
    WriteFormatString(TcpP5025, "The active Solution is ID[ActiveSolution%s][10][13]" )
   endif
endif
```

TCP/IP: Changing Job ID—"S#" Command

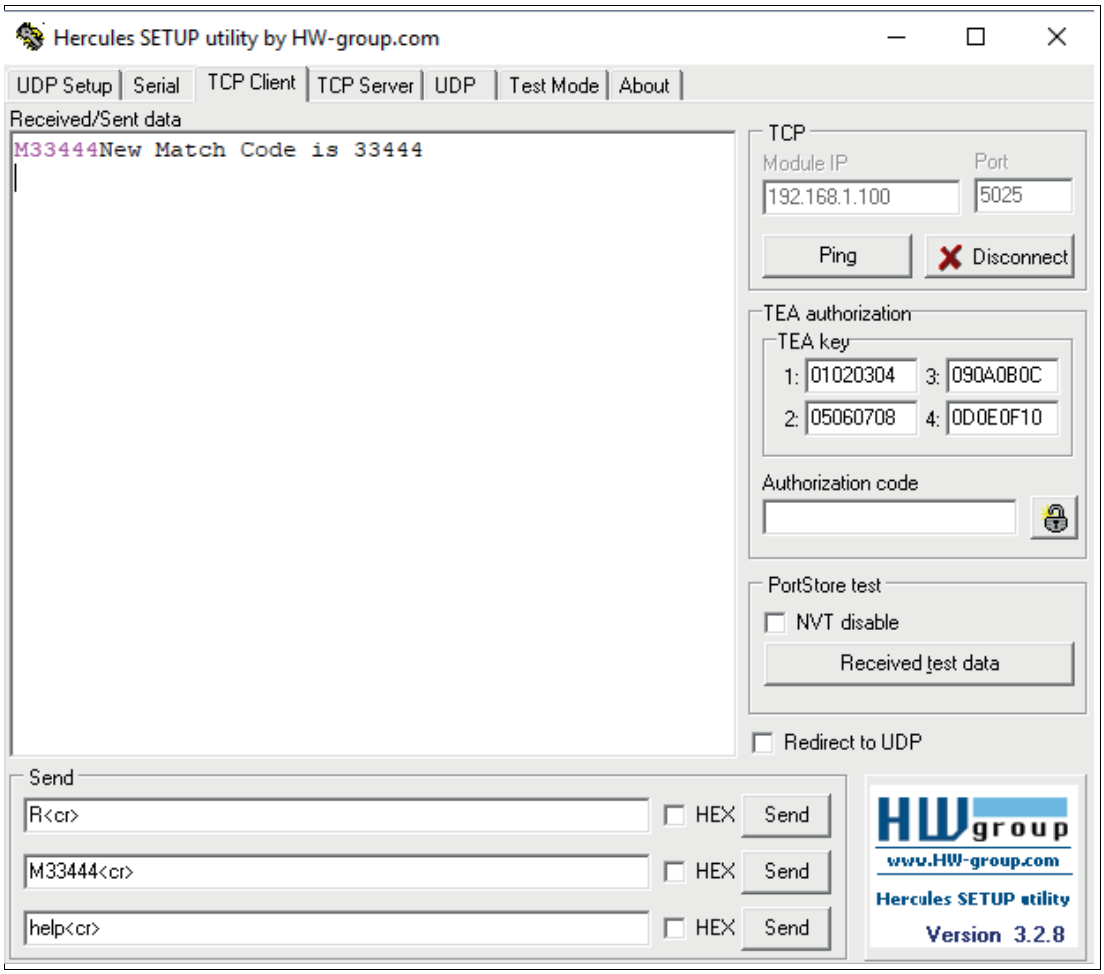
```
ReadBuffer = ReadString( TcpP5025 , 13 ) // 13 (<cr>) is the end character
if(ReadBuffer!= "") // if NOT an empty data string
  CommandString= ReadBuffer
  Counter = Counter + 1
  CommandCharacter =Substring(CommandString, 0, 1) // Trigger command
  JobNumber = Substring(CommandString, 1, 1)
   if(CommandCharacter= "S" AND INT(JobNumber) >0 AND INT(JobNumber)<9)
    JobNumber = INT(JobNumber)
    // Conversion of string to INT
    ChangeSolution(JobNumber)
    // Switch to the specified job
    WriteFormatString( TcpP5025, "Changed to Solution ID[JobNumber%s][10][13]" )
   endif
endif
```

TCP/IP: Re-teaching in Match Code—R<cr> Command




```
ReadBuffer = ReadString( TcpP5025 , 13 ) // 13 (<cr>) is the end character
if(ReadBuffer!= "") // if NOT an empty data string
  CommandString= ReadBuffer
  Counter = Counter + 1
  CommandCharacter =Substring(CommandString, 0, 1) // Trigger command
if(CommandCharacter= "R") // The letter R causes the barcode tool to be taught in again with the variable bar. Only a specific set of tools can be taught in: Match, OCR, Barcode, 2-D Code, and Locator.
  RequestRelearn(Bar)
  WriteFormatString( TcpP5025, "New Match Code is [Bar%s][10][13]")
 endif
endif
```

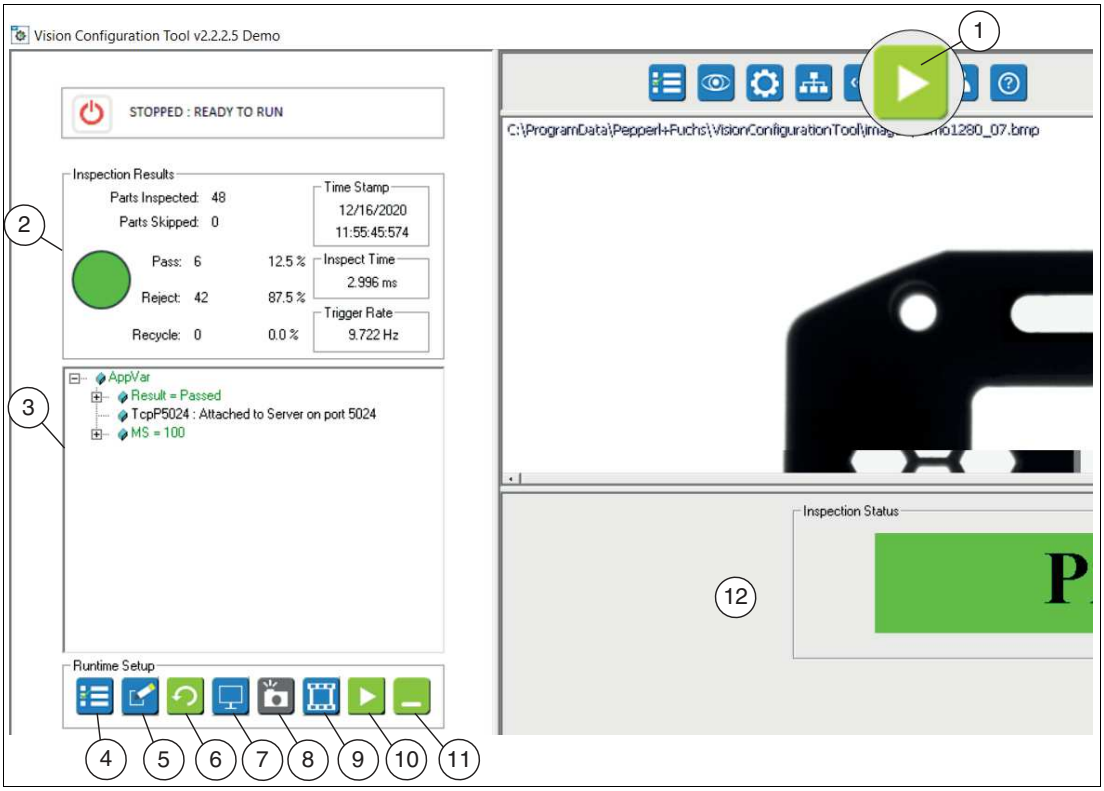
TCP/IP: Match Code Sent from the PLC M##### <cr>




```
ReadBuffer = ReadString( TcpP5025 , 13 ) // 13 (<cr>) is the end character
if(ReadBuffer!= "") // if NOT an empty data string
  CommandString= ReadBuffer
  Counter = Counter + 1
  CommandCharacter =Substring(CommandString, 0, 1) // Trigger command
  MatchCode = Substring(CommandString, 1, 0)
if(CommandCharacter="M" AND MatchCode!= 0 )
  SetMatchString (bar, MatchCode) // SetMatchString(mesurementVar, perfectMatch) sets a measurementVar string to a new perfectMatch value. For measurements that return a string (barcode, 2-D barcode, OCR), a new "perfect" value is set.
 WriteFormatString( TcpP5025, "New Match Code is [Matchcode%s][10][13]")
 endif
endif
```

7.6 Application Test

The "Run solution" button (1) in the navigation bar takes you to the main menu of the "Application Test." Here you can check whether your tools are working as expected.



Item	Menu	Description
1	Run solution	"Run solution" main menu
2	Inspection Results	The inspection results are displayed in this field. • **Parts Inspected**: Total number of parts or images processed. This should be equal to the sum of "Pass, Reject, and Recycle." • **Parts Skipped**: Number of unprocessed parts or images if the trigger or internal timer is too fast to process all images. VOS is internally buffered, so you can take a new image while processing the previous image. Skipped parts are classified as failed. • **Pass, Reject, Recycle**: The (total) counter values for all camera images or for all parts (or since clicking "Reset Statistics") and the counter values for the current or selected camera. In some cases, the counter values of the camera images may be greater than the counter values for all cameras. • **Inspect Time**: The total time spent locating (finding) and measuring (testing or checking) the part.
3	AppVar	Application variables, measurements taken and logged, and other program variables.
4	Select Solution	Use this button to open a saved job. This button changes to "Return to Monitor" when you select a job.
5	Edit Tolerances	This displays a table containing all variables and measurements in the status area. You can change the measurement tolerances or adjust the size and position of the tool search area. This button changes to "Return to Monitor" when you edit the tolerances.

2021-03

Item	Menu	Description
6	Reset Statistics	Resets the "Pass", "Reject", and "Recycled" counters on this display to 0. If your test runs faster than one part per second, you may not see the values go to 0. But you will see that they are changing to smaller numbers.
7	Setup Display	Select what appears in the image area and status window. This button changes to "Return to Monitor" when you configure the display.
8	Manual Trigger	Triggers the camera shot when "Inspection Trigger" is selected in the "Sensor Settings" menu. This button is disabled if "Inspection Trigger" is not enabled.
9	History Recall	History logging for tested parts. This button is not active if you have disabled "History Recall" in the "History Recall Setup" section. This button changes to "Return to Monitor" when you display history logs.
10	Start	Restarts the inspection if "Stop" was clicked.
11	Minimize Maximize	Hides the status field so that only the "Image area" and "Settings" fields are displayed. The settings field appears at the bottom of the screen. A "Maximize" button is added to the setup field. Opens the status field again and returns the setup field to its normal position on the left-hand side.
12	Configuration and status window	Further properties of the selected menus are displayed in the configuration and status window.

Edit Tolerances

You can use the "Edit Tolerances" function in the configuration and status field to adjust the tolerance values (temporarily or permanently) in the table. In addition, you can move or resize the tool's search field in the image area when "Edit Tolerances" is open. Changes affect the values in the Setup Tools windows.

The new value takes effect when you press Enter, or click anywhere in the display outside the edited cell.

There are no "Cancel" or "Undo" buttons. Close or reload the project—without saving—to undo any changes.

The new settings are used while the project is running. You can restore your previous settings by reloading the previously saved project.

The new settings will be lost if you reload the project, load another project, or exit the project without saving.

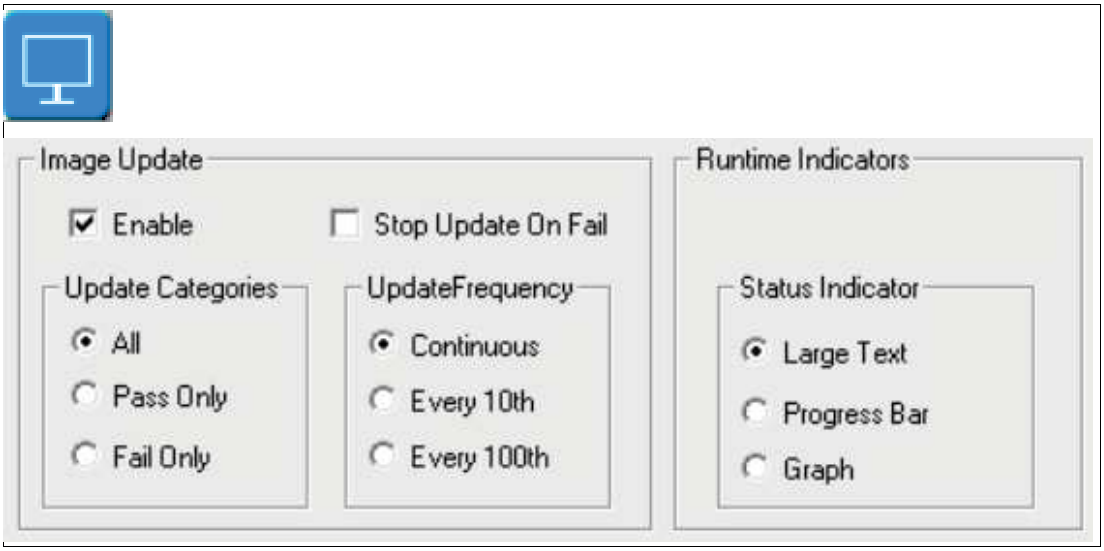
To retain your new settings, save the job files (under a new job number or overwrite the same job number) to keep these settings for reloading later.

Setup Display

You can use the "Setup Display" function to set how often the sensor updates the display on the PC during the application test.

Note

The display options are grayed out when image logging is enabled.

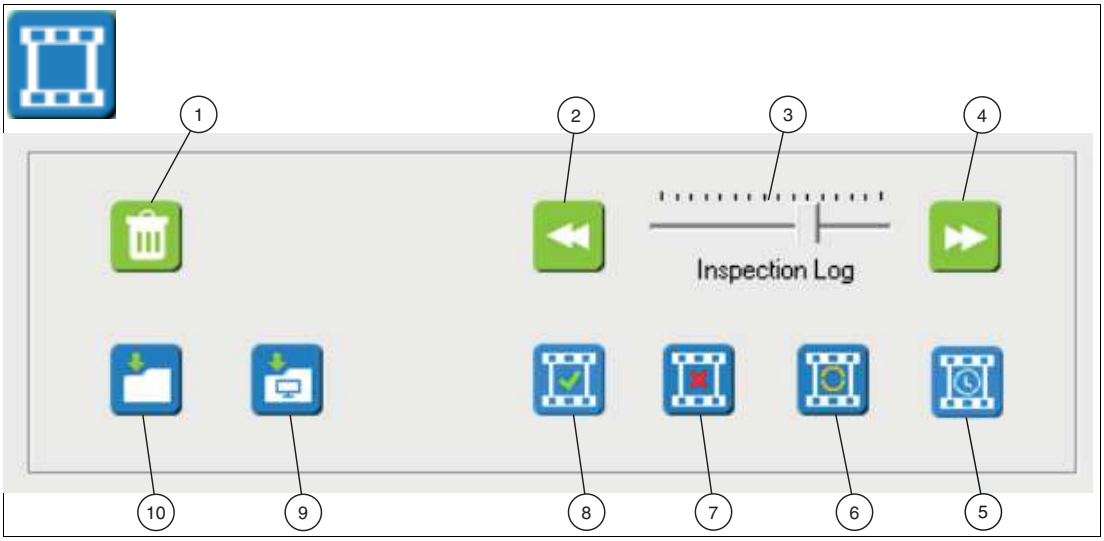


Function	Option	Description
Image Update		Controls the display of the image area during the inspection runtime:
	Enable	Enables update of the display. Uncheck the "Enable" check box to disable all updates or disable the display in the image area.
	Stop Update on Failure	Select the "Stop Update on Failure" check box to freeze the display in the event of an error. The sensor continues to run and tests parts. Only the display is affected. The "Reset Statistics" button changes to "Reset Display." Click "Reset Display" to resume a live display.
Update Categories		Select which images or results cause the display to update:
	All	All tested parts: Pass, Recycle, Fail
	Pass Only	The display is updated for "Pass" parts.
	Fail Only	The display is updated for "Fail" parts.
Update Frequency		Selects how often the display is updated.
	Continuous	Displays all parts in the selected category.
	Every 10th	Display every 10th part in the selected category.
	Every 100th	Display every 100th part in the selected category.

Function	Option	Description
Runtime Indicators		Controls the "Monitor" display in the configuration and status window during the inspection runtime.
	Large Text	Displays "Pass, "Recycle", or "Fail" in large font with a colored background (green, yellow, red).
	Progress Bar	Displays a horizontal bar with red, yellow, and green colored areas corresponding to the percentages of "Failed", "Recycled", or "Passed" parts.
	Graph	Displays a bar chart with three bars in red, yellow, and green, corresponding to the number (count) of "Failed", "Recycled", or "Passed" parts.

History Recall

You can use the "History Recall" function to display and save the recorded history of the measurement results.



Item	Function	Description
1	Clear History	Clears (deletes) all images and data from the history log.
2, 3, and 4	Inspection Log	Use the double-arrow keys to scroll through the records. The sensor saves the last 20 records for the resolution of 640 x 480 pixels and two recordings for the resolution of 1280 x 960 pixels.
5	Time Log	Displays all parts: passed, recycled, and failed.
6	Recycle Log	Displays the history of the recycled parts.
7	Fail Log	Displays the history of the "failed" parts.
8	Pass Log	Displays the history of the "passed" parts.
9	Save Screen	Saves the entire screen of the PC display, all three window areas.
10	Save Image	Saves the image of the object without graphics or measurement tools.

7.7 User Administration

To prevent others from changing your jobs, you can set a password for your sensor. Without entering a password, other users can only access the "Monitor" screen to display the "Run" screen.

Creating User Accounts

1. Enter the administrator password.
2. Click "Log In."
3. Check "Enable."
4. Click "Add" to add a new account.
5. Enter your user name and the login name. Assign a password.

Note

The user name is limited to a maximum of 31 characters.

The user's full name is limited to a maximum of 63 characters.

Passwords are limited to a minimum of 6 characters and a maximum of 15 characters.

6. Set the options for the user account.
7. When you are finished, click "Log out."

Note

If you leave "Enable" enabled and log out, all users will need to log back in to use the sensor menus. If you clear the "Enable" check box, a password is not required and access is not restricted.

Enabling or Disabling Access Permission

1. Click the "User administration" button in the navigation bar.
2. Log in as an administrator to make changes. Enter the administrator password and click "Log In."
3. You can check "Auto LogOff after" and set the time taken until it logs off automatically.
4. Click the "Enable" check box if you want to enable user accounts and require a password each time you attempt to access the sensor. Clear the check box if you do not want to require logins. Anyone can see the "Monitor" window without logging in.
5. Click "Log Out" when you are finished.

Note

If you leave "Enable" enabled and log out, all users will need to log back in to use the sensor menus. If you clear the "Enable" check box, a password is not required and access is not restricted.

8 Appendix

8.1 System Settings in Windows®

The application interface is accessed via the Ethernet connection to a PC running Microsoft® Internet Explorer 6 or higher.

Configuring the PC for Sensor Access

The paths to some tools depend on your Windows® configuration and display settings. Some elements may appear in the left or right window range of the menus.

Setting the System Performance in Windows® 10

Use your PC's performance options to customize the visual effects of Windows® 10 or Windows® 7. This setting increases the overall performance of your PC as well as the performance of the sensor applications. If you do not reduce or disable the performance options on your PC, it may cause malfunctions of the "VOS Configuration Tool" software.

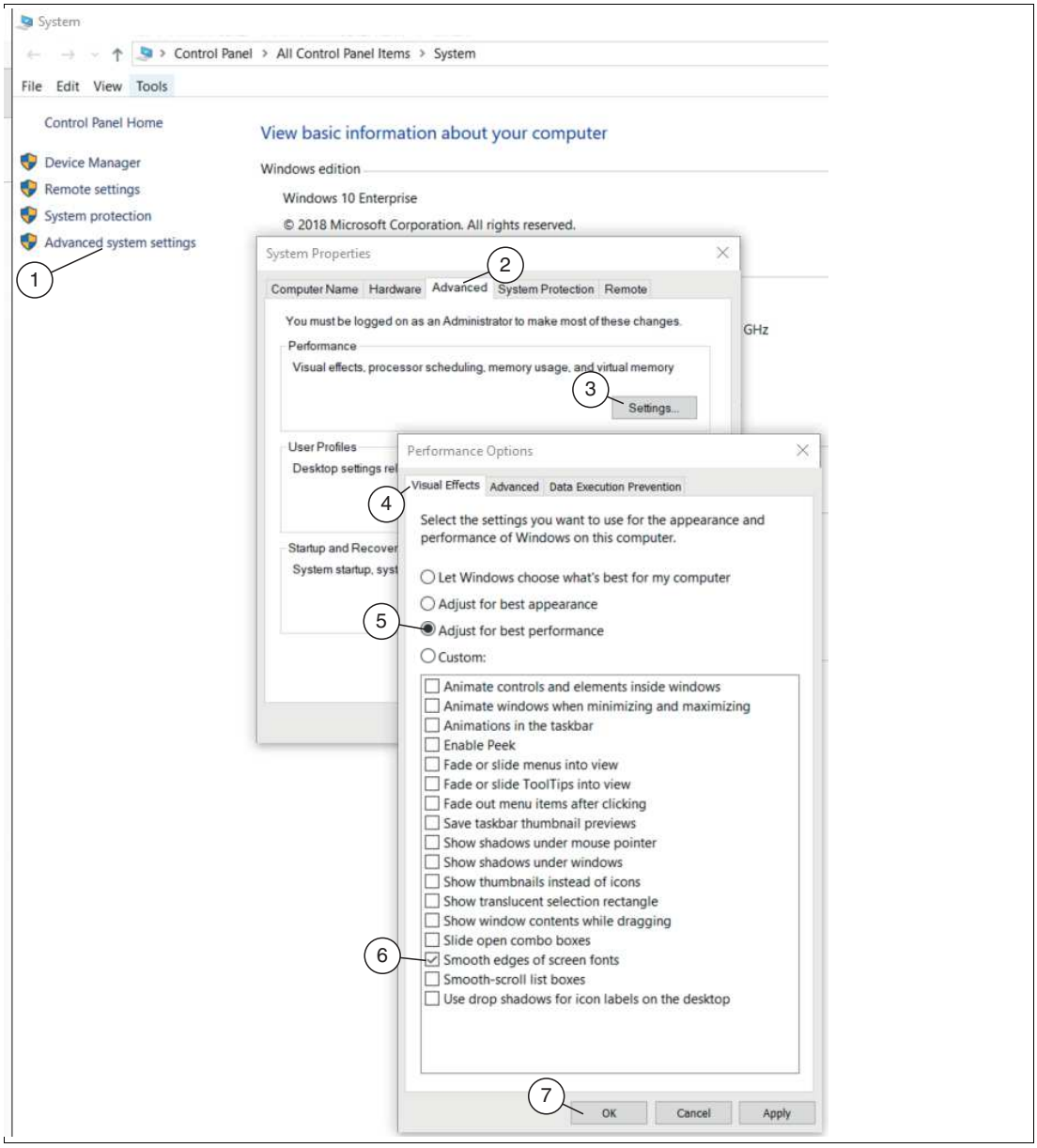


Figure 8.1 Performance options

1. Open the system control panel. Click on "System".

   ↳ The start page of the system control panel opens.

2. From the start page of the system control panel, select the "Advanced System Settings" menu (1).

   ↳ The "System Properties" window opens.

3. In the "System Properties" window, select the "Advanced" tab (2).

4. On the "Advanced" tab, select the "Settings" function (3).

   ↳ The "Performance Options" window opens.

5. In the "Performance Options" window, select the "Visual Effects" tab (4).

6. On the "Visual Effects" tab, select the "Adjust for best performance" setting (5).

7. Enable the following settings: "Refine edges of screen font" (6) and "Use visual styles for windows and buttons."

   ↳ The performance setting is changed to "Custom".

   Note

   Some systems may not have the "Use visual styles for windows and buttons" setting.

8. Click "OK" (7) to confirm your selection.

   ↳ The design is changed to resemble Windows® Classic.

Disabling or Customizing User Account Control in Windows® 10

User Account Control (UAC) is a security tool in Windows® 7 and Windows® 10. UAC controls the assignment of rights to individual programs. If a program wants to make changes to your system, UAC responds and you must actively agree to the change. This also prevents the system from loading and running the **"Vision Configuration Tool"** software. As soon as you start the **"Vision Configuration Tool"**, a warning message is issued by the system. Depending on the UAC level, you may need to enter an administrator password.
If your PC is in a closed and secure factory environment that is not connected to the Internet and an administrator account is also used, it is usually safe to switch off UAC.
If your PC is connected to an office network and the Internet, you should use the default setting or a higher security setting. The software should be manually approved when the prompts appear.
The procedure for customizing the UAC settings or disabling UAC completely is described below.

Figure 8.2 Settings

1. Press the [Windows®] + [X] key combination and then click on the "Settings" menu item (1).

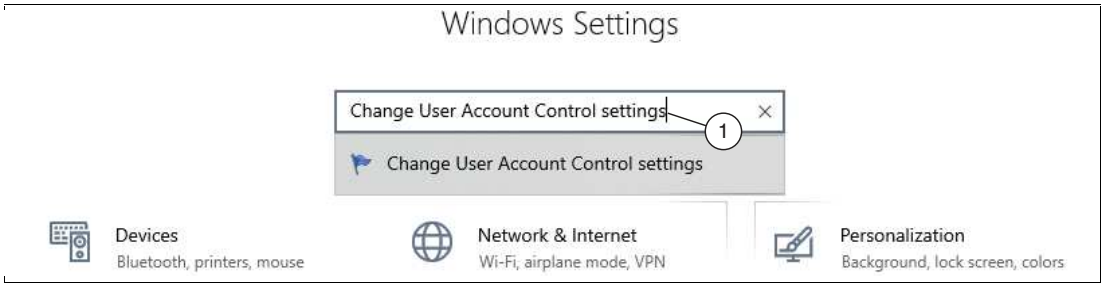


Figure 8.3 Search

2. Type "Change user account control settings" (1) in the upper search bar and confirm with [Enter].

↳ The "User account control settings" window opens.

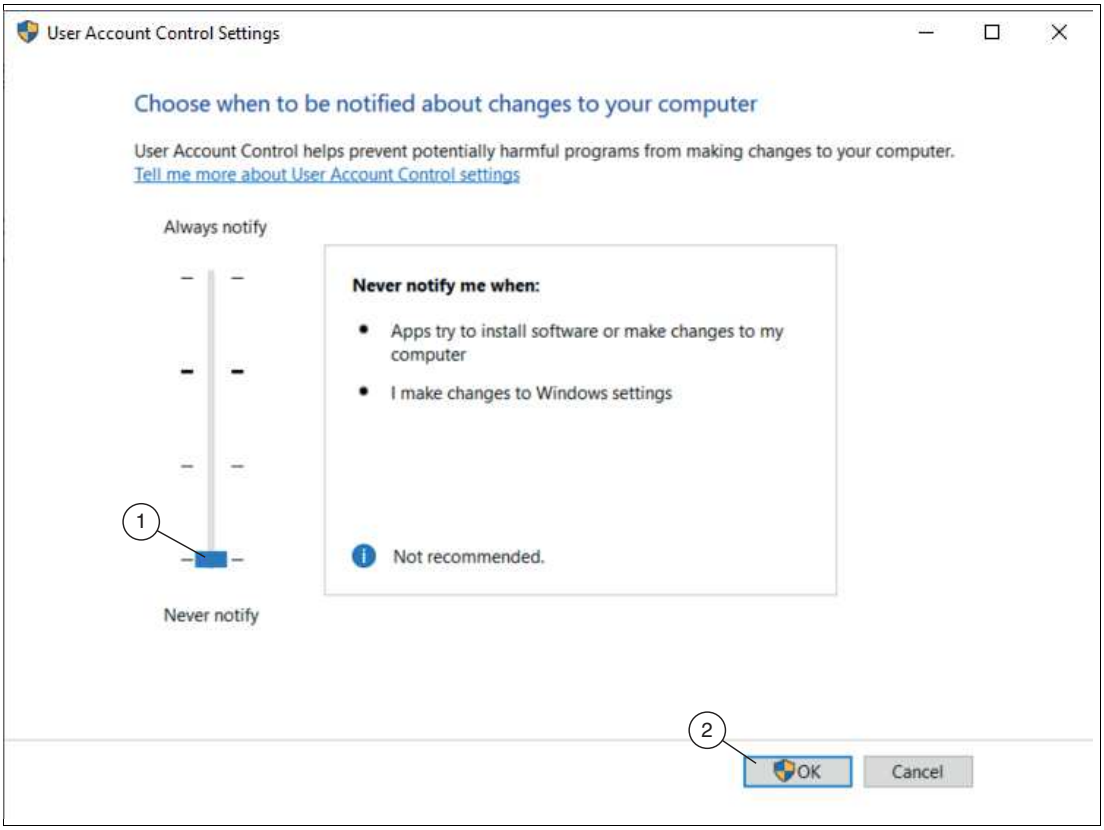


Figure 8.4 Disabling UAC

3. Use the **slider** (1) on the left to customize notifications from user account control. If the **slider is at the bottom**, **UAC is disabled** and you are no longer notified when a program changes settings on your system. If the slider is at the top, you will always be notified. Save your settings with "OK" (2).

Switching Windows® Firewall On or Off

Windows® Firewall protects your system from unauthorized access from the Internet. For your own security, the firewall should always be enabled. However, it is sometimes necessary to disable it. The sensor communicates with the PC via the network connection. Windows® Firewall blocks this communication. The easiest way to use the sensor software is to turn off the firewall. If you need to connect to an office network or the Internet, you can turn on the firewall again.

1. Go to the Windows® 10 Start menu at the bottom left of the screen. In the search, type "Windows® Defender Firewall" and select the appropriate entry.

   ↳ The Windows® Firewall settings will open.

   Note

   If you see a message that the service is not running, click on "No" and do not start the firewall. Continue downloading the **Nexus** software.

2. In the firewall options, select "Turn Windows Defender Firewall on or off".

   ↳ The "Customize Settings" window opens with the default settings for your Windows® Firewall.

3. In the settings, select the two "Disable Windows Defender Firewall" options and confirm your entry with "OK".

   Note

   To turn the firewall back on, select "Enable Windows Defender Firewall" for both options.

   ↳ Windows® Firewall is disabled, and network security for Windows® 10 is slightly eased.

   Note

   You also have the option to leave the firewall enabled. In this case, when you run the application, you must approve each security-critical step because the firewall reports it as "potentially unsafe".

Windows® Firewall—Adding an Exception

1. Go to the Windows® 10 Start menu at the bottom left of the screen. In the search, type "Windows Defender Firewall" and select the appropriate entry.

   ↳ The Windows® Firewall settings will open.
2. In the new window, select the option "Allow a program or feature through Windows Firewall".
3. Click on "Change settings", then click on "Allow other app..."
4. If the **"VOS Configuration Tool"** software is already in the list, simply tick the box for the rule.

   Note

   If this is not the case, or the enabling still does not work, you must use the "Browse..." option to select the program.
5. Confirm with "OK". All open windows can be closed.

Changing the IP Address in Windows® 10

The following section describes how to customize the network connection settings of your Windows® PC for connection to your sensor.

1. Open the system control panel.
2. Click on "View network status and tasks".
3. Select "Change adapter settings" from the list on the left.
4. Right-click on the appropriate network and click on "Settings".
5. Then double-click on the entry "Internet protocol, version 4 (TC/IPv4)".

   ↳ The settings window for the TCP/IP protocol opens.
6. Select "Use the following IP address:" and enter the IP address of the device, but only the first three segments of the IP address. The last segment must be different from the IP address of the sensor. The new IP address must not already be in use elsewhere in the network.

   Note

   Contact your system administrator for further instructions. The subnet mask should be set to 255.255.255.0.

2021-03

Your automation, our passion.

Explosion Protection

- Intrinsic Safety Barriers
- Signal Conditioners
- FieldConnex® Fieldbus
- Remote I/O Systems
- Electrical Ex Equipment
- Purge and Pressurization
- Industrial HMI
- Mobile Computing and Communications
- HART Interface Solutions
- Surge Protection
- Wireless Solutions
- Level Measurement

Industrial Sensors

- Proximity Sensors
- Photoelectric Sensors
- Industrial Vision
- Ultrasonic Sensors
- Rotary Encoders
- Positioning Systems
- Inclination and Acceleration Sensors
- Fieldbus Modules
- AS-Interface
- Identification Systems
- Displays and Signal Processing
- Connectivity

Pepperl+Fuchs Quality
Download our latest policy here:

www.pepperl-fuchs.com/quality

PEPPERL+FUCHS

www.pepperl-fuchs.com

Printed in Germany / DOCT-6801A