EN

Software wenglor uniVision 3 DNNF023

Operating instructions

Subject to change without notice
Available as PDF file only
Revision level: 11.12.2023
Version: 1.1.0
www.wenglor.com

Table of Contents

1. General 5
1.1 Information Concerning these Instructions 5
1.2 Explanations of Symbols 5
1.3 Limitation of Liability 6
1.4 Copyrights 6

2. For Your Safety 6
2.1 Use for Intended Purpose 6
2.2 General Safety Precautions 7

3. Technical Data 7

4. Software Installation 8
4.1 Installation Basics 8
4.2 Software Compatibility 8

5. Basics in wenglor uniVision 3 10
5.1 Mounting, Electrical and Network Connection 10
5.2 Network Overview 10
5.2.1 Network Cable Connection 10
5.2.2 Network Basics 11
5.2.3 Network Settings of PC with Software wenglor uniVision 3 12
5.2.4 Default Network Settings of Machine Vision Devices 12
5.3 Data Acquisition 13
5.4 Data Evaluation 14
5.5 uniVision Job 17
5.6 Modules 18
5.6.1 Inputs and Outputs 19
5.6.2 Basic Data Types 19
5.6.3 Status of Outputs and Error Handling 21
5.6.4 Module States 21
5.6.5 Status of Processing Instance 22
5.7 Interface Overview 24
5.7.1 Interface for Automated Production 24
5.7.2 Network Interfaces for Processing Instances 25
5.7.3 Network Interfaces for uniVision Simulator 27
5.8 SFTP Server on Machine Vision Devices 28
5.9 FTP/SFTP Client on Machine Vision Devices 30

6. Software uniVision 3 33
6.1 Connecting to Machine Vision Device 33
6.2 uniVision Simulator 36
6.3 Structure 38
6.3.1 Menu Bar 40
6.3.2 Icon Menu Bar 43
6.3.3 Job Tree 45
6.3.4 Status Bar 46
6.3.5 Scene with Data 47
6.3.6 Image Container Viewer 47
6.3.7 Extended View 47
6.3.8 Toolbox 48
6.4 Teach+ 49
6.5 Licenses for uniVision Modules 51
6.5.1 Smart Camera B60 51
6.5.2 uniVision Simulator 52
6.5.3 Requesting License Upgrades 53
6.6 Job Converter 58

7. Modules 59
7.1 Module Application 59
7.2 Device Smart Camera for B60 60
7.3 Image-Based Preprocessing Modules 65
7.3.1 Module Image Calibration 65
7.3.2 Module Image Region 73
7.3.3 Module Image Filter 75
7.3.4 Module Image Locator 76
7.3.5 Module Image Pattern Match 81
7.3.6 Module Image Coordinate System 86
7.4 Image-Based Processing Modules 89
7.4.1 Module Image Threshold 89
7.4.2 Module Image Threshold HSV 93
7.4.3 Module Image Blob 95
7.4.4 Module Image Comparison 99
7.4.5 Module Image Measure 100
7.4.6 Module Image OCR 105
7.4.7 Module Image Code 1D 110
7.4.8 Module Image Code 2D 112
7.5 Module HALCON Script 114
7.5.1 Basics in HDevelop 116
7.5.2 Basics in Module HALCON Script 121
7.6 Calculation Modules 126

7.6.1 Module Spreadsheet 126
7.6.2 Module Counter 131
7.6.3 Module Match Code 131
7.6.4 Module Teach Numeric 132
7.6.5 Module Math 133
7.6.6 Module Logic 134
7.6.7 Module Numeric Comparison 134
7.6.8 Module Statistic 135
7.7 Standard Interfaces 136
7.7.1 Device TCP 136
7.7.2 Device UDP 138
7.7.3 Device FTP 139
7.7.4 Device Industrial Ethernet 141
7.8 Robot Vision 142
7.8.1 Calibration Job in uniVision 142
7.8.2 Installation of URCap 143
7.8.3 UR Installation 145
7.8.4 Detect Objects Job in uniVision 151
7.8.5 UR Program 153

8. Webbased Visualization 156

9. LIMA Interface 161
9.1 Jobs 162
9.2 Edit Job Tree 163
9.3 Read Image 164
9.4 Trigger, Start and Stop 165
9.5 Get Status 166
9.6 Teach+ 169

10. Third-Party Licenses 170

11. Change Index of Operating Instructions 170

12. Attachments 170
12.1 Module States 170
12.2 Disabled HALCON Operators 207

1. General

1.1 Information Concerning these Instructions

- These instructions enable safe and efficient use of DNNF023
- These instructions are an integral part of the product and must be kept on hand for the entire duration of its service life.
- Local accident prevention regulations and national work safety regulations must be complied with as well.
- The product is subject to further technical development, and thus the information contained in these operating instructions may also be subject to change. The current version can be found at www.wenglor.com in the product's separate download area.

NOTE!
The operating instructions must be read carefully before using the product and must be kept on hand for later reference.

1.2 Explanations of Symbols

- Safety precautions and warnings are emphasized by means of symbols and attention-getting words.
- Safe use of the product is only possible if these safety precautions and warnings are adhered to.

The safety precautions and warnings are laid out in accordance with the following principle:

ATTENTION-GETTING WORD
Type and Source of Danger!
Possible consequences in the event that the hazard is disregarded.
- Measures for averting the hazard.

The meanings of the attention-getting words, as well as the scope of the associated hazards, are listed below:

DANGER!
This word indicates a hazard with a high degree of risk which, if not avoided, results in death or severe injury.

WARNING!
This word indicates a hazard with a medium degree of risk which, if not avoided, may result in death or severe injury.

CAUTION!
This word indicates a hazard with a low degree of risk which, if not avoided, may result in minor or moderate injury.

ATTENTION!
This word draws attention to a potentially hazardous situation which, if not avoided, may result in property damage.

NOTE!
A note draws attention to useful tips and suggestions, as well as information regarding efficient, error-free use.

1.3 Limitation of Liability

- The product has been developed taking into account the state of the art as well as the applicable standards and guidelines.
- We reserve the right to make technical changes.
- A valid declaration of conformity can be found at www.wenglor.com in the download area of the product.
- wenglor sensoric elektronische Geräte GmbH (hereinafter „wenglor“) accepts no liability for:
  - » Failure to observe the operating manual,
  - » Unsuitable or improper use of the product,
  - » Excessive use, incorrect or negligent treatment of the product,
  - » Incorrect installation or commissioning,
  - » Use of untrained personnel,
  - » Use of unauthorized spare parts or
  - » Improper or unauthorized changes, modifications or repair work to the products.
- This operating manual does not contain any guarantees/warrantees from wenglor with regard to the processes described or certain product properties.
- wenglor assumes no liability with regard to printing errors or other inaccuracies contained in this operating manual, unless it can be proven that wenglor was aware of the errors at the time the operating manual was created.

1.4 Copyrights

- The contents of these instructions are protected by copyright law.
- All rights are reserved by wenglor.
- Commercial reproduction or any other commercial use of the provided content and information, in particular graphics and images, is not permitted without previous written consent from wenglor.

2. For Your Safety

2.1 Use for Intended Purpose

The software wenglor uniVision 3 is an intuitive parameterization software for solving complexe image processing applications. It contains a comprehensive toolbox for individual job configuration, common industrial interfaces for easy integration and a flexible and adjustable webbased visualization. It allows simple initial start-up and readjustment via uniVision Simulator based on Teach+ functionality.

Typical applications are reading, 1D codes, 2D codes or OCR, presence check of best before date or checking the correct assembly at automatic manufacturing processes.

Supported Machine Vision Devices:

- Smart Camera B60
- uniVision Simulator (offline)

NOTE!
For details about the Machine Vision Devices check the operating instructions of the device.

2.2 General Safety Precautions

NOTE!

- These instructions are an integral part of the product and must be kept on hand for the entire duration of its service life.
- In the event of possible changes, the respectively current version of the operating instructions can be accessed at www.wenglor.com in the product's separate download area.
- Read the operating instructions carefully before using the product.

3. Technical Data

Technical Data	DNNF023
General Data	
Use	For B60
Version	3.1.0
Language	DE, EN, FR, IT, ES, PT, NL, HU, TR, ZH, RU
Licensing model	Freeware
Function	
Configuration Software	Yes
Display Software	Yes
Diagnostics Software	Yes
System Requirements	
Processor (minimum)	Intel Core i3 (6th generation)
RAM (minimum)	2 GB RAM
Free hard disc space	500 MB
Minimum resolution	1280 × 1024 Pixel
Maximum resolution	4096 × 2160 Pixel
Minimum Browser Version	Chrome 108 Chromium 111 Firefox 108 Microsoft Edge 108
Operating System	
Windows 10. 64 bit	Yes
Windows 11	Yes
Interface	
Ethernet	Yes

NOTE!

- 2 GB RAM are needed for each instance of the uniVision software as the software can be started several times in parallel (see “5.4 Data Evaluation”).
- Using the software wenglor uniVision 3 on virtual machines is not supported.

4. Software Installation

4.1 Installation Basics

The software wenglor uniVision 3 is available on the wenglor website:
https://www.wenglor.com/product/DNNF023

Download the relevant version of the software wenglor uniVision 3 from the “Downloads” category “Soft- and Firmware” (see “4.2 Software Compatibility”). Install the software on the PC by running the installation wizard.

NOTE!

- Supported operating systems:
  - » Windows 10, 64 bit
  - » Windows 11
- Supported minimum browser versions for the webbased operator interfaces (tested on Windows 10 22H2 and Windows 11 22H2 PCs):
  - » Chrome 108
  - » Firefox 108
  - » Microsoft Edge 108
  - » Chromium 111
- Accepting the license agreement is mandatory in order to install the software.
- Admin rights are necessary to install the software wenlgor uniVision 3.

4.2 Software Compatibility

The following uniVision 3 products have their own version number:

- Software wenglor uniVision 3
- Firmware Smart Camera B60
- uniVision job

The version numbers of software and firmware products consist of three digits (e.g. Software wenglor uniVision 3.1.0):

- Major release: The first digit is changed (no job compatability)
- Feature release: The second digit is changed (jobs must be converted via job converter)
- Bugfix release: The third digit is changd (jobs are completed)

NOTE!
In order to work with uniVision 2 jobs or uniVision 2 devices, install the software wenglor uniVision 2. For details, please check https://www.wenglor.com/product/DNNF020. Converting uniVision 2 jobs to uniVision 3 jobs and connecting to uniVision 2 devices with the software wenglor uniVision 3 is not supported. Installing and using the software wenglor uniVision 2 and wenglor uniVision 3 in parallel is possible.

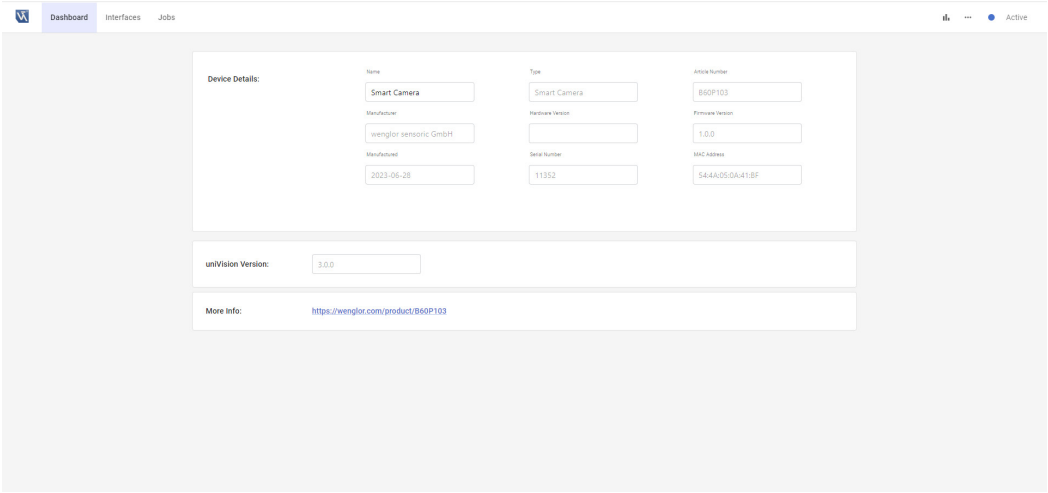
The following generally applies with regard to the compatability of uniVision jobs:

- For uniVision jobs, there are only two digits in the version number, as the job format does not change for bugfixes (third digit changed).
- Jobs with a specific two-digit version number can only be opened on Machine Vision Devices with a suitable version of the firmware or set up using a suitable version of the software.
- After a firmware update with new features (second digit changed) on a device, the corresponding software version must be installed and existing jobs must be converted.

Job version	Software wenglor uniVision 3 version	Firmware versions
3.0	uniVision 3.0.x	Firmware B60 1.0.x
3.1	uniVision 3.1.x	Firmware B60 1.1.x

Example
To work with a Smart Camera B60 with firmware version 1.0.0 use the compatible software version wenglor uniVision 3.0.0 and work with uniVision jobs with version 3.0

NOTE!
The uniVision version of the device (e.g. Smart Camera B60) is shown on the Dashboard of the device website. For details, check the operating instructions of the device.



NOTE!

Installing different wenglor uniVision 3 versions on Windows PCs with changes at the second digit (feature release) is possible in parallel. In case of changes at the third digit (bugfix release), the new version replaces the old version at installation.

5. Basics in wenglor uniVision 3

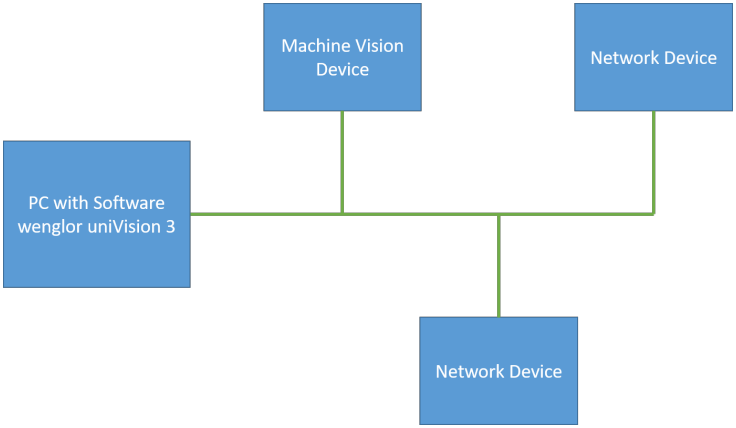
5.1 Mounting, Electrical and Network Connection

For details about mounting, electrical wiring and network connection, check the operating instructions of the relevant Machine Vision Device.

5.2 Network Overview

5.2.1 Network Cable Connection

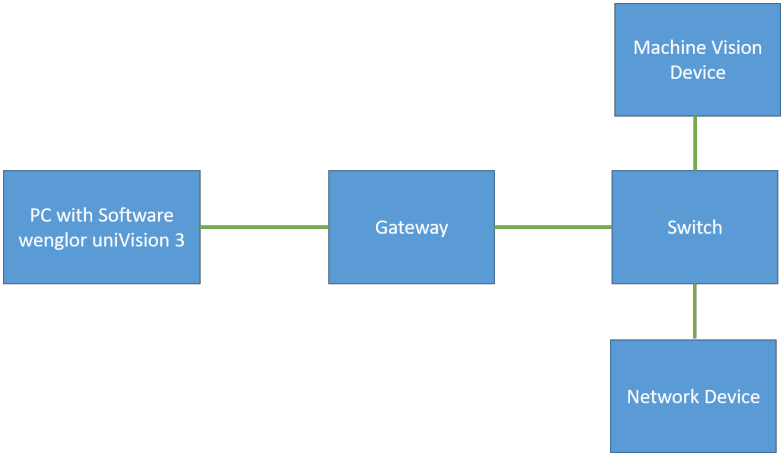
Connect the network cable from the Ethernet connector of the Machine Vision Device to the PC with the software wenglor uniVision 3 or to a switch in order to work with several Network Devices.



NOTE!

- For details about the network connector of the Machine Vision Device, check the operating instructions of the device.
- Cabling must be capable of 1 GBit/s throughout the entire network.
- It is recommended to use Machine Vision Devices only in local networks without internet access because of security reasons.

It is possible to use a gateway between the Machine Vision Device and the PC with the software wenglor uniVision 3.



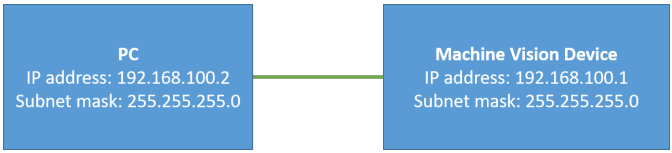
5.2.2 Network Basics

In order to establish a connection from the software wenglor uniVision 3 to the Machine Vision Device, the Machine Vision Device and the PC must be in the same network. The network part of the IP address of the Machine Vision Device must coincide with the network part of the IP address of the PC with the software wenglor uniVision 3. The device part of the IP address must be different for the Machine Vision Device and the PC.

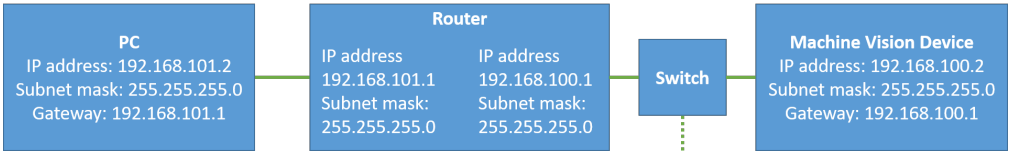
Address format for IPv4:

	Network Part	**Device Part** (host part)
IP address	192.168.100.	**001**
Subnet mask	255.255.255.	**000**

For example, use the following network configuration to connect from the PC with the software wenglor uniVision 3 to the Machine Vision Device.



For example, use the following network configuration if working with a gateway between the PC with the software wenglor uniVision 3 and the Machine Vision Device.



NOTE!
Use the software wenglor Discovery Tool to see the network settings of the PC and to see and edit the network settings of the Machine Vision Device in order to bring it in the same network like the PC. For details, please check the operating instructions of the software wenglor Discovery Tool (see https://www.wenglor.com/product/DNNF022). Alternatively, network setup is possible via the device website. For details, see the operating instructions of the Machine Vision Device.

5.2.3 Network Settings of PC with Software wenglor uniVision 3

Open the network settings of the PC and make sure that the network configuration fits, e.g. use the static IPv4 address:

- IP address: 192.168.100.2
- Subnet mask: 255.255.255.0
- Gateway: 0.0.0.0

NOTE!
- By default, the IP address of the network adapter card of the PC is set to obtain IP address by DHCP server (automatic allocation). Change the setting to a static IP address and select a unique IP address in the network.
- For details about the network settings of the PC, check the operating instructions of the operating system.

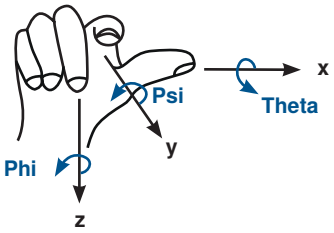
5.2.4 Default Network Settings of Machine Vision Devices

Default LAN network settings of Machine Vision Devices:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Gateway: 0.0.0.0

5.3 Data Acquisition

Machine Vision Devices capture data (e.g. images) which have an original coordinate system as reference. A right-handed coordinate system is used in the software wenglor uniVision 3. The following angles and rotations around the axis are indictated.



Phi (Z rotation)	Rotation around the Z-axis
Theta (X rotation)	Rotation around the X-axis
Psi (Y rotation)	Rotation around the Y-axis

Use the software wenglor uniVision 3 to configure the data acquisition in order to get suitable data quality. The better the data quality (e.g. the sharper the image), the easier the data evaluation. For example, the trigger settings define when the Machine Vision Device captures data.

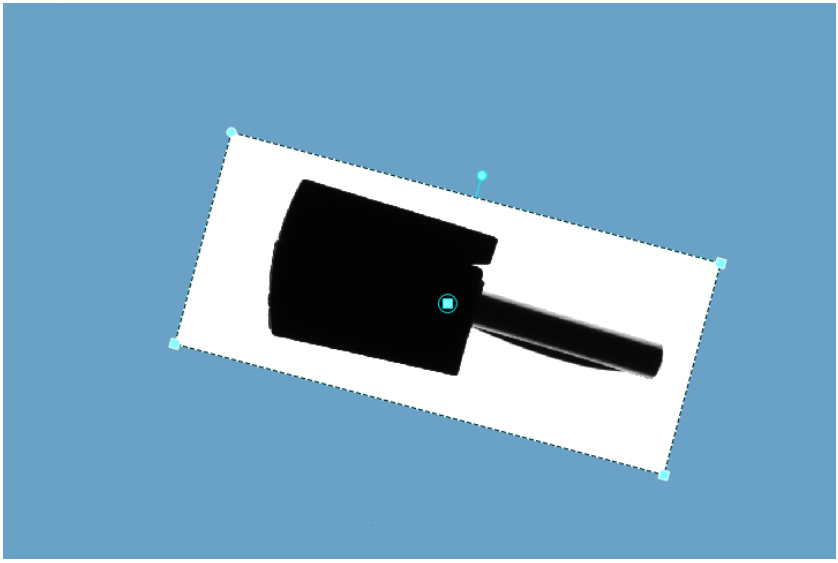
NOTE!
Make sure that the Machine Vision Device is ready to receive trigger signals (e.g. avoid over triggering, so that during exposuring the next trigger signal is already sent). For details about forbidden trigger timings, check the operating instructions of the Machine Vision Device.

Image Analysis

The origin of the coordinate system for Machine Vision Devices that capture images is located in the top left corner of the image (in the center of the top left pixel). Images are shown in the x-y plane:

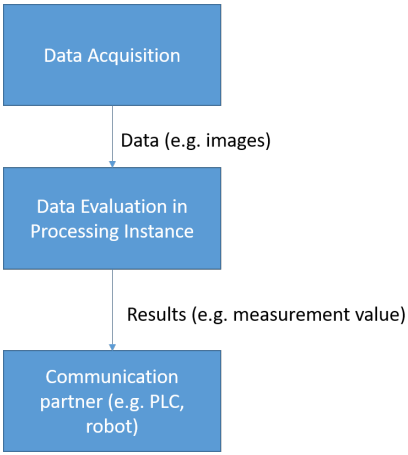
- X-Axis: Positive to the right
- Y-Axis: Positive downward

Rotations are thus possible around the Z-axis (Phi). Positive angles are in clockwise direction. The example shows a rectangle with a rotation of 10°.



5.4 Data Evaluation

Data (e.g. images) are evaluated in the Processing Instance and results are sent to the communication partner via different interfaces.

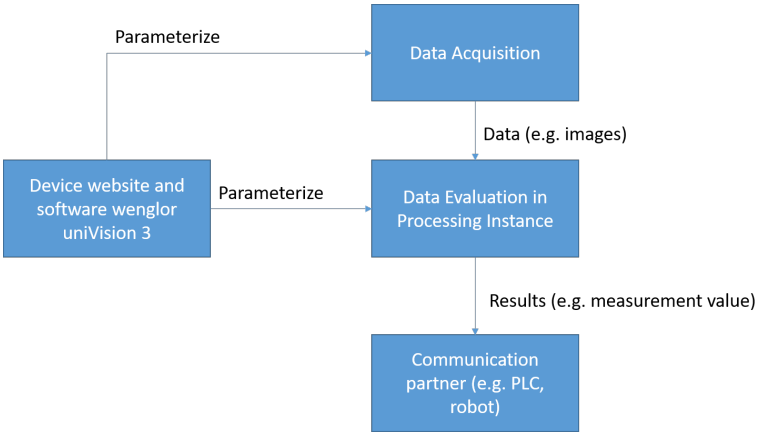


NOTE!
When setting up the data acquisition frequency, make sure that data evaluation (processing time of job), interfaces (e.g. network) and communication partners (e.g. PLC cycle times) are fast enough to receive all results, otherwise reliable applications cannot be guaranteed.

The configuration of the Processing Instance is done via the device website (see operating instructions of Machine Vision Device) and the software wenglor uniVision 3. Connect from the PC with the software wenglor uniVision 3 to the Processing Instance in order to parameterize it. Following the parameterization, the connection can be closed and the Machine Vision Device carries out evaluations independently.

- Processing Instance: Core to evaluate data (e.g. images)
- Software wenglor uniVision 3: Software to parameterize data acquisition and Processing Instances

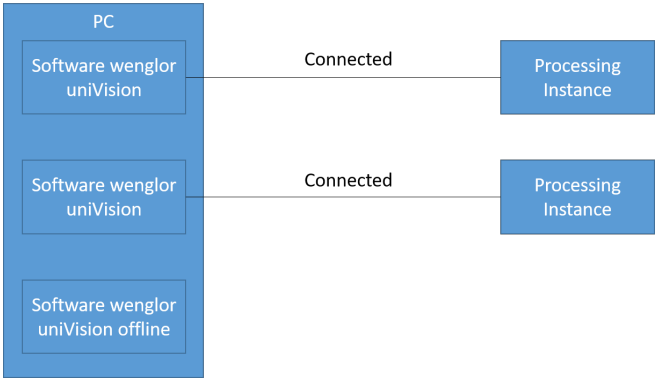
NOTE!
The software wenglor uniVision 3 is to parameterize the Processing Instance. It is no software to visualize results in automated production as it reduces the maximum performance of the Machine Vision Device significantly. Furthermore, it cannot be guaranteed that the software wenglor uniVision 3 is always informed about job changes by different interfaces (e.g. webbased visualization). Use instead the webbased visualization to visualize relevant results in a permanent way (see “8. Webbased Visualization”).



The Processing Instance runs on Machine Vision Devices and offline without a real Machine Vision Device in the uniVision Simulator. The uniVision Simulator allows to fine tune the job configuration and to validate the settings with stored data of the Machine Vision Device.

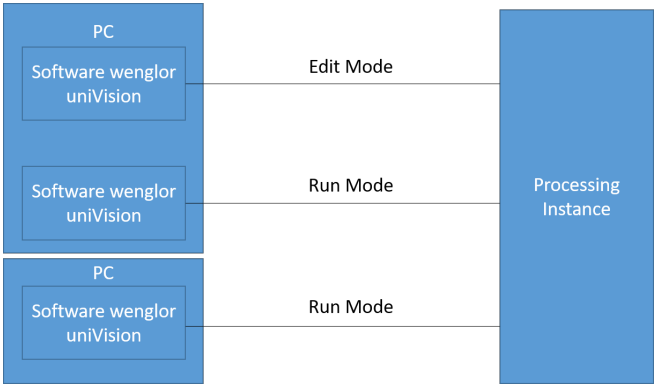
The software wenglor uniVision 3 can be started several times on the same PC in order to connect to several different Processing Instances. One instance of the software wenglor uniVision 3 can be started offline without a real Machine Vision Device by using the uniVision Simulator.

NOTE!
Close the offline job in the uniVision Simulator on a PC first (also by different users on the same PC), before opening another offline job.



The software wenglor uniVision 3 can connect several times to the same Processing Instance running on a real Machine Vision Device. Connections are possible from one or several PCs to the Processing Instance.

- One connection in edit mode
- Up to five connections in run mode



NOTE!
Ports used by the software wenglor uniVision 3 are described in “5.7.2 Network Interfaces for Processing Instances”.

5.5 uniVision Job

The Processing Instance evaluates one uniVision job. Creating and loading different jobs for different machine vision applications is possible. Jobs are stored directly on the Machine Vision Devices. Job exchange between the PC with the software wenglor uniVision 3 and the Machine Vision Device is possible (see “5.8 SFTP Server on Machine Vision Devices”). This enables also to edit uniVision jobs offline with the uniVision Simulator.

The startup behavior of the Processing Instance defines which job is loaded when booting the Machine Vision Device (see operating instructions of Machine Vision Device).

- Fix start job: The Machine Vision Device boots with a fix start job that can be defined.
- Last loaded job: The Machine Vision Device boots with the last loaded job (default).

NOTE!
The maximum storage space for jobs depends on the Machine Vision Device. For details, check the device website (Diagnosis → Storage) and the operating instructions of the Machine Vision Device.

Jobs contain one or several modules and devices.

- Module: Software module (e.g. Module Image Measure) that typically can be used several times in one job.
- Device: Defines an interface (e.g. Device TCP) that typically can be used only once in one job.

NOTE!
The maximum number of modules depends on the available RAM of the Machine Vision Device. For details, check the device website (Diagnosis → RAM) and the operating instructions of the Machine Vision Device.

The modules can be arranged in any desired order because the job tree runs until all results are available. If all results have not been calculated after 10 runs, the evaluation is aborted and an error is forwarded.

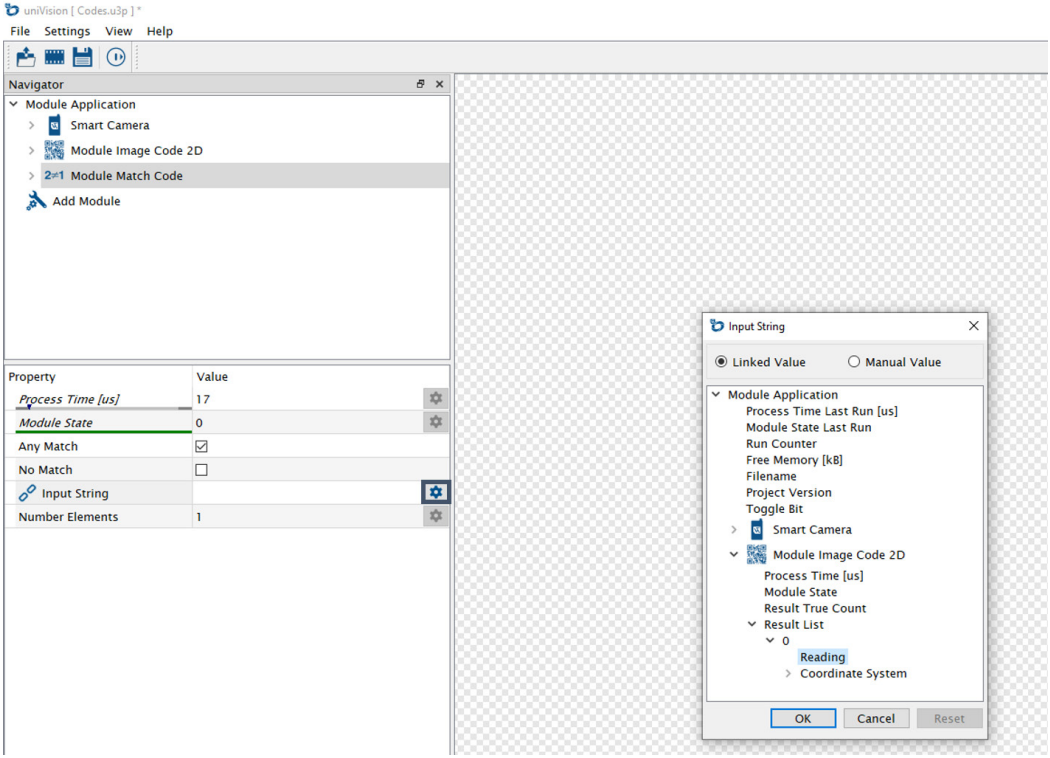
For each captured data the job runs through the Processing Instance and one result is sent to the communication partner. This means, there is typically one result for each trigger signal. If data acquisition is faster than data evaluation, data is lost and signaled via data overflow. Data overflow is also shown via the status of the Processing Instance (see “5.6.5 Status of Processing Instance”).

5.6 Modules

The toolbox (see “6.3.8 Toolbox”) contains all available modules for the Machine Vision Device. Modules extract information out of data. They can be added to the uniVision Job and typically contain:

- Inputs
- Parameters
- Outputs

NOTE!
Parameters are defined statically in the module and inputs can be linked with outputs of other modules. Certain values can be either parameters with a fix configuration or inputs if linked with the output of another module. Opening the settings icon of values provides in such case the option to select “Linked Value” or “Manual Value”.



5.6.1 Inputs and Outputs

Typical inputs and outputs are:

- Image
- Region
- Coordinate System
- Further data (e.g. points)
- Basic Data Types (e.g. BOOL, DINT, REAL, CHAR)

Linking inputs of one module with outputs of other modules is possible in a flexible way. Only suitable types can be linked.

Input Image

Depending on the camera type different images are generated:

- Monochrome cameras provide 8 bit grey images
- Color images provide HSV, RGB or BGRA image (default: HSV and BGRA images)

Link any available image from the job tree as Input Image of the module.

NOTE!

- The supported type of input image depends on the software module.
- Linking the input image is typically mandatory for most of the modules so that the module runs successfully.

Input Coordinate System

By default, the original coordinate system of the camera is statically used for modules with an Input Coordinate System. It is possible to link another coordinate system from the job tree as Input Coordinate System. Coordinate Systems can be at a fix position or dynamic for each data evaluation. If the position of objects is not fix, use a dynamic coordinate system to track it so that other modules work at the correct position (e.g. that the Region of Interest moves with the object).

Further data (e.g. input point)

Further inputs can be linked depending on the module, e.g. input points or input xlds.

5.6.2 Basic Data Types

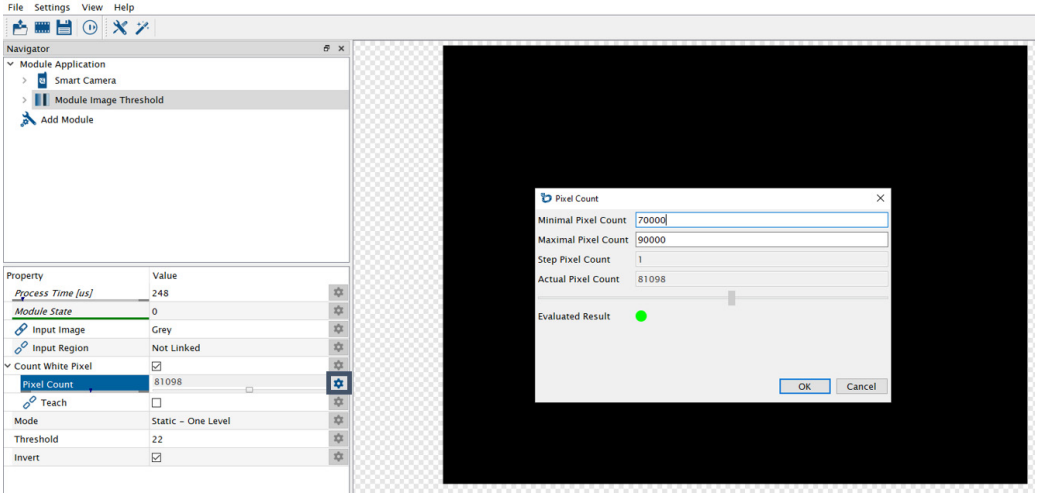
The software wenglor uniVision 3 uses the following basic data types:

- BOOL: For OK and NOK results (e.g. for digital outputs)
- DINT: For numbers without decimal places (e.g. run counter)
- REAL: For numbers with decimal places (e.g. coordinates of found point)
- CHAR: For strings (e.g. found 2D code)

Depending on the input of a module, linking to different basic data types is possible. Then the following conversion rules are applied:

- Module Input BOOL is linked to:
  - » Output BOOL: Returns true or false depending on the Boolean value
  - » Output DINT or REAL: Returns true if the current DINT or REAL value is within the thresholds (between minimum and maximum threshold) and returns false if the current value is out of tolerance (lower than minimum or higher than maximum threshold).
  - » Output CHAR: Returns true if the string is not empty and false if the string is empty
- Module Input DINT is linked to:
  - » Output BOOL: Returns 0 for BOOL value false and 1 for BOOL value true
  - » Output DINT: Returns the number
  - » Output REAL: Returns the number without decimal places (not rounded)
  - » Output CHAR: Returns the number of characters for the string (for example for the string “abc” the number 3 is returned)
- Module Input REAL is linked to:
  - » Output BOOL: Returns 0 for BOOL value false and 1 for BOOL value true
  - » Output DINT or REAL: Returns the number with decimal places
  - » Output CHAR: Returns the number of characters for the string (for example for the string “abc” the number 3 is returned)
- Module Input CHAR is linked to:
  - » Output BOOL: Returns “false” for BOOL value false and “true” for BOOL value true
  - » Output DINT or REAL: Returns the number
  - » Output CHAR: Returns the string

NOTE!
Click on the settings icon of outputs to adjust the minimal and maximal thresholds of outputs. Linking the output as input in another module will use the minimal and maximal thresholds for the data conversion.



5.6.3 Status of Outputs and Error Handling

Each output has one of the following states:

- Valid: Result is calculated successfully (typically)
- Error: Result could not be calculated successfully
- Not available: Result is not available so far (e.g. after booting or loading another job without captured data)

The error state of outputs is forwarded to further modules if the output is linked as input in other modules. If an output in error state is linked in one of the devices (e.g. Device TCP), the error handling of the relevant device defines the behavior in such a case. Typically, the value is substituted with another value that can be defined in the Error Handling of the Device. In Device TCP for example, strings are by default replaced with the value “Error###”.

5.6.4 Module States

The Module State shows the status of the module or the device. The following states exist:

- 0: Module was executed successfully
- Not 0: Module was not executed successfully

Overview of the most common Module States with possible solutions:

Module Status	Name	Description	Possible Solution
0	No Error	Module is running without error.	Nothing to do
1010	Input value error	One of the inputs of the module is in error state.	Identify the input in error state (shown in red color) and solve the error in the relevant module where the error state was inherited from.
1011	Return value error	One of the outputs of the module is in error state.	Solve the error within the module by adjusting the parameters.
1040	Image not linked	Input image in module is not linked.	Link an input image in the module.
1100	Module unlicensed	Unlicensed module	Activate the license of the module (see “6.5 Licenses for uniVision Modules”).
1104	Module not taught	Module not taught	Teach the module
1112	There is an error concerning the data memory access	Error in data memory access, e.g. by reaching the minimum storage limit of 200 MB in the local folder “output”	Delete files in local folder “output” (see “5.8 SFTP Server on Machine Vision Devices”) so that enough space is available to store new data via Device FTP (see “7.7.3 Device FTP”).

Module Status	Name	Description	Possible Solution
1113	There is an error concerning the FTP interface	Error in the FTP/SFTP interface caused by the following: • FTP/SFTP server is not available • No write permission for the FTP/SFTP user in the relevant folder • FTP/SFTP server is not fast enough to store all data	• Check that FTP/SFTP server is available. • Ensure that write permissions are activated for the FTP/SFTP user • Reduce the acquisition frame rate, reduce the number of stored images (e.g. via observer) or store only compressed data (e.g. in JPG format). See "7.7.3 Device FTP"
70010	Frame dropped because the queue was full	Data loss because of data overflow	• Reduce the acquisition frame rate or the trigger frequency. • Reduce the process time of the processing instance.
70052	Frame unavailable	Frame unavailable in Teach+ (see "6.4 Teach+")	• Record Teach+ with real data of the Machine Vision Device. Load real data of Machine Vision Device in Teach+ job.

NOTE!
The complete list of module states is available in the attachements (see section "12.1 Module States")

5.6.5 Status of Processing Instance

Level	Color	Name	Description	Possible Solution
Active	Blue	Running	Processing Instance is running without errors	Nothing to do
Warning	Yellow	Data overflow	Data evaluation takes longer than data acquisition	• Reduce the acquisition frame rate or the trigger frequency. • Reduce the process time of the processing instance.
Warning	Yellow	Command overflow	Commands are sent too fast to the Processing Instance	• Reduce the frequency to send commands to the Processing Instance. • Wait until the response is received before sending the next command (see "9. LIMA Interface").

Level	Color	Name	Description	Possible Solution
Warning	Yellow	FTP interface	Error in the FTP/SFTP interface caused by the following: • FTP/SFTP server is not available • No write permission for the FTP/SFTP user in the relevant folder • FTP/SFTP server is not fast enough to store all data • Reaching the minimum storage limit of 200 MB in local folder “output”.	• Check that FTP/SFTP server is available. • Ensure that write permissions are activated for the FTP/SFTP user. • Reduce the acquisition frame rate, reduce the number of stored images (e.g. via observer) or store only compressed data (e.g. in JPG format), see “7.7.3 Device FTP”. • Delete files in local folder “output” (see section “5.8 SFTP Server on Machine Vision Devices”) so that enough space is available to store new data via Device FTP (see section “7.7.3 Device FTP”).
Warning	Yellow	Unlicensed module(s)	One or several modules or devices within the job has no valid license.	Delete the modules or devices from the current job or request the license upgrade (see “6.5 Licenses for uniVision Modules”).
Error	Red	Incompatible project	The uniVision job version is not compatible to the firmware version of the Machine Vision Device.	Make sure that the uniVision job version and the firmware version of the Machine Vision Device are compatible. Use the job converter (see section “6.6 Job Converter) or install a suitable firmware version on the Machine Vision Device. For details about compatability see section “4.2 Software Compatibility”.
Error	Red	Project not available	The defined start job is not available after booting the device.	Select a valid start job on the device website. For details, see operating instructions of the Machine Vision Device.
Error	Red	Processing	One or several modules or devices within the job is in error state.	Connect to the Processing Instance on the device via the device website (Jobs tab) and check the job configuration. For details about typical module state errors, see “5.6.4 Module States”.

NOTE!
Make sure that the Machine Vision Device is ready to receive the next trigger signal (e.g. avoid over triggering, so that during exposuring the next trigger is already sent). For details about forbidden trigger timings, check the operating instructions of the Machine Vision Device.

NOTE!
For further details about status of Processing Instance via LIMA command see section “9.5 Get Status”.

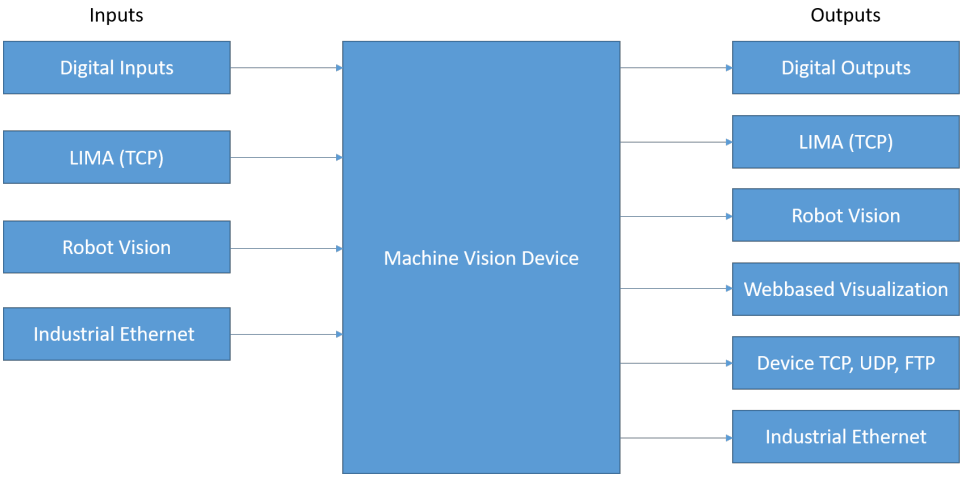
5.7 Interface Overview

Interacting with the Processing Instance is possible via different interfaces that can be used for automated production or for configuration.

NOTE!
Depending on the Machine Vision Device, different interfaces are available.

5.7.1 Interface for Automated Production

The following interfaces are available for an automated production use case.



NOTE!
- For details about the LIMA interface, see “9. LIMA Interface”.
- For details about Robot Vision, see “7.8 Robot Vision”.
- For details about webbased Visualization, see “8. Webbased Visualization”.
- For details about Industrial Ethernet, check the separate interface description for Industrial Ethernet.

5.7.2 Network Interfaces for Processing Instances

Processing Instances have the following network interfaces:

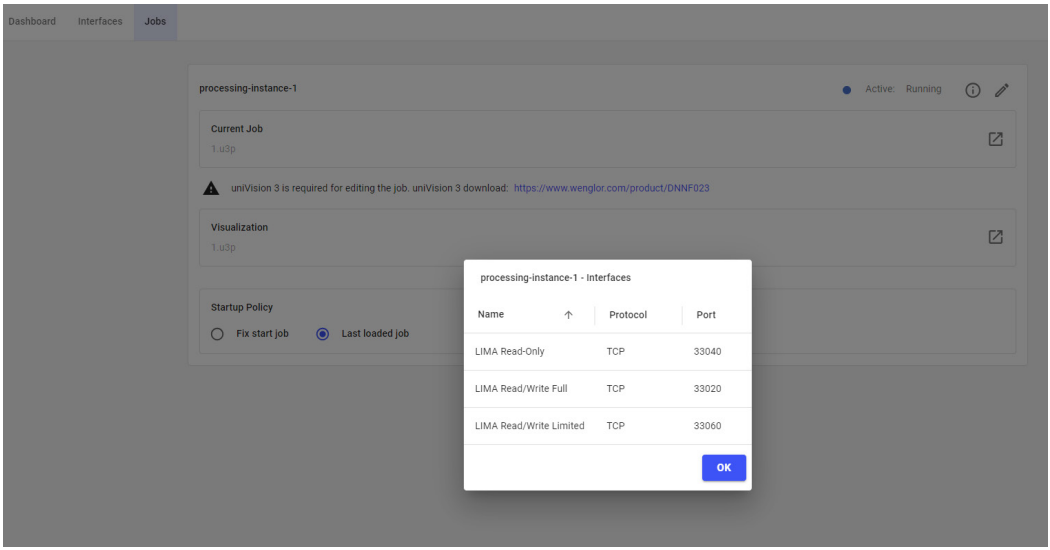
Name	Description	Protocol	Max number of connections	Port
LIMA Read Write Full	Port used by the software wenglor uniVision 3 in edit mode. If not blocked by the software wenglor uniVision 3, it is open for full LIMA Read Write communication from PLC, robot or any third-party software application.  **NOTE!** Loading another job via any other interface (e.g. via webbased visualization) sends automatically a job changed notification on the LIMA Read Write Full connection.	TCP	1	33020
LIMA Read Write Limited	Open port for limited LIMA communication from PLC, robot or any third-party software application. The port is also used by the robot interfaces.	TCP	5	33060
LIMA Read Only	Port used by the software wenglor uniVision 3 in run mode. Open port for read only LIMA communication from PLC, robot or any third-party software application.  **NOTE!** Loading another job (e.g. via webbased visualization) via any other interface sends automatically a job changed notification on all LIMA Read Only connections.	TCP	5	33040
Device website	Web port used by the device website	TCP	No limit	80
Firmware update	Web port used for firmware update	TCP	No limit	8081
Webbased visualization	Web port used by the webbased visualization	TCP	No limit	33080

Name	Description	Protocol	Max number of connections	Port
Device TCP	Port used by Device TCP to send process data to any communication partner (e.g. PLC, robot, third-party software application) **NOTE!** After job changes, it is necessary to close the connection and to reopen it from the communication partner to Device TCP.	TCP	5	34000
Device UDP	Port used by Device UDP to send process data to any communication partner (e.g. PLC, robot, third-party software application)	UDP	No limit	34020
Device Robot Vision	Port used by Device Robot Vision	TCP	5	34040

NOTE!

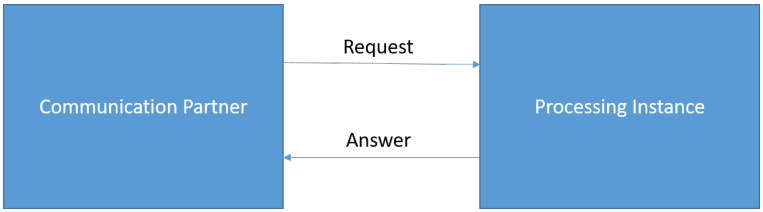
- Ports and maximum number of connections are fix and not adjustable.
- For details about the LIMA interface, see “9. LIMA Interface”.

LIMA ports are also visible on device website at tab jobs. Ports for Device TCP and UDP are visible directly at Device TCP and UDP in the software wenglor uniVision 3.



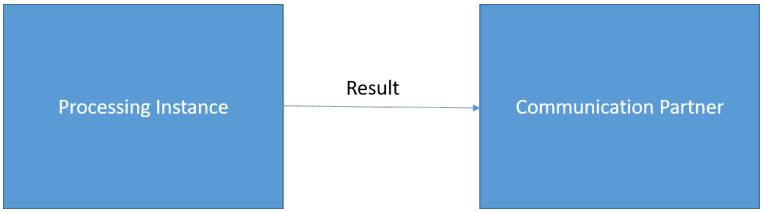
Depending on the port, the communication is based on request and answers or results are automatically sent when available:

- LIMA: Based on request and answer



NOTE!
Loading another job via any other interface (e.g. via webbased visualization) sends automatically a job changed notification on the LIMA Read Write Full connection and on all LIMA Read Only connections.

- Device TCP, UDP and Robot Vision: Results from Processing Instance are automatically sent to the communication partner (one-way communication) without requests.



5.7.3 Network Interfaces for uniVision Simulator

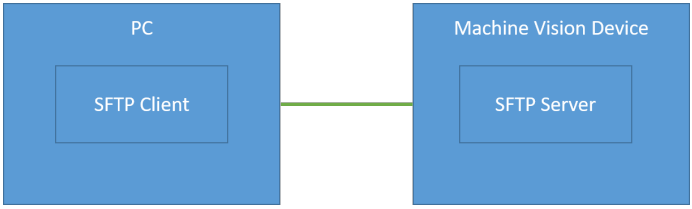
Offline without a real Machine Vision Device, the following network interfaces are available in the uniVision Simulator:

Name	Description	Protocol	Max number of connections	Port
LIMA Read Write Full	Port used by the software wenglor uniVision 3	TCP	1	33003
LIMA Read only	Port used by the software wenglor uniVision 3	TCP	5	33004
Webbased visualization	Web port used by the webbased visualization	HTTP	No limit	33005

NOTE!
Offline ports are limited to the described network ports

5.8 SFTP Server on Machine Vision Devices

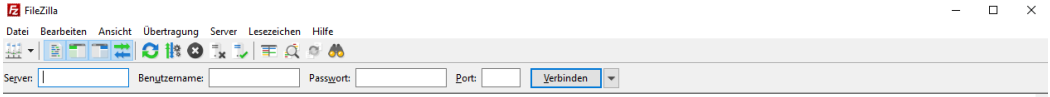
A SFTP Server runs on Machine Vision Devices in order to exchange files (e.g. jobs) via network between PC and Machine Vision Device.



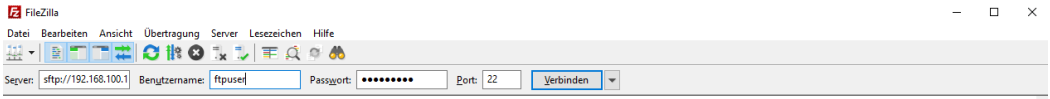
NOTE!
Accessing the SFTP Server on the Machine Vision Devices only works if PC and Machine Vision Device are in the same network (see “5.2.2 Network Basics”).

Creating a connection from the PC to the SFTP Server on the Machine Vision Device.

- Open SFTP client software on PC (e.g. FileZilla Client)



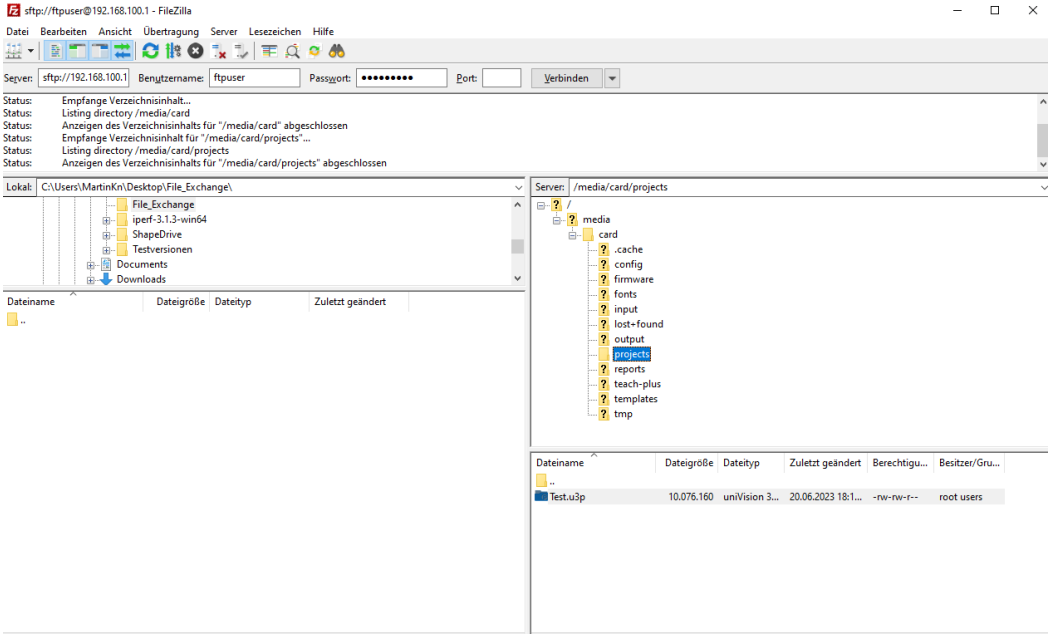
- Enter server address, username, password and port:
  - » Server: Protocol sftp with IP address of the Machine Vision Device, e.g. with default IP address of Machine Vision Device sftp://192.168.100.1
  - » Username: ftpuser
  - » Password: ftpvision
  - » Port: 22



NOTE!

- Editing Username and Password of the SFTP Server on the Machine Vision Device is not possible
- Username and password are also shown on the device website (see operating instruction of Machine Vision Device).

- Click on Connect shows on the Machine Vision Device the folder /media/card/



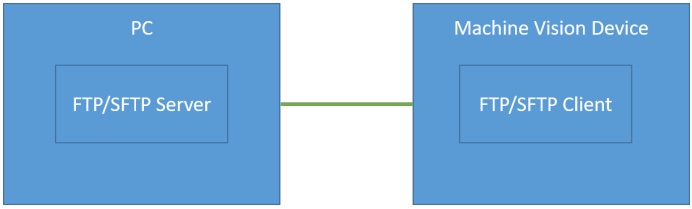
- Select the relevant folder to exchange data between PC and Machine Vision Device.

NOTE!

- Jobs are located in the folder “projects”.
- Teach+ files recorded via LIMA command are located in the folder “teach-plus”.
- Data stored via Device FTP locally on the Machine Vision Device are located in the folder “output”.

5.9 FTP/SFTP Client on Machine Vision Devices

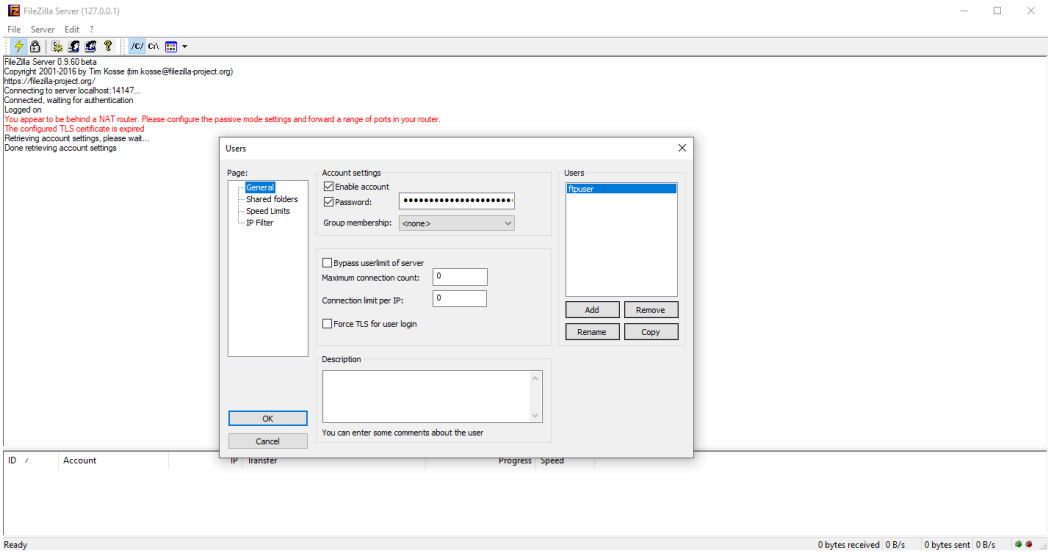
A FTP/SFTP Client runs on Machine Vision Devices to store process data on a FTP/SFTP Server in the network.



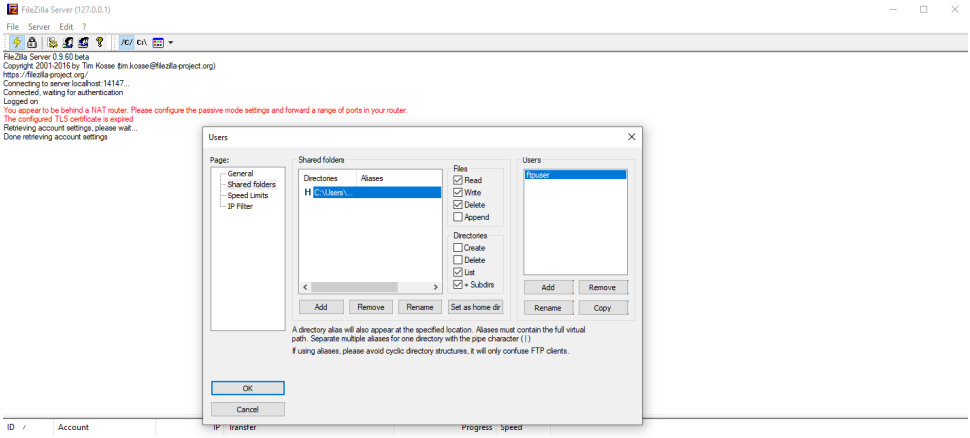
NOTE!
Accessing the FTP/SFTP Server in the network from the FTP/SFTP Client on the Machine Vision Devices only works if PC and Machine Vision Device are in the same network (see “5.2.2 Network Basics”).

Start FTP/SFTP Server in the network (e.g. on PC). E.g. use the software FileZilla Server and configure it accordingly:

- Add user (e.g. ftpuser) with password (optional) at FTP/SFTP server

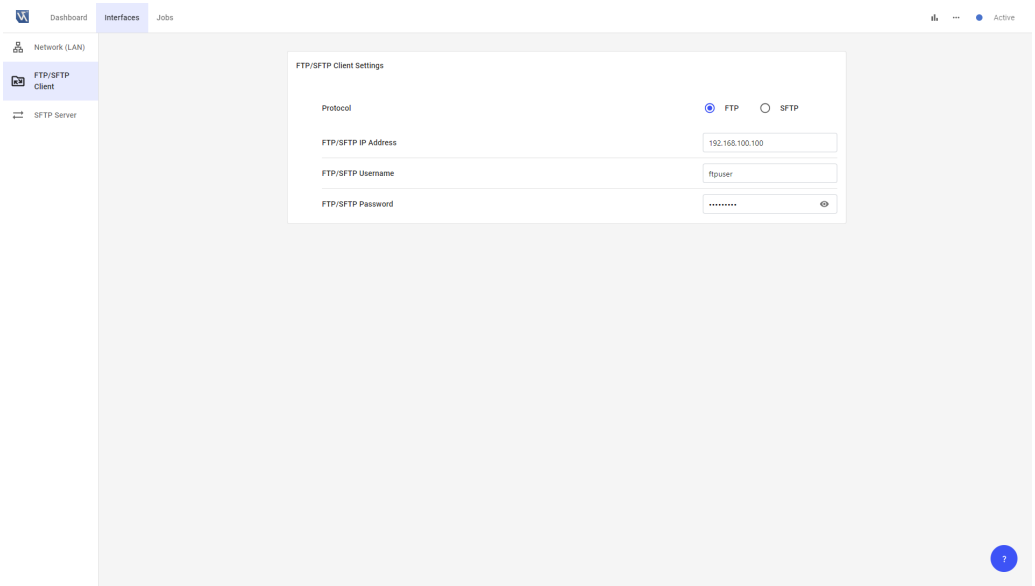


- Select shared folder and provide file read and write access to user



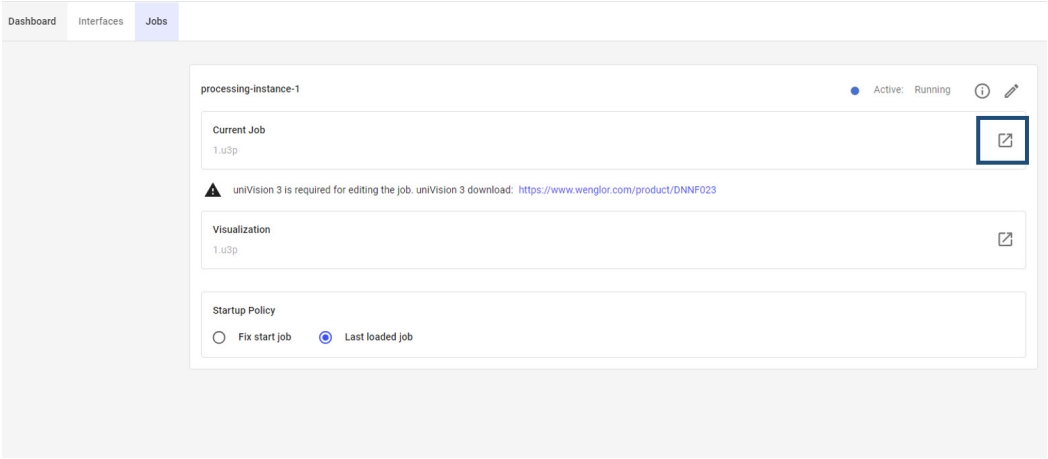
Open tab “Interfaces” and side navigation “FTP/SFTP Client” on the device website of the Machine Vision Device and enter the correct configuration for the FTP/SFTP server in the network:

- Select the protocol: FTP or SFTP (depending on the FTP/SFTP server)
- Enter the IP address of the device where the FTP/SFTP server is running on (e.g. the PC)
- Enter the username and password of the FTP/SFTP server



NOTE!
Check operating instructions of Machine Vision Device for details about the device website.

Open tab “jobs” on the device website, connect with the uniVision software to the device in order to configure the job.



Add Device FTP from the toolbox to the job and configure it accordingly to send process data to the FTP/SFTP server in the network (see “7.7.3 Device FTP”)

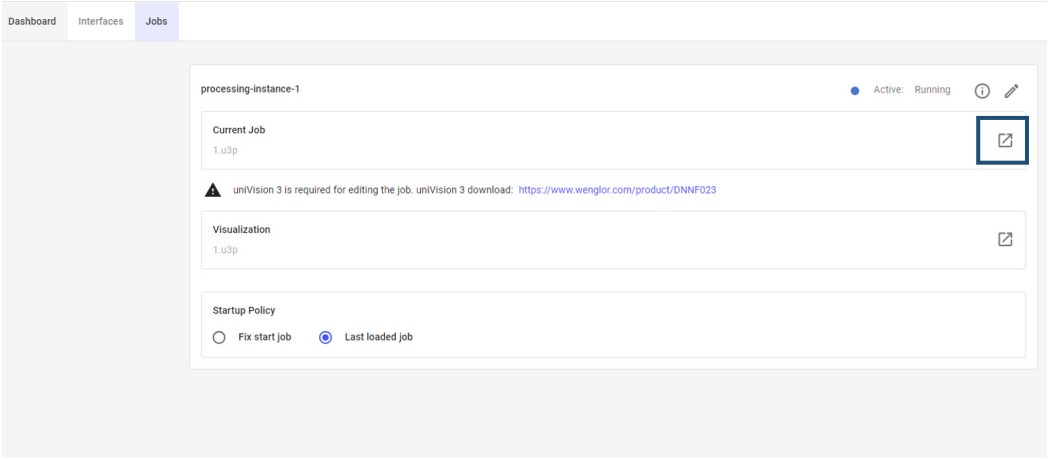
6. Software uniVision 3

The software wenglor uniVision 3 can create a connection to the Processing Instance on a Machine Vision Device or it can open an offline job.

6.1 Connecting to Machine Vision Device

Opening a connection to the Processing Instance is possible via the tab “Jobs” on the device website by clicking on the icon at Current Job

NOTE!
Make sure that the suitable version of the software wenglor uniVision 3 is installed to create the connection (see “4.2 Software Compatibility”).

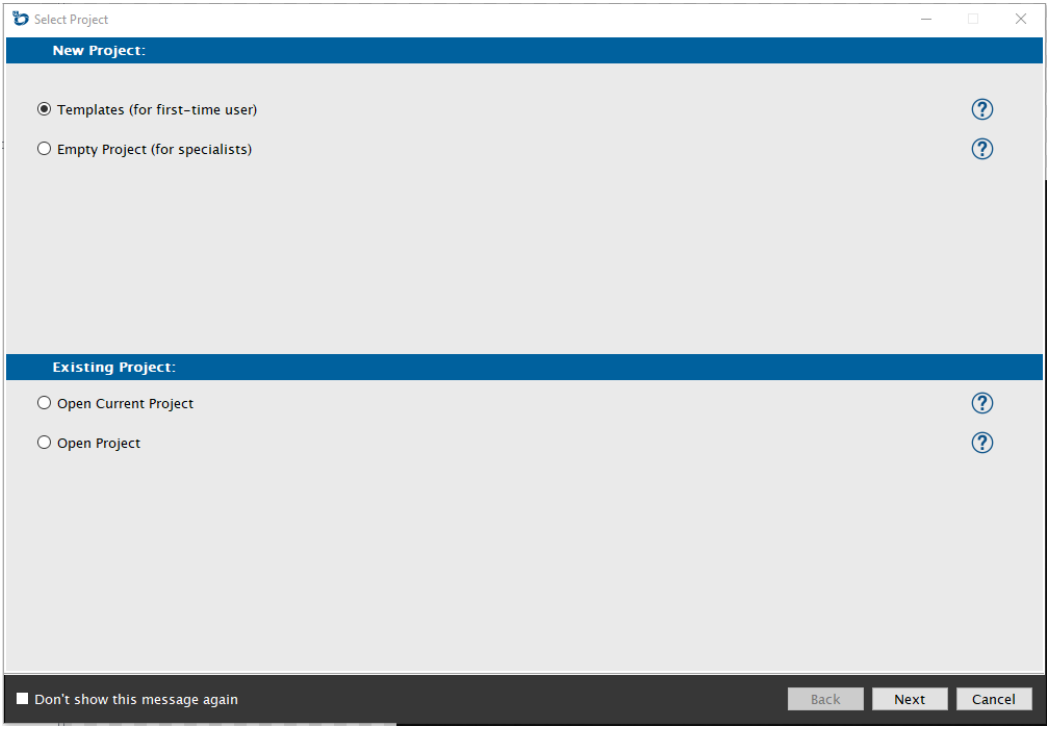


Creating a connection is also possible via shortcuts:

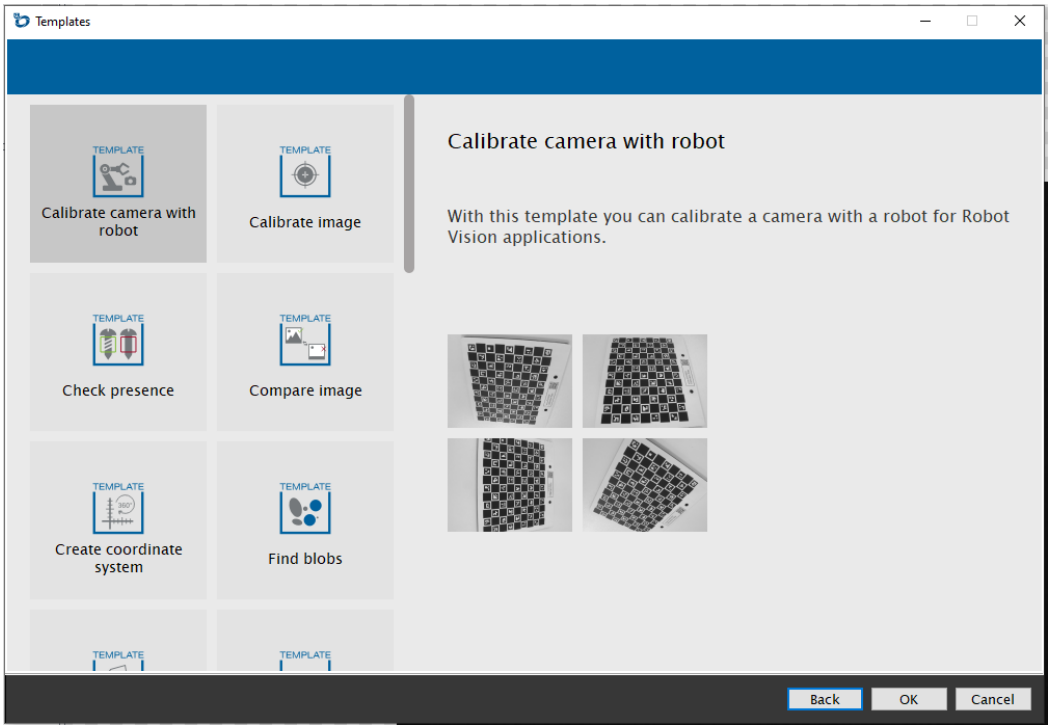
- Create a new shortcut at any location (e.g. on desktop)
- Link to the suitable version of the uniVision3 exe. For wenglor uniVision 3.0.0 e.g. link to “C:\Program Files\wenglor\uniVision3\3.0.0\bin\uniVision3.exe”
- Right click on the shortcut and open the properties.
- Change target and link to the IP address of the Machine Vision Device (e.g. default IP address 192.168.100.1), to the LIMA Read Write Full port (e.g. default rwport 33020) and to the LIMA Read Only port (e.g. default roport 33040): “C:\Program Files\wenglor\uniVision3\3.0.0\bin\uniVision3.exe” –connect –ip 192.168.100.1 –roport 33040 –rwport 33020

After connecting to the Processing Instance, select between the following options:

- Templates (for first-time user): Shows all available templates for the Processing Instance. Templates are predefined jobs for typical applications that contain relevant modules and links between them to reduce the setup time.
- Empty Job: Creates an empty job with only the input device
- Open Current Job: Opens the current job of the Processing Instance
- Open Job: Opens the file manager with the option to select and open one of the existing job files



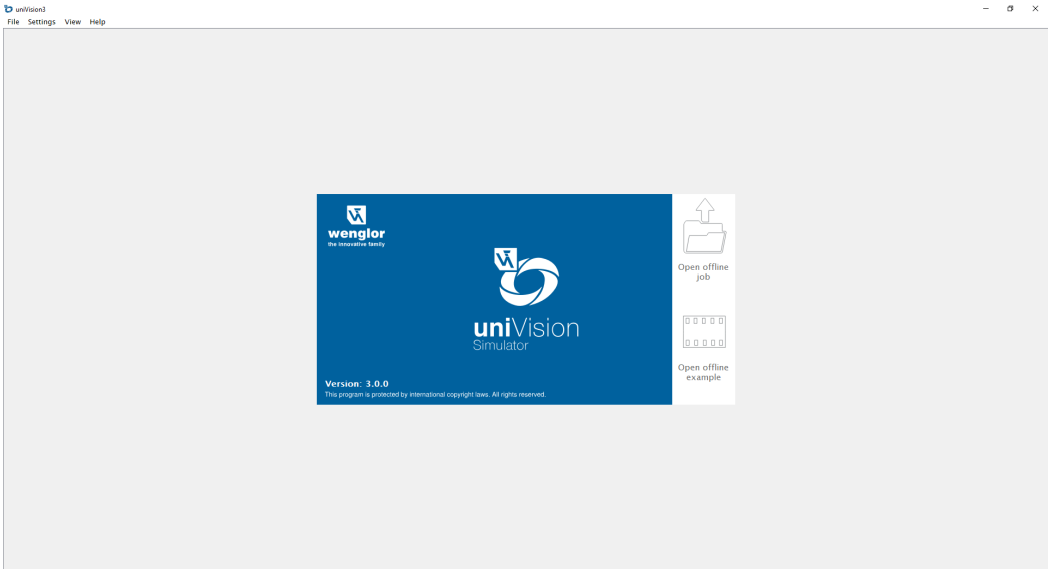
NOTE!
Depending on the Machine Vision Device, different template files are available.



6.2 uniVision Simulator

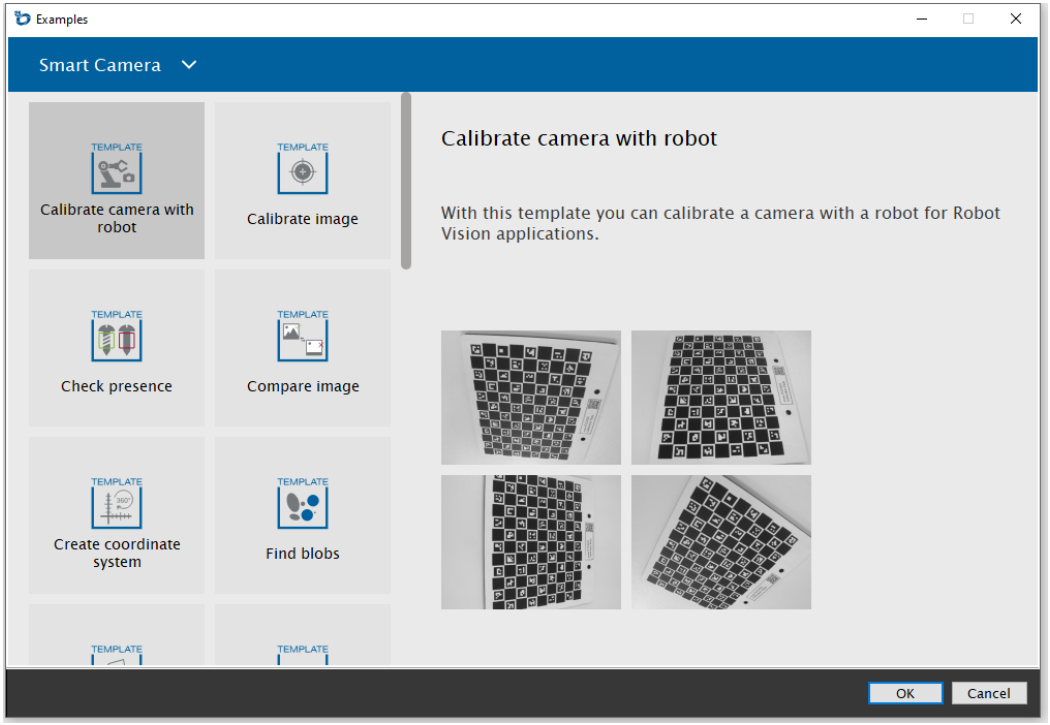
Open the software wenglor uniVision 3 shows the startscreen of the uniVision Simulator with the basic functionalities:

- Open offline job: Opens the file manager to selct one of the uniVision jobs (u3p file). By default on Windows the local job folder C:\ProgramData\wenglor\uniVision3\card\projects is selected
- Open offline example: Opens window to select one of the template job files (u3p file)



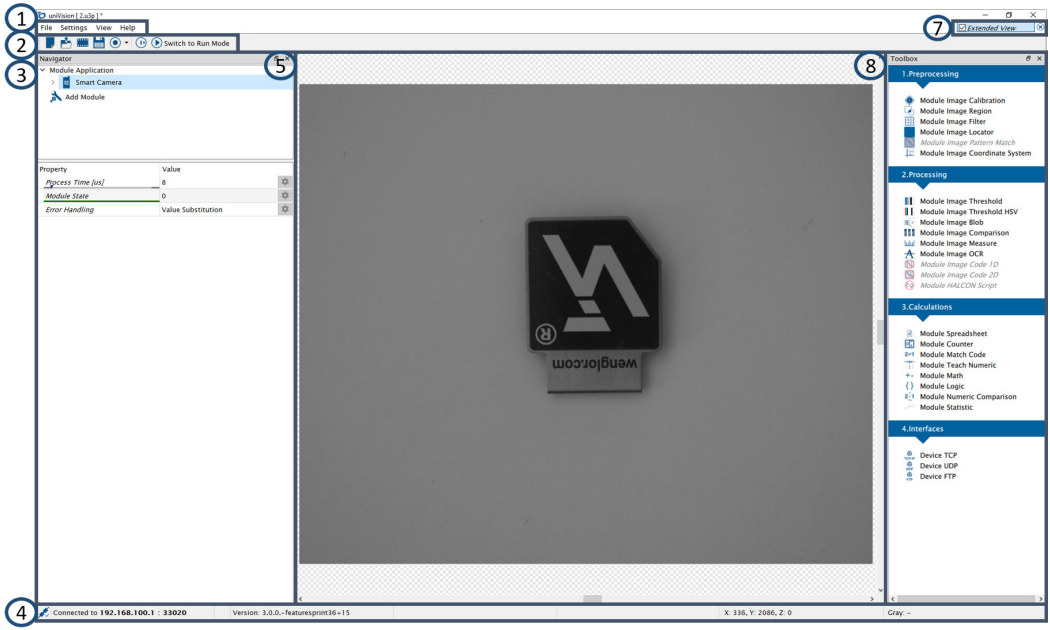
NOTE!
Opening uniVision jobs (*.u3p) works also via double click on the job file.

Clicking on “Open offline example” shows the available templates for all types of Machine Vision Devices.



6.3 Structure

The software wenglor uniVision 3 is divided in several areas.
If connected with a Machine Vision Device, the software wenglor uniVision 3 contains the following elements:



1: Menu bar

2: Icon menu bar

3: Job tree

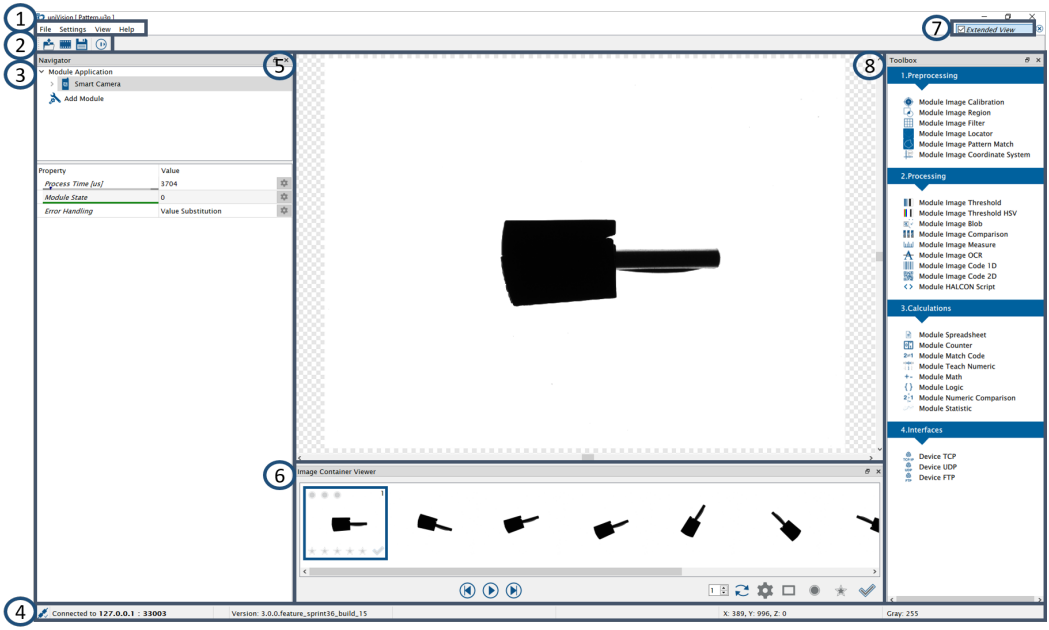
4: Status bar

5: Scene with data (e.g. image)

7: Extended view

8: Toolbox

If offline without a real Machine Vision Device, the software wenglor uniVision 3 contains the following elements:



1: Menu bar

2: Icon menu bar

3: Job tree

4: Status bar

5: Scene with data (e.g. image)

6: Image Container Viewer (only available offline in uniVision Simulator

7: Extended view

8: Toolbox

6.3.1 Menu Bar

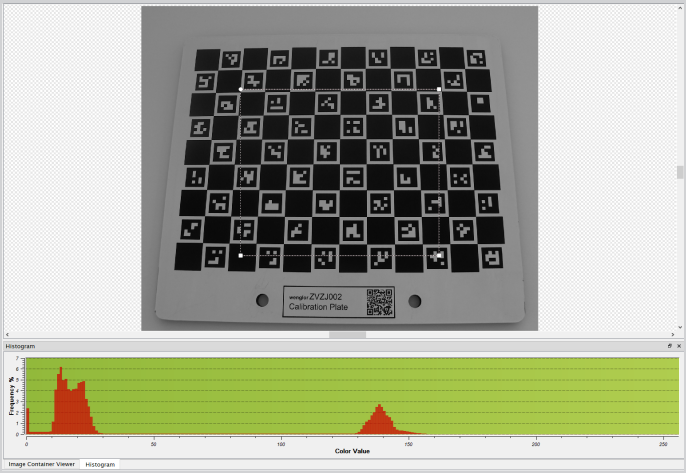
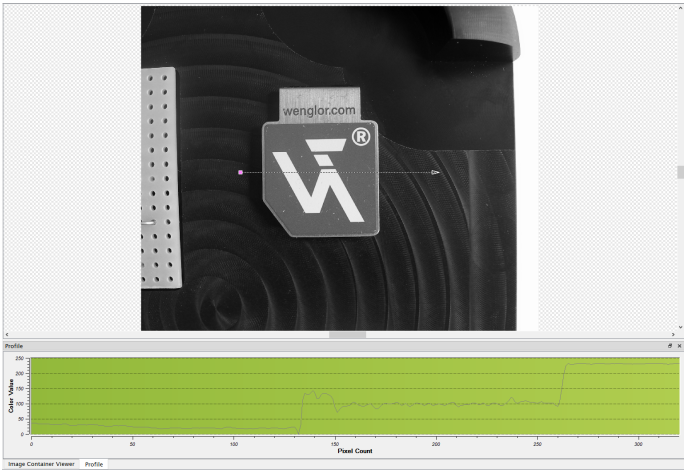
File

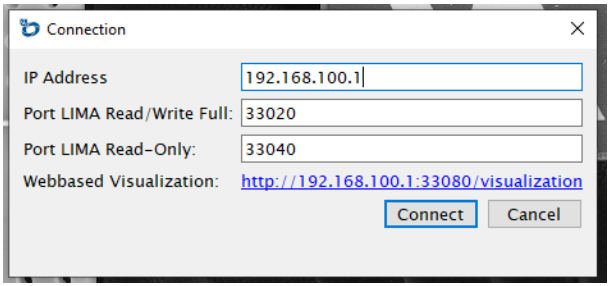
New...	Creates a new empty job at the Processing Instance on the Machine Vision Device (offline in the uniVision Simulator not available).
Open...	Opens file manager to select and open an uniVision job file (*.u3p). **NOTE!** • Default job location on Windows: C:\ProgramData\wenglor\uniVision3\card\projects • Default job location on Machine Vision Device (e.g. B60): /projects (complete path: /media/card/projects)
Templates or Examples	Opens overview of available templates on Machine Vision Device or examples offline for uniVision Simulator.
Save	Saves the current job at the job location.
Save as...	Opens the file manager to select a location where to store the job.
Close Job	Closes the connection to the Processing Instance on the Machine Vision Device or closes the uniVision offline job in the uniVision Simulator. Returns to the startscreen of the software wenglor uniVision 3.
Exit	Closes the software wenglor uniVision 3.

Settings

Options	Opens the options of the software wenglor uniVision 3. **Global** Number of records: Defines the number of records when recording Teach+ files (default: 100). **Visualization:** Inside ROI Area: Defines the color inside of the Region of Interest. Outside ROI Area: Defines the color outside of the Region of Interest.
Language	Defines the user language (DE, EN, FR, IT, ES, PT, NL, HU, TR, ZH, RU).

View

Histogram	Show or hide histogram window (only available at data type image). Select a region in the image to show the frequency distribution of the greyscale values. 
Image Container Viewer	Show or hide the Image Container Viewer window (only available offline in the uniVision Simulator, see “6.3.6 Image Container Viewer”).
Job Tree	Show or hide the Job Tree with the properties of the modules and devices (see “6.3.3 Job Tree”).
Profile	Show or hide profile window (only available at data type image). Move the line at the relevant position in the image to show the greyscale values along the line. 

Toolbox	Show or hide the toolbox with all available modules and devices (see “6.3.8 Toolbox”).
Connect to Device	Show or hide connection window to create a connection to a Processing Instance on a Machine Vision Device by entering IP address (e.g. default 192.168.100.1), LIMA Read Write Full port (e.g. default 33020) and LIMA Read Only port (e.g. default 33040). 
Webbased Visualization	Open webbased visualization of current job (only available offline in the uniVision Simulator, see “8. Webbased Visualization”).
Project Tools	Shows or hides project tools (see “6.3.2 Icon Menu Bar”).
Module Toolbar	Shows or hides module toolbar (see “6.3.2 Icon Menu Bar”). **NOTE!** Depending on the selected module in the navigator, the module toolbar disappears or shows different module tools.

Help

About	Shows the current version of the software wenglor uniVision 3. Provides link to the third-party licenses (see “10. Third-Party Licenses”).
Operating Instructions	Opens pdf file with operating instructions containing the description of the software wenglor uniVision 3.
Release Notes	Opens pdf file with release notes containing the changes between different software versions.
Third Party Licenses	Opens file manager with location of all third party licenses (see “10. Third-Party Licenses”)..
EULA	Opens pdf file with EULA that must be accepted before the software installation (see “4.1 Installation Basics”).
Software Highlights	Opens default browser with software highlights page of the software wenglor uniVision 3 (only available with internet access).
Tutorials	Opens default browser with tutorials page of the software wenglor uniVision 3 (only available with internet access).
Licenses	Opens license window to generate license request files or to load licensed files (see “6.5 Licenses for uniVision Modules”).

Job Conversion	Opens Job Converter to convert uniVision jobs after a feature upgrade (see chapter “6.6 Job Converter”).

6.3.2 Icon Menu Bar

The Icon Menu Bar contains job and module icons.

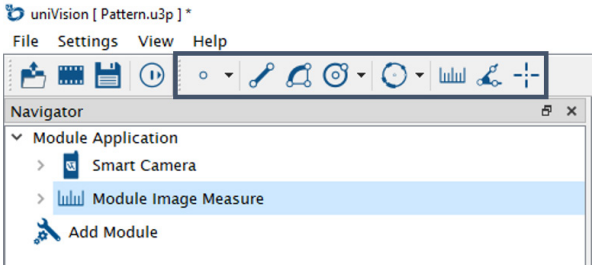
Project Tools

New Job	Creates a new empty job at the Processing Instance on the Machine Vision Device (offline in uniVision Simulator not available).
Open File	Opens the file manager to select a uniVision job file (*.3p). **NOTE!** • Default job location on Windows: C:\ProgramData\wenglor\uniVision3\card\projects • Default job location on Machine Vision Device (e.g. B60): \projects (complete path: /media/card/projects)
Templates or Examples	Opens overview of available templates on Machine Vision Device or examples offline for uniVision Simulator.
Save	Saves the current job at the job location.
Record Teach+	Teach+ recording and downloading is possible (see “6.4 Teach+”). **NOTE!** • Option is only available if connected with Processing Instance on Machine Vision Device (online). • The default number of records is set in the application settings (see “6.3.1 Menu Bar”).
Manual Trigger	In case of software trigger source, press the button to send single trigger signals from the software to the Processing Instance in order to get new results. In case of different trigger sources (e.g. trigger mode off or trigger source line), pressing the manual trigger button simply refreshes and shows the latest available results. **NOTE!** • In the uniVision Simulator clicking on the button moves to the next record of the Image Container Viewer. • Instead of pressing the button it is also possible to press F5 on the keyboard.

Run and Edit Mode	**Edit mode:** • Allows reading and writing of properties (uses LIMA Read Write Full port, see “5.7 Interface Overview”). Maximum one connection is possible via the port. • Software only refreshes if pressing manual trigger button or if selecting another module in the navigator. **Run mode:** • Only allows reading of properties (uses LIMA Read Only port, see “5.7 Interface Overview”). Maximum five connections at the same time are possible via the port. • Software refreshes regularly automatically and lowers the performance of the Machine Vision Device significantly. Do not use the run mode in automated production – instead use the webbased visualization (see “8. Webbased Visualization”).

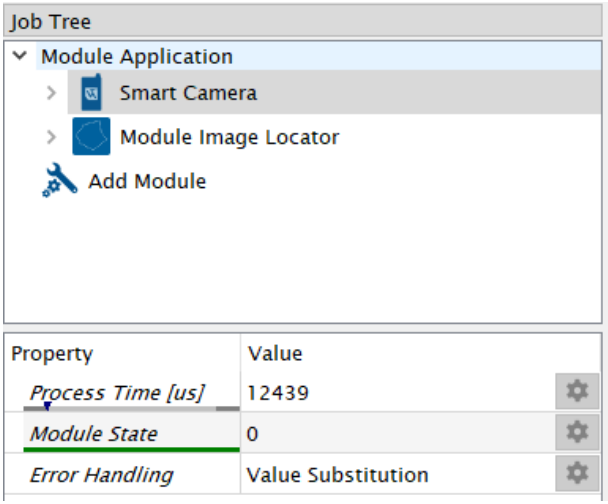
Module Tool Bar

Depending on the selected module in the Navigator different module icons appear in the Module Toolbar. Module Image Measure contains for example the following Module Toolbar.



6.3.3 Job Tree

The Job Tree shows all modules and devices of the current job. Adding modules and devices to the job is possible from the toolbox. Changing the position of the module or device within the job tree is possible via drag and drop. Depending on the selected module in the Navigator, the properties are listed below.



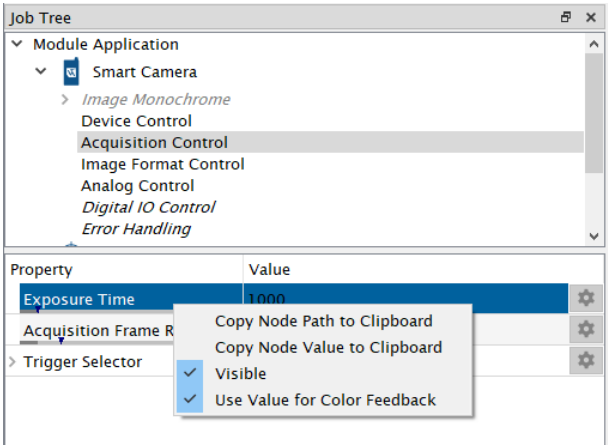
Open context menu on a certain module allows the following possibilities:

Copy Node Path to Clipboard	Copies the node path of the module or sub-module to clipboard in order to address modules or sub-modules in LIMA commands (see “9. LIMA Interface”). The node path for “Module Image Filter” is e.g. “Module Application.Module Image Filter”
Visible	Activate or deactivate visibility of the module or sub-module. Modules with deactivated visibility are hidden if extended view is deactivated. Editing visibility is only possible in extended view (see “6.3.7 Extended View”).
Rename	Rename the module (sub-modules cannot be renamed).
Delete	Delete the module from the current job. **NOTE!** Deleting a module from the current job removes also links in other module that use outputs of such module as inputs.
Copy Module	Copies the module with all of its settings (possible for most of the modules; not supported by all modules and not for devices).

Open context menu on a certain property allows the following possibilities:

Copy Node Path to Clipboard	Copies the node path of the property to clipboard in order to address it in LIMA commands (see “9. LIMA Interface”). The node path can be for example “Module Application.Module Image Measure.Set.Find Line.Quality of Fit [%]”
Copy Node Value to Clipboard	Copies the node value of the property to clipboard (only available for basic data types).
Visible	Activate or deactivate the visibility of properties. Properties with deactivated visibility are hidden if extended view is deactivated. Editing visibility is only possible in extended view (see “6.3.7 Extended View”).

For certain sub-modules and properties it is possible to define via the context menu to “Use Value for Color Feedback”. If the sub-module or property is in error state or if DINT or REAL values are out of tolerance, the value is shown in red color if the option is active. If the option is deactive, the sub-module or value is displayed with black color.



6.3.4 Status Bar

The status bar contains the following info:

- Connected to:
  - » Displays the IP address of the Machine Vision Device or 127.0.0.1 for the uniVision Simulator (if offline).
  - » Displays the port depending on edit mode (default, e.g. 33020) or run mode (default, e.g. 33040).
- Version: Shows the uniVision version of the connected Processing Instance.
- Coordinates X/Y/Z: Shows the coordinates of the mouse position (in pixel and/or mm depending on the module).
- Gray: Shows the gray value (for monochrome images) or the color values (for color images) at the mouse position.

6.3.5 Scene with Data

Shows the data (e.g. image) of the selected module in the Navigator. Zoom in and out via the mousewheel. Press the mousewheel while moving the mouse position to pan the data. The context menu in the scene allows the following possibilities:

Fit geometries	Fits the scene to the geometries within the scene
Fit image	Fits the scene to the image
Fit image width	Fits the scene to the image width
Fit image height	Fits the scene to the image height

NOTE!
The scene is empty if a module without an input image is selected.

6.3.6 Image Container Viewer

The Image Container Viewer (only available offline in the uniVision Simulator) provides all data that were recorded at a Teach+ file. Adding, deleting or saving data is possible. Adding of images with maximum size of 5 MP is supported.
Marking (e.g. statically or via linking with job result) and filtering data is possible.

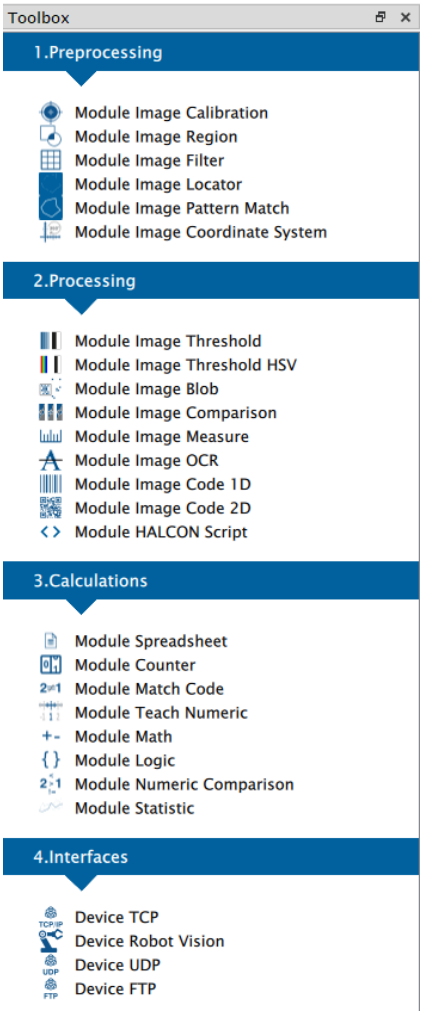
6.3.7 Extended View

Activating the extended view shows all modules and all parameters of modules.
Deactivating the extended view shows only visible modules and visible parameters of modules.

NOTE!
Changing the visibility of modules and parameters is possible in the Navigator via the context menu.

6.3.8 Toolbox

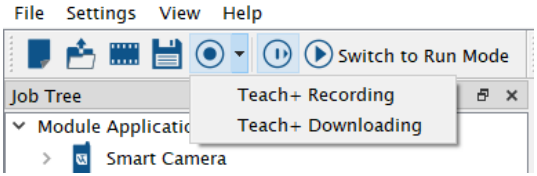
Contains all available modules and devices of the Processing Instance. Add modules or devices to the current job via double click on the module or device or via drag and drop from the toolbox to the Navigator.



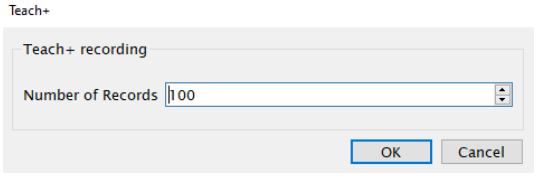
6.4 Teach+

Teach+ Recording:

- Connect to Processing Instance on a Machine Vision Device (online) in order to record a Teach+ (see “6.1 Connecting to Machine Vision Device”).
- Click on “Teach+ Recording”

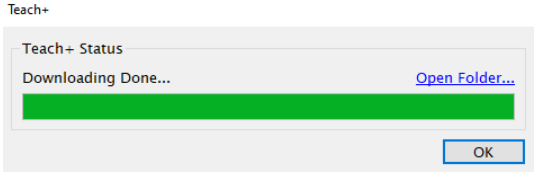


- Enter the “Number of Records” to be stored in the Teach+



- Trigger the device until the number of records is reached. In case of trigger source software, press F5 on the keyboard to create single software trigger signals. Pressing on Cancel button, saves the Teach+ file with the records that were captured so far.

- “Open Folder” shows the location of the local job folder. On Windows PC, Teach+ files are stored in C:\ProgramData\wenglor\uniVision3\card\projects.



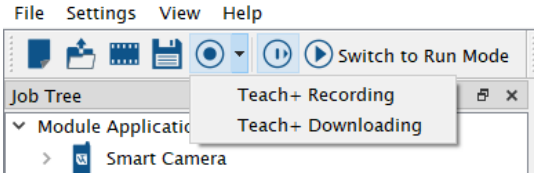
NOTE!

- Editing Teach+ files is possible offline in the uniVision Simulator.
- Loading changed Teach+ files again on the Machine Vision Device works by connecting to it, clicking on “Open” and selecting the local job folder.

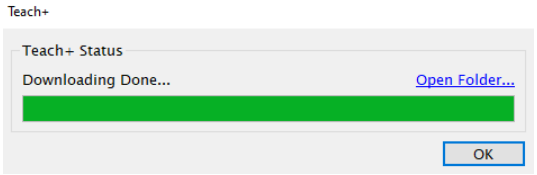
Teach+ Downloading:

Download the latest Teach+ file from the folder /teach-plus (complete path /media/card/teach-plus) on the Machine Vision Device to the local job folder on the PC. It enables easy copying of Teach+ files created via LIMA command (see “9. LIMA Interface”) from the Machine Vision Device to the PC.

- Record Teach+ via LIMA command. Only the latest Teach+ file is stored directly on the Machine Vision Device in the folder /teach-plus.
- Use the software wenglor uniVision 3 and connect to the Processing Instance on the Machine Vision Device (see “6.1 Connecting to Machine Vision Device”).
- Click on Teach+ Downloading to copy the Teach+ file from the Machine Vision Device to your local PC.



- “Open Folder” shows the location of the local job folder. On Windows PC Teach+ files are strored in C:\ProgramData\wenglor\uniVision3\card\projects.



6.5 Licenses for uniVision Modules

Depending on the software package of the Machine Vision Device, the licenses for certain modules and devices are activated by default.

6.5.1 Smart Camera B60

The following overview shows the licensed modules and devices for the different software packages of the Smart Camera B60:

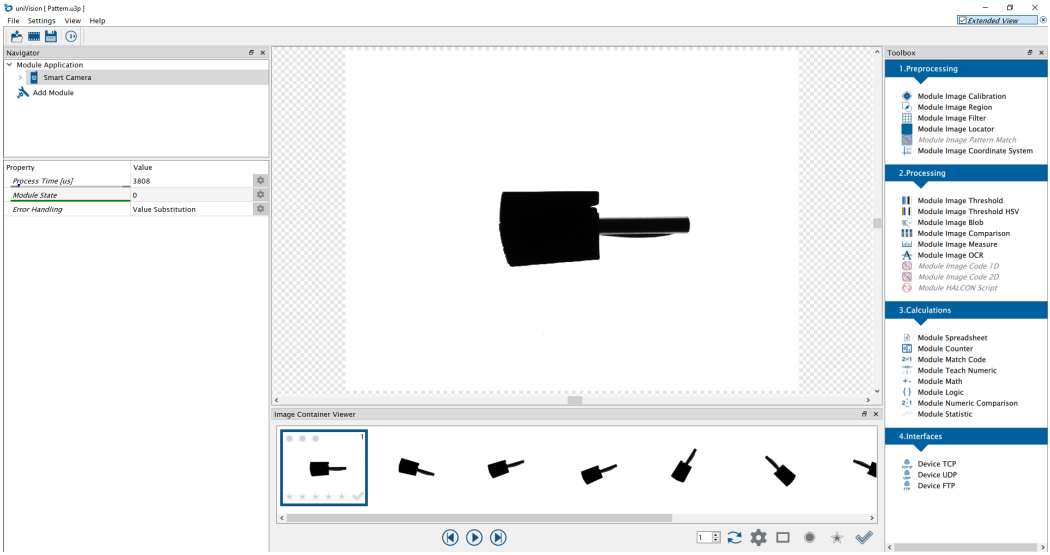
Module or Device	uniVision	uniVision Extended	uniVision Script
Device Smart Camera	licensed	licensed	licensed
Module Image Calibration	licensed	licensed	licensed
Module Image Region	licensed	licensed	licensed
Module Image Filter	licensed	licensed	licensed
Module Image Locator	licensed	licensed	licensed
Module Image Pattern Match	unlicensed	licensed	licensed
Module Image Coordinate System	licensed	licensed	licensed
Module Image Threshold	licensed	licensed	licensed
Module Image Threshold HSV	licensed	licensed	licensed
Module Image Blob	licensed	licensed	licensed
Module Image Comparison	licensed	licensed	licensed
Module Image Measure	licensed	licensed	licensed
Module Image OCR	licensed	licensed	licensed
Module Image Code 1D	unlicensed	licensed	licensed
Module Image Code 2D	unlicensed	licensed	licensed
Module HALCON Script	unlicensed	unlicensed	licensed
Module Spreadsheet	licensed	licensed	licensed
Module Counter	licensed	licensed	licensed
Module Match Code	licensed	licensed	licensed
Module Teach Numeric	licensed	licensed	licensed
Module Math	licensed	licensed	licensed
Module Logic	licensed	licensed	licensed
Module Numeric Comparison	licensed	licensed	licensed
Module Statistic	licensed	licensed	licensed
Device TCP	licensed	licensed	licensed
Device UDP	licensed	licensed	licensed
Device FTP	licensed	licensed	licensed
Device Robot Vision	unlicensed	unlicensed	unlicensed

The following overview shows the license upgrades for the Smart Camera B60:

Article Number	Name of license upgrade	Modules
DNNL018	License B60 uniVision Extended	Module Image Code 1D Module Image Code 2D Module Image Pattern Match
DNNL019	License B60 uniVision Script	Module Image Code 1D Module Image Code 2D Module Image Pattern Match Module HALCON Script
DNNL020	License B60 uniVision Extended to uniVision Script	Module HALCON Script
DNNL021	License B60 uniVision Robot Vision	Device Robot Vision

6.5.2 uniVision Simulator

Offline without a real Machine Vision Device all modules and devices habe a valid license for testing and fine-tuning of the job settings via Teach+ except modules based on the external third-party image library HALCON.



Order the license upgrade DNNL022 in order to use modules based on the external third-party image library HALCON offline in the uniVision Simulator.

Article Number	Name of license upgrade
DNNL022	License Windows Simulator uniVision

NOTE!

- The licensed files for the offline Windows Simulator work only for the specific Windows PC for which the licenses were requested.
- In order to use Module HALCON Script offline, a valid HDevelop license from the company MVTec is additionally necessary.

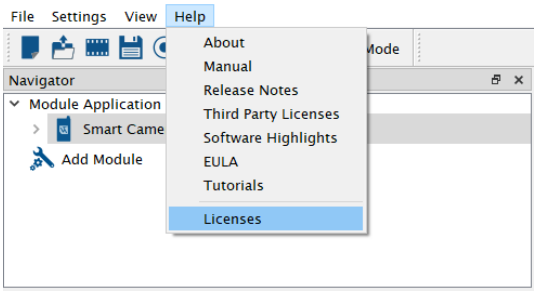
6.5.3 Requesting License Upgrades

Requesting license upgrades is different for Machine Vision Devices (e.g. Smart Camera B60) and the offline Windows Simulator.

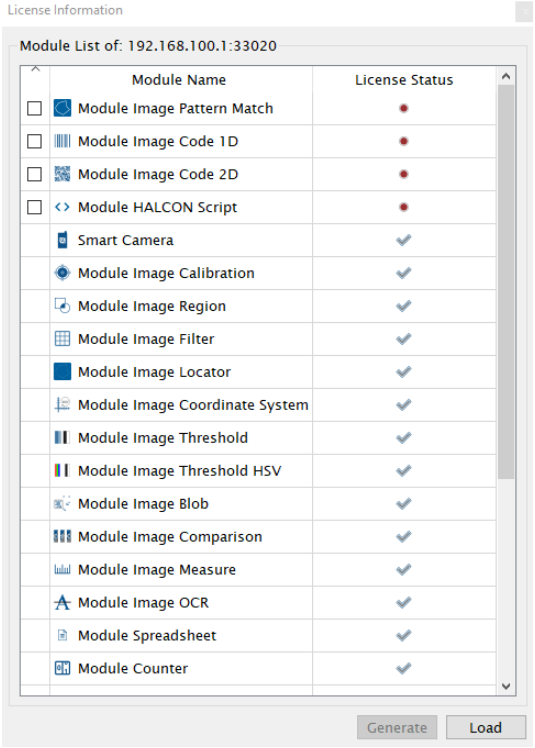
- Machine Vision Devices: Connect the software wenglor uniVison 3 to the Processing Instance on the Machine Vision Device (see “6.1 Connecting to Machine Vision Device”).
- Offline Windows Simulator: Open an offline job or example with the software wenglor uniVision 3 (see “6.2 uniVision Simulator”).

Generating license request files:

- Click in the Menu Bar on Help → Licenses

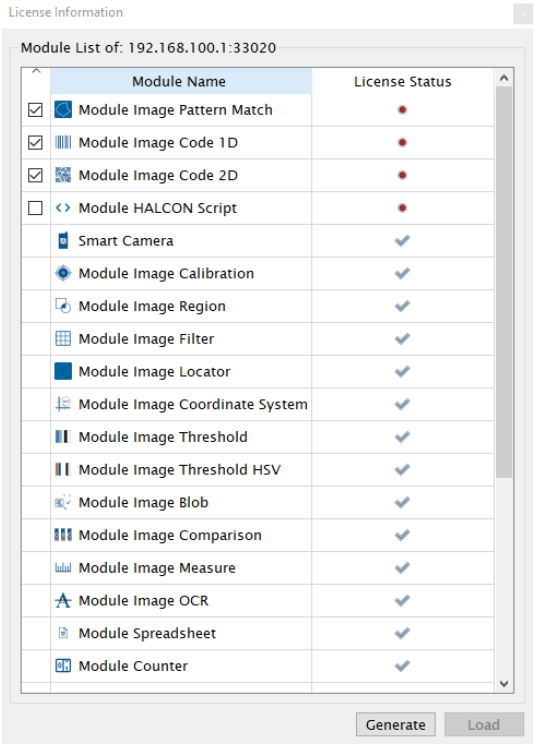


- The overview shows the license status for every module and device.

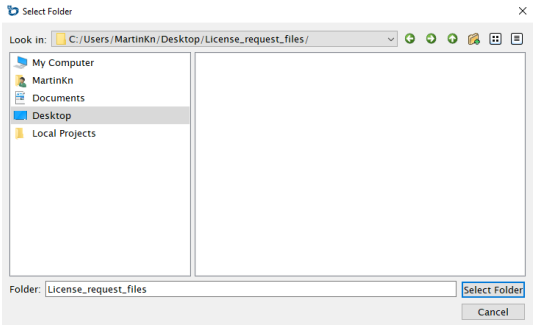


- Select all modules included in the relevant license upgrade and click on Generate.

NOTE!
Check the license upgrade possibilities for the Machine Vision Device.



- Select the folder location where to store the license request files (*.u3k) and click on Select Folder



- Send the license request files (*.u3k) together with the order via email to the local wenglor customer service (order@wenglor.com)

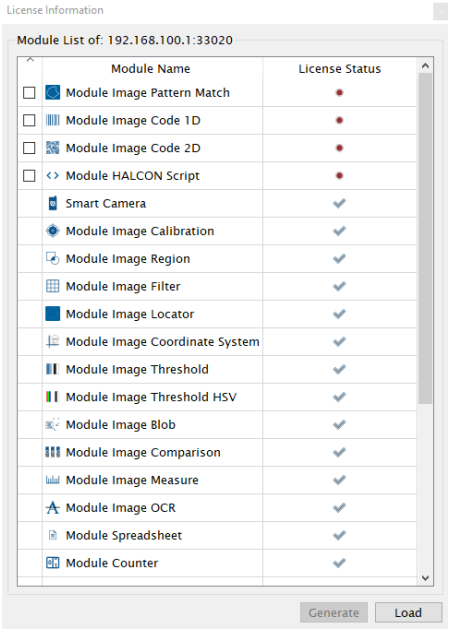
NOTE!
Processing the order and generating the licensed files takes approx. 1-3 working days. The licensed files are provided via email.

Activating license upgrade is different for Machine Vision Devices (e.g. Smart Camera B60) and the offline Windows Simulator.

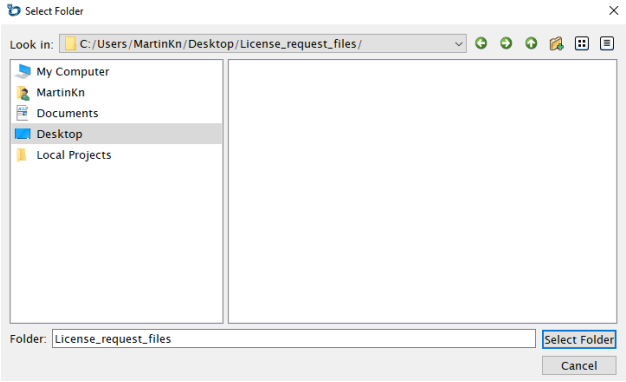
- Machine Vision Devices: Connect the software wenglor uniVison 3 to the Processing Instance on the Machine Vision Device (see “6.1 Connecting to Machine Vision Device”).
- Offline Windows Simulator: Open an offline job or example with the software wenglor uniVision 3 (see “6.2 uniVision Simulator”).

Activating license upgrades:

- Click in the Menu Bar on Help -> Licenses
- Click on Load. Make sure that no module is selected.



- Select the licensed files (.u3l) and click on Open



- Close the licensed window
- The relevant modules and devices are shown licensed in the Toolbox

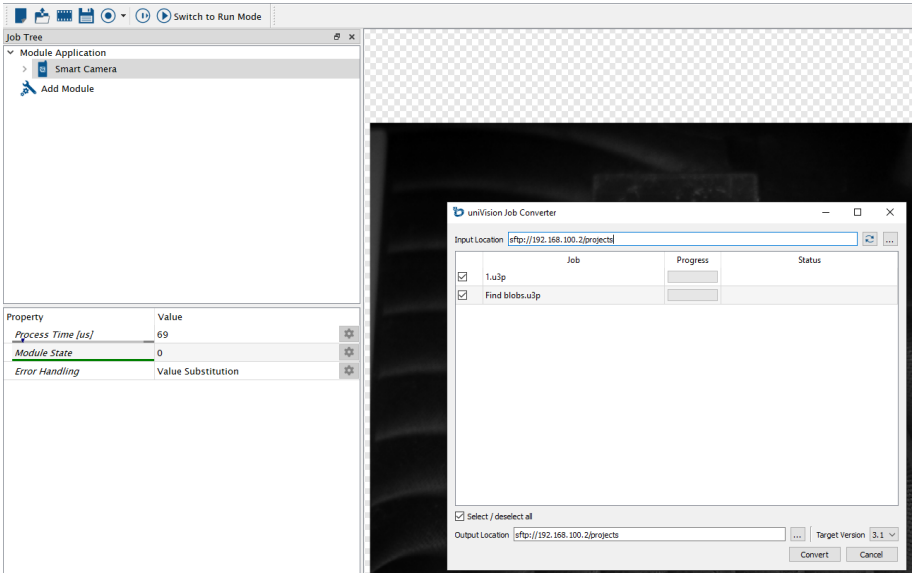
NOTE!
Only if one of the just licensed modules or devices was already part of the current uniVision job, reboot the Machine Vision Device in order to use the module or device.

6.6 Job Converter

If using existing jobs after a feature update (change of the second digit in the version number), job conversion is required.

Convert jobs:

- Start the Job Converter (Menu Bar "Help" -> "Job Conversion")
- Set path for Input Location
- Select jobs to be converted
- Set path for Output Location
- Define the Target Version for the jobs
- Click on Convert to start the conversion



NOTE!

- Original jobs are stored in backup folder. Converted jobs are stored in the Output Location.
- Job conversion is only possible after a feature update, not after a feature downgrade.
- For details about compatability, see section “4.2 Software Compatibility”.

7. Modules

Adding modules from the Toolbox to the Job Tree list the following properties depending on the selected module.

7.1 Module Application

Process Time [µs]	Process time to run the complete job tree. Starts when reading input data and ends when all results are sent. The process time does not include the image acquisition as it is running independently. Not linkable in other modules or devices as the value is only available after all modules and devices have been executed.
Module State	Shows aggregated error messages for all modules and devices (see “5.6.4 Module States”). Not linkable in other modules or devices as the value is only available after all modules and devices have been executed.
Process Time Last Run [µs]	Process time for the last data evaluation of the complete job tree. Linkable in other modules.
Module State Last Run	Shows aggregated error messages for all modules and devices of the last data evaluation (see “5.6.4 Module States”).
Run Counter	Increases by one with every data evaluation. Starts with value 1 for the first data evaluation after booting or job loading. After the maximum value of 2 147 483 647 is reached, the Run Counter starts again at 1 for the next data evaluation.
Free Memory [kB]	Shows the free memory of the device in kB.
Filename	Shows the current name of the job.
Project Version	Shows the project version (see “4.2 Software Compatibility”).
Toggle Bit	Changes status with every data evaluation. Link Toggle Bit e.g. to interfaces (e.g. digital outputs) to find out if new results are available. Starts with true status for the first data evaluation after booting or job loading.
Author	Enter author info of the job.
Info	Enter additional job info.

7.2 Device Smart Camera for B60

Process Time [µs]	Process time to run the module in µs. **NOTE!** The time does not include the exposure time, the readout time of the image chip and the internal transfer time – it only considers reading the image from the internal buffer.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States"").
Error handling	If a result linked to a digital output (at Digital IO Control) is in error state, the behavior for the digital output is to substitute the value with the BOOL substitution value defined at the sub-module "Error Handling".
Create HSV Image	Activate (default) or deactivate the HSV image (only for color images). The HSV image is needed as input in Module Image Threshold HSV.
Create RGB Image	Activate or deactivate (default) the RGB image (only for color images). The RGB image is e.g. needed for color images at Module HALCON Script.
Create BGRA Image	Activate (default) or deactivate the BGRA image (only for color images). The BGRA image is e.g. needed for robot vision applications or to store color images via Device FTP.

Sub-Module "Image Monochrome", "Image BGRA", "Image HSV" and "Image RGB"

Shows the image of the selected image channel in the scene. For RGB images, the single channels red, blue and green, and for HSV, the single channels hue, saturation and value, are available.

NOTE!

- Monochrome cameras provide only grey images.
- Color cameras provide by default HSV and BGRA images. It is possible to activate also the RGB image and to deactivate not used images in order to reduce the process time. At least one image channel must remain activated for color images.

Sub-Module "Device Control"

Device Type	Shows "Smart Camera"
Device Vendor Name	Shows "wenglor sensoric GmbH"
Device Model Name	Shows article number
Device Version	Shows hardware version
Device Firmware Version	Shows firmware version
Device Serial Number	Shows serial number

Sub-Module "Acquisition Control"

Exposure Time	Defines the exposure time for the image chip in µs (default: 1.000 µs; minimum: 50 µs; maximum: 500.000 µs)
Acquisition Frame Rate	Defines the acquisition frequency in Hz. Only visible if Trigger Mode of Exposure Start Trigger is set to Off (default: 5 Hz; minimum 1 Hz; maximum 40 Hz). **NOTE!** Recommended maximum frequencies (e.g. to avoid data overflow) are: • 35 Hz for B60 1.6 MP monochrome • 30 Hz for B60 1.6 MP color • 30 Hz for B60 5 MP monochrome • 20 Hz for B60 5 MP color
Trigger Selector	Trigger Selector “Exposure Start”: • Trigger Mode: If the Exposure Start trigger is set to Off, the Smart Camera captures images with the defined Acquisition Frame Rate. If the Exposure Start trigger is set to On, the Trigger Source can be defined accordingly. • Trigger Source: If Trigger Mode of Exposure Start trigger is set to On, the Trigger Source can be set to Software (e.g. to trigger via LIMA command or via Software wenglor uniVision 3) or to one of the lines (IOs). • Trigger Activation: Set fix to Rising Edge. Only visible if Trigger Mode of Exposure Start trigger is set to On and Trigger Source is set to one of the lines.

NOTE!

- Please consider that trigger signals while exposuring and while reading out the image chip must be prevented as they are ignored by the software. For details about trigger prohibited timings, check the operating instruction of the Machine Vision Device.
- Make sure that the complete process time of the job is fast enough for the trigger frequency. If image acquisition is faster than image evaluation, data is lost which is signaled via Module State of Device Smart Camera and via Data Overflow at status of processing instance (see section “5.6.5 Status of Processing Instance”). In case of data overflow, the error handling of devices takes care of providing lost results to the communication partner (see section “5.6.3 Status of Outputs and Error Handling”).

Sub-Module "Image Format Control"

Pixel Format	Shows pixel format: • Mono 8 for monochrome images • BGR 8 for color images
Sensor Width	Shows the current sensor width in pixel
Sensor Height	Shows the current sensor height in pixel
Invert X	Inverts x of image if activated.
Invert Y	Inverts y of image if activated.
Region Selector	Shows Region Selector “Region 0”: • Region Mode is set to Off (read-only; not used)

Sub-Module "Optic Control"

Only available at B60 Smart Cameras with Autofocus module.

Auto Focus	If set to „Once“, the auto focus is done once based on the auto focus region. It uses the image sharpness to find in an iterative approach the optimal focus position. After finishing the auto focus, the parameter is automatically set back to Off and the image should be sharp. **NOTE!** Applying the Auto Focus Once takes some time as the algorithm captures and evaluates several images with different focus positions in order to find the most suitable focus position.
Focus Position [mm]	Defines the Focus Position in mm (distance from front of protective tube to object). After setting Auto Focus to “Once”, the Focus Position updates to the new distance value. If needed, adjust the focus position value manually.
Auto Focus Region	The auto focus region defines the area where the image should be sharp after the auto focus: • Width: Defines the width of the auto focus region. • Height: Defines the height of the auto focus region. • Offset X: Defines the x offset of the auto focus region. • Offset Y: Defines the y offset of the auto focus region.

Sub-Module "Analog Control"

Gain Selector	Shows Gain Selector “Analog All”: • Shows the gain value (read-only; not used)

Sub-Module "Digital IO Control"

For each line (selected via Line Selector) the following properties appear:

Line Mode	Defines the selected Line as Input or Output.
Line Inverter	Defines if the selected Line is inverted or not.
Line Format	Defines if the output is PNP, NPN or Push Pull.
Line Source	For Outputs the Line Source can be set to Exposure Active (for flashing external illuminations), to Processing Active (active during data processing to check for new results after processing has been finished) or to User Output (to link job result at User Output Value).
User Output Value	Link any job result to User Output Value if the Line Source of the Output is set to User Output.

NOTE!

- By default, all lines are set to:
  - » Line Mode: Output
  - » Line Inverter: False
  - » Line Format: PNP
  - » Line Source: User Output
  - » User Output Value: False
- If using a line as Exposure Active, make sure to adjust the following settings so that wenglor external illuminations work correct:
  - » Line Mode: Output
  - » Line Inverter: True
  - » Line Format: Push Pull
  - » Line Source: Exposure Active
- Setting Trigger Source of Exposure Start trigger to one of the lines, automatically changes the Line Mode of such line to Input.

Sub-Module "Light Control"

Only available if using an internal illumination module at B60 Smart Cameras with Autofocus.

Internal Light Mode	Defines if internal illumination module is set to On or Off.
Internal Light Pattern	Defines if internal illumination module is set to Continuous Light or to Strobe Light. If possible, use Strobe Light to activate the illumination only when relevant. Only available if Internal Light Mode is set to On.
Light Brightness	Defines the brightness of the illumination module in percent. Only available if Internal Light Mode is set to On at white or infrared illumination modules.
White, Red, Green, Blue Brightness	Defines the brightness of the single channels for the RGBW illumination module in percent. Only available if Internal Light Mode is set to On.

Sub-Module "Test Object Status Control"

Only available if using an internal illumination module at B60 Smart Cameras with Autofocus.

For the green (OK) and red (NOK) LEDs on the illumination module the following settings appear.

Test Object Status Selector	Defines if settings are for red or green LED.
Test Object Status Value	Link any job result to Test Object Status Value. LED is active if linked job result is set to TRUE.
Test Object Status Inverter	Defines if logic is inverted or not.

Sub-Module "Chunk Data Control"

Meta data are available together with the image as chunk data. Activating them is possible individually for each chunk data.

Chunk Selector	Select one of the following chunk data: • Frame ID: Returns the frame ID provided by the image chip. Not relevant in real applications (instead use run counter in Module Application). After booting, the device records several images internally in order to get stabile brightness values. Consequently, the Frame ID value does not start with one after booting. • Position X: Returns the x value of the position sensor. • Position Y: Returns the y value of the position sensor. • Line 1-6 Status: Returns the status of the digital IO line. Link Line Status in further modules in order to teach or reset values (e.g. counter value of Module Counter) when digital input is set to TRUE during data evaluation. • Missed Trigger Counter: Returns the number of missed triggers. Missed triggers are trigger signals sent during trigger prohibited times (exposure time + readout time). Value starts automatically at zero, after job loading. If Missed Trigger Counter is bigger than zero, check the trigger settings and the trigger frequency. **NOTE!** For details about missed trigger counter and position sensor, check the operating instructions of the Machine Vision Device.
Chunk Enable	Defines if selected chunk data is enabled or not.

Sub-Module "Error Handling"

Substitute BOOL Types by	Applies substitution value false or true if the linked job result for any digital output is in error state.

7.3 Image-Based Preprocessing Modules

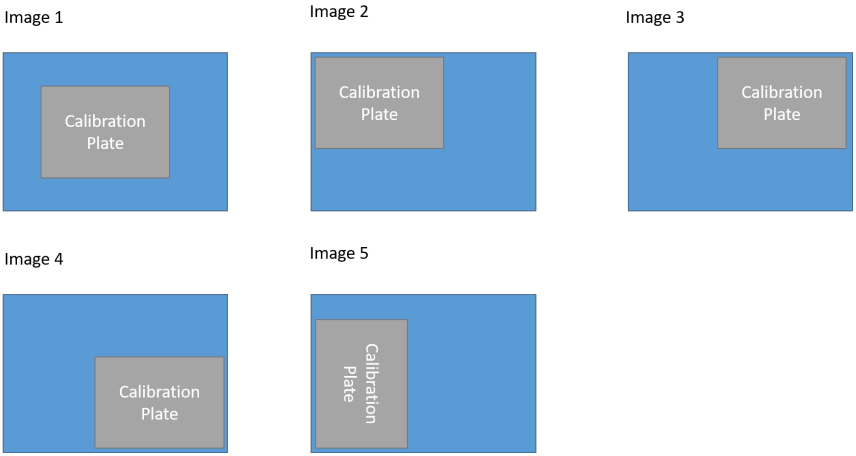
7.3.1 Module Image Calibration

Use Module Image Calibration to calibrate the camera image and to get mm values out of pixel values. The following two calibration modes exist:

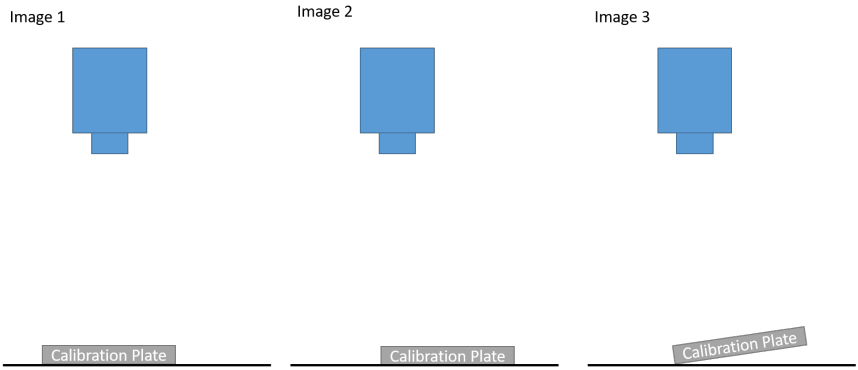
- Linear calibration: Put an object with a known length in the field of view (e.g. ruler) to calculate a pixel per mm factor. The mode does not take into account lens distortion creating less accurate results (especially in the image corners). Furthermore, the measurement plane must be on the same height level as the calibration (Z offset is not supported).
- Charuco calibration: Put a charuco calibration plate in the field of view to calibrate and to eliminate the lens distortion. Charuco calibration plates printed on stiff materials are available on the wenglor website (see www.wenglor.com → ZVZJ).

Charuco calibration

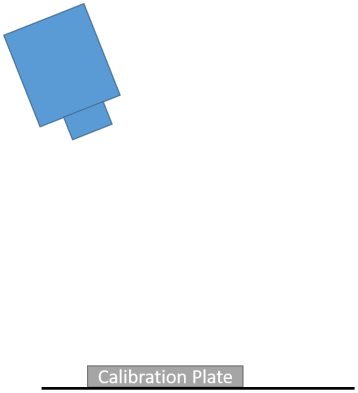
- It is recommended to order the wenglor calibration plates ZVZJ for accurate results as they are printed on stiff and temperature resistant materials.
- If using PDF files, make sure to print the charuco pattern on a stiff material with a high accurate printer. Print PDFs in real size and check the real dimensions after printing.
- Check “Reprojection Error Calibration” after calibration. Typical results with the wenglor calibration plates are about 0.1. Printing on less stiff materials results in bigger (worse) values for the “Reprojection Error Calibration”.
- Select a suitable size of the calibration plate. It should cover as much as possible of the camera image (at least half of the camera image) and should be visible completely if possible for best accurate results.
- Capture several images in different positions and orientations of the calibration plate to cover with the calibration plate all areas of the camera image (especially the image corners) in order to increase the accuracy.



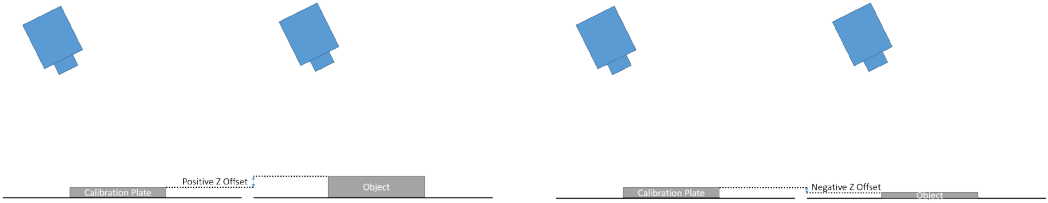
- It is possible to tilt the calibration plate differently for the calibration images (keep the distance from the camera to the calibration plate as constant as possible).



- One of the calibration images is used as reference calibration image and defines the measurement plane. The camera does not have to look perpendicular on the reference calibration plate. Slight tilting is supported (e.g. to avoid reflections).

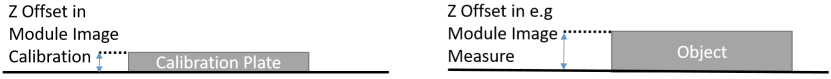


- The measurement plane (object) must be on the same level or parallel to the plane of the reference calibration image (Tilting the measurement plane to the reference calibration plane is not supported). If measuring on a different plane, use the z offset. The z offset should be as small as possible for best accurate results. If the object height is bigger than the calibration height, use positive values for the z offset. If the object height is smaller than the calibration height, use negative values for the z offset.



NOTE!
Adjust the z offset within Module Image Calibration if it is the same z offset for all further modules that use the calibration result as input. It is also possible to use different z offsets directly within the modules that use the calibration result as input.

- Alternatively, it is possible to use the z offset in Module Image Calibration to calculate back to the surface. The z offset of other modules that use the calibration result as input can then use the real object height (compared to the surface).



Do the calibration once and use the calibration result afterwards in further modules (if supported by the module). If the relation between camera and calibration plate changes or if the lens is adjusted (e.g. via changing the focus position of the lens), a re-calibration is necessary.

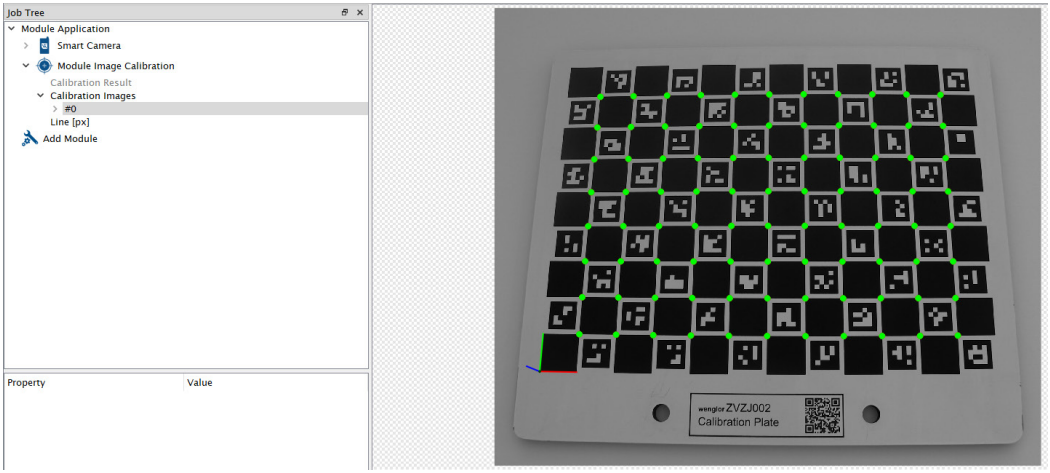
Parameters for Calibration Mode “Charuco on Device” and “Charuco from File”

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Reprojection Error Calibration	Returns a numeric value after the calibration. It is the mean difference over all images used for the calibration between the computed and the found corners (for details check the red and green corner points at the Calibration Images list). The smaller the value, the better the calibration. A typical value for the reprojection error with the wenglor calibration plates ZVZJ is 0.1. If the error is larger than 1, it is recommended to check the configuration (calibration images, calibration size). If the reprojection error is bigger than 20, the calibration fails (e.g. in case of a wrong selection at the calibration size).
Length of Line [mm]	Returns the calculated length of the line in mm. Place the line in the image at a known distance and check if the calculated length fits to the real distance. Adjust the z offset first, so that the measuring is done at the correct height level.
Input Image	Link 8 bit single channel image as input image of the module.
Calibration Mode	Select the calibration mode: • Charuco on Device: Calibration via charuco calibration plate. • Charuco from File: Load an existing calibration result (e.g. to use the same calibration result in several jobs). Several parameters are not available after loading a calibration result. • Linear: Calibration via pixel to millimeter factor. A separate table shows the parameters for the linear calibration mode.
Calibration Size	Select the size of the calibration plate. Several sizes are available: • ZVZJ001 / ZVZJ005 48 x 60 mm • ZVZJ002 / ZVZJ006 90 x 120 mm • ZVZJ003 168 x 210 mm • ZVZJ004 252 x 324 mm • 24 x 30 mm (only available as PDF) Parameter is not available if loading a calibration result from file.
Capture Image	Capture the current image. The captured image appears in the Calibration Images list (see Sub-Module). Parameter is not available if loading a calibration result from file.
Reference Calibration Image	Choose one of the calibration images by setting the corresponding number. For example, set the number 1 for the calibration image #1. The measurement plane must be on the same level as the reference calibration plane or parallel to it. Tilting the measuring plane to the reference calibration plane is not supported. Parameter is not available if loading a calibration result from file.
Calibrate	Press calibrate to start the calibration on the captured images. Parameter is not available if loading a calibration result from file.
Compute Undistorted Image	Defines if undistorted image is computed or not. If activated, the Sub-Module "Undistorted Image" appears.

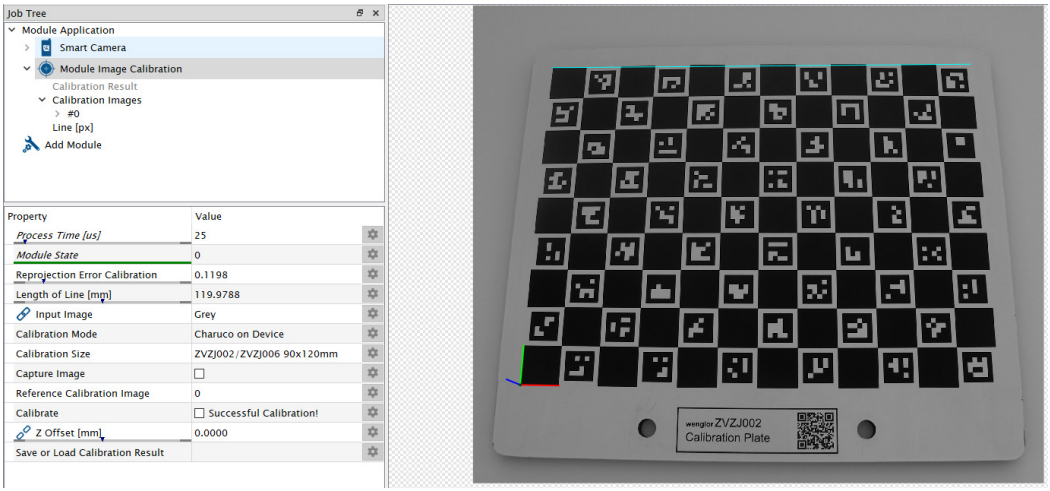
Z Offset [mm]	If the measurement plane is on a different height level than the reference calibration image, adjust the z offset (Tilting the measuring plane to the reference calibration plane is not supported). • If the object height is bigger than the calibration height, use positive values for the z offset. • If the object height is smaller than the calibration height, use negative values for the z offset.
Save or Load Calibration Result	Save the calibration result in order to use it in other jobs. Use the option also to load an existing calibration. After loading the calibration, not all options are accessible as the calibration images are not available.

Sub-Modules for Calibration Mode “Charuco on Device” and “Charuco from File”

- Undistorted Image: Returns the undistorted image as output image in order to link it in further modules as input image. Only available if parameter "Compute Undistorted Image" is activated.
- Calibration Result: Returns the calibration result. Further modules (e.g. Module Image Measure) can use the calibration result.
- Calibration Images: Lists all stored calibration images. After the calibration, the actual corner points are visualized in green color. The position where the corner points are supposed to be – based on the calibration result approximation (the algorithm tries to minimize the reprojection error by adjusting the camera and distortion parameters) – are visualized in red. The green points cover the red ones, if they are at a similar position (depending on the zoom level). Deleting wrong captured images is possible via the context menu “Delete” on the specific image in the Job Tree. The Sub-Module Calibration Images is not available if loading a calibration file.



- Line: Edit the line position to verify at a known distance if the calibration is successful. The result “Length of Line [mm]” returns then the calculated length of the line in mm. Adjust the z offset first, so that the measuring is done at the correct height level.

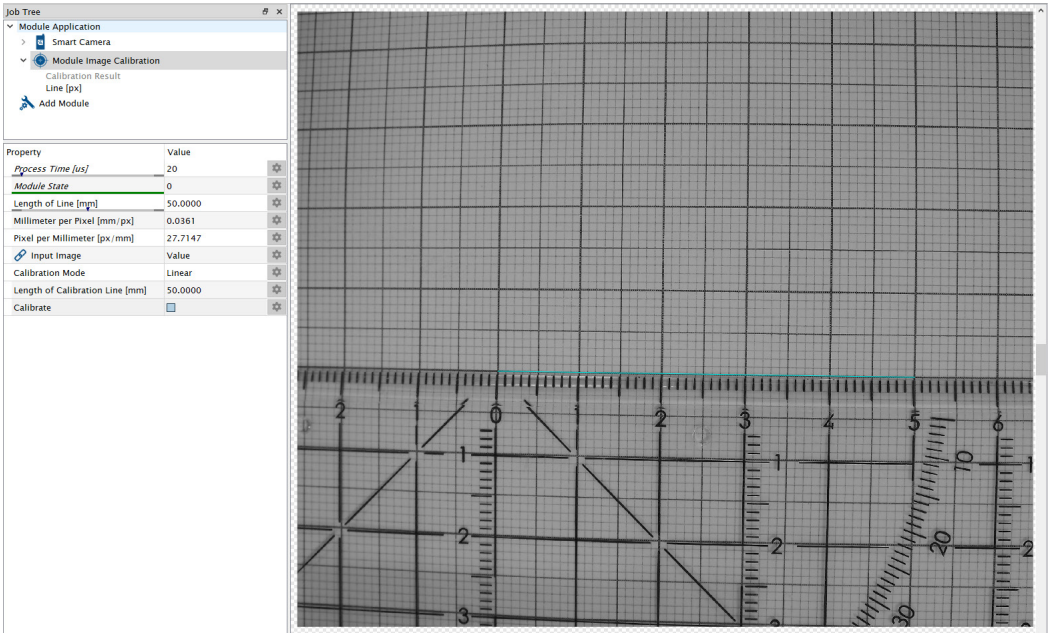


Parameters for Calibration Mode “Linear”

Process Time [us]	Process time to run the module in μs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Length of Line [mm]	Returns the calculated length of the line in mm. Place the line in the image at a known distance and check if the calculated length fits to the real distance.
Millimeter per Pixel [mm/px]	Returns the millimeter per pixel factor after a successful calibration.
Pixel per Millimeter [px/mm]	Returns the pixel per millimeter factor after a successful calibration.
Input Image	Link 8 bit single channel image as input image of the module.
Calibration Mode	Select the calibration mode: • Charuco on Device: Calibration via charuco calibration plate. A separate table shows the parameters for the charuco on device calibration mode. • Charuco from File: Load an existing calibration result (e.g. to use the same calibration result in several jobs). A separate table shows the parameters for the charuco from file calibration mode. • Linear: Calibration via pixel to millimeter factor
Length of Calibration Line [mm]	Put an object with a known length in the field of view of the camera (e.g. ruler). Place the line (see Sub-Module) on the object within the image and enter the length of the known object in mm.
Calibrate	Press calibrate to calculate the pixel per millimeter factor with the current line position and the entered Length of Calibration Line.

Sub-Modules for Calibration Mode “Linear”

- Calibration Result: Returns the calibration result. Further modules (e.g. Module Image Measure) can use the calibration result.
- Line: Edit the line position to calibrate and to verify at further known distances if the calibration was successful. The result “Length of Line [mm]” returns then the calculated length of the line in mm.



7.3.2 Module Image Region

Use Module Image Region to create a Region of Interest in the image. Other modules can use it in order to reduce the process time by running algorithms only on a specific region and not on the complete image. Optionally, it is possible to calculate interesting grey values within the Region of Interest, e.g. to calculate the mean grey value.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Input Image	Link 8 bit single channel image as input image of the module.
Coordinate System	Optionally, link input coordinate system to track the Region of Interest. The Region of Interest moves then together with the position of the coordinate system.
Find Min Max	If activated, additional results are calculated (by default deactivated to reduce the process time): • Min Grey Value: Returns the minimum grey value of all pixels within the current Region of Interest. • Max Grey Value: Returns the maximum grey value of all pixels within the current Region of Interest.
Compute Mean	If activated, the mean grey value is calculated (by default deactivated to reduce the process time). Mean Grey Value: Returns the mean grey value of all pixels within the current Region of Interest.

Adding new shapes is possible via the Module Toolbar.

1. Select the mathematical operation.

	Add	Add new shape to existing shapes.
	Subtract	Subtract new shape from existing shapes.
	Subtract symmetrically	Add new shape to existing shapes and substract common area of the new shape and the existing shapes.
	Intersect	Select the common area of the new shape and the existing shapes.

NOTE!
The order of shapes is fix and defined by the time when the shapes are created. As a result, the overall shape of all previously existing shapes is used for the offsetting of shapes.

2. Select the new shape

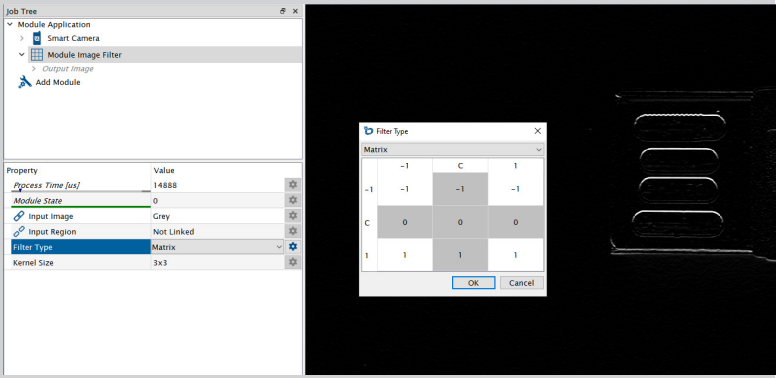
	Rectangle by two or three points	Create a rectangle with two or three points.
	Circle by two or three points	Create a circle with two or three points.
	Polygon	Create a polygon with flexible number of points (minimum of three points). Each click on the left mouse button defines a corner of the polygon. Finalize the polygon with a double click on the left mouse button at the last corner position. Delete individual points by pressing and holding the Ctrl+Shift keys and clicking the point with the left mouse button. Add new points to the polygon by pressing and holding the Alt+Shift keys and clicking the left mouse button at the line connection between the two relevant points.

3. Draw the new shape in the image.

Sub-Modules contain the Region and the Bounding Box. Furthermore, it is possible to select and edit single shapes at the Sub-Module “Set”.

7.3.3 Module Image Filter

Use Module Image Filter to emphasize or suppress certain image characteristics and to improve the image quality. Use the filtered image then as input in further modules.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Input Image	Link 8 bit single channel image as input image of the module.
Input Region	Optionally, link input region to apply the filter only on a Region of Interest (to reduce the process time).
Filter Type	Select the filter type: • Off: No filter used. • Closing: Consists of a dilation filter followed by an erosion filter in order to remove small holes (typically black pixels). • Dilation: Filter to enlarge the boundaries of objects (typically white pixels). • Erosion: Filter to shrink the boundaries of objects (typically white pixels). • Gauss: “Low-pass filter” to reduce noise by blurring the image. • Matrix: User-defined filter (for experts). Click on the settings icon to adjust the weightings for the neighbors. Each neighbor gray value is multiplied with its weighting factor. The sum of the values is then divided by the number of neighbors and provides the new gray value for the central pixel. The calculation is applied separately for all pixels. Example with horizontal edge filter:  • Median: Filter to reduce noise (can preserve edges while removing noise). Each pixel value is replaced with the mean gray value of the neighbors defined by the kernel size. • Opening: Consists of an erosion filter followed by a dilation filter in order to remove small objects (typically white pixels). • Sharpen: Filter to amplify edges and also noise in the image. • Sobel: Filter to detect edges. Edges get white and homogeneous areas get black.

Kernel Size	Select the kernel size for filtering: • 3x3 • 5x5

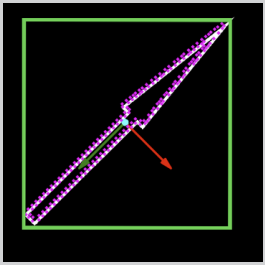
The Sub-Module Output Image provides the filtered image for further modules.

7.3.4 Module Image Locator

Use Module Image Locator to detect, to locate and to count objects by feature points. Teach one or several shape models. At teaching, the algorithm finds feature points automatically at characteristic points (e.g. edges, contours). Find objects afterwards independent of the position, the rotation and the scaling level (depending on the set parameters).

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Result True Count	Returns the number of found objects (independent of parameter Result Max Count).
Input Image	Link 8 bit single channel image as input image of the module.
Input Region	Optionally, link input region to limit the search for objects to the Region of Interest (to reduce the process time and to make the search more robust). **NOTE!** • If the center of gravity of a potential object (without considering the Origin Offset) is within the input region, it is a valid result – otherwise it is dismissed. • Reducing the process time is possible if using small Input Regions and small sizes of the shape models.
Calibration	Optionally link calibration as input to calculate in mm. In case of linked calibration, the following additional parameters appear: • Z Offset (available if Calibration Mode of input calibration is "Charuco on Device" or "Charuco from File"): Defines the height difference compared to the height defined in Module Image Calibration. Enter positive values, if the height for the module is bigger than for Module Image Calibration. Enter negative values, if the height for the module is smaller than for Module Image Calibration • Unit: Defines if values are in pixel and millimeter or only in pixel. **NOTE!** • Keep the height difference compared to the calibration height as small as possible for best accuracy. • For details about the calibration, see section “7.3.1 Module Image Calibration”.
Result Max Count	Defines the size of the Result List (see Sub-Module).

Sort Rule	Defines how to sort the results in the Result List (see Sub-Module): • Score (ascending/descending) • Center X (ascending/descending) • Center Y (ascending/descending) • Scale (ascending/descending)
Shape Models	Defines the number of different shapes (see Sub-Module Shape Models).
Angle Start [deg]	Defines the first angle in which the model rotates in reference to the orientation of the taught in model (e.g. -20°). Positive angles are in clockwise direction.
Angle End [deg]	Defines the last angle in which the model rotates in reference to the orientation of the taught in model (e.g. 20°). Positive angles are in clockwise direction.
Angle Step [deg]	Defines the individual increments within the selected angle range.  **NOTE!** Angle Start, End and Step define the number of angle steps that the algorithm must check. Keep the number of angle steps as small as possible to reduce the process time.
Scale Min	Defines the lower boundary of the possible scaling range (e.g. 0.9). The value 1 corresponds to the original size of the model.
Scale Max	Defines the upper boundary of the possible scaling range (e.g. 1.1). The value 1 corresponds to the original size of the model.
Scale Step	Defines the individual increments within the selected scaling range.  **NOTE!** Scale Min, Max and Step define the number of scaling steps that the algorithm must check. Keep the number of scaling steps as small as possible to reduce the process time.
Min Score	Defines the minimum required correspondence between model and potential results. Lower the value to find more results. Increase the value in case of finding wrong results.

Max Overlap	Defines the accepted overlapping of results. The relevant area for overlapping is the bounding box of the results (enclosing rectangle without rotation).  The parameter Max Overlap refers to the intersection over union (area of overlap divided by area of union).  **NOTE!** The overlap check can only consider detected objects. In case of overlapping results, the result with the lower score is dismissed and the one with the higher score remains.
Auto Accuracy	By default, the module calculates the accuracy automatically when teaching models. If deactivating “Auto Accuracy”, the additional parameter “Accuracy [1 (high) – 6 (low)]” appears. The smaller the value for the parameter, the higher the accuracy (visible via higher feature points density), but the bigger the process time. **NOTE!** • Small objects require smaller values for the accuracy, whereas bigger objects also work with bigger values. • The accuracy is reduced by searching on an image with a lower resolution resulting in significantly faster process times. Increase the parameter in order to reduce the process time especially for big objects. • For the parameter “Accuracy [1 (high) – 6 (low)]” values of 1 and 2 use the full image resolution, values of 3 and 4 use half of the image resolution and values of 5 and 6 use a quarter of the image resolution.

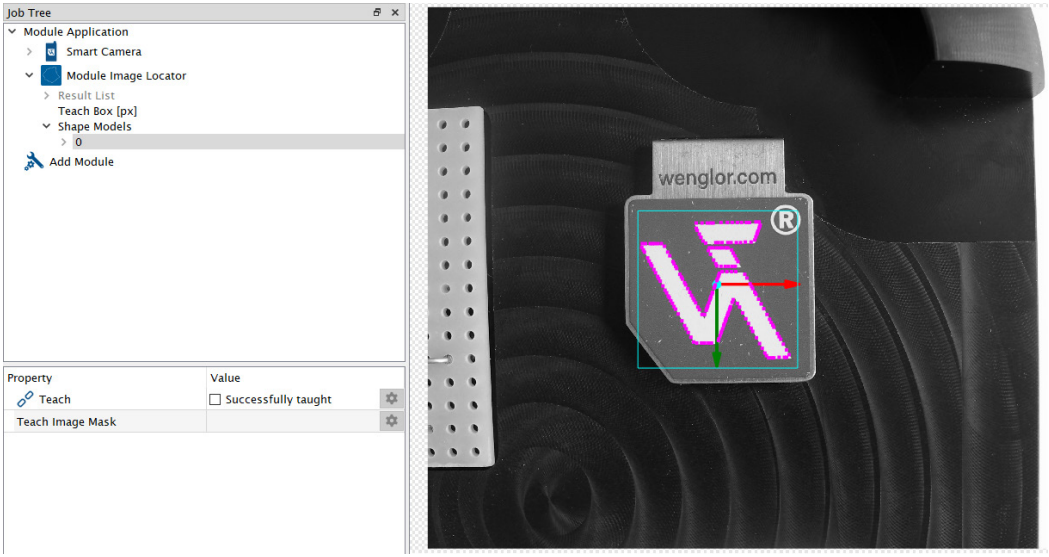
The Sub-Module Result List provides the results for all found objects. For each result, the following outputs are calculated:

- Shape Model: Returns the number of the shape model that belongs to the result (see Sub-Module Shape Model).
- Scale: Returns the scale value of the found object.
- Score: Returns the score value of the found object. The higher the value, the better the match. Increase the parameter “Min Score” to avoid finding wrong results.
- Coordinate System: Returns the position of the found object (including the x and y coordinates and the rotation). It is the center of gravity of the shape model (including the Origin Offset). If needed, link the coordinate system as input in other modules (e.g. in Module Image Measure).

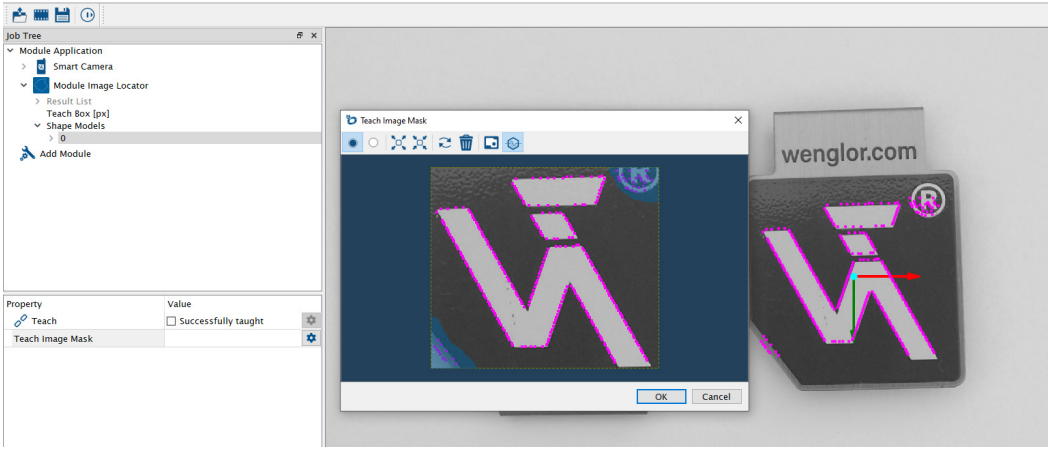
The Sub-Module Teach Box defines the teach area. Move it on the relevant part of the object and teach the shape via the Sub-Module Shape Model.

The Sub-Module Shape Model enables to teach all shape models. The parameter “Shape Models” defines the number of shape models. Selecting the specific shape model, lists the following options:

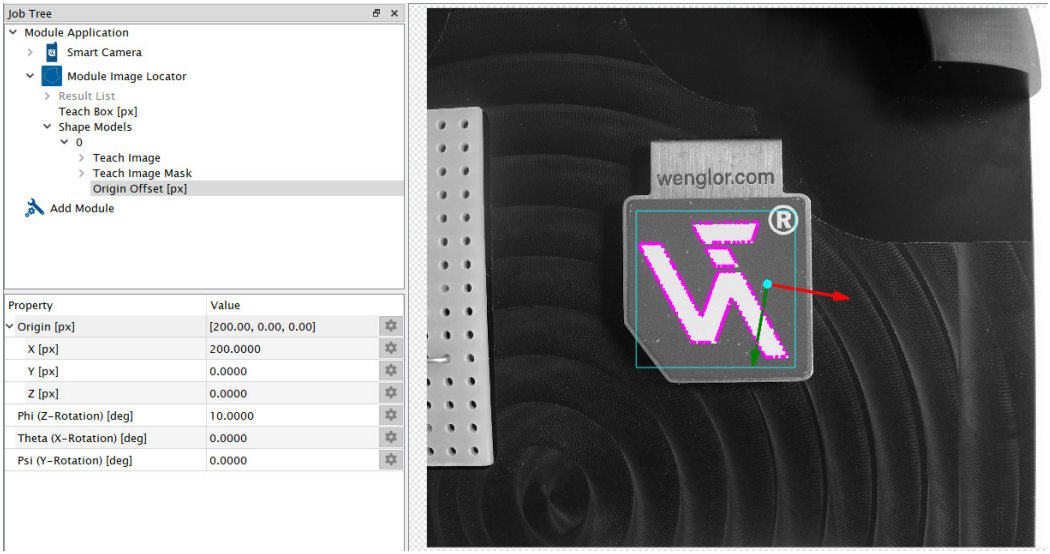
- Teach: Place the Teach Box on the relevant object and press the teach button. Linking the teach functionality to a job result is supported. In case of failed teaching because of not enough feature points, set a smaller value for the parameter “Accuracy [1 (high) – 6 (low)]” or increase the size of the teach box. In case of failed teaching because of a teach timeout, set a bigger value for the parameter “Accuracy [1 (high) – 6 (low)]” or reduce the number of angle and scaling steps.



- Teach Image Mask: Click on the settings next to Teach Image Mask to check and edit the teach image. Remove areas where no feature points should be found in order to make the search more robust and faster.



NOTE!
Each shape model contains the Teach Image, the Teach Image Mask and the option to enter offsets for the origin (for x and y coordinates and the z rotation via angle Phi). It allows changing the gripping point for pick and place applications (for example with robots).



7.3.5 Module Image Pattern Match

Use Module Image Pattern Match to detect, to locate and to count objects via shape-based matching. The module uses the shape-based matching algorithm of the HALCON 22.11 library of the company MVTec. Teach one or several shape models. Find objects afterwards independent of the position, the rotation and the scaling level (depending on the parameter settings).

NOTE!

- Sharp images with high contrast are essential as the algorithm analyses contours.
- Avoid ambiguous contours, e.g. round objects with big span angle (defined via parameters Angle Start and Angle End) or contours close together with big scaling range (defined via parameters Scale Min and Scale Max).
- Optimize the performance:
  - » Generally, the algorithm works faster for bigger objects as more pyramid levels are possible. It means that at first the algorithm tries to find objects on images with reduced image resolutions. Then it only investigates the potential results on images with higher resolutions. Consider that at reduced image resolutions “small and thin contours” might disappear.
  - » Increase the Greediness parameter as much as possible.
  - » Increase the parameter “Min Score” as much as possible (especially for big objects).
  - » Reduce the span angle defined by the parameters Angle Start and Angle End as much as possible to avoid checks on unnecessary angle steps.
  - » Reduce the scaling range defines by the parameters Scale Min and Scale Max as much as possible to avoid checks on unnecessary scaling steps.
  - » Reduce the Search Box to the position where the objects are supposed to be.
  - » Check and optimize the contour after teaching via the parameter “Contour DXF”. Remove unnecessary contours.
- For details about the shape-based matching algorithm, check the HALCON solution guide for matching.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Result True Count	Returns the number of found objects (maximum value of Result True Count depends on parameter Result Max Count).
Input Image	Link 8 bit single channel image as input image of the module.

Calibration	Optionally link calibration as input to calculate in mm. In case of linked calibration, the following additional parameters appear: • Z Offset (available if Calibration Mode of input calibration is "Charuco on Device" or "Charuco from File"): Defines the height difference compared to the height defined in Module Image Calibration. Enter positive values, if the height for the module is bigger than for Module Image Calibration. Enter negative values, if the height for the module is smaller than for Module Image Calibration • Unit: Defines if values are in pixel and millimeter or only in pixel. **NOTE!** • Keep the height difference compared to the calibration height as small as possible for best accuracy. • For details about the calibration, see section "7.3.1 Module Image Calibration".
Read Timeout [µs]	Defines the maximum time the algorithm searches for objects. If searching is not finished until the timeout, all results of the module are set to error. **NOTE!** The process time of the module can take slightly longer than the parameter Read Timeout.
Result Max Count	Defines the size of the Result List (see Sub-Module) and the maximum value for Result True Count. Increase Result Max Count if several objects with a similar score are expected.
Sort Rule	Defines how to sort the results in the Result List (see Sub-Module): • Score (ascending/descending) • Center X (ascending/descending) • Center Y (ascending/descending) • Scale (ascending/descending)
Shape Models	Defines the number of different shapes (see Sub-Module Shape Models).
Angle Start [deg]	Defines the first angle in which the model rotates in reference to the orientation of the taught in model (e.g. -20°). **NOTE!** The algorithm must check on all angle positions. Keep the span angle as small as possible to reduce the process time. As the angle result of found objects is never 0°, use maximum -5° for the Angle Start value. Positive angles are in clockwise direction.
Angle End [deg]	Defines the last angle in which the model rotates in reference to the orientation of the taught in model (e.g. 20°). **NOTE!** The algorithm must check on all angle positions. Keep the span angle as small as possible to reduce the process time. As the angle result of found objects is never 0°, use minimum 5° for the Angle End. Positive angles are in clockwise direction.

Scale Min	Defines the lower boundary of the possible scaling range. The value 1 corresponds to the original size of the model. **NOTE!** The algorithm must check on all scaling steps. Keep the scaling range as small as possible to reduce the process time.
Scale Max	Defines the upper boundary of the possible scaling range. The value 1 corresponds to the original size of the model. **NOTE!** The algorithm must check on all scaling steps. Keep the scaling range as small as possible to reduce the process time.
Min Score	Defines the minimum required correspondence between model and potential results. Lower the value to find more results (if e.g. the algorithm does not find relevant results). Increase the value in case of finding wrong results. **NOTE!** The parameter Min Score has a significant influence on the process time of the module. Increase the Min Score parameter at objects with high score values to reduce the process time.
Max Overlap	Defines the accepted overlapping of results. If two results overlap each other by more than the parameter Max Overlap, the algorithm only returns the result with the higher score. The relevant area for overlapping is the smallest enclosing rectangle of arbitrary orientation. **NOTE!** The overlap check can only consider detected objects.
Greediness	Defines how “aggressive” the search heuristics work. It is a trade between thoroughness and speed. Minimum 0 allows reliable search, but has high process times as the algorithm also needs to examine very unlikely candidates. Maximum 1 has small process times, but may miss potential results. **NOTE!** Greediness and Min Score influence each other. Generally, higher values for Greediness and sufficiently lower values for Min Score reduce the process time.

The Sub-Module Result List provides the results for all found objects. For each result, the following outputs are calculated:

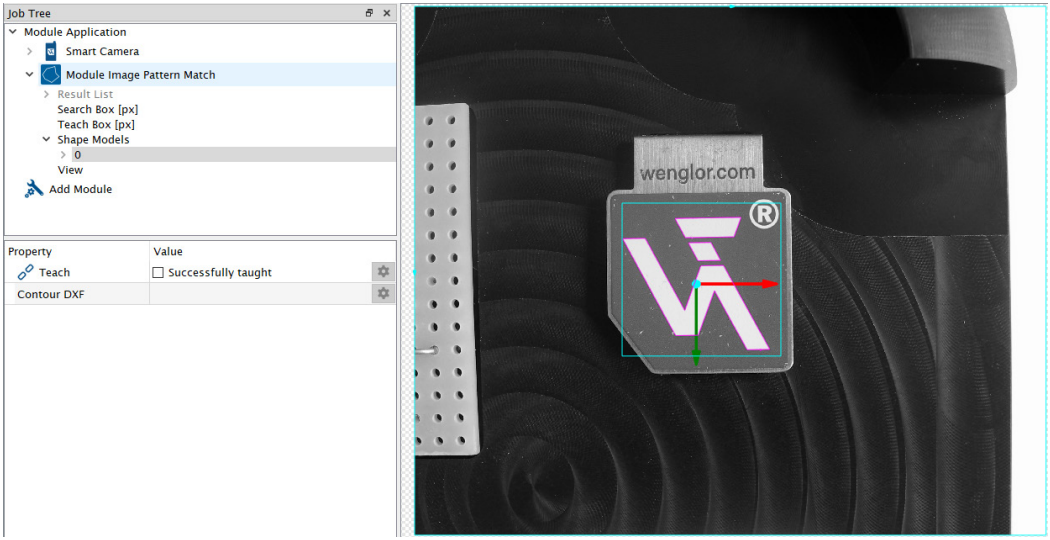
- Shape Model: Returns the number of the shape model that belongs to the result (see Sub-Module Shape Model).
- Scale: Returns the scale value of the found object.
- Score: Returns the score value of the found object. The higher the value, the better the match.
- Coordinate System: Returns the position of the found object (including x and y coordinates and rotation). It is the center of gravity of the shape model (including the Origin Offset).

The Sub-Module Search Box defines the area where the algorithm looks for potential objects. The Search Box is always at a fix position (not linkable).

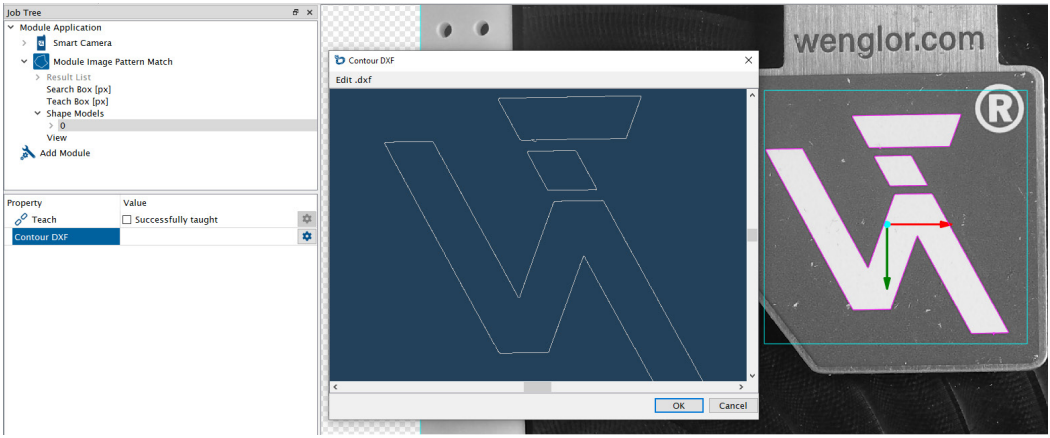
The Sub-Module Teach Box defines the teach area. Move it on the relevant part of the object and teach the shape via the Sub-Module Shape Model.

The Sub-Module Shape Model enables to teach all shape models. The parameter “Shape Models” defines the number of shape models. Selecting the specific shape model, lists the following options:

- Teach: Place the Teach Box on the relevant object and press the teach button. Linking the teach functionality to a job result is supported. Teaching can take some time.



- Contour DXF: Click on the settings icon next to Contour DXF to check and edit the contour. Remove unnecessary contours in order to make the search more robust and faster. Clicking on “Edit.dxf” turns the mouse into a red dot. Delete individual contours via clicking on the left mouse button and moving the mouse on the relevant contours. Zoom in or out via pressing the button Ctrl and turning the mouse wheel.



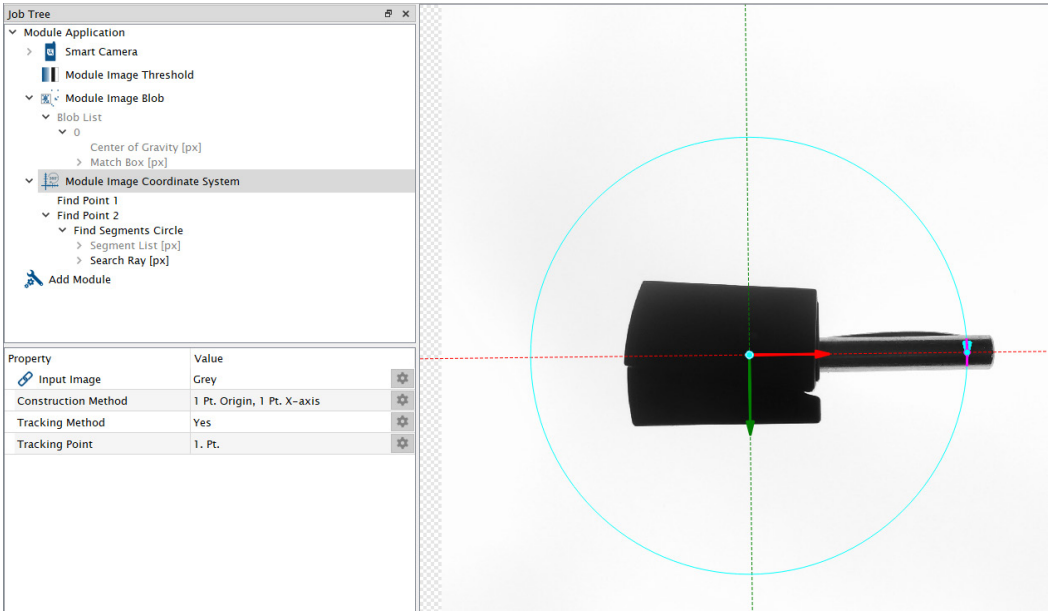
NOTE!
Each shape model contains the option to enter offsets for the origin (for x and y). It allows changing the gripping point for pick and place applications (for example with robots).

The Sub-Module View allows to activate or to deactivate certain UI elements for found objects (contour, coordinate system, extension lines of coordinate system).

7.3.6 Module Image Coordinate System

Use Module Image Coordinate System to track an object by creating an individual coordinate system. Further modules (e.g. Module Image Measure or Module Image Region) can use the output coordinate system as input to track e.g. regions or search geometries.

Process Time [us]	Process time to run the module in μs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Input Image	Link 8 bit single channel image as input image of the module.
Construction Method	Defines how to construct the coordinate system: • 1 Pt. Origin: One point defines the origin of the translatory coordinate system. • 1 Pt. X-axis, 1 Pt. Y-axis: One point defines the x-axis and one point the y-axis of the translatory coordinate system. • 1 Pt. Origin, 1 Pt. X-axis: One point defines the origin and one point the x-axis of the rotary coordinate system. • 1 Pt. Origin, 1 Pt. Y-axis: One point defines the origin and one point the y-axis of the rotary coordinate system. • 2 Pt. X-axis, 1 Pt. Y-axis: Two points define
Tracking Method	Defines the tracking method (only available if at least two points create the coordinate system): • No: No tracking of points. • Yes: Track the points horizontally and vertically by one of the points. • Horizontally: Track points only horizontally by one of the points. • Vertically: Track points only vertically by one of the points. If tracking method is active, the parameter “Tracking Point” appears. It defines the point used for tracking of the other points.  **NOTE!** The template “Create coordinate system” contains an example of tracking. The coordinate system uses tracking method “1 Pt. Origin, 1 Pt. X-axis”. The origin point is linked to the center of gravity of a blob. The point on the x-axis is the center of the first arc segment. Via tracking by the first point, the search geometry for Find Segments on Circle moves together with the center of gravity of the blob.



The Sub-Module Coordinate System provides the output coordinate system that other modules can use as input.

The Sub-Modules Find Point (1, 2 or 3 Points depending on Construction Method) contains the properties for each point:

- Found Point: Returns the coordinates of the found point.
- Input Point: Enter fix values for the input point or link the input point with the found point of another module (job result). Only available for Find Method “Point (fix or linked)”.
- Find Method: Defines the find method.
  - » Point (fix or linked): Enter a fix point or link it to the found point of another module
  - » Edge on Line: Use an edge on a line as point.
  - » Edge on Arc: Use an edge on an arc as point.
  - » Segment on Line: Use any point of the first line segment on a search line as point.
  - » Segment on Arc: Use any point of the first arc segment on a search arc as point.
  - » Segment on Circle: Use any point of the first arc segment on a search circle as point.
  - » Find Line: Use any point on a line as point.
  - » Find Arc: Use any point on an arc as point.

Edge on Line and Edge on Arc

- Edge Point: Returns the coordinates of the edge point.
- Edge Polarity: Defines the polarity of the edge.
  - » Either: Finds edges at transitions from bright to dark or from dark to bright along the search ray.
  - » Bright to Dark: Finds edges only at transitions from bright to dark along the search ray.
  - » Dark to Bright: Finds edges only at transitions from dark to bright along the search ray.
- Find by: Define the relevant edge result.
  - » Best Score: Uses the result with the highest score as edge point on the search ray.
  - » First Score: Uses the first result as edge point on the search ray.
  - » Last Score: Uses the last result as edge point on the search ray.
- Edge Width: Defines the edge width. The higher the value, the bigger the smoothing effect on the gray-values along the search ray.
- Threshold Gradient Pos [GrM]: Defines the minimum edge sensitivity for edge points from dark to bright along the search ray.
- Threshold Gradient Neg [GrM]: Defines the minimum edge sensitivity for edge points from bright to dark along the search ray.
- Orientation: Defines the orientation of the search ray (default or swap).

Segment on Line, Segment on Arc and Segment on Circle

- For details about properties of “Find Segments Line”, “Find Segments Arc” or “Find Segments Circle” see section “7.4.5 Module Image Measure”
- For details about properties of “Point on Geometry” → see section “7.4.5 Module Image Measure”

Find Line and Find Arc

- For details about properties of “Find Geo Line” or “Find Geo Arc” → see section “7.4.5 Module Image Measure”
- For details about properties of “Point on Geometry” → see section “7.4.5 Module Image Measure”

7.4 Image-Based Processing Modules

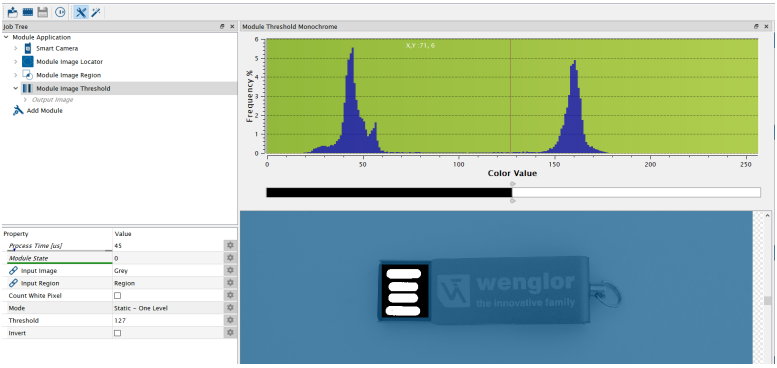
7.4.1 Module Image Threshold

Use Module Image Threshold to transform a single channel 8-bit image (e.g. gray image) into a binary image (black and white image). Pixels with gray values within the thresholds get white (gray value 255) and all other pixels get black (gray value 0). By counting the white pixels, a simple presence check is possible.

NOTE!

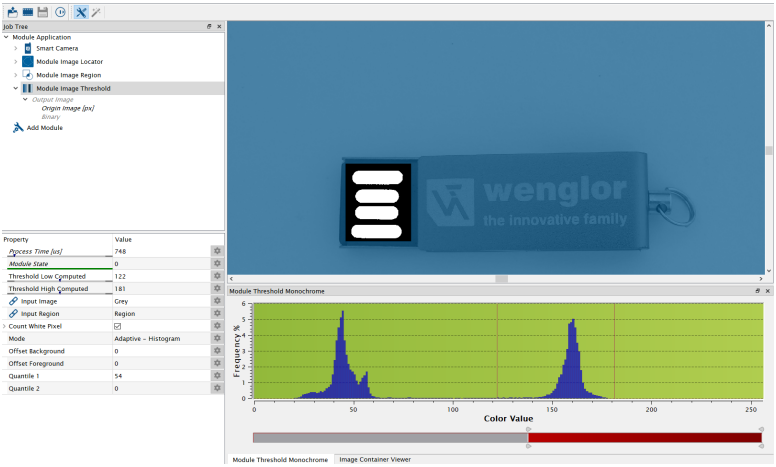
- Default values and parameter names assume that the object is brighter than the background. The result of the module is supposed to show the object in white color on a black background.
- Use the tools “Profile” and “Histogram” at the menu bar (→ View) to investigate the gray values within the image.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Input Image	Link 8 bit single channel image as input image of the module.
Input Region	Optionally, link input region to apply the threshold only on a specific region. All pixels outside of the region are black in the output image. In case of no linked input region, the algorithm uses the complete image. **NOTE!** Linking an input region allows to search only at relevant positions. In general, there is no performance boost by linking an input region.
Count White Pixel	Activate or deactivate counting of white pixels (by default, activated). **NOTE!** In some modes, deactivating the option reduces the process time significantly. In the modes “Static - Two Level”, “Adaptive - Reference Boxes” and “Adaptive - Histogram” deactivating the option has no significant performance advantage. If activated, the following parameters appear: • Pixel Count: Returns the number of detected white pixels in the output image. • Teach: Adjusts the minimum and maximum values of the result “Pixel Count” so that the current number of detected pixels is in the middle of the two values. The window width between minimum and maximum remains untouched. Teaching is possible once manually. Linking the value to a job result, teaches every time the linked value is set to TRUE.

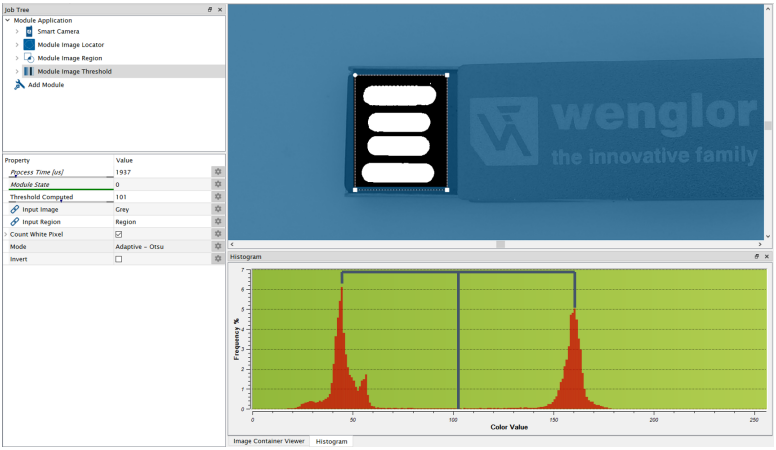
Mode	The following threshold modes are available: • Static - One Level: Pixels with gray values equal or lower than parameter “Threshold” get black and all other pixels get white. Option to invert the colors. • Static - Two Levels: If parameter “Threshold Low” is smaller than parameter “Threshold High”, pixels with gray values between the two thresholds get white. All other pixels get black. If parameter “Threshold Low” is bigger than parameter “Threshold High”, pixels with gray values between the two thresholds get black. **NOTE!** • Use the magic wand tool in the Module Toolbar to find automatically suitable threshold values for both static modes. The algorithm adds or subtracts 20 gray value steps to the selected pixel in the image. Activate the tool in the module toolbar, click on one of the pixels belonging to the object within the image and the algorithm calculates suitable threshold values. • Use the histogram in the Module Toolbar to visualize the histogram for the relevant region and to adjust the threshold parameters in a graphical way.  • Adaptive - Reference Boxes: The algorithm calculates the threshold values dynamically for each image from two reference areas (background and foreground). The mean gray value of the pixels in the background gets Threshold High Computed - the mean gray value of the pixels in the foreground gets Threshold Low Computed. Applying offsets for foreground and background is possible. • Adaptive - Histogram: The algorithm calculates the threshold values dynamically for each image using the quantiles from the histogram. Threshold Low Computed is the gray value at which the amount of lower gray values (in percent) reaches the value of Quantile 1. Threshold High Computed is the gray value at which the amount of upper gray values (in percent) reaches the value of Quantile 2. Applying offsets for foreground and background is possible.

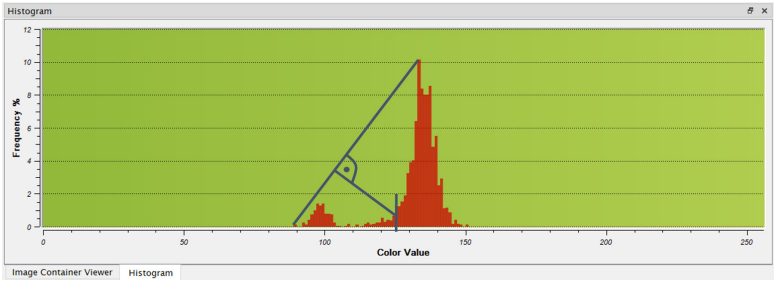
Mode

NOTE!
Use the histogramm widget in the Module Toolbar to find proper values for the quatities.



- Adaptive - Otsu: The algorithm calculates the threshold dynamically for each image assuming two peaks in the histogram. Threshold Computed is set in the middle of both peaks. Option to invert white and black color for pixels.



Mode	• Adaptive - Triangle: The algorithm calculates the threshold dynamically for each image analyzing the histogram and determining the threshold by constructing a triangle. One line is from the histogram peak to the farthest end of the histogram. Threshold Calculated is the point of maximum distance between the line and the histogram. Option to invert white and black colors for pixels.  • Adaptive - Mean: The algorithm calculates the thresholds dynamically for each image and for each pixel. The threshold at a pixel is the mean of the neighborhood defined by the parameter Box Size. Adjust the threshold by the fix parameter Offset. Option to invert the colors. Especially helpful at changing light conditions within one image. • Adaptive - Gaussian: The algorithm calculates the thresholds dynamically for each image and for each pixel. The threshold at a pixel is the weighted sum (cross-correlation with a Gaussian window) of the neighborhood defined by the parameter Box Size. Adjust the threshold by the fix parameter Offset. Option to invert the colors. Especially helpful at changing light conditions within one image.

The Sub-Module Output Image provides the binary image for further modules (e.g. for Module Image Blob).

7.4.2 Module Image Threshold HSV

Use Module Image Threshold HSV to transform a three-channel HSV image (color image) into a binary image (black and white image). If the pixel values for all activated channels are within the thresholds, the pixel gets white (gray value 255). All other pixels get black (gray value 0). By counting the white pixels, a simple presence and color check is possible.

NOTE!
HSV images consist of the three channels Hue, Saturation and Value. Activate or deactivate the single channels and set suitable threshold values for the channels.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Input Image	Link three-channel HSV image as input image of the module.
Input Region	Optionally, link input region to apply the threshold only on a specific region. All pixels outside of the region are black in the output image. In case of no linked input region, the algorithm uses the complete image. **NOTE!** Linking an input region allows to search only at relevant positions. In general, there is no performance boost by linking an input region.
Count White Pixel	Activate or deactivate counting of white pixels (by default, activated). If activated, the following parameters appear: • Pixel Count: Returns the number of detected white pixels in the output image. • Teach: Adjusts the minimum and maximum values of the result “Pixel Count” so that the current number of detected pixels is in the middle of the two values. The window width between minimum and maximum remains untouched. Teaching is possible once manually. Linking the value to a job result, teaches every time the linked value is set to TRUE.

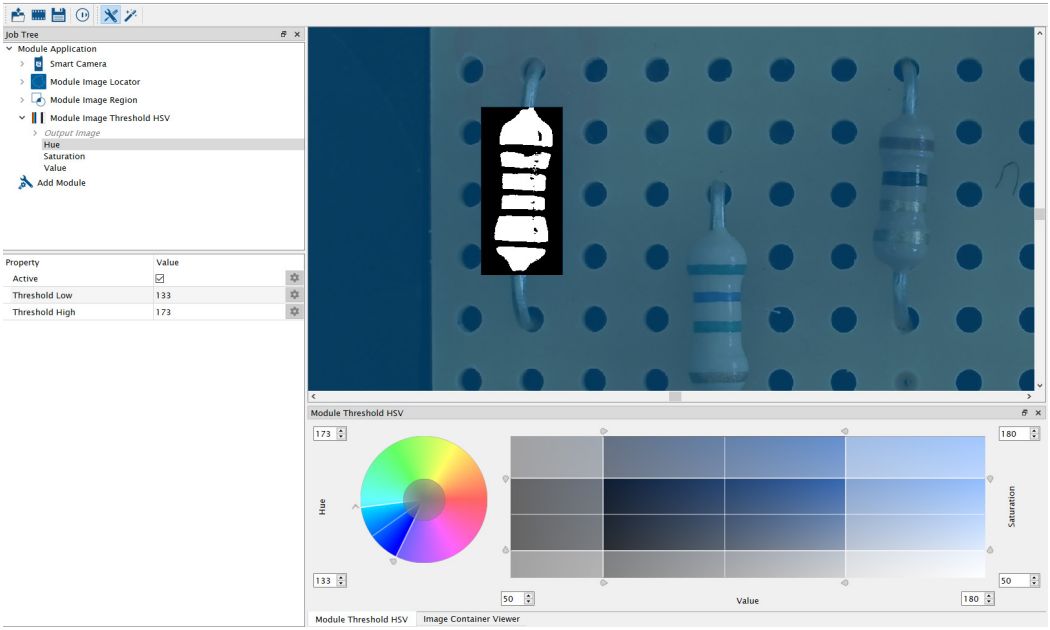
The Sub-Module Output Image provides the binary image for further modules (e.g. for Module Image Blob).
The Sub-Modules Hue, Saturation and Value contain the following parameters:

- Active: Option to activate or deactivate the specific channel. If deactivated, the thresholds of the channel are not used.
- Threshold Low: Defines the lower threshold of the specific channel.
- Threshold High: Defines the upper threshold of the specific channel.

NOTE!

- Use the Magic Wand tool in the Module Toolbar to find automatically suitable thresholds for all channels. Activate the tool in the module toolbar, click on one of the pixels belonging to the object within the image and the algorithm calculates suitable threshold values.

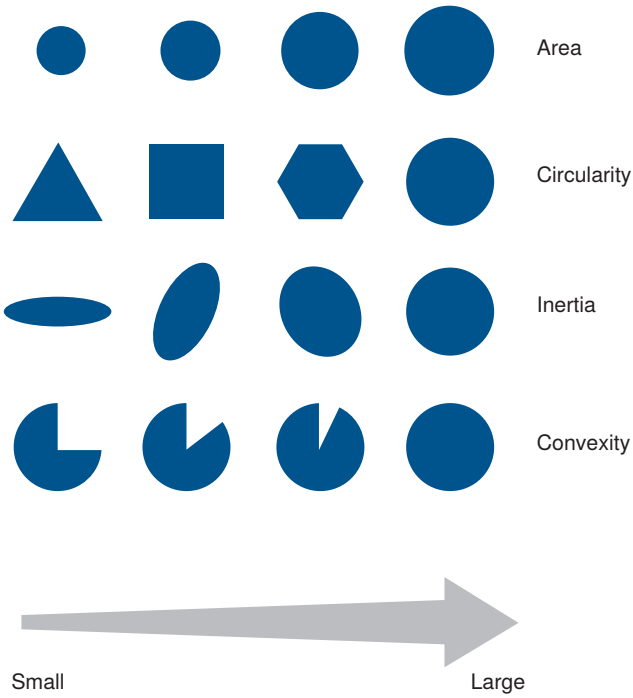
- To fine-tune, open the settings in the Module Toolbar and adjust the thresholds in a graphical way.



7.4.3 Module Image Blob

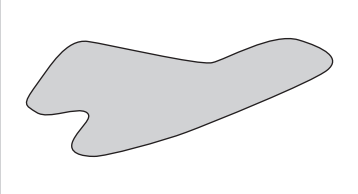
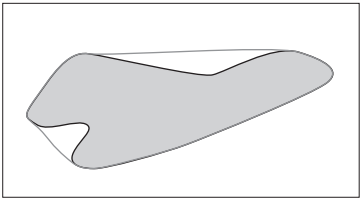
Use Module Image Blob to detect, to count and to sort objects. Associated white pixels in a binary image with certain criteria (e.g. area or convexity) belong to one object (blob).

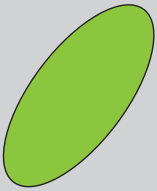
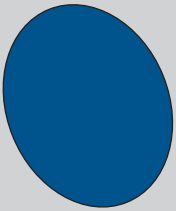
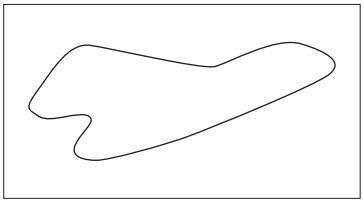
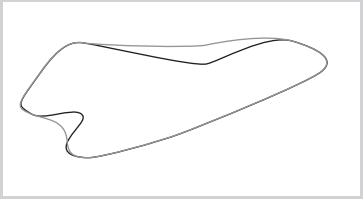
NOTE!
The following graphic shows various criteria from small to large .



Process Time [us]	Process time to run the module in μs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Blob True Count	Returns the number of found blobs (independent of parameter Blob Max Count).
Input Image	Link binary image (e.g. output image of Module Image Threshold or Module Image Threshold HSV) as input image of the module.

Calibration	Optionally link calibration as input to calculate in mm. In case of linked calibration, the following additional parameters appear: • Z Offset (available if Calibration Mode of input calibration is "Charuco on Device" or "Charuco from File"): Defines the height difference compared to the height defined in Module Image Calibration. Enter positive values, if the height for the module is bigger than for Module Image Calibration. Enter negative values, if the height for the module is smaller than for Module Image Calibration • Unit: Defines if values are in pixel and millimeter or only in pixel. **NOTE!** • Keep the height difference compared to the calibration height as small as possible for best accuracy. • For details about the calibration, see section "7.3.1 Module Image Calibration".
Blob Max Count	Defines the size of the Blob List (see Sub-Module).
Contour Mode	Defines the contour mode: • Outer: Finds only outer blobs (no blobs within blobs). • All: Finds all blobs (including blobs within blobs).
Sort Rule	Defines how to sort the results in the Blob List (see Sub-Module): • Off • Center of Gravity X (ascending/descending) • Center of Gravity Y (ascending/descending) • Area (ascending/descending) • Area Hull (ascending/descending) • Circularity (ascending/descending) • Convexity (ascending/descending) • Inertia (ascending/descending) • Perimeter (ascending/descending) • Perimeter Hull (ascending/descending)
Blob Bounding Box Orientation	Defines if the orientation of blobs is calculated or not. **NOTE!** The box enclosing the blob is defined by the maximum width and height of the blob. Width corresponds to the largest value and height to the smallest value of the blob.
Create Output Blob Image	Defines if binary output image is calculated containing all detected blobs (see Sub-Module).

Feature Area	If activated, the algorithm only returns blobs where the area is within the Minimum and Maximum values. Area is the number of white pixels belonging together. 
Feature Area Hull	If activated, the algorithm only returns blobs where the area hull is within the Minimum and Maximum values. Area Hull is the area of an imaginary band surrounding the blob. 
Feature Circularity	If activated, the algorithm only returns blobs where the circularity is within the Minimum and Maximum values. Blob circularity is defined as: $\frac{\text{Area}}{\text{Circumference}^2} \times 4\pi$ Circularity can only assume values within a range of 0 to 1. An ideal circle has a circularity of 1.
Feature Convexity	If activated, the algorithm only returns blobs where the convexity is within the Minimum and Maximum values. Convexity is the area divided by the envelope area. Convexity can only assume values within a range of 0 to 1. An ideal circle has a convexity of 1.

<table>
<tr><td>Feature Inertia</td><td>If activated, the algorithm only returns blobs where the inertia is with the Minimum and Maximum values.
Inertia is the inertial resistance of the blob to rotation about its principal axes. Find suitable values by testing for certain blobs.
Inertia can only assume values within a range of 0 to 1. An ideal circle has an Inertia of 1 and a line has an Inertia of 0.
<table><tr><th>Low value</th><th>High value</th></tr><tr><td>  </td><td>  </td></tr></table></td></tr>
<tr><td>Feature Perimeter</td><td>If activated, the algorithm only returns blobs where the perimeter is within the Minimum and Maximum values.
  </td></tr>
<tr><td>Feature Perimeter Hull</td><td>If activated, the algorithm only returns blobs where the perimeter hull is within the Minimum and Maximum values.
Perimeter Hull is the perimeter of an imaginary band surrounding the blob.
  </td></tr>
</table>

The Sub-Module Output Image returns a binary image containing all found blobs as white pixels. All other white pixels of the binary input image that do not fulfill the criteria for the blob analysis are black pixels in the output image. Black pixels (holes) within blobs in the binary input image are white (filled) in the output image. The image is only available if the parameter Create Output Blob Image is active.

The Sub-Module Blob List provides the results for all found blobs. For each result, the following outputs are calculated:

- Area: Returns the area of the blob.
- Area Hull: Returns the area hull of the blob.

- Circularity: Returns the circularity of the blob.
- Convexity: Returns the convexity of the blob.
- Inertia Ratio: Returns the inertia ratio of the blob.
- Perimeter: Returns the perimeter of the blob.
- Perimeter Hull: Returns the perimeter hull of the blob.
- Center of Gravity: Returns the center of gravity of the blob (including the x and y coordinates and the rotation).
- Match Box: Returns origin, width and height of Match Box.

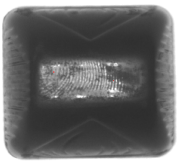
NOTE!
Depending on activated features, only certain results are available. Not available results are set to error.

7.4.4 Module Image Comparison

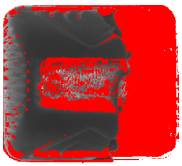
Use Module Image Comparison to compare images or regions of an image with a reference image. Teach in a reference image and the algorithm compares the following images to the reference. Use it e.g. to detect defects at an object.

NOTE!

- The algorithm divides the reference image into background and edges. The parameters Edge Sensitivity and Edge Broadening define edges (in contrast to the background).
- Deviations to the reference are display in the image scene with red pixels .



Reference object



Displayed deviation from the reference object

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Pixel Count	Returns the number of pixels with deviations from the current image to the reference image.
Input Image	Link 8 bit single channel image as input image of the module.
Input Region	Optionally, link input region to perform the image comparison at the tracked region.
Threshold Background	Defines the threshold for differences in the background brightness. The higher the value, the less sensitive to differences in background brightness.

Threshold Border	Defines the threshold for differences in brightness at edges (borders). Typically set to 255 (default) in order to accept brightness changes at edges (borders).
Edge Broadening [px]	Defines the width of the edges in pixels.
Teach	Teach current image as new reference image. **NOTE!** Changing the size of the input region require to teach again.
Edge Sensitivity [%]	Defines the percentage of detected edges evaluated as edges for the reference image. Typical value of 20% is suitable for most applications.

The Sub-Module Output Image provides a binary image containing all pixels with deviations to the reference as white pixels.
The Sub-Module Reference Image shows the taught in image.
The Sub-Module Threshold Image shows the threshold value for each pixel relevant for the comparison. It shows the pixels belonging to the background and the pixels belonging to edges.

7.4.5 Module Image Measure

Use Module Image Measure to find lines, circles or arcs or to find segments on lines, circles or arcs. With the found elements, it is possible to measure angles, distances or to find special points.

Process Time [us]	Process time to run the module in μs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Input Image	Link 8 bit single channel image as input image of the module.
Coordinate System	Optionally, link coordinate system as input to move all search geometries with the input coordinate system.
Calibration	Optionally, link calibration as input to calculate in mm. In case of linked calibration, the following additional parameters appear: • Z Offset (available if Calibration Mode of input calibration is “Charuco on Device” or “Charuco from File”): Defines the height difference compared to the height defined in Module Image Calibration. Enter positive values, if the height for the module is bigger than for Module Image Calibration. Enter negative values, if the height for the module is smaller than for Module Image Calibration. • Unit: Defines if values are in pixel and millimeter or only in pixel. **NOTE!** • Keep the height difference compared to the calibration height as small as possible for best accuracy. • For details about the calibration, see chapter “7.3.1 Module Image Calibration”.

The Sub-Module Set contains all tools added to the module from the Module Toolbar.

	Find Point (Point) Find Point (Coordinates)	Add a fix point or link the point with a job result. Add a fix point or link the coordinates of a point with job results.
	Find Line	Find a line via edges on the search rays.
	Find Arc	Find an arc via edges on the search rays.
	Find Circle (Two Points) Find Circle (Three Points)	Find a circle via edges on the search rays.
	Find Segments on Circle	Find segments on a circle via edges on the search ray.
	Find Segments on Line	Find segments on a line via edges on the search ray.
	Find Segments on Arc	Find segments on an arc via edges on the search ray.
	Distance Measurement	Measure the distance between two geometries (e.g. point or line).
	Intersection Point Detection	Intersect two lines.
	Property of Geometry	Find special points on geometries (e.g. center of surface).

NOTE!
Activate the tool in the Module Toolbar and draw it in the scene. Depending on the type of the tool, different properties appear.

Find Point Tool (Point)

- Found Point: Returns the found point and the coordinates of the found point.
- Input Point: Define a fix point or link the point to a job result.

Find Point Tool (Coordinates)

- Point Coordinates: Returns the found point and the coordinates of the found point.
- Input Point Coordinates: Define a fix point or link the coordinates of the point to job results.

Find Line, Find Arc or Find Circle

The algorithm finds edge points (at gray value transitions) on the search rays that are perpendicular to the search geometry. It then approximates the geometry through the edge points in an iterative best-fit approach.

- Quality of Fit [%]: Returns quality info for the best fit approach. The higher the result, the better the fit.
- Edge Polarity: Defines the polarity of the edge.
  - » Either: Finds edges at transitions from bright to dark or from dark to bright along the search rays.
  - » Bright to Dark: Finds edges only at transitions from bright to dark along the search rays.
  - » Dark to Bright: Finds edges only at transitions from dark to bright along the search rays.
- Find by: Define the relevant edge result.
  - » Best Score: Uses the result with the highest score as edge point on the search rays.
  - » First Score: Uses the first result as edge point on the search rays.
  - » Last Score: Uses the last result as edge point on the search rays.
- Edge Width [px]: Defines the edge width. The higher the value, the bigger the smoothing effect on the gray-values along the search rays.
- Threshold Gradient Pos [GrM]: Defines the minimum edge sensitivity for edge points from dark to bright along the search rays.
- Threshold Gradient Neg [GrM]: Defines the minimum edge sensitivity for edge points from bright to dark along the search rays.
- Threshold Outlier Distance [px]: Defines the width of the tube (the width is two times the Threshold Outlier Distance) around the found geometry for edge points to be considered as valid and used for further fitting iterations.
- Fitting Iterations: Defines the number of fitting iterations applied to optimize the result of the found geometry.
- Search Ray Length [px]: Defines the length of the search rays.
- Search Ray Interval [px]: Defines the width between the search rays.
- Search Ray Orientation: Defines the orientation of the search ray (default or swap).
- Points to Use [%]: Defines the percentage of edge points on the search rays for the first fitting iteration.
- Points to Use Strategy: Defines if the first or the last edge points on the search rays are used for the first fitting iteration (if parameter Points to Use [%] is smaller than 100 %).
- Fit Maximal Geometry: Defines if finding the maximal geometry is active or not (only available for Find Line and Find Arc). Starting from the center of the found geometry, the algorithm checks in both directions for gaps and outliers in a row. If set to “On”, the following additional parameters appear.
  - » Maximal Gap Between Valid Points: Defines the maximal gap between valid points. In case of bigger gaps, the last valid point projected on the found geometry defines the start or end point of the geometry.
  - » Maximal Outliers In a Row: Defines the maximal number of outliers in a row. In case of more outliers in a row, the last valid point projected on the found geometry defines the start or end point of the geometry.

The tool returns the following results:

- For Find Line:
  - » Point 1: Returns the start point.
  - » Point 2: Returns the end point.
  - » Midpoint: Returns the midpoint.
  - » Length: Returns the length of the line.
  - » Angle: Returns the angle from the search geometry to the found geometry (positive counterclockwise)
- For Find Arc:
  - » Diameter: Returns the diameter (only available in pixel).
  - » Angle Start: Returns the start angle of the found geometry (depending on the input coordinate system; positive clockwise)
  - » Angle Span: Returns the span angle of the found geometry (depending on the input coordinate system; positive clockwise).
  - » Start, Middle and End of Arc: Returns the start, middle and end point of the found arc.
  - » Arc Length: Returns the length of the arc (only available in pixel).
  - » Angle: Returns the angle from the search geometry to the detected geometry (positive clockwise). The orientation for arcs is from the midpoint of the arc to the center on the arc.
  - » Center: Returns the center point of the found geometry.
- For Find Circle:
  - » Diameter: Returns the diameter (only available in pixel).
  - » Center: Returns the center point of the found geometry.

Find Segments on Line, Arc or Circle

The algorithm finds edge points (at gray value transitions) and corresponding segments on the search ray.

- Segments True Count: Returns the number of found segments on the geometry (independent of parameter Segment Max Count).
- Edge Width [px]: Defines the edge width. The higher the value, the bigger the smoothing effect on the gray-values along the search ray.
- Threshold Gradient Pos [GrM]: Defines the minimum edge sensitivity for edge points from dark to bright along the search ray.
- Threshold Gradient Neg [GrM]: Defines the minimum edge sensitivity for edge points from bright to dark along the search ray.
- Segments Max Count: Defines the size of the Segment List (see Sub-Module).
- Segments Minimal Length: Defines minimum length of segments.
- Segments Maximal Length: Defines maximum length of segments.
- Sort Rule: Defines how to sort the results in the Segment List (see Sub-Module):
  - » Position on Search Geometry
  - » Size [Longest First]
  - » Size [Shortest First]
- Segment Brightness: Defines if the segments on the search ray are bright or dark.
- Orientation: Defines the orientation of the search ray (default or swap).

The tool returns the following results (for each segment of the Segment List):

- For Find Segments on Line:
  - » Point 1: Returns the start point.
  - » Point 2: Returns the end point.
  - » Midpoint: Returns the midpoint.
  - » Length: Returns the length of the segment.
- For Find Segments on Arc or Circle:
  - » Diameter: Returns the diameter (only available in pixel).
  - » Angle Start: Returns the start angle of the found geometry (depending on the input coordinate system; positive clockwise)
  - » Angle Span: Returns the span angle of the found geometry (depending on the input coordinate system; positive clockwise)
  - » Start, Middle and End of Arc: Returns the start, middle and end point of the found arc.
  - » Arc Length: Returns the length of the arc (only available in mm).
  - » Center: Returns the center point of the found arc.

Measure Distance

- Output Distance: Returns the distance.
- Calculation Method: Define the calculation method.
  - » Geometrical Distance: Measure geometrical (shortest) distance. E.g. relevant if measuring from point perpendicular to a line.
  - » Center to Center: Measure from center point to center point. E.g. relevant if measuring from point to center point of a line.
- Output Geometry with Point 1 and Point 2: Returns the coordinates of point 1 and point 2.

Intersection Point Detection

- Output Intersection Point: Returns the coordinates of the intersection point.
- Output Angle: Returns the angle from the first to the second line (positive clockwise). The orientation of lines is from the start point to the end point of the line.

Property of Geometry

- Output Point: Returns the coordinates of output point.
- Type of Property: Defines the type of property.
  - » Center of Surface: Find the center of the surface (e.g. on arcs or circles).
  - » Start of Surface: Find the start of the surface.
  - » End of Surface: Find the end of the surface.
  - » Highest Point: Find the highest point of a geometry (e.g. arc or circle).
  - » Lowest Point: Find the lowest point of a geometry (e.g. arc or circle).
  - » Leftmost Point: Find the leftmost point of a geometry (e.g. arc or circle).
  - » Rightmost Point: Find the rightmost point of a geometry (e.g. arc or circle).

7.4.6 Module Image OCR

Use Module Image OCR to read letters, numbers or symbols.

NOTE!
The OCR algorithm works best if characters have a height of 25 pixels and if the width of the gap between characters is half the width of characters. Furthermore, the background should be as homogeneous as possible without noise.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Segments True Count	Returns the number of found segments (independent of parameter Segment Max Count).
Reading Result	Returns the reading result. In case of reading characters in several rows, the reading results of several rows are separated from each other by the line feed symbol.
Input Image	Link 8 bit single channel image as input image of the module.
Coordinate System	Optionally, link a coordinate system to move the Search Box with the coordinate system if the position of the characters is not fix.
Read Timeout [us]	Defines the maximum time the algorithm searches for characters. If searching is not finished until the timeout, all results of the module are set to error. **NOTE!** The process time of the module can take slightly longer than the parameter Read Timeout.
Segment Max Count	Defines the size of the Segment List (see Sub-Module).

The Sub-Module Result List contains the reading result separately for each row. The number of reading results depends on the settings of the Sub-Module Row Find.

The Sub-Module Segment List contains a list of all segments. The number of segments in the list depends on the parameter Segments Max Count. For each segment, the following results are available:

- Assigned Character: Returns the character for the segment. In case of no trained characters or not finding a suitable character, the algorithm returns the value defined by the parameter Substitution Character (default: ?).
- Lower Threshold: Returns the lower threshold value used by the algorithm.
- Upper Threshold: Returns the upper threshold value used by the algorithm.
- Height [px]: Returns the height of the segment.
- Width [px]: Returns the width of the segment.
- Score: Returns the score value of the character.

The Sub-Module Search Box defines the region where character are expected.

NOTE!

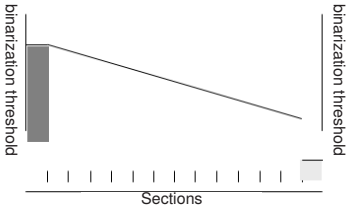
- If the position of characters is not fix, link a coordinate system to move the position of the search box together with the coordinate system.
- Make sure that the search box is only as big as necessary to avoid finding wrong results.
- Ensure that the search area is always visible completely within the image.

The Sub-Module Row Find defines if the algorithm searches for characters in one or several lines.

- Angle [deg]: Returns the angle between the search box and the rows.
- Row True Count: Returns the number of found rows.
- Row Recognition: Defines if the algorithm searches in one or several lines.
  - » Off: The algorithm searches for characters only in one row (default). No further parameters or results appear at the Sub-Module Row Find.
  - » Standard: Search for characters in specific number of rows. Further parameters at Sub-Module Row Find appear.
- Row Max Count: Defines the number of rows. The number of rows specified in the parameter must be equivalent to the real number of rows – otherwise the algorithm does not find any characters.
- Angle Range [deg]: Defines the expected angle between the search box and the rows.
- Row Height Min [px]: Defines the minimum height of the rows.
- Row Height Max [px]: Defines the maximum height of the rows.
- Row Space Min [px]: Defines the minimum space between rows.

The Sub-Module Binarization defines the binarization of the image.

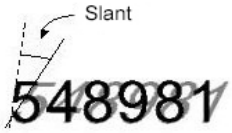
- Contrast: Defines the contrast.
  - » Bright on Dark: The algorithm searches for bright characters on dark background.
  - » Dark on Bright: The algorithm searches for dark characters on bright background.
- Threshold Mode: Define the threshold mode for binarization.
  - » Manual: Define values for Manual Lower Threshold Value and Manual Upper Threshold Value manually.
  - » Computed: The algorithm computes the thresholds automatically.
  - » Linear: The algorithm automatically adjusts the thresholds in a linear way along the characters. Helpful in case of linear changing light conditions along the characters.



» Nonlinear: The algorithm divides the search box in several parts defined by the parameter “Linear/ Nonlinear: Threshold Division”. It calculates the thresholds independently for each division. Helpful in case of not homogeneously illuminated images.

The Sub-Module Segmentation defines how to separate the single segments.

- Character Height Min [px]: Defines the minimum height of character.
- Character Height Max [px]: Defines the maximum height of characters.
- Character Width Min [px]: Defines the minimum width of characters.
- Character Width Max [px]: Defines the maximum width of characters.
- Cluster Size Min [px]: Defines the minimum cluster size of characters.
- Cluster Size Max [px]: Defines the maximum cluster size of characters.
- Discard Undersized: Define if the algorithm ignores undersized characters (defined by min parameters) or not.
- Discard Oversized: Defines if the algorithm ignores oversized characters (defined by max parameters) or not.
- Dot Space Vertical [px]: Defines vertical dot space.
- Dot Space Horizontal [px]: Defines horizontal dot space.
- Splitting: Define the splitting
  - » Default: Assumes fix distribution of characters (fix values for Character Space and angle).
  - » Variable: Assumes variable distribution of characters (variable values for Character Space and angle).
  - » Dynamic: Assumes dynamic distribution of characters.
- Character Space [px]: Defines the space between characters.
- Substitution Character: Defines the substitution character that is used in case of no trained characters or if the algorithm does not find characters for segments.
- De-Slanting Mode: Defines tilting of symbols.
  - » Fix: Use fix inclination angle with additional parameter De-Slanting Angle [deg]. The angle defines the inclination of characters in reference to the orientation of the row.



» Automatic: Calculate inclination angle automatically.

NOTE!
The following example shows the definition for the property.

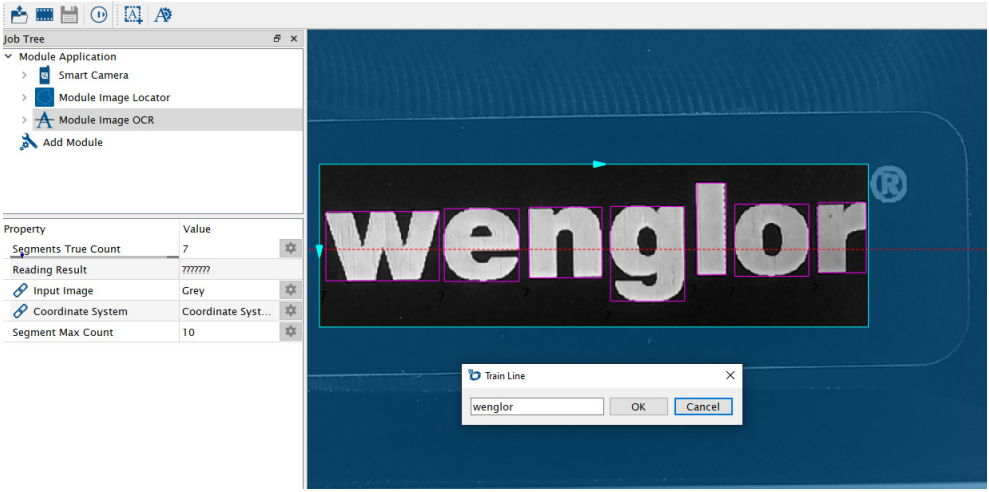
The Sub-Module Classification contains the parameters for classification:

- Acceptance Level: Defines the minimum acceptance level for characters.

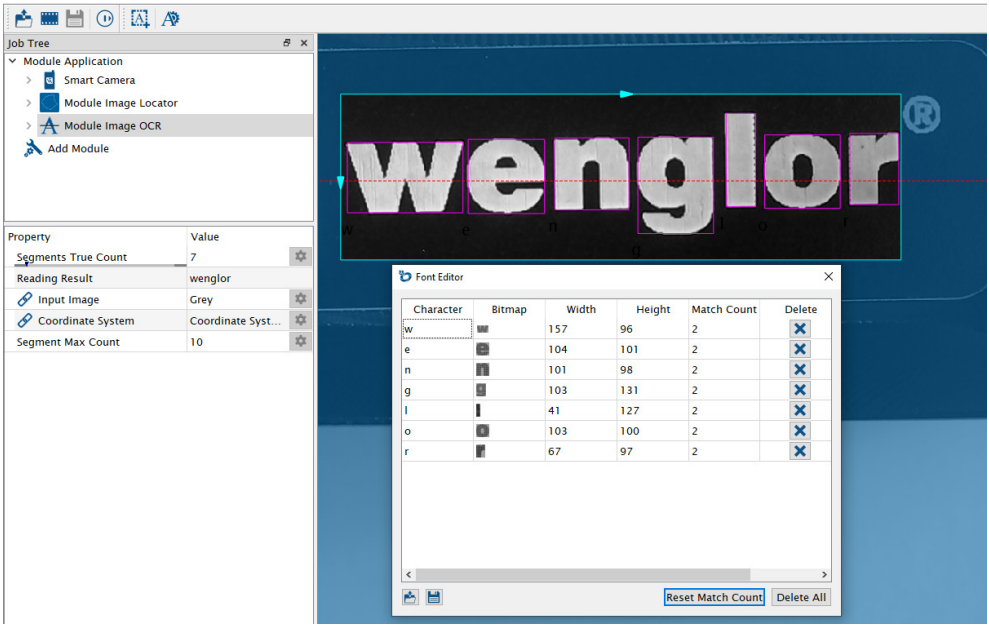
The Sub-Module Fielding contains the patterns. The following patterns are available:

- N: 0123456789
- A: ABCDEFGHIJKLMNOPQRSTUVWXYZ
- a: abcdefghijklmnopqrstuvwxyz
- H: 0123456789ABCDEF
- h: 0123456789abcdef
- O: 01234567
- Add new: Add further patterns.

Module Toolbar
Use the Module Toolbar in order to train the line. Click on the icon “Train Line”, enter the characters for the complete line and press Ok.



Open the Fond Editor in the Module Toolbar to see all trained characters. It is possible to delete single or all characters. Save the complete character set to load it in other jobs. The standard font types OCR-A and OCR-B are available on the Machine Vision Devices.



Training single characters is also possible via clicking on a specific character in the image and by assigning the character to it.

NOTE!
The module only supports training of single characters and complete lines, if using only one single row (not supported for several rows). To train the characters for several rows, reduce the search box to find only one row and set the parameter Row Recognition to “Off”. Train the characters and then set it back to several rows.

7.4.7 Module Image Code 1D

Use Module Image Code 1D to read 1D codes.

NOTE!
Module Image Code 1D uses HALCON algorithms of the company MVTec. For details, check the operating instructions of HALCON.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Result True Count	Returns the number of found codes (maximum value of Result True Count depends on parameter Result Max Count).
Input Image	Link 8 bit single channel image as input image of the module.
Calibration	Optionally link calibration as input to calculate in mm. In case of linked calibration, the following additional parameters appear: • Z Offset (available if Calibration Mode of input calibration is "Charuco on Device" or "Charuco from File"): Defines the height difference compared to the height defined in Module Image Calibration. Enter positive values, if the height for the module is bigger than for Module Image Calibration. Enter negative values, if the height for the module is smaller than for Module Image Calibration • Unit: Defines if values are in pixel and millimeter or only in pixel. **NOTE!** • Keep the height difference compared to the calibration height as small as possible for best accuracy. • For details about the calibration, see section "7.3.1 Module Image Calibration".
Read Timeout [us]	Defines the maximum time the algorithm searches for code. If searching is not finished until the timeout, all results of the module are set to error. **NOTE!** The process time of the module can take slightly longer than Read Timeout.
Result Max Count	Defines the size of the Result List (see Sub-Module) and the maximum value for Result True Count.
Sort Rule	Defines how to sort the results in the Result List (see Sub-Module): • Reading (ascending/descending) • Center X (ascending/descending) • Center Y (ascending/descending) • Area (ascending/descending)

Code Type	Defines the code type: • Code 39 • Code 128 • 2/5 Industrial • 2/5 Interleaved • Codabar • EAN-13 • EAN-13 Add-On 2 • EAN-13 Add-On 5 • EAN-8 • EAN-8 Add-On 2 • EAN-8 Add-On 5 • UPC-A • UPC-A Add-On 2 • UPC-A Add-On 5 • UPC-E • UPC-E Add-On 2 • UPC-E Add-On 5 • Code 93 • MSI • PharmaCode • GS1-128 • GS1 DataBar Omnidir • GS1 DataBar Truncated • GS1 DataBar Stacked • GS1 DataBar Stacked Omnidir • GS1 DataBar Limited • GS1 DataBar Expanded • GS1 DataBar Expanded Stacked • Auto  **NOTE!** Use option “auto” only to identify the code type. Once the code type is familiar, select it accordingly to optimize the performance.
Quality Grading	Define the quality grading: • None: Quality grading deactivated (fast) • Quality ISO/IEC 15416: Additional quality grading results are calculated for each result (see Sub-Module Result List).

The Sub-Module Result List provides the results for all found codes. For each result, the following outputs are calculated:

- Reading: Returns the code content
- Quality: Returns the quality results of the code (in case of activated quality grading)
- Coordinate System: Returns the position of the found code (including x and y coordinates and rotation). It is the center of the found code.

NOTE!

- Depending on code type and quality grading, different quality grading results are available. Not calculated results are set to error.
- For details about quality grading, check the code standards and the operating instruction of HALCON.

The Sub-Module Search Box defines the area where the algorithm looks for codes. The Search Box is always at a fix position (not linkable).

The Sub-Module Enhanced Parameters contains additional parameters depending on the code type (for experts). For details, see operating instruction of HALCON.

7.4.8 Module Image Code 2D

Use Module Image Code 2D to read 2D codes.

NOTE!
Module Image Code 2D uses HALCON algorithms of the company MVTec. For details, check the operating instructions of HALCON.

Process Time [us]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Result True Count	Returns the number of found codes (maximum value of Result True Count depends on parameter Result Max Count).
Input Image	Link 8 bit single channel image as input image of the module.
Teach	Teach current 2D code to optimize the algorithm settings. Requires the 2D code to be present in the current image. Teach manually or link it to job result.
Calibration	Optionally link calibration as input to calculate in mm. In case of linked calibration, the following additional parameters appear: • Z Offset (available if Calibration Mode of input calibration is "Charuco on Device" or "Charuco from File"): Defines the height difference compared to the height defined in Module Image Calibration. Enter positive values, if the height for the module is bigger than for Module Image Calibration. Enter negative values, if the height for the module is smaller than for Module Image Calibration • Unit: Defines if values are in pixel and millimeter or only in pixel. **NOTE!** • Keep the height difference compared to the calibration height as small as possible for best accuracy. • For details about the calibration, see section "7.3.1 Module Image Calibration".

Read Timeout [us]	Defines the maximum time the algorithm searches for code. If searching is not finished until the timeout, all results of the module are set to error. **NOTE!** The process time of the module can take slightly longer than Read Timeout.
Result Max Count	Defines the size of the Result List (see Sub-Module) and the maximum value for Result True Count.
Sort Rule	Defines how to sort the results in the Result List (see Sub-Module): • Reading (ascending/descending) • Center X (ascending/descending) • Center Y (ascending/descending) • Area (ascending/descending)
Code Type	Defines the code type: • Data Matrix ECC 200 • QR Code • Micro QR Code • PDF417 • Aztec Code • DotCode • GS1 DataMatrix • GS1 QR Code • GS1 Aztec Code • GS1 DotCode
Recognition	Defines the recognition mode (Standard, Enhanced or Maximum). **NOTE!** Increase the recognition mode from Standard to Enhanced or Maximum to find difficult codes with poor quality. This increases also the process time.
Quality Grading	Define the quality grading: • None: Quality grading deactivated (fast) • Quality ISO/IEC 15415 • Quality ISO/IEC TR 29158 • Quality Semi T10 Values **NOTE!** Additional quality grading results are calculated for each result (see Sub-Module Result List) in case of active quality grading.

The Sub-Module Result List provides the results for all found codes. For each result, the following outputs are calculated:

- Reading: Returns the code content.
- Quality: Returns the quality results of the code (in case of activated quality grading).
- Coordinate System: Returns the position of the found code (including x and y coordinates and rotation). It is the center of the found code.

NOTE!

- Depending on code type and quality grading, different quality grading results are available. Not calculated results are set to error.
- For details about quality grading, check the code standards and the operating instruction of HALCON.

The Sub-Module Search Box defines the area where the algorithm looks for codes. The Search Box is always at a fix position (not linkable).

The Sub-Module Enhanced Parameters contains additional parameters depending on the code type (for experts). For details, see operating instruction of HALCON.

7.5 Module HALCON Script

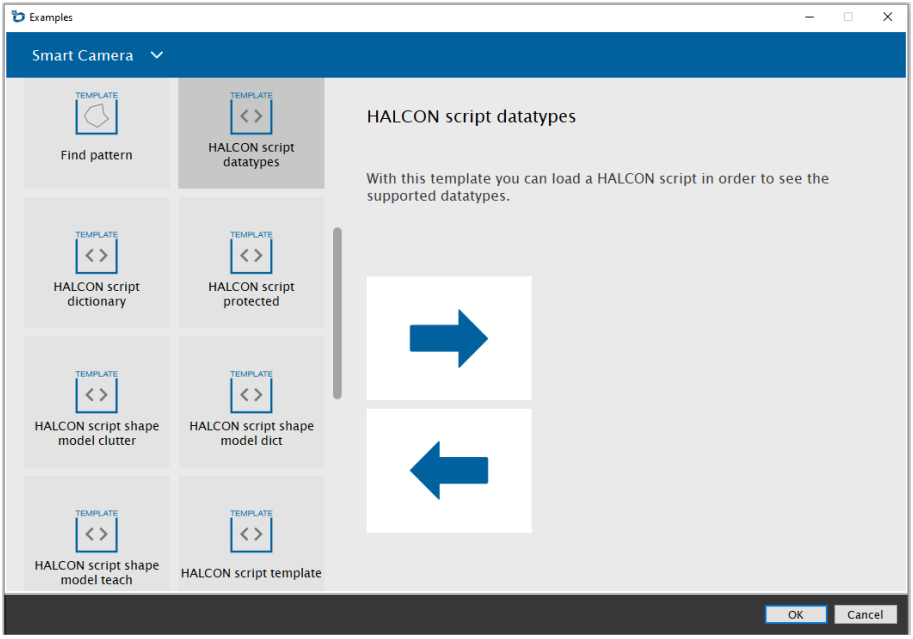
Module HALCON script enables to load HALCON scripts created with the software HDevelop from the company MVTec. Licenses for HDevelop are available by the sales partner of MVTec (not by wenglor).

NOTE!

- The Machine Vision Devices work with HALCON steady version 22.11. It is possible to create HALCON scripts with the progress version or other HALCON versions, but it is necessary to check the compatibility notes of the HALCON library versions.
- The license of Module HALCON Script supports the HALCON modules Foundation, Calibration, 1D-Metrology, 2D-Metrology, Bar Code, Data Code, OCR/OCV and Matching.

Templates within the software wenglor uniVision 3 contain also HALCON script examples to get familiar with the supported datatypes and example applications like blob analysis, code reading or shape based matching:

- Machine Vision Devices: Connect the software wenglor uniVision 3 to the Processing Instance of the Machine Vision Device (see section “6.1 Connecting to Machine Vision Device”) and open one of the HALCON templates.
- Offline Windows Simulator: Open an offline example with the software wenglor uniVision 3 (see section “6.2 uniVision Simulator”).



NOTE!
The HALCON script examples are also available on https://www.wenglor.com/product/DNNF023 → Downloads → Soft- and Firmware.

The following HALCON script examples are available:

- 00_init_run_finalize_dictionary: Contains minimum HALCON script requirements.
- 01_read_recorded_images: Reads images of uniVision Teach+ file (see section "6.4 Teach+"). Available within uniVision template "HALCON script template".
- 02_conversion_region_to_xldcont: Copies input region to output xldcont.
- 02_conversion_region_to_xldpoly: Copies input region to output xldpoly.
- 02_conversion_xldpoly¬_to_region: Copies input xldpoly to output region.
- 02_supported_datatypes: Shows all supported datatypes and copies all inputs to outputs. Available within uniVision template "HALCON script datatypes".
- 03_blob: Does blob analysis.
- 04_datacode: Reads datamatrix code.
- 05_shape_model_clutter: Teaches and finds shape model with clutter region. Available within uniVision template "HALCON script shape model clutter".
- 05_shape_model_dictionary: Shows how to save and load a shape model from a HALCON dictionary file (see separate dictionary file "shapemodel.hdict"). Available within uniVision template "HALCON script shape model dictionary".

- 05_shape_model_teach: Teaches and finds shape model. Available within uniVision template “HALCON script shape model teach”.
- 06_protect_script_with_hostid: Shows how to password protect procedures in HDevelop (Default password: admin) and how to limit the script to one specific device via the Host ID in the procedure Init. Available within uniVision template “HALCON script protected”.
- 07_multi_image_dictionary: Shows how to store data (e.g. images) in the dictionary in order to use several historic data together, e.g. to combine the last x images. Available within uniVision template “HALCON script dictionary”.

Recommended workflow for creating HALCON scripts

- Record Teach+ file with real data (see section “6.4 Teach+”).
- Create HALCON script in HDevelop and use the previously recorded real word data.

NOTE!
Use the HALCON script example “01_read_recorded_images” in order to extract the images of the Teach+ file and to create and test the HALCON script file.

- Save the finished HALCON script to the file system.
- Import the HALCON script file in Module HALCON script of the software wenglor uniVision 3 to process real data on the Machine Vision Device.

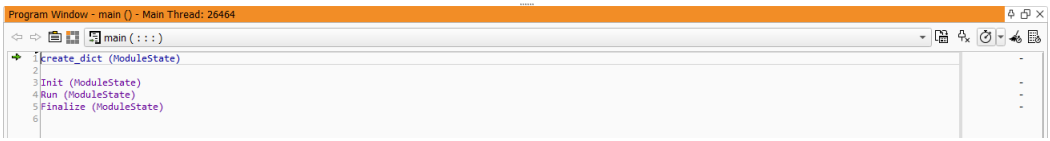
7.5.1 Basics in HDevelop

Module HALCON Script in the software wenglor uniVision 3 expects the following HALCON procedures:

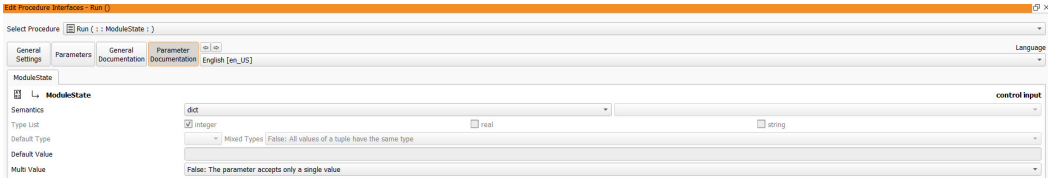
- Init (): Procedure is called when adding a HALCON script to Module HALCON Script or when loading a uniVision job including a HALCON script (e.g. to create a data code 2d model).
- Run (): Procedure is called with every data evaluation.
- Finalize (): Procedure is called when switching to another job, when loading another HALCON script or when deleting Module HALCON Script from the job tree.

All three procedures must be called within the main procedure. Everything else within the main procedure (e.g. to initialize variables) is ignored by Module HALCON Script (except the creation of the dictionary with the name ModuleState).
Calling further procedures within the procedures Init, Run or Finalize and protecting procedures with a password within the software HDevelop is supported. Even limiting a script to one specific device is possible via the host ID (see HALCON example 06_protect_script_with_hostid). If the host ID is not equal, an error message shows the host ID of the device in Module HALCON Script. Furthermore, the Host ID of the Machine Vision Device is also shown on the dashboard of the device website.

The procedures Init, Run and Finalize support exclusively one dictionary called “ModuleState” to store and exchange data within procedures. If the dictionary “ModuleState” is used, then it is necessary to create it within the procedure main and it is necessary to use it in all Init, Run and Finalize procedures. No additional or differently named dictionary is supported.



Make sure to set the “Semantics” of “ModuleState” to “dict”. Dictionaries do not support Multi Value (only option FALSE).

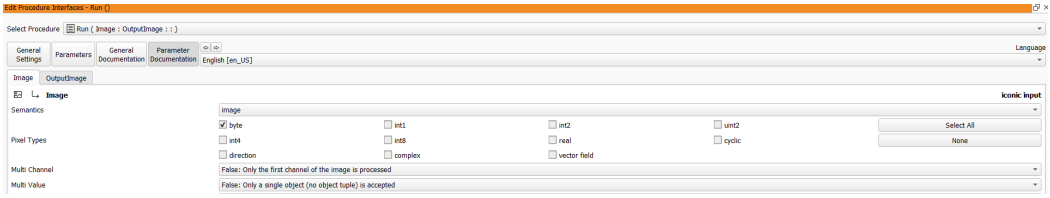


Using dictionaries

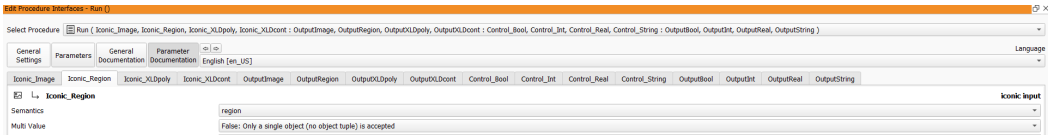
- Use the dictionary to exchange data between the different procedures (e.g. from Init to Run Procedure) so that e.g. the 2D code model is only created once when loading the uniVision job (see HALCON example 04_datacode).
- It is also possible to store data (e.g. shape models) in a permanent way in the dictionary once the script was loaded in Module HALCON Script of the Software wenglor uniVision 3 (see HALCON example 05_ shape_model_teach).

Only the procedure "Run" additionally supports the following data types

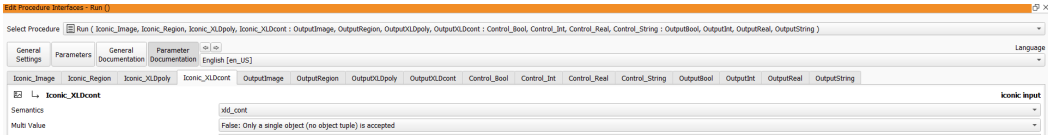
- Iconic parameters:
  - » Semantics “image”
    - Only “Pixel Types” “byte”
    - “Multi Channel” “True” for color images and “False” for monochrome images
    - “Multi Value” (tuple) with only option FALSE.



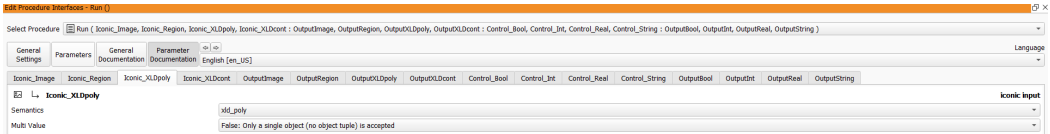
» Semantics “region”
- “Multi Value” (tuple) with only option FALSE



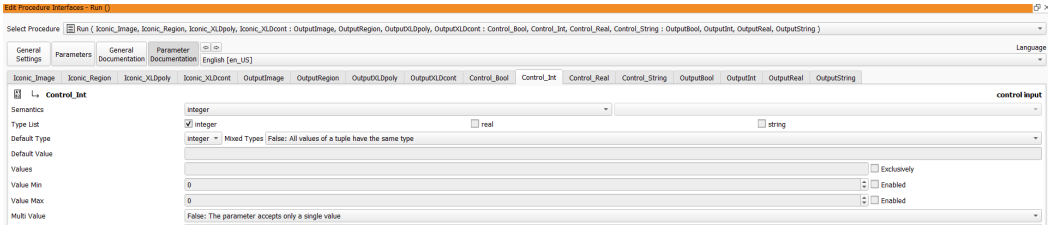
» Semantics “xld_cont”
- “Multi Value” (tuple) with only option FALSE



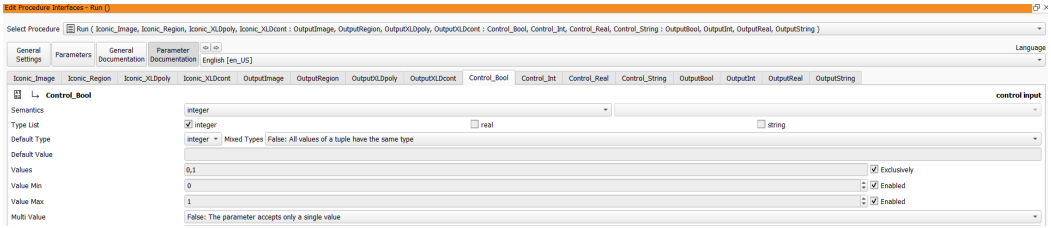
» Semantics “xld_poly”
- “Multi Value” (tuple) with only option FALSE



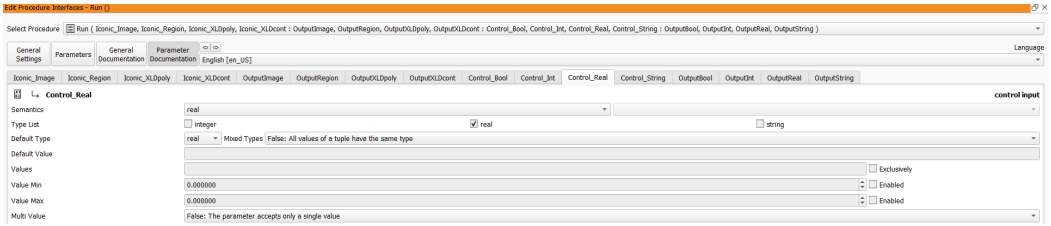
- Control parameters:
  » Semantics “integer”
  - “Mixed Types” with only option FALSE
  - Value Min/Max (if enabled) are considered if using as inputs for Module HALCON Script.
  - “Multi Value” (tuple) with only option FALSE.



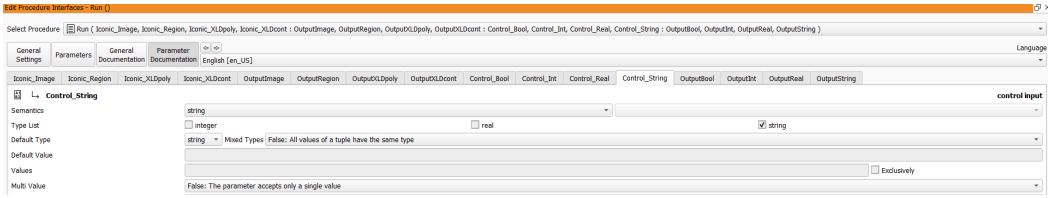
» Semantics “integer” with “Values” “0,1” (exclusively), “Value Min” “0” (enabled) and “Value Max” “1” (enabled) to work with Boolean data types as inputs or outputs for Module HALCON Script.
  - “Mixed Types” with only option FALSE
  - “Multi Value” (tuple) with only option FALSE.



» Semantics “real”
  - “Mixed Types” with only option FALSE
  - Value Min/Max (if enabled) are considered if using as inputs for Module HALCON Script.
  - “Multi Value” (tuple) with only option FALSE.



» Semantics “string”
  - “Mixed Types” with only option FALSE
  - “Multi Value” (tuple) with only option FALSE.

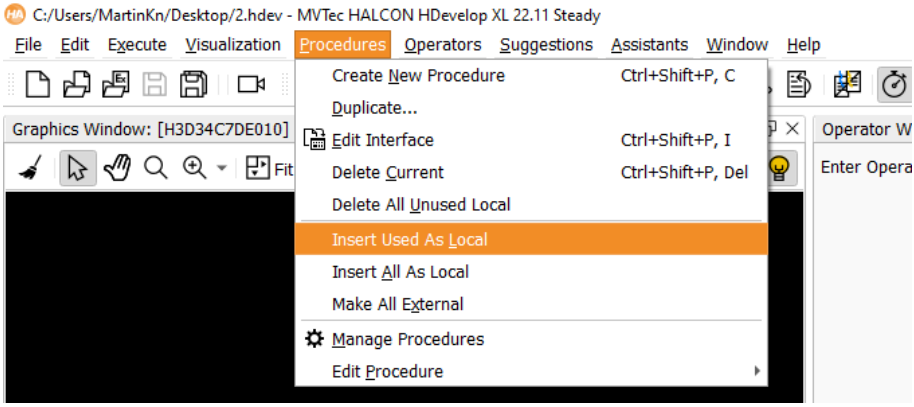


NOTE!

- It is necessary to set the “Semantics” for each parameter in the software HDevelop so that Module HALCON Script in the software wenglor uniVision 3 detects the parameters.
- The parameters of the “run” procedure will appear within Module HALCON Script as inputs and outputs.

Store the HALCON script as *.hdev file with all external procedures inserted as local procedures so that all relevant info is available in one HALCON script without platform dependencies.

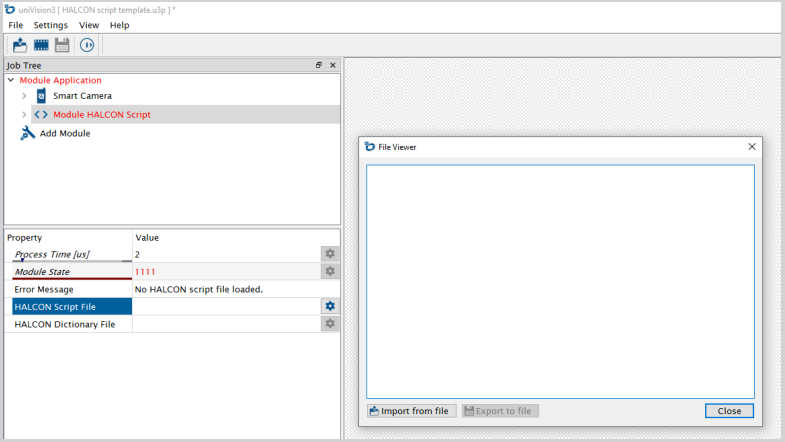
NOTE!
If working with many external procedures, store the HALCON script file once with links to external procedures and once for the export to Machine Vision Devices with a separate name and all procedures added as local procedures. Then it is possible to handle changes in external procedures in an easy way by only inserting the external procedures again as local procedures and by saving the scripts again.

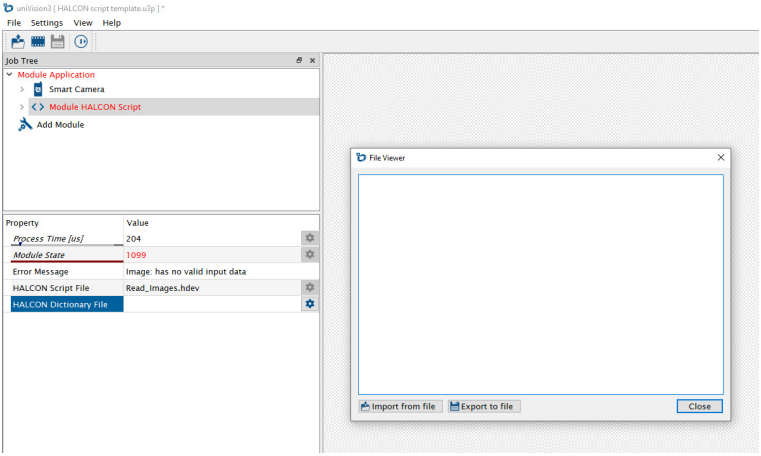


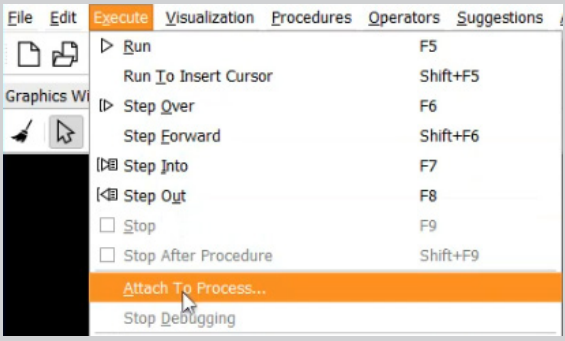
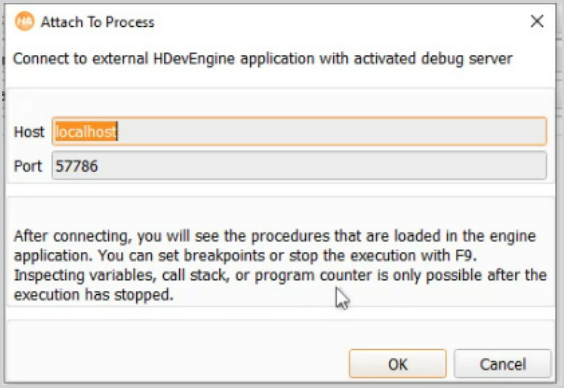
All HALCON operators can be used except not supported (deactivated) operators (see section “12.2 Disabled HALCON Operators”).

NOTE!
In general, read and write operations on the file system and canvas (GUI) actions are not supported because of the platform independent implementation on all Machine Vision Devices. Furthermore, HALCON interface operators are disabled because the Machine Vision Device interfaces can be used directly as separate modules within the uniVision job.

7.5.2 Basics in Module HALCON Script

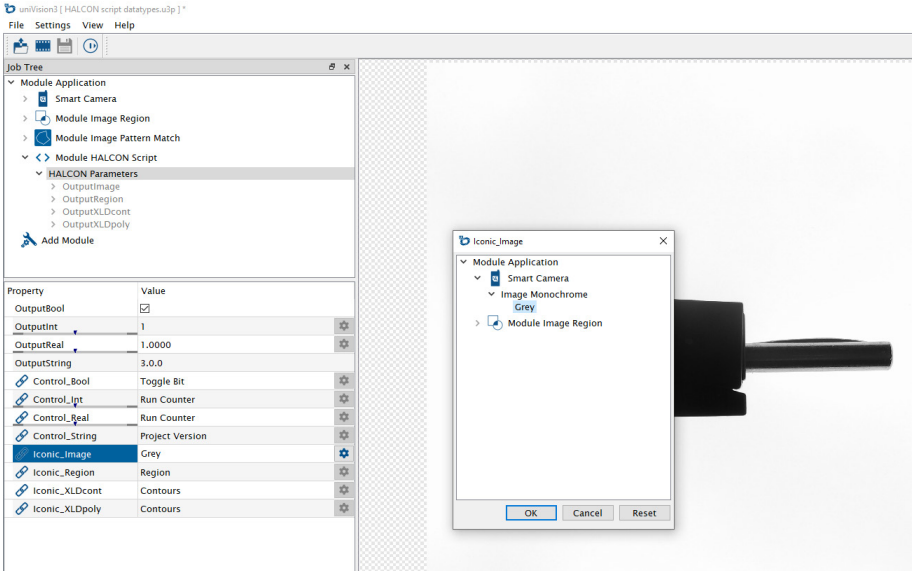
Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Error Message	Shows additional error messages (e.g. No HALCON script file loaded; Image: has no valid input data) and HALCON exceptions.
HALCON Debug Server Port	Shows the port used for the HALCON debug server (read-only port 57786). Only available if parameter HALCON Debug Server is activated in offline uniVision jobs (in uniVision Simulator).
HALCON Script File	Click on the settings icon at HALCON Script File to import a HALCON script file (*.hdev). In case of an already imported script, a preview of the script is shown and it is possible to export the HALCON script in order to open it again in the software HDevelop. 

HALCON Dictionary File	Click on the settings icon at HALCON Dictionary File to optionally import a separate HALCON Dictionary File (*.hdict). The dictionary file can e.g. contain a shape model created in the software HDevelop. It is also possible to export the HALCON Dictionary File. 

HALCON Debug Server	Defines if HALCON debug server is activated or not. Only available in uniVision offline jobs (uniVision Simulator). Not supported on real Machine Vision Devices. Open the software HDevelop and click in menu bar "Execute" on "Attach to Process...".  Select Host "localhost" and default port 57786 and press Ok to connect to the HALCON debug server.  Use the available HALCON debug options.

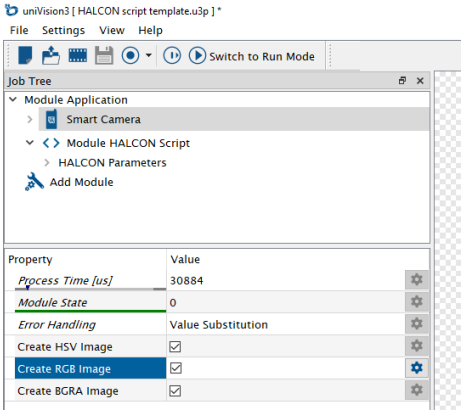
Sub-Module "HALCON Parameters"

The parameters of the procedure Run defined in the software HDevelop are shown at HALCON Parameters of Module HALCON Script. Link all input images, regions, xldconts and xldpolys so that the HALCON script runs without errors.

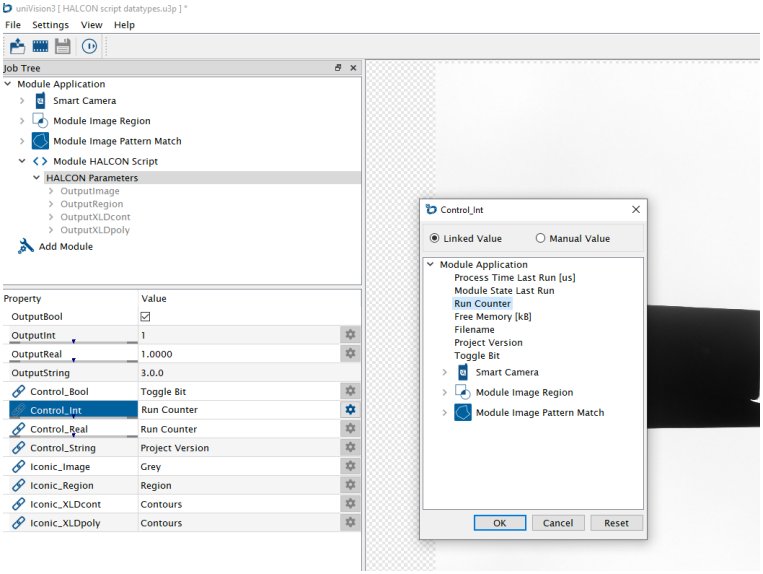


NOTE!

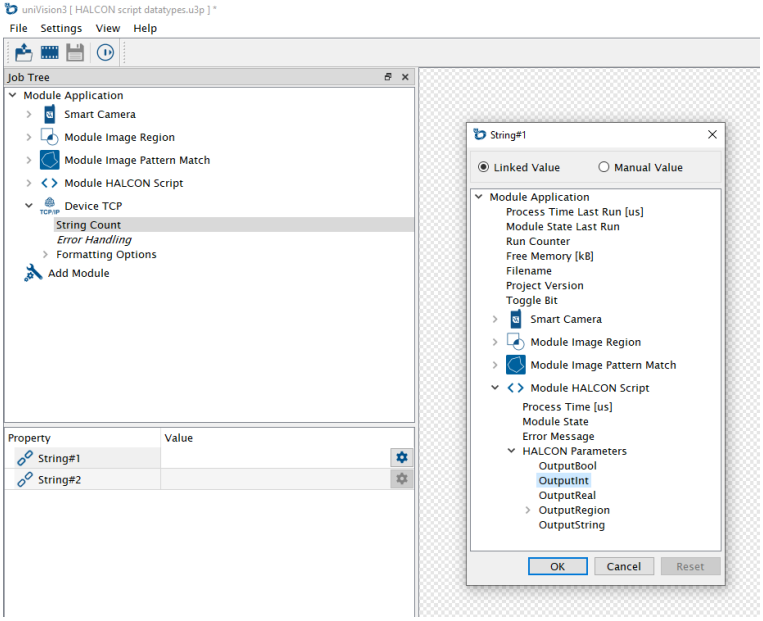
- Avoid regions with intersections as they can lead to invalid geometries (see HALCON example 02_conversion_xldpoly_to_region).
- If using color images in HDevelop, make sure that “Create RGB Image” in the input device (e.g. at Smart Camera) is activated so that the RGB color image is available for Module HALCON Script.



For all other “Control Parameters” of the procedure Run, it is possible to set fix values or to link them with dynamic results of the uniVision job tree.



Link the outputs of Module HALCON Script in other modules or devices (e.g. at Device TCP).



Output images, regions, xldpolys and xldconts are shown as sub-modules of HALCON Parameters. Regions are displayed as black and white images. Outer borders of xldpolys are shown in blue color and inner borders of xldpolys are shown in red color.

NOTE!
All inputs of Module HALCON Script must be valid (not in Error state) so that Module HALCON Script runs successfully. If one of the parameters (e.g. image or real value) is in error state, then all outputs of Module HALCON Script are in error state. If linked to any output (e.g. Device TCP), the error handling of the device will take care of it accordingly (see section “5.6.3 Status of Outputs and Error Handling”).

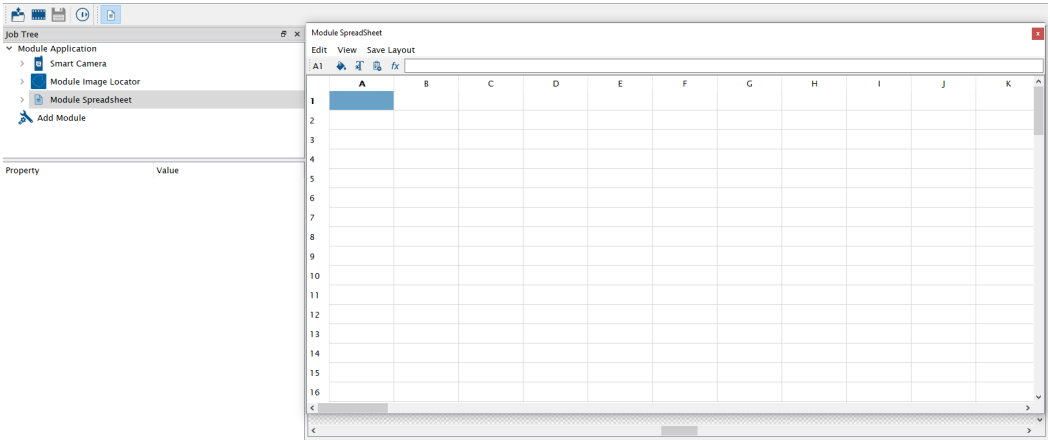
7.6 Calculation Modules

7.6.1 Module Spreadsheet

Use Module Spreadsheet to do calculations and comparisons with several results in one spreadsheet. Add job results to the spreadsheet, calculate and compare results and use results as output so that other modules can use them as input.

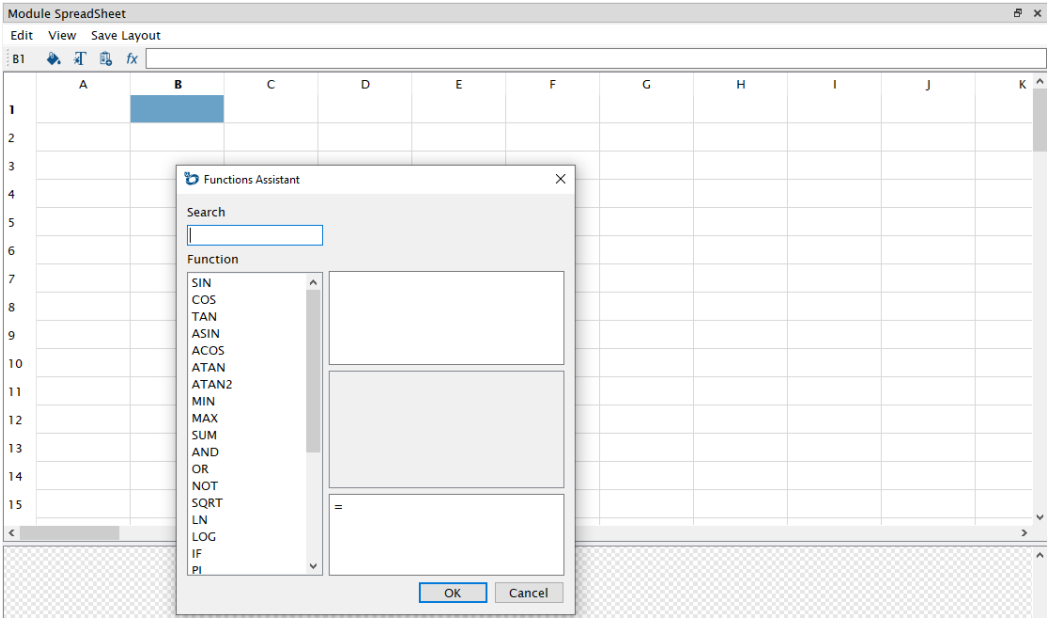
Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).

Open the spreadsheet from the Module Toolbar.



Copying, cutting or pasting cells, showing the formula and saving the layout is possible via the menu bar of the spreadsheet.

For each cell, it is possible to change background and font color, to add a job result in the cell and to open the assistant.



Via the context menu of the cell, it is possible to:

Cut	Cut cell content.
Copy	PCopy cell content.
Paste	Paste previously cut or pasted cell content. **NOTE!** Adds only the cell content (Setting for “Use as output” is lost).
Delete	Delete cell content.
Insert row	Insert a new row above the selected cell.
Insert column	Insert a new column left to the selected cell.
Remove row	Remove the selected row.
Remove column	Remove the selected column.
Add project result	Link job result to the cell of the spreadsheet. **NOTE!** Adding job results of datatype BOOL or DINT converts them automatically to values with datatype REAL. Linking job results of datatype REAL and STRING keeps the original datatype.

Use as output	Add cell to the export list (see Sub-Module Export List) so that other modules can link to it. **NOTE!** Make sure that the cell content is valid (not in ERROR state) when activating “Use as output”.

NOTE!
Do not insert or remove rows or columns if cells are used as output for the webbased visualization. Inserting or removing rows or columns changes the cell address. The webbased visualization is not notified about updated cell paths.

Basics about formulas in cells

- Formulas begin with an equal sign.
- Semicolons separate elements in formulas.
- Comparisons are possible with =; ≥; ≤, < or >.

Formula	Explanation	Example
Reference to cell	Reference to the content of another cell	=B1
Reference to job result	Add job result to the cell	=INPUT("Module Application.Run Counter")
Link numeric values	Link two or more numeric values	Examples: =A1+""+B1 =1+""+2=12
Compare numeric values	Compare numeric values	=A1>A2
Add	Add two or more values	=(2+3)
Subtract	Subtract value from another value	=(2-1)
Negation of value	Negate a value	=(-3)
Multiply	Multiply two or several values	=(2*3)
Divide	Divide value by another value	=(4/2)
Modulo	Remainder after division of one value by another.	=(8%3)
Exponent	Exponential value	=(2^3)
Min, Max	Minimum or maximum of two or more values	=MIN(2;3;1) =MAX(2;3;1)
PI	Use PI value	=PI()

Formula	Explanation	Example
Sin, cos, tan, asin, acos, atan, atan2	Various trigonometric functions. **NOTE!** Angles are in radians. Convert angles in degree to angles in radians via formula  $x = \frac{\pi}{180°} \times \alpha$ "X": Angle in radians "Alpha": Angle in degrees	=SIN(10)
Sqrt, log, ln	Root or logarithm of value	=SQRT(100)
AND, OR	Logical AND or OR operation of two or more values.	=AND(2>1;3>2)
NOT	Logically negate a value.	=NOT(1<2)
If Then Else	IF THEN query: First element contains condition. If the condition is true, the second element is the result. If the condition is false, the third element is the result.	=IF(2<3;4;5)
ISERROR	Check if the result is in ERROR state or not. Returns true if value is in ERROR state and false if value is not in ERROR state. **NOTE!** Combine ISERROR formula with IF ELSE in order to create user-defined error handling in modules.	=ISERROR(A1)
2BIN, DEC2HEX, HEX2DEC, HEX2BIN, BIN2DEC, BIN2HEX	Convert number between decimal, binary and hexadecimal. **NOTE!** Take into account maximum size of BIN, HEX and DEC. E.g. 255 is maximum decimal number that can be converted to a binary number. If the number of digits exceeds, an error is returned.	=DEC2BIN(A1)

Formula	Explanation	Example
LEFT, RIGHT	Return the first x digits of a character from left or right. For example, the first two left digits of 12345 with formula =LEFT(12345;2) are 12. **NOTE!** • Formula provides always results with datatype STRING. • If number of characters is bigger than the value, blank characters are not used as fillers, but rather the available value is output (e.g. (LEFT(ABC;5)=ABC).	=LEFT(A1;2)
TEXT	Format value by defining number of digits before and after comma. **NOTE!** • Only values of datatype DINT, REAL and BOOL can be formatted (no values of datatype STRING). • Formatting requires at least one number before and after the decimal point, e.g. “0.0” and the number before the decimal point must be high enough for all possible values. Otherwise, an error is output. • Result is always a STRING datatype.	

The following error massages may appear.

Name	Explanation	Example
ERROR_INPUT	Linked job result is not available, for example because the module has been deleted or the result is in ERROR state.	Check input data.
ERROR_PARSER	Syntax error in case of incorrect use of characters, e.g.: =(2+3)	Check the syntax of the formula.
ERROR_INF	Value is plus or minus infinity.	Check mathematical formula.
ERROR_NAN	Division by 0 or root of a negative number.	Check mathematical formula.
ERROR_VALUE	Semantic error, e.g. in the substraction of two numeric values.	Check the formula.

The Sub-Module Exports contains all cells with activated option “Use as output”. Set minimum and maximum values to convert numbers to BOOL results.

7.6.2 Module Counter

Use Module Counter to count good and bad parts.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Number of Counters	Defines the number of counters (see Sub-Module Counters).
Global Reset	Enables to reset the Counter Values for all counters to default values. Teach manually or link it with any job result.

The Sub-Module Counters contains a list of all counters. The parameter "Number of Counters" defines the number of counters. Each counter has the following settings and results:

- Counter Value: Returns the current counter value. After the maximum counter value of 2,147,483,647 is reached, an overrun occurs and the counter value starts again at 0. After booting and job loading, the counter value starts with 0.
- Counter Event: Set Counter Event manually or link it with any job result.
- Reset: Enables to reset the Counter Value to its default value. Teach manually or link it with any job result.
- Counting Method: Defines if counter value increases or decreases.
- Counting Mode: The following counting modes are available:
  - » All: Counts with every data evaluation in the Processing Instance.
  - » Within tolerance: Counts only if value of Counter Event is TRUE (e.g. linked BOOL result is true or linked number is within tolerance).
  - » Out of tolerance: Counts only if value of Counter Event is FALSE (e.g. linked BOOL result is false or linked number is out of tolerance).
  - » Error: Counts only if value of Counter Event is in error state.
- Default Counter Value: Defines the default counter value after reset.

7.6.3 Module Match Code

Use Module Match Code to compare a code or a text with a match code.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Any Match	Returns TRUE if any match code in one of the Sub-Modules is TRUE.
No Match	Returns TRUE if no match code in all Sub-Modules is TRUE.
Input String	Link to any string (e.g. reading result of Module Image Code 1D or 2D or result of Module Image OCR).
Number Elements	Defines the number of match codes (see Sub-Module Number Elements).

The Sub-Module “Number Elements” contains a list of all match codes. The parameter “Number Elements” defines the number of match codes.

- Match: Returns TRUE if the value of the current input string is equal to the Match Code.
- Mismatch: Returns TRUE if the value of the current input string is not equal to the Match Code.
- Match Code: Enter match code manually or link match code with any job result.
- Match Teach: Teaches the current input string as new match code. Teach manually or link the match teach with any job result. Match Teach updates the value of the Match Code.

NOTE!

- In case of a linked Match Code, the parameter Match Teach disappears. Then Module Match Code compares the current input string of the module with the job result linked to the Match Code.
- The following placeholders are available for entering manual match codes:
  - » *: Any string
  - » ?: Exactly one string
  - » [abc]: a, b or c may appear at this position
  - » [^A]: Any character different to “A” can appear at this position.

7.6.4 Module Teach Numeric

Use Module Teach Numeric to teach and compare numbers.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Any Ok	Returns TRUE if the Ok result of any numeric comparison in the Sub-Module is TRUE.
No Ok	Returns TRUE if no Ok result of all numeric comparisons in the Sub-Module is TRUE.
Input Numeric	Link to any number (e.g. x coordinate).
Number Elements	Defines the number of match codes (see Sub-Module Number Elements).

The Sub-Module “Number Elements” contains a list of all numbers. The parameter “Number Elements” defines the amount of numbers.

- Ok: Returns TRUE if the value of the current input numeric is within the tolerance.
- Nok: Returns TRUE if the value of the current input numeric is not within the tolerance.
- Current Input Numeric: Returns the current input numeric.
- Difference: Returns the difference between Current Input Numeric and Compare Numeric (Current Input Numeric – Compare Numeric).
- Compare Numeric: Enter compare numeric manually or link it with any job result.

- Comparison Teach: Teaches the current input numeric as new value for Compare Numeric. Teach manually or link the Comparison Teach with any job result. Comparison Teach updates the value of Compare Numeric.
- Tolerance +: Enter “Tolerance +” manually or link it with any job result.
- Tolerance -: Enter “Tolerance –“ manually or link it with any job result.

NOTE!
In case of a linked value for “Compare Numeric”, the parameter Comparison Teach disappears. Then Module Teach Numeric compares the current input numeric of the module with the job result linked to the parameter “Compare Numeric”.

7.6.5 Module Math

Use Module Math to do mathematical operations with several numbers.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Output	Returns the result of the mathematical calculation.
Math Function	Defines the mathematical operation: • +: Add value to another value. • -: Subtracts value from another value. • *: Multiply value with another value. • /: Divide value by another value. **NOTE!** If the parameter “Inputs Max Count” is bigger than 2, the mathematical operation is applied to the first two inputs. Then it is applied to the result of the calculation and the third input – until all calculations are done.
Inputs Max Count	Defines the number of inputs in the Sub-Module.

The Sub-Module Inputs allows to set manual values for the inputs or to link them with job results.

7.6.6 Module Logic

Use Module Logic to do logical operations with several values.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Output	Returns the result (TRUE or FALSE) of the logical operation.
Logic Function	Defines the logical operation: • AND: TRUE if both inputs are TRUE • OR: TRUE if at least one of the inputs is TRUE • XOR: TRUE if only one of the inputs is TRUE (exclusive OR) • NAND: TRUE if both inputs are not TRUE • NOR: TRUE if both inputs are FALSE **NOTE!** If the parameter "Inputs Max Count" is bigger than 2, the logical operation is applied to the first two inputs. Then it is applied to the result of the operation and the third input – until all operations are done.
Inputs Max Count	Defines the number of inputs in the Sub-Module.

The Sub-Module Inputs allows to set manual values for the inputs or to link them with job results.

7.6.7 Module Numeric Comparison

Use Module Numeric Comparison to compare two numeric values.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Output	Returns the result (TRUE or FALSE) of the numeric comparison.
Compare Function	Defines the operation for the numeric comparison: • A > B: TRUE if A is bigger than B • A < B: TRUE if A is smaller than B • A >= B: TRUE if A is bigger or equal to B • A <= B: TRUE if A is smaller or equal to B • A == B: TRUE if A is equal to B • A != B: TRUE if A is not equal to B
Input A	Set input A manually or link it with job result.
Input B	Set input B manually or link it with job result.

7.6.8 Module Statistic

Use Module Statistic to analyse statistic results for one or several values.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Channel Count	Defines the number of channels (see Sub-Module Channel Count).

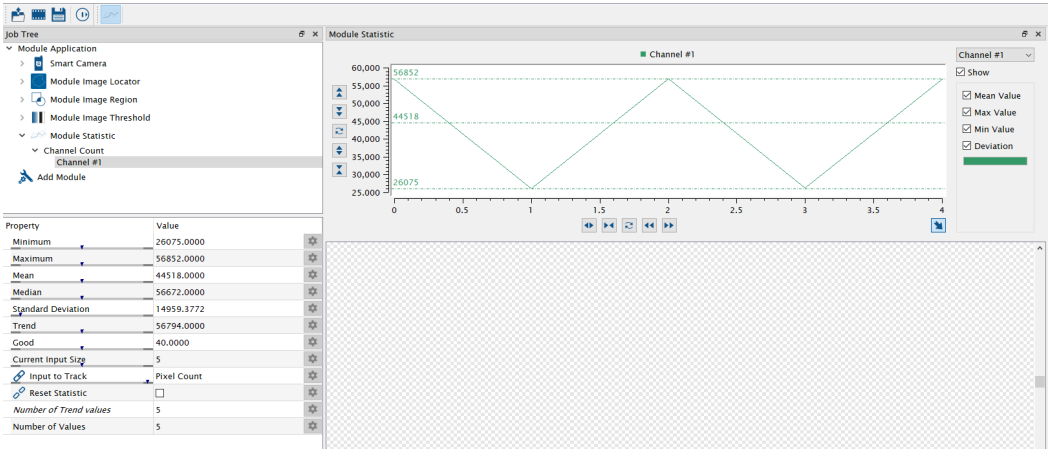
The Sub-Module Channel Count contains the number of channels set via the parameter Channel Count and contains the following parameters and results:

- Minimum: Returns the minimum of all historic results.
- Maximum: Returns the maximum of all historic results.
- Mean: Returns the mean of all historic results (sum of all historic results divided by the number of historic results).
- Median: Returns the median of all historic results (center value when sorting all historic results by magnitude).
- Standard Deviation: Returns the standard deviation (square root of the sum of the squared differences from the mean value divided by the number of values).

$$s = \sqrt{\frac{1}{n} * \sum_{i=1}^{n} (x_i - \bar{x})^2}$$

- Trend: Returns the next expected result. The algorithm calculates the trend via a linear regression of the last historic results defined by the parameter “Number of Trend Values”.
- Good: Returns the percentage value of historic TRUE results compared to all historic results.
- Current Input Size: Returns the current number of results used for the statistic calculation.
- Input to Track: Link to any job result.
- Reset Statistic: Reset statistic manually or link it with job result.
- Number of Trend Values: Defines the number of the last historic results used for calculating the trend (see result Trend).
- Number of Values: Defines the number of the last historic results used for calculating the statistic (Minimum, Maximum, Mean, Median, Standard Deviation, Good, Current Input Size). Changing the parameter “Number of Values”, resets the statistic results.

The Module Toolbar contains a window to visualize historic results.



7.7 Standard Interfaces

7.7.1 Device TCP

Use Device TCP to send process data via TCP/IP to a communication partner. For each data evaluation in the Processing Instance (e.g. image evaluation), one result is sent to the communication partner.
The Machine Vision Device is a TCP/IP server. Use a TCP/IP client at the communication partner to establish a connection (e.g. the software Free IP Tools). Use the IP address of the LAN network at the Machine Vision Device and the port that is shown in Device TCP (for details, see section “5.7.2 Network Interfaces for Processing Instances”).

NOTE!

- Make sure that the Machine Vision Device and the communication partner are in the same network and that the network load is ok in order to receive all results.
- Only supported at real Machine Vision Devices (not in offline jobs via uniVision Simulator).

After booting and after loading another job, it is necessary to re-connect to the TCP/IP server.

NOTE!
If working with PLCs, close the connection from the PLC at job changes and re-establish the connection after the job is loaded so that the closed connection is detected by the PLC.

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Output	Returns a preview of the output sent to the communication partner.
Connections	Shows the maximum number of five simultaneous connections supported by the TCP/IP server.
TCP Port	Shows the TCP port the TCP/IP client needs to use to connect.
Preamble	Defines the characters sent in the beginning of each output. **NOTE!** Click on the settings icon to add further ASCII symbols.
Postamble	Defines the characters sent at the end of each output. **NOTE!** Click on the settings icon to add further ASCII symbols.
Delimiter	Defines the characters that separate the strings. **NOTE!** Click on the settings icon to add further ASCII symbols.
String Count	Defines the number of strings (see Sub-Module String Count). The maximum value of String Count is 100.
Output Mode	Defines if the strings are sent formatted or unformatted (see Sub-Module Formatting Options).
Error Handling	If a job result linked to one of the strings is in error state, the behavior is to substitute the value with the STRING substitution value defined at the Sub-Module Error Handling.

The Sub-Module String Count defines the strings. Set the strings to fix values or link them to job results. The parameter “String Count” defines the number of strings.

The Sub-Module Error Handling defines the substitution value for strings that is applied if linked job results are in error state. By default, the string substitution value is Error###.

The Sub-Module Formatting Options defines for each datatype the formatting (only visible if Output Mode is set to Formatted):

- Integer: Defines the number of digits and if + is printed or not.
- Floating Point: Defines the number of digits before and after the comma and if + is printed or not.
- Boolean: Defines the mode and the number of digits.
- String: Defines the number of digits.

NOTE!

- Make sure that the number of digits at the Formatting Options is big enough for all results (including the substitution value at the error handling). Otherwise, the output result contains more characters than defined.
- If the result contains less digits, strings are filled with empty space characters in the beginning - integers, floating points and Boolean results are filled with 0.

7.7.2 Device UDP

Use Device UDP to broadcast process data via UDP (without connection). For each data evaluation in the Processing Instance (e.g. image evaluation), one result is sent.

Use a software at the communication partner to listen to the UDP data (e.g. the software Free IP Tools). Use the port that is shown in Device UDP (for details, see section “5.7.2 Network Interfaces for Processing Instances”).

NOTE!

- Make sure that the Machine Vision Device and the communication partner are in the same network and that the network load is ok in order to receive all results.
- Only supported at real Machine Vision Devices (not in offline jobs via uniVision Simulator).

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Output	Returns a preview of the output sent to the communication partner.
UDP Port	Shows the UDP port the software needs to listen to in order to receive the process data.
Preamble	Defines the characters sent in the beginning of each output. **NOTE!** Click on the settings icon to add further ASCII symbols.
Postamble	Defines the characters sent at the end of each output. **NOTE!** Click on the settings icon to add further ASCII symbols.
Delimiter	Defines the characters that separate the strings. **NOTE!** Click on the settings icon to add further ASCII symbols.
String Count	Defines the number of strings (see Sub-Module String Count). The maximum value of String Count is 100.
Output Mode	Defines if the strings are sent formatted or unformatted (see Sub-Module Formatting Options).

Error Handling	If a job result linked to one of the strings is in error state, the behavior is to substitute the value with the STRING substitution value defined at the Sub-Module Error Handling.

The Sub-Module String Count defines the strings. Set the strings to fix values or link them to job results. The parameter “String Count” defines the number of strings.

The Sub-Module Error Handling defines the substitution value for strings that is applied if linked job results are in error state. By default, the string substitution value is Error###.

The Sub-Module Formatting Options defines for each datatype the formatting (only visible if Output Mode is set to Formatted):

- Integer: Defines the number of digits and if + is printed or not.
- Floating Point: Defines the number of digits before and after the comma and if + is printed or not.
- Boolean: Defines the mode and the number of digits.
- String: Defines the number of digits.

NOTE!

- Make sure that the number of digits at the Formatting Options is big enough for all results (including the substitution value at the error handling). Otherwise, the output result contains more characters than defined.
- If the result contains less digits, strings are filled with empty space characters in the beginning - integers, floating points and Boolean results are filled with 0.

7.7.3 Device FTP

Use Device FTP to send process data to a FTP or SFTP server running on a communication partner (e.g. PC). The FTP or SFTP protocol supports to send files, for example image or text files. For each data evaluation in the Processing Instance (e.g. image evaluation), one result consisting of one or several files is sent.

NOTE!

- Make sure that the Machine Vision Device and the communication partner with the FTP/SFTP server are in the same network.
- The FTP protocol uses the ports 20 and 21 - SFTP uses the port 22.

Configure the FTP/SFTP Client Settings on the Machine Vision Device in order to connect to the FTP/SFTP Server in the network (see section “5.9 FTP/SFTP Client on Machine Vision Devices”).

NOTE!

- File transfer via FTP or SFTP takes time. Make sure that the acquisition frequency and the network load are low enough to transfer all results.
- If data saving fails (e.g. because of too high acquisition frequency), the module state signals the data loss in one of the following evaluations (see “5.6.4 Module States”). The status of the Processing Instance also signals the FTP error (see “5.6.5 Status of Processing Instance”).
- Make sure to provide enough storage space at the FTP/SFTP server and empty the storage regularly. Also, avoid a big number of files inside of a single folder.

It is also possible to store the files in the local output folder of the Machine Vision Device. Make sure to empty the storage regularly. Access the files on the Machine Vision Device via the SFTP server on the Machine Vision Device (see section “5.8 SFTP Server on Machine Vision Devices”).

NOTE!
In case of less than 200 MB free memory on the Machine Vision Device, no more files are stored and module status and status of Processing Instance signal errors.

Saving data in local output folder and on FTP or SFTP server is only supported at real Machine Vision Devices (not in offline jobs via uniVision Simulator).

Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section “5.6.4 Module States”).
Output	Returns a preview for the content of the text file.
Preamble	Defines the characters sent in the beginning of each text file. **NOTE!** Click on the settings icon to add further ASCII symbols.
Postamble	Defines the characters sent at the end of each text file. **NOTE!** Click on the settings icon to add further ASCII symbols.
Delimiter	Defines the characters that separate the strings within the text file. **NOTE!** Click on the settings icon to add further ASCII symbols.
String Count	Defines the number of strings for the text file (see Sub-Module String Count). The maximum value of String Count is 100.
Output Mode	Defines if the strings are formatted or unformatted (see Sub-Module Formatting Options).
Error Handling	If a job result linked to one of the strings is in error state, the behavior is to substitute the value with the STRING substitution value defined at the Sub-Module Error Handling.
Filename	The filename consists of: • Fix part with date and time stamp (to guarantee unique file names). • Flexible part of the file name: Fix value or linked to job result. **NOTE!** Fix and flexible components of the filename are separates by underscore.
Input Pointcloud	Optionally, link an input pointcloud to send or store the pointcloud file.

Input Image	Optionally, link an input image to send or store the image file. Depending on the parameter “Save Image Type”, link a monochrome or a color image. Make sure that the BGRA image channel is active at the input device in order to see an available color input image.
Data Sink	Defines where the data is stored: • FTP: Store files on FTP or SFTP server in the network. • Local Folder “output”: Store files in local folder “output” on the Machine Vision Device. Access the files on the Machine Vision Device via the SFTP server on the Machine Vision Device (see section “5.8 SFTP Server on Machine Vision Devices”).
Save Image Type	Defines if monochrome (8 bit single channel) or color (BGRA) images can be linked at the parameter Input Image.
Save Image Compression	Defines if images are stored in BMP or JPG format.
Observer	Defines if the files at the current data evaluation are stored or not: • True: Files are not stored. • False: Files are stored. Set value fix or link it to job result.

The Sub-Module String Count defines the strings for the text file. Set the strings to fix values or link them to job results. The parameter “String Count” defines the number of strings.

The Sub-Module Error Handling defines the substitution value for strings that is applied if linked job results are in error state. By default, the string substitution value is Error###.

The Sub-Module Formatting Options defines for each datatype the formatting (only visible if Output Mode is set to Formatted):

- Integer: Defines the number of digits and if + is printed or not.
- Floating Point: Defines the number of digits before and after the comma and if + is printed or not.
- Boolean: Defines the mode and the number of digits.
- String: Defines the number of digits.

NOTE!

- Make sure that the number of digits at the Formatting Options is big enough for all results (including the substitution value at the error handling). Otherwise, the output result contains more characters than defined.
- If the result contains less digits, strings are filled with empty space characters in the beginning - integers, floating points and Boolean results are filled with 0.

7.7.4 Device Industrial Ethernet

For details about Device Industrial Ethernet, check the separate interface protocol for Industrial Ethernet (see https://www.wenglor.com/product/DNNF023)

7.8 Robot Vision

Device Robot Vision enables direct communication between Machine Vision Device and robot.

Supported robots:

- E series of UR (CB series of UR is not supported)

Mounting options:

- Camera not at robot (statically):
  - » Mount the Machine Vision Device in a static way e.g. above the object plane. Slight tilting is supported to avoid reflections. For details about mounting solutions, check the operating instruction of the Machine Vision Device.
  - » During calibration (installation) procedure, the calibration plate is mounted on the robot (for details about mounting solutions check the product detail page of the calibration plates ZVZJ on the wenglor website). In a second calibration step, the calibration plate is fix on the object plane.
  - » During program procedure, the Machine Vision Device can detect the objects while the robot arm is not in the field of view of the camera.
- Camera at robot:
  - » Mount the Machine Vision Device on the robot arm. For details about mounting solutions, check the operating instruction of the Machine Vision Device.
  - » During calibration (installation) procedure, the calibration plate is fix on the object plane and the robot with the camera moves to different positions.
  - » During program procedure, the robot needs to move to the detection pose to trigger the next detection.

NOTE!
For safe cable management, use the joint limits in the safety settings of the robot and the angle start and end limits within the localization module.

7.8.1 Calibration Job in uniVision

Connect the software wenglor uniVision 3 to the Machine Vision Device (see section “6.1 Connecting to Machine Vision Device”). Open the template “Calibrate camera with robot” on the Machine Vision Device and adjust camera parameters and lens to get a sharp and well illuminated image.

NOTE!

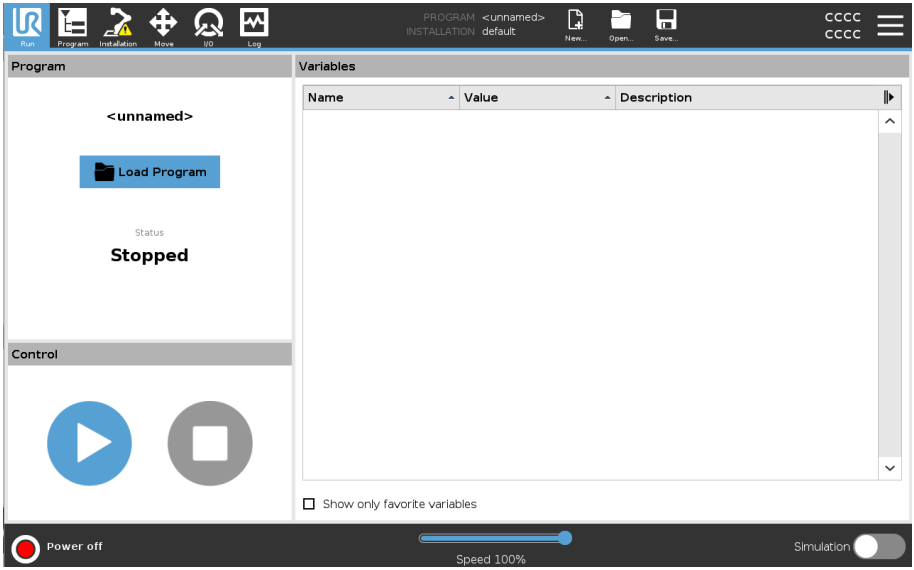
- In the template “Calibrate camera with robot” the Trigger Source is set to Software (mandatory for the calibration procedure). If needed, set Trigger Mode to Off and switch to Run Mode in order to adjust focus and brightness (especially if working with C-Mount cameras). Do not forget to set Trigger Mode back to On and Trigger Source back to Software.
- If working with color cameras, make sure that “Create BGRA Image” is active at the input device (e.g. at Smart Camera).

Save the calibration job in the device projects folder on the Machine Vision Device.

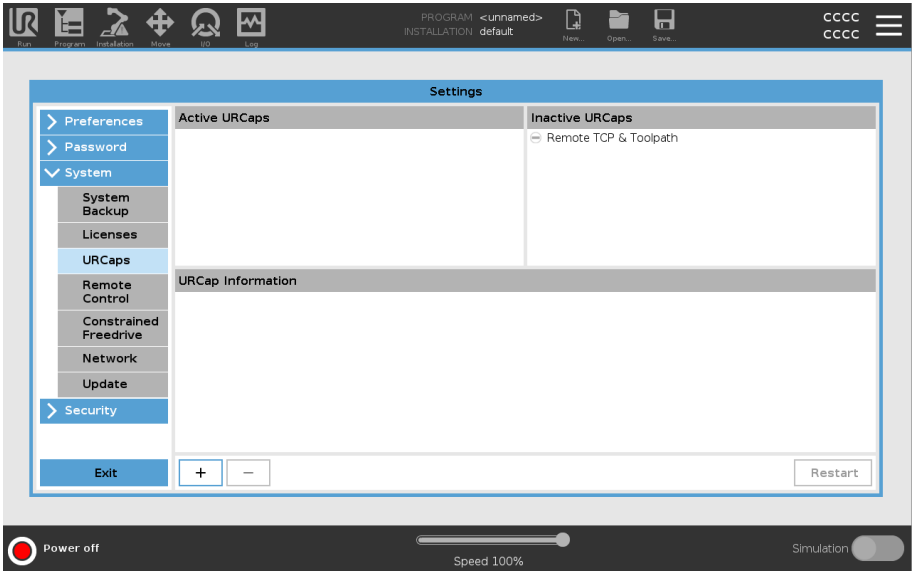
7.8.2 Installation of URCap

Download the latest URCap version from https://www.wenglor.com/product/DNNF023 → Downloads → Soft- and Firmware and put the file on a freshly formatted USB stick.

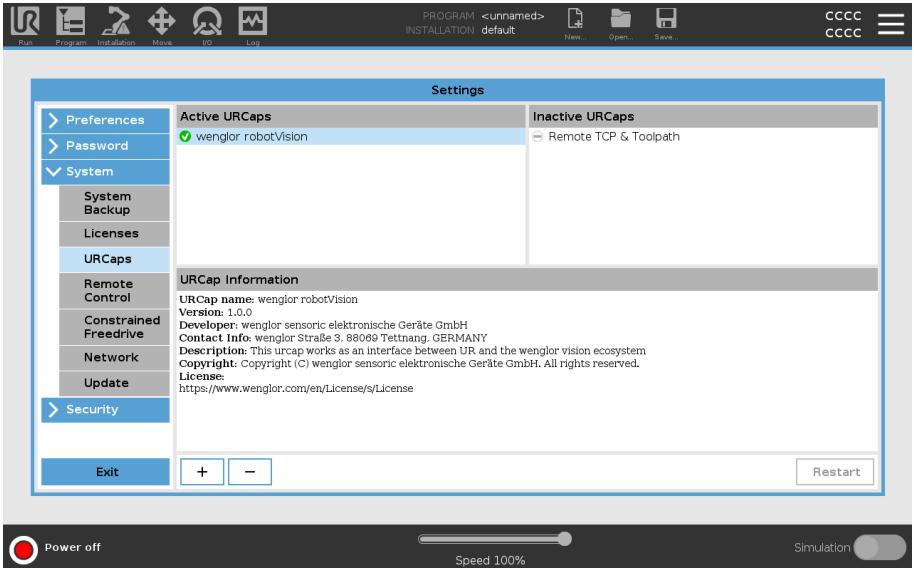
Plug in the USB stick into the UR Teach Panel in order to install the URCap. Start the robot and go to the menu in the top right corner of the UR Teach Panel.



Navigate then to the System menu and select URCaps. Click on the + symbol.



Navigate through the system until you find the URCap file “wenglor robot vision”. Select it and proceed by re-starting the robot. After the reboot, the URCap is displayed with a green hook after the successful installation.

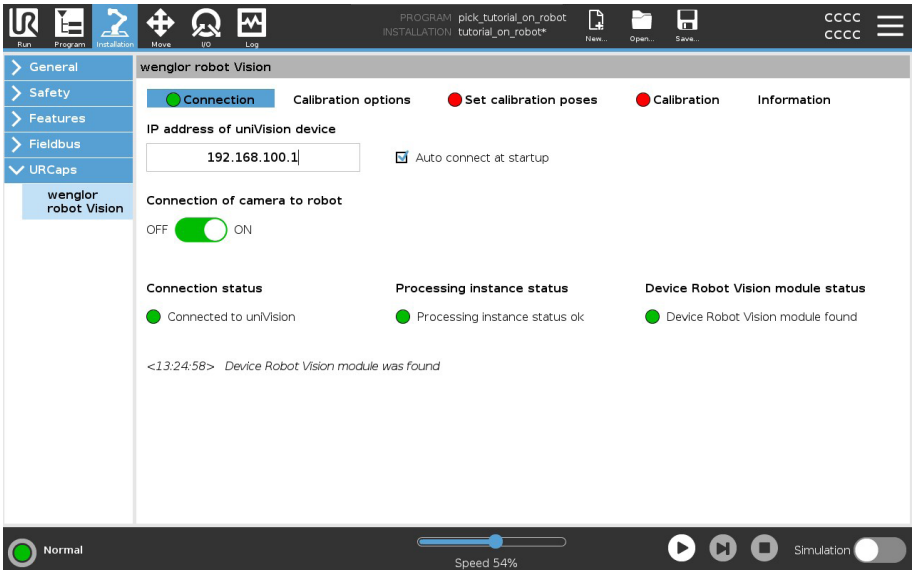


7.8.3 UR Installation

The installation of the URCap “wenglor robot vision” enables to connect to the Machine Vision Device and to calibrate camera and robot. Select the “Installation” on the UR Teach Panel and click on “URCaps” -> “wenglor robot vision”.

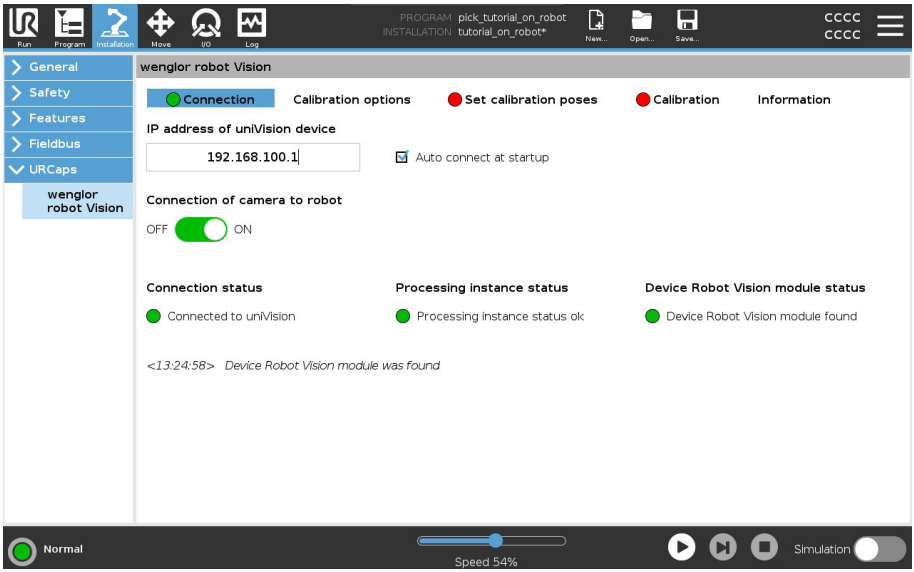
LED status in tabs:

- Connection tab
  - » Red: No connection to Machine Vision Device possible. Check the network setup.
  - » Green: Connection to Machine Vision Device established and valid.
- Set calibration poses tab
  - » Red: Not all calibration poses were set. Check the calibration poses and set them accordingly.
  - » Green: All calibration poses were set.
- Calibration tab
  - » Grey: Could not check for previous calibration data as no connection to the Machine Vision Device is available. Establish the connection to the Machine Vision Device first.
  - » Red: No valid calibration data available. Run the calibration procedure again.
  - » Yellow: Calibration is incomplete (if second calibration step is missing when camera is static and not at robot).
  - » Green: Calibration data was set during the current session or was loaded from a previous calibration.



Connection tab
Connect to the Machine Vision Device via entering the IP Address (default: 192.168.100.1) and setting the connection slider to on. By default, auto connect at startup is active.

NOTE!
Make sure that robot and Machine Vision Device are in the same network.



Status:

- Connection status
  - » Red: No connection to Machine Vision Device possible. Check the cables and the network setup.
  - » Green: Connection to Machine Vision Device is available.
- Processing Instance status
  - » Red: Processing Instance status displays an error. Check the status of the Processing Instance on the tab Information or on the device website of the Machine Vision Device.
  - » Green: No error at status of Processing Instance.
- Device Robot Vision module status
  - » Red: No Device Robot Vision within current uniVision job. Add Device Robot Vision to current uniVision job.
  - » Green: Device Robot Vision is available in current uniVision job.

Calibration options tab

Select if the camera is mounted at the robot or no at the robot (static).

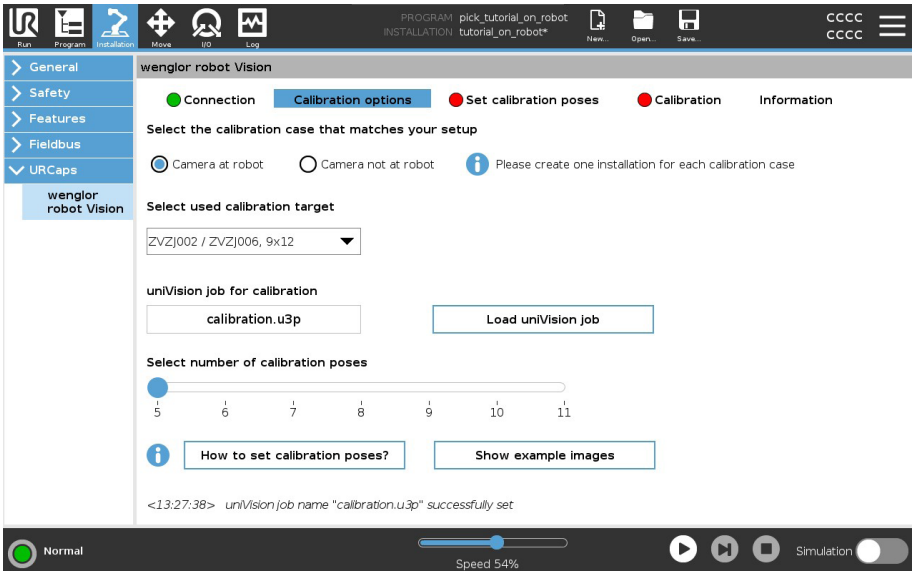
Select the calibration target. Use one of the calibration targets ZVZJ on the wenglor website or print the pdf on a stiff and flat material (see www.wenglor.com -> ZVZJ). For best accuracy, use the wenglor calibration plates ZVZJ.

NOTE!

- In case of printing, make sure to print the pdfs at actual size.
- Typically, the reprojection error for the ZVZJ calibration plate is five times better compared to the printed version.
- For direct light applications, non-transparent calibration targets with the material carbon fiber are available. For backlight applications, transparent calibration targets with the material glass are available.
- The calibration target should cover at least half of the image and should be visible completely by the camera if possible for best accuracy results.

Enter the name of the uniVision job for calibration and load it. The job is loaded every time the calibration procedure is started. Set the number of calibration poses between 5 and 11 (default: 5 poses). The calibration results get better with more calibration poses.

Further info is provided about how to set the calibration poses and some example images from camera perspective for the tab “Set calibration poses”.

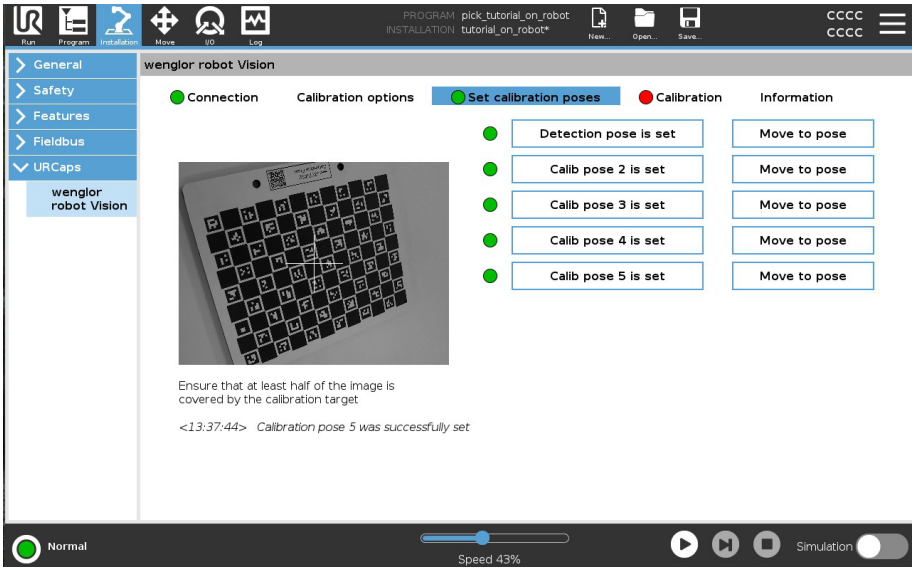


Set calibration poses tab

The calibration poses are different for the camera at robot and for the camera not at robot (static). Therefore, it is recommended to create separate installations in case of using both mounting solutions. The number of buttons to set the calibration poses depends on the settings made in the calibration options tab.

- If the Machine Vision Device is mounted at the robot, it is necessary to put the calibration plate on the object plane. Different poses of the robot must be set so that the wenglor robot vision URCap can run the calibration. The first pose is the detection pose.
- If the Machine Vision Device is static (e.g. above the object plane), it is necessary to mount the calibration plate on the robot. Different poses of the robot must be set so that the URCap “wenglor robot vision” can run the calibration.

NOTE!
Make sure when setting the poses with the calibration plate mounted at the robot that the height difference between object plane and calibration target is as small as possible to achieve best accurate results.



NOTE!

- Check the camera live image before setting the poses.
- Make sure that the difference between one pose and the next one is as big as possible for best accurate results. The same pose could be used several times if not consecutive (e.g. pose one and three can be similar). If space is limited in the application, small differences between one pose and the next one are also possible, but result in less accurate results.
- The distance from the camera to the calibration target shall remain as constant as possible.
- The calibration target should cover as much as possible of the camera image and should be visible completely if possible for best accuracy results.
- The poses can be considered on a half sphere surface where the blue arrows represent the image axis of the camera for the use case “Camera at robot”.

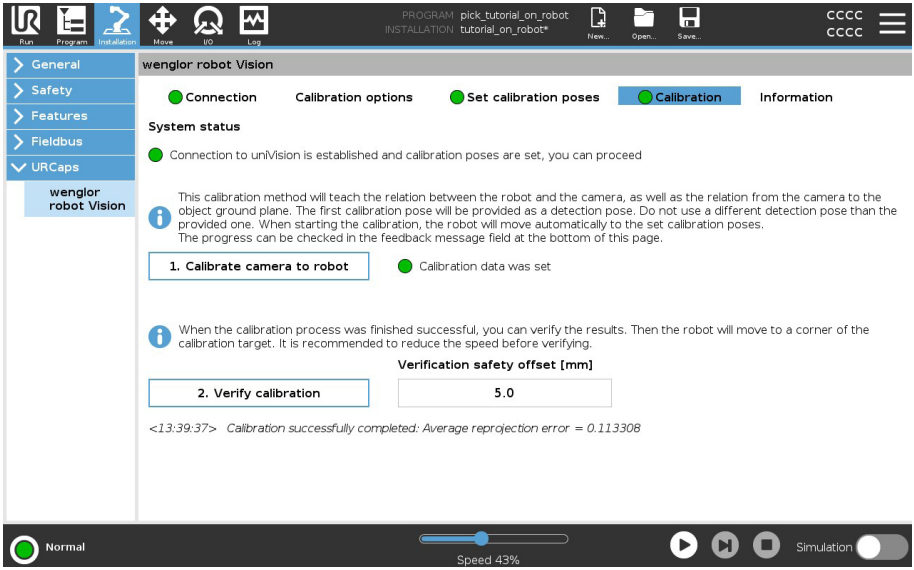
Calibration tab
System status LED for calibration:

- Red: No active connection to Machine Vision Device and not all calibration poses set. Connect first to the Machine Vision Device and set the calibration poses.
- Yellow: Only one condition is true
  - » Active connection to Machine Vision Device, but not all calibration poses are set. Set the calibration poses.
  - » No active connection to Machine Vision Device, but the calibration poses are set. Connect first to the Machine Vision Device.
- Green: Active connection to Machine Vision Device and all calibration poses are set.

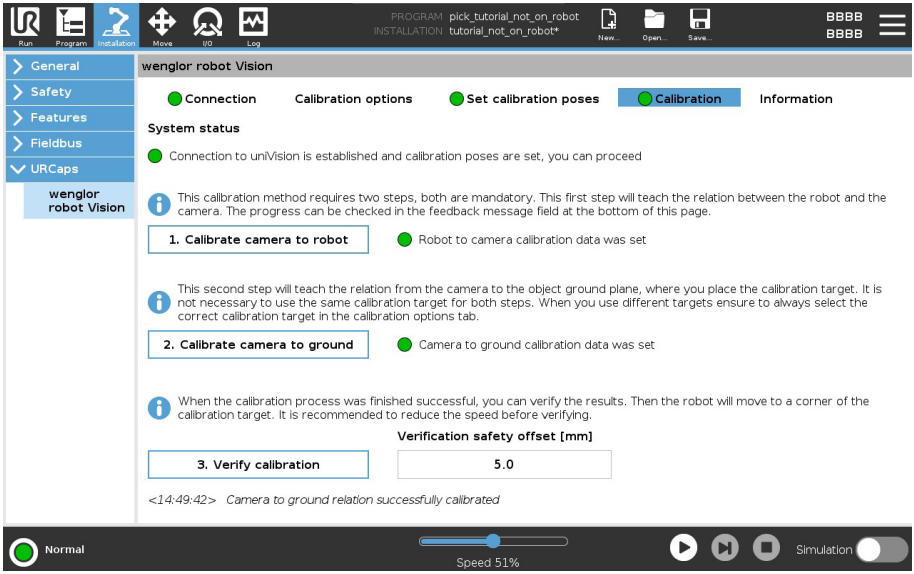
It is only possible to start the calibration if the system status LED is green. At the start of the calibration, the “uniVision job for calibration” is loaded.

The calibration is different for the camera at robot and for the camera not at robot (static).

- If the camera is mounted at the robot, only one calibration step is needed. When starting the calibration, the robot will load the calibration job and move automatically to the calibration poses set in the tab “Set calibration poses”.



- If the camera is mounted static above the object plane, the calibration process requires two steps.
  - » When starting the first calibration step, the calibration target must be mounted at the robot. The robot will load the calibration job and move automatically to the calibration poses set in the tab “Set calibration poses”.
  - » For the second step, unmount the calibration target from its holder on the robot and place it on the object plane. Make sure that the calibration target is visible for the camera (robot arm can be moved away in order to not interfere). Check the camera image in the tab “Set calibration poses”. When the calibration target is placed, you can trigger the second calibration step. Only one image is captured for the second calibration step.



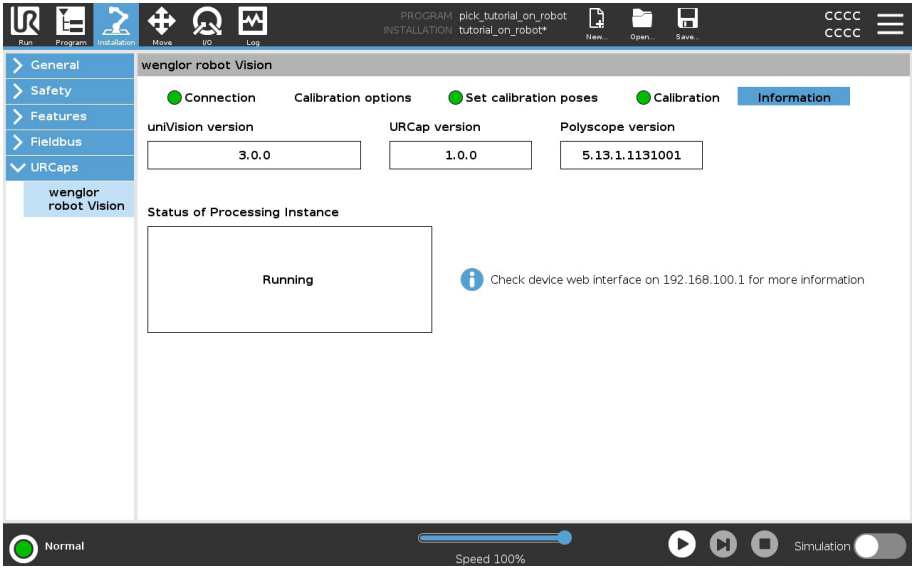
NOTE!
Calculating the relations takes some time – wait until the calibration is completed.

After the calibration is done successfully, you can perform the optional verification where you can guide the robot to drive to the bottom left corner of the calibration target. Therefore, it is necessary that the calibration target was not moved between the calibration and the verification step. The verification offset will add a height offset, so that collisions with the calibration target are avoided.

NOTE!

- For the verification step, the z axis must be oriented downwards (towards the object).
- The reprojection error shows how good the calibration was. Values of 0.1 for ZVZJ calibration plates are typical. Values of 0.5 are typical for printed calibration plates.

Information tab
The information tab contains info about different versions as well as the status of the Processing Instance.



7.8.4 Detect Objects Job in uniVision

Connect the software wenglor uniVision 3 to the Machine Vision Device (see section “6.1 Connecting to Machine Vision Device”). Open the template “Pick objects with robot” on the Machine Vision Device and adjust camera parameters and lens to get a sharp and well illuminated image.

NOTE!

- In the template “Pick objects with robot” the Trigger Source is set to. If needed, set Trigger Mode to Off and switch to Run Mode in order to adjust focus and brightness (especially if working with C-Mount cameras). Do not forget to set Trigger Mode back to On and Trigger Source back to Software.
- If working with color cameras, make sure that “Create BGRA Image” is active at the input device (e.g. at Smart Camera).

Create uniVision job for detecting objects e.g. via Module Image Locator or via Module Image Pattern Match. In the template, Module Image Locator is used to detect objects.

NOTE!

- For details about Module Image Locator see section “7.3.4 Module Image Locator” and about Module Image Pattern Match see section “7.3.5 Module Image Pattern Match”.
- Do not combine Module Image Calibration with the calibration on robot side.

In the template, Device Robot Vision is already pre-configured to send the results to the robot interface.

Process Time [µs]	Process time to run the module in µs.
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see section "5.6.4 Module States").
Output	Shows preview of the output sent to robot interface.
Shape Model Count	Defines the number of shape models (object types) used in the submodule "Shape Model Height Difference".
Result Max Count	Defines the number of results used in the sub-module "Result List".
Result True Count	Link the value with Result True Count of Module Image Locator or Module Image Pattern Match so that the robot knows how many objects are actually in the current image.

Sub-Module "Shape Model Height Difference" (number of entries depends on parameter Shape Model Count)

Height Difference to Calibration [mm] #0...x	Defines the heigth difference of each shape model to the calibration plate.

NOTE!

- It is possible to pick different object types on different heights. Make sure to keep the height differences (especially the height difference to the calibration target) as small as possible for best accurate results.
- Enter the object height relative to the surface of the calibration target in mm. If the object is smaller than the calibration plate plane, enter a negative number. If the object is higher, enter a positive number. Use the height difference also in case of height variations after the calibration process.

Sub-Module "Result List" (number of entries depends on parameter Result Max Count)

Link the results list of Module Image Locator or Module Image Pattern Match to the Result List of Device Robot Vision.

Shape Model	Link the shape model (object type) for each result.
X [px]	Link the x position for each result.
Y [px]	Link the y position for each result.
Phi (Z-Rotation) [deg]	Link the Phi value (z rotation) for each result.
Additional Value	Optionally link an additional string result (e.g. score value) for each result.

NOTE!

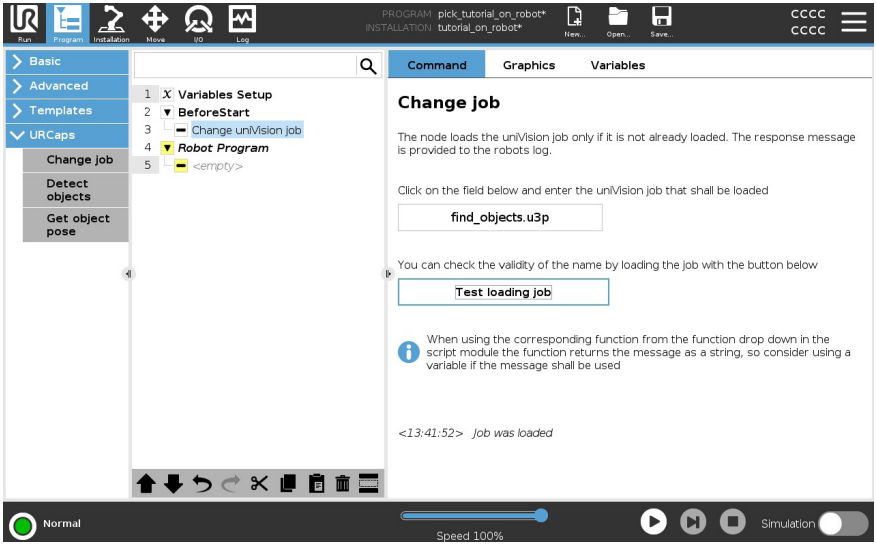
- It is possible to detect several objects with one image capturing.
- Offsets in x and y can be set directly in Module Image Locator or Module Image Pattern Match.
- Pixel results for x and y can only be used by the robot if created by modules that use the original camera image or any image based on it. No support of x and y coordinates created by modules that use the undistorted image of Module Image Calibration or any image based on it.

Save the detect objects job in the device projects folder on the Machine Vision Device.

7.8.5 UR Program

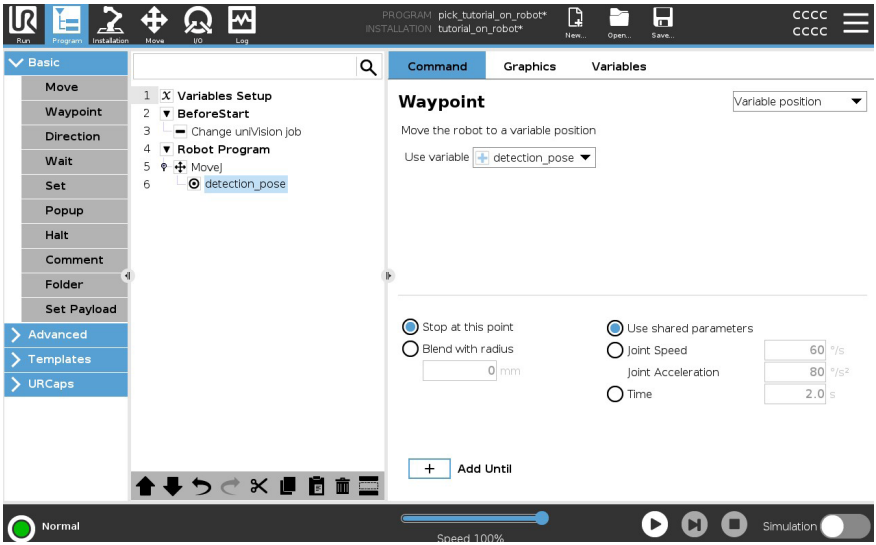
The URCap “wenglor robot vision” adds modules to change the uniVision job, to detect objects or to get the latest object pose. Switch to the program tab on UR side in order to create the robot program. First step is to load the uniVision job for detecting objects. Use the “Change job” module from the URCaps section on the left drop-down menu.

Enter the name of the uniVision job file. To test the validity of the job name, load it by clicking the button “Test loading job”.

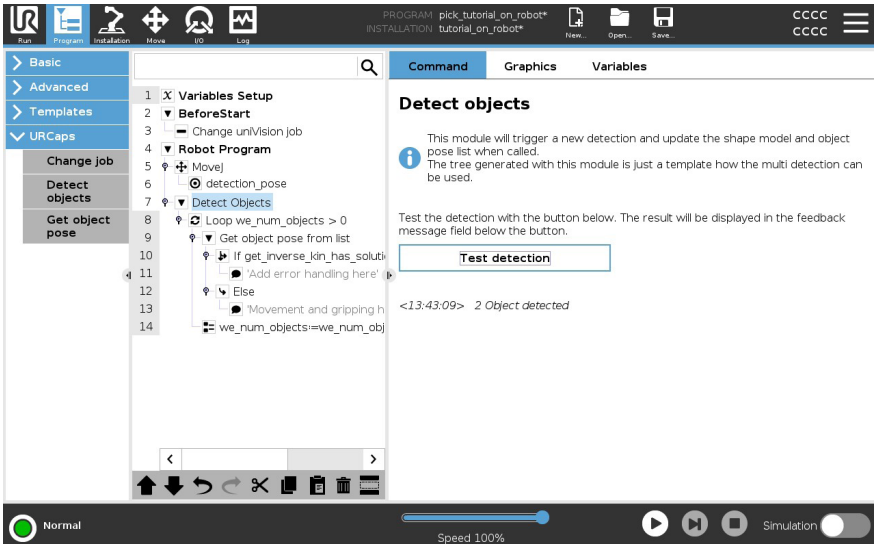


In case of static camera above the object plane, you can teach a detection pose to prevent the robot from interfering the field of view of the camera while capturing images.

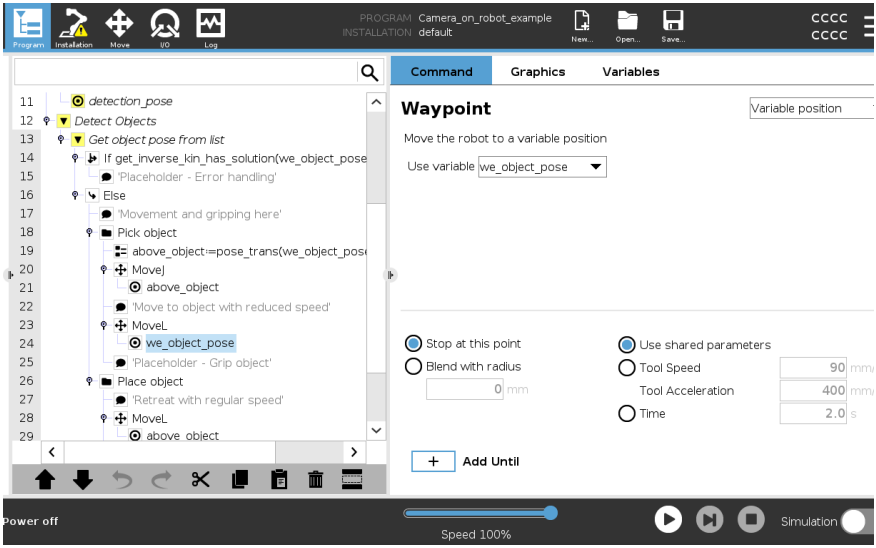
If the camera is mounted at the robot, it is necessary to move the robot to the detection pose so that the camera can check for objects. The detection pose from the URCap must be used. To update the detection pose, it is necessary to set the detection pose in the URCap installation page and to re-run the calibration process.



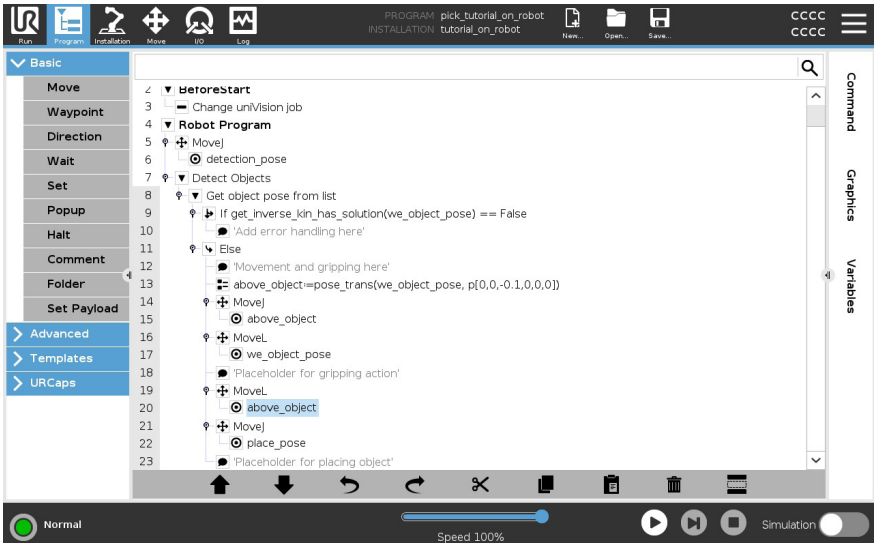
Then add the module "Detect objects". It adds a subtree, which iterates through all found objects.



You can access the object pose as a variable in order to move the robot to any position.



In case of camera on robot, the full program for pick and place can look like this:



NOTE!
The UR example robot programs for camera on and not on robot are available on https://www.wenglor.com/product/DNNF023 → Downloads → Soft- and Firmware.

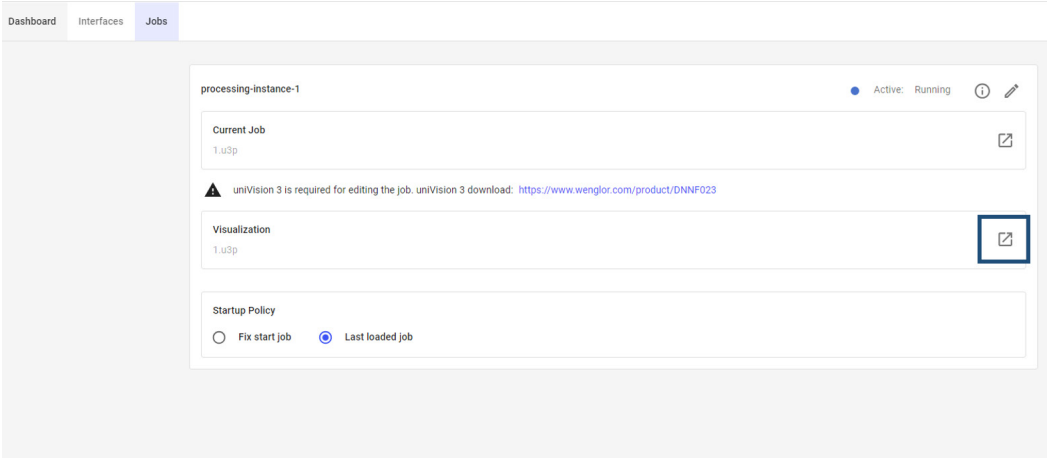
8. Webbased Visualization

Every Processing Instance contains a webbased visualization to show all relevant results for a job at one glance. Each uniVision job can have its own individual visualization. The browser updates the content of the visualization at regular intervals.

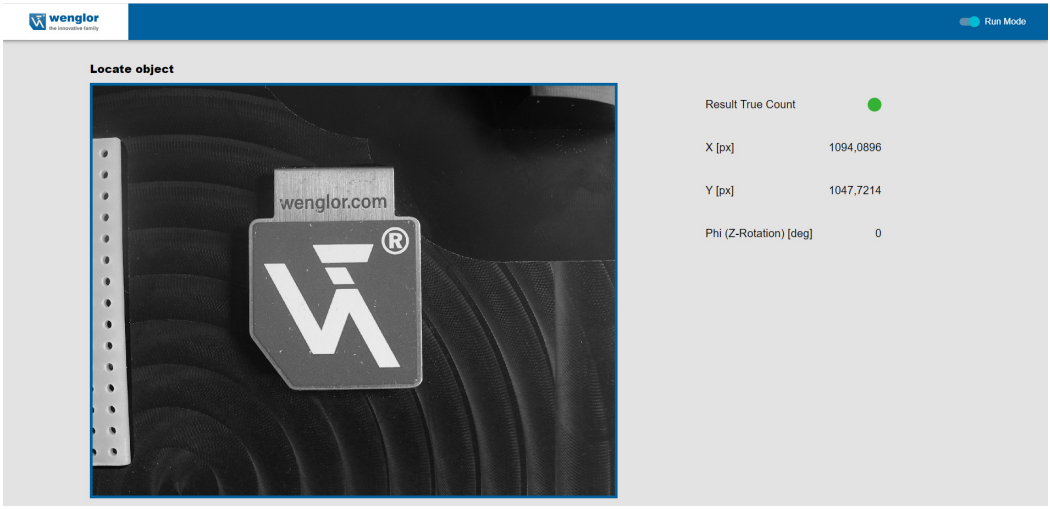
Supported minimum browser versions (tested on Windows 10 22H2 and Windows 11 22H2 PCs):

- Chrome 108
- Firefox 108
- Microsoft Edge 108
- Chromium 111

Opening the webbased visualization on a real Machine Vision Device is possible via the tab „Jobs“ on the device website by clicking on the icon at Visualization .



A new tab with the webbased visualization opens in Run Mode.

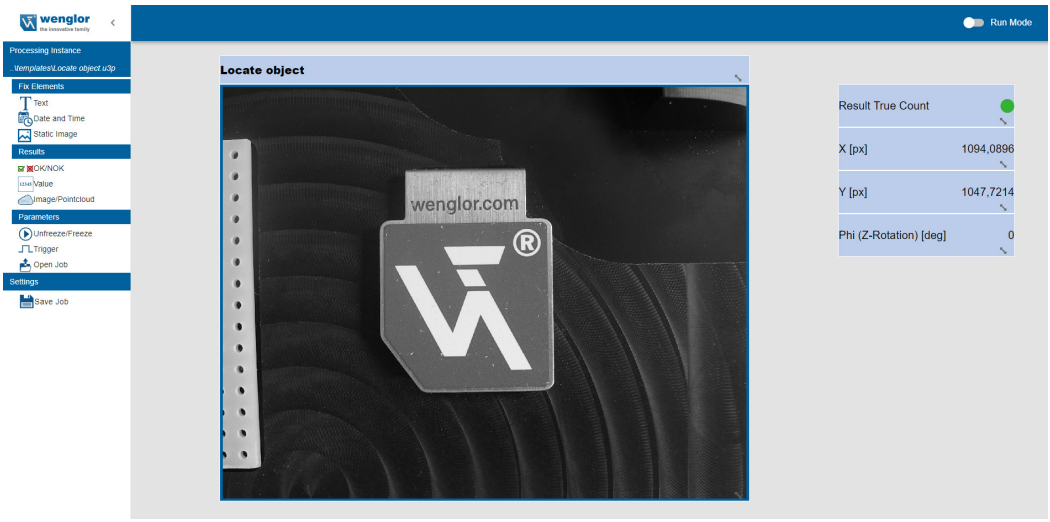


Accessing the webbased visualization is also possible for offline jobs in the uniVision Simulator via the menu bar “View” → “Webbased Visualization” in the Software wenglor uniVision 3 (see section “6.3.1 Menu Bar”).

If authentification is activated on the Machine Vision Device, login is necessary to interact within the visualization or to edit the visualization.

NOTE!
For details about different user rights, check the operating instruction of the Machine Vision Device.

Deactivate Run Mode to create or adjust the visualization. The toolbox appears on the left side. Drag and drop elements from the toolbox to the visualization scene. Save the uniVision job together with the visualization.



Toolbox elements

Fix elements:

Text	Add a text field to the visualization and enter a fix text (e.g. job name).
Date and Time	Add date and time (of the PC) in the visualization scene.
Static Image	Add a static image (e.g. logo) or OK/NOK images in the visualization scene. If using OK/NOK images, link additionally a job result.

Results:

OK/NOK	Add an indicator with a red or green LED to show OK/NOK results. Link any job result to the indicator.
Value	Add a job result. Link any job result to the value (e.g. string or number).
Image/Pointcloud	Add an image or a pointcloud. Link any image or pointcloud of the job. Zoom in or out via mouse wheel. Press mouse wheel and move the position of the mouse to move the image or the pointcloud. **NOTE!** • Add a fix frame color or link a job result to display a green or red frame depending on the job result. • Add overlays in the image or the pointcloud. The following overlay types are available: » Point » Line » Arc » Circle » Coordinate System » Rectangle » Box » Polygon » Image » Multi Polygons 2D » Multi Lines 2D » Matching Result • Link input for overlay, select the representation and add text to the overlay (depending on the overlay type).

NOTE!
The visualization does not detect if the path at linked job results changes (e.g. when renaming modules). Make sure to finish renamings in the job tree first (within the software wenglor uniVision 3) and only then start creating the webbased visualization.

Parameters:

Unfreeze/Freeze	Add button “Unfreeze/Freeze” to the visualization scene. Pressing the button in Run Mode, freezes the current values for detailed analysis. Pressing it again continues to show regular updates of the visualization scene.
Trigger	Add button “Trigger” to the visualization scene. Pressing the button in Run Mode, sends a software trigger command to the Machine Vision Device if the Trigger Mode is set to On and the Trigger Source is set to Software. In offline mode, it remains on the current data of the Image Container Viewer.
Open Job	Add button “Open Job” to the visualization scene. Pressing the button in Run Mode, lists all available jobs of the projects folder on the device. Select one of the uniVision jobs to load it. **NOTE!** Saving the job in the software wenglor uniVision 3 includes also the visualization.

Settings:

Save Job	Save the job together with the visualization. **NOTE!** Saving the job in the software wenglor uniVision 3 includes also the visualization.

Browser with touch support

In order to edit the visualization on devices with touchscreen (e.g. tablet), adjust the browser-dependent settings.

Chrome:

- Open the Chrome browser.
- Enter “chrome://flags” to the address line.
- Activate “Touch Events AP” and “Touch initiated drag and drop”.

9. LIMA Interface

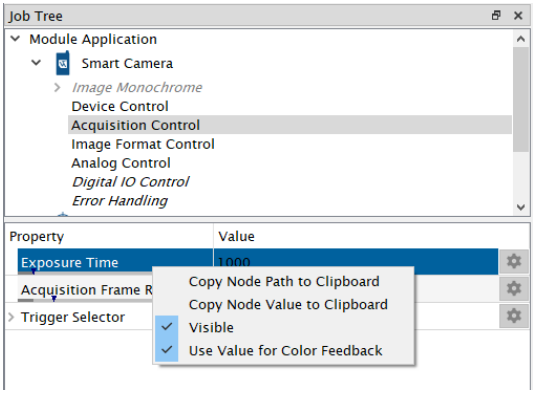
LIMA is an open wenglor network protocol based on XML in order to communicate with uniVision devices. The following points must be considered when working with LIMA commands:

- LIMA commands are based on query and associated response. Evaluate the LIMA response to know if and when the query was done successfully.
- Send LIMA commands sequentially. Do only send the next LIMA query after you have received the LIMA response to the previous query.

NOTE!

- Do not send e.g. with every PLC cycle the same LIMA command to the uniVision device. Just send the command once and then remove the command in the next PLC cycle at least until you have received the answer from the uniVision device.
- In case of too many commands (command overflow) the device answers with a LIMA error (Device Busy).

- Ensure data consistency (e.g. evaluate run counter in order to know if results are new).
- Adjust example commands accordingly via parameters (e.g. job name, path) so that they work accordingly. E.g. right click on properties in the software wenglor uniVision 3 and select “Copy Node Path to Clipboard” to get the path.



9.1 Jobs

Load Job

Description	Load a job from local projects folder of Machine Vision Device (/projects or complete path /media/card/projects)
Supported ports	LIMA Read Write Full LIMA Read Write Limited
Parameter	FILE: Job name (job with correct name must be available in projects folder on Machine Vision Device)
LIMA Command	<LIMA DIR="Request" CMD="Project_Load" FILE="Test.u3p"/>
LIMA Response	<LIMA DIR="ReplyOk" CMD="Project_Load"/>

NOTE!

- Loading another job sends automatically a project changed notification on all other LIMA Read Only connections and on the LIMA Read Write Full connection.
- If the job is not available in the projects folder, the LIMA response shows an error message (Lima Error on File Read).

Save Job

Description	Save current job in local projects folder of Machine Vision Device (/projects or complete path /media/card/projects)
Supported ports	LIMA Read Write Full
Parameter	FILE: Job name (job with such name is saved in projects folder on Machine Vision Device)
LIMA Command	<LIMA DIR="Request" CMD="Project_Save" FILE="Test.u3p"/>
LIMA Response	<LIMA DIR="ReplyOk" CMD="Project_Save"/>

Read Available Jobs

Description	Get list of all available jobs in projects folder of Machine Vision Device. Projects in sub-folders are not included in the LIMA response.
Supported ports	LIMA Read Write Full LIMA Read Write Limited LIMA Read Only
Parameter	VALUE: Returns the available jobs on the Machine Vision Device separated by semicolon.
LIMA Command	<LIMA DIR="Request" CMD="ProcessingInstance_GetJobs" />
LIMA Response	<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetJobs" VALUE="1.u3p;Find blobs.u3p"/>

9.2 Edit Job Tree

Read value of project path

Description	Read value of any project path in current job.
Supported ports	LIMA Read Write Full LIMA Read Write Limited LIMA Read Only
Parameter	PATH: Project path (get the relevant project path via context menu “Copy Node Path to Clipboard” in extended view of wenglor uniVision 3 software)
LIMA Command	<LIMA CMD="Project_GetNode" DIR="Request" PATH="Module Application. Smart Camera.Acquisition Control.Exposure Time"/>
LIMA Response	VALUE: Returns the value of the project path. <LIMA DIR="ReplyOk" CMD="Project_GetNode" PATH="Module Application. Smart Camera.Acquisition Control.Exposure Time" VALUE="1000"/>

Write value to project path

Description	Write value to project path in current job.
Supported ports	LIMA Read Write Full
Parameter	PATH: Project path (get the relevant project path via context menu “Copy Node Path to Clipboard” in extended view of wenglor uniVision 3 software) VALUE (for parameters): Desired new value of parameter defined by project path MIN (for outputs): Desired new minimum value of output defined by project path MAX (for outputs): Desired new maximum value of output defined by project path
LIMA Command	Changing value at parameter (e.g. Exposure Time at Smart Camera): <LIMA CMD="Project_SetNode" DIR="Request" PATH="Module Application. Smart Camera.Acquisition Control.Exposure Time" VALUE="2000"/> Changing minimum and maximum values at outputs (e.g. Min/Max for output Mean Grey Value at Module Image Region): <LIMA CMD="Project_SetNode" DIR="Request" PATH="Module Application. Module Image Region.Compute Mean.Mean Grey Value" MIN="100" MAX="200"/> Changing an option at a selector (e.g. activate auto focus option "Once" on B60 Smart Camera with Autofocus): <LIMA CMD="Project_SetNode" DIR="Request" PATH="Module Application. Smart Camera.Optic Control.Auto Focus" VALUE="1" />
LIMA Response	<LIMA DIR="ReplyOk" CMD="Project_SetNode"/>

Write value to cell in spreadsheet

Description	Write value to one cell in Module Spreadsheet.
Supported ports	LIMA Read Write Full

Parameter	PATH: Project path (get the relevant project path via context menu “Copy Node Path to Clipboard” in extended view of wenglor uniVision 3 software) CELLS: x@y (x: number of line starting with 0; y: number of column starting with 0; e.g. 0@1 means in first line and second column) VALUE: Value of project path (only possible for parameters)
LIMA Command	<LIMA DIR="Request" CMD="Project_SetSpreadSheetCells" CELLS= "0@1" PATH="Module Application.Module Spreadsheet" VALUE="Test"/>
LIMA Response	<LIMA DIR="ReplyOk" CMD="Project_SetSpreadSheetCells"/>

NOTE!
If the path is not valid, the device responds with a LIMA response error (e.g. Invalid path). Also if values for further parameters (e.g. VALUE, MIN, MAX) are not allowed, a LIMA error response is sent.

9.3 Read Image

Description	Read current image.
Supported ports	LIMA Read Write Full LIMA Read Write Limited LIMA Read Only
Parameter	TYPE: Defines the image format type RAW or BMP PATH: Defines the project path (get the relevant project path via context menu “Copy Node Path to Clipboard” in extended view of wenglor uniVision 3 software)
LIMA Command	**Monochrome camera** <LIMA DIR="Request" CMD="Project_GetImage" TYPE="RAW" PATH= "Module Application.Smart Camera.Image Monochrome.Grey"/> **Color camera (only possible if BGRA image is activated)** <LIMA DIR="Request" CMD="Project_GetImage" TYPE="RAW" PATH= "Module Application.Smart Camera.Image BGRA.BGRA"/> **Color camera (only possible if HSV image is activated)** <LIMA DIR="Request" CMD="Project_GetImage" TYPE="RAW" PATH= "Module Application.Smart Camera.Image HSV.Value"/> **Output image of module (e.g. Module Image Threshold)** <LIMA DIR="Request" CMD="Project_GetImage" TYPE="RAW" PATH= "Module Application.Module Image Threshold.Output Image.Binary"/>
LIMA Response	Returns the data length in byte (DATALEN) and the image. <LIMA DIR="ReplyOk" CMD="Project_GetImage" DATALEN="1555200"/> with image included

NOTE!
If the path is not valid, the device responds with a LIMA response error (e.g. Invalid path).

9.4 Trigger, Start and Stop

Trigger device

Description	Send software trigger signal to Machine Vision Device to capture new data (e.g. image). Trigger Mode must be On and Trigger Source must be set to Software.
Supported ports	LIMA Read Write Full LIMA Read Write Limited
LIMA Command	<T/>
LIMA Response	<TOk/>

Start/stop acquisition

Description	Start or stop acquisition on Machine Vision Device. If started, the device is ready to receive trigger signals. If stopped, the device is not ready to receive trigger signals. **NOTE!** • Trigger settings to capture data (e.g. images) are done separately (see section “7.2 Device Smart Camera for B60”). • After booting or loading job, acquisition is started automatically by default.
Supported ports	LIMA Read Write Full LIMA Read Write Limited
Parameter	STATE: on or off (on: starts acquisition; off: stops acquisition)
LIMA Command	**Start acquisition** <LIMA DIR="Request" CMD="Device_Acquisition" STATE="on"/> **Stop acquisition** <LIMA DIR="Request" CMD="Device_Acquisition" STATE="off"/>
LIMA Response	<LIMA DIR="ReplyOk" CMD="Device_Acquisition"/>

9.5 Get Status

Get device status

Description	Get device status
Supported ports	LIMA Read Write Full LIMA Read Write Limited LIMA Read Only
LIMA Command	<LIMA DIR="Request" CMD="Device_GetStatus"/>
LIMA Response	INFO: Provides detailed info about device status (e.g. Running) STATE: Returns a value for the color of the device status • ACTIVE (blue active) • LOCALIZE (blue blinking) • WARNING (yellow) • ERROR (red) VALUE: Returns decimal number for detailed device status. Converting it in binary number shows a bit converted info (starting with bit number 0 from right to left):

Bit number	STATE	INFO
0	WARNING	High CPU load
1	WARNING	High RAM
2	WARNING	High data memory
3	WARNING	High LAN Ethernet
7	WARNING	Too big changes in position
20	ERROR	Hardware error (e.g. "Autofocus module" if autofocus module on B60 Smart Camera is blocked)
21	ERROR	Crashed software service (with name of crashed software service)
30	LOCALIZE	Device is blinking

Examples for device status

<LIMA DIR="ReplyOk" CMD="Device_GetStatus" INFO=
"Running" STATE="ACTIVE" VALUE="0"/>

<LIMA DIR="ReplyOk" CMD="Device_GetStatus" INFO=
"High data memory" STATE="WARNING" VALUE="4"/>

<LIMA DIR="ReplyOk" CMD="Device_GetStatus" INFO=
"Too big changes in position" STATE="WARNING" VALUE="128"/>

<LIMA DIR="ReplyOk" CMD="Device_GetStatus" INFO=
"Autofocus module" STATE="ERROR" VALUE="1048576"/>

<LIMA DIR="ReplyOk" CMD="Device_GetStatus" INFO=
"Service software crashed" STATE="ERROR" VALUE="2097152"/>

<LIMA DIR="ReplyOk" CMD="Device_GetStatus" INFO=
"Device is blinking" STATE="LOCALIZE" VALUE="1073741824"/>

Get Processing Instance status

Description	Get status of processing instance
Supported ports	LIMA Read Write Full LIMA Read Write Limited LIMA Read Only
LIMA Command	<LIMA DIR="Request" CMD="ProcessingInstance_GetStatus"/>

LIMA Response

INFO: Provides detailed info about status of Processing Instance (e.g. Running or data overflow)
STATE: Returns a value for the color of the Processing Instance status (shown on Tab Jobs on device website, see operating instruction of Machine Vision Device for details).

- ACTIVE (blue)
- WARNING (yellow)
- ERROR (red)

VALUE: Returns decimal number for detailed status of Processing Instance. Converting it in binary number shows a bit converted info (starting with bit number 0):

Bit number	STATE or INFO	Level (WARNING or ERROR)
0		WARNING
1		ERROR
3	Data overflow	WARNING
4	Command overflow	WARNING
9	FTP interface	WARNING
10	Unlicensed module(s)	WARNING
20	Incompatible project	ERROR
21	Project not available	ERROR
22	Processing	ERROR

Examples for status of Processing Instance

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "Running" STATE="ACTIVE" VALUE="0"/>

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "Data overflow" STATE="WARNING" VALUE="9"/>

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "Command overflow" STATE="WARNING" VALUE="17"/>

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "FTP interface" STATE="WARNING" VALUE="513"/>

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "Unlicensed Module(s)" STATE="WARNING" VALUE="1025"/>

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "Incompatible project" STATE="ERROR" VALUE="1048578"/>

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "Project not available" STATE="ERROR" VALUE="2097154"/>

<LIMA DIR="ReplyOk" CMD="ProcessingInstance_GetStatus" INFO= "Processing" STATE="ERROR" VALUE="4194304"/>

LIMA Response	**NOTE!** For details about status of Processing Instance, see section “5.6.5 Status of Processing Instance”.

9.6 Teach+

Start Tech+ recording

Description	Start recording of Teach+ file (see section “6.4 Teach+”).
Supported ports	LIMA Read Write Full
Parameter	IMAGECOUNT: Number of data to be recorded in the Teach+ file.
LIMA Command	<LIMA DIR="Request" CMD="TeachPlus_Start" IMAGECOUNT="10"/>
LIMA Response	<LIMA DIR="ReplyOk" CMD="TeachPlus_Start" IMAGECOUNT="10"/>

Get status of Teach+ recording

Description	Get status of Teach+ recording (see section “6.4 Teach+”).
Supported ports	LIMA Read Write Full
LIMA Command	<LIMA DIR="Request" CMD="TeachPlus_GetState"/>
LIMA Response	PENDING: Lists still open records for Teach+ recording. STATE: Returns the state of Teach+ recording (e.g. off, recording images, writing archive) <LIMA DIR="ReplyOk" CMD="TeachPlus_GetState" PENDING="10" STATE= "recording images"/>

Cancel Teach+ recording

Description	Cancel Teach+ recording and save the Teach+ file with the so far recorded data (see section “6.4 Teach+”).
Supported ports	LIMA Read Write Full
LIMA Command	<LIMA DIR="Request" CMD="TeachPlus_Cancel"/>
LIMA Response	<LIMA DIR="ReplyOk" CMD="TeachPlus_Cancel"/>

10. Third-Party Licenses

Third Party License information of the software are available on the PC where the software is installed (see “6.3.1 Menu Bar”) or on the wenglor website:

https://www.wenglor.com/License/s/License

11. Change Index of Operating Instructions

Version	Date	Description	Associated software version
1.0.0	19.09.2023	Initial version for the market introduction	Software uniVision 3.0.1
1.0.1	10.10.2023	Description added for Robot Vision and Module HALCON Script	Software uniVision 3.0.1
1.0.2	24.11.2023	Description added for software modules and webbased visualization.	Software uniVision 3.0.2
1.1.0	11.12.2023	Updated description for version 3.1.0 of Software wenglor uniVision.	Software uniVision 3.1.0

12. Attachments

12.1 Module States

For details about module states see section “5.6.4 Module States”.

List of all module states:

0	no error
1001	undefined
1010	Input value error
1011	Return value error
1012	Internal data error
1020	Alignment error
1030	Function not implemented
1040	Image not linked
1041	Point cloud not linked
1050	Invalid operation
1060	Module Timeout
1098	Exception bad allocation
1099	Exception
1100	Module unlicensed
1101	Module init failed
1102	Device not available
1103	Data Loss

1104	Module not taught
1105	Unsupported pixel format
1106	No filtered tracking points found
1107	Invalid shape models
1108	No valid calibration
1111	Module configuration error
1112	There is an error concerning the SD card access or the SSD access.
1113	There is an error concerning the FTP interface
1114	There is an error concerning the TCP interface
10000-19999	Internal error of data structure
21201	Wrong type of control parameter: 1
21202	Wrong type of control parameter: 2
21203	Wrong type of control parameter: 3
21204	Wrong type of control parameter: 4
21205	Wrong type of control parameter: 5
21206	Wrong type of control parameter: 6
21207	Wrong type of control parameter: 7
21208	Wrong type of control parameter: 8
21209	Wrong type of control parameter: 9
21210	Wrong type of control parameter: 10
21211	Wrong type of control parameter: 11
21212	Wrong type of control parameter: 12
21213	Wrong type of control parameter: 13
21214	Wrong type of control parameter: 14
21215	Wrong type of control parameter: 15
21216	Wrong type of control parameter: 16
21217	Wrong type of control parameter: 17
21218	Wrong type of control parameter: 18
21219	Wrong type of control parameter: 19
21220	Wrong type of control parameter: 20
21301	Wrong value of control parameter: 1
21302	Wrong value of control parameter: 2
21303	Wrong value of control parameter: 3
21304	Wrong value of control parameter: 4
21305	Wrong value of control parameter: 5
21306	Wrong value of control parameter: 6
21307	Wrong value of control parameter: 7
21308	Wrong value of control parameter: 8
21309	Wrong value of control parameter: 9
21310	Wrong value of control parameter: 10
21311	Wrong value of control parameter: 11
21312	Wrong value of control parameter: 12
21313	Wrong value of control parameter: 13

21314	Wrong value of control parameter: 14
21315	Wrong value of control parameter: 15
21316	Wrong value of control parameter: 16
21317	Wrong value of control parameter: 17
21318	Wrong value of control parameter: 18
21319	Wrong value of control parameter: 19
21320	Wrong value of control parameter: 20
21350	Wrong value of component (see reset_obj_db())
21351	Wrong value of gray value component (see reset_obj_db())
21401	Wrong number of values of control parameter: 1
21402	Wrong number of values of control parameter: 2
21403	Wrong number of values of control parameter: 3
21404	Wrong number of values of control parameter: 4
21405	Wrong number of values of control parameter: 5
21406	Wrong number of values of control parameter: 6
21407	Wrong number of values of control parameter: 7
21408	Wrong number of values of control parameter: 8
21409	Wrong number of values of control parameter: 9
21410	Wrong number of values of control parameter: 10
21411	Wrong number of values of control parameter: 11
21412	Wrong number of values of control parameter: 12
21413	Wrong number of values of control parameter: 13
21414	Wrong number of values of control parameter: 14
21415	Wrong number of values of control parameter: 15
21416	Wrong number of values of control parameter: 16
21417	Wrong number of values of control parameter: 17
21418	Wrong number of values of control parameter: 18
21419	Wrong number of values of control parameter: 19
21420	Wrong number of values of control parameter: 20
21500	Number of input objects too big
21501	Wrong number of values of object parameter: 1
21502	Wrong number of values of object parameter: 2
21503	Wrong number of values of object parameter: 3
21504	Wrong number of values of object parameter: 4
21505	Wrong number of values of object parameter: 5
21506	Wrong number of values of object parameter: 6
21507	Wrong number of values of object parameter: 7
21508	Wrong number of values of object parameter: 8
21509	Wrong number of values of object parameter: 9
21510	Number of output objects too big
22000	Wrong specification of parameter (error in file: xxx.def)
22001	Initialize HALCON: reset_obj_db(Width,Height,Components)
22002	Used number of symbolic object names too big

22003	No license found
22004	Lost connection to license server
22005	No modules in license (no VENDOR_STRING)
22006	No license for this operator
22007	Time zone offset from GMT is > 24 hours
22008	Vendor keys do not support this platform
22009	Bad vendor keys
22010	Unknown vendor key type
22011	malloc() call failed
22012	Vendor keys have expired
22013	Second call to lc_init() (multiple jobs), and vendor keys do not support multiple jobs
22014	Vendor key data not supplied
22015	lmclient.h/liblmgr.a version mismatch
22016	Networking software not available on this machine
22017	Old vendor keys supplied
22018	License key in license file does not match other data in file
22019	Encryption handshake with daemon failed
22020	‘key’ structure is incorrect type, or feature Err:520 NULL, or num_licenses Err:520 0
22021	System clock has been set back. This error can only occur when the FEATURE line n contains an expiration date
22022	Version argument is invalid floating point format
22023	License server busy starting another copy of itself -0 retry
22024	Cannot establish a connection with a license server
22025	Feature is queued. lc_status will determine when it is available
22026	Vendor keys do not support this function
22027	Checkout request filtered by the vendor-defined filter routine
22028	Checkout exceeds MAX specified in options file
22029	All licenses in use
22030	No license server specified for counted license
22031	Cannot find feature in the license file
22032	Server has different license file than client -0 client’s license has feature, but server’s does not
22033	License file does not support a version this new
22034	This platform not authorized by license -0 running on platform not included in PLATFORMS list
22035	License server busy -0 the request should be retried (This is a rare occurrence)
22036	could not find license.dat
22037	Invalid license file syntax
22038	Cannot connect to a license server
22039	No TCP license service exists
22040	No socket connection to license manager server
22041	Invalid host
22042	Feature has expired
22043	Invalid date format in license file

22044	Invalid returned data from license server
22045	Cannot find SERVER hostname in network database
22046	Cannot read data from license server
22047	Cannot write data to license server
22048	Error in select system call
22049	Feature checkin failure detected at license
22050	Users are queued for this feature
22051	License server does not support this version of this feature
22052	Request for more licenses than this feature supports
22053	Cannot read /dev/kmem
22054	Cannot read /vmunix
22055	Cannot find ethernet device
22056	Cannot read license file
22057	Feature not yet available (wrong time/date set?)
22058	No such attribute
22059	Clock difference too large between client and server
22060	Feature database corrupted in daemon
22061	Duplicate selection mismatch for this feature
22062	User/host on EXCLUDE list for feature
22063	User/host not on INCLUDE list for feature
22064	Feature was never checked out
22065	Invalid FLEXlm key data supplied
22066	Clock setting check not available in daemon
22067	Date too late for binary format
22068	FLEXlm not initialized
22069	Server did not respond to message
22070	Request rejected by vendor-defined filter
22071	No FEATURESET line present in license file
22072	Incorrect FEATURESET line in license file
22073	Cannot compute FEATURESET line
22074	socket() call failed
22075	setsockopt() failed
22076	Message checksum failure
22077	Cannot read license file from server
22078	Not a license administrator
22079	lmremove request too soon
22080	Attempt to read beyond the end of LF path
22081	SYS$SETIMR call failed
22082	Internal FLEXlm Error -0 Please report to Globetrotter Software
22083	FLEXadmin API functions not available
22084	Invalid PACKAGE line in license file
22085	Server FLEXlm version older than client's
22086	Incorrect number of USERS/HOSTS INCLUDED in options file – see server log

22087	Server doesn't support this request
22088	This license object already in use
22089	Future license file format or misspelling in license file
22090	Feature removed during lmreread or wrong SERVER line hostid
22091	This feature is available in a different license pool
22092	Network connect to THIS_HOST failed
22093	Server node is down or not responding
22094	The desired vendor daemon is down
22095	The decimal format license is typed incorrectly
22096	All licenses are reserved for others
22097	Terminal Server remote client not allowed
22098	Cannot borrow that long
22099	License server out of network connections
22100	Wrong index for output object parameter
22101	Wrong index for input object parameter
22102	Wrong index for image object (too big or too small)
22103	Wrong number region/image component (see: HGetComp)
22104	Wrong relation name
22105	Access to undefined gray value component
22106	Wrong image width
22107	Wrong image height
22108	Undefined gray value component
22200	Inconsistent data of data base (typing)
22201	Wrong index for input control parameter
22202	Data of data base not defined (internal error)
22203	Number of operators too big
22205	User extension not properly installed
22206	Number of packages too large
22207	No such package installed
22300	Dongle not attached, or can't read dongle
22301	Missing Dongle Driver
22302	FLEXlock checkouts attempted
22303	SIGN attribute required
22304	CRO not supported for this platform
22305	BORROW failed
22306	BORROW period has expired
22307	FLOAT_OK license must have exactly one dongle hostid
22308	Unable to delete local borrow info
22309	Support for returning a borrowed license early is not enabled
22310	Error returning borrowed license on server
22311	Error when trying to checkout just a PACKAGE(BUNDLE)
22312	Composite Hostid not initialized
22313	An item needed for Composite Hostid missing or invalid

22314	Borrowed license doesn't match any known server license
22315	Error enabling event log
22316	Event logging is disabled
22317	Error writing to event log
22318	Timeout
22319	Bad message command
22320	Error writing to socket, peer has closed socket
22321	Attempting to generate version specific license tied to a single hostid, which is composite
22322	Version-specific signatures are not supported for uncounted licenses
22323	License template contains redundant signature specifiers
22324	Invalid V71_LK signature
22325	Invalid V71_SIGN signature
22326	Invalid V80_LK signature
22327	Invalid V80_SIGN signature
22328	Invalid V81_LK signature
22329	Invalid V81_SIGN signature
22330	Invalid V81_SIGN2 signature
22331	Invalid V84_LK signature
22332	Invalid V84_SIGN signature
22333	Invalid V84_SIGN2 signature
22334	License key required but missing from the license certificate
22335	Bad AUTH signature
22336	TS record invalid
22337	Cannot open TS
22338	Invalid Fulfillment record
22339	Invalid activation request received
22340	No fulfillment exists in trusted storage which matches the request
22341	Invalid activation response received
22342	Can't return the fulfillment
22343	Return would exceed max count(s)
22344	No repair count left
22345	Specified operation is not allowed
22346	User/host on EXCLUDE list for entitlement
22347	User/host not in INCLUDE list for entitlement
22348	Activation error
22349	Invalid date format in trusted storage
22350	Message encryption failed
22351	Message decryption failed
22352	Bad filter context
22353	SUPERSEDE feature conflict
22354	Invalid SUPERSEDE_SIGN syntax
22355	SUPERSEDE_SIGN does not contain a feature name and license signature
22356	ONE_TS_OK is not supported in this Windows Platform

22357	Internal error -178
22358	Only one terminal server remote client checkout is allowed for this feature
22359	Internal error -180
22360	Internal error -181
22361	Internal error -182
22362	More than one Ethernet hostid not supported in composite hostid definition
22363	The number of characters in the license file paths exceeds the permissible limit
22364	Invalid TZ keyword syntax
22365	Invalid time zone override specification in the client
22366	The time zone information could not be obtained
22367	License client time zone not authorized for license rights
22368	Invalid syntax for VM_PLATFORMS keyword
22369	Feature can be checked out from physical machine only
22370	Feature can be checked out from virtual machine only
22371	Vendor keys do not support Virtualization feature
22372	Checkout request denied as it exceeds the MAX limit specified in the options file
22373	Binding agent API -0 Internal error
22374	Binding agent communication error
22375	Invalid Binding agent version
22452	HALCON id out of range
22800	Wrong hardware knowledge file format
22801	Wrong hardware knowledge file version
22802	Error while reading the hardware knowledge
22803	Error while writing the hardware knowledge
22804	Tag in hardware knowledge file not found
22805	No cpu information in hardware knowledge file found
22806	No aop information in hardware knowledge file found
22807	No aop information for this HALCON variant found
22808	No aop information for this HALCON architecture found
22809	No aop information for specified Operator found
22810	Unknown aop model
22811	Wrong tag derivate in hardware knowledge file
22812	Internal error while processing hardware knowledge
22813	Optimizing aop was canceled
22830	Wrong access to global variable
22831	Used global variable does not exist
22832	Used global variable not accessible via GLOBAL_ID
22835	HALCON server to terminate is still working on a job
22837	No such HALCON software agent
22838	Hardware check for parallelization not possible on a single-processor machine
22839	Sequential HALCON does not support parallel hardware check (use Parallel HALCON instead)
22840	Initialization of agent failed

22841	Termination of agent failed
22842	Inconsistent hardware description file
22843	Inconsistent agent information file
22844	Inconsistent agent knowledge file
22845	The file with the parallelization information does not match to the currently HALCON version/revision
22846	The file with the parallelization information does not match to the currently used machine
22847	Inconsistent knowledge base of HALCON software agent
22848	Unknown communication type
22849	Unknown message type for HALCON software agent
22850	Error while saving the parallelization knowledge
22851	Wrong type of work information
22852	Wrong type of application information
22853	Wrong type of experience information
22854	Unknown name of HALCON software agent
22855	Unknown name and communication address of HALCON software agent
22856	cpu representative (HALCON software agent) not reachable
22857	cpu refuses work
22858	Description of scheduling resource not found
22859	Not accessible function of HALCON software agent
22860	Wrong type: HALCON scheduling resource
22861	Wrong state: HALCON scheduling resource
22862	Unknown parameter type: HALCON scheduling resource
22863	Unknown parameter value: HALCON scheduling resource
22864	Wrong post processing of control parameter
22867	Error while trying to get time (time query)
22868	Error while trying to get the number of processors
22869	Error while accessing temporary file
22900	Error while forcing a context switch
22901	Error while accessing the cpu affinity
22902	Error while setting the cpu affinity
22950	Wrong synchronization object
22952	Wrong thread object
22953	Input Object was not initialized
22954	Input control parameter is not initialized
22955	Output Object parameter is not initialized
22956	Output control parameter is not initialized
22970	creation of pthread failed
22971	pthread-detach failed
22972	pthread-join failed
22973	Initialization of mutex variable failed
22974	Deletion of mutex variable failed
22975	Lock of mutex variable failed

22976	Unlock of mutex variable failed
22977	failed to signal pthread condition variable
22978	failed to wait for pthread condition variable
22979	failed to init pthread condition variable
22980	failed to destroy pthread condition variable
22981	failed to signal event
22982	failed to wait for an event
22983	failed to init an event
22984	failed to destroy an event
22985	failed to create a tsd key
22986	failed to set a tsd key
22987	failed to get a tsd key
22988	failed to free a tsd key
22989	aborted waiting at a barrier
22990	‘Free list’ is empty while scheduling
22991	Communication partner not checked in
22992	you cannot start the communication system while running it
22993	Communication partner not checked in
23010	Region completely outside of the image domain
23011	Region (partially) outside of the definition range of the image
23012	Intersected definition range region / image empty
23013	Image with empty definition range (> no gray values)
23014	No common image point of two images
23015	Wrong region for image (first row < 0)
23016	Wrong region for image (column in last row > image width)
23017	Number of images unequal in input parameters
23018	Image height too small
23019	Image width too small
23020	Internal error: multiple call of HRLInitSeg()
23021	Internal error: HRLSeg() not initialized
23022	Wrong size of filter for Gauss
23033	Filter size exceeds image size
23034	Filter size have to be odd
23035	Filter is too big
23036	Input region is empty
23040	Row value of a coordinate > 2^15-1
23041	Row value of a coordinate < -2 ^15
23042	Column value of a coordinate > 2 ^15-1
23043	Column value of a coordinate < -2 ^15
23100	Wrong segmentation threshold
23101	Unknown feature
23102	Unknown gray value feature
23103	Internal error in HContCut

23104	Error in HContToPol: distance of points too big
23105	Error in HContToPol: contour too long
23106	Too many rows (IPImageTransform)
23107	Scaling factor 0.0 (IPImageScale)
23108	Wrong range in transformation matrix
23109	Internal error in IPvvf: no element free
23110	Number of input objects is zero
23111	At least one input object has an empty region
23112	Operation allowed for rectangular images 2**n only
23113	Too many relevant points (IPHysterese)
23114	Number of labels in image too big
23115	No labels with negative values allowed
23116	Wrong filter size (too small ?)
23117	Images with different image size
23118	Target image too wide or too far on the right
23119	Target image too narrow or too far on the left
23120	Target image too high or too far down
23121	Target image too low or too far up
23122	Number of channels in the input parameters are different
23123	Wrong color filter array type
23124	Wrong color filter array interpolation
23125	Homogeneous matrix does not represent an affine transformation
23126	Inpainting region too close to the image border
23127	Source and destination differ in size
23128	Too many Features
23129	Reflection axis undefined
23131	Concurrence Matrix: too little columns for quantization
23132	Concurrence Matrix: too little rows for quantization
23133	Wrong number of columns
23134	Wrong number of rows
23135	Number has too many digits
23136	Matrix is not symmetric
23137	Matrix is too big
23138	Wrong structure of file
23139	Lesser than 2 matrices
23140	Not enough memory
23141	Cannot read the file
23142	Cannot open file for writing
23143	Too many lookup table colors
23145	Too many Hough points (lines)
23146	Target image has got wrong height (not big enough)
23147	Wrong interpolation mode
23148	Region not compact or not connected

23170	Wrong filter index for filter size 3
23171	Wrong filter index for filter size 5
23172	Wrong filter index for filter size 7
23173	Wrong filter size only 3/5/7
23175	Number of suitable pixels too small to reliably estimate the noise
23200	Different number of entries/exits in HContCut
23250	Wrong XLD type
23252	Internal error: border point is set to FG
23253	Internal error: maximum contour length exceeded
23254	Internal error: maximum number of contours exceeded
23255	Contour too short for fetch_angle_xld
23256	Regression parameters of contours already computed
23257	Regression parameters of contours not yet entered! Please compute them by calling regress_cont_xld
23258	Data base: XLD object has been deleted
23259	Data base: object has no XLD-ID
23260	Internal error: wrong number of contour points allocated
23261	Contour attribute not defined
23262	Ellipse fitting failed
23263	Circle fitting failed
23264	All points classified as outliers (ClippingFactor too small)
23265	Quadrangle fitting failed
23266	No points found for at least one side of the rectangle
23267	A contour point lies outside of the image
23274	Not enough valid points for fitting the model
23275	No ARC/INFO world file
23276	No ARC/INFO generate file
23278	Unexpected end of file while reading DXF file
23279	Cannot read DXF-group code from file
23280	Inconsistent number of attributes per point in DXF file
23281	Inconsistent number of attributes and names in DXF file
23282	Inconsistent number of global attributes and names in DXF file
23283	Cannot read attributes from DXF file
23284	Cannot read global attributes from DXF file
23285	Cannot read attribute names from DXF file
23286	Wrong generic parameter name
23289	Internal DXF I/O error: Wrong data type
23290	Isolated point while contour merging
23291	Constraints (MaxError/MaxDistance) cannot be fulfilled
23300	Syntax error in file for training
23301	Maximum number of attributes per example exceeded
23302	Not possible to open file for training
23303	Too many data sets for training

23304	Wrong key for data for training
23305	Too many examples for one data set for training
23306	Too many classes
23307	Maximum number of cuboids exceeded
23308	Not possible to open classificator's file
23309	Error while saving the classificator
23310	Not possible to open protocol file
23311	Classificator with this name is already existent
23312	Maximum number of classificators exceeded
23313	Name of classificator is too long, > 20
23314	Classificator with this name is not existent
23315	Current classificator is not defined
23316	Wrong id in classification file
23317	The version of the classifier is not supported
23318	Serialized item does not contain a valid classifier
23330	Wrong covariance initialization
23331	The version of the GMM training samples is not supported
23332	Wrong training sample format
23333	Invalid file format for Gaussian Mixture Model (GMM)
23334	The version of the Gaussian Mixture Model (GMM) is not supported
23335	Internal error while training the GMM
23336	Singular covariance matrix
23337	No samples for at least one class
23338	Too few samples for at least one class
23340	GMM has not been trained yet
23341	No training samples stored in the classifier
23342	Serialized item does not contain a valid Gaussian Mixture Model (GMM)
23350	Unknown output function
23351	Target vector not in 0-1 encoding
23352	No training samples stored in the classifier
23353	Invalid file format for MLP training samples
23354	The version of the MLP training samples is not supported
23355	Wrong training sample format
23356	MLP is not a classifier use OutputFunction 'softmax' in create_class_mlp
23357	Invalid file format for multilayer perceptron (MLP)
23358	The version of the multilayer perceptron (MLP) is not supported
23359	Wrong number of image channels
23360	Number of MLP parameters too large
23361	Serialized item does not contain a valid multilayer perceptron (MLP)
23370	Wrong number of image channels
23371	A look-up table can be build only for a 2 or 3 channel classifier
23372	Cannot create a look-up table. Please choose a larger 'bit_depth' or select 'fast' for 'class_selection'.

23380	No training samples stored in the classifier
23381	Invalid file format for SVM training samples
23382	The version of the SVM training samples is not supported
23383	Wrong training sample format
23384	Invalid file format for support vector machine (SVM)
23385	The version of the support vector machine (SVM) is not supported
23386	Wrong class
23387	Nu was chosen too big
23388	SVM training failed
23389	Old SVM and new SVM do not match
23390	SVM contains no trained support vectors
23391	Kernel is not an RBF kernel
23392	Train data does not contain all classes
23393	SVM not trained
23394	Classifier not trained
23395	Serialized item does not contain a valid support vector machine (SVM)
23401	Wrong rotation number
23402	Wrong letter for Golay element
23403	Wrong reference point
23404	Wrong number of iterations
23405	Mophology: system error
23406	Wrong type of boundary
23407	Morphology: wrong number of input objects
23408	Morphology: wrong number of output objects
23409	Morphology: wrong number of input control parameter
23410	Morphology: wrong number of output control parameter
23411	Morphology: structuring element is infinite
23412	Morphology: wrong name for structuring element
23500	Wrong number of run length rows (chords): smaller than 0
23501	Number of chords too big. Increase ‘current_runlength_number’ using set_system!
23502	Run length row with negative length
23503	Run length row > image height
23504	Run length row < 0
23505	Run length column > image width
23506	Run length column < 0
23507	For CHORD_TYPE: Number of row too big
23508	For CHORD_TYPE: Number of row too small
23509	For CHORD_TYPE: Number of column too big
23510	Exceeding the maximum number of run lengths while automatical expansion
23511	Internal error: Region->compl neither TRUE/FALSE
23512	Internal error: Region->max_num < Region->num
23513	Internal error: number of chords too big for num_max
23514	Operator cannot be implemented for complemented “

23520	Image width < 0
23521	Image width > MAX_FORMAT
23522	Image height < 0
23523	Image height > MAX_FORMAT
23524	Image width < 0
23525	Image height < 0
23550	Too many segments
23551	‘int8’ images are available on 64 bit systems only
23600	Point at infinity cannot be converted to a Euclidean point
23601	Covariance matrix could not be determined
23602	RANSAC algorithm didn’t find enough point correspondences
23603	RANSAC algorithm didn’t find enough point correspondences
23604	Internal diagnosis: fallback method had to be used
23605	Projective transformation is singular
23606	Mosaic is under-determined
23607	Input covariance matrix is not positive definite
23620	Inconsistent number of point correspondences
23621	At least one image cannot be reached from the reference image
23622	The image with specified index does not exist
23623	Matrix is not a camera matrix
23624	Skew is not zero
23625	Illegal focal length
23626	Distortion is not zero
23627	It is not possible to determine all parameters for variable camera parameters
23628	No valid implementation selected
23629	Kappa can only be determined with the gold-standard method
23630	Conflicting number of images and projection mode
23631	Error in projection: Point not in any cube map
23632	No solution found
23640	Illegal combination of estimation method and parameters to be determined
23650	Invalid file format for FFT optimization data
23651	The version of the FFT optimization data is not supported
23652	Optimization data was created with a different HALCON variant (Sequential HALCON / Parallel HALCON)
23653	Storing of the optimization data failed
23654	Serialized item does not contain valid FFT optimization data
23660	No contours suitable for self-calibration found
23661	No stable solution found: please change the inlier threshold or select contours manually
23662	Instable solution: please choose more or different contours
23663	Not enough contours for calibration: please select contours manually
23700	Epipoles are within the image domain: no rectification possible.
23701	Fields of view of both cameras do not intersect each other.
23750	Invalid sheet-of-light handle

23751	No sheet-of-light model available
23752	Wrong input image size (width)
23753	Wrong input image size (height)
23754	The bounding-box around the profile region does not fit the domain of definition of the input image
23755	Calibration extend not set
23756	Undefined disparity image
23757	Undefined domain for disparity image
23758	Undefined camera parameter
23759	Undefined pose of the lightplane
23760	Undefined pose of the camera coordinate system
23761	Undefined transformation from the coordinate system of the camera to the coordinate system of the lightplane
23762	Undefined movement pose for xyz calibration
23763	Wrong value of scale parameter
23764	Wrong parameter name
23765	Wrong type of parameter method
23766	Wrong type of parameter ambiguity
23767	Wrong type of parameter score
23768	Wrong type of parameter calibration
23769	Wrong type of parameter number_profiles
23770	Wrong type of element in parameter camera_parameter
23771	Wrong type of element in pose
23772	Wrong value of parameter method
23773	Wrong type of parameter min_gray
23774	Wrong value of parameter ambiguity
23775	Wrong value of parameter score_type
23776	Wrong value of parameter calibration
23777	Wrong value of parameter number_profiles
23778	Wrong type of camera
23780	Wrong number of values of pose
23850	The light source positions are linearly dependent
23851	No sufficient image indication
23852	Internal error: Function has equal signs in HZBrent
23900	Kalman: Dimension n,m or p has got a undefined value
23901	Kalman: File does not exist
23902	Kalman: Error in file (row of dimension)
23903	Kalman: Error in file (row of marking)
23904	Kalman: Error in file (value is no float)
23905	Kalman: Matrix A is missing in file
23906	Kalman: Matrix C is missing in file
23907	Kalman: Matrix Q is missing in file
23908	Kalman: Matrix R is missing in file
23909	Kalman: G or u is missing in file

23910	Kalman: Covariant matrix is not symmetric
23911	Kalman: Equation system is singular
24050	Image data management: object is a object tuple
24051	Image data management: object has been deleted already
24052	Image data management: wrong object-ID
24053	Image data management: object tuple has been deleted already
24054	Image data management: wrong object tuple-ID
24055	Image data management: object tuple is a object
24056	Image data management: object-ID is NULL (0)
24057	Image data management: object-ID outside the valid range
24058	Image data management: access to deleted image
24059	Image data management: access to image with wrong key
24060	Image data management: access to deleted region
24061	Image data management: access to region with wrong key
24062	Image data management: wrong value for image channel
24063	Image data management: index too big
24064	Image data management: index not defined
24100	No OpenCL available
24101	OpenCL Error occured
24102	No compute device available
24104	Out of compute device memory
24105	Invalid work group shape
24106	Invalid compute device
25100	Wrong (logical) window number
25101	Error while opening the window
25102	Wrong window coordinates
25103	It is not possible to open another window
25104	Device resp. operator not available
25105	Unknown color
25106	No window has been opened for desired action
25107	Wrong filling mode for regions (fill or margin)
25108	Wrong gray value (0..255)
25109	Wrong pixel value (use value of get_pixel(P) only)
25110	Wrong line width (see: query_line_width(Min,Max))
25111	Wrong name of cursor
25112	Wrong color table (see: query_lut(Name))
25113	Wrong representation mode (see: query_insert(Mode))
25114	Wrong representation color (see: query_color(List))
25115	Wrong dither matrix (binary image representation)
25116	Wrong image transformation (name or image size)
25117	Unsuitable image type for image transformation
25118	Wrong zooming factor for image transformation
25119	Wrong representation mode

25120	Wrong code of device
25121	Wrong number for father window
25122	Wrong window size
25123	Wrong window type
25124	No current window has been set
25125	Wrong color combination or range (RGB)
25126	Wrong number of pixels set
25127	Wrong value for comprise (object or image)
25128	set_fix with 1/4 image levels and static not valid
25129	set_lut not valid in child windows
25130	Number of concurrent used color tables is too big
25131	Wrong device for window dump
25132	Wrong window size for window dump
25133	System variable DISPLAY (setenv) not defined
25134	Wrong thickness for window margin
25135	System variable DISPLAY has been set wrong (<host>:0.0)
25136	Too many fonts loaded
25137	Wrong font name
25138	No valid cursor position
25139	Window is not a textual window
25140	Window is not an image window
25141	String too long or too high
25142	Too little space in the window rightwards
25143	Window is not suitable for the mouse
25144	Here Windows on a equal machine is permitted only
25145	Wrong mode while opening a window
25146	Wrong window mode for operation
25147	Operation not possible with fixed pixel
25148	Color tables for 8 image levels only
25149	Wrong mode for pseudo real colors
25150	Wrong pixel value for LUT
25151	Wrong image size for pseudo real colors
25152	Error in procedure HRLUT
25153	Wrong number of entries in color table for set_lut
25154	Wrong values for image area
25155	Wrong line pattern
25156	Wrong number of parameters for line pattern
25157	Wrong number of colors
25158	Wrong value for mode of area creation (0,1,2)
25159	Spy window is not set (set_spy)
25160	No file for spy has been set (set_spy)
25161	Wrong parameter output depth (set_spy)
25162	Wrong window size for window dump

25163	Wrong color table: wrong file name or query_lut()
25164	Wrong color table: empty string ?
25165	Using this hardware set_lut('default') is allowed only
25166	Error while calling online help
25167	Row cannot be projected
25168	Operation is unsuitable using a computer with fixed color table
25169	Computer represents gray scales only (no colors)
25170	LUT of this display is full
25171	Internal error: wrong color code
25172	Wrong type for window attribute
25173	Wrong name for window attribute
25174	Negative height of area (or 0)
25175	Negative width of area (or 0)
25176	Window not completely visible
25177	Font not allowed for this operation
25178	Operation not possible (window was created in different thread)
25179	Depth was not stored with window
25180	Internal error: only RGB-Mode
25181	No more (image-)windows available
25182	Object index was not stored with window
25183	Operator does not support primitives without point coordinates
25184	Operator not available with Windows Remote Desktop
25185	No OpenGL support available
25186	No depth information available
25187	OpenGL error occurred
25188	Required framebuffer object is unsupported
25189	OpenGL accelerated hidden surface removal not supported on this machine
25190	Invalid window parameter
25191	Invalid value for window parameter
25192	Unknown mode
25195	Invalid value for navigation mode
25196	Internal file error
25197	Error while file synchronization
25198	Insufficient rights on file
25199	Bad file descriptor
25200	File not found
25201	Error while writing image data (sufficient memory ?)
25202	Error while writing image descriptor (sufficient memory ?)
25203	Error while reading image data (format of image too small ?)
25204	Error while reading image data (format of image too big ?)
25205	Error while reading image descriptor: file too small
25206	Image matrices are different
25207	Help file not found (setenv HALCONROOT <Halcon- Homedirectory>)

25208	Help index not found (setenv HALCONROOT <Halcon- Homedirectory>)
25209	File <standard_input> cannot be closed
25210	<standard_output/error> cannot be closed
25211	File cannot be closed
25212	Error while writing to file
25213	Exceeding of maximum number of files
25214	Wrong file name
25215	Error while opening the file
25216	Wrong file mode
25217	Wrong type for pixel (e.g. byte)
25218	Wrong image width (too big ?)
25219	Wrong image height (too big ?)
25220	File already exhausted before reading an image
25221	File exhausted before terminating the image
25222	Wrong value for resolution (dpi)
25223	Wrong output image size (width)
25224	Wrong output image size (height)
25225	Wrong number of parameter values: format description
25226	Wrong parameter name for operator
25227	Wrong slot name for parameter
25228	Operator class is missing in help file
25229	Wrong or inconsistent help/*.idx or help/*.sta
25230	File help/*.idx not found (setenv HALCONROOT <Halcon- Homedirectory>)
25231	File help/*.sta not found (setenv HALCONROOT <Halcon- Homedirectory>)
25232	Inconsistent file help/*.sta
25233	No explication file (.exp) found
25234	No file found in known graphic format
25235	Wrong graphic format
25236	Inconsistent file halcon.num
25237	File not a TIFF file
25238	Wrong file format
25239	gnuplot could not be started
25240	Output file for gnuplot could not be opened
25241	Not a valid gnuplot output stream
25242	No PNM format
25243	Inconsistent or old help file ($HALCONROOT/help)
25244	Wrong file handle
25245	File not open
25246	No files in use so far (none opened)
25247	Invalid file format for regions
25248	Error while reading region data: Format of region too big.
25250	Invalid handle for a serial connection
25251	Serial port not open

25252	No serial port available
25253	Could not open serial port
25254	Could not close serial port
25255	Could not get serial port attributes
25256	Could not set serial port attributes
25257	Wrong baud rate for serial connection
25258	Wrong number of data bits for serial connection
25259	Wrong flow control for serial connection
25260	Could not flush serial port
25261	Error during write to serial port
25262	Error during read from serial port
25270	Serialized item does not contain valid regions
25271	The version of the regions is not supported
25272	Serialized item does not contain valid images
25273	The version of the images is not supported
25274	Serialized item does not contain valid XLD objects
25275	The version of the XLD objects is not supported
25276	Serialized item does not contain valid objects
25277	The version of the objects is not supported
25280	File has not been opened in text format
25281	File has not been opened in binary file format
25282	Cannot create directory
25283	Cannot remove directory
25300	No image acquisition device opened
25301	Image acquisition: wrong color depth
25302	Image acquisition: wrong device
25303	Image acquisition: determination of video format not possible
25304	Image acquisition: no video signal
25305	Unknown image acquisition device
25306	Image acquisition: failed grabbing of an image
25307	Image acquisition: wrong resolution chosen
25308	Image acquisition: wrong image part chosen
25309	Image acquisition: wrong pixel ratio chosen
25310	Image acquisition: handle not valid
25311	Image acquisition: instance not valid (already closed?)
25312	Image acquisition: device cannot be initialized
25313	Image acquisition: external triggering not supported
25314	Image acquisition: wrong camera input line (multiplex)
25315	Image acquisition: wrong color space
25316	Image acquisition: wrong port
25317	Image acquisition: wrong camera type
25318	Image acquisition: maximum number of acquisition device classes exceeded
25319	Image acquisition: device busy

25320	Image acquisition: asynchronous grab not supported
25321	Image acquisition: unsupported parameter
25322	Image acquisition: timeout
25323	Image acquisition: invalid gain
25324	Image acquisition: invalid field
25325	Image acquisition: invalid parameter type
25326	Image acquisition: invalid parameter value
25327	Image acquisition: function not supported
25328	Image acquisition: incompatible interface version
25329	Image acquisition: could not set parameter value
25330	Image acquisition: could not query parameter setting
25331	Image acquisition: parameter not available in current configuration
25332	Image acquisition: device could not be closed properly
25333	Image acquisition: camera configuration file could not be opened
25334	Image acquisition: callback type not supported
25335	Image acquisition: device lost
25400	Image type is not supported
25401	Invalid pixel format
25402	Internal JPEG-XR error
25403	Invalid format string
25404	Maximum number of channels exceeded
25405	Unspecified error in JPEG-XR library
25406	Bad magic number in JPEG-XR library
25407	Feature not implemented in JPEG-XR library
25408	File read/write error in JPEG-XR library
25409	Invalid file format in JPEG-XR library
25500	Error while closing the image file
25501	Error while opening the image file
25502	Premature end of the image file
25503	Image dimensions too large for this file format
25504	Image too large for this HALCON version
25505	Too many iconic objects for this file format
25510	File is no PCX-File
25511	PCX: unknown encoding
25512	PCX: More than 4 image plains
25513	PCX: Wrong magic in color table
25514	PCX: Wrong number of bytes in span
25515	PCX: Wrong number of bits/pixels
25516	PCX: Wrong number of plains
25520	File is no GIF-File
25521	GIF: Wrong version (not 87a/89a)
25522	GIF: Wrong descriptor
25523	GIF: Wrong color table

25524	GIF: Premature end of file
25525	GIF: Wrong number of images ‘ ‘
25526	GIF: Wrong image extension ‘!’
25527	GIF: Wrong left top width
25528	GIF: Cyclic index of table
25529	GIF: Wrong image data
25530	File is no Sun-Raster-File
25531	SUN-Raster: Wrong header
25532	SUN-Raster: Wrong image width
25533	SUN-Raster: Wrong image height
25534	SUN-Raster: Wrong color map
25535	SUN-Raster: Wrong image data
25536	SUN-Raster: Wrong type of pixel
25540	XWD: Wrong type of pixel
25541	XWD: Wrong visual class
25542	XWD: Wrong X10 header
25543	XWD: Wrong X11 header
25544	XWD: Wrong X10 colormap
25545	XWD: Wrong X11 colormap
25546	XWD: Wrong pixmap
25547	XWD: unknown version
25548	XWD: Error while reading an image
25550	TIFF: Error while reading a file
25551	TIFF: Wrong colormap
25552	TIFF: Too many colors
25553	TIFF: Wrong photometric interpretation
25554	TIFF: Wrong photometric depth
25555	TIFF: Image is no binary file
25556	TIFF: Image format not supported by HALCON
25557	TIFF: Wrong specification of the TIFF file format
25558	TIFF: TIFF file is corrupt
25559	TIFF: A required TIFF tag is missing the TIFF file
25560	File is no BMP-File
25561	BMP: Premature end of file
25562	BMP: Incomplete header
25563	BMP: Unknown bitmap format
25564	BMP: Unknown compression format
25565	BMP: Wrong color table
25566	BMP: Write error on output
25567	BMP: File does not contain a binary image
25570	JPEG: wrong number of components in image
25571	JPEG: unknown error from libjpeg
25572	JPEG: no implemented feature in libjpeg

25573	JPEG: file access error in libjpeg
25574	JPEG: tmp file access error in libjpeg
25575	JPEG: memory error in libjpeg
25576	JPEG: Error in input image
25580	PNG: File is not a PNG file
25581	PNG: Unknown interlace type
25582	PNG: Unsupported color type
25583	PNG: Image is no binary file
25590	JPEG-2000: File corrupt
25591	JPEG-2000: Image has more than 28 significant bits
25592	JPEG-2000: Error while encoding
25600	Socket cannot be set to block
25601	Socket cannot be set to unblock
25602	Received data is no tuple
25603	Received data is no image
25604	Received data is no region
25605	Received data is no xld object
25606	Error while reading from socket
25607	Error while writing to socket
25608	Illegal number of bytes with get_rl
25609	Buffer overflow in read_data
25610	Socket cannot be created
25611	Bind on socket failed
25612	Socket information is not available
25613	Socket cannot listen for incoming connections
25614	Connection could not be accepted
25615	Connection request failed
25616	Hostname could not be resolved
25617	No data on socket
25618	Unknown tuple type on socket
25619	Timeout occurred on socket
25620	No more sockets available
25621	Socket is not initialized
25622	Invalid socket
25623	Socket is NULL
25624	Received data type is too large
25625	Wrong socket protocol
25626	Received data does not contain packed data
25627	Error when handling the parameter
25628	Format specification does not match the data
25629	Invalid format specification
25630	Received data is no serialized item
25678	XLD object data can only be read by HALCON XL

25700	Too many contours/polygons for this file format
25750	The version of the quaternion is not supported
25751	Serialized item does not contain a valid quaternion
25752	The version of the homogeneous matrix is not supported
25753	Serialized item does not contain a valid homogeneous matrix
25754	The version of the homogeneous 3D matrix is not supported
25755	Serialized item does not contain a valid homogeneous 3D matrix
25756	The version of the tuple is not supported
25757	Serialized item does not contain a valid tuple
25758	Tuple data can only be read on 64-bit systems
25759	The version of the camera parameters (pose) is not supported
25760	Serialized item does not contain valid camera parameters (pose)
25761	The version of the internal camera parameters is not supported
25762	Serialized item does not contain valid internal camera parameters
26000	Access to undefined memory area
26001	Not enough memory available
26002	Memory partition on heap has been overwritten
26003	HAlloc: 0 bytes requested
26004	Tmp-memory management: Call freeing memory although nothing had been allocated
26005	Tmp-memory management: Null pointer while freeing
26006	Tmp-memory management: could not find memory element
26007	Memory management: wrong memory type allocated
26021	Not enough video memory available
26040	System parameter for memory-allocation inconsistent
26041	No memory block allocated at last
26500	Process creation failed
27000	Wrong index for output control parameter
27001	Wrong number of values: output control parameter (see: HPut*Par)
27002	Wrong type: output control parameter (see: HPut*Par)
27003	Wrong data type for object key (input objects)
27004	Range for integer had been passed
27005	Inconsistent HALCON version
27006	Not enough memory for strings allocated
27007	Internal error: Proc is NULL
27100	Wrong list structure using input objects
27101	Wrong input object parameter (not bound)
27102	Wrong input control parameter (not bound)
27103	Wrong output object parameter (already bound)
27104	Wrong output control parameter (already bound)
27105	Unknown symbolic object key (input objects)
27200	Wrong number of output object parameter
27300	Wrong number of input parameter
27400	System error: output type <string> expected

27401	System error: output type <long> expected
27402	System error: output type <float> expected
27403	Object parameter is a zero pointer (‘_’ not allowed)
27404	Tuple had been deleted values are not valid any more
27430	CPP-interface internal error: wrong object mode
27431	Wrong number of regions (> 1) for type HRegion
27432	Wrong number of images (> 1) for type HImage
27433	Tuple with undefined values
27500	No contact to RPC server
27501	Error in remote procedure call
27600	Parameter value is neither a list nor a atom
28000	Unknown operator name
28001	register_comp_used is not activated (see set_system)
28002	Unknown operator class
28101	convol/mask: error while opening the file
28102	convol/mask: premature end of file
28103	convol/mask: conversion error
28104	convol/mask: wrong row-/column number
28105	convol/mask: mask size overflow
28106	convol/mask: too many elements entered
28107	convol: wrong margin type
28108	convol: no mask object has got empty region
28110	convol: Weight factor is 0
28111	convol: inconsistent number of weights
28112	rank: wrong rank value
28113	convol/rank: error while handling margin
28114	Error while parsing filter mask file
28120	Wrong number of coefficients for convolution (sigma too big?)
28200	No valid ID for data set
28201	No data set active (set_bg_esti)
28202	ID already used for data set (is not possible)
28204	No data set created (create_bg_esti)
28205	Not possible to pass an object list
28206	Image has other size than the background image in data set
28207	Up-date-region is bigger than background image
28208	Number of statistic data sets is too small
28209	Wrong value for adapt mode
28210	Wrong value for frame mode
28300	Maximum number of fonts exceeded
28301	Wrong ID (Number) for font
28302	OCR internal error: wrong ID
28303	OCR not initialized: no font was read in
28304	No font activated

28305	OCR internal error: wrong threshold in angle determination
28306	OCR internal error: wrong attribute
28307	The version of the OCR classifier is not supported
28308	OCR File: inconsistent number of nodes
28309	OCR File: File too short
28310	OCR: internal error 1
28311	OCR: internal error 2
28312	Wrong type of OCR tool (no ‘box’ or ‘net’)
28313	The version of the OCR training characters is not supported
28314	Image too large for training file
28315	Region too large for training file
28316	Protected training file
28317	Wrong password for protected training file
28318	Serialized item does not contain a valid OCR classifier
28320	Invalid file format for MLP classifier
28321	The version of the MLP classifier is not supported
28322	Serialized item does not contain a valid MLP classifier
28330	Invalid file format for SVM classifier
28331	The version of the SVM classifier is not supported
28332	Serialized item does not contain a valid k-NN classifier
28333	Invalid file format for k-NN classifier
28340	Invalid text model
28341	Invalid text result
28350	OCV system not initialized
28351	The version of the OCV tool is not supported
28353	Wrong name for an OCV object
28354	Training has already been applied
28355	No training has been applied to the character
28356	Serialized item does not contain a valid OCV tool
28370	Wrong number of function points
28371	List of values is not a function
28372	Wrong ordering of values (not ascending)
28373	Illegal distance of function points
28374	Function is not monotonic
28375	Wrong function type
28400	You have to indicate at least 3 calibration points
28402	No calibration table found
28403	Error while reading calibration table description file
28404	Minimum threshold while searching for ellipses
28405	Read error / format error in calibration table description file
28406	Error in projection: s_x 0 or s_y 0 or z 0
28407	Error in inverse projection
28408	Not possible to open camera parameter file

28409	Format error in file: no colon
28410	Format error in file: 2. colon is missing
28411	Format error in file: semicolon is missing
28412	Not possible to open camera parameter (pose) file
28413	Format error in camera parameter (pose) file
28414	Not possible to open calibration target description file
28415	Not possible to open postscript file of calibration target
28416	Error while norming the vector
28417	Fitting of calibration target failed
28418	No next mark found
28419	Normal equation system is not solvable
28420	Average quadratic error is too big for 3D position of mark
28421	Non elliptic contour
28422	Wrong parameter value slvand()
28423	Wrong function results slvand()
28424	Distance of marks in calibration target description file is not possible
28425	Specified flag for degree of freedom not valid
28426	Minimum error did not fall below
28427	Wrong type in Pose (rotation / translation)
28428	Image size does not match the measurement in camera parameters
28429	Point could not be projected into linescan image
28430	Diameter of calibration marks could not be determined
28431	Orientation of calibration plate could not be determined
28432	Calibration plate does not lie completely inside the image
28433	Wrong number of calibration marks extracted
28434	Unknown name of parameter group
28435	Focal length must be non-negative
28436	Function not available for cameras with telecentric lenses
28437	Function not available for line scan cameras
28438	Ellipse is degenerated to a point
28439	No orientation mark found
28440	Camera calibration did not converge
28441	Error in calibration data, try to recalibrate with improved input data!
28442	Point cannot be distorted
28451	Model not optimized yet -0 no results can be queried
28452	Model not post processed yet -0 no auxiliary results can be queried
28453	Calibration setup: fields of view do not intersect
28454	Camera type and camera parameters incompatible
28455	Calibration setup: incompatible camera types
28456	Camera type not supported
28457	Invalid camera index
28458	Invalid calibration object index
28459	Invalid calibration object pose index

28460	Undefined camera
28461	Indices: ambiguous observation index
28462	Undefined calibration object
28463	Invalid file format for calibration data model
28464	The version of the calibration data model is not supported
28465	Zero-motion in line scan camera parameters
28466	Calibration setup: multiple cameras and/or calibration objects not supported for camera type
28467	Incomplete observation data
28468	Invalid file format for camera setup model
28469	The version of the camera setup model is not supported
28470	Full HALCON calibration plate description required
28471	Invalid observation index
28472	Serialized item does not contain a valid camera setup model
28473	Serialized item does not contain a valid calibration data model
28474	Invalid tool pose index
28475	Undefined tool pose
28476	Feature or operation not supported for current calibration data model type
28490	Feature or operation not supported for current stereo model type
28491	Feature or operation available only in ‘persistent’ mode
28492	Invalid bounding box
28493	Image sizes must be identical with the corresponding camera parameters from the camera setup
28494	Bounding box lies partially or completely behind the base line of at least one camera pair
28495	Ambiguous calibration: Please, recalibrate with improved input data!
28496	Pose of calibration plate could not be determined!
28500	Invalid file format for template
28501	The version of the template is not supported
28502	Error during changing the file mode (t/b)
28503	Inconsistent match file: coordinates out of range
28505	The image(s) is not a pyramid (wrong zooming factor?)
28506	Number of template points too small
28507	Template data can only be read by HALCON XL
28508	Serialized item does not contain a valid NCC model
28509	Serialized item does not contain a valid template
28510	Number of shape model points too small
28511	Gray-value-based and color-based shape models cannot be searched simultaneously
28512	Shape model data can only be read by HALCON XL
28513	Shape model was not created from XLDs
28514	Serialized item does not contain a valid shape model
28530	Initial components have different region types
28531	Solution of ambiguous matches failed
28532	Computation of the incomplete gamma function not converged

28533	Too many nodes while computing the minimum spanning arborescence
28534	Component training data can only be read by HALCON XL
28535	Component model data can only be read by HALCON XL
28536	Serialized item does not contain a valid component model
28537	Serialized item does not contain a valid component training result
28540	Size of the training image and the variation model differ
28541	Variation model has not been prepared for segmentation
28542	Invalid variation model training mode
28543	Invalid file format for variation model
28544	The version of the variation model is not supported
28545	Training data has already been cleared
28546	Serialized item does not contain a valid variation model
28550	No more measure objects available
28551	Measure object is not initialized
28552	Invalid measure object
28553	Measure object is NULL
28554	Measure object has wrong image size
28555	Invalid file format for measure object
28556	The version of the measure object is not supported
28557	Measure object data can only be read by HALCON XL
28558	Serialized item does not contain a valid measure object
28570	Metrology model is not initialized
28571	Invalid metrology model
28572	Invalid metrology object
28573	Not enough valid measures for fitting the metrology object
28575	Invalid file format for metrology model
28576	The version of the metrology model is not supported
28577	Fuzzy function is not set
28578	Serialized item does not contain a valid metrology model
28600	Dynamic library could not be opened
28601	Dynamic library could not be closed
28602	Symbol not found in dynamic library
28650	Not enough information for radiometric calibration
28700	Unknown bar code
28701	Wrong number of modules
28702	Wrong number of elements
28703	Unknown character (for this code)
28705	wrong name for attribute in barcode descriptor
28706	Wrong thickness of element
28707	No region found
28708	Wrong type of bar code
28720	Invalid bar code handle
28721	List of bar code models is empty

28722	Training cannot be done for multiple bar code types
28723	Cannot get bar code type specific parameter with get_bar_code_param. Use get_bar_code_param_specific
28724	Cannot get this object for multiple bar code types. Try again with single bar code type
28725	Invalid file format for bar code model
28726	The version of the bar code model is not supported
28800	Specified code type is not supported
28801	Wrong foreground specified
28802	Wrong matrix size specified
28803	Wrong symbol shape specified
28804	Wrong generic parameter name
28805	Wrong generic parameter value
28806	Wrong symbol printing mode
28807	Symbol region too near to image border
28808	No rectangular module boundings found
28809	Couldn't identify symbol finder
28810	Symbol region with wrong dimension
28811	Classification failed
28812	Decoding failed
28813	Reader programing not supported
28820	General 2d data code error
28821	Corrupt signature of 2d data code handle
28822	Invalid 2d data code handle
28823	List of 2d data code models is empty
28825	Invalid 'Candidate' parameter
28829	Unexpected 2d data code error
28830	Invalid parameter value
28831	Unknown parameter name
28832	Invalid value for 'polarity'
28833	Invalid value for 'symbol_shape'
28834	Invalid symbol size
28835	Invalid module size
28836	Invalid value for 'module_shape'
28837	Invalid value for 'orientation'
28838	Invalid value for 'contrast_min'
28839	Invalid value for 'measure_thresh'
28840	Invalid value for 'alt_measure_red'
28841	Invalid value for 'slant_max'
28842	Invalid value for 'L_dist_max'
28843	Invalid value for 'L_length_min'
28844	Invalid module gap
28845	Invalid value for 'default_parameters'
28846	Invalid value for 'back_texture'

28847	Invalid value for ‘mirrored’
28848	Invalid value for ‘classificator’
28849	Invalid value for ‘persistence’
28850	Invalid model type
28851	Invalid value for ‘module_roi_part’
28852	Invalid value for ‘finder_pattern_tolerance’
28853	Invalid value for ‘mod_aspect_max’
28854	Invalid value for ‘small_modules_robustness’
28863	Invalid module aspect ratio
28864	Invalid layer num
28865	Wrong data code model file version
28866	Serialized item does not contain a valid 2D data code model
28900	Unknown parameter name
28901	Invalid value for ‘num_levels’
28902	Invalid value for ‘optimization’
28903	Invalid value for ‘metric’
28904	Invalid value for ‘min_face_angle’
28905	Invalid value for ‘min_size’
28910	The projected model is too large “
28920	Invalid value for ‘longitude_min’
28921	Invalid value for ‘longitude_max’
28922	Invalid value for ‘latitude_min’
28923	Invalid value for ‘latitude_max’
28924	Invalid value for ‘cam_roll_min’
28925	Invalid value for ‘cam_roll_max’
28926	Invalid value for ‘dist_min’
28927	Invalid value for ‘dist_max’
28928	Invalid value for ‘num_matches’
28929	Invalid value for ‘max_overlap’
28933	Invalid value for ‘border_model’
28940	Pose is not well-defined
28941	Invalid file format for 3D shape model
28960	Invalid file format for descriptor model
28961	The version of the descriptor model is not supported
28962	Invalid value for ‘radius’
28963	Invalid value for ‘check_neighbor’
28964	Invalid value for ‘min_check_neighbor_diff’
28965	Invalid value for ‘min_score’
28966	Invalid value for ‘sigma_grad’
28967	Invalid value for ‘sigma_smooth’
28968	Invalid value for ‘alpha’
28969	Invalid value for ‘threshold’
28970	Invalid value for ‘depth’

28971	Invalid value for ‘number_trees’
28972	Invalid value for ‘min_score_descr’
28973	Invalid value for ‘patch_size’
28974	Invalid value for ‘tilt’
28975	Invalid value for ‘guided_matching’
28976	Invalid value for ‘subpix’
28977	Too few feature points can be found
28978	Invalid value for ‘min_rot’
28979	Invalid value for ‘max_rot’
28980	Invalid value for ‘min_scale’
28981	Invalid value for ‘max_scale’
28982	Invalid value for ‘mask_size_grd’
28983	Invalid value for ‘mask_size_smooth’
28984	Model broken
28985	Invalid value for ‘descriptor_type’
28986	Invalid value for ‘matcher’
28987	Too many point classes -0 model storing in a file is not possible
28988	Serialized item does not contain a valid descriptor model
29000	Function not implemented on this machine
29001	Image to process has wrong gray value type
29002	Wrong image component (see: get_system(obj_images,H))
29003	Undefined gray values
29004	Wrong image format for operation (too big or too small)
29005	Wrong number of image components for image output
29006	String is too long (max. 1024 characters)
29007	Wrong pixel type for this operation
29008	Operation not realized yet for this pixel type
29009	Image is no color image with three channels
29010	Image acquisition devices are not supported in the demo version
29011	Packages are not supported in the demo version
29020	Internal error: Unknown value
29021	Image domain too small.
29022	Input dimension too small
29023	Draw operator has been canceled
29050	Operator is not available in this restricted version of HALCON
29051	Packages are not available in this restricted version of HALCON
29052	The selected image acquisition interface is not available in this restricted version of HALCON
29100	Too many unknown variables in linear equation
29101	No (unique) solution for the linear equation
29102	Too little equations in linear equation
29200	Matrix is not invertible
29201	Singular value decomposition did not converge

29202	Matrix has too few rows for singular value partition
29203	Eigenvalue computation did not converge
29204	Eigenvalue computation did not converge
29205	Matrix is singular
29206	Function matching did not converge
29207	Input matrix undefined
29208	Input matrix with wrong dimension
29209	Input matrix is not quadratic
29210	Matrix operation failed
29211	Matrix is not positive definite
29212	One element of the matrix is zero: Division by zero
29213	Matrix is not an upper triangular matrix
29214	Matrix is not a lower triangular matrix
29215	One element of the matrix is negative
29216	Matrix file: Invalid character
29217	Matrix file: Matrix incomplete
29218	Invalid file format for matrix
29219	Resulting matrix has complex values
29220	Wrong value in matrix of exponents
29221	The version of the matrix is not supported
29222	Serialized item does not contain a valid matrix
29230	Internal error: wrong Node
29231	Inconsistent red black tree
29250	Internal error: Wrong LAPACK parameter
29260	Number of points too small for spherical triangulation
29261	First three points are collinear in spherical triangulation
29262	Spherical triangulation contains identical input points
29263	Internal error: array not allocated large enough for spherical triangulation
29264	Spherical Voronoi diagram contains degenerate triangle
29265	Internal error: inconsistent spherical triangulation
29266	Spherical Voronoi diagram contains self-intersecting polygon
29267	Internal error: inconsistent spherical polygon data
29268	Internal error: Ambiguous great circle arc intersection
29269	Internal error: Ambiguous great circle arc
29270	Internal error: Illegal parameter
29280	Not enough points for planar triangular meshing
29281	The first three points of the triangular meshing are collinear
29282	Planar triangular meshing contains identical input points
29283	Invalid points for planar triangular meshing
29284	Internal error: allocated array too small for planar triangular meshing
29285	Internal error: planar triangular meshing inconsistent
29300	Eye point and reference point coincide
29400	Timeout occurred

29401	Invalid value for timeout
29450	Invalid value for ‘sub_object_size’
29451	Invalid value for ‘min_size’
29452	Invalid number of least-squares iterations
29453	Invalid value for ‘angle_step’
29454	Invalid value for ‘scale_r_step’
29455	Invalid value for ‘scale_c_step’
29456	Invalid value for ‘max_angle_distortion’
29457	Invalid value for ‘max_aniso_scale_distortion’
29458	Invalid value for ‘min_size’
29459	Invalid value for ‘cov_pose_mode’
29460	Model contains no calibration information
29461	Generic parameter name does not exist
29462	Provided camera parameters have different resolution than image
29463	Invalid file format for deformable model
29464	The version of the deformable model is not supported
29465	Invalid ‘deformation_smoothness’
29466	Invalid ‘expand_border’
29467	Model origin outside of axis-aligned bounding rectangle of template region
29468	Serialized item does not contain a valid deformable model
29500	3D Object Model has no points
29501	3D Object Model has no faces
29502	3D Object Model has no normals
29506	Invalid file format for 3D surface model
29507	The version of the 3D surface model is not supported
29508	Serialized item does not contain a valid 3D surface model
29510	Invalid 3D file
29511	Invalid 3D object model
29512	Unknown file type
29513	The version of the 3D object model is not supported
29514	Required attribute missing in 3D object model
29515	Required points missing in 3D object model
29516	Required normals missing in 3D object model
29517	Required triangulation missing in 3D object model
29518	Required polylines missing in 3D object model
29519	Required triangle neighborhood missing in 3D object model
29520	Required polygons missing in 3D object model
29521	Required 2D mapping missing in 3D object model
29522	Required primitive missing in 3D object model
29523	Required 3D shape model missing in 3D object model
29524	Required extended attribute missing in 3D object model
29525	Serialized item does not contain a valid 3D object model
29526	Primitive in 3D object model has no extended data

29527	Operation invalid, 3D object model already contains triangles
29528	Operation invalid, 3D object model already contains lines
29529	Operation invalid, 3D object model already contains faces or polygons
29530	For at least one input 3D object model no neighbor with sufficient surface overlap is available.
29531	All components of points must be set at once
29532	All components of normals must be set at once
29533	Number of values doesn't correspond to number of already existing points
29534	Number of values doesn't correspond to number of already existing normals
29535	Number of values doesn't correspond to already existing triangulation
29536	Number of values doesn't correspond to length of already existing polygons
29537	Number of values doesn't correspond to length of already existing polylines
29538	Number of values doesn't correspond to already existing 2D mapping
29539	Number of values doesn't correspond to already existing extended attribute
29550	Triangles of the 3D object model are not suitable for this operator
29551	Too few suitable 3D points in the 3D object model
29580	Invalid file format for serialized items
29581	Serialized item: premature end of file
29600	Invalid value for 'image_resize_method'
29601	Invalid value for 'image_resize_value'
29602	Invalid value for 'rating_method'
29603	At least one type of image information must be added
29604	Sample identifier does not contain color information
29605	Sample identifier does not contain texture information
29606	Sample image does not contain enough information
29607	Sample identifier does not contain unprepared data (use add_sample_identifier_preparation_data)
29608	Sample identifier has not been prepared yet (use prepare_sample_identifier)
29609	Sample identifier does not contain untrained data (use add_sample_identifier_training_data)
29610	Sample identifier has not been trained yet (use train_sample_identifier)
29611	Sample identifier does not contain result data
29612	Sample identifier must contain at least two training objects (use add_sample_identifier_training_data)
30000	no error
30001	Input invalid
30002	Input negative
30003	Input exceeded range
30004	Memory exceeded boundary
30004	Memory allocation failure
30006	Memory pointer null
30007	DMA failure
30008	File open failure

30009	File read failure
30010	File write failure
30011	File close failure
30012	File format failure
30013	Warning low memory
40000	No error occurred in camera device.
40001	Initialization of image chip driver failed.
40002	Converting image to RGB or HSV failed.
40003	The capture process timed out.
40004	Arming video driver failed -> driver is in error state.
40005	Setting up image chip failed while changing size.
40006	Setting up video driver failed while changing size.
40007	Setting up image chip failed while changing brightness.
40008	Setting light mode failed -> typically UART communication.
40009	Setting focus pos. failed -> typically UART communication.
40010	Auto focus process failed -> typically UART communication.
50001	Indicates the configuration is invalid.
50002	Indicates the configuration API was not initialized.
50003	Indicates the configuration API was already initialized.
50004	Indicates that a function argument was invalid.
50005	Indicates a channel was defined twice.
50006	One has tried to define more than 2 quadrature channels.
50007	Indicates that more than 1 TRIGGER inputs is defined.
50008	Indicates that more than 1 READY signal is defined.
50009	Indicates that more than 1 FLASH output is defined.
50010	Indicates that more than 1 PROCESS output is defined.
50011	Indicates that more than 1 CAPTURE output is defined.
50012	Indicates that more than 1 PROJECT_SELECT feedback output defined.
50013	Indicates that more than 1 PROJECT_SELECT input is defined.
50014	Indicates invalid configuration of timer/quadrature.
50015	Indicates PRU couldn't started.
70010	Frame dropped because the queue was full
70011	Frame lost in the GigE interface
70020	Payload type not supported
70021	Pixel format not supported
70030	Receive timeout
70031	Too many GigE resend requests sent
70032	Failed to recover frame
70040	Frame partially received
70041	Frame not received
70050	File load error
70051	File format error
70052	Frame unavailable

80001	Invalid pointer
80002	Timeout
80003	Not initialized
80004	No cameras
80005	Bad index
80006	Bad category
80007	Bad feature
80008	Bad feature type
80009	Bad value
80010	Out of range
80011	Socket error
80012	Bad reply
80013	Access denied
80014	Exception
80015	Overload
80016	Unknown error
80017	No more features
80018	No more enums

12.2 Disabled HALCON Operators

For details see section “7.5 Module HALCON Script”

List of not supported (disabled) HALCON operators for Module HALCON script:

[0] "read_polygon_xld_arc_info"
[1] "write_polygon_xld_arc_info"
[2] "read_contour_xld_arc_info"
[3] "write_contour_xld_arc_info"
[4] "read_world_file"
[5] "read_variation_model"
[6] "write_variation_model"
[7] "write_tuple"
[8] "read_tuple"
[9] "write_serial"
[10] "open_serial"
[11] "get_socket_param"
[12] "set_socket_param"
[13] "get_next_socket_data_type"
[14] "get_socket_descriptor"
[15] "close_all_sockets"
[16] "close_socket"
[17] "socket_accept_connect"
[18] "open_socket_connect"
[19] "open_socket_accept"
[20] "read_sheet_of_light_model"
[21] "write_sheet_of_light_model"
[22] "create_sheet_of_light_calib_object"

[23] "fwrite_serialized_item"
[24] "fread_serialized_item"
[25] "optimize_aop"
[26] "write_aop_knowledge"
[27] "read_aop_knowledge"
[28] "read_ocv"
[29] "write_ocv"
[30] "read_ocr_class_knn"
[31] "write_ocr_class_knn"
[32] "trainf_ocr_class_knn"
[33] "select_feature_set_trainf_knn"
[34] "select_feature_set_trainf_mlp_protected"
[35] "select_feature_set_trainf_mlp"
[36] "select_feature_set_trainf_svm_protected"
[37] "select_feature_set_trainf_svm"
[38] "import_lexicon"
[39] "read_ocr_class_svm"
[40] "write_ocr_class_svm"
[41] "trainf_ocr_class_svm_protected"
[42] "trainf_ocr_class_svm"
[43] "get_prep_info_ocr_class_svm"
[44] "read_ocr_class_mlp"
[45] "write_ocr_class_mlp"
[46] "trainf_ocr_class_mlp_protected"
[47] "trainf_ocr_class_mlp"
[48] "get_prep_info_ocr_class_mlp"
[49] "write_ocr"
[50] "read_ocr"
[51] "trainf_ocr_class_box"
[52] "protect_ocr_trainf"
[53] "write_ocr_trainf"
[54] "concat_ocr_trainf"
[55] "write_ocr_trainf_image"
[56] "append_ocr_trainf"
[57] "read_ocr_trainf_names_protected"
[58] "read_ocr_trainf_names"
[59] "read_ocr_trainf_select"
[60] "read_ocr_trainf"
[61] "read_gray_se"
[62] "write_metrology_model"
[63] "read_metrology_model"
[64] "write_measure"
[65] "read_measure"
[66] "read_matrix"
[67] "write_matrix"
[68] "read_sample_identifier"
[69] "write_sample_identifier"
[70] "read_shape_model"
[71] "write_shape_model"
[72] "read_descriptor_model"
[73] "write_descriptor_model"

[74] "read_deformable_model"
[75] "write_deformable_model"
[76] "read_ncc_model"
[77] "write_ncc_model"
[78] "read_component_model"
[79] "write_component_model"
[80] "read_training_components"
[81] "write_training_components"
[82] "read_deformable_surface_model"
[83] "write_deformable_surface_model"
[84] "read_surface_model"
[85] "write_surface_model"
[86] "read_shape_model_3d"
[87] "write_shape_model_3d"
[88] "write_object_model_3d"
[89] "read_object_model_3d"
[90] "read_kalman"
[91] "update_kalman"
[92] "create_rectification_grid"
[93] "get_window_background_image"
[94] "set_drawing_object_callback"
[95] "detach_background_from_window"
[96] "attach_background_to_window"
[97] "detach_drawing_object_from_window"
[98] "attach_drawing_object_to_window"
[99] "update_window_pose"
[100] "get_os_window_handle"
[101] "set_window_dc"
[102] "new_extern_window"
[103] "set_window_type"
[104] "set_window_extents"
[105] "get_window_attr"
[106] "set_window_attr"
[107] "query_window_type"
[108] "open_window"
[109] "open_textwindow"
[110] "get_window_type"
[111] "get_window_pointer3"
[112] "get_window_extents"
[113] "dump_window_image"
[114] "dump_window"
[115] "close_window"
[116] "clear_window"
[117] "write_string"
[118] "get_display_scene_3d_info"
[119] "set_scene_3d_to_world_pose"
[120] "set_scene_3d_param"
[121] "set_scene_3d_light_param"
[122] "set_scene_3d_instance_pose"
[123] "set_scene_3d_instance_param"
[124] "set_scene_3d_camera_pose"

[125] "render_scene_3d"
[126] "remove_scene_3d_light"
[127] "remove_scene_3d_instance"
[128] "remove_scene_3d_camera"
[129] "display_scene_3d"
[130] "add_scene_3d_light"
[131] "add_scene_3d_instance"
[132] "add_scene_3d_camera"
[133] "clear_scene_3d"
[134] "create_scene_3d"
[135] "get_window_param"
[136] "set_window_param"
[137] "set_draw"
[138] "get_draw"
[139] "write_lut"
[140] "set_lut"
[141] "gnuplot_plot_funct_1d"
[142] "gnuplot_plot_ctrl"
[143] "gnuplot_plot_image"
[144] "gnuplot_close"
[145] "gnuplot_open_file"
[146] "create_drawing_object_text"
[147] "get_drawing_object_iconic"
[148] "clear_drawing_object"
[149] "set_drawing_object_params"
[150] "get_drawing_object_params"
[151] "set_drawing_object_xld"
[152] "create_drawing_object_xld"
[153] "create_drawing_object_circle_sector"
[154] "create_drawing_object_ellipse_sector"
[155] "create_drawing_object_line"
[156] "create_drawing_object_circle"
[157] "create_drawing_object_ellipse"
[158] "create_drawing_object_rectangle2"
[159] "create_drawing_object_rectangle1"
[160] "draw_nurbs_interp_mod"
[161] "draw_nurbs_interp"
[162] "draw_nurbs_mod"
[163] "draw_nurbs"
[164] "draw_xld_mod"
[165] "draw_xld"
[166] "draw_rectangle2_mod"
[167] "draw_rectangle2"
[168] "draw_rectangle1_mod"
[169] "draw_rectangle1"
[170] "draw_point_mod"
[171] "draw_point"
[172] "draw_line_mod"
[173] "draw_line"
[174] "draw_ellipse_mod"
[175] "draw_ellipse"

[176] "draw_circle_mod"
[177] "draw_circle"
[178] "draw_region"
[179] "draw_polygon"
[180] "read_distance_transform_xld"
[181] "write_distance_transform_xld"
[182] "read_funct_1d"
[183] "write_funct_1d"
[184] "convol_image"
[185] "write_template"
[186] "read_template"
[187] "gen_filter_mask"
[188] "read_fft_optimization_data"
[189] "write_fft_optimization_data"
[190] "read_polygon_xld_dxf"
[191] "write_polygon_xld_dxf"
[192] "copy_file"
[193] "set_current_dir"
[194] "get_current_dir"
[195] "remove_dir"
[196] "make_dir"
[197] "list_files"
[198] "delete_file"
[199] "file_exists"
[200] "read_object"
[201] "write_object"
[202] "write_region"
[203] "write_image"
[204] "read_sequence"
[205] "read_region"
[206] "read_image"
[207] "open_file"
[208] "fwrite_string"
[209] "fread_line"
[210] "fread_string"
[211] "fread_char"
[212] "fnew_line"
[213] "close_file"
[214] "read_data_code_2d_model"
[215] "write_data_code_2d_model"
[216] "read_class_train_data"
[217] "write_class_train_data"
[218] "read_class_knn"
[219] "write_class_knn"
[220] "read_class_gmm"
[221] "write_class_gmm"
[222] "read_samples_class_gmm"
[223] "write_samples_class_gmm"
[224] "read_class_svm"
[225] "write_class_svm"
[226] "read_samples_class_svm"

[227] "write_samples_class_svm"
[228] "read_class_mlp"
[229] "write_class_mlp"
[230] "read_samples_class_mlp"
[231] "write_samples_class_mlp"
[232] "write_class_box"
[233] "read_sampset"
[234] "read_class_box"
[235] "learn_sampset_box"
[236] "gen_caltab"
[237] "create_caltab"
[238] "caltab_points"
[239] "read_pose"
[240] "write_pose"
[241] "read_cam_par"
[242] "write_cam_par"
[243] "sim_caltab"
[244] "disp_caltab"
[245] "find_marks_and_pose"
[246] "find_caltab"
[247] "write_camera_setup_model"
[248] "read_camera_setup_model"
[249] "read_calib_data"
[250] "write_calib_data"
[251] "read_bar_code_model"
[252] "write_bar_code_model"
[253] "control_io_channel"
[254] "write_io_channel"
[255] "read_io_channel"
[256] "set_io_channel_param"
[257] "get_io_channel_param"
[258] "close_io_channel"
[259] "open_io_channel"
[260] "query_io_device"
[261] "control_io_device"
[262] "set_io_device_param"
[263] "get_io_device_param"
[264] "close_io_device"
[265] "open_io_device"
[266] "control_io_interface"
[267] "query_io_interface"
[268] "get_framegrabber_param"
[269] "set_framegrabber_param"
[270] "get_framegrabber_callback"
[271] "set_framegrabber_callback"
[272] "info_framegrabber"
[273] "close_all_framegrabbers"
[274] "close_framegrabber"
[275] "open_framegrabber"
[276] "get_framegrabber_lut"
[277] "set_framegrabber_lut"

[278] "add_scene_3d_label"
[279] "remove_scene_3d_label"
[280] "set_scene_3d_label_param"
[281] "convert_coordinates_image_to_window"
[282] "convert_coordinates_window_to_image"
[283] "read_ocr_class_cnn"
[284] "read_texture_inspection_model"
[285] "write_texture_inspection_model"
[286] "read_dl_classifier"
[287] "read_structured_light_model"
[288] "write_dl_classifier"
[289] "write_structured_light_model"
[290] "create_dl_model_detection"
[291] "read_dl_model"
[292] "read_message"
[293] "write_dl_model"
[294] "write_message"
[295] "fread_bytes"
[296] "fwrite_bytes"
[297] "read_image_metadata"
[298] "write_image_metadata"
[299] "read_deep_ocr"
[300] "write_deep_ocr"
[301] "read_encrypted_item"
[302] "read_memory_block"
[303] "write_encrypted_item"
[304] "write_memory_block"
[305] "system_call"