SENSOPART

VISOR®

User Manual

Software version 2.8

V 04

Copyright

We accept no responsibility for printing errors or mistakes which occurred in drafting these document. Subject to delivery and technical alterations.

- Translation of Original operating instructions -

First publication 01 / 2019

SensoPart Industriesensorik GmbH
Nägelseestr. 16
79288 Gottenheim
Germany

c UL us
LISTED
IND. CONT. EQ.
77HA
For use in class 2 circuits
For use in NFPA 79 Applications only

CE

Open Source Licenses

The VISOR® software makes use of third party software packages that come under various licenses. This section is meant to list all these packages and to give credit to those whose code helped in the creation of the VISOR® software.

For components that reference the GNU General Public License (GPL) or the GNU Lesser General Public License (LGPL), please find these licenses in this software installation in \SensoPart\VISOR Vision Sensor\Eula\OpenSourceLicenses.

For at least three years from the date of distribution of the applicable product or software, we will give to anyone who contacts us using the contact information provided below, for a charge of no more than our cost of physically distributing, a complete machine-readable copy of the corresponding source code for the version that we may have distributed to you.

The packages containing the source code and the licenses for all of the open-source software are available upon request.
Contact information for requesting source code:
Email: Open.Source@sensopart.de

VISOR® Firmware version newer than V2.0

This software uses Linux Version 4.14 (Website: www.kernel.org), which is distributed under the GNU GPL version 2

This software is based on pugixml library (http://pugixml.org). pugixml is Copyright © 2006-2018 Arseny Kapoulkine.

This software makes use of Nlohmann JSON library, which is distributed under the MIT License. Copyright © 2013-2018 Niels Lohmann.

This software makes use of libcrypto, which is distributed under the openssl license. Copyright OpenSSL © 2018.

This software makes use of libcurl, which is distributed under the MIT license. Copyright (c) 1996 - 2018, Daniel Stenberg, daniel@haxx.se, and many contributors.

This software makes use of busybox, which is distributed under the GNU GPL version 2.

This software makes use of e2fsprogs which is distributed under the GNU GPL version 2 License, GNU LGPL version 2.1 License, BSD License and MIT License.

This software makes use of eudev, which is distributed under the GNU GPL version 2 License and GNU LGPL version 2.1 License or later.

This software makes use of glibc, which is distributed under the GNU LGPL version 2.1 License or later.

This software makes use of GNU MP library, which is distributed under the GNU GPL version 2 License.

This software makes use of libcap, which is distributed under the GNU LGPL version 2.1 License or later.

This software makes use of libidn, which is distributed under the GNU LGPL version 2.1 License or later.

This software is based in part on the work of the Independent JPEG Group. This software makes use of libjpeg-turbo, which is distributed under the Modified (3-clause) BSD License.

This software makes use of libnl, which is distributed under the GNU LGPL version 2.1 License or later.

This software makes use of libssh2, which is distributed under the BSD license.

This software makes use of libssh2, which is distributed under the openssl license.

This software makes use of libstdc++, which is distributed under the GPL-3.0-with-GCC-exception license.

This software makes use of libxml2, which is distributed under the MIT license.

This software makes use of lighttpd, which is distributed under the BSD 3-clause license. Copyright (c) 2004, Jan Kneschke, incremental.

This software makes use of mtd-utils, which is distributed under the GNU GPL version 2 License.

This software makes use of net-snmp-libs, which is distributed under the BSD-License License. Copyright (c) 2001-2003, Networks Associates Technology, Inc

This software makes use of netbase, which is distributed under the GNU GPL version 2 License.

This software makes use of OpenSSH, which is distributed under the OpenSSH License.

This software makes use of OpenSSL, which is distributed under the OpenSSL License.

This software makes use of GnuPG Version 1.4.10 (Website: https://www.gnupg.org/),which is distributed under the GNU GPL version 3 or higher.

This software makes use of netbase, which is distributed under the GNU GPL version 2 License U-Boot exception 2.0 license.

VISOR® Firmware version starting with V2.8.0

This software makes use of xxHash Library, whixh is distributed under the BSD 2-Clause License, Copyright (c) 2012-2021 Yann Collet. All rights reserved.

BSD 2-Clause License (https://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

This software makes use of asio, Copyright (c) 2003-2022 Christopher M. Kohlhoff (chris at kohlhoff dot com), which is distributed under the Boost Software License, Version 1.0.

This software makes use of WebSocket++ Main Library, which is distributed under the BSD license. Copyright (c) 2014, Peter Thorson. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the WebSocket++ Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Base 64 Library (base64/base64.hpp)

base64.hpp is a repackaging of the base64.cpp and base64.h files into a single header suitable for use as a header only library. This conversion was done by Peter Thorson (webmaster@zaphoyd.com) in 2012.

base64.cpp and base64.h: Copyright (C) 2004-2008 René Nyffenegger

SHA1 Library (sha1/sha1.hpp)

sha1.hpp is a repackaging of the sha1.cpp and sha1.h files from the shallsha1 library (http://code.google.com/p/smallsha1/) into a single header suitable for use as a header only library. This conversion was done by Peter Thorson (webmaster@zaphoyd.com) in 2013. All modifications to the code are redistributed under the same license as the original, which is listed below.

Copyright (c) 2011, Micael Hildenborg. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Micael Hildenborg nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

MD5 Library (common/md5.hpp)

md5.hpp is a reformulation of the md5.h and md5.c code from http://www.opensource.apple.com/source/cups/cups-59/cups/md5.c to allow it to function as a component of a header only library. This conversion was done by Peter

Thorson (webmaster@zaphoyd.com) in 2012 for the WebSocket++ project. The changes are released under the same license as the original (listed below)

UTF8 Validation logic (utf8_validation.hpp)

utf8_validation.hpp is adapted from code originally written by Bjoern Hoehrmann <bjoern@hoehrmann.de>. See http://bjoern.hoehrmann.de/utf-8/decoder/dfa/ for details.

The original license:

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of websockets nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

tinyaes-py: Copyright (c) 2020 Matteo Bertini <naufraghi@develer.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Copyright (c) 2005-2022, NumPy Developers. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the NumPy Developers nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Copyright (c) 2001-2002 Enthought, Inc. 2003-2022, SciPy Developers. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derivedfrom this software without specific prior written permission.

The Python Imaging Library (PIL) is

Copyright © 1997-2011 by Secret Labs AB

Copyright © 1995-2011 by Fredrik Lundh

Pillow is the friendly PIL fork. It is

Copyright © 2010-2022 by Alex Clark and contributors

Like PIL, Pillow is licensed under the open source HPND License:

By obtaining, using, and/or copying this software and/or its associated documentation, you agree that you have read, understood, and will comply with the following terms and conditions:

Permission to use, copy, modify, and distribute this software and its associated documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies, and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Secret Labs AB or the author not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

Copyright (c) 2010-2022, PyInstaller Development Team

Copyright (c) 2005-2009, Giovanni Bajo

Based on previous work under copyright (c) 2002 McMillan Enterprises, Inc.

PyInstaller is licensed under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

Bootloader Exception

In addition to the permissions in the GNU General Public License, the authors give you unlimited permission to link or embed compiled bootloader and related files into combinations with other programs, and to distribute those combinations without any restriction coming from the use of those files. (The General Public License restrictions do apply in other respects; for example, they cover modification of the files, and distribution when not linked into a combined executable.)

Tensorflow:

All contributions by the University of California:

Copyright (c) 2014, The Regents of the University of California (Regents)

All rights reserved.

All other contributions:

Copyright (c) 2014, the respective contributors

All rights reserved.

Redistribution and use in source and binary forms, with or without

modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Copyright (c) 2012-2014 Pierre Raybaut

Copyright (c) 2014-2017 The Winpython development team https://github.com/winpython/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

react-loader-spinner:

Copyright (c) 2018 Mohan Pd.

@emotion/styled:

@emotion/react:

@material/top-app-bar

@mui/icons-material

@mui/lab

@mui/material

@mui/styles

axios

c3

i18next

i18next-browser-languagedetector

i18next-xhr-backend

react-i18next

react

react-dom

react-app-polyfill

react-device-detect

VISOR® Firmware version older than V2.0

The VISOR® firmware makes use of Linux Version 2.6.33 (Website: www.kernel.org), which is distributed under the GNU GPL version 2.

The VISOR® firmware makes use of x-loader, an initial program loader for Embedded boards based on OMAP processors (Website: http://arago-project.org/git/projects/?p=x-load-omap3.git; a=summary) which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of u-boot, an initial program loader for Embedded boards based on OMAP processors (Website: http://arago-project.org/git/projects/?p=x-load-omap3.git; a=summary) which is distributed under the GNU GPL version 2 or higher

The VISOR® firmware makes use of spike Version 0.2,a SPI-driver (Website: https://github.com/scottellis/spike/blob/master/spike.c), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of Busy-Box Version 1.18.1 (Website: http://www.busybox.net/), which is distributed under the GNU GPL version 2 or higher

The VISOR® firmware makes use of vsftpd Version 2.0.3 (Website: https://security.appspot.com/vsftpd.html), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of mtd-utils Version 1.5.0 (Website: http://www.linux-mtd.infradead.org/doc/general.html), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of Boa Webserver Version 0.94.13 (Website: http://www.boa.org/), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of Procps Version 3.2.8 (Website http://procps.sourceforge.net/download.html), which is distributed under the GNU GPL version 2 or higher and GNU LGPL version 2.1 or higher.

The VISOR® firmware makes use of GnuPG Version 1.4.10 (Website: https://www.gnupg.org/), which is distributed under the GNU GPL version 3 or higher.

The VISOR® firmware makes use of glibc, which is distributed under GNU LGPL version 2.1 or higher.

The VISOR® firmware makes use of Dropbear - a SSH2 server Version 2012.55 (Website: https://matt.ucc.asn.au/dropbear/dropbear.html). The Dropbear SSH2 server is distributed under the terms of the Dropbear License, which is a MIT/X Consortium style open source license. Please find this license in this software installation in \SensoPart\VISOR Vision-Sensor\Eula\OpenSourceLicenses

VISOR® PC Software

SensoConfig software is based in part on the work of the Qwt project (http://qwt.sf.net).

SensoFind, SensoConfig, SensoRescue software is based in part on the work of the Qt-project (https://doc.qt.io/qt-5/licenses-used-in-qt.html).

SensoCalc uses icons from material-design-icons (https://material.io/tools/icons), which are distributed under the Apache License Version 2.0.

SensoCalc uses NodeJS, which is distributed under the MIT license. Copyright (C) 2012-2018 by various contributors.

SensoCalc uses electron and electron-store, which is distributed under the MIT license. Copyright (c) 2013-2019 GitHub Inc.

SensoCalc uses AngularJS, which is distributed under the MIT license. Copyright (c) 2010-2018 Google, Inc. http://angularjs.org.

SensoCalc uses fs-jetpack, which is distributed under the MIT license.

SensoCalc uses asar, which is distributed under the MIT license. Copyright (c) 2014 GitHub Inc.

SensoCalc uses clipboard.js, which is distributed under the MIT license. Copyright © 2019 Zeno Rocha <hi@zenorocha.com>

SensoCalc uses roboto font, which is distributed under the MIT license.

SensoCalc uses gulp, which is distributed under the MIT license. Copyright (c) 2013-2018 Blaine Bublitz <blaine.bublitz@gmail.com>, Eric Schoffstall <yo@contra.io> and other contributors.

SensoCalc uses rcedit, which is distributed under the MIT license.

SensoCalc uses q, which is distributed under the MIT license. Copyright 2009–2017 Kristopher Michael Kowal and contributors.

SensoFind uses libgpg-error, which is distributed under the GNU LGPL version 2.1 License or later. Copyright (C) 2012-2017 g10 Code GmbH.

SensoFind uses Libgcrypt, which is distributed under the GNU LGPL version 2.1 License or later. Copyright (C) 1989,1991-2017 Free Software Foundation, Inc. Copyright (C) 2012-2017 g10 Code GmbH. Copyright (C) 2013-2017 Jussi Kivilinna.

SensoConfig uses Mesa library, which is licensed according to the terms of the MIT license. Copyright (C) 1999-2007 Brian Paul. All Rights Reserved.

Table of contents

1 Information on this document

1.1 Explanation of symbols

Warnings

CAUTION / WARNING / DANGER
This symbol is used to indicate a potentially hazardous situation that, if not avoided, could result in death or serious injury.

WARNING
This symbol is used to indicate potentially hazardous situations arising from laser beams.

ATTENTION:
This symbol is used to indicate text that must be observed without fail. Failure to do so may result in bodily injury or property damage.

NOTE:
This symbol is used to highlight useful tips and recommendations, as well as information intended to help ensure efficient operation.

Detectors

Pattern matching
Contour
Contrast
Brightness
Gray
Caliper
BLOB
Contour 3D

Barcode
Datacode
OCR
Color Value
Color List
Color Area
Result processing
Target Mark 3D

Alignment

Pattern matching
Edge detector
Contour matching

1.2 Additional documents

The following documents for the VISOR® vision sensor are available for download in the Download area of the SensoPart website.

- VISOR® User Manual
- VISOR® Communications manual
- VISOR® Operating manual

Furthermore, these documents are part of the software installation and can be found in the subfolder "...\Documentation\", as well as via the Windows Start menu.

1.3 Document version

This manual describes the VISOR® software version 2.8.

2 Safety instructions

WARNING!
The vision sensor is not a safety component pursuant to the EU Machinery Directive. It is strictly prohibited to use it in applications in which the safety of people depends on device functions.

Comply with all applicable local accident prevention regulations and general safety regulations.

Follow all safety instructions and other instructions in the operating manual and in this user manual.

The connection should be made exclusively by trained qualified personnel.

Do not tamper with or make alterations to the unit!

For use with any listed, configured cable cables (CYJV).

WARNING
Vision sensors with a laser belong to laser class 1 as defined in IEC 60825-1:2014.
Wavelength: 655 nm, frequency: 9 kHz, pulse width: 2.6 µs, pulse limit: 11 mW.

For devices with internal UV illumination:

WARNING! UV RADIATION - RISK GROUP 1
Vision sensors with internal UV illumination comply with risk group 3 according to DIN EN 62471:2009-03.
Possibly hazardous optical radiation.
Do not look into the light source during operation. May cause eye damage.

3 Intended use

The VISOR® vision sensor is an optical sensor and is used for the non-contact acquisition /identification of objects. The vision sensor features a number of different evaluation methods (detectors), with the specific methods depending on the specific model sensor. The product is designed for industrial use only. In residential areas, it is possible that additional measures for noise suppression must be taken. The vision sensor is not suitable for use outdoors.

WARNING
The vision sensor is not a safety component pursuant to the EU Machinery Directive It is strictly prohibited to use it in applications in which the safety of people depends on device functions.

3.1 Field of application

The VISOR® vision sensor is a cost-effective alternative to conventional image processing systems.

VISOR® Object:

The VISOR® vision sensor detects defective parts and parts in the wrong position, angular position, order, or any combination of them with unrivalled precision and accuracy. For testing tasks and evaluations, different detectors are available: e.g. Pattern matching, Contour, Brightness, Gray, Contrast, Caliper or BLOB. The Advanced version of the VISOR® vision sensor features Alignment on top of this. In this way, it is also possible to reliably detect the features which do not appear repetitively in the taught-in position. All evaluations are made relative to the current part position and angular position, without you having to define your own characteristic for each possible position.
The Advanced version also offers calibration for correcting distortion, e.g. for simple measurement tasks.

VISOR® Object AI:

All VISOR® Object AI functions are available in the VISOR® Object, as are added the Detector Classification (AI).

VISOR® Object Color:

The VISOR® Object Color offers powerful object detection combined with color detection. This makes it possible to increase the stability of many applications in which there are too few differences in the gray image. In addition, e.g. self-luminous parts such as colored LEDs and "non-colors" such as white and black are detected.

VISOR® Code reader:

The identification of products, components, or packaging on the basis of printed or directly marked – nailed or lasered – codes or plain text is common today in many areas of industry. The Code reader from SensoPart recognizes at a glance which part it has in front of it: It effortlessly reads barcodes of many types as well as printed and directly marked ECC-200 standard Data matrix codes – and this from any carrier material (metal, plastic, paper, glass). The sensor can even routinely decipher applied codes on oblique, distorted, or convex, reflective, or transparent surfaces. The code reader evaluates the quality of printed and directly marked Data Matrix codes based on standardized ISO and AIM quality parameters. This makes it possible to take corrective measures early on, preventing scrap resulting from illegible codes. In addition, the sensor can also read directly printed fonts with the detector plain text reading.

VISOR® Solar:

The VISOR® Solar features an optimized inspection algorithm that can be used for the comprehensive quality control of sensitive silicon wafers while they are being produced. The functions relevant to wafer and cell inspection – from wafer geometry detection, location, and position to defect localization, processing speed adjustment, and test accuracy – are pre-configured so that the sensor is ready to go after a few mouse clicks.

VISOR® Allround:

The VISOR® Allround features all the functions of the VISOR® Object, Code reader and Object Color combined in a single device. The Professional version also offers the "Multishot" function to detect minute surface defects.

VISOR® Robotic:

All VISOR® Robotic functions are available in the VISOR® Object, as are added robot-specific functions.

3.2 Requirements for use

In order to configure the VISOR® vision sensor, a standard PC / notebook (at least 1 GHz processor with support for SSE2 and 1 GB RAM, with Microsoft Windows 8, Windows 8.1, Windows 10 or Windows 11 operating system) with a network connection with RJ-45 port and a network with TCP-IP protocol is required. We recommend a screen resolution of at least 1024 x 768 pixels.

The default network settings for the VISOR® vision sensor are 192.168.100.100 for the IP address, 255.255.255.0 for the subnet mask, and 192.168.100.1 for the gateway.

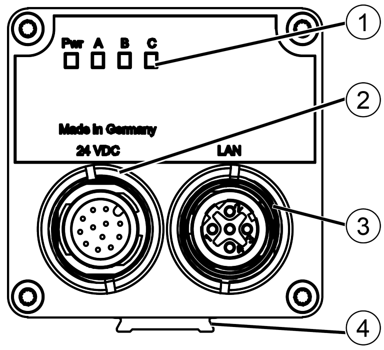
The VISOR® vision sensor does not need a PC or PLC to run. A PC / laptop is required only in order to configure the VISOR® vision sensor.

Attention must be paid to sufficient and constant object illumination to ensure reproducible results and avoid malfunction. Light reflections or changing extraneous light can distort evaluation results. If necessary, use an external light source and / or light-protection devices to protect against extraneous light / ambient light.

4 Product identification and characteristics

4.1 Product description

Overview



① LED display

② M12 connector for supply voltage (24 VDC) and digital I/O.

③ M12 connector socket for Ethernet connection.

④ Dovetail guide

LED display

All LEDs are driven without taking into account any delays used.

Fig. 1: *LED display*

Name	Color	Meaning	
Pwr. (Power)	Green	Operating voltage	No errors
	Red / Yellow		No PROFINET connection
	Yellow		No jobset available
	Red		Error / Starting device
A	Yellow	Result 1	
B	Yellow	Result 2	
C	Yellow	Result 3	

4.2 Type key

V	**20**	**-**	**ALL**	**-**	**P**	**3**	**-**	**W**	**-**	**M**	**D**	**-**	**M**	**2**	**-**	**L**
①	②		③		④	⑤		⑥		⑦	⑧		⑨	⑩		⑪

① **V = VISOR®**

② **Hardware / resolution**

- V10 / V10C: SVGA (800 x 600), QSVGA (400 x 300), QSVGA Zoom 2 (400 x 300)
- V20 / V20C: HDV2 (1440 x 1080), WGA (720 x 540), WGA Zoom 2 (720 x 540)
- V50 / V50C: QSXGA (2560 x 1936), SXVGA (1280 x 968), SXVGA Zoom 2 (1280 x 968)

③ **Sensor type**
ALL = Allround
OB = Object
OAI = Object AI
CR = Code reader
RO = Robotic
SO = Solar

④ **Variant**
S = Standard
A = Advanced
P = Professional

⑤ **Version**

⑥ **Lighting**
W = White
R = Red
I = Infrared

⑦ **Field of view lens**
C = C-Mount
W = Wide
M = Medium
N = Narrow

⑧ **Depth of field**
" " = Depth of field: Normal
D = Depth of field: Enhanced

⑨ **Focal point (optional)**
M = Motorized focal point

⑩ **Connections (optional)**
2 = Two connections (1 x I/O, 1 x Ethernet)

⑪ **Laser**

4.3 Scope of delivery and software

4.3.1 VISOR® vision sensor scope of delivery

The scope of delivery includes:

- VISOR® vision sensor
- Mounting bracket MK 45
- Allen key
- Operating manual (068-14799)

Upon receiving the delivery, check it immediately for any transit damage and make sure that it is complete. If there is any transit damage, inform the shipping agent. If returning the sensor, always make sure to pack it in sufficiently sturdy packaging.

4.3.2 VISOR® software

The VISOR® software consists of the three modules "SensoFind", "SensoConfig" and "SensoView". Additional information: VISOR® software – Overview and Quick Start Guide.

Download

The VISOR® software Setup is available for download at www.sensopart.com (Download / Software...).

USB drive

The VISOR® software setup is also available from SensoPart Industriesensorik GmbH on a USB drive with part number 651-01000 (subject to a charge).

4.4 Overview of functions

4.4.1 VISOR® Object

	VISOR® Object	
	Standard	**Advanced**
Applications	Presence, completeness, measuring, color, position control	
Resolution		
V10 (800 x 600): Mono | Color	✓ |✓	
Frames per second: Mono | Color	75 | 50	
V20 (1440 x 1080): Mono | Color	– | –	✓ |✓
Frames per second: Mono | Color	– | –	40 | 20
V50 (2560 x 1936): Mono | Color	– | –	✓ |✓
Frames per second: Mono | Color	– | –	22 | 8
Lighting	White | only for Mono: Red, Infrared	
Multishot (Mono)	–	–
Target laser	–	✓
Lenses		
V10 Wide | Medium | Narrow | C-Mount	✓ |✓ |✓ |–	✓ |✓ |✓ |✓
V20 Wide | Medium | Narrow | C-Mount	– | – | – | –	✓ |✓ |✓ |✓
V50 Wide | Medium | Narrow | C-Mount	– | – | – | –	✓ |✓ |– |✓
Interfaces	Ethernet | EtherNet/IP | PROFINET	
Inputs | Outputs | Selectable	2 | 2 | 4	2 | 2 | 6
Encoder input	–	✓
Ethernet | EtherNet/IP	✓ |✓	✓ |✓
PROFINET | SensoWeb	✓ |✓	✓ |✓
Service port	–	✓
Job | Detectors		
Number of jobs (max.)	32	255
Number of detectors per job (max.)	32	255

	VISOR$^{®}$ Object	
	Standard	Advanced
Calibration		
Scaling (Measurement)	✓	✓
Calibration plate (Measurement)	–	✓
Point pair list (Robotics)	–	–
Calibration plate (Robotics)	–	–
Hand-Eye calibration (Robotics)	–	–
Base-Eye calibration (Robotics)	–	–
Pre-processing		
Pre-processing filter	–	✓
Repeat mode	–	✓
Shutter variation	–	✓
Free-form search region	✓	✓
Alignment		
Contour matching (translation, rotation 360°)	✓	✓
Pattern matching (translation, rotation 360°)	–	✓
Edge detector (translation, rotation)	–	✓
Object detection		
Contour (translation, rotation 360°)	✓	✓
Multiple objects	–	✓
Contour 3D (translation, rotation 360°)	–	–
Multiple objects	–	–
Pattern matching (translation, rotation 360°)	✓	✓
Multiple objects	–	✓
Gray	✓	✓
Contrast	✓	✓
Brightness	✓	✓
Caliper	✓	✓
BLOB	–	✓
Identification		
Barcode | Barcode Advanced	– | –	– | –
Datacode | Datacode Advanced	– | –	– | –
OCR	–	–
Robotic functions		
Result offset Image | 2D | 3D	– | – | –	– | – | –
Gripping space check	–	–
Color detectors for Color variants		
Color Area	✓	✓

	VISOR® Object	
	Standard	Advanced
Color Value	–	✓
Color List	–	✓
Color distance | Binarization	–	✓
Result processing		
Result processing: Text	–	–
Result processing: Math	–	✓
Result processing: Robotics	–	–

4.4.2 VISOR® Code reader

	VISOR® Code reader		
	Standard	Advanced	Professional
Applications	Reading barcodes, data codes, text		
Resolution			
V10 (800 x 600): Mono | Color	✓ |–		
Frames per second: Mono | Color	75 | –		
V20 (1440 x 1080): Mono | Color	✓ |–		
Frames per second: Mono | Color	40 | –		
V50 (2560 x 1936): Mono | Color	– | –	✓ |–	
Frames per second: Mono | Color	– | –	22 | –	
Lighting	White | only for Mono: Red, Infrared		
Multishot (Mono)	–	–	–
Target laser	✓ (V20 only)	✓	✓
Lenses			
V10 Wide | Medium | Narrow | C-Mount	✓ |✓ |✓ |–	✓ |✓ |✓ |✓	
V20 Wide | Medium | Narrow | C-Mount	✓ |✓ |✓ |–	✓ |✓ |✓ |✓	
V50 Wide | Medium | Narrow | C-Mount	– | – | – | –	✓ |✓ |– |✓	
Interfaces	Ethernet | EtherNet/IP | PROFINET		
Inputs | Outputs | Selectable	2 | 2 | 4	2 | 2 | 6	
Encoder input	–	✓	✓
Ethernet | EtherNet/IP	✓ |✓	✓ |✓	✓ |✓
PROFINET | SensoWeb	✓ |✓	✓ |✓	✓ |✓
Service port	–	✓	✓
Job | Detectors			
Number of jobs (max.)	8	255	
Number of detectors per job (max.)	2	255	
Pre-processing			
Pre-processing filter	–	✓	✓
Repeat mode	–	✓	✓
Shutter variation	–	✓	✓
Free-form search region	–	–	✓
Alignment			
Contour matching (translation, rotation 360°)	–	–	✓
Pattern matching (translation, rotation 360°)	–	–	✓
Edge detector (translation, rotation)	–	–	✓
Object detection			
Contour (translation, rotation 360°)	–	–	–

	VISOR® Code reader		
	Standard	Advanced	Professional
Multiple objects	–	–	–
Contour 3D	–	–	–
Multiple objects	–	–	–
Target Mark 3D	–	–	–
Pattern matching (translation, rotation 360°)	–	–	✓
Multiple objects	–	–	–
Gray	–	–	✓
Contrast	–	–	✓
Brightness	–	–	✓
Caliper	–	–	–
BLOB	–	–	–
Identification			
Barcode | Barcode Advanced	✓ |–	✓ |✓	✓ |✓
Datacode | Datacode Advanced	✓ |–	✓ |✓	✓ |✓
OCR	–	–	✓
Robotic functions			
Result offset Image | 2D | 3D	–|–|–	–|–|–	–|–|–
Gripping space check	–	–	–
Result processing			
Result processing: Text	–	✓	✓
Result processing: Math	–	–	–
Result processing: Robotics	–	–	–

4.4.3 VISOR® Robotic

	VISOR® Robotic	
	Advanced	Professional
Applications	Robotics, localization, presence, completeness, measurement, position control	
Resolution		
V10 (800 x 600): Mono | Color	✓ |–	–
Frames per second: Mono | Color	75	–
V20 (1440 x 1080): Mono | Color	✓ |✓	
Frames per second: Mono | Color	40 | 20	
V50 (2560 x 1936): Mono | Color	–	✓ |✓
Frames per second: Mono | Color	–	22 | 8
Lighting	White | only for Mono: Red, Infrared	
Multishot (Mono)	–	–
Target laser	✓	✓
Lenses		
V10 Wide | Medium | Narrow | C-Mount	✓ |✓ |✓ |✓	–|–|–|–
V20 Wide | Medium | Narrow | C-Mount	✓ |✓ |✓ |✓	
V50 Wide | Medium | Narrow | C-Mount	–|–|–|–	✓ |✓ |–|✓
Interfaces	Ethernet | EtherNet/IP | PROFINET	
Inputs | Outputs | Selectable	2 | 2 | 6	
Encoder input	✓	✓
Ethernet | EtherNet/IP	✓ |✓	✓ |✓
PROFINET | SensoWeb	✓ |✓	✓ |✓
Service port	✓	✓
Job | Detectors		
Number of jobs (max.)	255	
Number of detectors per job (max.)	255	
Calibration		
Scaling (Measurement)	✓	✓
Calibration plate (Measurement)	✓	✓
Point pair list (Robotics)	✓	✓
Calibration plate (Robotics)	✓	✓
Hand-Eye calibration (Robotics)	–	✓
Base-Eye calibration (Robotics)	–	✓
Pre-processing		
Pre-processing filter	✓	✓
Repeat mode	✓	✓
Shutter variation	✓	✓
Free-form search region	✓	✓

	VISOR® Robotic	
	Advanced	**Professional**
Alignment		
Contour matching (translation, rotation 360°)	✓	✓
Pattern matching (translation, rotation 360°)	✓	✓
Edge detector (translation, rotation)	✓	✓
Object detection		
Contour (translation, rotation 360°)	✓	✓
Multiple objects	✓	✓
Contour 3D	–	✓
Multiple objects	–	✓
Target Mark 3D	✓	✓
Pattern matching (translation, rotation 360°)	✓	✓
Multiple objects	✓	✓
Gray	✓	✓
Contrast	✓	✓
Brightness	✓	✓
Caliper	✓	✓
BLOB	✓	✓
Identification		
Barcode | Barcode Advanced	– | –	✓ | ✓
Datacode | Datacode Advanced	– | –	✓ | ✓
OCR	–	✓
Robotic functions		
Result offset Image | 2D | 3D	✓ | ✓ | ✓	
Gripping space check	✓	✓
Color detectors for Color variants		
Color Value	✓	✓
Color Area	✓	✓
Color List	✓	✓
Color distance | Binarization	✓	✓
Result processing		
Result processing: Text	–	✓
Result processing: Math	✓	✓
Result processing: Robotics	✓	✓

4.4.4 VISOR® Object AI

	VISOR® Object AI	
	Standard n.a.	**Advanced**
Applications	Presence, completeness, measuring, color, position control	
Resolution		
V10 (800 x 600): Mono | Color	– | ✓	
Frames per second: Mono | Color	– | 50	
V20 (1440 x 1080): Mono | Color	– | –	✓ | ✓
Frames per second: Mono | Color	– | –	40 | 20
V50 (2560 x 1936): Mono | Color	– | –	✓ | ✓
Frames per second: Mono | Color	– | –	22 | 8
Lighting	White | only for Mono: Red, Infrared	
Multishot (Mono)	–	–
Target laser	–	✓
Lenses		
V10Wide | Medium | Narrow | C-Mount	– | – | – | –	✓ | ✓ | ✓ | ✓
V20Wide | Medium | Narrow | C-Mount	– | – | – | –	✓ | ✓ | ✓ | ✓
V50Wide | Medium | Narrow | C-Mount	– | – | – | –	✓ | ✓ | – | ✓
Interfaces	Ethernet | EtherNet/IP | PROFINET	
Inputs | Outputs | Selectable	– | – | –	2 | 2 | 6
Encoder input	–	✓
Ethernet | EtherNet/IP	– | –	✓ | ✓
PROFINET | SensoWeb	– | –	✓ | ✓
Service port	–	✓
Job | Detectors		
Number of jobs (max.)	–	255
Number of detectors per job (max.)	–	255
Calibration		
Scaling (Measurement)	–	✓
Calibration plate (Measurement)	–	✓
Point pair list (Robotics)	–	–
Calibration plate (Robotics)	–	–
Hand-Eye calibration (Robotics)	–	–
Base-Eye calibration (Robotics)	–	–
Pre-processing		
Pre-processing filter	–	✓
Repeat mode	–	✓
Shutter variation	–	✓
Free-form search region	–	✓

	VISOR® Object AI	
	Standard n.a.	Advanced
Alignment		
Contour matching (translation, rotation 360°)	–	✓
Pattern matching (translation, rotation 360°)	–	✓
Edge detector (translation, rotation)	–	✓
Object detection		
Contour (translation, rotation 360°) Multiple objects	– –	✓ ✓
Contour 3D (translation, rotation 360°) Multiple objects	– –	– –
Pattern matching (translation, rotation 360°) Multiple objects	– –	✓ ✓
Gray	–	✓
Contrast	–	✓
Brightness	–	✓
Caliper	–	✓
BLOB	–	✓
Classification		
Classification (AI)	–	✓
Identification		
Barcode | Barcode Advanced	– | –	– | –
Datacode | Datacode Advanced	– | –	– | –
OCR	–	–
Robotic functions		
Result offset Image | 2D | 3D	– | – | –	– | – | –
Gripping space check	–	–
Color detectors for Color variants		
Color Area	–	✓
Color Value	–	✓
Color List	–	✓
Color distance | Binarization	–	✓
Result processing		
Result processing: Text	–	–
Result processing: Math	–	✓
Result processing: Robotics	–	–

4.4.5 VISOR® Allround

	VISOR® Allround	
	Advanced	**Professional**
Applications	Presence, completeness, measurements, color Reading of barcodes, data codes, text, multishot, position control	
Resolution		
V10 (800 x 600): Mono | Color	✓ | ✓	– | –
Frames per second: Mono | Color	75 | 50	– | –
V20 (1440 x 1080): Mono | Color	✓ | ✓	
Frames per second: Mono | Color	40 | 20	
V50 (2560 x 1936): Mono | Color	– | –	✓ | ✓
Frames per second: Mono | Color	– | –	22 | 8
Illumination	White | only for Mono: Red, Infrared	
Multishot (Mono)	✓	✓
Target laser	✓	✓
Lenses		
V10 Wide | Medium | Narrow | C-Mount	✓ | ✓ | ✓ | ✓	– | – | – | –
V20 Wide | Medium | Narrow | C-Mount	✓ | ✓ | ✓ | ✓	
V50 Wide | Medium | Narrow | C-Mount	– | – | – | –	✓ | ✓ | – | ✓
Interfaces	Ethernet | EtherNet/IP | PROFINET	
Inputs | Outputs | Selectable	2 | 2 | 6	
Encoder input	✓	✓
Ethernet | EtherNet/IP	✓ | ✓	✓ | ✓
PROFINET | SensoWeb	✓ | ✓	✓ | ✓
Service port	✓	✓
Job | Detectors		
Number of jobs (max.)	255	
Number of detectors per job (max.)	255	
Calibration		
Scaling (Measurement)	✓	✓
Calibration plate (Measurement)	✓	✓
Point pair list (Robotics)	–	✓
Calibration plate (Robotics)	–	✓
Hand-Eye calibration (Robotics)	–	✓
Base-Eye calibration (Robotics)	–	✓
Pre-processing		
Pre-processing filter	✓	✓
Repeat mode	✓	✓
Shutter variation	✓	✓
Free-form search region	✓	✓

	VISOR® Allround	
	Advanced	**Professional**
Alignment		
Contour matching (translation, rotation 360°)	✓	✓
Pattern matching (translation, rotation 360°)	✓	✓
Edge detector (translation, rotation)	✓	✓
Object detection		
Contour (translation, rotation 360°)	✓	✓
Multiple objects	✓	✓
Contour 3D	–	✓
Multiple objects	–	✓
Target Mark 3D	–	✓
Pattern matching (translation, rotation 360°)	✓	✓
Multiple objects	✓	✓
Gray	✓	✓
Contrast	✓	✓
Brightness	✓	✓
Caliper	✓	✓
BLOB	✓	✓
Classification		
Classification (AI)	–	✓
Identification		
Barcode | Barcode Advanced	✓ |✓	✓ |✓
Datacode | Datacode Advanced	✓ |✓	✓ |✓
OCR	✓	✓
Robotic functions		
Result offset Image | 2D | 3D	–|–|–	✓ |✓ |✓
Gripping space check	–	✓
Color detectors for Color variants		
Color Value	✓	✓
Color Area	✓	✓
Color List	✓	✓
Color distance | Binarization	✓	✓
Result processing		
Result processing: Text	✓	✓
Result processing: Math	✓	✓
Result processing: Robotics	–	✓

5 Installation

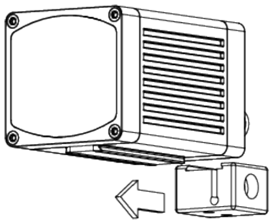
5.1 Mechanical installation

5.1.1 Installing the bracket

For installation, use the Mounting bracket MK 45 (543-11000) or the Mounting hinge MG 3A (543-11024) only.

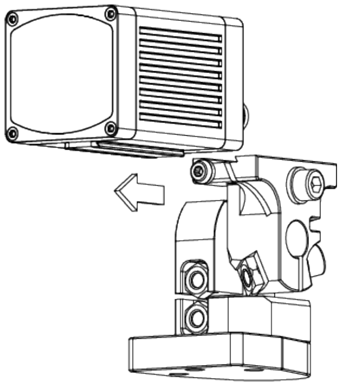
NOTE:
The Mounting hinge MG 3A is not included.
It is available under part number 543-11024 from SensoPart Industriesensorik GmbH.

Installing VISOR® on Mounting bracket MK 45



1. Slide the mounting bracket onto the sensor’s dovetail guide.
2. Use the Allen key to tighten the socket cap screw in the mounting bracket’s cross-hole.
3. Now install the mounting bracket on a suitable fixture.

Installing VISOR® on Mounting hinge MG 3A



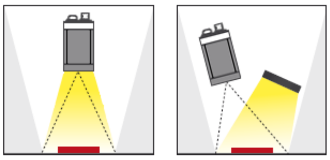
1. Slide the mounting hinge's dovetail onto the sensor’s dovetail guide.
2. Use the Allen key to tighten the socket cap screw in the mounting hinge’s cross-hole.
3. Now install the mounting hinge on a suitable fixture.

5.1.2 Illumination

5.1.2.1 Sensor and illumination configuration

The terms "bright field illumination," "dark field illumination," and "diffuse illumination" are used to distinguish between three sensor and illumination configurations.

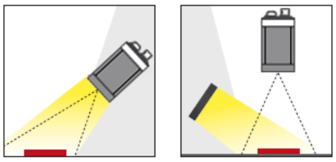
Arrangement for bright-field illumination



When using bright field illumination, the positions of the sensor, object, and illumination are chosen in such a way that the light will be reflected directly from the object surface towards the sensor.

Smooth object surfaces will appear bright, while bumps and depressions will appear dark.

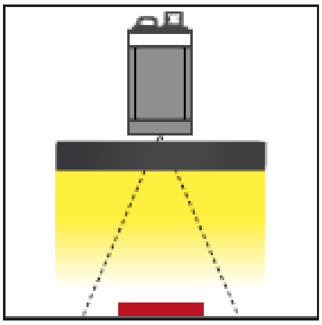
Arrangement for dark-field illumination



When using dark field illumination, the positions of the sensor, object, and illumination are chosen in such a way that the light will not be reflected directly from the object surface towards the sensor.

Smooth object surfaces will appear dark, while bumps and depressions will appear bright.

Diffuse illumination configuration



Diffuse illumination is only possible with an external source of lighting.
Diffuse illumination is used everywhere where highly reflective, curved, or, above all, irregularly shaped object surfaces are concerned (e.g. aluminum foil on blister packs, etc.). Such objects can only be illuminated with diffuse illumination (i.e. uniform illumination coming from all directions), and not spot-shaped illumination. Diffuse illumination is also known as "cloudy day" illumination, i.e. uniform light from behind the cover of clouds as a light source rather than from direct sunlight.

NOTE:
External sources of light are available from SensoPart Industriesensorik GmbH.
Additional information: Accessories

Fine adjustment

It will not be possible to carry out a fine adjustment on the vision sensor until after electrical installation and the initial setup (VISOR® software installation).

5.1.2.2 Mechanical isolation from ambient light

Ambient light from windows or skylights that disrupts the scene only temporarily on certain days / seasons of the year can often be blocked with mechanical elements.

5.1.2.3 Version with Infrared illumination

Another option for becoming more independent of ambient light is to use the corresponding VISOR® variant with IR illumination. Here, the test scene is illuminated with built-in, powerful IR illumination. The receiver is equipped with appropriate filters that only let light within this spectrum get through to the receiver, i.e., the sensor operates within a narrow wavelength range and, to the greatest extent possible, only with the light it emits itself.

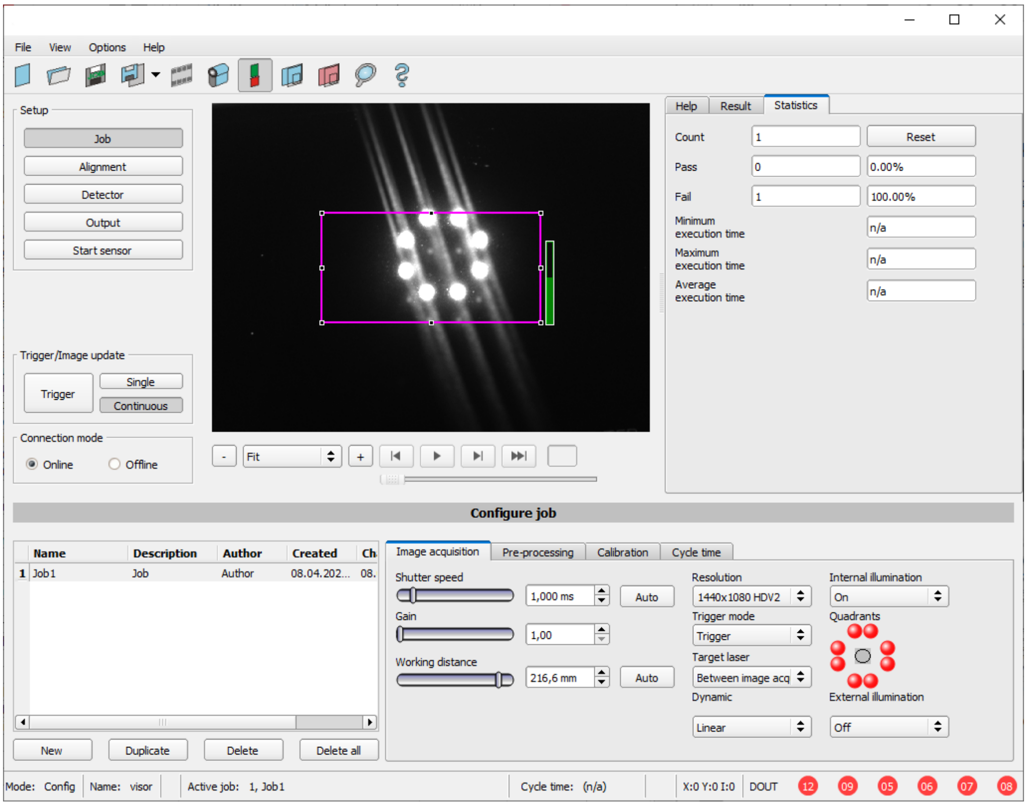
Another advantage of using infrared illumination consists of the fact that the flashes are invisible and accordingly will not inconvenience any people working in the area.

5.1.2.4 Version with UV illumination

The VISOR® vision sensor with UV illumination is used to detect marks applied with luminescent ink. These can be codes or other characters, for example. The mark remains invisible in the visible spectrum of light. Only when illuminated with UV light does the marking become visible and luminesce in a visible color such as red or green. This effect allows invisible marks to be applied or marks to be detected regardless of the background color.

5.1.2.5 Alignment for vertical illumination

In order to ensure that the VISOR® is perfectly perpendicularly aligned with the object surface, place a piece of reflective foil or a mirror on the object and start the VISOR® operating software as a test. For an image that is continually updated, select the trigger mode "Free run" and under Trigger / Image update "Continuous". Now align the sensor as perpendicular as possible to the reflective / mirror surface until the integrated illumination LEDs dazzle directly into the image of the user interface. Arrangement see Figure in Chapter Illumination



Connection mode: ● Online ○ Offline

Count	1
Pass	0 / 0.00%
Fail	1 / 100.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Name	Description	Author	Created	Ch
1	Job1	Job	Author	08.04.202...	08.

Shutter speed: 1,000 ms

Gain: 1,00

Working distance: 216,6 mm

Resolution: 1440x1080 HDV2

Trigger mode: Trigger

Target laser: Between image acq

Dynamic: Linear

Internal illumination: On

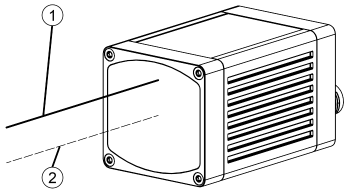
External illumination: Off

Mode: Config | Name: visor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 2: Figure for vertical illumination

5.1.3 Target laser

The laser can be configured in the VISOR® software in the SensoConfig module under Job / Image acquisition / Target laser.



The target laser's laser beam runs parallel to the optical axis at a distance of 12.5 mm and can be used to align the sensor.

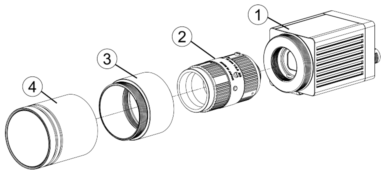
① Laser beam

② Optical axis

5.1.4 C-Mount lens and protective casing

Various lenses with different focal lengths are available for VISOR® C-Mount vision sensors. The lenses can be protected with a protective casing. Lenses and protective casings are available from SensoPart.

Installing the C-Mount lens and protective casing on the vision sensor



1. Screw the lens into the C-Mount vision sensor's internal thread.
2. Screw the protective casing's extension onto the VISOR® vision sensor's external thread.
3. Screw the protective casing onto the extension's external thread.

① C-Mount VISOR® vision sensor

② Lens

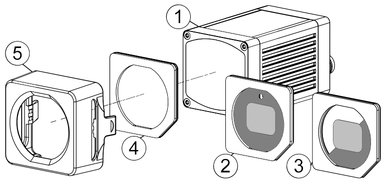
③ Extension for the protective casing

④ Protective casing

5.1.5 Polarizing filters and spark protection guard

ATTENTION:
If you use the spark protection guard, the maximum operating temperature is lowered to 45 °C.

Various polarizing filters and a spark protection guard are available for the VISOR® vision sensor.



① **VISOR® vision sensor**

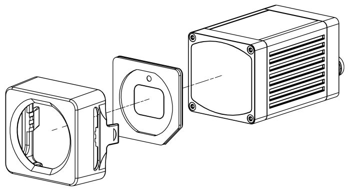
② **Polarizing filter for spark protection with 100% coverage.**

③ **Polarizing filter for spark protection with 50% coverage.**
Individual quadrants of the internal illumination can be turned on and off in the VISOR® software. If the only quadrants that are turned on are the ones within the 50% polarizing filter cover, polarized light will be emitted. If the quadrants that are turned on are the ones in the clear half of the polarizing filter, unpolarized light will be emitted instead.

④ **Spark protection guard**

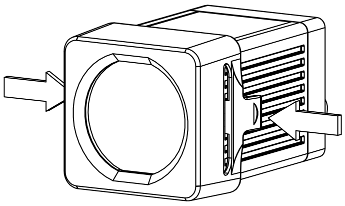
⑤ **Spark protection**

Installing the polarizing filter and spark protection guard



1. Insert the polarizing filter, aligned at the chamfer with the foam side facing towards the front of the sensor, into the spark protection guard.
2. Slide the polarizing filter disk and spark protection guard onto the front of the sensor until the guard locks into place on the sensor case and you hear two clicks.

Removing the polarizing filter and spark protection guard



1. Squeeze the two tabs on the side of the spark protection guard at the same time. The spark protection guard will come loose from the sensor case and you will be able to remove it.

5.2 Electrical installation

WARNING
The connection should be made exclusively by trained qualified personnel.
All live components must be de-energized when performing the electrical installation work.

ATTENTION:
When using the unit on a network, it is necessary to ensure that the vision sensor's default network address (IP address), 192.168.100.100 / 24, is free and that it is not being used by any other device connected to the network. If necessary, the vision sensor's IP address must be changed. For more information, please refer to "Network settings."
For error-free operation, the length of the connecting cables must not exceed 30 m. Failure to do this may cause malfunction.

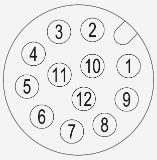
For stand-alone operation (independent of PC / PLC), only connection 24 V DC is required after startup.

5.2.1 24 V DC connection

M12 connection socket for voltage supply and digital I/O.

ATTENTION:
Use shielded cables exclusively, and terminate the shield across a large area.
Tightening torque for connector: 0.6 – 1 Nm.

I/O Mapping

	PIN	Color [3]	Signal
	1	BN	+ U_B (24 V DC)
	2	BU	GND
	3	WH	IN (external trigger)
	4	GN	READY (ready for next external trigger)
	5 [1]	PK	IN/OUT (encoder B+)
	6	YE	IN/OUT, (external illumination, external illumination south)[4]
	7	BK	IN/OUT, (external illumination, external illumination east)[4], LED B [2]
	8	GY	IN/OUT, (external illumination, external illumination north)[4], LED C [2]
	9	RD	IN/OUT, (external illumination, external illumination east)[4]
	10 [1]	VT	IN (encoder A+)
	11	GY/PK	VALID (indicator for valid results)
	12	RD/BU	IN/OUT (ejector), LED A [2]

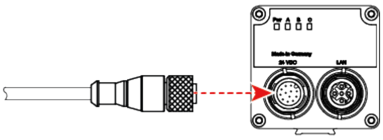
[1] Not available on all Standard models

[2] All indicator LEDs are set without consideration of any delay times used

[3] Colors match the SensoPart power cables. If other cables are used, there might be differences.

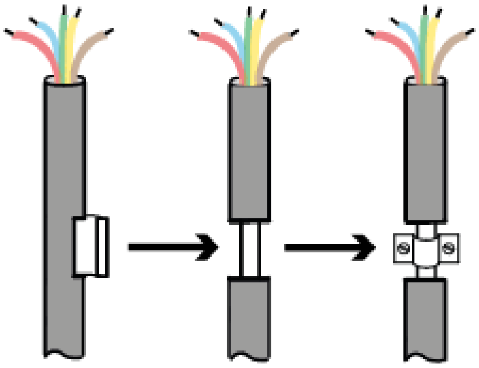
[4] Only if Multishot is active

Connecting the power cable



1. Connect the power cable to the 24 VDC connector on the vision sensor.
2. Fasten the connector and tighten it with a torque of 0.6 to 1 Nm.

Terminating the shield



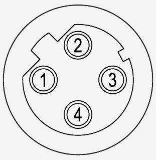
1. Strip a section of the power cable (remove a section of the jacket). The power cable’s shield will be exposed.
2. Place a shielding clamp or a similar component over the stripped cable section and secure this shielding clamp to a shielding plate.

5.2.2 LAN connection

M12 connector for Ethernet connection.

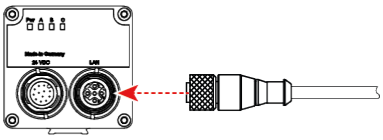
ATTENTION:
Use only the correct network cables.
Tightening torque for connector: 0.6 – 1 Nm.

I/O Mapping

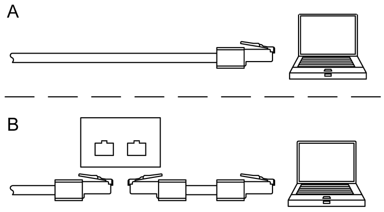
	PIN	Signal
	1	TxD+
	2	RxD+
	3	TxD-
	4	RxD-

Connecting the Ethernet cable

The vision sensor can be connected either directly to a PC (the preferred option) or to a PC through a network.



1. Connect the Ethernet cable to the vision sensor’s LAN connector.
2. Fasten the connector and tighten it with a torque of 0.6 to 1 Nm.



3. Use the RJ45 connector to connect the Ethernet cable either
   A) Directly to your PC (the preferred option).
   B) To your PC through a network.

5.2.3 Exemplary connection plan

Exemplary connection plan for the following configuration:

- Power supply
- Trigger
- 1x digital switch output
- Encoder
- Ethernet to PC or PLC

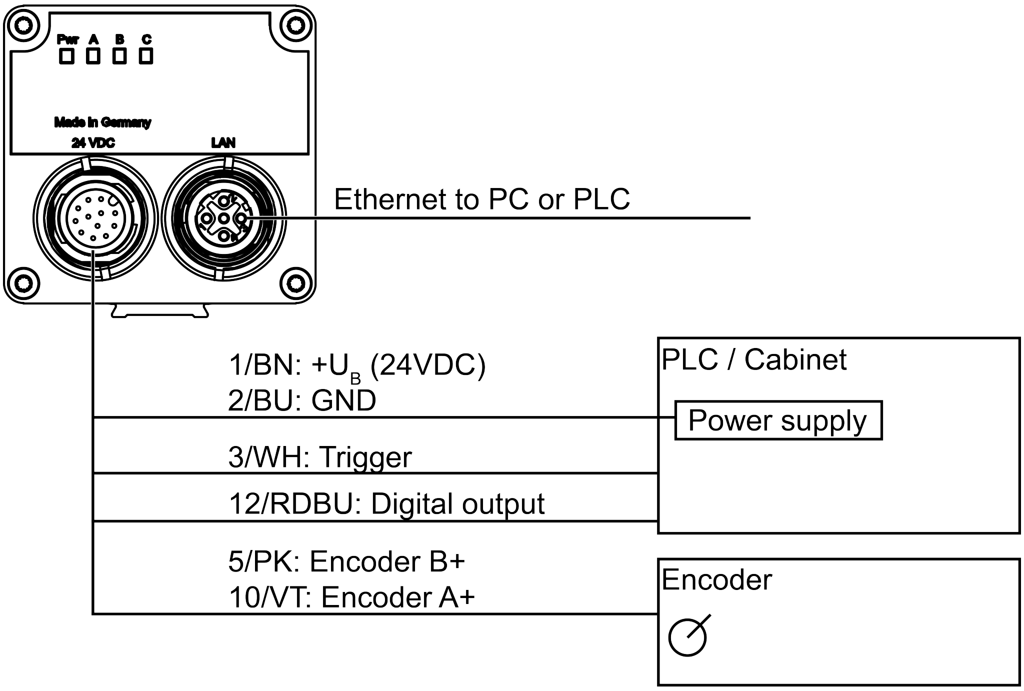


Fig. 3: *Exemplary connection plan*

5.2.4 Electrical connection Supply voltage with shield

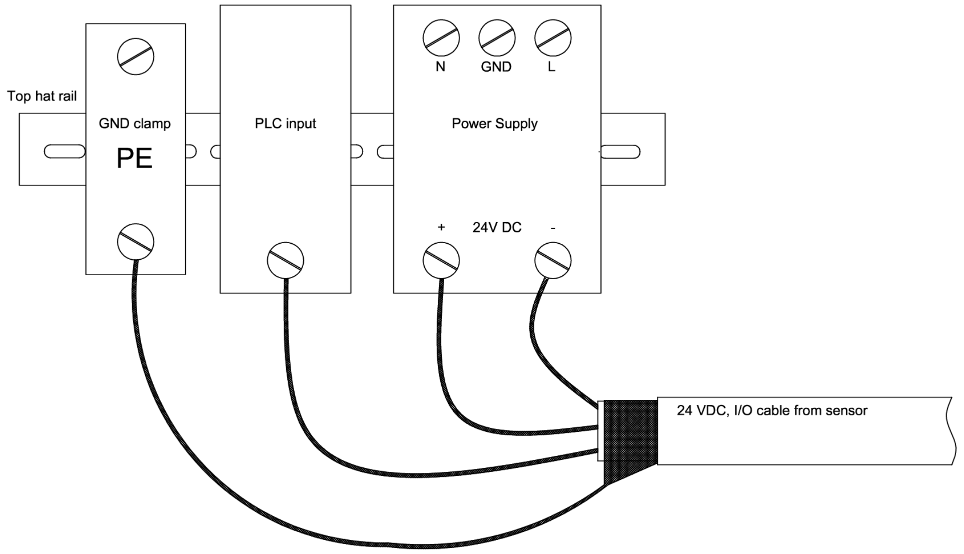


Fig. 4: *Power supply 24 V DC in the control cabinet with shield*

5.2.5 Electrical connection PNP / NPN

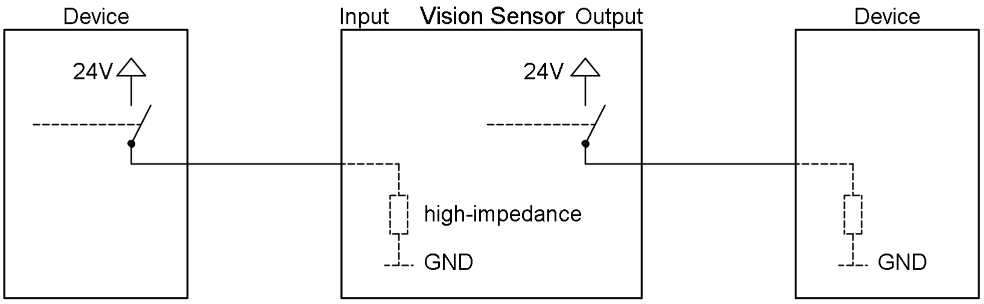


Fig. 5: *Connection example for VISOR® in PNP mode. Inputs/outputs switch to +24 V*

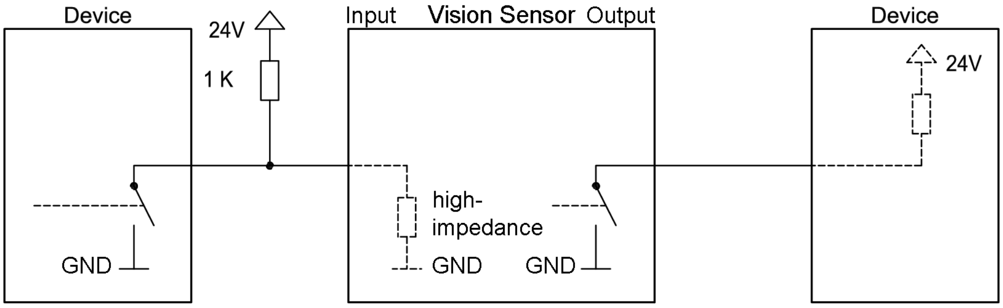


Fig. 6: *Connection example for VISOR® in NPN mode*

The inputs are ground-referenced. Accordingly, an additional pull-up resistor may be needed in certain cases to ensure that the input voltage will be increased to 24 V in an unswitched state. The outputs switch to ground.

5.3 Network connection

The following instructions explain how to modify the network configuration for the PC and the VISOR® vision sensor. If incorrect settings are used, the network connections in the computer may be lost. To be on the safe side, note the former settings and reuse them if required. Following this procedure, it may be necessary to restart the system. In order to determine which IP address is allowed in your network or locally on your PC, and to carry out the necessary settings on your PC, please contact the responsible system administrator or administrator beforehand. The illustrations, dialogues and menus used are taken from Microsoft Windows 10 operating system. The illustrations are similar in other operating systems.

5.3.1 Basic PC and VISOR® vision sensor settings

Prerequisite for configuring the VISOR® vision sensor with a PC: PC with network adapter and an installed TCP/IP LAN connection (even if the PC is not connected to a network). The VISOR® supports the automatic detection of the Ethernet transmission rate, but a maximum of 100 MBit. The internet protocol IPv4 must be activated. The VISOR® vision sensor can basically be configured and parameterized via two options: network connection and direct connection.

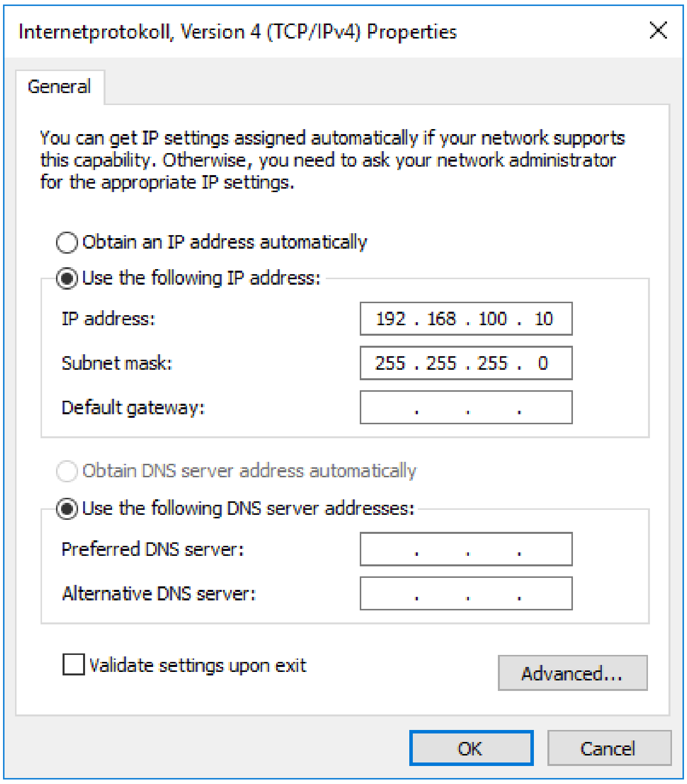
5.3.2 Direct connection - Setting the IP address of the PC

To connect the VISOR® vision sensor to a computer via Ethernet, the IP address settings of the two devices must correspond to each other. The default setting for the VISOR® vision sensor’s IP address is 192.168.100.100 / 24 with a subnet mask of 255.255.255.0. For direct connection, the PC must be set to a fixed IP address suitable for the sensor, as follows:

1. Clicking on Start / Control Panel / Network Connection / LAN Connection / Properties opens the dialog window "Local Area Connection Properties".
2. In the list "This connection requires the following elements", select the option "Internet Protocol (TCP/IP)" and click the button "Properties".
3. In the following window, set the desired IP address and subnet mask of the PC.
4. Confirm entries with OK.

Example:

The VISOR® vision sensor comes with its IP address set to 192.168.100.100 and its subnet mask set to 255.255.255.0. In this case, the IP address may be set to any value between 192.168.100.1 and 192.168.100.254, with a subnet mask 255.255.255.0, with the exception of the sensor IP address (192.168.100.100).
To alter the sensor’s IP address, see Network settings of the sensor. Do not use the network addresses .0 and .255 as device addresses, as these are mostly reserved for network infrastructure, such as servers, gateways, etc.



Internetprotokoll, Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 100 . 10

Subnet mask: 255 . 255 . 255 . 0

Default gateway: . . .

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternative DNS server: . . .

□ Validate settings upon exit

Fig. 7: *PC IP Setup*

5.3.3 Network connection - Setting the IP address of the VISOR® vision sensor

Before connecting the sensor to the network, check with the network administrator whether the sensor's address has already been assigned (default: 192.168.100.100 with subnet mask 255.255.255.0). This can otherwise cause network failure. The configured IP address should be written down on the enclosed label for the VISOR® vision sensor. The label must then be stuck on the sensor in a clearly visible place after installation.

Network connection speed:

Especially when using the VGA resolution and SensoView, the sensor should be operated with 100 Mbit /full-duplex only.

Sensor IP still free:

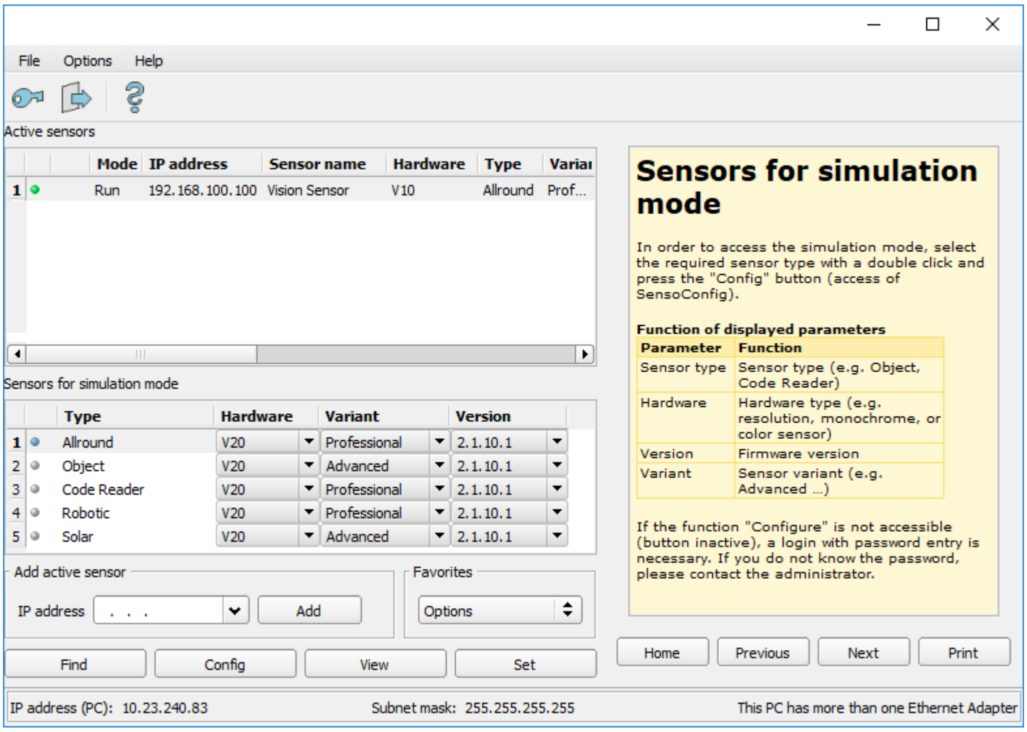
Connect the sensor to the network and then set the sensor's IP to match the administrator's specifications, as follows, beginning with Point 2.

Sensor IP already assigned:

1. First connect sensor and PC directly and set an authorized IP address in the sensor.
2. Connection via the network can then be carried out. First ensure electrical connection and installation of PC software has been completed. To set the IP address on the VISOR® vision sensor, carry out the following steps in the PC software:
   a. Start SensoFind.
   b. Select the VISOR® vision sensor you want from the list of active sensors.
   c. Set sensor's new IP address with the "Set" button. The IP address is assigned by your system administrator. The PC's IP address is shown in the status bar under the buttons. Note:: Certain PCs may have more than

one Ethernet connection, i.e., wireless and wired LAN connections.

d. Select the sensor and connect via SensoConfig or SensoView.



Active sensors

	Mode	IP address	Sensor name	Hardware	Type	Variai
1	Run	192.168.100.100	Vision Sensor	V10	Allround	Prof...

Sensors for simulation mode

	Type	Hardware	Variant	Version
1	Allround	V20	Professional	2.1.10.1
2	Object	V20	Advanced	2.1.10.1
3	Code Reader	V20	Professional	2.1.10.1
4	Robotic	V20	Professional	2.1.10.1
5	Solar	V20	Advanced	2.1.10.1

Add active sensor

IP address . . .

Favorites

Options

Sensors for simulation mode

In order to access the simulation mode, select the required sensor type with a double click and press the "Config" button (access of SensoConfig).

Function of displayed parameters

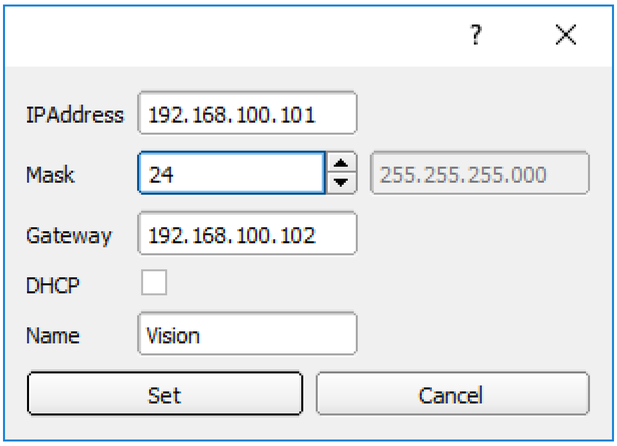
Parameter	Function
Sensor type	Sensor type (e.g. Object, Code Reader)
Hardware	Hardware type (e.g. resolution, monochrome, or color sensor)
Version	Firmware version
Variant	Sensor variant (e.g. Advanced ...)

If the function "Configure" is not accessible (button inactive), a login with password entry is necessary. If you do not know the password, please contact the administrator.

IP address (PC): 10.23.240.83 Subnet mask: 255.255.255.255 This PC has more than one Ethernet Adapter

Fig. 8: *SensoFind*

Modification of the standard gateway enables operation in different sub-networks. Only alter this setting after consultation with your administrator. Automatic integration of a new computer or sensor into the existing network without manual configuration is possible through DHCP. Normally, only the automatic reference of the IP address must be set at the sensor on the client. When the sensor is started on the network, it can obtain the IP address, net mask, and gateway from a DHCP server. Activation of the DHCP mode is carried out via the "Set" button by activating the checkbox "DHCP". Since this means that the exact same VISOR® may have different IP addresses at different times, a sensor name must be assigned when enabling DHCP. If there are multiple VISOR® sensors on a network, each one must be assigned its own unique name.



IPAddress 192.168.100.101

Mask 24 255.255.255.000

Gateway 192.168.100.102

DHCP □

Name Vision

Fig. 9: *VISOR® IP setup*

If a VISOR® with DHCP is turned on on a network without a DHCP server, the VISOR® will automatically set its IP address to 0.0.0.0. This can occur, for example, in the event of a power or server failure or in the event that the system is restarted. The reason for this is that the DHCP server may boot more slowly than the VISOR®. Make sure that the VISOR® is turned on only after the DHCP server is available.

6 VISOR® software – Overview and Quick Start Guide

The following describes the basic structure of the VISOR® software. The structure of the individual modules (SensoFind, SensoConfig and SensoView) is explained and short instructions are given for each module. This Quick Start Guide uses the example of an object sensor to explain the exemplary procedure for setting an inspection task on the VISOR® vision sensor.

6.1 Structure of the VISOR® software

The VISOR® software software is made up of the following three modules:

- **SensoFind**
  With this module, you can select the sensor to be configured or a sensor simulation, and start the applications SensoConfig or SensoView. Here, system settings such as IP addresses or firmware updates can be modified, and passwords and user rights can be managed.
- **SensoConfig**
  This module contains comprehensive functions for setting up sensors and configuring inspection tasks (jobs). If password protection is activated, you need the authorization of the user group administrator for the configuration.
- **SensoView**
  This module displays images and results. You can use it to monitor / check sensors and analyze measurement results. In addition, extensive archiving functions are available. Compared to SensoConfig, it only offers limited configuration options. If password protection is activated, the authorization of the user group administrator or worker is needed.

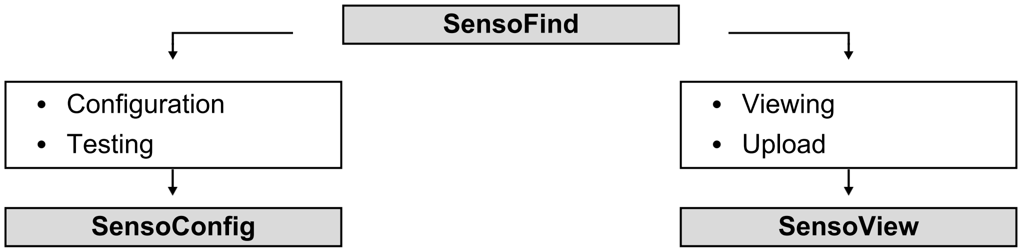


Fig. 10: *Software structure*

You can download free, up-to-date versions of the VISOR® software at www.sensopart.com.

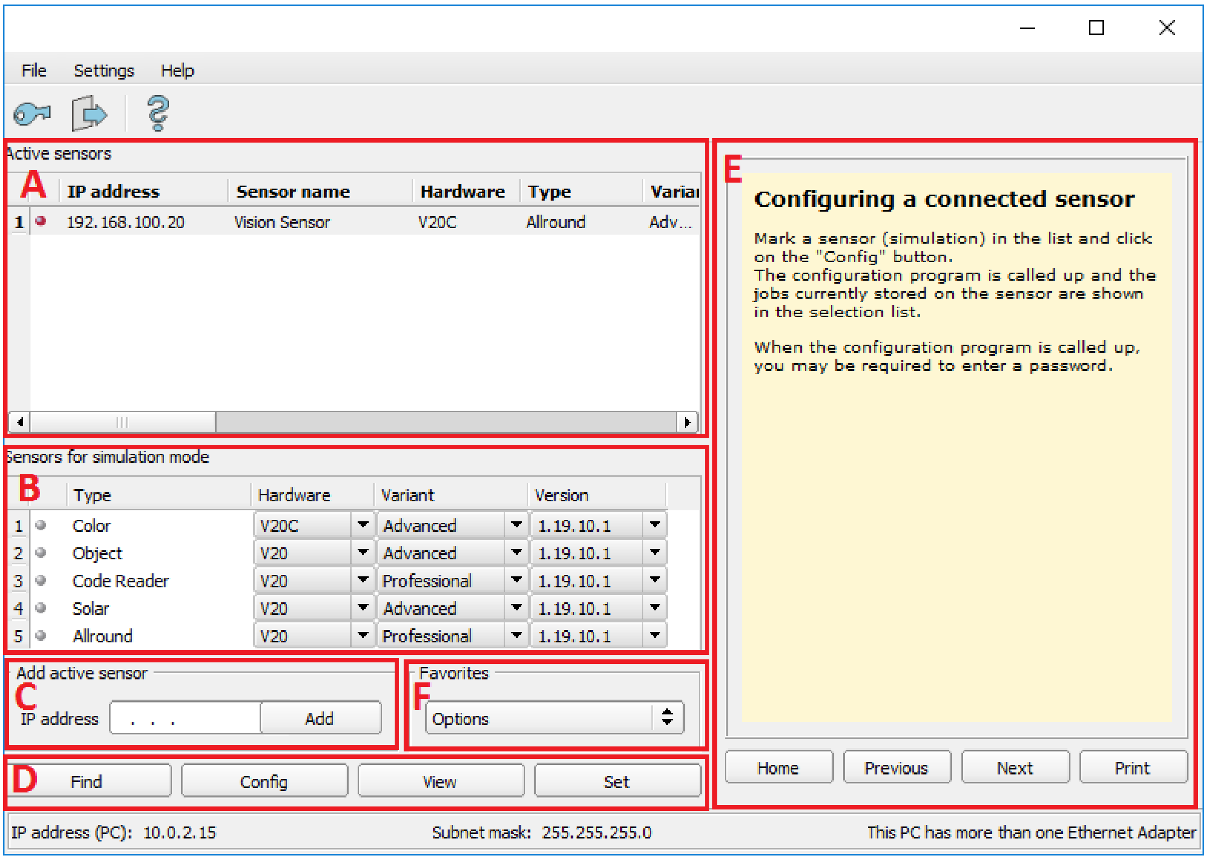
6.2 Start the VISOR® software

Click on the "VISOR Vision Sensor" desktop icon to start the VISOR® software.

Fig. 11: VISOR® software icon

6.3 SensoFind

6.3.1 SensoFind - Overview



Active sensors

		IP address	Sensor name	Hardware	Type	Varia
1	●	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

		Type	Hardware	Variant	Version
1	●	Color	V20C	Advanced	1.19.10.1
2	●	Object	V20	Advanced	1.19.10.1
3	●	Code Reader	V20	Professional	1.19.10.1
4	●	Solar	V20	Advanced	1.19.10.1
5	●	Allround	V20	Professional	1.19.10.1

Configuring a connected sensor

Mark a sensor (simulation) in the list and click on the "Config" button. The configuration program is called up and the jobs currently stored on the sensor are shown in the selection list.

When the configuration program is called up, you may be required to enter a password.

IP address (PC): 10.0.2.15   Subnet mask: 255.255.255.0   This PC has more than one Ethernet Adapter

Fig. 12: SensoFind Overview

A: Active sensors
This list shows all the VISOR® vision sensors that can be reached from the PC.

B: Sensors for simulation mode
Shows all the VISOR® vision sensors available for the offline simulation.

C: Add sensors via IP address
Sensors that do not appear in the list "Active sensors" after the software has started or after "Find" (triggering a further search run), but are definitely in the network (possibly behind a gateway, for example) and whose IP address is known, can be entered here with their IP address. By clicking the "Add" button, such sensors, if found, are also entered in the "Active sensors" list and can now be edited.

D: Functions

- **Find**
  Triggers another search
- **Configure**
  Configuration of a connected sensor or sensor simulation = SensoConfig
- **View**
  Displays image or result data from a connected sensor = SensoView
- **Web**
  Displays image or result data from a connected sensor in the standard web browser
- **Settings**
  Edits network settings such as the sensor's IP address, etc.

E: Context help
Context-sensitive help for the current topic

F: Favorites
VISOR® vision sensors can be saved as favorites. The favorites are used for quick access and management of the VISOR® vision sensors.

Additional information: VISOR® Software – SensoFind

6.3.2 SensoFind – Quick Start Guide

In this program, you can select a sensor or a sensor simulation for configuration or display (monitoring) and carry out different basic settings.

6.3.2.1 Open sensors or sensor simulations

6.3.2.1.1 Configuring or displaying sensors

In order to open a sensor for configuration or display, select with a single left mouse click the required sensor in the "Active sensors" list. Then, with a click on the **"**Config**"** button, the "SensoConfig" module starts. With the button "View", the "SensoView" module starts.

6.3.2.1.2 Sensor simulation

To open a sensor for offline simulation, highlight the desired sensor in the "Sensors for simulation mode" list. Then, with a click on the "Config" button, the "SensoConfig" module starts.

6.3.2.2 Passwords

6.3.2.2.1 Set up passwords

At the first start after the installation, the password input is completely deactivated and the auto-login as administrator is preset. If parameter settings should be protected against unauthorized access, passwords for the password levels "Admin" and "User" should be assigned. This can be accessed via the menu bar File / User administration or via the button with the key symbol in the toolbar.

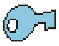


Fig. 13: *Password button*

6.3.2.2.2 Password levels

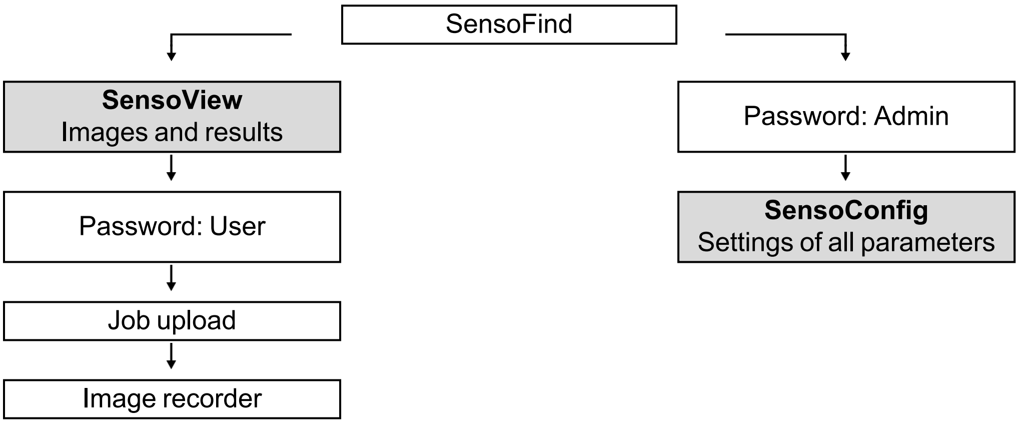


Fig. 14: *Password levels*

Password level	SensoFind	SensoConfig	SensoView
Administrator Password	All functions	All functions	All functions
Worker Password	All functions except • Configuration • Settings • Update	None	All functions, including Job Upload and Image Recorder
User (without password)	All functions except • Configurating • Settings • Update	None	Only display of images, inspection results, and statistics

In order to be able to use the "Config" function after assigning passwords, a login is now necessary: To login, click on the Login button in the toolbar and enter the previously assigned password.

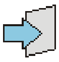


Fig. 15: *Login button*

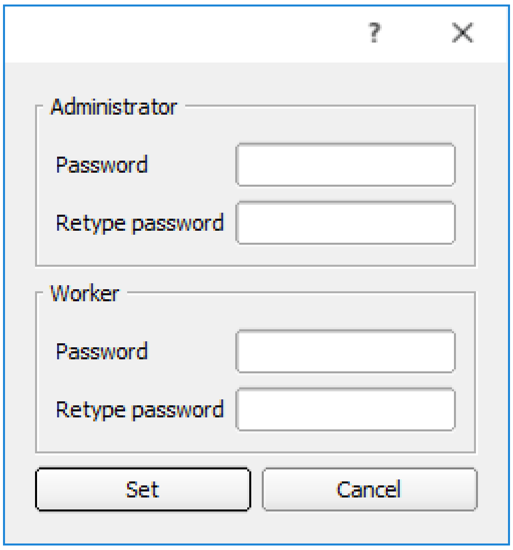


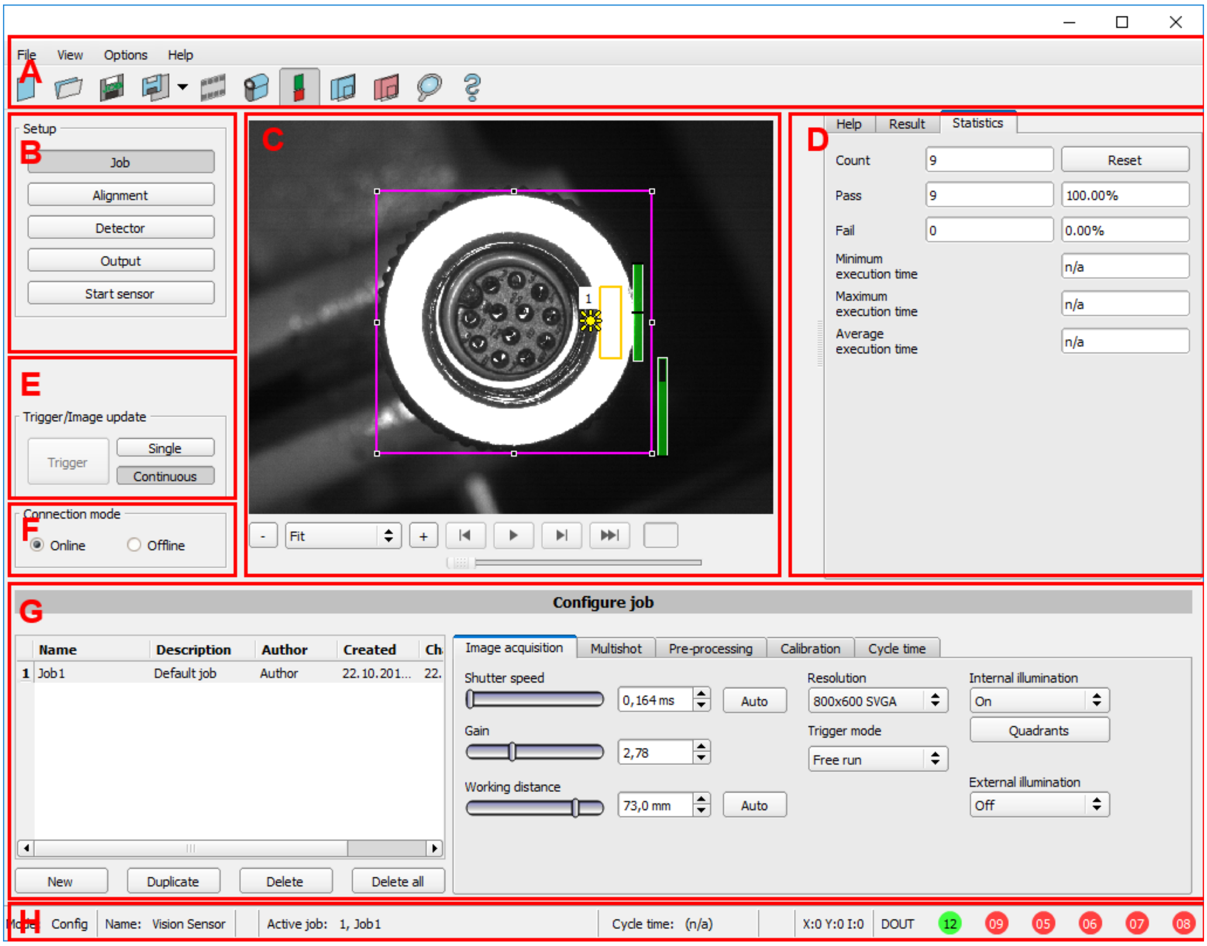
Fig. 16: *Password input*

By assigning an empty password, the query can be acknowledged again without further input. By activating the checkbox "Deactivate password query" the query will be deactivated permanently.

If passwords have been assigned and then forgotten, the software can be reset to the delivery status by reinstalling the software.

6.4 SensoConfig

6.4.1 SensoConfig - Overview



Count: 9
Pass: 9 100.00%
Fail: 0 0.00%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

Connection mode: ● Online ○ Offline

	Name	Description	Author	Created	Ch
1	Job1	Default job	Author	22.10.201...	22.

Shutter speed: 0,164 ms
Gain: 2,78
Working distance: 73,0 mm
Resolution: 800x600 SVGA
Trigger mode: Free run
Internal illumination: On
External illumination: Off

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT

Fig. 17: SensoConfig Overview

The different areas are:

A: Menu and toolbar. Under "Options" you can change the language of the user interface. Afterwards, a restart is not required.
Please find the menu item "Classification (AI): Open folder of training data sets" under Detector Classification (AI).

B: Setup steps: VISOR® Software – SensoConfig

C: Image
Image output with graphically adjustable working and regions of interest as well as zoom function also filmstrip navigation

D: Help, Result, Statistics

- Help: Context-sensitive help for the current topic
- Result: Detector results for selected parameters
- Statistic: Display statistic on evaluation and execution time

E: Image acquisition settings
Switchover between continuous mode and single-frame mode and software trigger

F: Connection mode
Switchover between online and offline mode (sensor present or simulation without sensor)

G: Configuration window
Variable, content changes to the corresponding action to set the associated parameters.

H: Status bar
Various status information, including: Mode / name of VISOR® / active job. In the Run mode: Cycle time, xy position of the cursor and pixel intensity / individual I/O on/off status (as configured below in "Output / Digital output").

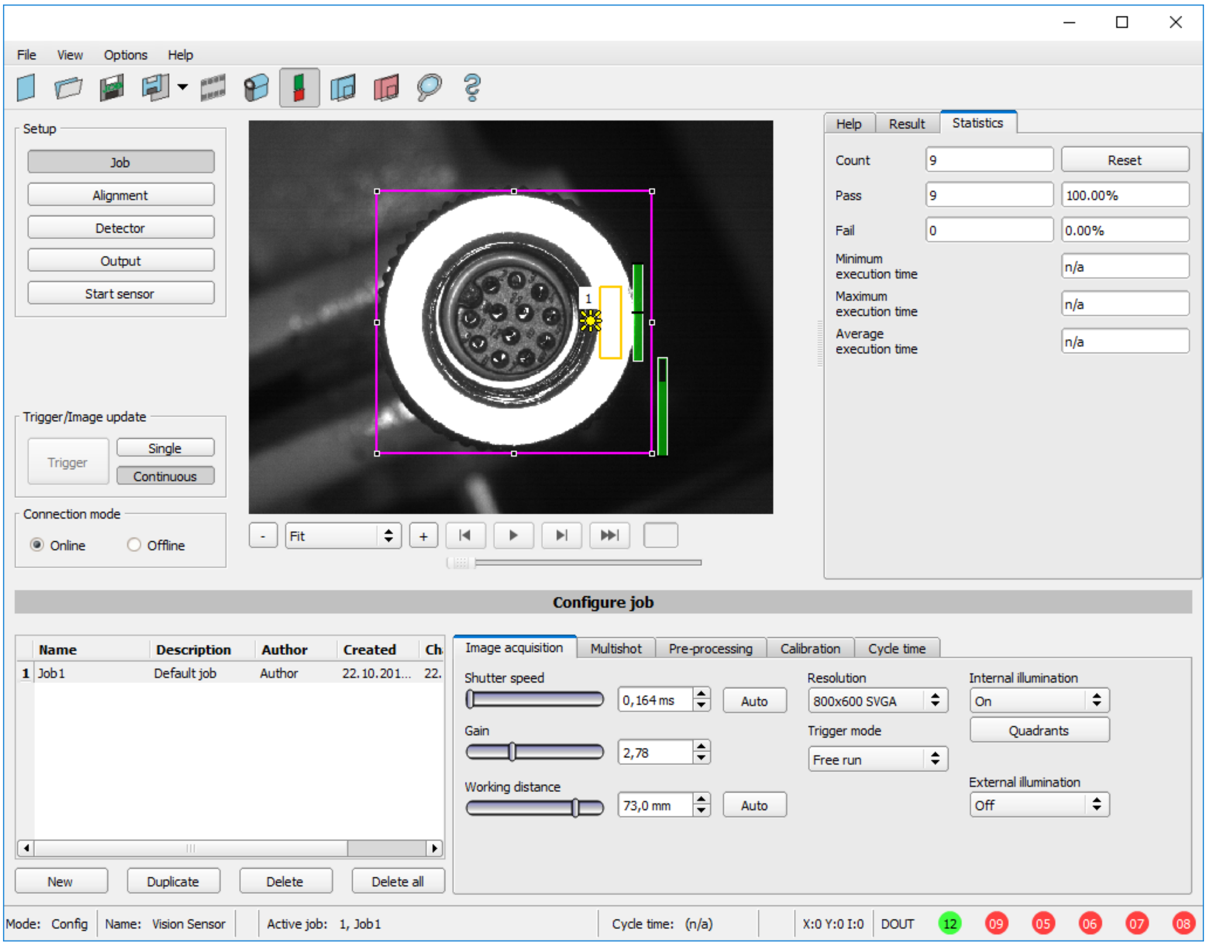
Additional information: VISOR® Software – SensoConfig

6.4.2 SensoConfig – Quick Start Guide

You can use this program to configure your VISOR® for one or more jobs in five setup steps.

6.4.2.1 Configuring a job

To configure a job: Edit the job entry under Setup / Job or create a new job.



Count 9
Pass 9 100.00%
Fail 0 0.00%
Minimum execution time n/a
Maximum execution time n/a
Average execution time n/a

	Name	Description	Author	Created	Ch
1	Job1	Default job	Author	22.10.201...	22.

Shutter speed 0,164 ms
Gain 2,78
Working distance 73,0 mm
Resolution 800x600 SVGA
Trigger mode Free run
Internal illumination On
External illumination Off

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 18: SensoConfig Job

Here, new jobs are created and the jobs are managed. In addition, this is where all global settings that are valid for the entire inspection task, such as shutter speed, gain, illumination settings, etc. are carried out.

A job contains all the settings and parameters required to perform a specific inspection task.

- The following basic image settings should first be made to ensure a high-contrast and sharp image:
  - Image brightness: Adjust Shutter speed or Gain (see setup step Job/ Image acquisition tab)
  - Sharpness: Adjust the sharpness of the image by adjusting the "Working distance" control until a sharp image is visible (see chapter Job / Image acquisition tab).
- In the delivery state, the settings are Trigger mode = "Free run" (see Job / Image acquisition) and "Trigger / Image update" = "Continuous". Thus, a new image is permanently fed into the focus and brightness setting, thus permanently updating the display.
- The subsequent adjustment of the Alignment and the detectors is preferably carried out in single image mode, since all settings are then based on a master image and the image input is not permanently executed. For this purpose set the Trigger mode = Trigger in the Image acquisition tab.

- Within a job, Alignment and (depending on the sensor type) 32 or 255 different detectors to solve the inspection task can be defined below.

It is possible to save a job as a template. To do this, right-click on the job in the job list and select "Save as template". For each new job, the settings and detectors are then copied from the job template. In the job list, the job template is identified with a "T" (Template). The job template cannot be edited. To remove the job template, right-click on the template and select "Remove".

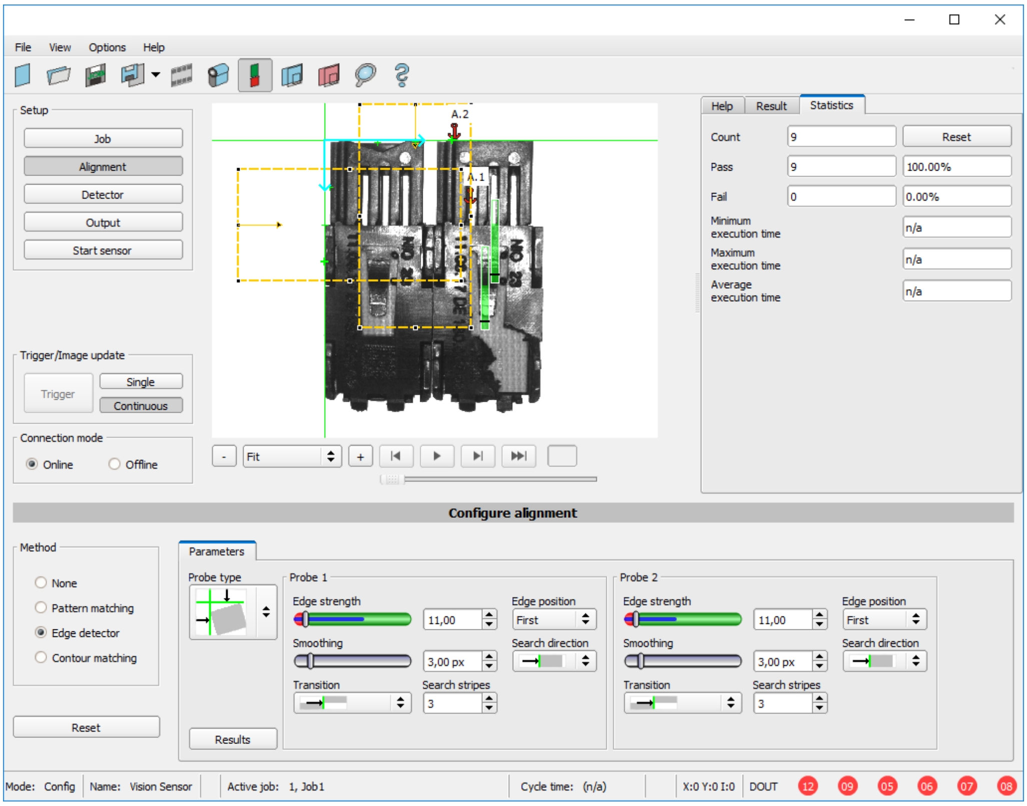
6.4.2.2 Configuring Alignment

For objects or features whose position varies in the image, Alignment may be useful or necessary.

Alignment is optional and is available with the methods Pattern matching, Edge detector and Contour matching.

First select the appropriate Alignment method. Then graphically set the position and size of the corresponding work areas on the screen to the characteristic that is to be used to determine the position. The associated parameters are displayed on the bottom right-hand side and can also be adjusted there.

Alignment affects all detectors defined below in this job. In the example here, the upper left corner of the rectangular component, which only varies in position translatorically in X and Y direction, is used to determine the position of the component. Therefore the left and the upper outer edge and their intersection point are determined. If the angular position of the component can also vary, the "Contour matching" method should be used for Alignment.



Count: 9
Pass: 9 — 100.00%
Fail: 0 — 0.00%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

Connection mode: ● Online ○ Offline

Method: ○ None ○ Pattern matching ● Edge detector ○ Contour matching

Probe 1: Edge strength 11,00; Edge position First; Smoothing 3,00 px; Search direction; Transition; Search stripes 3

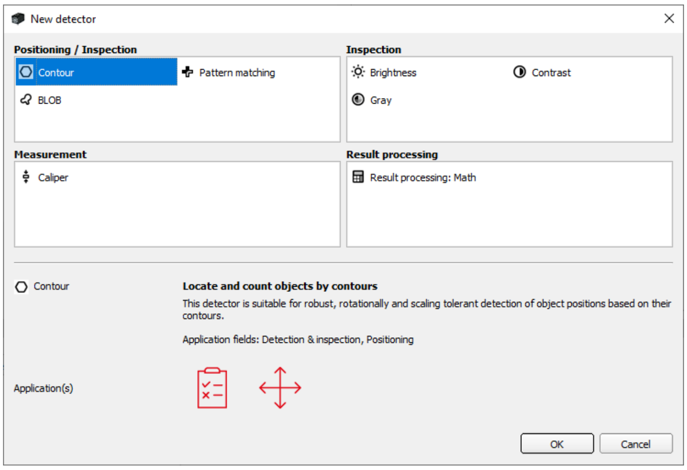
Probe 2: Edge strength 11,00; Edge position First; Smoothing 3,00 px; Search direction; Transition; Search stripes 3

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

__Fig. 19:__ SensoConfig, Alignment Edge detector

6.4.2.3 Configuring detectors

In the setup step Detectors, detectors can be selected and set to solve an inspection task.



Locate and count objects by contours

This detector is suitable for robust, rotationally and scaling tolerant detection of object positions based on their contours.

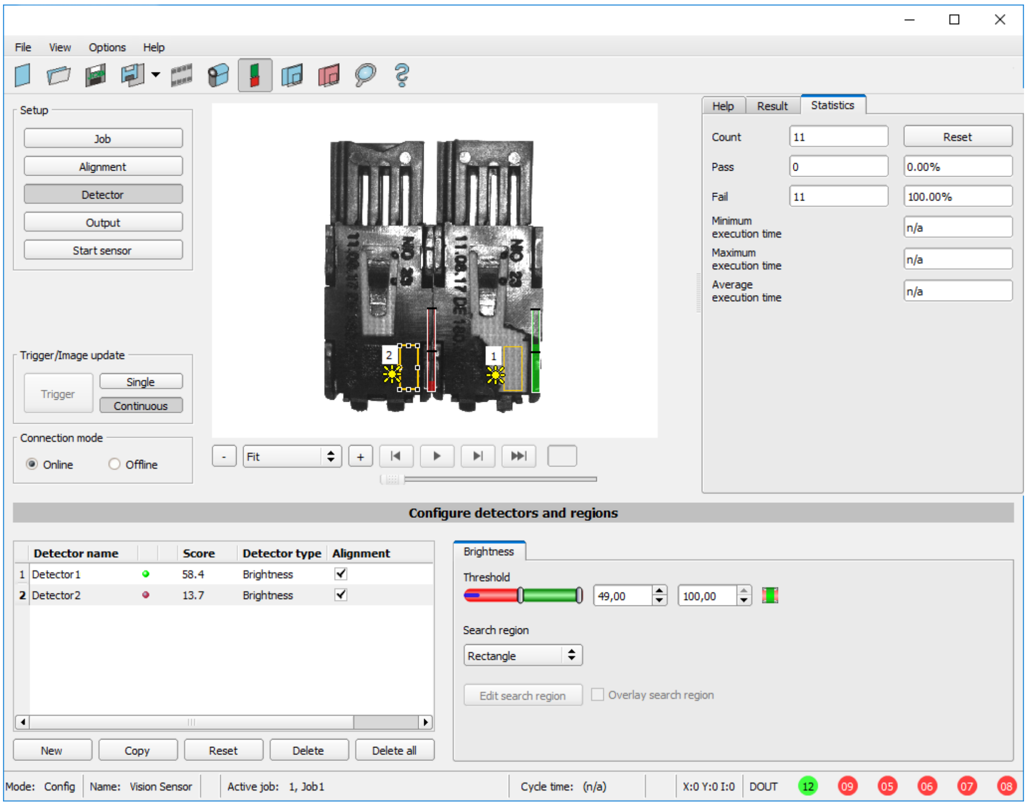
Application fields: Detection & inspection, Positioning

Fig. 20: *Detector selection list, example: VISOR® Object*

First select a suitable detector from the dialog shown above. Then set the work and regions of interest graphically in the image. If there are teach-in areas, they are taught-in immediately upon completing the setting. At the bottom left, all the detectors defined in this job are displayed in the detector list. At the bottom right, the parameters of the currently selected detector are displayed and can be adjusted there.

If additional features should be tested on the same part, you can use "New" to create any number of additional detectors, analogous to those described above. In the example, two brightness detectors were defined to check the presence of contacts in the test piece.

- Detector 1 finds a contact (brightness value is within the required range, as there is a metallically shiny, i.e. highly reflective contact) and therefore reports a positive result.
- Detector 2 finds no contact (brightness value is outside the required range, as there is hardly any reflection from dark plastic housing) and therefore reports a negative result.



Connection mode: ● Online ○ Offline

Fit

Statistics		
Count	11	
Pass	0	0.00%
Fail	11	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Configure detectors and regions

	Detector name	Score	Detector type	Alignment
1	Detector1	🟢 58.4	Brightness	☑
2	Detector2	🔴 13.7	Brightness	☑

Brightness

Threshold 49,00 100,00

Search region: Rectangle

☐ Overlay search region

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

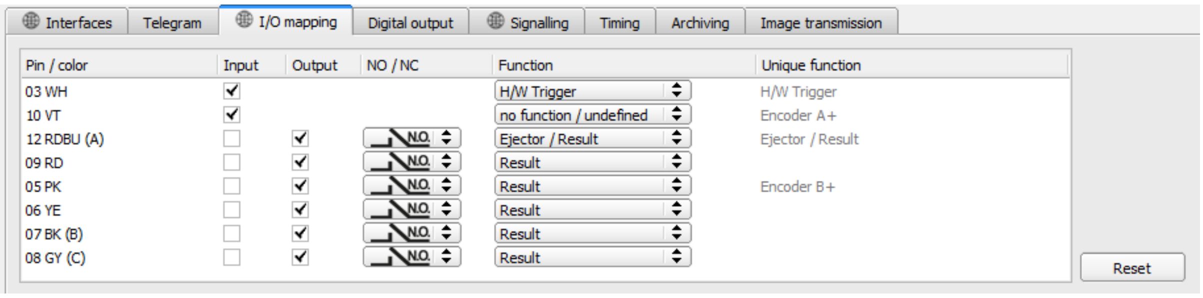
Fig. 21: Set detector

6.4.2.4 Output, I/O and data output

The setup step Output enables different settings of digital inputs / outputs and data output.

The interfaces can be selected and activated in the various tabs. Detector results can be logically linked and assigned to the existing I/Os.

The desired interface is also selected for the output of result data and the data string is compiled.


Interfaces Telegram I/O mapping Digital output Signalling Timing Archiving Image transmission
Reset

Pin / color	Input	Output	NO / NC	Function	Unique function
03 WH	☑			H/W Trigger	H/W Trigger
10 VT	☑			no function / undefined	Encoder A+
12 RDBU (A)	☐	☑	N.O.	Ejector / Result	Ejector / Result
09 RD	☐	☑	N.O.	Result	
05 PK	☐	☑	N.O.	Result	Encoder B+
06 YE	☐	☑	N.O.	Result	
07 BK (B)	☐	☑	N.O.	Result	
08 GY (C)	☐	☑	N.O.	Result	

Fig. 22: Output, Digital signals, and Data

Setting possibilities in the different tabs:

- **Interfaces**
  Selection, Setting, and Activation of the individual interfaces.
- **Telegram**
  Used to configure the data output string via Ethernet or PROFINET.
- **I/O mapping**
  Used to select and map digital switching inputs and outputs.

- **Digital output**
  Assignment of a logical link using the Boolean results of all detectors.
  Definition of complex logical links via table or via input of a logical formula.
  A separate logical link can be assigned to each existing digital output.
- **Signalling**
  Settings for statistics and for digital outputs.
- **Timing**
  Used to configure delays: Trigger delay, result delay, result duration
- **Archiving**
  Used to configure data archiving.
- **Image transmission**
  Used to configure image transmission via image recorder or RAM drive.

Selection of: Binary or ASCII protocol, start / trailer, standard content / flexibly configurable, special individual data of the individual detectors.
Any number of individual results of all defined detectors can be freely arranged in the output string.

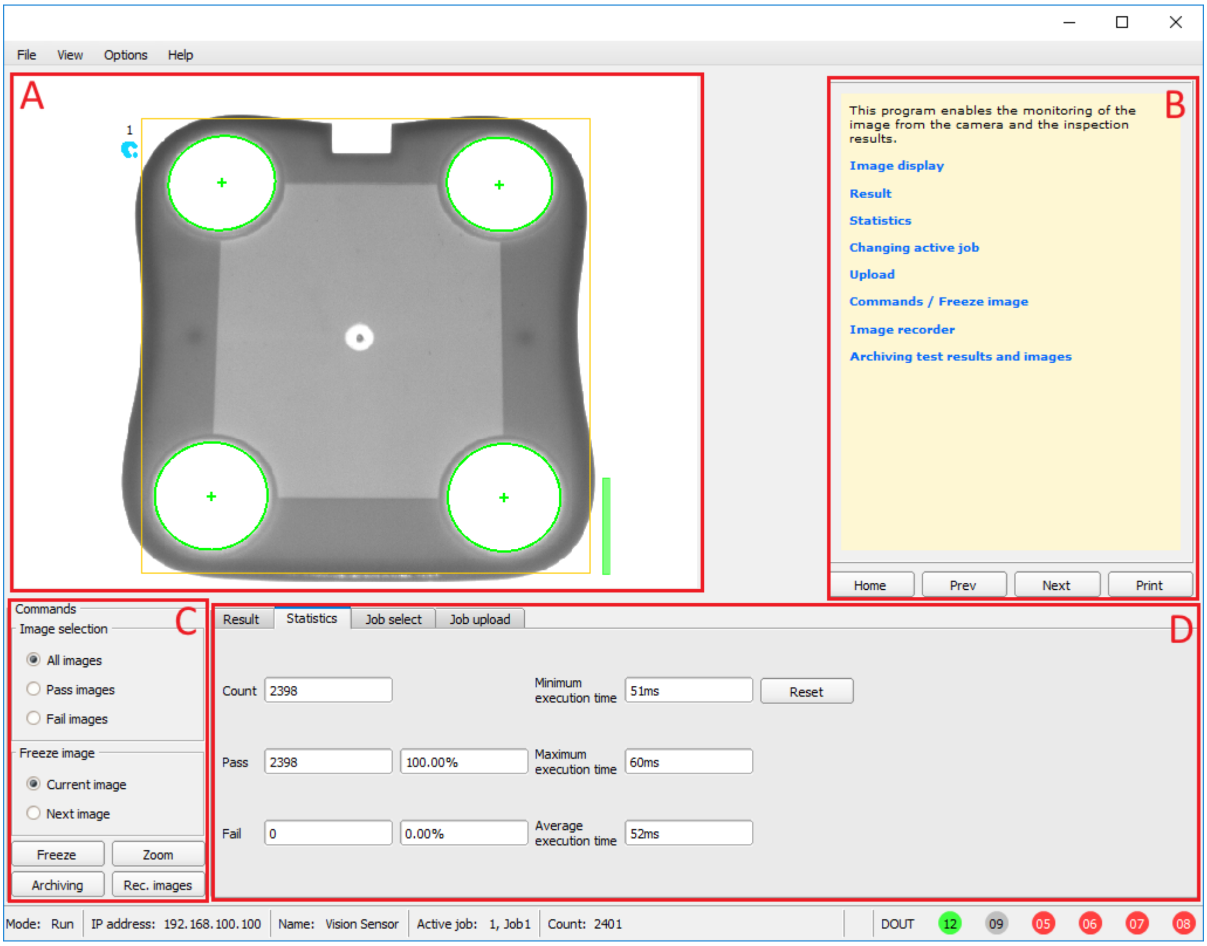
NOTE:
The settings in the "Interfaces", "I/O mapping" and "Signalling" tabs (indicated by the "globe" symbol) apply to the entire job set. Changes made in one job are applied to all other jobs.

6.4.2.5 Starting the sensor

When the sensor is started, all settings are transferred to the sensor, stored in the flash and executed there according to the settings made, e.g. in free run or triggered mode. All displays in the detector list, in the result field or under "Statistic", are updated here. With a click on "Start sensor" the transferred parameters are permanently stored and the corresponding hardware outputs are also set during execution.

6.5 SensoView

6.5.1 SensoView - Overview



This program enables the monitoring of the image from the camera and the inspection results.

Image display

Result

Statistics

Changing active job

Upload

Commands / Freeze image

Image recorder

Archiving test results and images

Commands

Image selection

● All images

○ Pass images

○ Fail images

Freeze image

● Current image

○ Next image

Count 2398

Minimum execution time 51ms

Pass 2398 100.00%

Maximum execution time 60ms

Fail 0 0.00%

Average execution time 52ms

Mode: Run | IP address: 192.168.100.100 | Name: Vision Sensor | Active job: 1, Job1 | Count: 2401

Fig. 23: SensoView Overview

A: Image display

B: Context help

Context-sensitive help for the current topic

C: Commands

Commands for displaying, transferring, and archiving images.

D: Job and result display

These tabs can display (statistical) results, switch jobs, and load jobs / job sets from SensoView to the sensor.

Additional information: VISOR® Software – SensoView

6.5.2 SensoView – Quick Start Guide

Display images and results

This program is used to monitor / check connected sensors, to analyze inspection results, as well as to archive inspection results and images.

After clicking on the button "**View**" in SensoFind the module SensoView starts.

The current image is displayed with overlays of the Alignment and detectors (if "Image Transfer = Active" is enabled in the configuration module under Job / General).

- The "**Result**" tab shows the individual detectors with their results and the overall result.
- The tab "**Statistic**" shows further statistical results.
- With "**Freeze image**", event-driven (e.g. bad part) images can be captured in the display.
- With "**Zoom**", displayed images can be enlarged.
- With "**Archive images**", images and result data can be archived on the hard disk of the connected PC, as previously set under File / "Configure archiving", with or without numerical result data.
- With "**Rec. images**", the image recorder can be read.
- In the "**Job**" tab, it is possible to switch between existing jobs on the sensor.
- In the "**Upload**" tab, other, previously defined jobs or entire job sets can be loaded from the viewer onto the sensor.

6.6 Context help

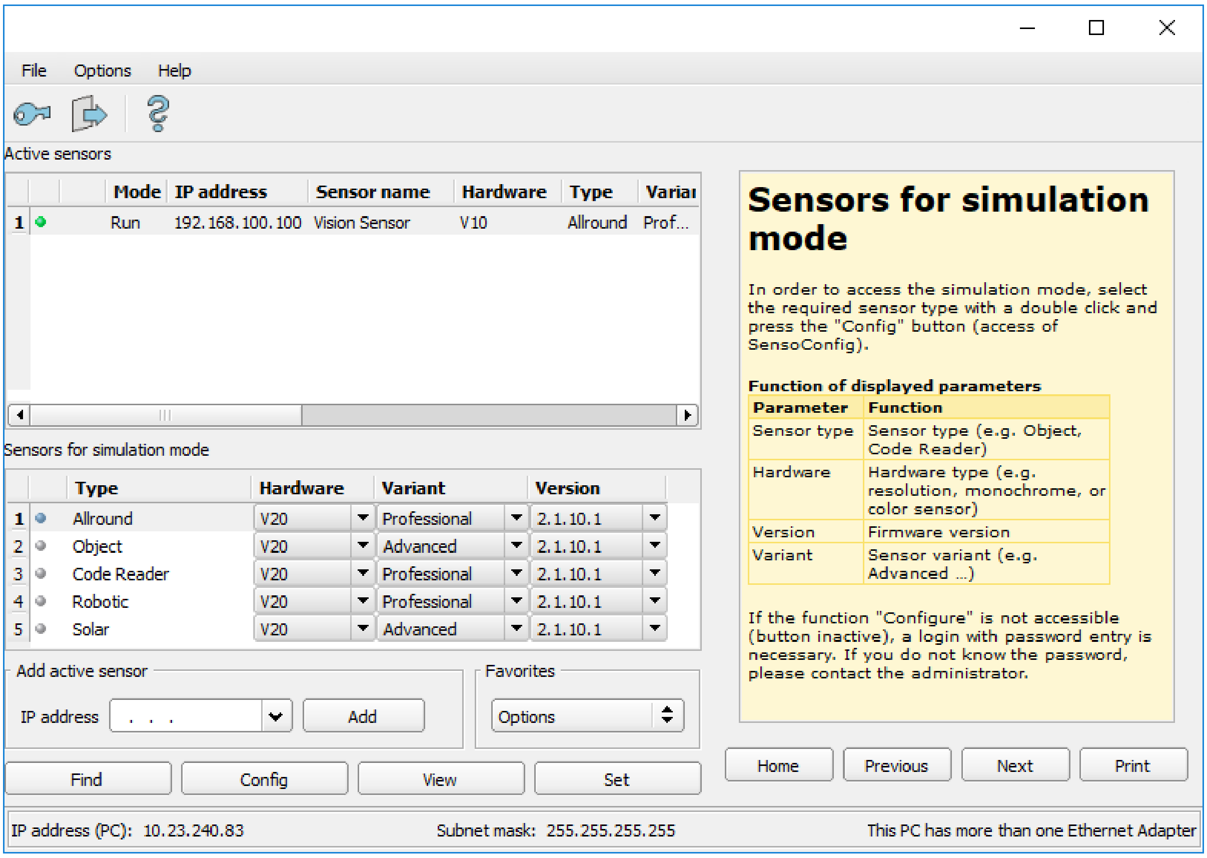
Context-sensitive help pages are available for all important program functions: As soon as you select a certain function on the program interface, you will receive the appropriate information in the help window at the top right (Help tab).

To view all available help pages, select "Help" from the menu or click the button with the "?" symbol or double-click in the context help window. There, you also can also search for terms or keywords. In comparison to the context-sensitive help, the size of this help window can be enlarged to view longer text more clearly.

Used open-source software: Open Source Licenses

7 VISOR® Software – SensoFind

In this program, you can select a sensor or a sensor simulation for configuration, or for display (monitoring), as well as various basic settings.



Active sensors

		Mode	IP address	Sensor name	Hardware	Type	Variai
1	●	Run	192.168.100.100	Vision Sensor	V10	Allround	Prof...

Sensors for simulation mode

		Type	Hardware	Variant	Version
1	●	Allround	V20	Professional	2.1.10.1
2	●	Object	V20	Advanced	2.1.10.1
3	●	Code Reader	V20	Professional	2.1.10.1
4	●	Robotic	V20	Professional	2.1.10.1
5	●	Solar	V20	Advanced	2.1.10.1

Add active sensor

IP address: . . .

Favorites

Sensors for simulation mode

In order to access the simulation mode, select the required sensor type with a double click and press the "Config" button (access of SensoConfig).

Function of displayed parameters

Parameter	Function
Sensor type	Sensor type (e.g. Object, Code Reader)
Hardware	Hardware type (e.g. resolution, monochrome, or color sensor)
Version	Firmware version
Variant	Sensor variant (e.g. Advanced ...)

If the function "Configure" is not accessible (button inactive), a login with password entry is necessary. If you do not know the password, please contact the administrator.

IP address (PC): 10.23.240.83 Subnet mask: 255.255.255.255 This PC has more than one Ethernet Adapter

Fig. 24: *SensoFind*

If the function "Config" is not accessible (button inactive), a login with password entry is necessary. Click on the button with the door symbol and arrow:

If you do not know the password, please contact the administrator.

7.1 Active sensors

All sensors available in the connected network are displayed in the drop-down list Active sensors.

Press the Ctrl key to select multiple individual sensors. With the Shift key you can select a whole continuous row of sensors. The following options are available for multiple selected sensors:

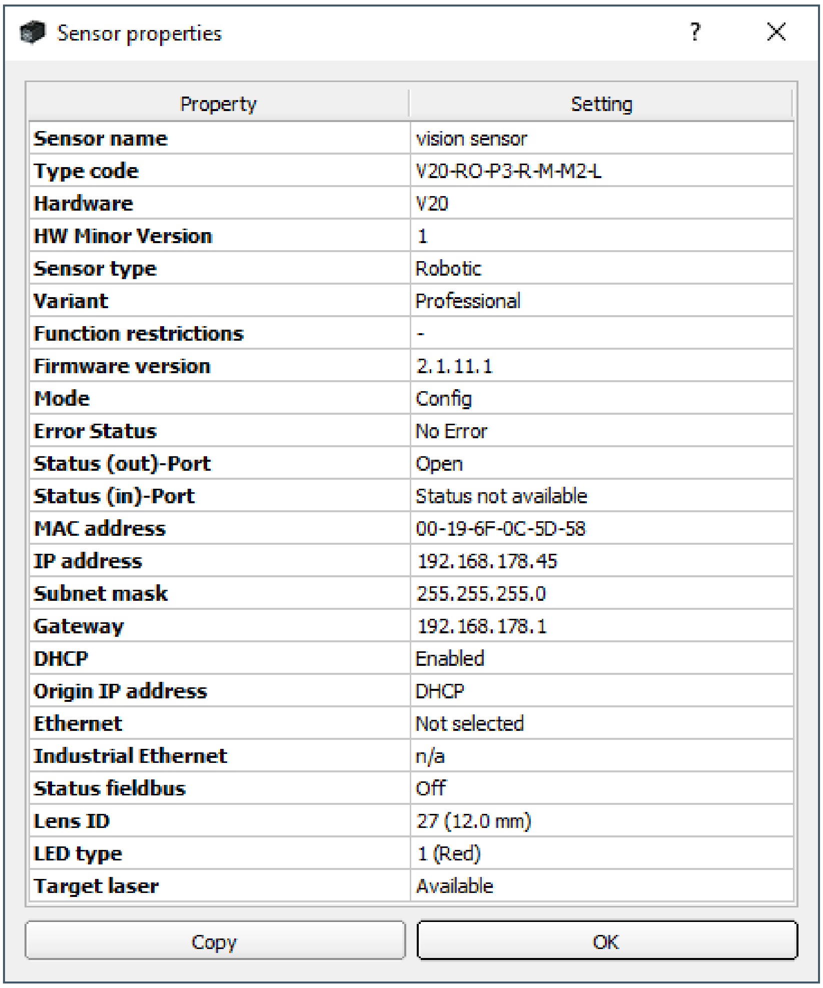
- File -> Backup sensor files to pc
- File -> Firmware update
- Web (if SensoWeb is activated)

In the first column, an LED indicates the operating mode of the VISOR®. Green: Device is in run mode, yellow: Device is in configuration mode, red: Error/device start

NOTE:

- If no entries are shown in the list, even though a sensor is connected, you can enter it using the "Find" button or the "Add" button.
- If no sensors are connected, the Sensors for simulation mode list will show available simulations for various sensor applications.

Clicking on the Details button (at the right end of the "Active sensors" parameter list) will open an even more detailed list of the VISOR® parameters.



Property	Setting
Sensor name	vision sensor
Type code	V20-RO-P3-R-M-M2-L
Hardware	V20
HW Minor Version	1
Sensor type	Robotic
Variant	Professional
Function restrictions	-
Firmware version	2.1.11.1
Mode	Config
Error Status	No Error
Status (out)-Port	Open
Status (in)-Port	Status not available
MAC address	00-19-6F-0C-5D-58
IP address	192.168.178.45
Subnet mask	255.255.255.0
Gateway	192.168.178.1
DHCP	Enabled
Origin IP address	DHCP
Ethernet	Not selected
Industrial Ethernet	n/a
Status fieldbus	Off
Lens ID	27 (12.0 mm)
LED type	1 (Red)
Target laser	Available

***Fig. 25:** Sensor properties*

Additional information:

Configuration of a connected sensor SensoConfig (Access of SensoConfig)

Display images and result data SensoView (Access of SensoView)

7.2 Sensors for simulation mode

In order to access the simulation mode, select the required sensor type with a double click or press the "Config" button (access of SensoConfig).

Function of displayed parameters

Parameter	Function
Sensor type	Sensor type (e.g. Object, Code reader,)
Hardware	Hardware type (e.g. resolution, monochrome, or color sensor)
Version	Firmware version
Variant	Sensor variant (e.g. Advanced …)

If the function "Config" is not accessible (button inactive), a login with password entry is necessary. If you do not know the password, please contact the administrator.

7.3 Add / find active sensor

If no entries are displayed in the "Active sensors" drop-down list, even though a sensor is connected, proceed as follows:

Search Find / sensor:

To search for sensors which are connected to the PC, or which are available in the network, click the button "Find".

Add active sensor:

If you know the IP-address of a sensor, please enter it in the field IP-address and click the button "Add".

Now the sensor appears in the list and can be configured (button "Config") or displayed (button "View").

If the function "Config" is not available (button not active / grayed out), a login with password input is necessary. If you do not know the password, please contact your system administrator.

7.4 Favorites

The favorites are used for quick access and management of the VISOR® vision sensors. The following parameters can be selected for the favorites (in the "Favorites" area in SensoFind or in support mode via right-click).

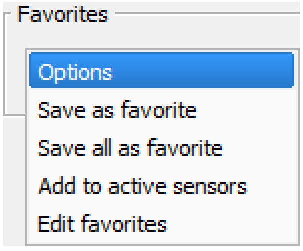
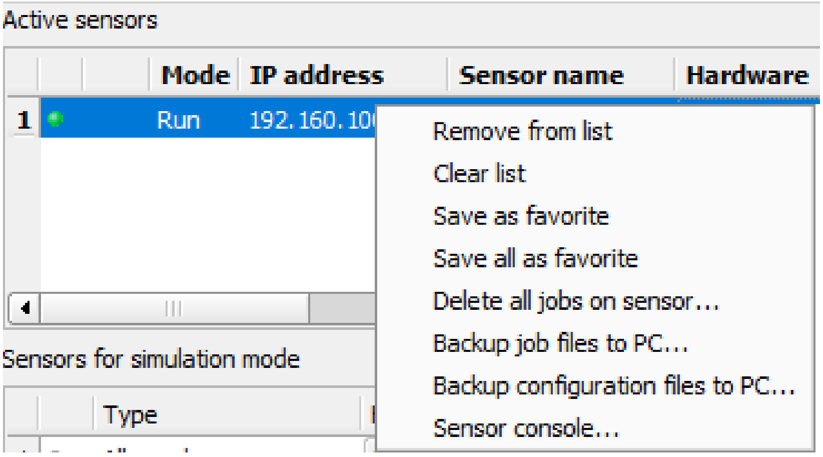


Fig. 26: *Favorites options*



Active sensors

	Mode	IP address	Sensor name	Hardware
1	Run	192.160.10		

Sensors for simulation mode

Fig. 27: *Parameter "Favorites" when right-clicking on an active sensor (only available in support mode)*

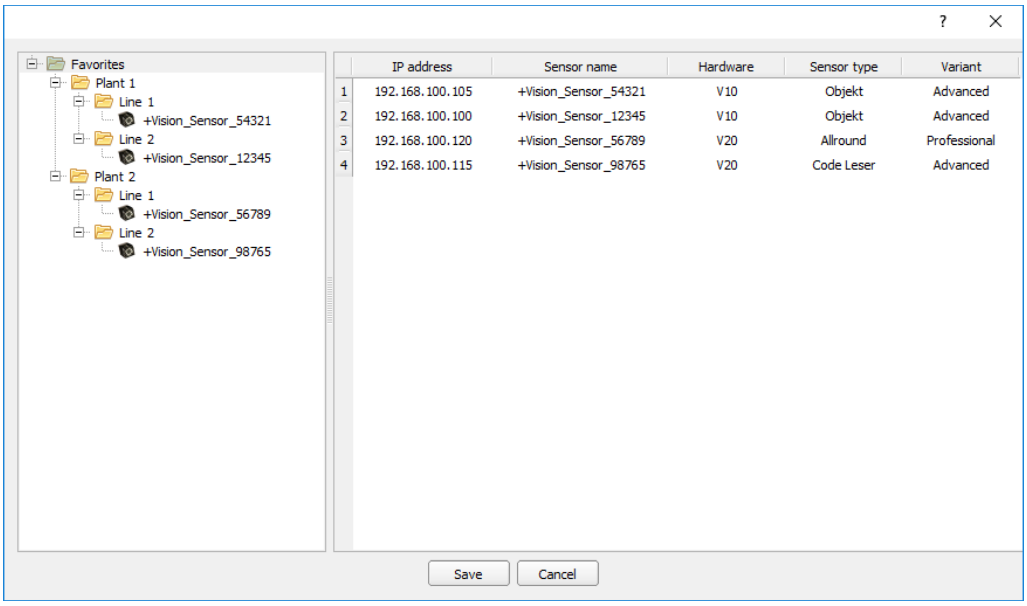
Parameter description:

Parameter	Function
Save as favorite	Opens the "Save as favorite" window where a desired destination can be selected in the tree structure in which the sensor from the "Active sensors" list should be saved as a favorite.
Save all as favorite	Opens the "Save all as favorite" window where a desired destination can be selected in the tree structure in which all the sensors from the "Active sensors" list should be saved as a favorite.
Add to active sensors	Opens the "Add to active sensors" window where a sensor/ sensor group can be selected that should be added to the "Active sensors" list.
Edit favorites	Opens the "Edit favorites" window in which the sensor groups can be managed / edited.

Edit favorites - create groups

In the left window area, the sensors are divided into groups via a tree structure, e.g. according to production sites and production lines. In the right window area, the sensors below a selected group are listed in tabular form, e.g. the group "Favorites" shows all sensors.

To create a group, right click on "Favorites" or an existing group / "Add group".



- Favorites
  - Plant 1
    - Line 1
      - +Vision_Sensor_54321
    - Line 2
      - +Vision_Sensor_12345
  - Plant 2
    - Line 1
      - +Vision_Sensor_56789
    - Line 2
      - +Vision_Sensor_98765

	IP address	Sensor name	Hardware	Sensor type	Variant
1	192.168.100.105	+Vision_Sensor_54321	V10	Objekt	Advanced
2	192.168.100.100	+Vision_Sensor_12345	V10	Objekt	Advanced
3	192.168.100.120	+Vision_Sensor_56789	V20	Allround	Professional
4	192.168.100.115	+Vision_Sensor_98765	V20	Code Leser	Advanced

Fig. 28: Group configuration

Favorites are saved as an XML file in the installation path for the VISOR® vision sensor on the PC. The file is located in the directory "SensoPart/VISOR Vision Sensor/SensoFind/Data". An exchanged between different PCs can take place.

Examples for using favorites:

Example 1:

VISOR® vision sensors that are integrated into various networks can be viewed and managed locally in SensoFind (please refer to the following figure as well). The sensors can be added to the "Active sensors" list by entering the IP address in the field "Add active sensor". The sensors are subsequently managed via the favorites. The sensors can be added to favorites by "SensoFind/Favorites/Save as favorite". Within the favorites, the sensors can be assigned to different groups.

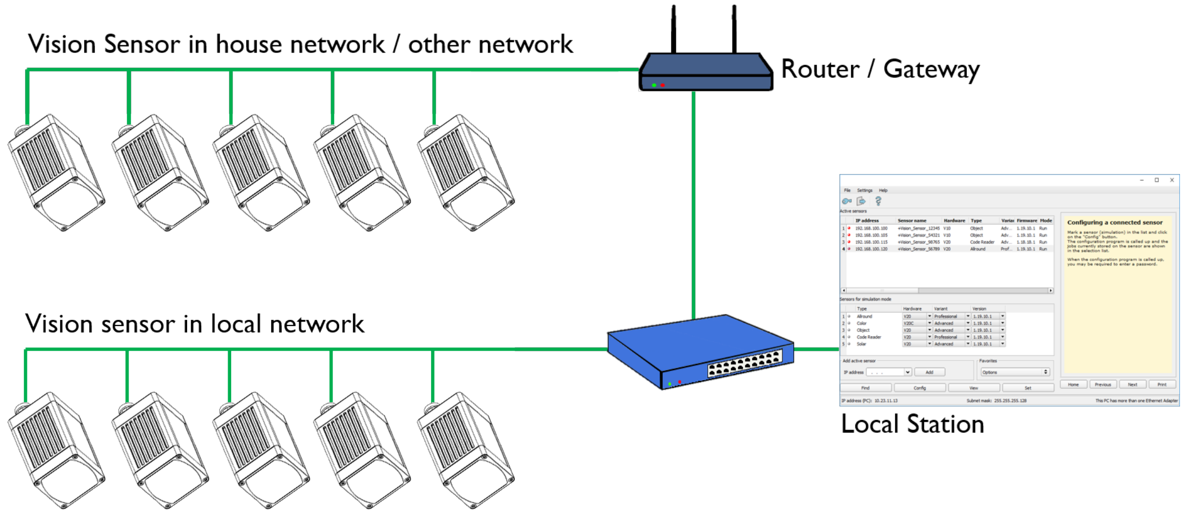

Vision Sensor in house network / other network
Router / Gateway
Vision sensor in local network
Local Station

Fig. 29: Example 1 - VISOR® on various networks

Example 2:

Multiple stations are in the same local network. All users have access to all VISOR® vision sensors, even though they only need some of these for their work (please refer to the following figure as well). When using the autostart function (please refer to Autostart file (file) as well), you can choose for only a specific selection of VISOR® vision sensors (favorites) to be shown. The sensors must therefore be added to the favorites and divided into groups. A group of favorites can then be selected in the Autostart file. Users will then only have access to the relevant sensors when opening SensoFind via the Autostart file.

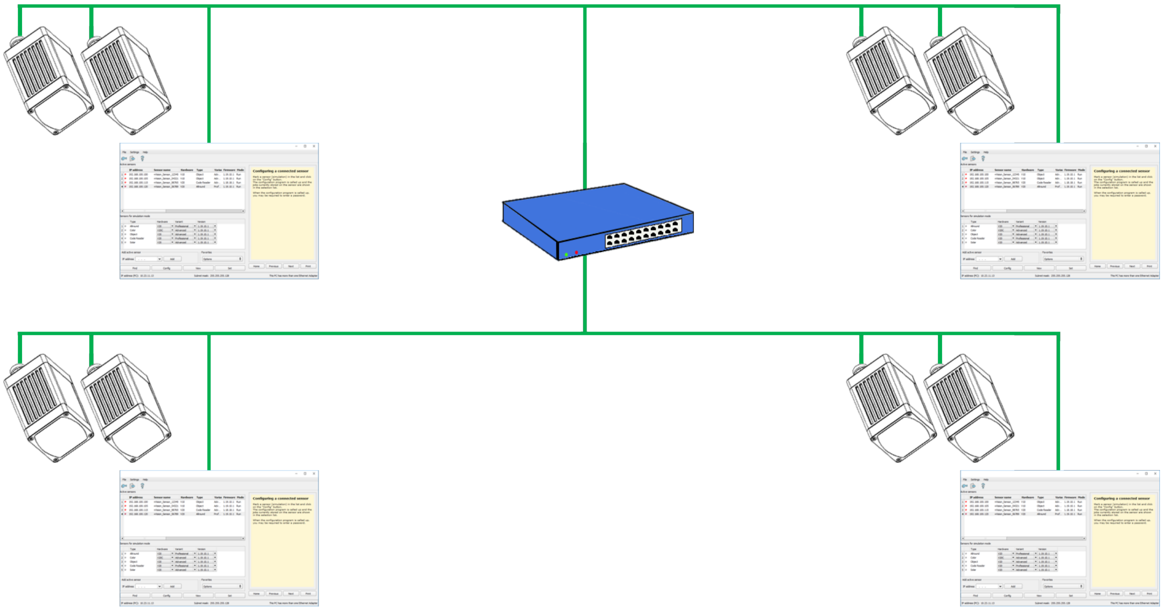
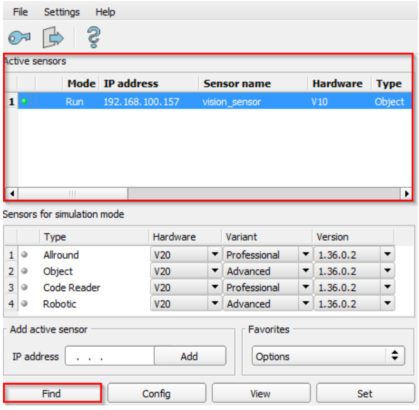


__Fig. 30:__ Example 2 - Favorites in the Autostart file

7.5 Configuration of a connected sensor

To configure the VISOR® vision sensor, proceed as follows.

1. Start the VISOR® software. SensoFind is opened.



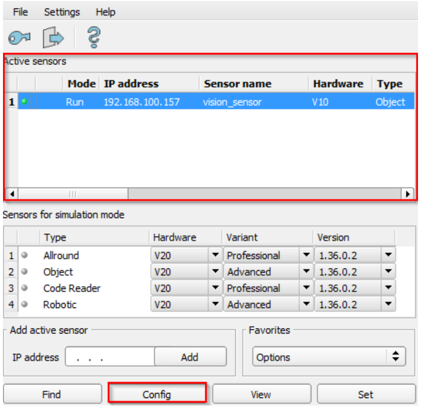
Active sensors

	Mode	IP address	Sensor name	Hardware	Type
1 ●	Run	192. 168. 100. 157	vision_sensor	V10	Object

Sensors for simulation mode

	Type	Hardware	Variant	Version
1 ●	Allround	V20	Professional	1.36.0.2
2 ●	Object	V20	Advanced	1.36.0.2
3 ●	Code Reader	V20	Professional	1.36.0.2
4 ●	Robotic	V20	Advanced	1.36.0.2

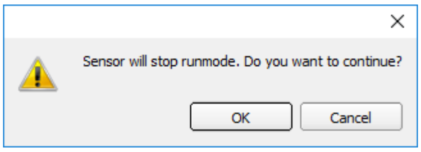
2. Click on the "Find" button. The vision sensor will be listed in the "Active sensors" window.



	Mode	IP address	Sensor name	Hardware	Type
1	Run	192.168.100.157	vision_sensor	V10	Object

	Type	Hardware	Variant	Version
1	Allround	V20	Professional	1.36.0.2
2	Object	V20	Advanced	1.36.0.2
3	Code Reader	V20	Professional	1.36.0.2
4	Robotic	V20	Advanced	1.36.0.2

3. Click on a sensor in the "Active Sensors" list.
4. Click the "Config" button to open SensoConfig.


Sensor will stop runmode. Do you want to continue?

5. Confirm the following dialog box with "OK" to stop SensoFind and start the configuration in SensoConfig.

   NOTE:
   When SensoConfig is accessed, you may be required to enter a password.
   See User administration / Passwords (file) for defining passwords.

Additional information: VISOR® Software – SensoConfig

7.6 Display images and result data

Select a sensor in the list and click on the " View " button. The SensoView program is started and images and measurement results from the active job are displayed. SensoView is not available for sensors in simulation mode.

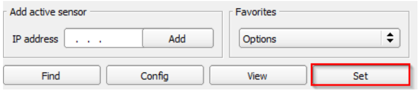
NOTE:
Accessing SensoView does not affect the operation of the selected sensor.
See Chapter: VISOR® Software – SensoView

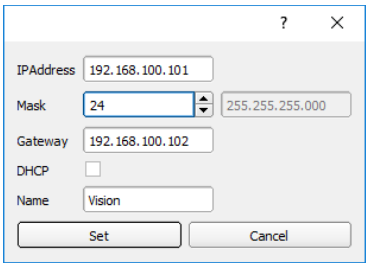
7.7 Network settings of the sensor

You can change the network settings of the selected sensor with the "Set" button. The IP address, subnet mask, standard gateway, DHCP, and sensor name can be set here. The PC's IP address and subnet mask are displayed below in the SensoFind status bar.

To connect the sensor to the PC, the address spaces must match. If necessary, set the IP address, etc. of the sensor accordingly here. Please contact your administrator to set network parameters. For more information, please refer to sections Network connection and Network settings.

If "DHCP = active" is selected, a name must be assigned for the sensor, since the IP address can then be reassigned every time the sensor is started and can thus change, i.e. is no longer unique. You require administrator rights for these functions (see User administration).





IPAddress: 192.168.100.101
Mask: 24 | 255.255.255.000
Gateway: 192.168.100.102
DHCP: □
Name: Vision

1. Click on a sensor in the "Active Sensors" list.
2. Click on the "Set" button. The dialog box for configuring the IP address and the sensor name will appear.
3. Assign an IP address and a name to the sensor.
4. Click on the "Set" button. The IP address and the name have now been updated.

7.8 User administration / Passwords (file)

The VISOR®configuration software distinguishes between three user groups with different authorizations: (button in the upper left corner with key symbol)

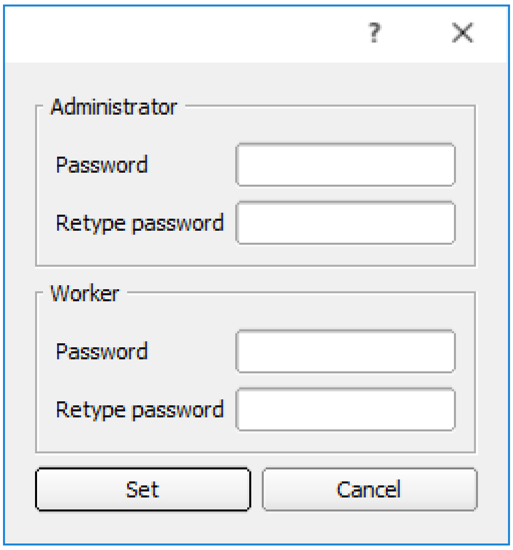


__Fig. 31:__ SensoFind, Passwords

Password level	SensoFind	SensoConfig	SensoView
Administrator Password	All functions	All functions	All functions
Worker Password	All functions except • Configuration • Settings • Update	None	All functions, including Job Upload and Image Recorder
User (without password)	All functions except • Configurating • Settings • Update	None	Only display of images, inspection results, and statistics

After installing the software, the login is automatically executed immediately when the application is called without password prompt. No passwords are assigned.

Define passwords

Select User Administration in the File menu or click on the button with the key symbol in the toolbar to assign or change passwords for the user categories Administrator and Worker. Once a password has been entered, a logout is automatically carried out, i.e. input of the new password is now necessary. When assigning an "empty" password, the entry can be simply confirmed with OK.

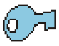


***Fig. 32:** Password button*

Login / Logout

After setting passwords, login is necessary, e.g. for configuring a sensor. To do this, click on the login button in the toolbar, enter the specified password, and confirm with "OK". If the checkmark is set to "Disable password query", the password will no longer be requested the next time the application is started. To log out of the user group, click on the logout button.

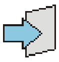


***Fig. 33:** Login button*

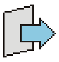


***Fig. 34:** Logout button*

7.9 Firmware update (file)

You can update the firmware of the selected sensors through the menu item "SensoFind/File/Firmware Update" (see following figure). For this, the corresponding firmware update file must first be obtained from the SensoPart homepage or from SensoPart support.

In the dialog that is opened, select the appropriate firmware file and follow the instructions. Do not disconnect the power to the sensors during this process unless prompted to do so by a screen instruction.

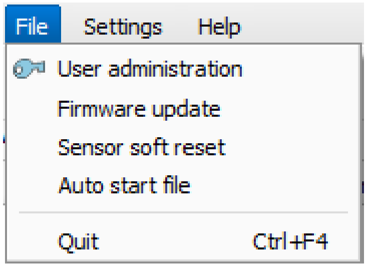


***Fig. 35:** SensoFind, firmware update*

⚠ **ATTENTION:**
Before executing the firmware update, please create a current backup!
To do this, save the jobsets via the menu item "SensoConfig/File/Save jobset as..."

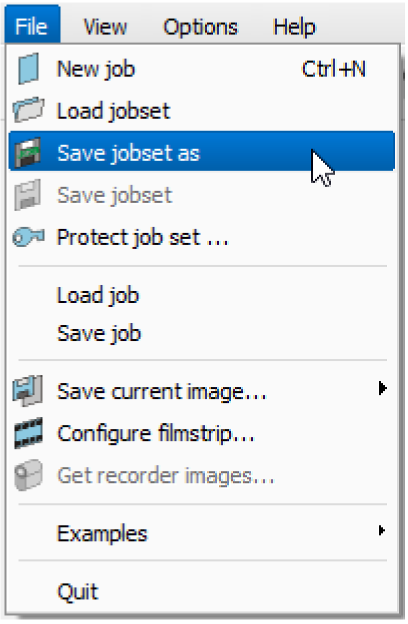


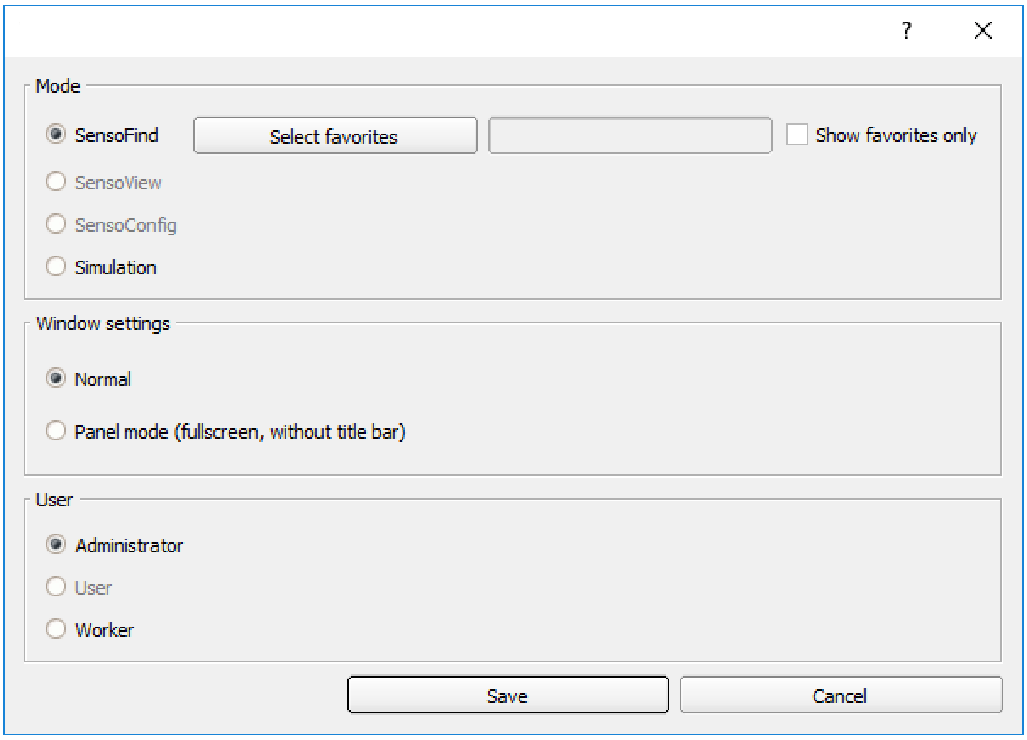
***Fig. 36:** Create backup, Save job set under*

7.10 Autostart file (file)

Autostart makes it possible to start the VISOR® software software automatically. For this purpose, a batch file is created which can be stored in the Windows system folder "Autostart" so that it can be accessed automatically every time the PC is started. The Autostart window is divided into the areas: mode, window settings, and user.

Procedure

1. Open the Autostart file in the SensoFind module with the file path: SensoFind/File/Autostart File
2. You can define which VISOR® software software module should be automatically started in the "Mode" section.
3. In the window settings, select the view of the module: Normal or full screen without title bar in the panel PC mode.
4. In the "User" area, the user for the Autostart file is specified. For more information on user role permissions, please refer to User administration / Passwords (file)
5. Select the "Save" button and save the batch file (.bat) to the desired destination. For an automatic start when the PC boots, the file must be stored in the Windows system folder "Startup".
6. Close the VISOR® software.
7. Execute the batch file. The VISOR® software software will start as specified in the configured settings.



Mode

● SensoFind □ Show favorites only

○ SensoView

○ SensoConfig

○ Simulation

Window settings

● Normal

○ Panel mode (fullscreen, without title bar)

User

● Administrator

○ User

○ Worker

__Fig. 37:__ Autostart file

The following parameters can be set in the Autostart window:

Mode	
Parameter	**Function**
SensoFind	VISOR® software software modules that should be opened automatically in the autostart file. For the start of the simulation mode, the model variant which is currently selected in SensoFind (marked in blue) is used.
SensoView	
SensoConfig	
SensoWeb	
Simulation	
Select favorites	This parameter can be used to add a favorites group to the Autostart file.
Show favorites only	If the "Select favorite list" parameter is selected, the "Active sensors" list will be emptied and then only filled with the selected favorites.

Window settings	
Parameter	**Function**
normal	In the autostart file, the selected VISOR® software module will be opened normally with the title bar.
Panel PC mode (Fullscreen without title bar)	In the autostart file, the selected VISOR® software module will be opened in full screen mode without a title bar. Typical application for touchscreen panel PCs.

User	
Parameter	**Function**
Administrator	The selection of the user depends on the rights that the user will have within the Autostart file. For more information on user role permissions, please refer to User administration / Passwords (file)
User	
Worker	

7.11 SensoFind Support Mode

In support mode, advanced functions are available and can be accessed via context menu.

1. To enter the support mode, open the installation directory of the VISOR® software (right click on the desktop shortcut of the VISOR® software / "Open file path").
2. Open the "SensoFind-support" shortcut.

In support mode, the following options are available by right-clicking on an active sensor or in the context menu:

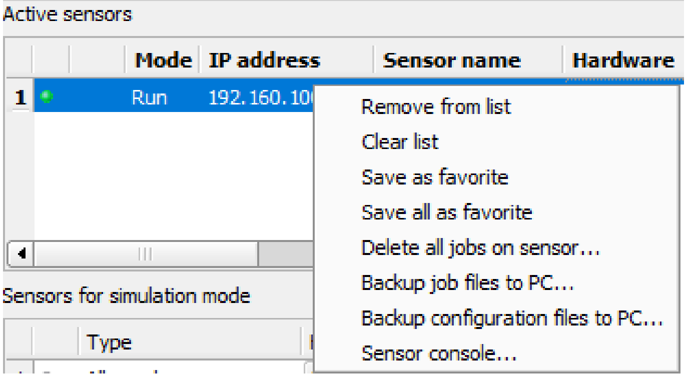


Fig. 38: *Right-click on Active sensor*

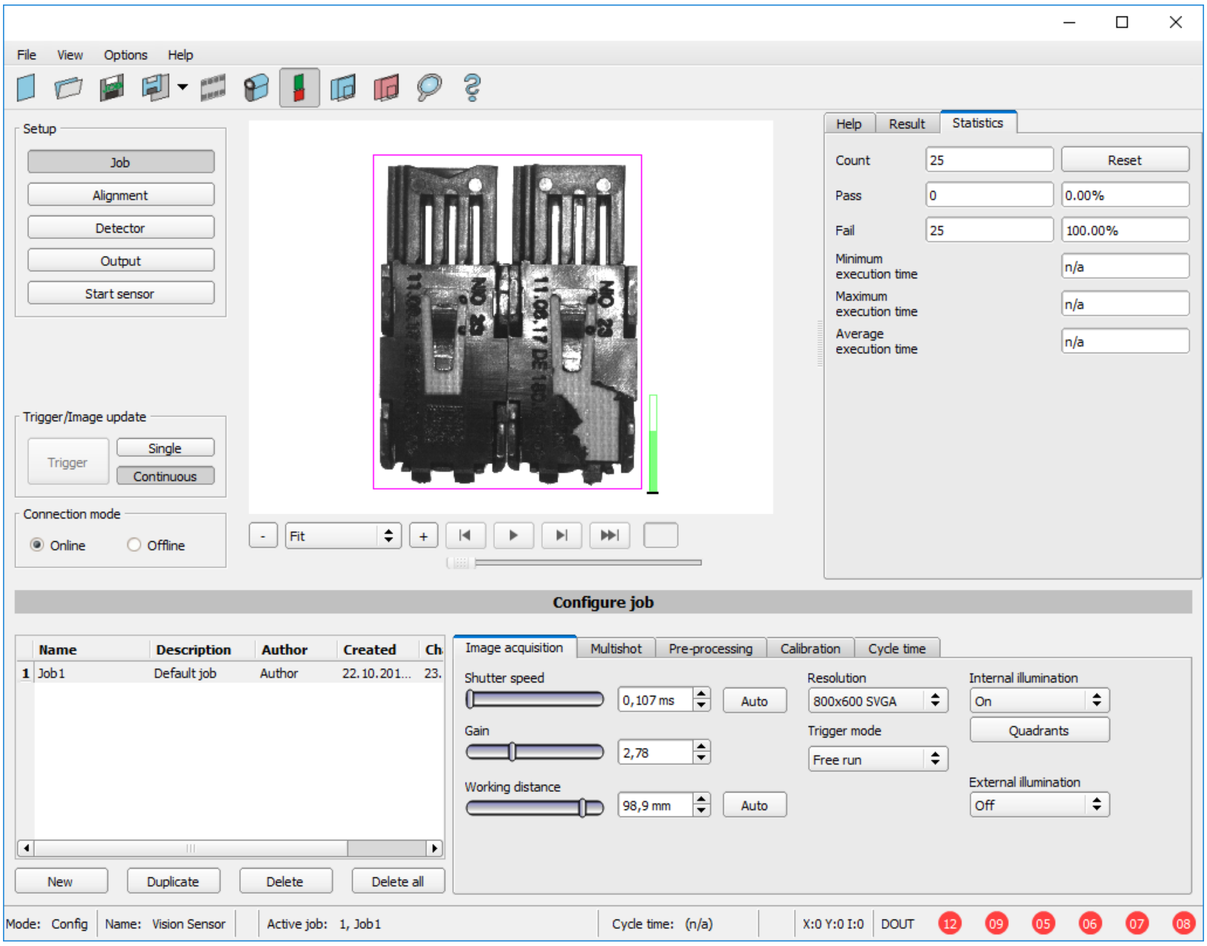
Parameter	Function
Remove from list	Removes the selected sensor from the "Active sensors" list.
Clear list	Clears the complete list "Active sensors".
Save as favorite	Saves the selected sensor or all sensors in the list as favorite(s). Additional information: Favorites
Save all as favorite	
Delete all jobs on sensor...	Deletes all jobs on the sensor. Information cannot be recovered. A restart is required after deletion.
Backup job files to PC...	Saves job files on the PC in the specified directory.
Backup configuration files to PC...	Saves the configuration files on the PC in the specified directory.
Sensor console...	Opens a console (ssh) to the selected device.

8 VISOR® Software – SensoConfig

You can use this program to configure your vision sensor in five setup steps for one or more inspection tasks (jobs):

8.1 Setup Job (Inspection tasks)

A job contains all the settings and parameters required to perform a specific inspection task.



Connection mode: ● Online ○ Offline

Count	25
Pass	0 0.00%
Fail	25 100.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Name	Description	Author	Created	Ch
1	Job1	Default job	Author	22.10.201...	23.

Shutter speed: 0,107 ms

Gain: 2,78

Working distance: 98,9 mm

Resolution: 800x600 SVGA

Trigger mode: Free run

Internal illumination: On

External illumination: Off

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 39: SensoConfig, setup step Job

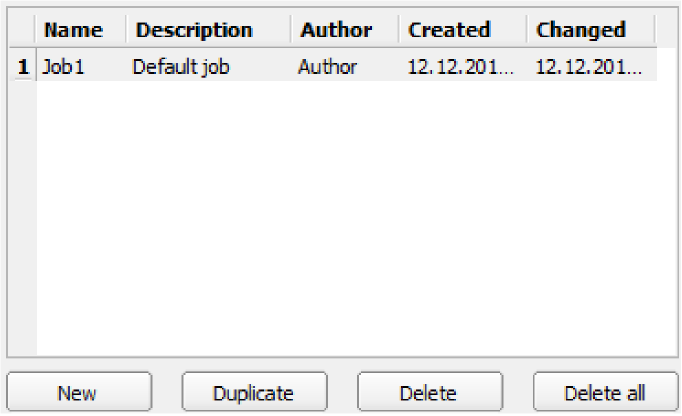
8.1.1 Creation, modification, and administration of jobs

You can edit a selected job (marked in the list on the bottom left) by entering parameters in the tabs of the configuration window (right, bottom).

If there is no job entry in the list, you must create a new job first.

Creating a new job

1. Click on the button "New" underneath the job drop-down list. A new job entry appears in the list.
2. Edit the entry with a double click on the respective line (Name, Description, Author)



	Name	Description	Author	Created	Changed
1	Job1	Default job	Author	12.12.201...	12.12.201...

Fig. 40: SensoConfig Job list

Further functions

Function	Description
New	Defines a new job
Duplicate	Adds a copy of the selected job to the job set.
Delete	Deletes the selected job from the list
Delete all	Deletes all the jobs in the list

If the capacity of the sensor memory is exhausted and no further jobs can be loaded onto the sensor, the color of the remaining memory indicator in the status line (below) changes to red.

Job templates

It is possible to save a job as a template. To do this, right-click on the job in the job list and select "Save as template". For each new job, the settings and detectors are then copied from the job template. In the job list, the job template is identified with a "T" (Template). The job template cannot be edited. To remove the job template, right-click on the template and select "Remove".

When creating new jobs from the job template, the job set parameters are not changed.

Copy job parameters

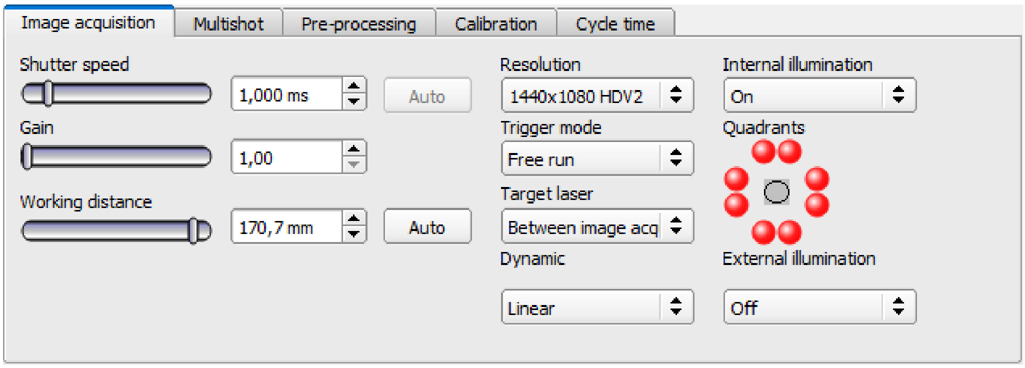
By right clicking on a job, job parameters such as Image acquisition or calibration settings can be copied to another job. The respective parameters and the target jobs can be selected in the dialog.

Additional information:

Open and save job or jobset (file)

Protect jobset (file)

8.1.2 Image acquisition tab



Shutter speed: 1,000 ms

Gain: 1,00

Working distance: 170,7 mm

Resolution: 1440x1080 HDV2

Trigger mode: Free run

Target laser: Between image acq

Dynamic: Linear

Internal illumination: On

External illumination: Off

Fig. 41: *Image acquisition tab*

In the Image acquisition tab, you define the basic parameters of image acquisition.

Parameter	Functions and setting possibilities
Shutter speed	Parameters for controlling the image brightness. Image brightness should preferably be set with the shutter speed. Only in the second step, if necessary, adjust the gain (default gain = 1). With moving objects, a slower shutter speed can cause motion blur in the image. Auto: With the button "Auto" the exposure can be set automatically. The maximum shutter speed that can be configured is 100 ms. The duration of the internal exposure pulse is limited to 8 s. Shutter speeds longer than 8 ms only make sense in cases in which external lighting (or both internal and external lighting) is used.
Gain	Parameters for controling the image brightness. The image brightness should preferably be adjusted with the shutter speed; only adjust the gain in the second step if necessary (default gain = 1)
Working distance	Parameters for configuring the working distance. Auto: With the button "Auto" the approximate working distance can be set automatically. Fine adjustment is possible using the slider or value adjustment (Additional information: Focusing / Working distance).
Resolution	Available resolutions: V10 / V10C: SVGA (800 x 600), QSVGA (400 x 300), QSVGA Zoom 2 (400 x 300) V20 / V20C: HDV2 (1440 x 1080), WGA (720 x 540), WGA Zoom 2 (720 x 540) V50 / V50C: QSXGA (2560 x 1936), SXVGA (1280 x 968), SXVGA Zoom 2 (1280 x 968) For time-critical applications or for compatibility reasons, a lower resolution can be selected. Attention: If the resolution is changed, all detectors that have already been defined are deleted!
Zoom	By selecting a resolution level with zoom, different image sections with different image sizes can be achieved.
Trigger mode	Selection option that can be used to define whether the vision sensor should be operated in trigger mode or in free run mode. Trigger: In the triggered mode, the pin 03 WH trigger input or one of the interfaces can be used to trigger an image acquisition. Free run: In free run mode, the vision sensor will continuously capture images and run evaluations.

Parameter	Functions and setting possibilities
Target laser	The target laser is used to align the sensor. Target laser options: Off / During image acquisition / Between image acquisition operations
Internal illumination	Switch for internal illumination (on / off) The internal illumination is limited to 50 ms. If a longer shutter speed is set, the internal illumination switches off at 50 ms.
Quadrants	By clicking on the LED quadrants, individual quadrants of the illumination can be switched on / off (shown as two red dots; perspective: view onto the VISOR® vision sensor). This function can suppress reflections at low working distances.
External illumination	External lighting options: Off / On / Permanent. The external lighting is switched using pin 09 RD.

In order to get a continuously updated live image without triggers, configure the following settings as shown:

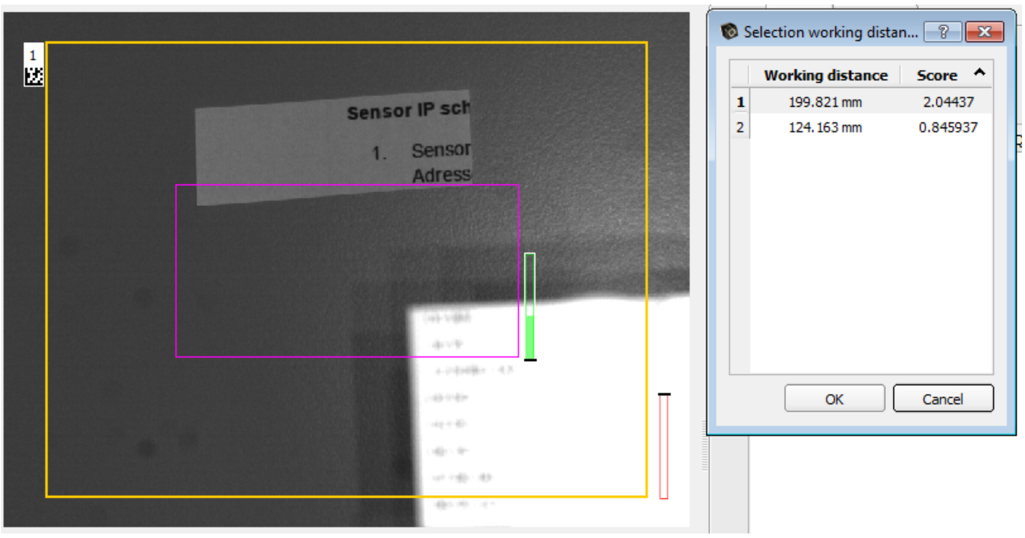
1. In the Job setup step, open the Image acquisition tab and set the Trigger mode to Free run.

2. Under Trigger / Image update, select "Continuous".

Focusing / Working distance

The parameter is Working distanceused to set the working distance at which the image is focused. You can use the slider, or edit the values, to do a fine adjustment.

Parameter	Function
Auto	With the button "Auto" the approximate working distance can be determined automatically. The violet region of interest is used for the determination (see Working distance). If several possible sharp layers are found in the region of interest, the dialog "Layer selection list" appears. The corresponding working distance can be selected here. The value "Score" indicates a measure for the sharpness of the image (greater = sharper). The corresponding working distance is accepted by clicking on it.



	Working distance	Score
1	199.821 mm	2.04437
2	124.163 mm	0.845937

Fig. 42: Working distance

Different jobs can be set to different working distances. The time required for the job change can be extended by up to 2 seconds by moving to the working distances. Approx. 1 job change per minute is possible.

8.1.3 Multishot tab

NOTE:
The Multishot function is available only in Allround and Professional versions and not for color sensors.

When using the Multishot function, each sequence involves acquiring 4 images of an object. Each time an image is taken, the object is illuminated from a different direction. The four images are then combined into one image. Due to the different reflections, a "virtual height image" can be calculated which contains information that is not visible in the individual images. This way, the finest depressions or elevations in the considered surface can be detected.

This technology is especially suitable for:

- Detecting defects on flat surfaces e.g. scrapes or scratches
- Reading raised or imprinted fonts using OCR
- Nailed Data matrix codes
- Detection of details on high-resolution surfaces
- Detection of Braille dots

This technology is not suitable for:

- Imaging of moving objects
- Heavily curved surfaces
- 3D applications
- Detection of details that are shaded by other parts of the component and thus cannot be illuminated from all 4 sides.

The following **conditions** must be considered:

1. Use ring light and place object in its focus. If no ring light is used, place the light source as far away from the measurement object as possible.
2. Optimize image acquisition: Avoid overexposed or dark areas, shadows and blurred areas in all four images.
3. The test object must be still during the image acquisition operations (four images in a single sequence), i.e., it must be stationary relative to the sensor. Exception: movement of constant speed in x-direction (see Image offset X axis).

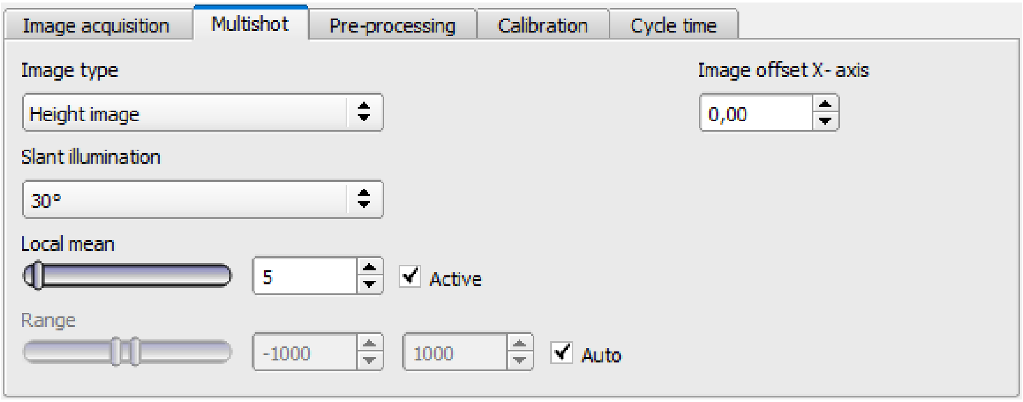


Image type: Height image

Slant illumination: 30°

Local mean: 5 ☑ Active

Range: -1000 1000 ☑ Auto

Image offset X- axis: 0,00

Fig. 43: "Multishot" function, parameters

8.1.3.1 Image types

Image type	Description
Image curvature	Shows virtual (estimated) curvature values (positive and negative) mapped to gray values. The curvature values indicate how much the tilt of the surface changes at a given point.
Curvature, amount	Shows virtual (estimated) curvature values, but here only the absolute amount, mapped to gray values.
Height image	Shows virtual (estimated) height values, scaled to gray values. ℹ **NOTE:** Selecting "Height image" causes longer execution time.
Albedo image	Shows virtual (estimated) reflectivity values mapped to gray values.
Mean image	Mean of four single images
Combined image, quadrants	All four single images combined in one quadrant image. This function can be used to adjust the lighting. Avoid overexposed or dark areas, shadows and blurred areas.
Combined image, horizontal	All four single images arranged horizontally on top of each other combined in one image. If the test object was moved uniformly during image acquisition, the position offset of the individual images in the X-direction can be made visible and compensated with the function "Image offset X-axis". The image order from top to bottom is: East, West, North, South.
East	Single image, illumination from east
North	Single image, illumination from north
West	Single image, illumination from west
South	Single image, illumination from south

Image type parameters

Parameter	Function
Image type	Select Image type (see above)
Slant illumination	Angle of illumination with respect to the surface of the object (0 ° = flat from the side; 90 ° = perpendicular from above)
Background flattening	If the sensor is not exactly perpendicular to the object or the illumination is inhomogeneous, the calculated height image may appear strongly tilted. The local smoothing of the height differences helps to correct the tilting. The smoothing is carried out via the set number of pixels.
Range	Value range of virtual height and angle values. The set range is mapped to a gray value image 0...255. With this range selection it is possible to obtain an optimal gray value spread of the area of interest. With "Auto", this value is automatically calculated from the minimum and maximum values found in the image.
Image offset X axis	Generally, the object should rest during the 4 image acquisitions. However, a movement of constant speed in x-direction can be compensated by the parameter "Image offset X-axis". This parameter specifies by how many pixels the object is shifted in the X direction in the successive images. If the inspection object was moved uniformly during image acquisition, the position offset of the individual images can be compensated for one another together with the image type "Combined image, horizontal".

8.1.3.2 Multishot Illumination

Correct illumination is important for use. The object must be illuminated from all four sides. The VISOR® sensor automatically controls the image acquisition sequence. For ease of illustration, the four illumination directions are referred to below analogously to the four directions (north top).

The result is the following arrangement of illumination:

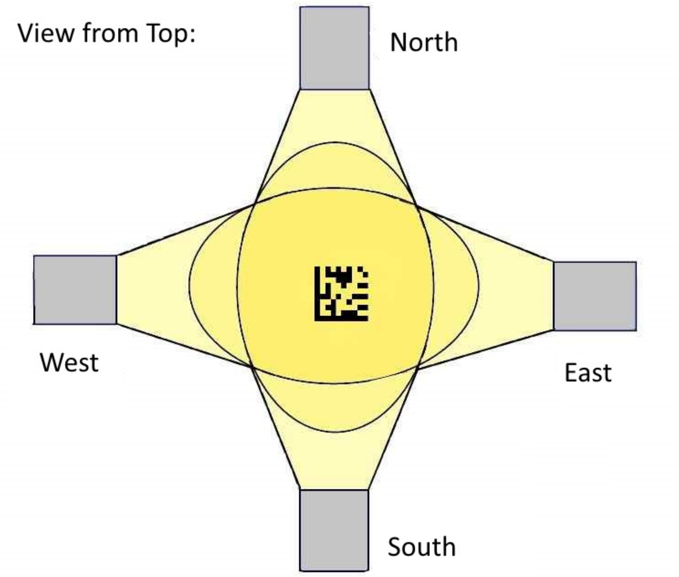


Fig. 44: *Multishot, Orientation of illumination*

Assignment of the illuminations to the VISOR® connections:

Direction	Output pin (old)	Output pin (new)
East	09	09
South	07	06
West	06	07
North	05	08

The correct connection of the illumination can be checked by using the function "Combined image, quadrants", which displays all 4 single images together in one image. Place an object in the image that casts a clear shadow (e.g. a screw). The images are combined according to the following scheme:

Top left: Illumination from north, shadow to south	**Top right:** Illumination from east, shadow to west
Bottom left: Illumination from west, shadow to east	**Bottom right:** Illumination from south, shadow to north

In the image it looks like this:

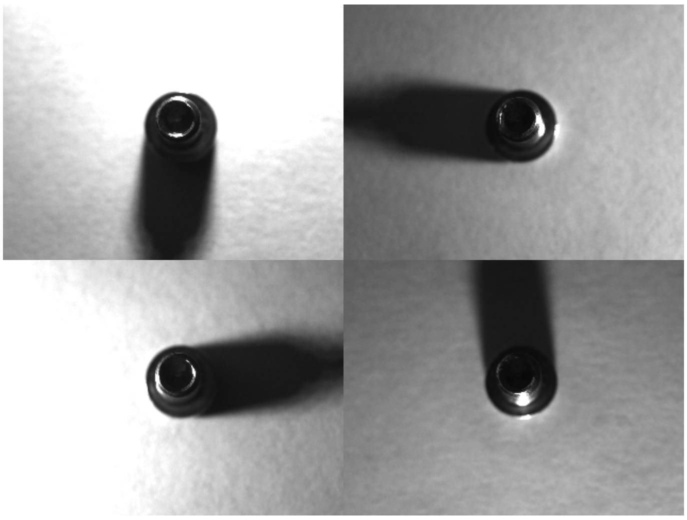


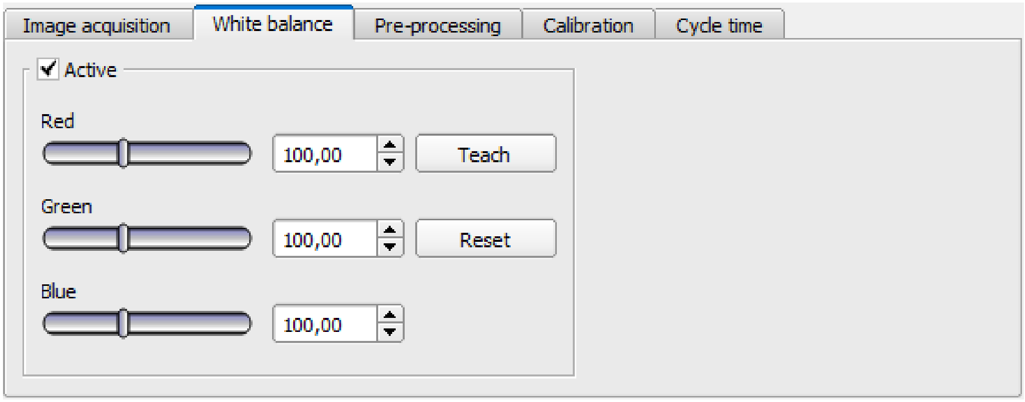
Fig. 45: Multishot, Single images

Further advice for illumination:

- Avoid both overdriven areas of the image and dark shadows
- The SensoPart illumination can be mounted in 30° or 60° angles
- Use 30° angle to illuminate parts flat from the side (avoids reflections)
- Use the 60 ° angle to illuminate parts steeply from above (amplified reflections)

8.1.4 White balance tab

White balance is used to compensate for a possible color cast in the image due to lighting conditions or camera chip. The White balance tab is only available for color sensors.



☑ Active

Red: 100,00

Green: 100,00

Blue: 100,00

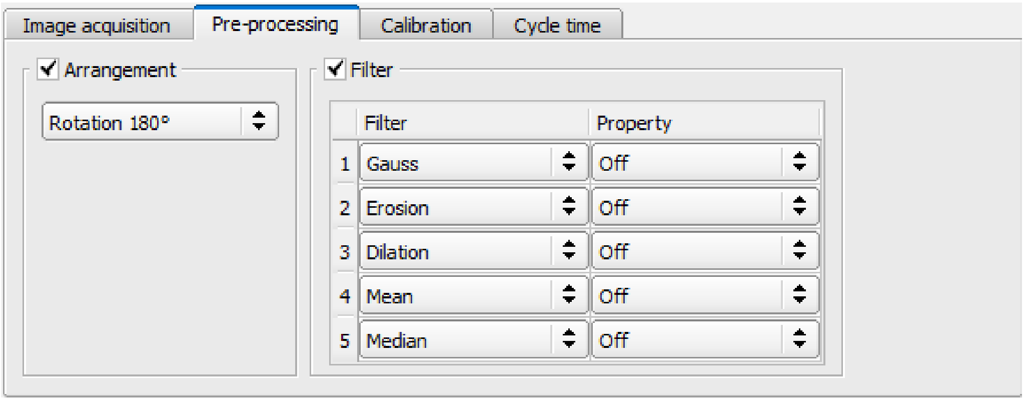
Fig. 46: White balance tab

Parameter	Function
Red	Mean of red channel in image
Green	Mean of green channel in image
Blue	Mean of blue channel in image
Teach	Execution of white balance: For white balance, there should be a homogeneous, white or slightly gray area under the camera to position

Parameter	Function
Reset	Reset values

8.1.5 Pre-processing tab

The Pre-processing tab can be used to filter or rearrange the images captured by the sensor before they are analyzed.



☑ Arrangement

Rotation 180°

☑ Filter

	Filter	Property
1	Gauss	Off
2	Erosion	Off
3	Dilation	Off
4	Mean	Off
5	Median	Off

Fig. 47: *Pre-processing tab*

- Up to 5 filters can be activated, which are executed in the specified order.
- All detectors (Alignment and standard detectors) will work on the preprocessed image, not on the original image.

Arrangement filters

Arrangement type	Effect
Rotation 180°	Rotation of image by 180°
Horizontal mirroring	Horizontal mirroring of the image
Vertical mirroring	Vertical mirroring of the image

Filter for image improvement

In particular, with the morphological operators (dilation and erosion), improvements of the image can also be achieved in combination, e.g. by successive erosion and dilatation or vice versa.

Example: Black spots with a bright background can be eliminated when dilatation and erosion follow one another.

Filter type	Effect
Gauss	The image is smoothed with a gaussian filter. This can be used to reduce noise, suppress interfering details and artifacts, and smooth edges.
Erosion	Extension of dark areas, elimination of bright pixels in dark areas, elimination of artifacts, separation of bright objects. Effect: Each gray value is replaced by the minimum gray value within the filter mask (e.g. 3x3 filter mask).
Dilation	Extension of bright areas, elimination of dark pixels in dark areas, elimination of artifacts, separation of dark objects. Each gray value is replaced by the maximum gray level found within the filter mask (e.g. 3x3).
Median	Each gray value is replaced by the median value of the pixels found within the filter mask (e.g. 3x3). Typical application: Smoothing the image, suppressing image noise, especially local light or dark areas / pixels (salt and pepper noise)

Filter type	Effect
Mean	Each gray value is replaced by the mean of the pixels found within the filter mask (e.g. 3x3). This can be applied for reduction of disturbances, suppression of disturbing details and artifacts and smoothing the image.
Range	Each gray value is replaced by the range value (maximum gray level − minimum gray level) of the pixels found inside the filter mask (e.g. 3x3). Typical applications: Detection and enhancement of edges and improvement of local image contrast.
Standard deviation	Each gray value is replaced by the standard deviation of the pixels found within the filter mask (e.g. 3x3). Typical applications: Highlighting surface defects or edges.
Edge filter (Sobel)	The filtered image contains edges that were found using the Sobel algorithm (see also image-processing literature). Typical applications: Detecting and improving edges, improving local contrast, and detecting surface defects.
Multiplication	The gray value of each pixel is multiplied by the chosen multiplier (2x, 4x, 8x, etc.). The value range is limited to 255.
Inversion	Inversion of image pixel / gray value
Background flattening	If the sensor is not aligned exactly perpendicular to the object or the illumination is inhomogeneous, a brightness gradient may be visible in the background of the image. Local flattening of the brightness values helps to correct this effect. The flattening is carried out via the set number of pixels.

The effect of an activated filter can be seen immediately in the image. The larger the filter core is selected, the stronger the filtering effect. The filters are executed in the order given from top to bottom (1-5).

Configuring the filters

1. Select the filters in the desired order using the drop-down menus in the Filter column.
2. Enter the size of the filter core in the drop-down menu in the "Setting" column. If the setting is "Off", the respective filter is deactivated.

8.1.6 Calibration tab

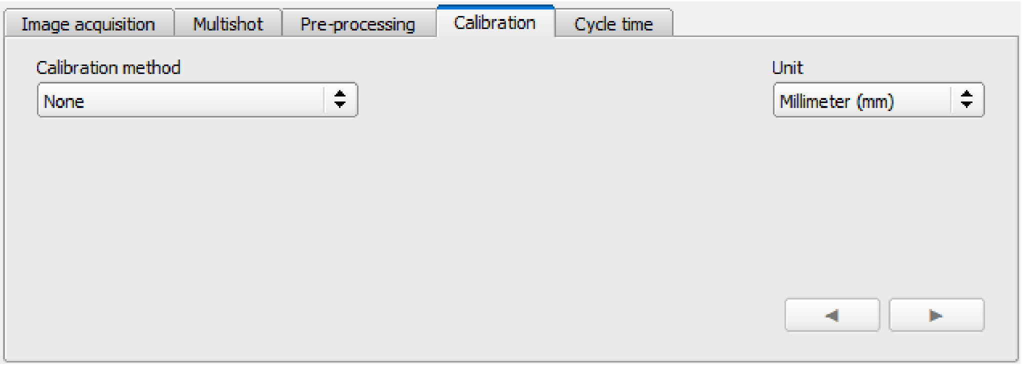
The calibration allows the conversion of image coordinates (pixels) into world coordinates (e.g. millimeters). When using this function, all coordinate outputs (positions and measurement results) are calculated and output in the selected unit.

8.1.6.1 Select the calibration method

The calibration methods are distinguished into two fields of application:

- "Measurement": Calibration methods for applications in the field of measurement and testing
- "Robotics": Calibration methods for applications in the field of robotics

NOTE:
The calibration methods described below are suitable for standard lenses, integrated or C-mount. Only the method "Scaling (Measurement)" is suitable for telecentric lenses.



Calibration method: None

Unit: Millimeter (mm)

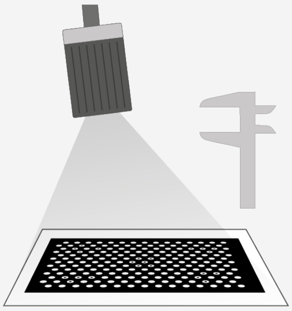
Fig. 48: Select the calibration method

Parameter	Function
Calibration method	Selection of a calibration method: • **None**: Calibration not active, coordinate determination, display and output in pixels [px] • Calibration plate (Robotics) • Point pair list (Robotics) • Hand-Eye calibration (Robotics) • Base-Eye calibration (Robotics) • Calibration plate (Measurement) • Scaling (Measurement)
Unit (user unit)	Desired unit for world coordinates. The following units are available: • Millimeter (mm) • Centimeter (cm) • Meter (m) • Inch (in) • Arbitrary unit (au) Note:If no calibration has been performed, all values refer to pixels.
Robot: Order of rotation	For 3D references, the order of the specified rotations must be observed. You can choose between the two most common pose types: • Yaw-Pitch-Roll (e.g. Stäubli) • Roll-Pitch-Yaw (e.g. Kuka, Fanuc, Hanwha, ABB*, UR**) Note: Here the rotation is referred to the "old" coordinate axes of the reference coordinate system. If you are using a robot whose rotation order refers to the new axes created by the rotation, the following applies: • Roll-Pitch-Yaw (new axes) = Yaw-Pitch-Roll (old axes) • Yaw-Pitch-Roll (new axes) = Roll-Pitch-Yaw (old axes)
◄ / ►	Go to next / previous step

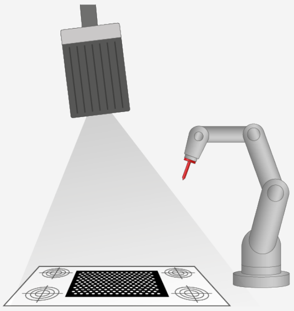
*ABB robots use quaternions as order of rotation. To communicate with the VISOR®, you need to convert the coordinates on the robot to Roll-Pitch-Yaw.

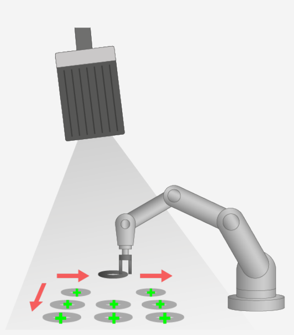
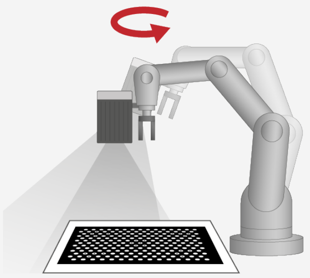
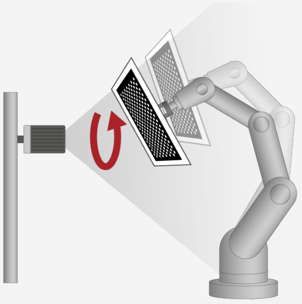
**UR robots use "Axis-Angle" as rotation order. However, they do support a function that converts them into the rotation order "Roll-Pitch-Yaw". Use this function and select Roll-Pitch-Yaw in SensoConfig.

8.1.6.1.1 Overview: Calibration methods "Measurement"

Calibration method	Function
Scaling (Measurement)	
	• Relative determination of e.g. distances in world coordinates (e.g. mm) using a simple ratio factor **Note:** • Limited accuracy • No correction of distortions. Additional information: Calibration method "Scaling (Measurement)"
Calibration plate (Measurement)	
	• Relative determination of e.g. distances in world coordinates (e.g. mm) via the image acquisition of a calibration plate • High accuracy • Correction of tilt and lens distortion • Two variants: • Single image calibration • Multi-image calibration Additional information: Calibration method "Calibration plate (Measurement)"

8.1.6.1.2 Overview: Calibration methods "Robotics"

Calibration method	Function
Calibration plate (Robotics)	
	• Determination of absolute positions in robot coordinates (e.g. mm) • Correction of tilt and lens distortion • Variants: Single image calibration, Multi-image calibration Additional information: Calibration method "Calibration plate (Robotics)"

Calibration method	Function
Point pair list (Robotics)	
	• Determination of absolute positions in robot coordinates (e.g. mm) • Correction of tilt and lens distortion Additional information: Calibration method "Point pair list (Robotics)"
Hand-Eye calibration (Robotics)	
	• Determination of absolute positions in robot coordinates (e.g. mm) • Correction of tilt and lens distortion • Determination of the Hand-Eye reference (reference robot TCP to Camera coordinate system) • Enables shifting the image acquisition position Additional information: Calibration method "Hand-Eye calibration (Robotics)"
Base-Eye calibration (Robotics)	
	• Determination of absolute positions in robot coordinates (e.g. mm) • Correction of tilt and lens distortion • Determination of the Base-Eye reference (reference robot base to Camera coordinate system) Additional information: Calibration method "Base-Eye calibration (Robotics)"

NOTE:
All result values for positions and measurement results are corrected. However, in order not to burden the cycle time, i.e. to extend it, the image data are not converted or equalized! Thus, even with active calibration, a high execution speed is guaranteed.

Status LED

As soon as a calibration method is selected, the status LED is shown on the left side next to the tab title "Calibration". If the calibration is active, all affected functions, e.g. detectors, are only carried out correctly if the calibration is valid (=green), i.e. if it was carried out successfully.

Color of Status LED	Status of the calibration	Meaning / Measures
🟢 Green	Valid	No action required
🟡 Yellow	Valid	Deviations. Recommendation: Re-calibrate
🔴 Red	Invalid	Check the calibration object and re-calibrate

NOTE:

- With method "Scaling (Measurement)" only "green" is possible: Default or input values result in the scaling factor. No error calculation possible.
- "Point pair list (Robotics)": "Green" is displayed for a new job. Default values (9 points) result in correct default calibration.
- All methods with calibration plate: When a new job is created, "Red" appears, because no calibration with calibration plate has yet been carried out.

The calibration affects the following Alignment methods:

Alignment	Result value
Contour matching	Position Coordinates
Pattern matching	Position Coordinates
Edge detector	Position coordinates, distance

The calibration affects the following detectors:

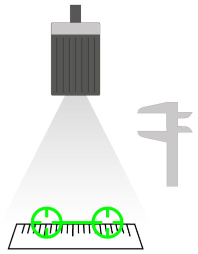
Detector	Result value
Contour	Position Coordinates
Contour 3D	Position Coordinates
Target Mark 3D	Position Coordinates
Pattern matching	Position Coordinates
Caliper	Position coordinates, distance
BLOB	Position coordinates, width, height

8.1.6.2 Calibration methods "Measurement"

By default, distances in the image are given in pixels [px]. These can be converted into metric units such as millimeters by means of a calibration. With calibration methods "Measurement" the origin remains in the upper left corner of the field of view. Besides position coordinates, distances are also converted.

NOTE:
World coordinates are not absolute. The coordinate values refer to the principal point in the left, upper corner or the field of view.

8.1.6.2.1 Calibration method "Scaling (Measurement)"



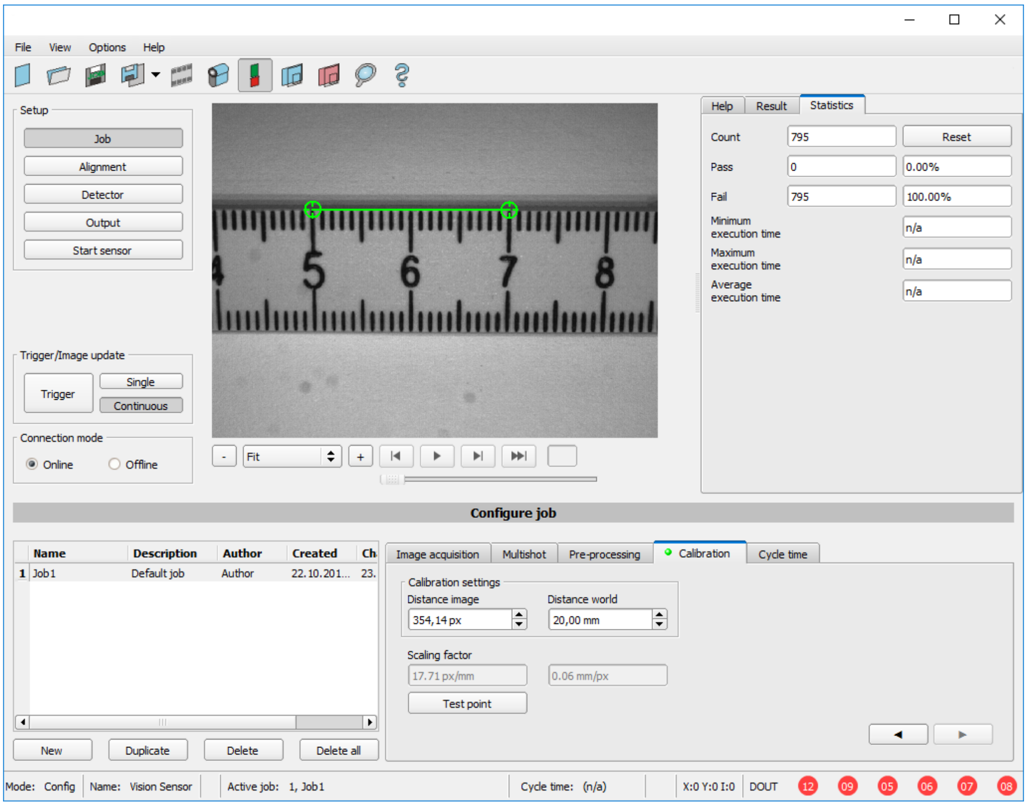
The calibration method "Scaling (Measurement)" is for relative determination of e.g. distances in world coordinates (e.g. mm). This is done using a simple ratio factor for both coordinate axes X and Y. The method is very simple to use, but the accuracy is limited because there is no correction for distortion.

Required objects: Working object

Example: Determination of the distance between two objects in millimeters (medium accuracy) - if the object typically appears in the same region of the field of view.

Calibration method provides:

- Conversion from pixel to measurement unit



Connection mode: ● Online ○ Offline

Count	795	Reset
Pass	0	0.00%
Fail	795	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Name	Description	Author	Created	Ch
1	Job1	Default job	Author	22.10.201...	23.

Calibration settings

Distance image: 354,14 px

Distance world: 20,00 mm

Scaling factor: 17.71 px/mm 0.06 mm/px

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 49: Calibration method "Scaling (Measurement)"

Parameters "Scaling (Measurement)"

Parameter	Function
Distance image	Distance in the image in pixels [px], by graphical or value input.
Distance world	Corresponding distance in world by numerical input (in previously selected unit, e.g. mm)
Test point	A test point (graphical or value input) can be set in the image, the coordinates of which are displayed in world coordinates to control the scaling in the test point window.

Parameter	Function
Scaling factor	Scaling factors in px/mm or mm/px resulting from the above settings "Distance image" and "Distance world".

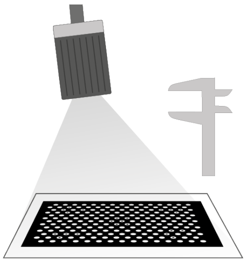
Calibration procedure "Scaling (Measurement)"

1. To parameterize, place an object of known extent (e.g. gauge) in the image.
2. Position the two graphical green crosshairs in the image on the points with the exact known distance.
   The size of the crosshair can be determined via the scroll wheel of the mouse.
   The distance in the image pixels between the two centers is displayed in the field "Distance image".
3. Now enter the known world distance in the field "Distance world" (e.g. in mm).
   The scaling factor is calculated and displayed. From now on, positions and distances are displayed and transferred in world coordinates.

Optimization of the calibration results

- Align the sensor as vertically as possible to the field of view plane in order to avoid excessively different distortions in the two axes X and Y.
- Ideally, the calibration object should be located at the point in the field of view where the measurement will be made later.
- After calibration, the working distance (focus) and the position of the sensor to the Measurement plane must not be changed.

8.1.6.2.2 Calibration method "Calibration plate (Measurement)"



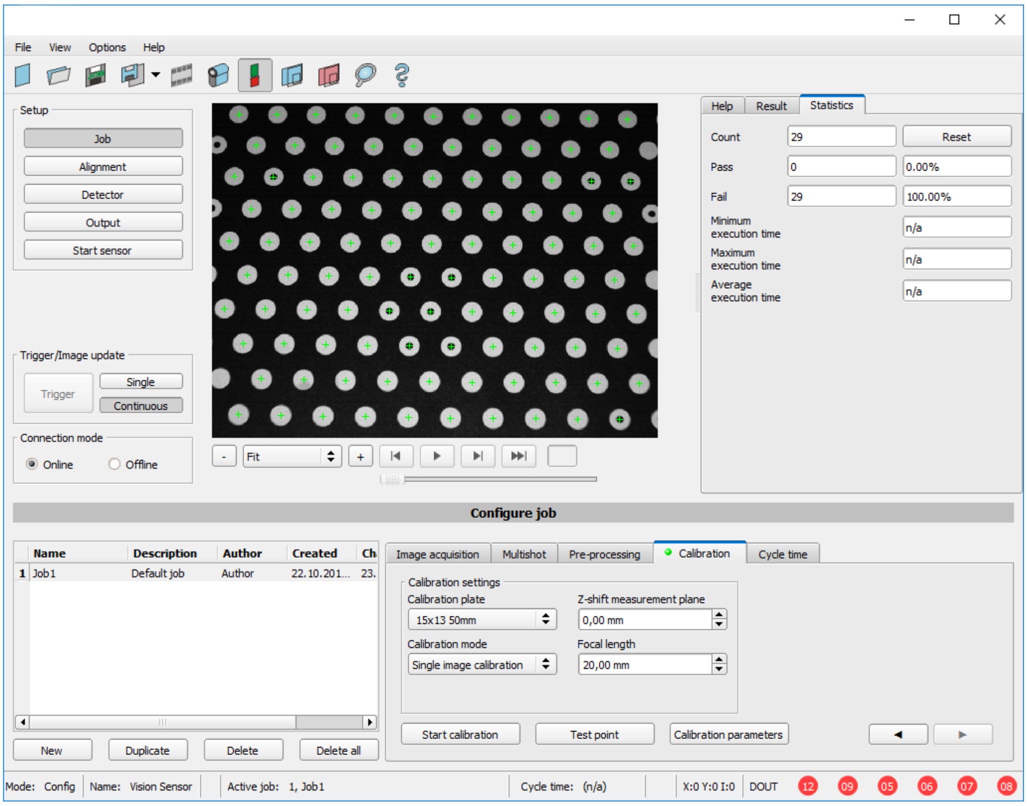
The calibration method "Calibration plate (Measurement)" is used for the relative determination of e.g. distances in world coordinates (e.g. mm). This is done via the image acquisition of a calibration plate

Required objects: Calibration plate

Example: Determination of the distance between two objects in millimetres (high accuracy) - the object appears at varying positions in the camera's field of view.

Calibration method provides:

- Conversion from pixel to measurement unit
- Distortion correction
- Correction of tilt between VISOR® and Measurement plane



Count	29	
Pass	0	0.00%
Fail	29	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Name	Description	Author	Created	Ch
1	Job1	Default job	Author	22.10.201...	23.

Calibration plate: 15x13 50mm

Z-shift measurement plane: 0,00 mm

Calibration mode: Single image calibration

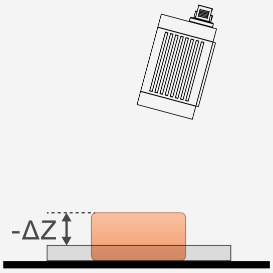
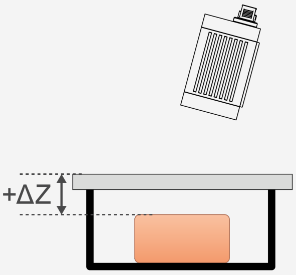
Focal length: 20,00 mm

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 50: *Calibration method "Calibration plate (Measurement)"*

Parameters "Calibration plate (Measurement)"

Parameter	Function
Calibration plate	Selection of the used calibration plate (size/type) (Additional information: Information on calibration plates)

Parameter	Function
Z-shift of Measurement plane	The "Z-shift of Measurement plane" parameter can be used to move the measuring plane along the Z axis (perpendicular to the plane) in order to obtain more accurate results, if necessary. For Z=0, the calibration and the measurement plane are identical. For Z ≠ 0, the calibration plane is shifted relative to the measurement plane. The planes are always parallel. The sign of the shift results from the Z direction of the right-handed calibration coordinate system (thumb = X, index finger = Y, middle finger = Z).  ***Fig. 51:*** *"Z-shift of Measurement plane" negative*  ***Fig. 52:*** *"Z-shift of Measurement plane" positive* **NOTE:** The depth of field of the sensor must cover the calibration plane and the measurement plane!
Calibration mode	Selection of Single image calibration or Multi-image calibration (see Calibration procedure)
Focal length	Focal length of the lens • With integrated lens: Value is entered automatically according to the internally installed lens. • For C-Mount-variant: Read and enter the value from the lens used.
Start calibration	Calibration is started. All visible points of the calibration plate are determined, all detected are marked, and calibration is calculated.
Test point	A test point (graphical or value input) can be set in the image, the coordinates of which are displayed in world coordinates in the test point window to check the calibration or as a plausibility test of the scaling.
Calibration parameters	In the "Calibration parameters" window, parameters determined from the calibration are displayed. These can be used to find errors and optimize the calibration. Additional information: Calibration parameters

Calibration procedure "Calibration plate (Measurement)"

The sensor can be mounted in any orientation (pose) in relation to the Measurement plane (whereby an orientation that is as perpendicular as possible to the Measurement plane requires less correction and is therefore advantageous). The scaling, in X and Y, the tilt of the sensor with respect to the field of view, and the lens distortion (depending on the selected calibration method) are corrected.

The image sharpness and brightness must first be set and the desired unit (to the right of the calibration method selection) must be selected. Two calibration modes are available: Single image calibration and Multi-image calibration.

Calibration mode "Single image calibration"

1. Set "Z-shift of Measurement plane" between the calibration plate and Measurement plane.
2. Place the calibration plate in the field of view (Additional information: "Information on calibration plates").
3. Select the appropriate calibration plate (size and type) in the "Calibration plate" selection box.
4. Start calibration via "Start calibration".
   All visible points of the calibration plate are determined, all detected are marked, and calibration is calculated.
5. Check calibration in the "Calibration parameters" window if necessary.

Calibration mode "Multi-image calibration" (increased accuracy)

1. Set "Z-shift of Measurement plane" between the calibration plate and Measurement plane.
2. Place the calibration plate in the field of view (Additional information: "Information on calibration plates").
3. Select the appropriate calibration plate (size and type) in the "Calibration plate" selection box.
4. Set calibration mode to "Multi-image calibration".
5. Start calibration via "Start calibration".
6. Acquire different images of the calibration plate (recommended: ≥ 6 pictures).
   Note: The **first** image determines the Measurement plane (which can be adjusted later if necessary). For the further images, the calibration plate should be tilted to the Measurement plane and shifted in Z-direction for best results.
7. Check calibration in the "Calibration parameters" window if necessary.

NOTE:
World coordinates are not absolute. The coordinate values refer to the principal point in the left, upper corner or the field of view.

For tips showing how to best use the calibration plate / boundary conditions, please refer to: Information on calibration plates

8.1.6.3 Calibration methods "Robotics"

The robot calibrations first convert pixels into metric units (e.g. mm) and correct distortion and tilted viewing angles. In addition, the camera coordinate system is projected onto that of the robot, so that the robot can now move directly in its coordinate system with the position data supplied by the sensor and can grip a part, for example.

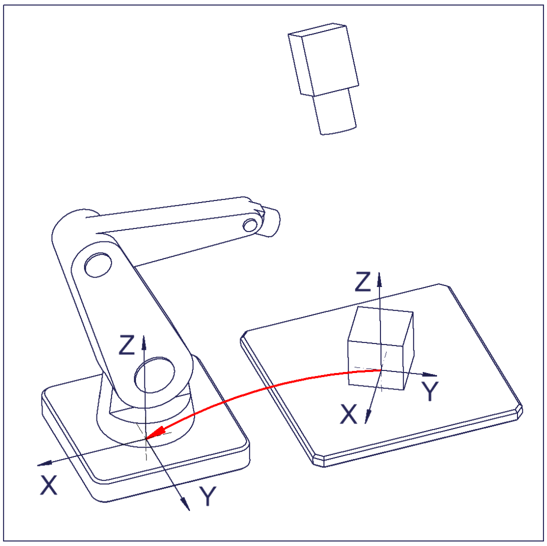
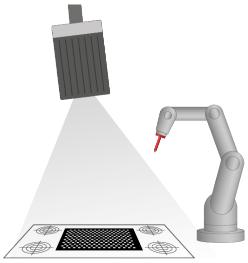


Fig. 53: *Position of the part to be gripped directly in the robot coordinate system*

8.1.6.3.1 Calibration method "Calibration plate (Robotics)"



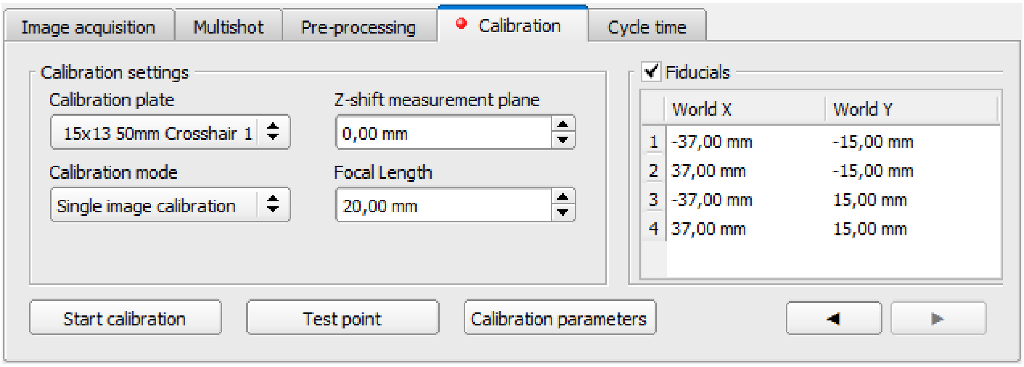
The calibration method "Calibration plate (Robotics)" is used to determine absolute positions in the robot coordinate system. This is done by acquiring one or more images of the calibration plate and teaching four fiducials.

Required objects: Calibration plate "Crosshair" (calibration plate with fiducials)

Example: Picking parts from a feeder with a stationary mounted VISOR®.

Calibration method provides:

- Conversion from pixel to measurement unit
- Distortion correction
- Correction of tilt between VISOR® and Measurement plane
- Output of world coordinates in robot coordinate system



Calibration settings

Calibration plate: 15x13 50mm Crosshair 1

Z-shift measurement plane: 0,00 mm

Calibration mode: Single image calibration

Focal Length: 20,00 mm

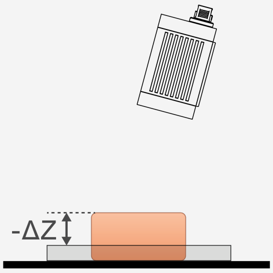
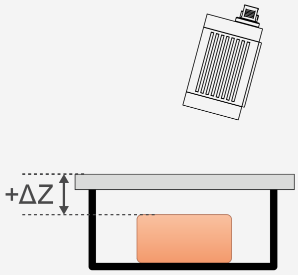
☑ Fiducials

	World X	World Y
1	-37,00 mm	-15,00 mm
2	37,00 mm	-15,00 mm
3	-37,00 mm	15,00 mm
4	37,00 mm	15,00 mm

Fig. 54: Calibration method "Calibration plate (Robotics)"

Parameters "Calibration plate (Robotics)"

Parameter	Function
Calibration plate	Selection of the used calibration plate (size/ type) (Additional information: "Information on calibration plates")

Parameter	Function
Z-shift of Measurement plane	The "Z-shift of Measurement plane" parameter can be used to move the measuring plane along the Z axis (perpendicular to the plane) in order to obtain more accurate results, if necessary. For Z=0, the calibration and the measurement plane are identical. For Z ≠ 0, the calibration plane is shifted relative to the measurement plane. The planes are always parallel. The sign of the shift results from the Z direction of the right-handed calibration coordinate system (thumb = X, index finger = Y, middle finger = Z).  *Fig. 55:* *"Z-shift of Measurement plane" negative*  *Fig. 56:* *"Z-shift of Measurement plane" positive* **NOTE:** The depth of field of the sensor must cover the calibration plane and the measurement plane!
Calibration mode	Selection of Single image calibration or Multi-image calibration (see Calibration procedure)
Focal length	Focal length of the lens - With integrated lens: Value is entered automatically according to the internally installed lens. - For C-Mount-variant: Read and enter the value from the lens used.
Fiducials - World X - World Y	Coordinate values in world in selected unit (e.g. mm), by directly entering values in the list of fiducials. In the case of Pick & Place, for example, these values are the X/Y coordinate values that can be read off from the robot controller when the calibration part is placed or the corresponding point is approached and transferred to the list of fiducials.
Start calibration	Calibration is started. All visible points of the calibration plate are determined, all detected are marked, and calibration is calculated.
Test point	A test point (graphical or value input) can be set in the image, the coordinates of which are displayed in world coordinates in the test point window to check the calibration or as a plausibility test of the scaling.
Calibration parameters	In the "Calibration parameters" window, parameters determined from the calibration are displayed. These can be used to find errors and optimize the calibration. Additional information: Calibration parameters

Calibration procedure "Calibration plate (Robotics)"

The image sharpness and brightness must first be set and the desired unit (to the right of the calibration method selection) must be selected. Two calibration modes are available: Single image calibration and Multi-image calibration.

Calibration mode "Single image calibration"

1. Set "Z-shift of Measurement plane" between the calibration plate and Measurement plane.
2. Place the calibration plate in the field of view so that it covers as much as possible (Additional information: "Information on calibration plates").
3. Select the appropriate calibration plate (size and type) in the "Calibration plate" selection box.
4. Start calibration via "Start calibration".
   All visible points of the calibration plate are determined, all detected are marked, and calibration is calculated.
5. Check "Fiducials" if not yet active.
6. For fiducial 1, select the first line in list box "Fiducials".
7. Approach the first fiducial with the robot.
8. In the field "World X" and "World Y", enter the corresponding known world coordinate values (for robots: the values from the robot controller).
9. For fiducials 2, 3 and 4: Select the next line in the list box "Fiducials". Move to the next fiducial and enter the corresponding values (see steps 6-8).
10. Check calibration in the "Calibration parameters" window if necessary.

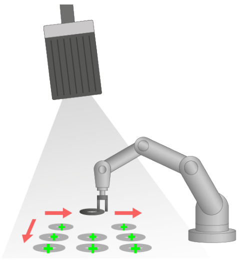
Calibration mode "Multi-image calibration" (increased accuracy)

1. Set "Z-shift of Measurement plane" between the calibration plate and Measurement plane.
2. Place the calibration plate in the field of view (Additional information: "Information on calibration plates").
3. Select the appropriate calibration plate (size and type) in the "Calibration plate" selection box.
4. Set calibration mode to "Multi-image calibration".
5. Start calibration via "Start calibration".
6. Acquire different images of the calibration plate (recommended: ≥ 6 pictures).
   Note: The **first** image determines the Measurement plane (which can be adjusted later if necessary). For the further images, the calibration plate should be tilted to the Measurement plane and shifted in Z-direction for best results.
7. Check "Fiducials" if not yet active.
8. For fiducial 1, select the first line in list box "Fiducials".
9. Approach the first fiducial with the robot.
10. In the field "World X" and "World Y", enter the corresponding known world coordinate values (for robots: the values from the robot controller).
11. For fiducials 2, 3 and 4: Select the next line in the list box "Fiducials". Move to the next fiducial and enter the corresponding values (see steps 8-10).
12. Check calibration in the "Calibration parameters" window if necessary.

NOTE:
Calibration can also be performed using telegrams.
Additional information: Communications manual, chapter "Calibration with telegrams".

For tips showing how to best use the calibration plate / boundary conditions, please refer to: Information on calibration plates

8.1.6.3.2 Calibration method "Point pair list (Robotics)"



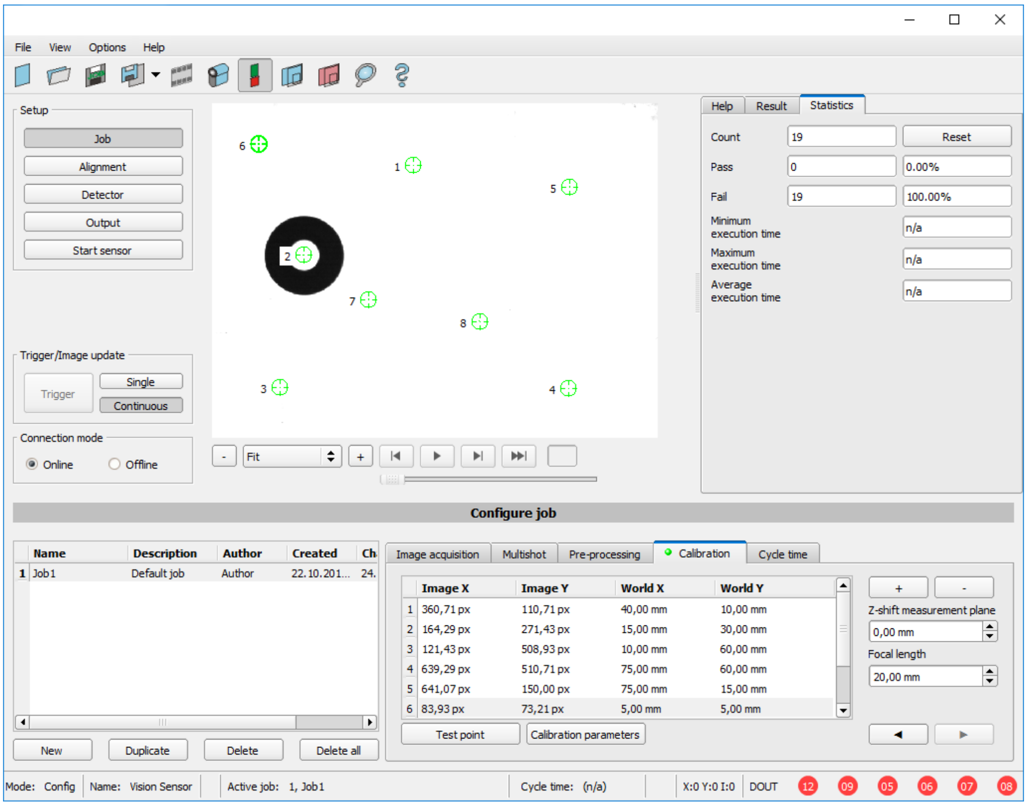
The calibration method "Point pair list (Robotics)" is a calibration at the working object - therefore no calibration plate is needed. After calibration of the sensor, the position of the part to be gripped by the robot is directly available in the absolute coordinate system of the robot.

Required objects: Working object

Example: Determine absolute position (and orientation) of objects in world coordinates (e.g. robot coordinate system).

Calibration method provides:

- Conversion from pixel to measurement unit
- Distortion correction
- Correction of tilt between VISOR® and Measurement plane
- Output of world coordinates in robot coordinate system



Connection mode: ● Online ○ Offline

Count	19
Pass	0	0.00%
Fail	19	100.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Name	Description	Author	Created	Ch
1	Job1	Default job	Author	22.10.201...	24.

	Image X	Image Y	World X	World Y
1	360,71 px	110,71 px	40,00 mm	10,00 mm
2	164,29 px	271,43 px	15,00 mm	30,00 mm
3	121,43 px	508,93 px	10,00 mm	60,00 mm
4	639,29 px	510,71 px	75,00 mm	60,00 mm
5	641,07 px	150,00 px	75,00 mm	15,00 mm
6	83,93 px	73,21 px	5,00 mm	5,00 mm

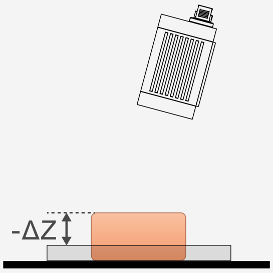
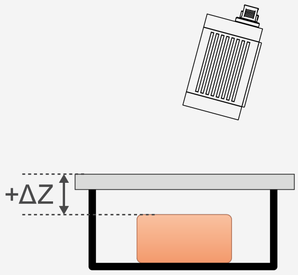
Z-shift measurement plane: 0,00 mm

Focal length: 20,00 mm

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT

Fig. 57: Calibration method "Point pair list (Robotics)"

Parameters "Point pair list (Robotics)"

Parameter	Function
- **Image X** - **Image Y** Values in point list	Coordinate values in pixels [px] in the image, via the exact graphical positioning of the crosshair on the center point of the calibration part, which is precisely placed in world coordinates. Or, via "Snap function": Right mouse click anywhere within the symmetrical calibration object. This way, the exact position of the center can be determined automatically. ℹ **NOTE:** The snap function is not available for Color variants.
- **World X** - **World Y** Values in point list	Coordinate values in selected unit (e.g. mm), by directly entering values in point pair list. In the case of Pick & Place, for example, these values are the X/Y coordinate values that can be read by the robot controller when the calibration part is placed or the corresponding point is approached and transferred to the reference mark list.
+ / -	Add or delete one line / list point. The selected line is deleted.
Z-shift of Measurement plane	The "Z-shift of Measurement plane" parameter can be used to move the measuring plane along the Z axis (perpendicular to the plane) in order to obtain more accurate results, if necessary. For Z=0, the calibration and the measurement plane are identical. For Z ≠ 0, the calibration plane is shifted relative to the measurement plane. The planes are always parallel. The sign of the shift results from the Z direction of the right-handed calibration coordinate system (thumb = X, index finger = Y, middle finger = Z).  ***Fig. 58:*** *"Z-shift of Measurement plane" negative*  ***Fig. 59:*** *"Z-shift of Measurement plane" positive* ℹ **NOTE:** The depth of field of the sensor must cover the calibration plane and the measurement plane!
Focal length	Focal length of the lens - With integrated lens: Value is entered automatically according to the internally installed lens. - For C-Mount-variant: Read and enter the value from the lens used.
Test point	A test point (graphical or value input) can be set in the image, whose coordinates are displayed in world coordinates for checking the calibration or as a plausibility test of the scaling in the test point window
Calibration parameters	In the "Calibration parameters" window, parameters determined from the calibration are displayed. These can be used to find errors and optimize the calibration. Additional information: Calibration parameters

Calibration procedure "Point pair list (Robotics)"

NOTE:

- The accuracy depends mainly on the high quality as well as on the sufficient number of calibration points (at least 6 points, recommended: ≥ 9 points).
- The accuracy can be optimized by a high precision in position determination and input of the individual points, e.g. if some points are displayed in yellow color.
- Preferably use flat, point-symmetrical calibration objects (e.g. washer), as this way the center of gravity is independent of the orientation. For calibration objects that are not point-symmetrical, ensure that the orientation is always the same when positioning.

The sensor can be mounted in any orientation (pose) with respect to the Measurement plane. An alignment as perpendicular as possible with the Measurement plane, however, requires less correction and is therefore more advantageous.

The image sharpness and brightness must first be set and the desired unit (to the right of the calibration method selection) must be selected.

1. Set "Z-shift of Measurement plane".
2. Select Line 1 in list box "Point pair list".
3. Place the calibration object at an exactly known world coordinate in the field of view (e.g. with a robot).
4. Graphically position the crosshairs (no. "n" to the corresponding line "n" in the point pair list) exactly in the center of the calibration object. Zoom the image if necessary.
   Alternatively: Use "Snap Function", i.e. right click somewhere inside the calibration object. The center of gravity of the calibration object is determined automatically (not available for Color variants).
   The size of the crosshair can be determined via the scroll wheel of the mouse.
   Result: Pixel values for image coordinates "Image X" and "Image Y" are automatically entered in Line "n".
5. Now enter the corresponding known world coordinate values in the "World X" and "World Y" fields (for example, for robots: the values from the robot controller).
6. Repeat steps 3-5 until the desired number of point pairs is entered (min. 6 points, recommended >10 points). If necessary, create further lines with "+".
7. Check calibration in the "Calibration parameters" window if necessary.

NOTE:
Calibration can also be performed using telegrams.
Additional information: Communications manual, chapter "Calibration with telegrams".

Meaning of the colors of the points in the image and in the point pair list

The entered points are displayed in the following colors to indicate the position quality, i.e. how well they correspond to the position determined by the adjustment calculation (can only be used effecitvely from a minimum number of 6 points).

Color of the crosshairs	Status of the calibration	Meaning / Measures
⊕ Green	Calibration valid, points accurately positioned	No action required
⊕ Yellow	Calibration valid, point is not exactly positioned	Check pair of points of the point
⊕ Red	No valid calibration, assignment of world points / pixels deviates strongly from the model.	Check assignment

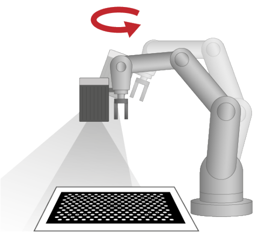
Error

In the case of yellow points, a line is visible from the point center. It is a measure of the direction and magnitude of the error with respect to the achieved position accuracy of the point input in the real world.

If the errors are large here, the X and Y values or entire pairs of points in the image and world may be interchanged at one or more points.

In the "Calibration parameters" dialog, the **deviation values** / errors are displayed: "Average", minimum error "Min." and maximum error "Max.". The exact position input of the existing points may be optimized with these values.

8.1.6.3.3 Calibration method "Hand-Eye calibration (Robotics)"



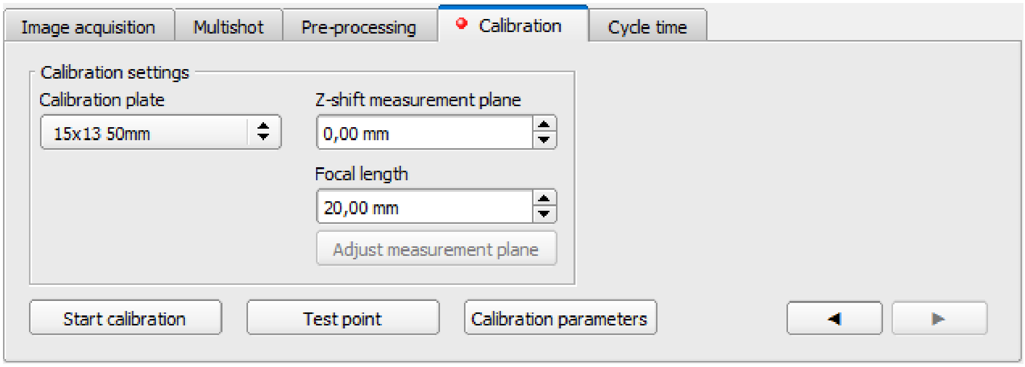
The "Hand-Eye calibration (Robotics)" calibration method is used to determine the reference between Tool Coordinate System (TCP) and Camera coordinate system (position and orientation) when the VISOR® is attached to the gripper.

Required objects: Calibration plate

Example: Screwing on components with multiple positions on VISOR® attached to robot arm.

Calibration method provides:

- Conversion from pixel to measurement unit
- Distortion correction
- Correction of tilt between VISOR® and Measurement plane
- Output of world coordinates in robot coordinate system, independent of image acquisition position
- References (see figure References: "Robotics" Calibration methods)
  - TCP_CF (Tool Coordinate System (TCP) - Camera coordinate system, corresponds to Hand-Eye)
  - CF_CPF (Camera coordinate system - Calibration Plate Coordinate System)
  - CPF_MF (Calibration Plate Coordinate System - Measuring coordinate system)



Calibration settings

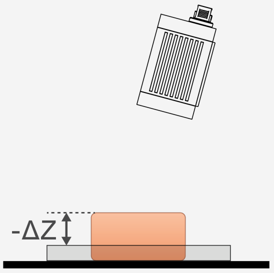
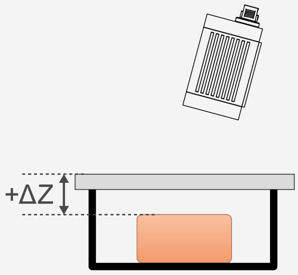
Calibration plate: 15x13 50mm

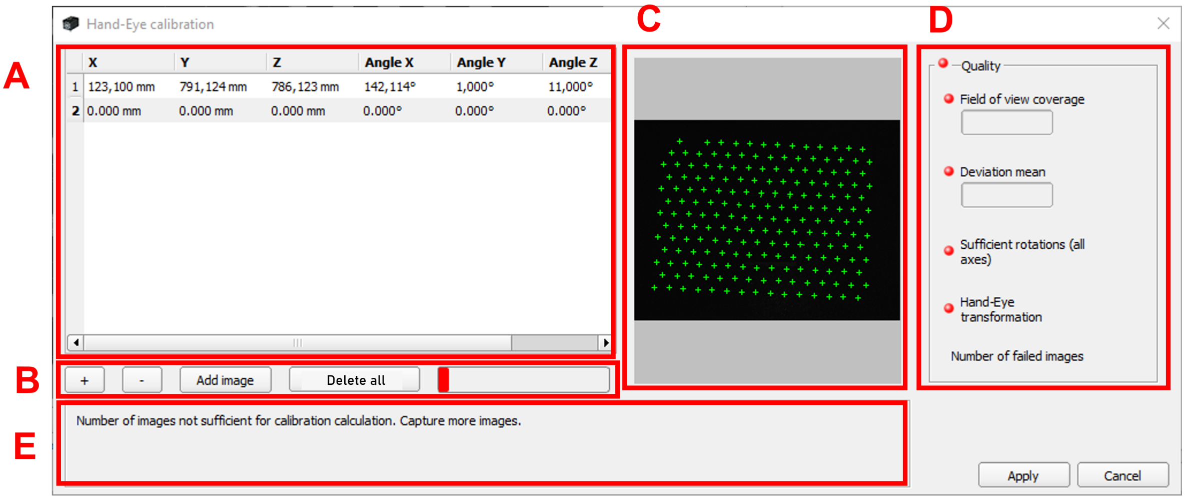
Z-shift measurement plane: 0,00 mm

Focal length: 20,00 mm

Fig. 60: Calibration method "Hand-Eye calibration (Robotics)"

Parameter "Hand-Eye calibration (Robotics)"

Parameter	Function
Calibration plate	Selection of the used calibration plate (size/ type) (Additional information: "Information on calibration plates")
Z-shift of Measurement plane	The "Z-shift of Measurement plane" parameter can be used to move the measuring plane along the Z axis (perpendicular to the plane) in order to obtain more accurate results, if necessary. For Z=0, the calibration and the measurement plane are identical. For Z ≠ 0, the calibration plane is shifted relative to the measurement plane. The planes are always parallel. The sign of the shift results from the Z direction of the right-handed calibration coordinate system (thumb = X, index finger = Y, middle finger = Z).  ***Fig. 61:*** *"Z-shift of Measurement plane" negative*  ***Fig. 62:*** *"Z-shift of Measurement plane" positive* **NOTE:** The depth of field of the sensor must cover the calibration plane and the measurement plane!
Focal length	Focal length of the lens - With integrated lens: Value is entered automatically according to the internally installed lens. - For C-Mount-variant: Read and enter the value from the lens used.
Adjust Measurement plane	This allows the Measurement plane to be changed subsequently. This is necessary, for example, if the orientation of the vision sensor or the distance to the Measurement plane have changed, typically if the calibration position and working position are different. The tool position (TCP) is used for the calculation. If the current position differs from the stored position, it can be entered in the dialog.
Start calibration	Calibration process is initiated: Dialog "Hand-Eye calibration (Robotics)" opens (see Dialog Hand-Eye calibration (Robotics)).
Test point	A test point (graphical or value input) can be set in the image, the coordinates of which are displayed in world coordinates in the test point window to check the calibration or as a plausibility test of the scaling.
Calibration parameters	In the "Calibration parameters" window, parameters determined from the calibration are displayed. These can be used to find errors and optimize the calibration. Additional information: Calibration parameters



	X	Y	Z	Angle X	Angle Y	Angle Z
1	123,100 mm	791,124 mm	786,123 mm	142,114°	1,000°	11,000°
2	0.000 mm	0.000 mm	0.000 mm	0.000°	0.000°	0.000°

Number of images not sufficient for calibration calculation. Capture more images.

Fig. 63: *Dialog Hand-Eye calibration (Robotics)*

A: Input area for current tool position (TCP)

B: Adding images for calibration; status display

C: Display area of the recorded images

D: Evaluation of the currently calculated calibration

E: Information area for notes

General notes on Hand-Eye calibration (Robotics)

- The position of the calibration plate must not change during calibration.
- Make sure that there is enough space around your robot tool (TCP) to perform 10 poses as described in the position selection.
- After calibration, the working distance (focus) must not be changed.
- The calibration is only valid for the tool (TCP) and the coordinate system that are active during the calibration.
- The accuracy of the calibration can often be further increased by adding more images.
- In order to have sufficient freedom of movement for the robot, the distance of flange to VISOR® should be significantly smaller (~< 25%) than the length of the robot arm.

Notes on position selection for a Hand-Eye calibration (Robotics)

- Use your Tool Coordinate System (TCP) to move the robot
- Tilt your tool (TCP) strongly around 2 axes between each pose you approach, ideally ~ 60° (min. 20°).
- Then perform the necessary translation to bring the calibration plate back into the field of view of the camera.
- In this procedure, try to achieve the greatest possible variation of the tilts between all poses used for calibration.

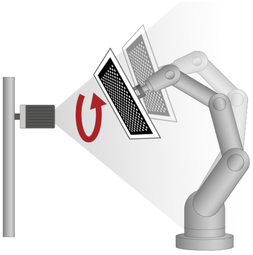
Procedure of the calibration method "Hand-Eye calibration (Robotics)"

1. Select the correct size and type of calibration plate.
2. Set "Z-shift of Measurement plane".
3. Position the calibration plate in the field of view or move the camera (attached to the robot arm) over the calibration plate.
4. Set image acquisition parameters (Shutter speed, Working distance of the VISOR®). The "Parameters" Working distance must not be changed anymore from now on.

5. Click "Start calibration". The calibration dialog opens.
6. By clicking „+“ (Fig. Dialog Hand-Eye calibration (Robotics) A) a new line is added. In offline mode, the next image from the filmstrip is automatically moved in. In online mode, a recording is triggered. By clicking Add image (Fig. Dialog Hand-Eye calibration (Robotics) B) a new recording can manually executed. This overwrites the existing image.
   Optional: Clicking on "-" (Fig. 63, B) deletes the currently selected line (Fig. 63 A).
7. Read the robot position from the robot and enter it in the corresponding line in the pose overview (Fig. Dialog Hand-Eye calibration (Robotics) A).
8. Change robot position and perform steps 6 and 7 for at least 6 (recommended: 10) robot positions. Please observe the notes on position selection!
9. Click "Apply".
10. Check calibration in the "Calibration parameters" window if necessary.

NOTE:
Calibration can also be performed using telegrams.
Additional information: Communications manual, chapter "Calibration with telegrams".

8.1.6.3.4 Calibration method "Base-Eye calibration (Robotics)"



The "Base-Eye calibration (Robotics)" calibration method is used to determine the reference from Camera coordinate system to the robot base (position and orientation) when the vision VISOR® sensor is mounted stationary.

Required objects: Calibration plate

Example: Position correction of gripped component in front of a stationary mounted VISOR®.

Calibration method provides:

- Conversion from pixel to measurement unit
- Distortion correction
- Correction of tilt between VISOR® and Measurement plane
- Output of world coordinates in robot coordinate system
- References
  - RF_CF (Robot Coordinate System - Camera coordinate system, corresponds to Base-Eye)
  - CF_CPF (Camera coordinate system - Calibration Plate Coordinate System)
  - CPF_MF (Calibration Plate Coordinate System - Measuring coordinate system)

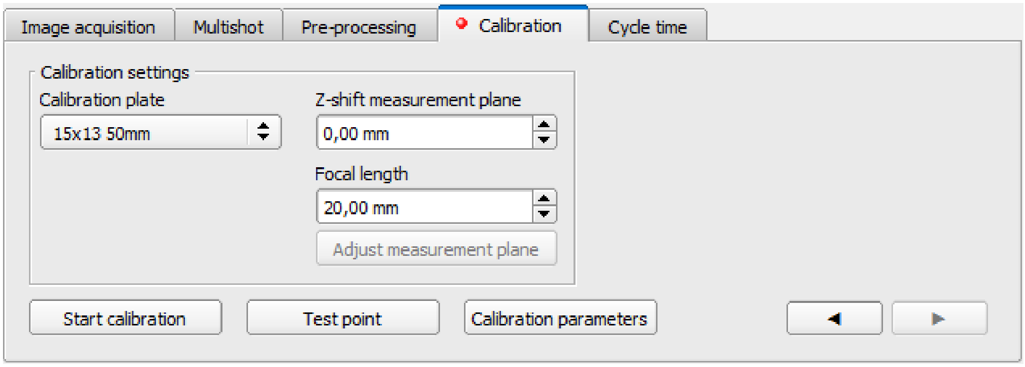


Image acquisition | Multishot | Pre-processing | Calibration | Cycle time

Calibration settings

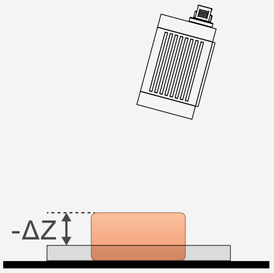
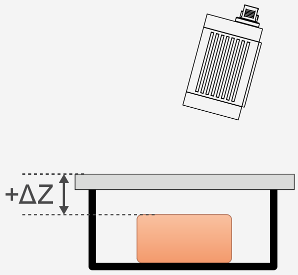
Calibration plate: 15x13 50mm

Z-shift measurement plane: 0,00 mm

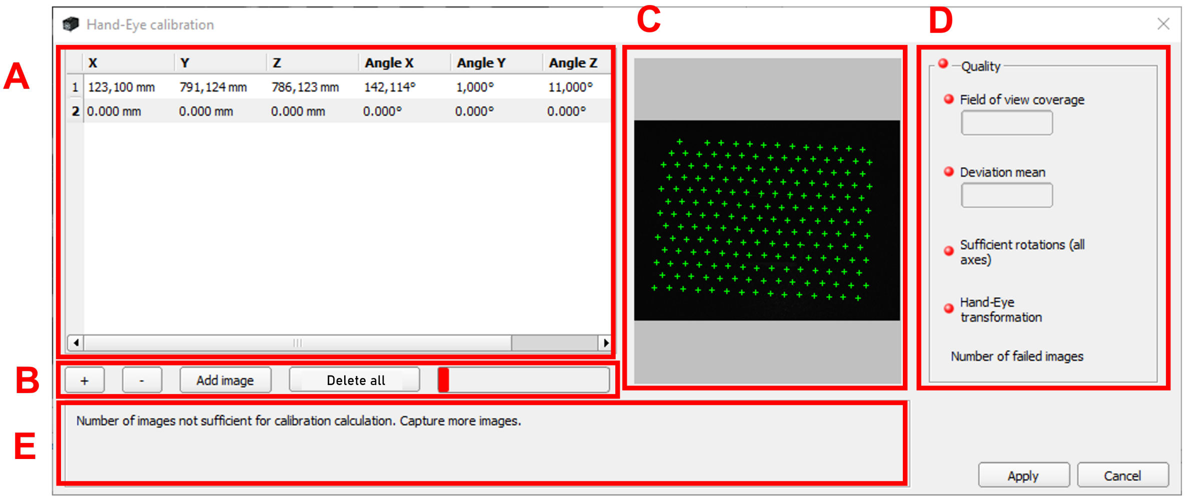
Focal length: 20,00 mm

Fig. 64: Calibration method "Base-Eye calibration (Robotics)"

Parameter "Base-Eye calibration (Robotics)"

Parameter	Function
Calibration plate	Selection of the used calibration plate (size/ type) (Additional information: "Information on calibration plates")
Z-shift of Measurement plane	The "Z-shift of Measurement plane" parameter can be used to move the measuring plane along the Z axis (perpendicular to the plane) in order to obtain more accurate results, if necessary. For Z=0, the calibration and the measurement plane are identical. For Z ≠ 0, the calibration plane is shifted relative to the measurement plane. The planes are always parallel. The sign of the shift results from the Z direction of the right-handed calibration coordinate system (thumb = X, index finger = Y, middle finger = Z).  *Fig. 65: "Z-shift of Measurement plane" negative*  *Fig. 66: "Z-shift of Measurement plane" positive* **NOTE:** The depth of field of the sensor must cover the calibration plane and the measurement plane!
Focal length	Focal length of the lens • With integrated lens: Value is entered automatically according to the internally installed lens. • For C-Mount-variant: Read and enter the value from the lens used.

Parameter	Function
Adjust Measurement plane	This allows the Measurement plane to be changed subsequently. This is necessary, for example, if the orientation of the vision sensor or the distance to the Measurement plane have changed, typically if the calibration position and working position are different. The tool position (TCP) is used for the calculation. If the current position differs from the stored position, it can be entered in the dialog.
Start calibration	Calibration process is initiated: Dialog "Base-Eye calibration (Robotics)" opens (see Dialog Base-Eye calibration (Robotics)).
Test point	A test point (graphical or value input) can be set in the image, the coordinates of which are displayed in world coordinates in the test point window to check the calibration or as a plausibility test of the scaling.
Calibration parameters	In the "Calibration parameters" window, parameters determined from the calibration are displayed. These can be used to find errors and optimize the calibration. Additional information: Calibration parameters



	X	Y	Z	Angle X	Angle Y	Angle Z
1	123,100 mm	791,124 mm	786,123 mm	142,114°	1,000°	11,000°
2	0.000 mm	0.000 mm	0.000 mm	0.000°	0.000°	0.000°

Number of images not sufficient for calibration calculation. Capture more images.

Fig. 67: *Dialog Base-Eye calibration (Robotics)*

A: Input area for current tool position (TCP)

B: Adding images for calibration; status display

C: Display area of the recorded images

D: Evaluation of the currently calculated calibration

E: Information area for notes

General notes on Base-Eye calibration (Robotics)

- The position of the calibration plate to the gripper must not change during calibration (calibration plate must not slip).
- Make sure that there is enough space around your robot tool (TCP) to perform 10 poses as described in the position selection.
- After calibration, the working distance (focus) must not be changed.
- The calibration is only valid for the coordinate system active during calibration.
- The accuracy of the calibration can often be further increased by adding more images.

Notes on position selection for a Base-Eye calibration (Robotics)

- Use your Tool Coordinate System (TCP), to move the robot
- Tilt your tool (TCP) strongly around 2 axes between each pose you approach, ideally ~ 60° (min. 20°).
- Then perform the necessary translation to bring the calibration plate back into the field of view of the camera.
- In this procedure, try to achieve the greatest possible variation of the tilts between all poses used for calibration.

Procedure of the calibration method "Base-Eye calibration (Robotics)"

1. Select the correct size and type of calibration plate.
2. Set "Z-shift of Measurement plane".
3. Position the calibration plate in the field of view or move the camera (attached to the robot arm) over the calibration plate.
4. Set image acquisition parameters (Shutter speed, Working distance of the VISOR®). The "Parameters" Working distance must not be changed anymore from now on.
5. Click "Start calibration". The calibration dialog opens.
6. By clicking „+“ (Fig. Dialog Base-Eye calibration (Robotics) B) a new line is added. In offline mode, the next image from the filmstrip is automatically moved in. In online mode, a recording is triggered. By clicking Add image (Fig. Dialog Base-Eye calibration (Robotics) B) a new recording can manually executed. This overwrites the existing image.
   Optional: Clicking on "-" (Fig. 63, B) deletes the currently selected line (Fig. Dialog Base-Eye calibration (Robotics) A).
7. Read the robot position from the robot and enter it in the corresponding line in the pose overview (Fig. Dialog Base-Eye calibration (Robotics) A).
8. Change robot position and perform steps 6 and 7 for at least 6 (recommended: 10) robot positions. Please observe the notes on position selection!
9. Click "Apply".
10. Check calibration in the "Calibration parameters" window if necessary.

NOTE:
Calibration can also be performed using telegrams.
Additional information: Communications manual, chapter "Calibration with telegrams".

8.1.6.4 Information on calibration plates

When using calibration plates, the scaling in X and Y, the tilt of the sensor relative to the plane of view and the lens distortion (depending on the calibration method selected) are corrected.

Calibration plates can be ordered via the website or they can be printed or applied to paper or any other flat medium. In the installation directory \SensoPart\VISOR Vision Sensor\Documentation\Calibrationplates, the calibration plates available for this purpose can be found as PDF files. When printing, use the "Actual Size" setting, and ensure not to scale the print. The edge length / label of the plate must match the name of the plate when selected in the software.

Calibration plates without fiducials

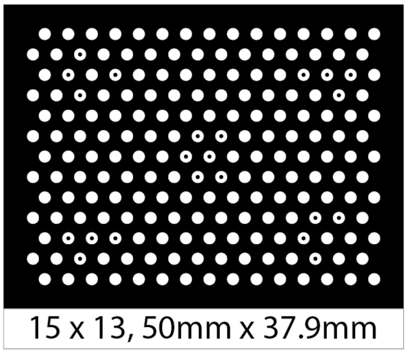


Fig. 68: *Calibration plate without fiducials*

Typically used for calibration methods: Calibration plate (Measurement), Hand-Eye calibration (Robotics), Base-Eye calibration (Robotics)

Calibration plates with fiducials

Typically used for calibration method: Calibration plate (Robotics)

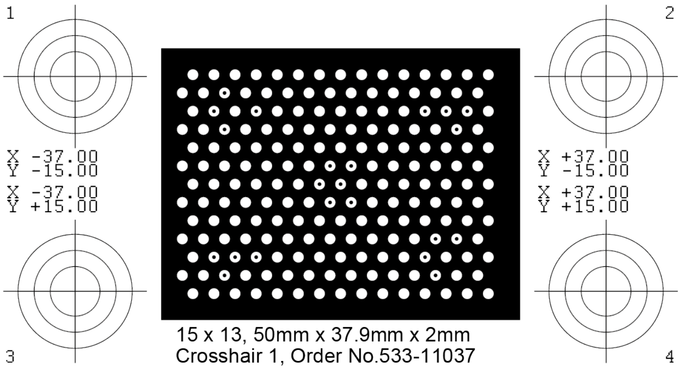


Fig. 69: *Calibration plate with fiducials*

Advice on optimized use of the calibration plate / boundary conditions

- The calibration plate must be clean and level.
- The illumination of the panel should be homogeneous throughout the field of view and not overexposed. The light areas should have a gray value of at least 100 and all below the value 255. The contrast between light and dark areas should be at least 100 gray levels. This means that the image may not be underexposed or overexposed.
- The calibration pattern should ideally cover the entire field of vision of the VISOR® vision sensor. This can be ensured either by a large calibration plate or by a Multi-image calibration.
- To perform a calibration, at least one search pattern must be found.
- For small calibration patterns, it may be necessary to use two search patterns.
- After calibration is complete, the working distance (focus) must not be changed anymore. If the position of the camera to the Measurement plane changes, the Measurement plane must be taught-in again.

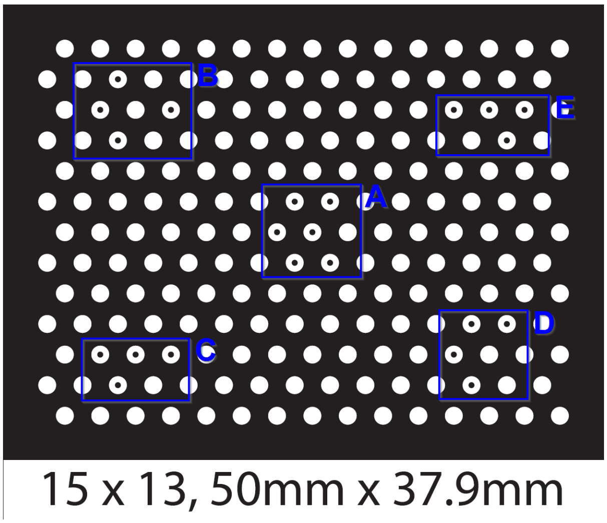


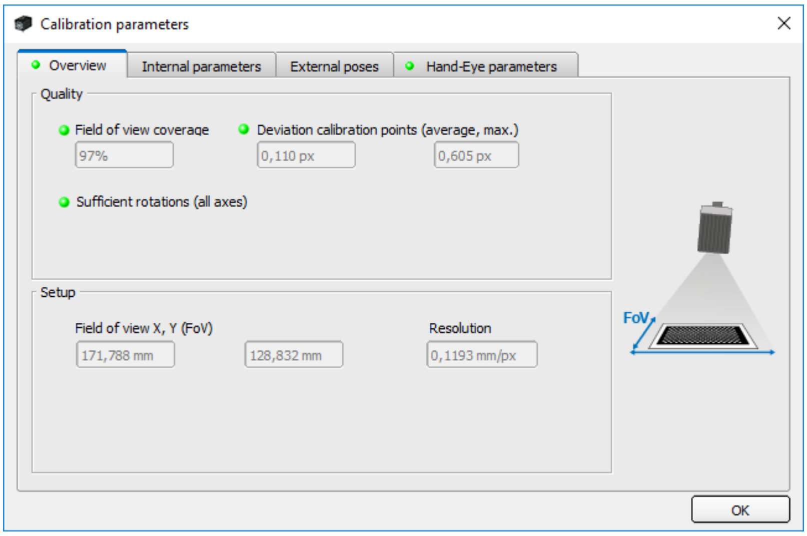
Fig. 70: Calibration plate, blue = search pattern

8.1.6.5 Calibration parameters

In the "Calibration parameters" window, parameters determined from the calibration are displayed. These can be used to find errors and optimize the calibration. The displayed parameters are read-only parameters.

All parameters are described below - however, not all parameters are available for every calibration method.

"Overview" tab



Quality

Field of view coverage: 97%

Deviation calibration points (average, max.): 0,110 px 0,605 px

Sufficient rotations (all axes)

Setup

Field of view X, Y (FoV): 171,788 mm 128,832 mm

Resolution: 0,1193 mm/px

Fig. 71: Calibration parameters, "Overview" tab

Parameter	Function
Quality	
Field of view coverage	Indicates in which part of the field of view calibration objects (e.g. calibration plate) were detected. In order to obtain accurate results, the highest possible value should be achieved here (100%).
Deviation Calibration points (Mean, Max.)	Deviation of the calibration points between detected and expected position in pixels.

Parameter	Function
Deviation fiducials (Average, Max.)	Deviation of the reference marks between specified and expected position in pixels.
Sufficient rotations (all axes)	Indicator for a good calibration. Green: exact calibration Yellow: Image width cannot be determined exactly, position information becomes less accurate. LED turns green when tilt difference between some of the recorded calibration plate images is at least 20 degrees.
Setup	
Field of view X, Y (FoV)	Size of the field of view detected by the VISOR®
Resolution	Resolution from customer unit to pixels (customer unit / px) in the determined field of view

"Internal Parameters" tab

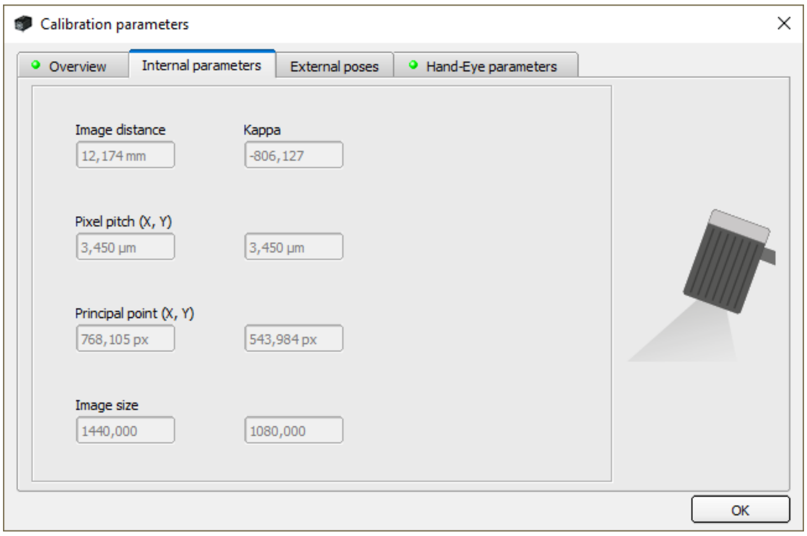
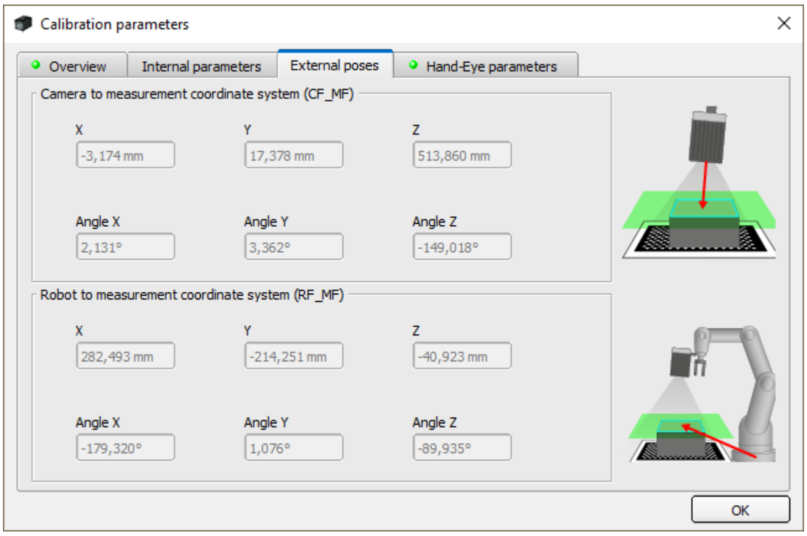


Image distance: 12,174 mm
Kappa: -806,127
Pixel pitch (X, Y): 3,450 µm | 3,450 µm
Principal point (X, Y): 768,105 px | 543,984 px
Image size: 1440,000 | 1080,000

***Fig. 72:** Calibration parameters, "Internal parameters" tab*

Parameter	Function
Image distance	Image distance determined from the calibration
Kappa	Calculated kappa value of the lens, display in micro kappa (x 10E-6)
Pixel pitch (X, Y)	Calculated grid / axis distance from pixel to pixel on the sensor. Reducing the resolution in the "Image acquisition" tab affects this parameter.
Coordinate origin (X, Y)	Puncture point of the optical axis through Measurement plane in the center of the sensor chip, opposite ideal center, in relation to upper left corner in pixels.
Image size	Image size in pixels

"External poses" tab



Camera to measurement coordinate system (CF_MF)

X: -3,174 mm
Y: 17,378 mm
Z: 513,860 mm
Angle X: 2,131°
Angle Y: 3,362°
Angle Z: -149,018°

Robot to measurement coordinate system (RF_MF)

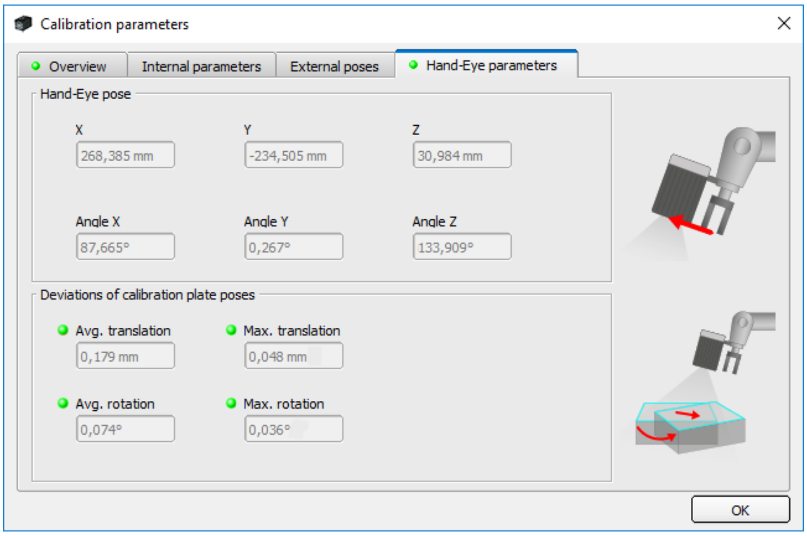
X: 282,493 mm
Y: -214,251 mm
Z: -40,923 mm
Angle X: -179,320°
Angle Y: 1,076°
Angle Z: -89,935°

Fig. 73: *Calibration parameters, "External poses" tab*

Note: The references in the following tabs are given in the rotation order that was selected when the calibration method was selected (Yaw-Pitch-Roll / Roll-Pitch-Yaw). Regardless of this, the order in which the values are specified is always (X, Y, Z, Angle X, Angle Y, Angle Z).

Parameter	Function
Camera- to Measuring coordinate system (CF_ MF)	Describes the 3D reference from Camera coordinate system (CF) to Measuring coordinate system (MF, determined by the calibration).
Robot- to Measuring coordinate system (RF_ MF)	Describes the 3D reference from the Robot Coordinate System (RF) that is active during calibration to the Measuring coordinate system (MF, determined by the calibration).
X Y Z	Translation values of the considered reference
Angle X Angle Y Angle Z	Rotation values (angles) of the reference under consideration: Rotation around X-/Y-/Z-axis

"Hand Eye Parameters" and "Base Eye Parameters" tab



Hand-Eye pose

X: 268,385 mm
Y: -234,505 mm
Z: 30,984 mm
Angle X: 87,665°
Angle Y: 0,267°
Angle Z: 133,909°

Deviations of calibration plate poses

Avg. translation: 0,179 mm
Max. translation: 0,048 mm
Avg. rotation: 0,074°
Max. rotation: 0,036°

Fig. 74: *Calibration parameters, "Hand-Eye parameters" tab*

Parameter	Function
Hand eye reference (TCP_CF) or base eye reference (RF_CF)	
X Y Z	Translation values of the Hand Eye or Base Eye reference
Angle X Angle Y Angle Z	Rotation values of the Hand Eye or Base Eye reference
Deviation Calibration plate poses	
Average translation	Average translation deviation, Root Mean Square Error (RMSE), calculated from the expected to the measured translations of the calibration plate poses (in user unit * 1000)
Maximum translation	Maximum translation deviation, calculated from the expected to the measured translations of the calibration plate poses (in user unit * 1000)
Average rotation	Mean rotation deviation, Root Mean Square Error (RMSE), calculated from the expected to the measured rotations of the calibration plate poses (in degrees * 1000)
Maximum rotation	Maximum rotation deviation, calculated from the expected to the measured rotations of the calibration plate poses (in degrees * 1000)

Interpretation of the deviation values: These deviations can be used as an indication of the approach accuracy of the robot, ...

- ...if the image acquisition positions during object acquisition are within the robot's movement range that was covered during calibration. If the image acquisition positions vary less, the deviations are smaller.
- ...if no result offset is used (if the approach point lies on the localization feature). The further the approach point is from the result point, the greater the deviations.
- ...if the Measurement plane corresponds to the calibration plane. The further away the Measurement plane is from the calibration plane, the greater the deviations.

8.1.6.6 Coordinate systems and transformations

The following figure shows the notations of the references used in the context of calibration.

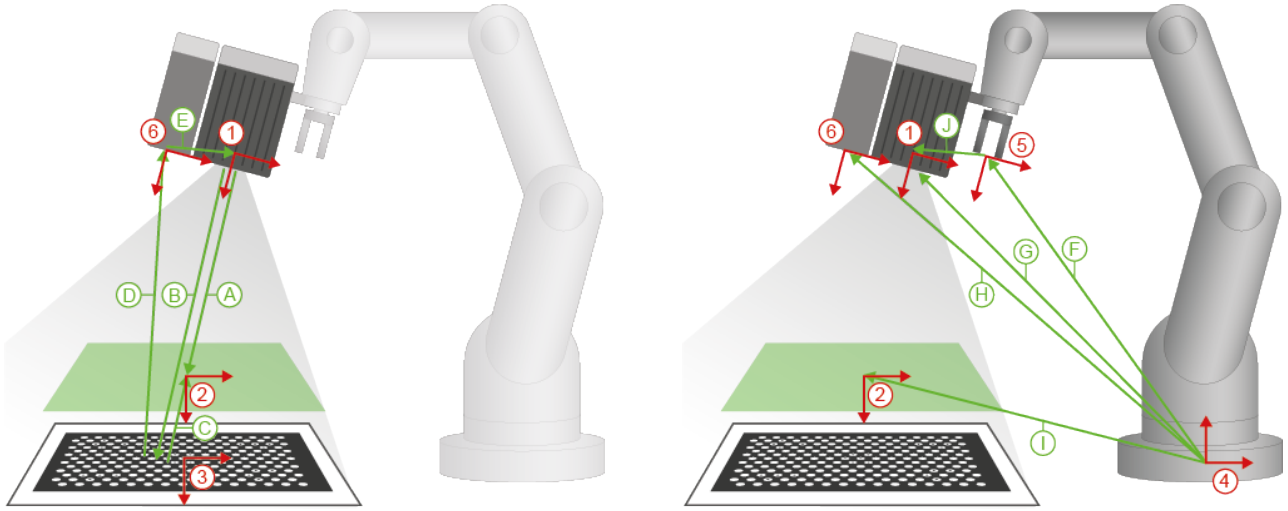


***Fig. 75:** References: "Robotics" Calibration methods*

(1)	Camera Frame (CF)	(A)	CF_MF	(F)	RF_TCP
(2)	Measurement Frame (MF)	(B)	CF_CPF	(G)	RF_CF
(3)	Calibration Plate Frame (CPF)	(C)	CPF_MF	(H)	RF_PF
(4)	Robot Frame (RF)	(D)	CPF_PF	(I)	RF_MF
(5)	Tool Center Point (TCP)	(E)	PF_CF	(J)	TCP_CF
(6)	Projector Frame (PF)				

Coordinate system	Description
(1) Camera Frame (CF)	Camera coordinate system Origin of Camera coordinate system lies inside the camera housing.
(2) Measurement Frame (MF)	Measuring coordinate system Measuring coordinate system is shifted parallel to Calibration Plate Coordinate System by the "Z-shift of Measurement plane" parameter.
(3) Calibration Plate Frame (CPF)	Calibration Plate Coordinate System Origin of Calibration Plate Coordinate System lies in the center of the calibration plate.
(4) Robot Frame (RF)	Robot Coordinate System The Robot Coordinate System is the coordinate system that is active during calibration or the base (depending on the manufacturer).
(5) Tool Center Point (TCP)	Tool Coordinate System (TCP)
(6) Projector Frame (PF)	Projector coordinate system (origin of an additional projector)

Reference	Description
(A) CF_MF	Reference: Camera coordinate system - Measuring coordinate system
(B) CF_CPF	Reference: Camera coordinate system - Calibration Plate Coordinate System
(C) CPF_MF	Reference: Calibration Plate Coordinate System - Measuring coordinate system

Reference	Description
(D) CPF_PF	Reference: Calibration Plate Coordinate System - Projector coordinate system
(E) PF_CF	Reference: Projector coordinate system - Camera coordinate system
(F) RF_TCP	Reference: Robot Coordinate System - Tool Coordinate System (TCP) This reference is visualized by most robots as "current position".
(G) RF_CF	Reference: Robot Coordinate System - Camera coordinate system
(H) RF_PF	Reference: Robot Coordinate System - Projector coordinate system
(I) RF_MF	Reference: Robot Coordinate System - Measuring coordinate system
(J) TCP_CF	Reference: Tool Coordinate System (TCP) - Camera coordinate system (corresponds to Hand-Eye reference).

8.1.6.7 Validation of a robotics calibration

A validation can be performed after a successful robot calibration to check whether the robot camera system still delivers the desired accuracy. For this purpose the calibration plate must not be moved between calibration and validation. Typically, it is screwed tight. Validation is only possible on the basis of requests. Depending on the calibration method, use the following requests:

Calibration plate (Robotics), Point pair list (Robotics)	CCD, CCP
Multi-image calibration	CCD, CMP
Hand-Eye calibration (Robotics), Base-Eye calibration (Robotics)	CCD, CAI, CRP

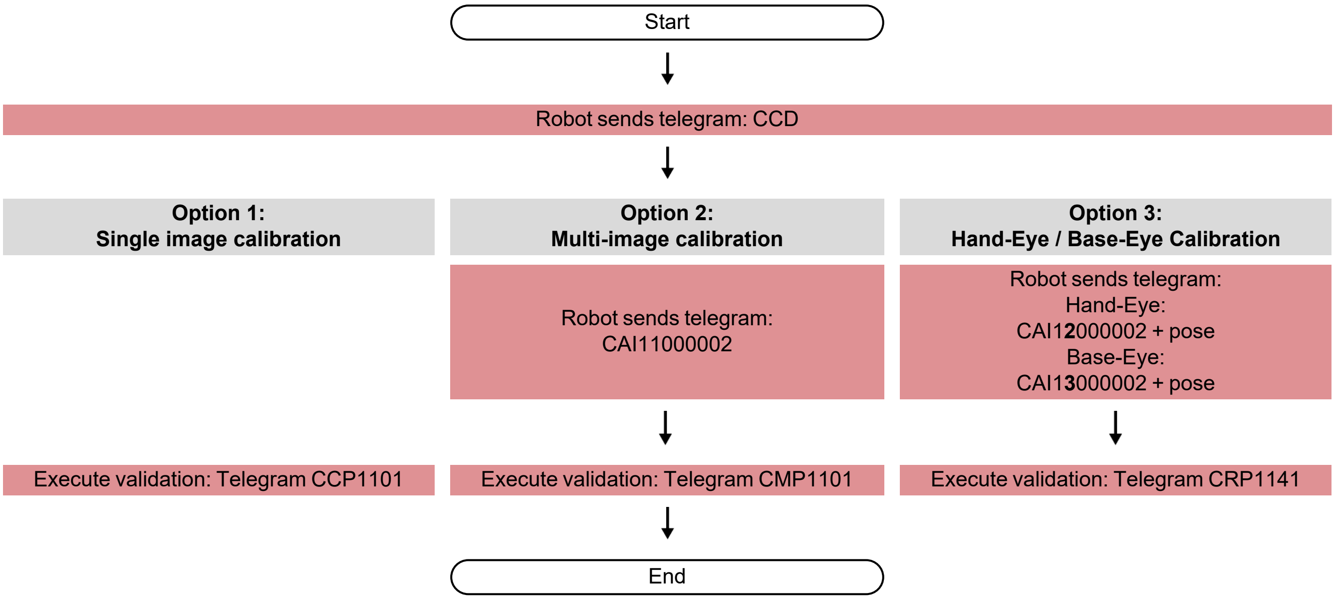
Additional information: see Communications manual

Select the option "Validation of the calibration" in the request CCP or CRP.

Procedure

1. Move the camera to the position above the calibration plate where the Measurement plane was set.
2. Call up the request (sequence) according to your selected calibration method. In the request-response (CCP or CRP), consider the mean deviation (RMSE).
   When mounted on the gripper, this deviation is typically higher than that of the calibration, since it includes the accuracy of the robot positioning.

Request sequences for validation



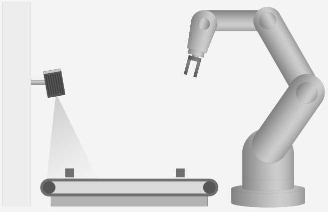
The deviations can be interpreted as follows:

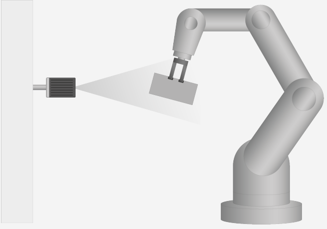
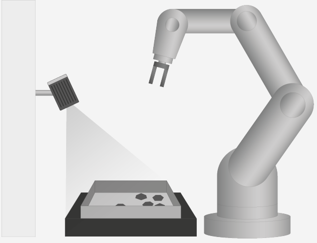
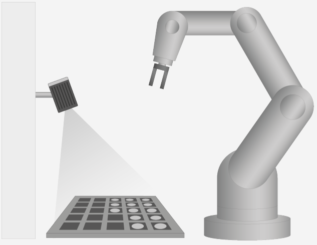
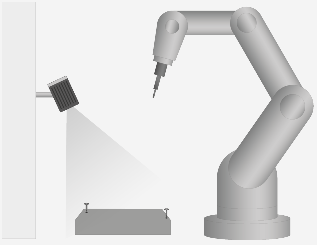
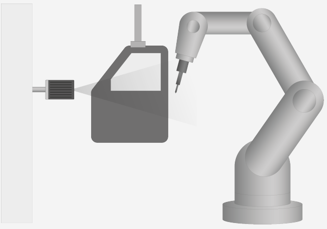
- The deviation is specified in pixels*1000, as this is a relative value to the field of view and therefore independent of the actual working distance of your application. If you divide the deviation value by the pixel resolution of your camera, you get the relative deviation of the field of view.
- With the request CGP it is possible to retrieve the average customer unit/pixel and thus convert the pixel value into the user unit, e.g. millimeter (parameter number 041).

As a guide value for mounting on the gripper, 0.4% deviation of the field of view is a realistic limit value above which a new calibration is recommended. Please check for yourself whether this applies to your application or whether you have to adapt it. For a stationary installation, the recommendation would be to set the limit value below 0.4% deviation of the field of view.

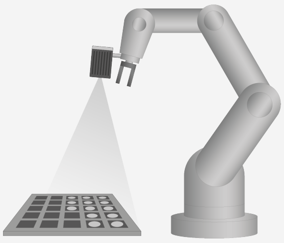
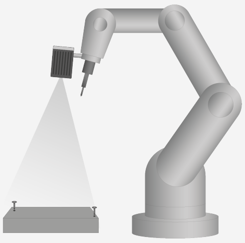
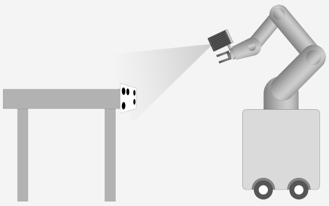
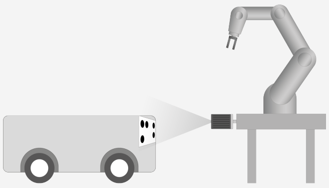
8.1.6.8 Application-specific calibration recommendations

Applications: VISOR® Vision Sensor stationary

Application example	Recommended calibration procedure
Part is moved by robot (pick and place)	
 Picking from a conveyor belt	Calibration plate (Robotics) Multi-image calibration: Performing the calibration • in SensoConfig • via telegram (see Communications manual) Base-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)

Application example	Recommended calibration procedure
 Fine positioning in gripper	Base-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)
 Picking from a vibration feeder	Calibration plate (Robotics) Multi-image calibration: Performing the calibration • in SensoConfig • via telegram (see Communications manual)
 Picking from a load carrier	Calibration plate (Robotics) Multi-image calibration: Performing the calibration • in SensoConfig • via telegram (see Communications manual) Base-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)
Part is processed by robot (screwing, gluing)	
 Automated screw insertion	Calibration plate (Robotics) Multi-image calibration: Performing the calibration • in SensoConfig • via telegram (see Communications manual) Base-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)
 Glue bead application	Base-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)

Applications: VISOR® Vision Sensor in motion

Application example	Recommended calibration procedure
Part is moved by robot (pick and place)	
 Picking from a load carrier	Hand-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)
Part is processed by robot (screwing, gluing)	
 Automated screw insertion	Hand-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)
Mobile workstation is calibrated	
 Calibration of mobile robots	Hand-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)
 Calibration of driverless transport systems	Base-Eye calibration (Robotics): Performing the calibration • in SensoConfig • via telegram (see Communications manual)

8.1.7 Cycle time tab

The Cycle time tab is used to configure the time response of the VISOR® vision sensor.

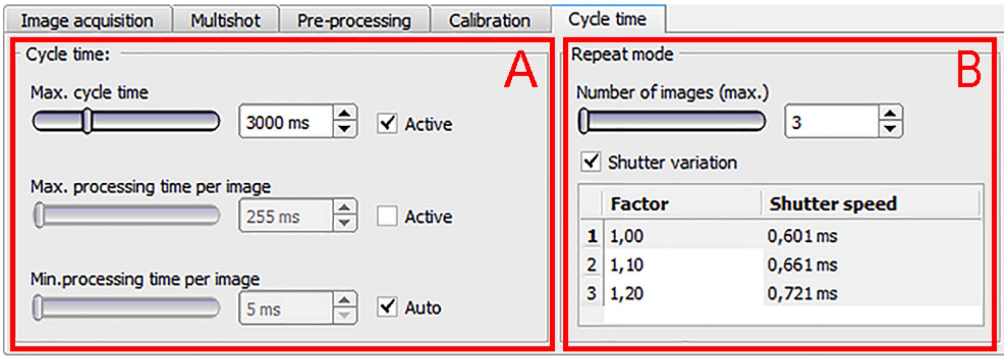


Image acquisition | Multishot | Pre-processing | Calibration | Cycle time

Cycle time:

Max. cycle time 3000 ms ☑ Active

Max. processing time per image 255 ms □ Active

Min.processing time per image 5 ms ☑ Auto

Repeat mode

Number of images (max.) 3

☑ Shutter variation

	Factor	Shutter speed
1	1,00	0,601 ms
2	1,10	0,661 ms
3	1,20	0,721 ms

***Fig. 76:** Setup Job, tab Cycle time*

A: Cycle time

The cycle time measures the time from the trigger to the setting of the digital switching outputs. If the cycle time should be limited, e.g. because the machine cycle must not be exceeded, the value for the maximum cycle time must be limited accordingly. The result of all unfinished detectors up to this time is set to faulty. When choosing the maximum cycle time, it must be taken into account that this is not adhered to strictly. However, depending on the detector that has just been executed, it can take several milliseconds before it can break off. It is recommended that this maximum cycle time over the actual execution time is checked and the set maximum cycle time is reduced accordingly.

Parameter	Function
Max. cycle time	Parameter for controlling the execution time of a cycle. Inside a cycle, some images can be evaluated (in case of "Number of images" >1). Maximum execution time is used to abort a cycle after a defined time. The result of the cycle after a timeout is always "not OK". The maximum cycle time should always be greater than the time required for a complete evaluation.
Max. processing time per image	Maximum duration of one evaluation inside a cycle including image acquisition
Min. Processing time per image	Minimum duration of one evaluation inside cycle including image acquisition. The minimum processing time can be used to suppress multiple triggers. In case of "Number of images" = 1 (default) the "Min. Processing time per image" corresponds to the minimum cycle time.
Auto	The "Auto" switch sets the "Min. Processing time per image" so that the LED strength is always 100% and the processing time is minimal.

B: Repeat mode

Parameter	Function
Number of images (max.)	Maximum number of frames taken after a trigger, if none of the following abort criteria is met: • "Job result" = positive (adjustable under "Output" / "Digital output") • "Max. cycle time" is not fulfilled (if activated). Optional: • Assign detector to an image, see also: Multiple image capture: Allocate the detector to an image

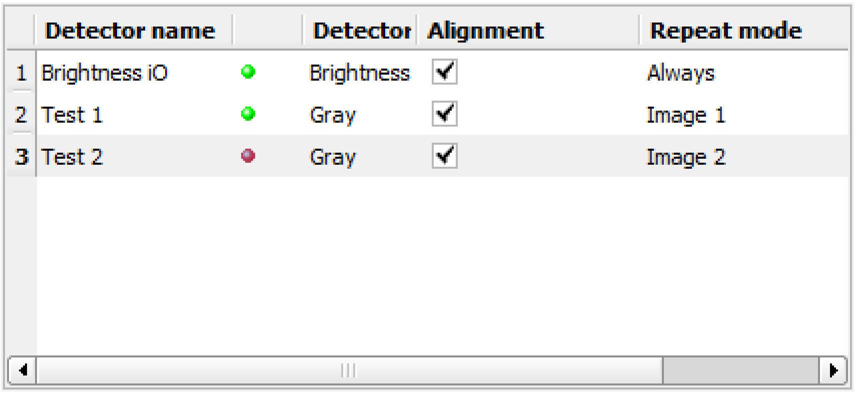
Parameter	Function
Shutter variation	If the shutter speed variation is "Active", a variation of several different shutter speeds can be created across a table. One image is then captured per set shutter speed, i.e. the first image is captured at shutter speed 1, the second image at shutter speed 2, etc. Default of the "Shutter variation" is "Off". In this case, the listbox is not displayed.
Factor and Shutter speed	The default value for the factor is: First value = 1.00 (the first value is always identical to 1.00 and is read-only). Subsequent default values each increase by 0.1, e.g. 1.10, 1.20, etc. The user can change the factor in the table, automatically adjusting the shutter speed (second column, read-only) and taking a picture. A mouse click in a row of the table will take a picture with the settings of the clicked table row. **NOTE:** If the parameter "Shutter speed" in the "Image acquisition" tab is changed, the shutter speed in the "Shutter variation" list box is recalculated.

Multiple image capture: Allocate the detector to an image

In the "Detector" setup, all set detectors are listed. If the parameter "Number of images (max.)" of the multiple image acquisition is greater than 1, you get the option to assign a detector to an image acquisition. In the column "Repeat mode", this setting can be made for each detector.

- Always: Run for all image acquisition operations
- Recording n: Executed in the corresponding image acquisition

Open the selection table by double-clicking.



	Detector name		Detector	Alignment	Repeat mode
1	Brightness iO	●	Brightness	☑	Always
2	Test 1	●	Gray	☑	Image 1
3	Test 2	●	Gray	☑	Image 2

Fig. 77: Detector list, multiple image acquisition

Min. Processing time per image

When using the internal illumination, the minimum cycle time depends on the set exposure time (longer => higher min. cycle time).

8.2 Setup Alignment

For objects or features whose position varies in the image, Alignment may be useful or necessary. The Alignment determines the object / feature position in the image. Three different detection methods (alignment detectors) are available for this purpose: Pattern matching, Edge detector, and Contour matching.

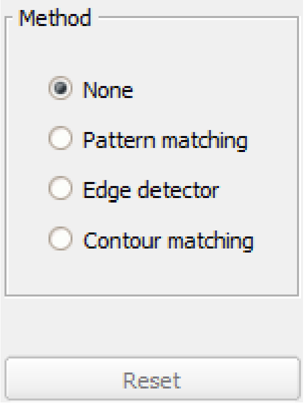
Alignment functionality:

An Alignment is an aligned coordinate system anchored to a selected feature. Defined detectors are alignment relative to this coordinate system. The aligned coordinate system is drawn in dark blue (for information on the meaning and adjustment of the different frames: see Search and feature ranges).

NOTE:

- A maximum of one alignment detector can be defined for each job.
- For each detector in the job, it can be selected whether the detector should be aligned with the Alignment.
- Since the Alignment is an additional calculation step and thus makes use of cycle time, it should only be used if the application requires it.

8.2.1 Selection and configuration of Alignment


Method

● None

○ Pattern matching

○ Edge detector

○ Contour matching

Fig. 78: *Selecting Alignment*

1. Click on the setup button "Alignment".
2. Select a detection method in the configuration window "Method".

Method	Selection criteria
None	Alignment not active
Pattern matching	Detection of contours and edges at any angle. Pattern matching can be used preferably if: • There are no / few high-contrast or paraxial edges, but areas with gray value patterns are present in the image • an angular offset (rotation offset against teach-in position) of up to 360° can occur.
Edge detector	The detection of edge should always be selected: • if an offset of the position in X and / or Y direction can occur. • at a maximum angle offset (rotational offset compared to the teach-in position) of approx. ± 20° (depending on the object and application). • if there are edges with strong contrast If the above-mentioned criteria are fulfilled, Edge detector is a very quick method of Alignment.

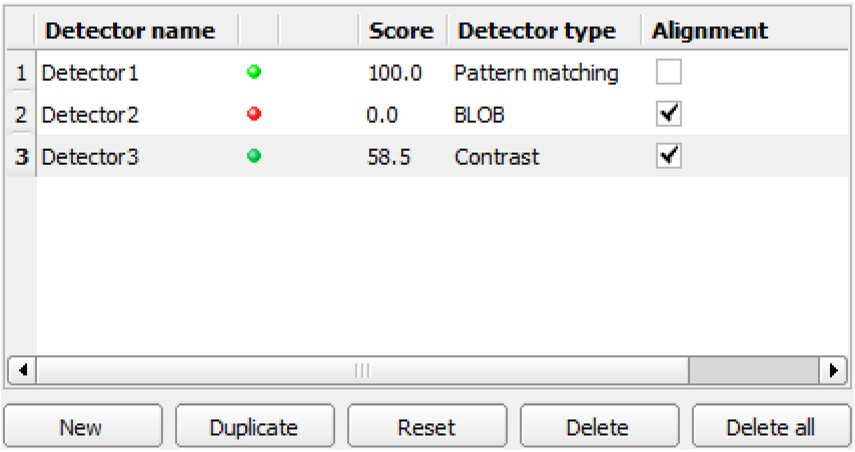
Method	Selection criteria
Contour matching	Detection of contours and edges at any angle Contour detection must always be used if: • An angular offset (rotation offset against teach-in position) of up to 360° can occur. It can be used preferably when high-contrast edges are present in the image. The relatively complex contour detection function usually results in a comparatively longer cycle time.

Configuration of alignment detector

1. Adapt the position and size of the search and function ranges displayed on the screen if necessary.
2. Configure the alignment detector in the Parameters tab.

Set Alignment active for detectors

In the "Detectors" setup, all set detectors are listed. In the "Alignment" column, it is possible to select for each detector whether it should be aligned by the adjusted Alignment. Default value is "Active".



	Detector name			Score	Detector type	Alignment
1	Detector1	🟢		100.0	Pattern matching	□
2	Detector2	🔴		0.0	BLOB	☑
3	Detector3	🟢		58.5	Contrast	☑

Fig. 79: Detector list, Alignment active / inactive

Reset

The "Reset" button can be used to restore the factory settings for the selected alignment detectors.

8.2.2 Alignment Pattern matching

This method is suitable for detecting patterns of any shape, even without clear edges or contours The patterns of the object in the search field are stored on the sensor during teach-in. In run mode, the sensor searches for the position of the greatest match with the taught pattern in the current image. If the match is greater than the set threshold value, the pattern is considered recognized and its position is used as the origin of the coordinate system of the Alignment. The pattern recognition is completely rotational tolerant, i.e. the searched object may appear in any position in the image (choose angle setting accordingly).

8.2.2.1 Color Channel tab

In the Color Channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray scale value image on a monochrome VISOR® vision sensor, contrasts can be increased significantly. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object recognition detectors: Monochrome image, display dependent on the selected color space and the color channels

Parameter description:

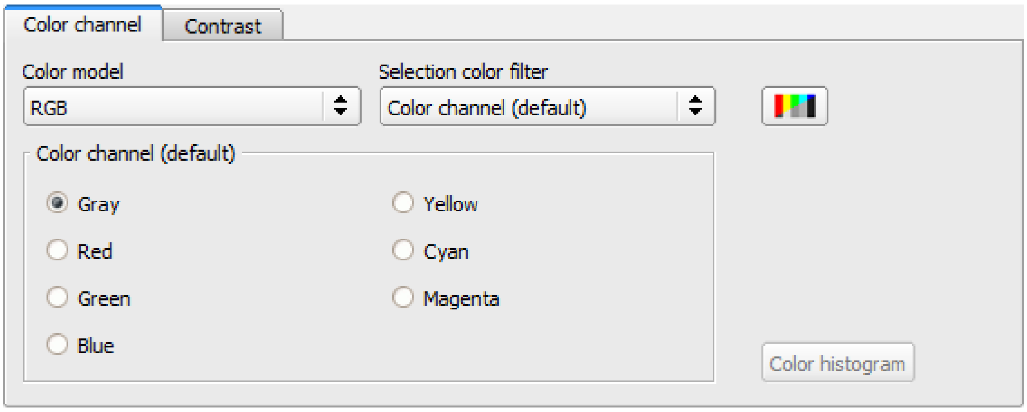
Parameter	Function
Color space	Color spaces: RGB, Color model RGB HSV, Color model HSV LAB, Color model LAB
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
🟥🟩⬛	Switching the image between color and monochrome.

8.2.2.1.1 Selecting a color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color model: RGB

Selection color filter: Color channel (default)

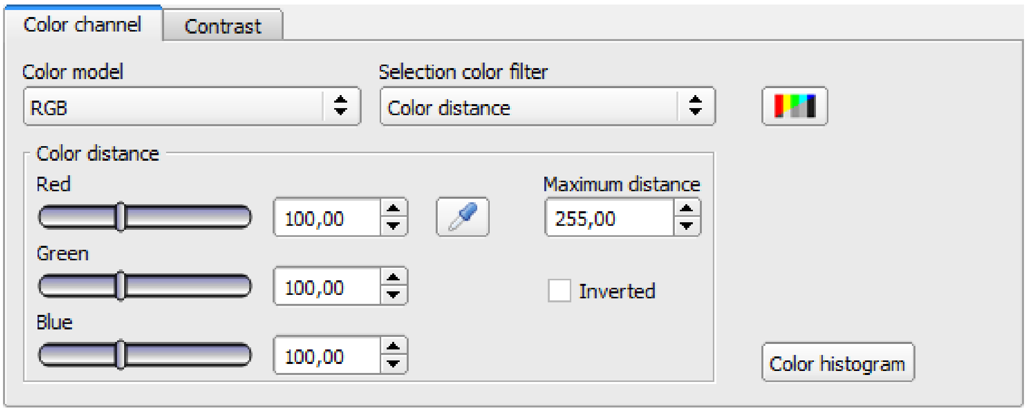
Color channel (default)

● Gray ○ Yellow
○ Red ○ Cyan
○ Green ○ Magenta
○ Blue

Fig. 80: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of letters for OCR.



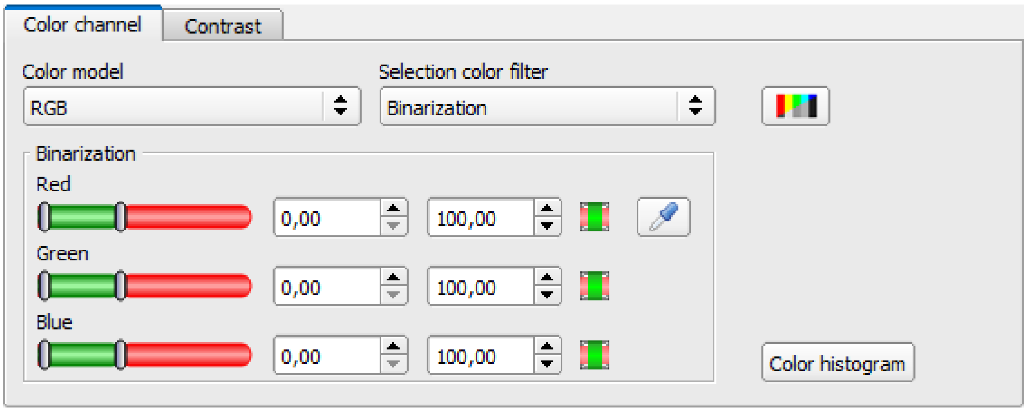
Color model: RGB
Selection color filter: Color distance
Color distance
Red: 100,00
Green: 100,00
Blue: 100,00
Maximum distance: 255,00
□ Inverted

***Fig. 81:** Color filter, Color distance*

Parameter		Function
Red Green Blue	Luminance A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette symbol		With the selection of the pipette button and then clicking on the image, the selected color channel is determined automatically.
Maximum color distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of "Inverted".
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB
Selection color filter: Binarization
Binarization
Red: 0,00 100,00
Green: 0,00 100,00
Blue: 0,00 100,00

***Fig. 82:** Color filter, Binarization*

Parameter			Function
Red Green Blue	Shade (hue) Saturation Brightness (value)	Luminance A B	Determination of the color range. The color ranges can be user-defined via the slider or set by entering a value.
Invert button 🟩/🟥			The current setting is inverted when selecting the button.
Pipette symbol			With the selection of the pipette button and then clicking on the image, the selected color channel is determined automatically.

8.2.2.2 Parameters tab

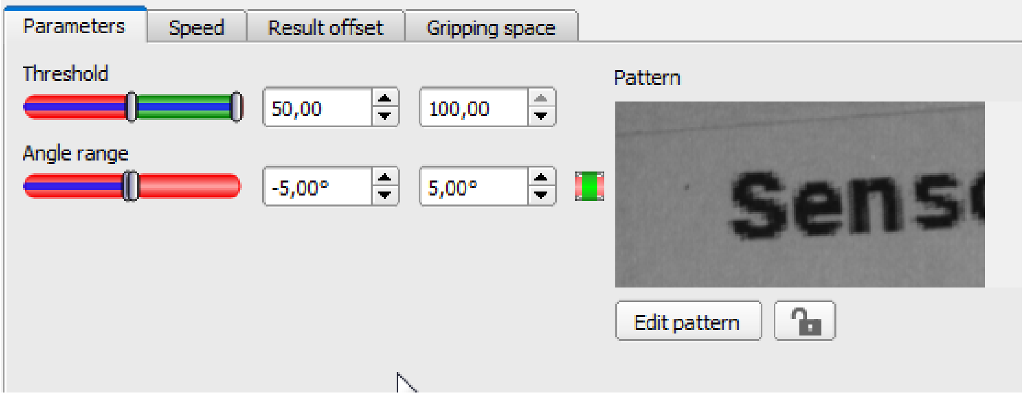


Fig. 83: Alignment Pattern matching, Parameters tab

The following parameters can set in the "Parameters" tab:

Parameter	Function
Threshold	Range for the required concordance of the found sample with the taught sample
Angle range	Angle range in which to search (larger range leads to longer processing times). Depending on the size and complexity of the image, the Angle range may be limited.
Pattern	Shows the taught pattern (red frame in the field of view)
Edit pattern	By masking the pattern, areas of the taught-in pattern can be deactivated (see also Function: Edit pattern / contour)
Lock 🔓	Lock / unlock pattern. When locked, the taught-in pattern is protected against (unintentional / accidental) changes, e.g. accidental adaptation of the teaching range. Unlock to modify taught pattern.

8.2.2.3 Speed tab

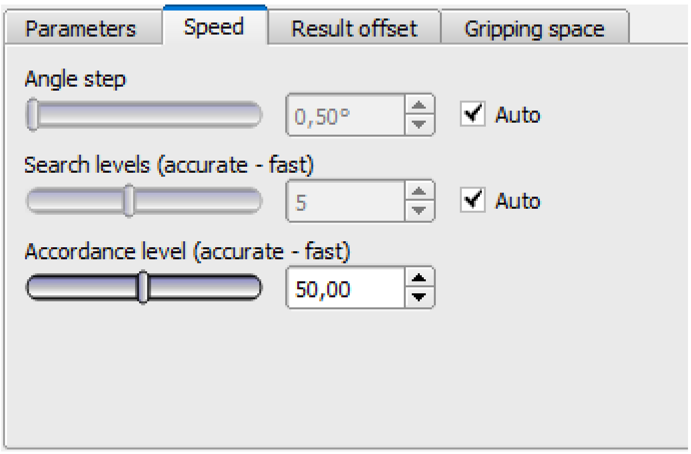


Fig. 84: Alignment, Pattern matching, Tab Speed

The execution speed is influenced by the adjustable speed parameters. The search is performed either less finely, i.e. earlier canceled and thus faster, or even finer details are taken into account in the search, i.e. search longer and the search is slower.

Parameter description:

Parameter	Function
Angle step	Sensitivity of the search throughout the selected angle range in degrees [°] (smaller value leads to longer machining times, but higher accuracy)
Search levels (accurate - fast)	Number of search levels (one search level corresponds to an image with half resolution) • Small value (accurate): Slow search = lower risk (less likely to overlook candidates) • High value (fast): Fast search = higher risk (candidates can be overlooked)
Accordance level (accurate - fast)	Candidates with a degree of compliance below the specified value are already discarded during the search. • Small value (accurate): Late rejection = slower = less risky • High value (fast): Early rejection = quicker = riskier In case of false results, this value can be decreased (more accurate).
Auto	Automatic setting

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

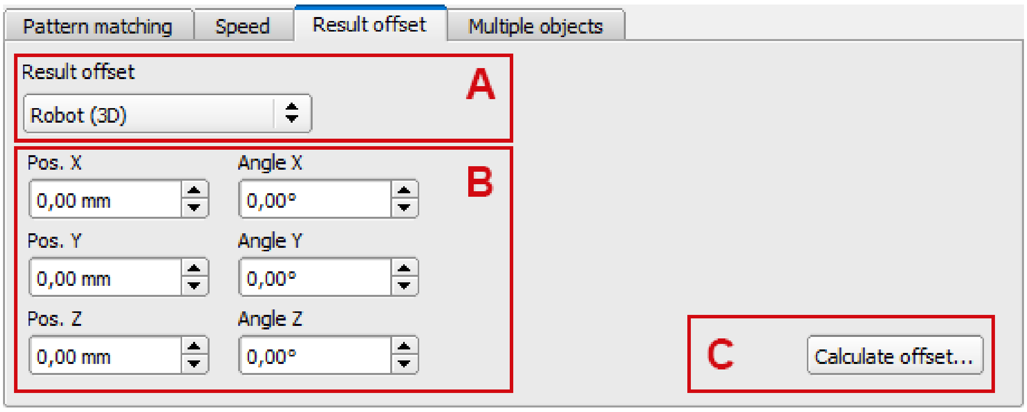
8.2.2.4 Result offset tab

The Result offset function can be used, on the one hand, to define a gripping point that deviates from the detected position. On the other hand, this offset can be used to define a zero position.

Positions detected with the VISOR® vision sensor contain either 2D values (X,Y, angle Z) or 3D values (pos. X, Y, Z, Angle X, Y, Z). The Result offset function can be used to add fixed offset values to these values.

Depending on the detector, different options are available.

NOTE:
If several objects are found (for BLOB detector or Multiple objects), the Result offset refers to the first result in the result list.



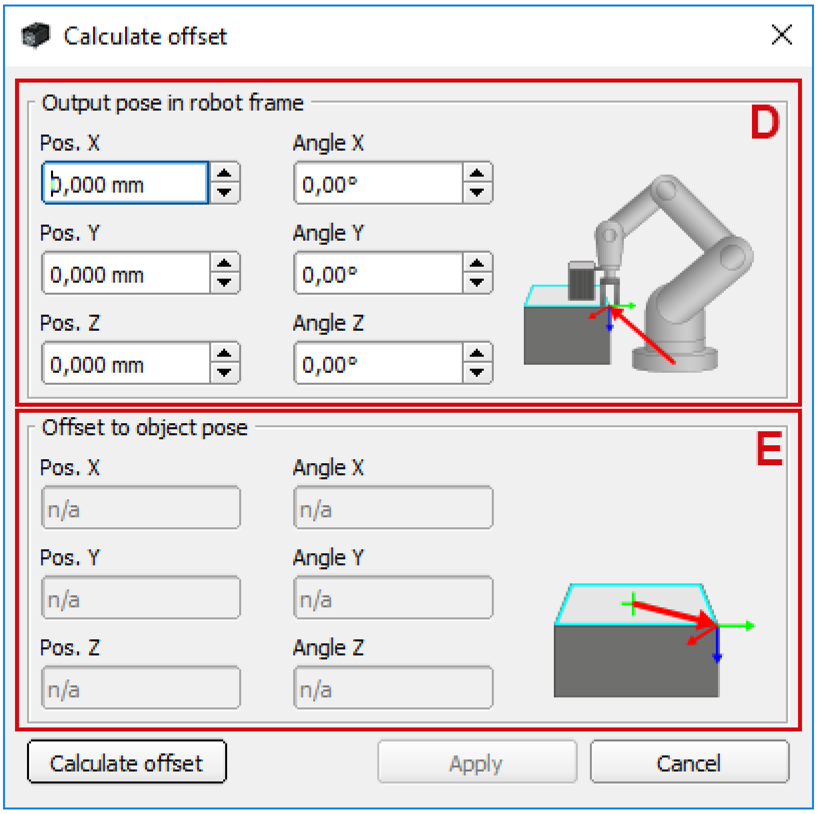
Result offset: Robot (3D)

Pos. X: 0,00 mm
Angle X: 0,00°
Pos. Y: 0,00 mm
Angle Y: 0,00°
Pos. Z: 0,00 mm
Angle Z: 0,00°

***Fig. 85:** Result offset tab, Option "Robot (3D)"*

Parameter description:

Parameter	Function
A: Offset selection based on the values and the coordinate system	
Off	No offset, i.e. automatically determined center of the found object / finder region.
Image plane (in pixels)	Values in the Image Coordinate System: Pos. X, Y, Angle Z
Align (2D)	Values in the Robot Coordinate System: Pos. X, Y, Angle Z
Robot (3D)	Values in the Robot Coordinate System: Pos. X, Y, Z, Angle X, Y, Z
B: Direct value input of the absolute offset in customer unit and in degrees. Note: If "Image plane (in pixels)" is selected, the input can also be made graphically.	
Pos. X	Offset in X direction
Pos. Y	Offset in Y direction
Pos. Z	Offset in Z direction
Angle X	Rotation around X-axis
Angle Y	Rotation around Y-axis
Angle Z	Rotation around Z-axis
C: Opens dialog "Calculate offset" Function for calculating the offset based on a given gripping point / zero position.	
Dialog "Calculate offset" (only for Align (2D) and Robot (3D))	The starting point is the pose that will eventually be achieved by the robot and thus output by the detector or the Alignment. After the robot gripper has been moved to the desired position, this pose can be read by the robot controller and entered as "Output pose in robot frame". The "Calculate offset" button is then used to calculate the result offset that leads to this desired pose. The calculation can only be performed if an object has been detected.



Calculate offset

Output pose in robot frame

Pos. X: 0,000 mm
Angle X: 0,00°
Pos. Y: 0,000 mm
Angle Y: 0,00°
Pos. Z: 0,000 mm
Angle Z: 0,00°

Offset to object pose

Pos. X: n/a
Angle X: n/a
Pos. Y: n/a
Angle Y: n/a
Pos. Z: n/a
Angle Z: n/a

***Fig. 86:** Dialog Calculate offset*

D: Data input, e.g. gripping point

E: Calculated value (read-only)

8.2.2.5 Gripping space tab

Robots grip objects, e.g. with a jaw gripper, on the outer contour of the objects. If the objects touch or overlap, gripping with the robot may not be possible. The VISOR® gripping space check can be used to check whether the gripping positions on the object are actually clear in the required size.

The gripping space check is available for Contour matching and Pattern matching Alignment.

The VISOR® gripping space check is an extension of the Alignment. For an Alignment **without** gripping space check, the position of the object that has the highest score value is output. **With** an active gripping space check, the position of the first object found is output, in which its tracking detectors (gripping ranges) are also "OK" (according to the logical links in the overall result).

If the gripping space check is to be performed for several objects, individual results can be selected from the list in the Result tab in order to optimize the detectors and regions of interest. Recommended setting therefor: Trigger / Image update = "Single"

Procedure:

1. The Alignment Contour matching / Pattern matching identifies objects as candidates whose contour / pattern matches the taught contour / pattern.
2. These candidates are sorted. The sorting takes place according to the set values for "Sorting criteria" and "Sorting order" in the "Gripping space" tab.
3. In this order, the candidates are checked to see if the detectors aligned by the Alignment (e.g. checking for gripping space) are all met. This happens under consideration of the logical links in the overall result.

In setup step "Output" / "Digital output" tab, logical links can be made to evaluate the objects. Here, for example, free spaces for different gripping positions can be defined (see figure below).

4. The position data of the candidates that meet all the criteria are displayed in the list and can be output via the Telegram tab. For all other detectors, only the data of the first object in the list is output.

NOTE:
If several objects are found, by default these will all be output ("No. of results": "0"). If only one result is to be output, the "No. of results" must be set to "1" under Output / Telegram / Payload.

The gripping positions X-X and Y-Y are possible for the object shown in the following figure. Of these gripping possibilities, only those that are actually necessary for one grip can then be checked for "free".

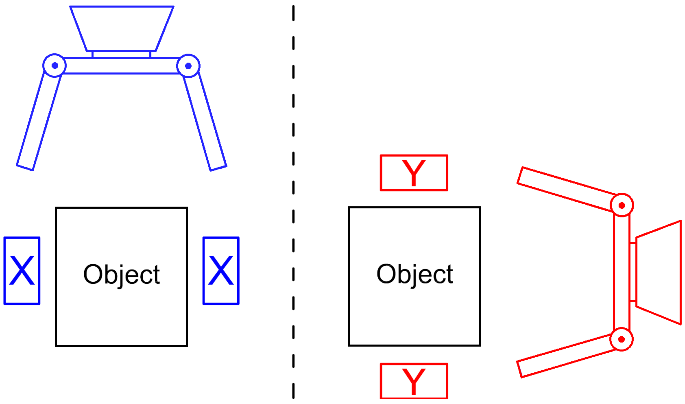
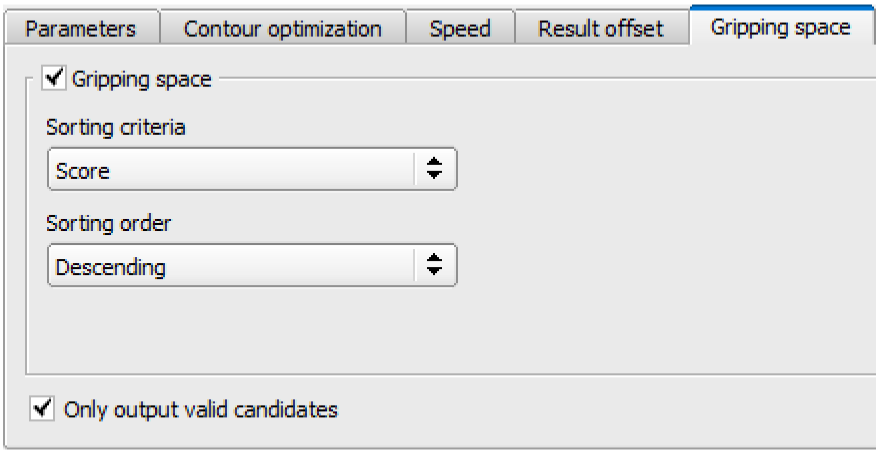


***Fig. 87:** Possible gripping position X-X (left) and possible gripping position Y-Y (right).*

NOTE:
In the detector list, Alignment per detector can be activated or deactivated (default: active). Only detectors activated here are effective for the gripping space check.

Prerequisite for successfully finding an object is at least one object per image / evaluation, where the overall result is "OK", i.e. also tracked detectors!



☑ Gripping space

Sorting criteria: Score

Sorting order: Descending

☑ Only output valid candidates

***Fig. 88:** Alignment Contour matching, Gripping space tab*

The following parameters can be set in the "Gripping space" tab:

Parameter	Function
Sorting criterion • Score • Position X • Position Y • Angle Z • Scaling	Sorting criteria according to which objects should be "pre-sorted". • Score value • X-position • Y-position • Angle Z • Scaling (only for Alignment Contour matching)
Sorting order • Ascending • Descending	Sorting order for the selected sorting criterion. The values of the sorting criteria are sorted in ascending order. The values of the sorting criteria are sorted in descending order.
Only output valid candidates	If this checkbox is activated, only objects whose score value is above the set threshold ("Parameters" tab) are displayed and output. It can be used e.g. for parameter optimization.

8.2.3 Alignment Edge detector

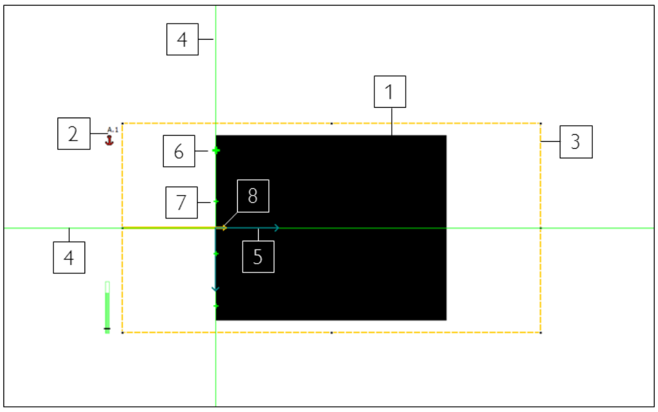
This Alignment determines the object position and thus the aligned coordinate system based on the intersection point of edges in the image. Angle positions up to approx. ±20% deviation (depending on the object) can be compensated.

8.2.3.1 Structure of the Edge detector

The "Edge detector" is carried out using "Probes". Depending on the probe type, there are between one and three probe(s). The region of interest of a probe is indicated by the yellow frame (ROI). Within this ROI, the object is searched for and the edge of the object is scanned. The scanning is performed in the direction of the yellow arrow, the "Search direction". This yellow arrow can also be used to turn the region of interest of the detector.

From the starting point of the region of interest, search stripes (number can be set as desired) are sent out into the search direction. If the search stripe touches the edge of the object, the "Touching point" of the search stripe is marked with a cross at this point. Depending on the number and the setting, there may be a "Winner search stripe", the touching point of which is shown in bold.

Which edge of an object is probed can be seen on the "Scanning line" in the search direction. If an object is not scanned from both the X and Y direction but only from one direction, the second scanning line is at the center of the region of interest. The arrows with the origin at the intersection of the scanning lines form the aligned coordinate system. In the following figure, the structure of the Alignment "Edge detector" is visualized.



[1] Object
[2] Probe 1
[3] region of interest of probe 1
[4] Scanning line
[5] Tracked coordinate system
[6] Touching point of the winning search stripe
[7] Touching point of a search stripe
[8] Search direction of the probe

Fig. 89: *Structure of the Edge detector*

8.2.3.2 Color Channel tab

See Chapter: Color Channel tab

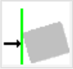
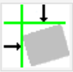
8.2.3.3 Parameters tab

The probe type must be selected to perform the Edge detector. The probe type determines which change in position of the object can be aligned: Shift in one or two directions, rotation. The following probe types are available and recommended for use with varying object positions ...

① ...in one direction

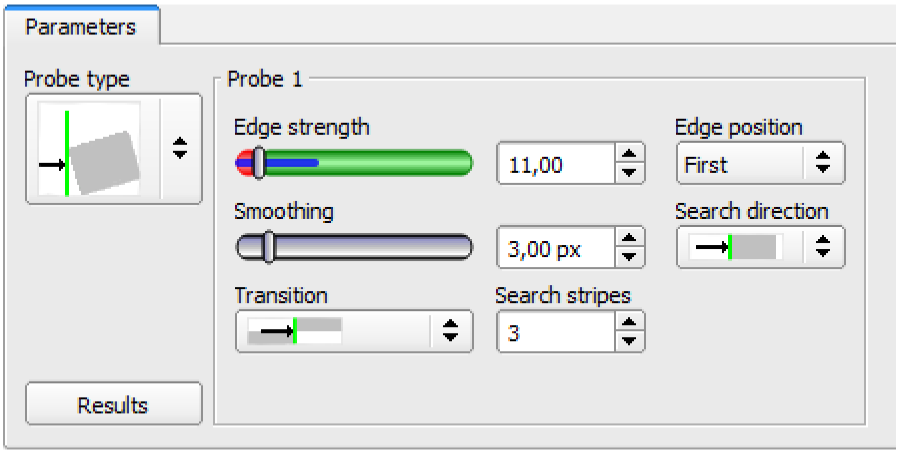
② ...in two directions

③ ...with rotation

Probe type	Function	①	②	③
1 	**One probe: Alignment for shift in one direction** Object position is aligned when moving in one direction. The position of the scanning line is determined by the probe's scanning direction. The other scanning line is in the middle of the region of interest (ROI). **NOTE:** A rotation of the object position is not aligned.	✓		
2 	**One probe: Alignment for shift in one direction and rotation** Alignment of object position by shift in one direction and rotation. The position of the scanning line is determined by the probe's scanning direction. The other scanning line is in the middle of the region of interest (ROI).	✓		✓
3 	**Two probes: Alignment for shift in two directions** Object position is aligned when moving in two directions. The position of the scanning line in X direction of the coordinate system is determined by Probe 1. The position of the scanning line in Y direction of the coordinate system is determined by Probe 2. The origin of the coordinate system lies at the intersection of both scanning lines. **NOTE:** A rotation of the object position is not aligned.		✓	
4 	**Two probes: Alignment for shift in two directions and rotation** Object position is aligned by shift in two directions and rotation. The position of the scanning line in X direction of the coordinate system is determined by Probe 1. The position of the scanning line in Y direction of the coordinate system is determined by Probe 2. The origin of the coordinate system lies at the intersection of both scanning lines. In addition, the orientation of the object is determined. Probe 2 is rotated and moved according to the object movement. The position of Probe 2 is aligned relative to the position and orientation of the scanning line of Probe 1.		✓	✓

Probe type	Function	①	②	③
5	**Three probes: Alignment for shift in two directions and rotation** Object position is aligned by shift in two directions and rotation. A straight line is drawn through the touching points of the winner search stripes of Probes 1 and 2. This scanning line (12) determines the position and orientation of the coordinate system. The origin of the coordinate system lies at the intersection of Scanning line 12 and Scanning line 3. Probe 3 is rotated and moved according to the object movement. The position of Probe 3 is aligned relative to the position and orientation of Scanning line 12.		✓	✓

After selecting the probing mode, the corresponding parameters must be determined.
The following parameters can set in the "Parameters" tab:



Edge strength: 11,00
Smoothing: 3,00 px
Edge position: First
Search stripes: 3

Fig. 90: Alignment Edge detector, Parameters tab

Parameter	Function
Edge thickness	Edge thickness / contrast at which an edge should be detected as an edge.
Smoothing	The edge contour is smoothed in the search direction. With larger values, noisy edges, blurred edges, or edges that are not perpendicular to the search direction, are detected more reliably. In addition, light-dark-light or dark-light-dark transitions which are close together with larger values can be ignored. Thus, interfering edges, e.g. scratches, can be hidden. The effect of smoothing can be displayed graphically using the button "Results".
Transition	With the "Transition" parameter, the edge transition can be determined.
Both directions	Edge transition from light to dark and vice versa.
Light → dark	Edge transition from light to dark.
Dark → light	Edge transition from dark to light.
Search stripes	Number of parallel search stripes into which the width of the region of interest is divided. Edge detector is carried out in each search stripe, and the first edge is decisive.

Parameter	Function
Edge position	The parameter "Edge position" determines which edge should be detected from the search direction. It is determined how the winner search stripes and thus the edge position are determined.
• First	The first edge in the search direction is detected. The distances from the beginning of the region of interest to the touching points of all search stripes in the search direction are determined. The winner search stripe is the one with the shortest distance to the beginning of the region of interest.
• Last	The last edge in the search direction is detected. The distances from the beginning of the region of interest to the touching points of all search stripes in the search direction are determined. The winner search stripe is the one with the longest distance to the beginning of the region of interest.
• Median	The distances from the beginning of the region of interest to the touching points of all search stripes in the search direction are determined. The median value of these distances is then formed.
• Mean	The distances from the beginning of the region of interest to the touching points of all search stripes in the search direction are determined. The mean of these distances is then formed.
Orientation	The "Orientation" parameter defines the type of the scanning line determination.
• Best-fit line	In this setting, the scanning line is determined by placing a best-fit line through all search stripes.
• Edge guide	In this setting, a scanning line is determined, which acts like a mechanical edge stop. This makes it possible to achieve more robust results for convex-shaped edges than with a simple best-fit line.
Search direction	This parameter determines the search direction of the probes. From this direction, the object edge is aligned. All probes can be rotated with the small black arrow.
→|	The search direction takes place in only one direction: the direction of the yellow arrow (ROI). The touching points and thus the origin of the coordinates lie at an edge of the object.
→|←	For each search stripe, a touching point is determined from both directions of the probe. The center between these touching points is then determined. The origin of the coordinate system is at the center of the winner search stripe, i.e. in the object.
Results	Opens the results and histogram window. For more information, please refer to: Results window

Improvement of execution speed

- Region of interest for position (yellow frame) only as large as necessary.
- Reduce search stripes
- Reduce smoothing value
- Reduce resolution from SVGA to QQVGA, QVGA or VGA

⚠ **ATTENTION:**
This parameter affects all detectors!

Robust detection

- If edges are blurred: Increase smoothing value
- If interfering edges such as scratches are detected: Increase switching threshold or / and smoothing value
- If edge is not perpendicular to search direction: Increase search stripes

8.2.3.4 Result offset tab

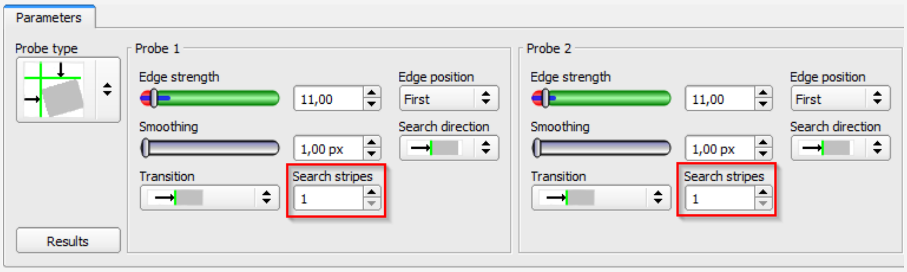
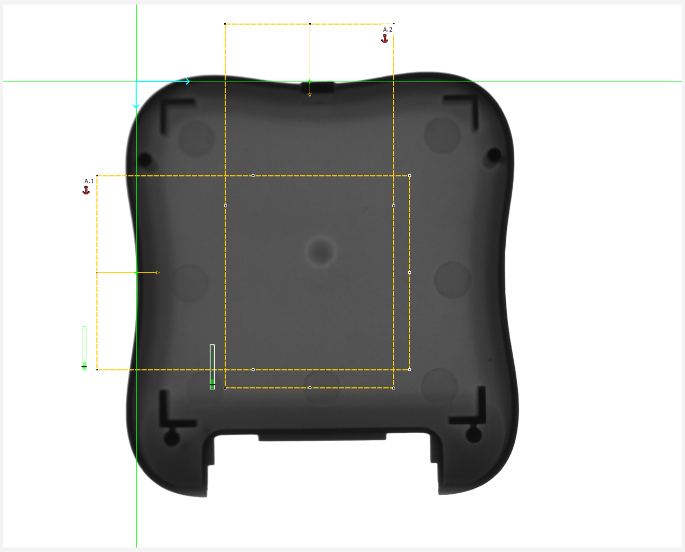
See Chapter: Result offset tab

8.2.3.5 More information on Edge detector (Alignment)

8.2.3.5.1 Effect of number of search stripes

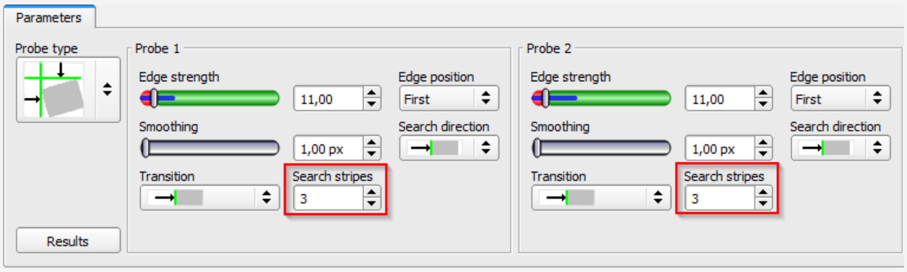
"Search stripes" represent how many parallel search stripes the region of interest is divided into. Edge detector is performed separately in each search stripe. The first edge in the search direction of all search stripes is considered to be the overall result. Increasing the value of "search stripes" ensures that the first edge is found in the region of interest.

When increasing "search stripes", the edge thickness found can fluctuate greatly, e.g. if only half the region of interest is occupied by the edge. The reason for this is that the edge thickness is displayed for the first (not the thickest) edge, which lies in the search direction above the switching threshold.

Edge detector with search stripes = 1 The dominating edge perpendicular to the search direction is found.	
Settings in the Parameters tab: Search stripes = 1	 Probe 1: Edge strength 11,00; Smoothing 1,00 px; Transition; Search stripes 1; Edge position First; Search direction Probe 2: Edge strength 11,00; Smoothing 1,00 px; Transition; Search stripes 1; Edge position First; Search direction ***Fig. 91:*** *Parameters "search stripes" = 1*
Image display: The dominating edge perpendicular to the search direction is found.	 ***Fig. 92:*** *Dominating edge perpendicular to the search direction*

Edge detector with search stripes > 1.
The first edge perpendicular to the search direction is found

Settings in the Parameters tab:
Search stripes = 3


Parameters

Probe type

Probe 1
Edge strength 11,00
Edge position First
Smoothing 1,00 px
Search direction
Transition
Search stripes 3

Probe 2
Edge strength 11,00
Edge position First
Smoothing 1,00 px
Search direction
Transition
Search stripes 3

Fig. 93: *Parameters "search stripes" = 3*

Image display:
The first edge perpendicular to the search direction is found

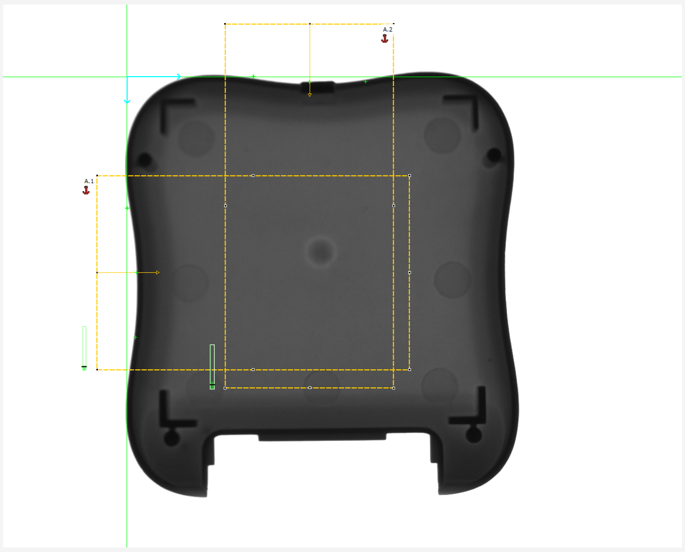


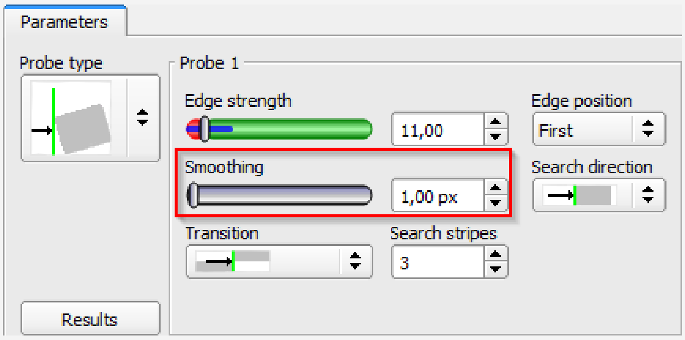
Fig. 94: *First edge perpendicular to the search direction is found*

8.2.3.5.2 Function of smoothing on sharp or blurred edges

The edge thickness results from the addition of edge steps over an area in the search direction, the size of which is given by the "smoothing" parameter.
For sharp edges, the edge thickness is not increased by increasing smoothing.
However, for blurred edges, the edge thickness is increased by increasing smoothing.

Detection of sharp and blurred edges with a low level of smoothing

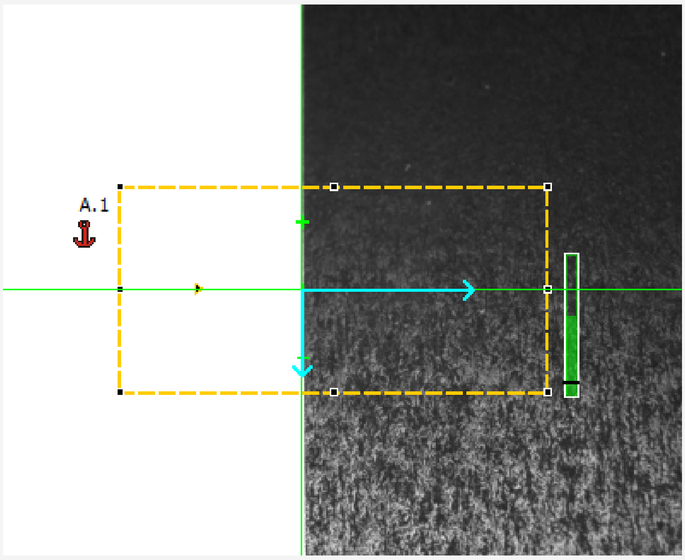
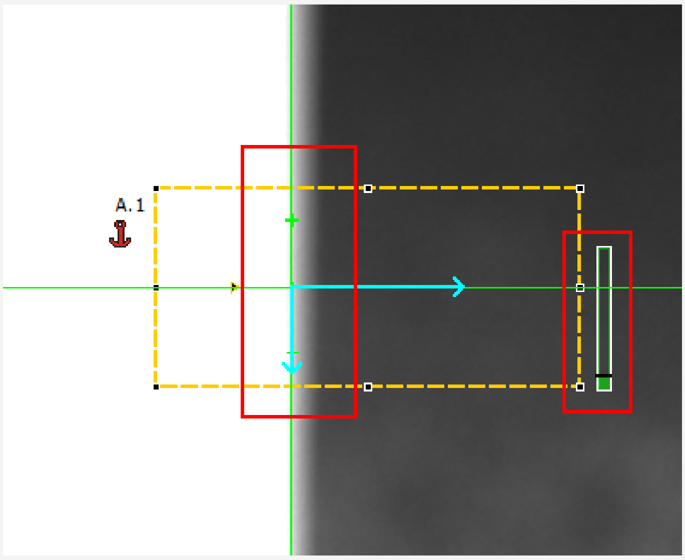
Settings in the Parameters tab:
Smoothing = 1


Parameters

Probe type

Probe 1
Edge strength 11,00
Edge position First
Smoothing 1,00 px
Search direction
Transition
Search stripes 3

Fig. 95: *Parameters smoothing = 1*

Image display: Detection of sharp edge. High edge thickness with a low level of smoothing.	 *Fig. 96: High edge thickness with a low level of smoothing*
Image display: Detection of blurred edge. Low edge strength with a low level of smoothing.	 *Fig. 97: Low edge strength with a low level of smoothing*

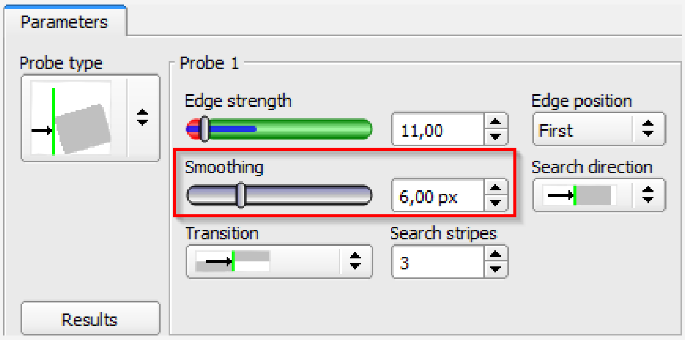
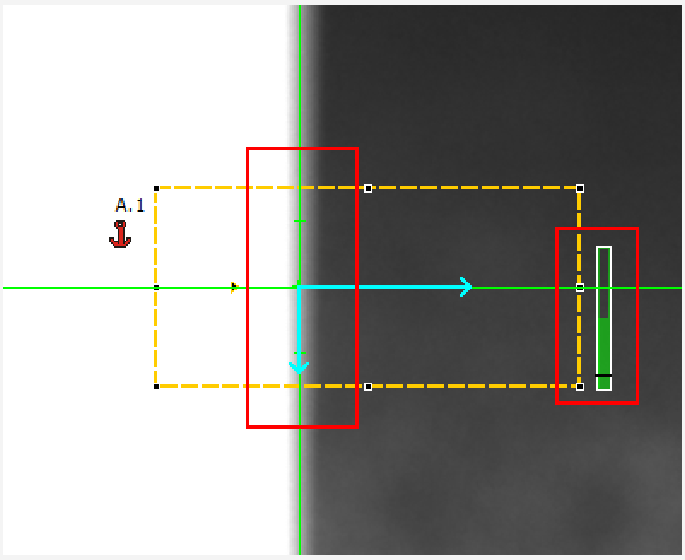
Detection of blurred edges with a high level of smoothing	
Settings in the Parameters tab: Smoothing = 6	 Parameters Probe type Probe 1 Edge strength: 11,00 Edge position: First Smoothing: 6,00 px Search direction Transition Search stripes: 3 Results *Fig. 98: Parameters smoothing > 1*

Image display: Detection of blurred edges. High edge strength with a high level of smoothing.	 ***Fig. 99:** High edge strength with a high level of smoothing*

8.2.3.5.3 Effect of smoothing on interfering edges

As mentioned above, the edge thickness results from the addition of edge steps over an area in the search direction, whose size is given by the "smoothing" parameter. If edges of different polarity lie in this area (dark-bright: positive polarity, bright-dark: negative polarity), their edge steps can neutralize each other. This can be used to eliminate interfering edges, by choosing "smoothing" which is sufficiently large.

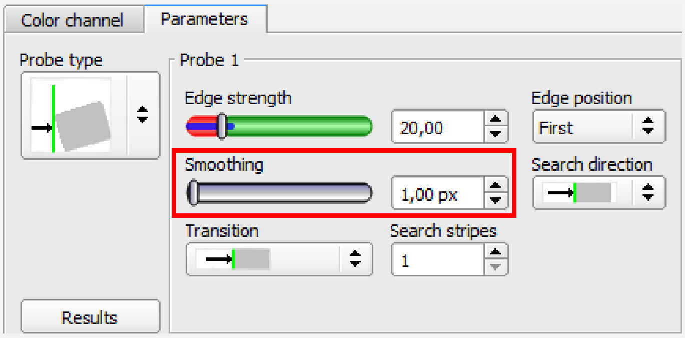
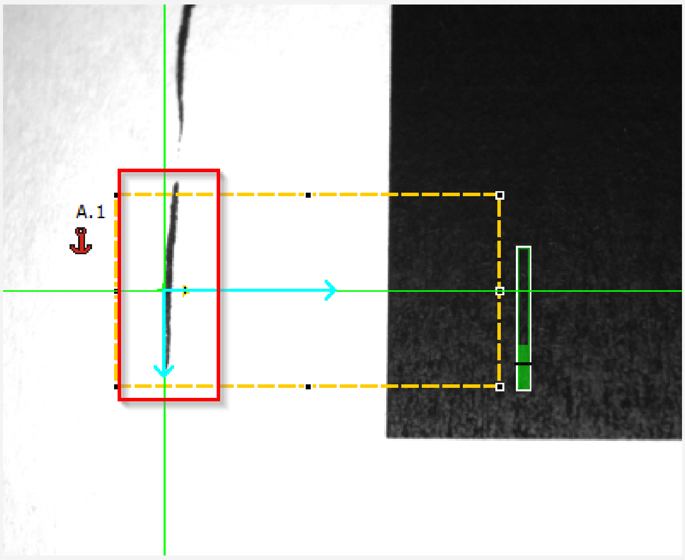
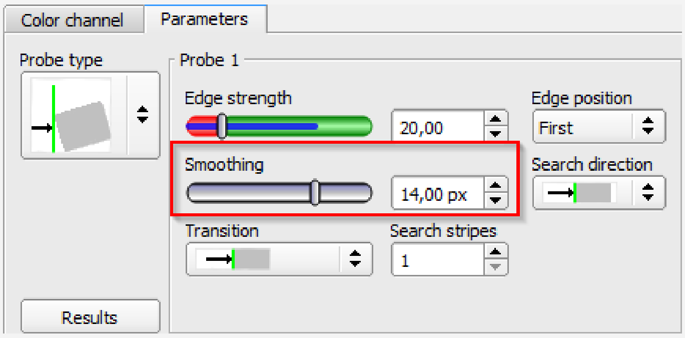
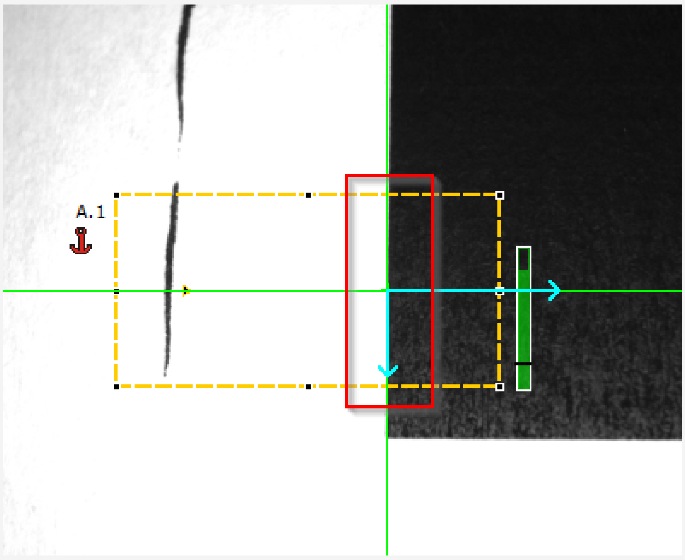
Detection with smoothing = 1. Interfering edge is not skipped.	
Settings in the Parameters tab: Smoothing = 1	 Color channel Parameters Probe type Probe 1 Results Edge strength 20,00 Smoothing 1,00 px Transition Search stripes 1 Edge position First Search direction ***Fig. 100:** Parameters smoothing = 1*

Image display: Interfering edge is not skipped.	 ***Fig. 101:** Interfering edge is not skipped*

Detection with smoothing > 1. Interfering edge is skipped.	
Settings in the Parameters tab: Smoothing > 1	 Color channel Parameters Probe type Probe 1 Edge strength 20,00 Edge position First Smoothing 14,00 px Search direction Transition Search stripes 1 Results ***Fig. 102:** Parameters smoothing > 1.*
Image display: Interfering edge is skipped.	 ***Fig. 103:** Interfering edge is skipped*

8.2.4 Alignment Contour matching

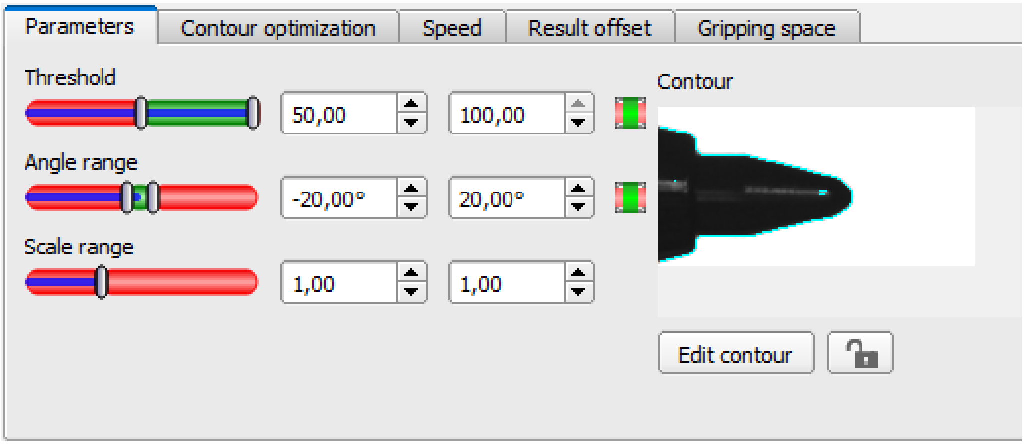
This detector is suitable for detecting contours based on edges. The contours of the object in the region of interest are taught and stored in the sensor. In Run mode, the sensor searches for the position of the best fit with the taught contour in the current image. If the fit is higher than the selected threshold, the result is considered positive. Contour detection is completely tolerant to rotation positions, i.e. the searched object may appear in any position in the image (choose angle setting accordingly).

8.2.4.1 Color Channel tab

See Chapter: Color Channel tab

8.2.4.2 Parameters tab

The most important parameters for contour detection can be set in the "Parameters" tab.



Threshold 50,00 100,00

Angle range -20,00° 20,00°

Scale range 1,00 1,00

Fig. 104: Alignment Contour matching, Parameters tab

The edges marked in light blue at the bottom right (high-contrast transitions in the image) were identified and drawn in on the basis of the parameter settings made in the teach-in area (red frame). These can be further altered by changing the parameters or with the function "Edit contour". The VISOR® vision sensor now searches for this contour in the image within the region of interest (yellow frame).

The following parameters can be set in the "Parameters" tab:

Parameter	Function
Threshold	Value for the required match of the found contour with the taught-in contour
Angle range	Angle range in which to search (larger range leads to longer processing times).
Scaling	Detection also of enlarged or reduced objects in a given scale range.
Contour	Shows the taught-in contour
Edit contour	Via the parameter "Edit contour", ranges of the taught-in contour can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The drawings can also be inverted via the option "Invert all". Additional information: Function: Edit pattern / contour
Lock 🔓	Locking / unlocking the settings: In the locked state, the taught-in settings are protected against (unintentional) change, e.g. accidentally adjusting the learning area. Unlock (click the lock icon again) for changing the settings.

Additional information:

Improvement of execution speed

- Region of interest for position (yellow frame) only as large as necessary.
  The region of interest indicates the area in which the center of gravity of the contour is searched.
- Region of interest for angles only as large as necessary.
- The region of interest for scaling is only as large as necessary.
- Reduce resolution (e.g. to WGA)

  ATTENTION:
  This parameter affects all detectors!
- Set slider (accurate - fast) to "fast"
- Increase the value for "Min. Contrast Model", since small values can lead to a larger number of extracted contours. In the display of the pattern, check whether the relevant contours are still present.
- Increase value for "Min. contrast image".
- Especially in the case of Alignment: Use an alternate search pattern. For example, with higher contrast, so that "Min. contrast pattern" and "Min. contrast image" can be increased.

Robust detection

- Region of interest (yellow frame) sufficiently large?
- Region of interest for angles sufficiently large?
- Region of interest for scaling sufficiently large?
- Min. contrast pattern and Min. contrast image sufficient? Is the contrast sufficient when acquiring the model and the images that are to be checked? Is the model detectable in the images that are to be checked?
- Set slider (accurate - fast) to "accurate"
- Objects should not overlap.
- Are there distinctive edges in the model? If necessary, teach in the model again so that prominent edges lie in the taught-in model.
- Is "Min. contrast pattern" suitably selected? If the relevant contours are not displayed in the taught-in model, reduce "Min. contrast pattern". If too many contour lines are shown, increase "Min. contrast pattern".
- Is "Min. contrast image" suitably chosen for the current image? If the current image has a smaller / larger contrast than the taught-in pattern, "Min. contrast image" should be smaller / larger than "Min. contrast pattern".
- Model found in wrong position? If the taught-in model is not unique, teach-in a new model.
- Does the result value fluctuate from image to image? If necessary, ensure that no "wrong" contours are taught in the image (edges due to shadows or contour fragments that are not desired in the contour model). This can be achieved by increasing "Min. contrast pattern". With the help of "Edit contour", regions of interest can be hidden.

Parameter angle range: Rotational direction of angle

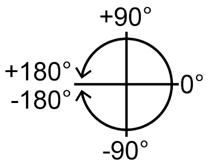
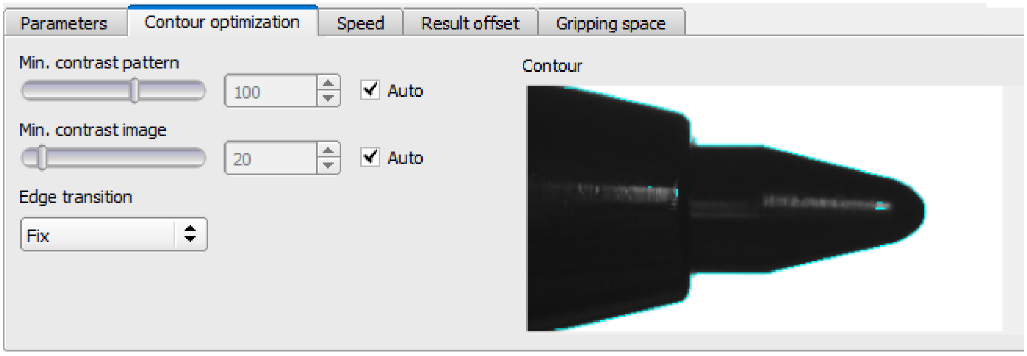


Fig. 105: *Rotational direction of angle*

8.2.4.3 Contour optimization tab

In the "Contour optimization" tab, further settings for the edge transition and the contrast can be made.



Min. contrast pattern 100 ☑ Auto

Min. contrast image 20 ☑ Auto

Edge transition Fix

__Fig. 106:__ Alignment Contour matching, tab Contour optimization

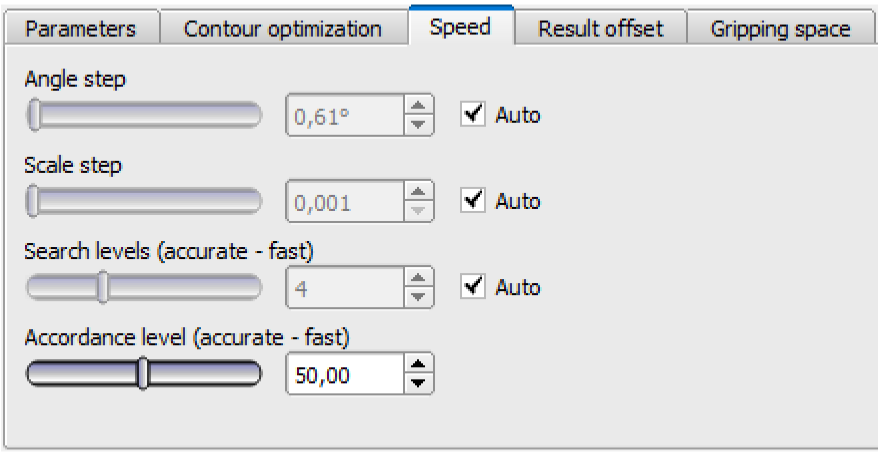
The following parameters can be set in the "Contour optimization" tab:

Parameter	Function
Min. contrast pattern	Minimum required contrast for the taught-in model, in which an edge is accepted as such.
Min. contrast image	Minimum contrast required in current image for an edge to be accepted as one.
Edge transition • fix • fix + inverted • flexible	The parameter "Edge transition" can be used to determine the transition between object or contour and background. The way in which the contour is to be recognized is selected: • "fix" = only on the taught-in background • "fix + inverted" = only on the taught-in and inverted background • "flexible" = on any background For more information, please refer to "Additional information:".
Auto	Automatic setting

Additional information: see description Edge transition Contour detector

8.2.4.4 Speed tab

Using the adjustable parameters in the Speed tab, execution speed of the sensor can be altered. Adjusting the search levels sets the level of detail of the search and thus the time for a particular search. The search is either performed less finely, i.e. the search is canceled sooner and is therefore faster, or even finer details are taken into account in the search, i.e. a longer search is performed and the search is slower.



Angle step 0,61° ☑ Auto

Scale step 0,001 ☑ Auto

Search levels (accurate - fast) 4 ☑ Auto

Accordance level (accurate - fast) 50,00

__Fig. 107:__ Alignment Contour matching, Speed tab

The following parameters can set in the "Speed" tab:

Parameter	Function
Angle step	Sensitivity of the search throughout the selected angle range in degrees [°]
Scale step	Sensitivity of the search throughout the selected scale range
Search levels (accurate - fast)	Number of search levels (one search level corresponds to an image with half resolution) • Small value (accurate): Slow search = lower risk (less likely to overlook candidates) • High value (fast): Fast search = higher risk (candidates can be overlooked)
Accordance level (accurate - fast)	Candidates with a degree of compliance below the specified value are already discarded during the search. • Small value (accurate): Late rejection = slower = less risky • High value (fast): Early rejection = quicker = riskier In case of false results, this value can be decreased (more accurate).
Auto	Automatic setting

8.2.4.5 Result offset tab

See Chapter: Result offset tab

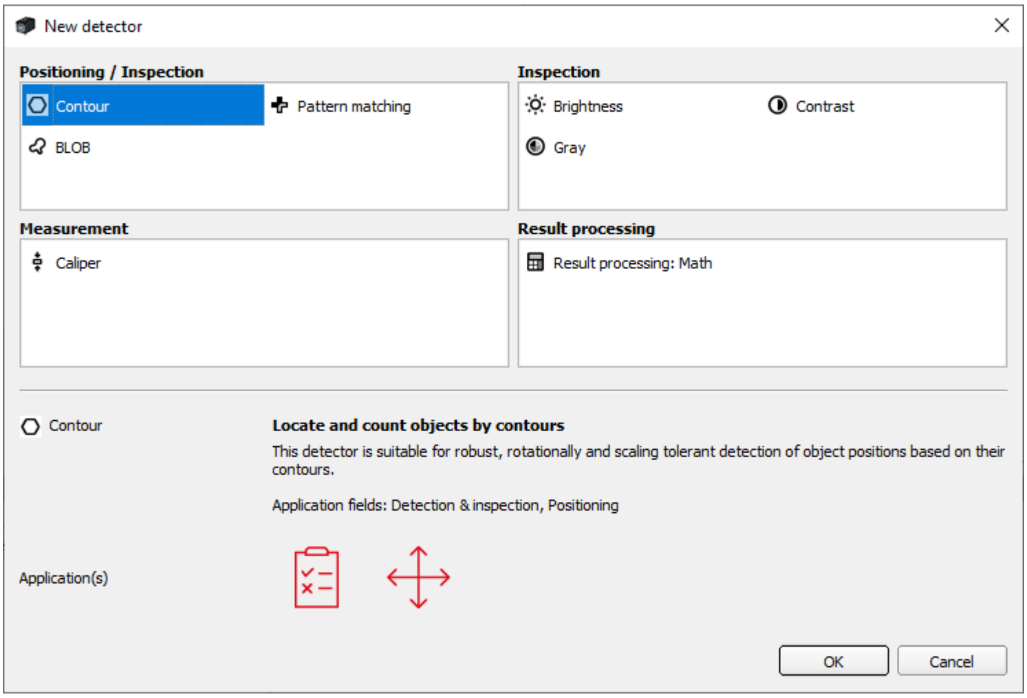
8.2.4.6 Gripping space tab

See Chapter: Gripping space tab

8.3 Setup Detectors

Each job contains one or several inspection steps (detectors), which you can define here. By clicking on the "Detectors" button, or the "New" button under the Detector list, a window with a list of all available detectors opens. The corresponding setting ranges are displayed graphically in the image as a frame in preset position and size. Now the frames and the parameters can be adjusted according to the inspection task.

For information on what the various frames in the image mean and how to adjust them, please refer to section: Search and feature ranges.



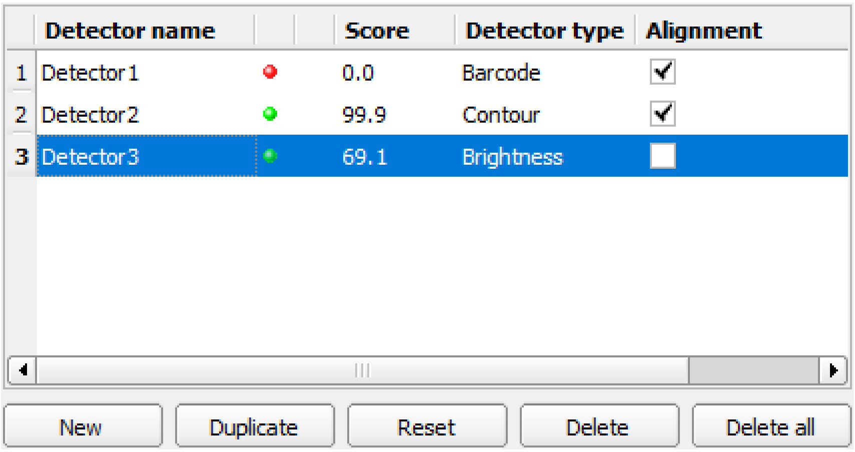
Locate and count objects by contours

This detector is suitable for robust, rotationally and scaling tolerant detection of object positions based on their contours.

Application fields: Detection & inspection, Positioning

Fig. 108: *Detector selection list, example: VISOR® Object*

8.3.1 Creating and adjusting detectors



	Detector name		Score	Detector type	Alignment
1	Detector1	🔴	0.0	Barcode	☑
2	Detector2	🟢	99.9	Contour	☑
3	Detector3	🟢	69.1	Brightness	☐

Fig. 109: *Detector list*

Create new detector

1. Click on the button "New" under the detector list in the configuration window and select the type of detector required. A new detector entry appears in the detector list.
2. Edit the name of the detector by double clicking on the field "Name".

Configuring the detector

1. Select a detector in the drop-down list and assign a name for each detector.
2. Define the corresponding Search and feature ranges graphically in the image.
3. Configure the detector by entering / setting parameters in the tabs in the configuration window to the right of the detector list. Which tabs are shown depends on the type of detector selected.

Configure overlay

In the menu "View"/"Overlay settings..." or in the toolbar, the overlays in the image (frame in yellow, red, etc.) can be switched on or off for each detector or category.

In "View" / "Overlay current detector only" or with the frame symbol button, all overlays in the image can be switched off except for those of the currently processed detector.

▣ Overlay of current detector only

▣ Overlay of failed detectors only

Functions for administration of detectors

Button	Function
New	Adds new detector > Dialogue with detector drop-down list appears
Duplicate	Create a new detector by copying the existing detector with all settings (incl. region of interest etc.).
Copy (right click on Detector > "Copy detector parameters...")	Copies all parameters from one detector to one or several others. All detectors must be from the same type. To copy the characteristic areas (different colored frames such as: Teach-in area, region of interest, etc.), select the corresponding checkboxes. Note: Size, shape (rectangle, circle, freeform), masking (in case of freeform) are copied. Position and orientation are **not** copied. Copy process: Create all desired target detectors of the same type as the source detector. Mark source detector in the detector list. Press to button "copy". In the appearing list, mark all desired target detectors (multiple marking with a held "Ctrl" key). Click "Copy" to confirm.
Reset	Reset the parameters and the search and feature range of the selected detector to the default values
Delete	Deletes the selected detector
Delete all	Deletes all of the detectors in the list

Error display in the detector list

The following icons can be displayed in the third column of the detector list in the event of an error:

⏰ = the computing time of a detector is longer than specified in the Cycle time tab, detector is NOK

📍 = no part is found during Alignment, all detectors dependent on it are NOK

📏 = Calibration has not been performed correctly, all following detectors are NOK

☒ = result buffer is exceeded (>10 MB)

8.3.2 Selecting a suitable detector

The following detectors are available:

Detector Type		Description
✚	Detector Pattern matching	Locate and count objects by patterns
○	Detector Contour	Locate and count objects by contours

Detector Type		Description
3D	Detector Contour 3D	Locate objects in space (3D)
3D	Detector Target Mark 3D	Locate objects in space using standardized markers (3D)
◐	Detector Contrast	Check contrast
☼	Detector Brightness	Check brightness
◐	Detector Gray	Check grey value distribution
	Detector BLOB	Count and evaluate objects
⇕	Detector Caliper	Measure distance between edges
	Detector Classification (AI)	Assign objects into defined classes
●	Detector Color Area	Check color value distribution
☰	Detector Color List	Assign objects to a color
☼	Detector Color Value	Check color values
▮▮▮	Detector Barcode	Read barcodes and evaluate quality
▦	Detector Datacode	Read datacodes and evaluate quality
ABC	Detector OCR	Read plain text (OCR)
▦	Detector Result processing: Text, numbers, robotics	Process and evaluate detector results
■	Detector Wafer	Locating and verifying of wafers
▯	Detector Busbar	Locating and verifying of busbars on wafers

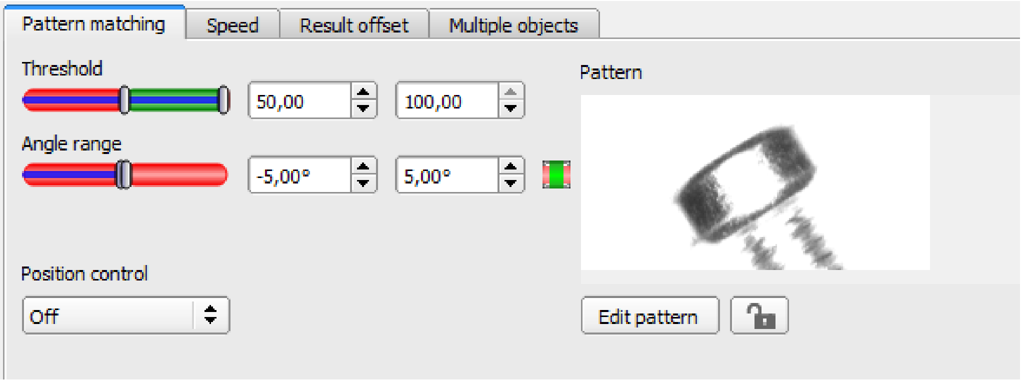
8.3.3 Detector Pattern matching

✚ This detector is suitable for detecting patterns of any shape, even without clear edges or contours

8.3.3.1 Color Channel tab

See Chapter: Color Channel tab

8.3.3.2 Pattern matching tab



Threshold: 50,00 – 100,00

Angle range: -5,00° – 5,00°

Position control: Off

Fig. 110: Pattern matching detector, Pattern matching tab

Parameter description:

Parameter	Function
Threshold	Range for the required match of the found pattern with the learned pattern in %.
Angle range	Angle range in which to search (larger range leads to longer processing times). Depending on the size and complexity of the image, the Angle range may be limited.
Position control	Checks whether the pattern found is in the right position. If position check is activated, the position frame is shown in blue (either rectangular or elliptic). The center of the pattern must be within the blue frame.
Pattern	Shows the taught pattern (red frame in the field of view)
Edit pattern	With the parameter "Edit pattern", areas of the taught-in pattern can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The marked areas can also be inverted (see also Function: Edit pattern / contour)
Lock 🔓	Lock / unlock pattern. When locked, the taught-in pattern is protected against (unintentional / accidental) changes, e.g. accidental adaptation of the teaching range. Unlock to modify taught pattern.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Improvement of execution speed

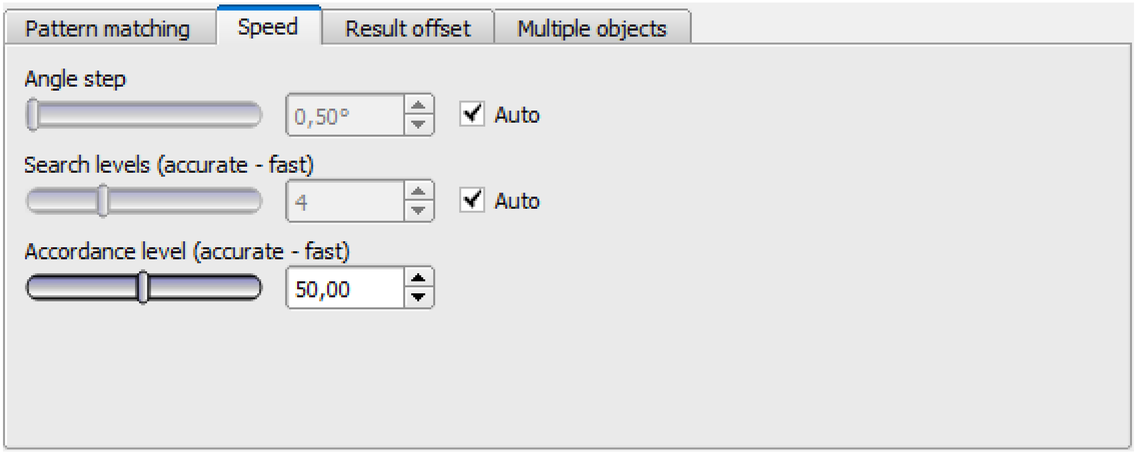
- Select the taught-in pattern (red frame) as small as possible.
- Region of interest for position (yellow frame) only as large as necessary.
  The region of interest indicates the area in which the center of gravity of the pattern is searched.
- Reduce resolution (e.g. to WGA)

  ⚠ **ATTENTION:**
  This parameter affects all detectors!
- Set slider (accurate - fast) to "fast"

Robust detection

- Region of interest (yellow frame) sufficiently large?
- Set slider (accurate - fast) to "accurate"
- Select a distinctive gray value pattern; re-teach if necessary.
- If found in the wrong position: Use a unique pattern, and re-teach and adjust the threshold if necessary.

8.3.3.3 Speed tab



Angle step 0,50° ☑ Auto

Search levels (accurate - fast) 4 ☑ Auto

Accordance level (accurate - fast) 50,00

Fig. 111: *Detector Pattern matching, Speed tab*

The execution speed is influenced by the adjustable speed parameters. The search is performed either less finely, i.e. earlier canceled and thus faster, or even finer details are taken into account in the search, i.e. search longer and the search is slower.

If, immediately after teach-in, the found position (green ROI) does not coincide with the teach-in range (red ROI), the "Search levels (accurate - fast)" slider should be set to "fast".

Parameter description:

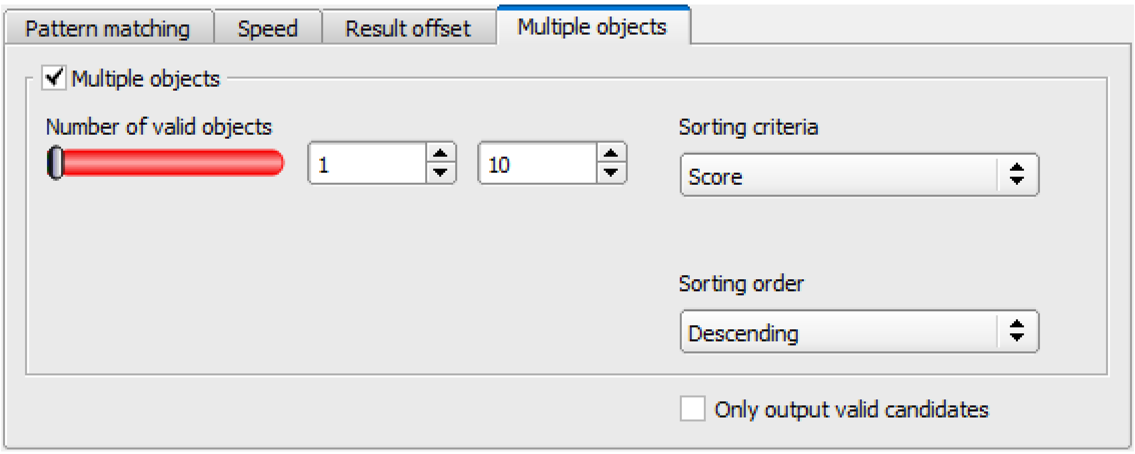
Parameter	Function
Angle step	Sensitivity of the search throughout the selected angle range in degrees [°]
Search levels (accurate - fast)	Number of search levels (one search level corresponds to an image with half resolution) • Small value (accurate): Slow search = lower risk (less likely to overlook candidates) • High value (fast): Fast search = higher risk (candidates can be overlooked)
Accordance level (accurate - fast)	Candidates with a degree of compliance below the specified value are already discarded during the search. • Small value (accurate): Late rejection = slower = less risky • High value (fast): Early rejection = quicker = riskier In case of false results, this value can be decreased (more accurate).
Auto	Automatic setting

8.3.3.4 Result offset tab

See Chapter: Result offset tab

8.3.3.5 Multiple objects tab

By default, a maximum of one instance of the taught object is found in the image. The Multiple objects tab also allows you to find multiple instances of a taught-in pattern or contour. Multiple detection identifies objects whose pattern or contour matches the taught pattern or contour. The output of the object results is sorted according to the set criteria in ascending or descending order.



Pattern matching | Speed | Result offset | Multiple objects

☑ Multiple objects

Number of valid objects: 1 – 10

Sorting criteria: Score

Sorting order: Descending

□ Only output valid candidates

__Fig. 112:__ Detector Pattern matching, Multiple objects tab

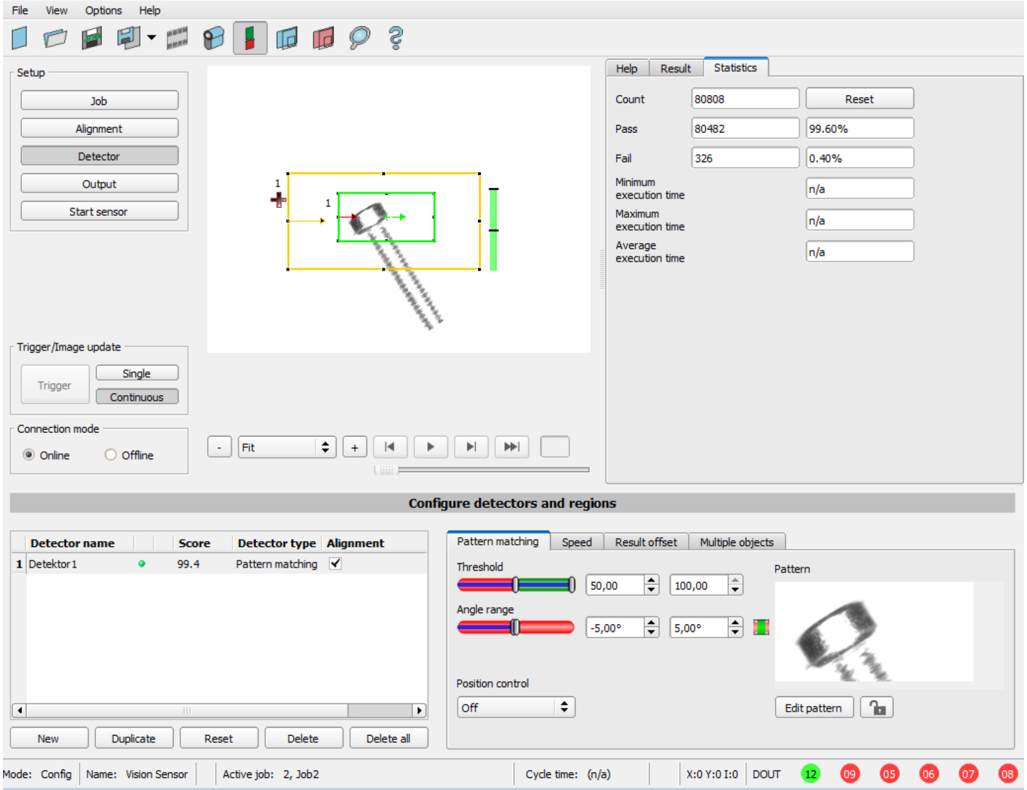
It is also possible to use this function to count objects in the image. The number of objects found can be output as a telegram. The minimum and maximum number of tolerated objects can be specified with the "Number of valid objects" parameter. If the number of objects found is outside this range, the detector result is NOK.

Parameter description:

Parameter	Function
Number of valid objects	This parameter allows you to check whether the number of objects found is within a specified range. If it is within the range, the detector result is OK, otherwise it is NOK.
Sorting criterion • Score • Position X • Position Y • Angle Z Contour detector only: • Scaling	Sorting criteria according to which objects should be "pre-sorted". • Score value • Position X • Position Y • Angle around Z • Scaling
Sorting order • Ascending • Descending	Sorting order for the selected sorting criterion. • The values of the sorting criteria are sorted in ascending order. • The values of the sorting criteria are sorted in descending order.
Only output valid candidates	If this checkbox is activated, only objects whose score value is above the set threshold ("Pattern matching" or "Contour" tab) are displayed and output. It can be used e.g. for parameter optimization.

8.3.3.6 Pattern matching application

In this example, a contact (far left) of the test piece was taught in as a pattern and is recognized at this point even with a high degree of compliance (switching threshold near 100%).



Count 80808

Pass 80482 99.60%

Fail 326 0.40%

Minimum execution time n/a

Maximum execution time n/a

Average execution time n/a

	Detector name	Score	Detector type	Alignment
1	Detektor1 ●	99.4	Pattern matching	☑

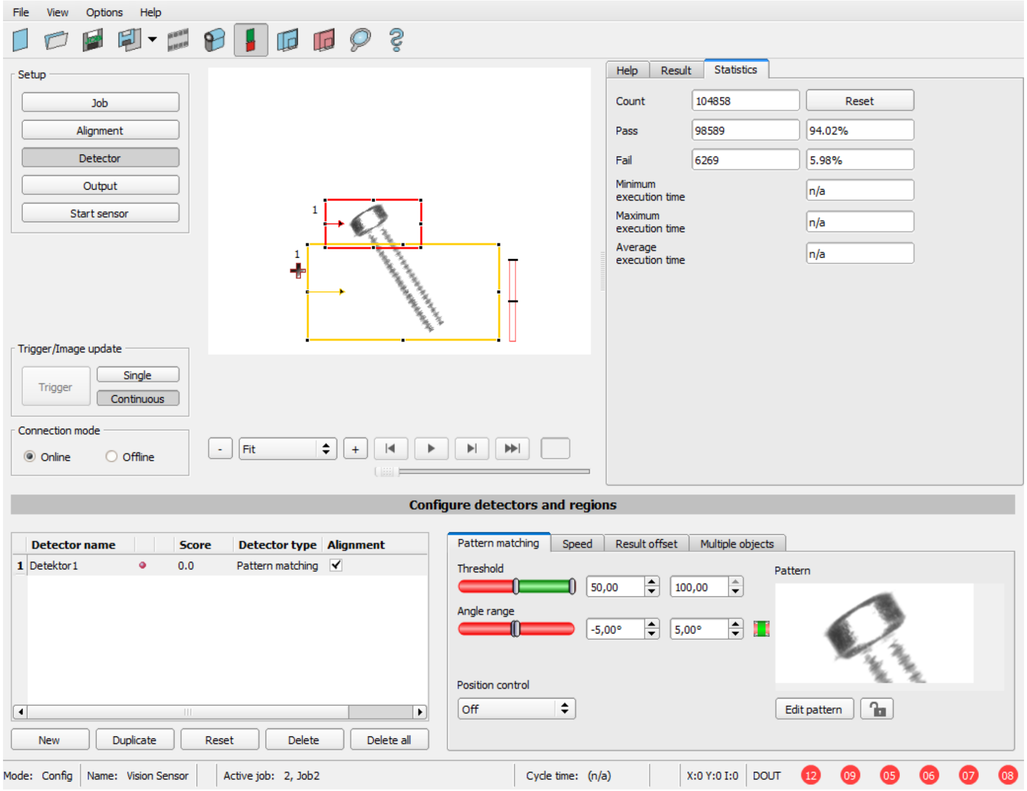
Threshold 50,00 100,00

Angle range -5,00° 5,00°

Position control Off

Mode: Config | Name: Vision Sensor | Active job: 2, Job2 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 113: *Pattern matching, application example, result OK*



Count 104858

Pass 98589 94.02%

Fail 6269 5.98%

Minimum execution time n/a

Maximum execution time n/a

Average execution time n/a

	Detector name	Score	Detector type	Alignment
1	Detektor1 ●	0.0	Pattern matching	☑

Threshold 50,00 100,00

Angle range -5,00° 5,00°

Position control Off

Mode: Config | Name: Vision Sensor | Active job: 2, Job2 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 114: *Pattern matching, application example, result NOK*

If the same Pattern matching is performed at a location of the test part where the contact sought is missing, the degree of conformity does not reach the required threshold value and the result becomes NOK. The contact is searched for here because of the gray values at the respective places in the image. Since the interior, very highly reflective and therefore bright area does not exist, and instead the image pixels correspondingly have darker values, the degree of conformity here is not as high as with an existing contact. However, because large parts of the pattern are identical to the one taught in (the entire outer, black area), the degree of conformity is still quite high, at around 70%!

The settings made here are chosen only to illustrate the operation of the Pattern matching detector. They should be further optimized in real operation (e.g. by reducing the search and feature range >> relevant pattern becomes more significant, etc.).

By teaching, the pattern inside the red frame is stored in the sensor as a reference. Size and position of the reference is defined by the red frame. In Run mode, the VISOR® will search the current image for the best reference image / pattern match inside the search region. Depending on the setting of the threshold value (= degree of conformity), the object is recognized as good or not.

Example:

The following pattern was taught:

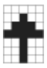


Fig. 115: *Pattern, reference*

For the following three example images, the 100% match object is recognized because the taught pattern is exactly the same even though it is in a different location in the image. However, it is only moved in the X or Y direction and not rotated.

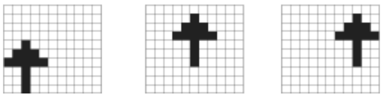


Fig. 116: *Pattern, result OK*

In the case of the three following example images, the object is also recognized, but with less than 100% match (about 70 - 80%), because in some pixels, it differs from the taught-in pattern. Depending on the setting of the threshold value (degree of match), good or bad results are returned.

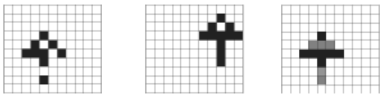


Fig. 117: *Pattern, borderline cases*

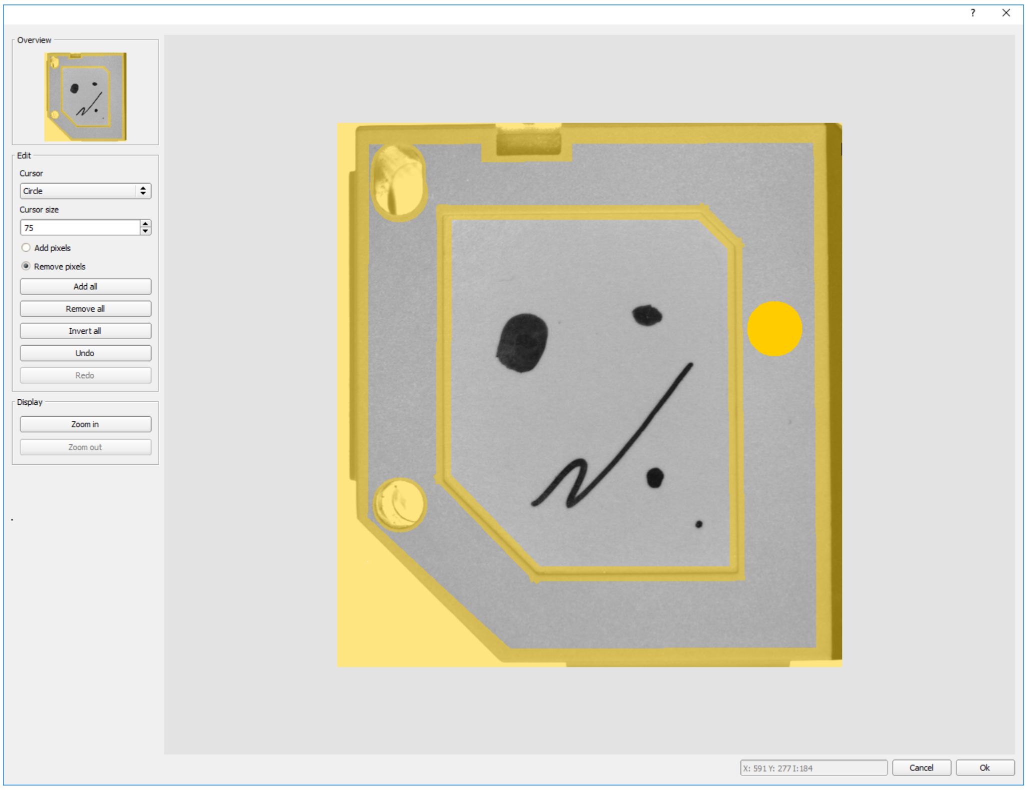
8.3.3.7 Function: Edit pattern / contour

With the function "Edit region of interest" or "Edit pattern" / "Edit contour", areas for the evaluation can be allowed or excluded within the search fields / feature fields of the various detectors.

Application example

Outer and inner contour lines, as well as holes should be irrelevant for the evaluation, but all surface defects should be detected.

After masking, only the unmarked areas within the ROI of the detector are used for the evaluation. The yellow marked areas are masked and thus no longer relevant for the evaluation.



Cursor: Circle

Cursor size: 75

○ Add pixels

● Remove pixels

X: 591 Y: 277 I:184

Fig. 118: *Edit region*

Handling

Parameter	Function
Cursor (shape)	Changing shape of the cursor (square, circle, or line) With the setting: "Cursor = Line", the angular position of the line jumps while the Shift key is pressed in 15 ° increments.
Cursor size	Changing the size of the cursor (possible setting 1-500, also e.g. with the mouse wheel)
Add / remove pixels	Select whether the cursor adds or excludes pixels for image processing
Add all	Adds all pixels to image processing
Removes all	Removes all pixels from image processing
Invert all	Inverts all pixels
Undo	Undoes the last action
Restore	Restores the last undo action
Display	Selects the display mode (zoom in / out)

The flexible selection of cursor shape and size, as well as whether an action adds or removes pixels, can easily and quickly define complex geometrical or free-form areas that are relevant or not taken into account (yellow).

Settings for using the "Edit region" function for the different detector types

Detector Type	Necessary setting for editing
Pattern matching Contour	Generally possible with "Edit pattern" / "Edit contour"
Contrast, Brightness, Gray, BLOB, Color Value, Color Area, Color List	Select region of interest "Free shape"

Masking of search and learn ranges, Examples

Adjustment of the region of interest to the object for detectors Contrast, Brightness, and Gray level

For the detectors mentioned above, there are three forms for the region of interest: Circle, Rectangle, and Free form. If the work area cannot be adjusted well enough to the object with a circle or rectangle that can be rotated over the control point on the arrow, use the freeform region of interest. With this feature, any geometry can be designed for the region of interest. The cursor for editing the region of interest can be selected as a square or a circle of any size. Below are some examples of freeform regions of interest with a brief description of how they were created in the freeform editor.

Example 1: Circles with relevant areas

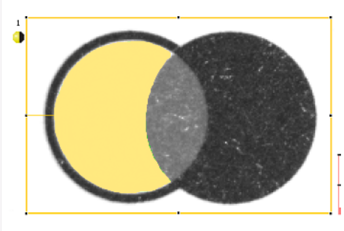


Fig. 119: *Edit region 1*

Generated by an adding and a withdrawing circle.

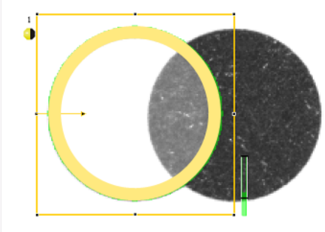


Fig. 120: *Edit region 2*

Generated by an adding and a withdrawing circle.

Example 2: Only surface defects are relevant, object contour lines are masked

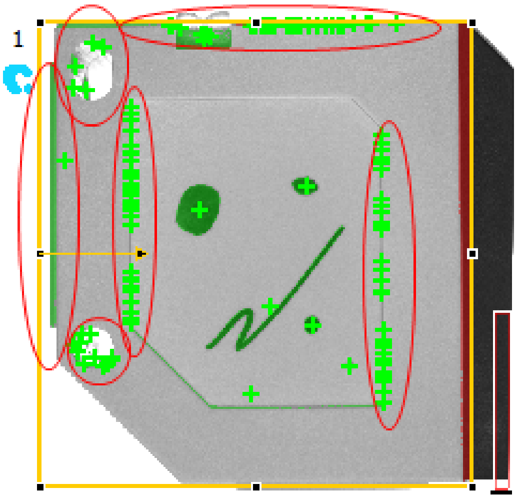
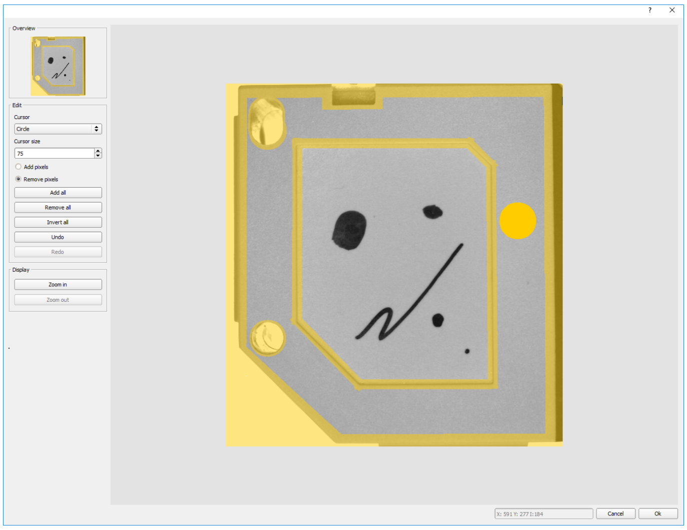


Fig. 121: *BLOB detector without the use of masking*

With the used BLOB detector, all surface defects as well as the outer and inner contour lines are detected.



Cursor: Circle

Cursor size: 75

○ Add pixels

● Remove pixels

X: 591 Y: 277 I:184

Fig. 122: *Masking the contour lines that are not to be recognized = yellow areas.*

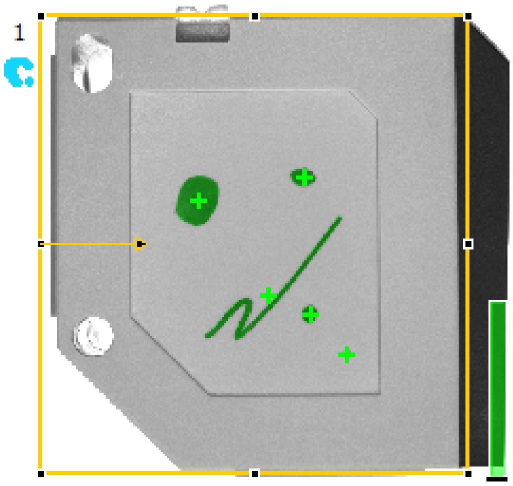


Fig. 123: BLOB detector with the use of masking

Only surface defects are detected; all contour lines / objects in the masking area are no longer recognized.

8.3.4 Detector Contour

○ This detector is suitable for robust, rotationally and scaling tolerant detection of object positions based on their contours.

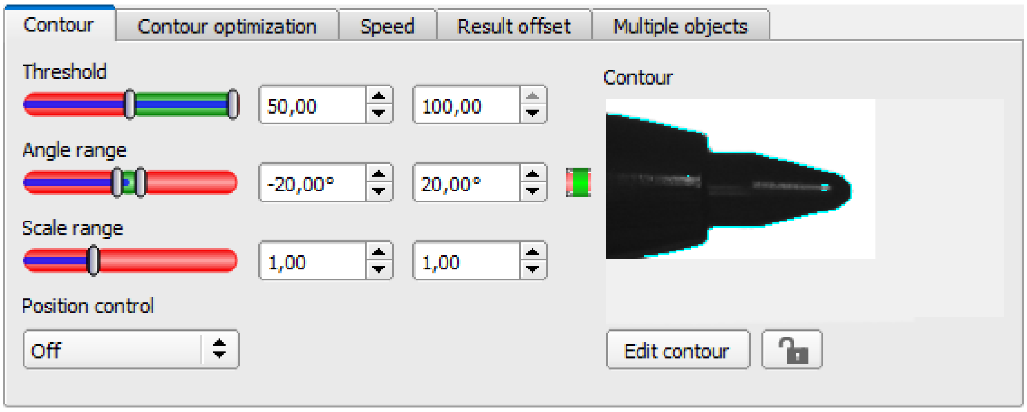
The contours of the object in the region of interest are taught and stored in the sensor. In Run mode, the sensor searches for the position of the best fit with the taught contour in the current image. If the fit is higher than the selected threshold, the result is considered positive. The contour recognition is completely rotational tolerant, i.e. the searched object may appear in any position in the image (choose angle setting accordingly!).

8.3.4.1 Color Channel tab

See Chapter: Color Channel tab

8.3.4.2 Contour tab

The most important parameters for contour detection can be set in the "Contour" tab.



Threshold: 50,00 – 100,00

Angle range: -20,00° – 20,00°

Scale range: 1,00 – 1,00

Position control: Off

Fig. 124: Detector Contour, Contour tab

The edges marked in light blue at the bottom right (high-contrast transitions in the image) were identified and drawn in on the basis of the parameter settings made in the teach-in area (red frame). These can be further altered by changing the parameters or with the function "Edit contour". The VISOR® vision sensor will search for this contour only in the image within the search region (yellow frame).

Parameter description:

Parameter	Function
Threshold	Value for the required match of the found contour with the taught-in contour
Angle range	Angle range in which to search (larger range leads to longer processing times).
Scaling	Detection also of enlarged or reduced objects in a given scale range.
Contour	Shows the taught-in contour
Edit contour	Via the parameter "Edit contour", ranges of the taught-in contour can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The drawings can also be inverted via the option "Invert all". Additional information: "Function: Edit pattern / contour"
Lock 🔓	Locking / unlocking the settings: In the locked state, the taught-in settings are protected against (unintentional) change, e.g. accidentally adjusting the learning area. Unlock (click the lock icon again) for changing the settings.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Improvement of execution speed

- Region of interest for position (yellow frame) only as large as necessary.
  The region of interest indicates the area in which the center of gravity of the contour is searched.
- Region of interest for angles only as large as necessary.
- The region of interest for scaling is only as large as necessary.
- Reduce resolution (e.g. to WGA)

  ⚠ **ATTENTION:**
  This parameter affects all detectors!
- Set slider (accurate - fast) to "fast"
- Increase the value for "Min. Contrast Model", since small values can lead to a larger number of extracted contours. In the display of the pattern, check whether the relevant contours are still present.
- Increase value for "Min. contrast image".
- Especially in the case of Alignment: Use an alternate search pattern. For example, with higher contrast, so that "Min. contrast pattern" and "Min. contrast image" can be increased.

Robust detection

- Region of interest (yellow frame) sufficiently large?
- Region of interest for angles sufficiently large?
- Region of interest for scaling sufficiently large?
- Min. contrast pattern and Min. contrast image sufficient? Is the contrast sufficient when acquiring the model and the images that are to be checked? Is the model detectable in the images that are to be checked?
- Set slider (accurate - fast) to "accurate"
- Objects should not overlap.
- Are there distinctive edges in the model? If necessary, teach in the model again so that prominent edges lie in the taught-in model.

- Is "Min. contrast pattern" suitably selected? If the relevant contours are not displayed in the taught-in model, reduce "Min. contrast pattern". If too many contour lines are shown, increase "Min. contrast pattern".
- Is "Min. contrast image" suitably chosen for the current image? If the current image has a smaller / larger contrast than the taught-in pattern, "Min. contrast image" should be smaller / larger than "Min. contrast pattern".
- Model found in wrong position? If the taught-in model is not unique, teach-in a new model.
- Does the result value fluctuate from image to image? If necessary, ensure that no "wrong" contours are taught in the image (edges due to shadows or contour fragments that are not desired in the contour model). This can be achieved by increasing "Min. contrast pattern". With the help of "Edit contour", regions of interest can be hidden.

Parameter angle range: Rotational direction of angle

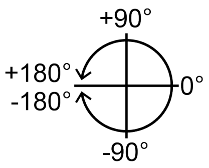
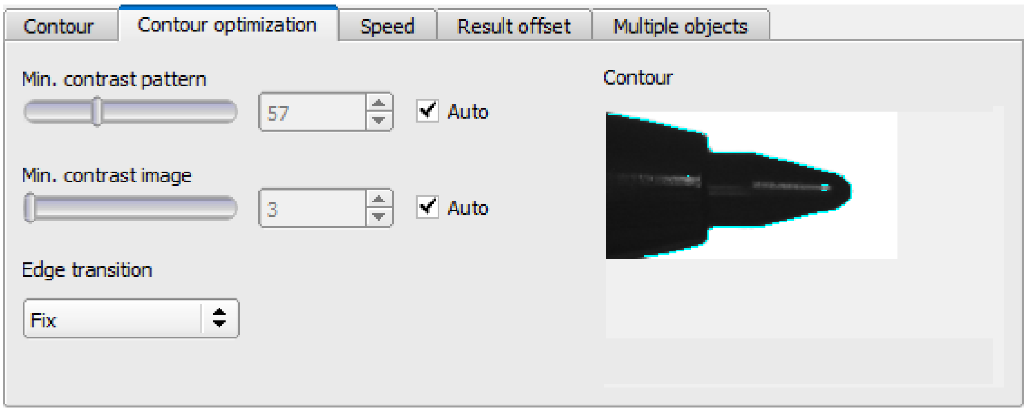


Fig. 125: Rotational direction of angle

8.3.4.3 Contour optimization tab

In the "Contour optimization" tab, further settings for the edge transition and the contrast can be made.



Min. contrast pattern 57 ☑ Auto

Min. contrast image 3 ☑ Auto

Edge transition Fix

Fig. 126: Detector Contour, Contour optimization tab

Parameter description:

Parameter	Function
Min. contrast pattern	Minimum required contrast for the taught-in model, in which an edge is accepted as such.
Min. contrast image	Minimum contrast required in current image for an edge to be accepted as one.
Edge transition • fix • fix + inverted • flexible	The parameter "Edge transition" can be used to determine the transition between object or contour and background. The way in which the contour is to be recognized is selected: • "fix" = only on the taught-in background • "fix + inverted" = only on the taught-in and inverted background • "flexible" = on any background Additional information: see below
Auto	Automatic setting

Edge transition

Example:

A gray object is taught-in in front of a brighter background, as shown in the following figure.

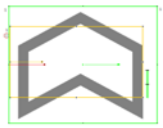
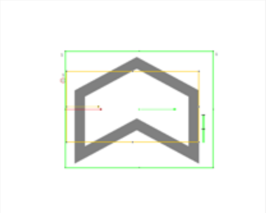
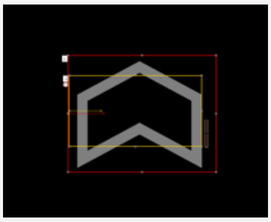
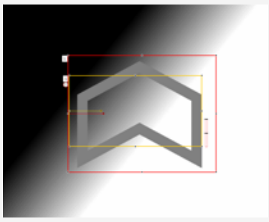
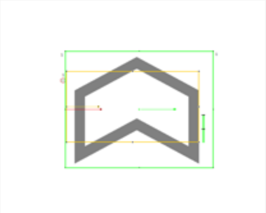
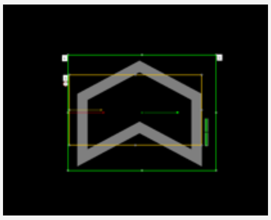
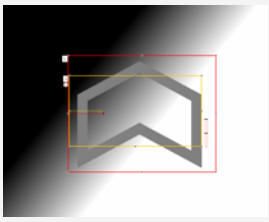
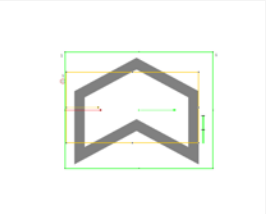
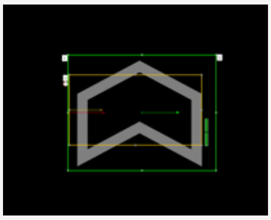
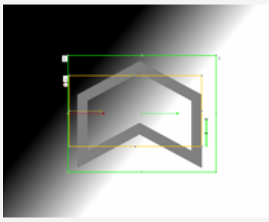


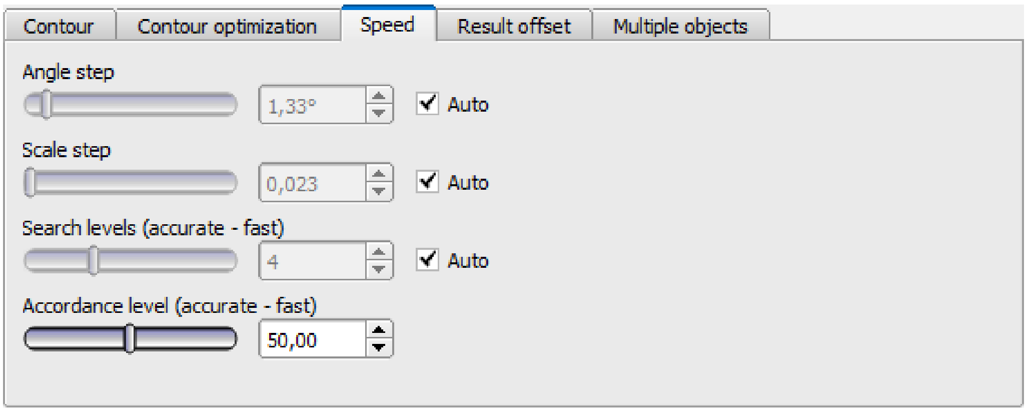
Fig. 127: *Taught-in contour in front of a bright background*

The following table shows how the contour detector behaves with the respective edge transition setting.

Settings for parameter "Edge transition"	Bright background	Dark background	Inconsistent background
fix	 Contour detector: OK	 Contour detector: NOK	 Contour detector: NOK
fix + inverted	 Contour detector: OK	 Contour detector: OK	 Contour detector: NOK
flexible	 Contour detector: OK	 Contour detector: OK	 Contour detector: OK

8.3.4.4 Speed tab

The execution speed is influenced by the adjustable speed parameters. The search is performed either less finely, i.e. earlier canceled and thus faster, or even finer details are taken into account in the search, i.e. search longer and the search is slower.



Angle step 1,33° ☑ Auto

Scale step 0,023 ☑ Auto

Search levels (accurate - fast) 4 ☑ Auto

Accordance level (accurate - fast) 50,00

Fig. 128: *Detector Contour, Speed tab*

Parameter description:

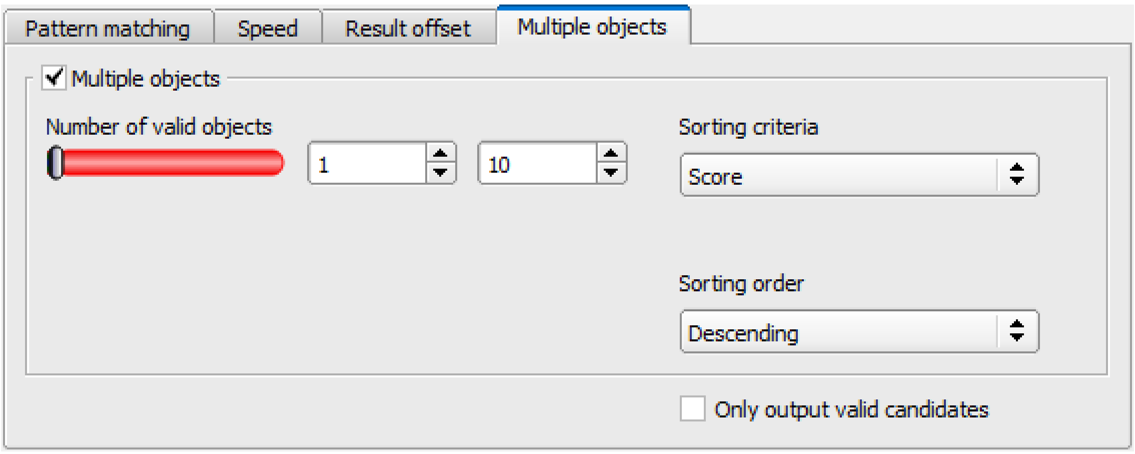
Parameter	Function
Angle step	Sensitivity of the search throughout the selected angle range in degrees [°]
Scale step	Sensitivity of the search throughout the selected scale range
Search levels (accurate - fast)	Number of search levels (one search level corresponds to an image with half resolution) • Small value (accurate): Slow search = lower risk (less likely to overlook candidates) • High value (fast): Fast search = higher risk (candidates can be overlooked)
Accordance level (accurate - fast)	Candidates with a degree of compliance below the specified value are already discarded during the search. • Small value (accurate): Late rejection = slower = less risky • High value (fast): Early rejection = quicker = riskier In case of false results, this value can be decreased (more accurate).
Auto	Automatic setting

8.3.4.5 Result offset tab

See Chapter: Result offset tab

8.3.4.6 Multiple objects tab

By default, a maximum of one instance of the taught object is found in the image. The "Multiple objects" tab makes it possible to find several instances of a taught contour. Multiple recognition identifies objects whose contours match the taught-in contour. The output of the object results is sorted according to the set criteria in ascending or descending order.



☑ Multiple objects

Number of valid objects: 1 – 10

Sorting criteria: Score

Sorting order: Descending

□ Only output valid candidates

Fig. 129: Contour Detector, Multiple objects tab

It is also possible to use this function to count objects in the image. The number of objects found can be output as a telegram. The minimum and maximum number of tolerated objects can be specified with the "Number of valid objects" parameter. If the number of objects found is outside this range, the detector result is NOK.

Parameter description:

Parameter	Function
Number of valid objects	This parameter allows you to check whether the number of objects found is within a specified range. If it is within the range, the detector result is OK, otherwise it is NOK.
Sorting criterion • Score • Position X • Position Y • Angle Z • Scaling	Sorting criteria according to which objects should be "pre-sorted". • Overall result • Position X • Position Y • Angle Z • Scaling
Sorting order • Ascending • Descending	Sorting order for the selected sorting criterion. The values of the sorting criteria are sorted in ascending order. The values of the sorting criteria are sorted in descending order.
Only output valid candidates	If this checkbox is activated, only objects whose score value is above the set threshold ("Contour" tab) are displayed and output. It can be used e.g. for parameter optimization.

8.3.5 Detector Contour 3D

This detector allows the localization of objects in space in six degrees of freedom (position X, Y, Z and Angle X, Y, Z) with the acquisition of a single image.

To teach-in the Contour 3D detector, the contours used must lie in **one** plane. The default contour plane is the Measurement plane that was defined during calibration (Measurement plane). However, the contour plane can also be adjusted in the "Contour plane" tab using the calibration plate (Contour plane).

NOTE:
For reliable detection, the use of the Contour 3D detector is recommended for printed objects.

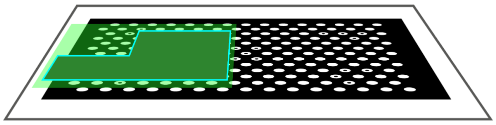


Fig. 130: Measurement plane

Contour plane (green) is parallel to the Measurement plane from calibration. The Measurement plane can be used as contour plane via Z-shift measurement plane.

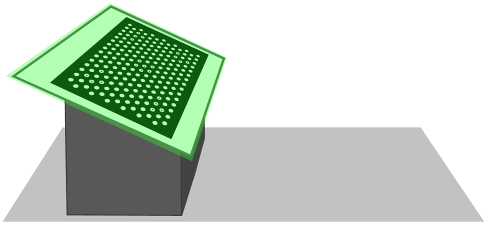


Fig. 131: Contour plane

The contour plane (green) is not parallel to the Measurement plane defined in the calibration (grey). The contour plane must be taught-in separately by taking an image of a calibration plate.

NOTE:

- To use the Contour 3D detector, a calibration must first be performed (calibration methods: Calibration plate (Robotics), Hand-Eye calibration (Robotics) or Base-Eye calibration (Robotics)).
- If no valid calibration is available, no contour can be taught.

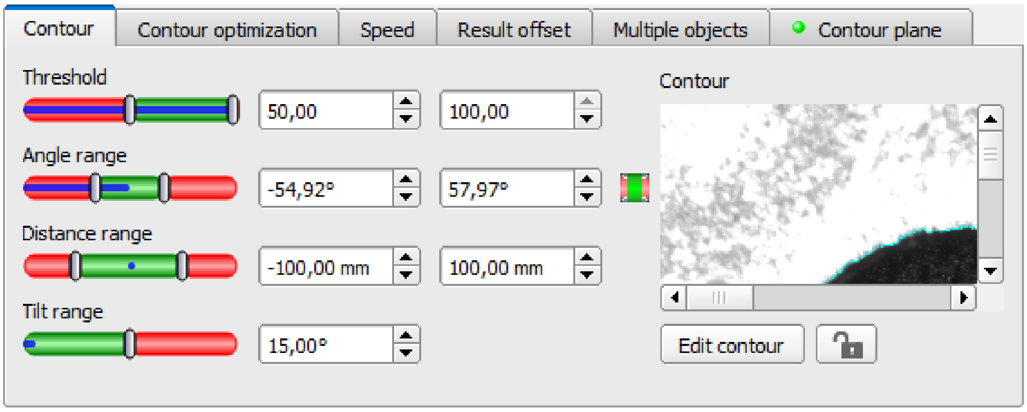
For reliable detection, the object or the desired contours should not be perpendicular to the optical axis, but slightly tilted. This causes the object to appear distorted in perspective when the image is captured. The more distinctive the perspective distortion is, the better a clear pose can be assigned to this distortion.

8.3.5.1 Color Channel tab

See Chapter: Color Channel tab

8.3.5.2 Contour tab

The most important parameters for contour detection can be set in the "Contour" tab.



Threshold: 50,00 – 100,00

Angle range: -54,92° – 57,97°

Distance range: -100,00 mm – 100,00 mm

Tilt range: 15,00°

Contour

Fig. 132: Contour 3D detector, Contour tab

The edges in the lower right corner marked in light blue (high-contrast transitions in the image) were identified and drawn on the basis of the parameter settings made. These can be further altered by changing the parameters or with the function "Edit contour". The contours used here must lie in one plane to obtain correct localization results. The VISOR® vision sensor now searches for this contour in the image within the region of interest (yellow frame).

Parameter description:

Parameter	Function
Threshold	Value for the required match of the found contour with the taught-in contour
Angle range	Angle range in which to search (larger range leads to longer processing times). Depending on the size and complexity of the image, the Angle range may be limited.
Distance range	Depth range in mm in which the search is carried out, starting from the contour plane.
Tilt range	Maximum tilt of the object in relation to the contour plane, i.e. the maximum rotation around the X and Y axes of the contour plane.
Contour	Shows the taught-in contour
Edit contour	Via the parameter "Edit contour", ranges of the taught-in contour can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The drawings can also be inverted via the option "Invert all". Additional information: "Function: Edit pattern / contour" **NOTE:** The result pose is always in the center of gravity of the taught contours. It therefore changes every time the contour is edited. This can be compensated via Result offset (Result offset tab).
Lock 🔓	Locking / unlocking the settings: In the locked state, the taught-in settings are protected against (unintentional) change, e.g. accidentally adjusting the learning area. Unlock (click the lock icon again) for changing the settings.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Improvement of execution speed

- Reduce resolution (e.g. to WGA) (→Setup step Job)

  ATTENTION:
  This parameter affects all detectors!
- Remove all unneeded areas around the taught-in contour with the "Edit contour" function.
- Region of interest for position (yellow frame) only as large as necessary.
  The region of interest indicates the area in which the center of gravity of the contour is searched.
- Increase the value for "Min. Contrast Model", since small values can lead to a larger number of extracted contours. In the display of the pattern, check whether the relevant contours are still present. (→ Contour optimization tab)
- Increase value for "Min. contrast image".
- Search levels / Accordance level: Set slider (accurate - fast) to "fast" (→ Speed tab)
- Especially in the case of Alignment: Use an alternate search pattern. For example, with higher contrast, so that "Min. contrast pattern" and "Min. contrast image" can be increased.

Robust detection

- The contours to be taught in must all be in the same plane. If necessary, use "Edit contour" to remove contours that do not lie in the contour plane.
- Set slider (accurate - fast) to "accurate". However, this leads to an increased cycle time.
- The object or the desired contours should not be perpendicular to the optical axis, but slightly tilted so that the object appears distorted in perspective when the image is taken. This increases the accuracy.

- The taught-in model should have as few symmetries as possible, since the results are not clear for symmetrical contours (example: teach-in of a circle: angle of rotation is undefined; teach-in of an ellipse: angle of rotation is undefined at 180°). If necessary, add further contour features to create an asymmetry.
- The contours used must not be caused by reflections, as these change their shape and position when the objects are tilted. Use illumination that is as homogeneous as possible.
- Objects should not overlap.
- The objects to be found should appear as large as possible in the image.
- The taught-in contour should have as many features / points as possible (which of course must correspond to real contours / points of the object).

Parameter angle range: Rotational direction of angle

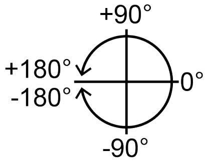
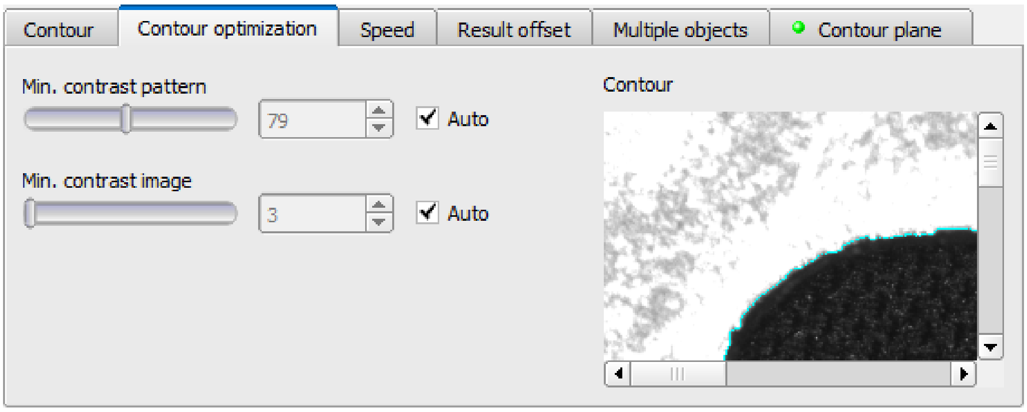


***Fig. 133:** Rotational direction of angle*

8.3.5.3 Contour optimization tab

In the "Contour optimization" tab, further settings for the edge transition and the contrast can be made.



Min. contrast pattern 79 ☑ Auto

Min. contrast image 3 ☑ Auto

***Fig. 134:** Detector Contour 3D, Contour optimization tab*

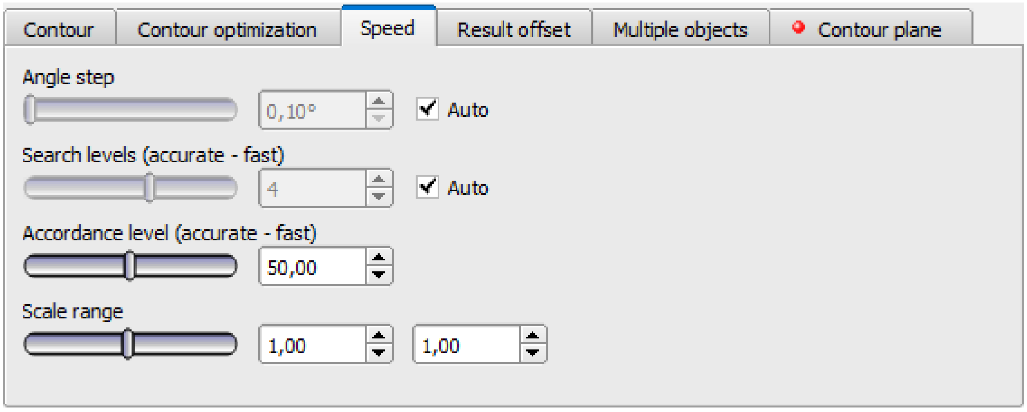
Parameter description:

Parameter	Function
Min. contrast pattern	Minimum required contrast for the taught-in model, in which an edge is accepted as such.
Min. contrast image	Minimum contrast required in current image for an edge to be accepted as one.
Edge transition • fix • fix + inverted	The parameter "Edge transition" can be used to determine the transition between object or contour and background. The way in which the contour is to be recognized is selected: • "fix" = only on the taught-in background • "fix + inverted" = only on the taught-in and inverted background Additional information: see below
Auto	Automatic setting

Additional information: see description Edge transition Detector Contour

8.3.5.4 Speed tab

The execution speed is influenced by the adjustable speed parameters. The search is performed either less finely, i.e. earlier canceled and thus faster, or even finer details are taken into account in the search, i.e. search longer and the search is slower. For further information on how to influence the speed, see also: Contour tab.



Angle step: 0,10° ☑ Auto

Search levels (accurate - fast): 4 ☑ Auto

Accordance level (accurate - fast): 50,00

Scale range: 1,00 1,00

Fig. 135: Contour 3D detector, Speed tab

Parameter description:

Parameter	Function
Angle step	Sensitivity of the search throughout the selected angle range in degrees [°]
Search levels (accurate - fast)	Number of search levels (one search level corresponds to an image with half resolution) • Small value (accurate): Slow search = lower risk (less likely to overlook candidates) • High value (fast): Fast search = higher risk (candidates can be overlooked)
Accordance level (accurate - fast)	Candidates with a degree of compliance below the specified value are already discarded during the search. • Small value (accurate): Late rejection = slower = less risky • High value (fast): Early rejection = quicker = riskier In case of false results, this value can be decreased (more accurate).
Auto	Automatic setting

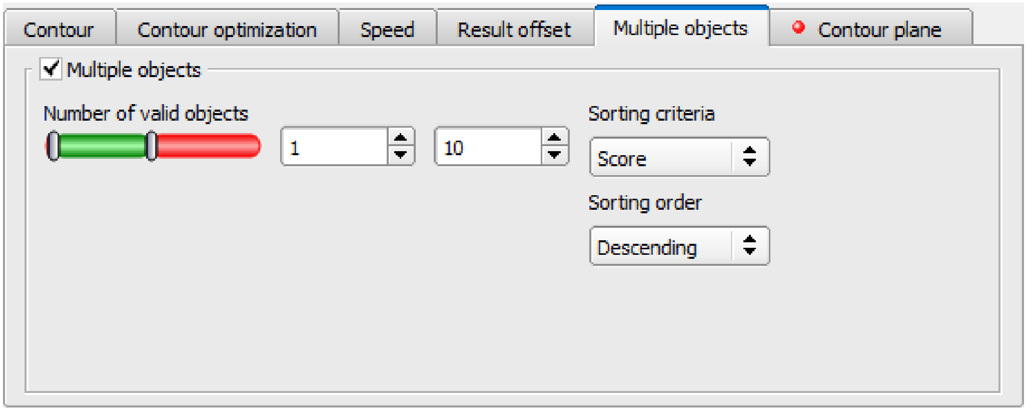
8.3.5.5 Result offset tab

See Chapter: Result offset tab

8.3.5.6 Multiple objects tab

By default, a maximum of one instance of the taught object is found in the image. The "Multiple objects" tab makes it possible to find several instances of a taught contour.

Multiple recognition identifies objects whose contours match the taught-in contour. The output of the object results is sorted according to the set criteria in ascending or descending order.



☑ Multiple objects

Number of valid objects: 1 – 10

Sorting criteria: Score

Sorting order: Descending

Fig. 136: Detector Contour 3D, Multiple objects tab

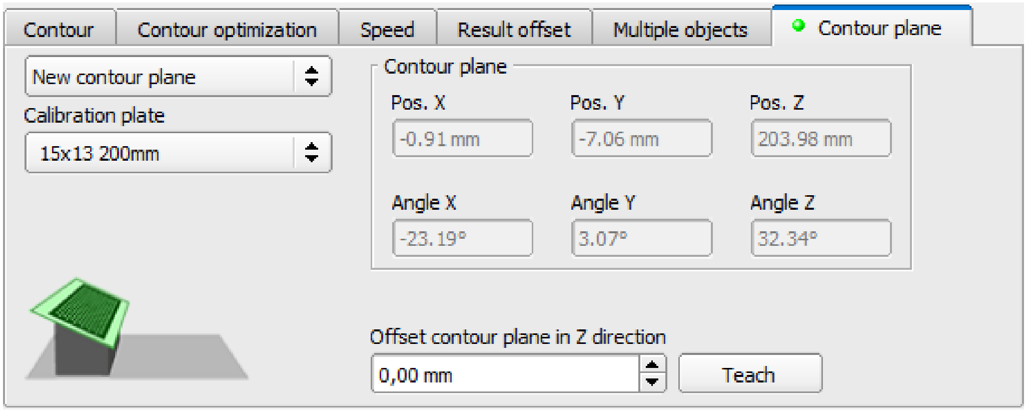
It is also possible to use this function to count objects in the image. The number of objects found can be output as a telegram. The minimum and maximum number of tolerated objects can be specified with the "Number of valid objects" parameter. If the number of objects found is outside this range, the detector result is NOK.

Parameter description:

Parameter	Function
Number of valid objects	This parameter allows you to check whether the number of objects found is within a specified range. If it is within the range, the detector result is OK, otherwise it is NOK. Note: By default, only objects whose score value is above the set threshold are displayed and output.
Sorting criterion • Score • Position X • Position Y • Position Z • Angle X • Angle Y • Angle Z	Sorting criteria according to which objects should be "pre-sorted". • Score value • X-position • Y-position • Z-position • Angle around X • Angle around Y • Angle around Z
Sorting order • Ascending • Descending	Sorting order for the selected sorting criterion. The values of the sorting criteria are sorted in ascending order. The values of the sorting criteria are sorted in descending order.

8.3.5.7 Contour plane tab

The Contour 3D detector requires knowledge of the plane in which the taught-in contours lie (contour plane). In the Contour plane tab, a contour plane can be taught in that differs from the Measurement plane. By default, the contour plane corresponds to the Measurement plane defined during calibration. However, by teaching in a calibration plate, the contour plane can be transformed in the X, Y and Z directions and rotated around the X, Y and Z axes.



New contour plane

Calibration plate: 15x13 200mm

Contour plane

Pos. X	Pos. Y	Pos. Z
-0.91 mm	-7.06 mm	203.98 mm

Angle X	Angle Y	Angle Z
-23.19°	3.07°	32.34°

Offset contour plane in Z direction: 0,00 mm

Fig. 137: Contour 3D detector, Contour plane tab

Parameter description:

Parameter	Function
Selection of "Use measurement plane"	By default, the contour plane corresponds to the Measurement plane defined during calibration.
Selection of "New contour plane"	• Selection of the calibration plate (from the drop-down list) to be used for the definition of the contour plane • "Offset contour plane in Z direction": relative to the contour plane, perpendicular to it • "Teach" button: Contour plane is taught in.
Contour plane	Display (read-only) of X, Y and Z position and X, Y and Z rotation of the contour plane

8.3.6 Detector Target Mark 3D

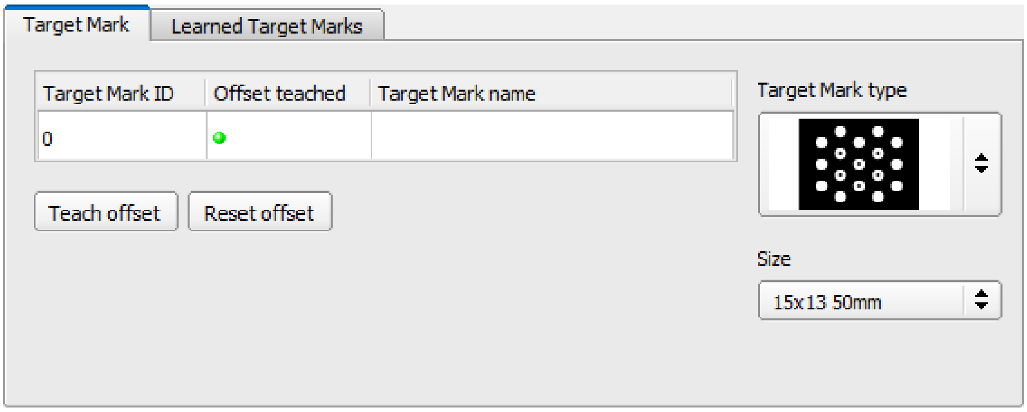
This detector allows the localization of objects in space in six degrees of freedom (position X, Y, Z and Angle X, Y, Z) with the evaluation of a single image.

To use the Target Mark 3D detector, a calibration must first be performed (calibration methods: Calibration plate (Robotics), Hand-Eye calibration (Robotics) or Base-Eye calibration (Robotics)). If no valid calibration is available, no target mark can be localized!

A standardized target mark is required To use the Target Mark 3D detector: a Target Mark with ID function or a Calibration plate.

8.3.6.1 Target Mark tab

In the Target Mark tab, the target mark type can be selected and a target mark name and a result offset can be taught.



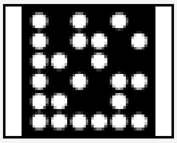
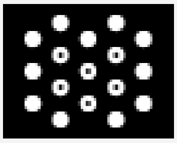
Target Mark ID	Offset teached	Target Mark name
0	●	

Target Mark type

Size: 15x13 50mm

Fig. 138: Target Mark 3D detector, Target Mark tab

Parameter description:

Parameter	Function
Target Mark ID	Unique ID of a target mark (string). (Only for Target Mark type "Target Mark with ID function")
Target Mark name	This can be used to assign a string to a target mark ID.
Target Mark type	Selection of the used target mark from the drop-down list: • Target Mark with ID function  • Calibration plate  A Target Mark with ID function cannot be printed by yourself, because these targets have unique IDs to be identified automatically. The Calibration plate type target marks, however, do not have consecutive IDs and can therefore be printed by yourself. In the installation directory \SensoPart\VISOR Vision Sensor\Documentation\Calibrationplates, the calibration plates available for this purpose can be found as PDF files.
Target Mark size	Selection of the size of the used target mark from the drop-down list (only required for the "Calibration plate" target mark type): • 50 mm • 100 mm • 200 mm • 500 mm
Teach offset	Calculates the result offset, related to the origin of the robot coordinate system, and links it to the currently found target mark. For this purpose, a target mark must be detected. The numerical value can be found in the tab Learned Target Marks.
Reset offset	The offset associated with the currently detected target mark is set to 0.

NOTE:
If there are several marks found in the image, the one closest to the camera is output.

Improvement of execution speed

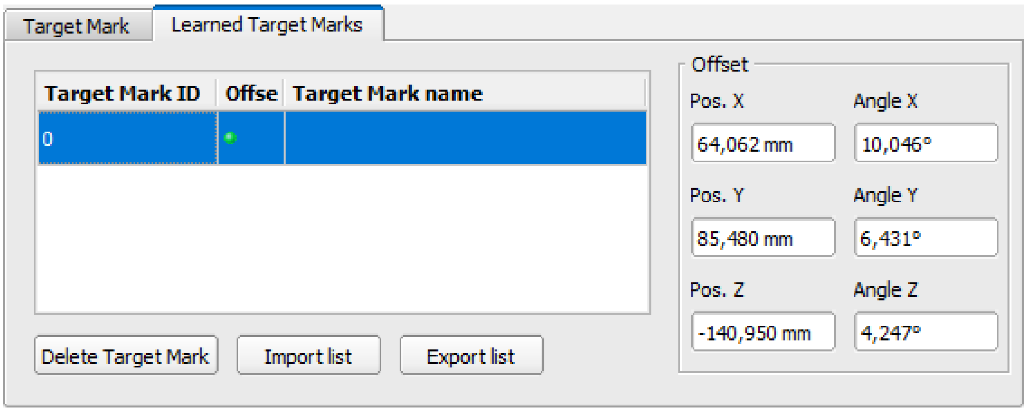
- Reduce resolution (e.g. to WGA) (→Setup step Job)

  ATTENTION:
  This parameter affects all detectors!

- Region of interest for position (yellow frame) only as large as necessary.

8.3.6.2 Learned Target Marks tab

The Learned Target Marks tab lists all target marks detected so far and their offsets. In configuration mode, all detected target marks are automatically added to the list. The list of target marks is part of the job set.



Target Mark ID	Offse	Target Mark name
0	●	

Offset

Pos. X	Angle X
64,062 mm	10,046°

Pos. Y	Angle Y
85,480 mm	6,431°

Pos. Z	Angle Z
-140,950 mm	4,247°

***Fig. 139:** Target Mark 3D detector, Learned Target Marks tab*

Parameter description:

Parameter	Function
Delete Target Mark	Deletes the selected target mark and the associated information (offset and Target Mark name)
Import list	Imports a target marker list as a file (*.tml). The existing list will be overwritten.
Export list	Exports a target mark list as a file (*.tml)
Offset	Calculated values for the selected target mark (read-only)
Pos. X	Offset in X direction
Pos. Y	Offset in Y direction
Pos. Z	Offset in Z direction
Angle X	Rotation around X-axis
Angle Y	Rotation around Y-axis
Angle Z	Rotation around Z-axis

8.3.6.3 Accuracy of detector Target Mark 3D

In order to optimize the accuracy of the detector Target Mark 3D, it is recommended to place the target mark as close as possible to the final working position and to choose the smallest possible distance between the vision sensor and the target mark. Statements on the absolute accuracy of the detector can be taken from the figure "Accuracy depending on distance dO-T Target mark - object (for VISOR® V20, Field of view Medium)".

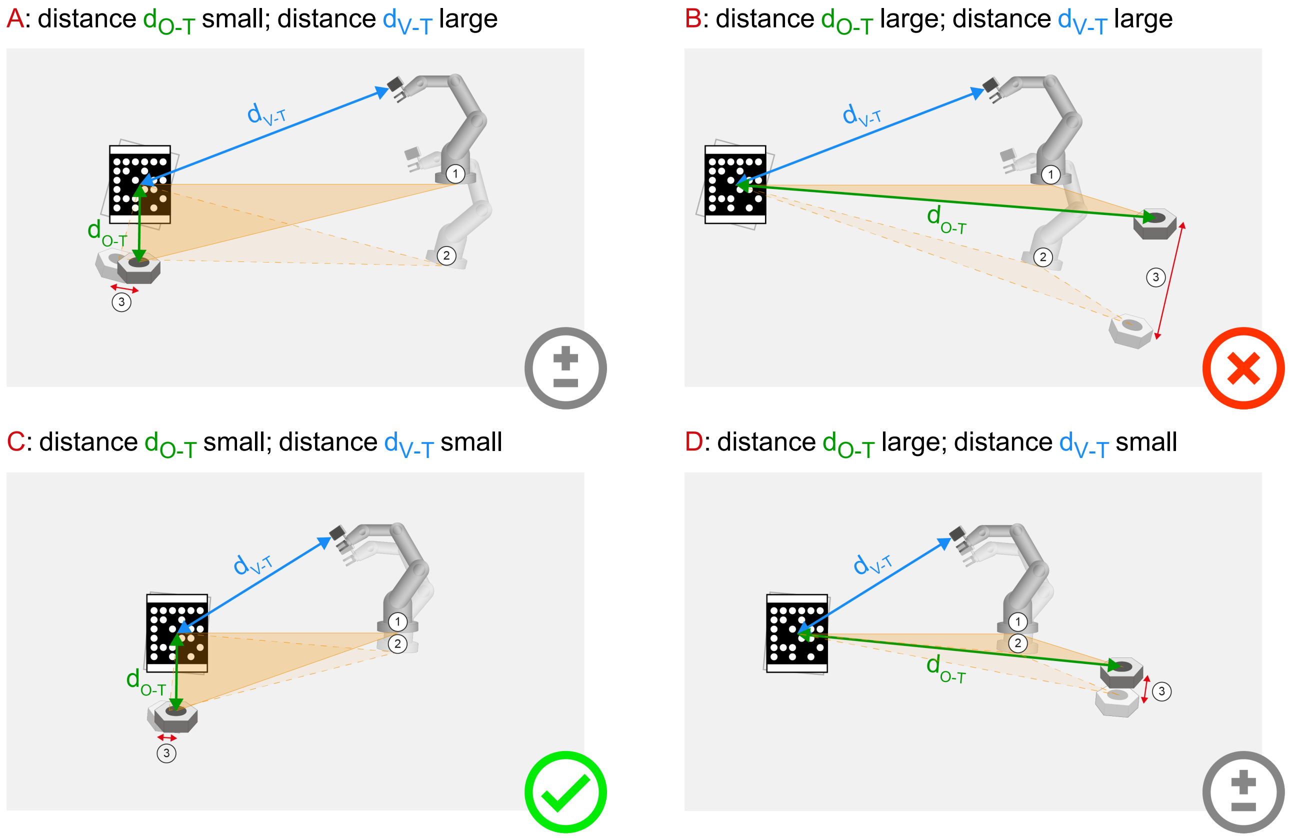
The absolute accuracy of the detector Target Mark 3D depends on the geometric properties of the setup and is influenced by any inaccuracies due to external influences and measurement noise when detecting the target mark.

Example: In the following figure ("Deviations between real and calculated positions"), noise is generated upon detection of the target mark. As a result, the calculated robot, working and gripping positions (②) deviate from the actual positions (①).

A larger distance (d_{V-T}) of the vision sensor (V) to the target mark (T) increases this noise. A smaller distance (d_{V-T}) between the vision sensor (V) and the target mark (T), on the other hand, reduces the field of view of the vision sensor. This increases the proportion of the field of view covered by the target mark. This minimizes the noise influence on the positioning of the target.

In the images on the left (A and C) the target mark is placed close to the working position / object (O) (distance d_{O-T} = small). This minimizes the absolute error at the working position / object (③).

In the images on the right (B and D) the target marker is placed further away from the working position / object (O) (distance d_{O-T} = large). This increases the absolute error at the working position / object (③).



A: distance d_{O-T} small; distance d_{V-T} large

B: distance d_{O-T} large; distance d_{V-T} large

C: distance d_{O-T} small; distance d_{V-T} small

D: distance d_{O-T} large; distance d_{V-T} small

Fig. 140: Deviations between real and calculated positions

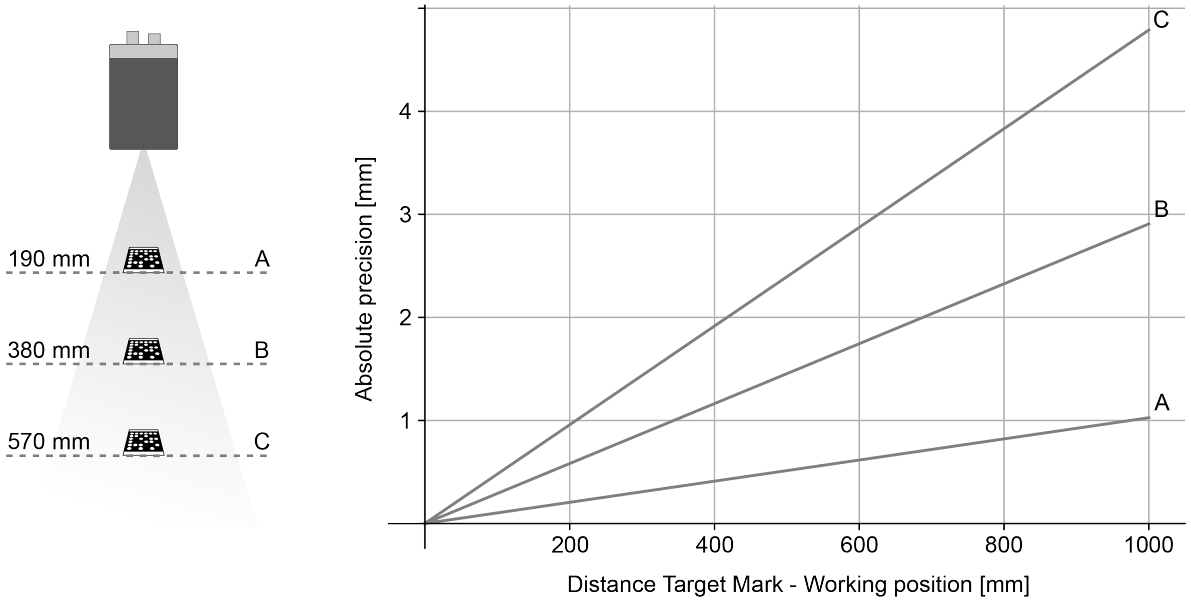


***Fig. 141:** Accuracy depending on distance d_{O-T} Target mark - object (for VISOR® V20, Field of view Medium)*

8.3.7 Detector Contrast

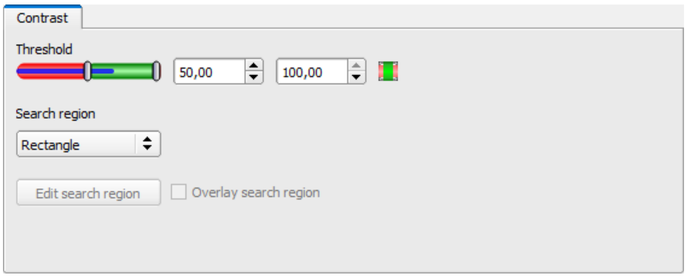
◑ This detector determines the contrast in the selected search field.

For this purpose, all pixels within the region of interest are evaluated with their gray values, and the contrast is calculated. If the contrast value is inside the limits set under "Threshold", the result is positive (OK). The position of the light or dark pixels is not relevant here. It only depends on the spreading of light and dark pixels and their quantity ratio. Highest contrast value with 50% gray value "0" (= black) and 50% gray value "255" (= white).

8.3.7.1 Color Channel tab

See Chapter: Color Channel tab

8.3.7.2 Contrast tab


Contrast

Threshold: 50,00 – 100,00

Search region: Rectangle

***Fig. 142:** Contrast detector, Contrast tab*

Parameter description:

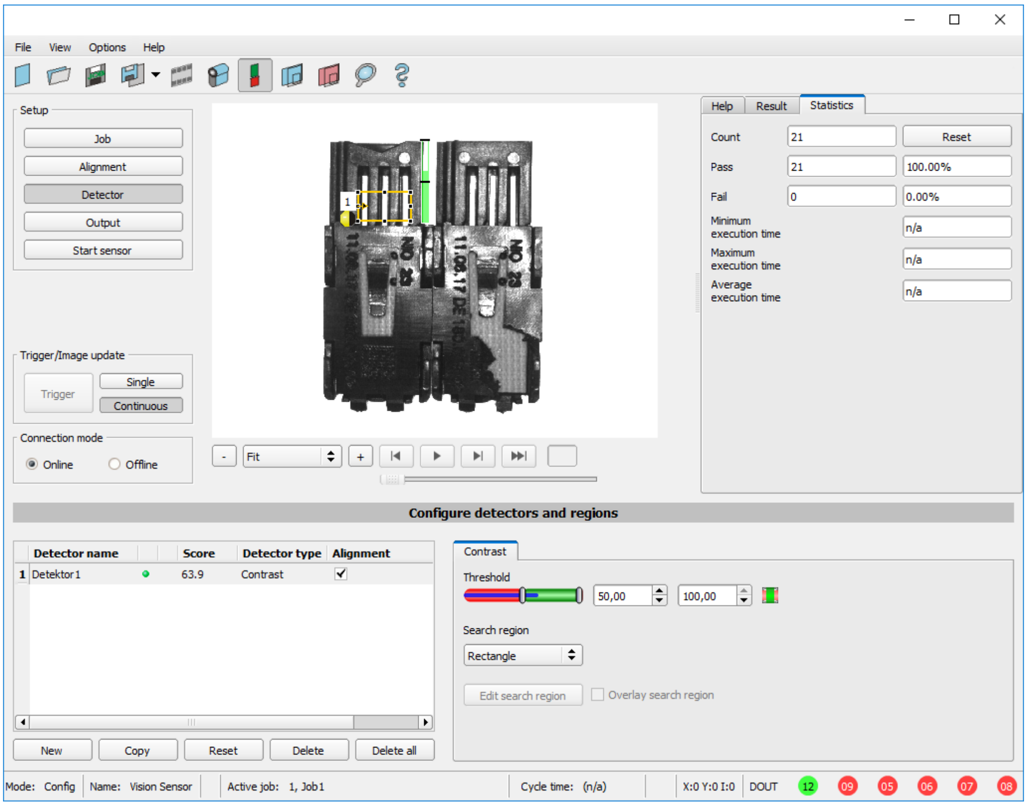
Parameter	Function
Threshold	Specification of the contrast range that is accepted

Parameter	Function
Region of interest (shape)	The shape of the region of interest can be set as Rectangle, Circle, or Free shape. If Freeform is selected, "Edit region of interest" is active.
Edit region of interest	With the parameter "Edit region of interest", areas of the region of interest can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The marked areas can also be inverted. This marks the areas that are important for the execution. See Chapter: Function: Edit pattern / contour
Display region of interest	Enable / disable the display of region of interest edits

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.7.3 Contrast application

In the example, the presence of a contact is checked by means of a contrast detector.



Connection mode: ● Online ○ Offline

Count	21	Reset
Pass	21	100.00%
Fail	0	0.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name		Score	Detector type	Alignment
1	Detektor 1	●	63.9	Contrast	☑

Threshold 50,00 100,00

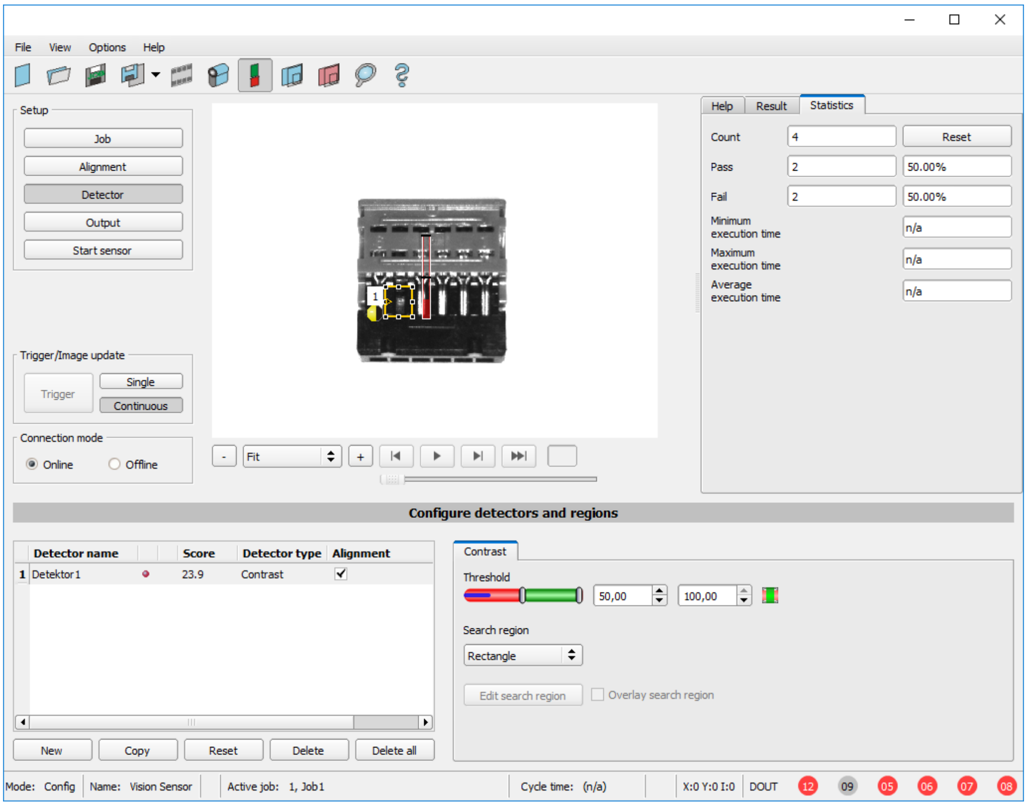
Search region: Rectangle

☐ Overlay search region

Mode: Config | Name: Vision Sensor | Active job: 1, Job 1 | Cycle time: (n/a) | X:0 Y:0 I:0

Fig. 143: Contrast, application example, result OK

The highly reflective, i.e. bright metallic contact, in the midst of the surrounding black plastic housing, is checked for presence with a contrast detector. Since the contrast is very high in this range, the detector provides a high value and thus in conjunction with Alignment a reliable result.



Count	4	Reset
Pass	2	50.00%
Fail	2	50.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Score	Detector type	Alignment
1	Detektor1	23.9	Contrast	☑

Contrast

Threshold 50,00 100,00

Search region: Rectangle

□ Overlay search region

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 144: *Contrast, application example, result NOK*

If the same detector is used at a location where contact is missing, the detector will give a negative result. This is because the contrast is too low between the black environment and the now visible black background of the contact.

Function detector Contrast

The dark and light pixels are scored for number and light / dark intensity.

The position of the bright or dark pixels in the region of interest is irrelevant.

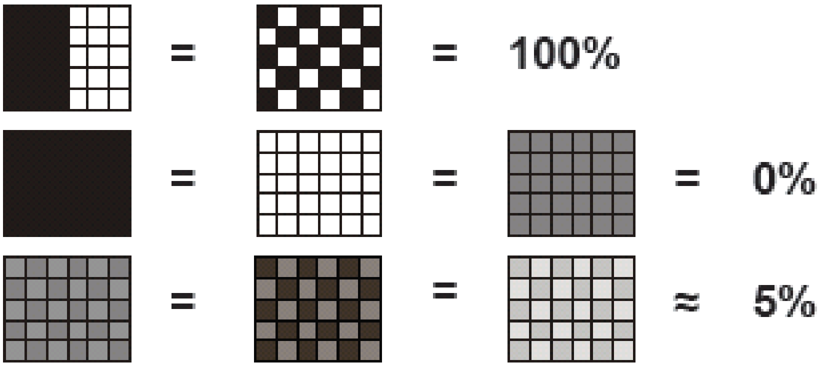
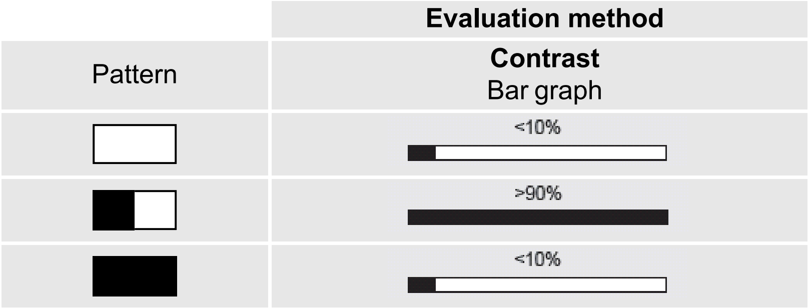


Fig. 145: *Contrast examples*



	Evaluation method
Pattern	**Contrast** Bar graph
	<10%
	>90%
	<10%

Fig. 146: Contrast explanation

8.3.8 Detector Gray

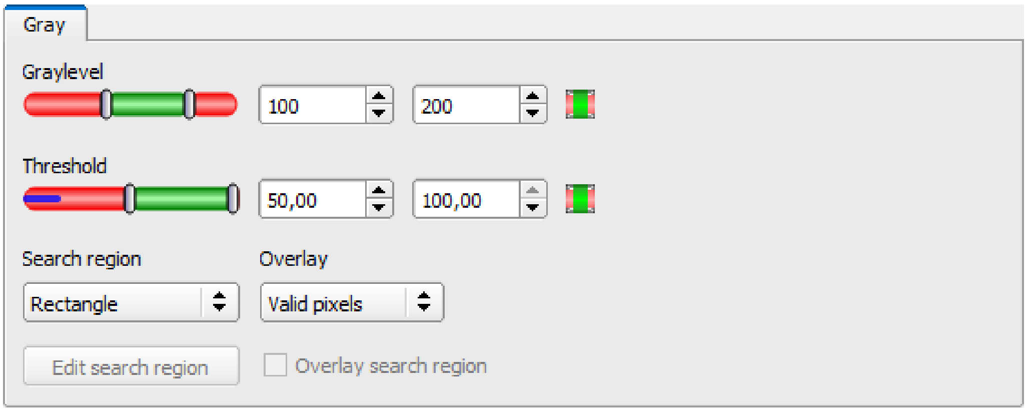
This detector is suitable for determining the gray values in the selected search field.

In the first step of this detector, the parameter "Graylevel" is used to determine the value range of the gray values that may occur in the region of interest. In the second step, under "Threshold", the area percentage (in %) of the region of interest is defined which must have the gray values defined in Step 1 in order to provide a positive result.

8.3.8.1 Color Channel tab

See Chapter: Color Channel tab

8.3.8.2 Gray tab



Gray

Graylevel: 100 – 200

Threshold: 50,00 – 100,00

Search region: Rectangle

Overlay: Valid pixels

Fig. 147: Gray detector, Gray tab

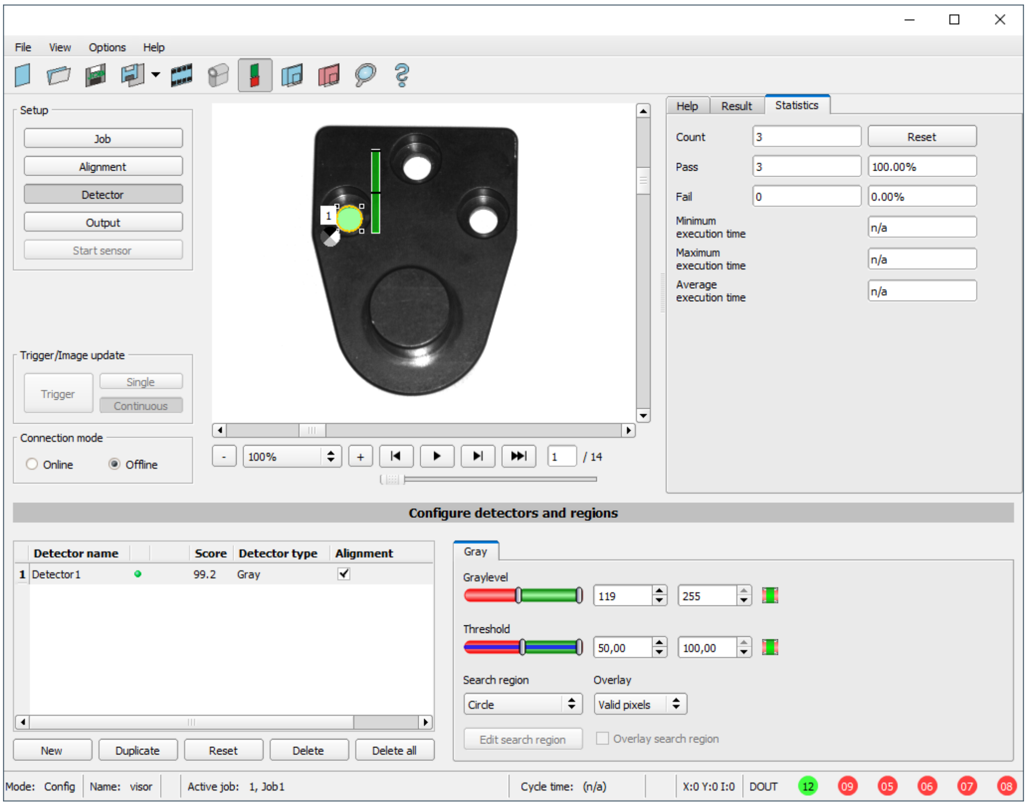
Parameter description:

Parameter	Function
Graylevel	Value range of the gray values that may occur in the region of interest
Threshold	Percentage of the area that must have the gray values defined under "Graylevel"
Invert button 🟩 / 🟥	With the respective inversion of "Graylevel" or "Threshold", all conceivable combinations can be set, even those in which e.g. only gray values are allowed at the top and bottom of the value range. The position of the light or dark pixels is not relevant here.
Region of interest (shape)	The shape of the region of interest can be set as Rectangle, Circle, or Free shape. If Freeform is selected, "Edit region of interest" is active.

Parameter	Function
Overlay	Selection of the pixels that have a gray value within (valid pixels) or outside (invalid pixels) of the gray level definition under "Graylevel". These are then marked in color as a selection aid. Thus, e.g. noise pixels / areas that are not covered by the gray value range can very easily be detected.
Edit region of interest	With the parameter "Edit region of interest", areas of the region of interest can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The marked areas can also be inverted. This marks the areas that are important for the execution. See Chapter: Function: Edit pattern / contour
Display region of interest	Enable / disable the display of region of interest edits

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.8.3 Gray application



Connection mode: ○ Online ● Offline

Count	3	
Pass	3	100.00%
Fail	0	0.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name		Score	Detector type	Alignment
1	Detector1	●	99.2	Gray	☑

Graylevel: 119 – 255

Threshold: 50,00 – 100,00

Search region: Circle

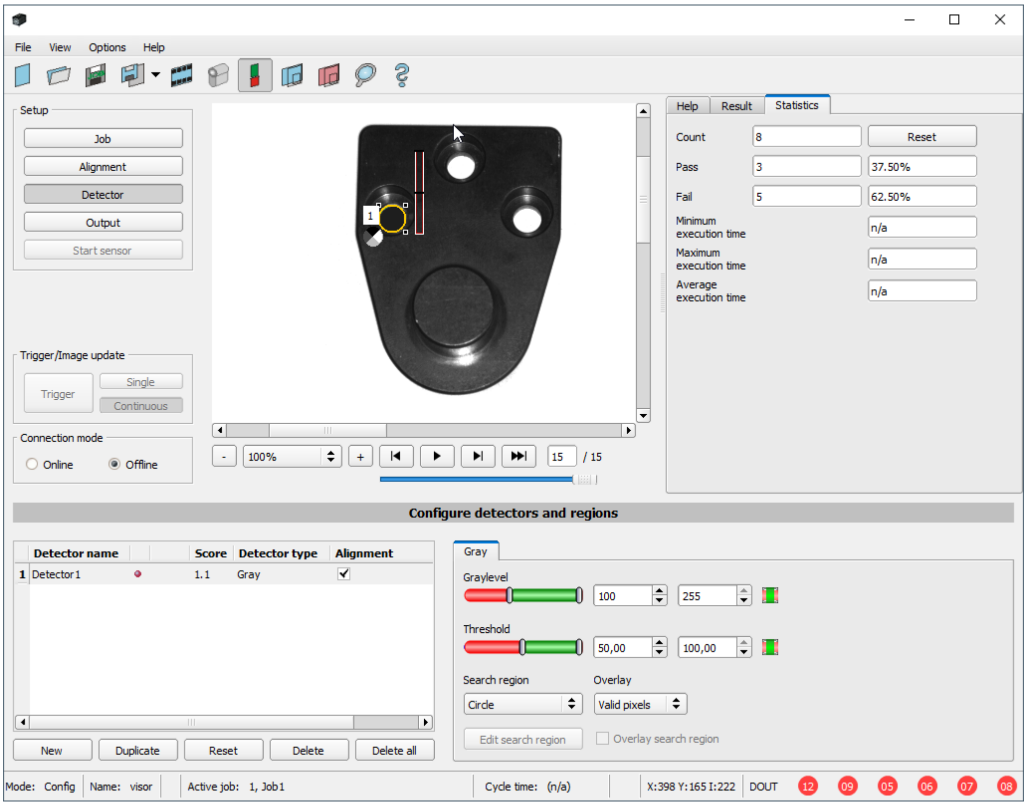
Overlay: Valid pixels

□ Overlay search region

Mode: Config | Name: visor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 148: *Gray, application example, positive result*

Requested characteristic is present in the region of interest, gray values within the required threshold values = positive result



Count: 8 — Reset
Pass: 3 — 37.50%
Fail: 5 — 62.50%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

Connection mode: ○ Online ● Offline

100% — 15 / 15

Detector name	Score	Detector type	Alignment
1 Detector1	1.1	Gray	☑

Gray

Graylevel: 100 – 255

Threshold: 50,00 – 100,00

Search region: Circle

Overlay: Valid pixels

□ Overlay search region

Mode: Config | Name: visor | Active job: 1, Job1 | Cycle time: (n/a) | X:398 Y:165 I:222

Fig. 149: Gray, application example, negative result

Searched characteristic is not present in the region of interest. I.e. mean value of the gray values in the region of interest not within the threshold values = negative result.

Note for determining the gray values

If the cursor is placed anywhere in the image area, the corresponding X and Y coordinates as well as the gray value ("I" = intensity) are displayed in the second field from the right in the status line at the lower edge of the image.

Detector Gray functionality:

The permissible gray value range is defined with the two limits of the "Graylevel" slider.

All pixels within this gray value range and within the defined working zone (yellow frame) are added together. The proportion of the number of all the pixels in the working zone (yellow frame) and of the number of pixels in the authorized gray value range represents the result of this detector.

If this result is within the limits set on the "Threshold" slider, the result is positive.

The position of the gray value pixels on the screen is of no importance.

Example: (when the gray level slider is set to very dark values):

The two images deliver exactly the same result with detector Gray, because 9 of 25 pixels are recognized as dark.

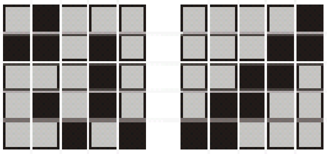


Fig. 150: Graylevel, Example 1

If the threshold value were set to 10 in this example, the following images would produce a positive result:

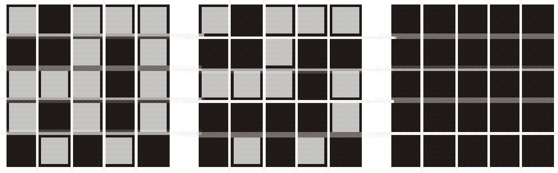


Fig. 151: Graylevel, Example 2

8.3.9 Detector Brightness

☼ The Brightness detector calculates the mean of the gray values of all pixels within the region of interest.

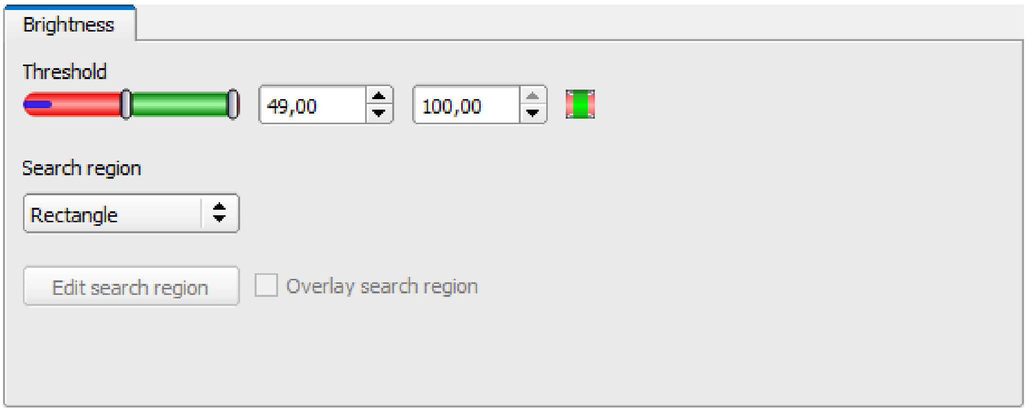
The two threshold value sliders of the "Threshold" parameter are used to set the permissible range for this brightness average. As soon as the calculated mean is within these two limits, the result is positive. The result is standardized to %. The position of the bright or dark pixels in the region of interest is irrelevant.

The detector can be used effectively if the position of the searched object in the image is absolutely unchanged from inspection to inspection. If deviations in the position may occur, Alignment must be used.

8.3.9.1 Color Channel tab

See Chapter: Color Channel tab

8.3.9.2 Brightness tab



Brightness

Threshold

49,00 100,00

Search region

Rectangle

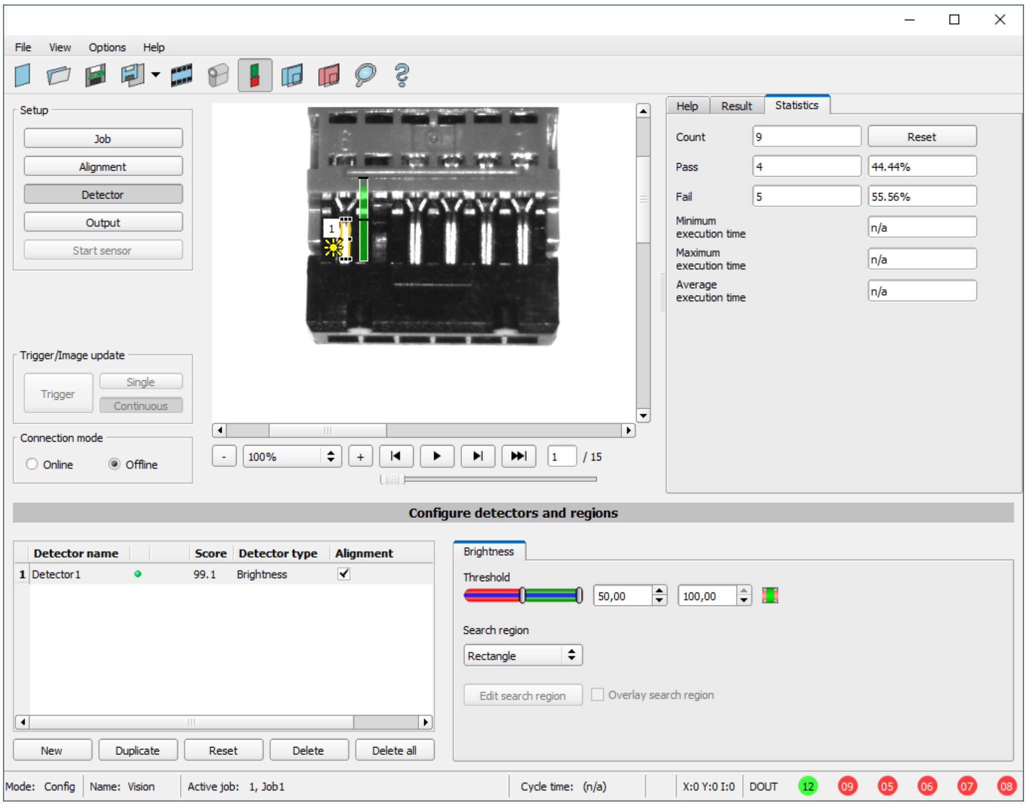
Fig. 152: Brightness detector, Brightness tab

Parameter description:

Parameter	Function
Threshold	Specification of the brightness range that is accepted
Region of interest (shape)	The shape of the region of interest can be set as Rectangle, Circle, or Free shape. If Freeform is selected, "Edit region of interest" is active.
Edit region of interest	With the parameter "Edit region of interest", areas of the region of interest can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The marked areas can also be inverted. This marks the areas that are important for the execution. See Chapter: Function: Edit pattern / contour
Display region of interest	Enable / disable the display of region of interest edits

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.9.3 Brightness application



Connection mode: ○ Online ● Offline

100% 1 / 15

Count	9	Reset
Pass	4	44.44%
Fail	5	55.56%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name		Score	Detector type	Alignment
1	Detector1	●	99.1	Brightness	☑

Threshold: 50,00 100,00

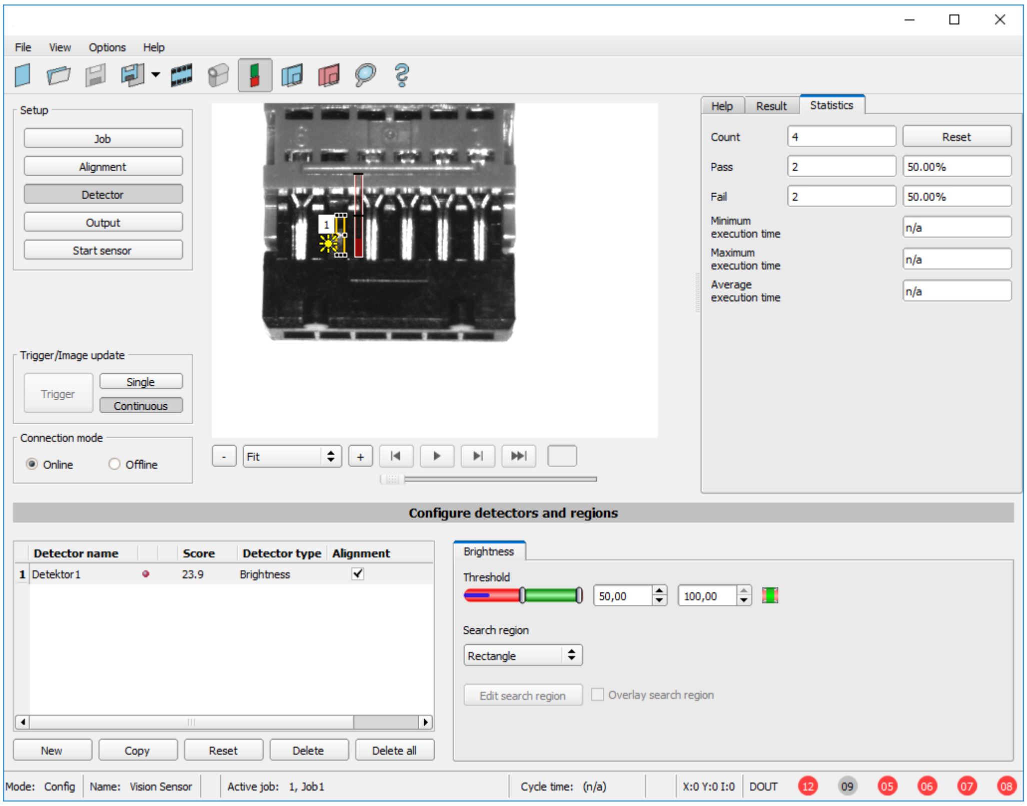
Search region: Rectangle

☐ Overlay search region

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 153: Brightness, application example, result OK

The contact is present at the searched location and the mean value of the gray values in the region of interest therefore provides a very high value (close to 100%). This means the current value is within the requested threshold limits and the result is positive = contact present.



Connection mode: ● Online ○ Offline

Count	4
Pass	2	50.00%
Fail	2	50.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Detector name	Score	Detector type	Alignment
1	Detektor1	23.9	Brightness	☑

Threshold: 50,00 – 100,00

Search region: Rectangle

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 154: *Brightness, application example, result NOK*

Contact does not exist at the requested location, and the mean of the gray values in the search area therefore provides a very low value (close to 0%). This means the current value is not within the requested threshold limits and the result is negative = contact not available.

Examples: Brightness value as mean of the gray values.

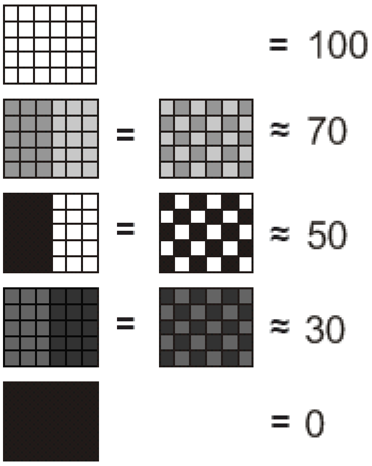


Fig. 155: *Brightness, examples*

8.3.10 Detector BLOB

The BLOB detector can be used to identify and count one or more objects that share common characteristics such as the same gray scale areas, area, perimeter, etc. The objects can be evaluated and sorted according to these user-defined criteria and their position can be detected.

- "BLOB", abbreviation for "Binary Large Object" or "Binary Labeled Object".
- Basic image processing function for evaluating **contiguous** surfaces and objects in an image.
- Differentiation of the individual objects on the basis of simple features, such as area, width, height, etc.

Typical applications

- Counting of objects
- Differentiation / classification of objects in the image by:
  - Size, area, contour
  - Shape, geometry
  - Position, Orientation
- Position, side
- Position detection, e.g. in robotics
- Surface inspection

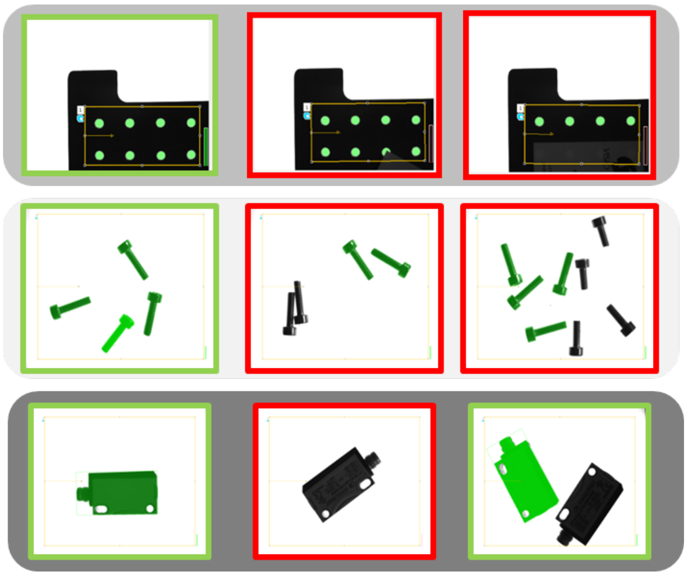
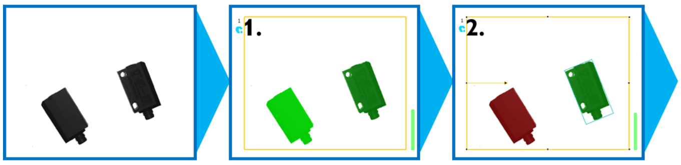


Fig. 156: Typical applications: Count, classify / sort, location / side

Simple configuration in 3 steps



3.
Configuration and data output,
Ethernet, M12, 4-pin
- Ethernet
- EtherNet/IP
- PROFINET
- FTP/SMB Archiving

1. Binarization

Separation between background and relevant object

Absolute threshold

Dynamic threshold

2. Filtering of detected BLOBs

Based on various characteristics such as: area, perimeter, orientation, location, etc.

Features tab

3. Data output

Definition of data output telegram and sorting of results.

Sorting tab

Telegram tab

8.3.10.1 Color Channel tab

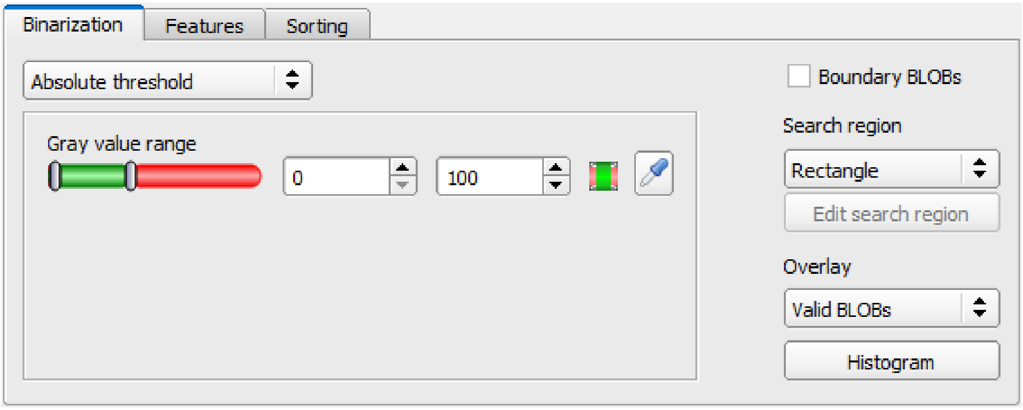
See Chapter: Color Channel tab

8.3.10.2 Binarization tab

In this tab, all parameters for binarization of a BLOB can be set.

Binarization is the first step in BLOB evaluation and is used to separate relevant objects from the background of the image. This is done by converting the gray level image to a pure black and white image, i.e. binary image.

Two methods of binarization are available: "Absolute threshold" and "Dynamic threshold".



Absolute threshold

Gray value range: 0 – 100

□ Boundary BLOBs

Search region: Rectangle

Overlay: Valid BLOBs

Fig. 157: Detector BLOB, tab Binarization

Selection of binarization method

Parameter	Function
Absolute threshold	The binarization switching threshold is set to an absolute gray value in the value range of 0 ... 255. Additional information: Absolute threshold

Parameter	Function
Dynamic threshold	The dynamic switching threshold is automatically adjusted for each image to a position optimized according to statistical methods in order to be able to differentiate between foreground and background as well as possible. Additional information: Dynamic threshold

Boundary objects, overlays, and histogram

Parameter	Function
Boundary objects	If the "Boundary BLOBs" checkbox is activated, the selected BLOBs / objects are taken into account, even if they are not completely within the yellow region of interest. Note: BLOBs are also considered to be boundary BLOBs when they touch or are partially covered by an area that has been masked with "Edit pattern" / "Edit region of interest" (even if the masked / hidden area is inside the region of interest). Additional information: Boundary objects
region of interest	The shape of the region of interest can be set to: "Rectangle", "Circle" or "Free shape". In the mode "Free shape", the masking function "Edit region of interest" can be used to hide areas that are not relevant.
Edit region of interest	By clicking on "Edit region of interest", the window opens for editing the region of interest. Additional information: Function: Edit pattern / contour
Overlay	"Valid BLOBs": All valid BLOBs that meet the feature criteria within the set feature thresholds are marked green. Invalid BLOBs are marked red. "BLOB contour": All valid BLOBs which fulfill the feature criteria within the set feature thresholds are marked with a green contour line. Invalid BLOBs are not marked.
Histogram	Clicking on "Histogram" opens the Histogram window. Additional information: Histogram

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.10.2.1 Absolute threshold

Parameter description:

Parameter	Function
Absolute threshold	The upper and lower limit defines the range of valid gray values of pixels assigned to the respective BLOB.
Gray value range	Setting the upper and lower limit of the gray values for binarization.
Invert button 🟥🟩🟥 / 🟩🟥🟩	The Invert button can be used to include or exclude the relevant range between the thresholds in the selection. Default: red/green/red = included.
Pipette symbol ✎	By clicking on the Pipette button, the cursor changes into a pipette symbol. If the cursor is now moved and a pixel within the image is clicked, the thresholds of the "Absolute switching threshold" are set to +/- 10 gray values above or below the gray value of the selected pixel (maximum value range 0 ... 255).

Additional information:

Histogram

8.3.10.2.2 Dynamic threshold

The dynamic switching threshold can be used if the searched BLOBs / objects have significantly different gray values than the background, and the brightness / illumination fluctuates evenly over the entire image.

If the image brightness changes evenly across the entire image, the two switching thresholds are automatically readjusted for each image. (When using "Absolute threshold", the thresholds ought to be readjusted manually!)

Please note:

- When using the dynamic switching threshold, the thresholds are recalculated and readjusted with each new image / evaluation.
- Changing lighting conditions or surface conditions / reflectivity can affect the result.

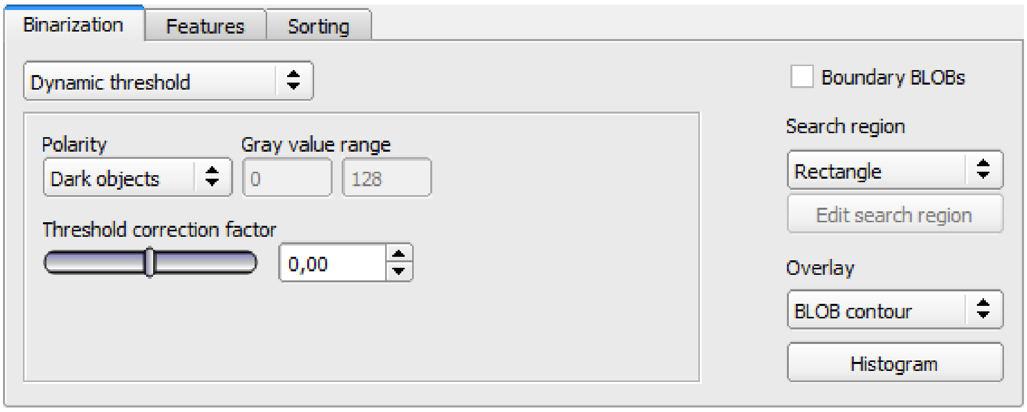


Fig. 158: *BLOB detector, Binarization tab, Dynamic threshold*

Parameter description:

Parameter	Function
Dynamic threshold	The switching threshold is automatically adjusted for each image to a position optimized according to statistical methods in order to be able to distinguish as well as possible between foreground and background.
Polarity	Defines whether BLOB / object is brighter or darker than the background.
Gray value range	Gray value thresholds for binarization
Correction factor for threshold	The binarization threshold value can be shifted in the direction of the foreground or the background brightness via this correction factor.

Additional information:

Histogram

8.3.10.2.3 Histogram

In this window, the histogram of the gray values inside the yellow region of interest and the chosen thresholds are shown.

In the example shown here, clear maxima for foreground and background can be recognized. The switching threshold for binarization is positioned approximately in the middle in between.

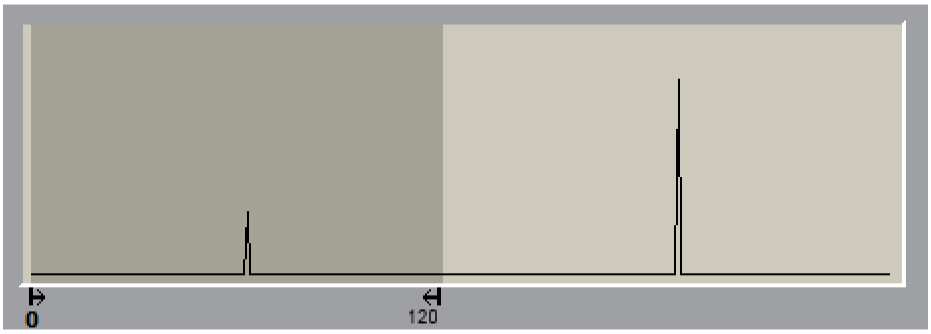


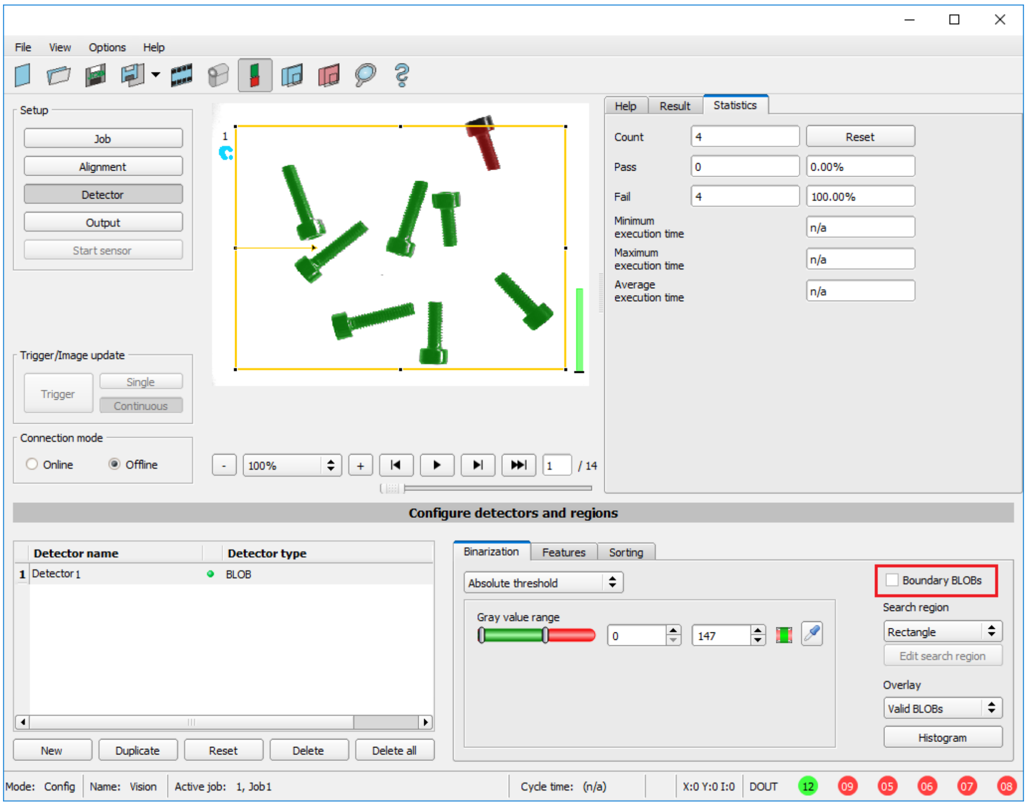
Fig. 159: Binarization tab, histogram

8.3.10.2.4 Boundary objects

If the "Boundary BLOBs" checkbox is activated, the selected BLOBs / objects are taken into account, even if they are not completely within the yellow region of interest. (Of course, the objects must always meet the BLOB features within the selected thresholds.)

NOTE:
BLOBs are also considered to be boundary BLOBs when they touch or are partially covered by an area that has been masked with "Edit pattern" / "Edit region of interest" (even if the masked / hidden area is inside the region of interest).

Example 1: Boundary objects, touches outer yellow region of interest



Connection mode: ○ Online ● Offline

100% 1 / 14

Count: 4

Pass: 0 0.00%

Fail: 4 100.00%

Minimum execution time: n/a

Maximum execution time: n/a

Average execution time: n/a

Detector name	Detector type
1 Detector 1	BLOB

Absolute threshold

Gray value range: 0 147

□ Boundary BLOBs

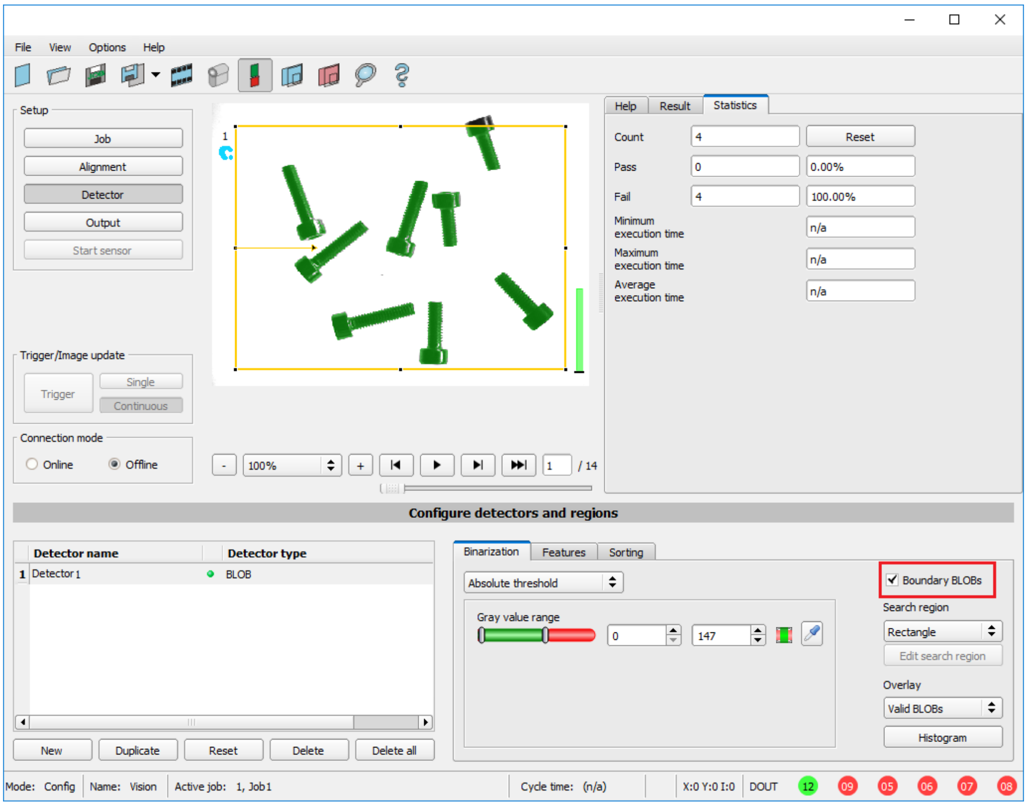
Search region: Rectangle

Overlay: Valid BLOBs

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 160: Edge objects, example 1.1

The BLOB touches the outer, yellow region of interest. It is not detected / marked as a valid BLOB because "Boundary objects" is NOT enabled.



Count: 4
Pass: 0, 0.00%
Fail: 4, 100.00%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

Connection mode: ○ Online ● Offline
100% | 1 / 14

Detector name	Detector type
1 Detector 1	BLOB

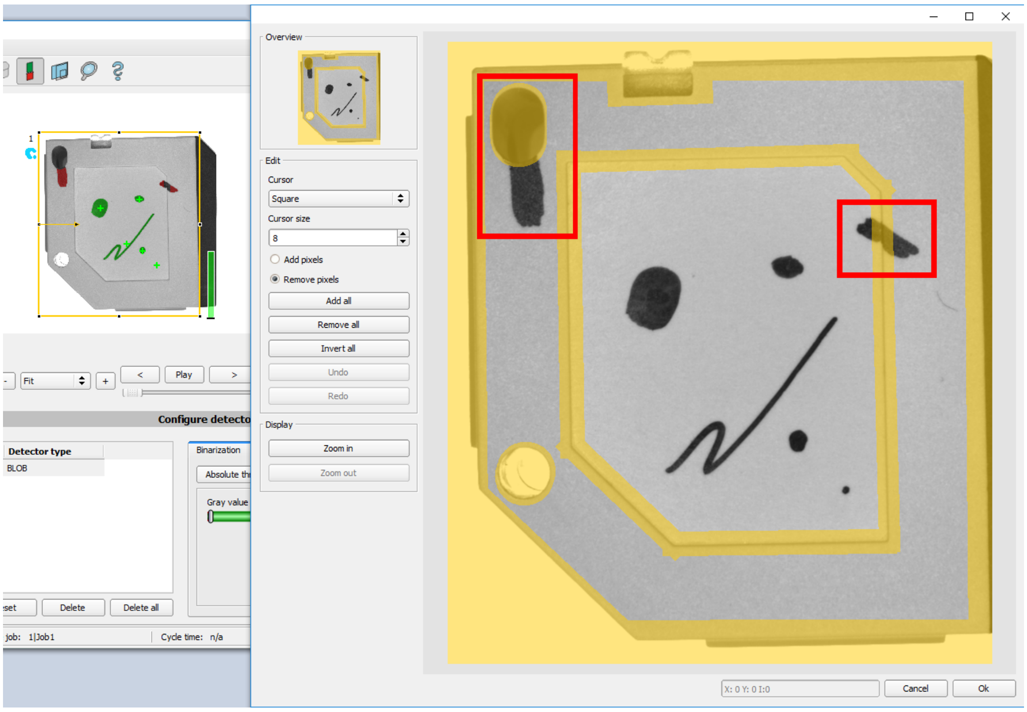
Absolute threshold
Gray value range: 0 – 147
☑ Boundary BLOBs
Search region: Rectangle
Overlay: Valid BLOBs

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT

Fig. 161: Edge objects, example 1.2

The BLOB touches the outer, yellow region of interest. It will now be detected / marked as a valid BLOB because "boundary objects" is ACTIVE.

Example 2, boundary objects touching the inner, masked area.

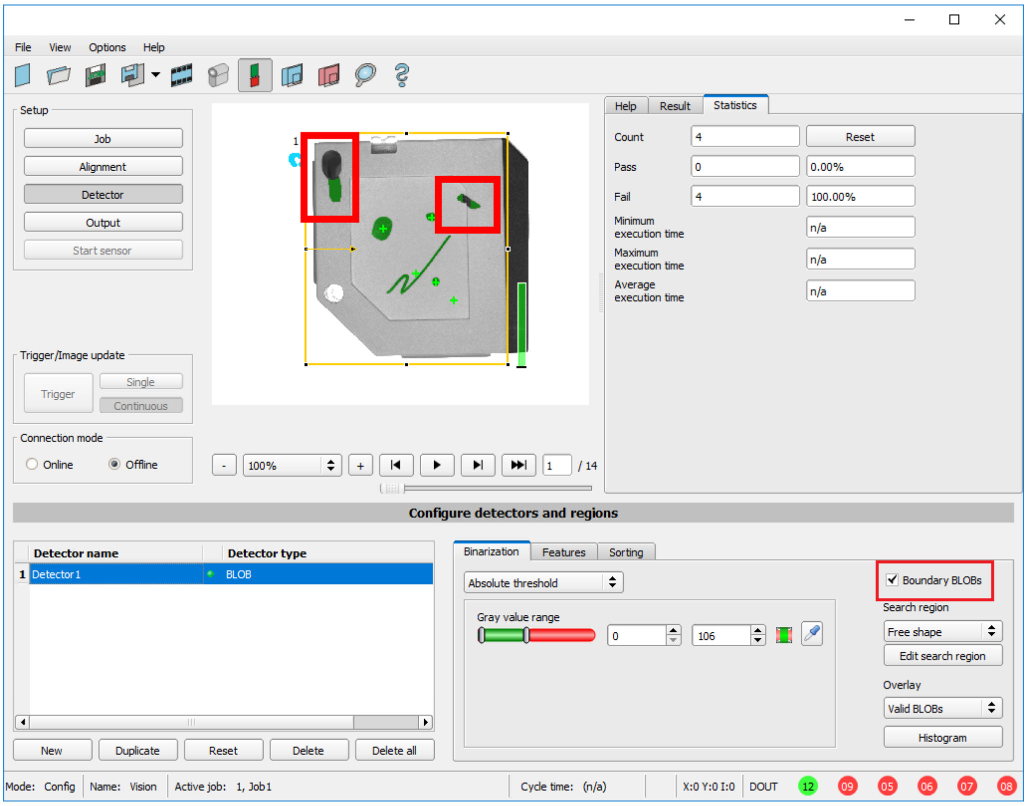


Cursor: Square
Cursor size: 8
○ Add pixels
● Remove pixels

Fit
Absolute th
Gray value
job: 1|Job1 | Cycle time: n/a
X: 0 Y: 0 I:0

Fig. 162: Edge objects, example 2.1

The BLOBs touch the inner, yellow, masked areas. They are not detected as valid BLOBs, as "Boundary objects" is NOT activated.



Connection mode: ○ Online ● Offline

100% 1 / 14

Count	4	
Pass	0	0.00%
Fail	4	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type
1	Detector1	BLOB

Absolute threshold

Gray value range 0 106

☑ Boundary BLOBs

Search region: Free shape

Overlay: Valid BLOBs

Mode: Config Name: Vision Active job: 1, Job1 Cycle time: (n/a) X:0 Y:0 I:0

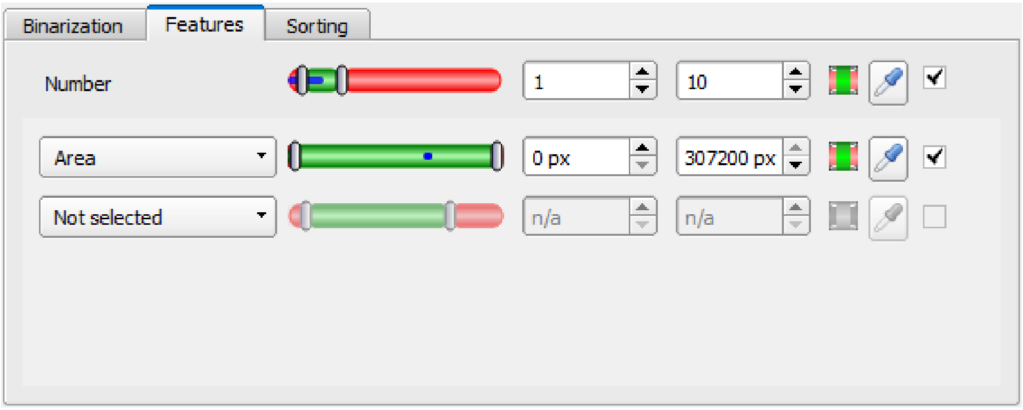
Fig. 163: Edge objects, example 2.2

The BLOBs touch the inner, yellow masked areas, but they are now detected / marked as valid BLOBs, because the option "border objects" is ACTIVE.

8.3.10.3 Features tab

In the Features tab, the features / filter criteria to distinguish between valid and invalid BLOBs / objects are defined. Only the valid BLOBs are processed further, e.g. for data output.

Example: If the thresholds for the "Area" feature have been set to the range 100 ... 150 (pixels), only blobs with an area within this range will be recognized as valid (green).



Number	1	10	☑
Area	0 px	307200 px	☑
Not selected	n/a	n/a	□

Fig. 164: BLOB detector, Features tab

Parameter description:

Parameter	Function
Number	The parameter "Number" can be used to check the number of found and valid BLOBs in addition to the characteristics. For this purpose, the upper and lower limit for the accepted number of BLOBs (max. 10,000) is determined. • Detector result positive: Number of valid (filtered) BLOBs is within the range of "Number". • Detector result negative: Number of valid (filtered) BLOBs is outside the range of "Number". If the number of BLOBs is outside the defined limits, the detector result is negative, although valid BLOBs in the image are marked green. If the detector counts more than 10,000 BLOBs (maximum), the detector result is negative and processing is stopped. ℹ **NOTE:** Defect detection via number = 0
Invert button 🟩 / 🟥	The Invert button can be used to include or exclude the relevant range between the thresholds in the selection. Default: red/green/red = included.
Pipette symbol 🖉 (Number)	By clicking this symbol, both limits of "Number" are set to the exact number of BLOBs found in the image.
Pipette symbol 🖉 (Feature)	By clicking on the "Pipette button", the cursor changes into a pipette symbol. When the cursor is moved and a pixel within a valid (green) BLOB is clicked, the thresholds of the selected feature are set to +/- 10% of the value of the BLOB that was clicked. Example: If the feature "Area" was selected and clicked with an active pipette on a pixel within a valid BLOB, the two thresholds for "Area" are set to +/-10% of the calculated number of pixels (= area) of the selected BLOB.
Checkbox (Default: Active)	• Active: Feature is calculated, filtered (limits adjustable), and available for data output. • Inactive: Feature is calculated, NOT filtered, but is available for data output.

First level features: BLOB type / geometric model

The features of the first level ("Area", "Area (incl. holes)", "Contour length", etc.) are calculated directly from the BLOB data, i.e. the pixels belonging to the BLOB. For further features, a geometric model is first fitted to the data via a best-fit line. The features are then based on this model and not directly on the pixels belonging to the BLOB.

Feature	Function
Area	Area of the BLOB, without holes, in pixels. Corresponds to the number of pixels belonging to the BLOB.
Area (incl. holes)	Area of the BLOB, with holes, in pixels. Corresponds to the number of pixels within the outer contour.
Contour length	Number of pixels of the outer contour of the BLOB.
Compactness	Compactness of the BLOB (ideal circle = 1, all other > 1) The stronger the shape of the BLOB deviates from an ideal circle, the larger the value for compactness will be. Value range 1 ... 100 (limited at 100; BLOBs with higher values are marked as invalid)
Average brightness	Average gray scale value of all the pixels that belong to the BLOB.

Some features are calculated based on a given geometric model. For example, eccentricity is based on an ellipse fit of the object.

Note: When the calibration is activated in the "Job" setup, the value output can also be in world coordinates, e.g. in millimeters.

BLOB type / Geometric Model	Function
Center of gravity (CG)	Focus of all pixels belonging to the BLOB.
Rectangle, paraxial (R1)	Enclosing rectangle parallel to Y axis and X axis. Outliers are not eliminated.
Rectangle, minimum area (R2)	Enclosing rectangle with smallest area. Outliers are not eliminated.
Circle, fit (C1)	Circle-fit, not enclosing, outlier correction (robust against outliers)
Ellipse, equivalent (E1)	Equivalent ellipse, based on moments of area.

Features / second level: BLOB type parameter

[*] Possibility of value output in world coordinates [mm] when calibration is activated

Feature	Relevant for	Function	[*]
Height	R1, R2, E1	Height of geometric element. Height ≥ 0, height ≤ width. The orientation is chosen in a way that width is always greater than the height. (Exception: R1, Rectangle, paraxial: Height always in vertical orientation = parallel to Y axis)	✓
Width	R1, R2, E1	Width of geometric element. Width ≥ 0, width ≥ height. The orientation is chosen in a way that width is always greater than the height. (Exception: R1, Rectangle, paraxial: Width always in horizontal orientation = parallel to X axis)	✓
Pos. X	CG, R1, R2, C1, E1	X-coordinate of the center of the fitted geometric element (center of gravity, rectangle, circle, ellipse).	✓
Pos. Y	CG, R1, R2, C1, E1	Y-coordinate of the center of the fitted geometric element (center of gravity, rectangle, circle, ellipse).	✓
Pos. Z	CG, R1, R2, C1, E1	Z coordinate in the world coordinate system (derived from result offset)	✓
Angle X	CG, R1, R2	Rotation values (angle) around the X-axis of the world coordinate system (derived from result offset)	
Angle Y	CG, R1, R2	Rotation values (angle) around the Y-axis of the world coordinate system (derived from result offset)	
Angle Z	CG	Rotation values (angle) around the Z-axis of the world coordinate system (derived from result offset)	

Feature	Relevant for	Function	[*]
Angle Z (360)	R1, R2, E1	Orientation of the width (long axis) of the object in degrees (°), (range -180 ... + 180 °, 0 ° = east, counterclockwise). Please refer to the following as well: Feature Angle	
Angle Z (180)	R1, R2, E1	Orientation of the width (long axis) of the object in degrees (°), (range -90 ... + 90 °, 0 ° = east, counterclockwise). Please refer to the following as well: Feature Angle	
Pose 3D (X, Y, Z, Angle X, Y, Z)	CG	X, Y and Z coordinates as well as Angle X, Y and Z Note: Only one pose 3D can be selected.	✓
Pose 3D (X, Y, Z, Angle X, Y, Z 360)	R1, R2, C1, E1	X, Y and Z coordinates as well as Angle X, Y and Z (360)	✓
Pose 3D (X, Y, Z, Angle X, Y, Z 180)	R1, R2, C1, E1	X, Y and Z coordinates as well as Angle X, Y and Z (180)	✓
Axial ratio	E1	Ratio of the long to the short axis (a / b)	
Face up / down, area	E1	Face up / down distinction, based on area, indicated by sign. Please refer to the following as well: Feature Face up / down	
Radius	C1	Specifies the radius of the fitted circle.	✓
Deviation, inside	C1	Returns the largest deviation between the BLOB contour and the contour of the geometric element (deviation inside the fitted circle). Please refer to the following as well: Feature Deviation	✓
Deviation, outside	C1	Returns the largest deviation between the BLOB contour and the contour of the geometric element (deviation outside the fitted circle). Please refer to the following as well: Feature Deviation	✓
Deviation, mean	C1	Returns the mean of the absolute "inside" and "outside" deviation values between the BLOB contour and the contour of the geometric element. Please refer to the following as well: Feature Deviation	✓

Feature Angle

With the features "Angle Z (180)" and "Angle Z (360)" the orientation of the object can be determined. The angle always indicates the orientation of the width axis (width is the longest side of an object). The angles are specified in [degrees °].

The "Angle (180)" feature has a rotational range of -90° to +90°.

The "Angle (360)" feature depends on the selected geometric model (e.g. E1 Ellipse, R2 Rectangle minimal area, etc.). It has a rotational range of -180° to +180°, as shown in the following figure.

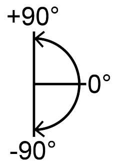


Fig. 165: Rotational direction of "Angle (180)"

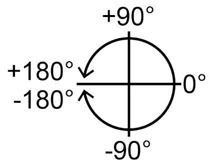


Fig. 166: Rotational direction of "Angle (360)"

In contrast to the "Angle (180)" feature, in the "Angle (360)" feature, the orientation of the width axis is set depending on a vector. This vector indicates the direction to the contour point with the longest distance to the center and the orientation (180°) of the vector point to the same side. Whether an object lies in half plane (-90° ... +90°) or in (-180° ... -90°; 90° ... 180°) is determined by the half plane in which the vector lies. The following figures show two examples of the angle determination of the feature "Angle (360)".

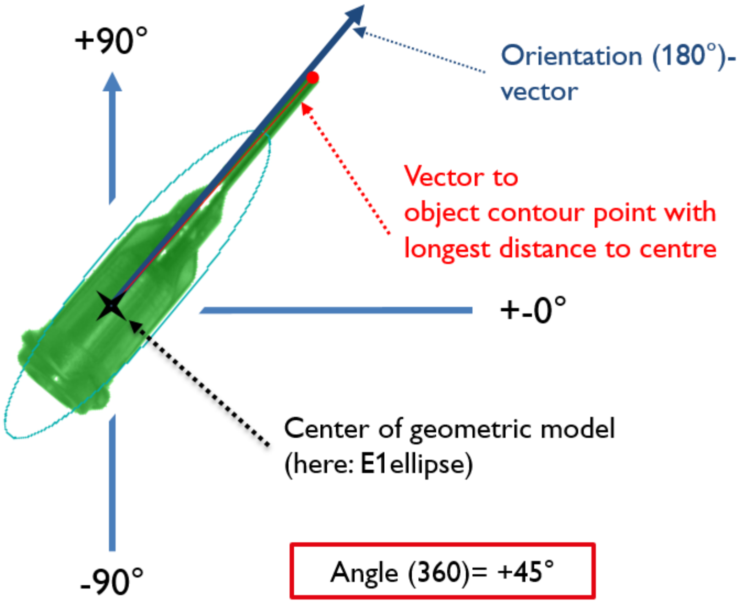

+90°
Orientation (180°)-vector
Vector to object contour point with longest distance to centre
+-0°
Center of geometric model (here: E1ellipse)
-90°
Angle (360)= +45°

Fig. 167: *Example 1: Angle (360) with +45°*

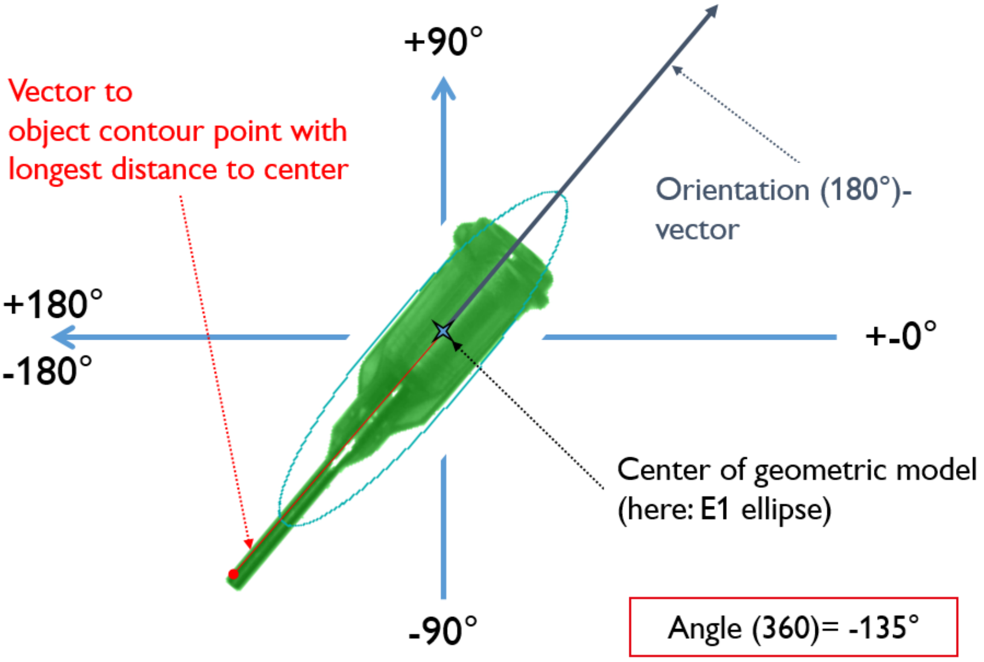
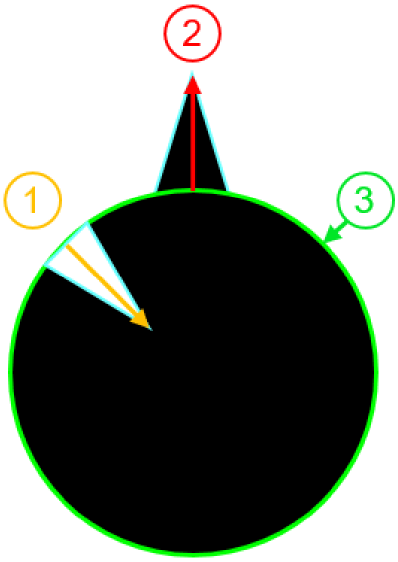

+90°
Vector to object contour point with longest distance to center
Orientation (180°)-vector
+180°
-180°
+-0°
Center of geometric model (here: E1 ellipse)
-90°
Angle (360)= -135°

Fig. 168: *Example 2: Angle (360) with -135°*

Once the system is calibrated, the Angle Z is first determined according to the criteria described above. Then it is transformed according to the robot calibration and the result offset. In this case, the value range increases accordingly to [-180° ... 180°].

Feature Deviation

The deviation feature calculates measures that quantitatively describe the deviation of the actual object from the fitted model. The features "Deviation, inside", "Deviation, outside", and "Deviation, mean" evaluate indentations and outstanding elements of the BLOB / object contour. The deviations always refer to the fitted circle. All indentations inside the fitted circle are "Deviation, inside". All elements that protrude from the fitted circle are judged by the feature "deviation, outside". The orientation directions of the features are shown graphically in the following figure.



(1): Deviation, inside
(2): Deviation, outside
(3): Fitted circle C1

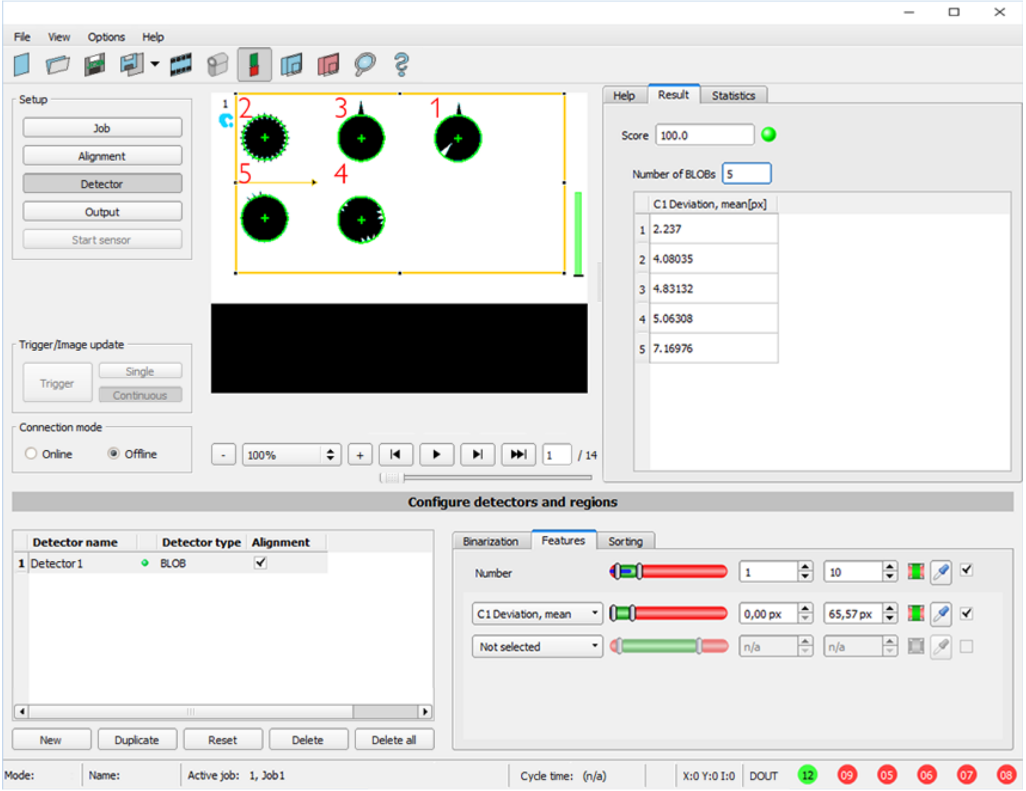
Fig. 169: Orientation direction "Deviation, inside" and "Deviation, outside"

In the "Result" tab of the VISOR® software, the value of the largest deviation towards the inside and the value of the largest deviation towards the outside (provided they are "enabled") are always shown for each fitted circle.

The characteristic "deviation, mean" gives the average of the amounts of the deviations to all positions, i.e. to all pixels of the fitted circle.

Example for the assessment of the mean deviation

Jagged elements are examined for the feature "Deviation, mean", see figure "Deviation, mean".



Score 100.0

Number of BLOBs 5

	C1 Deviation, mean[px]
1	2.237
2	4.08035
3	4.83132
4	5.06308
5	7.16976

	Detector name	Detector type	Alignment
1	Detector1	BLOB	☑

Number	1	10	☑
C1 Deviation, mean	0,00 px	65,57 px	☑
Not selected	n/a	n/a	☐

Mode: Name: Active job: 1, Job1 Cycle time: (n/a) X:0 Y:0 I:0 DOUT 12 09 05 06 07 08

Fig. 170: Deviation, mean

The feature "Deviation, mean" calculates all deviations from the fitted circle (green) to the contour of the object / BLOB (cyan) per pixel of the fitted circle inwards and outwards. The following figure shows a zoomed out section of the circle number "2" from the previous figure. The red arrows indicate the deviations per pixel of the fitted circle to the BLOB contour. The amounts of all determined values are averaged and form the result of the "Deviation, mean" feature.

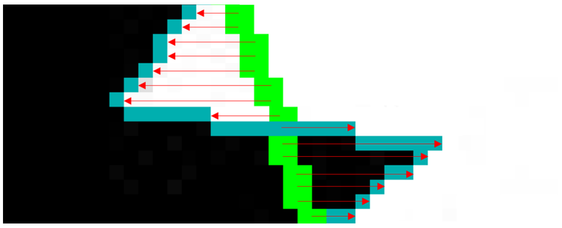


Fig. 171: Deviation, mean: Circle 2 zoomed

Example for assessing deviations from circles inside and outside

Six circles with different notches and protruding elements must be examined for the features "Deviation, inside" and "Deviation, outside".

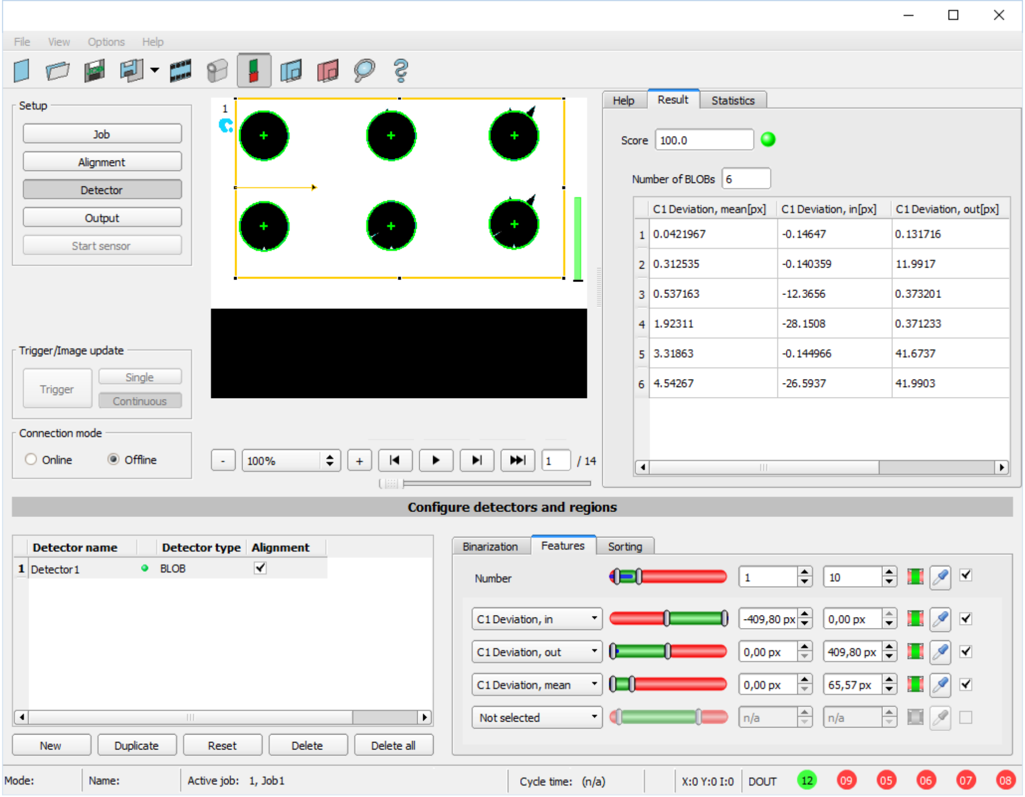
In order to improve the presentation, "BLOB contour" is selected in the "Binarization" tab of the "BLOB" detector. Now the detector marks the contours of all circles in the search field in cyan.

The following features are selected in the "Features" tab:

- "C1 Circle, fit" (first-level feature), "Deviation, inside" (second-level feature)
- "C1 Circle, fit" (first-level feature), "Deviation, outside" (second-level feature)

- "C1 Circle, fit" (first-level feature), "Deviation, mean" (second-level feature)

In the "Result" tab, the results of the characteristics per circle can now be read, see also the following figure. (Note: The results can be assigned to the circles by moving the mouse over the circles in the field of view.)



Score 100.0

Number of BLOBs 6

	C1 Deviation, mean[px]	C1 Deviation, in[px]	C1 Deviation, out[px]
1	0.0421967	-0.14647	0.131716
2	0.312535	-0.140359	11.9917
3	0.537163	-12.3656	0.373201
4	1.92311	-28.1508	0.371233
5	3.31863	-0.144966	41.6737
6	4.54267	-26.5937	41.9903

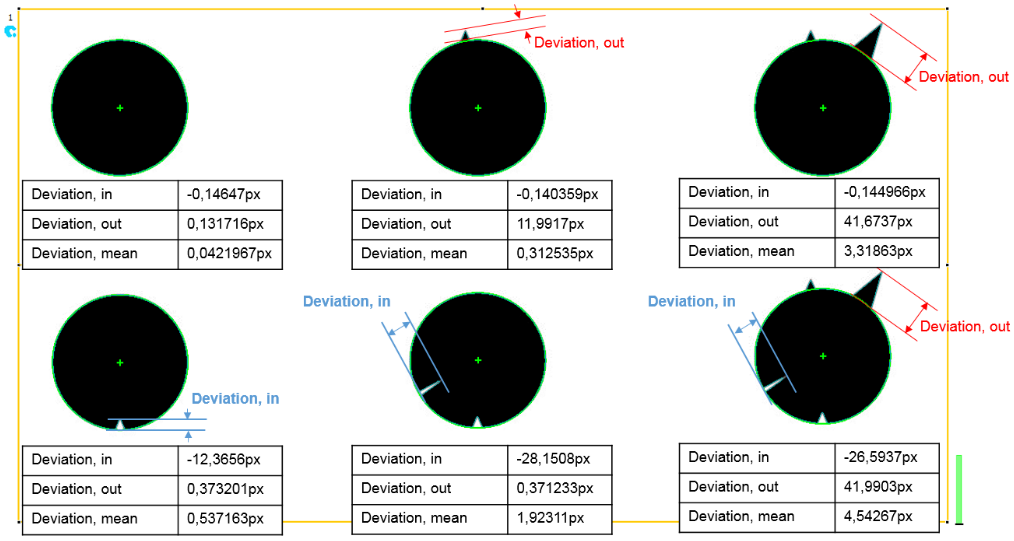
	Detector name	Detector type	Alignment
1	Detector1	BLOB	☑

Number	1	10	☑
C1 Deviation, in	-409,80 px	0,00 px	☑
C1 Deviation, out	0,00 px	409,80 px	☑
C1 Deviation, mean	0,00 px	65,57 px	☑
Not selected	n/a	n/a	□

Mode: Name: Active job: 1, Job1 Cycle time: (n/a) X:0 Y:0 I:0 DOUT 12 09 05 06 07 08

Fig. 172: "Deviation, inside", "Deviation, outside", and "Deviation, mean" results

The figure below serves to allocate and interpret the results from the screenshot above.



Deviation, in	-0,14647px
Deviation, out	0,131716px
Deviation, mean	0,0421967px

Deviation, in	-0,140359px
Deviation, out	11,9917px
Deviation, mean	0,312535px

Deviation, in	-0,144966px
Deviation, out	41,6737px
Deviation, mean	3,31863px

Deviation, in	-12,3656px
Deviation, out	0,373201px
Deviation, mean	0,537163px

Deviation, in	-28,1508px
Deviation, out	0,371233px
Deviation, mean	1,92311px

Deviation, in	-26,5937px
Deviation, out	41,9903px
Deviation, mean	4,54267px

Fig. 173: Result interpretation of "Deviation, inside", "Deviation, outside", and "Deviation, mean"

Feature Face up / down

"Face up / down, area" assesses the symmetry of a blob with respect to an axis determined by the center and the orientation of the blob. If a BLOB is fully symmetric with respect to this line, the result value will be 0, otherwise it will deviate from 0. The sign of the value indicates whether the side to the left or right is "stronger".

"Face up / down, area" can be used to distinguish the face up / face down position of an object, as is necessary in many areas of feeding technology. Applications can be found, for example, on vibratory conveyors or in robotics.

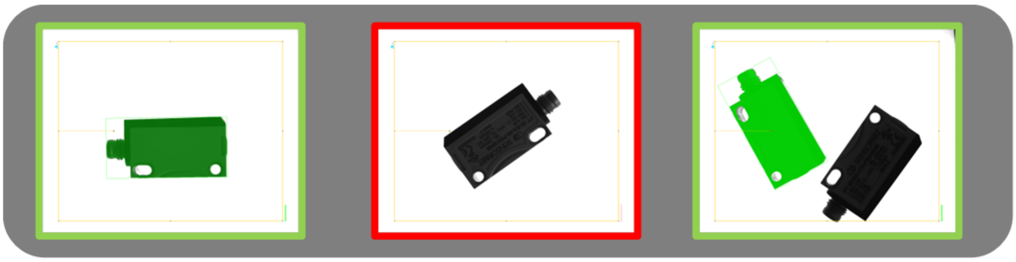


Fig. 174: *Face up / down, area*

The left image shows the searched object e.g. lying on the ventral side. The thresholds are chosen so that this upward-facing position leads to a positive result.
The image in the middle shows the same object face up and is considered "not OK".
The right-hand image displays both objects in an image where only the object facing up is considered "ok".

"Face up / down, area" uses each pixel associated with the BLOB for the calculation.

The axis used for the calculation is determined by the center and angle (360 °) of the geometric model chosen that was chosen, e.g. smallest enclosing rectangle (R2) or ellipse (E1).

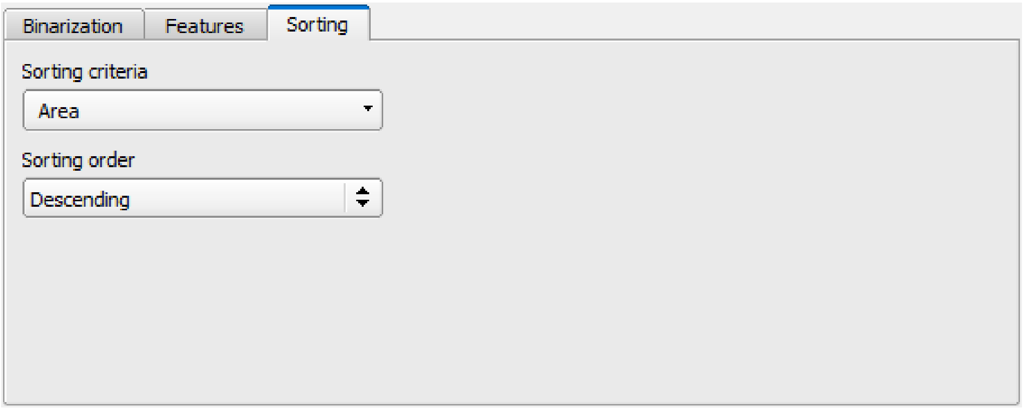
The geometric model for the calculation should be chosen so that the orientation returns a stable and unambiguous value. Thus highly symmetric objects (e.g. rectangles, circles, squares, or point-symmetric objects) cannot be reliably evaluated with this method. For objects where the smallest enclosing rectangle does not provide a unique orientation indication (e.g., "L"-shaped geometries), the ellipse may be a better choice than a geometric model.

8.3.10.4 Sorting tab

The characteristics (Features tab) defined in the "Features" tab are calculated for each blob. The results from these calculations can be sent to a PC or PLC in a data telegram, provided that the property has been selected in the corresponding Telegram tab. The order in which the results of the individual BLOBs are sent is specified in the "Sorting" tab.

If e.g. the feature "Center of gravity Y" is calculated and there are 5 BLOBs in the image, the output data telegram comprises the results of all 5 BLOBs.

If the sort criterion is "Area" and the sort order is "Descending", the result (here: center of gravity Y) of the BLOB with the largest area is output first.



Sorting criteria: Area

Sorting order: Descending

Fig. 175: Detector BLOB, tab Sorting

Parameter description:

Parameter	Function
Sorting criterion	Each feature mentioned under "Detector BLOB / Features tab" can be used as a sorting criterion.
Sorting order	Sorting order for the selected sorting criterion. Sorting can be "ascending" or "descending".

8.3.10.5 Result offset tab

See Chapter: Result offset tab

8.3.11 Detector Caliper

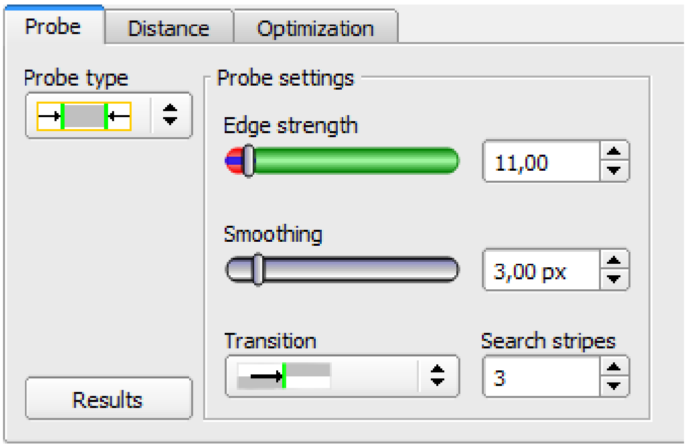
This detector can be used to check objects for dimensional accuracy by measuring the distance between edges. Different types of probes are available for this purpose. Minimum, maximum or averaged distance values can be measured.

8.3.11.1 Color Channel tab

See Chapter: Color Channel tab

8.3.11.2 Probe tab

All caliper parameters can be set here, and the result Histogram can be retrieved.



Probe type

Probe settings

Edge strength 11,00

Smoothing 3,00 px

Transition

Search stripes 3

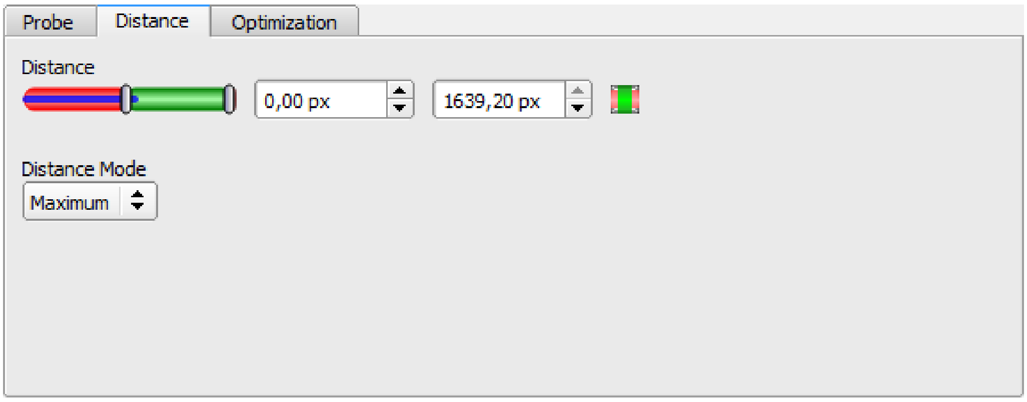
Fig. 176: *Detector Caliper, tab Probe*

Parameter description:

Parameter	Function
Probe type	Selection of probe type: • One probe, both sides • One probe, one side • Two probes, antiparallel (opposite direction) • Two probes, same direction
Edge thickness	Edge width / contrast from which (0 ... 100) an edge should be detected as an edge.
Smoothing	The edge contour is smoothed in the search direction. With larger values, noisy edges, blurred edges, or edges that are not perpendicular to the search direction, are detected more reliably. In addition, light-dark-light or dark-light-dark transitions which are close together with larger values can be ignored. Thus, interfering edges, e.g. scratches, can be hidden. The effect of smoothing can be displayed graphically using the button "Results".
Transition	Selection between: • Light → dark • Dark → light • Both directions (light-dark and dark-light transition)
Number of search stripes	Number of parallel search stripes into which the width of the region of interest is divided. Edge detector is processed in each search stripe over the whole width. The bigger the number of search stripes, the more probable the very first edge will be found. (Finer sampling will result in a longer execution time)
Results	Opens result and histogram window

8.3.11.3 Distance tab

Here, all parameters for the desired distance can be set.



Distance: 0,00 px – 1639,20 px

Distance Mode: Maximum

Fig. 177: *Detector Caliper, tab Distance*

Parameter description:

Parameter	Function
Distance	Distance range in pixels. Blue bar: Current distance value.
Distance mode	For each search stripe, one probe point is determined. If the number of search stripes is greater than 1, there are several possibilities to determine a result from these probe points: • Maximum: The probe point that gives the maximum distance is selected. • Minimum: The probe point that gives the minimum distance is selected. • Mean: All probe points are arithmetically averaged. If there are outliers, they go into the result and falsify it. • Median: The probe points are sorted in ascending order and the mean distance is used. Outliers do not influence the result. Only available for the double-sided probe "→▭←" • Maximum by search stripe (of a search stripe): The probe points of the search stripe with the greatest distance are selected. • Minimum by search stripe (of a search stripe): The probe points of the search stripe with the smallest distance are selected. • Minimum/Maximum by search stripe: combination of "Maximum by search stripe" and "Minimum by search stripe" ℹ **NOTE:** For "Minimum/Maximum by search stripe", also those search stripes in which no edges are found are taken into account. These are output with the result 0.

Additional information:

Distance mode

Information about the structure of the Edge detector can be found at: Structure of the Edge detector

To determine a distance, different distance modes can be selected in the "Distance" tab. The following examples explain the difference between "Minimum" and "Minimum by search stripe" and the difference between "Maximum" and "Maximum by search stripe".

The following object is detected:

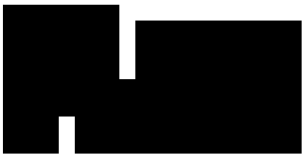


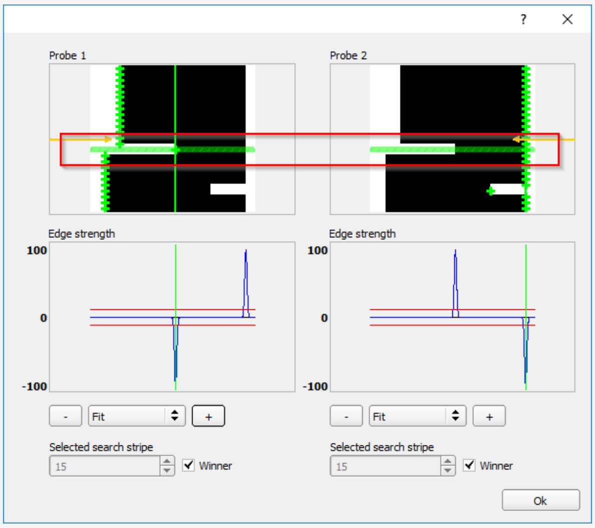
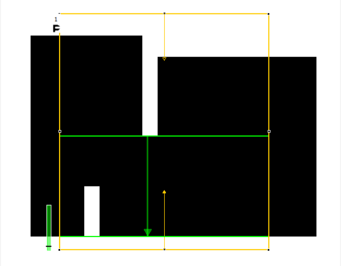
__Fig. 178:__ Demonstration object

For the examination the double-sided probe type "→▢←" is selected in the Probe tab and the number of search stripes is increased to 20.

Difference between "Minimum" and "Minimum by search stripe"

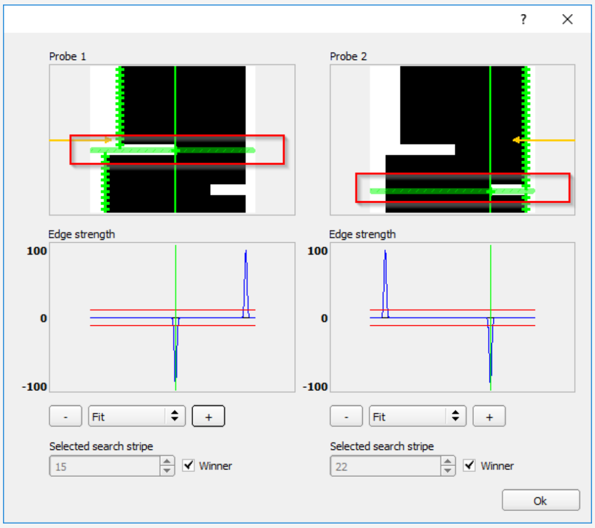
With the Distance mode "Minimum by search stripe", in contrast to the Distance mode "Minimum", the probe points are evaluated by only one search stripe. The search stripe with the smallest distance between the probe points is selected (see figure: Results histogram, Distance mode "Minimum by search stripe").

With the Distance mode "Minimum" the smallest distance is determined in which the probe lines with the smallest distance to each other are selected. However, in contrast to the Distance mode "Minimum by search stripe", the probe points of the probe lines can come from two different search stripes (see figure: Results histogram, Distance mode "Minimum").

Distance mode "Minimum by search stripe"	
Minimum by search stripe determined at the probe points of a search stripe.	
 ? × Probe 1 Probe 2 Edge strength 100 0 -100 Edge strength 100 0 -100 - Fit + - Fit + Ok Selected search stripe 15 ☑ Winner Selected search stripe 15 ☑ Winner ***Fig. 179:** Results histogram, Distance mode "Minimum by search stripe"*	 ***Fig. 180:** Image output with overlay, Distance mode "Minimum by search stripe"*

Distance mode "Minimum"

Minimum distance determined at the scanning lines with the smallest distance to each other. The probe points of the scanning lines can, but do not have to, come from just one search stripe.



Selected search stripe 15 ☑ Winner

Selected search stripe 22 ☑ Winner

Fig. 181: *Results histogram, Distance mode "Minimum"*

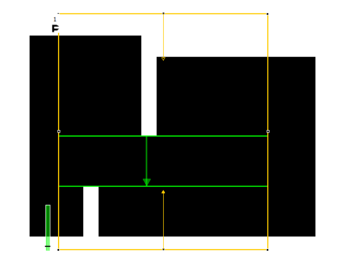


Fig. 182: *Image output with overlay, Distance mode "Minimum"*

Difference between "Maximum" and "Maximum by search stripe"

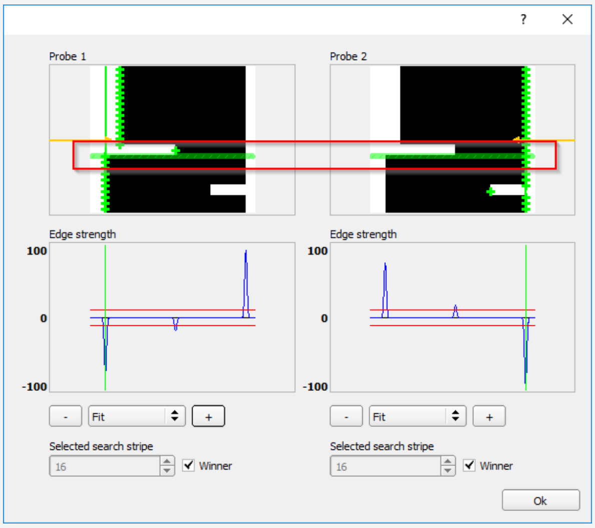
The selected example object shows identical image outputs with the Distance mode "Maximum" and the Distance mode "Maximum by search stripe" when determining the distance (see figures in the right columns ofi the tables below). However, the distances are not determined identically.

With the Distance mode "Maximum by search stripe" the probe points are evaluated by only one search stripe. The search stripe with the largest distance between the probe points is selected (see figure: results histogram, Distance mode "Maximum by search stripe").

With Distance mode "Maximum", the maximum distance is determined by selecting the probe lines with the greatest distance to each other. However, the probe points of the probe lines can come from two different search stripes (see figure: Results histogram, Distance mode "Maximum").

Distance mode "Maximum by search stripe"

Maximum by search stripe determined at the probe points of a search stripe.



Selected search stripe 16 ☑ Winner

Selected search stripe 16 ☑ Winner

Fig. 183: Results histogram, Distance mode "Maximum by search stripe"

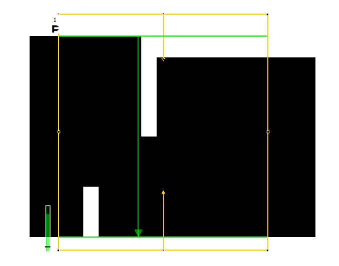
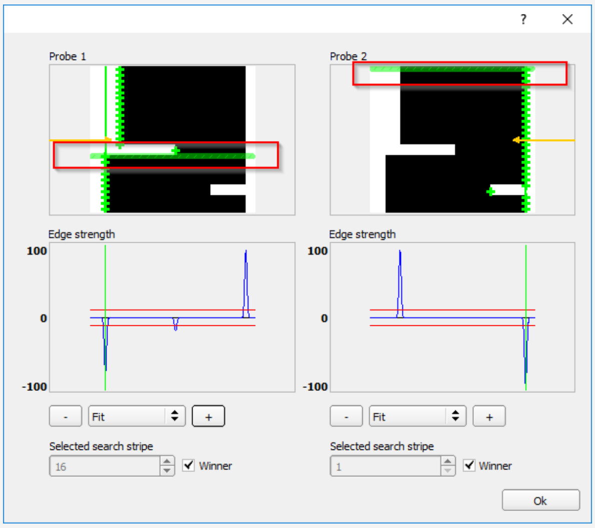


Fig. 184: Image output with overlay, Distance mode "Maximum by search stripe"

Distance mode "Maximum"

Maximum distance determined at the scanning lines with the largest distance to each other. The probe points of the scanning lines can, but do not have to, come from just one search stripe.



Selected search stripe 16 ☑ Winner

Selected search stripe 1 ☑ Winner

Fig. 185: Results histogram, Distance mode "Maximum"

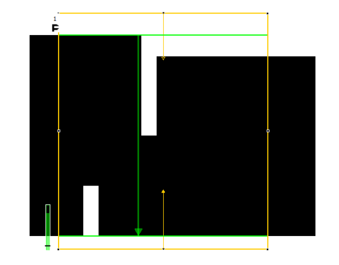


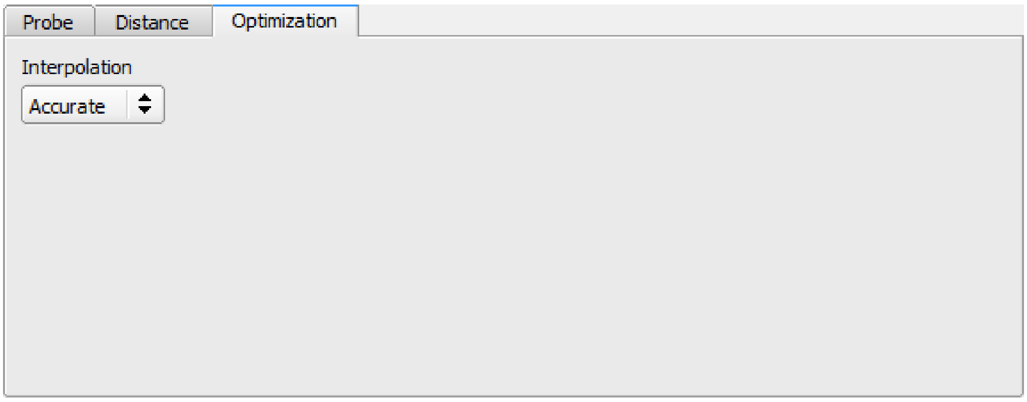
Fig. 186: Image output with overlay, Distance mode "Maximum"

Output of probe positions

NOTE:
For the result output, probe P1 is the starting point of the green arrow in the region of interest, probe P2 is the end point of the green arrow. In case of only one probe, P1 is therefore at the edge of the yellow region of interest.

8.3.11.4 Optimization tab

In the "Optimization" tab, further settings for optimizing the Edge detector can be made.



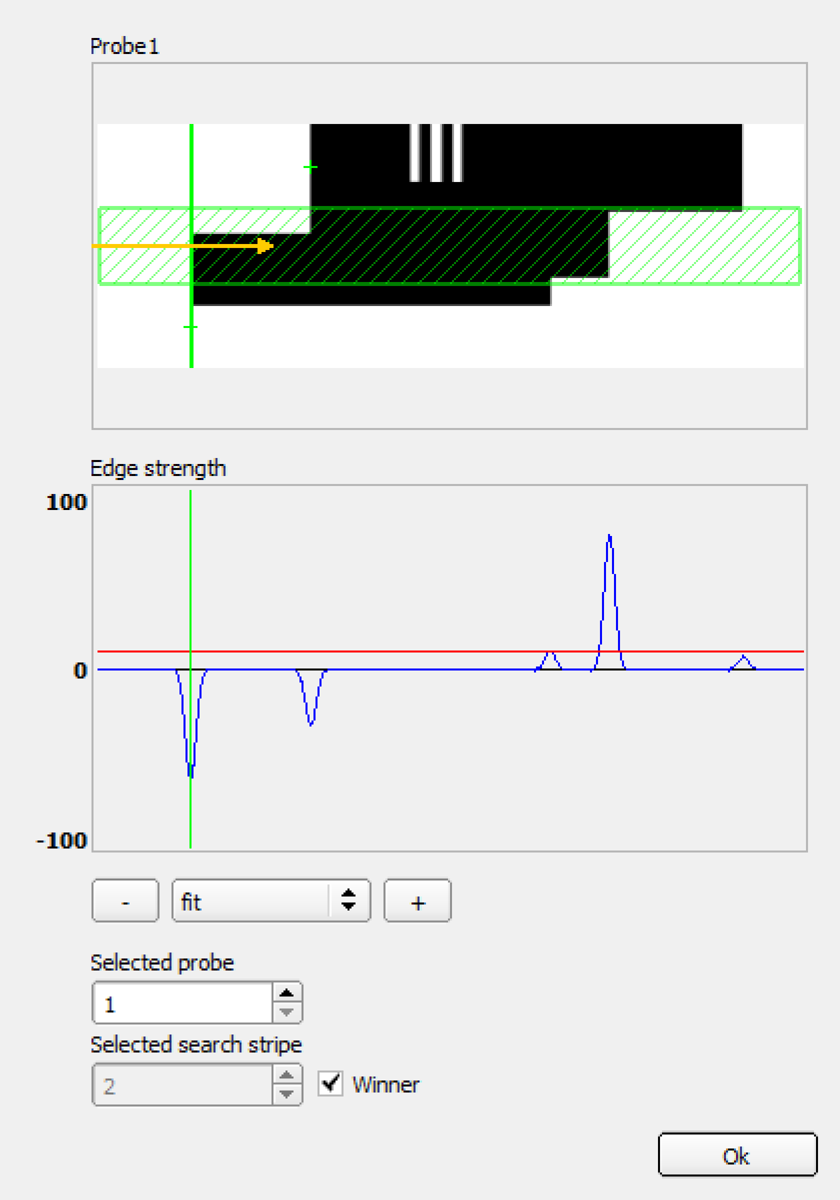
Interpolation: Accurate

Fig. 187: *Detector Caliper, tab Optimization*

Parameter description:

Parameter	Function
Interpolation	The calculation of the edge position can be performed either with subpixel accuracy (up to 1/10 pixels) or with simple accuracy.
Accurate	Subpixel accuracy
Fast	Simple accuracy: This setting partly provides calculations which are over 50% faster.

8.3.11.5 Results window



Selected probe: 1

Selected search stripe: 2

☑ Winner

Fig. 188: Results window / Histogram

Parameter	Function
Probe (x)	Image of probe (x) with: • Green line: Detected result edge • Green crosses: Determined edge transition (touch point) per search stripe • Light blue or light green zone: Display of "Selected search stripe"
Edge thickness	Histogram with: • Blue line: Contrast gradient in the image, depending on the selected search stripe • Red line: Required contrast for edge ("Switching threshold") • Light blue or light green line: Determined edge transition depending on selected search stripe

Parameter	Function
Fit, "+", "-"	Fit or zoom for "edge width" histogram
Selected probe	Display of the selected probe
Selected search stripe	Selection of the search stripe to be displayed in the probe image • Winner: Winner search stripe (depending on settings in tab "Distance") • "1, 2, .." Number of the displayed search stripe

8.3.12 Detector Classification (AI)

The detector Classification (AI) assigns the object in the region of interest to a previously defined class. You can make simple good / bad differentiations or differentiate between up to 200 classes. The Classifier is trained on the basis of sample images representing the individual classes and then returns the assigned class and a score (degree of matching) as the result.

Examples of typical applications

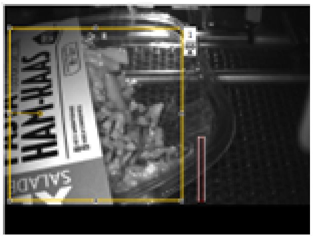
With this Detector you can ensure that the correct product variant is manufactured correctly.

In a production line, the same type of car could be manufactured in different variants, for example, whether the correct switch is installed on the dashboard.

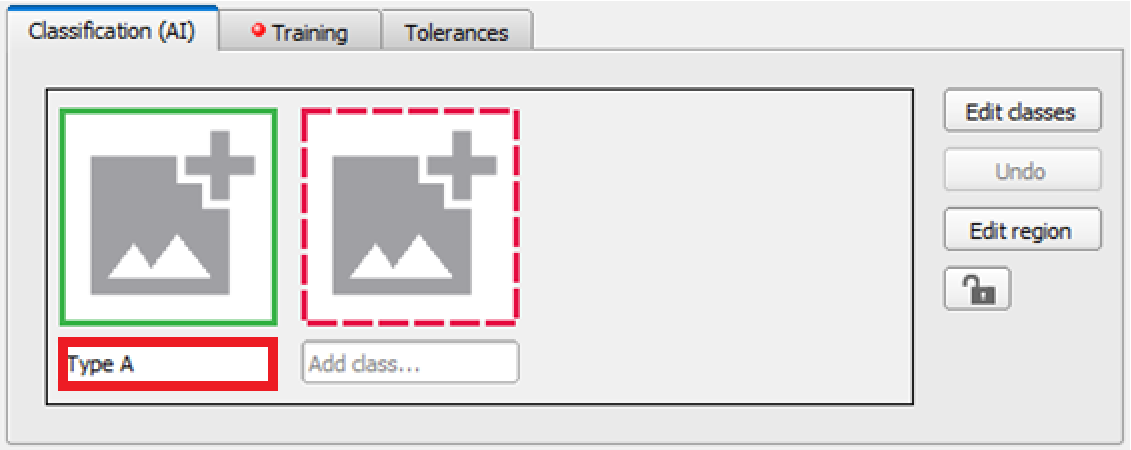
You can also use the Detector for quality inspection, to distinguish between good and bad components. Example: If there is a pin in a component, check whether the pin has been correctly inserted into the component or whether it is missing altogether. To do this, train the Detector with training images of all states that are possible here: pin correctly inserted, pin incorrectly inserted, pin missing.

The Detector can be taught as follows:

1. Define the region of interest (ROI - yellow frame).



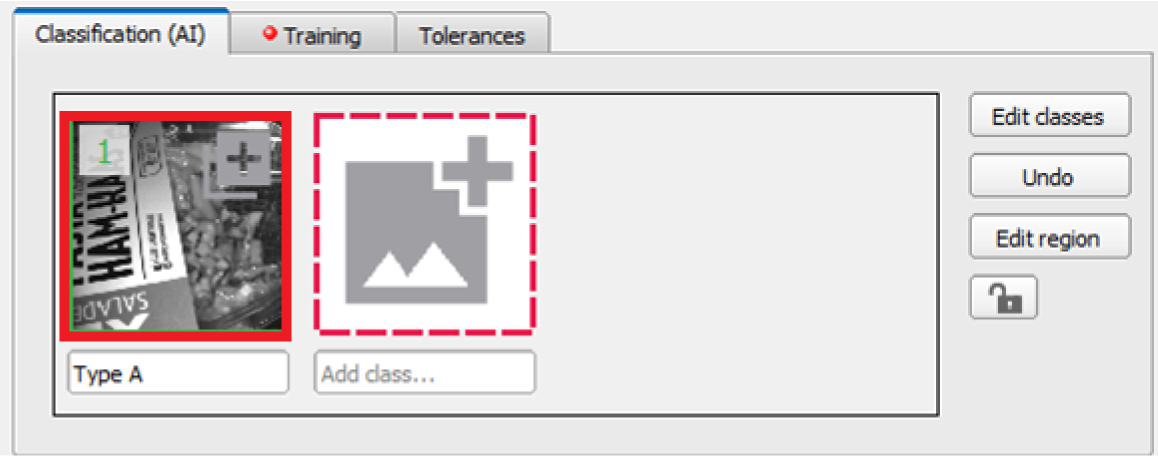
2. In the "Add class" field, enter the name of the class. You can freely choose the name.



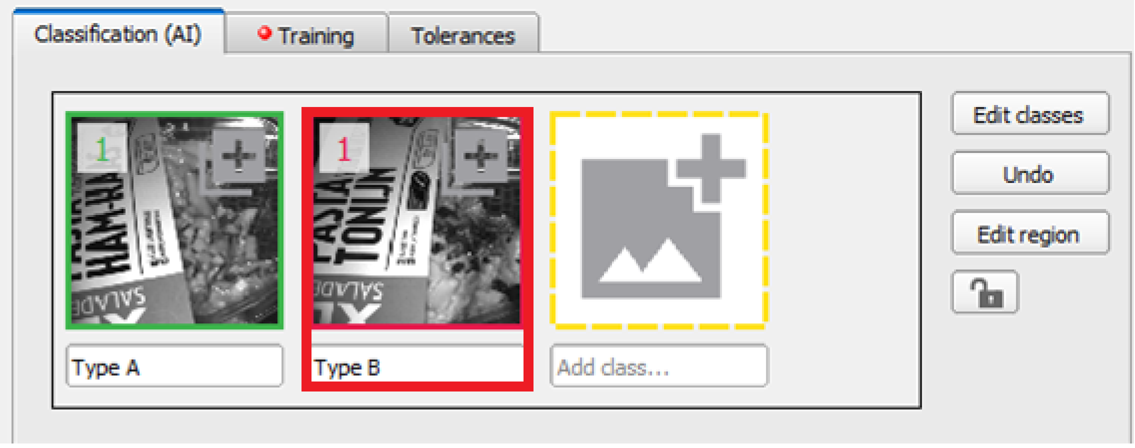
Type A

Add class...

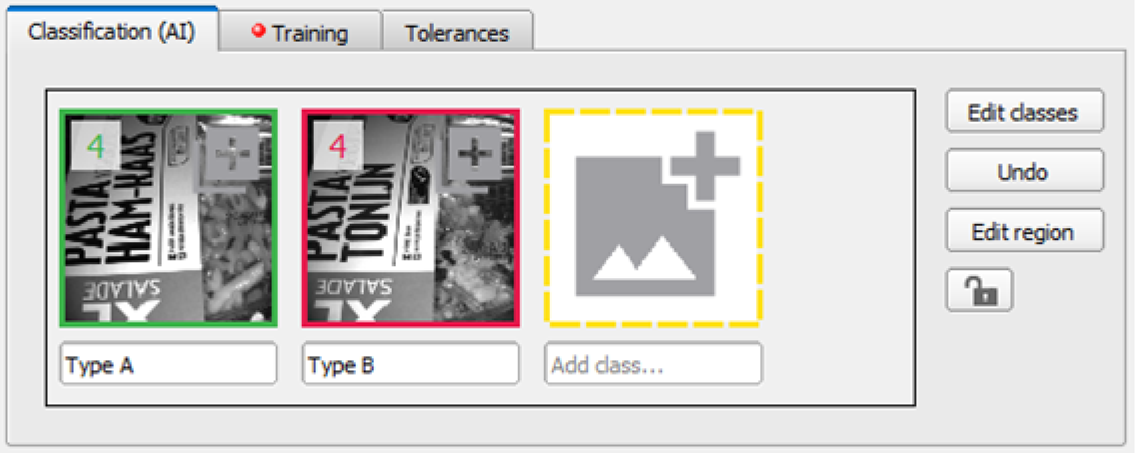
3. Click on the "Add image" icon. This assigns the current contents of the region of interest to the corresponding class.



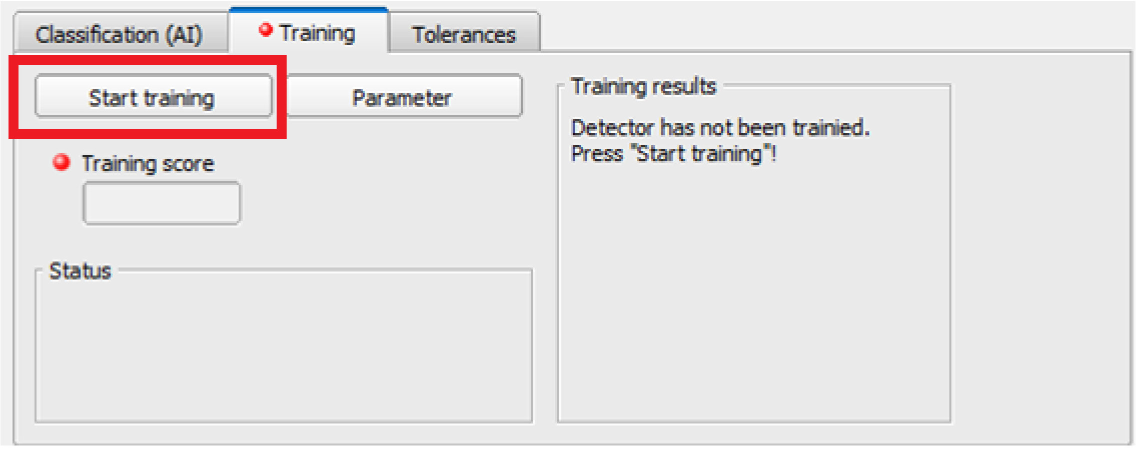
4. Repeat this process for at least one more class.



5. Assign as many additional images to the classes as you like by clicking on the class icon. When doing so, keep in mind:
   - The more training images you add, the higher the detection performance of the detector.
   - Add approximately the same number of training images to each class.

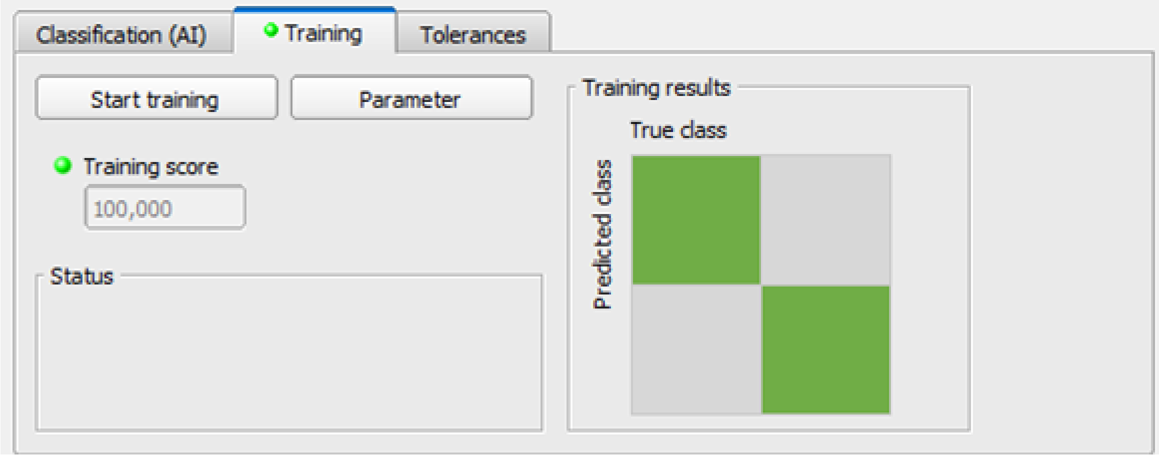


6. Go to the Training tab and click on "Start training".



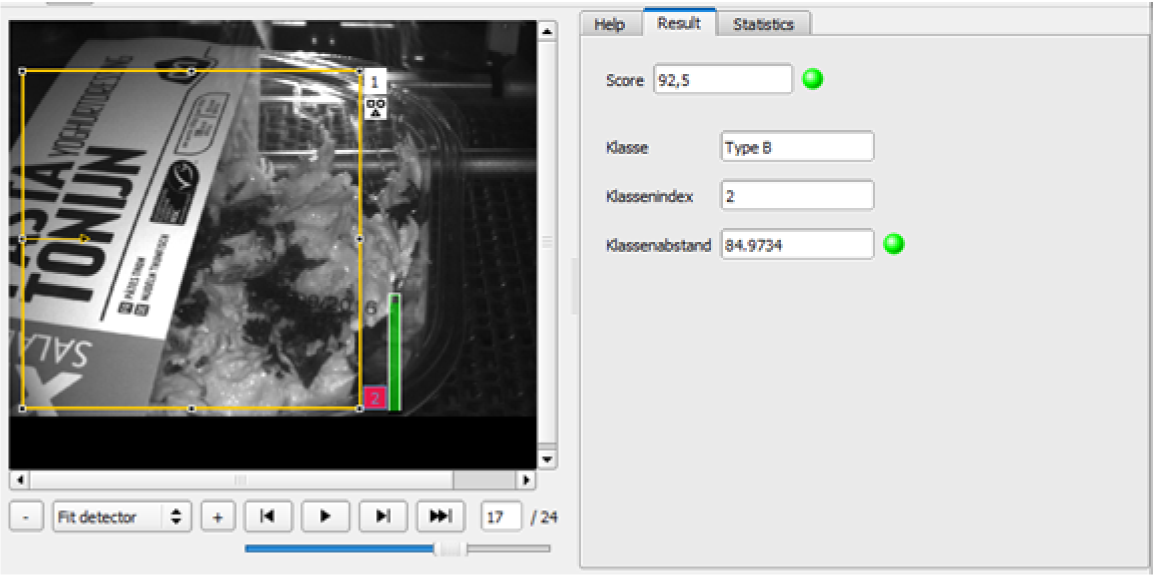
Detector has not been trainied.
Press "Start training"!

7. After completion of the training, the Detector is ready for use.



● Training score: 100,000

8. Test the Detector by running it. The assigned class is displayed as the result.



Score: 92,5 ●

Klasse: Type B

Klassenindex: 2

Klassenabstand: 84.9734 ●

NOTE:
Under "Options" you can click on "Classification (AI): Open folder with training data records" to open the folder with all training images in Windows Explorer.

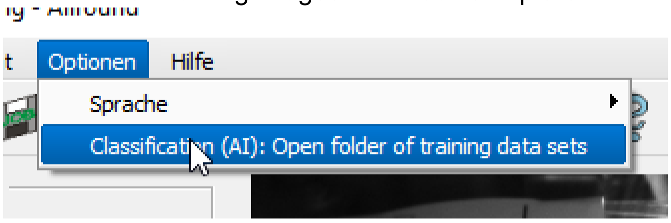


Fig. 189: *Open folder with training records*

8.3.12.1 Notes on use

Setting up detectors that learn is an iterative process.

If you have only a few, clearly distinguishable classes, you can initially assign about five images per class. This will give you a rough estimate of the classification performance after an initial Training. Use images in the filmstrip or live from the system to test whether all classes have been correctly assigned by the Detector. If this is not the case, repeat the process by assigning the incorrectly classified images to the correct class. Then re-train the Detector.

General notes:

- A larger number of training images per class leads to better and more stable classification performance.
- The more classes you want to distinguish, the more training images per class you need.
- Train example images of all the variations that result from the process, such as contamination, so that the Classifier can correctly identify the variations.
- The Detector always assigns a class to each region of interest, even if the content of the region of interest does not correspond to what was trained.
- There are three different types of Classifiers: Siena, Indigo and Olive (see Parameters window).

Notes on defining the region of interest
Select the region of interest so that all relevant features are included and visible with a sufficient size.
If this is not the case, adjust the region of interest:

- If necessary, use the Alignment and reduce the region of interest's size.
- Edit the region of interest if necessary. Hide fluctuations and limit yourself to the essentials in the image.
- Keep the region of interest as small as possible. The essentials should be visible and the structures to be distinguished should be clearly recognizable in the images in the training database (see Managing training images).

Managing training images

Once you add the training images to a class, the images are added to the training image database. This database is stored on the PC. Only the trained Classifier is transferred to the sensor. The training dataset is stored only on the PC running SensoConfig and is not part of the job dataset that is transferred to the sensor.

As soon as another PC without the training dataset connects to the sensor, the Detector can be displayed and executed, but not changed, which results in the following:

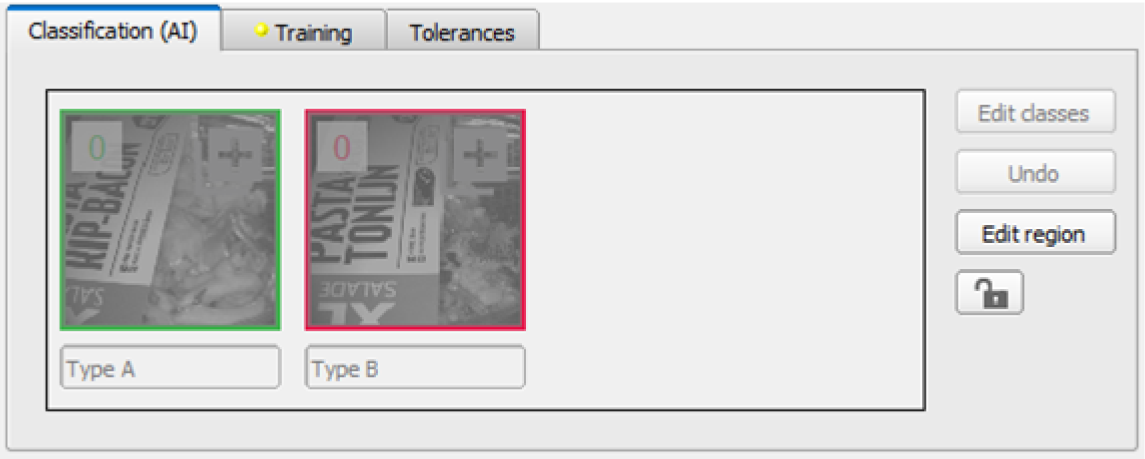
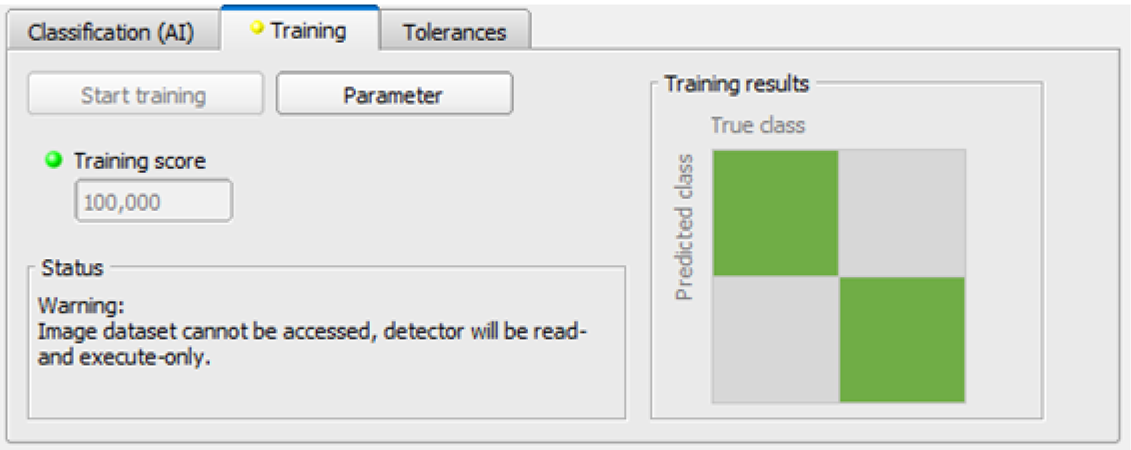


Fig. 190: No access to the image dataset



Status

Warning:
Image dataset cannot be accessed, detector will be read- and execute-only.

Fig. 191: No access to the Training

8.3.12.2 Classification (AI) tab

Use this tab to define the classes and manage the training images.

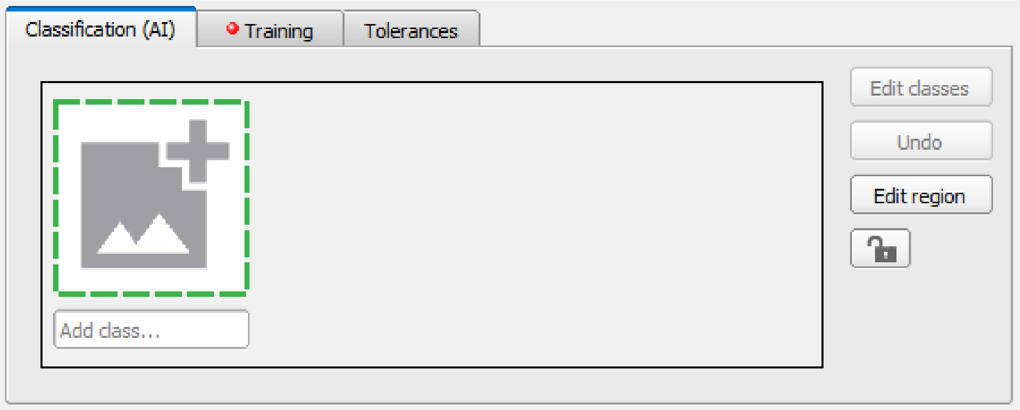


Fig. 192: Detector Classification (AI), Classification (AI) tab - Before assigning classes

Assign the images to the desired class by clicking on the icon.

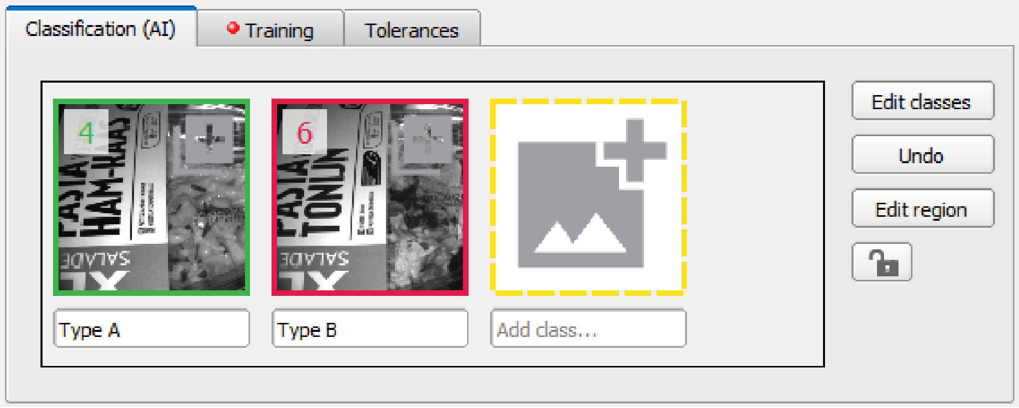
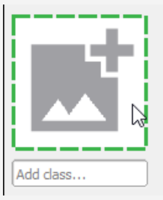
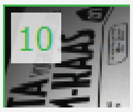
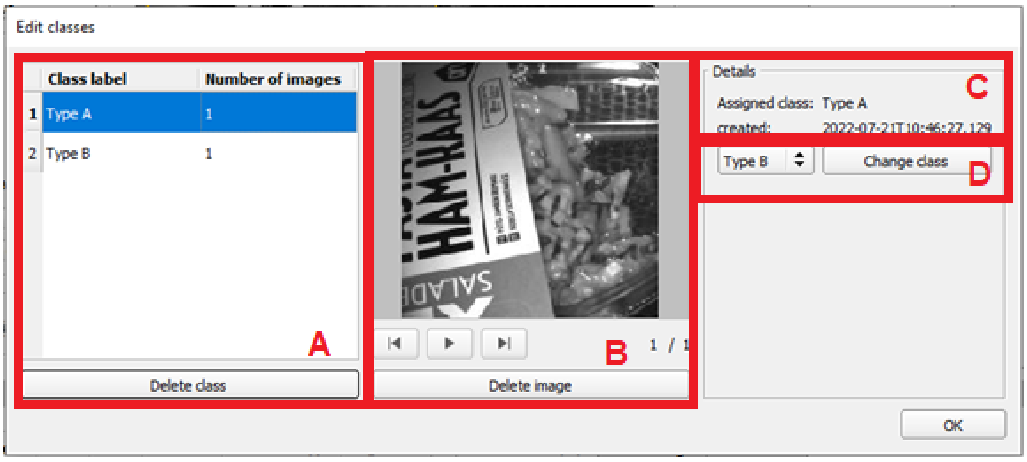


Fig. 193: Detector Classification (AI), Classification (AI) tab - After assigning classes

NOTE:
For efficient assignment of images to classes, use the number keys 1 - 9 on the keyboard. The number keys automatically assign the current image to the corresponding class and then immediately display the next image in the filmstrip. Number key 1 = Class 1, number key 2 = Class 2, etc.

Parameter	Function
Icon: 	Assigns the contents of the region of interest (ROI) to the selected class so that it can be used for Training the detector.
Icon: 	Number of images of this class used for Training.
Text field: Add class ...	Adds a name to the class. The name can be changed as desired with a limit of 50 characters. Another class can only be defined once the field has been filled or an image has been added.
Undo	The last image added to a class is removed from that class.
Edit classes	see Edit classes window.
Edit area	see Function: Edit pattern / contour
Lock 🔓	Locking / unlocking the settings: In the locked state, the taught-in settings are protected against (unintentional) change, e.g. accidentally adjusting the learning area. Unlock (click the lock icon again) for changing the settings.

8.3.12.2.1 Edit classes window



Edit classes

	Class label	Number of images
1	Type A	1
2	Type B	1

Details

Assigned class: Type A

created: 2022-07-21T10:46:27.129

Fig. 194: Edit classes window

A: List of existing classes and number of images assigned to the class.

- "Delete Class": Completely deletes the class selected in the list and the images assigned to it.

B: Displays the images assigned to the class marked in list A.

- ▶| One image forward
- |◀ One image back.
- ▶ Automatically plays the images one after the other.
- "Delete Image": Deletes the currently selected image.

C: Details

- Class to which the currently selected image is assigned.
- Date of recording or modification.

D: Change class

- Change the assignment of the selected image by selecting a different class from the drop-down list and then clicking "Change Class".

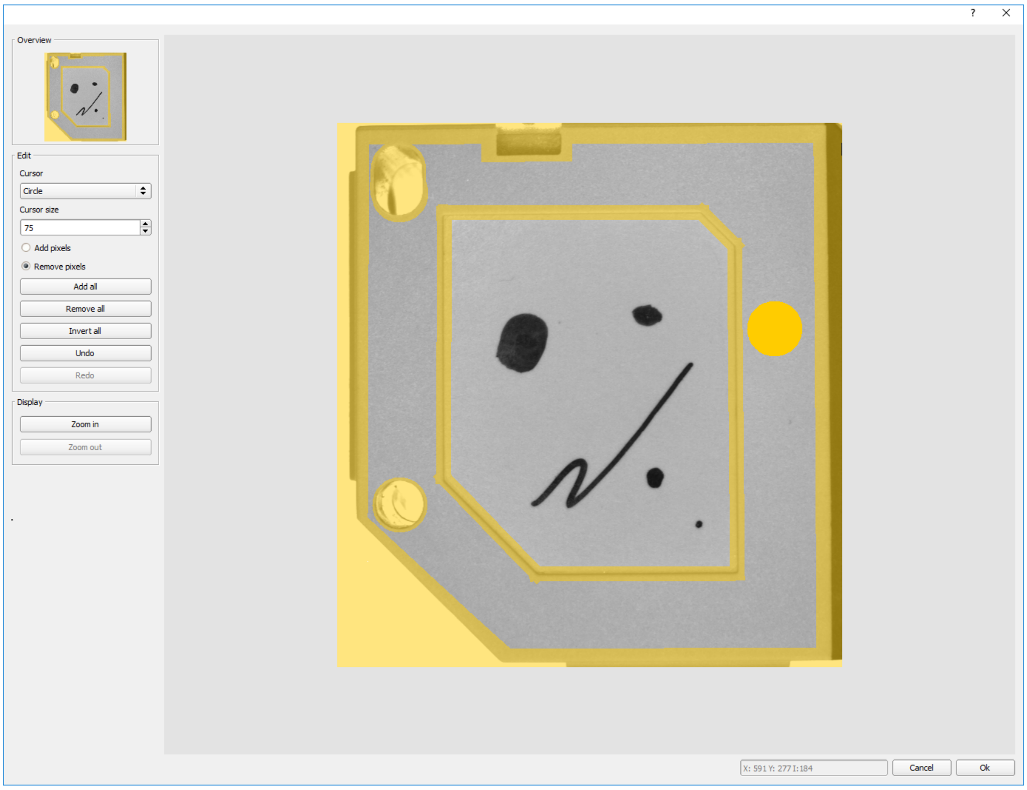
8.3.12.2.2 Function: Edit pattern / contour

With the function "Edit region of interest" or "Edit pattern" / "Edit contour", areas for the evaluation can be allowed or excluded within the search fields / feature fields of the various detectors.

Application example

Outer and inner contour lines, as well as holes should be irrelevant for the evaluation, but all surface defects should be detected.

After masking, only the unmarked areas within the ROI of the detector are used for the evaluation. The yellow marked areas are masked and thus no longer relevant for the evaluation.



Cursor: Circle

Cursor size: 75

○ Add pixels

● Remove pixels

X: 591 Y: 277 I: 184

Fig. 195: Edit region

Handling

Parameter	Function
Cursor (shape)	Changing shape of the cursor (square, circle, or line) With the setting: "Cursor = Line", the angular position of the line jumps while the Shift key is pressed in 15 ° increments.
Cursor size	Changing the size of the cursor (possible setting 1-500, also e.g. with the mouse wheel)
Add / remove pixels	Select whether the cursor adds or excludes pixels for image processing
Add all	Adds all pixels to image processing
Removes all	Removes all pixels from image processing
Invert all	Inverts all pixels
Undo	Undoes the last action
Restore	Restores the last undo action
Display	Selects the display mode (zoom in / out)

The flexible selection of cursor shape and size, as well as whether an action adds or removes pixels, can easily and quickly define complex geometrical or free-form areas that are relevant or not taken into account (yellow).

Settings for using the "Edit region" function for the different detector types

Detector Type	Necessary setting for editing
Pattern matching Contour	Generally possible with "Edit pattern" / "Edit contour"
Contrast, Brightness, Gray, BLOB, Color Value, Color Area, Color List	Select region of interest "Free shape"

Masking of search and learn ranges, Examples

Adjustment of the region of interest to the object for detectors Contrast, Brightness, and Gray level

For the detectors mentioned above, there are three forms for the region of interest: Circle, Rectangle, and Free form. If the work area cannot be adjusted well enough to the object with a circle or rectangle that can be rotated over the control point on the arrow, use the freeform region of interest. With this feature, any geometry can be designed for the region of interest. The cursor for editing the region of interest can be selected as a square or a circle of any size. Below are some examples of freeform regions of interest with a brief description of how they were created in the freeform editor.

Example 1: Circles with relevant areas

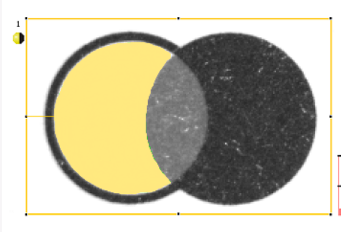


Fig. 196: *Edit region 1*

Generated by an adding and a withdrawing circle.

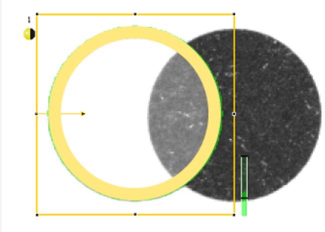


Fig. 197: *Edit region 2*

Generated by an adding and a withdrawing circle.

Example 2: Only surface defects are relevant, object contour lines are masked

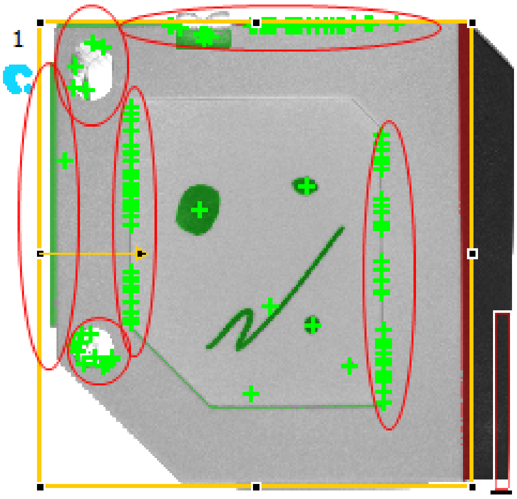
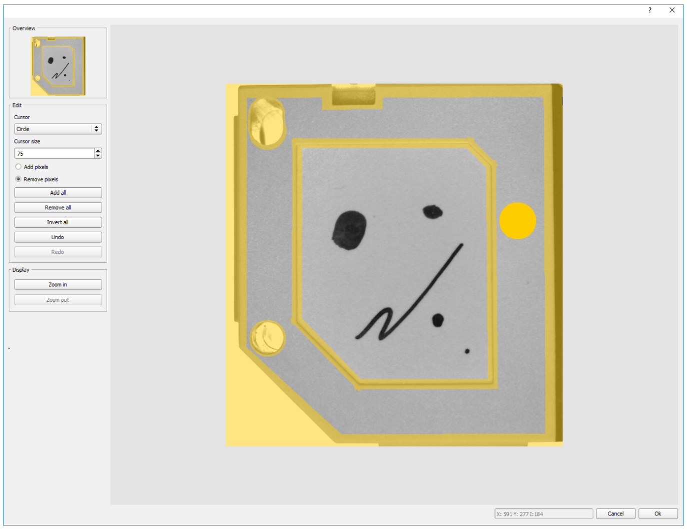


__Fig. 198:__ BLOB detector without the use of masking

With the used BLOB detector, all surface defects as well as the outer and inner contour lines are detected.



Cursor: Circle

Cursor size: 75

○ Add pixels

● Remove pixels

X: 591 Y: 277 I:184

__Fig. 199:__ Masking the contour lines that are not to be recognized = yellow areas.

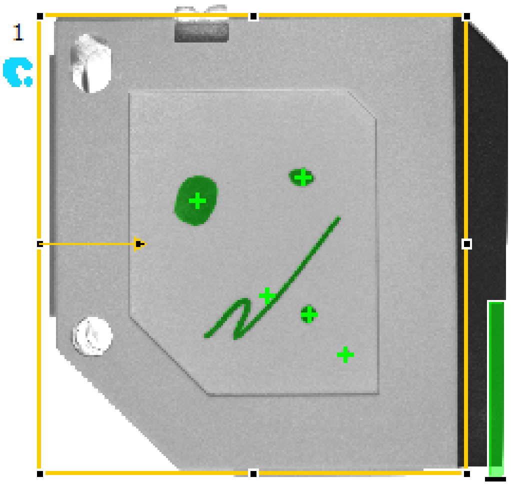


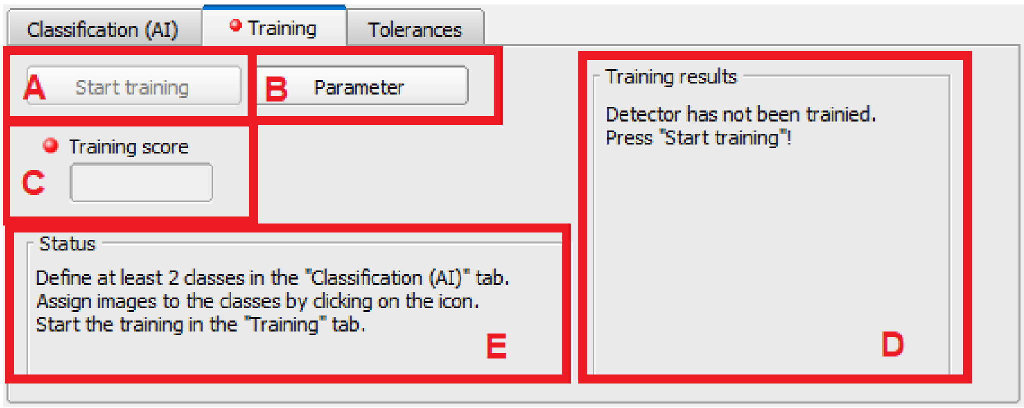
Fig. 200: BLOB detector with the use of masking

Only surface defects are detected; all contour lines / objects in the masking area are no longer recognized.

8.3.12.3 Training tab

Once you have assigned training images to the individual classes, you can start the Training of the classifier with "Start Training" (**A**). When the Training is completed, the Detector can be executed.

The training score (**C**) indicates how well the Classifier performed on the training image dataset during a final test run at the end of the training procedure. Example, 100%: All images were assigned to the correct classes.



Define at least 2 classes in the "Classification (AI)" tab.
Assign images to the classes by clicking on the icon.
Start the training in the "Training" tab.

Detector has not been trainied.
Press "Start training"!

Fig. 201: Detector Classification (AI), Training tab

A: Start of the training.

B: Opens the parameter dialog.

C: Training score.

D: Training results.

E: Detector status.

States of the Classification (AI) detector:

Detector status	Message	Cause	Remedy
🔴	Define at least 2 classes in the "Classification (AI)" tab. Assign images to the classes by clicking on the icon. Start the Training in the "Training" tab.	Detector was newly created.	Classification (AI) tab: 1. Define at least two classes. 2. Add new images to the classes. Training tab: 3. Train the Detector.
🔴	Define at least 2 classes.	Only one class was defined, which means the Training cannot be started.	Define at least two classes.
🟢	- no message -	The Training was successful. The Detector can be executed.	
🟢	Warning: Image dataset cannot be accessed, Detector will be read- and execute-only.	A sensor has been connected that was configured using a different PC. The image database has been deleted.	Use the PC you used for Training. Check the path for the image database.
🟡	Changes have been made. Restart the Training!	Changes to classes and assignment of images to classes.	Changes have been made. Restart the Training.
🟡	Assign images to the classes.	The training score after the Training is below 100%. Not all training images were classified correctly after the test run.	Use more training images, assign incorrectly classified images to the correct class and restart the Training.
🟡	Region of interest has been modified: re-train Detector.	The region of interest has been changed.	Re-train the Detector.
🟡	Mask has been modified: re-train Detector.	The masking of the region of interest has been changed.	Re-train the Detector.
🟡	Parameters have been modified: re-train Detector.	Training parameters have been changed.	Re-train the Detector.
Legend:			

🔴 Detector cannot be executed because no Training has been performed yet.

🟢 The Training was successful. Detector can be executed.

🟡 Detector has been trained and can be executed, but changes have been made since the last Training.

Training result

There are no training results from before the Training, so nothing is displayed in the area. After the Training, all training images are tested with the trained classifier in a final test run.

If the class recognized by the Classifier corresponds to the previously assigned class in all cases, all fields of the diagonal are displayed in green. Incorrectly classified images are displayed in yellow to red, depending on the proportion of the incorrect classification.

In this example, 100% of the images of this class (i.e. row or column) had the correct class detected. The class recognized for all images corresponded to the assigned class.

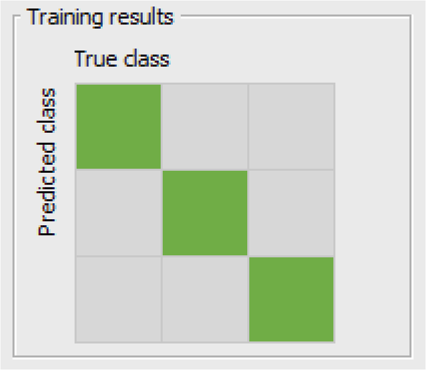


Fig. 202: Training results - 100% detected correctly

If not all images were assigned to the correct class, the proportions of the incorrectly assigned images are displayed in the matrix.

In this example, out of eleven training images assigned to class 1, one was incorrectly identified as class 2. The matrix shows a yellow field in the second row of the first column. If the image had been incorrectly classified as Class 3, the yellow field would be in the third row of the first column. The tooltip shows you what percentage of the images (here: 9.09%) have been classified incorrectly. This is shown with color coding.

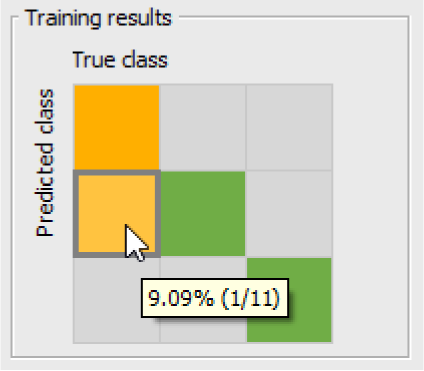


Fig. 203: Training results - 9.09% detected incorrectly

Another Example:

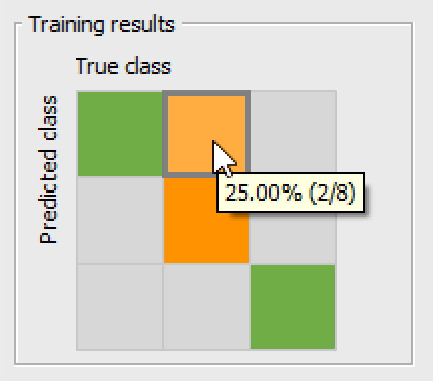


Fig. 204: Training results - 25% detected incorrectly

Here, an image that should actually be assigned to class 2 has been recognized as class 1 by the Classifier. Consequently, the matrix shows an orange field in the first row of the second column. Since 25% of the images here - namely 2 out of 8 - were detected incorrectly, the field is colored darker than in the first example.

Clicking on the field with the left mouse button opens the Training results dialog dialog

Legend:

Columns: Assigned class

Rows: Recognized class

Green: 100% detected.

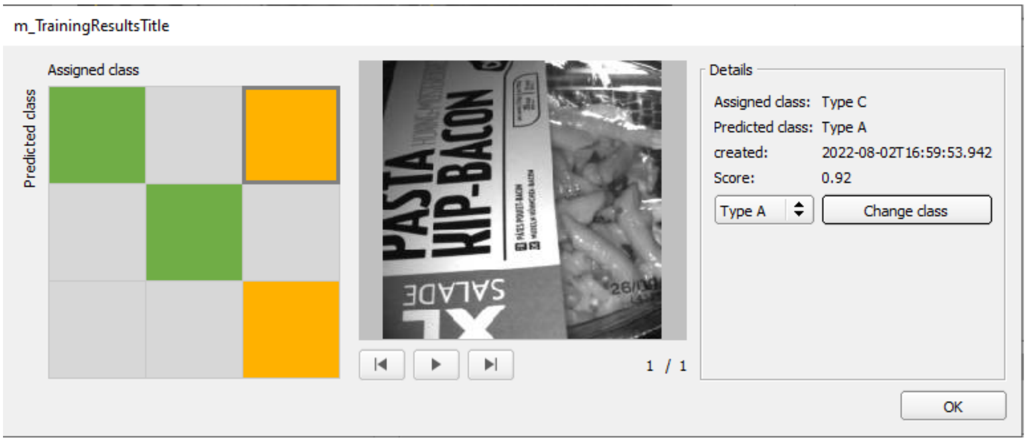
Yellow color gradient to red: The higher the percentage of images classified incorrectly, the more intense the hue of the fields.
100% classified incorrectly = Red.

Gray: 0% classified incorrectly.

Training results dialog

This dialog is used for detailed analysis of the training results. It shows training results in great detail and you can change the assignment to classes. This dialog is only available if a training run has already taken place. Before the training run, you can change the assignment to the classes in the Edit classes window on the Classification (AI) tab . The description of the individual buttons can be found in the chapter Edit classes window

The dialog opens when you click with the left mouse button on a field in the "Training results" window (see Advanced Detector Classification (AI), Training tabarea D).


m_TrainingResultsTitle

Details

Assigned class: Type C
Predicted class: Type A
created: 2022-08-02T16:59:53.942
Score: 0.92

Type A

Fig. 205: Training Results dialog

The matrix of the Training results is shown on the left side. Clicking on a filled field displays the affected images, which you can quickly assign to the correct class using "Change class". Then re-train the Classifier.

A: Matrix of the Training results, see Description above. Select a field to display the correctly or incorrectly classified images.

B: Displays the images assigned to the field in area A.

- ⏭ One image forward
- ⏮ One image back.
- ▶ Automatically plays the images one after the other.

C: Details

- Assigned class: Class to which the currently selected image is assigned.
- Detected class: Class that was detected by the Classifier during the last training run.
- Created: Date and time when the image was assigned to the class.
- Score: The score value achieved by the Classifier for this image during the last training run.

D: Change class

- Change the assignment of the selected image by selecting a different class from the drop-down list and then clicking "Change class".

8.3.12.3.1 Start Training window

The duration of the training time depends on many factors, including hardware performance, number of images in the training dataset, and number of classes.

The progress bar provides a rough overview.

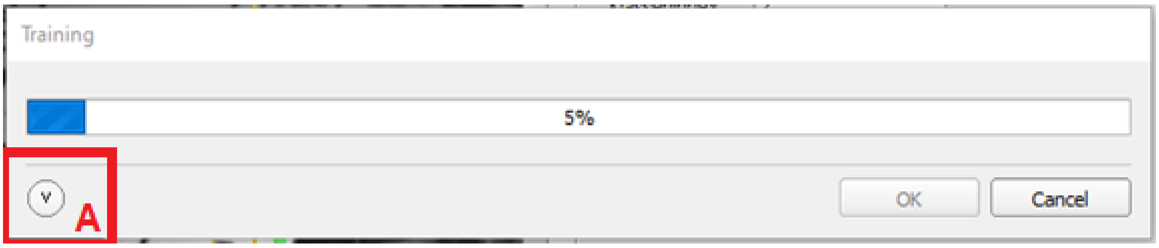


Fig. 206: *Training progress*

Optionally, click on the arrow (A) for more details about the training progress (see Training score). Training is an iterative process in which internal parameters are continuously optimized. For each training step, a subset of the training image data is used for the actual parameter optimization (training run) and a subset is used to check whether this optimization step leads to an improvement (validation run). The curves represent the training score (proportion of correctly classified image data) for each iteration:

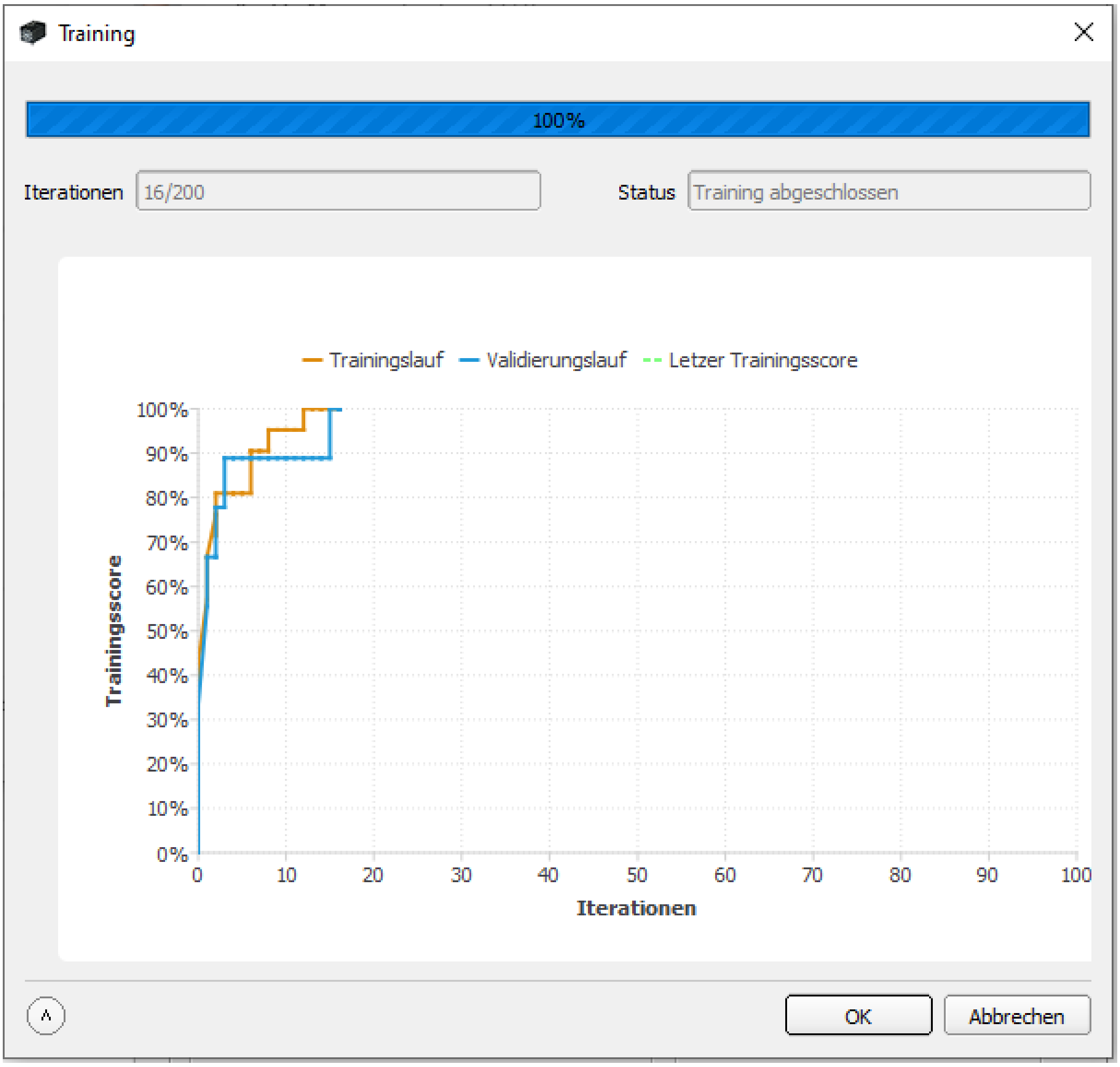
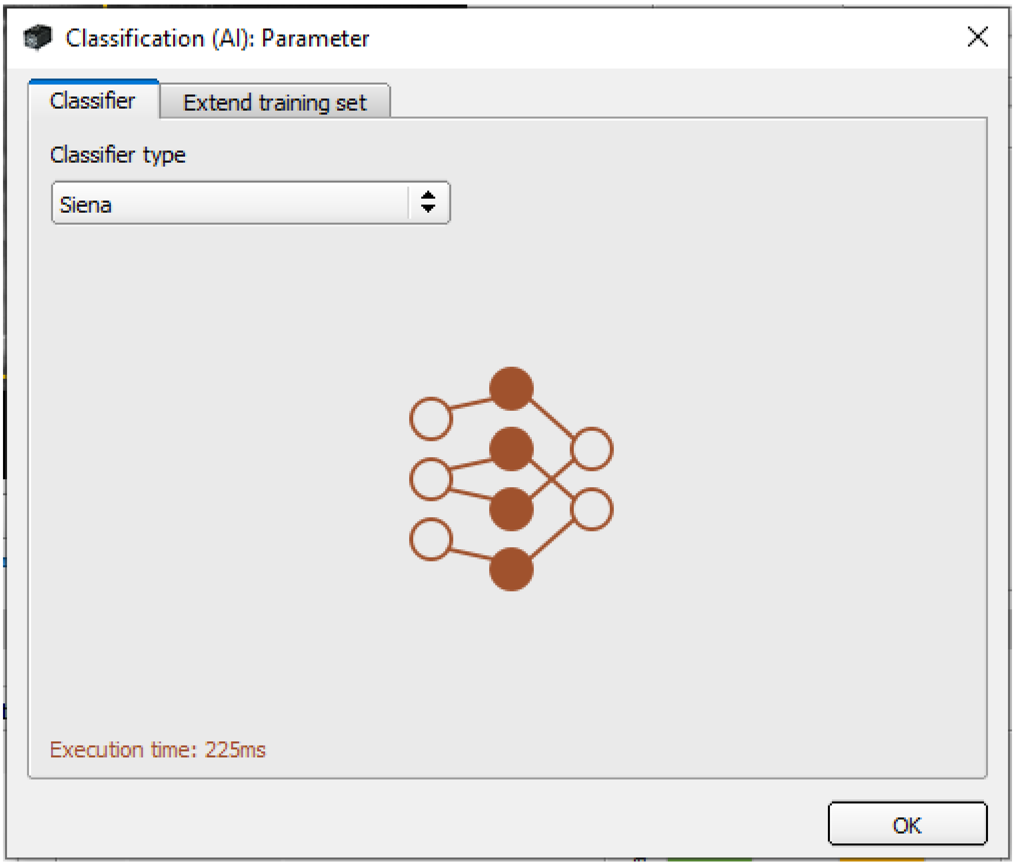


Fig. 207: Training score

8.3.12.3.2 Parameters window

Classifier tab

There are three Classifiers to choose from. These each have a different internal structure. The different Classifiers provide better or worse results depending on the task and image data. Which Classifier is best suited for the given task must be determined experimentally.



Classifier type

Siena

Execution time: 225ms

Fig. 208: Classifier tab

The execution time of the Classifiers is by and large constant, regardless of the ROI size or the resolution of the image. Due to the size of the Classifiers, their maximum number per job dataset is limited. Due to their size, the time required by the device for a job change is also higher than for other Detectors (typically > 1s).

	Siena	Indigo	Olive
Execution time	225 ms	725 ms	900 ms
Size of a Detector in job dataset [1)]	12.1 MByte	17.2 MByte	24.2 MByte
Number Detectors of this type per job dataset (max.) [2)]	10	7	5

[1)] A job dataset can have a maximum size of 120 Mbytes.

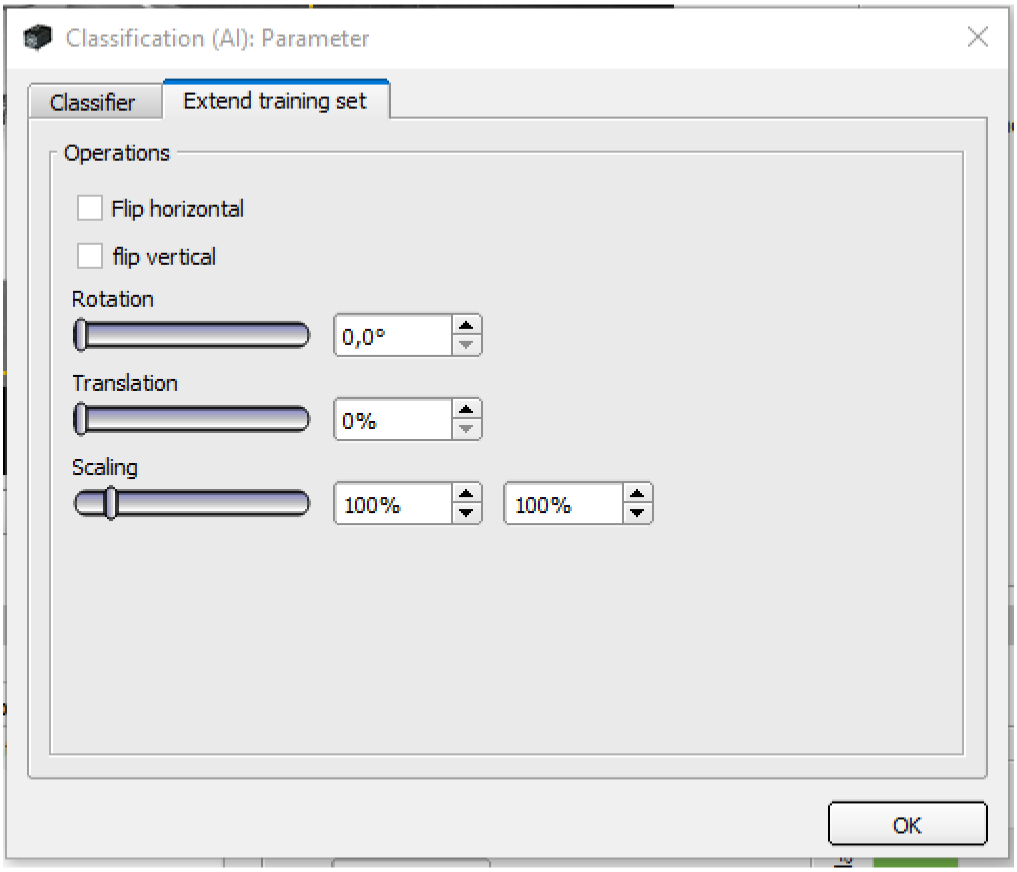
[2)] Also depends on the other Detectors used in the job dataset.

Enlarge training dataset tab

This feature allows the training image dataset to be artificially enlarged using calculated images to cover variations in the process that may occur but for which no image data are available at the time of training.

For example, if you know that an object can occur rotated up to 360° in the image, but you have no or not enough training images of all rotations, you can use the "Rotation" parameter to calculate rotated images that can then be used for Training the Classifier.

NOTE:
The process takes place automatically and the images cannot be viewed in the training image dataset.



Classification (AI): Parameter

□ Flip horizontal

□ flip vertical

Rotation: 0,0°

Translation: 0%

Scaling: 100% 100%

Fig. 209: *Enlarge training dataset tab*

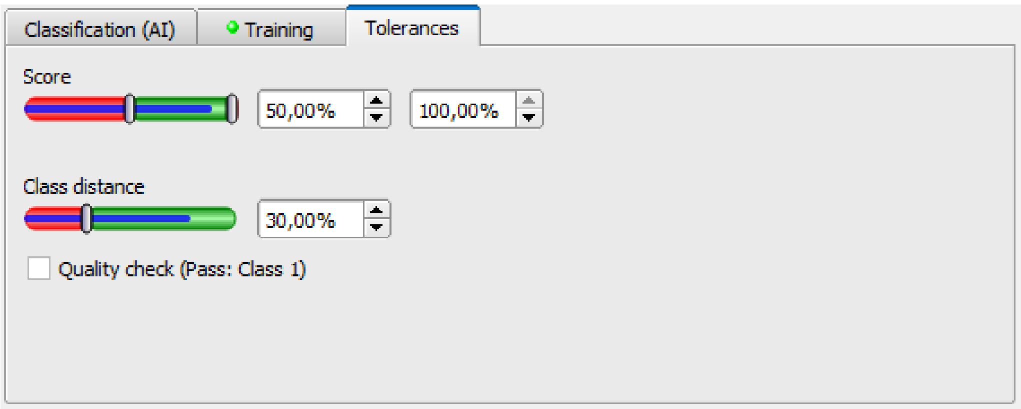
Parameter description:

Parameter	Function
Mirror horizontally	The image content is mirrored horizontally
Mirror vertically	The image content is mirrored vertically
Rotation	The image content is rotated at most to the value specified. 0%: No rotation
Translation	Image content is shifted at most to the specified value in the X and Y direction. 0%: No shift. The shift takes place relative to the image size.
Scaling	The image content is scaled between the given limits. 100% - 100%: No scaling.

8.3.12.4 Tolerances tab

Use this tab to set the tolerances for the Pass/Fail decision. The detector result is Pass if the score **and** the class distance are within the given limits. The score is a confidence measure for the assignment to the class.

The "Quality control (Good parts: Class 1)" function causes the Detector to output passes automatically only if the class has been detected with index 1 (the very first class). This can be used, for example, to perform simple quality checks where there are only the classes "good" and "bad" and where the result is to be communicated as a digital IO signal.



Score: 50,00% – 100,00%

Class distance: 30,00%

□ Quality check (Pass: Class 1)

Fig. 210: *Detector Classification (AI), Tolerances tab*

Parameter description:

Parameter	Function
Score	If the score is in the defined range, the detector result is Pass.
Class distance	If the class distance is greater than specified, the detector result is Pass. Internally, a separate score is calculated for each class. The score issued is the one from the class that has the highest score. This is the most likely class. The difference between the most likely class and the second most likely class is the class distance. A higher class distance means higher reliability in the classification. Default: 30%.
Quality control (Class 1)	If this check mark is set, the detector result is Pass if the result class is Class 1 and the above two criteria are met.

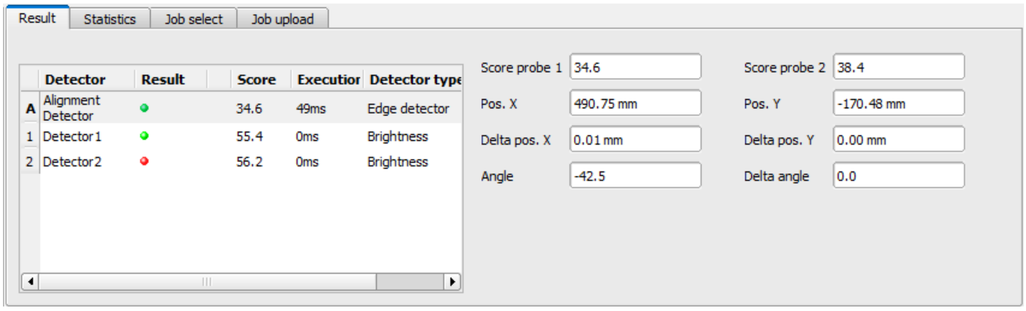
8.3.12.5 Result tab

This function executes the defined job on the PC and displays the "Result Statistic" window with detector list and evaluation results. The execution times are not updated in this mode because they are not available from the sensor.

In Run mode, the detailed test results of the detector marked in the drop-down list are displayed.

In the image window – if adjusted – the image, the search and feature ranges, and the result graphs are displayed.

The displayed parameters vary depending on the selected detector type. To see the test results of another detector, mark it in the drop-down list.



	Detector	Result	Score	Executior	Detector type
A	Alignment Detector	🟢	34.6	49ms	Edge detector
1	Detector1	🟢	55.4	0ms	Brightness
2	Detector2	🔴	56.2	0ms	Brightness

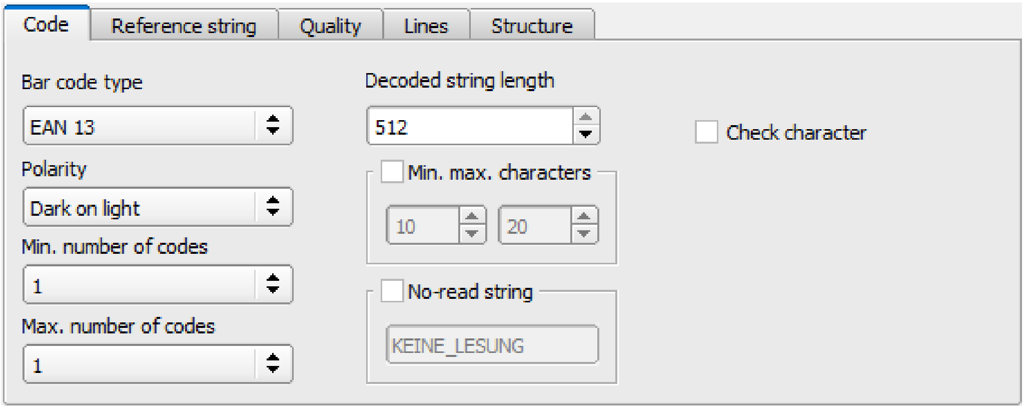
Score probe 1: 34.6
Score probe 2: 38.4
Pos. X: 490.75 mm
Pos. Y: -170.48 mm
Delta pos. X: 0.01 mm
Delta pos. Y: 0.00 mm
Angle: -42.5
Delta angle: 0.0

Fig. 211: *SensoView, Result*

8.3.13 Detector Barcode

This detector is suitable for barcode reading of common 1D codes such as EAN, UPC, RSS, 2/5 Interleaved, 2/5 Industrial, Code 32, Code 39, Code 93, Code 128, GS1, Pharmacode or Codabar.

8.3.13.1 Code tab



Bar code type: EAN 13
Polarity: Dark on light
Min. number of codes: 1
Max. number of codes: 1
Decoded string length: 512
□ Min. max. characters: 10 20
□ No-read string: KEINE_LESUNG
□ Check character

Fig. 212: Detector Barcode, Code tab

Parameter description:

Parameter	Function
Barcode type	Select here the type of barcode to be read. • 2/5 Industrial • 2/5 Interleaved • Codabar • Code 39 • Code 93 • GS1-128 • EAN 8 • EAN 8 Add-On 2 • EAN 13 • EAN 13 Add-On 2 • EAN 13 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 • PharmaCode • RSS 14 / GS 1 • RSS 14 Truncated • RSS 14 Stacked • RSS 14 Stacked Omnidir • RSS Limited • RSS Expanded • RSS Expanded Stacked • Code 32
Polarity	Choices for ink of code "dark on light" or "light on dark".
Min. Number of codes	Minimum number of codes to be read inside the region of interest. If the set minimum number of codes is not found, the detector result will be NOK
Max. number of codes	Maximum number of codes to be read inside the region of interest. If this value is chosen higher than actually necessary, the execution time of the detector may increase slightly.

Parameter	Function
Max. string length	Max. length of a barcode If the contents of the code are longer than the maximum length, the rest will be cut off. If more than one code is read simultaneously, this value must be set to the longest length of the longest code.
Min. Max. characters Number of characters	Number of expected characters in the barcode. Codes with a different number of characters are ignored. If the number of characters of the code is known in advance, this increases detection certainty. If codes with a certain number of characters must be found among several codes, the parameter "Max. number of codes" must be set higher than the number of searched codes.
Text output for incorrect reading	Specifies the text which is output via the interfaces in case of an incorrect reading. The text does not appear in the result display.
Check digit	This parameter activates the recognition of a check character if it is available in the code. Barcodes with check characters are e.g. Code 39, Codabar, 25 Industrial or 25 Interleaved. If this parameter is not selected, the check character is interpreted as a normal data character and output in the string.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Improvement of execution speed

- Region of interest for position (yellow frame) only as large as necessary.

Robust detection

- Region of interest (yellow frame) sufficiently large?
- Distinctive contrast present?
- Was the selection "Check character" activated, even though there is no check character in the code?

8.3.13.2 Reference string tab

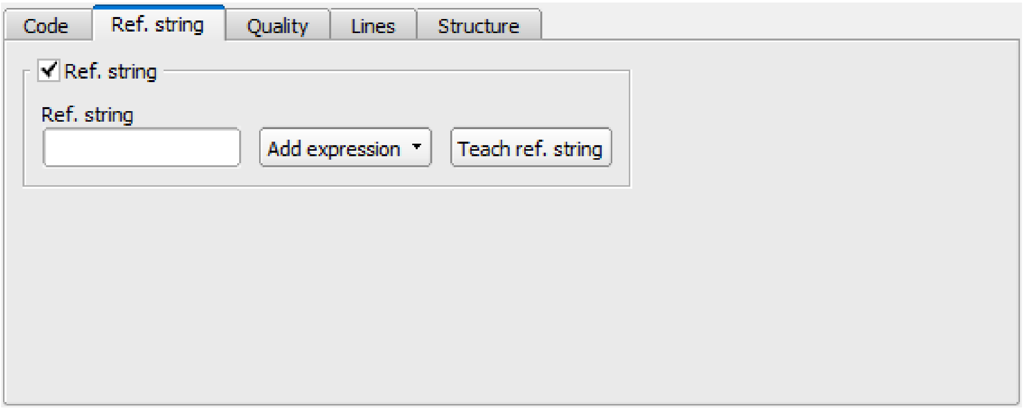


Fig. 213: Barcode detector , Reference string tab

Parameter description:

Parameter	Function
Reference string	This text or regular expressions is taken for verification. This is where specific characters can be found that are directly compared or regular expressions to check the structure of the read result. If codes with a certain reference string must be found among several codes, the parameter "Max. number of codes" in the "Code" tab must be selected higher than the number of searched codes. If the taught-in reference string is not found, the detector result will be NOK
Add expression	Opens a list with examples for regular expressions.
Teach reference string	Reads the code that is currently under the code reader and accepts the read content as a comparison text. This text can be changed later. If in the operating step Output / I/O mapping tab an input is assigned with the "Teach" function, the Reference string is also taught-in again.

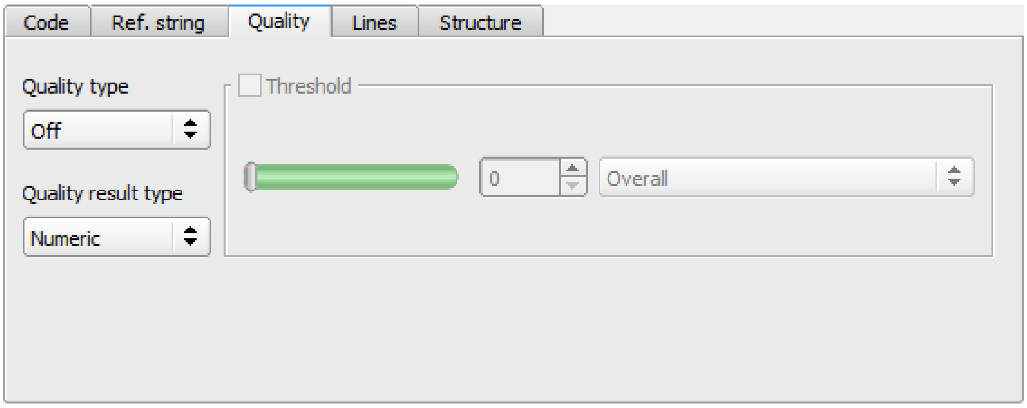
Examples of reference character strings defined by regular expressions

Reference character string	Hit	Example for hit
123	String containing 123	01234
\A123	String beginning with 123	1234
123\Z	String ending with 123	0123
\A123\Z	String matching 123 exactly	123
[123]	Character string containing one of the characters	33
[123]{2}	String containing a sequence of 2 of the characters	23
[12]|[34]	String containing a character of one of both groups (OR function)	4

The most important elements of regular expressions:

^ or \A	Represents the beginning of the character string
$ or \Z	Represents the end of the character string, and possibly includes a newline as the last character
.	Represents every character apart from newline
[...]	Represents any literal listed in the square brackets. If the first character is an '^', the expression is negated. You can use the '-' character, as in '[A-Z0-9]', to specify value ranges. Other characters lose their special meaning within square brackets, except '\'.
*	Allows 0 or more repetitions of the preceding literal / group
+	Allows 1 or more repetitions
?	Allows 0 or 1 repetitions
{n,m}	Allows n to m repetitions
{n}	Allows exactly n repetitions
|	Separates alternative search expressions (OR function)

8.3.13.3 Quality tab



Quality type: Off

Quality result type: Numeric

□ Threshold: 0, Overall

Fig. 214: Detector Barcode, tab Quality

Parameter	Function
Quality type	Selection of the ISO/IEC 15416 standard for the quality check of the code.
Output format	Selection of the output format of the quality parameters: • "Numeric" • "Letter" Both formats correspond to the norms. A and 4 are the best possible grades. The setting made here affects both the display of the quality parameters on the screen and the output of the quality parameters via the interfaces. The assignment is the following: Letter: A B C D F Numeric: 4 3 2 1 0
Threshold	If the determined quality parameter is below the set threshold, the detector result will be NOK

The evaluation of the print quality is carried out according to the international standard ISO/IEC 15416.

For a standard-compliant quality assessment, certain minimum requirements for the mapping of the code in the camera (resolution), the arrangement of the camera, and the type and arrangement of the lighting are prescribed. These are printed in the respective standards.

While the overall quality is the final symbol level, the remaining degrees indicate potential quality-reducing causes. A list of frequently occurring defects and their effect on the individual quality grades can be found in the standard ISO / IEC 15416.

To evaluate the quality, 10 scan lines are placed over the barcode and each of these is evaluated individually. The individual scores are calculated as the average of these 10 scan lines.

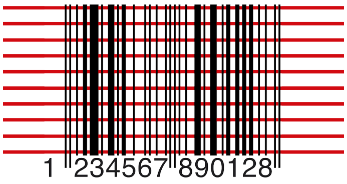


Fig. 215: Scan lines for quality assessment

The quality parameters are composed of individual evaluations, the significance of which can be taken from the respective standards.

Standard 1D barcodes - Parameter description:

Parameter	Function
Q1 Overall	Average of the minimum values of the following quality parameters. The minimum value of a quality parameter results from the poorest value of the 10 scan lines.
Q2 Not used	-
Q3 Not used	-
Q4 Decode	Is set to 4 if the examined barcode symbol could be read, otherwise set to 0
Q5 Symbol Contrast	Difference between maximum and minimum reflectance value of the greyscale profile; higher contrast results in a better degree
Q6 Minimal Reflectance	Is set to 4 if the minimum reflection value of the gray value profile is less than or equal to 0.5 of the maximum reflection value. Otherwise a value of 0 is assigned.
Q7 Minimal Edge Contrast	Evaluates the minimum edge contrast in the greyscale profile. "Edge contrast" is the contrast between two adjacent symbol elements (line-to-hole or hole-to-line).
Q8 Modulation	Evaluates the amplitude between the symbol elements. Higher amplitudes mean that lines and holes can be distinguished more reliably from one another and this degree is rated higher
Q9 Defects	Irregularities in the grey tone profile within individual symbol elements or the squeaky zones, the presence of which is indicated by a lower degree
Q10 Decodability	Designates deviations of the symbol element widths from their nominal value, which is defined in the corresponding symbology standard.
Q11 Additional Requirements	Other symbology specific requirements such as: quiet zone widths, wide/narrow ratio, inter character gaps, guarding patterns or others.

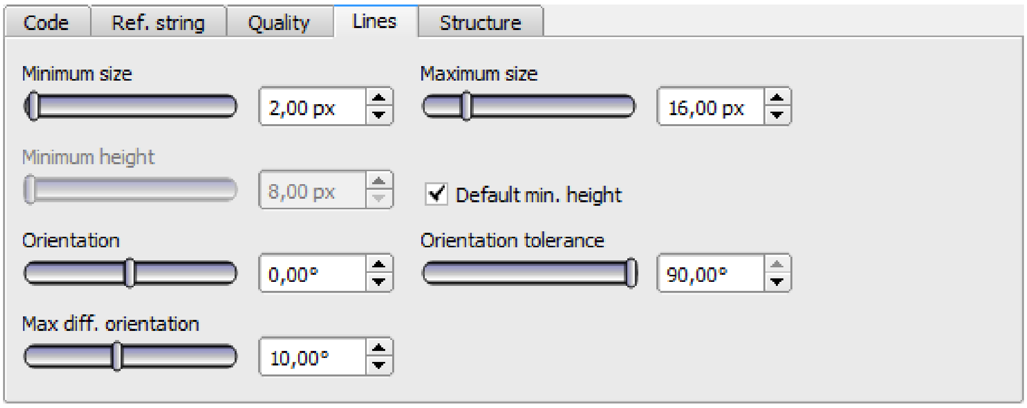
Composite and stacked barcodes - Parameter description:

The print quality rating of a "composite" barcode includes the following 24 grades:

Parameter	Function
Q1 Overall	Minimum value of all remaining degree values
Q2 Overall Linear	Minimum value of Q parameters Q4-Q11; represents the total degree of the linear (1D) part of the composite symbol
Q3 Overall Composite	Minimum value of Q parameters Q12-Q24; represents the total degree of the composite (2D) part of the composite symbol
LINEAR: Q4 Decode Q5 Symbol Contrast Q6 Minimal Reflectance Q7 Minimal Edge Contrast Q8 Modulation Q9 Defects Q10 Decodability Q11 Additional Requirements	The grades from the group LINEAR correspond to those from the simple 1D barcode case described above.

Parameter	Function
COMPOSITE: Q12 Decode Q13 Rap Overall	The grades from the group COMPOSITE correspond to the PDF 417 quality grades, whereby rap overall is named after the so-called RAP start-stop pattern, which is specific for composite symbols.
COMPOSITE RAP: Q14 Contrast Q15 Minimal Reflectance Q16 Minimal Edge Contrast Q17 Modulation Q18 Defects Q19 Decodability Q20 Codeword Yield Q21 Unused Error Correction Q22 Modulation Q23 Decodability Q24 Defects	In addition, the COMPOSITE RAP subgroup represents the individual gray level profile grades of the RAP pattern. These are consistent with the simple 1D barcode quality grades.

8.3.13.4 Lines tab



Minimum size: 2,00 px

Maximum size: 16,00 px

Minimum height: 8,00 px

☑ Default min. height

Orientation: 0,00°

Orientation tolerance: 90,00°

Max diff. orientation: 10,00°

Fig. 216: Detector Barcode, Lines tab

Parameter description:

Parameter	Function
Minimum line width	The minimum size of an element, i.e. minimum width of all lines and spaces. For very small barcodes, the value should be reduced to 1.5. In the case of huge barcodes, the value should be increased, which results in a shorter execution time.
Maximum line width	The maximum size of an element, i.e the maximum thickness of all lines and spaces. This value should be large enough to find the candidate region for the entire symbol. On the other hand, the value should not be so large that two adjacent barcodes merge into a single candidate.
Minimum line height	The minimum barcode height For very flat barcodes with a height of less than 16 pixels, it makes sense to set the height manually so that the barcode can be found and read. The minimum height is 8 pixels. If the barcode is very high, e.g. 70 pixels and more, manually adjusting to the respective height can lead to quicker reading. If the "Default" checkbox is active, a default preset value is used.

Parameter	Function
Orientation	The Orientation parameter can be used to restrict the angle range for the code reading. Barcodes with rotational positions outside the specified orientation are not read. The parameter is used, for example, if a barcode can lie in different rotational positions in front of the code reader and not all rotational positions should be read. If codes with a certain rotational position need to be found among several codes, then the parameter max. number of codes in the Code tab should be selected higher than the number of searched codes. If the barcodes only appear in the edited images with a certain orientation, then you can reduce the value range accordingly. This will detect wrong candidates sooner. The execution time of the operator is shortened if the orientation angle is restricted. This strategy is especially true when the edited images contain a lot of background texture with misoriented, barcode-like structures.
Orientation tolerance	Tolerance of orientation. See "Orientation" for further details.
Max. deviation (orientation)	A potential barcode comprises lines, and hence edges with a consistent orientation. The size "Maximum orientation deviation" indicates how strong the difference in the orientation of adjacent edges may be. The maximum orientation deviation is a differential angle in degrees. If a barcode is frayed, i.e. the line edges are interefered, the maximum orientation deviation should be selected as large. However, with small values, the number of wrong barcode candidates can be reduced.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Parameter Orientation

The following figure illustrates the orientation.

NOTE:
The specification of the orientation refers to the image and not to the rotational position of the region of interest.

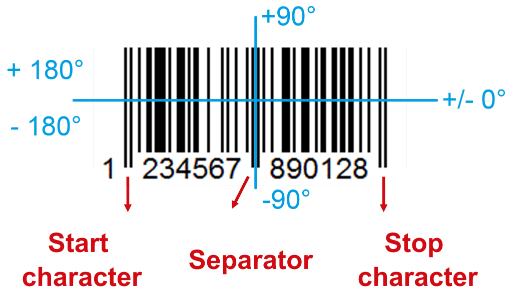


Fig. 217: Parameter Orientation

Improvement of execution speed

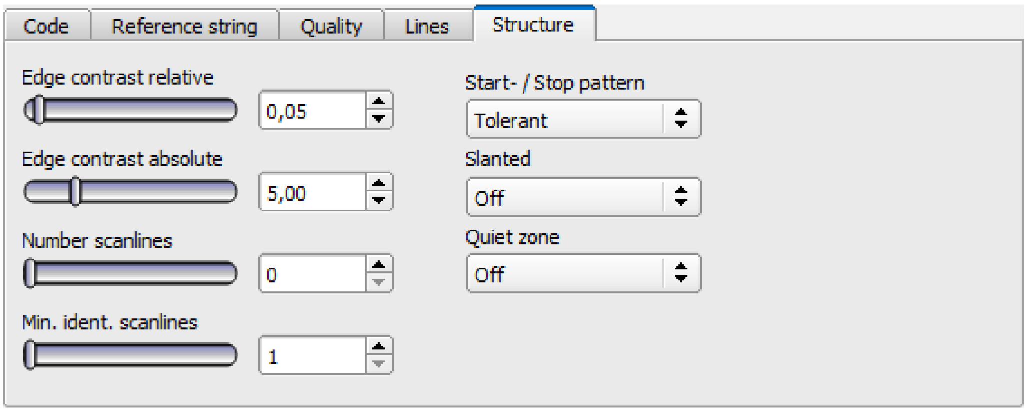
- Region of interest for position (yellow frame) only as large as necessary.

Robust detection

- Region of interest (yellow frame) sufficiently large?
- Distinctive contrast present?
- Are the thresholds set correctly?

- Code size sufficient in the field of view?
- Is the line width sufficiently large?

8.3.13.5 Structure tab



Edge contrast relative: 0,05

Edge contrast absolute: 5,00

Number scanlines: 0

Min. ident. scanlines: 1

Start- / Stop pattern: Tolerant

Slanted: Off

Quiet zone: Off

Fig. 218: Detector Barcode, tab Structure

Parameter description:

Parameter	Function
Edge contrast relative	Edges are found within a scan line using a threshold. The parameter "Edge contrast relative" defines how this threshold is calculated relative to the dynamic range of the gray values along the scan line. For large background noise or noise, edge contrast should be set relatively higher. Typical value range: [0.05 ... 0.2]; Default: 0.05
Edge contrast absolute	"Edge contrast absolute" prevents the misdetections of edges. For high noise images, this parameter should be set higher. For low-contrast, noise-free images, over-reading may interfere with correct edge detection. In such cases, it is advisable to reduce the value or to set it to 0.0. Typical values: [0.0 ... 10.0]; Default: 5.0
Number scan lines	Number of scan lines used during the scanning of a code. Reducing the number of scan lines improves speed. Better quality images require fewer scan lines than lower quality images. For average image quality, a value between 2 and 5 are adequate. If a barcode cannot be found after reducing the scan lines, the number of scan lines must be increased again. Typical values: [0, 5, 10, 20 ...]; Default: 0
Min. ident. scan lines	The minimum number of identical scan lines to accept a code as read. If this parameter is not set (value 0), the barcode is read as soon as the scan line has been successfully decoded. Faulty readings can be reduced if this parameter is set to 2 or higher. Typical values: [0, 2, 3, ...]; Default: 0
Start / Stop pattern	Sets the search for start or stop patterns within a scan line to "Tolerant" or "Exact". "Tolerant" increases the general read rate, especially in images with poor contrast. "Exact" increases the robustness against incorrect decoding but can also reduce the general reading rate. Standard value: "Tolerant"

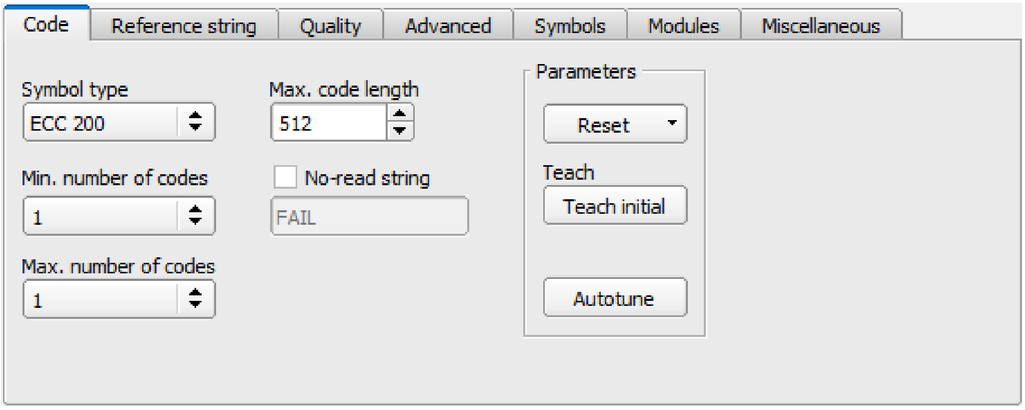
Parameter	Function
Slanted	If "slanted" = "On", improved readability when individual lines of the barcode are aligned obliquely to the main direction of the code, e.g. if the code appears distorted by an uneven surface. If "slanted" = "Off", default setting when all lines of the barcode appear parallel in the image. If "Tilt" = "Auto", first the "Off" position and then the "On" position is tested, which can increase reading time. Values: "Off", "Auto", "On"; Default: "Off"
Quiet zone	Controls the detection of quiet zones of a barcode. With "Quiet zone" = "On", the quiet zones must be at least as wide as specified by the corresponding barcode standard. With "Quiet zone" set to an integer (≥ 1), the quiet zones of at least "Quiet zone" x X pixels must be observed. With "Quiet zone" = "Tolerant", a limited number of edges are allowed in the quiet zone, but at most 1 per 4 module widths. The goal is to prevent you from recognizing only part of the barcode, but still be able to read codes with a simple violation of the quiet zone. With "Quiet zone" = "Off", the detection of the quiet zones is disabled. Detection of the quiet zone prevents simple barcodes from being found within a stroke sequence of a longer and / or more complex barcode. Usually, values between 2 and 4 achieve optimal results because they suppress false barcodes while still tolerating small interferences such as text, label edges, etc. Typical values: "Off" "On", 1, 2, 3, 4, 5; Default: "Off"

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.14 Detector Datacode

This detector is suitable for reading 2D DataMatrix codes such as ECC200, QR Code, ECC200 (GS1), QR Code (GS1), PDF 417, Aztec Code and Micro QR Code.

8.3.14.1 Code tab



Symbol type: ECC 200

Max. code length: 512

Min. number of codes: 1

☐ No-read string: FAIL

Max. number of codes: 1

__Fig. 219:__ Datacode detector, Code tab

Parameter description:

Parameter	Function
Code type	Select here the type of the data code to be read. • ECC 200 • QR Code • PDF 417 • ECC 200 (GS1) • QR Code (GS1) • Micro QR code • Aztec code • Aztec code (GS1) Addition GS1: Check whether the code complies with the GS1 standard. Codes that do not comply with the standard will not be read with this setting.
Min. Number of codes	Minimum number of codes to be read inside the region of interest. If the set minimum number of codes is not found, the detector result will be NOK
Max. number of codes	Maximum number of codes to be read inside the region of interest. If this value is chosen higher than actually necessary, the execution time of the detector may increase slightly.
Max. string length	Max. length of a code. If the contents of the code are longer than the maximum length, the rest will be cut off. If more than one code is read simultaneously, this value must be set to the longest length of the longest code.
Text output for incorrect reading	Specifies the text which is output via the interfaces in case of an incorrect reading.
Reset	Reset parameter is used to reset the learned parameters to the initial state before teaching. There are the options "Standard", "Advanced", and "Maximum". "Standard" sets the limits of the region of interest so that the majority of the possible codes are recognized during training. If your code is still not recognized, select the setting "Advanced". If the code still cannot be read, set the setting to "Maximum". The settings "Advanced" and "Maximum" may slow down the execution time. The difference to a complete resetting of the sensor is that only the parameters for the Data matrix code are reset. The basic parameters of the sensor, e.g. for lighting, inputs, serial interface, etc. are retained. After resetting the parameters, a new teach-in process can be started again with "Teach-in".
Initial teach / Additive teach	Teach-in: The region of interest of the sensor is searched for a Data matrix code. If a code was found, the parameters are stored for this code. After successful teaching, the found code is marked with a green frame. In the "Run" mode, only this very taught-in code is searched for. After completing the teach-in, the button "Additional teach-in" appears at the same place. This allows the extension of the taught-in parameters to either read several different codes in one configuration or to capture any existing spreads in the print quality of a single code. "Additional teach-in" extends the already taught-in parameter set. If in the setup step Output / I/O mapping tab an input is assigned with the "Teach" function, an "Additionl teach-in" is performed for each image acquisition. If the Reference string is activated, it will also be taught.
Autotune	Automatic setting (pre-processing filter and image settings) for the optimization of code reading.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Improvement of execution speed

- Region of interest for position (yellow frame) only as large as necessary.

Robust detection

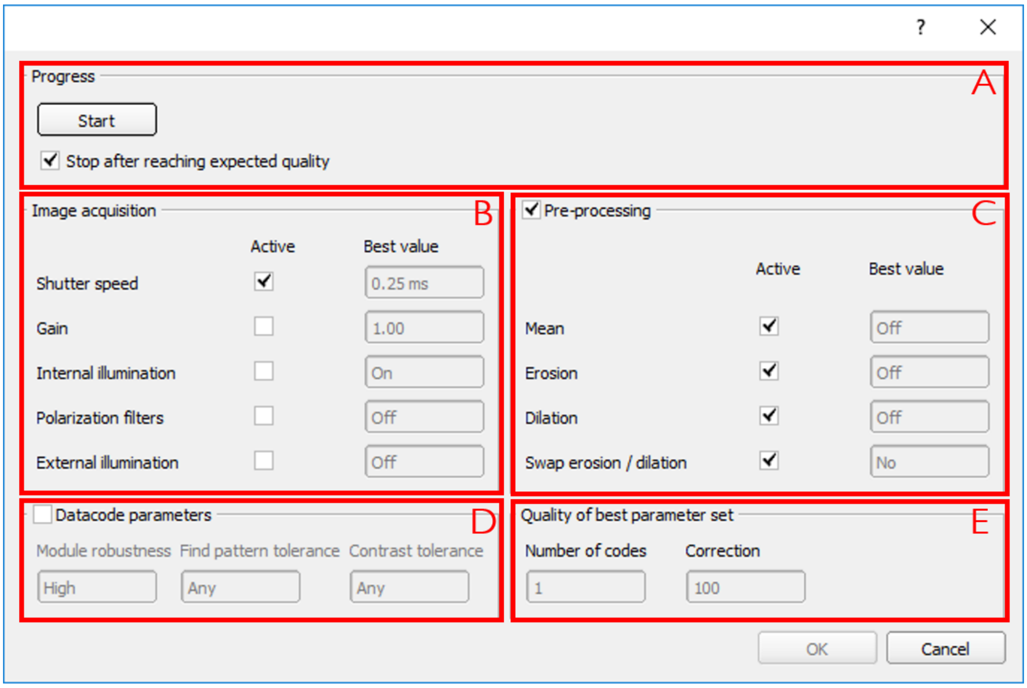
- Region of interest (yellow frame) sufficiently large?
- Distinctive contrast present?
- Are thresholds set correctly?

8.3.14.1.1 Autotune

If you select the "Autotune" function, the vision sensor will automatically configure its settings in order to optimize its code reading functionality.

The function always starts with the parameters already set by the user. So if parameters are roughly set before starting the "Autotune" function, then Autotune fine tunes to optimize the result.

After the optimization run of "Autotune", "OK" or "Cancel" can be selected. In case of "OK", the newly found parameters are used. When "Cancel" is selected, the old parameters are restored to the values from before the "Autotune" execution.



Progress

☑ Stop after reaching expected quality

Image acquisition

	Active	Best value
Shutter speed	☑	0.25 ms
Gain	□	1.00
Internal illumination	□	On
Polarization filters	□	Off
External illumination	□	Off

☑ Pre-processing

	Active	Best value
Mean	☑	Off
Erosion	☑	Off
Dilation	☑	Off
Swap erosion / dilation	☑	No

□ Datacode parameters

Module robustness	Find pattern tolerance	Contrast tolerance
High	Any	Any

Quality of best parameter set

Number of codes	Correction
1	100

Fig. 220: "Autotune" window

The "Autotune" function consists of the following elements:

A: Progress	
Start	Start of "Autotune" function. After pressing start, the progress is displayed.
Stop after reaching expected quality	The "Autotune" function stops the automatic settings when the minimum required quality is reached.

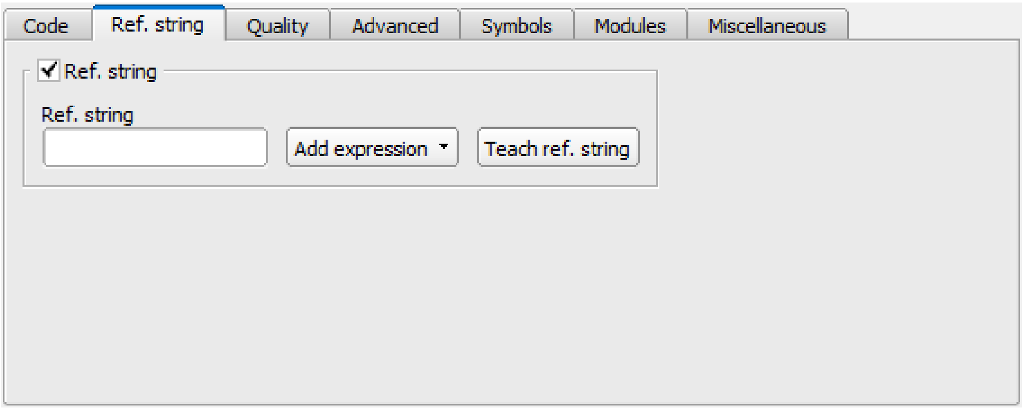
B: Image acquisition	
Shutter speed	**Enabled:** If the "Enabled" checkbox is enabled, the parameters that the VISOR® vision sensor should determine automatically will be determined. Non-activated parameters remain unchanged. **Best Value:** The "Best Value" field shows the last setting which was determined by the Autotune function.
Gain	
Internal illumination	
Polarization filters	
External illumination	

C: Pre-processing	
Mean	**Enabled:** If the "Enabled" checkbox is enabled, the parameters that the VISOR® vision sensor should determine automatically will be determined. Non-activated parameters remain unchanged. **Best Value:** The "Best Value" field shows the last setting which was determined by the Autotune function.
Erosion	
Dilation	
Inversion order: erosion / dilatation	

D: Code parameters	
Module robustness	The best settings found by the Autotune function are displayed.
Search pattern tolerance	
Contrast tolerance	

E: Quality of best parameter set	
Number of codes	Number of codes in the field of view tested by the Autotune function.
Correction	Decode error, which is achieved with activated parameters.

8.3.14.2 Reference string tab



☑ Ref. string

Ref. string

__Fig. 221:__ Detector Datacode, tab Reference string

Parameter description:

Parameter	Function
Reference string	This text or regular expressions is taken for verification. This is where specific characters can be found that are directly compared or regular expressions to check the structure of the read result. If codes with a certain reference string must be found among several codes, the parameter "Max. number of codes" in the "Code" tab must be selected higher than the number of searched codes. If the taught-in reference string is not found, the detector result will be NOK
Add expression	Opens a list with examples for regular expressions.
Teach reference string	Reads the code that is currently under the code reader and accepts the read content as a comparison text. This text can be changed later. If in the operating step Output / I/O mapping tab an input is assigned with the "Teach" function, the Reference string is also taught-in again.

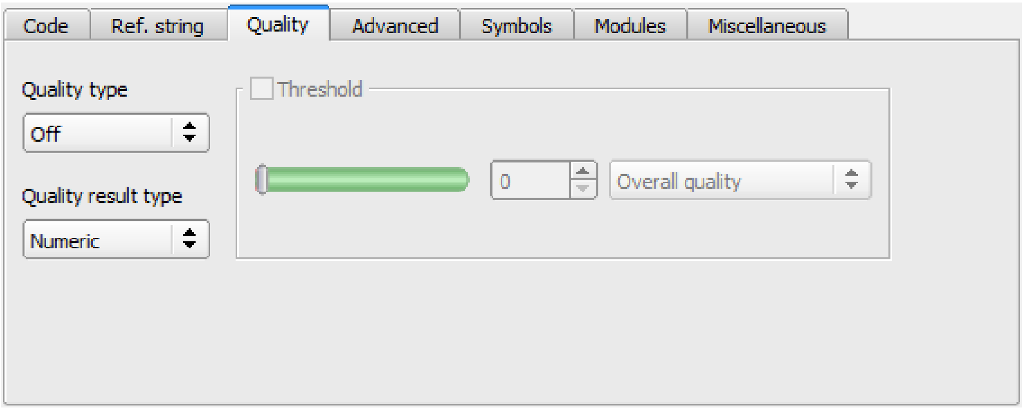
Examples of reference character strings defined by regular expressions

Reference character string	Hit	Example for hit
123	String containing 123	01234
\A123	String beginning with 123	1234
123\Z	String ending with 123	0123
\A123\Z	String matching 123 exactly	123
[123]	Character string containing one of the characters	33
[123]{2}	String containing a sequence of 2 of the characters	23
[12]|[34]	String containing a character of one of both groups (OR function)	4

The most important elements of regular expressions:

^ or \A	Represents the beginning of the character string
$ or \Z	Represents the end of the character string, and possibly includes a newline as the last character
.	Represents every character apart from newline
[...]	Represents any literal listed in the square brackets. If the first character is an '^', the expression is negated. You can use the '-' character, as in '[A-Z0-9]', to specify value ranges. Other characters lose their special meaning within square brackets, except '\'.
*	Allows 0 or more repetitions of the preceding literal / group
+	Allows 1 or more repetitions
?	Allows 0 or 1 repetitions
{n,m}	Allows n to m repetitions
{n}	Allows exactly n repetitions
|	Separates alternative search expressions (OR function)

8.3.14.3 Quality tab



Quality type: Off

Quality result type: Numeric

□ Threshold: 0 Overall quality

Fig. 222: Detector Datacode, tab Quality

Parameter	Function
Quality type	Selection of the standard for the quality check of the code. • ISO/IEC 15415 • AIM DPM 2006 • ISO/IEC TR 29158 • SEMI T10
Output format	Selection of the output format of the quality parameters (depending on the selected standard): • "Numeric" • "Letter" • "Numeric decimal" The formats "Numeric" and "Letter" correspond to the standards, "Numeric decimal" is an additional output format beyond the standard. A and 4 are the best possible grades. The setting made here affects both the display of the quality parameters on the screen and the output of the quality parameters via the interfaces. The assignment is the following: Letter: A B C D F Numeric: 4 3 2 1 0 Numeric decimal: ≥ 3.5 2.5...3.4 1.5...2.4 0.5...1.4 ≤ 0.4
Threshold	If the determined quality parameter is below the set threshold, the detector result will be NOK

Quality parameters are provided as additional information to evaluate the quality of the codes.

For a standard-compliant quality assessment, certain minimum requirements for the mapping of the code in the camera (resolution), the arrangement of the camera, and the type and arrangement of the lighting are prescribed. These are printed in the respective standards.

The quality parameters are composed of individual evaluations, the significance of which can be taken from the respective standards.

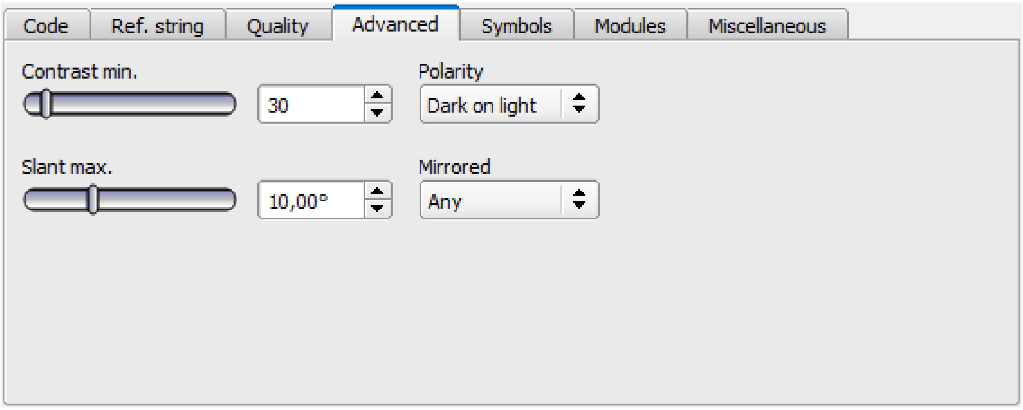
Overview quality parameters

ISOIEC 15415			
	ECC200 Atzec	**QR Code MicroQR**	**PDF417**
Q1	OverallQuality	OverallQuality	OverallQuality
Q2	Contrast	Contrast	Start/StopPattern
Q3	Modulation	Modulation	CodewordYield
Q4	FixedPatternDamage	FixedPatternDamage	UnusedErrorCorrection
Q5	Decode	Decode	Modulation
Q6	AxialNonuniformity	AxialNonuniformity	Decodability
Q7	GridNonuniformity	GridNonuniformity	Defects
Q8	UnusedErrorCorrection	UnusedErrorCorrection	Aperture
Q9	ReflectanceMargin	ReflectanceMargin	
Q10	PrintGrowth	PrintGrowth	
Q11	ContrastUniformity	ContrastUniformity	
Q12	Aperture	FormatInformation	
Q13		VersionInformation	
Q14		Aperture	
Q15			
Q16			
Q17			
Q18			
Q19			
Q20			
Q21			
	Legends for area:		
	0-4, A-F		
	0.0-1.0*100	0 is the best value	
	0.0-1.0*100	100 is the best value	
	float		

	Aimdpm 1 2006 ISOIEC TR 29158		**Semi T10**	
	ECC200 Atzec	**QR Code MicroQR**	**ECC200**	
Q1	OverallQuality	OverallQuality	P1Row	Q1
Q2	CellContrast	CellContrast	P1Column	Q2
Q3	CellModulation	CellModulation	P2Row	Q3
Q4	FixedPatternDamage	FixedPatternDamage	P2Column	Q4
Q5	Decode	Decode	P3Row	Q5
Q6	AxialNonuniformity	AxialNonuniformity	P3Column	Q6

	Aimdpm 1 2006 ISOIEC TR 29158		Semi T10	
	ECC200 Atzec	QR Code MicroQR	ECC200	
Q7	GridNonuniformity	GridNonuniformity	P4Row	Q7
Q8	UnusedErrorCorrection	UnusedErrorCorrection	P4Column	Q8
Q9	MeanLight	MeanLight	Rows	Q9
Q10	ReflectanceMargin	ReflectanceMargin	Columns	Q10
Q11	PrintGrowth	PrintGrowth	SymbolContrast	Q11
Q12	ContrastUniformity	ContrastUniformity	SymbolContrastSNR	Q12
Q13	Aperture	FormatInformation	HorizontalMarkGrowth	Q13
Q14		VersionInformation	VerticalMarkGrowth	Q14
Q15		Aperture	DataMatrixCellWidth	Q15
Q16			DataMatrixCellHeight	Q16
Q17			HorizontalMarkMisplacement	Q17
Q18			VerticalMarkMisplacement	Q18
Q19			CellDefects	Q19
Q20			FinderPatternDefects	Q20
Q21			UnusedErrorCorrection	Q21
	Legends for area:			
	0-4, A-F			
	0.0-1.0*100	0 is the best value		
	0.0-1.0*100	100 is the best value		
	float			

8.3.14.4 Advanced tab



Contrast min. 30

Polarity: Dark on light

Slant max. 10,00°

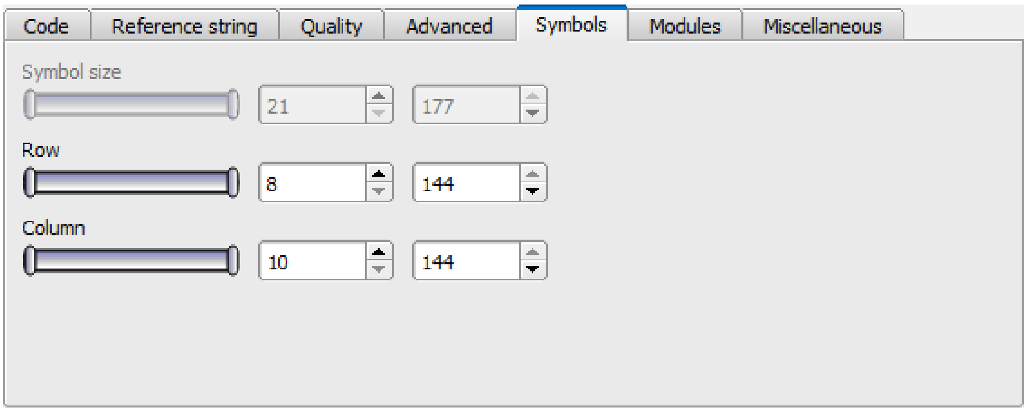
Mirrored: Any

Fig. 223: Detector Datacode, tab Advanced

Parameter description:

Parameter	Function
Min. Contrast	Minimum contrast in grayscale between light and dark parts of the code, value range (1 ... 100).
Slant max.	Maximum deviation of the angle in the L-shaped finder pattern from the (ideally) right angle. The specification corresponds to perspective distortions that may occur when the symbol is printed or when the image is captured.
Polarity	Possible settings: Should light code be read on a dark background or dark code on a light background?
Mirrored	Setting option, whether the code was applied mirrored or not. Due to the symmetry of the code, this is not visible to the eye. The function is helpful if e.g. codes on a transparent surface should be read from behind.
Quiet zone required	Controls the behavior of detecting symbols that have missing or severe defects in their quiet zone: • Yes: Symbols with missing or defective quiet zone are discarded (compliant with standards) • No: Symbols are returned as a result if they can be read (non-standard)

8.3.14.5 Symbols tab



Symbol size 21 177

Row 8 144

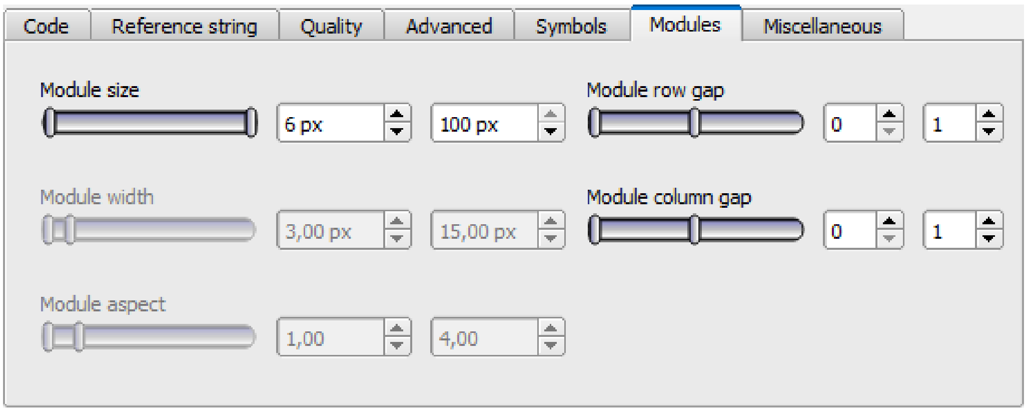
Column 10 144

Fig. 224: Detector Datacode, tab Symbols

Parameter description:

Parameter	Function
Symbol size min. / max.	Only QR code: Size of symbol in the image in pixels.
Rows min. / max.	Only ECC200 and PDF 417: Number of rows including finder pattern.
Columns min. / max.	Only ECC200 and PDF 417: Number of columns including finder pattern.

8.3.14.6 Modules tab



Module size 6 px 100 px

Module row gap 0 1

Module width 3,00 px 15,00 px

Module column gap 0 1

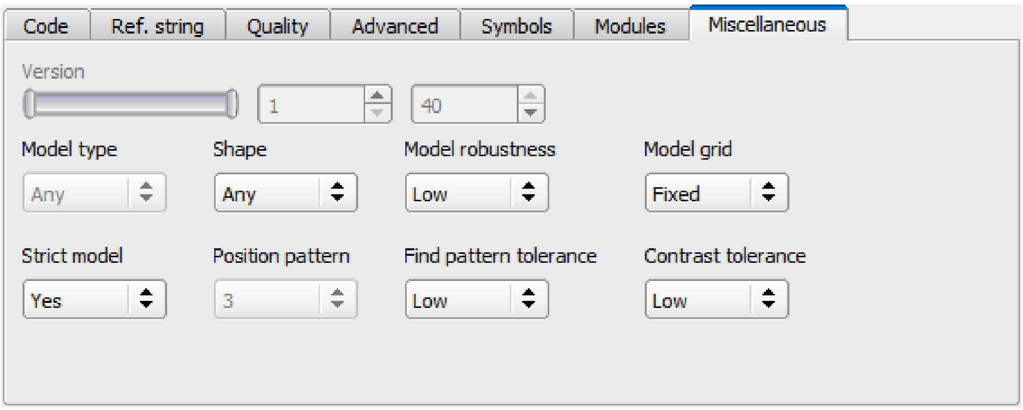
Module aspect 1,00 4,00

Fig. 225: Detector Datacode, tab Modules

Parameter description:

Parameter	Function
Module size min. / max.	Size of modules in pixels.
Module width min. / max.	Only PDF 417: Width of the modules in the image in pixels.
Module aspect min. / max.	Only PDF 417: Minimum aspect of modules in the image (height too wide).
Column / line spacing	Only ECC200 and QR code: Allowable space between two columns, e.g. with nailed codes that have no area-wide modules.

8.3.14.7 Miscellaneous tab



Version 1 40

Model type Any

Shape Any

Model robustness Low

Model grid Fixed

Strict model Yes

Position pattern 3

Find pattern tolerance Low

Contrast tolerance Low

Fig. 226: Datacode detector, Miscellaneous tab

Parameter description:

Parameter	Function
Version	Only QR code: Version of the code, with Model 1 range of values 1 ... 14, with Model 2 range of values 1 ... 40. Version indicates the size of the code. Version 1 means 21 x 21 modules, Version 2 means 25 x 25 modules . . . (4 more modules per version)

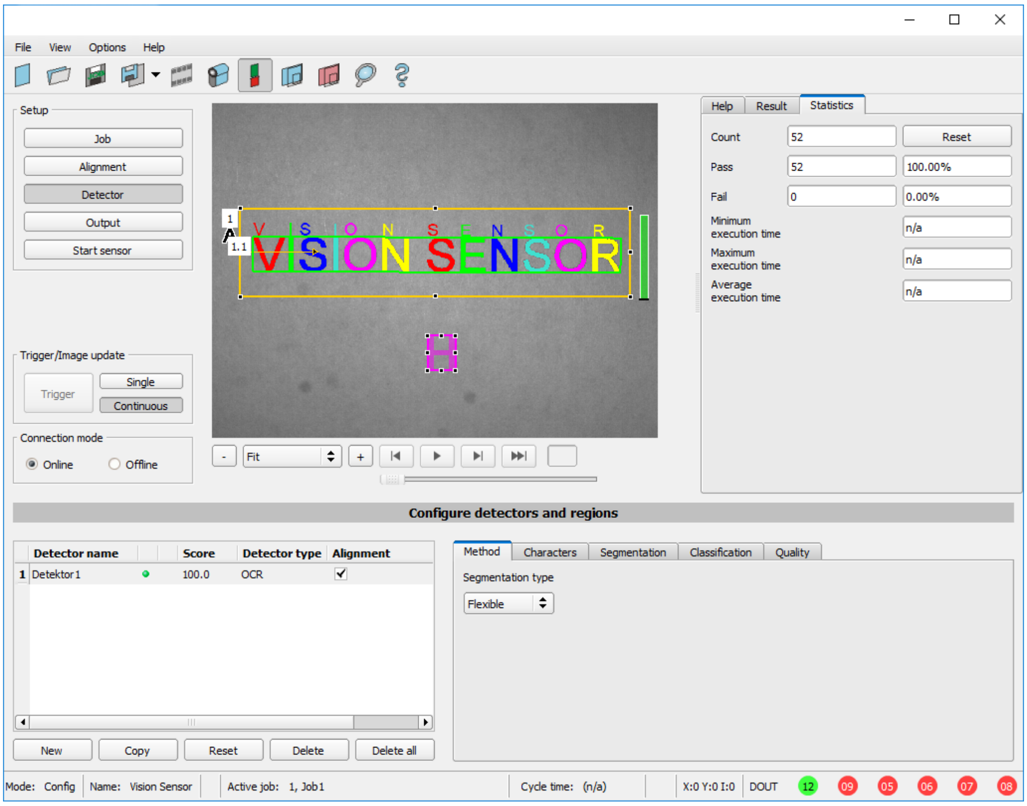
Parameter	Function
Search strategy	Determines the number of candidates for reading attempts. If this parameter is set to "Fault tolerant", more potential candidates are taken into account and the probability increases of finding codes that are difficult to read. The "Fast" setting produces a faster reading result for clearly printed codes.
Model type	Only QR code: Model 1 or Model 2. Model 2 is the newer version, supporting larger codes.
Shape	Only ECC200 and QR code: Used to specify whether rectangular or quadratic codes should be read.
Module robustness	Robustness of the decoding of data codes with very small module size. Setting the parameter to 'high' increases the likelihood of being able to decode data codes with very small module sizes. In addition, in this case, the minimum module size should also be adjusted accordingly, i.e. be set to the assumed minimum module size or module width.
Module tab	Only ECC200: Indication of whether the size of the modules may vary within a certain range. Depending on this parameter, different algorithms are used to calculate the module positions. In one case ('fixed'), a fixed grid in which the distances between the module centers are all the same is used. In the other case ('variable'), the grid is aligned on the alternate side of the finder pattern. With 'any', both variants for the grid are tried one after the other. Note that the value of 'module_grid' is ignored if 'finder_pattern_tolerance' is set to 'high'. In this case, a solid grid is always assumed. List of values: 'fixed', 'variable', 'any' Default: 'fixed' (enhanced: 'any').
Strict model	Specifies whether the entered parameters must be exactly adhered to. If you select "Yes", codes outside of the parameter limits will be ignored.
Position pattern	Only QR code: Number of position detection patterns that must be clearly visible in the image in order to search for a code.
Search pattern tolerance	Only ECC200: Tolerance of the search with respect to a disturbed or missing finder pattern. The finder pattern encompasses both the L-shaped and the opposite alternating sides. In one case ('low'), it is assumed that the finder pattern is present to a high degree and shows almost no disturbances. In the other case ('high'), the finder pattern may be heavily disturbed or missing completely without influencing the recognition. It should be noted, however, that in this variant, an increased computing time should be expected.
Contrast tolerance	Tolerance in code search for strong local contrast fluctuations.

8.3.15 Detector OCR

This detector is suitable for plain text reading of printed, lasered or needled fonts and offers a high reading rate for difficult fonts or fluctuating marking quality by using neural networks. It is suitable, for example, for reading dot print, as in the automotive industry, and fonts for the pharmaceutical, semiconductor and food industries are also pre-installed.

8.3.15.1 Procedure

The following describes the procedure for setting up an OCR detector for clear text reading step by step. Since the setup steps are based on the results of the previously performed steps, this sequence must be followed for a correct, process-reliable function.



Count: 52 | Reset

Pass: 52 | 100.00%

Fail: 0 | 0.00%

Minimum execution time: n/a

Maximum execution time: n/a

Average execution time: n/a

	Detector name		Score	Detector type	Alignment
1	Detektor 1	●	100.0	OCR	☑

Segmentation type: Flexible

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 227: Detector OCR

Basic sequence of setting parameters

1. Optimization of the output image; use of the "Pre-processing" tab under "Job".
2. Segmentation (separating characters from the background) using the "Characters" and "Segmentation" tabs.
3. Classification (reading of characters) using the "Classification" tab: selection of a character set, specification of a reference string, optimization of reading results with regular expressions .
4. "Quality" tab: Discard characters that could not be classified with sufficient quality.

NOTE:
For the OCR detector, it is not sufficient to adjust the parameters on one single image. Process reliability can only be judged on the largest possible image series - the more images are used, the more stable the result! Save typical image sequences and load them into simulation mode for parameter optimization. Use at least a few dozen images that show the variability of the process.

Step 1: Optimization of the original image

1. Setup step "Job" / Tab "Image acquisition": Optimize contrast, optimize brightness
2. Adjust external lighting if necessary. If standard illumination is not sufficient for raised or recessed fonts, a better result can be achieved with the "Multishot" option if necessary.
3. The segmentation can be improved using preprocessing filters under "Job" / "Pre-processing" tab.
   E.g. smoothing filters "Gauss", "Mean" (for stable segmentation) or "Dilatation" / "Erosion" or a combination of these.
   (Additional information: Pre-processing tab)
4. Display characters to be read as large as possible in the image

Step 2: Segmentation

1. Select the segmentation method in the "Method" tab: "Flexible" or "Fast" (Additional information: Method tab).
2. Optimize segmentation with the tabs "Characters" (Method: Flexible / Method: Fast) and the tab "Segmentation" (Method Flexible) or "Threshold" (Methode Fast).
   Each segment is displayed in a different color.
   Result: all desired characters must be cleanly segmented.
   Note: Which symbol was assigned to each segmented character (result of the classification) is not yet important here.
3. Check correct segmentation of all characters before classification.
   Note: Classification has no influence on segmentation. **Incorrectly segmented** characters are **classified incorrectly**. If the segmentation is unstable despite correct settings, return to step 1 (optimizing the original image).

Examples: Segmentation

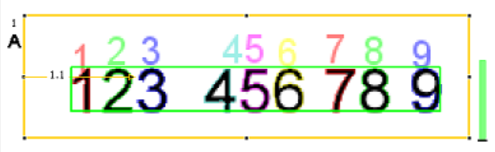


***Fig. 228:** Segmentation without default for parameter "Grouping of characters":*

All characters are found.

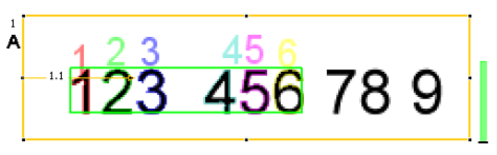


***Fig. 229:** Segmentation with default for parameter "Grouping of characters" "3 3":*

Only the two groups of three are segmented.

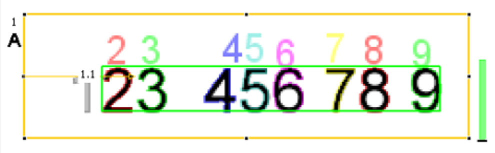


***Fig. 230:** Segmentation without default for parameter "Grouping of characters":*

The first character "1" is not segmented because its brightness differs too much from the other characters.

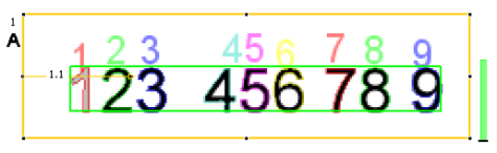


***Fig. 231:** Segmentation with default for parameter "Grouping of characters" "3 3 2 1":*

The character that differs in brightness is also segmented.

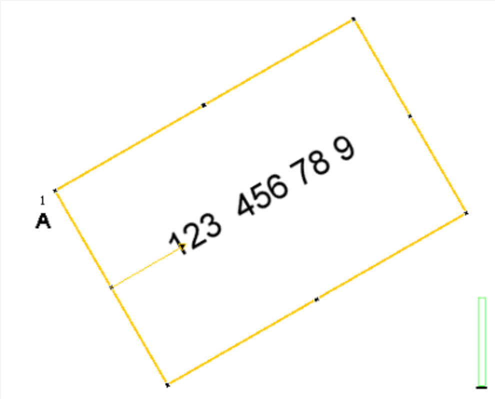


Fig. 232: *Segmentation with parameter "Text alignment" = "Font horizontal in the image":*

No segmentation because characters do not lie horizontally in the image.

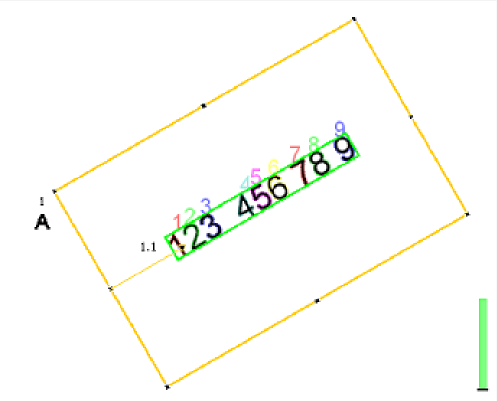


Fig. 233: *Segmentation with parameter "Text alignment" = "Font horizontal in region of interest":*

Segmentation works because characters lie horizontally in the search **area**.

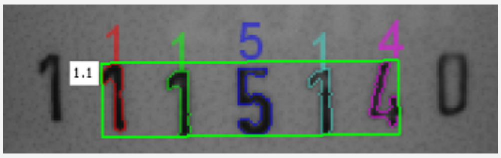
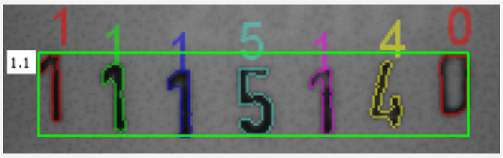


Fig. 234: *Segmentation with value 15% for parameter "Max. deviation baseline":*

Only the inner characters are segmented.



Segmentation with value 25% for parameter "Max. deviation baseline":
All characters are segmented.

Step 3: Classification

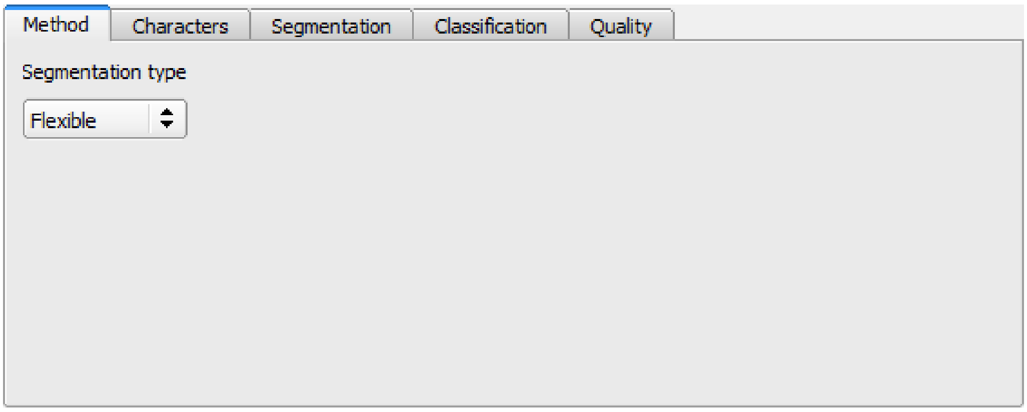
1. Tab "Classification": Select a suitable font ("Character set").
2. Each character set is offered with a different number of characters (e.g. numbers, capital letters, special characters).
   Select the character set that best suits the application.
   Note: The larger the character set, the greater the probability of misreading - therefore use the smallest possible character set!
3. Defining Reference string, adding regular expressions.
   The Reference string has two functions:
   - **Influencing the classification** using the quality value (reliability)
   - **Influence on the detector** result based on the specified minimum quality for the entire character string (threshold).

Step 4: Quality

- If the reliability of one of the classified characters is below the threshold (minimum reliability), the detector result becomes negative.
- Low reliability indicates that the character was not classified safely. High reliability, however, is no guarantee for a safe classification!

8.3.15.2 Method tab

In the Method tab the type of segmentation is set.



Segmentation type: Flexible

Fig. 235: *OCR Detector, Method tab*

NOTE:
The available tabs differ depending on the selected segmentation method (Flexible / Fast).

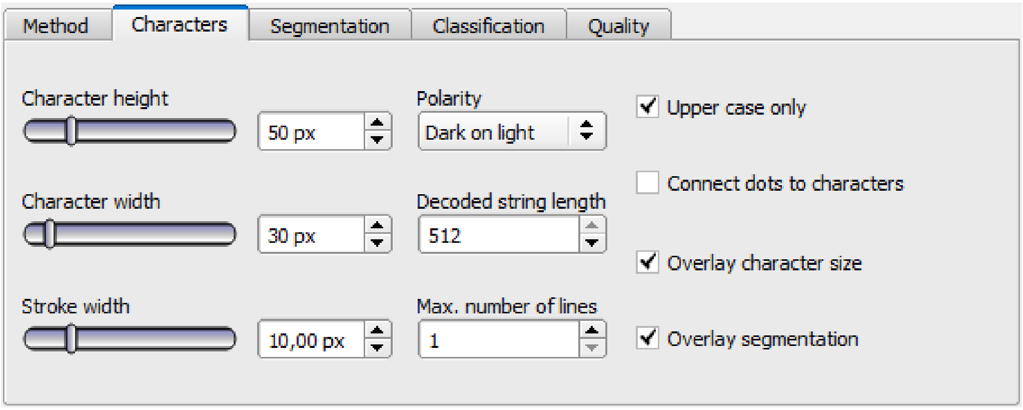
Parameter	Function
Segmentation type: Flexible	Segmentation searches for groups of characters that are close to each other
Segmentation type: Fast	Segmentation takes place via binarization thresholds to separate the characters from the background (BLOB analysis principle)

Comparison of the segmentation methods "Flexible" and "Fast"

Segmentation type: Flexible	Segmentation type: Fast
• At least 3 characters required	• Works already from 1 character
• Background noise and inhomogeneous characters can interfere with segmentation and reading quality	• Background noise can be filtered out by setting parameters
• Also suitable for low-contrast fonts with changing illumination conditions or demanding dot fonts	• Only conditionally suitable for low-contrast fonts or in changing illumination conditions
• Covers a wide range of variations between font and background	• Depending on the application approx. factor 2-8 faster than "flexible"
• Better performance under simple conditions (e.g. printed labels with clearly defined characters)	• Better performance under difficult conditions
• Only a few parameters need to be adjusted - well suited for users with little experience in image processing	• Requires knowledge of image processing

8.3.15.3 Characters tab (Method: flexible)

In the Characters tab the basic settings of the characters to be read are defined.



Character height 50 px

Character width 30 px

Stroke width 10,00 px

Polarity Dark on light

Decoded string length 512

Max. number of lines 1

☑ Upper case only

☐ Connect dots to characters

☑ Overlay character size

☑ Overlay segmentation

Fig. 236: *Detector OCR, tab Characters*

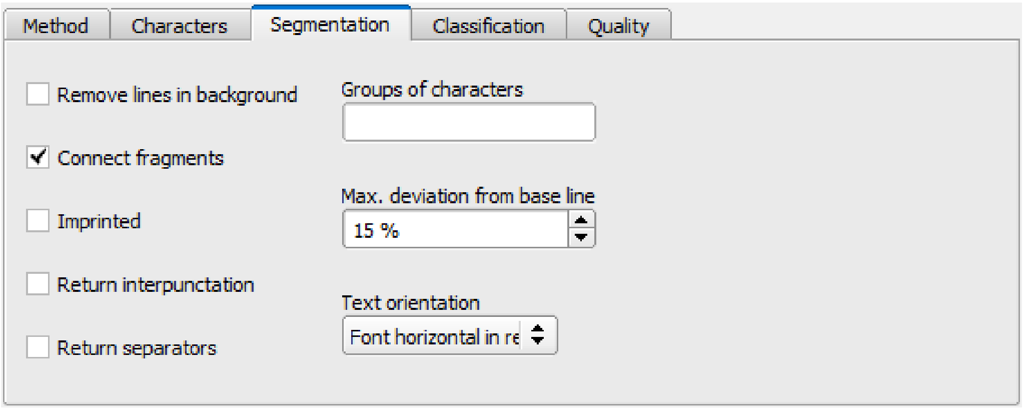
Parameter description:

Parameter	Function
Character height	Max. height of a character in pixels.
Character width	Max. width of character in pixels.
Stroke width	Average width of lines of characters in pixels.
Polarity	Possibility to select between dark characters on bright background or vice versa.
Max. string length	Maximum allowed length of the character string to be read. Characters that exceed this value are not read.
Max. number of lines	Max. number of lines to be read
Upper case only	Limitation: read capital letters only
Connect dots to characters	Connects single dots, e.g. of dot print or a blurry printed font to complete characters Check the influence of punctuation and separators like "." or "," or "-".
Overlay character size	Turns the rectangle (green drawing) for marking the letter size on / off
Overlay segmentation	Turns the color marking of the segmented characters on / off

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.15.4 Segmentation tab (Method: flexible)

In the Segmentation tab the basic settings of the characters to be read are defined.



☐ Remove lines in background

☑ Connect fragments

☐ Imprinted

☐ Return interpunctation

☐ Return separators

Groups of characters

Max. deviation from base line 15 %

Text orientation Font horizontal in re

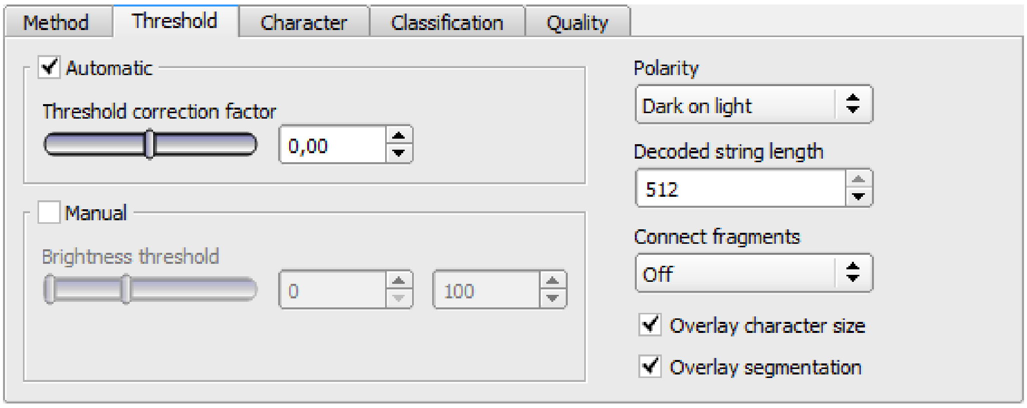
Fig. 237: *OCR Detector, Segmentation tab*

Parameter description:

Parameter	Function
Suppress background lines	Can be used to hide disturbing lines in the background.
Connect fragments	Connects characters that are e.g. fragmented in two parts due to unclean pressure or damage.
Raised / imprinted font	Enables the reading of raised or imprinted writing, e.g. if, due to lighting, the characters e.g. appear in white with a black border (shadow) or vice versa.
Return punctuation	Activates output of special characters such as periods or commas.
Return separators	Activates output of special characters e.g. dashes.
Groups of characters	Allows to specify how the characters in the string to be read are grouped (number of characters per character group). If the characters are e.g. always printed in two groups of four, this can be specified by entering "4 4". This function should be used if different lengths of strings are read in the same picture for several evaluations.
Max. deviation from base line	Maximum permissible vertical shift of characters relative to the line (the straight line between the first and last character); specified as a percentage of the character height. This function can be used when the characters are not printed on a horizontal line.
Text orientation	"Font horizontal in image": the font must always be horizontal in the image. Rotated text will not be read or read wrong. "Font horizontal in region": The rotation angle of the region of interest can be used to specify the rotation of the font with respect to the horizontal.

8.3.15.5 Threshold tab (Method: fast)

In the Threshold tab the basic settings of the characters to be read are defined.



☑ Automatic

Threshold correction factor: 0,00

☐ Manual

Brightness threshold: 0 100

Polarity: Dark on light

Decoded string length: 512

Connect fragments: Off

☑ Overlay character size

☑ Overlay segmentation

Fig. 238: *OCR detector, Threshold tab*

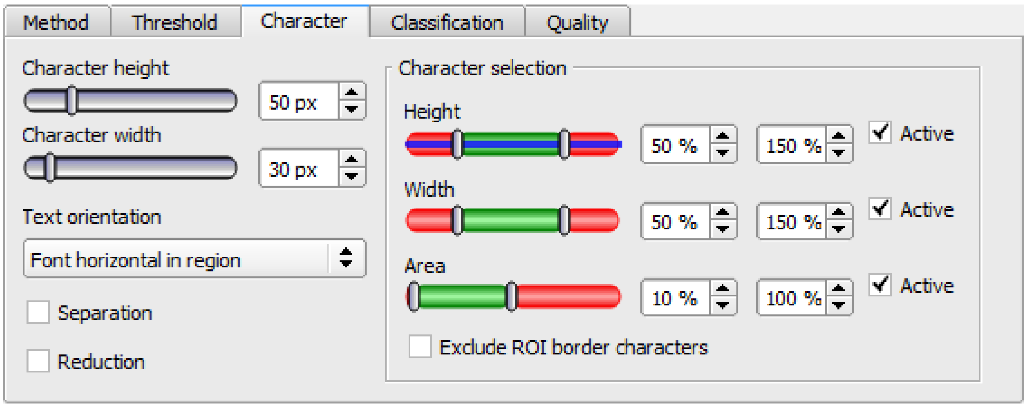
Parameter description:

Parameter	Function
Automatic	Enable automatic binarization (separation of the character from the background). The binarization threshold is automatically calculated in each image from the brightness distribution of foreground and background (in the yellow region of interest). Ensure homogeneous brightness in both the character and the background. Changes from image to image are possible, but not changes in brightness in the structure of the characters.
Correction factor for threshold	The binarization threshold can be selected lighter or darker. It can be moved manually in the direction of the brightness of the character or of the background. Positive value = shift towards brighter regions of the region of interest
Manual	Activate manual binarization
Brightness threshold	The binarization threshold can be manually set to a fixed value. Use several test parts (>50) to ensure homogeneous brightness in both the character and the background.
Invert button 🟥🟩🟥 / 🟩🟥🟩	The Invert button can be used to include or exclude the relevant range between the thresholds in the selection. Default: red/green/red = included.
Polarity	Possibility to select between dark characters on bright background or vice versa.
Decoded string length	Maximum allowed length of the character string to be read. Characters that exceed this value are not read.
Connect fragments	Connect fragmented characters from several fragments. Selection: "Off" / "1" / ... / "20" [px]. Number of nearest neighboring pixels (by radius): If another segment is found within these pixels, they are combined into one character.
Overlay character size	Turns the rectangle (green drawing) for marking the letter size on / off
Overlay segmentation	Turns the color marking of the segmented characters on / off

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.15.6 Characters tab (Method: fast)

In the Characters tab the basic settings of the characters to be read are defined.



Character height: 50 px

Character width: 30 px

Text orientation: Font horizontal in region

□ Separation

□ Reduction

Character selection

Height: 50 % – 150 % ☑ Active

Width: 50 % – 150 % ☑ Active

Area: 10 % – 100 % ☑ Active

□ Exclude ROI border characters

Fig. 239: *OCR Detector, Characters tab*

Parameter description:

Parameter	Function
Character height	Typical vertical height of a character in pixels. This setting is used in conjunction with the "Character selection" setting. Consider the height of upper and lower case letters and symbols (such as "-" or "*").
Character width	Typical horizontal width of a character in pixels. This setting is used in conjunction with the "Character selection" setting. Consider the width of different characters (for example, "I" or "W") in the font you choose.
Text orientation	• "Font horizontal in image": the font must always be horizontal in the image. Rotated text will not be read or read incorrectly. • "Font horizontal in region": The rotation angle of the yellow search region can be used to specify the rotation of the font with respect to the horizontal.
Separation	Forces the separation of horizontally connected characters that deviate significantly from the defined character width.
Reduction	Reduces segmented (grown) areas that have been enlarged by the "Join Fragments" function (Threshold tab) to approximately their original size.
Character selection (check "Active" to activate the setting)	
Height	Specification of tolerances for the defined character height (50% to 150%).
Width	Specification of tolerances to the entered character width (50% to 150%).
Area	Specifying tolerances for the drawing area resulting from the settings (10% to 100%).
No characters on border region of interest	Discard characters that extend beyond the boundary of the region of interest.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Example: Evaluation of the type face

Say you want the following type face to be read:

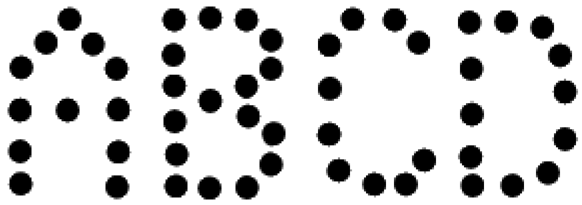
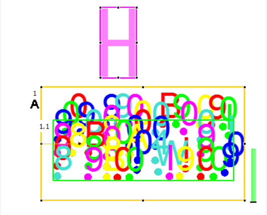
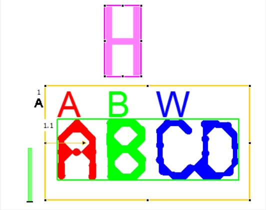
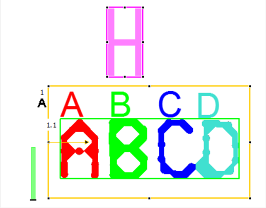


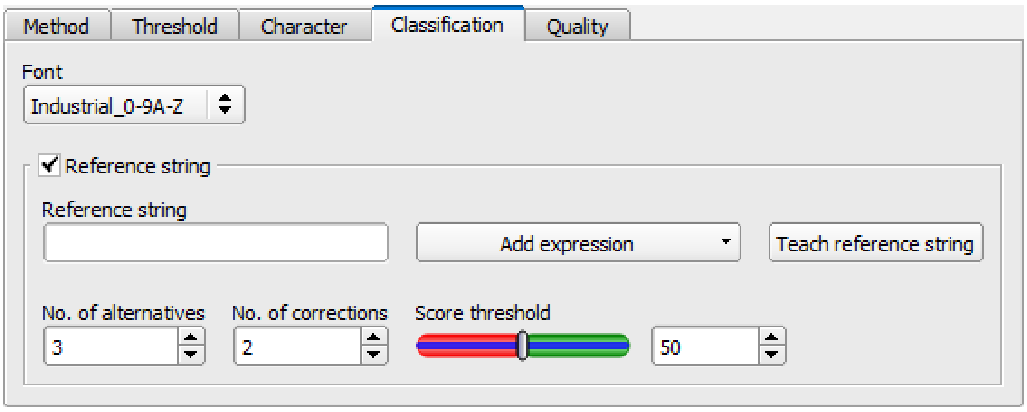
Fig. 240: *Type face*

The following table describes the individual steps used to evaluate the type face.

Step 1	Step 2	Step 3
If you place a OCR Detector over the text image and select the "Fast" mode for segmentation in the "Method" tab, an evaluation will be run for each fragment of a letter.	In order for the individual fragments to be put together in order to form letters / characters, you need to select the right value for the "Connect fragments" in the "Threshold" tab. In this example, the "Connect fragments" parameter has been set to 14.	The letters "C" and "D" are still being recognized as the letter "W". The configured character width (pink "H"), however, corresponds to the character width of a letter. To separate the characters, the "Separation" parameter needs to be enabled in the "Characters" tab.
 ***Fig. 241:*** *"Fast" segmentation mode*	 ***Fig. 242:*** *Connect fragments*	 ***Fig. 243:*** *Separation*

8.3.15.7 Classification tab

In the Classification tab the basic settings of the characters to be read are defined.



Font: Industrial_0-9A-Z

☑ Reference string

Reference string:

No. of alternatives: 3

No. of corrections: 2

Score threshold: 50

Fig. 244: *Detector OCR, tab Classification*

Parameter description:

Parameter	Function	
Character set	Available fonts. Additional information: OCR - available fonts	
	0-9	numeric characters only
	0-9+	Numbers and special characters
	A-Z+	Capital letters and special characters
	0-9A-Z	Numbers and capital letters
	no extension	All characters (upper and lower case letters, digits and special characters)
Reference string (Checkbox)	Activates verification of the contents of the read information. The content of the read string is checked using regular expressions.	

Parameter	Function
Reference string	This text or regular expressions is taken for verification. This is where specific characters can be found that are directly compared or regular expressions to check the structure of the read result.Characters which look very similar as number or as letter, e.g. "8" and "B" can be corrected automatically by using regular expressions in the reference string. Additional information: see Reference string: Details.
Add expression	Opens a list with examples for regular expressions.
Teach reference string	Reads the code that is currently under the code reader and enters the read content as a comparison text. This text can still be edited manually afterwards or it can be configured via a digital input or via the telegrams SPT / SPP.
Number of alternatives	Specifies how many possible alternatives may be used to find and automatically replace a character according to the regular expression in the reference string.
Number of corrections	Max. number of characters that may be changed after being checked by the regular expression. Example: The segmentation is day, three digits (MON / TUE / WED / etc). The decoding outputs the letters "W6O" instead of "WED". With a setting of '2' in this field, the camera software will automatically 'correct' the (number) 6 and (letter) O to become (letter) E and D. If the setting in this field was 1, then the detector would fail.
Score threshold	Threshold for good/bad decision: If, based on the set threshold, the number of corrections is too high, the entire text is rated as "not read".

Reference string: Details

The Reference string has two functions:

1. Affecting the classification, i.e. of the recognized characters. For each segmented character, a quality value (reliability) is determined in relation to each character contained in the character set.
If no reference string is specified, the character with the highest quality value (reliability) is output as the "correct" character.
If a reference string is specified, on the other hand, the *'n'* best alternatives are considered (Number of alternatives). For the reference string, each character may be selected a maximum of 'm' times (Number of corrections) that did not have the maximum reliability.

Example: A string "BZ156" is marked and read without Reference string. Due to a bad quality of the marking, the VISOR® reads the string as "8ZI5G". Using "Number of corrections" and "Number of alternatives" it would be possible to influence the reading to correct itself to "BZ156".

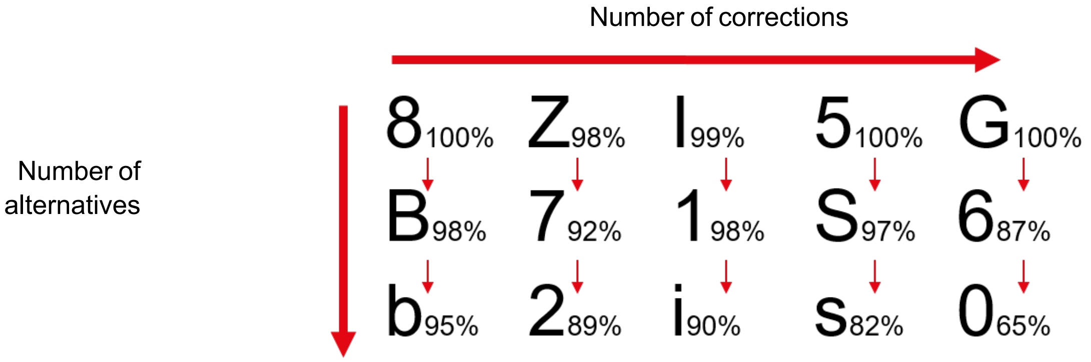


Fig. 245: *Reference string operating mode*

2. Influencing the detector result:
A minimum quality for the entire character string (threshold) is specified. If this value is undershot, the detector result is negative.

Examples of reference character strings defined by regular expressions

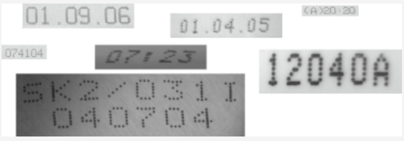
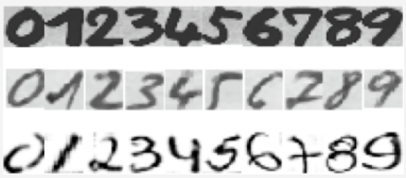
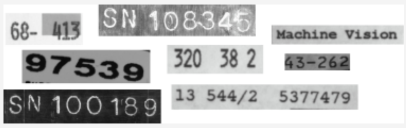
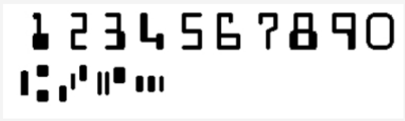
Reference character string	Hit	Example for hit
123	String containing 123	01234
\A123	String beginning with 123	1234
123\Z	String ending with 123	0123
\A123\Z	String matching 123 exactly	123
[123]	Character string containing one of the characters	33
[123]{2}	String containing a sequence of 2 of the characters	23
[12]|[34]	String containing a character of one of both groups (OR function)	4

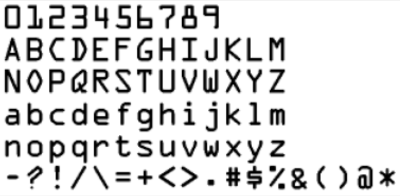
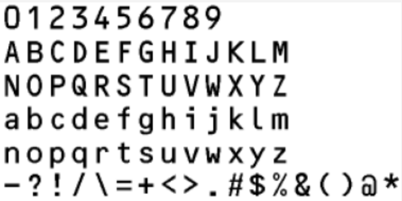
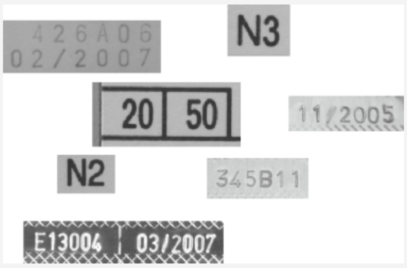
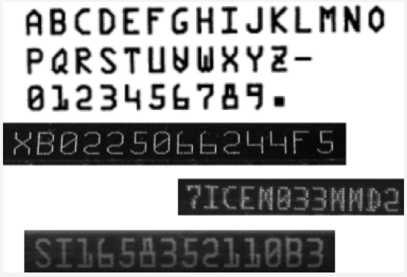
The most important elements of regular expressions:

^ or \A	Represents the beginning of the character string
$ or \Z	Represents the end of the character string, and possibly includes a newline as the last character
.	Represents every character apart from newline
[...]	Represents any literal listed in the square brackets. If the first character is an '^', the expression is negated. You can use the '-' character, as in '[A-Z0-9]', to specify value ranges. Other characters lose their special meaning within square brackets, except '\'.
*	Allows 0 or more repetitions of the preceding literal / group
+	Allows 1 or more repetitions
?	Allows 0 or 1 repetitions
{n,m}	Allows n to m repetitions
{n}	Allows exactly n repetitions
|	Separates alternative search expressions (OR function)

8.3.15.7.1 OCR - available fonts

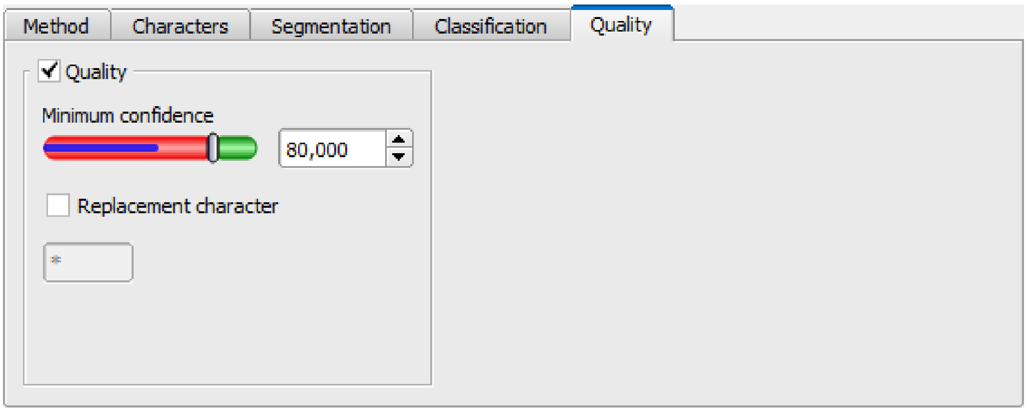
Examples of available fonts:

Dot print	Handwritten
	
Industrial	**MICR**
 68-413 SN 108345 Machine Vision 97539 320 38 2 43-262 SN 100189 13 544/2 5377479	

OCRA	OCRB
 0123456789 ABCDEFGHIJKLM NOPQRSTUVWXYZ abcdefghijklm nopqrtsuvwxyz -?!/\=+<>.#$%&()@*	 0123456789 ABCDEFGHIJKLM NOPQRSTUVWXYZ abcdefghijklm nopqrtsuvwxyz -?!/\=+<>.#$%&()@*
Pharma	**Semi**
 426A06 02/2007 N3 20 50 11/2005 N2 345B11 E13004 03/2007	 ABCDEFGHIJKLMNO PQRSTUVWXYZ- 0123456789. XB0225066244F5 7ICEM033MMD2 SI16583521I0B3

8.3.15.8 Quality tab

Definition of basic settings of characters to read.



☑ Quality

Minimum confidence: 80,000

□ Replacement character

*

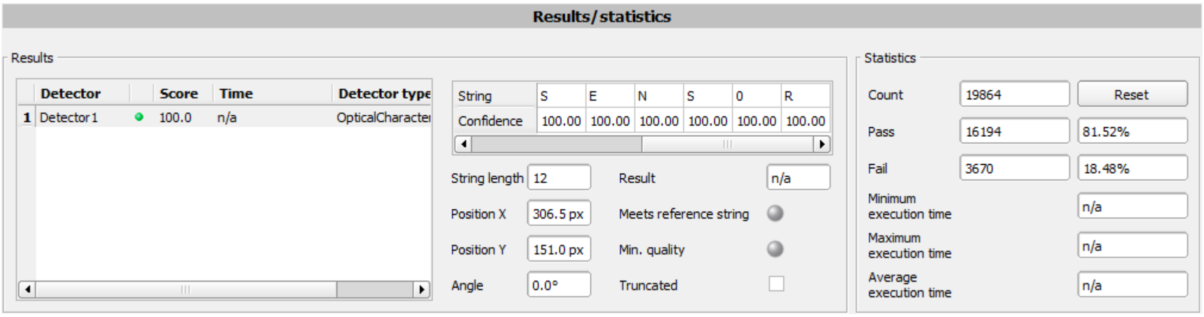
Fig. 246: Detector OCR, tab Quality

Parameter description:

Parameter	Function
Quality	Quality of each character gets a value of 0-100%. The higher the value, the safer the read character was determined. Small values indicate a rather uncertain assignment.
Minimum reliability	If the reliability is below the set threshold, the character is rated unread and replaced with a placeholder.
Replacement character	Placeholder in case the minimum reliability was not reached.

8.3.15.9 OCR Result

In the Result tab (next to the help tab or, in run mode, at the bottom of the screen) evaluation results are displayed.



Results/statistics

Results

	Detector	Score	Time	Detector type
1	Detector 1	● 100.0	n/a	OpticalCharacte

String	S	E	N	S	0	R
Confidence	100.00	100.00	100.00	100.00	100.00	100.00

String length 12
Result n/a
Position X 306.5 px
Meets reference string ●
Position Y 151.0 px
Min. quality ●
Angle 0.0°
Truncated □

Statistics

Count	19864	Reset
Pass	16194	81.52%
Fail	3670	18.48%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Fig. 247: *Detector OCR, Result display*

Parameter description:

Parameter	Function
Score	Detector result: 0% (NOK) or 100% (OK)
Text	Characters read
Security	Value from 0-100%, indicates how reliably the detector could evaluate a character.
String length	Length of string read
Position X	X position of the read string in pixels
Position Y	Y position of the read string in pixels
angle	Angle to the horizontal line
Result	Indication for the quality of a result. If no characters had to be replaced according to the reference string, this value is at 100%. The value decreases with rising number of corrections.
Result comparison	Indicates if the output string meets the reference string.
Min. Quality	Indicates if minimum reliability was reached.
Truncated	Indicates if a part of the string was truncated.

8.3.16 Detector Color Value

This detector determines mean color values RGB / HSV / LAB for output via the interfaces.

8.3.16.1 Color Channel tab

The Color Channel tab is used to select the Color models/color channel(s) on which the detector is to operate.

An image recorded with a color chip contains more information than a monochrome image due to the color component. This feature can be used with the color channel selection. By selecting the color channels, specific areas can be intensified or weakend. The display of the image depends on the image chip and the selected detector.

- Monochrome chip: Display always black/ white
- Color chip + Color detector: Display always colored
- Color chip + Object detector: Monochrome image, display depending on selected color model and color channel

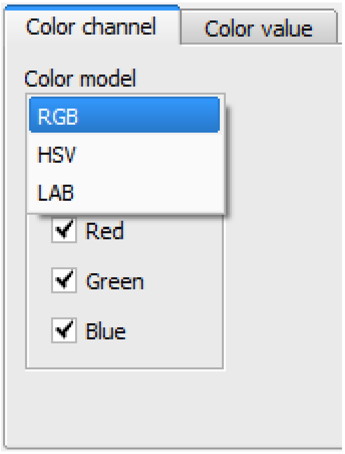


Fig. 248: Color Channel

Parameter	Function
Color space	Color spaces: RGB, Color model RGB, HSV, Color model HSV, LAB, Color model LAB
Color channel	One or more channels can be selected.

8.3.16.2 Color Value tab

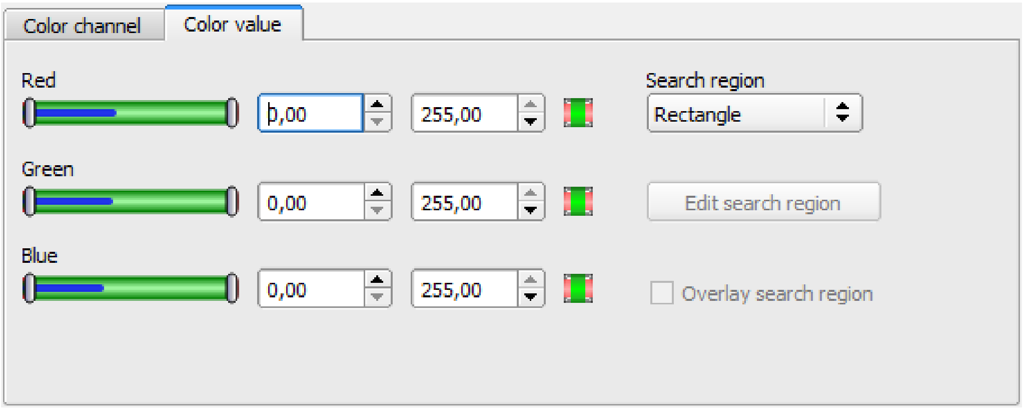


Fig. 249: Color Value tab

Parameter description:

Parameter*	Function
*differs depending on the setting in color channel	
Red (Hue / Luminance)	Threshold for selected channel min./ max.
Green (Saturation / A)	Threshold for selected channel min./ max.
Blue (Value / B)	Threshold for selected channel min./ max.
Region of interest (shape)	The shape of the region of interest can be set as Rectangle, Circle, or Free shape. If Freeform is selected, "Edit region of interest" is active.

Parameter*	Function
Edit region of interest	The parameter "Edit region of interest" can be used to hide areas of the region of interest. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. These marked areas can also be inverted, i.e. the areas that are important for the execution, etc. are marked.
Display region of interest	Enable / disable the display of region of interest edits

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Predestined applications:

- Output of calculated color parameters via one of the data interfaces for further processing.

8.3.17 Detector Color Area

This detector determines the area covered by a color or a color range. Depending on the proportion of the area, a good / bad result can be produced.

8.3.17.1 Color Channel tab

For VISOR® Color: See Chapter: Color Channel tab

8.3.17.2 Color Area tab

Determines the area covered by a color or color range. Depending on the proportion of the area, a good / bad result can be produced.

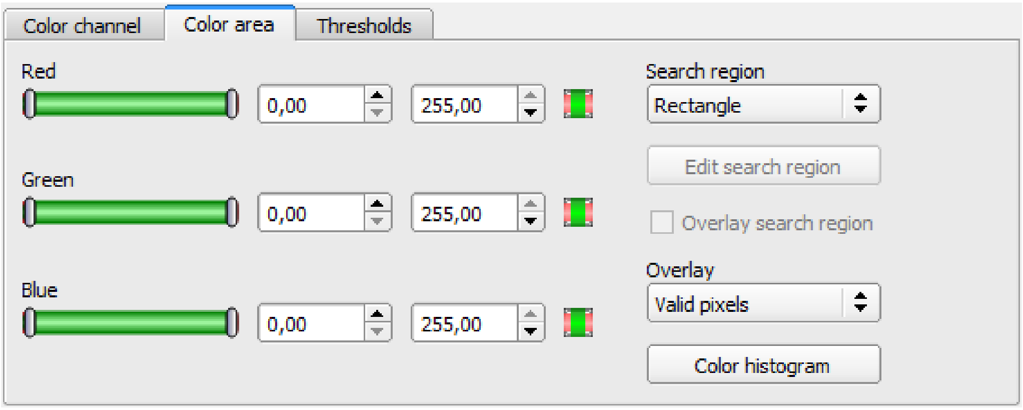


Fig. 250: *Color Area*

Parameter description:

Parameter*	Function
*differs depending on the setting in color channel	
Red (Hue / Luminance)	Threshold for selected channel min./ max.
Green (Saturation / A)	Threshold for selected channel min./ max.
Blue (Value / B)	Threshold for selected channel min./ max.
Region of interest (shape)	The shape of the region of interest can be set as Rectangle, Circle, or Free shape. If Freeform is selected, "Edit region of interest" is active.

Parameter*	Function
Edit region of interest	With the parameter "Edit region of interest", areas of the region of interest can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The marked areas can also be inverted. This marks the areas that are important for the execution.
Display region of interest	Enable / disable the display of region of interest edits
Overlay	Color marking of pixels inside or outside of specified color range. This is an aid during setup to visualize detector results and to set thresholds more accurately.
Color histogram	Allows graphical adjustment of the thresholds using a histogram

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Predestined applications:

- Colored object with certain size and variable position in the ROI

8.3.17.2.1 Color histogram

Depending on the selected color model, the histograms for RGB, HSV, or LAB are displayed.
The histogram shows the distribution of colors in the region of interest. Using the buttons, single channels can be switched on or off. Small markings below the histogram can be used to move the color detection limits. The marked area is highlighted in the corresponding color. Crossing the limits results in inversion of the selection. If a color can be reliably detected with only one channel, the limit values of the other channels must be set to the lower or upper end value so that they do not interfere with detection.

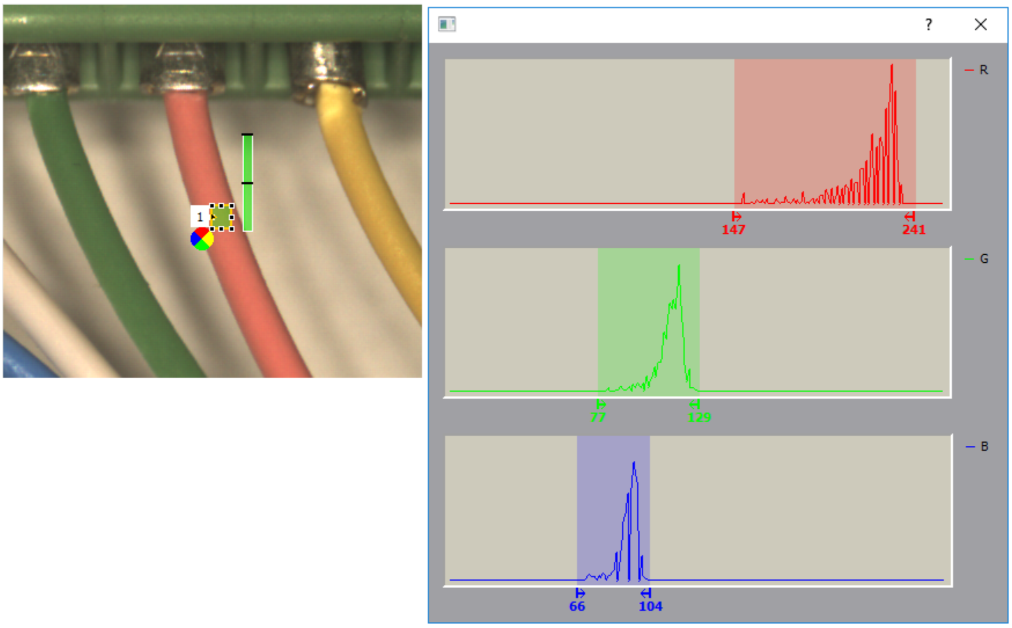
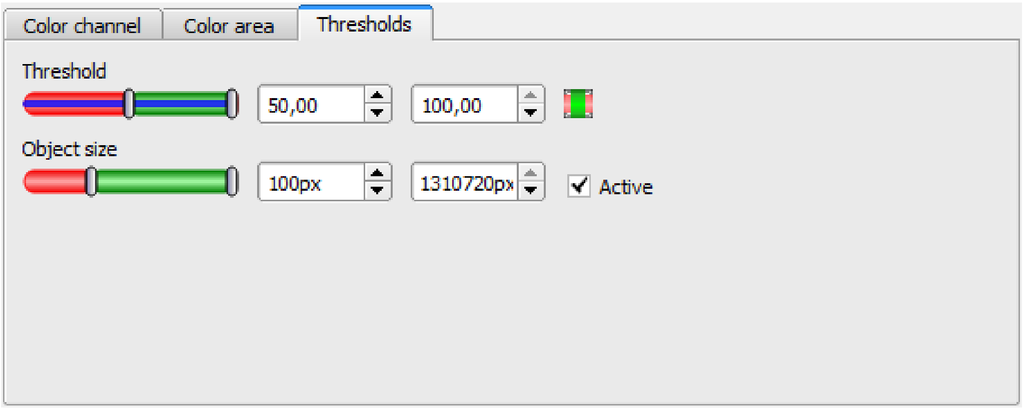


Fig. 251: *Color histogram*

8.3.17.3 Thresholds tab

Determines the area covered by a color or color range. Setting of thresholds.



Threshold 50,00 100,00

Object size 100px 1310720px ☑ Active

Fig. 252: Color Area, Thresholds

Parameter description:

Parameter	Function
Threshold	Threshold for percentage of the area min./ max.
Object size	Min. / Max. object size (connected area)

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

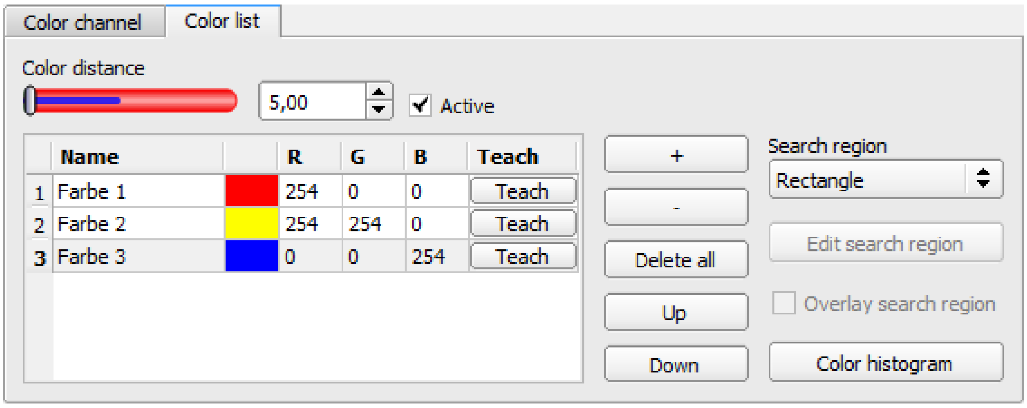
8.3.18 Detector Color List

This detector compares a color with a list of known colors. Result: Number or name of the closest color. This enables sorting of parts by color.

8.3.18.1 Color Channel tab

For VISOR® Color: See Chapter: Color Channel tab

8.3.18.2 Color List tab



Color distance 5,00 ☑ Active

	Name		R	G	B	Teach
1	Farbe 1	🟥	254	0	0	Teach
2	Farbe 2	🟨	254	254	0	Teach
3	Farbe 3	🟦	0	0	254	Teach

Search region: Rectangle

□ Overlay search region

Fig. 253: Color List

Parameter description:

Parameter	Function
Color distance	Distance of the current color versus the taught-in color. The metric of the color distance depends on the Color models used; only the selected color channels are considered. *1)
Name	Name of the color, can be changed by double-clicking on the name, e.g. red, yellow, blue
Pattern color	Representation of the taught-in color as a pattern and in numerical values. Differs depending on the setting in the color channel (RGB / HSV / LAB)
Teach-in	If you learn the color or the color range in the region of interest, and if several different colors are to be taught-in, a small region of interest must be moved to the color to be taught in each case.
+	New line at end of the table
-	Delete selected line
Delete all	Delete all entries in the list
Up	Move marked line one line down
Down	Move marked line one line down
Region of interest (shape)	The shape of the region of interest can be set as Rectangle, Circle, or Free shape. If Freeform is selected, "Edit region of interest" is active.
Edit region of interest	With the parameter "Edit region of interest", areas of the region of interest can be hidden. As with an eraser, the areas that are not needed for the evaluation can be removed in the region of interest. The marked areas can also be inverted. This marks the areas that are important for the execution.
Display region of interest	Enable / disable the display of region of interest edits
Color histogram	Allows graphical adjustment of the thresholds using a histogram

1*) In the RGB and LAB color model, the color distance is the Euclidean distance.
In the color model LAB, the color distribution over the entire space is nearly homogeneous, i.e. color differences of the same amount lead to a very similar perception of the color difference over the entire model. Therefore in this model, it can be said that a distance of ≥ 5 leads to the perception of another color.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

Predestined applications:

- Sorting of colored object via the list index
- Simple testing of homogeneous color areas (color is averaged over region of interest, learn color, set small color distance (tolerance band) ... done)

8.3.18.2.1 Color histogram

Depending on the selected color model, the histograms for RGB, HSV, or LAB are displayed.
The histogram shows the distribution of colors in the region of interest. Using the buttons, single channels can be switched on or off. Small markings below the histogram can be used to move the color detection limits. The marked area is highlighted in the corresponding color. Crossing the limits results in inversion of the selection. If a color can be reliably detected with only one channel, the limit values of the other channels must be set to the lower or upper end value so that they do not interfere with detection.

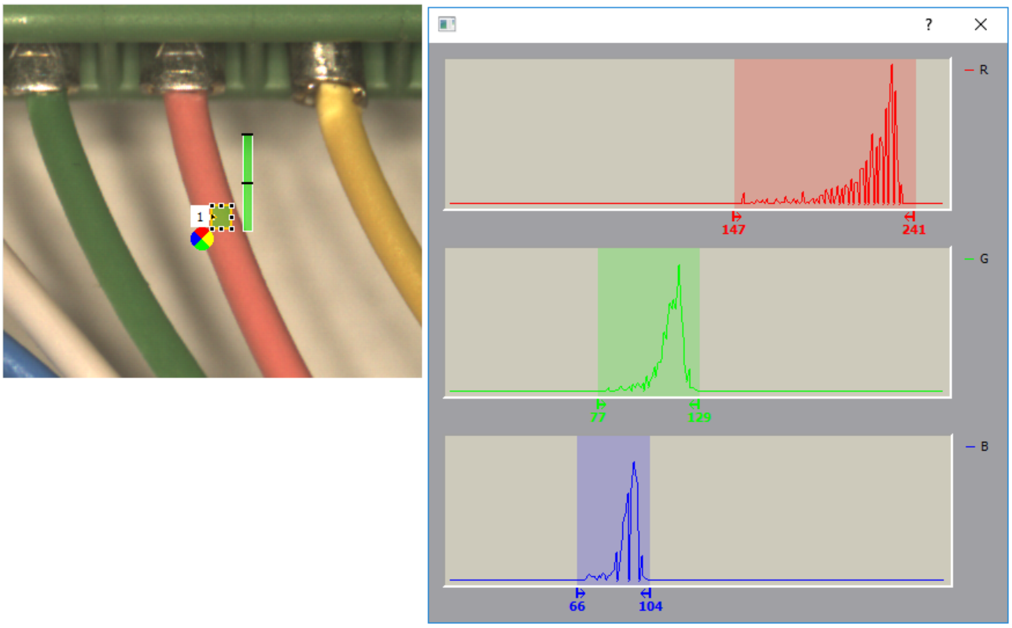


Fig. 254: Color histogram

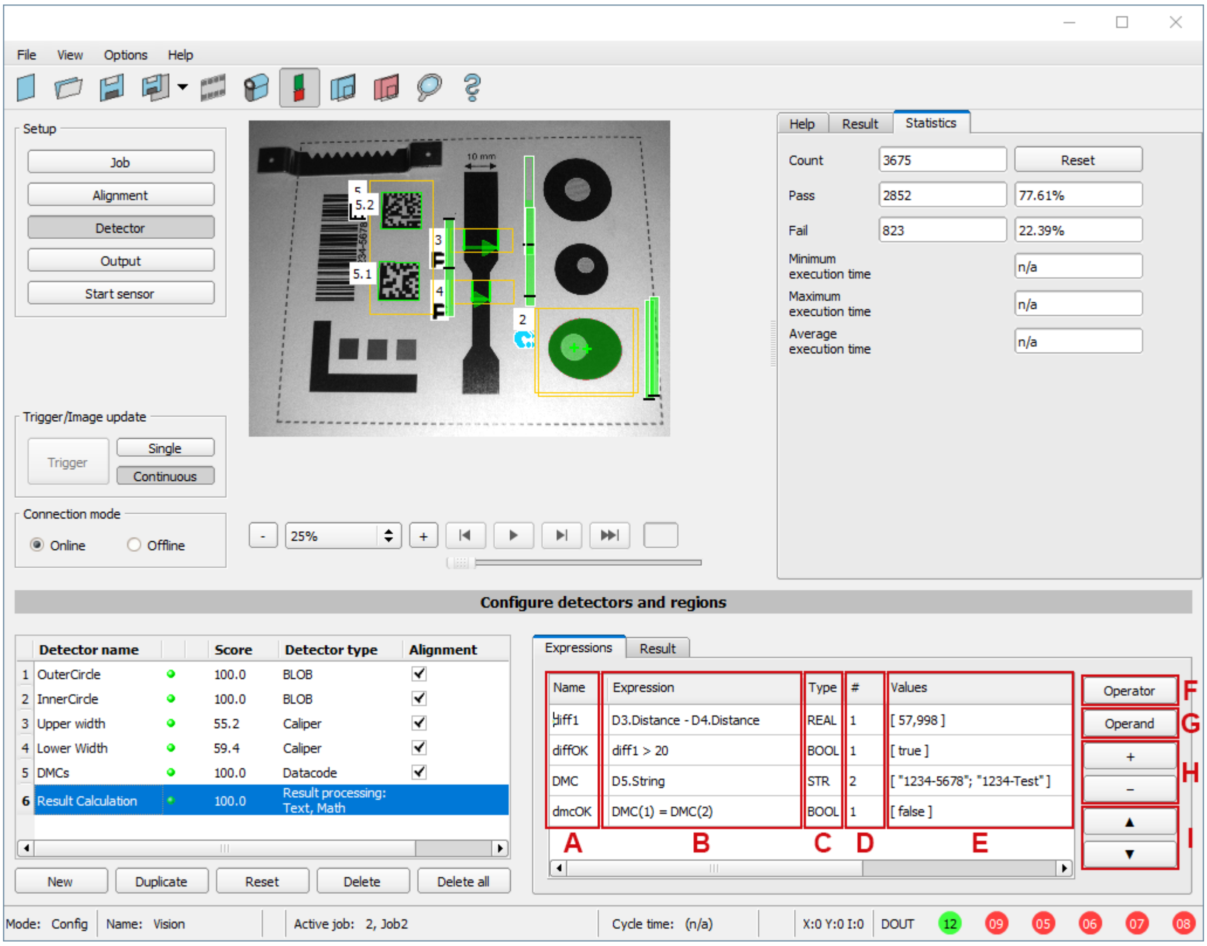
8.3.19 Detector Result processing: Text, numbers, robotics

This detector allows to calculate and evaluate detector results from previously executed detectors, and to compare, sort, format, add and crop strings. This means that the calculation can be performed directly on the VISOR® and the correct functioning does not have to be distributed among different systems (e.g. on the PLC).

In addition to simple arithmetic operations, more complex operations can also be performed, such as sorting output vectors, calculating distances and angles, or logical operations.

The score value of the detector is 100 if **all** expressions are valid. Otherwise, the score value is 0.

8.3.19.1 Expressions tab



Count: 3675
Pass: 2852 | 77.61%
Fail: 823 | 22.39%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

	Detector name	Score	Detector type	Alignment
1	OuterCircle	100.0	BLOB	☑
2	InnerCircle	100.0	BLOB	☑
3	Upper width	55.2	Caliper	☑
4	Lower Width	59.4	Caliper	☑
5	DMCs	100.0	Datacode	☑
6	Result Calculation	100.0	Result processing: Text, Math	

Name	Expression	Type	#	Values
diff1	D3.Distance - D4.Distance	REAL	1	[57,998]
diffOK	diff1 > 20	BOOL	1	[true]
DMC	D5.String	STR	2	["1234-5678"; "1234-Test"]
dmcOK	DMC(1) = DMC(2)	BOOL	1	[false]

Mode: Config | Name: Vision | Active job: 2, Job2 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 255: Result processing detector, Expressions tab

The columns and buttons of the detector are described below. The expression in column **B** is evaluated and its result is written into the variable (column **A**). This variable can be accessed in the setup step Output / Telegram.

Parameter description: Columns and buttons in the Expressions tab

Column	Description
A: Name	Name of the expression (changeable), used as a variable. Numbers 0-9, letters A-Z, underscores are allowed, no special characters.
B: Expression	Consists of operators and operands, can be typed in manually or inserted using the buttons
C: Type	Data types: BOOL (Boolean number), INT (Integer, value range: +- 2.147.483.647), REAL (Floating-point number, value range: +- 2.147.483), STR (Text).
D: #	Number of components of the result, e.g. for vectors.
E: Values	Result of the expression and possible errors that may occur (e.g. "Invalid expression name", "Unexpected symbol", "Undefined variable").

Buttons	Description
F: Operators	Operators: divided into groups. Additional information: Chapter Operators.

Buttons	Description
G: Operands	Variables, constants as well as access to detector results (by detector number) Additional information: Chapter Operands.
H: + / -	"+" adds a new line at the end, "-" deletes the currently selected line.
I: ▲ / ▼	▲ / ▼ moves the current line one position up / down.

Syntax

Input	Function
D1.Score	Access e.g. to score value of detector no. 1 1
[ ]	Vectors or scalars, spaces possible
;	Separator for vectors
"xyz"	Strings
(n)	Access to the *n*-th component of a vector

Syntax highlighting

Type	Color
Operators / operands	Blue
Error / Undefined	Red
Constants / strings	Dark Magenta
Brackets ((),[]) and semicolon (;)	**Black**
Unclosed brackets	**Red**
Everything else	Black

Scalars and vectors

The input data and the results can be scalars (single values) or vectors with several components (fields).

Examples:

Data type	Scalar	Vector
REAL	[10,543]	[10,543; 2,000; 8,500]
INT	[23]	[23; 45; 6]
BOOL	[true]	[true; true; false]
STR	["Object"]	["Object"; "Code"]

Individual components of a vector can be accessed by specifying the index. The numbering of the indices begins with 1.

Example: **v1** = [11; 12; 13; 14]

Single component:
v1(2) = [12]

Multiple components:
v1(2; 4) = [12; 14]

Component range:
v1(2:4) = [12; 13; 14]

Component range to the end:
v1(2::) = [12; 13; 14]

Component ranges:
v1(1; 3:4) = [11; 13; 14]

Invalid index:
v1(5) = []

Handling of empty vectors:

In case of incorrect input or runtime errors, the "empty vector" [ ] result will be returned. The operator size() can be used to check how many elements a vector contains. Error cases can be checked by querying whether the case "empty vector" has occurred. See also Handling empty vectors.

Static variables

Static variables can be accessed globally, i.e. from any job. Furthermore, they are static, i.e. they persist beyond a trigger and are not overwritten like normal variables. Additional information: Static variables tab

8.3.19.1.1 Operators

The operators are diveded into the groups Algebra, Text, Logic, Geometry, Poses as well as Trigonometry, Rounding, Vector, Vector properties and Sorting. Within the groups of operators, there are "Advanced" groups containing more complex and less common operators. These subgroups can be expanded by clicking on the arrow (▼).

NOTE:
Basically, the Data types: BOOL (Boolean number), INT (Integer, value range: +- 2.147.483.647), REAL (Floating-point number, value range: +- 2.147.483), STR (Text) can be processed. If there are restrictions, these are listed in the respective group or operator.

Operators can be applied:

- to 2 scalars (e.g. 2 + 1 = [3])
- to 2 vectors: Operator acts component by component (e.g. ["a"; "b"] + ["x"; "y"] = ["ax"; "by"].
- to a vector and a scalar: Scalar affects every component of the vector. (e.g. [2; 3] + [1] = [3; 4])

The different operators are described below.

8.3.19.1.1.1 Algebra group

IN:/OUT: Data types INT / REAL

Operator name	Description	Example:
+	Addition of scalars or of vector components. Corresponding components are added.	[2;3;5]+[4;6;7] =[6;9;12]
	Addition of scalar to vector. In this case, the scalar is added to each component of the vector.	[2;3;5]+5 =[7;8;10]
-	Subtraction of scalars or of vector components.	[4;6;7]-[2;3;5] =[2;3;2]
	Subtraction of a scalar from a vector. In this case, the scalar is subtracted from each component of the vector.	[4;6;7]-2 =[2;4;5]
*	Multiplication of scalars or of vector components.	[4;6;7]*[2;3;5] =[8;18;35]
	Multiplication of scalars and vectors. In this case, the scalar is multiplied by each component of the vector.	[4;6;7]*2 =[8;12;14]
/	Division of scalars or of vector components.	[4;6;7]/[2;3;5] =[2;2;1.4]
	Division of vector by scalar. In this case, each component of the vector is divided by the scalar.	[4;6;7]/2 =[2;3;3.5]

Operator name	Description	Example:
sqr	Square the scalar or vector components.	sqr([2;3;5]) =[4;9;25]
sqrt	Square root of the scalar or vector components.	sqrt(9) =[3]
pow	Power of the scalar or vector components. **IN: base** (base), **expn** (exponent) For negative numbers, only integer exponents can be used.	pow (2;3) =[8] pow ([2;3;5];3) =[8;27;125]
log	Logarithm of the scalar or vector components with base 10.	log(100) =[2]
abs	Absolute value of the scalar or vector components	abs(-3.4) =[3.4]
min	Returns the smallest vector components (component by component)	min([1;5];[2;4]) =[1;4]
max	Returns the largest vector components (component by component)	max([1;5];[2;4]) =[2;5]
()	Access vector components via their indexes **IN:** Indexes of the desired vector components, Data type INT	[2;4;6;8;10](2:4) =[4;6;8] [2;4;6;8;10](4::) =[8;10]
div	Integer division with remainder (outputs integer result) **IN: x** (dividend), **y** (divisor), (Data type INT) **OUT:** Data type INT	div(5;2) =[2]
mod	Remainder of an integer division **IN: x** (dividend), **y** (divisor), (Data type INT) **OUT:** Data type INT	mod(5;2) =[1]

8.3.19.1.1.2 Text group

IN:/OUT: Data type STR

Operator name	Description	Example:
""	Create string variable	"Object" =["Object"]
+	Add string vectors	["abc-"; "def-"] + ["123";"456"] =["abc-123";"def-456"]
	Add a single string to a vector. In this case, the string will be added to each of the vector components.	["pick";"place"] + [" object"] =["pick object";"place object"]
str_sub	Extract substring. Range refers to the position according to UTF-8 symbols (not to bytes). **IN: v** (String from which a substring is to be extracted), **pos1** (start position, Data type INT), **pos2** (end position, optional, Data type INT) If **pos2** is not specified, the partial string is output to the end.	str_sub ("object";1;3) =["obj"] str_sub ("object";4) =["ect"]

Operator name	Description	Example:
str_insert	Insert a string at a predefined position **IN: v** (String into which a string is to be inserted), **pos1** (position at which the text is to be inserted, Data type INT), **str** (Text to be inserted, Data type STR)	str_insert ("xxzz";3;"yy") =["xxyyzz"]
str_delete	Delete part of a string **IN: v** (String of which a part is to be deleted), **pos** (position at which the text is to be deleted, Data type INT), **len** (number of characters to be deleted, Data type INT)	str_delete ("abcde";4;2) =["abc"]
str_replace	Replace parts of a text. **IN: v** (String whose text is to be replaced), **str1** (string to be replaced), **str2** (string to be inserted instead)	str_replace ("abcde";"abc";"x") =["xde"]
str_search	Search forward for string and return position of the first result. If string is not found or input string is empty, the result is -1. **IN: v** (string to be scanned) **w** (String to search for)	str_search ("xy-ab-xy";"xy") =[1]
str_length	String length: Number of characters (not number of bytes). **IN: v** (string whose length is to be output) **OUT:** Data type INT	str_length (["abcde"]) =[5]
str_length_byte	String length: Number of bytes for UTF-8 representation (not number of characters). **IN: v** (string whose length is to be output)	str_length_byte ("►") =[3]
str_correction	Perform text correction using the Reed-Solomon algorithm. Number of check digits: 2. **IN: code** (string to be checked) **OUT:** corrected string, Data type STR	str_correction ("0110UOL5MI5") =["0100UOL5MI5"]
str_correction_ errors	Perform text correction using the Reed-Solomon algorithm and output corrected position. Number of check digits: 2. **IN: code** (string to be checked) **OUT:** Index of the corrected position (Data type INT). If no correction is made, the result is an empty vector [ ].	str_correction_ errors ("0110UOL5MI5") =[3] str_correction_ errors ("0100UOL5MI5") =[ ]
to_upper	Convert all letters to capital letters **IN: v** (String)	to_upper ("Object") =["OBEJCT"]
to_lower	Convert all letters to lower case letters **IN: v** (String)	to_lower ("Object") =["object"]
to_string	Convert value / number of type BOOL, INT, REAL to string. **IN: v** (Data type BOOL /INT /REAL), optional: **width** (minimum width of the output string (right-justified), if necessary filled up with blanks, Data type INT), **precision** (accuracy of rounding to decimal places, Data type INT) **OUT:** Data type STR	to_string ([2.22; 9.99; 5.1]; 4; 1) =[" 2.2"; "10.0"; " 5.1"]
to_number	Convert string to number. If the string contains letters, the result is an empty vector [ ]. **IN: v** (String) **OUT:** Data type INT, REAL	to_number ("000.123") =[0.123]

Operator name	Description	Example:
vec_sum	Link (concatenate) components of the string vector **IN: v** (String vector, Data type STR), optional: separator (Data type STR)	vec_sum (["ab";"cd";"ef"];"_") = ["ab_cd_ef"]
str_split	Separate text at the position of the specified separator **IN:** v (string), separator (Data type STR)	str_split("ab-cd-ef"; "-") = ["ab";"cd";"ef"]

8.3.19.1.1.3 Logic group

Part 1

When comparing a vector with a scalar, the scalar is compared with each component of the vector.

IN: Data type BOOL / INT / REAL / STR

OUT: Data type BOOL

Operator name	Description	Example:
<	"Less than" operation of scalars or vector components	[2;5;4] < [2;4;5] =[false;false;true] ["A"]<["B"] = [true]
<=	"Less than or equal to" operation of scalars or vector components	[2;5;4] <= [2;4;5] =[true;false;true]
>	"Greater than" operation of scalars, or vector components	[2;5;4] > [2] =[false;true;true]
>=	"Greater than or equal to" operation of scalars or vector components	[2;5;4] >= [2] =[true;true;true]
=	"Is equal" operation of scalar or vector components	["OK";"NOK"] = ["OK"] =[true;false]
!=	"Is unequal" operation of scalar or vector components	["OK"] != ["NOK"] =[true]

Part 2

For the following operators: **IN:/OUT:** Data type BOOL
Example vectors: **v1**=[true;true] **v2**=[true;false] **v3**=[true]

Operator name	Description	Example:
&	AND operation of scalars or vector components	v1&v2 =[true;false]
	AND operation of scalar and vector. In this case, the scalar is combined with each component of the other vector.	v2&v3 =[true;false]
|	OR operation of scalars or vector components	v1|v2 =[true;true]
	OR operation of scalar and vector. In this case, the scalar is combined with each component of the other vector.	v2|v3 =[true;true]
!	Negation (NOT) of scalars or vector components	!v2 =[false;true]

Operator name	Description	Example:
if	Check a condition and issue a value accordingly **IN: b** (condition to be tested, Data type BOOL, must be scalar), **v1** (Then value), **v2** (Otherwise value), Data types BOOL / INT / REAL / STR. Output of **v1** if **b**=true, output of **v2** if **b**=false Note: Both **v1** (then value) and **v2** (o**th**erwise value) must be executable. Otherwise, the expression cannot be executed.	if(v3; "OK"; "NOK") =["OK"]

8.3.19.1.1.4 Geometry group

IN:/OUT: Data types INT / REAL

Overlays for geometry operators

For some Geometry operators there are graphical overlays in the image. The overlays are displayed for the expression that is currently active (if there are several Geometry operators in one line, only the first one is displayed).

These overlays are also only displayed if **no** calibration is active or a **2D** calibration (i.e. "Scaling (Measurement)", "Calibration plate (Measurement)" or "Calibration plate (Robotics)").

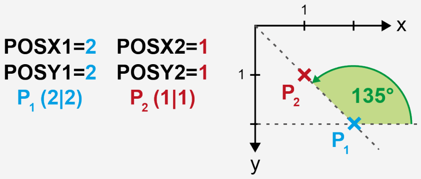
The input positions are shown with green crosses and a small number next to them (this number indicates the parameter, e.g. "1" for x1, y1).

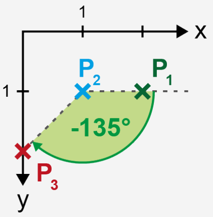
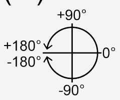
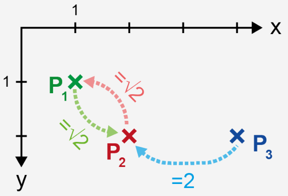
For distance calculations ("distance" and "nn_distance") the distances are drawn in.

For angles, the respective straight lines and the angle are drawn in.

- For the "angle_2points" angle: X axis and the line defined by the 2 entered points.
- For the "angle_3points" angle: the two straight lines defined by the 3 entered points.

If a calibration is active, the units set in the calibration are used and the corresponding coordinate system is used for angle calculation.

Operator name	Description	Example:
distance	Calculate distance between 2 points **IN: x1**, **y1**, **x2**, **y2**	distance(1;1;4;1) =[3]
angle_2points	Angle between the X axis and the line given by 2 points. Value range -180° to +180°. (Direction of rotation: see graphic below) If the calibration is active, the calculation is performed in the World coordinate system. If the calibration is inactive, the calculation is performed in the Image Coordinate System. **IN: PosX1**, **PosY1**, **PosX2**, **PosY2**  POSX1=2 POSX2=1 POSY1=2 POSY2=1 P1 (2|2) P2 (1|1) 1 x 1 y P2 135° P1 angle_2points (2;2;1;1) = [135] ***Fig. 256:*** *Example for angle_2points without calibration (IF)*	angle_2points (2;2;1;1) =[135]

Operator name	Description	Example:
angle_3points	Angle between 2 straight lines given by 3 points. Value range -180° to +180°. (Direction of rotation: see graphic below) If the calibration is active, the calculation is performed in the World coordinate system. If the calibration is inactive, the calculation is performed in the Image Coordinate System. **IN: PosX1**, **PosY1**, **PosX2**, **PosY2**, **PosX3**, **PosY3** **POSX1=2 POSX2=1 POSX3=0** **POSY1=1 POSY2=1 POSY3=2** **P_1 (2|1) P_2 (1|1) P_3 (0|2)**  angle_3points (**2**;**1**;**1**;**1**;**0**;**2**) = [**-135**] ***Fig. 257:** Example for angle_3points without calibration (IF)*	angle_3points (2;1;1;1;0;2) =[-135]
angle_diff	Difference between 2 angles. The function returns the smaller distance within the circle (or the specified period) with a sign. Direction of rotation (IF):  **IN: a1**, **a2** (angles 1 and 2), optional: **period** (period: possible values 90, 180 or 360; 360 is the default value) For period = 180, the results are between -90 and 90. For period = 90, the results are between -45 and 45.	angle_diff (200;10;360) =[170] angle_diff (200;10; 90) =[-10]
nn_distance	Calculate distance to nearest neighbor for each of the points **IN: x**, **y** (vectors with the same number of vector components (≥2)) **x = [1;2;4]** **y = [1;2;2]** → **P_1 (1|1) P_2 (2|2) P_3 (4|2)**  1 x 1 P_1 =√2 =√2 P_2 P_3 =2 y nn_distance (x,y) = [√**2**;√**2**;**2**]	nn_distance (x,y) = [1.414; 1.414; 2.000]
nn_distance_idx	Calculate the index of the nearest neighbor for each of the points: "At which position is the nearest point to this one?" **IN: x**, **y** (vectors with the same number of vector components (≥2)) **OUT:** Data type INT	x=[1;2;4] y=[1;2;2] nn_distance_idx (x;y) =[2;1;2]

8.3.19.1.1.5 Poses group

IN:/OUT: Data types INT / REAL

Operators of this group are used for processing (robot) poses. All operators of this group start with "pose3d_", and offer the possibility to process six-dimensional poses - that are vectors with 3 translations and 3 rotations each. The rotations are transferred in degrees [°].

Operator name	Description	Example:
pose3d_invert	Corresponds to the inversion of a displacement by one 3D pose. Determination of the inverse pose is done by transformation into and inversion of the corresponding homogeneous transformation matrix H followed by transformation back into the inverse pose. **IN:pose** **pose** = [X,Y,Z,Rx,Ry,Rz]	x = [1;0;0;0;0;90] pose3d_invert(x) = [0; 1; 0; 0; 0; -90]

pose3d_multiply	Determines a result pose equivalent to a consecutively executed shift by **pose1** and **pose2**. It is determined by transforming both poses **pose1** and **pose2** into the respective transformation matrices H_1 and H_2, multiplying both matrices and then transforming them back into a 3D pose.
	IN:pose1; **pose2**
	pose1 = [X1,Y1,Z1,Rx1,Ry1,Rz1];
	pose2 = [X2,Y2,Z2,Rx2,Ry2,Rz2]
		x = [1;0;0;0;0;90]
		y = [0;1;0;0;90;0]
		pose3d_multiply (x,y) = [0; 0; 0; -90; 0; 90]

pose3d_points	3D pose determination of an object by comparing *n* known geometric characteristics of the object with the coordinates in Image Coordinate System or (with automatic calibration) in World coordinate system.
	Note:**x_wf**, **y_wf** are vectors with X and Y coordinates of the respective determined characteristics.
	x_of, **y_of** and **z_of** are vectors with X, Y and Z coordinates of the real static characteristics on the object (ground truth).
	IN:x_wf; **y_wf**; **x_of**; **y_of**; **z_of**
	x_wf = [x1, x2, … , xN];
	y_wf = [y1, y2, … , yN];
	x_of = [xOF_1, xOF_2, … , xOF_N];
	y_of = [yOF_1, yOF_2, … , yOF_N];
	z_of = [zOF_1, zOF_2,…, zOF_N]
		Depending on calibration

8.3.19.1.1.6 Trigonometry group

Note: The calculations are performed based on degree values (not in radians).

IN:/OUT: Data type REAL

Operator name	Description	Example:
sin	Sine of the scalar or vector components	sin(90) =[1]
cos	Cosine of the scalar or vector components	cos(180) =[-1]
tan	Tangent of the scalar or vector components	tan(45) =[1]
arcsin	Arc sine of the scalar or vector components **IN:** Data type REAL -1 ≤ **v** ≤ 1	arcsin(0,5) =[30]
arccos	Arc cosine of the scalar or vector components **IN:** Data type REAL -1 ≤ **v** ≤ 1	arccos(0,5) =[60]
arctan	Arc tangent of the scalar or vector components	arctan(1) =[45]
arctan2	Arc tangent of the scalar or vector components with 2 arguments	arctan2(1;0) =[90]

Operator name	Description	Example:
to_degree	Convert radian to degrees	to_degree (pi) =[180]
to_radian	Convert degrees to radian	to_radian (180) =[3.142]

8.3.19.1.1.7 Rounding group

IN: Data type REAL

OUT: Data types REAL / INT

Operator name	Description	Example:
round	Rounding of the scalar or vector components **IN:** optional: **prec** (precision of rounding to decimal places, Data type INT; no specification or "0" leads to rounding to integer)	round (3.667;1) =[3.7]
ceil	Ceiling function: INT value greater than or equal to the scalar or vector components	ceil ([2.3;-3.5]) =[3;-3]
floor	Floor function: INT value less than or equal to the scalar or vector components	floor ([2.3;-3.5]) =[2;-4]
trunc	Create INT-value by truncating the scalar or the vector components (truncating the decimal places)	trunc ([2.3;-3.5]) =[2;-3]

8.3.19.1.1.8 Vector group

IN:/OUT: Data types BOOL / INT / REAL / STR

Example vector: **v1**=[2;3;5] **v2**=[20;30;50]

Operator name	Description	Example:
[ ]	Create vector	[2;3;5]
	Link (concatenate) vectors	[[2;3];[4;6]] =[2;3;4;6]
:	Create vector with components from "**i**" to "**j**" **IN: i**, **j** (vector components from/to, Data type INT, values ≥ 0)	0:3 = [0;1;2;3]
::	Return range of vector components up to the end of the vector **IN:** Index of the desired vector components, Data type INT	v1(**2**::) =[3;5]

new	Create new vector by specifying length and default value
	IN: length (number of vector components, Data type INT)
	v0 (value)
		new(3;1.2)
		=[1.2; 1.2; 1.2]
		new(3;"xy")
		["xy";"xy";"xy"]

size	Return the number of components of the vector **OUT:** Data type INT	size(v1) =[3]
is_empty	Check if vector is empty **OUT:** Data type BOOL	is_empty(v1) =[false]
concat	Link (concatenate) vectors **IN: v**, **w** (vectors)	concat (v1;[0;0;0]) =[2;3;5;0;0;0]
interleave	Interleave vector components **IN: v**, **w** (vectors)	interleave (v1;v2) =[2;20;3;30;5;50]

Operator name	Description	Example:
()	Access vector components via their indexes **IN: v** (vector), **w** (index of the desired vector components, Data type INT)	v1(3) =[3]
bound	Return values that are within the specified range or output vector components that are ≥ lower bound or ≤ upper bound (use **[ ]** for this purpose). **IN: v** (Data type REAL), **vmin**, **vmax** (upper/lower limit, Data type REAL) **OUT:** Data type REAL	bound (v1; 3; 6) =[3; 5] bound (v1; [ ]; 3) =[2;3] bound (v1; 3; [ ]) =[3;5]
bound_idx	Return indexes of vector components whose values are within the specified range. If the lower or upper limit is irrelevant, use **[ ]** for it. **IN: v** (Data type REAL), **vmin**, **vmax** (upper/lower bound, Data type REAL) **OUT:** Data type INT	bound_idx (v1;3;5) =[2;3] bound_idx ([2;7;5;3;4]; 4; [ ]) =[2;3;5]
select	Access vector components via their indexes	select(v1;1) =[2]
insert	Insert vector **w** into vector **v** at each of the positions indicated by the indices. The indices refer to the position in the origin vector **v**. If an index is greater than the length of **v**, it is padded with 0 accordingly. **IN: v** (vector), **indexes** (Data type INT), **w** (vector)	insert([1;2;3;4];[2;6];[8;9]) =[1;8;9;2;3;4;0;8;9]
replace	Replace all the vector components specified by the indices with the vector **w.** **IN: v** (vector), **indexes** (Data type INT), **w** (vector)	replace (["a";"b";"c"];[2];["x";"y"]) =["a";"x";"y";"c"]
replace_section	Replace vector components from the specified start "start" to end "end" with the vector **w.** Start and end can be vectors. **IN: v** (vector), **start** (Data type INT), **end** (Data type INT), **w** (vector)	replace_section ([1;1;1;1;1];2;4;[8;9]) = [1;8;9;1] replace_section ([1];3;10;[8;9]) = [1;0;8;9] replace_section ([1;1;1;1;1];2;4;[]) = [1;1]
append_shift	Append new element **w** to the end of vector **v.** The length of the result vector should remain constant or ≤ **max_length**. Accordingly, elements at the beginning of the vector are deleted. **IN: v** (vector), **w** (vector), optional: **max_length** (Data type INT)	append_shift ([1;2;3;4;5],7) = [2;3;4;5;7]
duplicates	Return repeated vector components. **IN: v** (vector)	duplicates ([5;6;5;5]) = [5]
duplicates_idx	Return all indexes of all repeated vector components. **IN: v** (vector)	duplicates_idx ([5;6;5;5]) = [3;4]

Operator name	Description	Example:
delete	Removes the elements at the given indices from a vector. **IN:v** (vector), indices (Data type INT)	delete ([10;20;30;40]; [1;3]) = [20;40] delete ([10;20;30;40];2::) = [10]

8.3.19.1.1.9 Vector properties group

This group provides operators to combine ("aggregate") all components of a vector. These operators start with the abbreviation "vec_" to distinguish them from operators of the same name, which process vectors component by component.

IN:/OUT: Data type REAL
Example vector: **v1**=[2;4;5]

Operator name	Description	Example:
vec_sum	Sum of the vector components	vec_sum(v1) =[11]
vec_product	Product of the vector components	vec_product(v1) =[40]
vec_mean	Mean of the vector components	vec_mean(v1) =[3.667]
vec_stddev	Standard deviation of the vector components	vec_stddev(v1) =[1,528]
vec_median	Median of the vector components	vec_median(v1) =[4]
vec_median_idx	Index of the median of the vector components **OUT:** Data type INT	vec_median_idx ([4;2;5]) =[1]
vec_min	Returns the smallest of the vector components (by aggregating)	vec_min(v1) =[2]
vec_min_idx	Index of the minimum of the vector components **OUT:** Data type INT	vec_min_idx(v1) =[1]
vec_max	Returns the largest of the vector components (by aggregating)	vec_max(v1) =[5]
vec_max_idx	Index of the maximum of the vector components **OUT:** Data type INT	vec_max_idx(v1) =[3]
vec_and	AND operation within a vector **IN:/OUT:** Data type BOOL	vec_and ([true;false]) =[false]
vec_or	OR operation within a vector **IN:/OUT:** Data type BOOL	vec_or([true;false]) =[true]

8.3.19.1.1.10 Sorting group

Example vector: **v1**=[2;4;5]

Operator name	Description	Example:
sort	Sort vector in ascending order.	sort (["z";"x";"y"]) =["x";"y";"z"]
sort_idx	Return indexes of the vector components according to their size (ascending order): "At which position is the smallest / ... / greatest vector component?" **OUT:** Data type INT	sort_idx ([4;5;2]) =[3;1;2]
sort_by_idx	Sort vector by specified index vector **IN: v** (Vector), **idx** (Vector: Data type INT)	sort_by_idx([v1]; [2;3;1]) =[4;5;2]
invert	Inverse order of vector components	invert ([2;5;4;1]) =[1;4;5;2]

8.3.19.1.2 Operands

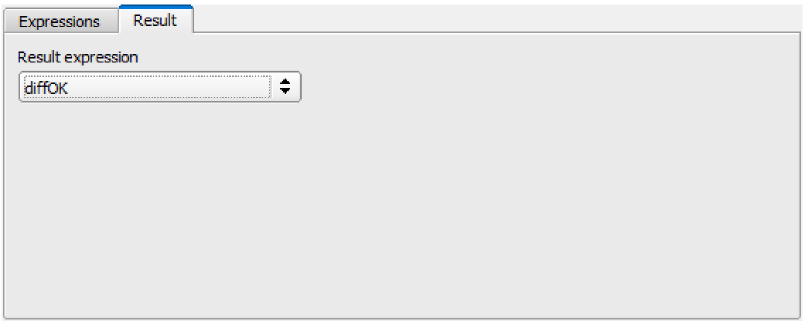
The "Operands" button can be used to access detector results and variables, and constants can be inserted.

Operand	Description
Detector (detector name)	Access to all detector results, these can also be accessed in the telegram (setup step Output / Telegram) Only those detectors that are listed **before** / **above** the current detector in the detector list can be accessed by the Result processing (i.e. which were first created or moved up). To access a result, the detector number is used in the expression, e.g.: "D1.Score".
Variables	Access to results of expressions (only from this Result processing detector), which are **before** / **above** the current expression (by default "v1", "v2", ...)
Constants	Inserting the constants "true", "false", "pi", "e".

For more information on the results of the individual detectors, see Communications manual in the Data Output chapter.

8.3.19.2 Result tab

The "Result" tab defines how the detector result (green or red LED) is generated.



Result expression

diffOK

Fig. 258: *Detector Result processing, Result tab*

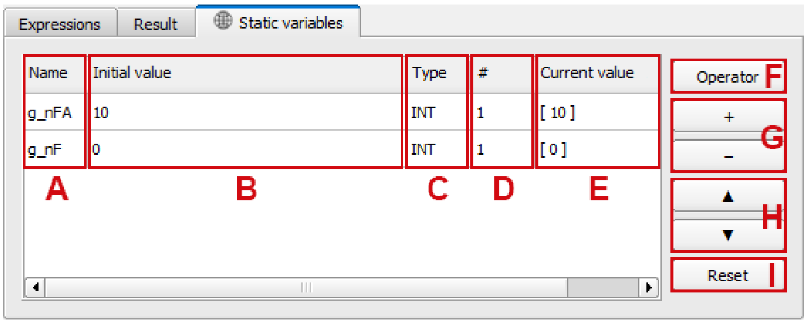
Parameter description:

Parameter	Function
Result expression	- "All expressions valid" (default): By default the detector is "OK" (green) if all expressions are valid. - Select an expression from the drop-down list (only expressions of Data type BOOL are displayed): The detector can thus be assigned the result of a Boolean variable. It is displayed accordingly: for "true" = green / "OK" for "false" = red / "NOK".

8.3.19.3 Static variables tab

In the Static variables tab variables can be created that have the attributes "static" and "global". The Static variables tab is identical in all created jobs. They have the prefix "g_".

These variables can be accessed globally, i.e. from any job. Furthermore, they are static, i.e. they persist beyond a trigger and are not overwritten like normal variables.



Name	Initial value	Type	#	Current value
g_nFA	10	INT	1	[10]
g_nF	0	INT	1	[0]

Fig. 259: *Detector Result processing, Static variables tab*

Parameter description: Columns and buttons in the Static variables tab

Column	Description
A: Name	Name of the expression (changeable), used as a variable. Numbers 0-9, letters A-Z, underscores are allowed, no special characters.
B: Initial value	Initial value of the variable after switching on the vision sensor.
C: Type	Data types: BOOL (Boolean number), INT (integer), REAL (floating point number), STR (text)
D: #	Number of components of the result, e.g. for vectors.
E: Current value	Current content of the variable.

Buttons	Description
F: Operators	Operators: divided into groups. Additional information: Chapter Operators.
G: + / -	"+" adds a new line at the end, "-" deletes the currently selected line.
H: ▲ / ▼	▲ / ▼ moves the current line one position up / down.
I: Reset	Resets the current value of the variable to the initial value.

8.3.19.4 Application examples: "Result processing" detector

8.3.19.4.1 Examples "Result processing: Math"

8.3.19.4.1.1 Simple calculations and checks

- Check whether the label of a package is centered and the angle of the label is correct (in the example demonstrated here: position X)
- Detect the position of the packaging and the label with "Contour" detector.
- Determine difference and check thresholds

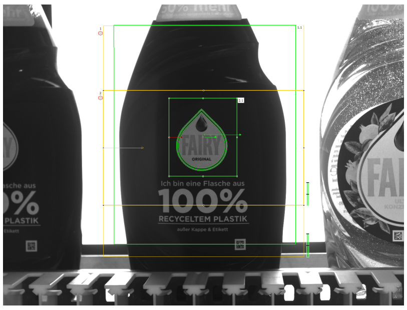


Fig. 260: *Case 1: Label correctly applied*

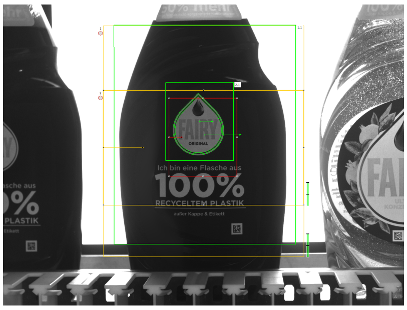


Fig. 261: *Case 2: Label not applied correctly*

Requirements and settings in SensoConfig:

- Detector 1 (**D1**): Contour detector for detecting the bottle
- Detector 2 (**D2**): Contour detector for detecting the label
  Both detectors output values for X position.
- Detector 3: Result processing: Math

Result processing - Expressions:

Case 1: Label correctly applied

Name	Expression	Type	#	Values
xOffset	abs(**D1**.PosX - **D2**.PosX)	REAL	1	[3,745]
Result	xOffset < 4	BOOL	1	**[true]**

Case 2: Label not applied correctly

Name	Expression	Type	#	Values
xOffset	abs(**D1**.PosX - **D2**.PosX)	REAL	1	[18,178]
Result	xOffset < 4	BOOL	1	**[false]**

Then select the expression "Result" in the "Result" tab, so that the overall detector result is output accordingly.

8.3.19.4.1.2 Calculate 2D distances

- Check if the white circle is centered within the black circle
- Detecting center of gravity with "BLOB" detector
- Calculate distance

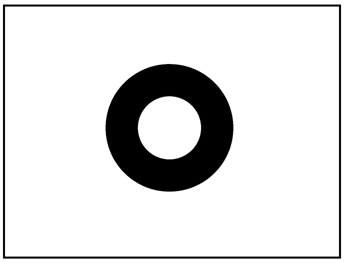


__Fig. 262:__ Case 1: White circle centered

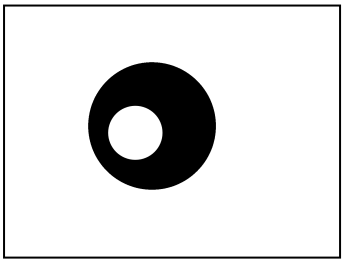


__Fig. 263:__ Case 2: White circle not centered

Requirements and settings in SensoConfig:

- Detector 1 (**D1**): BLOB detector for detecting the black circle
- Detector 2 (**D2**): BLOB detector for detecting the white circle
- For D1 and D2, in the "Features" tab: Activate C1 Circuit > Pos. X and C1 Circuit > Pos. Y
- Detector 3: Result processing: Math

Result processing - Expressions:

Case 1: White circle centered

Name	Expression	Type	#	Values
CenterDistance	distance(**D1**.C1_PosX; **D1**.C1_PosY; **D2**.C1_PosX; **D2**.C1_PosY)	REAL	1	[0,045]
Threshold	CenterDistance < 1	BOOL	1	**[true]**

Case 2: White circle not centered

Name	Expression	Type	#	Values
CenterDistance	distance(**D1**.C1_PosX; **D1**.C1_PosY; **D2**.C1_PosX; **D2**.C1_PosY)	REAL	1	[77,822]
Threshold	CenterDistance < 1	BOOL	1	**[false]**

Then select the expression "Threshold" in the "Result" tab, so that the overall detector result is output accordingly.

8.3.19.4.1.3 Handling empty vectors

- Check if vector is empty
- Depending on this, output the result to the PLC

Result processing - Expressions:

Error, because result of expression "Calculation" is empty

Name	Expression	Type	#	Values
Value	0	INT	1	[0]
Calculation	10 / Value			[ ] Division by 0
Calculation_OK	size(Calculation)!=0	BOOL	1	[false]
Result_to_PLC	if(Calculation_OK; Calculation; -1)	INT	1	[-1]

8.3.19.4.1.4 Delay of plant stop at NOK production

This example shows how the system stop can be controlled in the event of a repeat fault. When 10 NOK products have been detected in succession, a signal is output. A detector is used to count the number of holes. An error can occur due to a defective tool. This can be detected if there are NOK components as a result.

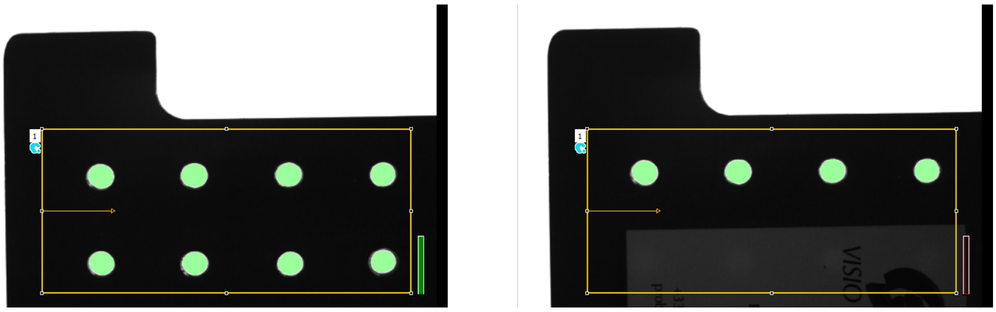


***Fig. 264:** Drill holes OK (left), drill holes NOK (right)*

Requirements and settings in SensoConfig:

- Detector 1(**D1**): BLOB detector for counting the holes
- Detector 2: Result processing: Math

Result processing - Create static variables:

Name	Initial value	Type	#	Values
g_numFailsAllowed	10	INT	1	-
g_numFails	0	INT	1	-

Result processing - Expressions:

Case 1: No product NOK

Name	Expression	Type	#	Values
g_numFails	if((g_numFails >= g_numFailsAllowed & **D1**) | **D1**; 0; if (!**D1**; g_numFails + 1; g_numFails))	INT	1	0
Result	if(g_numFails >= g_numFailsAllowed; true; false)	BOOL	1	false

Case 2: 10 NOK products detected in a row

Name	Expression	Type	#	Values
g_numFails	if((g_numFails >= g_numFailsAllowed & **D1**) | **D1**; 0; if (!**D1**; g_numFails + 1; g_numFails))	STR	4	10
Result	if(g_numFails >= g_numFailsAllowed; true; false)	REAL	4	true

Case 3: More than 10 NOK products detected in a row

Name	Expression	Type	#	Values
g_numFails	if((g_numFails >= g_numFailsAllowed & **D1**) | **D1**; 0; if (!**D1**; g_numFails + 1; g_numFails))	STR	4	13
Result	if(g_numFails >= g_numFailsAllowed; true; false)	REAL	4	true

Then select the expression "Result" in the "Result" tab, so that the overall detector result is output accordingly.

8.3.19.4.2 Examples "Result processing: Text"

8.3.19.4.2.1 Text comparison

- Check whether the content of the DataMatrix-Code matches the content of the barcode
- Output result as result on digital switching output

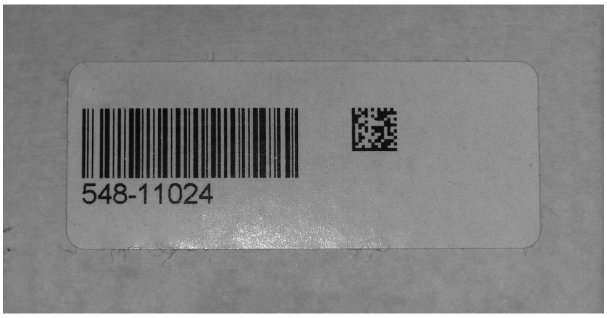


Fig. 265: *Comparison of DataMatrix code and barcode*

Requirements and settings in SensoConfig:

- Detector 1 (**D1**): Datacode detector
- Detector 2 (**D2**): Barcode detector
- Detector 3: Result processing: Text

Result processing - Expressions:

Name	Expression	Type	#	Values
DMC_Result	**D1**.String	STR	1	["543-11024"]
Barcode_ Result	**D2**.String	STR	1	["548-11024"]
Result	DMC_Result = Barcode_Result	BOOL	1	**[false]**

Then select the expression "Result" in the "Result" tab, so that the overall detector result is output accordingly.

8.3.19.4.2.2 Sorting the Results Output Based on Position

- Output the results of several codes based on their Y-position from top to bottom

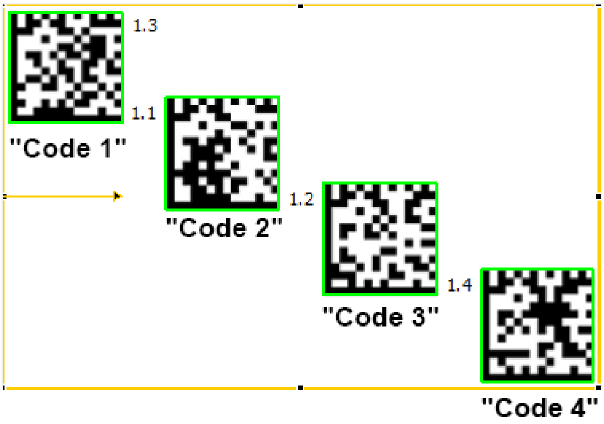


Fig. 266: *Sorting of data codes*

Requirements and settings in SensoConfig:

- Detector 1 (**D1**): Datacode detector
  Here: "Max. number of codes" Parameter is set to 4.
- Detector 2: Result processing: Text

Result processing - Expressions:

Name	Expression	Type	#	Values

ResultString	**D1**.String	STR	4	["Code 2";
				"Code 3";
				"Code 1";
				"Code 4"]

yPosition	**D1**.PosY	REAL	4	[359; 564; 154; 772]
IndexPos	sort_idx(yPosition)	INT	4	[3; 1; 2; 4]

Result	sort_by_idx(ResultString; IndexPos)	STR	4	["Code 1";
				"Code 2";
				"Code 3";
				"Code 4"]

8.3.20 Detector Wafer

This detector is suitable for inspecting breakouts on the edges of wafers or cells during production and for measuring geometric parameters such as width, height, position, angle of rotation, etc. It is extremely accurate in measuring the size and position of the wafer and can be used as a tool to pick up and place robotic systems.

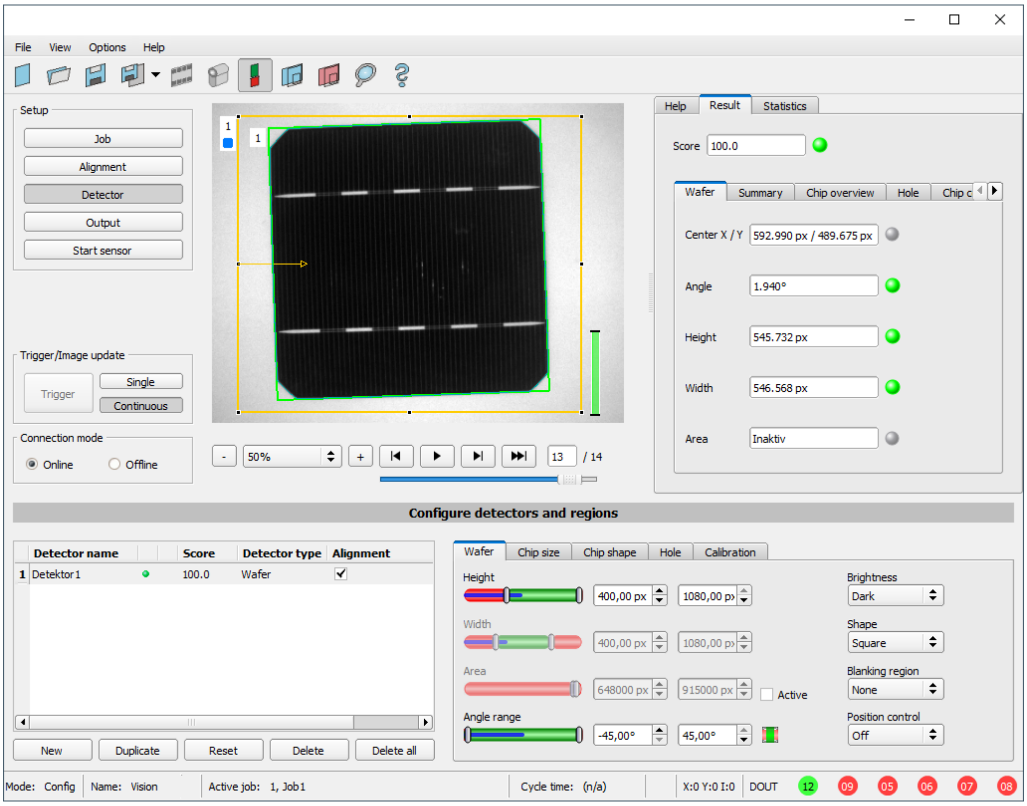
NOTE:
The tabs "Binarization", "Rectangle fit" and "Miscellaneous" are only accessible in the expert mode. Activation via menu bar "Options / Expert Mode".

The VISOR® Solar is characterized by the following properties:

- Automatic recognition of wafer and cell geometry
- Reliable detection of imperfections on straight and curved contours
- Flexible setting of the test criteria: e.g. tolerances for wafer size and rotational position, size, and number of allowed contour defects

- Easy optimization of the sensor with regard to evaluation speed and inspection accuracy (subpixel method)
- Free cutting function for larger defects
- Distortion removal

8.3.20.1 Wafer tab



Score 100.0

Center X / Y	592.990 px / 489.675 px
Angle	1.940°
Height	545.732 px
Width	546.568 px
Area	Inaktiv

	Detector name	Score	Detector type	Alignment
1	Detektor 1	100.0	Wafer	☑

Height	400,00 px	1080,00 px
Width	400,00 px	1080,00 px
Area	648000 px	915000 px
Angle range	-45,00°	45,00°

□ Active

Brightness: Dark

Shape: Square

Blanking region: None

Position control: Off

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

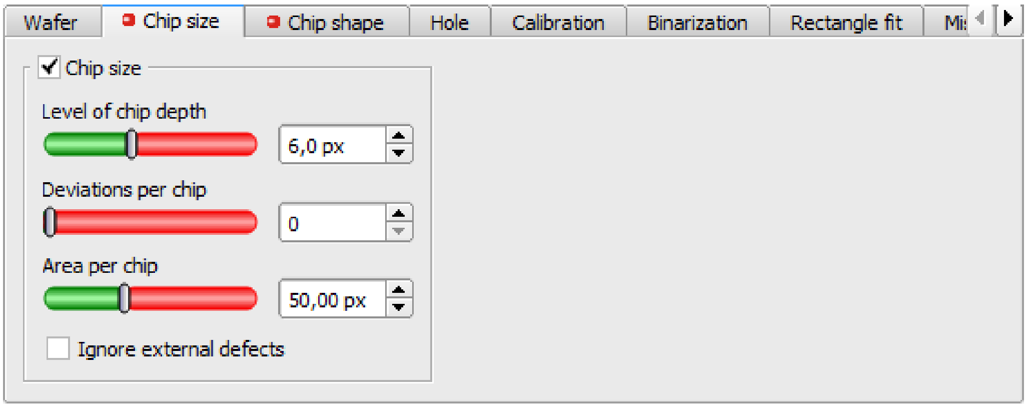
Fig. 267: Detector Wafer, Wafer tab

Parameter description:

Parameter	Function
height	Range to accept the height of a wafer.
Width	Range to accept the width of a wafer.
Area	Range to accept the area of a wafer.
Angle range	Range to accept the current value of rotation.
Brightness	Select the brightness of your object comparing to the background.
Shape	Choice between rectangular or square wafer shape
Blanking region	This selection box allows up to 12 rectangular areas to be positioned freely in the image. The areas within these zones are not used for wafer control.
Position control	To control the position of the wafer center of gravity, a rectangle or an ellipse can be freely positioned in the image area as the desired area.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.20.2 Chip size tab



☑ Chip size

Level of chip depth: 6,0 px

Deviations per chip: 0

Area per chip: 50,00 px

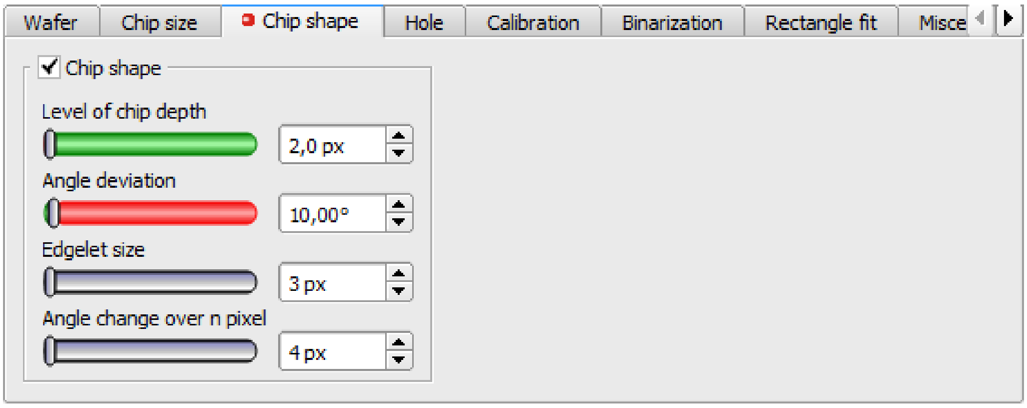
□ Ignore external defects

Fig. 268: Detector Wafer, Chip size tab

Parameter description:

Parameter	Function
Chip size	Activate chip size.
Threshold of chip depth	From the detected contour points, an ideally approximated, symmetrical right-hand contour is calculated. Based on this, the distances are calculated for all contour points. The threshold "max. deviation" defines the value for a faulty distance.
Deviations per chip	Defines the max. number of faulty distances for GOOD / BAD detection.
Area per chip	Defines the threshold of a faulty area for GOOD / BAD detection.
Ignore external defects (Expert mode)	All detected error contour points that lie outside the box enclosing the wafer (color: turquoise) are not taken into account during the contour check.

8.3.20.3 Chip shape tab



☑ Chip shape

Level of chip depth: 2,0 px

Angle deviation: 10,00°

Edgelet size: 3 px

Angle change over n pixel: 4 px

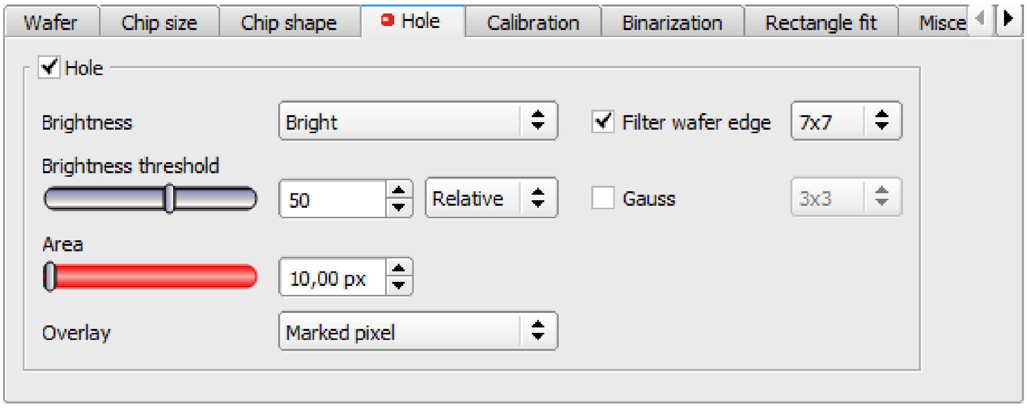
Fig. 269: Detector Wafer, tab Chip shape

Parameter description:

Parameter	Function
Chip shape	Activate chip shape.
Threshold of chip depth	If a protrusion has been detected and there is a chip in the fault region, then the internal errors below the set deviation (in pixels, relative to the enclosing box) are classified as faults.

Parameter	Function
Angle deviation	A contour point is detected as an error if the calculated angle change is above the threshold value.
Edge segment (Expert mode)	For each individual contour point, an edge segment (length, rotational position in the image) is determined on the basis of two adjacent contour points. Parameter: Distance to the neighboring points.
Angle change over n pixels (Expert mode)	For each individual contour point, the maximum difference of the rotational positions of the associated n edge segments is determined from n adjacent contour points.

8.3.20.4 Hole tab



☑ Hole

Brightness: Bright

☑ Filter wafer edge: 7x7

Brightness threshold: 50 Relative

□ Gauss: 3x3

Area: 10,00 px

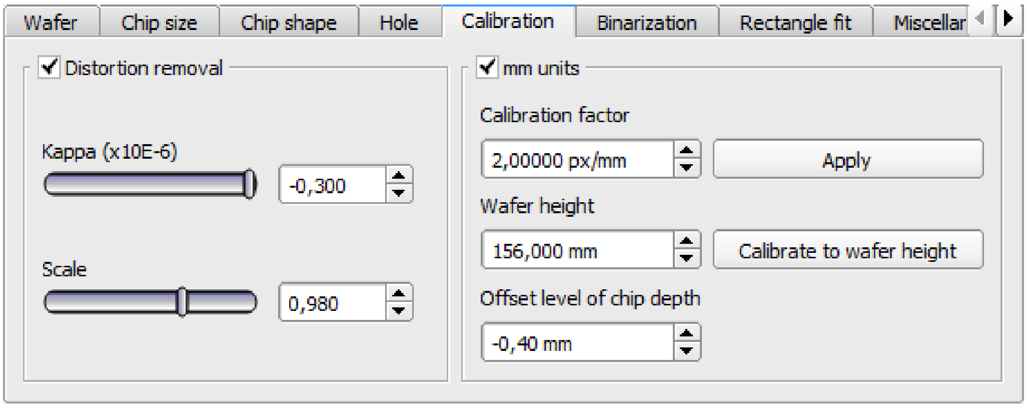
Overlay: Marked pixel

Fig. 270: Detector Wafer, tab Hole

Parameter description:

Parameter	Function
Hole	Activate hole recognition.
brightness	Select the brightness of the object based on the brightness of the wafer.
Brightness threshold / absolute	Define the intensity threshold to detect an faulty object as a fixed gray value.
Brightness threshold / relative	Defining the brightness threshold for object recognition as an offset value in addition to the dynamically determined, average gray value of the wafer.
Area	Definition of the minimum hole / object area to be detected. Value in pixels * pixels or calibrated in mm * mm.
Overlay	(De)activation of the overlay of detected objects.
Edge filter (Expert mode)	Extending dark areas, eliminating bright pixels in dark areas, eliminating artifacts, separating bright objects.
Gauss (Expert mode)	Reduction of noise, suppression of spurious details and artifacts, smoothing of edges.

8.3.20.5 Calibration tab



☑ Distortion removal

Kappa (x10E-6) -0,300

Scale 0,980

☑ mm units

Calibration factor 2,00000 px/mm

Wafer height 156,000 mm

Offset level of chip depth -0,40 mm

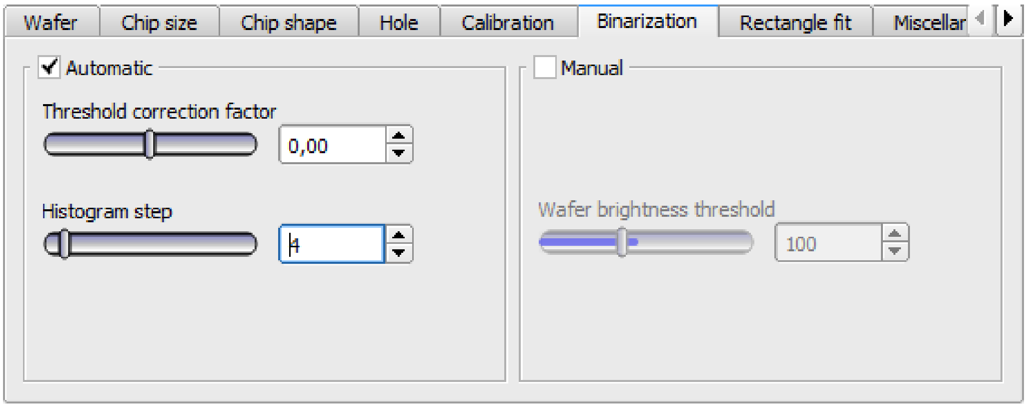
__Fig. 271:__ Detector Wafer, tab Calibration

Parameter description:

Parameter	Function
Distortion removal	Activate distortion removal.
• Kappa (x10E-6)	Distortion coefficient for modeling the radial distortion.
• Scaling	Multiplicative correction.
mm units	Activate mm units.
• Calibration factor	Pixel per mm; calibration factor to convert image data into world data.
• Apply	By pressing "Apply", the dimensions in the other tabs are automatically adjusted to the new calibration factor.
• Wafer height	Program automatically adjusts the calibration factor based on the measured wafer height in pixels.
• Calibrate to wafer height	Calibration factor is calculated from the value "Wafer height".
• Offset chip depth	Correction factor for the measured value of chip depth. The factor is added to the actual measured value.

8.3.20.6 Binarization tab

NOTE:
The tabs Binarization, Rectangle fit and Miscellaneous are only accessible in the expert mode. Activation via menu bar "Options / Expert Mode".



Wafer | Chip size | Chip shape | Hole | Calibration | Binarization | Rectangle fit | Miscellar

☑ Automatic

Threshold correction factor: 0,00

Histogram step: 4

□ Manual

Wafer brightness threshold: 100

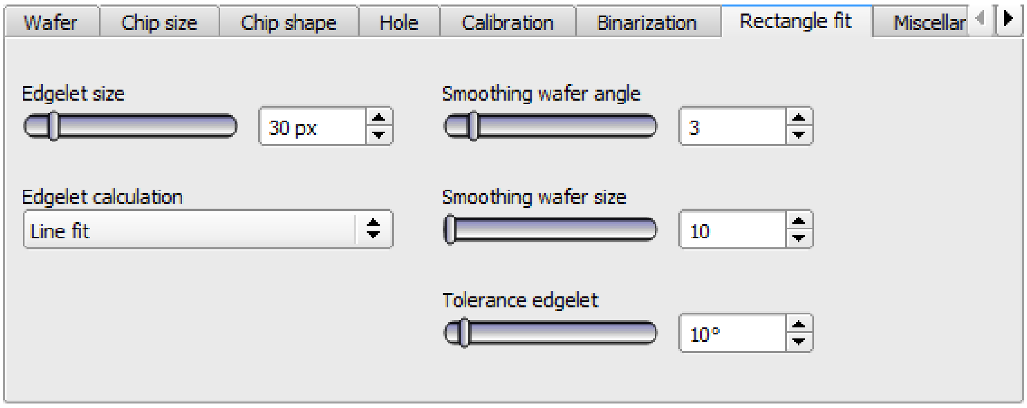
Fig. 272: Detector Wafer, tab Binarization

Parameter description:

Parameter	Function
Automatic	Activate automatic binarization.
Correction factor for threshold	The threshold value for contour detection is calculated automatically from the brightness distribution of foreground and background. The position of the contrast threshold in the histogram can be shifted in the direction of the foreground or the background color.
Histogram increment	Gray scale value resolution of histogram history.
Manual	Activate manual binarization.
Wafer brightness threshold	Fixed entry of brightness threshold.

8.3.20.7 Rectangle fit tab

NOTE:
The tabs Binarization, Rectangle fit and Miscellaneous are only accessible in the expert mode. Activation via menu bar "Options / Expert Mode".



Wafer | Chip size | Chip shape | Hole | Calibration | Binarization | Rectangle fit | Miscellar

Edgelet size: 30 px

Edgelet calculation: Line fit

Smoothing wafer angle: 3

Smoothing wafer size: 10

Tolerance edgelet: 10°

Fig. 273: Detector Wafer, Rectangle fit tab

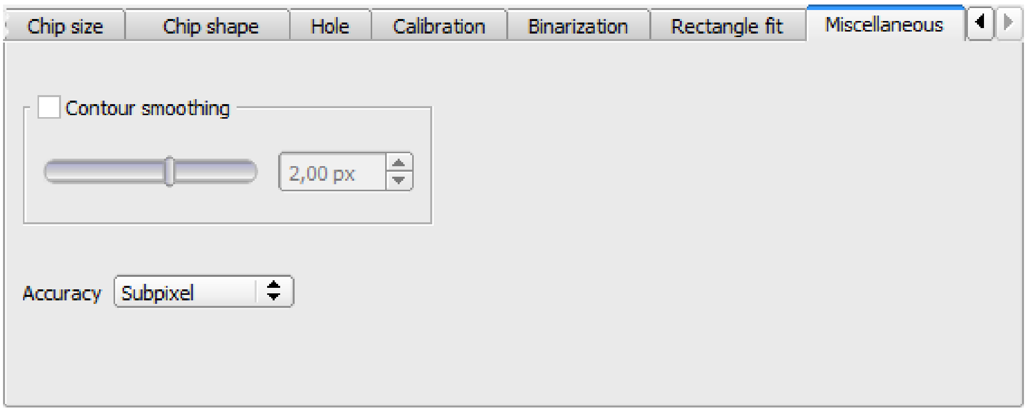
Parameter description:

Parameter	Function
Edge segment	Increment or number of contour elements for calculating local contraction.
Calculation edge segment	There are two modes for rectangular calculation: Secant or straight line adaptation.

Parameter	Function
Smoothing wafer angle	+/- range, based on the maximum of the Gaussian distribution of the individual angles, which is used for the angle calculation.
Smoothing wafer size	+/- range, based on the maximum of the Gaussian distribution of the individual quantities, which is used for the size calculation.
Tolerance edge segment	+/- angular range, based on the actual angle of the wafer of the local continuous line, which are taken into account for the calculation of the wafer size.

8.3.20.8 Miscellaneous tab

NOTE:
The tabs Binarization, Rectangle fit and Miscellaneous are only accessible in the expert mode. Activation via menu bar "Options / Expert Mode".



□ Contour smoothing

2,00 px

Accuracy: Subpixel

Fig. 274: Detector Wafer, tab Miscellaneous

Parameter description:

Parameter	Function
Contour smoothing (positive / negative)	With the help of the downstream contour smoothing, errors can be retrieved later: • increased (OPENING, parameter <0) or • reduced (CLOSING, parameter> 0).
Accuracy	De(activate) subpixel method for evaluation of all wafer / cell data.

8.3.20.9 Threshold value settings

Excerpt from: VISOR® SolarUserManual1WIP 05-14 V.pdf

Whether test parts are recognized as good or bad parts depends largely on the threshold settings. A typical setup will be illustrated by the following task: All good parts pass the test and all bad parts are recognized as bad parts and sorted out. In order to achieve this objective, some parts (pass and fail parts) need to be checked and the threshold limits must be adjusted until the results meet the relevant production requirement. Yes

In order to prevent borderline bad parts from passing through the test as good parts, the thresholds may need to be tightened, which increases the number of rejected parts but reduces the risk of downtime by passing borderline bad parts. If the thresholds are set too narrow, there is a risk that too many parts will be tested as bad parts and thrown out.

To ensure a high output, it may be necessary to set the thresholds less sharply. This carries the risk that a bad part will not be ejected, with all the negative consequences for further production.

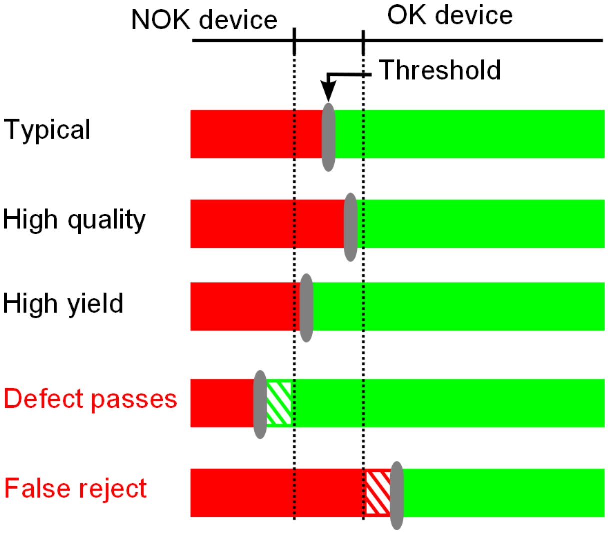
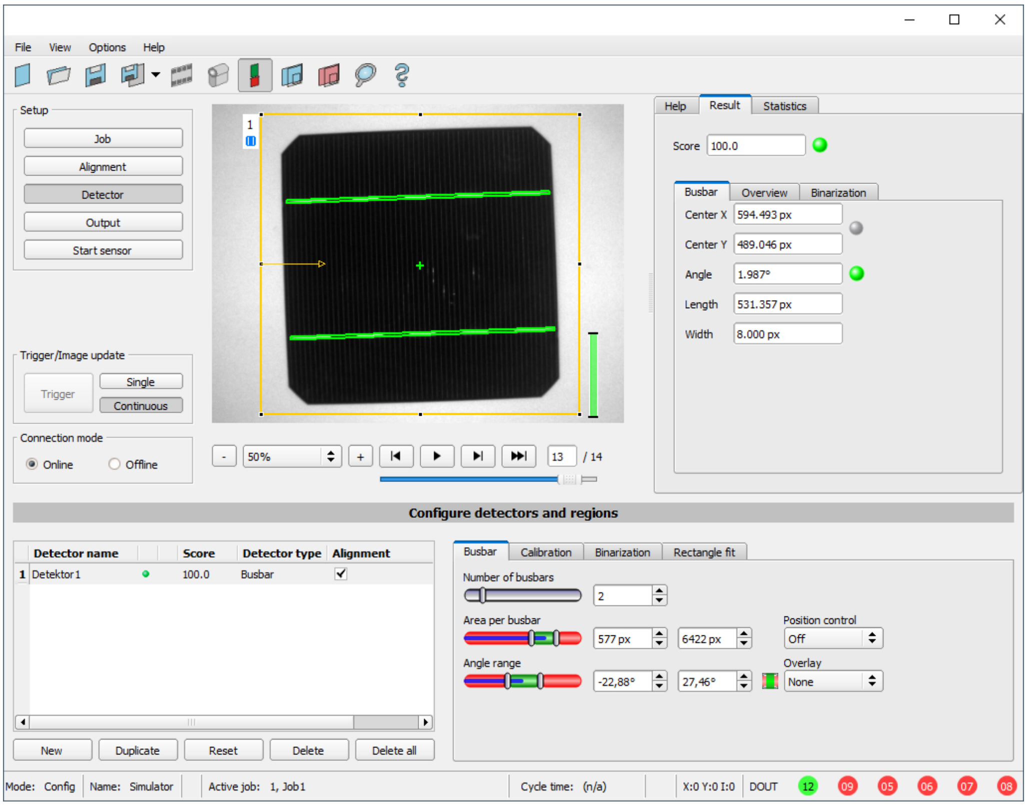


Fig. 275: Settings of the switching threshold

8.3.21 Detector Busbar

This detector is suitable for locating and testing busbars.

8.3.21.1 Busbar tab



Score 100.0

Center X 594.493 px

Center Y 489.046 px

Angle 1.987°

Length 531.357 px

Width 8.000 px

	Detector name		Score	Detector type	Alignment
1	Detektor 1	●	100.0	Busbar	✓

Number of busbars 2

Area per busbar 577 px 6422 px

Angle range -22,88° 27,46°

Position control Off

Overlay None

Mode: Config | Name: Simulator | Active job: 1, Job1 | Cycle time: (n/a) | X:0 Y:0 I:0 | DOUT

Fig. 276: Detector Busbar, tab Busbar

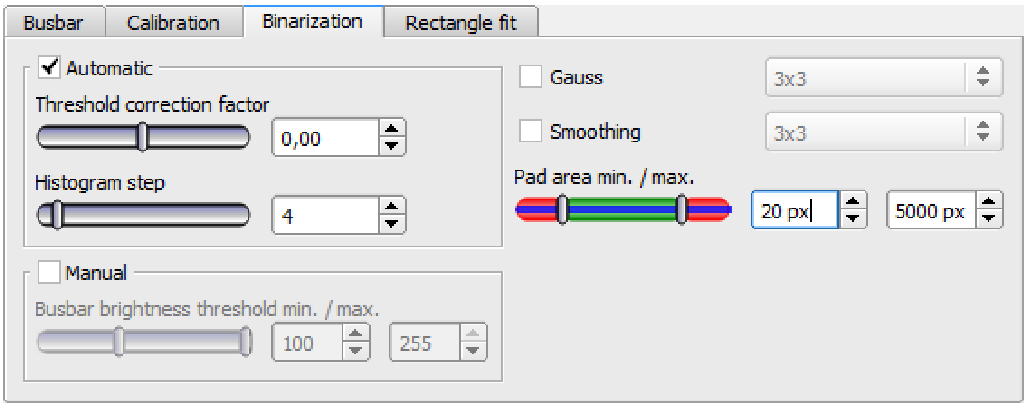
Parameter description:

Parameter	Function
Number of busbars	Setting for expected number of busbars.
Area per busbar	Range to accept the area of a busbar. The surface criterion is an abort criterion, i.e. if the calculated area is larger or smaller, then any further calculation is aborted. The area of the busbars is calculated from the total number of selected pixels.
Angle range	Range for the accepted rotational position
Position control	To control the position of the wafer center of gravity, a rectangle or an ellipse can be freely positioned in the image area as the desired area.
Overlays	Activate graphical overlays for busbar pixels.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.21.2 Binarization tab

NOTE:
The tabs Binarization and Rectangle fit are only accessible in the expert mode. Activation via menu bar "Options / Expert Mode".



☑ Automatic
Threshold correction factor 0,00
Histogram step 4
□ Manual
Busbar brightness threshold min. / max. 100 255
□ Gauss 3x3
□ Smoothing 3x3
Pad area min. / max. 20 px 5000 px

Fig. 277: Detector Busbar, tab Binarization

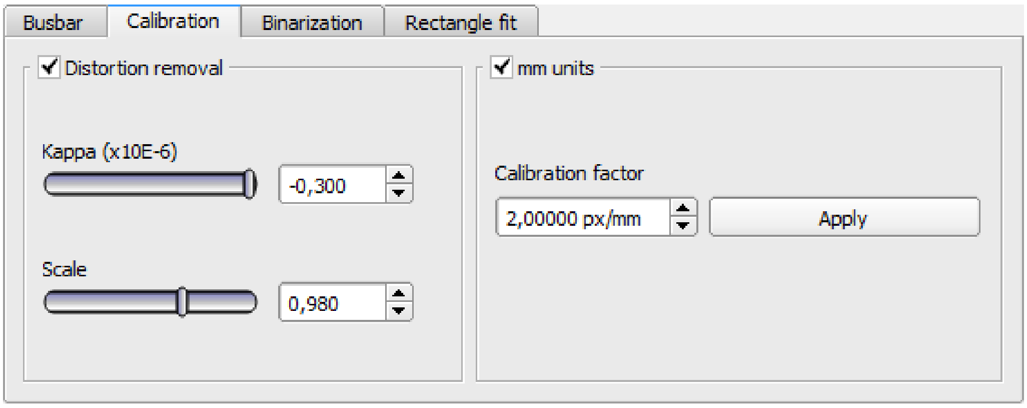
Parameter description:

Parameter	Function
Automatic	Activate automatic binarization.
Correction factor for threshold	The threshold value for contour detection is calculated automatically from the brightness distribution of foreground and background. The position of the contrast threshold in the histogram can be shifted in the direction of the foreground or the background color.
Histogram increment	Gray scale value resolution of histogram history.
Manual	Activate manual binarization.
Busbar brightness threshold min. max.	Min./max. grey threshold for pixels belonging to the busbar.

Parameter	Function
Contour smoothing	With the help of the downstream contour smoothing, errors can subsequently be: • increased (OPENING, parameter <0) or • reduced (CLOSING, parameter> 0).
Gauss	Reduction of noise, annoying details, and rounding of the corners.
Pad area min. max.	Minimum and maximum area to detect a single pad.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.21.3 Calibration tab



☑ Distortion removal

Kappa (x10E-6) -0,300

Scale 0,980

☑ mm units

Calibration factor 2,00000 px/mm

Fig. 278: Detector Busbar, tab Calibration

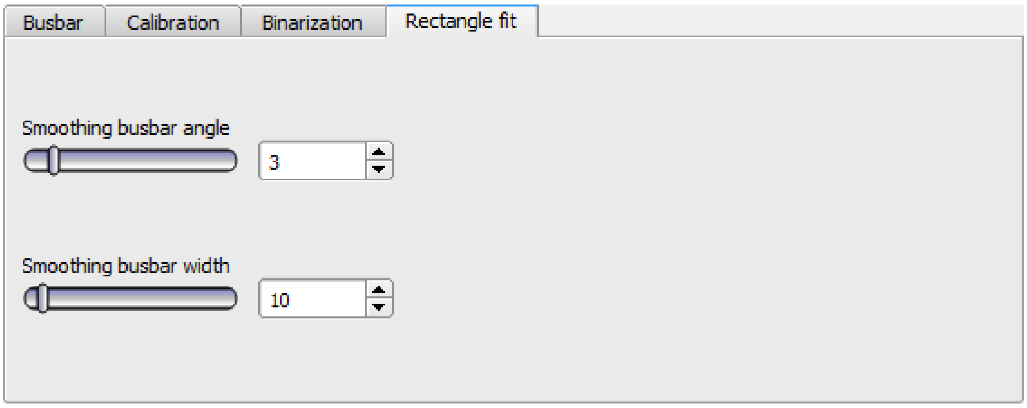
Parameter description:

Parameter	Function
Distortion removal	Activate distortion removal.
Kappa (x10E-6)	Distortion coefficient for modeling the radial distortion
Scaling	Multiplicative correction.
mm units	Activate mm units.
Calibration factor	Pixel per mm; calibration factor to convert image data into world data.
Apply	By pressing "Apply", the dimensions in the other tabs are automatically adjusted to the new calibration factor.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

8.3.21.4 Rectangle fit tab

NOTE:
The tabs Binarization and Rectangle fit are only accessible in the expert mode. Activation via menu bar "Options / Expert Mode".



Smoothing busbar angle: 3

Smoothing busbar width: 10

Fig. 279: Detector Busbar, tab Rectangle fit

Parameter description:

Parameter	Function
Smoothing busbar angle	+/- range, based on the maximum of the Gaussian distribution of the individual angles, which is used for the angle calculation.
Smoothing busbar width	+/- range, based on the maximum of the Gaussian distribution of the single widths, which is used for the size calculation.

For newly generated detectors, all parameters are preset as standard values, which are suitable for many applications.

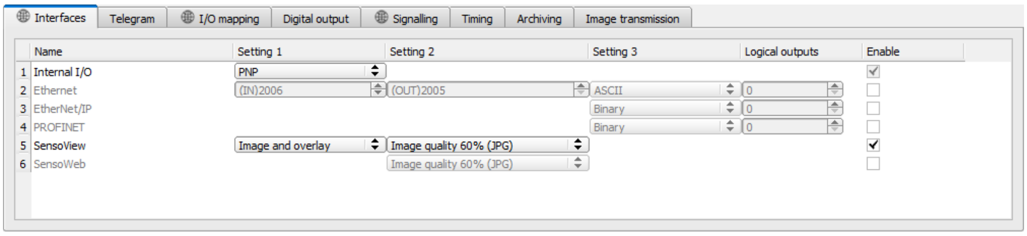
8.4 Setup step Output

The Output setup step can be used to define the assignment and logic gates for the digital signal outputs, as well the interfaces and output data for the VISOR®.

8.4.1 Interfaces tab

NOTE:
The settings in this tab will affect the job set globally.

In this tab, you select and activate the used digital inputs / outputs and the interfaces for data output. In the "Active" column, the outputs and interfaces can be activated or deactivated separately. In the column "Setting 3" you can select the format in which the data should be output.



	Name	Setting 1	Setting 2	Setting 3	Logical outputs	Enable
1	Internal I/O	PNP				☑
2	Ethernet	(IN)2006	(OUT)2005	ASCII	0	☐
3	EtherNet/IP			Binary	0	☐
4	PROFINET			Binary	0	☐
5	SensoView	Image and overlay	Image quality 60% (JPG)			☑
6	SensoWeb		Image quality 60% (JPG)			☐

Fig. 280: Output, Interfaces tab

Parameter description:

Parameter	Function
Digital I/O	Selection of the function of the digital I/O: PNP or NPN
Ethernet	Ethernet TCP/IP for data output. The sensor is always a socket server. Two different ports are used, which can be defined by the user. Default setting: Port 2006 (IN) for commands to the sensor (control commands and response protocol) and port 2005 (OUT) for the actual data output. Setting 3 can be used to select whether data should be output in binary (hex) or ASCII format. Additional information: see VISOR® Communications manual.
EtherNet/IP	Field bus EtherNet/IP for data output. Additional information: see VISOR® Communications manual.
PROFINET	Field bus PROFINET for data output and PLC communication. **NOTE:** The sensor starts the PROFINET stack as soon as a job with PROFINET is selected. This reduces the evaluation speed slightly. Switching to another job without PROFINET does not stop the PROFINET stack. In order to stop the stack, the device must be turned off. Additional information: see VISOR® Communications manual.
SensoView	Activation or deactivation of the module "SensoView". When the checkbox is deactivated, the button "View" in SensoFind can no longer be used to access SensoView. If the checkbox "SensoView" is activated (default), the following settings can be selected for image transferal: • **Overlay** When "Overlay" is selected, only the overlay is transferred to SensoView. Image and pre-processing settings are not transferred. • **Image and overlay** For the setting "Image and Overlay", the image and the overlays are transferred to SensoView. Pre-processing settings are not transferred. • **Image with pre-processing and overlay** For this setting, both the image with the pre-processing settings and the overlay will be transferred to SensoView. Additional information: Configure overlay and Pre-processing tab
SensoWeb	Turns on the web server on the vision sensor. Similar to the locally installed module "SensoView", images and results can be visualized via a web browser via "SensoWeb". Additional information: VISOR® – SensoWeb

Logical outputs:

Using the Ethernet and EtherNet/IP interfaces makes it possible to define additional logic outputs that only exist logically and can only be communicated via the data output interface.

Logical outputs can e.g. be associated with a detector result or a logical expression (formula).

8.4.2 Telegram tab

In the "Telegram" tab, configuration of data output for interfaces and archiving in .csv files is carried out. Result data that will be output via the previously activated interface can be configured here.

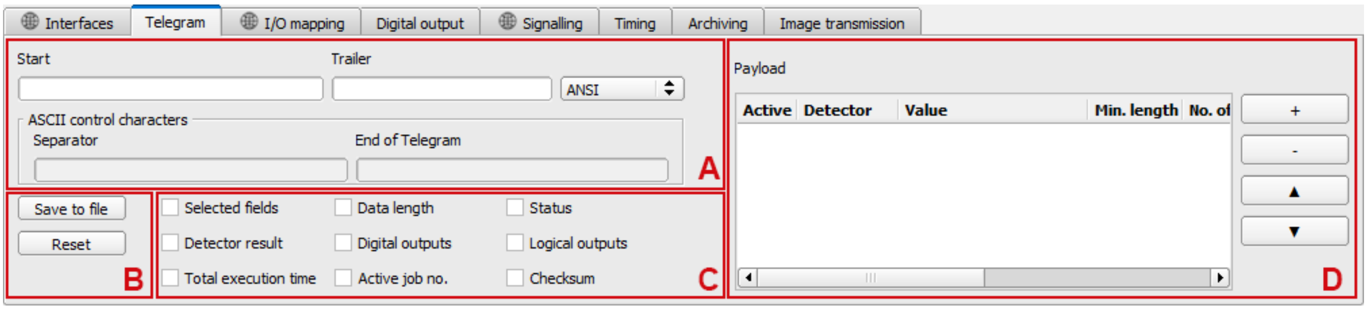


__Fig. 281:__ Output, Telegram tab

A: Control characters

B: General

C: Checkboxes

D: Payload

A: Control characters

Protocol standard contents (start .... to .... check sum)

The desired control characters for the data output can be set here.

Parameter description:	Length ASCII [Byte]	Length Binary [Byte]	Data type	Available for
Start				
Characters inserted at the beginning of the output data (prepended)	0 ... 8	0 ... 8	–	All types
Trailer				
Characters added at the end of the output data (appended)	0 ... 8	0 ... 8	–	All types
Separator				
Separates the values of detectors and selected fields in the output data	0 ... 4	na	–	All types
End of response				
Characters added to the response to a request	0 ... 4	na	–	All types

B: General

Parameter description:
Export
Exportation of file format with current results as .csv. Detailed data format output for the custom output string as a .csv file with: Byte position (start position in string), data type, field name, detector name, value, length (in bytes), detector number, and detector type.
Reset
Resets all the input in this tab.

C: Checkboxes

Standard contents that are used frequently can be added to the output string by simply filling them out or enabling the checkboxes.

Parameter description:	Length ASCII [Byte]	Length Binary [Byte]	Data type	Available for
Selected fields				
This checkbox displays all selected fields. The checkbox for selected fields is not displayed.	16	2	**ASCII**: The order of output is from left to right and from top to bottom, i.e. one byte is set per active checkbox starting with the LSB. **Binary**: The output sequence is from left to right and from top to bottom, i.e. one bit is set per active checkbox, starting with the least significant.	All types
Telegram length				
Number of characters including the characters for the telegram length itself.	1 ... 10	2	**ASCII**: E.g. output string with 10 characters; telegram length 10 + 2 characters (one byte per decimal place) = 12	All types
Status byte				
Returns the Trigger mode.	3	2	**ASCII**: PPF = Trigger; PFP = Free run **Binary**: 0X06 0x00 = Trigger mode 0X05 0x00 = Free run mode	All types
Detector results				
Output of overall result for each detector.	4 ... 261	3 ... 35	**ASCII**: Byte 1 = AND conjunction of all detectors Byte 2 = Overall Alignment result Byte 3 = Overall current job result Followed by the number of detectors; one byte per decimal place Following one byte per detector P = Detector pass F = Detector fail	All types

Parameter description:	Length ASCII [Byte]	Length Binary [Byte]	Data type	Available for
Digital outputs				
Returns the logic gate result for each digital output.	2 ... 7	n	**ASCII**: Byte 1 Number of active outputs (logic gate result assigned) Followed by bytes 2 – 7; one byte per output P = Detector pass F = Detector fail 0 = Inactive output (gap between two active outputs) **Binary**: Bytes 1 and 2: Number of active Outputs Bytes 3 – n: Outputs, bit-coded	All types
log. Outputs				
Returns the logic gate result for each logic output.	1 ... 259	N	**ASCII**: From byte 1 ... n: Number of active outputs to which a logic result has been assigned. Length: 1 byte per decimal place. The following bytes n ... m: 1 byte per logical output P = Detector pass F = Detector fail 0 = Inactive output (gap between two active outputs) **Binary**: Byte 1 - 2: Number of active outputs to which a logic result was assigned. Byte 3 ... n: all active logic outputs 1 = Detector Pass 0 = Detector Fail	All types
Execution time				
Returns the execution time for the last evaluation.	1 ... 3	4	Signed integer	All types
Active job				
Returns the job for the last evaluation.	1 ... 3	1	Unsigned int U8	All types
Telegram checksum				
XOR check sum of all bytes in the telegram. Is transmitted as the last byte.	1	1	Unsigned int	All types

D: Payload

The required data to be output is set here. The check boxes can be used to flexibly add detector-specific payload, in any order you want, to the data telegram.

1. Use the "+" button to generate new entry.

   Function of the buttons:

   - "+": Insert new entry
   - "-": Delete marked entry
   - "Up", "Down": Displace marked entry

2. Select the detector you want in the "Detector" column.

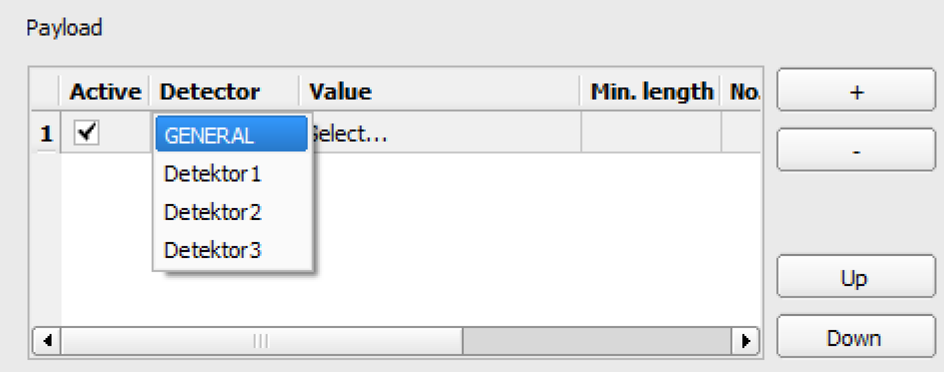


Fig. 282: *Telegram tab, Payload*

3. Select the detector value you want in the "Value" column so that this value will be output through the enabled interface.
   For more information on the available interfaces, please refer to: Interfaces tab

Column	Function
Active	Enables / disabled the selected output value
detector	Detector name (select from drop-down list)
Value	Available detector results (select from drop-down menu): Overview Payload see Communications manual, chapter Data output ASCII / BINARY
Min. Length	Used to define the minimum length for the "Value" cell. If the actual length is smaller than the set value, the field will be padded with spaces (ASCII) or zeros (binary).
No. of results	Number of results to be output from a detector for which multiple results are available. A number of results can be specified if it is a vector (several results of one type), e.g. for the detectors BLOB, Result processing, Contour or Pattern matching with Multiple objects. Example: The feature "area" was used as a filter and 15 BLOBs / objects were found. If "No. of results" is set to 10, a maximum of 10 results will be output, others will be discarded. If fewer objects are found, this number is output. Note: The value "0" means that **all re**sults are output. If a constant length of the telegram is to be output, the parameters "Min. length" and "No. of results" must be combined. All available output data see Communications manual, Chapter Data output ASCII / BINARY
Factor	Factor by which the result data is multiplied (determines the number of decimal places)

<table>
<tr><th>Column</th><th>Function</th></tr>
<tr><td>Bit depth (binary)</td><td>Specifies the length in bits and thus the value range of the transmitted value.
If the value is outside the value range, it is displayed with the highest / lowest possible value.
Usable value ranges:
<table>
<tr><th>Bit</th><th>Signed</th><th>unsigned</th></tr>
<tr><td>8</td><td>-127 to +126</td><td>0 to 254</td></tr>
<tr><td>16</td><td>-32.767 to 32.766</td><td>0 to 65.535</td></tr>
<tr><td>32</td><td>-2.147.483.647 to 2.147.483.646</td><td>0 to 4.294.967.295</td></tr>
</table></td></tr>
<tr><td>Sign (binary)</td><td>Specifies whether the transmitted value is signed or unsigned (Signed / Unsigned)</td></tr>
</table>

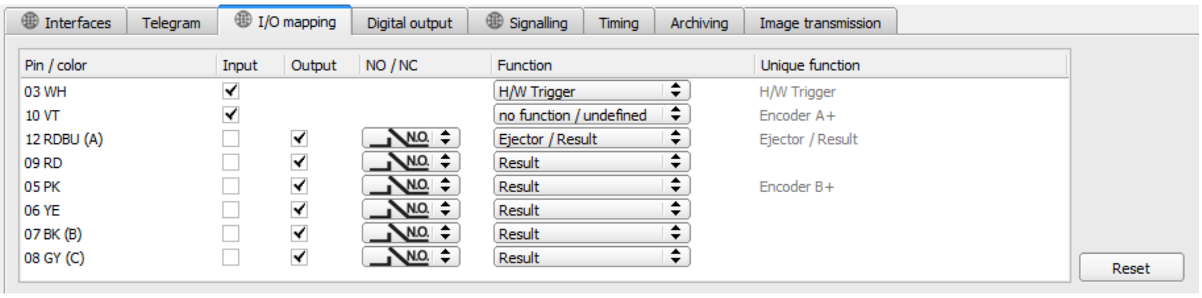
Overview Payload see Communications manual, chapter Data output ASCII / BINARY

8.4.3 I/O mapping tab

NOTE:
The settings in this tab will affect the job set globally.

Here, the following settings can be made:

- Determination of which of the variably usable I/Os are to be used as input or output.
  Pin 05 pink, pin 06 yellow, pin 07 black (LED B), and pin 08 gray (LED C) can be used as input or output.
- Assignment of functions to the inputs / outputs.
  The respective list boxes show the functions available for this input or output and can also be defined here. The functions listed under "Sole functions" are **only** available via this pin / line.



Pin / color	Input	Output	NO / NC	Function	Unique function
03 WH	☑			H/W Trigger	H/W Trigger
10 VT	☑			no function / undefined	Encoder A+
12 RDBU (A)	□	☑	N.O.	Ejector / Result	Ejector / Result
09 RD	□	☑	N.O.	Result	
05 PK	□	☑	N.O.	Result	Encoder B+
06 YE	□	☑	N.O.	Result	
07 BK (B)	□	☑	N.O.	Result	
08 GY (C)	□	☑	N.O.	Result	

Fig. 283: *Output, I/O mapping tab*

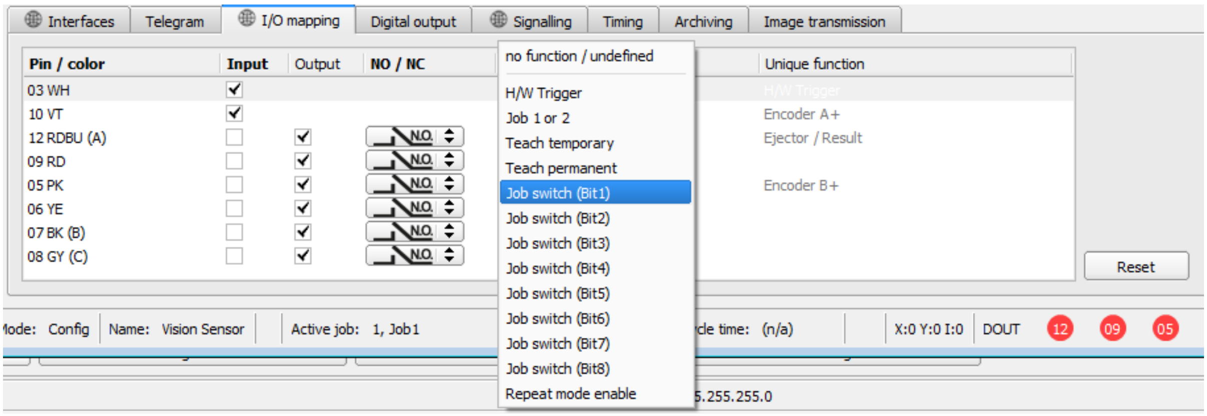
Functions of inputs

Function	Description
H/W trigger	Hardware Trigger (only available via Pin 03 white)
Encoder A+	Input for encoder, Track A+ (only available via Pin 10 purple)
Encoder B+	Input for encoder, Track B+ (only available via Pin 05 pink)
Enable trigger	Function for enabling or suppressing trigger signals. Reading this function takes about 1 ms. This creates a pause in which a trigger signal is ignored, even though the Enable Trigger signal is present.
Job 1 or 2	Job change between Job 1 and Job 2, depending on level at this input. Low = Job 1, High = Job 2.

Function	Description
Teach temporarily / Teach permanently	Teaching of all detectors. As soon as a high signal is applied **and** a trigger occurs, teach-in starts. Temporary: Storage in RAM, i.e. fleeting after reset, Permanent: Storage in Flash, i.e. permanent, even after reset
Job switch (BitX), binary coded	Job switching via binary bit pattern to up to 8 inputs, which can be defined for this purpose, i.e. switching between 1 to 255 jobs. Ranking of the bits according to assigned ascending input designation 1 - 8. Bit 1 = LSB. See Chapter: Job 1 … 255 via a binary input bit pattern
Repeat Mode Enable	Images are captured and evaluated as long as: There is a high level at this input and none of the following termination criteria is met: - "Job result" = positive (adjustable under "Output" / "Digital output") - "Max. cycle time" is not fulfilled (if activated). If the input Repeat Mode Enable is used, it also acts as Trigger Enable. I.e., triggers will be accepted and processed only if there is a high level at this input; see below: Input: Repeat Mode Enable, with Trigger
Multishot trigger (only if Multishot is active)	Default setting if Multishot is active, instead of H/W Trigger
No function, undefined	No function, not used

Connection Encoder

Functions that are already fully exploited appear pale gray in the listbox because they are no longer available. All input signals must have a minimum signal length of 2 ms.



Pin / color	Input	Output	NO / NC	Unique function
03 WH	☑			H/W Trigger
10 VT	☑			Encoder A+
12 RDBU (A)	□	☑	N.O.	Ejector / Result
09 RD	□	☑	N.O.	
05 PK	□	☑	N.O.	Encoder B+
06 YE	□	☑	N.O.	
07 BK (B)	□	☑	N.O.	
08 GY (C)	□	☑	N.O.	

lode: Config | Name: Vision Sensor | Active job: 1, Job1 | cle time: (n/a) | X:0 Y:0 I:0 | DOUT 12 09 05

5.255.255.0

Fig. 284: Output, I/O mapping tab, Inputs

If both tracks A+ and B+ are used, forward/backward differentiation or counting is possible. The encoder inputs can process a maximum frequency of 40 kHz.

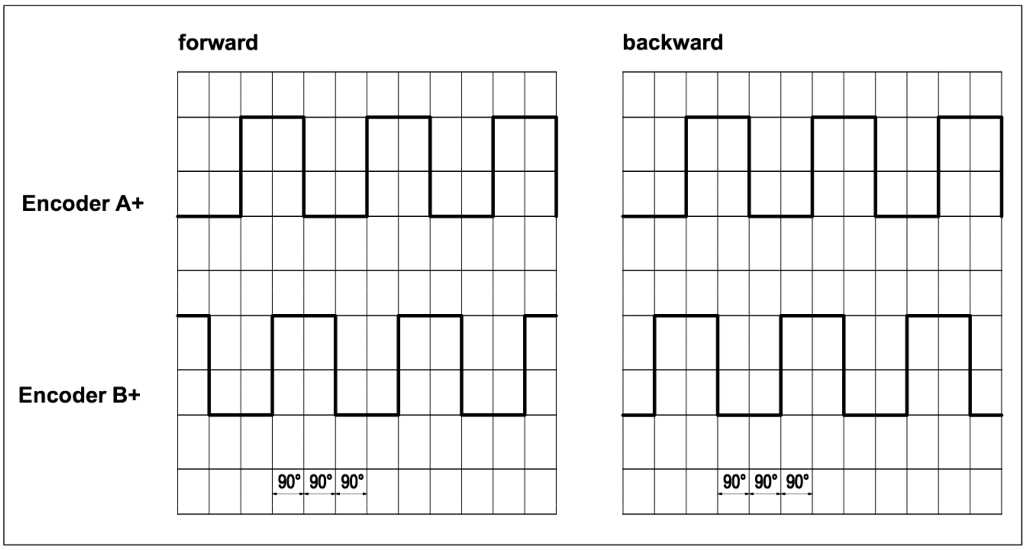


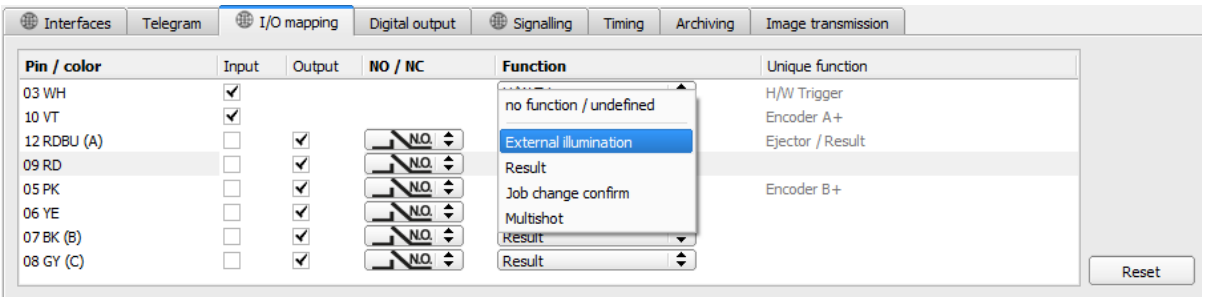
Fig. 285: Encoder tracks A+ / B+

Functions of outputs

The output’s default state is defined with NO / NC:

- NO (normally open) = Open if logical expression = False
- NC (normally closed) = Closed if logical expression = False

Function	Description
Ejector	Special ejector output can be loaded with up to 100 mA (all other outputs = 50 mA), only available via Pin 12 RDBU (corresponds to indicator LED "A").
Result	Result output; each of the result outputs defined here can be assigned a detector result or a combination of detector results in the tab "I /O Logic".
Confirmation job change	When changing jobs with the digital I/O ("Job pin X, binary-coded"), a falling / rising edge can be configured here in order to confirm a successful change. The high edge is set after the new job content is loaded and active, i.e. at the same time as the high edge of the ready signal after switching (see Timing ...). The high level stops for 20 ms and is then deleted again. If the switchover was unsuccessful, no high level is output and the signal is permanently low.
External illumination	If this setting is selected (only available via Pin 09 RD), external lighting can be connected / triggered here.
No function, undefined	No function, not used



Pin / color	Input	Output	NO / NC	Function	Unique function
03 WH	☑				H/W Trigger
10 VT	☑				Encoder A+
12 RDBU (A)	□	☑	N.O.		Ejector / Result
09 RD	□	☑	N.O.		
05 PK	□	☑	N.O.		Encoder B+
06 YE	□	☑	N.O.		
07 BK (B)	□	☑	N.O.	Result	
08 GY (C)	□	☑	N.O.	Result	

Fig. 286: Output, I/O mapping tab, Output

There are two permanently defined outputs:

- Ready: Indicates whether the sensor is ready to receive a trigger / next evaluation.
- Valid: Indicates whether the data is valid at the outputs.

N.O. / N.C.

For each output, you can define which switching function should be used: Normally open (N.O.) or normally closed (N.C.).

Programmable functions of the digital inputs:

In operation with a process control, the following functions can be performed via the inputs:

- Inactive
- Enable / Disable
- Load job (binary coded)
- Teach temporarily
- Teach permanently

Description of different cases with signal diagram.

All signals shown here are based on the setting "PNP"

Input: "Enable trigger"

Activates the trigger input of the sensor (high signal) or blocks the hardware trigger (low signal).

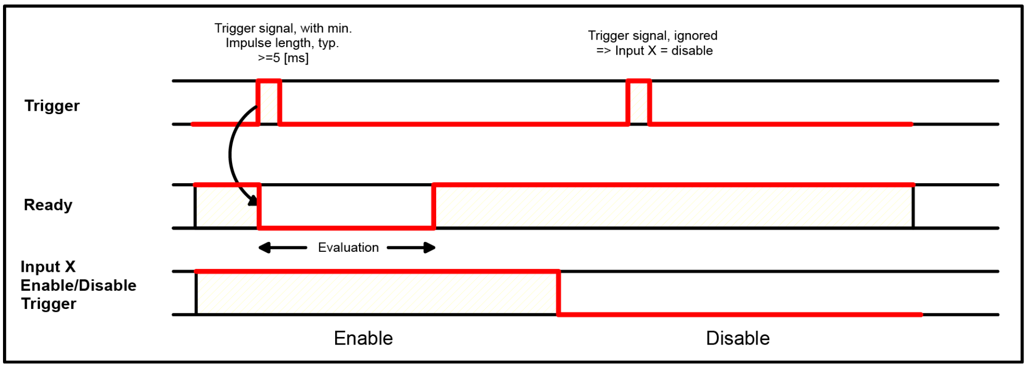


Fig. 287: Input timing, Enable trigger

Input: Job change via binary signals, or via function Job 1 or 2

Binary job change via up to 5 inputs (Job 1 - max. 255):

When changing the binary input signals, Ready is set to Low. Ready remains Low until the switchover to the new job occurs. If the optional job change confirmation signal is used, it will occur after the job change, and Ready will only become High again after that. During job switching, no trigger signals may be sent. The level change of the associated inputs must be made simultaneously (within a maximum of 10 ms, all levels must be stable). If the level changes of individual inputs are further apart, several job switches are executed one after the other if necessary).

Job change through function Job 1 or 2:

When changing the level of the correspondingly defined input, Ready is set to Low. Ready remains Low until the switchover to the new job occurs. If the optional job change confirmation signal is used, it will occur after the job change, and Ready will only become High again after that. During job switching, no trigger signals may be sent. With Job 1 or 2, low level switches to Job 1 and high level to Job 2.

Difference between binary signals and Job 1 or 2:

When using the switchover via binary signals, the desired job number must always be coded in binary code. At least 2 inputs must be used for 2 jobs.

With Job 1 or 2, low level switches to Job 1 and high level to Job 2. In this way, two jobs can be selected via one input.

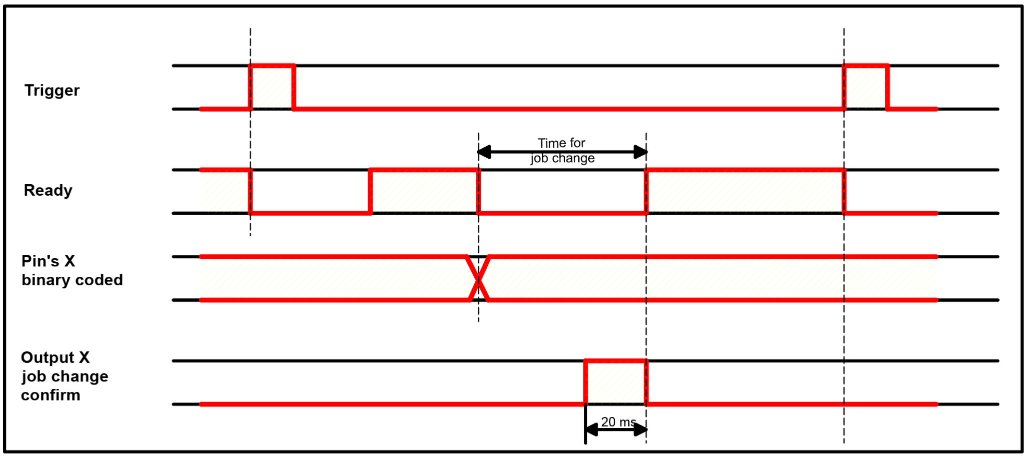


***Fig. 288:** Input timing, Job change via Binary / Job 1 or 2*

Input: Teach temporary / permanent

For re-teaching samples of all detectors of the current job. A rising edge initiates teaching, whereby the high level must be present at least until the next trigger so that an image of a test piece can be recorded in the correct position. Ready is set to Low and remains low until teaching is complete. Depending on the setting, storage is either temporary (in RAM only) or permanent (in Flash).

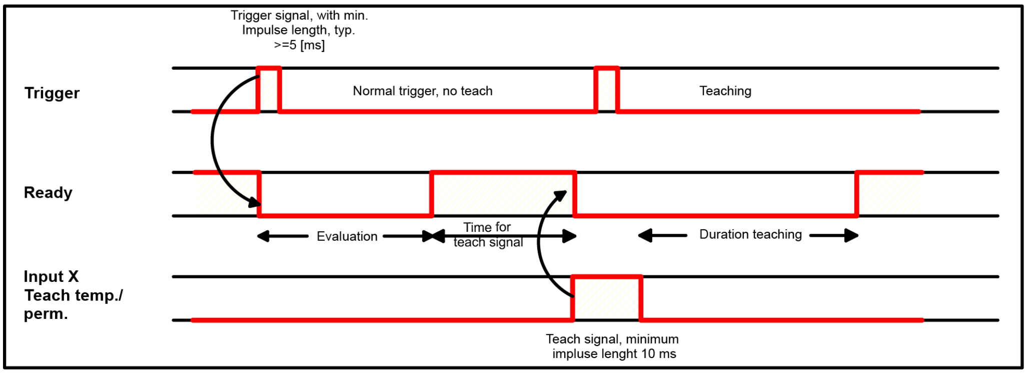

Trigger signal, with min. Impulse length, typ. >=5 [ms]
Trigger
Normal trigger, no teach
Teaching
Ready
Evaluation
Time for teach signal
Duration teaching
Input X Teach temp./ perm.
Teach signal, minimum impluse lenght 10 ms

***Fig. 289:** Input timing, Teach*

NOTE:
The functions Job 1 or 2 and Teach-in (temporary/permanent) can only be used in trigger mode.

Input: Repeat Mode Enable, with Trigger

Images are acquired and evaluated as long as a high level is present at this input and none of the following criteria is fulfilled:

- "Job result" = positive (adjustable under "Output" / "Digital output")
- - "Max. cycle time" is not fulfilled (if activated).

If the Repeat Mode Enable input is used, it also acts as a Trigger Enable. This means only if a high level is applied to this input, Trigger will be accepted and processed

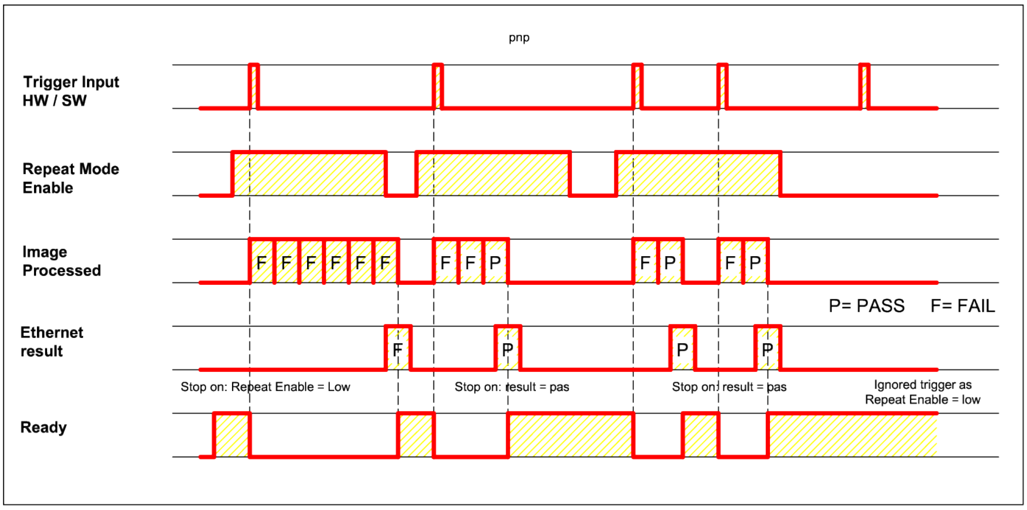


Fig. 290: Input: Repeat Mode Enable, with Trigger

Input, Repeat Mode Enable, in Freerun

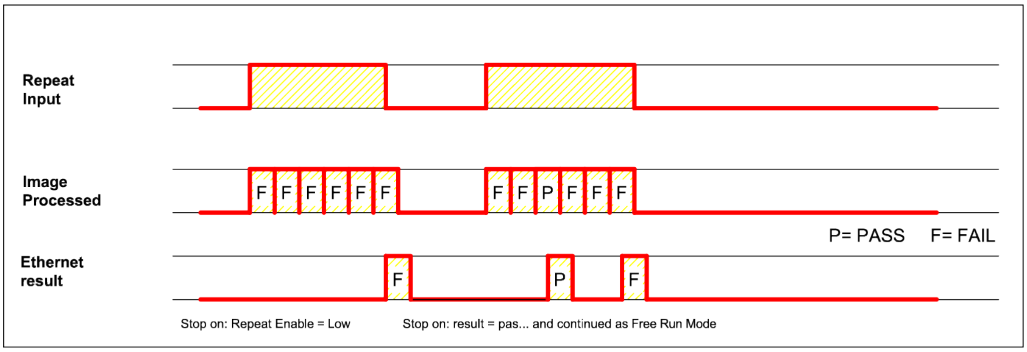
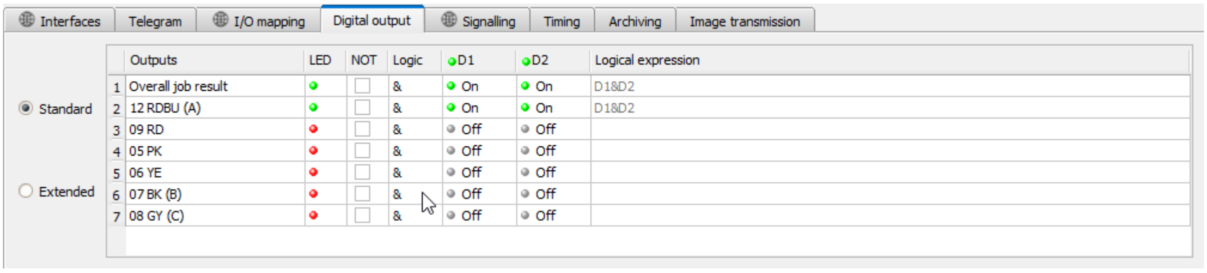

Repeat Input
Image Processed
F F F F F F
F F P F F F
P= PASS F= FAIL
Ethernet result
F
P
F
Stop on: Repeat Enable = Low
Stop on: result = pas... and continued as Free Run Mode

Fig. 291: Input, Repeat Mode Enable, in Freerun

8.4.4 Digital output tab (Digital outputs / logic)

In this tab you can define the switching behavior and the logical connection of the individual detectors with the digital outputs. The number of outputs depends on the settings under the tab I/O mapping. Additionally, an I/O extension can be controlled via the serial interface.



● Standard
○ Extended

	Outputs	LED	NOT	Logic	D1	D2	Logical expression
1	Overall job result	🟢	□	&	On	On	D1&D2
2	12 RDBU (A)	🟢	□	&	On	On	D1&D2
3	09 RD	🔴	□	&	Off	Off	
4	05 PK	🔴	□	&	Off	Off	
5	06 YE	🔴	□	&	Off	Off	
6	07 BK (B)	🔴	□	&	Off	Off	
7	08 GY (C)	🔴	□	&	Off	Off	

Fig. 292: *Output, Digital output tab, Logic*

Select logical combination of detectors for each output:

For each pin (output), there are the following possibilities:

Parameter	Function
Overall job result	No physical output. Affects logic for recorder, statistic, and archiving functions
Invert	Invert total result from the following settings for this pin (output)
Mode	Standard: Several detectors can be combined into a logical expression using the logical operators AND (&) / OR (|) / NOT (!). Advanced: The logical formula for combining the detectors can be freely created.
NOT	Select: Operator NOT (!)
Logic	Select: Operator AND (&) / OR (|)
D1 - D ...	Depending on the number of activated detectors, all detectors are inserted in this list. These can be logically assigned to each listed output. Each detector can be switched on, inverted, or off for the respective pin (output).
Logical Expression	Either the logical expression compiled in standard mode is displayed, or the logical expression can be compiled here in advanced mode.

Defining logical connection:

Define the logical link between the test results of the individual detectors and the status of the selected output. You have two input possibilities:

- Standard mode (checkboxes and operators)
- Extended mode (formulas)

NOTE:
If an external multishot illumination is connected, the pins: 09, 06, 07, and 08 can no longer be assigned with output signals.

8.4.4.1 Logical connection – Standard mode

In standard mode, connection of the detector test results for the selected output is made via the radio buttons Operator and the Checkboxes in the detector drop-down list. The result is displayed in the field "Logical formula" (not editable).

Connecting results:

1. In the Operator field, select the logical operator for linking the detectors in the drop-down list.

2. In the drop-down list, activate the detectors that should contribute to the result (tick in the Active column).

By activating the column "Inverted", you can invert the respective detector result.

The entry in the Result column changes accordingly.

Examples:

Here, the detector results can only be linked by a logical operation such as:

- (D1&D2&D3) or
- !((!D1)|D2|D3), etc.

NOTE:
If a detector is assigned to an image acquisition (see "Multiple Image Acquisition" Chapter Cycle time tab), its result in the other images does not affect the result of the combination.

8.4.4.2 Logical connection - extended mode

In extended mode, the combination of detector test results for the selected output is defined by direct input of a logical formula. For this you have the operators AND, OR, and NOT as well as parentheses.

To edit the formula, please use the following characters for the logical operators:

- "&" for AND
- "|" for OR
- "!" "!" for NOT

Examples:

Here, logical expressions of any complexity can be created, e.g.:

- (D1&D2)|(D3&D4)
- !((D1|D2)&(D3|D4))
- (D1|D2)&(D3|D4)&(D5|D6)

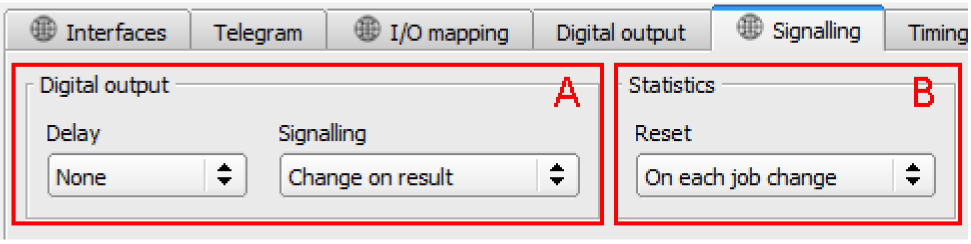
etc.

NOTE:
If a detector is assigned to an image acquisition (see "Multiple Image Acquisition" Chapter Cycle time tab), its result is set to logical "0" in the remaining image acquisitions. The result of the combination must be adjusted accordingly.

8.4.5 Signalling tab

NOTE:
The settings in this tab will affect the job set globally.

The Signalling tab can be used to configure the settings for statistics and for the digital outputs.



Digital output

Delay: None

Signalling: Change on result

Statistics

Reset: On each job change

__Fig. 293:__ Output, Signalling tab

A: Digital outputs

Parameter	Function
Delay	You can choose to either delay all outputs or only the ejector output.
Signaling	Result outputs can be reset based on various settings / events: • Change on next result (default): The output changes its level according to the logical result only when the next result is obtained. Typically used in switch control for e.g. sorting, etc. • Change on trigger: The output will be set to "inactive" (low when using PNP mode) when the next trigger occurs. Typically used for operation on a PLC. • Duration of result: The output will switch back to an inactive state after the result duration (in ms) configured here elapses. Typically used for e.g. pneumatic ejectors (blower)

B: Statistics

Parameter	Function
Reset	Used to select whether the statistics will be reset with ever job change or only with "Start sensor."

ATTENTION:
For job change and change from run to config mode, the following special states apply: The buffer of the delayed outputs is deleted.

Digital outputs:

These are reset to the default settings (defaults) when changing jobs and changing the operating mode from "Run" to "Config". The basic settings are defined by "Invert" in the SensoConfig/Output/Digital output tab. The selection "Inverse" inverts the basic setting of the digital output and the result at the same time.

Ready and Valid

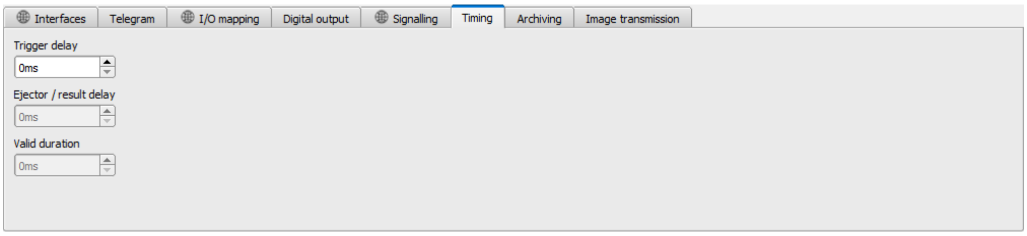
- Ready signals when High, readiness for new image acquisition.
- Valid signals when High that results at the outputs are valid.

PNP or NPN operating mode

All examples described here are executed in operating mode "PNP". If the setting "NPN" is set, the examples apply analogously with reversed levels.

8.4.6 Timing tab

This tab can be used to configure the time response characteristics for the selected output signal. All timing parameters are specified either in milliseconds or (if enabled) encoder pulses. Encoders can be enabled in the I/O mapping tab.



Trigger delay: 0ms

Ejector / result delay: 0ms

Valid duration: 0ms

Fig. 294: *Output, Timing tab*

Parameter description:

Parameter	Function
Trigger delay	Time between the trigger and the start of the image acquisition operation in ms or encoder pulses. The maximum possible value is 3000 ms / encoder pulses. How the delay setting will work if you use: • H/W Trigger (digital input): This delay is effective. • Trigger (via Ethernet, PROFINET): The delay will not be applied. The image acquisition operation will be carried out directly after the trigger.
Ejector / result delay	Time between the trigger and the presence of the event level, in ms or encoder pulses. The maximum number of components between the trigger and the ejector is 20 (buffer size). The maximum possible value is 3000 ms / encoder pulses. How the delay setting will work if you use: • H/W Trigger (digital input): This delay is effective and starts with the H/W trigger. • Trigger (via Ethernet, PROFINET): This delay is effective but only starts after the image is processed (not with the trigger!). • Please select Delay in the Signalling tab.
Duration of result	Duration of result signal in ms or encoder pulses. Maximum value of 3000 ms / encoder pulses. Please select Result duration in the Signalling tab.

ATTENTION:
When there are job changes, and when there is a change from Run mode to Config mode, the delayed output buffer will be cleared.

Ready and Valid

- Ready signals when High, readiness for new image acquisition.
- Valid signals when High that results at the outputs are valid.

The following cases in the time behavior can be distinguished:

All examples described here are executed in operating mode "PNP". If the setting "NPN" is set, the examples apply analogously with reversed levels.

Normal trigger without the use of delay times:

Flow: (in this case: Change on next)

- Rising edge at Trigger input (Pin03 WH)
- Consequence of Trigger = High: Ready = Low, and Valid = Low
- After the vision sensor has evaluated the image and the corresponding results are available, all defined outputs switch to the corresponding logical states and Ready and Valid go back to High level (outputs valid, vision sensor ready for next evaluation)

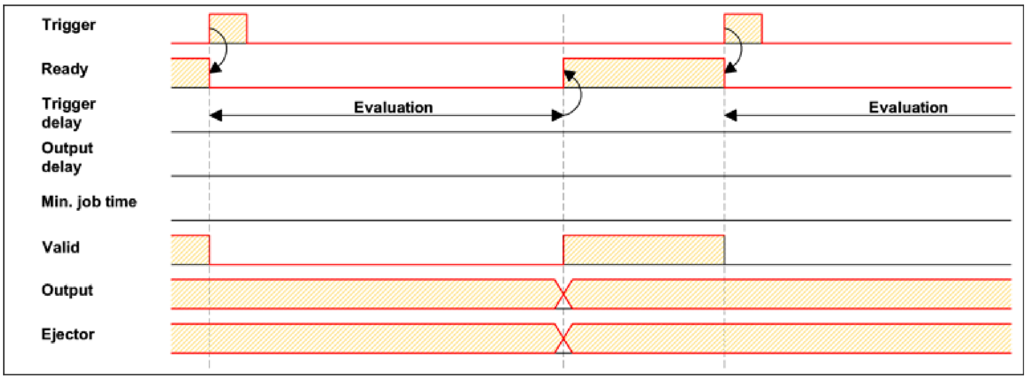


Fig. 295: Digital outputs timing, standard sequence with normal trigger

Trigger delay active

The trigger delay will only be applied to hardware trigger

This setting is used to selectively delay image acquisition / start of the evaluation compared to the actual physical trigger, which is e.g. triggered by a trigger photocell or the machine control. This allows the fine adjustment of the trigger time without any changes to the mechanics or the control program.

Procedure:

Image is acquired after the trigger delay time is elapsed. The cycle time is: Trigger delay time + evaluation time.

See SensoConfig/Output/Timing/Trigger/Delay

- Rising edge at Trigger input (Pin03 WH)
- As a result of Trigger = High: Ready = Low, Valid = Low, all defined result outputs = Low (signaling = change at trigger)
- Before the image is taken for evaluation, the set trigger delay time (trigger delay) elapses.
- Now the evaluation starts. As soon as the corresponding results are available, all defined outputs change to the corresponding logical states. Ready and Valid will switch back to a high state (outputs valid, VISOR® ready for next evaluation).

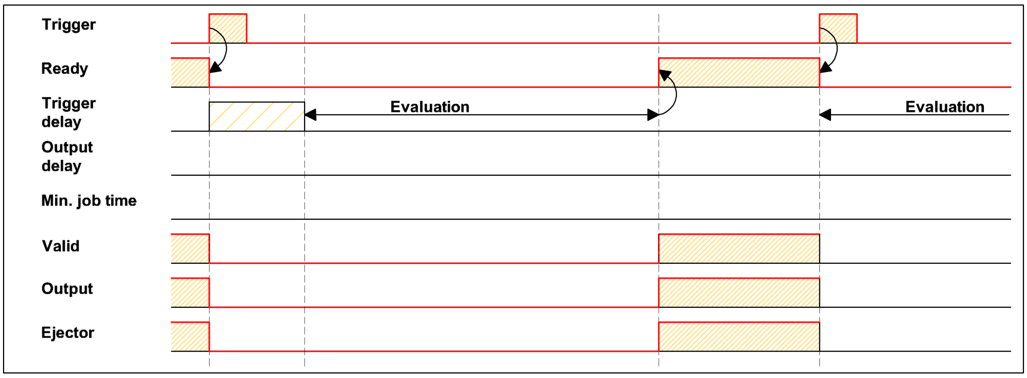


Fig. 296: *Digital outputs timing, Trigger delay*

Trigger delay + Result delay acting on ejector:

(ejector only in this case)

The trigger delay will only be applied to hardware trigger

The result delay (whether for all outputs or only ejectors) is used to fine tune e.g. the ejector time regardless of the evaluation time, especially since this may also have slight fluctuations.

Procedure:

Image is acquired after the trigger delay time is elapsed. In addition, the result delay works. In this example, however: only on the ejector output (Pin 12 RDBU).

For the defined result outputs except the ejector output, the cycle time is: trigger delay + evaluation time

The cycle time for the ejector output is: The result delay itself (counted from trigger time, only meaningful if longer than sum times!). See SensoConfig/Output/Timing/Digital Output/Delay.

- Rising edge at Trigger input (Pin03 WH)
- As a result of Trigger = High: Ready = Low, Valid = Low, all defined result outputs = Low. Except ejector, a fixed duration of results is defined for this,
- Before the image is taken for evaluation, the set trigger delay time (trigger delay) elapses.
- Now the evaluation starts. As soon as the corresponding results are available, all defined outputs (except ejector) change to the corresponding logical states. Ready and Valid return to a high level.
- In this operation mode, only the Ejector output is set after the Result delay has elapsed. The ejector output is also provided with a result duration in this example and is therefore set to Inactive after this result duration.

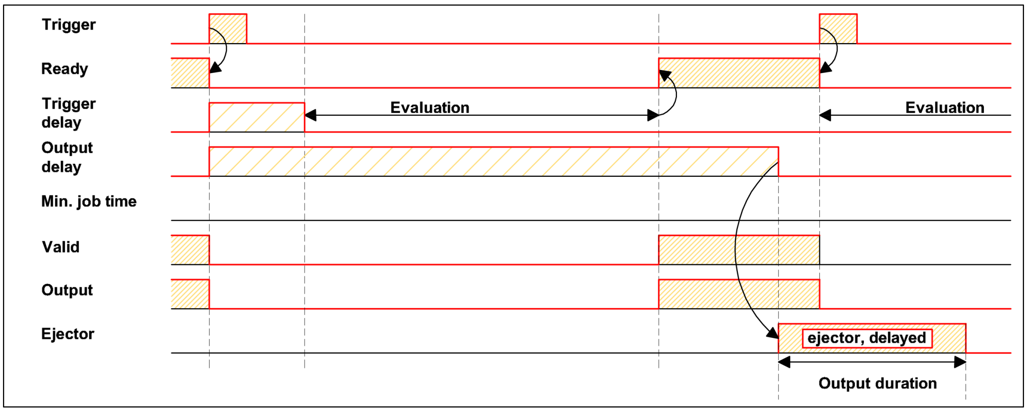


Fig. 297: Digital outputs timing, Result delay ejector

Trigger delay + Result delay acting on all outputs:

The trigger delay will only be applied to hardware trigger

The result delay (whether for all outputs or only ejectors) is used to fine tune e.g. the ejector time regardless of the evaluation time, especially since this may also have slight fluctuations.

Procedure:

Image is acquired after the trigger delay time is elapsed. Furthermore, in this example, the result delay affects ALL defined result outputs.

The cycle time is the same for all result outputs: The result delay itself (counted from trigger time, only meaningful if longer than sum of trigger delay + evaluation time!). See SensoConfig/Output/Timing/Digital Output/Delay.

- Rising edge at Trigger input (Pin03 WH)
- Consequence of Trigger = High: Ready = Low, Valid = Low.
- Before the image is taken for evaluation, the set trigger delay time (trigger delay) elapses.
- Now the evaluation starts. After the results are available, only the Ready signal is set to High again (ready for the next evaluation). Otherwise, the delay in the result delay is still awaited. Only then do all defined outputs change to the corresponding logical states. Valid goes back to high level (Valid = High: result outputs valid, Signaling = Change in next result).

In this operating mode, only the "Ready" signal changes after elapse of the trigger delay + image acquisition + evaluation time. Ready = High: Ready for the next evaluation. The reason this makes sense is that the VISOR® is already ready for the next evaluation regardless of whether the outputs are set later on.

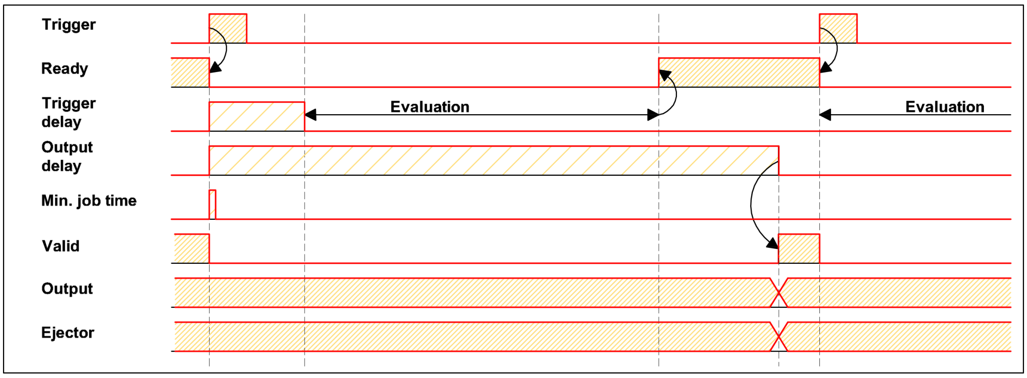


Fig. 298: Digital outputs timing, Result delay all outputs

Result duration, e.g., acting on all outputs:

This time setting is used to achieve an output pulse with a defined length, such as for controlling a pneumatic ejector (blower) in the case of a bad part, or the like.

All defined result outputs are reset to low level (inactive in PNP mode) after the result duration in ms is elapsed.

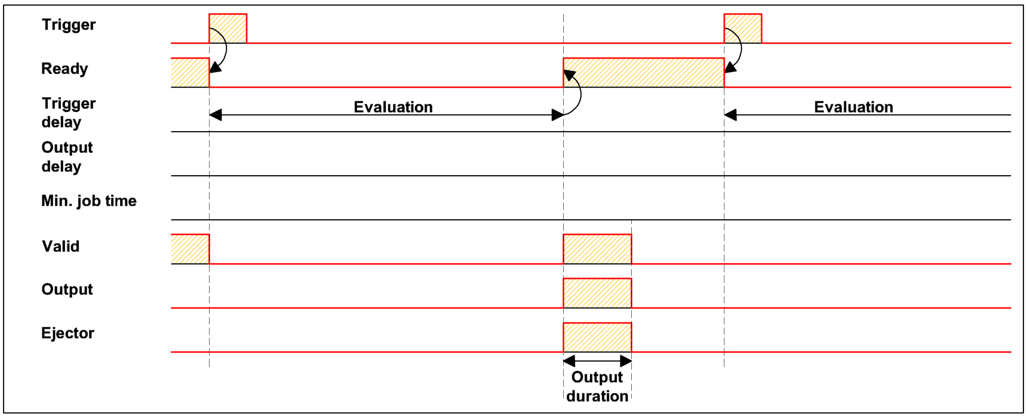


Fig. 299: Digital output timing, Result duration

Cycle time (min., max.) active:

(Here: Signaling: Change at Trigger)

Parameters for controlling the execution time of a job.

The minimum execution time can be used to suppress multiple triggers and can affect the LED performance. (i.e. if another trigger is received within the minimum job time, it will be ignored)

The maximum execution time is used to cancel a job after a defined time. The result of the job after termination is always "not OK". The maximum execution time should always be greater than the time required for an evaluation.

The cycle time measures the time from the trigger to the setting of the digital switching outputs. If the cycle time should be limited, e.g. because the machine cycle must not be exceeded, the value for the maximum cycle time must be limited accordingly. The result of all unfinished detectors up to this time is set to faulty. When choosing the maximum cycle time, it must be taken into account that this is not adhered to strictly. However, depending on the detector that has just been executed, it can take several milliseconds before it can break off. It is recommended that this maximum cycle time over the actual execution time is checked and the set maximum cycle time is reduced accordingly.

Procedure:

All outputs and the signal "Valid" (Outputs valid) are set directly after evaluation.

However, the signal "Ready" (Ready for next evaluation) is set only after the min. Job time has elapsed, and thus triggers are only accepted for the next evaluation from this moment.

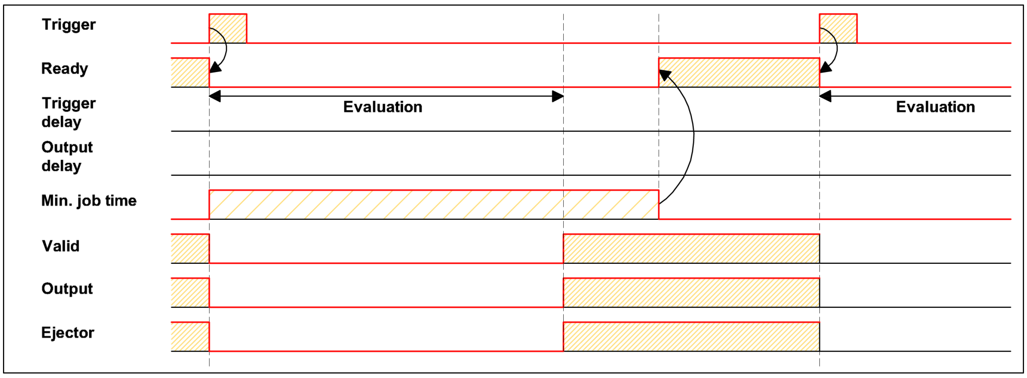


Fig. 300: Digital outputs timing, Min. job time

Multiple result delay for ejector

This operating mode is used if, between the trigger/evaluation for test part A and its ejection, there is so much time / such a large distance that the VISOR® already has to check n (max. of 20) additional test parts and manage their accordingly belated ejection time.

(only available in mode: SensoConfig/Output/Timing/delay: "ejector / ejector / result delay only")

Here: Signaling = Result duration (alternatively also "Change with next result" applicable)

A maximum of 20 components may fit between the trigger and the ejector.

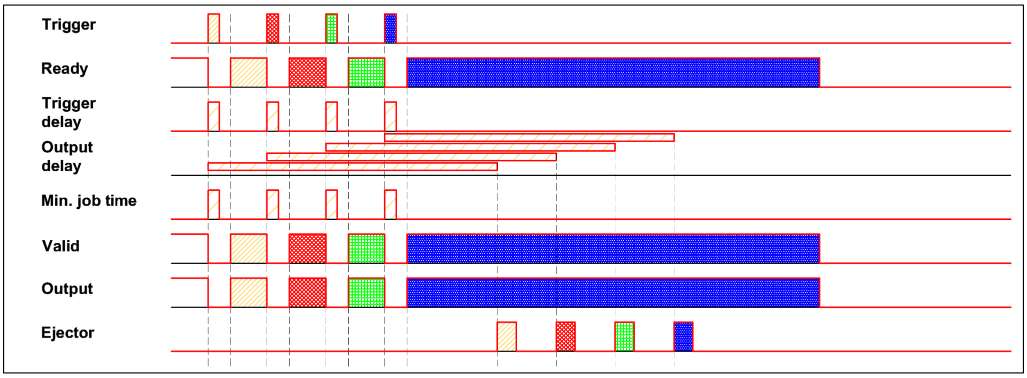
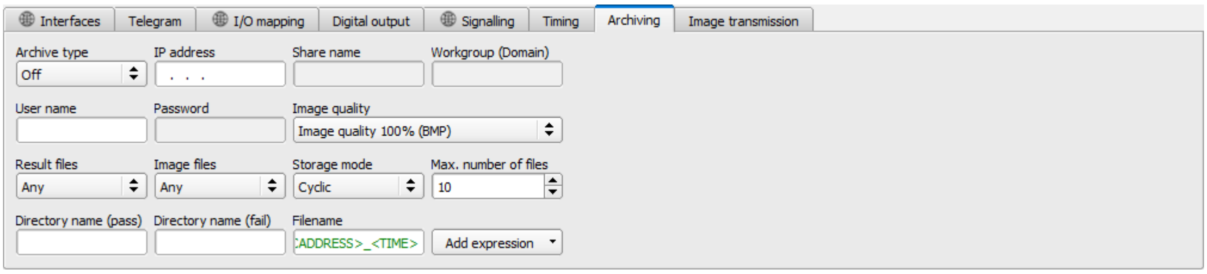


Fig. 301: Digital output timing, Multiple result delay ejector

8.4.7 Archiving tab

In the Archiving tab, you can configure the archiving of the data.



Archive type: Off
IP address: . . .
Share name
Workgroup (Domain)
User name
Password
Image quality: Image quality 100% (BMP)
Result files: Any
Image files: Any
Storage mode: Cyclic
Max. number of files: 10
Directory name (pass)
Directory name (fail)
Filename: :ADDRESS>_<TIME>

Fig. 302: Output, Archiving tab

Parameter description:

Parameter	Function and setting options
Archive type	**Off**: No archiving. **FTP**: Archiving on an FTP server. Note: The storage mode "Cyclic" requires the FTP extension RFC3659. **SFTP**: Archiving on an SFTP server with the SSH FTP protocol. **SMB**: Archiving on a drive with the SMB service (Server Message Block, up to Version 3.1.1). **NOTE:** When using archive servers in other subnets, first set the gateway in SensoFind.
IP address	IP address of the server
Sharing name	Share name of the drive on the server for archive type SMB

Parameter	Function and setting options
Workgroup (domain name)	Workgroup / domain name of the server (Optional)
User name	Username for FTP / SFTP / SMB connection.
Password	Password for FTP / SFTP / SMB connection.
Image quality	Selection of the compression rate. With reduced image quality, the file size is reduced. This speeds up image transfer to the server and reduces the amount of memory required on the server.
Result files	If the result file is activated, all results specified under "Output" / "Telegram" are additionally logged to a .csv file. A file is created for each evaluation (trigger). The files are filed by storage mode and file name. The following options are available via the Result files selection list: **None**: disabled. No files are written. **Any / Pass / Fail**: All images, only the images of the good parts or only the images of the bad parts, depending on the overall result of the job as defined in "Output" / "Digital output" / "Job result".
Image files	Activates archiving of images. The files are filed by storage mode and file name. The following options are available via the Image Files selection list: **None**: disabled. No files are written. **Any / Pass / Fail**: All images, only the images of the good parts or only the images of the bad parts, depending on the overall result of the job as defined in "Output" / "Digital output" / "Job result". **NOTE:** ℹ • Images are stored without preprocessing settings, but with the settings for the arrangement (e.g. rotated or mirrored) • FTP, SFTP, and SMB always save images without overlays only. To store images with overlays, please use SensoView. Additional information: VISOR® Software – SensoView
Storage mode	The following options are available via the Storage Mode selection list: **Limited**: When the maximum number of files (parameter "Max. number of files") is reached, the transfer is stopped. **Unlimited**: Files are written until the target drive is full. **Cyclic**: After reaching the maximum number of files (parameter "Max. number of files") the oldest is overwritten by the newest. The criterion for this is the creation date. Depending on the archive type and server, the time resolution varies. If the time resolution is below the image acquisition rate, the number of files in the destination directory may differ from the "Max. number of files" setting. **NOTE:** ℹ • All files with the currently selected extension (*.bmp, *.jpg, *.csv) will be overwritten. • By adding the MAC address ("MACADDRESS") to the file name, it is possible to prevent files with the same extension from another source from being overwritten. • The storage mode "Cyclic" requires the FTP extension RFC3659.
Max. number of files	Maximum number of data records that can be stored in the target directory.

Parameter	Function and setting options
Directory name (pass)	Via Directory name (good parts / bad parts) a subfolder can be created or selected in the target directory on the server, e.g. "TESTPASS" for ...[server target directory]\TESTPASS.
Directory name (fail)	
File name	Name for image files and result files. This name can consist of constants as well as variable contents. Maximum length: 100 characters. Files with a longer filename are not stored on the server.
Add expression	Extension of the file name by a variable part (like e.g. date or time), see also table below

The following table shows the expressions that can be added to the file name.

Expression	Description	Example
TIME	HHhMMmSSsSSSms	09h05m11s034ms
HOUR	hh	09
MIN	mm	05
SEC	ss	11
MSEC	sss	034
DATE	YYYY-MM-DD	2011-09-21
YEAR	YYYY	2011
2YEAR	YY	11 (for 2011)
MONTH	MM	09
DAY	DD	21
STRINGID	"Data" entry from extended trigger request "TRX"	Part 34
COUNTER	Counter from statistic	3824
XXCOUNTER	Counter taken from statistics with a defined number of digits. XX indicates the number of displayed digits and can accept values from 01 to 09 ℹ **NOTE:** If the number of digits of the counter is too small, it is prefilled with "0". If the number of digits of the counter is too large, digits are discarded.	06COUNTER → 003824
result	Overall result of job	Pass or Fail
SENSOR NAME	As specified in SensoFind	
JOB NAME	As specified in SensoConfig	
Detectors	Access to detector or Alignment results Further information on the results of the individual detectors can be found in the Communications manual: Chapters Data output in ASCII / Data output in BINARY	

8.4.8 Image transmission tab

In the Image transmission tab, the image transmission and / or the image recorder and the Ram disk can be activated.

ATTENTION:
When this icon appears on the life image, it is indicating the image visualization / image storage on the PC is running slower than the image processing on the VISOR®. Not all images taken by the VISOR® are displayed anymore. This may lead to image loss when using poor image archiving.
If the icon appears frequently, programs opened on the PC should be closed in the background to provide more PC performance.

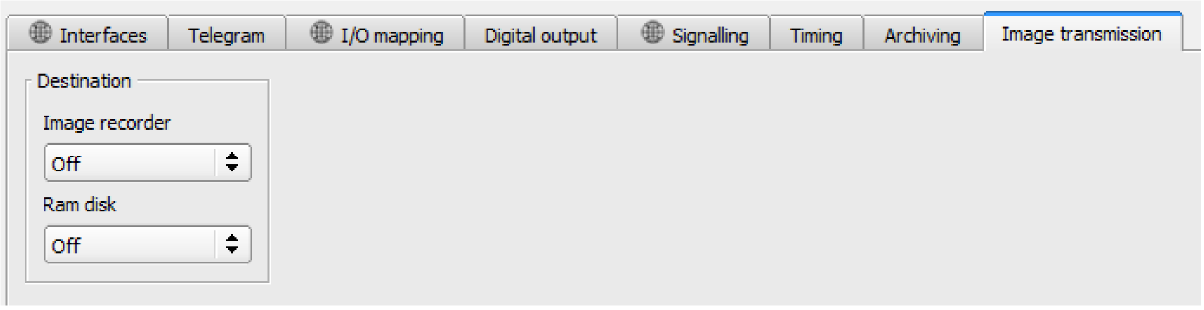


Image recorder: Off

Ram disk: Off

Fig. 303: Output, Image transmission tab

Parameter description:

Parameter	Function
Image recorder	Storage of max. 10 images in the sensor's internal ring buffer. Setting options: Off, Any, Pass, Fail
RAM disk	Storage of the last image in the internal RAM memory; this image can be fetched from an sFTP client. Setting options: Off, Any, Pass, Fail. The image is stored in the RAM of the VISOR® under the name "image.bmp" in the directory /tmp/results/. Access data for SFTP client: Username: *user*, Password: *user* When switched on, the result data (all defined in "Output / Data Output", with separator ";") can be obtained in the same way via the "result.csv" file.

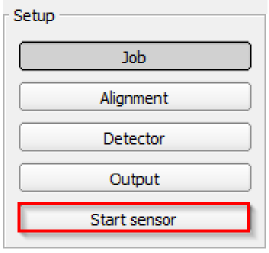
Different types of image archiving

Access	Description	Max. number of images	Image filter	Drawings	Compression
Image recorder in VISOR® (RAM)	In Run mode, images are stored on the VISOR®. The images can be transferred to a PC by SensoConfig or SensoView.	10	as predefined in the settings "Filter".	No	No
SensoView Archiving / SensoConfig saves image	Images transferred tSensoViewo can be stored on hard disk of PC.	unlimited (Limit is size of hard disk in PC)	as predefined in the settings "Filter".	Yes / No can be selected	Yes (optional)

Access	Description	Max. number of images	Image filter	Drawings	Compression
Saving of filmstrips in SensoConfig	Current images from filmstrip can be saved as filmstrip (*.flm) or as bitmap (*.bmp) on hard disk of PC.	50	without filtering	No	No
Last image in VISOR®	The last image will be stored on the RAM drive of the VISOR® and can be fetched from "directory /tmp/results" via FTP.	1	without filtering	No	No
Images archived via FTP, SFTP, or SMB	Images archived via FTP, SFTP, or SMB	unlimited (Limit is size of hard disk in PC)	selectable with / without filtering	No	Yes (optional)
Get Image Request	The last image of the VISOR® can be transmitted to a program on the PLC or PC with the "GetImage" command	unlimited (Limit is size of hard disk in PC)	as predefined in the settings "Filter".	No	No

8.5 Setup Start sensor

This function sets the sensor to Run mode and executes the job.



To start the sensor, click on "Start sensor".
The data will be transferred to the vision sensor and the vision sensor will be started.

NOTE:
A detector must have been created.

Starting execution of a job:

Click on the "Start sensor" button.

The active (= marked in the drop-down list) job is transferred to the sensor, stored in the sensor's non-volatile memory, and started (Run mode). This status is also automatically saved as a jobset backup (see Automatic job backup).

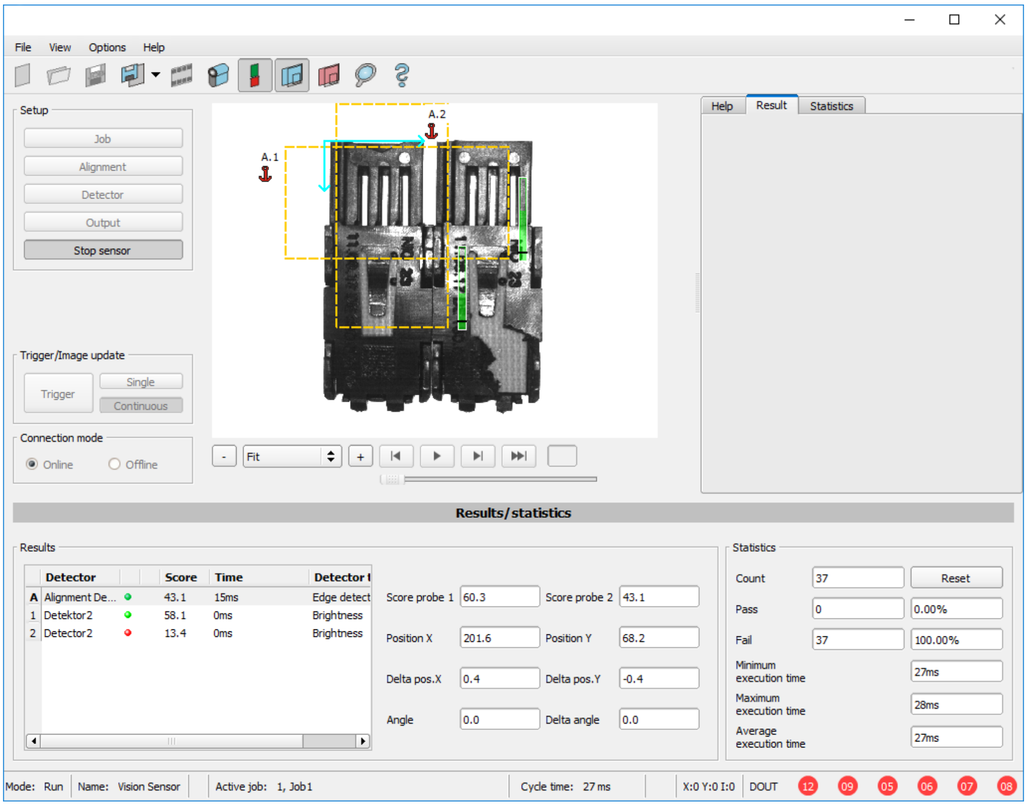
In the image window, the following are displayed: the found features, the test results for the first or selected detector in the drop-down list, and statistical parameters.

Changing detector display:

To display the inspection results for another detector, mark it in the detector drop-down list (bottom left) or click on its graphical representation in the image window.

Terminating job execution:

Click on the "Stop Sensor" button. You are now back in configuration mode and can edit your job.



Results

	Detector		Score	Time	Detector t
A	Alignment De...	🟢	43.1	15ms	Edge detect
1	Detektor2	🟢	58.1	0ms	Brightness
2	Detector2	🔴	13.4	0ms	Brightness

Score probe 1: 60.3
Score probe 2: 43.1
Position X: 201.6
Position Y: 68.2
Delta pos.X: 0.4
Delta pos.Y: -0.4
Angle: 0.0
Delta angle: 0.0

Statistics

Count: 37 Reset
Pass: 0 0.00%
Fail: 37 100.00%
Minimum execution time: 27ms
Maximum execution time: 28ms
Average execution time: 27ms

Mode: Run | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: 27 ms | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 304: *Starting the sensor*

8.6 Trigger / Image update

Select the Trigger mode you want in the job settings in the "Image acquisition" tab:

Parameter	Function
Trigger	Operation with external trigger, or "Trigger" button on the SensoConfig surface
Free run	Operation with automatically running self-trigger; the sensor supplies images with the maximum possible frequency

Select the format in which the sensor should deliver images by using the buttons in the Trigger / Image update update section:

Parameter	Function
Single	Capturing a single image; image capture occurs once when: 1. Trigger mode = Trigger: First external trigger signal or with the "Trigger" button on the SensoConfig interface 2. Trigger mode = Free run: First click on the "Single" button on the SensoConfig interface (important e.g. in setup mode)
Continuous	Continuous supply of images; image capture occurs continuously when: 1. Trigger mode = Trigger: Each external trigger or with each click on the "Trigger" button on the SensoConfig interface 2. Trigger mode = Free run: Continuous through internal self-triggering with maximum frequency

Changing the parameters exposure time, gain, lighting, or resolution in the job settings will automatically request a new image from the sensor.

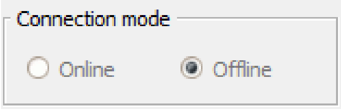
To obtain a continuously updated live image without trigger, carry out the following settings:

- Set "Free run" under Job / Image acquisition
- Set "Continuous" under "Trigger / Image update"

8.7 Connection mode

There are two operating modes available for configuring and testing the sensor, which you can select in the "Connection mode" field.

- Online mode: Configuration with connected sensor.
- Offline mode: Simulation of a sensor with the help of images stored in filmstrips.



Connection mode

○ Online ● Offline

__Fig. 305:__ Connection mode

When the sensor is connected, both modes are available; it is possible to switch between the two. If no sensor is available, it is only possible to work in Offline mode, i.e. with sensor simulation.

8.8 Displays in the image window

8.8.1 Image section and zoom

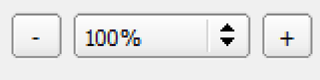


__Fig. 306:__ Zoom

Use the buttons or the drop-down menu below the image window to select the desired image section. Alternatively, the image section can be enlarged or reduced by holding down the "Ctrl" key and using the scroll wheel.

8.8.2 Graphical display of results

In the View menu, you can activate or deactivate the following graphical representations:

- Bar graph result: Displays the inspection result as a bar graph
- Overlays: Displays regions of interest, feature ranges, and position frames of detectors and Alignment
- Focusing aid: Displays image sharpness (see also Job settings)
- Enlarged display: Displays a separate, enlarged image window that can be scaled to any size via handles on the frame corners

The program SensoView offers a limited selection of these functions.

8.8.3 Controlling the image display

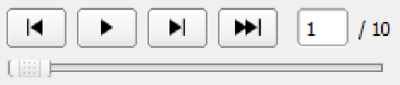


__Fig. 307:__ Image reproduction

Use the buttons and the scroll bar below the image area to control the selection and playback of saved images. The frame counter displays the number of the current image and the number of images in the active filmstrip.

Buttons	Function
|◀	Jump to previous image.
▶	Starts / Stops the reproduction of the stored images.
▶|	Jump to next image.
▶▶|	Jump to last image. The statistics are reset and all images are evaluated.

8.9 Open and save job or jobset (file)

Jobs can be loaded and stored individually or as a set of jobs as a job set. If several jobs are stored on the sensor, they form a job set which you can save as a single job as an XML file on your PC or an external storage medium.

Saving a job / job set:

1. Select "Save job as ..." from the File menu.
2. Select "Save job set under (Backup) ..." from the File menu.

Opening a job / job set:

1. Select "Load job ..." or "Load job set (Backup) ..." from the File menu.
2. Activate the button "Start Sensor" to transfer jobs to the sensor.

All the jobs stored on the sensor are deleted when a new job / job set is loaded!

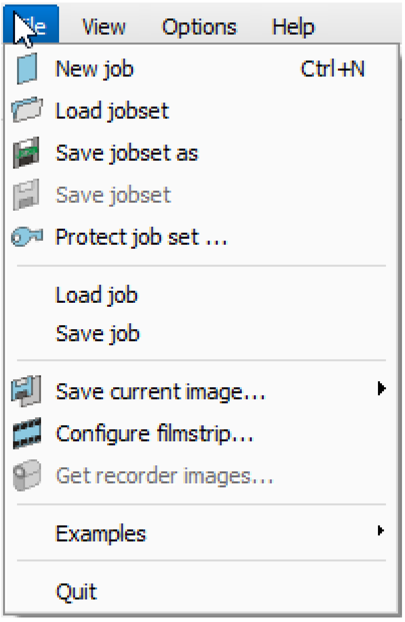


***Fig. 308:** SensoConfig Load / save jobs*

NOTE:

Job files created with software version 2.x.x.x or later have a different format than job files created with older versions.

Job files created with version 1.x.x.x can be loaded in version 2.x.x.x. The conversion is performed automatically. It may be necessary to adjust the regions of interest and fine tune the detector parameters.

Job files created with version 2.x.x.x cannot be used in version 1.x.x.x.

Automatic job backup

By clicking on "Start sensor" the current status of the jobset is transferred to the sensor. This status is also automatically saved as a jobset backup. The file can be found in the corresponding folder of the sensor type used (e.g. Allround / Robotic / ...) in the subfolder "Backup". This file is overwritten when the sensor is restarted.

- File path: \[User]\Documents\Sensopart\VISOR Vision Sensor\[SensorType]\Backup
- File name: "_last_upload_jobset.job"

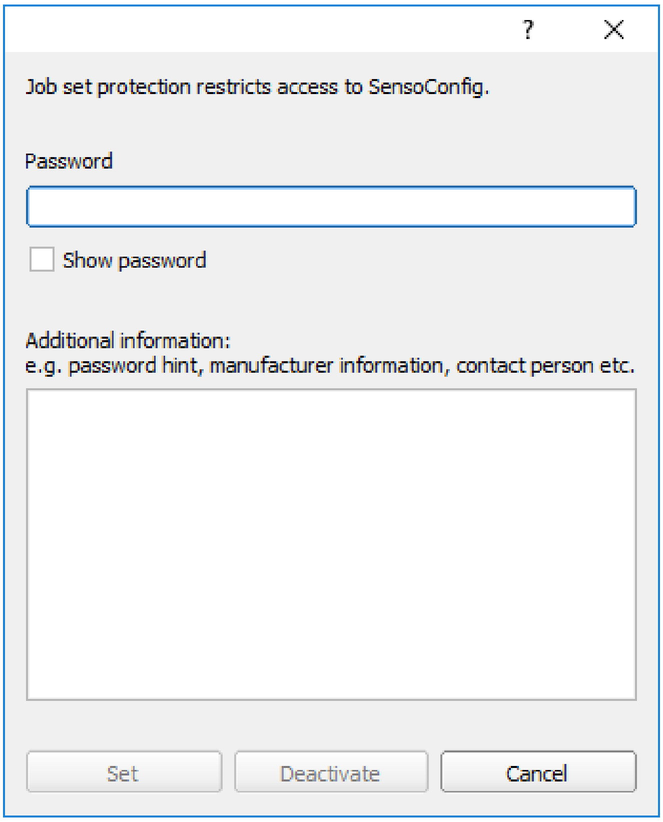
8.10 Protect jobset (file)

In the File menu of SensoConfig it is possible to protect the jobset with a password using the function "Protect job set...". Both the job set and all jobs are protected with a password. They can only be opened with SensoConfig, if the correct password has been entered. If the password is not entered correctly, the jobset cannot be displayed or changed. The VISOR® vision sensor or access to the vision sensor is not blocked, i.e. it operates normally in run mode.

ATTENTION:
There is **no** way to recover forgotten or lost passwords.
In case of forgotten or lost passwords, the entire job set must be recreated.

Assign a password

1. Select "Protect job set ..."via:"SensoConfig/Menu/Protect job set ..."
2. Enter a password and provide additional information if desired.



Job set protection restricts access to SensoConfig.

Password

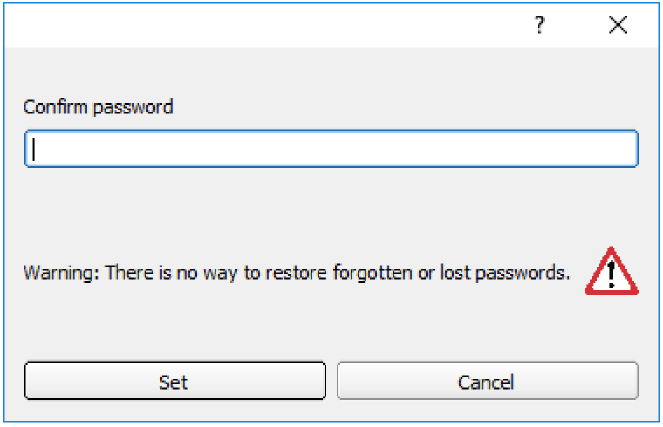
□ Show password

Additional information:
e.g. password hint, manufacturer information, contact person etc.

Fig. 309: Enter a password

NOTE:
The password must be between 1 and 100 characters long.

3. Confirm the entries with the button "Set". Another window to confirm the password opens.
4. Confirm the password by re-entering the password.



Confirm password

Warning: There is no way to restore forgotten or lost passwords.

Fig. 310: Confirm password

5. Press "Set".
6. Save the protected job set
   a. ... on your VISOR®, by selecting the setup step "Start sensor"
   b. ... via File/Saving a job / job set:.

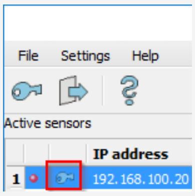
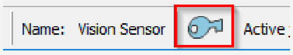
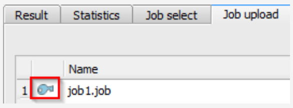
NOTE:
When saving the job or the job set, you can choose between the file types "With password protection (*.job)" and "Without password protection (*.job)".

- "With password protection (*.job)": The job / job set with the entered password for the job / job set is saved. The job / job set can only be opened by entering the correct password for the job / job set.
- "Without password protection (*.job)": The job / job set is saved without password protection. The job / job set can be opened and edited at any time without entering the password.

The following table explains how to open a protected / unprotected job in a protected / unprotected job set:

	Protected job set on the VISOR® vision sensor	Non-protected job set on the VISOR® vision sensor
Open a protected job	Job set protection remains. To open, the password of the protected job must be entered, then the password of the active job set is accepted.	After opening the protected job and saving the job set or starting the sensor, the password protection is applied to the entire job set.
Open an unprotected job	Job set protection remains unchanged and is applied to the unprotected job when saving.	Job set remains unprotected.

A protected job set is marked with a "key symbol". See also the following table:

SensoFind	SensoConfig / SensoView	SensoView
 File Settings Help Active sensors IP address 1 192.168.100.20 ***Fig. 311:** Protected job set, displayed in SensoFind*	 ***Fig. 312:** Protected job set, displayed in SensoConfig / SensoView*	 Result Statistics Job select Job upload Name 1 job1.job ***Fig. 313:** Protected job set, displayed in SensoView*
A VISOR® with a protected job set will be shown with a key symbol in the "Active sensors" list.	A protected job / job set is marked with a key symbol in the status bar.	A protected job / job set is marked with a key symbol in the "Upload" tab.

Change password

1. Select "Protect job set ..."in:"SensoConfig/Menu/Protect job set ...".
2. Enter the existing old password and press the "Change" button.
3. Confirm the password by re-entering the password and press the button "Set".
4. Save the new password
   a. ... on your VISOR® vision sensor by selecting the "Start sensor" setup step.
   b. ... via File/Saving a job / job set:.

Deactivate password

1. Select "Protect job set ..."via:"SensoConfig/Menu/Protect job set ..."
2. Enter the existing password and press the "Deactivate" button.
3. Save the job set
   a. ... on your VISOR® vision sensor by selecting the "Start sensor" setup step.
   b. ... via File/Saving a job / job set:.

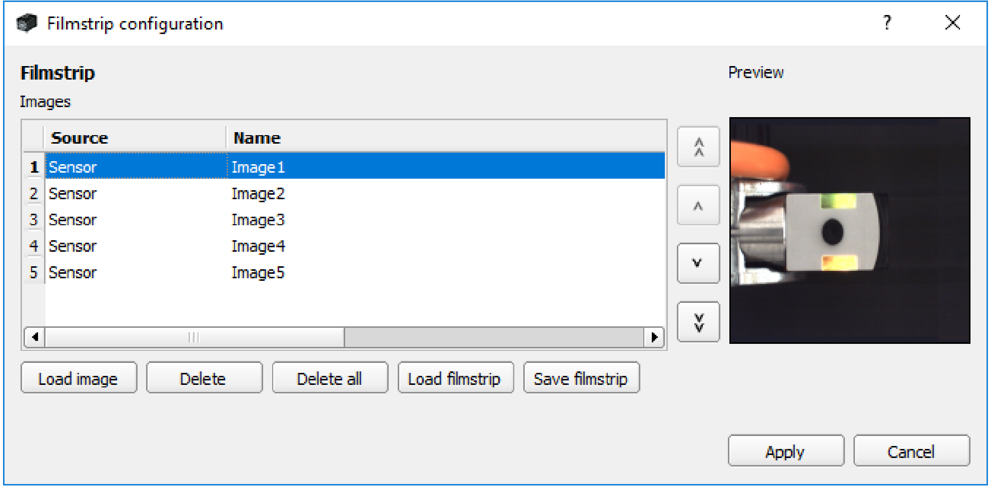
8.11 Filmstrips (file)

In the online mode configuration mode, images are continuously loaded from the sensor into the RAM of the PC. After switching from online to offline mode, you have a maximum of 30 images that you can save as a series of images in a filmstrip file. As an alternative or in addition to the images stored on the sensor, you can load image series or individual images that are stored on your PC or on an external storage medium and combine them to form new filmstrips.

If you select an image in the list, it will be displayed on the right in the preview window in the small format.

8.11.1 Storing images from the sensor as filmstrips:

1. First connect the PC to the sensor. Load the memory with images in Free run (connection mode = online).
2. Select radio button "Offline" in the field Connection mode.
3. Choose "Configure filmstrip" from the File menu or click the Filmstrip icon in the toolbar. In the drop-down list that opens, the images loaded by the sensor appear:



	Source	Name
1	Sensor	Image1
2	Sensor	Image2
3	Sensor	Image3
4	Sensor	Image4
5	Sensor	Image5

***Fig. 314:** Filmstrip*

Now the images can be viewed, rearranged or individual images can be deleted or added. The maximum number of images in a filmstrip is 30.

4. Click on the button "Save filmstrip" under the drop-down list.

All images in the list are saved in the order shown in a filmstrip file (extension .flm) and are available for future simulations.

8.11.2 Loading filmstrips and individual images from the PC:

1. Select radio button "Offline" in the field "Connection mode".
2. Select "Configure Film Strip" from the File menu or click on the "Film Strip" icon in the toolbar.
3. Select a filmstrip file from the selection list and click on the button "Load filmstrip" or load individual images from your PC or an external storage medium with the button "Open image".

The loaded images are added to the drop-down list.

The Source column displays the type and location of the file: Filmstrip stored on the PC (film), frame (file) saved on the PC, image in the sensor memory (sensor). After switching from online to offline mode, all entries are from the sensor type.

8.11.3 Edit filmstrips:

You can create new movies from the frames in the drop-down list, regardless of their source.

The following functions are available for this purpose:

Button	Function
"<", "<<", ">", ">>"	Change picture order: The selected picture is moved up or down by one place or to the end of the list.
Open image	Load further image
Delete, Delete all	Delete image from the list / Delete all images from the list. (The images on the PC are not deleted.)
Cancel	Quit the list without any modification
Apply	Load all images into the movie memory on the PC in the order shown. These are then available for display and evaluation in offline mode.
Open / save filmstrip	Load filmstrip from PC or save it there

8.12 Image recorder

An image recorder is available in the programs SensoConfig and SensoView. When the recorder is activated, either all images or only fault images are continuously loaded into the internal memory of the sensor. This captures 10 images; the oldest images are each overwritten (ring buffer). The recorded images can then be retrieved and viewed on a PC, stored on the PC or on an external storage medium, and then be available for analysis or simulation purposes in offline mode.

In the program SensoView, you may need to enter a password (if enabled) to access the recorder images (for the user group "Worker", see User Management).

Activate recorder:

Activate the video recorder in the "Output" step in the "Image transmission" tab. The following options are available via the Image Recorder selection list:

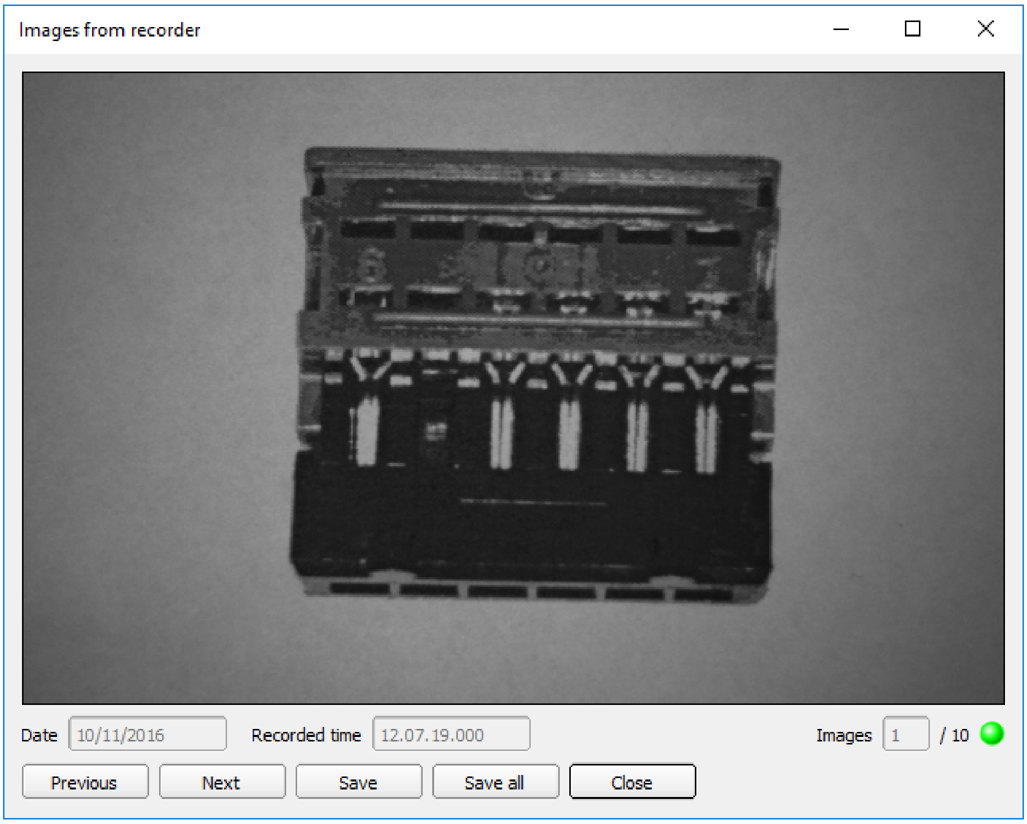
Off: disabled. No files are written.

Any / Pass / Fail: All images, only the images of the good parts or only the images of the bad parts, depending on the overall result of the job as defined in "Output" / "Digital output" / "Job result".

Selecting and recording images:

Select "Get recorder images" from the File menu or click the "Rec. images" button (in SensoView).

An image window appears in which you can load, view, and save the images stored in the sensor on the PC:



Images from recorder

Date 10/11/2016 Recorded time 12.07.19.000 Images 1 / 10

Fig. 315: Image recorder

Parameter	Function
Date Recording time	Date and time of the recording.
Images	The sequential number of the selected image and the total number of images (max. 10) are displayed in the counter below the image window.
Back	Displays the previous image
Next	Displays the next image
Save	Saves the image displayed on the PC or an external storage medium
Save all	Saves all images

When saving, the images are saved in bitmap format (extension .bmp). The test result (OK or error) and the date associated with the respective image are saved in the file name (format YYMMTT_numbering no._pass/fail.bmp, e.g. 090225_123456_Pass.bmp). If you want to record detailed test results together with the images, use the "Archiving" function in SensoView. If you want to record only a single image with or without an overlay, you can use the "Save Current Image" function in the File menu instead of the recorder.

NOTE:
Loading the images from the sensor to the PC erases the data on the sensor. If the recorder window is closed without saving the images first, the images will be lost. In the event of a power failure, images will be lost from the buffer.

8.13 Examples (file)

In the menu "File / Examples", some predefined application examples can be loaded. A filmstrip is loaded together with a job file.

8.14 Search and feature ranges

You can define search and feature ranges in the configuration steps Alignment and Detectors. These are marked in the image window by different colored frames.

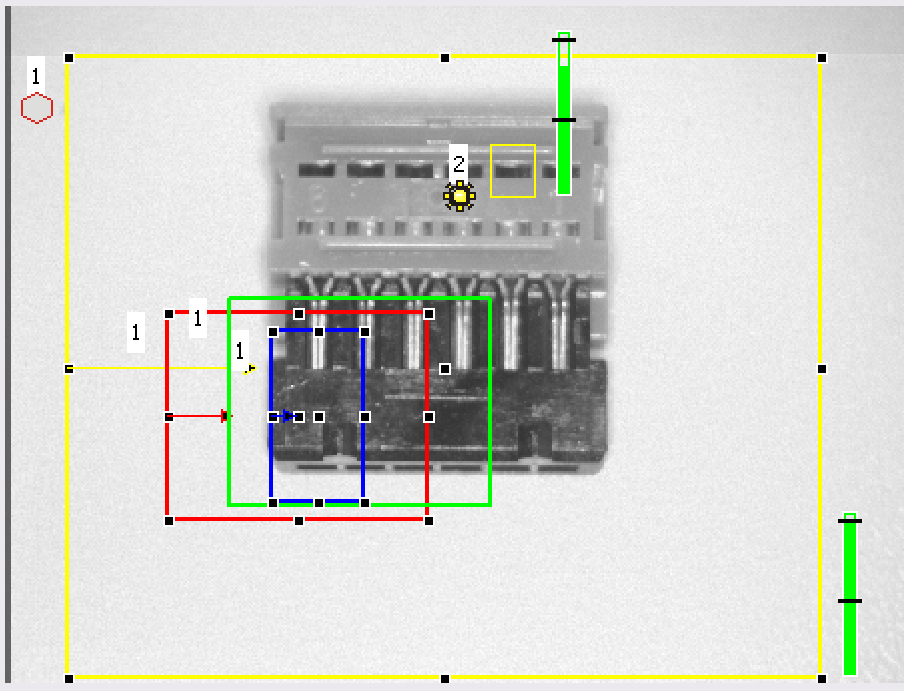


Fig. 316: *Search and feature ranges*

Frame colour	Meaning
Yellow	region of interest (ROI)
Red	Teach-in area
Green	Found features
Blue	Position control
Yellow dotted	Alignment

8.14.1 Definition of search and feature ranges

When a new detector is created, a yellow frame is displayed which defines the detector's region of interest. The standard shape of the region of interest is a rectangle. Depending on the type of detector, the shape "Circle" or "Free shape" can also be selected.

Drag the corner points with the left mouse button to enlarge or reduce the area. If you press the Shift key while doing this, the rectangle changes to a square. Drag the corner points with the left mouse button to enlarge or reduce the region of interest. If you press the Shift key while doing this, the shape snaps into place in 15-degree increments.

The defined features (red frame) are found (green frame) provided its center is within the region of interest (yellow frame).

For the detectors Pattern matching, Contour and Contour 3D there is an additional feature area within the region of interest, which is represented by a red or green frame:

- Red frame = Teach feature
- Green frame = Feature found

If a Position control is defined, an additional blue frame appears (either rectangle, circle, or ellipse).

If an Alignment is defined, its frame is shown in dotted yellow lines.

The respective detector number is displayed at the upper left corner of the frame.

8.14.2 Adapting search and feature ranges

The ranges first displayed in standard size and position can be selected / marked in the image or in the detector list and changed in position and size. Eight handles on the frame enable you to adapt the format and size of the frame. Its position can be displaced by clicking anywhere inside the frame. The arrow pointing to the center can be used to change the rotational position of the frame.

The taught-in pattern or contour is displayed in its original size in the first tab in the lower right corner of the screen. Only the frames of the currently active detector selected in the image or in the detector list are displayed in thick line width and with touch points. All other frames that are not selected at this time are displayed with thin or dashed lines (Alignment).

NOTE:

- For optimum detection, features must be distinct and not contain any variable parts, e.g. shadows.
- Significant contours, edges, and contrast distinctions are advantageous.
- To minimize the evaluation time, the region of interest should only be as large as necessary.

Result bar

To the right of the region of interest, the degree of conformity of the sought-after feature with the feature found is displayed as a standing result bar with a set threshold value:

- Green bar = The sought-after feature was found and the preset threshold of minimum match was reached.
- Red bar = The object could not be found with the required degree of conformity.

Overlays and display

In the "View" menu you can select which graphical representations are displayed.

"Current detector only": show only overlays of the detector currently being processed

"Failed detectors only": show only overlays of failed detectors

Under "View" / "Overlay settings", the overlays in the image (frames in yellow, red, etc.) can be switched on or off as required for each detector or category.

"Result Bar Graph": Show or hide result bar

8.15 Simulation mode: Simulation of jobs (offline mode)

You can also create and test your configuration without a connected sensor using saved filmstrips (= image series). A simulation may e.g. be sensible to prepare a configuration or to optimize an online configuration.

NOTE:

- In the delivery state of SensoConfig, there are some prepared filmstrips available.
- Additional image acquisition options: Image recorder.

To start the simulation mode, select a sensor type in SensoFind from the list "Sensors for simulation mode". Via double click or click on "Config", SensoConfig opens in simulation mode for this sensor type.

8.16 Color models

There are so-called color models for the description of colors. The VISOR® vision sensor Color can work with various color models.

The following color models can be used:

Color model RGB

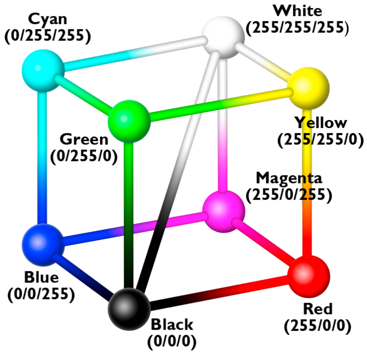
Color model HSV

Color model LAB

8.16.1 Color model RGB

An RGB color space is an additive color space that replicates color perceptions by additive mixing of three primary colors (red, green, and blue).

The RGB color space is described as a linear color space, as a cube with the three axis: red, green, and blue.



Red, green, blue, each 0-255
The RGB color space is used by both the image chip and the screen to define the colors. However, the image chip and the screen have different sensitivities within the individual color channels. Because of this, there must always be a compensation, so RGB is never equal to RGB.

Fig. 317: *Color model RGB*

Linear RGB

RGB values are calculated as linear RGB values, as the sensor chip delivers linear RGB values. The advantage of the linear RGB values is that there is a linear relationship between physical influence and RGB values.

Example: Doubling the exposure time leads to a doubling of the RGB value under constant lighting conditions.

8.16.2 Color model HSV

In terms of color adjustment, the HSV color space is preferred over the alternatives RGB and Lab because it resembles human color perception.

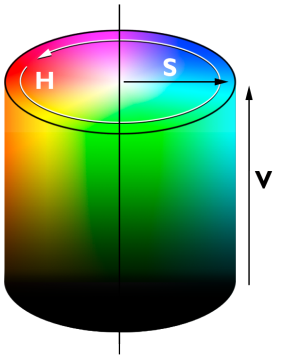


Fig. 318: *Color model HSV*

- H (hue) as a color angle on the color circle (e. g. 0° = red, 120° = green, 240° = blue)
- S (saturation) in percent (0 % = light gray, 50 % = low saturated color, 100 % = maximum saturated color)
- V (value) in percent (0 % = dark, 100 % = full brightness)

8.16.3 Color model LAB

LAB or L*a*b* color space is described by a three-dimensional coordinate system:

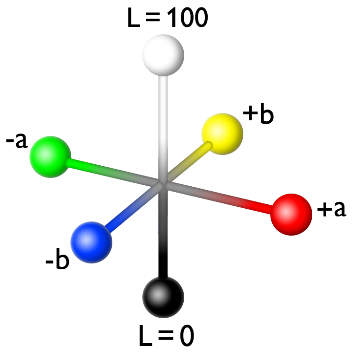


Fig. 319: *Color model LAB*

- An a*-axis describes the green and red components of a color; negative values stand for green and positive values stand for red. Range of values from -87 to +99.
- A b*-axis describes the blue and yellow components of a color; negative values stand for blue and positive values stand for yellow. Range of values from -108 to +95.
- An L*-axis describes the lightness (luminance) of the color with values from 0 to 100.

One of the most important features of the L*a*b* color model is its device independence. This means that the colors are defined independently of the way they are produced and reproduced. LAB values are calculated from linear RGB values. This is based on the standard illuminant D65 and 2° observer.

9 VISOR® Software – SensoView

This program is used to monitor / verify connected sensors and to analyze test results. No new settings can be made on the sensor.

Beyond the mere display, it is only possible to switch between pre-existing jobs on the sensor, or predefined job sets can be uploaded from the PC / PLC to the sensor by the authorized worker. Thus, this display tool mainly serves to visualize images and results and to change jobs at e.g. part change on the machine.

You can archive inspection results and statistical evaluations, including the selected graphical representations, in the program SensoView.

9.1 Image display

The graphic display of the image and the test results in the image window depends on the settings in the "Image transmission" tab in the job settings ("Image transmission" tab, Chapter "Image transfer parameters" in program SensoConfig):

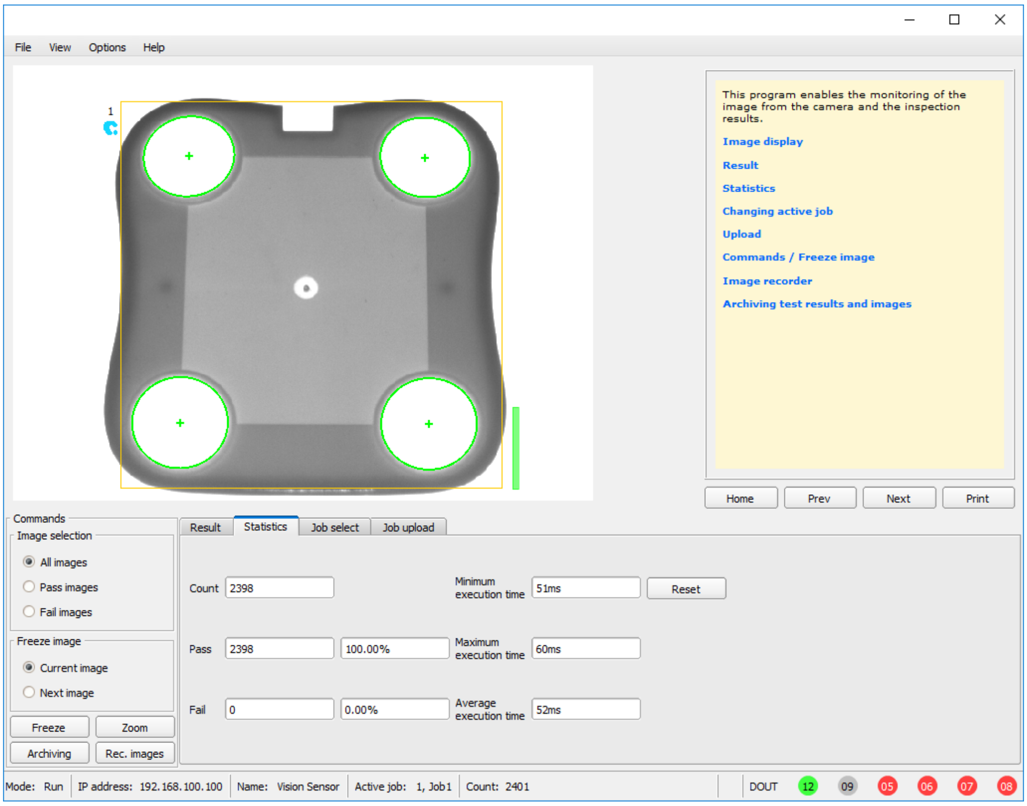
- Image transfer active: The current image as well as the frames for the defined search, feature and position ranges, and the found characteristics are displayed.
- Image transferal inactive: Only the frames for the defined search, feature and position areas, and the found features are displayed (the current picture is not displayed).

To the right of the region of interest of the respective detector, the degree of agreement of the sought-after with the found feature is displayed as a standing result bar with a set threshold value:

- Green bar: The parameter searched for has been found and the preset threshold value for concordance has been reached.
- Red bar: The object could not be found with the required degree of conformity

ATTENTION:
When this icon appears on the life image, it is indicating the image visualization / image storage on the PC is running slower than the image processing on the VISOR®. Not all images taken by the VISOR® are displayed anymore. This may lead to image loss when using poor image archiving.
If the icon appears frequently, programs opened on the PC should be closed in the background to provide more PC performance.

In the mSensoViewenu / view, you can configure the graphical representation of the test results.



This program enables the monitoring of the image from the camera and the inspection results.

Image display

Result

Statistics

Changing active job

Upload

Commands / Freeze image

Image recorder

Archiving test results and images

Image selection

● All images
○ Pass images
○ Fail images

Freeze image

● Current image
○ Next image

Count 2398 | Minimum execution time 51ms

Pass 2398 100.00% | Maximum execution time 60ms

Fail 0 0.00% | Average execution time 52ms

Mode: Run | IP address: 192.168.100.100 | Name: Vision Sensor | Active job: 1, Job1 | Count: 2401

Fig. 320: *SensoView*

Except archiving, all functions of SensoView are also available in the SensoConfig program .

9.2 Commands

9.2.1 Freeze image

With the "Freeze image" button, you can request individual images of the desired type (Current image, Next image, Next defect image) and hold them in the image window for display.

The desired frame is displayed and the frame counter remains at the corresponding frame number.

Press "Continue" to end the frozen image state.

9.2.2 Zoom

With the button "Zoom", the image is opened in a new window with enlarged display.

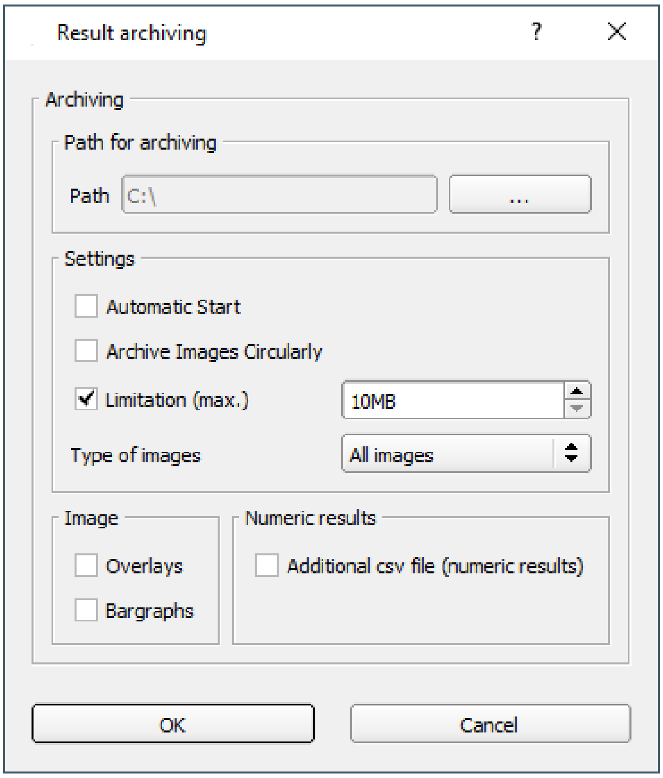
9.2.3 Archiving of test results and images

You can archive images with and without markups and check results on your PC or on an external storage medium for analysis or simulation purposes (see Offline Mode).

The execution of these functions may require the entry of a password (user group worker, see User Administration).

Configuring archiving:

1. Select "Configure Archiving" from the File menu.
   A dialogue box appears with the following options:



Result archiving

Archiving

Path for archiving

Path: C:\

Settings

□ Automatic Start

□ Archive Images Circularly

☑ Limitation (max.): 10MB

Type of images: All images

Image

□ Overlays

□ Bargraphs

Numeric results

□ Additional csv file (numeric results)

__Fig. 321:__ Configure archiving

Parameter	Function
Path for archiving	Directory in which the archived file(s) are stored.
Settings, Automatic start	Starts archiving automatically after starting SensoView.
Settings, Cyclic overwriting	Activates cyclic overwriting of oldest images if storage limit is reached.
Settings, Storage limit	Here, it is possible to limit the data volume.
Settings, Image type	In this selection list you can specify which images (all images or only pass or fail images) should be saved.
Overlays, Bar graph result	Selection of the graphic representations to be archived in the image.
Numerical results	If "Log" is activated, numerical result data such as coordinate values, etc. are archived in an additional .csv file. The "Legacy" / "Configured" setting determines the format of the .csv file. With "Legacy" *1), the contents of the .csv file are specified; "Configured" is freely configurable via "Output / Data output". *1) The storage mode "Legacy" is obsolete and only provided for reasons of backward compatibility. It will be omitted in one of the next versions.

2. Select the desired options and confirm your choice with OK.

Start / end archiving:

Click on the "Archive images" button in the "Commands" window to start or end the archiving function in the above-mentioned settings. The status bar displays the name of the image file currently being saved. Archiving is carried out for as long as the button "Archive Images" is pressed.

9.2.4 Image recorder

An image recorder is available in the programs SensoConfig and SensoView. When the recorder is activated, either all images or only fault images are continuously loaded into the internal memory of the sensor. This captures 10 images; the oldest images are each overwritten (ring buffer). The recorded images can then be retrieved and viewed on a PC, stored on the PC or on an external storage medium, and then be available for analysis or simulation purposes in offline mode.

In the program SensoView, you may need to enter a password (if enabled) to access the recorder images (for the user group "Worker", see User Management).

Activate recorder:

Activate the video recorder in the "Output" step in the "Image transmission" tab. The following options are available via the Image Recorder selection list:

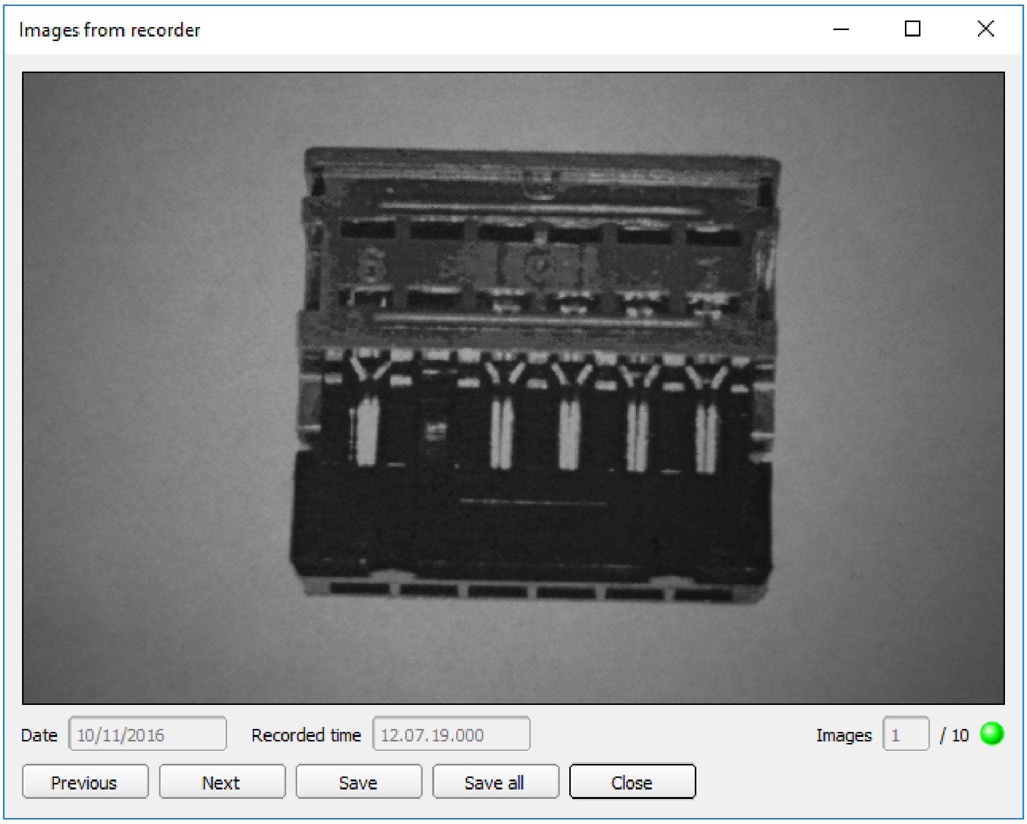
Off: disabled. No files are written.

Any / Pass / Fail: All images, only the images of the good parts or only the images of the bad parts, depending on the overall result of the job as defined in "Output" / "Digital output" / "Job result".

Selecting and recording images:

Select "Get recorder images" from the File menu or click the "Rec. images" button (in SensoView).

An image window appears in which you can load, view, and save the images stored in the sensor on the PC:



Images from recorder

Date 10/11/2016

Recorded time 12.07.19.000

Images 1 / 10 ●

Fig. 322: *Image recorder*

Parameter	Function
Date Recording time	Date and time of the recording.
Images	The sequential number of the selected image and the total number of images (max. 10) are displayed in the counter below the image window.
Back	Displays the previous image
Next	Displays the next image
Save	Saves the image displayed on the PC or an external storage medium
Save all	Saves all images

When saving, the images are saved in bitmap format (extension .bmp). The test result (OK or error) and the date associated with the respective image are saved in the file name (format YYMMTT_numbering no._pass/fail.bmp, e.g. 090225_123456_Pass.bmp). If you want to record detailed test results together with the images, use the "Archiving" function in SensoView. If you want to record only a single image with or without an overlay, you can use the "Save Current Image" function in the File menu instead of the recorder.

NOTE:
Loading the images from the sensor to the PC erases the data on the sensor. If the recorder window is closed without saving the images first, the images will be lost. In the event of a power failure, images will be lost from the buffer.

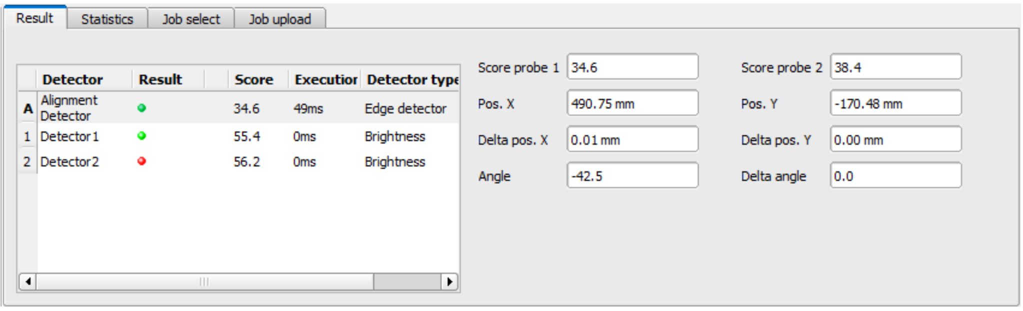
9.3 Result tab

This function executes the defined job on the PC and displays the "Result Statistic" window with detector list and evaluation results. The execution times are not updated in this mode because they are not available from the sensor.

In Run mode, the detailed test results of the detector marked in the drop-down list are displayed.

In the image window – if adjusted – the image, the search and feature ranges, and the result graphs are displayed.

The displayed parameters vary depending on the selected detector type. To see the test results of another detector, mark it in the drop-down list.



	Detector	Result	Score	Executior	Detector type
A	Alignment Detector	🟢	34.6	49ms	Edge detector
1	Detector1	🟢	55.4	0ms	Brightness
2	Detector2	🔴	56.2	0ms	Brightness

Score probe 1: 34.6
Score probe 2: 38.4
Pos. X: 490.75 mm
Pos. Y: -170.48 mm
Delta pos. X: 0.01 mm
Delta pos. Y: 0.00 mm
Angle: -42.5
Delta angle: 0.0

Fig. 323: SensoView, Result

9.4 Statistic tab

In the Run mode, the statistical data of the test process is displayed in the Statistic tab. The statistical data displayed is identical for all types of detectors:

Parameter	Function
All evaluations	Total number of inspections
Good parts	Number of inspections with result "OK"
Bad parts	Number of inspections with result "Error"
Min./max./mean execution time	Min./max./mean execution time for evaluation in ms

All statistical values can be reset to zero with the "Reset" button.

You can archive inspection results and statistical evaluations, including the selected graphical representations, in the SensoView program.

9.5 Job tab

In the Job tab, the jobs available in the sensor are displayed in the selection list. Here you can switch between different jobs stored in the sensor. The green arrow (➤) marks the active job.

Running functions that would cause the active sensor to be stopped (job change, job upload, and fetching recorder images), will also require for a password, if enabled in SensoFind, to be entered (Worker user group; please refer to User administration).

Password levels

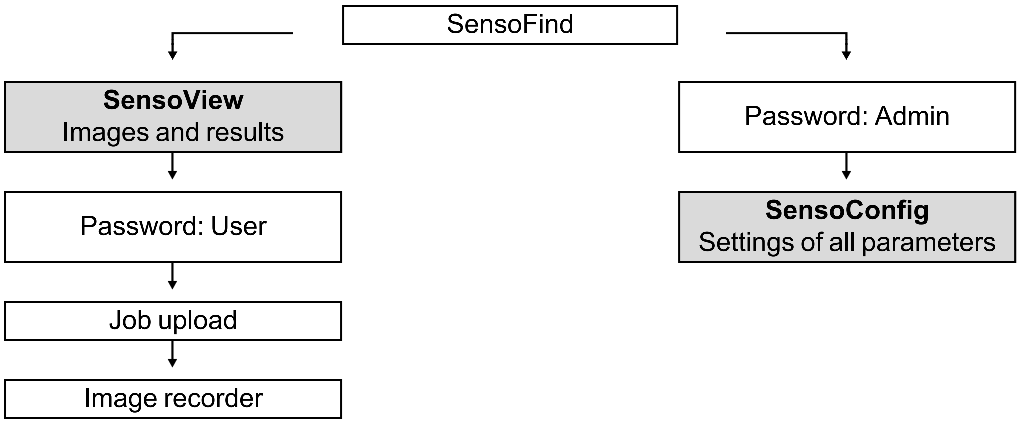
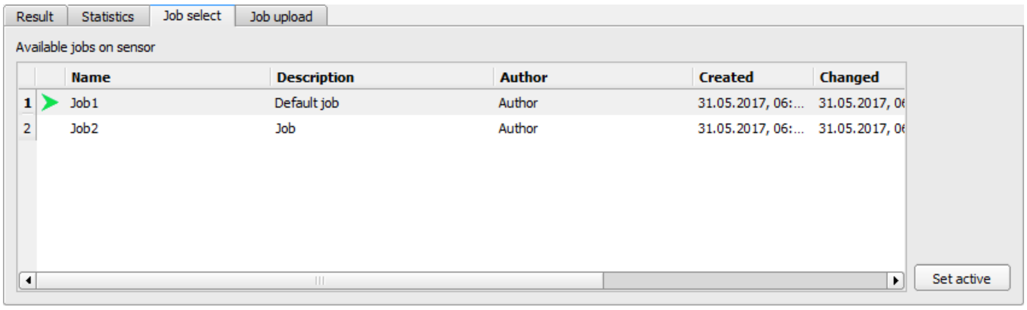


Fig. 324: Password levels



Available jobs on sensor

		Name	Description	Author	Created	Changed
1	➤	Job1	Default job	Author	31.05.2017, 06:...	31.05.2017, 0€
2		Job2	Job	Author	31.05.2017, 06:...	31.05.2017, 0€

Fig. 325: SensoView, Job switch

Procedure

Select a job from the list and activate it with the "Activate" button.

The previous job is deactivated; the selected job is now active.

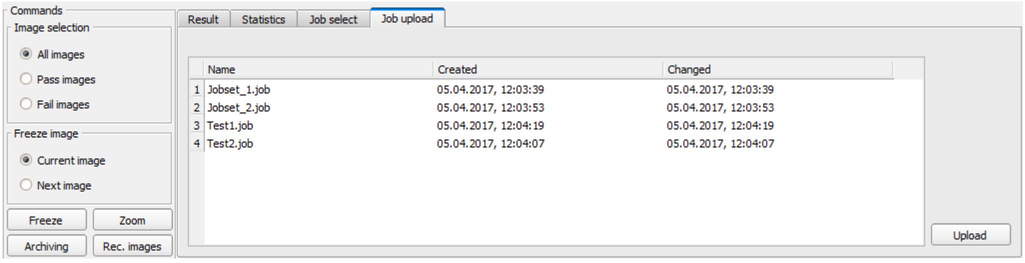
NOTE:
When the job changes and the operating mode changes from Run to Config mode, the following special states of the outputs occur:

- The buffer of the delayed outputs is deleted from "Run" to "Config" when the job is changed and the operating mode is changed.
- Digital outputs: These are reset to the default settings (defaults) when the job is changed and the operating mode is changed from "Run" to "Config". The basic settings are defined by "Invert" in the "Output" / "Digital output" tab. "Invert" inverts the basic setting of the digital output and, at the same time, the result.
- Ready and Valid: Ready and Valid signals when the job is changed and the operating mode changes from "Run" to "Config" that the sensor is not ready and the results are not valid (low level).

9.6 Upload tab

In the Upload tab, you can load new jobs or entire job sets from the PC into the sensor memory. The available jobs or job sets are displayed in the drop-down list.

Jobs and job sets can be created in the SensoConfig program and stored there under File / Save Job / Save job set as ...



Commands

Image selection

● All images
○ Pass images
○ Fail images

Freeze image

● Current image
○ Next image

	Name	Created	Changed
1	Jobset_1.job	05.04.2017, 12:03:39	05.04.2017, 12:03:39
2	Jobset_2.job	05.04.2017, 12:03:53	05.04.2017, 12:03:53
3	Test1.job	05.04.2017, 12:04:19	05.04.2017, 12:04:19
4	Test2.job	05.04.2017, 12:04:07	05.04.2017, 12:04:07

Fig. 326: *SensoView, Upload tab, Load job set*

NOTE:

- A job set consists of one or more jobs stored in the sensor or on the hard disk.
- Performing functions that cause the active sensor to stop may require the entry of a password (user group worker, see User Administration).
- Select a job or job set from the list and load it onto the sensor with the "Upload" button.
- This action deletes all jobs previously stored on the sensor!

10 VISOR® – SensoWeb

SensoWeb is used (as well as SensoView) to monitor / check connected sensors and to analyze test results. No new settings can be made on the sensor.

In contrast to the program SensoView, the display takes place in the browser (no software installation is required for display).

The following browsers are supported (specified version and newer):

- Microsoft Edge 90, Mozilla Firefox 88.0, Google Chrome 90
- Apple Safari 14.0
- Internet Explorer 11

10.1 Launching SensoWeb

1. In SensoConfig, operating step Output, select Interfaces tab.
   Activate the checkbox on the right side of the line SensoWeb.
2. In SensoConfig, start the sensor via button "Start sensor" .
3. Open browser and enter the IP address of the sensor in the address line of the browser (displayed in SensoFind).
   Format: "http://Your Sensor IP", e.g. "http://192.168.100.100" (default).

NOTE:

- 10 browser connections are allowed per VISOR® vision sensor.
- The use of SensoWeb affects the processing time of the VISOR® vision sensor. Opening multiple connections to SensoWeb can slow down the overall execution time.

10.2 Functions of SensoWeb

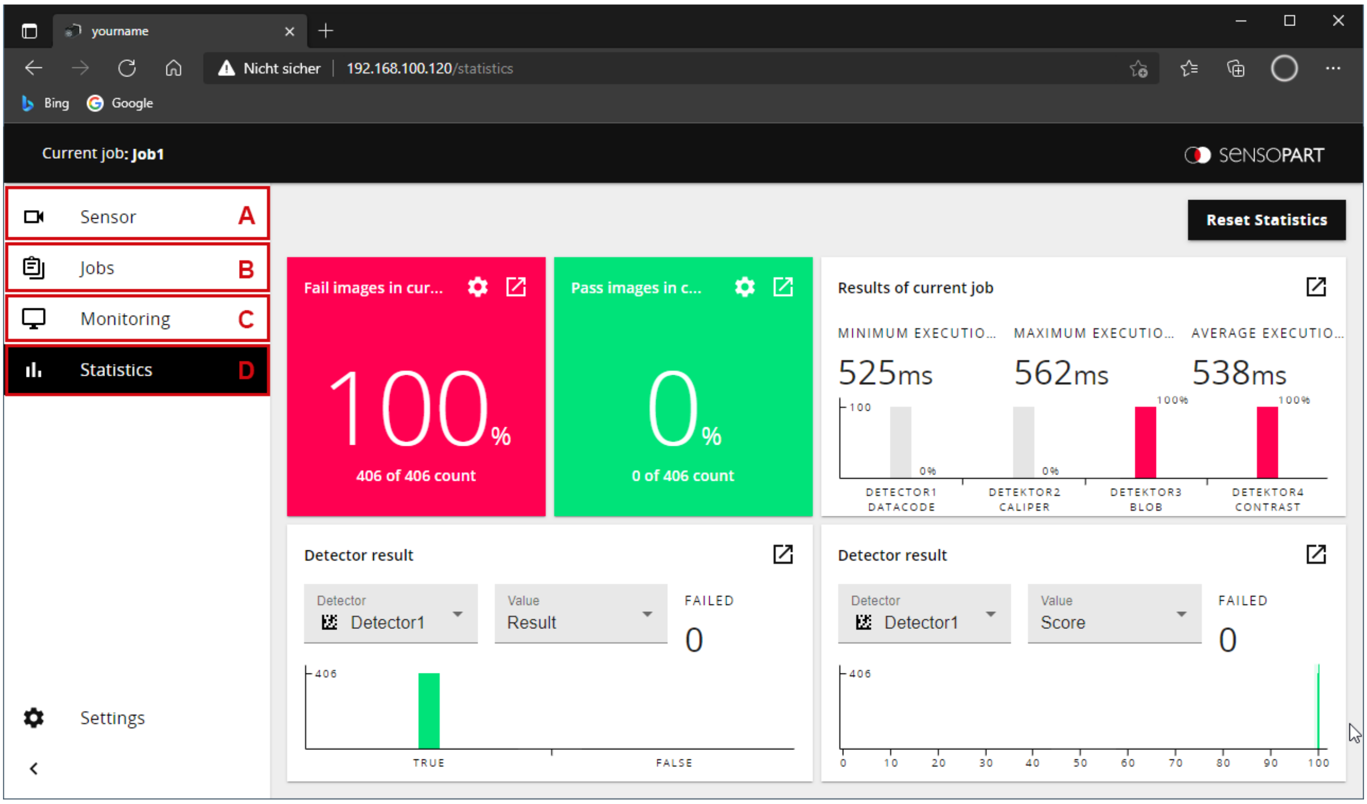


Fig. 327: View of SensoWeb in browser

SensoWeb consists of the following subpages:

A: Sensor

Display of the sensor details of the connected sensor (name, type, software version).

B: Jobs

Display of the jobs present on the sensor and display of the active job.

C: Monitoring

Display of image and result data of the active job.

Depending on the selected detector, tabular or individual results are output. If a detector returns both types of results, it is possible to switch between the "Result table" and "Result values" tabs.

D: Statistic

Detailed statistics on individual detectors of the active job (overview of evaluations, good and bad parts, and cycle and execution times).

NOTE:
Under "Detector Result", the result and score are displayed by default. If further individual values are to be displayed, these must first be selected in setup step Output / Telegram / Payload.

10.3 URLs in SensoWeb

In "SensoWeb" it is possible to arrange the individual displays (e.g. Monitoring or Statistic) modularly via URL in separate windows.

To open a single section of SensoWeb in a new window, click the icon in the upper right corner: ↗

The corresponding URL can now be saved as a favorite, for example, and the window can thus be called up directly.

Example: `http://192.168.100.100/monitor/image/live-any/overlay-all/1/errorhighlight`

In addition, the enlarged view can be accessed directly with `http://192.168.100.100/zoom.html` (or alternatively the IP address of the sensor).

Handling of invalid URLs

Usecase	Device mode	Default URL
An invalid URL entered in configuration mode redirects the application to the Sensor page.	Configuration mode	http://[device-IP]/sensor
An invalid URL entered in Run mode redirects the application to the Monitoring page.	Run mode	http://[device-IP]/monitor

11 Communication

NOTE:
For more information on the topic of communication, you can download the VISOR® Communications manual from the Download area of the SensoPart website (www.sensopart.com). The VISOR® Communications manual is also part of the VISOR® installation and can be found in the \Documentation subfolder.

11.1 Network settings

11.1.1 Integrating the VISOR® into the network / gateway

SensoFind/Active sensors will show a list with all the VISOR® vision sensors that are found on the same network segment on the PC on which is running SensoFind. To update the list, press the "Find" button, e.g. for sensors that were only activated after viewing SensoFind.

For sensors which are installed in the network but are located in a different network segment via a gateway, please enter the corresponding sensor IP address under "Add active sensor" and press the button "Add". The corresponding sensor will now also appear in the "Active sensors" list, and you will be able to access it and work with it.

11.1.2 Direct connection of the VISOR® vision sensor

NOTE:
To configure the PC via the direct connection, the IP configuration of the PC must be changed.
In advance, inform the administrator about changes.

1. Connect the sensor: power supply 24 V; network cable.
   → Green PWR LED of VISOR® is on. Status LED on network adapter is active and/or LAN status shows "Connection established" (Control Panel / Network Connection / LAN Connection / Status)
2. Check whether the IP address of the PC matches the vision sensor. Sensor (Default): 192.168.100.100/24. (/24 means: 24-bit subnet mask = 255.255.255.0).
   If not, check / correct the settings.
3. Open SensoFind and click the "Find" button. Select sensor.
4. Click the "Config" button or double-click on Vision Sensor.
   → SensoConfig opens.
5. Parameterize VISOR® in SensoConfig.

11.1.3 Connection of the VISOR® vision sensor via network

Establishing an Ethernet connection between the VISOR® vision sensor and the PC through a network.

NOTE:
To configure the VISOR® for the network, it must be integrated into the network.
To establish communication between the vision sensor and the PC, the sensor requires a free IP address. Before connecting, check whether the address of the sensor is already assigned to other devices!
Default: 192.168.100.100/24 (24 means: 24-bit subnet mask = 255.255.255.0)
Contact your administrator to obtain a free IP address.

1. Connect the sensor: power supply 24 V; network cable.
   → Green PWR LED of VISOR® is on. Status LED on network adapter is active and/or LAN status shows "Connection established" (Control Panel / Network Connection / LAN Connection / Status)
2. Open SensoFind and click the "Find" button. Select sensor.
3. Configure the IP address via the "Set" button and enter the subnet mask. Confirm with "Set".
4. Click the "Find" button.
5. Click the "Config" button or double-click on Vision Sensor.
   → SensoConfig opens.
6. Parameterize VISOR® in SensoConfig.

11.1.4 Used Ethernet ports

If you are integrating the VISOR® into a network, make sure that an admin opens the following ports if necessary. This is only the case if these ports were previously explicitly blocked in the company network or by a firewall installed on the PC.

The following ports are used for communications between the VISOR® software (PC) and the VISOR®:

- Port 2000, TCP
- Port 2001, UDP Broadcast (to find sensors via SensoFind)
- Port 2002, TCP
- Port 2003, TCP
- Port 2004, TCP

The following ports are used for communications between the PLC (PLC or control PC) and VISOR® vision sensor:

Process interfaces:

- Ethernet
  - Port 2005, TCP (Implicit results, i.e. user-configured result data)
  - Port 2006, TCP (Explicit requests, e.g. trigger or job switch)
- EtherNet/IP:
  - Port 2222, UDP
  - Port 44818, TCP
- PROFINET:
  - Port 161, UDP
  - Port 34962, UDP
  - Port 34963, UDP
  - Port 34964, UDP
- Service:
  - Port 22, TCP
  - Port 1998, TCP

- SensoWeb:
  - Port 80

NOTE:
If Ports 2005 or 2006 are changed in the configuration software, they must also be changed accordingly in the firewall by an administrator.

11.1.5 Access to VISOR® through network

Exemplary values for IP, etc.

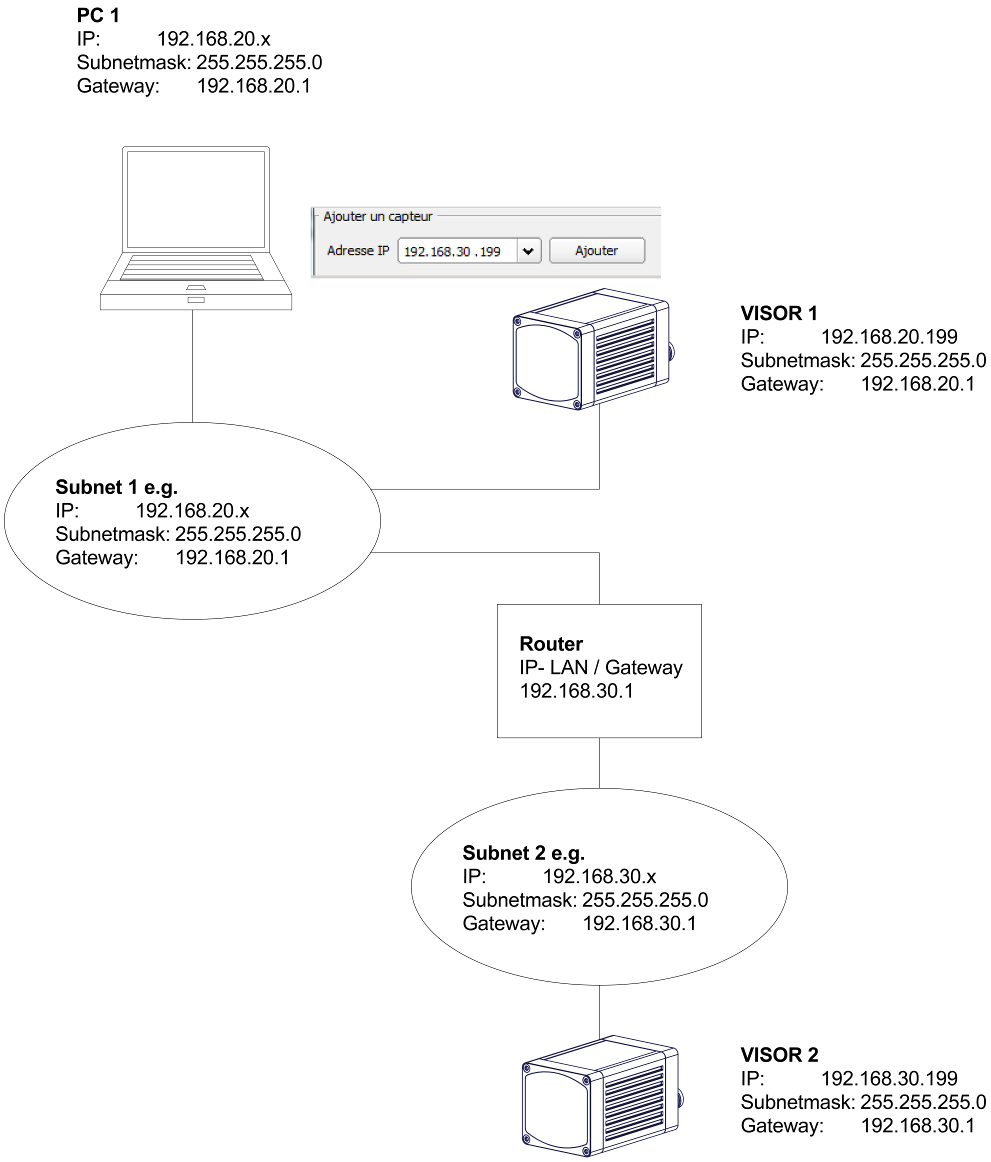
Access to VISOR® 1 from PC 1, if on the same subnet

- Via SensoFind (/find)

Access to VISOR® 2 from PC 1, if on a different subnet

Only if:

- Gateway is set correctly in Sensor 2 (here to 192.168.30.1) - and
- in SensoFind via Add IP, the sensor IP of Sensor 2 is set correctly
  > after this, VISOR® 2 will also appear in the "Active sensors" list in SensoFind!



PC 1
IP: 192.168.20.x
Subnetmask: 255.255.255.0
Gateway: 192.168.20.1

Ajouter un capteur
Adresse IP 192.168.30.199
Ajouter

VISOR 1
IP: 192.168.20.199
Subnetmask: 255.255.255.0
Gateway: 192.168.20.1

Subnet 1 e.g.
IP: 192.168.20.x
Subnetmask: 255.255.255.0
Gateway: 192.168.20.1

Router
IP- LAN / Gateway
192.168.30.1

Subnet 2 e.g.
IP: 192.168.30.x
Subnetmask: 255.255.255.0
Gateway: 192.168.30.1

VISOR 2
IP: 192.168.30.199
Subnetmask: 255.255.255.0
Gateway: 192.168.30.1

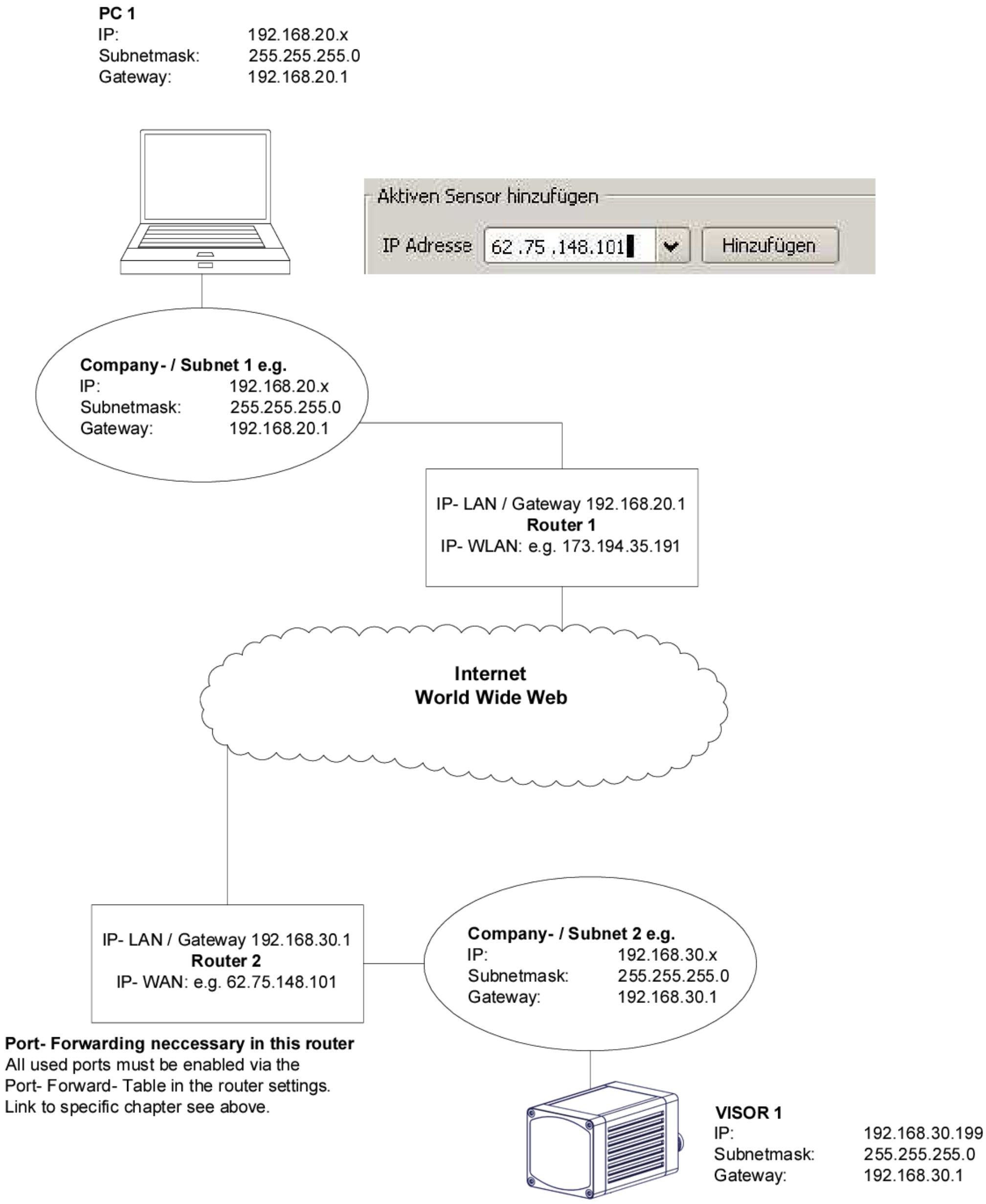
Fig. 328: *Access to VISOR® through network, same or other subnet*

11.1.6 Access to VISOR® through the Internet / World Wide Web

Exemplary values for IP, etc.

Access from PC 1 (company network 1), through the Word Wide Web, to company network 2 to VISOR® 1.

1. On PC 1 (company network 1SensoFind) enter and add the IP WAN of Router 2 (company network 2) under "Add active sensor" in (here in this example: 62.75.148.101)
2. On router 2, open the ports that the sensor will be using (please refer to section: Used Ethernet ports).See Chapter:



Aktiven Sensor hinzufügen

IP Adresse 62 .75 .148.101

Hinzufügen

PC 1
IP: 192.168.20.x
Subnetmask: 255.255.255.0
Gateway: 192.168.20.1
Company- / Subnet 1 e.g.
IP: 192.168.20.x
Subnetmask: 255.255.255.0
Gateway: 192.168.20.1
IP- LAN / Gateway 192.168.20.1
Router 1
IP- WLAN: e.g. 173.194.35.191
Internet
World Wide Web
IP- LAN / Gateway 192.168.30.1
Router 2
IP- WAN: e.g. 62.75.148.101
Company- / Subnet 2 e.g.
IP: 192.168.30.x
Subnetmask: 255.255.255.0
Gateway: 192.168.30.1
Port- Forwarding neccessary in this router
All used ports must be enabled via the
Port- Forward- Table in the router settings.
Link to specific chapter see above.
VISOR 1
IP: 192.168.30.199
Subnetmask: 255.255.255.0
Gateway: 192.168.30.1

Fig. 329: *Access to VISOR® through the Internet / World Wide Web*

11.2 Job change

11.2.1 Job change with digital inputs

To switch between several jobs which are already stored on the sensor, the following options are available via digital inputs:

See also Chapter I/O mapping tab, timing diagrams and comments

11.2.1.1 Job 1 or Job 2

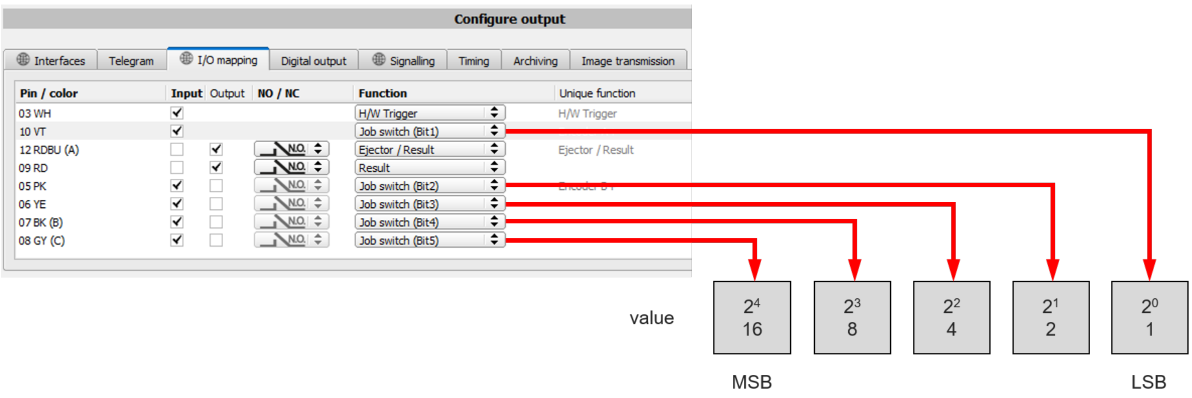
To switch between Job 1 and Job 2, an arbitrary value can be defined in SensoConfig/Output/I/O Mapping with the function "Job 1 or 2". After applying the according level at this input, Job 1 or Job 2 will then be executed (Low = Job 1, High = Job 2). See also Chapter I/O mapping tab / Function of inputs et seqq.

11.2.1.2 Job 1 … 255 via a binary input bit pattern

To switch between up to 255 jobs via a binary input combination of up to 8 inputs, all required inputs are assigned under SensoConfig / Output /I/O Mapping with the corresponding function "Job change bit x". The corresponding binary input patterns, as shown in the image below, then directly switch to the corresponding job when it is created. Please refer to the following section as well: I/O mapping tab / Input functions.

NOTE:

- The job change will start immediately after the input combination changes.
- The display of the active job in the status bar changes with the first, following trigger.
- The mapping of the I/O's is not fixed. It depending on the settings in SensoConfig/Output/I/O Mapping.
- The level change of the associated inputs must be made simultaneously (within a maximum of 10 ms, all levels must be stable).



Pin / color	Input	Output	NO / NC	Function	Unique function
03 WH	☑			H/W Trigger	H/W Trigger
10 VT	☑			Job switch (Bit1)	
12 RDBU (A)	□	☑	N.O.	Ejector / Result	Ejector / Result
09 RD	□	☑	N.O.	Result	
05 PK	☑	□	N.O.	Job switch (Bit2)	
06 YE	☑	□	N.O.	Job switch (Bit3)	
07 BK (B)	☑	□	N.O.	Job switch (Bit4)	
08 GY (C)	☑	□	N.O.	Job switch (Bit5)	

e.g. Job No.	Input 5	Input 4	Input 3	Input 2	Input 1
1	false	false	false	false	true
2	false	false	false	true	false
6	false	false	true	true	false
13	false	true	true	false	true
31	true	true	true	true	true
No change	false	false	false	false	false

***Fig. 330:** Job change binary (in this example with 5 inputs and therefore up to 31 jobs)*

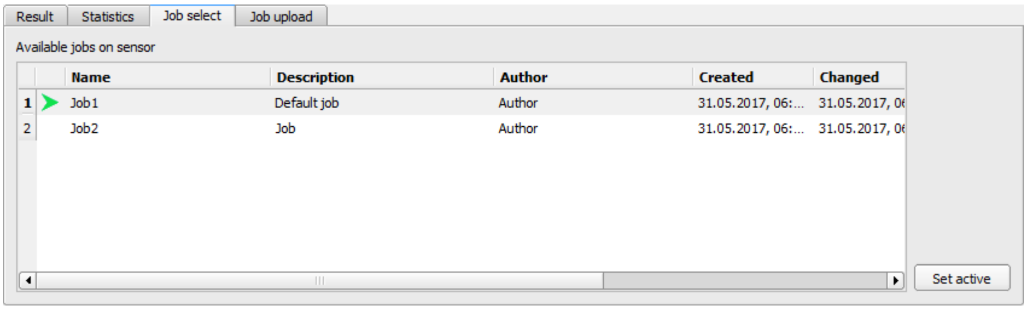
11.2.2 Job change Ethernet

For more information, please refer to the VISOR® Communications manual.

11.2.3 Job change with SensoView

In the applicatioSensoViewn , a job switch can be made, or completely new job sets can be uploaded to the sensor. See also Chapter Job tab

In the tab "SensoView/Job", all jobs stored on the sensor are displayed. If there is more than one job in the sensor memory, one of them can be marked in the list and activated with "Activate".



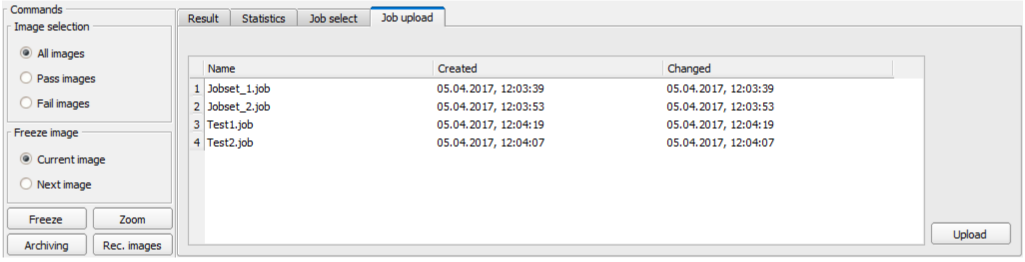
Available jobs on sensor

	Name	Description	Author	Created	Changed
1	➤ Job1	Default job	Author	31.05.2017, 06:...	31.05.2017, 06
2	Job2	Job	Author	31.05.2017, 06:...	31.05.2017, 06

Fig. 331: SensoView, job change

In the tab "SensoView/Upload", all available job sets on the PC are displayed. These can be marked in the list and then uploaded to the sensor with "Upload".

ATTENTION:
By uploading a new job set, all jobs in the sensor memory are deleted.



	Name	Created	Changed
1	Jobset_1.job	05.04.2017, 12:03:39	05.04.2017, 12:03:39
2	Jobset_2.job	05.04.2017, 12:03:53	05.04.2017, 12:03:53
3	Test1.job	05.04.2017, 12:04:19	05.04.2017, 12:04:19
4	Test2.job	05.04.2017, 12:04:07	05.04.2017, 12:04:07

Fig. 332: SensoView, Upload job

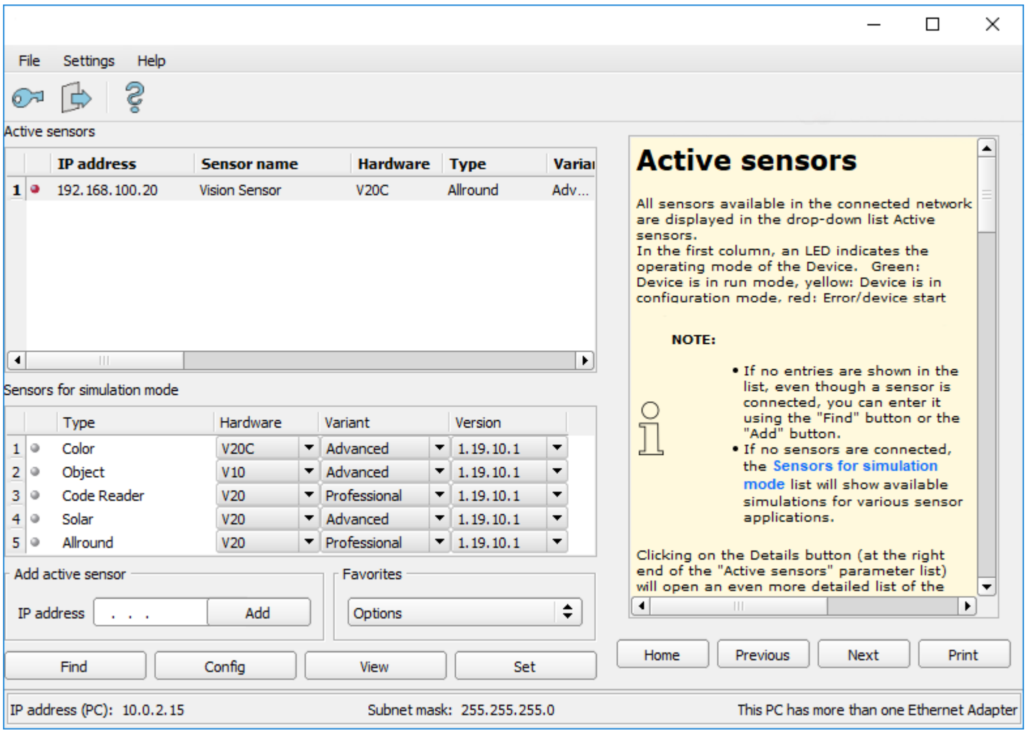
11.3 PC archiving (SensoView)

Via SensoFind, images and numerical data (in .csv format) can be stored in a folder on the PC.

Configuration (directory, etc. ...) of the archiving is done via SensoView in the menu "File/Configure Archiving". This function is available on PC only.

Step 1

Start SensoView by clicking on the button "View" in SensoFind.



	IP address	Sensor name	Hardware	Type	Variai
1	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

	Type	Hardware	Variant	Version
1	Color	V20C	Advanced	1.19.10.1
2	Object	V10	Advanced	1.19.10.1
3	Code Reader	V20	Professional	1.19.10.1
4	Solar	V20	Advanced	1.19.10.1
5	Allround	V20	Professional	1.19.10.1

Active sensors

All sensors available in the connected network are displayed in the drop-down list Active sensors.
In the first column, an LED indicates the operating mode of the Device. Green: Device is in run mode, yellow: Device is in configuration mode, red: Error/device start

NOTE:

- If no entries are shown in the list, even though a sensor is connected, you can enter it using the "Find" button or the "Add" button.
- If no sensors are connected, the Sensors for simulation mode list will show available simulations for various sensor applications.

Clicking on the Details button (at the right end of the "Active sensors" parameter list) will open an even more detailed list of the

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Fig. 333: SensoFind

SensoView is opened.

The conditions for correct image display are:

- Free run is set or
- at least one trigger occurs
- Image transferal is active at: SensoConfig/Job/Image Transferal

Step 2

Select "Configure Archiving" under SensoView/File.

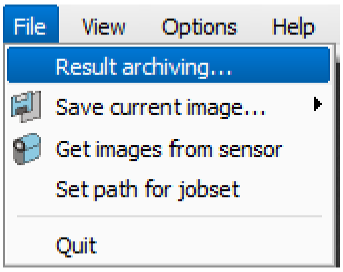


Fig. 334: SensoView, Archiving

The following dialog for setting the parameters for archiving is then displayed.

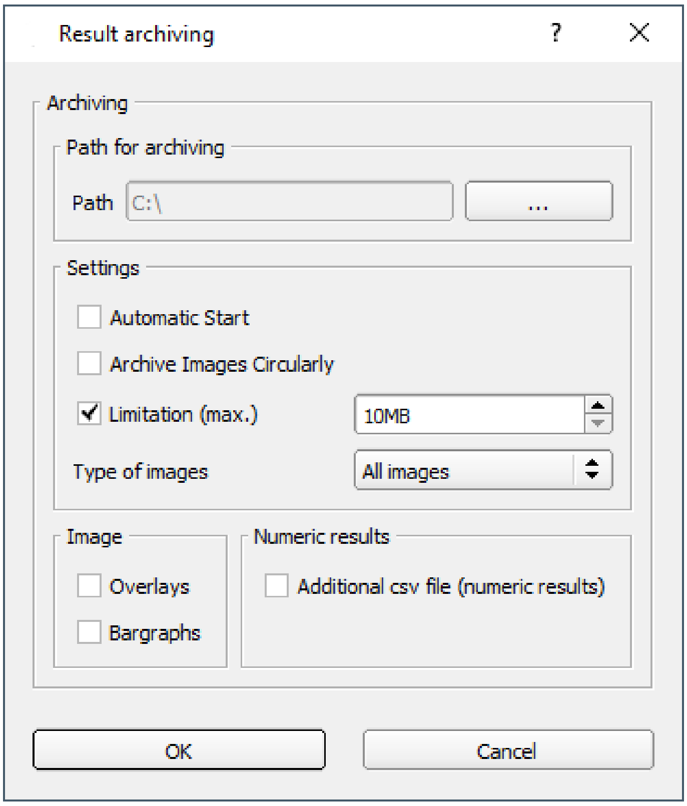
Parameter	Function
Path for archiving	Directory in which the archived file(s) are stored.
Settings, Automatic start	Starts archiving automatically after starting SensoView.
Settings, Cyclical Overwriting	Activates cyclic overwriting of oldest images if storage limit is reached.

Parameter	Function
Settings, Storage limit	Here, it is possible to limit the data volume.
Settings, Image type	In this drop-down list, it can be specified which images (all image or only good or bad images) should be stored.
Overlays, Bar graph result	The image data can be stored in various ways. By activating "Overlays", the frames of detectors and Alignment are also saved. By activating "Result Bar Graph", the result bars of detectors and Alignment are also saved. If none of the options are activated here, the image data are saved in a raw state.
Numerical results	If "Log" is activated, numerical result data such as coordinate values, etc. are archived in an additional .csv file. The Legacy / Configured determines the format of the CSV file. For "Legacy", the content of the .csv file is predefined; for "Configured", this can be freely configured via "Output / Data Output".

Select the desired options and confirm your choice with OK.

Start / end archiving:

Click on the button "Archive Images" in the window "Commands" to start or end the archiving function. The status bar displays the name of the image file currently being saved. Archiving is carried out for as long as the button "Archive Images" is pressed.



Result archiving

Archiving

Path for archiving

Path: C:\

Settings

□ Automatic Start

□ Archive Images Circularly

☑ Limitation (max.): 10MB

Type of images: All images

Image

□ Overlays

□ Bargraphs

Numeric results

□ Additional csv file (numeric results)

Fig. 335: SensoView, Configure archiving

11.4 Archiving via FTP or SMB

This allows images and numerical data (in csv format) to be archived by the sensor via FTP / SMB.

This archiving is configured under "Output/Archiving".

For this type of archiving:

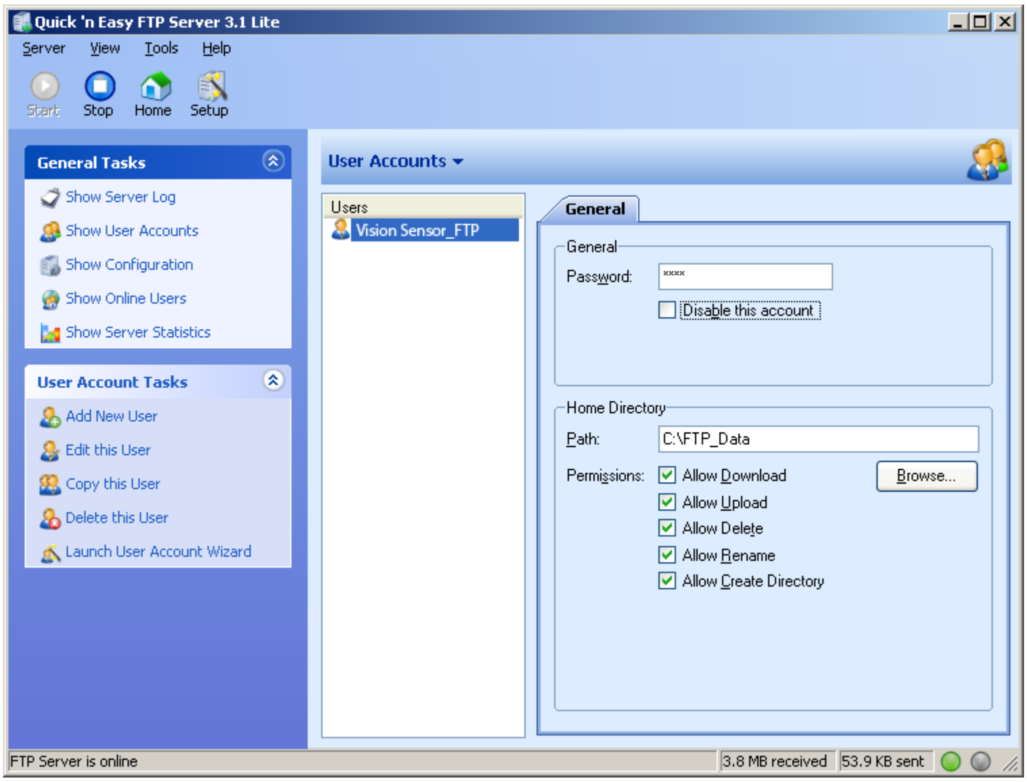
a. **for FTP**: The sensor is an "FTP client" and "writes" the data to an "FTP server" available in the network. With Job Start, the sensor connects to the FTP server.
b. **for SMB**: The sensor "writes" its data directly into a network-shared directory. With Job Start, the sensor connects with this directory.

When using this type of image and result data archiving in normal use cases, neither the SensoFind nor the SensoConfig PC application is active. Instead, only an FTP or SMB server configured to communicate with the VISOR® will be active.

11.4.1 Example: Archiving via FTP

In the example here, an exemplary FTP communication was established with the freely available FTP server software "Quick'n Easy FTP Server" and image and result data were saved on the hard disk of the PC.

The account wizard was used to create a user account named "VISOR_FTP" on the FTP server. A password and a path for data storage are specified, and upload and download are enabled.



Users: Vision Sensor_FTP

General

Password: ****

☐ Disable this account

Home Directory

Path: C:\FTP_Data

Permissions:
☑ Allow Download
☑ Allow Upload
☑ Allow Delete
☑ Allow Rename
☑ Allow Create Directory

FTP Server is online | 3.8 MB received | 53.9 KB sent

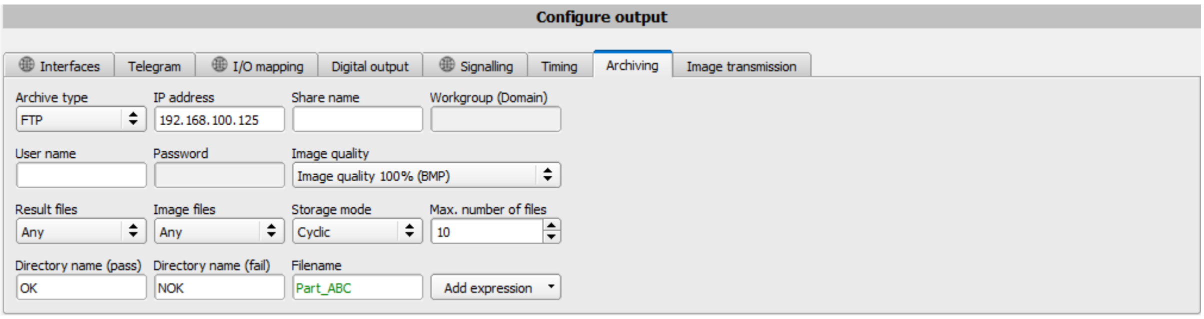
Fig. 336: FTP Server

In SensoConfig, in Output/Archiving, the appropriate settings for the FTP server still need to be configured on the VISOR®. In addition:

- Archive type = FTP
- IP address = IP of the PC on which the FTP server is running (can be seen in SensoFind in the status bar, bottom left)
- User name = Name of the user account in the FTP server
- Password = Password assigned in the FTP account (optional)

With this, the matching settings necessary for the FTP are made.

Here, additional settings, e.g. data name, max. number of files, and storage method (e.g. "cyclic") can / should be made.

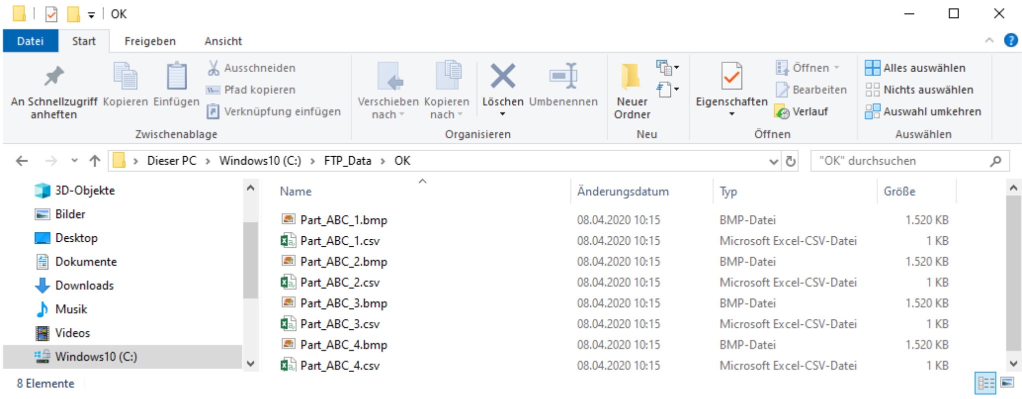


Configure output

Archive type: FTP	IP address: 192.168.100.125	Share name:	Workgroup (Domain):
User name:	Password:	Image quality: Image quality 100% (BMP)	
Result files: Any	Image files: Any	Storage mode: Cyclic	Max. number of files: 10
Directory name (pass): OK	Directory name (fail): NOK	Filename: Part_ABC	

Fig. 337: FTP server, settings in SensoConfig

Once these settings have been configured and transferred to the VISOR® using "Start sensor," the image and result data will be saved on the PC in the specified directory without the SensoFind, SensoConfig or SensoView applications needing to be active.



Name	Änderungsdatum	Typ	Größe
Part_ABC_1.bmp	08.04.2020 10:15	BMP-Datei	1.520 KB
Part_ABC_1.csv	08.04.2020 10:15	Microsoft Excel-CSV-Datei	1 KB
Part_ABC_2.bmp	08.04.2020 10:15	BMP-Datei	1.520 KB
Part_ABC_2.csv	08.04.2020 10:15	Microsoft Excel-CSV-Datei	1 KB
Part_ABC_3.bmp	08.04.2020 10:15	BMP-Datei	1.520 KB
Part_ABC_3.csv	08.04.2020 10:15	Microsoft Excel-CSV-Datei	1 KB
Part_ABC_4.bmp	08.04.2020 10:15	BMP-Datei	1.520 KB
Part_ABC_4.csv	08.04.2020 10:15	Microsoft Excel-CSV-Datei	1 KB

Fig. 338: Transferring files with FTP

Archiving via SMB takes place analogously via an smb server, which must be set accordingly.

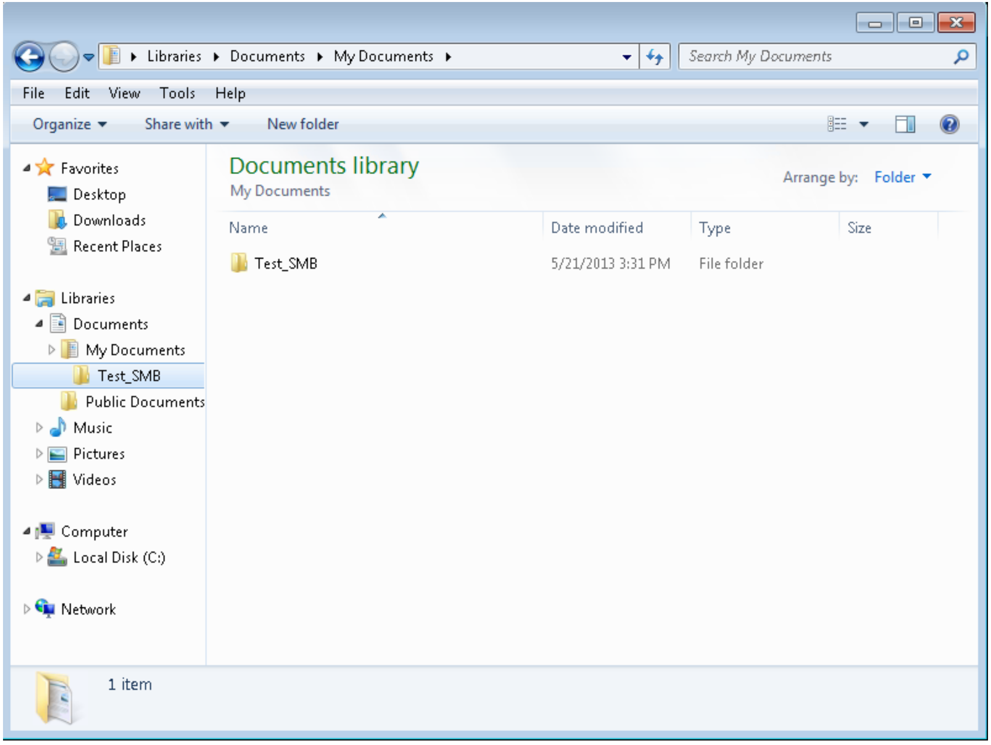
11.4.2 Example: Archiving via SMB

For data and / or image archiving via SMB (server message block), a folder must be shared from the PC.

The following example shows some exemplary settings for setting up data archiving via SMB.

11.4.2.1 Settings for SMB on PC: Create folder and share it

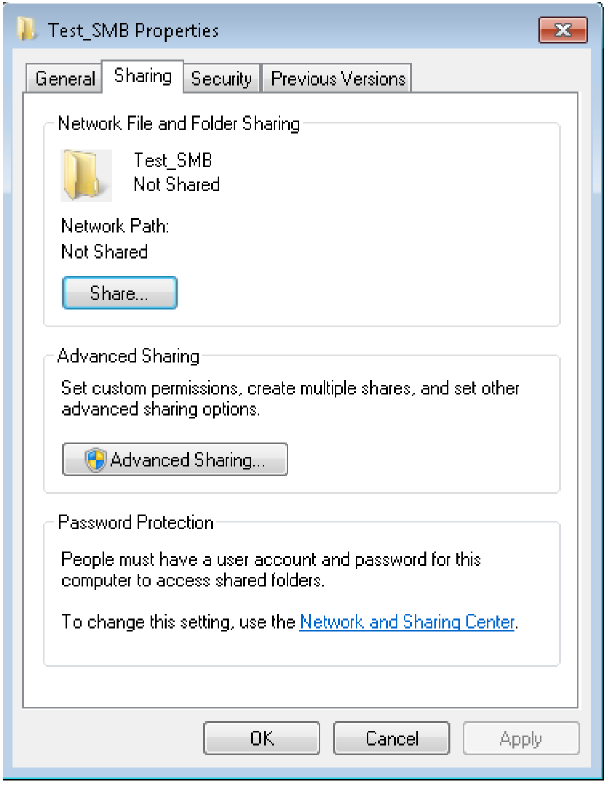
1. By right clicking on the folder (here "Test_SMB"), select the menu item "properties".



Name	Date modified	Type	Size
Test_SMB	5/21/2013 3:31 PM	File folder	

Fig. 339: Create folder to be described, here for example: "Test_SMB".

2. In the following dialog, "Properties of Test_", open the tab "Sharing" and click on "Advanced Sharing".



Test_SMB
Not Shared

Network Path:
Not Shared

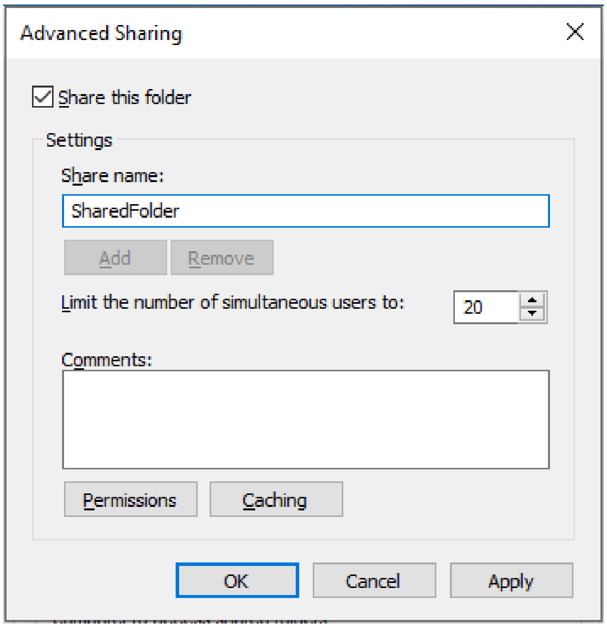
Set custom permissions, create multiple shares, and set other advanced sharing options.

People must have a user account and password for this computer to access shared folders.

To change this setting, use the Network and Sharing Center.

Fig. 340: Folder sharing > Advanced sharing

3. In the dialog "Advanced Sharing", activate "Share this folder". Here, the name of the folder "Test_SMB" is suggested as a "share name". Here, any other name can be set. In this example the suggested folder name is used.
   Important: You will later on need to enter this share name in the VISOR® SMB interface exactly the way it is entered here!



Advanced Sharing

☑ Share this folder

Settings

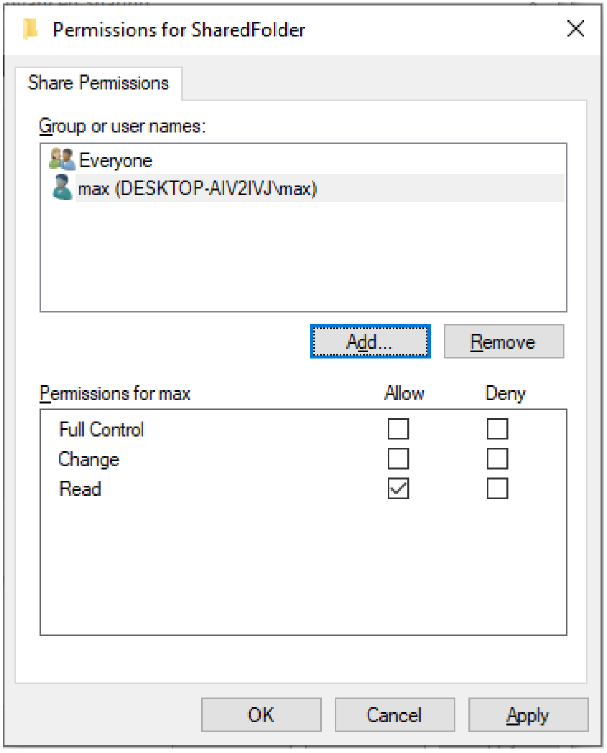
Share name: SharedFolder

Limit the number of simultaneous users to: 20

Comments:

Fig. 341: Set share name

4. By clicking on "Permissions", the following dialog appears.



Permissions for SharedFolder

Share Permissions

Group or user names:

Everyone
max (DESKTOP-AIV2IVJ\max)

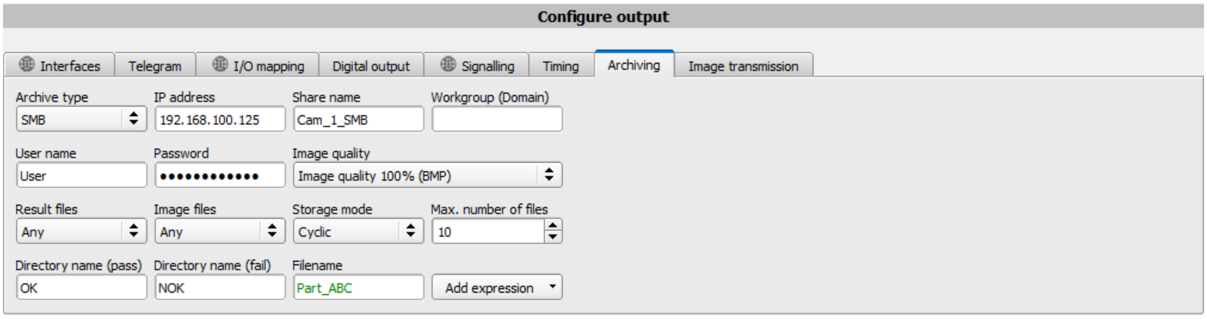
Permissions for max	Allow	Deny
Full Control	□	□
Change	□	□
Read	☑	□

Fig. 342: Set permissions

5. In the window "Test_SMB Permissions", select a user (here "fsc") (for which the user name and password are known). You will need to enter the username and password in the VISOR® SMB interface later on.

6. Activate "Full control" and close the dialog with "Apply" and "OK".
7. Then also close the dialog "Advanced Sharing" and "Test_SMB Properties" with "Apply" and "OK".
8. Access for the user selected here has now been set up on the PC. The corresponding settings in the VISOR® "SensoConfig" interface can be configured now.

11.4.2.2 SMB setup



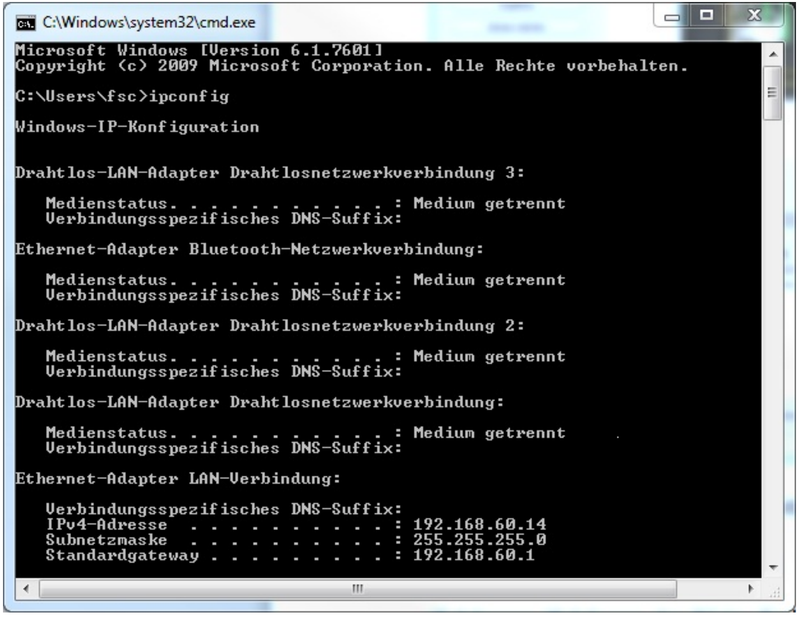
Archive type: SMB
IP address: 192.168.100.125
Share name: Cam_1_SMB
Workgroup (Domain):
User name: User
Password: ●●●●●●●●●●●●
Image quality: Image quality 100% (BMP)
Result files: Any
Image files: Any
Storage mode: Cyclic
Max. number of files: 10
Directory name (pass): OK
Directory name (fail): NOK
Filename: Part_ABC

Fig. 343: Settings in VISOR® SMB Interface

After starting SensoConfig, select "SMB" under Output/Archiving/Archiving type.

Make the following entries:

- IP address: IP address of the PC (can be found with command "ipconfig" under Start/Run/cmd, see following screenshot). In this example: 192.168.60.14



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\fsc>ipconfig

Windows-IP-Konfiguration


Drahtlos-LAN-Adapter Drahtlosnetzwerkverbindung 3:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Ethernet-Adapter Bluetooth-Netzwerkverbindung:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Drahtlos-LAN-Adapter Drahtlosnetzwerkverbindung 2:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Drahtlos-LAN-Adapter Drahtlosnetzwerkverbindung:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Ethernet-Adapter LAN-Verbindung:

   Verbindungsspezifisches DNS-Suffix:
   IPv4-Adresse  . . . . . . . . . . : 192.168.60.14
   Subnetzmaske  . . . . . . . . . . : 255.255.255.0
   Standardgateway . . . . . . . . . : 192.168.60.1
```

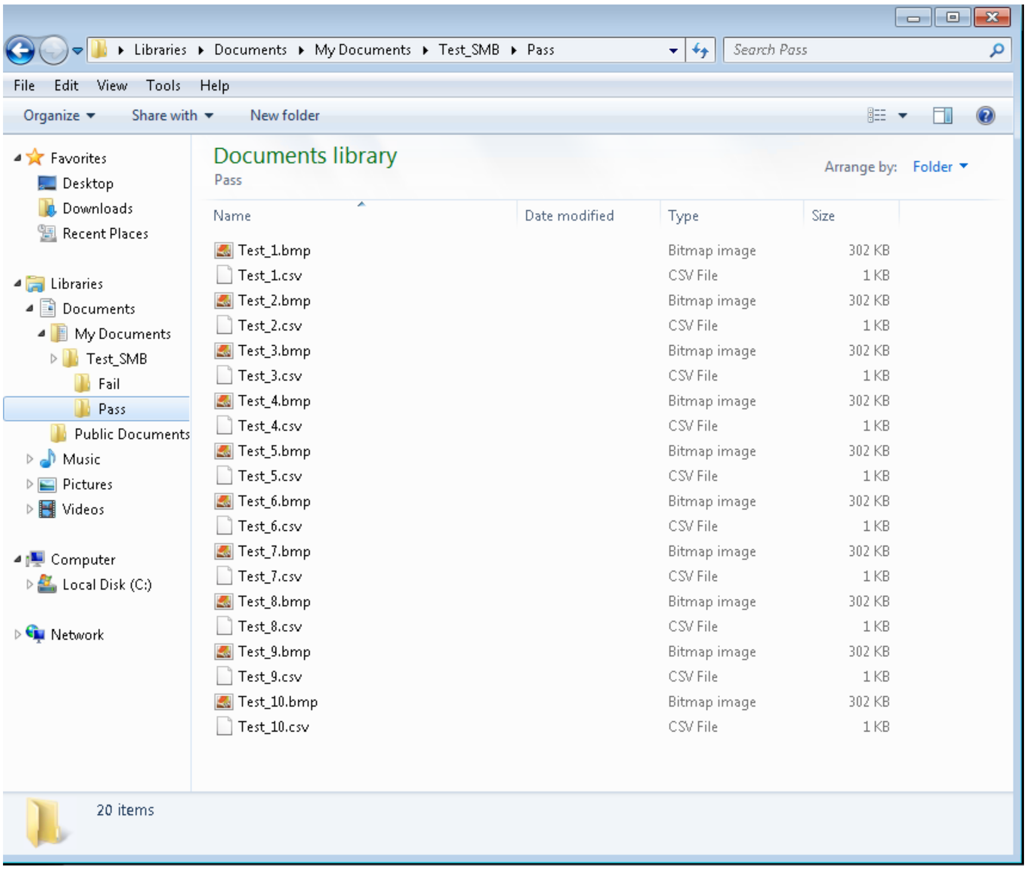
Fig. 344: IP address of the PC via Start/Run/cmd/ipconfig

- Share name: Here, enter share name as previously specified on the PC in the dialog "Advanced Sharing".
- Workgroup (domain): Optional! Enter name of workgroup.
- User name and password: Depending on the selection made in the dialog "Test_SMB Permissions", the following must be performed in the case of:
  1. User group "Everyone": Keep user name and password empty
  2. Enter corresponding user name with password (here in this example for "fsc")

- Directory Good parts, Directory Bad parts: Here is a name for the folder in which the data and images to be archived should be stored in the case of a good or bad part. These folders are created below the folder to be described and shared (here: "Test_SMB").
- File name: Enter any file name here for the output file.
- Result files: If the output for the result data is activated, all data specified under "Output / Data Output" are protocolled in a .csv file. A file is created for each evaluation (trigger). The files are numbered consecutively.
- Image files: Archiving of images as .bmp: None, All, Only good parts, Only bad parts
- Storage mode: Limited: Once the maximum number of files is reached, the transferal is terminated. Unlimited: Files are written until the target drive is full. Cyclic: After reaching the maximum number of files, the oldest file is overwritten by the newest.
- Max. number of files: Maximum number of file sets which are allowed to be stored in the target directory.

11.4.2.3 Archiving via SMB, Output data

After starting of the sensor, images are archived in the shared directory in the corresponding subfolder, and the data are archived as a .csv file, which has been specified under SensoConfig/Output/Data Output.



Name	Date modified	Type	Size
Test_1.bmp		Bitmap image	302 KB
Test_1.csv		CSV File	1 KB
Test_2.bmp		Bitmap image	302 KB
Test_2.csv		CSV File	1 KB
Test_3.bmp		Bitmap image	302 KB
Test_3.csv		CSV File	1 KB
Test_4.bmp		Bitmap image	302 KB
Test_4.csv		CSV File	1 KB
Test_5.bmp		Bitmap image	302 KB
Test_5.csv		CSV File	1 KB
Test_6.bmp		Bitmap image	302 KB
Test_6.csv		CSV File	1 KB
Test_7.bmp		Bitmap image	302 KB
Test_7.csv		CSV File	1 KB
Test_8.bmp		Bitmap image	302 KB
Test_8.csv		CSV File	1 KB
Test_9.bmp		Bitmap image	302 KB
Test_9.csv		CSV File	1 KB
Test_10.bmp		Bitmap image	302 KB
Test_10.csv		CSV File	1 KB

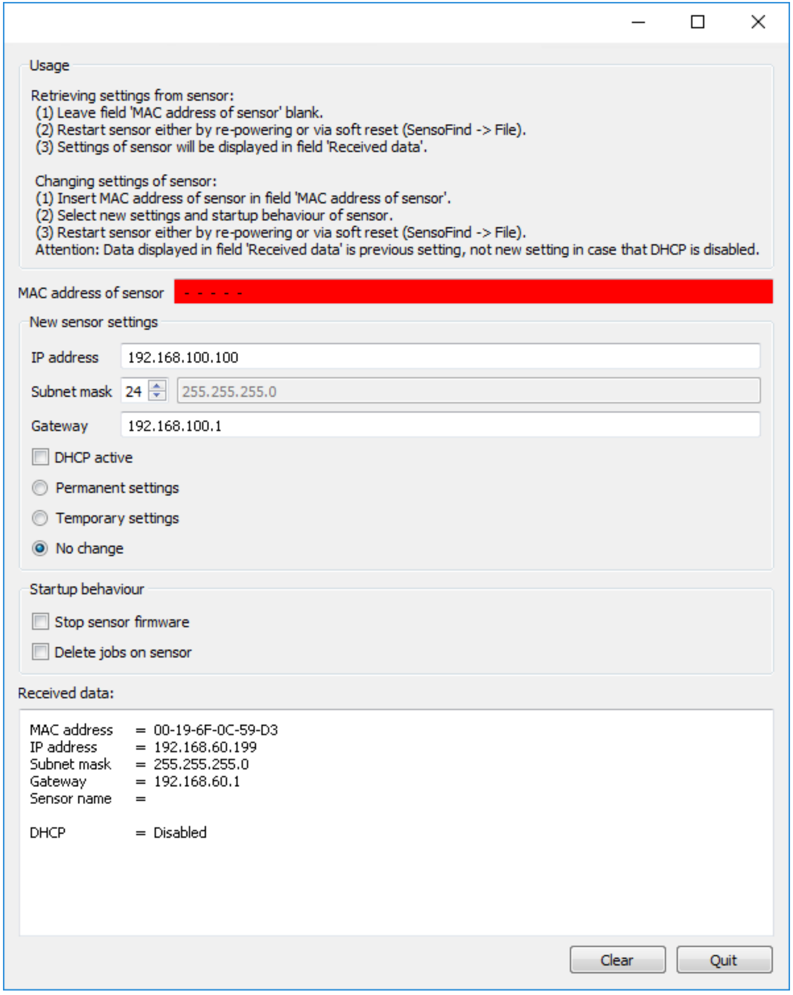
20 items

Fig. 345: *Successfully performed archiving via SMB*

11.5 SensoRescue

The utility "SensoRescue" is used to reset VISOR® vision sensors that can no longer be found with SensoFind to a state from which they can be addressed and parametrized again by SensoFind and SensoConfig.

1. Start SensoRescue (leave field "Mac address of Sensor" empty)
2. Restart VISOR®, Power off / on or SensoFind/File (VISOR® must be connected to the same network as the PC via Ethernet connection)
3. The "Received Data" field below will show the settings for the VISOR® vision sensor.



Usage

Retrieving settings from sensor:
(1) Leave field 'MAC address of sensor' blank.
(2) Restart sensor either by re-powering or via soft reset (SensoFind -> File).
(3) Settings of sensor will be displayed in field 'Received data'.

Changing settings of sensor:
(1) Insert MAC address of sensor in field 'MAC address of sensor'.
(2) Select new settings and startup behaviour of sensor.
(3) Restart sensor either by re-powering or via soft reset (SensoFind -> File).
Attention: Data displayed in field 'Received data' is previous setting, not new setting in case that DHCP is disabled.

MAC address of sensor - - - - -

New sensor settings

IP address 192.168.100.100

Subnet mask 24 255.255.255.0

Gateway 192.168.100.1

□ DHCP active

○ Permanent settings

○ Temporary settings

● No change

Startup behaviour

□ Stop sensor firmware

□ Delete jobs on sensor

Received data:

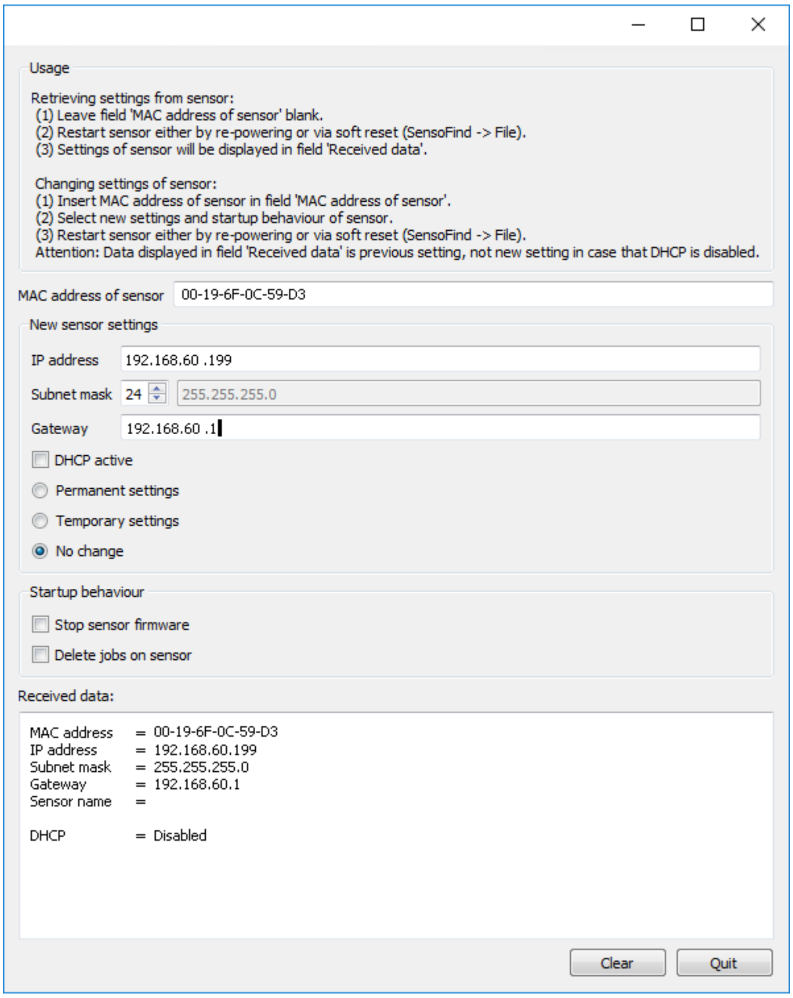
```
MAC address   = 00-19-6F-0C-59-D3
IP address    = 192.168.60.199
Subnet mask   = 255.255.255.0
Gateway       = 192.168.60.1
Sensor name   =

DHCP          = Disabled
```

Fig. 346: *SensoRescue /1*

4. Now the Mac address shown below can be entered into the field "Mac address of Sensor"
5. You can enter all the network settings, e.g., IP address, subnet mask, etc. that you want the VISOR® to have after the next restart (power off / on) underneath.
   Restart the VISOR®

NOTE:
The data displayed in the lower field will not be updated upon reboot.



Usage

Retrieving settings from sensor:
(1) Leave field 'MAC address of sensor' blank.
(2) Restart sensor either by re-powering or via soft reset (SensoFind -> File).
(3) Settings of sensor will be displayed in field 'Received data'.

Changing settings of sensor:
(1) Insert MAC address of sensor in field 'MAC address of sensor'.
(2) Select new settings and startup behaviour of sensor.
(3) Restart sensor either by re-powering or via soft reset (SensoFind -> File).
Attention: Data displayed in field 'Received data' is previous setting, not new setting in case that DHCP is disabled.

MAC address of sensor: 00-19-6F-0C-59-D3

New sensor settings

IP address: 192.168.60 .199

Subnet mask: 24 | 255.255.255.0

Gateway: 192.168.60 .1

□ DHCP active

○ Permanent settings

○ Temporary settings

● No change

Startup behaviour

□ Stop sensor firmware

□ Delete jobs on sensor

Received data:

```
MAC address   = 00-19-6F-0C-59-D3
IP address    = 192.168.60.199
Subnet mask   = 255.255.255.0
Gateway       = 192.168.60.1
Sensor name   =

DHCP          = Disabled
```

Fig. 347: *SensoRescue / 2*

NOTE:
If no data is received, it is possible that the connection was closed by the operating system. This connection can be re-established via the "Port Reset" button. Then start again with step 2.

12 Accessories

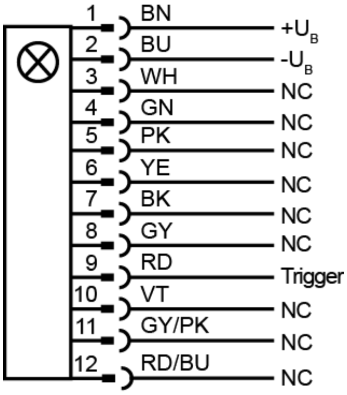
External illumination

There is an extensive range of accessories available for the VISOR®. It includes a variety of external illuminations, which can be used additionally or instead of the internal illumination.

For more information on vision accessories, please refer to: http://www.sensopart.com/en/download.

Both types, LF 45 xxxLFR 115 and xxx, can be connected directly to the sensor.

Connection



Connection Ring light with sensor

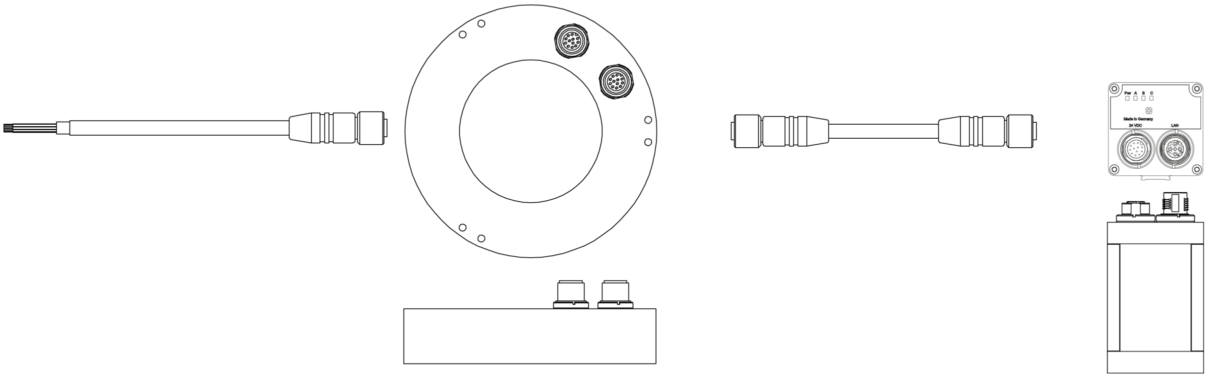
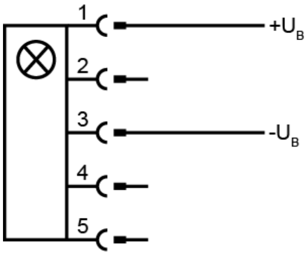


Fig. 348: *Connection of external illumination LF 45LFR 115 xxx and xxx*

All other models are connected to the VISOR® as follows.

Connection



Connection Ring light - connection adapter - sensor

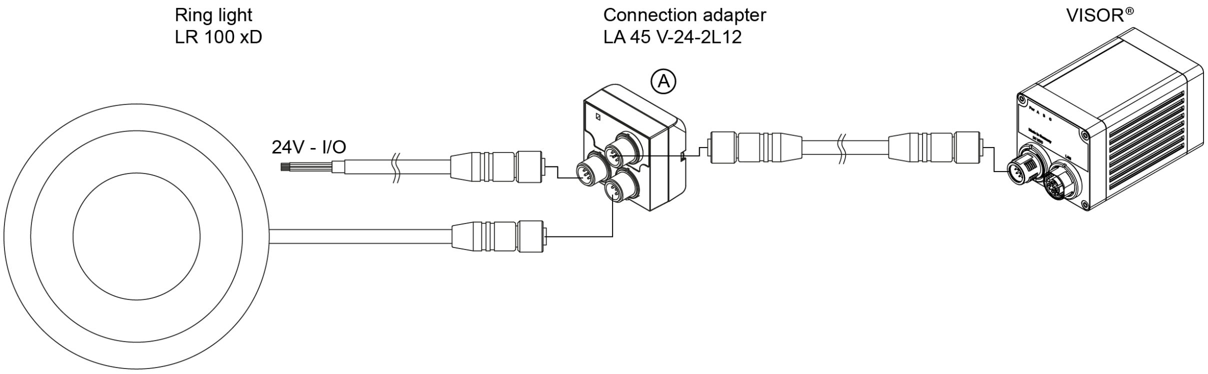


Fig. 349: *Connection of external illumination, all types except LF 45 xxx and LFR 115 xxx*

13 Technical data

Electrical data	
Operating Voltage U_B	24 V DC (18 V - 30V)
Reverse polarity protection	✓
Short circuit protection	✓
Residual ripple	< 5 Vss, level 3 EN 61000-4-17
Startup time	< 14 s
Current consumption (without I/O)	≤ 300 mA
Input/output polarity	PNP / NPN
Switching threshold for all inputs, incl. encoder	High > V_B - 1 V, Low < 3 V
Input resistance	> 20 kOhm
Encoder input	40 kHz
Max. output current per output	50 mA, Ejector (Pin 12 / RDBU) 100 mA
Total current (all active outputs)	Max. 200 mA
Inductive load	Typical: Relay: 17 K / 2 H (50 mA outputs), Pneumatic valve: 1.4 K / 190 mH (100 mA output)
Capacitive load	900 nF for ejector (pin 12 / RDBU); otherwise 500 nF
t_{rise} steepness of switching outputs	If 50 mA or 4.7 kohm pull-up / pull-down PNP: < 300 ns NPN: < 4 us
t_{fall} steepness of switching outputs	If 50 mA or 4.7 kohm pull-up / pull-down NPN: < 200 us PNP: < 400 us
Interfaces	100 Mbit LAN, PROFINET, EtherNet/IP, SensoWeb

Optical data	V10 / V10C	V20 / V20C	V50 / V50C
Number of pixels (H x V)	800 x 600	1440 x 1080	2560 x 1936
Sensor size	1 / 3.6"	1 / 2.9"	1 / 1.8"
Pixel size	4.8 µm x 4.8 µm	3.45 µm x 3.45 µm	2,8 µm x 2,8 µm
Technology	CMOS Mono / Color		
Light type	LED Red / White / Infrared / UV		
Target laser	Red, laser class 1		
Projector laser	Red, laser class 2 [1)]		
Integrated lens, Focal length	5.2 (Wide) 9.6 (Medium) 20 (Narrow)	6.5 (Wide) 12 (Medium) 20 (Narrow)	9.6 (Wide) 20 (Medium) -
Focus (Working distance)	Motorized		
Max. number of adjustment cycles per hour	60		

[1)] Only Robotic VISOR® +Z

Mechanical data	
Weight	Approx. 200 g
Ambient air temperature: operation	0 °C ... 50 °C (80 % humidity, non-condensing) [1) 2)]
Ambient air temperature: storage	-20 °C ... 60 °C (80% humidity, non-condensing)
Protection class	IP 65/ IP 67 EN 60529
Housing material	Die-cast aluminum
[1)] If you use the spark protection guard, the maximum operating temperature is lowered to 45 °C. For ambient temperatures above 45 °C, it is recommended to use the Mounting hinge MG 3A. [2)] Robotic VISOR® +Z 0 °C ... 40 °C	

Tests	
Vibration resistance	EN 60068-2-6
Shock resistance	EN 60068-2-27
LABS-free	Yes
EMC	EN 61000-6-2 / EN 55011
ESD Safe	IEC 61340- 5-1:2016 / DIN EN 61340- 5-1:2017 with the accessory LPC Vxx ESD, order number 651-01010

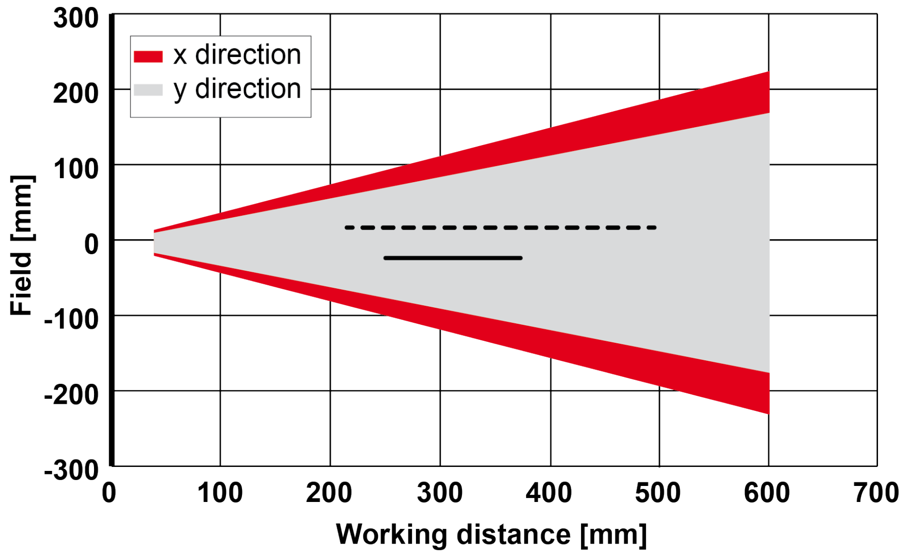
Other certification	
FDA CFR 21, Part 11	The VISOR® vision sensor provides features that can be used to implement an auditing program that complies with "CFR 21 Part 11". It is the user's responsibility to set up a compliant system using the functions provided. These functions include: • System or software backup and restore (backup of jobsets or firmware) • Protection of system backup files against modifications (jobset password protection) • System software security (jobset password protection) • Timestamp information during data output (extended triggers or archiving options) • Verification of the integrity of jobs and results (checksum function)

Typical cycle time	
Monochrome detectors	Barcode: 30 ms Datacode: 40 ms OCR: 15 ms per character Pattern matching: 20 ms Contour: 30 ms Contour 3D: 300 ms Target Mark 3D: 50 ms Contrast: 2 ms Brightness: 2 ms Gray: 2 ms Caliper: 8 ms Classification (AI): 225 ms - 900 ms BLOB: 30 ms Wafer: 100 ms Busbar: 100 ms
Color Detectors	Color Value: 2 ms Color Area: 30 ms Color List: 2 ms

14 Field of view and depth of field

14.1 VISOR® V50

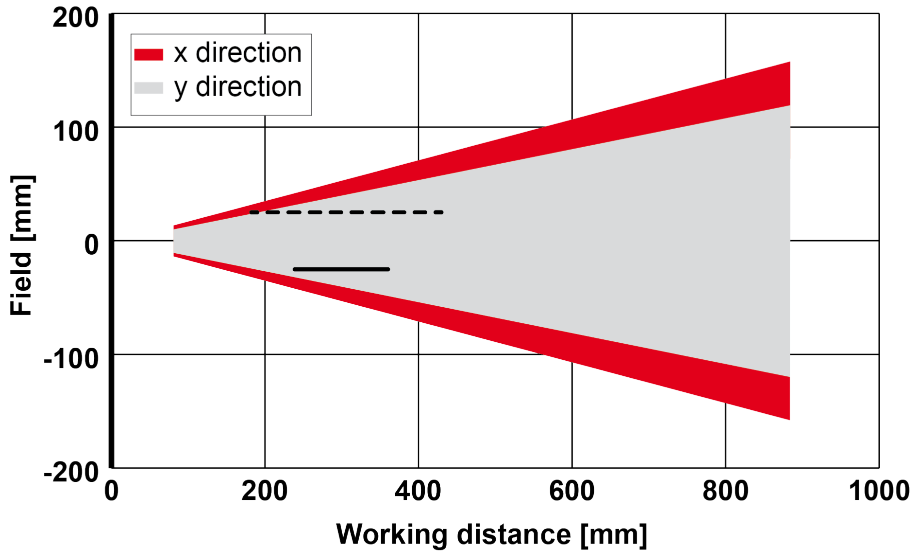
V50, Field of view Wide, internal lens



- - - - - Depth of field: Enhanced ——— Depth of field: Normal

Fig. 350: *Depth of field V50, Field of view Wide, internal lens*

V50, Field of view Medium, internal lens

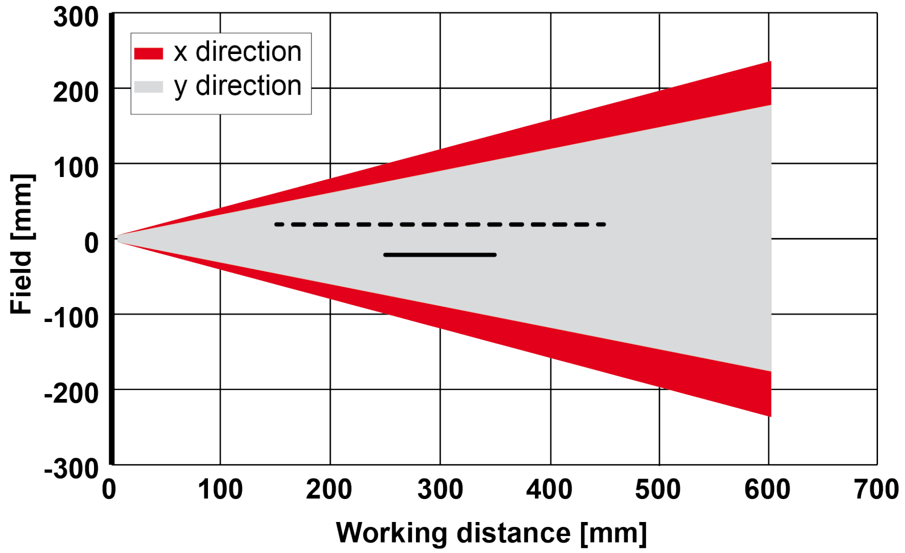


- - - - - Depth of field: Enhanced ——— Depth of field: Normal

Fig. 351: *Depth of field V50, Field of view Medium, internal lens*

14.2 VISOR® V20

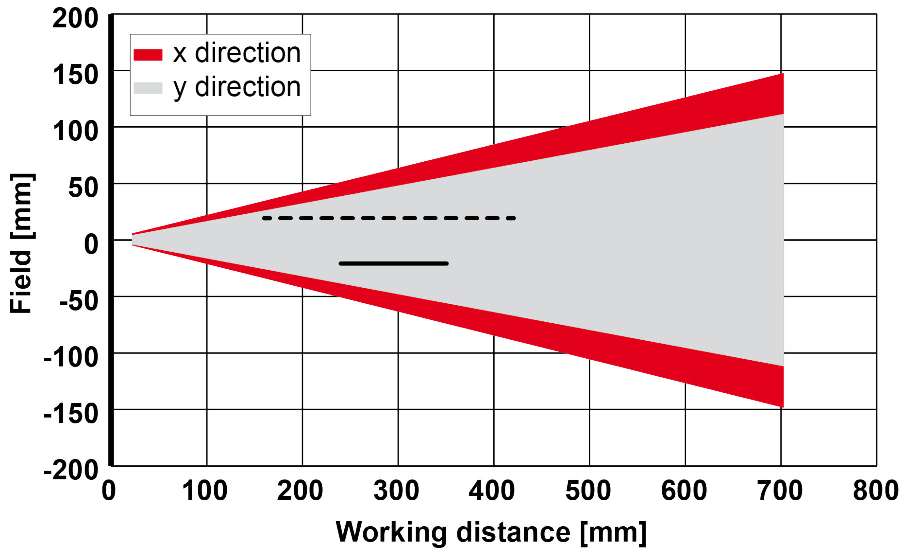
V20, Field of view Wide, internal lens



- - - - - Depth of field: Enhanced ——— Depth of field: Normal

***Fig. 352:** Depth of field V20, Field of view Wide, internal lens*

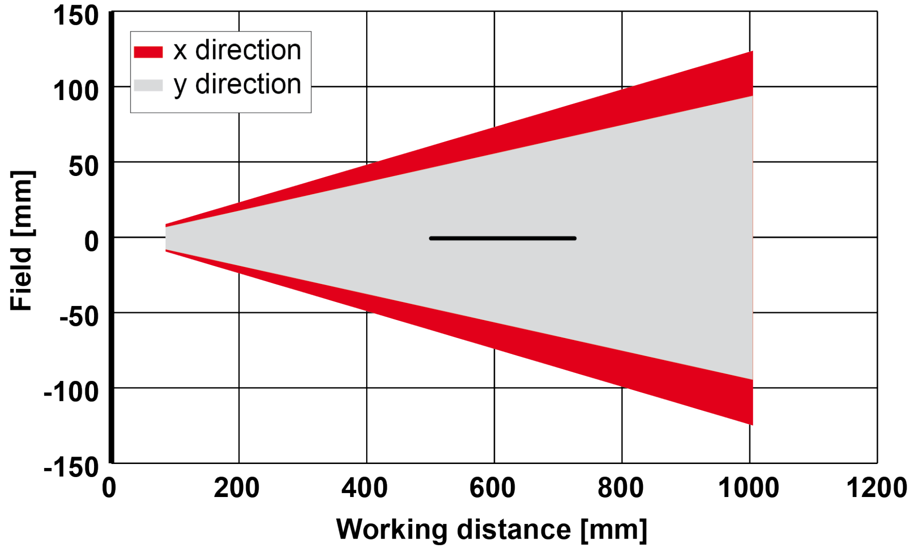
V20, Field of view Medium, internal lens



- - - - - Depth of field: Enhanced ——— Depth of field: Normal

***Fig. 353:** Depth of field V20, Field of view Medium, internal lens*

V20, Field of view Narrow, internal lens

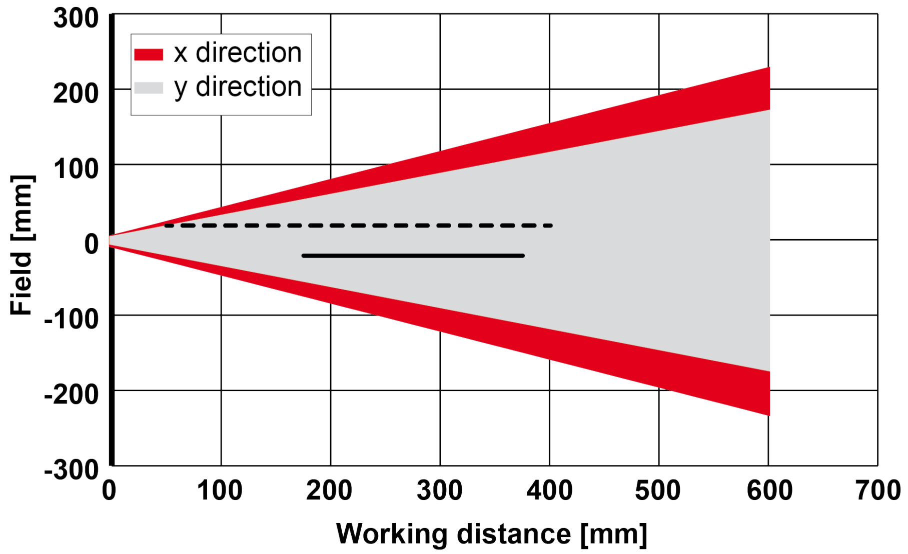


- - - - - Depth of field: Enhanced ——— Depth of field: Normal

Fig. 354: Depth of field V20, Field of view Narrow, internal lens

14.3 VISOR® V10

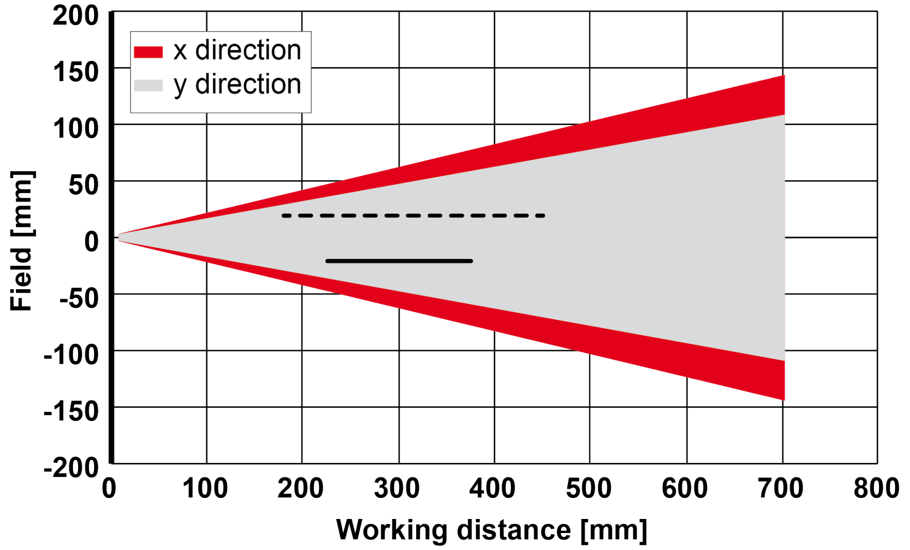
V10, Field of view Wide, internal lens



- - - - - Depth of field: Enhanced ——— Depth of field: Normal

Fig. 355: Depth of field V10, Field of view Wide, internal lens

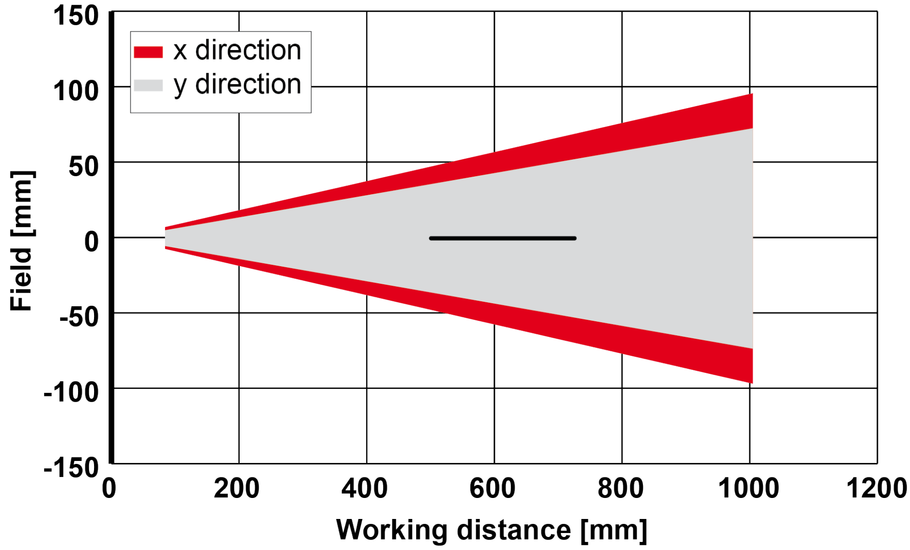
V10, Field of view Medium, internal lens



- - - - - Depth of field: Enhanced ——— Depth of field: Normal

Fig. 356: Depth of field V10, Field of view Medium, internal lens

V10, Field of view Narrow, internal lens



- - - - - Depth of field: Enhanced ——— Depth of field: Normal

Fig. 357: Depth of field V10, Field of view Narrow, internal lens

15 Maintenance

15.1 Maintenance

The following maintenance work must be performed for the vision sensor at regular intervals:

- Clean the vision sensor
- Check all connectors and fittings

15.2 Cleaning

The vision sensor's housing must be cleaned with a clean, dry cloth.

If the sensor's front is soiled, it must be cleaned with a soft cloth and a bit of plastic cleaner if necessary

ATTENTION:
Please note that improperly cleaning the front can damage it:

- Never use aggressive detergents such as solvents or benzine.
- Do not use any sharp objects; do not scratch the front.

15.3 Repairs

The vision sensor should be repaired exclusively by the manufacturer. The manufacturer's warranty will be void if you open, alter, or otherwise modify the product.

16 Disposal

This device must be disposed of in accordance with all applicable national environmental regulations and waste disposal regulations.
Since it is e-waste, it is strictly prohibited to dispose of it in household waste.

We look ahead

Yesterday, today and in the future

SENSOPART

SensoPart Industriesensorik GmbH
79288 Gottenheim
Tel.: +49 7665 94769-0
info@sensopart.de

www.sensopart.com