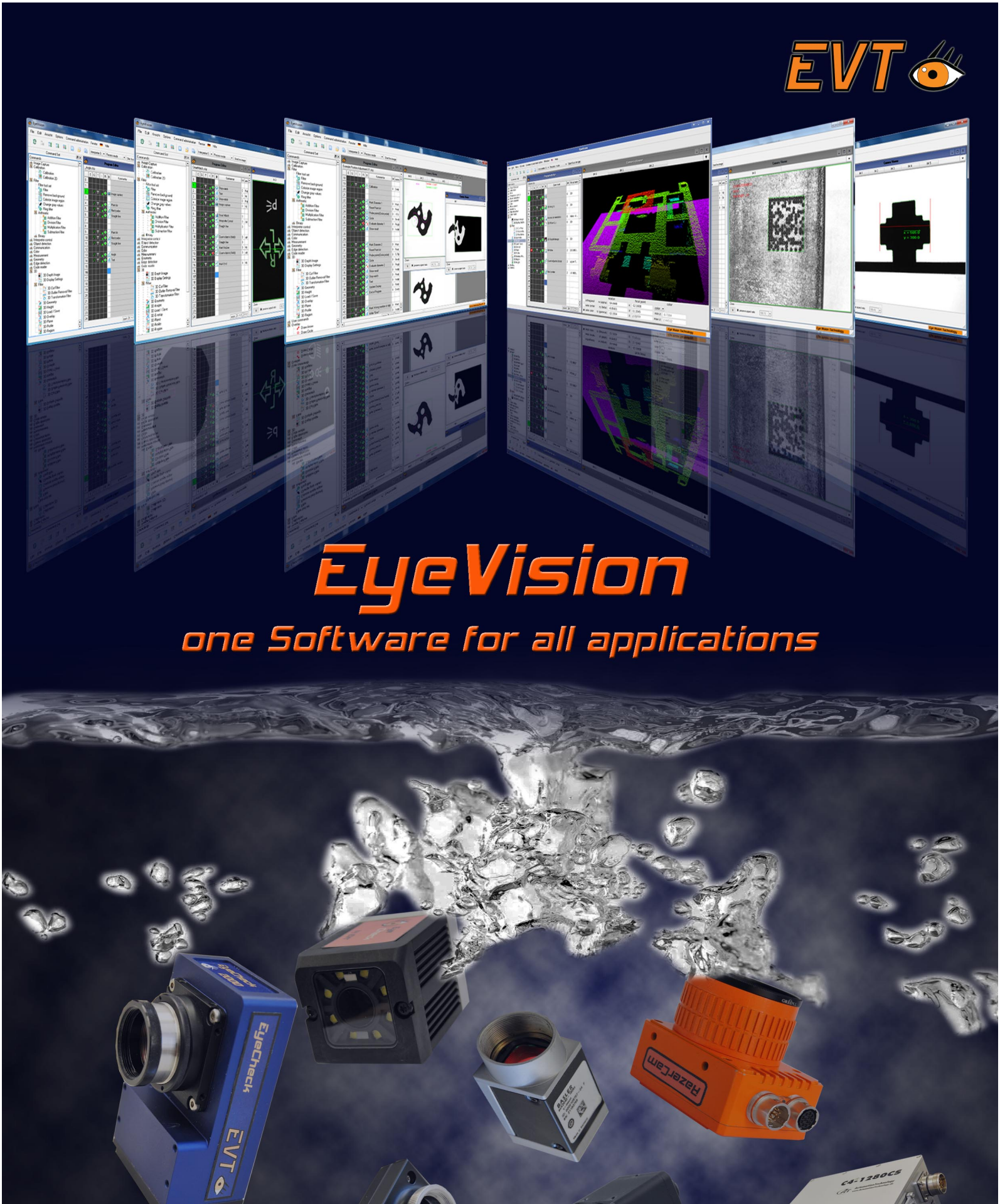

EVT
EyeVision
one Software for all applications
EyeCheck
EVT
HazerCam
C4-1280CS

Documentation EyeVision

Manual for EVT EyeVision Version 3.0

Copyright and Notices

EVT Eye Vision Technology Haid-und-Neu-Str. 7 76131 Karlsruhe Germany

Table of Contents

List of Figures

List of Tables

List of Examples

Introduction

The software EyeVision from EVT Eye Vision Technology in combination with the respective hardware is a complete machine vision tool kit for any industrial automation application, meant to provide inspection solutions quickly and efficiently. In the development and implementation of this image-processing software, it was a guiding idea to provide the professionals as well as the non-experts a tool, in order to solve easily the optical measuring and testing.

Maximum flexibility is offered to these two target groups and at the same time the operation should be as simple as possible. Fortunately the modern graphical user interface allows to solve even complicated tasks by manipulating graphic icons on the screen. The Drag and Drop programming is as powerful as it is intuitive and easy to use. EyeVision have a cross-platform support. So you can use it in the most common operating systems, like Windows, Linux, Mac OS and embedded systems. Even the usage of smart or intelligent cameras(iCam) is allowed by the current computer technology in order to execute inspection procedures in video real-time.

An extensive and modular set of libraries coupled with a Drag and Drop programming enables users to easily implement a wide array of inspections in numerous production industries including automotive, semi-conductor, pharmaceutical, packaging, and medicine. Thanks to the innovative developers at EVT, object detection, blob analysis, and comprehensive measurement can be accomplished in just minutes.

Chapter 1. Getting started

This section contains the following topics:

System requirements

The following list includes the minimum system requirements for the use of EyeVision. Note that for optimum performance, you need a faster CPU, more RAM and hard disc space than indicated in the list.

Table 1.1. System requirements

Property	Architecture	Description
CPU	x86	1 GHz or faster (Intel ATOM, AMD E350 and faster CPU’s)
	ARM	600 MHz or faster (Cortex A9 or higher)
Memory	x86	1 GByte
	ARM	512 MByte
Operating System	x86	Windows XP, Vista, 7 or 8
	ARM	LINUX Ubuntu/Debian/ Angstrom/BusyBox

Storage	**HDD/SSD/Flash** :1 GByte for GUI Systems 512 MByte for non GUI Systems
Licensing	USB/ Serial Number
Graphic board	A minimum screen resolution of 1280 x 800 Pixels
Installation	CD/DVD/USB/SD/Internet

Preparation for the setup

- Make sure that your system’s date and time are set correctly.
- Make sure that you have enough free disk space available on the drive where you want to install the application (have a look at the section called “System requirements”).
- Install EyeVision in a new directory to avoid conflicts with previous versions.
- Get the newest stable EyeVision Version

- Make sure you have the newest operating system version and the drivers for your camera installed.

Installing EyeVision

If you got the EyeVision version (DVD or Download) you can run the setup. Following steps are recommended. This will guide you step by step threw the installation..

Close all other applications

To avoid complications with other software we recommend to close all other applications.

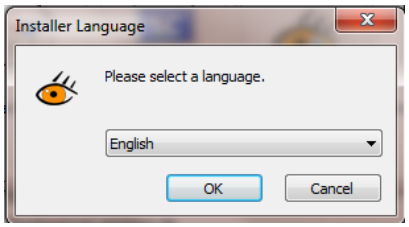
Start the Setup

Start the Setupfile from your DVD or hard disk to run the setup.

Detailed guide installing EyeVision

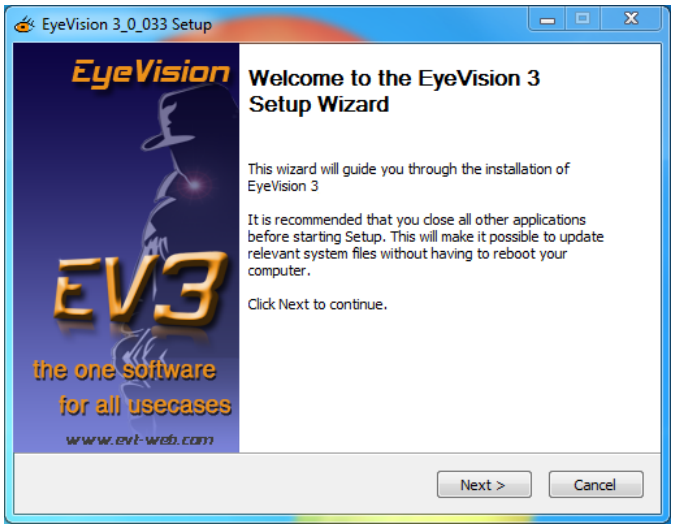
1. You get a small dialog (shown in Fig.: Setup language choice) with an Drop down List button. You should choose your preferred language for the installation and your software. If you choose the wrong language you can restart or change the language in the EyeVision software later.

Figure 1.1. Fig.: Setup language choice



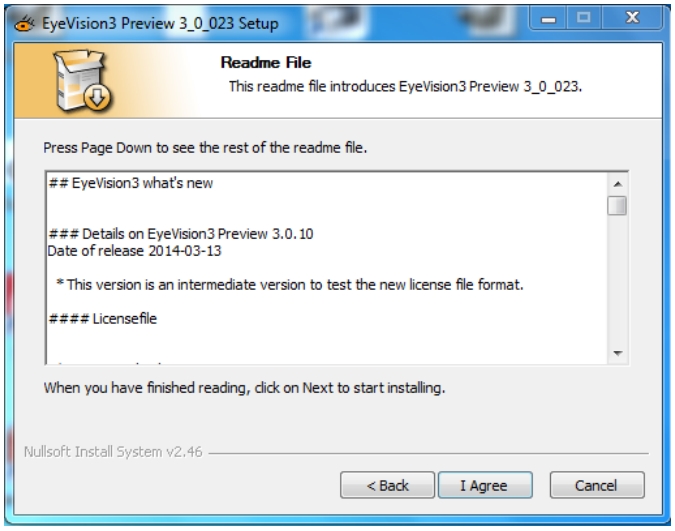
2. By clicking on the "ok" button the setup (see Fig: Setup Startdialog) will start.

Figure 1.2. Fig.: Setup Startdialog



3. After you push the "next" button the Fig: Terms and condition for EyeVision are shown. Read them carefully and accept them by pushing "I Agree". Else you can cancel the setup.

Figure 1.3. Fig: Terms and condition



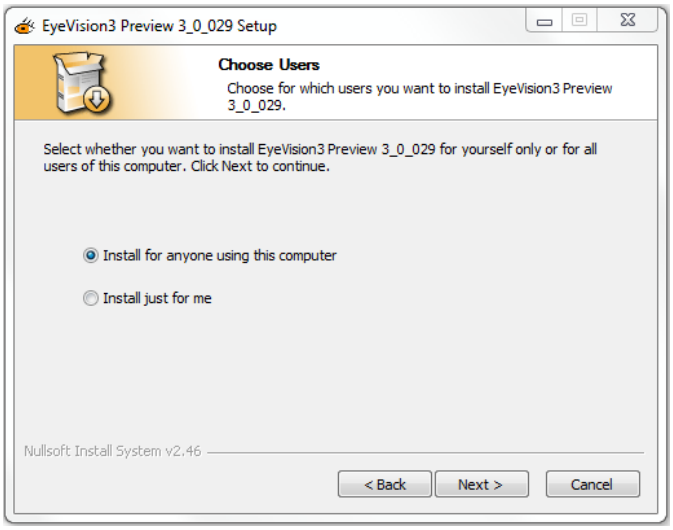
4. Next you have to Fig: Choose the user(s) which will use EyeVision. Here you should choose if the EyeVision software is used by the current user or by all user of this device.

Warning

Depending on your choice, the path of EyeVision directories will be different. If you choose the current user no other user can use EyeVision without fixing the paths.

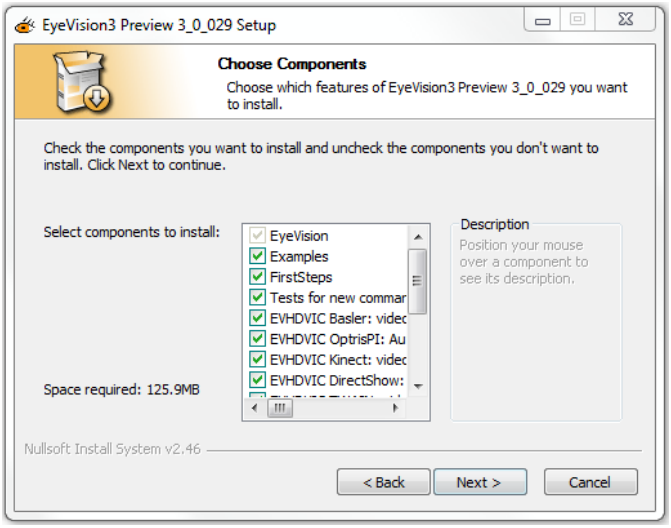
Install for anyone using this computer	This will create a EyeVision directory in the global user directory, (for example Windows 7) *C:\ProgramData\EVT.*
Install just for me	This will create a EyeVision directory in the local user directory (for example Windows 7, user: test) *C:\Users\test\Roaming\EVT.*

Figure 1.4. Fig: Choose the user(s)



5. By pushing next you will get a checklist for all Fig: Components to be installed with EyeVision. Recommended components are already checked. Uncheck components is just for advanced users. All the components are described detailed below.

Figure 1.5. Fig: Components

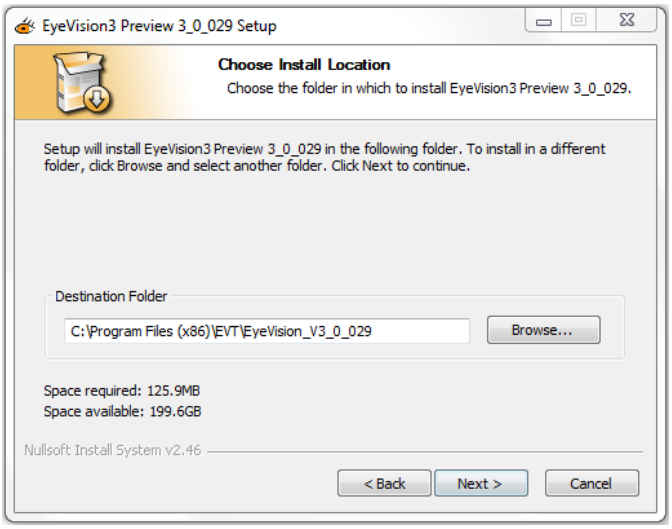


Examples	Some small examples to see how EyeVision can work. This is recommended for standard images to test the software.
First steps	A documentation will guide you threw your first steps. This is recommended for novice users
EVHDVIC Basler/IDS/...	Every option stands for its own video input channel implementations. If you are sure that you only need one of them you can uncheck the rest. Be sure that you always use this video input channel.

6. After you choose the components and push the "next" button you can choose the Fig: Installation location. The recommended location is already chosen. Choose an other location if you want to.

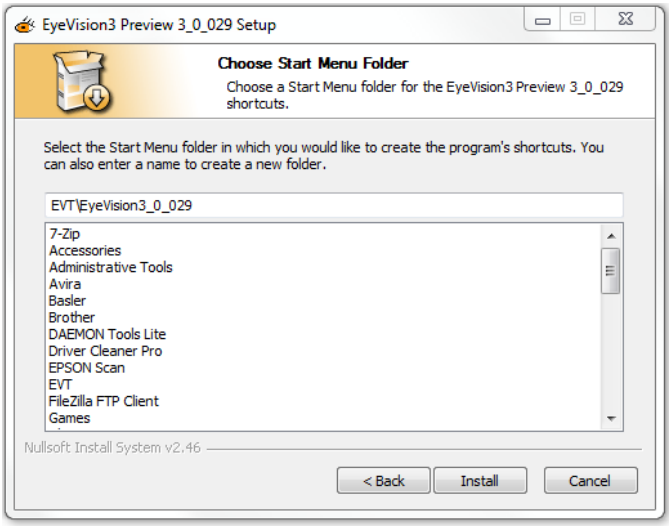
   Click on *Browse* to browse threw your storage.

Figure 1.6. Fig: Installation location



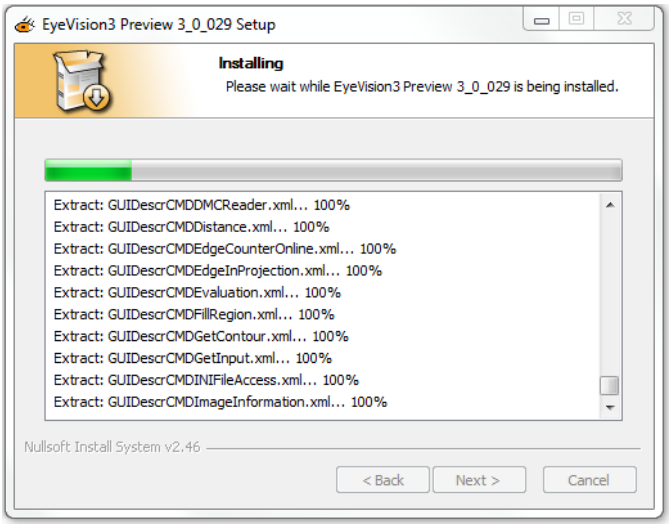
7. After you push next you can choose the Fig: Start menu folder name where you want to get EyeVision shown.

Figure 1.7. Fig: Start menu folder name



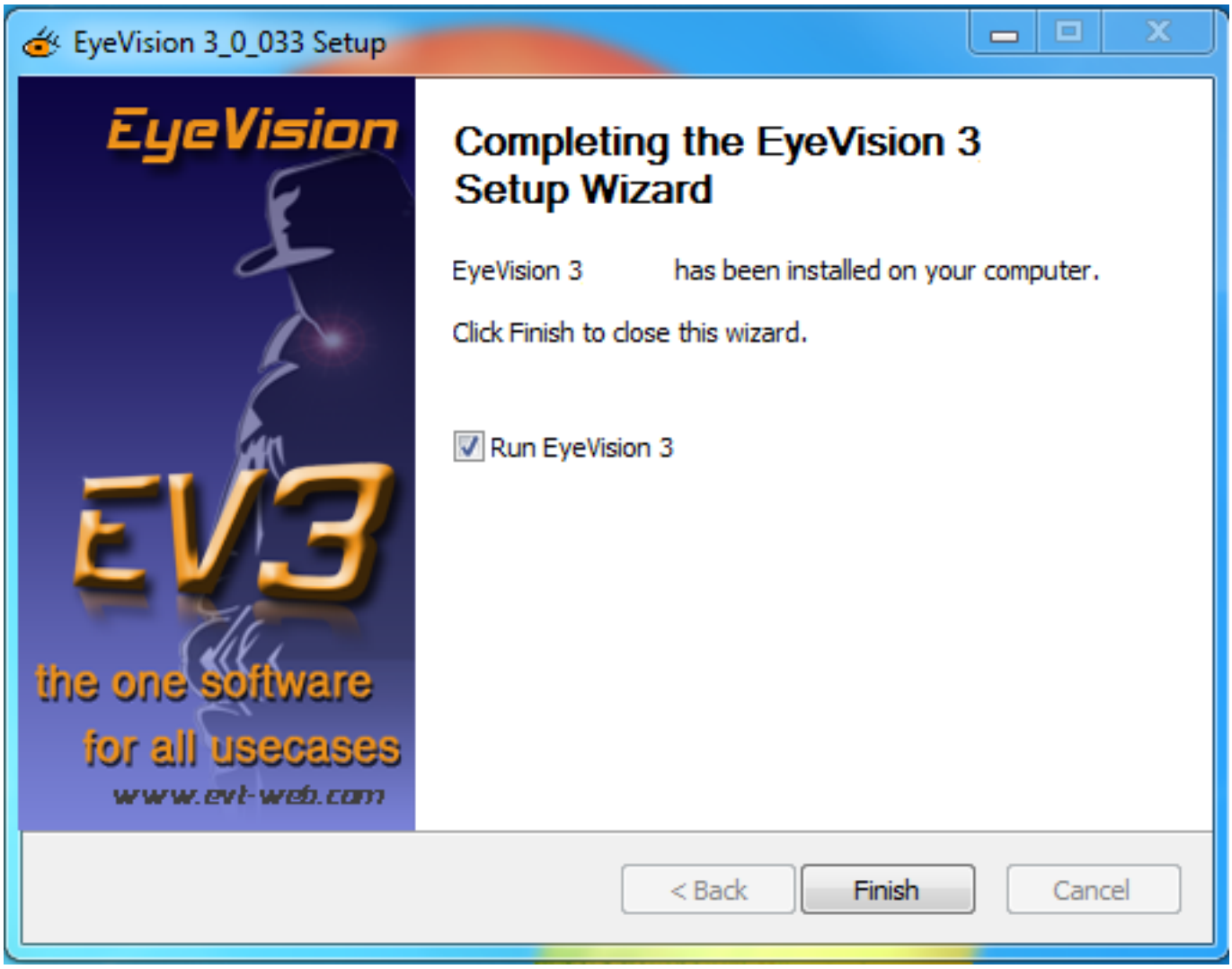
8. By clicking the "install" button the EyeVision will be installed and you can see Fig: Progress of the installation.

Figure 1.8. Fig: Progress

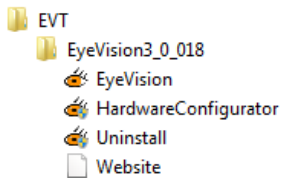


9. When the setup is Fig: Finished you will get the possibility to start EyeVision immediately if you don't uncheck the option and push "Finish".

Figure 1.9. Fig: Finished



The EyeVision start menu in Windows XP, Vista and 7 will looks like

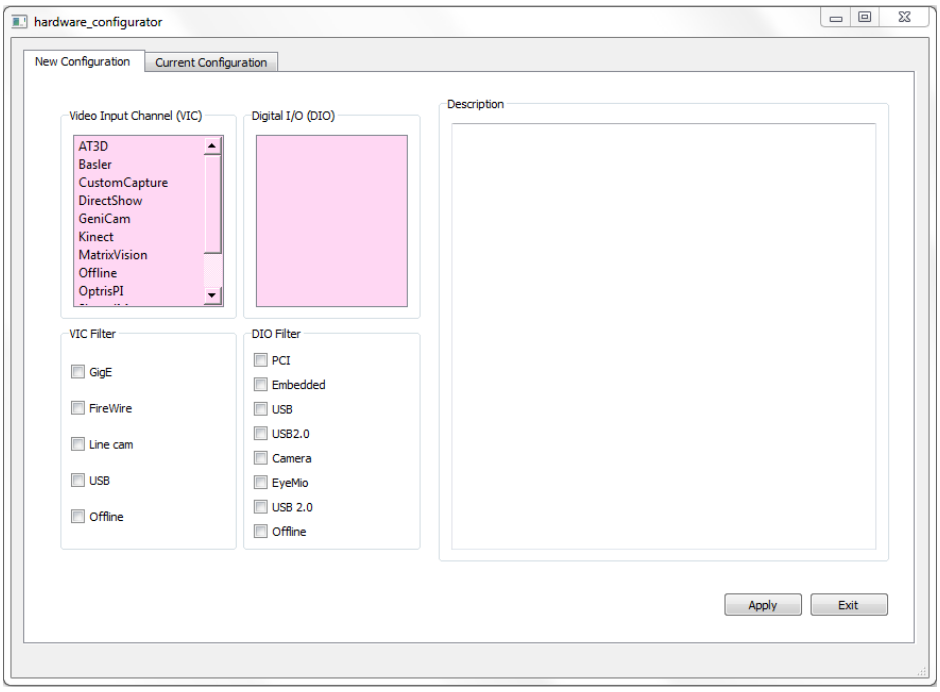


Eyevision	A link for starting Eyevision3
Hardware configurator	This link open the hardware configurator (Refer to the section called “Hardware configurator”)
Website	By clicking on this item, you are redirected to homepage http://www.evt-web.com

Uninstall	Remove/uninstall Eyevision3 from computer.

Hardware configurator

Cause of EyeVision supports many different hardware you need the configure your video input channel implementation for EyeVision once you change the hardware. The EyeVision hardware configurator allows you to easily choose the right implementation for your Input/Output device(s). This also can be changed in EyeVision but you need a restart of EyeVision. So it is recommended to use the hardware configurator instead. Starting the hardware configurator you can choose your hardware from a list. Every supported hardware can be found here. If you don't know exactly your hardware, click the possible hardware in the list and you will see some information at the right side. You will see two lists, "Video Input Channel (VIC)" and "Digital I/O (DIO)". Below the both lists you can find some filters for the two lists.



Video Input Channel (VIC)

The video Input channel is your image source for your EyeVision. In the list you can find all supported camera types. First you have to choose your camera type. Choose the "Offline" mode if you only want to use images from your hard disk.

- **Table 1.2. List of Hardware**

Name	Information
Active Silicon	
Adlink	
AT Automation Technology	
AVT Allied Vision Technology	
Basler	It means a series of cameras produced by Basler [a]

Name	Information
BitFlow	
Custom Capture	
DirectShow	USB Cameras with a DirectShow support/filter are summarized under this name. DirectShow API [b] produced by Microsoft [c] is used to perform capture and another operations on media files or stream.
EVT	
Fire Wire UCD	
Fire Wire UniBrain	
IDS	Cameras of the group IDS Imaging Development Systems GmbH [d]
ISG Imaging Solution Group	
Matrix Vision	
Offline	It is a default input channel. A device could be simulated on your PC (e.g `EVT\EyeVision\Devices\PC_Local\Images`)
PointGrey	
PrimeSens	

[a] See the Basler homepage [http://www.baslerweb.com/?language=en]

[b] Application Programming Interface

[c] See the Microsoft homepage [http://msdn.microsoft.com/de-de/library/windows/desktop/dd375454(v=vs.85).aspx]

[d] See the IDS homepage [http://en.ids-imaging.com/]

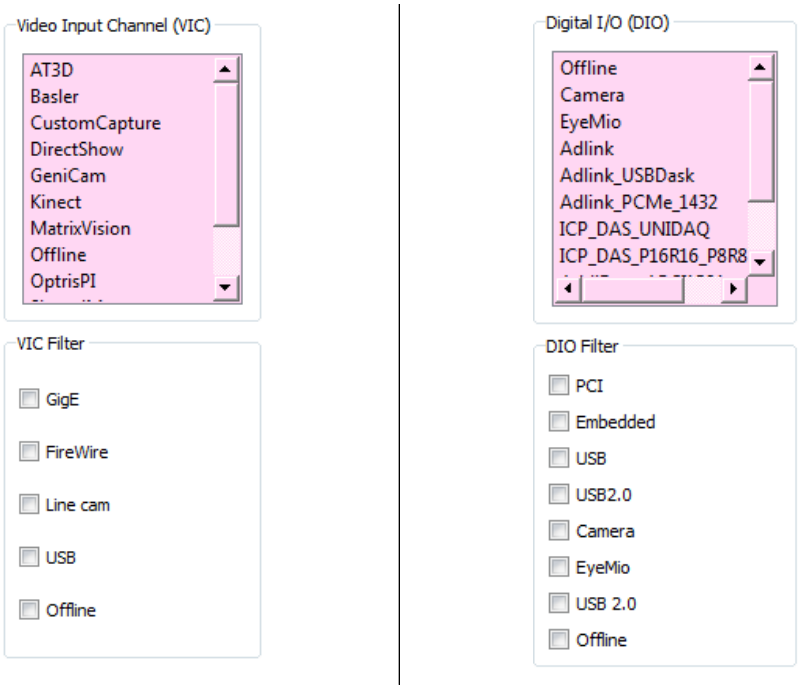
Digital I/O (DIO)

The digital I/O used by EV3 to use signals. Signals are for example leds, triggers and so on.

VIC Filters

With the VIC filters you can filter the video input channel list by the connections type of the video input. So you can select your connection type from your VIC to easily find your VIC.

Table 1.3. Filter direction



- **Table 1.4. Filter List for VIC**

Name	Information
3D Area	
Analog	
AreaScan	
Camera Link	
CoaXPress	
GigE	GigE [a]Vision is an interface standard introduced in 2006 for high-performance industrial cameras
FireWire	
LineScan	
Offline	
Remote	
Scanner	
SmartCamera	
USB	Camera with a USB-Interface.

[a]GigE: Giga Ethernet

DIO Filters

The DIO filters allows you to filter the digital I/O list. You can filter the list by the connection type of the dio. Choose your connection type to easily find your DIO.

Table 1.5. Overview Hardware

Computer Bus	Digital I/O		Video Input Device
	Name	**Typ**	**Name**
Camera	Camera	--/--	• Basler
EyeMio	EyeMio	--/--	• Basler • DirectShow • IDS • Offline
USB/ USB2.0	AdLink [a]	USB Dask(???PCI DAC???)	
Embedded	AdLink	PCMe-1432	
PCI	AdLink		
	ICP DAS [b]	(PCI-)P16R16/ P8R8	
		UNIDAC(Supported I/O Boards)	
	ADDI DATA [c]	APCI-1516	
		APCI-1564	
		APCI-1500	
		CPCI-1500	

[a] http://www.adlinktech.com/index.php

[b] http://www.icpdas.com/index.php

[c] http://addi-data.com/

Frequently asked questions (FAQ)

Some frequently asked questions to help.

Which EyeVision version is the right one for me?

If you want to use a stable version, you should use the version 2.6.051.

If you want to try out the new features and take part in the development, we recommend you to use the latest version 3.0.034

If you already use an older software version and do not want to change this version, then you indicate the desired version number when buying an EyeVision image processing system.

I can not set up a connection to the smart camera.

With the following points offer a solution in most cases:

- Is the camera in the same network as the PC?
- Windows network set up (adapter settings)
- tools you might check: ipconfig, ping
- Is it the correct network cable you use (crossover, uncrossed)?
- Is the firmware installed on the camera?
- tools: ftp (total commander), telnet

- Is a Firewall blocking the network communication?

For more details have a look at Quickstart EyeSpector.

If the firmware version of an EyeSpector camera is unknown, it can be readout in the following way

Install the latest EyeVision-Setup provided by a download link or download from the homepage www.evt-web.com. Afterwards start EyeVision\bin\EyeControl.exe all EyeSpector cameras in the network are shown in a list by Scan Network. Select the desired camera version and operating system are shown.

When starting the software an error message "Error in CUR_INIT" is shown

Regional and Language Settings on Asian PCs:

EyeVision is not running on systems that have unicode settings for programs that do not support UniCode. Please be sure to change the language settings to English or German before the installation. If EyeVision is already installed on the system, please uninstall EyeVision and reinstall after having done the correct language settings.

Control panel -> Regional and Language Settings -> Advanced

Systemsteuerung -> Regions - und Sprachoptionen -> Erweitert

An error message that a dll was not found occurs when starting the software

An error “CKData.dll not found” normally indicates a missing dependent dll. This includes missing driver dlls or an incomplete operating system installation. Please proceed as follows:

- download the Dependency walker (Dependency Walker Homepage [http://www.dependencywalker.com/], Dependency Walker direct download [http://www.dependencywalker.com/depends22_x86.zip])
- extract depends.exe from the zip file
- start depends.exe.

  ### Note

  depends.exe should be started with Administrator rights (rightclick->As administrator)

- File->Open->EyeVision.exe
- Profile->Start Profiling->OK (changes in the menu “Profile Module” are not necessary)
- execute EyeVision and close all error messaged until the EyeVision software is closed again as well.
- The missing dll is listed in the dependencies tree on the left side.
- If you have any questions, save the dependencies tree. (File->Save As)
- Please send the Dependency Walker File to the EVT Support (support@evt-web.com), for faster revision you might call us on phone.

EyeVision and Windows embedded

The following steps are necessary to run EyeVision on Windows embedded: Windows embedded might miss important features required by EyeVision, the following items were necessary for successfully run EyeVision on Windows embedded:

- manual creation of Windows/System before installation
  - Errors during installation if not existing:

    Installing fonts and wibukey failing
- manual installation of opengl
  - opengl32.dll
  - gdi32.dll
  - glu32.dll
- manual installation of windows/System32/xcopy.exe

  Errors in Filesync if not existing

Uninstall EyeVision

Here are some operating systems to show how to uninstall EyeVision

Windows 98

- Go to the start menu of windows (by pressing the Windows key) and select "Control panel".
- Press the program icon
- A new dialog open up with your installed programs. Choose EyeVision and press the uninstall button.

Windows Vista

- Go to the start menu of windows (by pressing the Windows key) and select "Control panel".
- Press the program Icon
- A new dialog open up with your installed programs. Choose EyeVision and press the uninstall button.

Windows 7

- Go to the start menu of windows (by pressing the Windows key) and select "Control panel".
- Push uninstall program the program category
- A new dialog open up with your installed programs. Choose EyeVision and press the uninstall button.

Windows 8

- Go to the start menu of windows (by pressing the Windows key + C) and select "Control panel".

- Push uninstall program the program category
- A new dialog open up with your installed programs. Choose EyeVision and press the uninstall button.

Chapter 2. Graphical User interface

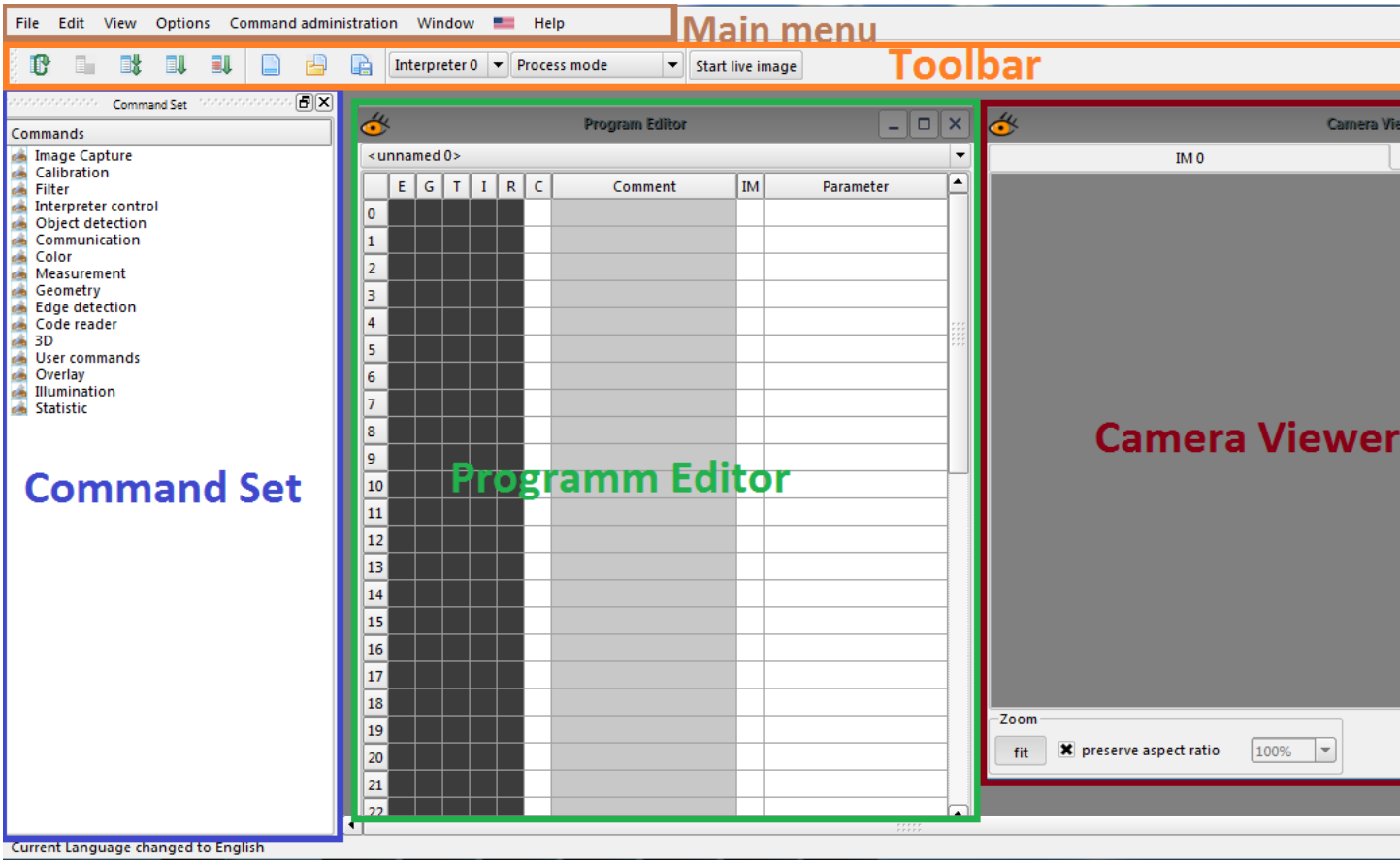
The graphical user interface is the interface between the user and EyeVision.

Main Graphical User Interface

After starting EyeVision you will get the EyeVision GUI. The interface contains the following elements:

- The Main menu
- Program editor
- Camera viewer
- Toolbar
- Command Set

Figure 2.1. EyeVision GUI



The Main menu

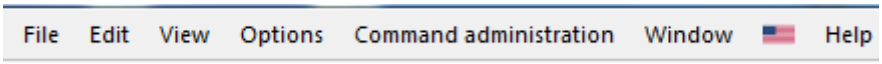
In the main menu is a graphical control element which contains drop down menus. The menu bar's purpose is to supply a common housing for window- or application-specific menus which provide access to such functions as opening files,

interacting with an application, or displaying help documentation or manuals. The main menu can be found in the Top of EyeVision.

Note

Menu items will be greyed out if the corresponding feature isn't available. For example, you cannot load a new program while the current program is running.

Figure 2.2. The main menu



The main menu contains the following items:

File — This sub menu contains all functions working with the inspection program, like save and load. Also you can load and save an image from/into the image memory. An additional option to close EyeVision exist and the recently opened files are been listed. Please refer to the section called “The "File" submenu” for more detailed instructions.

Edit — This menu contains functions for the selected line in the program editor. You can copy, cut, paste or delete a line. Please refer to the section called “The "Edit" submenu” for more detailed instructions.

View — Allows you to customize your EyeVision. You can hide and show elements of the GUI and get special elements for the image analysis. Please refer to the section called “The "View" submenu” for more detailed instructions.

Options — This menu contains items to launch the device control (not yet implemented), to edit/save a layout for the user interface(e.g how much and which window are been displayed), to set the camera options or the application parameters. Please refer to the section called “The "Options" submenu” for more detailed instructions.

Command administration — This menu opens the command set configuration and provides details about the license. Please refer to the section called “The "Command administration" submenu” for more detailed instructions.

Window — This menu contains items to arrange the windows, hide the menu bar or modify the aspect of design (color, shape...etc) . Please refer to the section called “The "Window" submenu” for more detailed instructions.

Flag — The selection of the GUI language could be done through a click on this icon. You get a drop down menu with different flags with the language name that refers to the used language. Please refer to the section called “The "Language" submenu” for more detailed instructions.

Help — This menu contains items to help the user, e.g. access to the manual, launch a the context help, usual information about EVT, and little overview for news on the version. Please refer to the section called “The "Help" submenu” for more detailed instructions.

Each of these menu items is described in more detail in the following sections.

The "File" submenu

The EyeVision file sub menu looks like Figure 2.3, “The "File" submenu”.

Figure 2.3. The "File" submenu

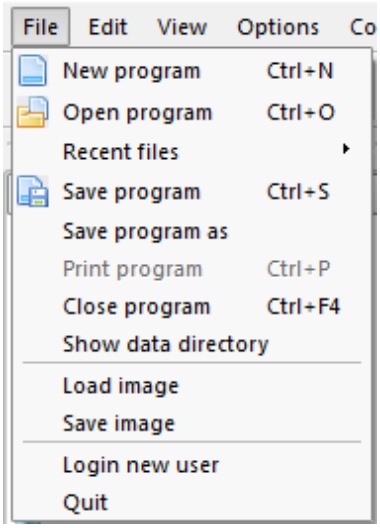


Table 2.1. File sub menu items

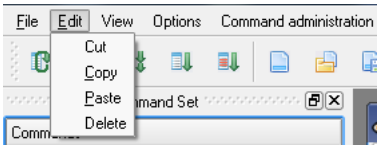
Menu Item	Short-cut	Description
New program	**Ctrl+N**	Creates a new empty inspection program.
Open program	**Ctrl+O**	Open a dialog box that allows the user to browse the computer for an inspection program file (*.ckp) and load it in the program editor.
Recent Files		A list off all recently opened inspection programs.
Save Program	**Ctrl+S**	Save the current inspection program to the given path. If the current inspection program has not been saved yet, it brings up the file save dialog box that allows the user to define a *.ckp file path and name the file.
Save Program as		This brings up the file save dialog box that allows the user to define a *.ckp file path and name the file.
Print program		This menu entry allows to print the program.
Close program	**Ctrl+W**	The current inspection program will be closed. If there are changes in the inspection program, the user may save or drop them.
Load image		Start the file open dialog box that allows the user to load an image from your disk into all shown camera viewer. The destination image(IM) memory must be chosen in a new dialog window.
Save image		This menu item will save the image from the camera viewer that currently has the focus. The user can select whether to save an image with overlay or without. If you select to save with overlay, additional information will be added to the image. Gray scale images will be converted to RGB24 images. When images are saved with intention to be processed again, be sure to not save the overlay.

Menu Item	Short-cut	Description
Login		Here you can switch between the users of EyeVision
Quit	**Ctrl+Q**	This menu item allows you to quit from EyeVision. EyeVision will ask to save your open inspection programs if you haven't previously saved them.

The "Edit" submenu

The EyeVision Edit sub menu looks like Figure 2.4, “The "Edit" submenu”

Figure 2.4. The "Edit" submenu



.

Table 2.2. Edit menu items

Menu Item	Short-cut	Description
Copy	**Shift+Ctrl+C**	Copy the selected line(s) in the program editor to the internal command clipboard.
Cut	**Ctrl+X**	Cut the selected line(s) in the program editor and copy them into the internal command clipboard.
Paste	**Shift+Ctrl+V**	Paste command line(s) from the internal command clipboard into the column of the program editor that has the focus. If there are multiple lines, new empty lines will be added.
Delete	**Del**	Removes the selected line(s) from the program editor.

The "View" submenu

The EyeVision View sub menu looks like Figure 2.5, “The "View" submenu”.

Figure 2.5. The "View" submenu

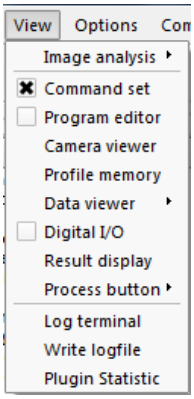


Table 2.3. View menu items

Menu Item	Description
Image analysis	Open up a sub menu with several image analysis tools. Please refer to the section called “The "Image analysis" submenu”
Command set	Show/Hide the command set tree view.
Program editor	Show/Hide the program editor view.
Camera viewer	Add a new camera viewer with image memory 0 (IM 0).
Data viewer	Opens the data viewer sub menu. The five registers can be shown. Please refer to the section called “The "Data viewer" submenu ”.
Digital I/O	A new dialog open up. Eight Inputs and eight Outputs. If there are connected DIOs they will be colored, else greyed out. You can also use a simulation mode to test the DIO.
Result display	Create a new result display for a custom process mode GUI. Please refers to Result display
Log terminal	A new dialog is shown up. The log terminal allows you to see every EyeVision message. You can filter them by the the severity or the sending module.
Write logfile	Write a log file with all EyeVision output (like warnings, errors and debugging information). The path of the log file is returned in a message box.
Process button	Shows a sub menu with the all buttons from the toolbar. Every button will be displayed in a new dialog. Integrators can select some buttons to generate a custom process GUI.

The "Image analysis" submenu

Figure 2.6. The "Image analysis" submenu

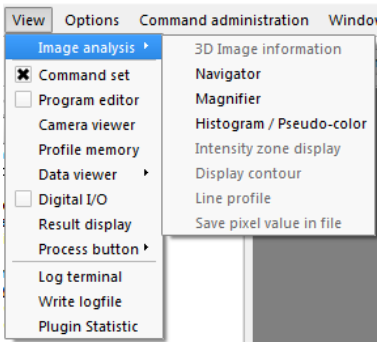


Table 2.4. "Image analysis" items

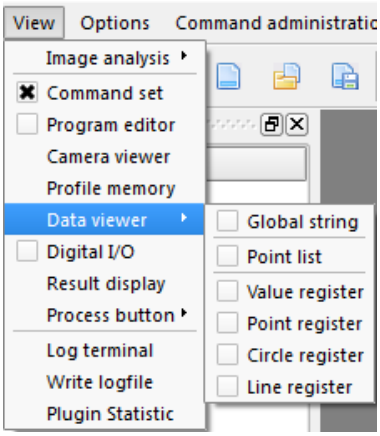
Menu Item	Description
3D image information	Gives some information about the point cloud. Also you can get some mouse over features to inspect the point cloud.
Magnifier	When the magnifier function is activated, a dialog is opened that zooms the area of the mouse cursor.
Navigator	This item opens a dialog. For this operation, the option "fit" in Camera Viewer must be deactivated, to permit the change change of the zoom slider. A shift

Menu Item	Description
	of the zoom slider define a rectangle form on the navigator. So the user can scale up/ down the rectangle size and shows a zoom of the selected rectangular zone in the camera viewer.
Histogram/ Pseudo-color	An area can be selected in the image in the Camera Viewer. The histogram displays the distribution of the brightness values in the selected area.
Intensity zone display	You can display different intensity zones in different colors.
Display contour	This option allows you to adjust a brightness value via a slide control. The contours of the object will be displayed (in pink) in the camera viewer with the adjusted threshold.
Line profile	A line can be positioned in the image in the camera viewer. The brightness values of the image along this are displayed graphically.
Save pixel in file	Selected pixel values can be saved in a file. The saved values can be used by other commands.

The "Data viewer" submenu

With the Data viewer sub menu you get a direct view of the wished register. You can choose between five different registers. For more information about the different registers, please refer to the section called “Register”.

Figure 2.7. The "Data viewer" submenu



The "Process button" submenu

The process buttons sub menu contains all buttons from the tool box. By clicking them a new dialog with the corresponding button will show up. You can control your inspection program with this tools. This option allows you to customize EyeVision.

Figure 2.8. The "Process button" submenu

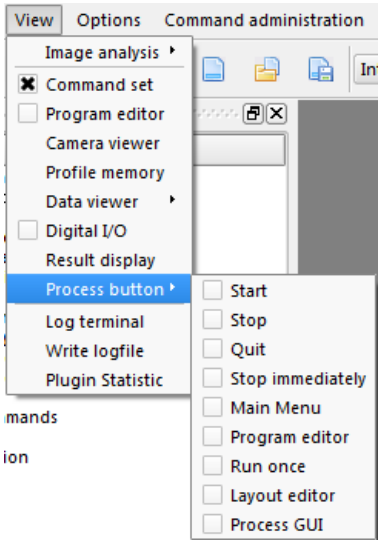
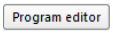
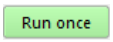


Table 2.5. Process button menu items

Menu Item	Icons	Description
Start		Shows up a dialog with a Start button which can start the inspection program.
Stop		Shows up a dialog with a Stop button which can stop the inspection program after the running command.
Quit		Shows up a dialog with a Quit button which can close EyeVision.
Stop immediately		Shows up a dialog with a Stop immediately button which can stop the inspection program after the running instruction.
Main menu		Shows up a dialog where you can show/hide the main menu
Program editor		Shows up a dialog where you can show/hide the program editor
Run once		Shows up a dialog with a Run once button which can run the inspection program once.
Layout editor		Shows up a dialog where you can show/hide the layout editor
Process GUI		This opens a new process GUI dialog. A process GUI dialog can be defined from the integrator to combine several controls with custom functionality. This includes buttons, sliders and spin boxes that are equipped with a script interpreter tool.

The "Options" submenu

The EyeVision Options sub menu looks like Figure 2.9, “The "Options" submenu”.

Figure 2.9. The "Options" submenu

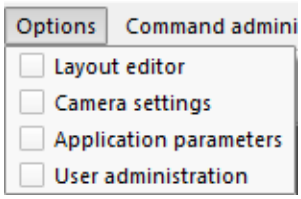


Table 2.6. Options menu items

Menu Item	Short-cut	Description
Device configuration		Opens the device configuration dialog. See Chapter 3, *Device control and configuration*
Layout editor		Opens the layout editor. Use this layout editor to design custom layouts to support you during integration and for process mode.
Camera settings		This options allows you to adjust the camera parameters of the active camera. Please refer to the section called “Camera settings”.
Application parameters		Specific application parameters for the EyeVision software. Have a look at Chapter 5, *Application parameter*.
User administration		Opens the user administration dialog. See the section called “User Administration”

The "Command administration" submenu

The EyeVision Command administration sub menu looks like Figure 2.10, “The "Command administration" submenu”.

Figure 2.10. The "Command administration" submenu

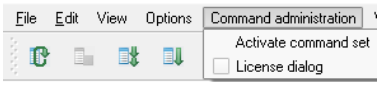


Table 2.7. Command administration menu items

Menu Item	Short-cut	Description
Activate command set		You can activate different command sets when you get the corresponding license
License dialog		The license dialog shown up. Here you can activates the license code or ask for a new one.

The "Window" submenu

The EyeVision Window sub menu looks like Figure 2.11, “The "Window" submenu”.

Figure 2.11. The "Window" submenu

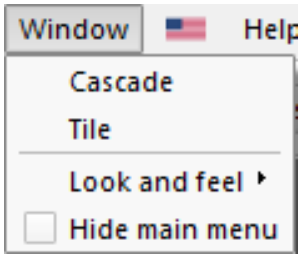


Table 2.8. Window menu items

Menu Item	Short-cut	Description
Cascade		Window arrangement: cascading windows.
Tile		Window arrangement: tiled windows.
Look and feel		This menu allows to select one of the predefined styles for your EyeVision.
Hide menu main		This item menu hide the menu tool bar.

The "Language" submenu

The EyeVision Language sub menu looks like Figure 2.12, “The language submenu”. A drop down list with the flags and names of the supported languages is shown. To select a new language just click it.

Figure 2.12. The language submenu

The "Help" submenu

The EyeVision Help menu looks like Figure 2.13, “The "Help" submenu”. You will get information about EyeVision, see this manual or create a support file

Figure 2.13. The "Help" submenu

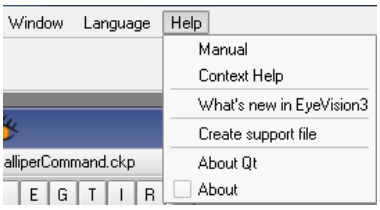


Table 2.9. Help menu items

Menu Item	Short-cut	Description
Manual		Opens this manual.
Context help		Activates the context help.
What's new in EyeVision		Shows the change log.
About		Usual information about EVT, operating system or connected devices.
Create support file		It is also possible to create a support file with all necessary information of the software that is needed for a professional support The support file contains information about the current installation (such as the version, installed hardware, installed plugins, current configuration) and internal states. Kindly provide a support file if you contact the technical support. `<support@evt-web.com>`
About Qt		Information about the running Qt.

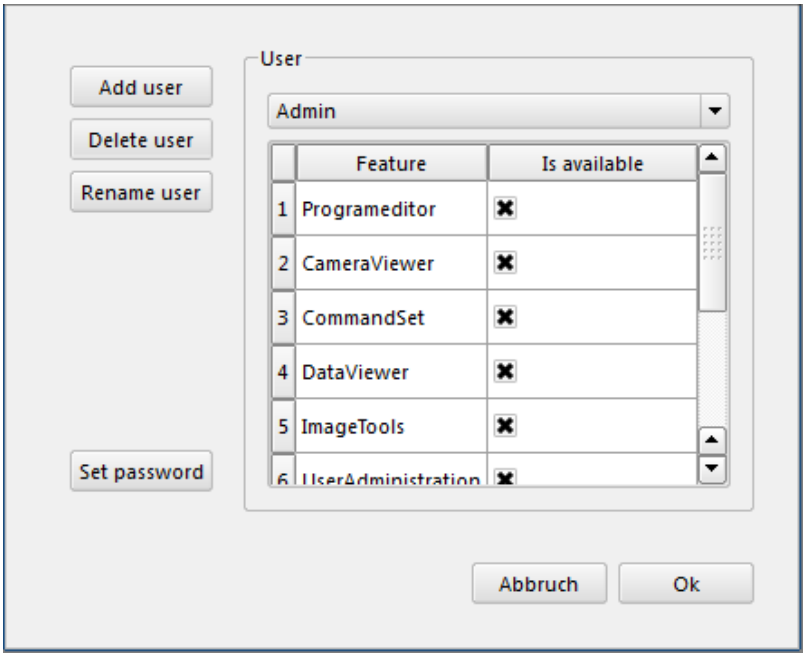
Device configuration

Here you can configure your input/output device(s). Please check Chapter 3, *Device control and configuration*

Application parameter

Here you can change many important application parameter. Please check Chapter 5, *Application parameter*

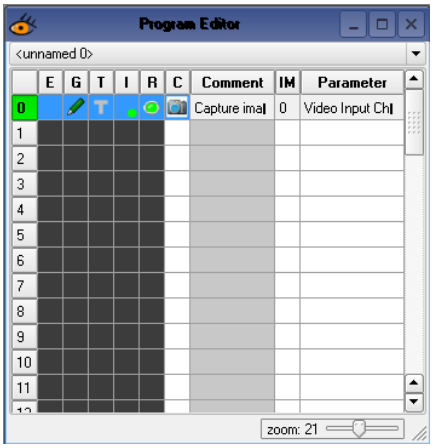
User Administration



With the user administration you can easily administrate your user for EyeVision. You can add, rename and remove user. You can change the available elements for every user by (un)checking the corresponding elements and set a password for the user.

Program editor

The program editor looks like Program editor. The EyeVision program editor can be explained as the EyeVision command list of your inspection program. The commands will be sequential processed. There can only be one command per line. It is recommended to use empty lines between commands to get a better overview in the program editor. You can add new empty lines later, if you need them. Commands can be easily put in this List by dragging them out of the command set and drop them in their new position in the program editor. A new dialog will appear to modify the command for your use. Commands can be updated by clicking with the right mouse button on it and choosing "Quick edit".



Execute a part of the program

You can control your inspection program with the Toolbar. You can also run the inspection program to a defined point by:

- Left mouse button clicking into the commands column will partly execute the program. This executes the program from the first row to the selected row. Some keyboard modifiers will change the exact behavior. Note that the program might wait until any condition has been triggered. You can always stop the program by clicking on the stop button (it is positioned between another shortcuts or in a single side window.).
- **Ctrl**+Left mouse button Executes the program without applying any jumps.

Toolbar

The toolbar fasten the use of EyeVision by direct accessing some often used functions. The toolbar is found below the Main menu. You get the following functions:

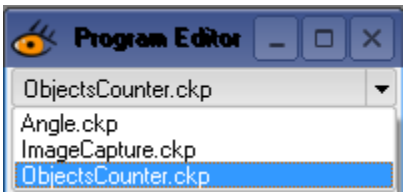
	A new, empty inspection program file is generated.
	Opens the file menu. The user can select a inspection program which is loaded in the program editor.
	The current inspection program is saved.
	The inspection program is executed completely (only one time).
	Only selected commands of the inspection program are executed.
	A single step in your inspection program. You can go step by step threw your inspection program.
	The running inspection program stops.
	The inspection program runs cyclic. You have to stop it manually.

The options of the program editor

The window of the Program editor is composed of a combo box, a slider for zooming and many columns.

`Combo box`

Shows the active inspection program. By clicking on it you get a drop down menu. You can switch between the recently used inspection programs.

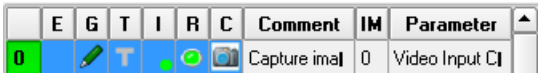


Zoom

A slider at bottom side of the window serve for zooming the interior of program editor.

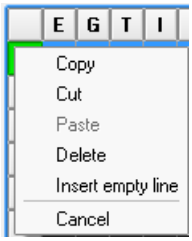
Columns

The different columns give access to general parameters of a command.



	The first column contains the column number specifies the program line of the command. See the paragraph of Column N.
	“E” means “End Execution”. You will get different possibilities when you want to execute this line. See the paragraph of Column E.
	“G” means “Graphic Mode”. You will get different possibilities when you want to show the processed image. See the paragraph of Column G.
	“T” means “Text Mode”. You will get different possibilities when you want to show the textual output. See the paragraph of Column T.
	“I” means “Invert the result of the command”. The column I (Inversion) inverts the result of the command.
	“R” means “Result of the command”. The Led of the column R displays a green LED if the result was good if not a red LED and if there is a warning a yellow LED is displayed.
	“C” means “Command”. This is the column where you drop the commands that you have dragged of the instruction set. If you drop the command into a field of the C-column the command is inserted. A small icon identifies the command.
	This column allows you to enter a comment to each command in the list. A double click in the comment column allows you to enter or change a comment. It is recommended to comment your inspection program for a better overview.
	“IM” means “Image Memory”. The number in this line shows which image memory is used for the command. The number in this column represent the used image memory which the command uses, e.g. if it says “1” in this column, it means that the command works on the image memory 1. Change the number by double clicking on it. Multiple numbers mean multiple In/Output image memories.
	This column displays the command parameters you defined during the configuration.

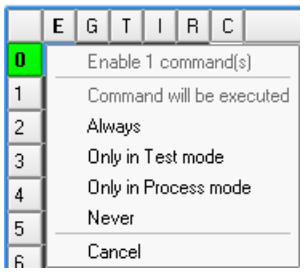
- Column N



If you click with the right mouse button in the column No. this menu appears.

`Cut`	Cut the selected line(s).
`Copy`	Copy the selected line(s)
`Paste`	Commands of the buffer are pasted at the current position.
`Delete`	The selected line(s) are deleted.
`Insert empty line`	An empty line is inserted at the selected line. All lines below (included the selected line) were moved down.
`Cancel`	The menu is closed.

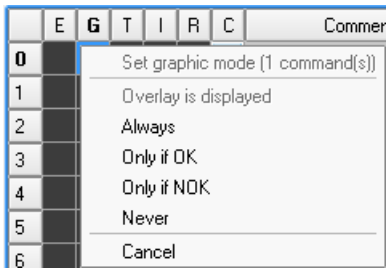
- `Column E`



A right click in a field of the column E of the current program line open up this menu. A left click switch between "Always" and "Never".

Always		The command in this line is always executed.
Only in Test mode		The command is not executed in the configuration mode. So the system does e.g. not try to capture an image of a camera that is not connected.
Only in process mode		The command is not executed during the checking mode. You can e.g. configure an image capture command that is replaced by an image capture command of the camera as soon as the system is in the checking mode.
Never		The command is never executed.
Cancel		This menu is closed.

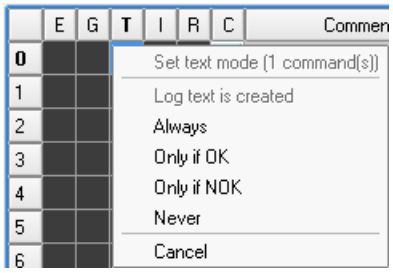
- `Column G`



A right click in a field of the column G open up this menu. Clicking with the left mouse button switch the different options (see the table below).

Always		The image is always displayed in the Camera Viewer.
Only if OK		The image is only displayed in the Camera Viewer if the result of the command ok.
Only if NOK		The image is only displayed in the Camera Viewer if the result of the command is not ok.
Never		The image is never displayed in the Camera Viewer.
Cancel		The menu is canceled.

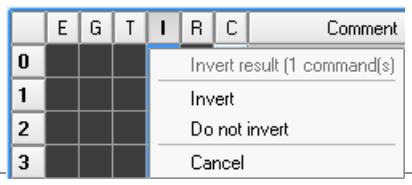
- Column T



A right click open up this dialog. If you click with the left mouse button you switch between the four different options (see the table below).

Always		There is always a text output independent of the success of the command.
Only if OK		There is only a text output in the text terminal if the command is successful.
Only if NOK		There is only a text output in the text terminal if the command is not successful.
Never		There is never a text output in the text terminal.
Cancel		The menu is canceled.

- Column I



A right click in a field of the column I open up this menu. If you click with the left mouse button you can switch between the two different options.

Invert		The result of the command is inverted.
Do not invert		The result of the command is not inverted.
Cancel		The menu is canceled.

- `Column R`

	If the command run successfully.
	If an error occurs in program or the results of evaluation are erroneous.

- `Column C`

A right click in a field (with a command) of the column C open up this menu. If you click with the left mouse button in the field there are two possibilities:

1. If you click on a white (empty) field all commands till this position are executed.
2. If there is a command in this field the "Quick edit" window opens.

Quick edit	The configuration window of the command opens up.
Use last result	A window opens with the set values and the tolerance values of the command. The last measured value is used as set value.
Cancel	The menu is canceled.

Camera viewer

This section introduce the EyeVision camera viewer. The camera viewer displays the images EyeVision is working on. Any command may draw its result on his graphical overlay. It allows to focus on regions of the image, to scale, or to transfer live the capture of the image.

The camera viewer can show the images from your video input channel as well as the processing steps of your inspection program.

Also it is used for some settings of the camera.

The storage types of the image data

Storage types in general

There a two categories for stored images. EyeVision supports 2D and 3D images.

2D Images

2D Images were divided in 3 types. 8 bit, 16 bit and 24 bit depth images. The representation of a grey scale image corresponds to the bmp-format of windows. Using bmp-format means basically: the image is defined by its width (number of pixels per line), its height (number of lines per frame) and the size of a pixel. The bytes for the pixels are stored sequentially one after the other, line by line.

8 bit

8 bit depth images are grey scale images with an intensity from 0 (black) to 255 (white). There are 3 different file formats which are supported from EyeVision (.bmp, .jpg and .png (not available on EyeSpector Cams)).

16 bit

16 bit depth images have an intensity of 0 (black) to 65535 (white). This depth is not avaible on smart cameras. Can be used for depth information of sensors. Right now there a less algorithm based on 16bit grey scale images. This type can be used as a temporary type. The only supported file format is .png.

24 bit

24 bit depth images are color pictures. This are RGB images with 3 different channels (red, green, blue) with an intensity of 0 (non of this color) to 255 (full color). Some examples are 0,0,255 means blue, 255,0,255 means violet. The three supported file formats are .bmp, .jpg and .png (.jpg/.png are not supported on EyeSpector Cams).

3D Images

Point clouds are an independent container for 3D data. Points are saved into a list. The three supported file formats in EyeVision are .e3d, .csv and .txt. The values are stored as float (single precision (32bit))

EyeVision 3D file format (.e3d)

e3d is a binary file format to store points.

Comma separated values (.csv)

In this file there is one point per line and no header needed. The values can be a floating decimal or integer. Decimal floating point is point ("."). Comma separates the values. Space will be ignored. The file could look like:

```
0, 0.5, 10.0
0, 1, 12.0
0, 1.5, 14.0
```

Text file (.txt)

In this file there is one point per line and no header needed. The values can be a floating decimal or integer. Decimal floating point is point ("."). Space separates the values. The file could look like:

```
0 0.5 10.0
```

```
0 1 12.0
0 1.5 14.0
```

Coordinate system

The coordinate system is by default right handed. The visualization is set by default, the observer will look in the direction of the positive z-axis. The point cloud will be positioned, so the center of the point cloud can be found in the middle of the image. The entire cloud will fit in the camera viewer.

Table 2.10. Axis in the point cloud

X-axis	Coordinates increasing from left to right
Y-axis	Coordinates increasing from top to bottom
Z-axis	Coordinates increasing into the depth of the image

Cursor shapes (Camera viewer)

The shape of the cursor changes, if you move the cursor across the different elements or sub-domains of the configuration elements. Depending on the displayed shape, it is then possible to shift the whole element or to change its shape.

If the mouse pointer is shaped like this, you can move the whole object arbitrary in the image without changing the size or the rotational direction.

If the mouse pointer is shaped like this, the object can be resized.

This shape of the mouse pointer means that you can rotate the hole object.

Mouse interaction in the camera viewer

The camera viewer allows mouse interaction to zoom and focus into images. Depending on the image type (2D or 3D), there are different functionalities available.

2D Images support the following mouse interactions:

Box zoom **Shift**+Left mousebuttonZooming into a specified box.

3D point clouds support the following mouse interactions:

Manual navigation using the mouse for 3D point clouds is possible, if option "autoCenter" is deactivated.

The following list shows the possible interactions:

`Rotation` Left mousebutton

Rotation of the point cloud around focal point.

`Zoom in / zoom out` Mouse wheel

Zooming into the point cloud

`Move focal point` — **Shift**+Mouse wheel

Moving the focal point responds directly when orthogonal projection is switched off. Moving the focal point effects the rotation.

`Translate object vertical and horicontal` — **Shift**+Left mousebutton

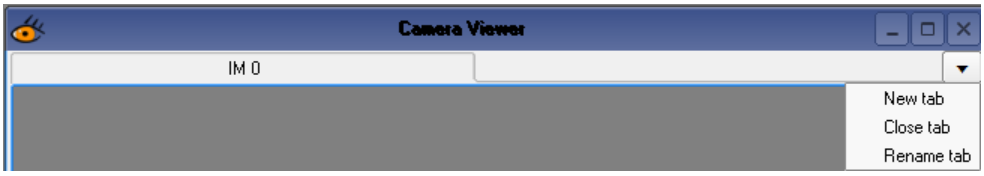
Translates the pointcloud in x and y direction.

The elements of the Camera Viewer

The components of the camera viewer will be described in this section. The goal of this section is to give a quick overview of all various options that could be seen on the user interface.

Image memory

Images must be stored in an image memory to work and show them to the user. You get also the possibility to show the actual selected video input channel taken images as live images in the Image memory.



New Tab	Opens a new tab in the camera viewer. The name of the tab will be "IM " + Counter for the IMs.
Close Tab	A click allows the user to close the currently selected IM tab.
Rename Tab	A new dialog open up where you can enter a new name for the tab.

IM 0, IM 1, ..., IM n — It is possible to display up to n different image memories. At the beginning you start with one Image memory but you can easily add new Image memories.

Zoom

Zoom

You can choose the zooming factor by the drop down menu or the slider. The slider allows you smaller steps as the drop down menu. If you want to choose an exact zoom you can enter the number in the drop down menu.

Zoom fit

You get two options. You can select if the aspect ratio is checked or not. If you check it, the a aspect ratio of 1:1 is used and your image will not be stretched. So the camera viewer could be not completely filled with the image (see figure CheckRatio). When you uncheck this option the image is fit in the camera viewer. No matter of the aspect ratio. So the image can be stretched (see figure UncheckRatio).

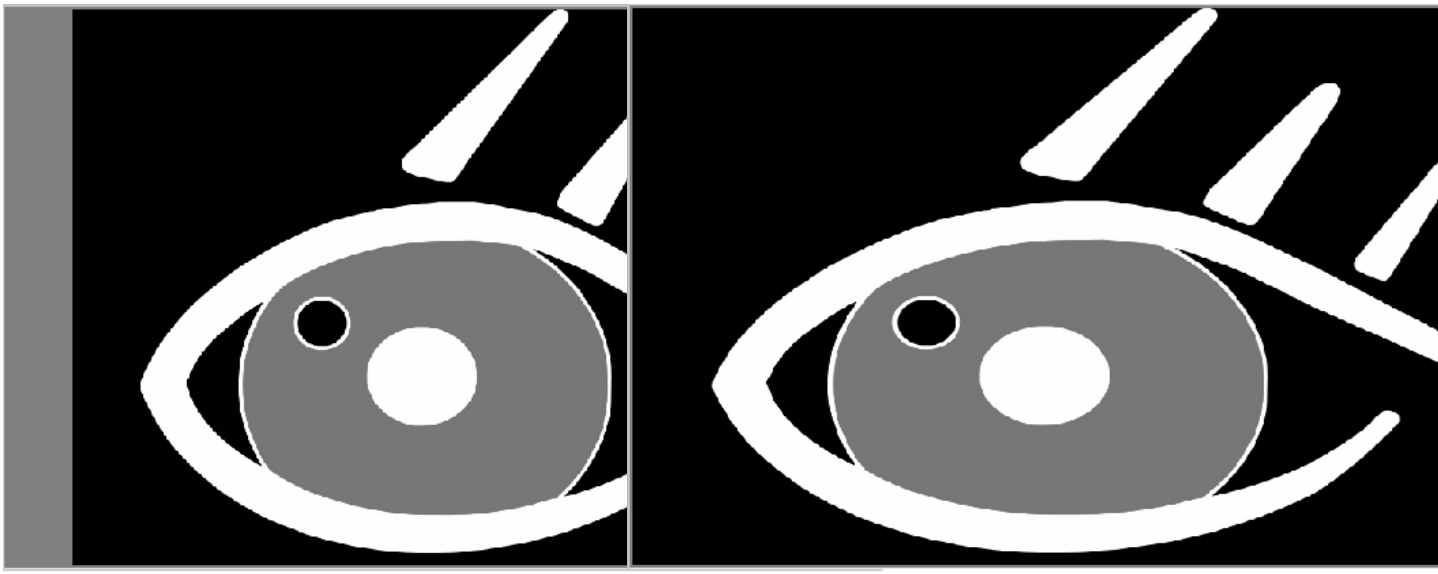


figure CheckRatio: Fit with selected checked aspect ratio

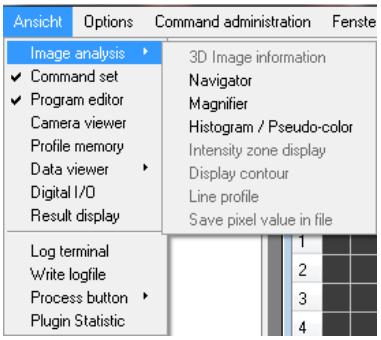
figure UncheckRatio: Fit without checked aspect ratio

Image information

There are two boxes with different information (Image and Value).

"Image" show the cursor position in the camera viewer. If your image is not zoomed and stretched (100%) you get the position in pixel. Starting top left with x=0 and y=0. If your image is stretched the cursor position can be floating point number. "Value" gives the color value of the cursor position. This can differ by the image types.

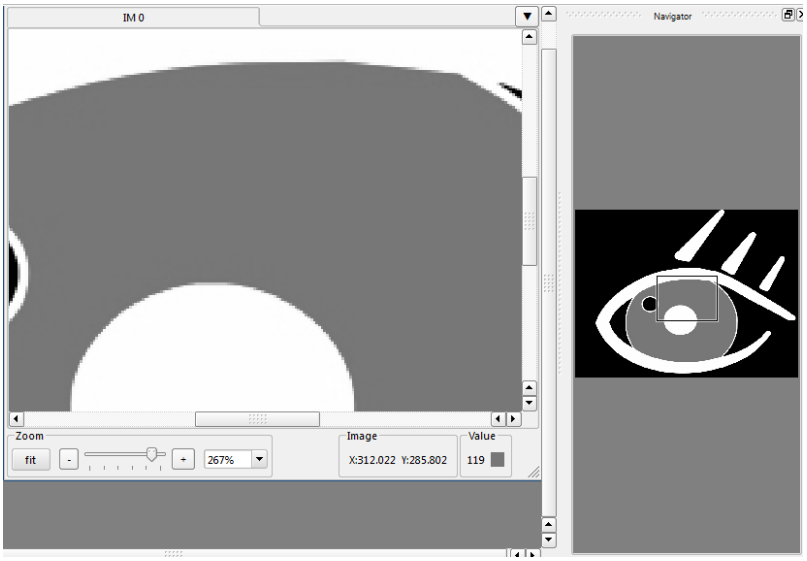
Image analysis



The menu "image analysis" allows you to inspect the image. There are also specialized display functions that simplify a parametrization of the different libraries and the graphical programming that is based on the libraries of the EVT Software.

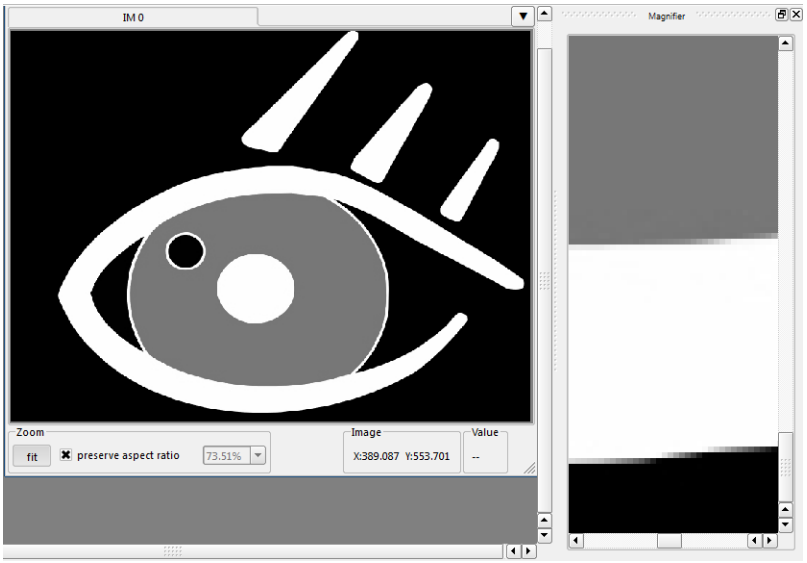
Navigator

The navigator support a better navigation threw a zoomed image. It open up a new dialog where the whole image is shown. You get a rectangle in the navigator which symbol the field of view for the camera viewer. So you can see, where your camera viewer stands in the image. You can easily navigate threw the image by clicking or holding the left mouse button in the navigator. The mouse pointer is then the middle point of the field of view.



Magnifier

The magnifier open up a new dialog. In this new dialog the surroundings of the cursor in the camera viewer are zoomed by 800%.

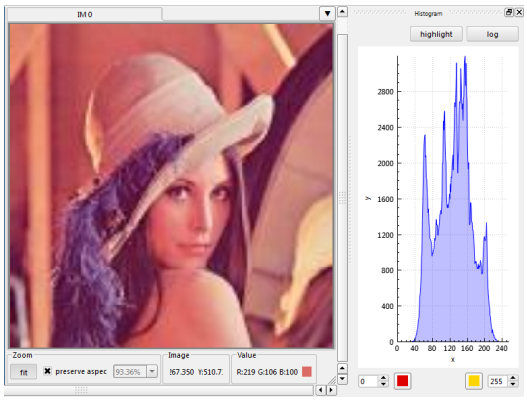


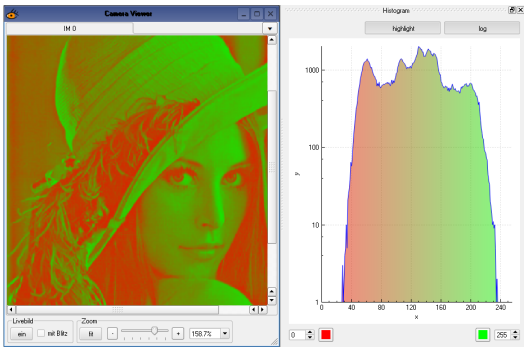
Histogram/Intensity zone display

Histogram — The relative frequency of the gray values of the pixel are displayed in a histogram (Figure 2.14, “Histogram”). This tool is very helpful for the adjustment of a camera (aperture). The gray values (0-255) can be read on the x-axis and the relative frequency can be read on the y-axis.

Intensity zone display — A gray image can be displayed by the two dedicated colors of the intensity zone display. This tool is very helpful in order to get a general idea of the lightning conditions of the image. If both colors are adjusted as the same color the display corresponds to a gray image. For example see Figure 2.15, “Intensity zone display”.

Figure 2.14. Histogram **Figure 2.15. Intensity zone display**

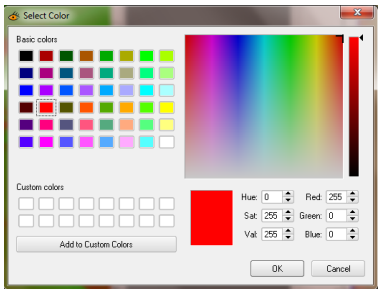




- **Determine thresholds**

Determine thresholds

The lower and the upper threshold can be entered directly into the accordant box or they can be determined via the sliders (click and hold left mouse button and move the sliders). The values refer to the current image in the Camera Viewer. The coloration is directly visible in the Camera Viewer.



Change colors

A mouse click on the button color 1 or color 2 opens a Windows form for the color selection. You can choose a color for the upper and the lower threshold by simply clicking on your preferred color.

- **Highlight**

By clicking on this button the change of gray value as "Intensity zone display" will be displayed in the Camera viewer. For example Figure 2.15, “Intensity zone display” is the result of Figure 2.14, “Histogram” after activating this button.

- **Log**

Figure 2.15, “Intensity zone display”shows the histogram after applying this function of "Log" button.

Camera settings

Figure 2.16. Example of camera settings

The interface holds the option of a camera control for various cameras, while for FireWire cameras there are even further adjustments such as brightness, saturation, zoom, trigger mode etc.

The view can change by different camera types. This is cause of the different available settings of the cameras. An example of output for a camera is shown in Figure 2.16, “Example of camera settings”

GUI Designer/ Layout

This section make references to all way for defining different layouts or styling the windows in the graphical user interface.

The layout make it possible for user to create an individual user interface for the runtime system, that means for example:

- to display the live-images from the camera with its overlays
- to display the image of the machine or the component parts and their technical drawings in a background image

Therefore the user can keep an overview of the inspection and the inspection characteristics. With the optionally available statistic package it is also possible to display the production values clearly arranged in graphs on the user interface.

Note

The statistic package is for layout editor currently not available.

To create a user-defined interface, the user need to open the layout under (Options → Layout editor). Many layout forms could be generated and locally saved, because the design can be different according the inspection program. If the user only wants an overview of the live-images from the camera, individual camera viewer can be added in (Options → Camera viewer). But for a user-defined display, which can supplied with a logo or various characteristics, the Layout editor is the right choice.

Customize Graphic User Interface (GUI)

The user can personalize the appearance of the GUI. In order to simplify the own view of software, minimize the count of visible elements or enhance some element the user has the possibility to style his window himself.

- *Arrange windows*

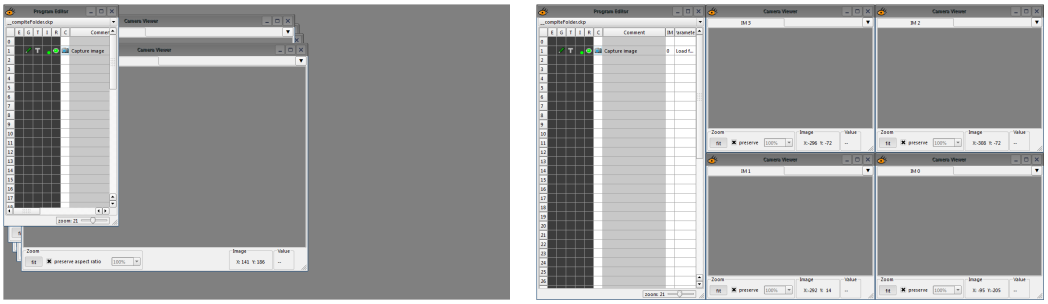
  Figure 2.17, “Cascaded and tiled window” show a screen shot of the different style, that are been proposed.

  Tile — By selecting this option(Window#Title) the "Program editor", "Camera viewer", "Profile memory", "Result display", "Digital I/O" and the process buttons will divide the Space in the middle of the GUI. Some Dialog like "Command set", "Navigator", "Magnifier", "Histogram" and "Log terminal" stay untouched.

  Cascade — By clicking this menu(Window#Cascade) the "Program editor", "Camera viewer", "Profile memory", "Result display", "Digital I/O" and the process buttons will cascade from top left to bottom right. Some Dialog like "Command set", "Navigator", "Magnifier", "Histogram" and "Log terminal" stay untouched.

Figure 2.17. Cascaded and tiled window

Cascade **Tile**

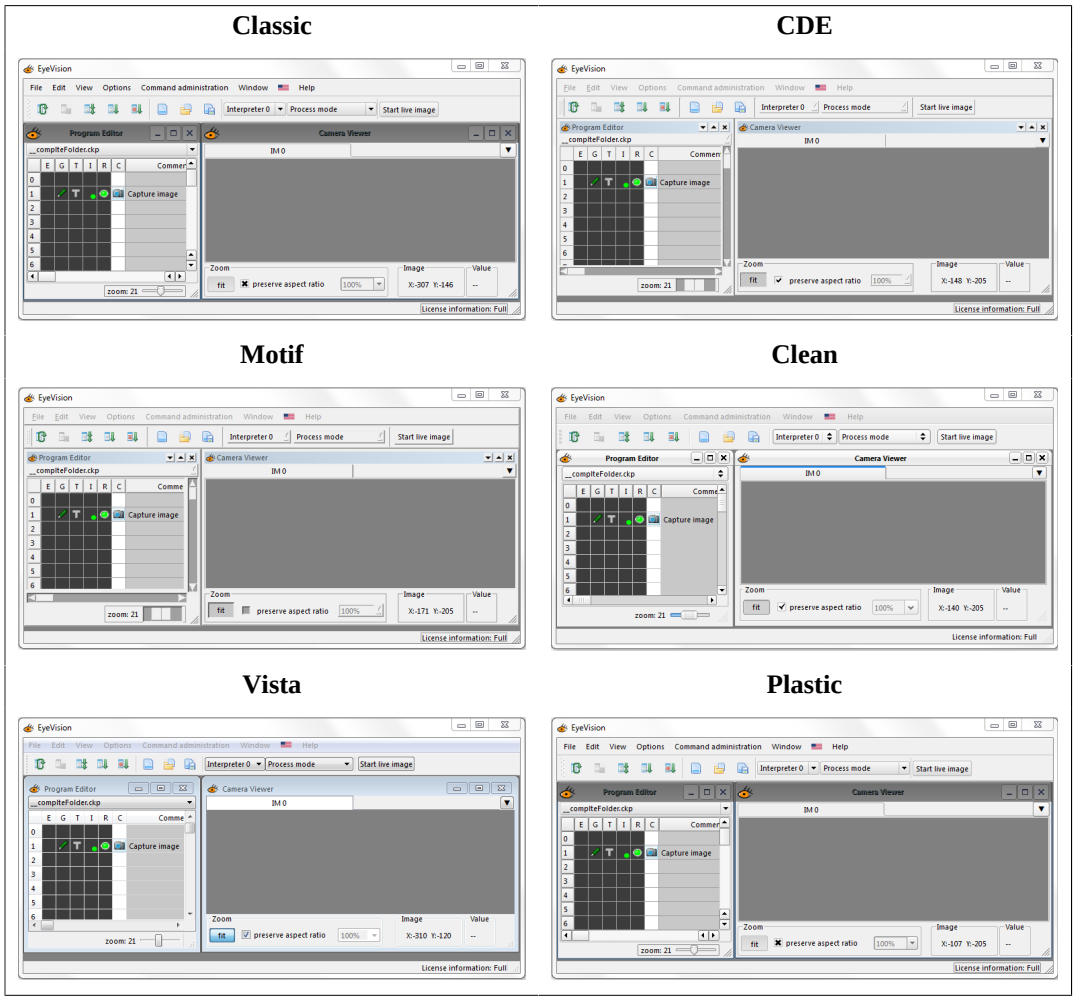


You can hide he main menu bar: To use this option go to (Window#Hide menu bar). This can also be done by pressing the key **F11**. Notice that without the main menu bar you only can switch the menu bar on by the key **F11**.

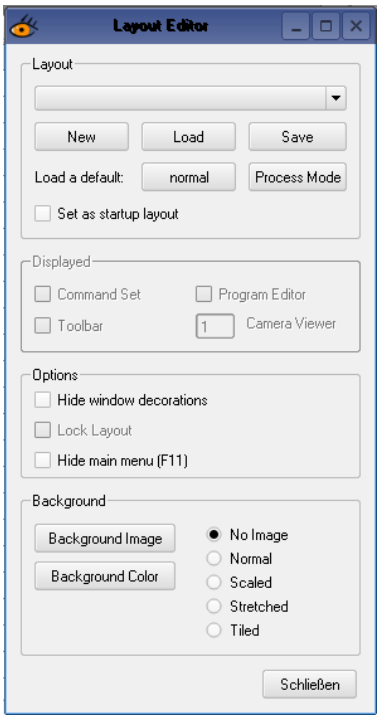
Figure 2.18. Hide menu

Different color, shape and design could been chosen as shown below in Figure 2.19, “Window design (Look and feel)”. To change the look of the window open the menu (Window#Look and Feel), then click on one of the items.

Figure 2.19. Window design (Look and feel)



Layout editor



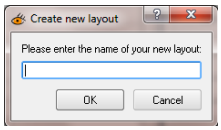
The *layout editor* can be used to create custom application layouts. The layout manager allows to generate and store different views. This involves all dialogs like *camera viewer*, *program editor*, *register viewers*. Layouts might be used for efficient integration, development of an industrial vision solution or for providing a clear and user-friendly end user GUI. During integration several views containing for example different image memories in their camera tabs and register viewers can be arranged. Open (Options #Layout editor) to create a new layout.

For example: An end user GUI might simply contain one image, a start- and stop button and a result display.

Components of the layout manager

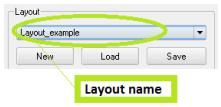
The following options are provide to created, changed or saved the layout of the user interface:

- `New`



A click on this button allows the user to input a name for a new layout. The new layout is created after pressing "OK", then this name is displayed in a combo box at the top of the window.

- `Load`



Whenever a layout should be created with a name, the name as well as the configured layout has been saved. To load an already existing layout, is it enough to scroll upward the selection list of the combo, pick the required item and then click on the button "Load". The selected layout will be loaded instantaneously.

- `Save`

  Through this options the layout has will be saved.

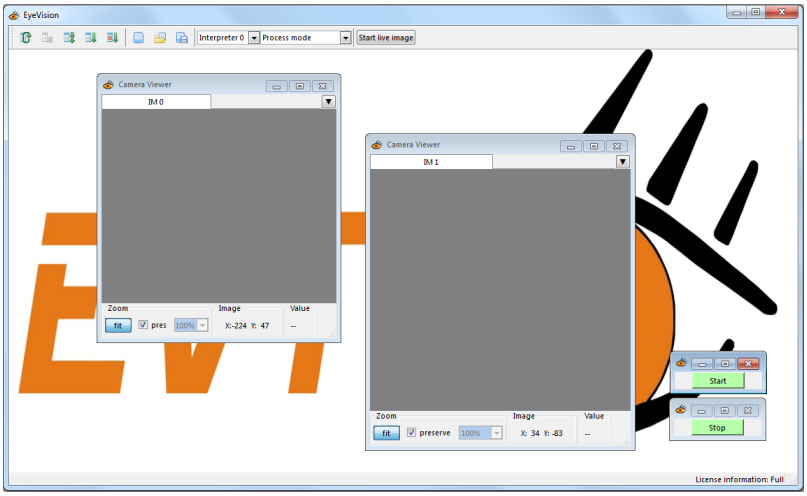
- `Normal`

  The actual view will be set to standard settings of layout.

- `Process Mode`

In this mode the user can define manually the number, the type and the position of the required dialogs (register- and camera viewer, program editor...etc) in the configured layout. An example for a view im process mode is shown in Figure 2.20, “Process mode”

Figure 2.20. Process mode



- `Set as start-up layout`

  It is enable to select the currently layout as standard . It means that the selected layout (instead the standard layout) will be loaded automatically by restart of the software

- `Displayed`

  You can select *Command set*, *Toolbar*, *Program editor* or a number of *Camera viewers* to be shown in the layout.

- Options

Hide window decorations	Hides all window title bars.
Lock layout	Lock the layout. So you can't change it.
Hide main menu	Hide the main menu bar.

- `Background`

  The customization of background in various ways. Either the user choose an image or a color or both as background.

Background image	The user can pick out an image from a modal file dialog. This image could be displayed in different ways:
Background color	A color can be chosen from color palette. This color is underneath the image, it means that the selected color is only visible, when the image not cover the entire background.
No image	The image is not displayed. If a color has been selected in the options for background definition, only this color is shown.
Normal	The image is displayed with the original size of the file.
Scaled	The image is scaled and centered.

Stretched	The image is enough stretched to cover the whole background.
Tiled	The image is used with the original width and height, and duplicated to obtain tiled window in the whole background.

Internals: config file format

All settings of the layout manager are stored using the EyeVision config module. This section provides information about the files.

User process Graphical User Interface

This section introduces how to create a custom user interface.

View -> Process Button -> Process GUI opens a custom GUI that allows to modify parameters of the active programs. User controls (Buttons, Sliders, ...) will be displayed in the Process GUI. All controls are connected to EyeVision's inbuilt script interpreter and will execute a script on activity.

The GUI definition file

Controls can be manually defined in the file `processbutton.txt`. This file is located in the data directory. File -> Show data directory will open this folder. If `processbutton.txt` does not exist, it should manually be created.

The following listing shows an example GUI definition file. All contents will be described in later section. List of available GUI controls

```
CommandGUI; commandGUI.sbi; Show parameters of Blob 1; Blob1
CommandGUI; commandGUI.sbi; Show parameters of Blob 2; Blob2
CommandGUI; commandGUI.sbi; Show parameters of Blob 3; Blob3
Button; blobColor.sbi; White objects; $userconfig.ProcessButton.Colormode; 1
Button; blobColor.sbi; Dark objects; $userconfig.ProcessButton.Colormode; 2
Button; blobExecutionMode.sbi; Stop Blob detection; $userconfig.ProcessButton.Executionmode; 1
Button; blobExecutionMode.sbi; Start Blob detection; $userconfig.ProcessButton.Executionmode; 2
Slider; blobXPosition.sbi; X position; $userconfig.ProcessButton.SliderX; 0; 500; 0
Slider; blobYPosition.sbi; Y position; $userconfig.ProcessButton.SliderY; 0; 480; 0
SpinBox; blobXPosition.sbi; X width; $userconfig.ProcessButton.Width; 50; 70; 50
```

List of available GUI controls

A list of available GUI controls

Slider — A slider is defined by the line

```
Slider; <ScriptName>; <Caption>; <ParameterName>; <Minimum>; <Maximum>; <InitialValue>
```

The following example will provide two sliders that are labeled "X position" and "Y position". Whenever the value of the slider is changed, the current value will be written into the user configuration "ProcessButton.SliderX" or "ProcessButton.SliderY" before the according scripts are executed.

```
Slider; blobXPosition.sbi; X position; $userconfig.ProcessButton.SliderX; 0; 500; 0
Slider; blobYPosition.sbi; Y position; $userconfig.ProcessButton.SliderY; 0; 480; 0
```

PushButton — A push button is defined by the line

```
Button; <ScriptName>;<Caption>;<ParameterName>;<ValueOfParameter>
```

The following example will create two push buttons with caption "White objects" and "Black objects", that will execute a script called "blobColor.sbi" whenever pressed. Before executing this script, a value "1" or "2" will be written into the user configuration "ProcessButton.Colormode"

```
Button; blobColor.sbi; White objects; $userconfig.ProcessButton.Colormode; 1
Button; blobColor.sbi; Black objects; $userconfig.ProcessButton.Colormode; 2
```

SpinBox

A SpinBox is defined by the line

```
SpinBox; <ScriptName>; <Caption>; <ParameterName>; <Minimum>; <Maximum>; <InitialValue>
```

A script called <ScriptName> can be defined. This script will be executed, whenever the value of the spinbox changes. Before executing this script, the current value of the spin box will be set to the parameter <ParameterName>.

The following example will provide a spin box that is labeled "Width". Whenever the value of the spinbox is changed, the current value will be written into the user configuration variable "ProcessButton.Width" before a script "blobXPosition.sbi" is executed.

```
SpinBox; blobXPosition.sbi; Width; $userconfig.ProcessButton.Width; 0; 500; 0 Slider; \
blobYPosition.sbi; Y position; $userconfig.ProcessButton.Width; 100; 110; 105
```

LineEdit

A LineEdit is defined by the line

```
LineEdit; <ScriptName>; <Caption>; <ParameterName>; <InitialValue>
```

A script called <ScriptName> can be defined. This script will be executed, whenever the value of the line edit changes. This happens, whenever the user calls the button apply. Before executing this script, the current entry of the line edit will be set to the parameter <ParameterName>.

The following example will provide a line edit that is labeled "Please enter your text". Whenever the entry is applied, the current value will be written into the user configuration variable "ProcessButton.LineEdit" before a script called "lineEdit.sbi" is executed.

```
LineEdit; lineEdit.sbi; Please enter your text; $userconfig.ProcessButton.LineEdit;
```

InputBox

An InputBox is defined by the line

```
InputBox; <ScriptName>; <Caption>; <ParameterName>; <InitialValue>
```

A button is provided that will open a modal input box, whenever the user presses this button. The input box allows to add any text. This text can be applied by clicking to OK or ignored by clicking Cancel. A script called <ScriptName> can be defined. This script will be executed, whenever a new value has been set using the input box. Before executing this script, the text will be set to the parameter <ParameterName>.

The following example will provide a input box that is labeled "Please enter your text". Whenever the entry is applied, the current value will be written into the user configuration variable "ProcessButton.InputBox" before a script called "lineEdit.sbi" is executed.

```
InputBox; inputBox.sbi; Please enter your text; $userconfig.ProcessButton.InputBox;
```

CommandGUI

A CommandGUI button allows to open the configuration GUI of a specific command. Pressing this button will always stop the running program and open the GUI afterwards.

```
CommandGUI; <ScriptName>; <Caption>; <CommandName>
```

The below example will provide a button labeled with "Show parameters of Blob 1". Pressing this button will stop the running program. If a command called "Blob1" exits, the accorging configuration

GUI dialog will pop up and allow to modify the parameters of the command. If more than one command "Blob1" exist in the program, the dialog of the first instance of the command will be opened.

```
CommandGUI; commandGUI.sbi; Show parameters of Blob 1; Blob1
```

Creating a result display

This is a quick guide to use the EyeVision result display. This guide shows how to add a new result display to the GUI and how to access it from the script interpreter.

Creating a new result display

Create a new result display in menu (View#Result display). The new result display is automatically named, starting with “Result Display”, “Result Display 0”. The result display names can be manually modified in the drop-down menu of each result display.

Every new result display provides initially two text fields. Fields can be removed and new fields can be added. Initially the fields are named with captions “LE0”, “LE1”.

Accessing the result display from EyeVision

The fields can be accessed from the script interpreter using the Result Display (RD) functions.

Short function description

The following function can be used to access the result displays:

```
RDSet(string completeElementPath, string value, long colorIndex)
RDFlush(string displayName)
RDFlushAll()
```

The function DISet stores a value inside an internal list. The name of the list can be chosen. elementName must be chosen as the name of the field was defined.

Parameter colorIndex defines the objects color, colorIndex:

- 1: good
- 0 : bad,
- 2 : black

Function `DIFlush` flushes a list to the defined result display window. Note, `DIFlush` doesn’t empty the list, use `DIReset` for this. DIReset resets the List.

Parameter timeStamp is currently not evaluated and will be handled in future releases internally. Please provide an empty String (“”) instead. Future versions might remove this parameter

Example:

The following script interpreter example accesses two result displays named "Result display 1" and "Result display 2".

```
RDSet("Result display 1.LE0","abc",0)
RDSet("Result display 2.LE0","anc",0)
RDFlush("Result display 1")
```

```
RDFlush("Result display 2")
```

The above example calls RDFlush for both displays. This is interesting, if different displays should be updated at different points of time. Below example uses RDFlushAll() to do the same.

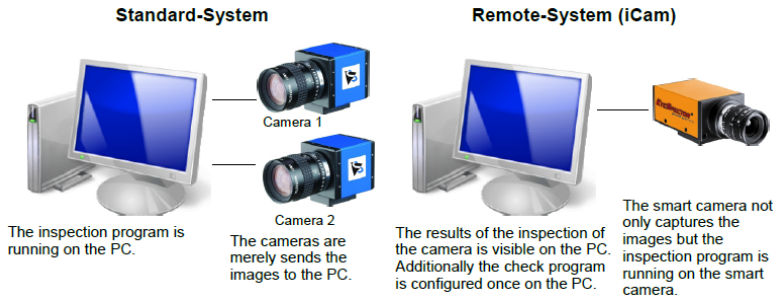
```
RDSet("Result display 1.LE0","abc",0)
RDSet("Result display 2.LE0","anc",0)
RDFlushAll()
```

Chapter 3. Device control and configuration

Standard system/ remote system

You can connect a camera directly to your computer (via LAN, USB, Firewire, Camera link). The inspection program run on the computer. This is called a standard system. You can also have a smart camera or an other computer which runs the inspection program. This is a remote system. Depending on the selected hardware and system configurations the software starts in two different modes. If the target system Remote is programmed the system starts with the remote menu. If the system is run with a connected camera, it starts with the standard menu

The difference between the Remote System and the Standard system can be described as follows:



In the “Configuration” mode the program is configured. In the “Check” mode the inspection is carried out. In a Standard-system, the check mode runs on the PC. Contrary to a Remote-system, where the inspection program runs on the smart cameras.

In a Remote system: If the user want to change to the configuration mode, the camera (which is in the check mode) has to be connected at first. Now an inspection program can be configured, which then is saved on the camera. If the camera gets disconnected again the camera changes into the check mode once more and is carrying out the newly assigned task.

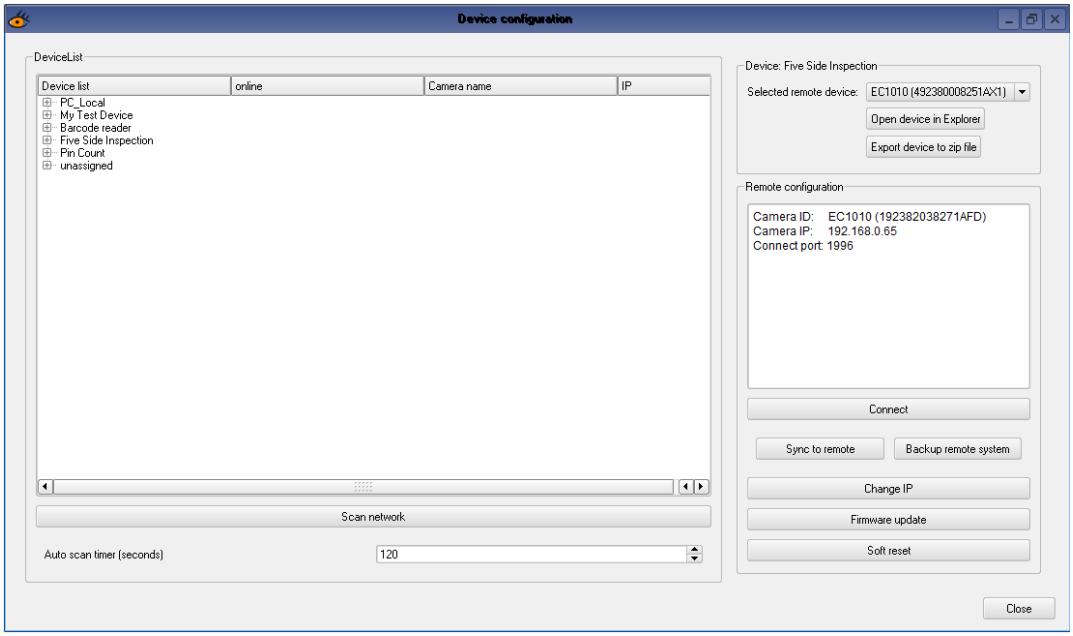
- Remote mode

  The remote mode starts when the inspection process runs on the smart camera. If the camera is disconnected from the programming software, the smart camera changes into the inspection mode automatically. To change into the configuration mode the user has to connect with the camera first via (Options → Device configuration (see Figure 3.1, “Fig: Device control and configuration.”).

Introduction

EyeVision can manage the configuration of a large number of *remote systems* inside the local network. A remote_system can be a smart camera or a runtime that runs on the same or different PC. To manage the remote systems, the user can group his work in separate projects.

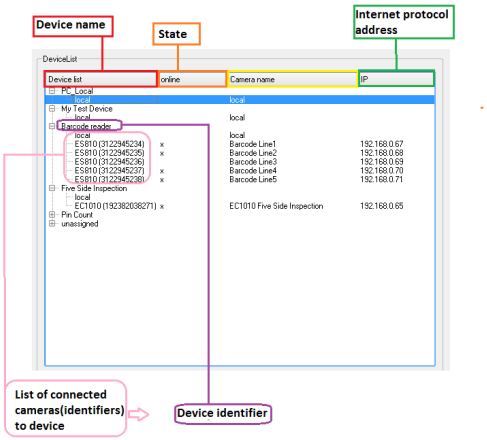
Figure 3.1. Fig: Device control and configuration.



A project should always be stored and managed in it's own project folder called *device*. The default location of projects is the folder Devices in the application directory. A *device* can contain one or more inspection programs, images and data.

One or more remote systems can be assigned to one *device*. This allows to duplicate an application via few mouse clicks. One remote system can only be assigned to one device at the same time.

Device lists



By the Remote Device control the communication medium for all non-monolithic systems is available, thus the systems with those the programming surface and the run time system are separate.

With a newly installed system there is maybe no camera set up. To find the camera the network have to be scan. A list of all cameras remote devices with which a connection has been established in the past, is listed.

- *`Device list`*

  The list of all existing (and in system found) devices is shown.

  device list — Each device that exist in the programming system has a name und this name is here listed. The device of all remote systems that been connected to a certain local device are saved and shown

below the name of this device as a list in tree view. So the device name ist at the bottom of a tree view, where leaves are cameras identifiers.

online — This describe the state of (remote) camera , that has been assigned to a local device. The state correspond to the camera state in the same row. There are three state messages:

1. Not connected – the system is not connected to the camera
2. Connected - the system is connected to the camera
3. Demo Mode is indicated, if the camera is connected but no valid license file on the camera is present.

camera name — That is the name given to the camera to identify it.

IP — The IP address on the camera, which is connected to the programming system.

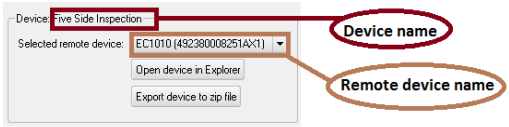
- *Scan network function*

  To refresh the view of device list or find out, if some came has been disconnected or not, a scan can be realized. Press down the button "Scan network", activate the search in network.

- *Scan timer*

  A frequently scan can be automatically started after a fixed time.

Device selection

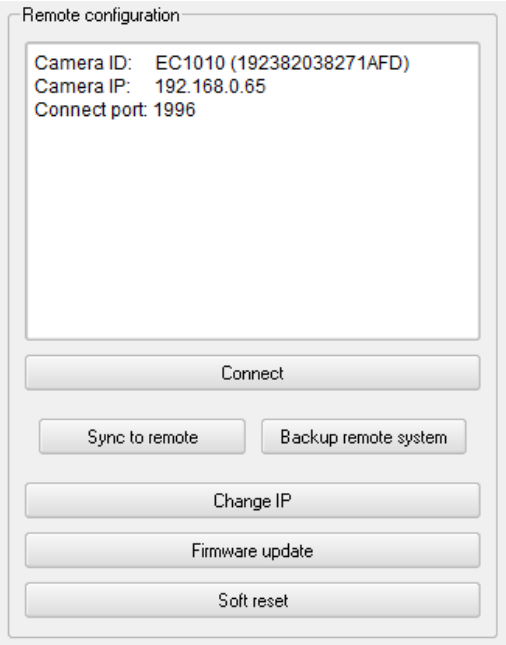


A device can be selected of the device list. After this selection, specifics operations could be execute with on the selected remote device.

Open device in explorer (Manage device files) — With the button Explorer a file viewer is opened to indicate the local copy to the camera file structure. Files can now be deleted or shifted and then be changed after Sync# Remote.

Export device to zip — Create a zip-file from the remote device.

Device configuration



In this section, it will be described, how to configure a selected remote device. A description of the selection of remote device is given in the section called “Device selection”

- **`Properties of remote device`**

  Camera ID — Name of the camera.

  Camera IP — The IP address on the camera, which is connected to the programming system.

  Connect port — Communication port for data exchange with the camera.

- **`Connect`**

  After a device was selected, a connection with the system can be done. If the option inspection program on the camera is activated then the test program will be interrupted.

- **`Syn to remote`**

  With this button the current file system image, which is on the programming system, is synchronized into the camera. That makes sure that all for the execution needed files are also on the camera (e.g. the character set files for the OCR reading).

- **`Change Ip`**

  With this button a request for input is opened, over which a new IP for the camera can be assigned. This address must be entered in the IP address area above otherwise no connection with the camera can be developed.

- **`Firmware update`**

  With this button the Camera Update process is activated.

- **`Soft reset`**

  Restart the camera.

- **`Backup remote system`**

  With Create Backup a complete Backup is made by the camera. This contains all files that are on the camera. Now it is possible to put the camera always back into the same condition as it was at the Backup time.

Chapter 4. Menus

Data Viewer: Register and Pickup lists

The basis element of all instructions which take measurements are the data structures of EyeVision. They represent geometrical objects and values in form of points coordinates, radius...etc. A differentiation can be made between six data structures: points, lines, circle, values, points list, window, string.

The individual values, points and straight lines are stored in the respective registers. The register viewer allows to see the flow of the entries of new values that have been added during the instructions are running. Registers and pickup lists are the main ways to pass results such as data, pass/fail results, point lists etc between tools.

Register

A register is a memory made as FIFO ring buffer (First in First Out). The different objects can be written in and read again.

Each object that is stored in the register receives an index. The register is a stack, where the value is always added at the bottom. For this reason is the index register a variable value and change each time, when a command send a value to register.

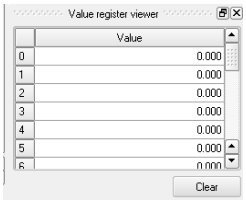
There are four register types in EyeVision:

- Value register
- Point register
- Straight line register
- Circle register

For each of the register types 20 storage locations are available. Contents of the Value, Point and Straight line registers can be seen in the Register Viewer: Select menu (View → Data viewer) to display the register.

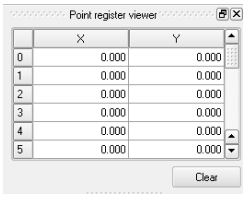
Excepts the window all objects are administered in registers.

- **Value**



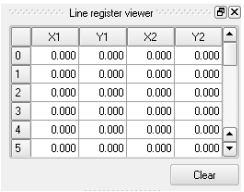
A value is a positive or negative floating point value. The values are stored in the value register.

- **Point**



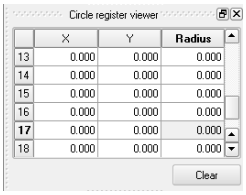
The points represent an individual point or pixel in a X-/Y-coordinate system. The points are stored in the point register as two values, an X and a Y coordinate.

- **Straight lines**



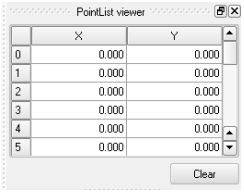
Straight lines have the size of a point and a user-defined expansion. The straight lines are stored as four values in the straight line register – the point that the straight line passes and the gradient of the straight lines.

- **Circles**



A circle consists of a point (the center of a circle) and a radius. For the circle there is no own register, the center of a circle is stored on the point register and the radius is stored in the value register.

- **Points list**



The point list is similar to the point register, this will be used however from commands that provide a large amount of data like a caliper The points in the point list will always be stored as two values the X- and Y- coordinate of the point. Against the register structure, new points are added always at the end of the point list.

- **String**

Pickup list

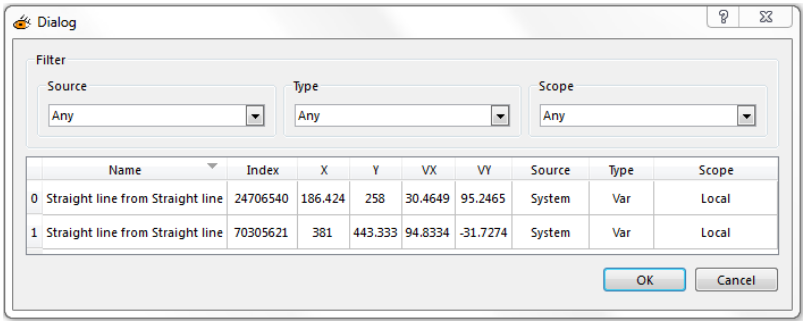
The “Pickup List” refers to the process of choosing through a tool the value or result generated by a command anywhere in the inspection program.

The pick-up lists are a global data structure which connects the result values of the different commands to their position and stores them. Now the user has a specific access to the results and is not dependent any more on possible shifts in the test program by added evaluations.

If for example two straight lines are determined in a Command "Straight line" and saved in the register. And for these lines an angle has to be determined, then the reference for the both lines can be selected in a pickup list in a command "Angle" to calculated the angle between them. Alternatively this can be realized by the pick-up List, then the respective straight line can be selected directly and is also firmly linked with the command.

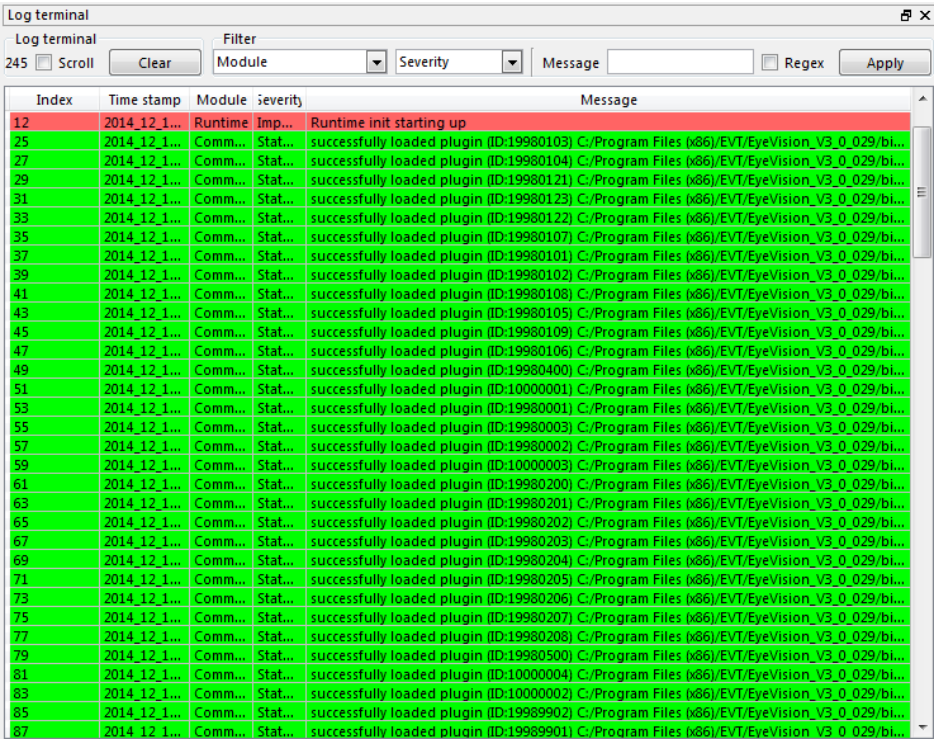
Note that the pickup shown only the value of instructions that are situated above the opened command. The view of pickup is different and depend of the type of elements such as seen in Figure 4.1, “Example of a pickup list”

Figure 4.1. Example of a pickup list



Log terminal

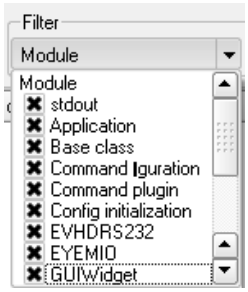
The log terminal allows you to see every EyeVision output. You can search for a special term (also in regex). You can filter the output by its severity or the sending module. This can be helpful to debug.



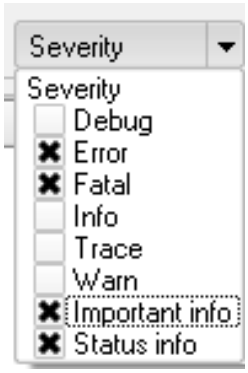
Log terminal

Row number	The first number is the numbers of rows in the log file
Scroll	If you check this option, the log terminal always jump to the newest line in the log.
Clear	The outputs in the text terminal will be deleted.

Filter



Module — Only the output from the selected module(s) is shown.

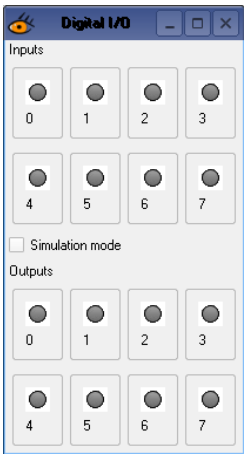


Severity — The filter select only the output with a certain severity e.g. when the user is looking for a notification with a certain priority.

Message — All the output is filtered by the entered word. You can also filter with regular expressions (check the option).

Apply — The filter is applied to the actual visible notifications: they are be filtered.

Digital I/O



The digital In-/Output display shows all digital In-/Output interfaces that are embedded into the system. Depending on the available interfaces all virtual LED will be displayed. Those displays reflect the real in/outputs and their conditions. You can also simulate the digital I/O

Simulation mode — After activating the check box "Simulation mode" the digital I/O are simulated. During the simulation mode the connected I/O hardware will be ignored. Independent from the connected hardware the inputs can be set by clicking onto the input-LED. The simulation mode is terminated when the window State of the digital in-/output is closed.

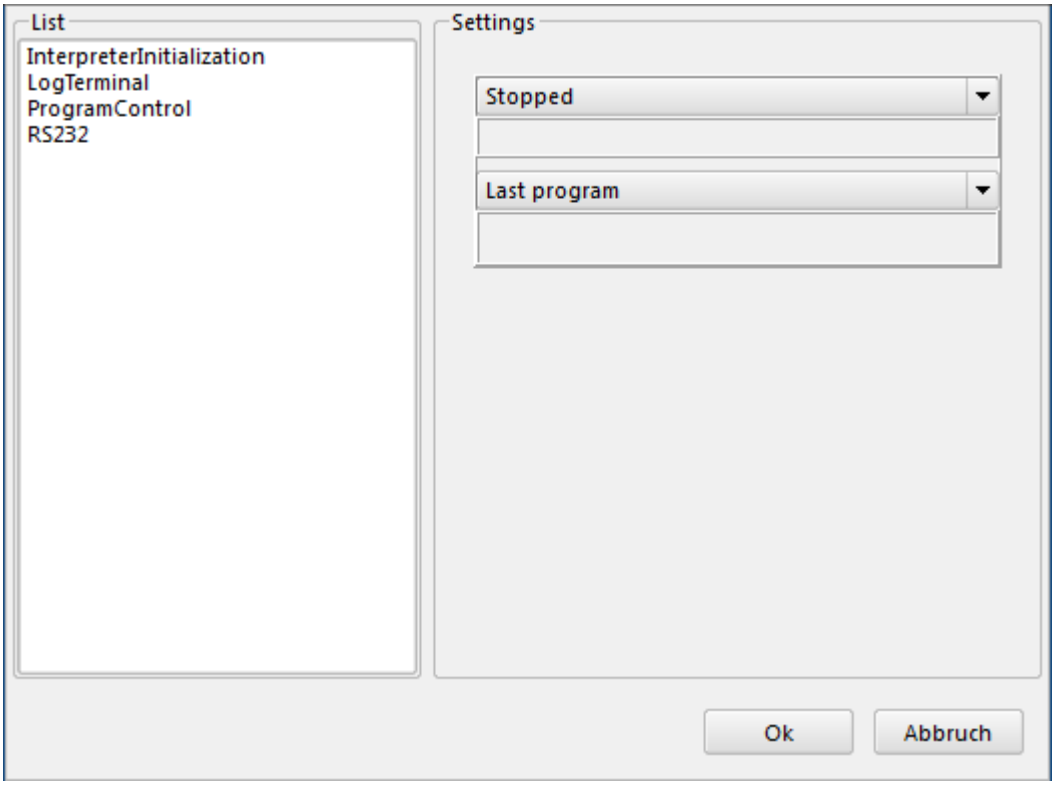
Chapter 5. Application parameter

There are four elements in the list.

- the section called “InterpreterInitialization”
- the section called “LogTerminal”
- the section called “ProgramControl”
- the section called “RS232”

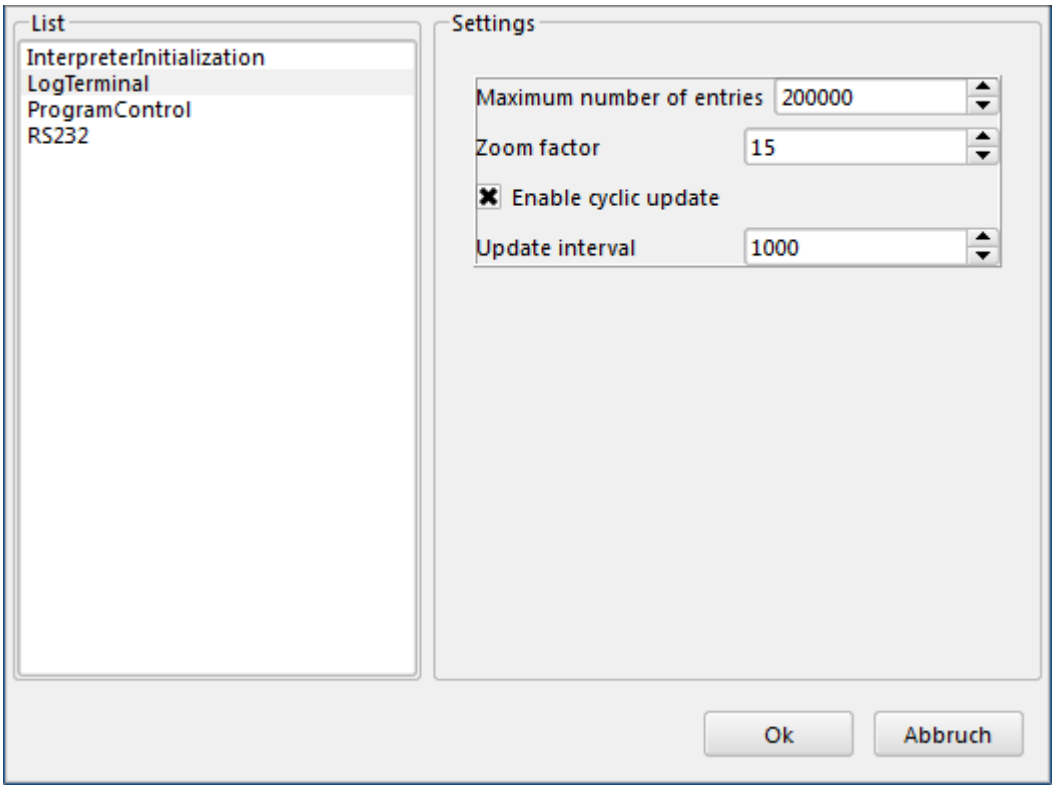
InterpreterInitialization

Consists of two drop down menus. The first drop down menu consists the three elements ("Stopped", "Endless" and "Number of cycles"). You can choose how the loaded inspection program will start. The second drop down menu consists the three elements ("Empty program", "Last program" and "Defined program"). You can choose which inspection program would be loaded at the start of EyeVision. You can always get an empty inspection program by selecting "Empty program". You can load the last inspection program by choosing "Last program". Or you can choose an inspection file from your hard disk by choosing "Defined program".



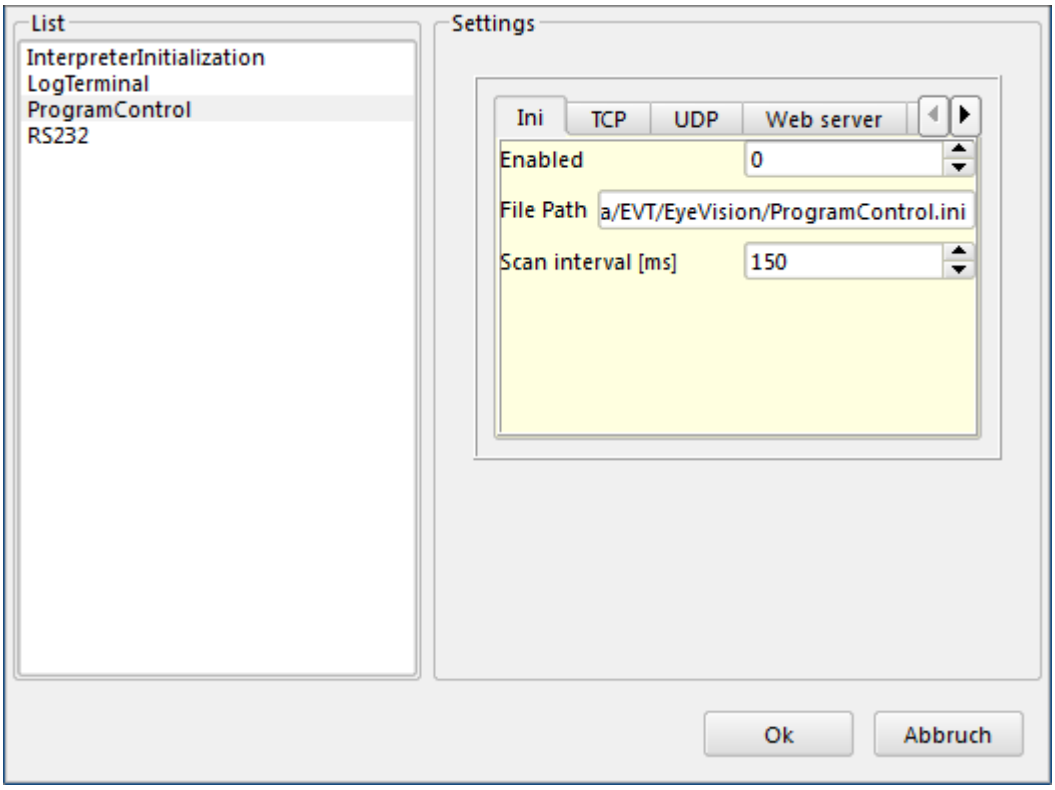
LogTerminal

The log terminal consists of three numbers and a check box. You can choose how many entries will be added and the zoom factor. The check box allows you to choose if there are cyclic updates and the update interval the time between the updates.



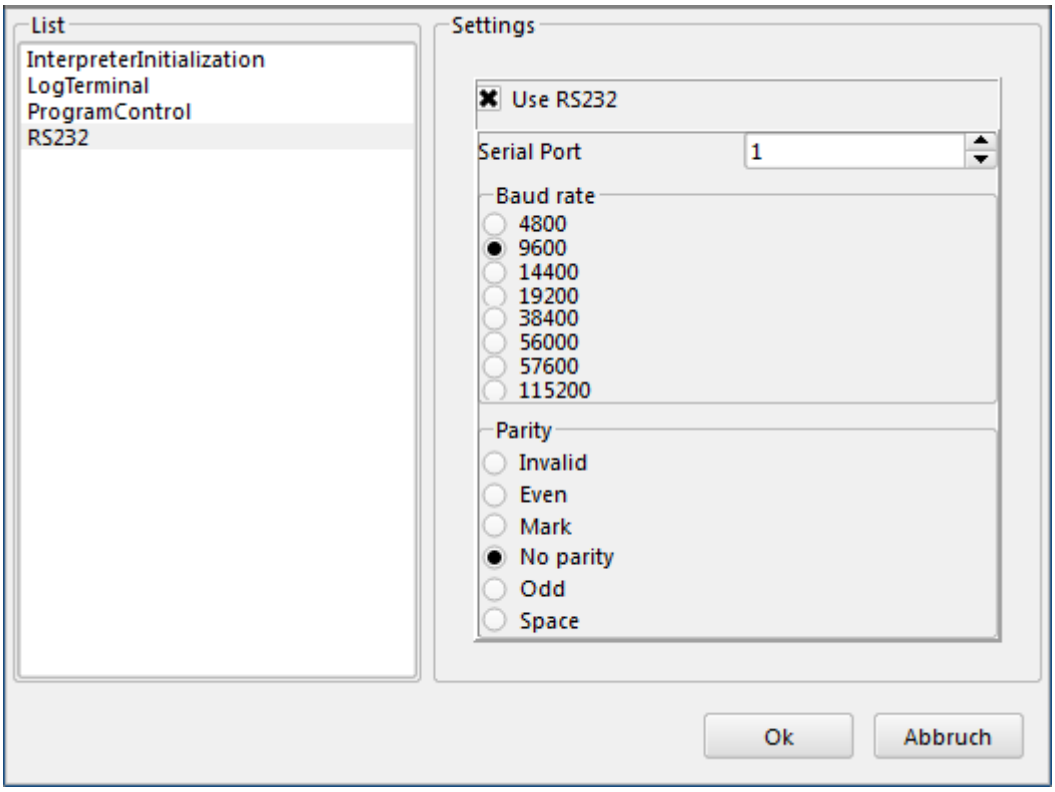
ProgramControl

Here you specify the control options for EyeVision. You can choose between an ini-file, TCP, UDP, Webserver or RS232 to controll EyeVision. For every option is a tabs in here. To enable a option you have to change the 0 behind the "Enabled" to 1. In the "Ini" tab you can choose the ini file and the interval to scan the file. For TCP and UDP you can define the listen port. The webserver allows you to specify the listen port, number of connections, maximum received file size, maximum send file size and the root path to your html files.



RS232

Here you can choose if you want to use RS232 by checking the check box. If you will use it, you can set up the right serial port numbers, the baud rate and the parity.


List
InterpreterInitialization
LogTerminal
ProgramControl
RS232
Settings
Use RS232
Serial Port
1
Baud rate
4800
9600
14400
19200
38400
56000
57600
115200
Parity
Invalid
Even
Mark
No parity
Odd
Space
Ok
Abbruch

Chapter 6. Probe elements and Tolerances

This chapter provides a detailed information about all probe elements and Tolerances.

Definitions: Probe elements and Tolerances

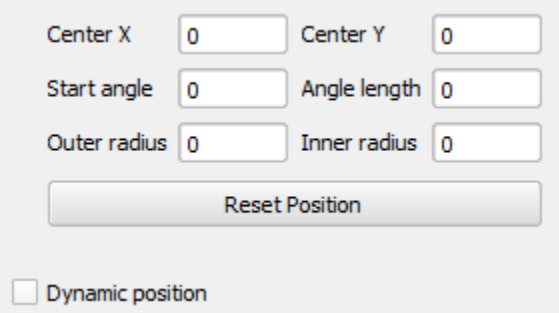
This section lists the definitions of probe elements and tolerances.

The probe parameter menu is the same for all commands. Here it is possible for the user to define different parameters for the edge detection. The standard parameters of the program are aligned so that they fit to images with lighting conditions of a bright background and a dark object. If there are different lighting conditions, the values have to be adjusted to those. In case the edge detections are made under different lighting conditions, the easiest would be to adjust the values in the global edge parameters. They will be transferred automatically while teaching in the commands. So it is not necessary to adjust the edge parameters for each command. The Configuration area for the different probes always looks similar.

Probe Circle

Description

By this menu the probe parameters for the edge detection are modified.



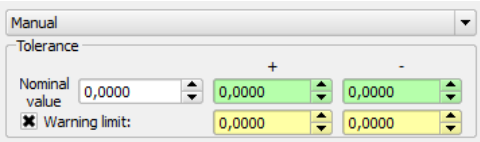
Parameters

Center X	x-coordinate of the probe center
Center Y	y-coordinate of the probe center
Start angle	Value of the start angle. The start angle means the angle between the x-Axis and the straight line defined by the circle center and the start point in the outer radius.
Angle length	It is the value of angle formed by intersection of the two straight lines that describe
Outer radius	It is the radius of the biggest circle of the probe.
Inner radius	It is the radius of the smallest circle of the probe.
Reset position	This option reset the position of the probe with the initial settings.
Dynamic position	From the detected points one point is selected depending on probe's parameters and written in the point register 0. This resulting point is the first, the last or the middle point seen from

the start of the probing rectangle. The amount points written in the point list is written in the value register 0.

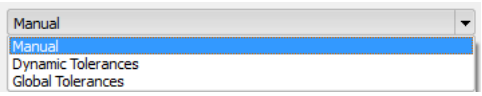
Evaluation (Tolerances)

Description



All commands that provide a result value are featuring a field where you can determine under which conditions the instruction executions should be recorded as successful or not successful or as a warning. The option “Warning limit” allows the user to react in time, before a measure exceeds a limit that is outside the tolerances, and not only when the measure has exceeded the limit. In the following the input dialogue box, which is the same dialogue in all commands, is described. Henceforth, at the individual instruction description there will only be a reference to the following description.

Parameters



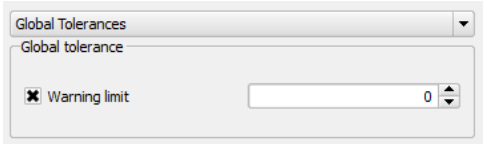
A click on the combo box scroll the options down.

- `Tolerance`

  Nominal Value — The nominal value and its modifications determine whether the result is going to be evaluated as OK or NOK.

  Warning limit — If the warning limit is active the result can also be a warning. This means that the value has not reached the NOK criteria yet, but tends towards it.

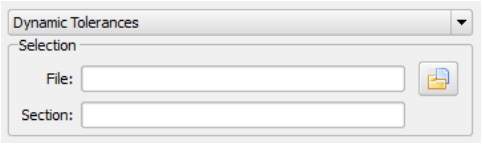
- `Globale tolerance`



If the flag is active, a tolerance can be selected out of the globally adjusted tolerances. The values from the chart will then be adopted for this command. The advantage of this approach is that the user does not need to change the tolerances at the command by entering the test program. This can be done by the set form for the global tolerances. Therefore, could it be useful to use distinct terms for the tolerances.

  Warning limit — If the warning limit is active the result can also be a warning. This means that the value has not reached the NOK criteria yet, but tends towards it.

- `Dynamic evaluation`



The user can use a own list of tolerance. The file that contain the list muss be loaded and the section name muss be written. For more information about such a list, please see the ini file definition.

File	File name (.ini)
Section	Name of the selected section.

Using dynamic tolerances

To use dynamic tolerances you need a inifile (.ini) with the tolerances declared. Your file should look like:

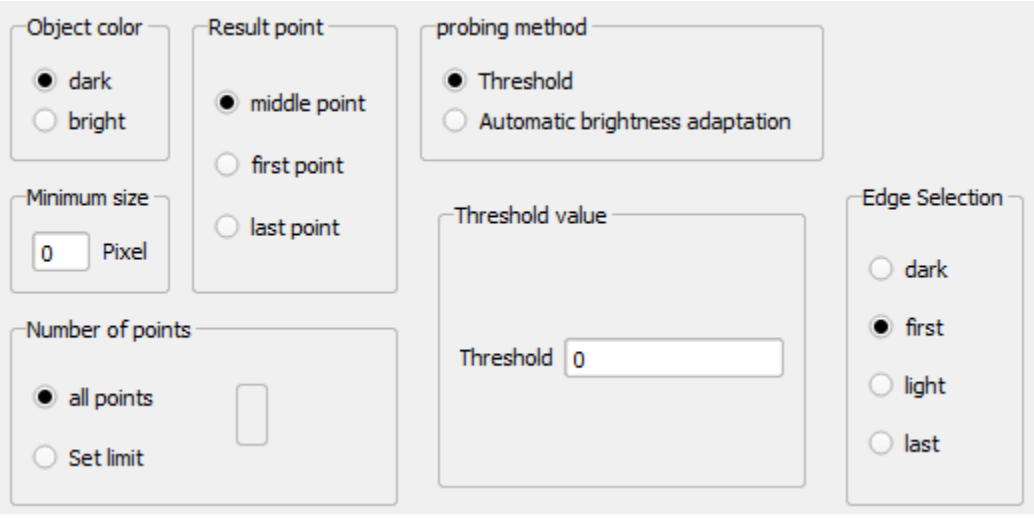
```
[Tolerances_MyCommand]
UseWarningTolerances=1
NominalValue = 10
PlusTolerance = 20.10
MinusTolerance = 20
WarnPlusTolerance = 2.3
WarnMinusTolerance = 2
```

In [] you write the name of your section. So you can have multiple section in your inifile. "UseWarningTolerances" you can switch off the warnings by "0" (false) and switch it on by "1" (true). "NominalValue" is the nominal value for your command. "PlusTolerance" and "MinusTolerance" are the limitations for the tolerances. If the warning limits ("UseWarningTolerances") is active the result can also be a warning. This means that the value has not reached the NOK criteria yet, but tends towards ("WarnPlusTolerance" and "WarnMinusTolerance") it.

Probe Parameter determination

Description

By this menu the probe parameters for the edge detection are modified.



Parameters

- `Object color`

dark	The color for the object in demand is set to dark.
bright	The color for the object in demand is set to bright.

- `Result point`

middle point	The middle point of the search basis is provided as result point.
first point	The first found point of the search basis is provided as result point.
last point	The last point of the search basis is provided as result point.

- `Probing method`

Automatic brightness adoption	Within the probe adequate parameters are searched for the edge detection.

- `Expert parameter`

Edge selection	The position of the edges is detected with the set edge parameters.
Threshold	A minimum value that should be exceeded.
Minimum contrast	Contrast that should exist at the edge position.
Pixel distance	Pixel distance concerning the contrast.
Edge symmetry	Symmetry position that is relative to the edge position.

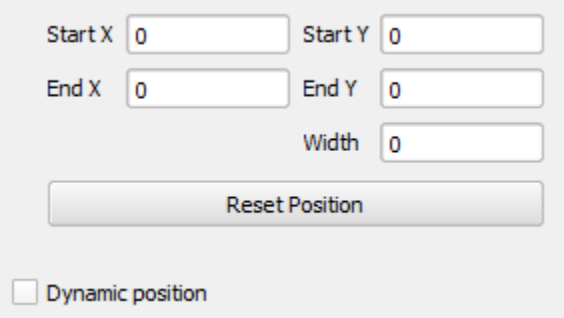
- `Number of points`

  Here you can set the number of points that should be used within the probe. By default this means that all points are reduced to one point. That way, the processing can be sped up.

Probe Rectangle

Description

By this menu the probe parameters for the edge detection are modified.



Parameters

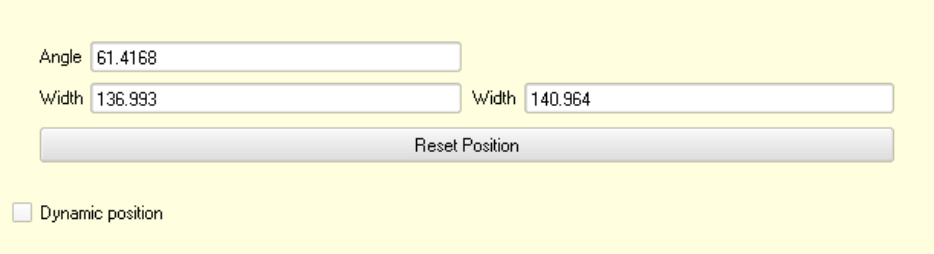
Start X	x-coordinate of the start point
End X	x-coordinate of the end point

Start Y	y-coordinate of the start point
End Y	y-coordinate of the end point
Width	Width of the rectangle probe
Reset position	Set to default values.
Dynamic position	From the detected points one point is selected depending on probe's parameters and written in the point register 0. This resulting point is the first, the last or the middle point seen from the start of the probing rectangle. The amount points written in the point list is written in the value register 0.

Caliper

Description

By this menu the probe parameters for the edge detection are modified.



Parameters

Angle	Displays the angle between the caliper and the x-axis. The measurement is anticlockwise.
Width(1)	Width of the first jaw.
Width(2)	Width of the second jaw.

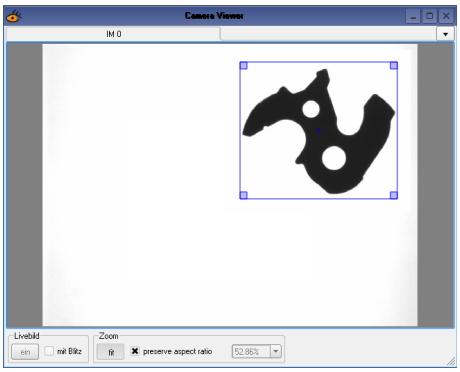
Configuration

This chapter provides a detailed informations about how to configure the probe elements.

Configure Circle

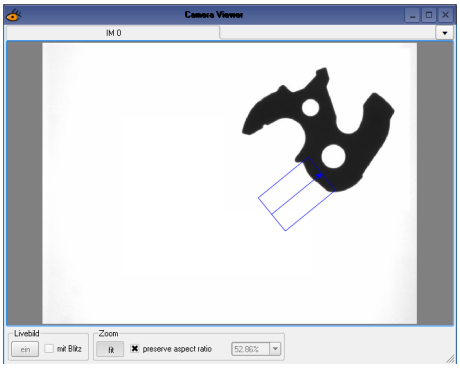
A circle can be configured analogous to a circle probe except the fact that you cannot configure sectors but only change the size of the circle. It draws a circle on the points, which were detected by the circle probe.

Configure window



The configuration of a window is analogue to the configuration of a probe.

Configure rectangular probe

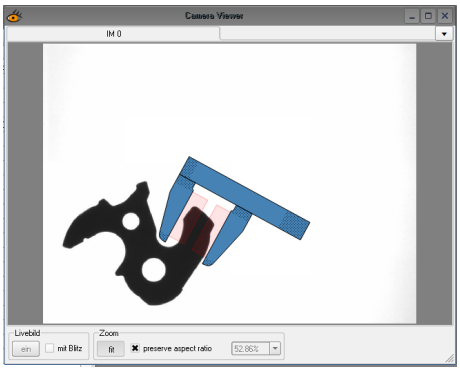


If you configure a probe a context menu is shown next to the Camera Viewer. In the Camera Viewer you can see the probe (blue). If you do not see a probe the option Reset return the probe into its starting position. Afterwards the probe has to be configured anew. If the user already knows where to position the probe the values can be also entered directly. There are different cursor forms for positioning the probe, for enlarging the probe etc. (see the section called “Cursor shapes (Camera viewer)”). You can modify the probe by clicking and holding the left mouse button.

Configure Circle Probe

A circle probe searches points on a circle or part of a circle. If you configure a circle probe a context menu is shown next to the Camera Viewer. In the Camera Viewer you can see the probe (blue). If you do not see a probe the option Reset return the probe into its starting position. Afterwards the probe has to be configured new. If the user already knows where to position the probe the values can be also entered directly. There are different cursor forms for positioning the probe, for enlarging the probe etc. (see the section called “Cursor shapes (Camera viewer)”). You can modify the probe by clicking and holding the left mouse button.

Configure Circle Probe



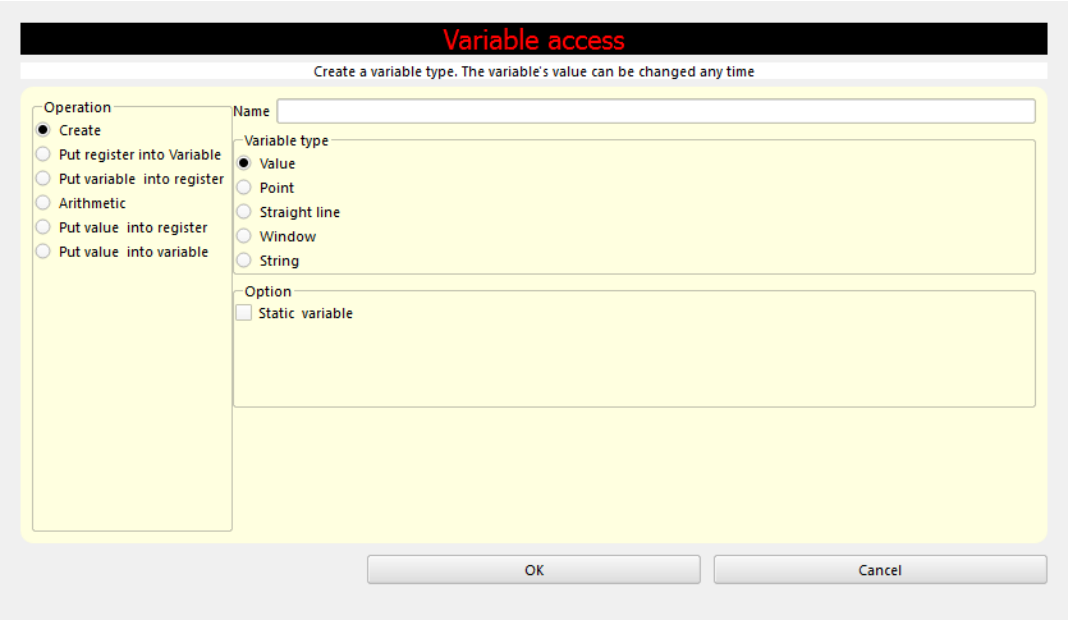
Here the distance of the jaws as well as the position of the caliper can be configured. With the caliper it is possible to measure the distance between two edges, which are positioned between the jaws of the caliper The caliper does not necessarily measure vertically, such as the shortest distance, but measures the direction the user has configured, from one jaw to the other. Clicking onto the button Configure calliper launches the window to configure the caliper as shown in the image below. The caliper can be positioned with the yellow points on the caliper. With the cursor the caliper can be dragged and turned into the desired position. Also you can drag apart the jaws to match the distance you want to measure. (see the section called “Cursor shapes (Camera viewer)”). The only parameter which maybe needs to be adjusted is the color of the search area.

Chapter 7. Command GUI

This chapter provides detailed information of the graphical user interface of the inbuild commands.

 Access to Variable

You can deal with the system variables like in a programming language with the instruction access to variables. It is possible to establish and save variables and to read them again or calculate with them. The variables can be stored in registers so they are available to the other functions of the system.



- Name

  Here you can give the variable any name. This field is not enable for all operations of command.

- Operations

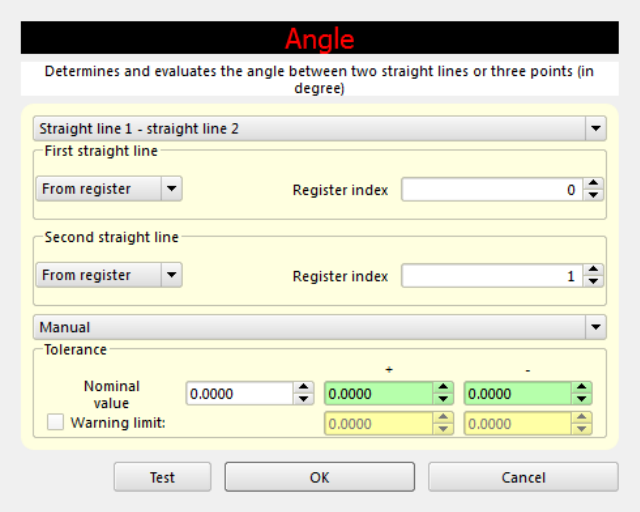
  This operations give the possibility to create a variable or manipulate an existing variable.

Create	A variable out of the selected type of variable is created. The user can write the variable name and select the variable typ (value, point, straight line, vwindow or string).You can use the option static variable to create static variables.
Put register into variable	A registrer entry is inputed into a existing variable. This variable can be a point, a value or a straight line und is choosen in a pickup list. The register index of the entry, which is copied into the selected variable must be inputed.

Put variable onto register	The content of the selected variable is put automatic onto the register with index value 0. The user have only to select in the pick list, which variable have to be copied in register.
Arithmetic	The arithmetic operation is used for the two chosen elements. The element could be selected from register, pickup list or manually inputed. The operation could be done with elements of typ point or numerical value. • **Value.** When this type is selected, the possible operators are: Addition, Subtraction, Multiplication, Division, Exponentiation , Amount,Square root , Sine, Cosine , Tangent, Degree>Radian, Radian>Degree. • **Point.** When this type is selected, the possible operators are: Addition, Subtraction, Multiplication, Division, Norm (length of a vector) **Element 1 / Element 2.** You can choose elements either from the register or from the pick-up list. **Operator.** You can choose an operation out of the list of the listed operations. Then this operator is used for the two elements.
Put value onto register	The entered value is stored in the register.
Put value into variable	The entered value is stored in the variable.

register information

Angle



With this command you can determine an angle either between two straight lines, three points or a vector between two points

- `Angle definition`

  The resulting angle in degrees is saved on the value register index 0. In this section you have to determine how the angle should be defined.

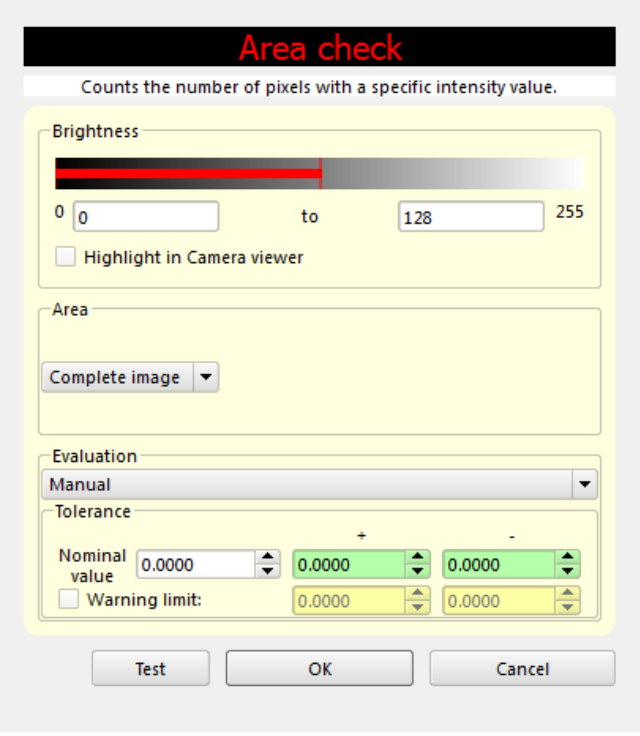
Straight line 1 – Straight line 2	The angle between straight line 1 and 2 is determined. First straight line / Second straight line Here you can select from where the straight line should be taken – from the register or by selection from the pick-up list. • **First straight line / Second straight line.** Here you can select from where the straight line should be taken – from the register or by selection from the pick-up list.
Straight line 2 – Straight line 1	The angle between straight line 2 and 1 is determined.
Three points	Three points are chosen form the point register and the spanned angle is calculated.
Vector (btw. 2 points)	The angle for the spanned vector between two points is calculated.

- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

register information

Area check



This command determines the pixels with a defined brightness in a configured window and compares this pixels with the configured number of pixels.

- `Brightness`

  The brightness area can be determined either by direct numerical input or by clicking with the left and the right mouse button into the corresponding area.

Highlight in camera viewer

- `Area`

  A window, in which filter operations are to be carried out, is defined (see also the section called “Probe Rectangle”).

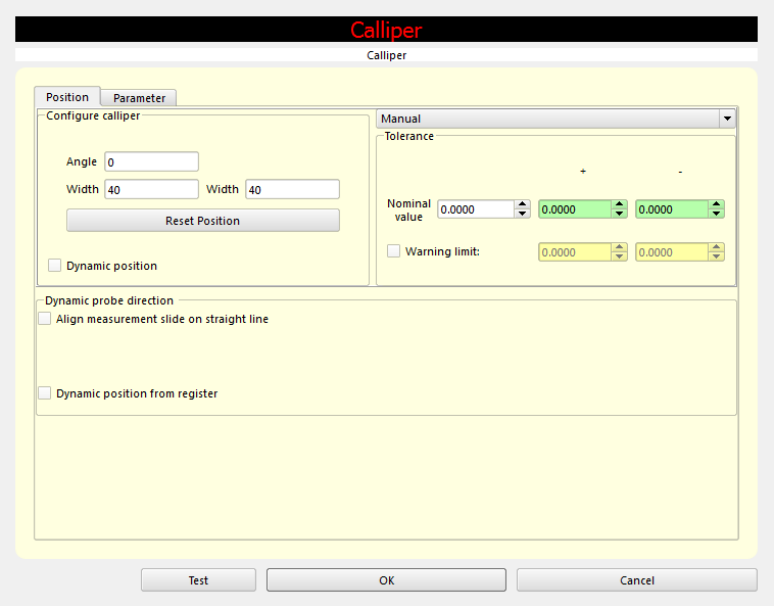
- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

Register Information

The amount of detected pixels is saved on the value register index 0.

Calliper free



The free caliper works just like a real caliper. The free calliper is positioned at a place, where it should measure. By the sliders F1 and F2 the measuring area of the two jaws can be controlled. Thus the entire measuring task is configured.

Position

- `Configure calliper`

  See also caliper.

- `Dynamic probe direction`

Align measurement slide on straight line	The caliper is aligned at an already existing straight line in the straight line list.
Dynamic position from register	The starting point is taken from the point register 0, whereas the endpoint is taken from the point register 1. The width is taken from the value register.

Evaluation and parameters

- `Use same parameters`

Define if the same parameters are used for both jaws or the parameters are separetely defined for each jaws.

- Parameter upper/ lower jaw : See also the section called “Probe Parameter determination”

- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

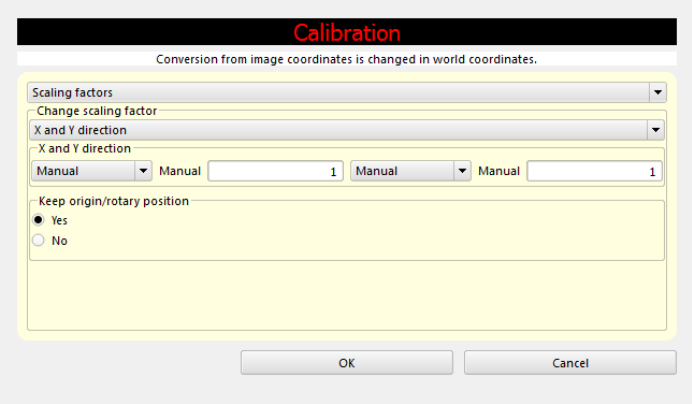
register information

The result of the measurement will be written on the value register index 0.

the caliper it is possible to measure the distance between two edges, which are positioned between the jaws of the caliper The caliper does not necessarily measure vertically, such as the shortest distance, but measures the direction the user has configured, from one jaw to the other. Clicking onto the button Configure caliper launches the window to configure the caliper as shown in the image below. The caliper can be positioned with the yellow points on the caliper. With the cursor the caliper can be dragged and turned into the desired position. Also you can drag apart the jaws to match the distance you want to measure. (see also the section called “Cursor shapes (Camera viewer)”).

The only parameter which maybe needs to be adjusted is the color of the search area. If for example you want to measure a white circle on a black background, the search color from the “Parameters of both jaws” have to be changed via double click on “G170” from white to black.

Calibration



The command calibration can be used for scaling the image area. The options of the command allows also a position tracking in X and Y as well as the rotation position.

Change of

- `Origin`

  A new origin can be defined. Either only the X- or Yvalue is changed, which means a parallel shift of the coordinate system or, both, the X- and Y- values are changed, which means that a completely new starting point of the coordinate system.

- `Scaling factors`

  The scaling factors can convert the camera coordinates, which are displayed in pixel, into global coordinates for a measure system. This can be done either by firm values or dynamically on the basis of computed values.

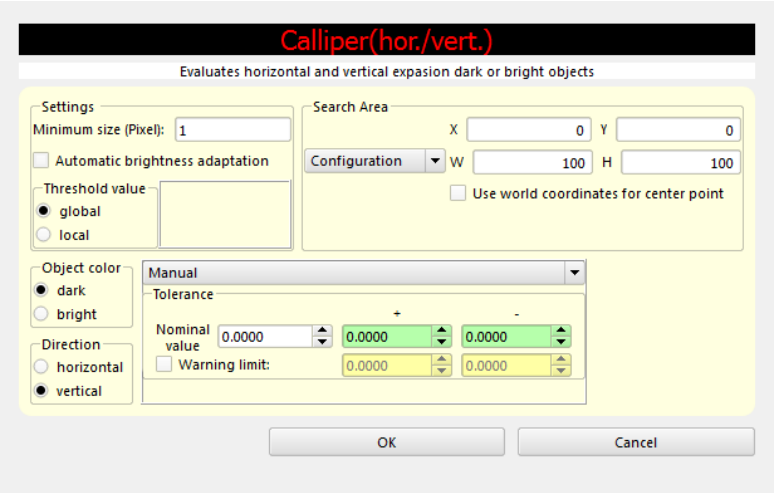
- Origin and Orientation

  With this option not only a new coordinate origin is defined but also the turning position is traced by a further point.

- Complete (3 Points)

  The calibration over 3 points is used not only when a new origin and rotation has to be defined, but also when an independent scaling becomes necessary.

Calliper (Hor./Ver.)



With this command an area is configured, in which the occurrence of a defined object color is searched. Searched horizontally, the occurrence of a defined object color starting from bottom and top, is determined. As soon as both search directions have found an object, the distance between those two positions will be determined and an imaginary straight line probe is drawn.

- Settings

  Minimum size(Pixel) — Indicates the minimum size of an object in pixel, so that it is recognized as valid.

  Automatic brightness adaptation — A threshold value will be determined automatically for recognizing an object edge.

  Threshold value
  - Global(Threshold value global):The global threshold is used for edge detection.
  - Local(Threshold value local):The indicated edge value is used.

- Search area

  Configure window — See also the section called “Probe Rectangle”.

- Object color

  Dark — It is searched for the first dark object.

  Bright — It is searched for the first bright object.

- Direction

Horizontal	Search for a defined object color starting from bottom and top.
Vertical	Search for a defined object color starting from right to left.

- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

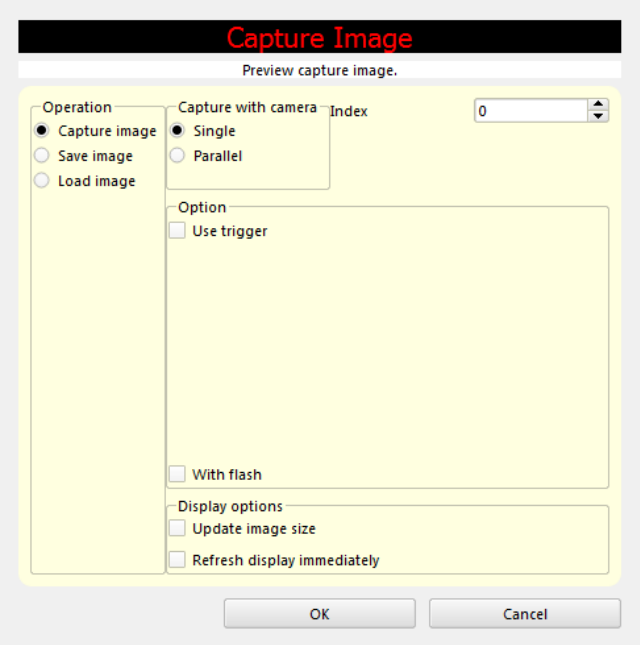
register information

The command calculates the 4 most distant points and places them on the point register:

- Point register index 0: top-most point
- Point register index 1: right-most point
- Point register index 2: bottom-most point
- Point register index 3: left-most point

The distance is saved in the value register index 0.

Image capture



With the command image capture one can master more than the name suggests. Not only is it possible to capture an image with a capture image hardware, it is also possible to load images that are saved on a file system. Furthermore is it possible to save the images that are in the image memory on a file system. The images can be saved as a .bmp or a .jpg file.

Options

Because the configure window will change according the selected operation, the respective configure windows are detailed below.

Capture image from camera

If the option Capture image from camera is activated, the Image Capture command is in its actual mode. It tries to capture an image with an image capture hardware.

- `Capture with camera`

  If several cameras are connected to the system, with this index it can be determined from which image capture canal, that is connected to a camera, (Video Input Channel) the image will be taken.

- `Options`

  - Use trigger

    Apply Timeout — If this flag is activated a timeout time can be adjusted. This means that after a certain time the waiting for a trigger signal will be cancelled. If a timeout occurs, the image capture could defective.

    Timeout: Input

    HW-triggered(activated) — If this flag is also active the hardware trigger will be activated automatically as long as it is supported by the image capture hardware. The hardware trigger should always be preferred to the software trigger.

    HW-triggered(Not activated) — If only this flag is set, the system proceeds from a soft trigger. A digital input has to be determined as trigger input and also a debounce time.

    Trigger input: In the soft-trigger mode, the input that is supposed to be used as trigger, is determined.
    Debounce time:In the soft-trigger mode the debounce time for the trigger input is determined.

  - With flash

    If this option is active an existing flash output is activated according the image capture timing.

- `Display options`

  Update image size — This option for the image capture, forces the software to adjust the image size to the current capture platform. This is necessary if e.g. between two picture recordings the image capture mode changes to Binning or a ROI for the image capture is selected.

  Refresh display immediately — If the machine vision system is equipped with a monitor output than with this option it can be forced to display the just captured image on the output unit. Otherwise it is left to the system at which time the update is carried out.

Save image

If the option Save image is activated, the Image Capture command saves image which are stored in an image memory that is part of the command, in a file that is part of the file system.

- `With overlay`

  If this flag is activated also the overlay data are registered and stored into the image. Otherwise only the gray-value image will be saved.

- `Save image`

  Fixed file — When choosing this option a name and a storage position for the file can be assigned by the file selection button. In case the file is also to be used later in a target system it is useful not to change the storage position since it will fall short of the synchronization.

Continious	If the option progressive is selected a prefix name is defined by the file selection. Date and time will be added and together will be used as a file name.
Cyclic	With this option an index number is added cyclically to the basis file name. The index runs within the default from 0... N and starts again with 0. If the option entire folder is activated all images in the selected folder are loaded sequential. With the button Reset Index the index is reset to the default settings.
Text string as name	With this option the file name is determined by the global string. Thus it is now possible to assign a name to an image file dependent on the testing part and its evaluation, e.g. NIOPart_4711.

Load image

If the option Load image is activated, the Image Capture command will load an image, which is stored in a file at the file system, into the main storage.

- `Load image`

Fixed file	When choosing this option a name and a storage position for the file can be assigned by the file selection button. In case the file is also to be used later in a target system it is useful not to change the storage position since it will fall short of the synchronization. • Delete after succeful loading If this is active the image file will be deleted after loading.
Cyclic	With this option an index number is added cyclically to the basis file name. The index runs within the default from 0... N and starts again with 0. If the option entire folder is activated all images in the selected folder are loaded sequential. With the button Reset Index the index is reset to the default settings.
Text string as name	With this option the file name is determined by the global string. Thus it is now possible to assign a name to an image file dependent on the testing part and its evaluation, e.g. NIOPart_4711.

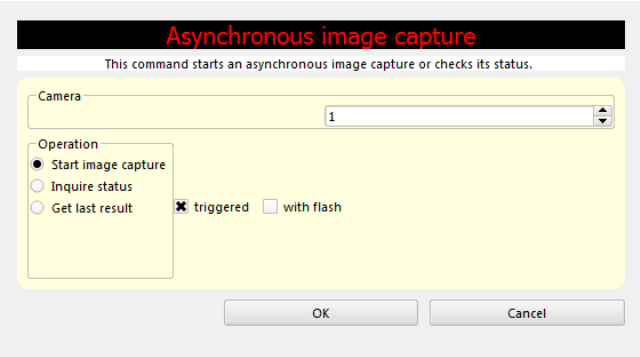
- `Display options`

Update image size	This option for the image capture, forces the software to adjust the image size to the current capture platform. This is necessary if e.g. between two picture recordings the image capture mode changes to Binning or a ROI for the image capture is selected.
Refresh display immediately	If the machine vision system is equipped with a monitor output than with this option it can be forced to display the just captured image on the output unit. Otherwise it is left to the system at which time the update is carried out.

- `Image source`

Unknown	The image geometries are unknown.
Camera	The image geometries, which are adjusted in camera N, are used.

Asynchronous image capture



This command starts an asynchronous image capture or checks its status. This means that the image capture in the camera can be launched through a trigger pulse (or flash impulse) at any time. An asynchronous image capture is used for fast running test objects as well as for irregularly occurring objects.

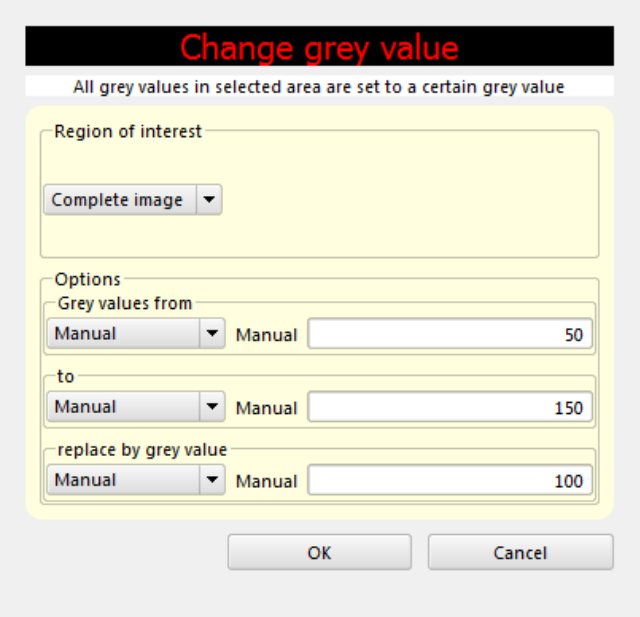
- `Camera`

  The camera index must be fixed.

- `Operations`

Start image capture	The image acquisition can be implemented via trigger or flash.
Inquire Status	Here one can inquire the status of the triggered image acquisition or wait until the image capture is finished. The command starts the image capture and comes back immediately. If “inquire status” is active it comes back and tells you if its ready for configuration. Additionally one is able to define if the image capture should be restarted (with flash) or if a timeout should be applied.
Get last result	Here one can inquire the result of last image capture.

Change grey value



All gray values in the selected area are set to a specified gray value.

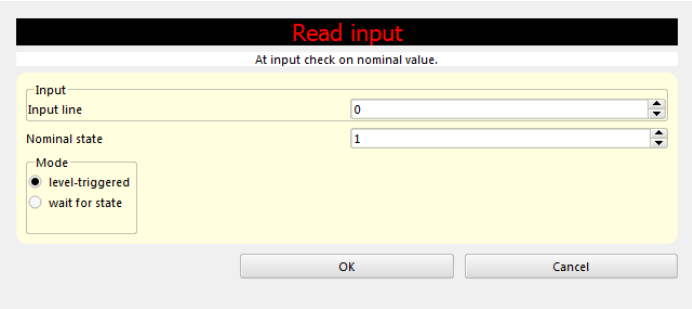
- Circle

  If the center of the circle and the radius are either taken from the register ,determined by the pickup list or configured manually, depends on the selection.See the section called “Probe Circle”.

- Options

From	This means "Gray values from". Here you can specify the minimum gray value manually, or from the register or select it by the pick-up list.
to	Here you can specify the maximum gray value manually, or from the register or select it by the pick-up list.
replace by grey value	You can determine which gray value, the gray value of the source image should be set to.

Check input



A digital input is checked for the expected condition. The input can be accessed either via the index or via the input name in the In-/Output the menu for the respective assigned input.

- `Input line`

  By a sequential index (beginning with 0 for the first input) the entrance can be accessed.

- `Nominal state`

  By comparing the input with the target condition the command announces OK when equal or otherwise NOK.

- `Mode`

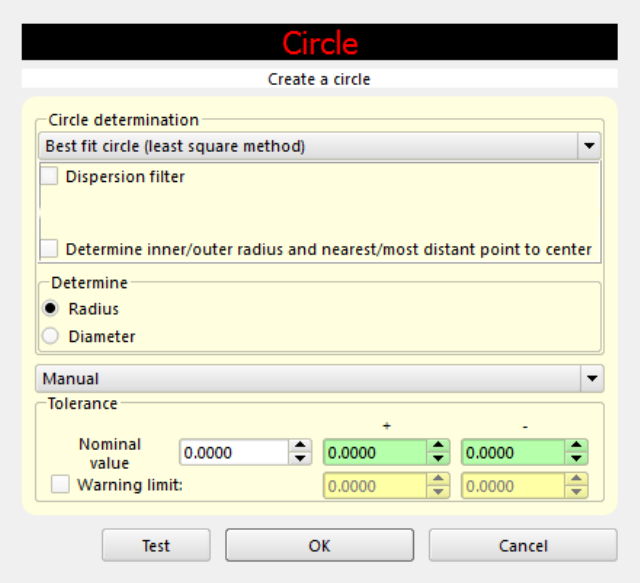
  level-triggered — If this option is active, the IO announces if the target condition is reached.

  wait for state — If this option is selected, the IO announces if the signal changes from the current condition into the target condition.

  Timeout — By clicking on this option, the user can input a value for timeoout instead the standard condition.

register information

Circle



This command determines a circle either out of the points of the point list or by three points which are supposed to create a circle. As a result the command puts the center point onto the point register and optionally the radius and the diameter to the value register.

- `Circle determination`

  This operations give the possibility to create a variable or manipulate an existing variable.

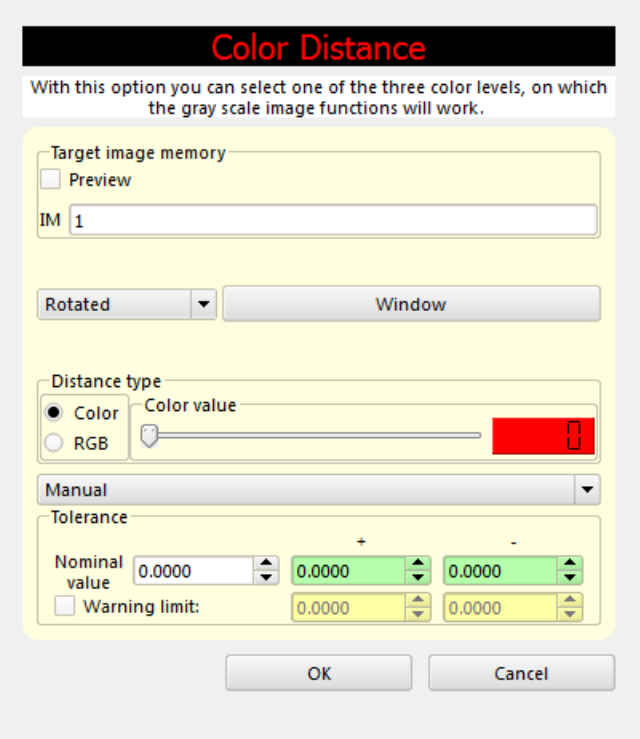
Best fit circle (least square method)	The optimum circle via the points of the point list is computed.
Dispersion filter	Makes sure that points, that would escape, are deleted.
Center and radius from register	The center point for the circle will be taken out of the point register and the radius out of the value register.
Determine inner/ outer radius and nearest/ most distant point to center	The inner/ outer radius will be determined. The results are saved on the register as follows: • Point register index 0: The center of the circle • Point register index 1: The nearest point to the circle center • Point register index 2: The most distant point to the circle center • Value register index 0: The radius of the circle • Value register index 1: Distance from the nearest point to the circle center • Value register index 2: Distance from the most distant point to the circle center
Circle through 3 points	The last three points of the point register are taken to determine the circle via them.
Save diameter into register	The diameter is put on the register.

- Tolerance

  For more informations, see the section called “Evaluation (Tolerances)”.

register information

Color distance



This command generates a gray scale image, in which the distance according the selected distance type to the reference color value is determined.

- Target image memory

  Preview

  IM In Image memory (IM 0,..., IM N) the target image is displayed.

- Window confoguration

  In the selected window the distance concerning the reference color is determined. The more the color value differs from the reference value, the darker the window is displayed. On the contrary, it is logic that the closer the color value is to the reference value, the brighter the window is displayed.

- Distance typ

  Color Determines a reference color value. This color value can either set manually (red input window) or by selecting the eyedropper (cursor now appears in the shape of a eyedropper) and than click onto the desired color value of the image in the Camera Viewer. The “new” color value is taken on automatically:

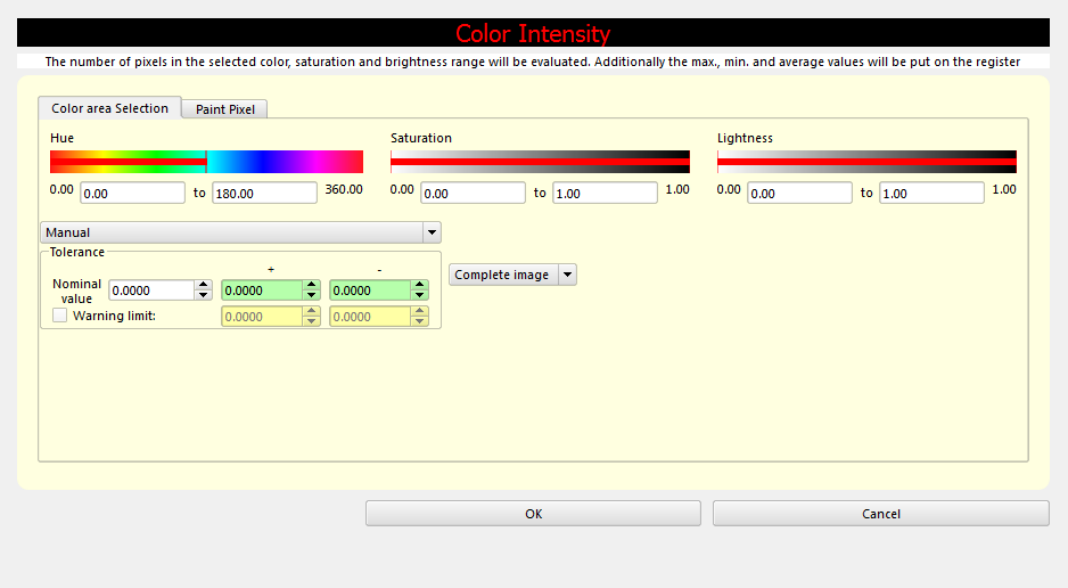
RGB The gray scale image is generated with regard to the distance to a RGB value. The values can be determined either by the slide control or manually typed into the respective field. The still black field on the left image then display the values according color.

- Tolerance

  For more informations, see the section called “Evaluation (Tolerances)”.

Color intensity

With this command, the amount of pixels in the selected color range will be evaluated. For this, it can be defined by the color, saturation and brightness range. The desired color value of the image can be selected with the eyedropper.



Color area selection

- Hue

  The color range can be limited either by numerical values or by moving the slider.

- Saturation

  Here you can determine the color saturation. The saturation can be adjusted either by direct value input or by moving the slider.

- Lightness

  Here you can set the brightness value. The brightness value can be adjusted either by direct numerical input or by moving the slider.

- Tolerance

  For more informations, see the section called “Evaluation (Tolerances)”.

- `Window position`

  The evaluation window is either defined by direct numeric input or is interactively defined by pressing the button window position.(see also the section called “Probe Rectangle”).

Paint pixel

- `No painting`

  The pixel are outputed in the original color.

- `Paint result`

  If this option is active the pixels matching the criteria will be displayed in color.

  In grey image
  - **IM**: Choose the Image memory
  - **Reset** :
  - **Good**:
  - **Bad**:

  Good

  Bad

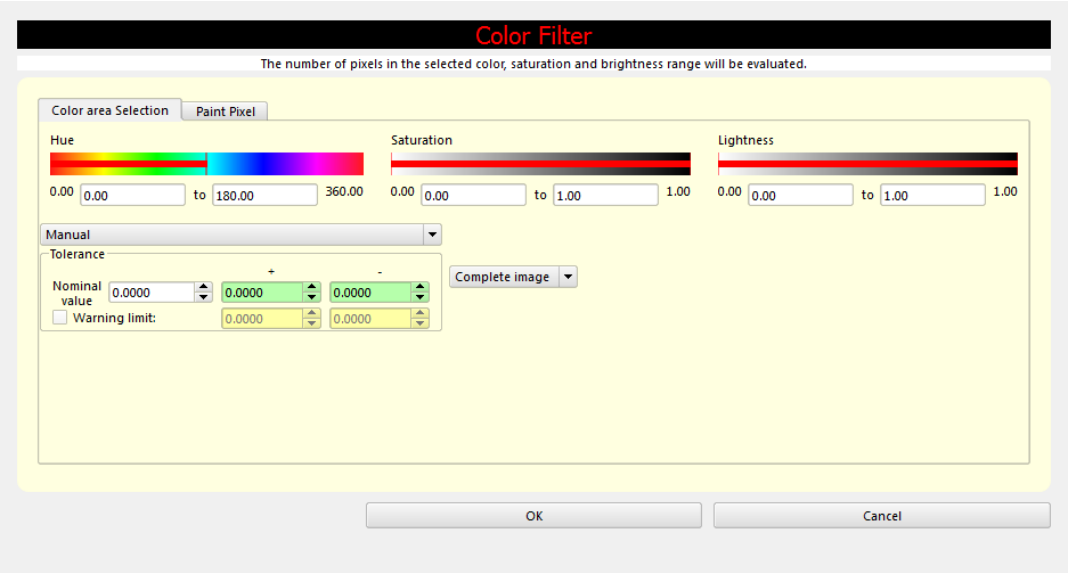
register information

When pixels are detected, the following data will be written onto the register:

- Value register index 0: the amount of pixels
- Value register index 1: hue, average
- Value register index 2: hue, minimum
- Value register index 3: hue, maximum
- Value register index 4: saturation, average
- Value register index 5: saturation, minimum
- Value register index 6: saturation, maximum
- Value register index 7: brightness, average
- Value register index 8: brightness, minimum
- Value register index 9: brightness, maximum

 Color filter

With this command, the amount of pixels in the selected color range will be evaluated.



Color area selection

- `Hue`

  The color range can be limited either by numerical values or by moving the slider.

- `Saturation`

  Here you can determine the color saturation. The saturation can be adjusted either by direct value input or by moving the slider.

- `Lightness`

  Here you can set the brightness value. The brightness value can be adjusted either by direct numerical input or by moving the slider.

- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

- `Window position`

  The evaluation window is either defined by direct numeric input or is interactively defined by pressing the button window position.(see also the section called “Probe Rectangle”).

Paint pixel

- `No painting`

  The pixel are outputed in the original color.

- `Paint result`

  If this option is active the pixels matching the criteria will be displayed in color.

  In grey image
  - **IM**: Choose the image memory.

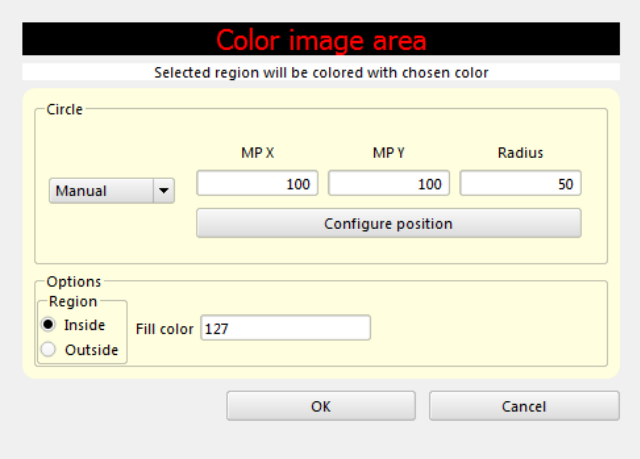
- **Reset** :
- **Good**:
- **Bad**:

Good

Bad

register information

Color image area



Inside or outside the defined circle range the image will be colored with the selected color.

- Circle

  If the center of the circle and the radius are either taken from the register, determined by the pickup list or configured manually, depends on the selection. See the section called “Probe Circle”.

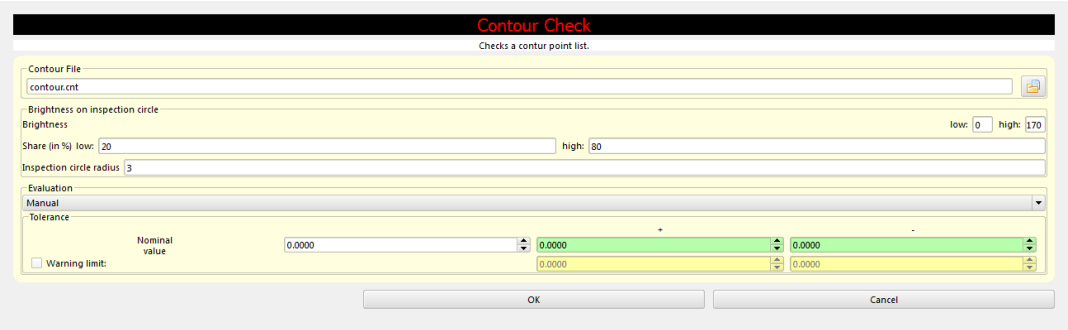
- Options

  Region — inside: Determines whether the inside of the circle should be colored with the fill color. outside: Determines whether the outside of the circle should be colored with the fill color.

  Fill color — The fill color marks a value between 1.. 255.

Contour check

For the object, which should be checked an existing contour point list is loaded and for each point of the object contour an inspection circle is determined to test if the brightness value in the inspection circle corresponds to the criteria. The points that do not meet the criteria will be evaluated.



- Contour file

  With the file selection an existing contour file, which was developed for the object, can be selected.

Brightness	The expected brightness on the inspection circle is entered here.
Share	The percentage share of the brightness on the inspection circle is given.
Inspection circle radius	The radius of the inspection circle is specified

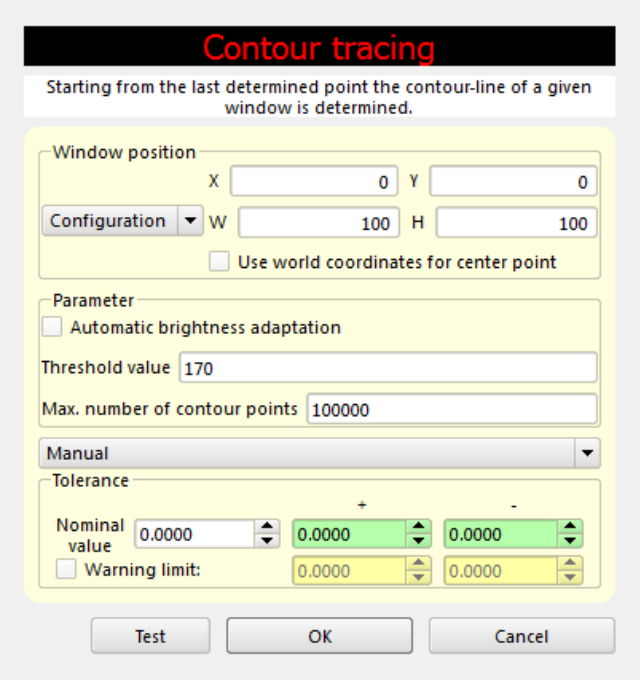
- Brightness on inspection circle
- Tolerances

  For more informations, see the section called “Evaluation (Tolerances)”.

Register information

This command writes to the value register index 0 the amount of those points, for which no counterparts could be found.

Contour tracing



With this command the outer edge of an object is traced and the number of points belonging to the object are returned as a result. If the point list was reset before, the points, belonging to the object contour, are on the point list and can be saved. The command also needs a starting point that belongs to the traced object contour. This can be defined e.g. by the probe command.

- `Window position`

  Configuration of window in camera viewer (see the section called “Probe Rectangle”).

- `Parameters`

  The resulting angle in degrees is saved on the value register index 0. In this section you have to determine how the angle should be defined.

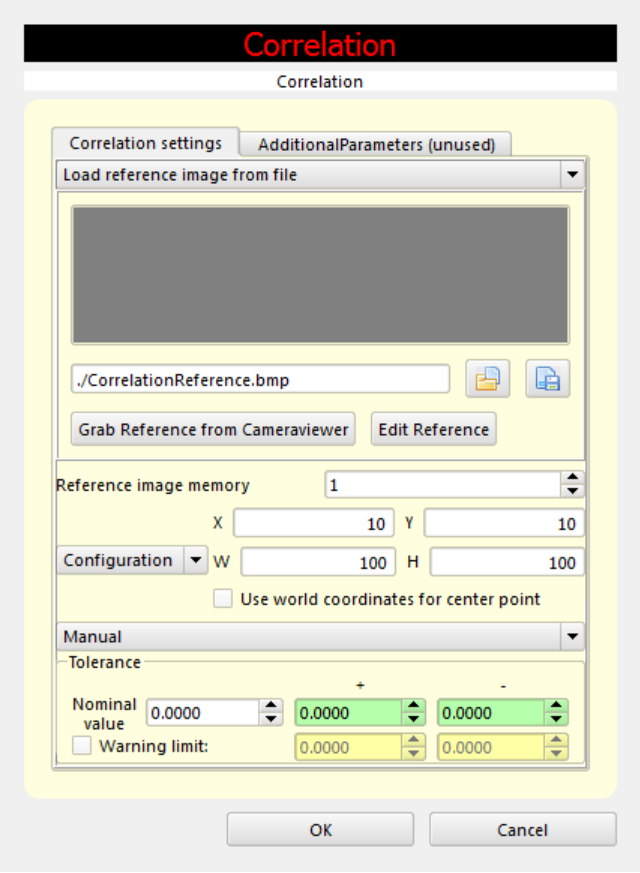
Automatic brightness adaption	If this option is active, a threshold value, which represents an edge at the neighboring point, is searched.
Threshold value	A given threshold for the edge search of the contour points is used.
Max. number of contour points	Determines how many contour points should be collected maximally. When the value is reached, the function stops searching. Hence areas of the contour can remain disregarded.

- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

register information

Correlation



The correlation command looks for an already configured gray pattern in the search form. If it detects the pattern its position is reported. The correlation with the configured pattern is evaluated. There are different correlation methods available, depending on the hardware platform. Platforms with Intel or AMD processors e.g. Sony smart Cam, additionally provide the real time correlator Kojak, which allows to find objects in any turning position.The correlation command looks for an already configured gray pattern in the search form. If it detects the pattern its position is reported. The correlation with the configured pattern is evaluated. There are different correlation methods available, depending on the hardware platform. Platforms with Intel or AMD processors e.g. Sony smart Cam, additionally provide the real time correlator Kojak, which allows to find objects in any turning position.

Correlation settings

- Pattern

In this register card you determine the source from where your reference image is loaded. You either choose directly a bitmap, which will be loaded or you choose this image dynamically out of the image memory. Furthermore, a pattern, which has been compiled with the pattern editor, can be chosen as reference.

Important

It should be indicated, that, of course, the reference image and the image that has to be checked are not to be saved in the same image memory. Therefore you have to choose an image float index in a free image memory.

Load reference image from file

Save/Open image: Depending on the method a bitmap will be loaded.
Grab reference from Camera viewer : Alternatively such a pattern can also be created dynamically, i.e. an image will be readout of the image memory. In this image sample points will be marked in the to be tested section and so, a pattern is created. After pattern will be created in the pattern editor, it could be saved (See save/open image).

Use exiting reference image — Use a generated image from a image memory. This is discribed in detail below.

- Reference image

  Reference image memory — Frame buffer index.

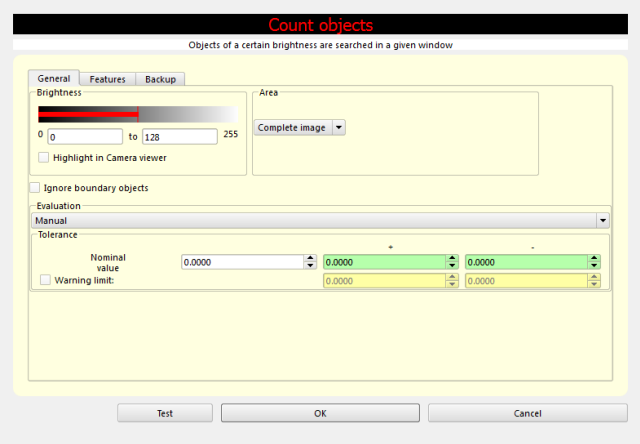
  Search Area (Window) — The search area in which a pattern should be searched and the granularity of the search, is determined.

  The search area in which correlation should take place, should be determined. Either by direct input or by Configure. (see also the section called “Probe Rectangle”)

- Tolerances

  For more informations, see the section called “Evaluation (Tolerances)”.

Count objects



The command count objects is a very universally applicable command. It is based on the fact that the image contains various regions of brightness, which are separated clearly from each other. With the input of a range or with the eyedropper the object color is determined, which is then used for the separation of the object surface. In order to find out whether the correct objects are determined, it is sufficient to click onto the test button. Thereafter all ranges with the same color are indicated. If too many objects are found they can be limited by activating one of the possible object filters.

Example 7.1. For example:

if you want to find only objects with a surface around 1400 pixels then you indicate in the object filter the min. value with 1300 and the max. value with 1500. Thus the fluctuations on the surface caused by changing the lighting, can be balanced and only objects with the desired size is indicated. In order to use the filter it is important that the filter is activated.

General

- Brightness

  The determination of the brightness range can be carried out either by direct input of the values or by moving the slider.

Highlight in Camera Viewer — Color the found object(s) in the camera viewer.

- Ignore boundary

  If this flag is active, objects, whose object color is identical with the window border, are deleted.

- Area

  Window — Configuration of an area in which the objects are to be determined (see also the section called “Probe Rectangle”).

- Evaluation

  Tolerances — For more informations, see the section called “Evaluation (Tolerances)”.

Features

- Output in different image memory
- Area

  If active only objects, whose area is between these two values, are output

- Distance (from center)

  If distance from center is active, only objects, whose distance from center lies between min. and max. are output.

- Width

  If width is active, only objects, whose width lies between min. and max. are output.

- Height

  If height is active, only objects, whose height lies between min. and max. are output.

- Point of balance

  Marks the point of balance of an object.

- Object borders

  Marks the bounding box.

- Ext. features

  If this flag is active, then Object-filter are extended to the options “Angle” and “Excentricity”, and Display is extended to the options “Mj. Axis”, “Angle” and “Excentricity”.

  Angle — *Filter*:If this option is selected, only objects with angles between minimum and maximum are shown.
  Display: The angle of an object is displayed.

  Excentricity — *Filter*:If this option is selected, only objects with an eccentricity between minimum and maximum are shown.
  Display: The excentricity of an object is displayed.

  Mj. axis — *Display*: The major axis of an object is displayed.

Backup

- Amount

  Here the number of object-results, which should be saved, are provided

Save all	All objects are saved.
Save a limited number	Maximum Objects Save in:

- Save in

  Here the user can select, where the object-results should be saved, e.g. in the pickup list.

  Point list: Saves the point of balance coordinates of the found objects into the point list.
  Point register: Saves the point of balance coordinates of the found objects into the point register.
  Extern access:

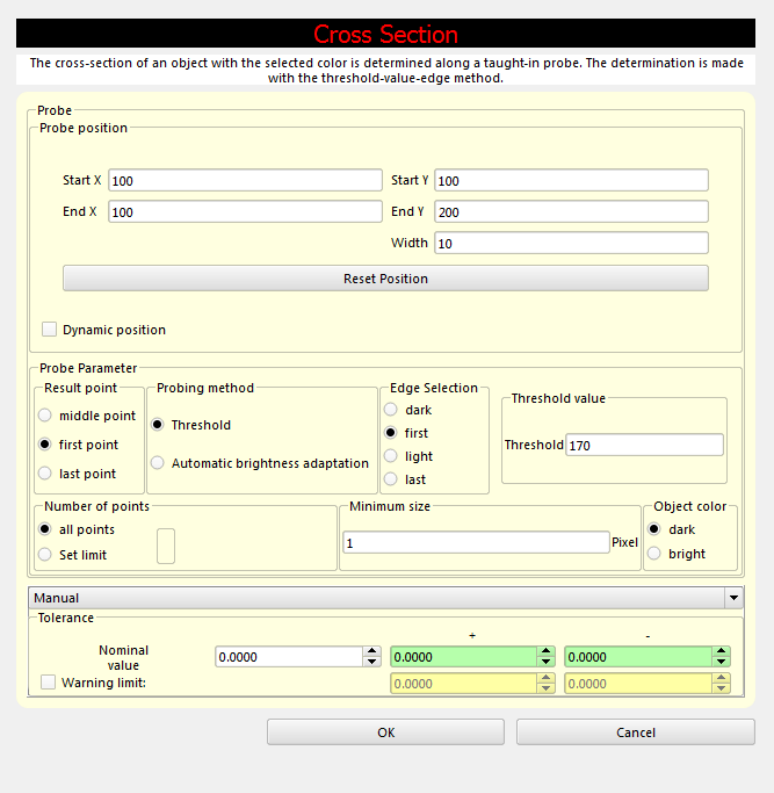
- Sorting

  Here it can be determined, how the objects should be sorted and in which order.

Register Information

The amount of found objects is written in the value register index 0.

Cross section



A cross-section of an object is determined on a configured position.

- Probe position

  The position of the search line is configured, see also the section called “Probe Rectangle”.

- Probe parameter

  The parameter for the edge determination can be defined (see also the section called “Probe Parameter determination”).

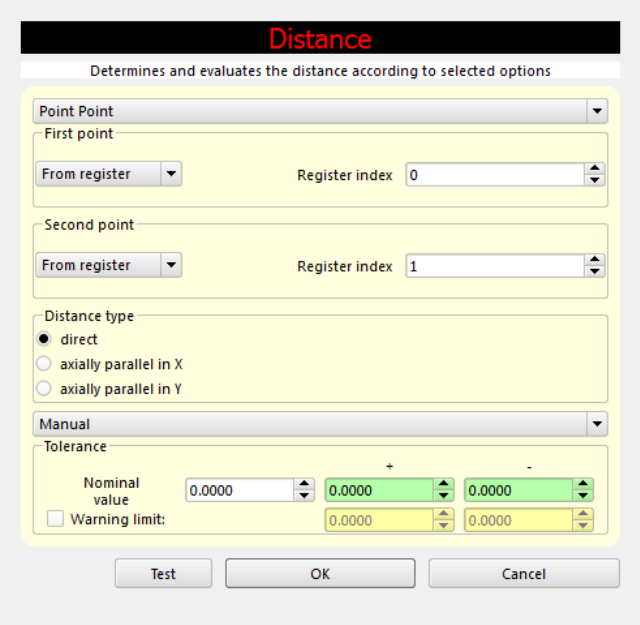
- Tolerance

  For more informations, see the section called “Evaluation (Tolerances)”.

register information

Two points are written on the point register indexes 0 and 1 and the distance between them on the value register 0.

Distance



According to selected options, the command Distance determines the distance between two points (see also the section called “Data Viewer: Register and Pickup lists”) or between a point and a straight line.

- Distance determination

  Following options can be choosen to determinate the distance:

Point Point	Determines the distance between tow points. How the points are determined is defined in “distance type”.
Point Straight line	Determines the distance between a point and a straight line. How those are determined is defined in “distance type”.

  First point/ second point. You can choose where to pick the two points. You can choose between the different registers and the pickup list.

- Distance typ

Direct	The direct distance is determined.
Axially parallel to x	The distance on the parallel x-axis is determined.
Axially parallel to y	The distance on the parallel y-axis is determined.

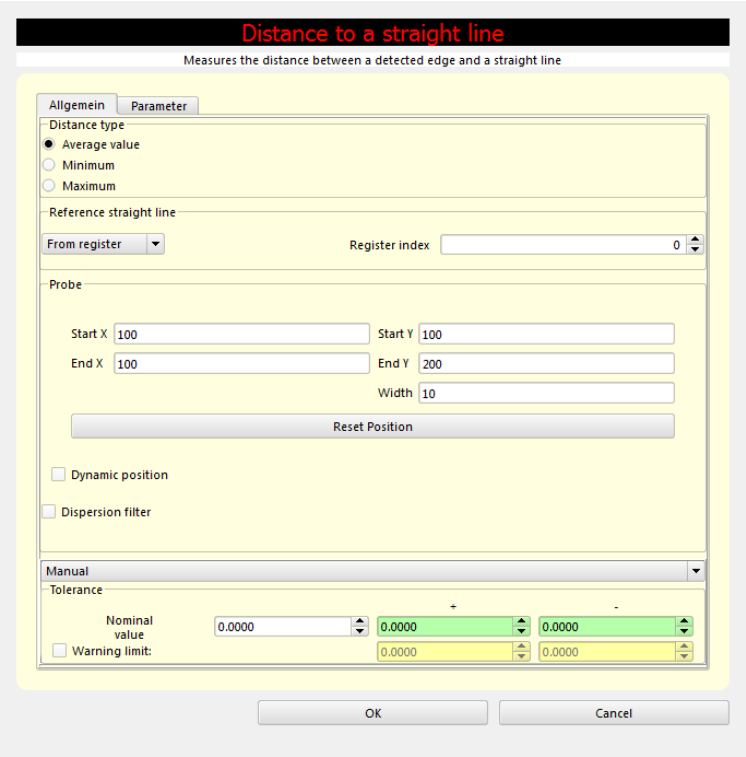
- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

register information

The distance is written in the value register index 0.

Distance to straight line



The distance between a caliper and a configured straight line will be determined.

General

- `Distance typ`

Average	The medial value between the calipered values and a reference straight line, will be determined.
Minimum	The minimum distance between the calipered values and the reference straight line is determined.
Maximum	The maximum distance between the callipered values and the reference straight line is determined.

- `Reference straight line`

This means that the selection of reference straight lines are determines either by the straight line register or by the pick-up list.

Probe see also the section called “Probe Rectangle”

- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

Parameter

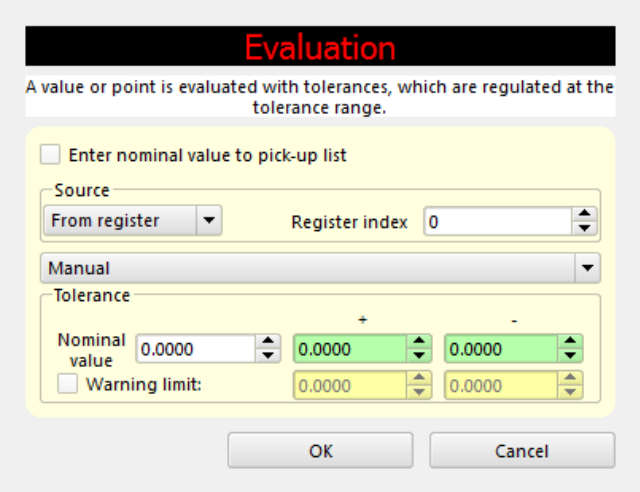
- `Probe parameter`

  see also the section called “Probe Parameter determination”

register information

A straight line is read from the register and all points from the point list. The resulting distance is written on the value register 0.

Evaluation



A value or point is evaluated with tolerances, which are regulated at the tolerance range.

- `Enter nominal value to pickup list`

- `Source`

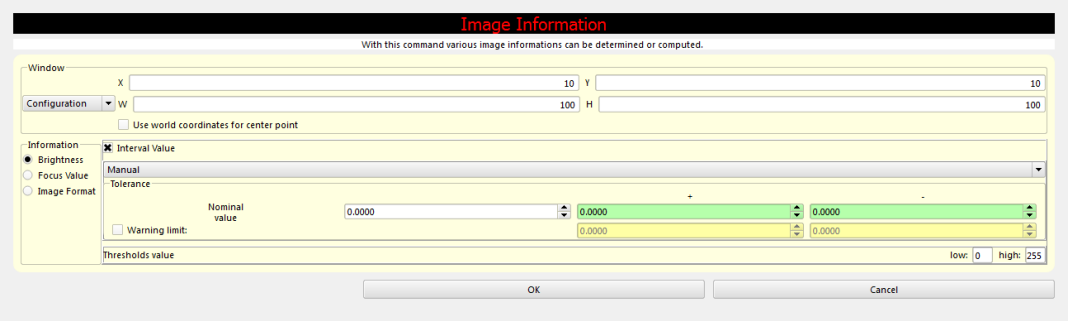
  See also the section called “Data Viewer: Register and Pickup lists”

- `Tolerances`

  For more informations, see the section called “Evaluation (Tolerances)”.

Image information

With this command various image informations can be determined or computed. Provided that a directly assessable result is achieved it can also be used for the evaluation.



- `Window`

  see also the section called “Probe Rectangle”

- `Information`

  Brigthness — This option writes the highest, lowest and the average gray value detected on the value register.

  - `Interval value:`
  - `Tolerance:` For more informations, see the section called “Evaluation (Tolerances)”.

  Focus value — The focus value for the selected range or the entire image is determined.

  - `Focus parameter:`

    Contrast threshold: This threshold is used to determine the focus value. With this threshold it can be defined which pixels are used for evaluation.

    Pixel threshold:To determine the contrast, pixels with a distance between two pixels are used.

  - `Grid distance:`

    Pixel:

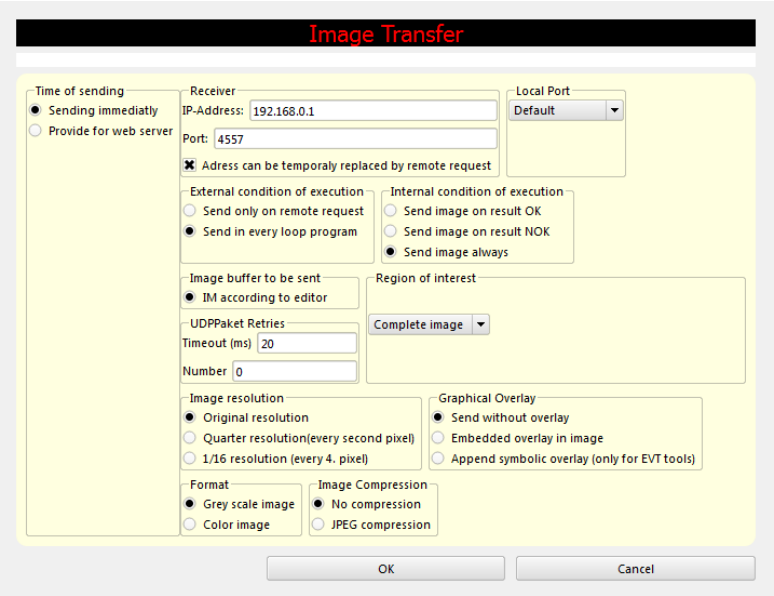
  Image format — The current image format is determined.

Register information

Value register for the option `Brigthness`

- Value register index 0: the average gray value
- Value register index 1: the lowest gray value
- Value register index 2: the highest gray value

Image transfer



Here you can send the image to any IP address .

Time of sending

Sending immediatly

- Receiver

  Type in the IP address and the port of the receiver.

- Local port

- External condition of execution

  Here you can chose if the image should be send only on remote request or if the image should be send in every program loop.

- Internal condition of execution

  You can select if the images should be sent always or on result OK or on result NOK.

- Image buffer to be send

  Here you can select which IM should be sent.

- UDP paket retries

  If a timeout occurs, you can determine if and how often it should retry to send.

- Region of interest

Complete Image	The entire image is sent.
Static Region-of-Interest	You can type in the values for the Static ROI manually or configure it on the actual image. (see also the section called “Probe Parameter determination”).

Dynamic Region-of-Interest	You can select the values for the dynamic ROI from the register or from the pick-up list.
World coordinates	The world coordinates are attached optionally.

- Image resolution

  Here you can determine the image resolution. You can chose between Original resolution, quarter resolution or 1/16 resolution.

- Graphical overlay

  You can send the image either with or without overlay.

- Format

  The image can be sent as gray scale image or as color image.

- Image compresion

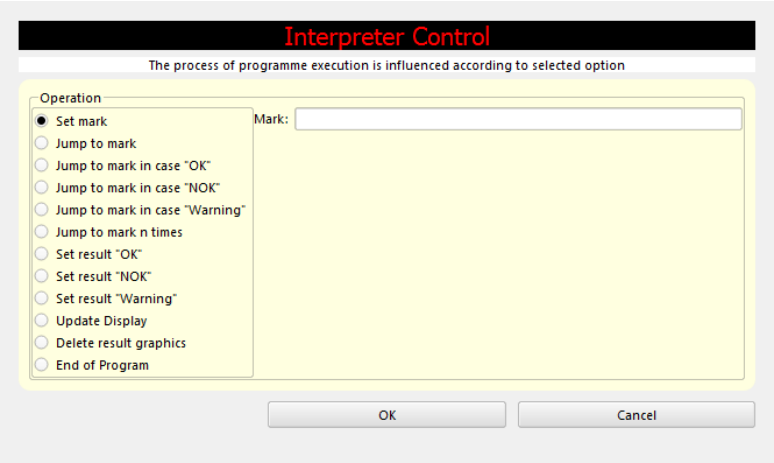
  The image can be sent either not compressed or as JPEG format.

Provide for web server

A short description of this option is already written in (send immediatly...)

- External condition of execution
- Internal condition of execution
- Image buffer to be send
- Image resolution
- Graphical overlay
- Format
- Image compresion

Interpreter control



The process sequence is controlled by the Interpreter Control. Labels can be created, which are then started by an unconditional jump (Goto) or by conditions (ifclause true than goto)..

Options

- `Set mark`

  For the new mark a not yet used name must be entered.

  Mark In this field can the name of the mark inputed.

- `Jump to mark`

  An unconditional jump to the selected mark will be executed. The by “set mark” already defined marks are available in the pull down menu.

  Mark A list of all existing mark is schown.

- `Jump to mark in case “OK”`

  If the result is “OK” a conditional jump to the selected mark will be executed.

  Mark See ().

  According to Total result
  Command
  Command, N back

- `Jump to mark in case “NOK”`

  If the result is “NOK” a conditional jump to the selected mark will be executed.

  Mark See (case ok).

  According to See (case ok)

- `Jump to mark in case “Warning”`

  If the result is “Warning” a conditional jump to the selected mark will be executed.

Mark See (case ok).

According to See (case ok)

- `Jump to mark n times`

  A conditional jump to the selected mark will be executed until the counter reaches default.

  Mark See (case ok).

  N times Enter numer of loops that will be excecuted.

- `Set result “OK”`

  The result is set on “OK”. So all previous results with “OK” will be overwritten.

- `Set result “NOK”`

  The result is set on “NOK”. This does make sense in case a negative logic is to be used.

- `Set result “Warning”`

  The result is set on “Warning”

- `Update Display`

  With this option an image can be transferred immediately from the image memory to the display memory. Since the update process can take some time for the processor at some hardware platforms, this option is useful as it ensures that an update is performed without influencing the processor time.

- `Delete result graphics`

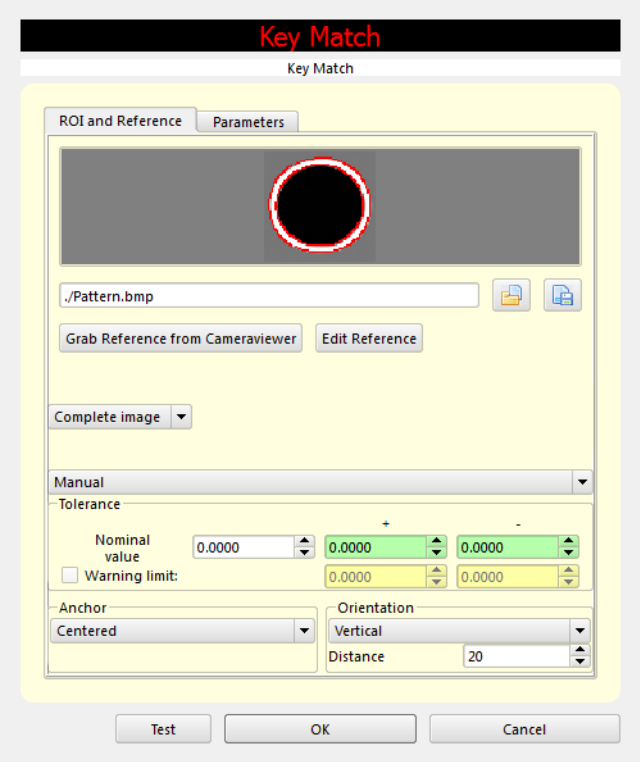
  By this option the result graphic memory will be deleted. The process will not be influenced coincidentally as it happens at a set point of time.

  From all images memories Delete every result graphic memory.

- `End of Program`

  The Interpreter will be indicated that the main program expires. So the program will again start at the beginning.

Key match



This tool finds a previously configured pattern again in the image – this does not depend on rotation and position any more.

ROI and reference

- Pattern

  In this register card you determine the source from where your reference image is loaded. You either choose directly a bitmap.

Save/Open image	Depending on the method a bitmap will be loaded.
Grab reference from Camera viewer	Alternatively such a pattern can also be created dynamically, i.e. an image will be readout of the image memory. In this image sample points will be marked in the to be tested section and so, a pattern is created. After pattern will be created in the pattern editor, it could be saved.

- Search Area (Window)

  Here you can define where the image should be found. Should the image be found in the entire image or only in a user-defined area. Either by direct input or by Configure. (see also the section called “Probe Rectangle”)

- Tolerances

  For more informations, see the section called “Evaluation (Tolerances)”.

Parameters

- Find object

Indicate the maximum number of objects, which should be found, and the minimum correlation value (between 0%....100%).

- Angle

  Here you can limit the search area for rotating objects by a maximum and a minimum angle. In this case the increment indicates the noise for the search. Recommended increment: 1.

- Speed and safety

  This mode indicates how fast the command should perform.

- Sort by

  The results of the KeyMatch command are stored in the point list. The user can determine, if the objects should sorted and how they should be sorted.

- Sort direction

  The results of the KeyMatch command can be sorted. You can choose between increasing and decreasing.

- Display

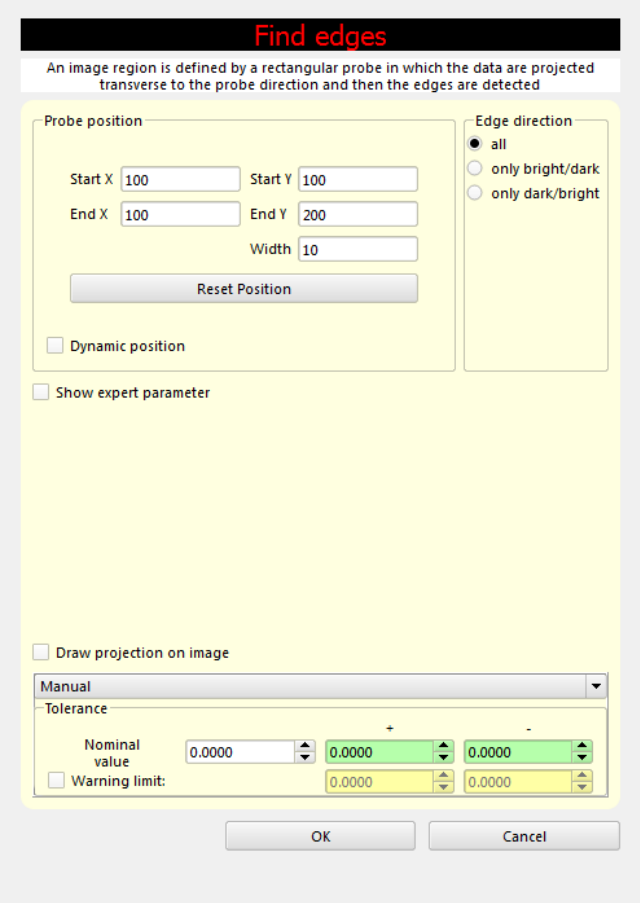
  The active values are displayed in the Camera Viewer e.g. position and angle of the object.

- Save

  By ticking the boxes and you can define where and which results should be stored. .

Register information

Edge in projection



By the point list command the point list can be influenced. It can not only be set back but also it can be stored, sorted and evaluated.

- `Probe position`

  The position of the probe is configured. See also the section called “Probe Rectangle”

- `Edge direction`

all	Show all edges in the probe area.
only bright / dark	Show only edges which change from light to dark
only dark / bright	Show only edges which change from dark to light.

- `Show expert parameter`

  With this projection type the middle value, the minimum, the maximum and the median can be determined. By the filter size, the minimum, the maximum, the middle value and the median can be determined.

Projection type	average, minimum, maximum or median
Min. gradient	Values in the projected image, where an edge is detected.

Filter size	Local neigbourhood for gradient calculation
Min. grad. diff.	Defines the minimum gradient difference for an edge. Defines the sharpness of the edges.

- `Draw projection on image`

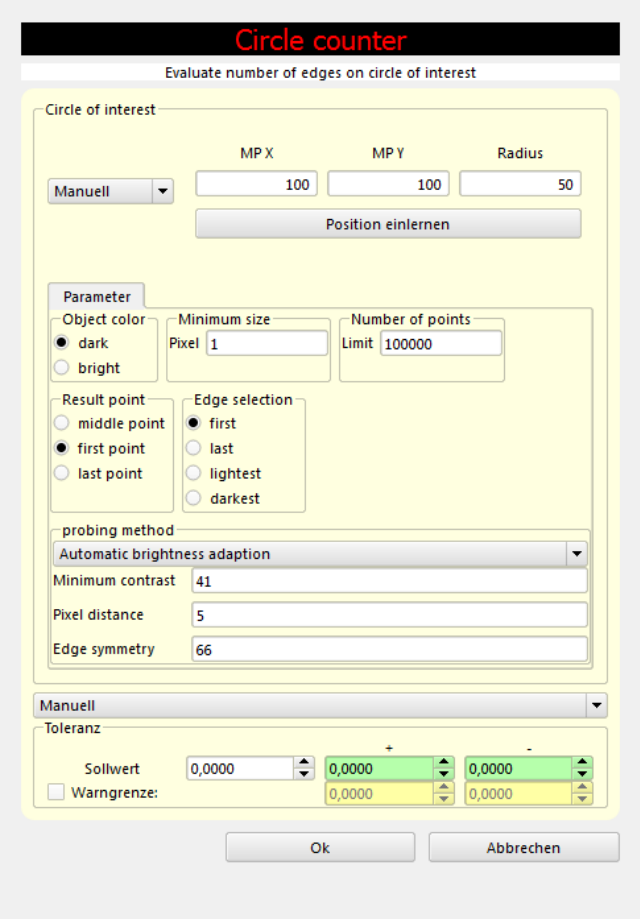
  The summation result is displayed in the image.

- `Tolerances`

  For more informations, see the section called “Evaluation (Tolerances)”.

register information

Edge counter Circle



This command is searching for the chosen changeover of edges on a configured circle and counts them. The position of the edges is stored in the point register.

Circle of interest

- `Probe position`

  Position and size of the probe are configured (manually or from pickup list), see also the section called “Probe Rectangle”

- Parameters

  See also the section called “Probe Parameter determination”

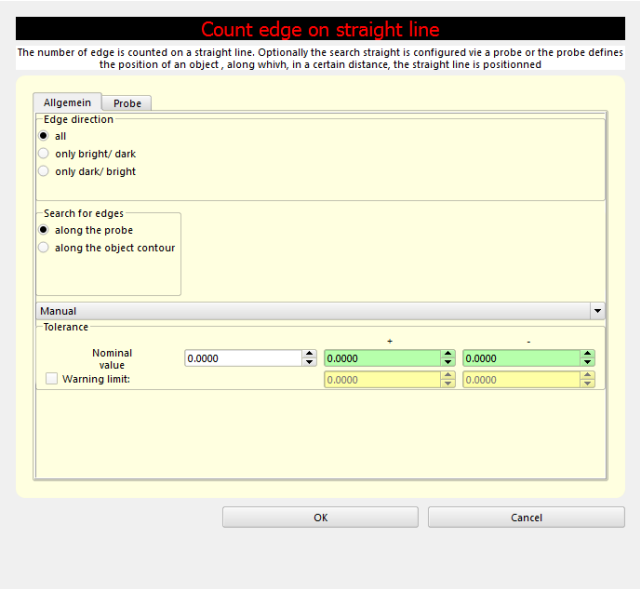
Tolerance

- Tolerances

  For more informations, see the section called “Evaluation (Tolerances)”.

register information

Edge counter line



All edges are counted along a configured probe.

General

- Edge direction

all	All edges that are changing from dark to bright and from bright to dark are detected and also counted.
only bright / dark	Only edges that are changing from bright to dark are counted and displayed.
only dark / bright	Only edges that are changing from dark to bright are counted and displayed.

- Search for edge

Along the probe	Searching for edges along the configured probe.
Along the object contour	Searching for edges along an object contour.

- Tolerance

  For more informations, see the section called “Evaluation (Tolerances)”.

Probe

- `Probe`

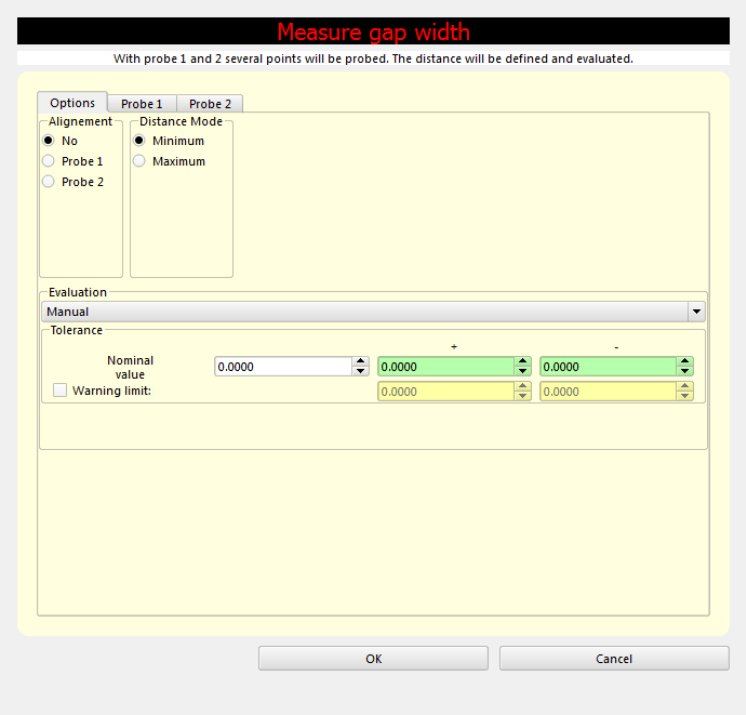
  see also the section called “Probe Parameter determination”

register information

This command writes in the following registers:

- Value register index 0: amount of detected edges
- Point register index 0: coordinates of the first detected edge
- Point list: coordinates of all detected edges.

Measure gap width



This command defines by means of two search areas, two point lists, whose distance is evaluated. The column width between the two probes is tested for the adherence to the tolerance. The probes have to be configured in such a way that there is a reasonable determination of the distance between the point lists possible.

General

- `Alignement`

  No — The probes are independent from each other.

  Probe 1 — The probes orient towards probe 1.

  Probe 2 — The probes orient towards probe 2

- `Distance mode`

  Minimum — The minimum column width is measured.

Maximum The maximum column width is measured.

- `Evaluation (Tolerance)`

  For more informations, see the section called “Evaluation (Tolerances)”.

Probe 1 / Probe 2

- `Configure position`

  Probe rectangle see also the section called “Probe Rectangle”

  Dispersion filter Determines the permitted dispersion; Standard is 1

- `Probe parameter`

  See also the section called “Probe Parameter determination”

register information

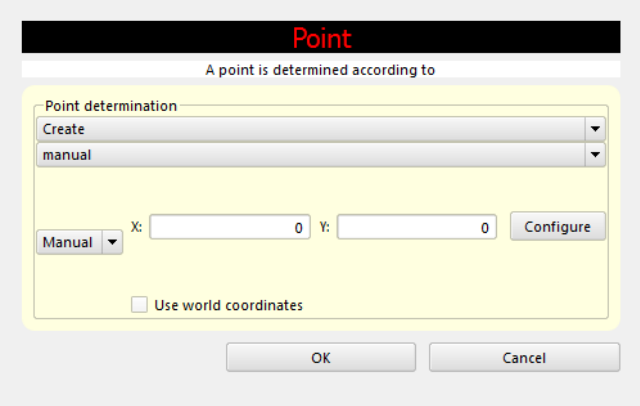
This command writes on the following register indexes:

- Value register 0: The shortest distance between detected points
- Value register 1: The longest distance between detected points
- Point register 0: The nearest point detected by probe 1
- Point register 1: The nearest point detected by probe 2
- Point register 2: The most distant point detected by probe 1
- Point register 3: The most distant point detected by probe 2
- Point list: All detected points.

Note

Value 2 through 19 and point register indexes 4 through 19 are also written by this command. These values and coordinates are used internally by the software and do not have any special significance.

Point



This command enables to define points in the image area in different ways.

- `Point determination`

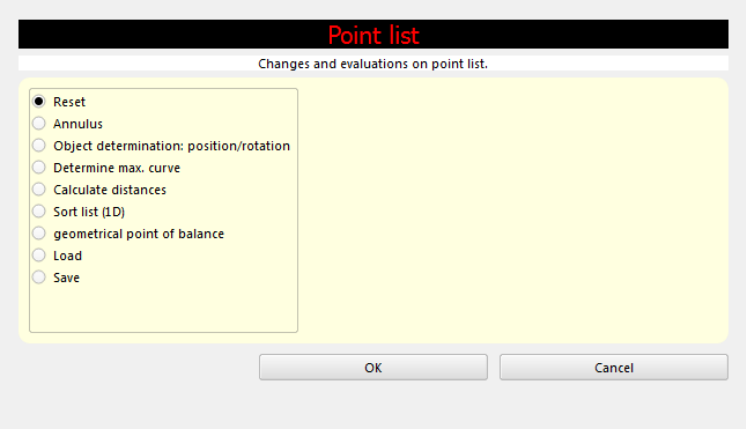
  A click on the first combo box scroll all options down.

Create manual	The point can be defined either by direct number input or by moving the cursor in the Camera Viewer (by the configuration button).
Create dynamic	The point can be created dynamically, by selecting values from the register or from the pick-up list.
Interception of two straight lines	Two straight lines are determined and the intersection point of both of them is defined as the new point.
Middle point between 2 P.	The middle point, between two selected points, is defined.
Image center	The image centre is determined as new point.
Nearest point on straight line	A point which is nearest to the straight line, is determined as new point.
Copy point on top of register	The point is put on top of the register.
From point list into point register	The point is taken out of the point list and into the register.
Transformation from image into world coordinates	The point coordinates are transformed from image coordinates into world coordinates.
Transformation from world coordinates into image coordinates	The point coordinates are transformed from world coordinates into image coordinates.
Window center	The image center point is determined.
From point register into point list	The register point is adopted into the point register.
Transform coordinates into values	The X- and the Y-value of the point is successively entered into the value register.

register information

The point's coordinates are always written on the point register index 0 and the last point list index.

Points list



By the point list command the point list can be influenced. It can not only be set back but also it can be stored, sorted and evaluated.

- `Points list operations`

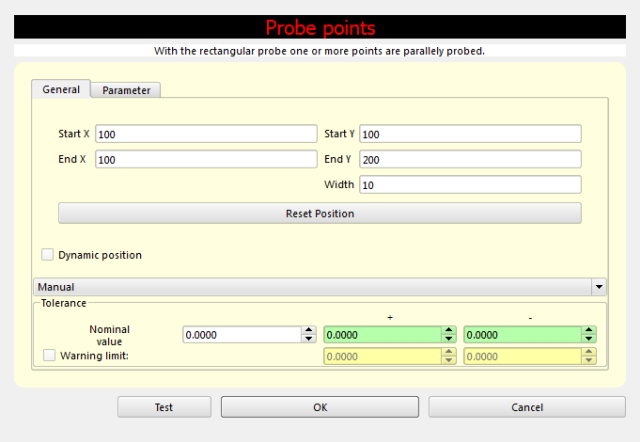
  To operate changes on the points list, the command provide various operations.

Reset	The point list is empty. That makes sense whenever the following commands evaluate the point list at a certain point of time.
Annulus valuation	If the option annulus valuation is selected, on the object, which is illustrated by the points of the point list, a ring is put. It is determined the number of points, which are situated in this ring. Number of points proportionally delivers the number as percentage, and number of points absolutely delivers the absolute number. You register the parameters for the ring into the input fields internal radius and external radius.
Object determination: position/ rotation	The option object determination: position/ rotation serves for the determination of the center of mass of the object, on which the points of the point list are. Around the object an ellipse is formed. The intersection of the axis of the ellipse and one point which is situated on the longer axis are saved in the point register. This command is useful for a dynamic calibration of the determination of the origin and the x-axis. The return values depend on the selected options.
Determine max. curve	By selection of determine max. curve the maximum curvature of the point list can be determined with a selectable existence of pixel. By the indication of a set point value with deviations the curvature can be evaluated (See the section called “Evaluation (Tolerances)”).
Calculate distances	With calculating distances you can determine distances of the points among themselves. The distances are stored in the register of values.
Sort list (1D)	With the option sort list (1D) you can sort the point list based on the situation of the individual points. Naturally this is also possible with points which are e.g. written down in a point list by the command count objects. The point list order is sorted with following keys: • Left: horizontally form left to right. The leftmost item is selected.

	• Right: horizontally form right to left. The rightmost item is selected. • Top: vertically from top to bottom. The topmost item is selected. • Down: vertically from bottom to top. The under most item is selected. The selected item is moved to the point list index 0 and so on.
Geometrical point of balance	The geometric center of all points of the point list is determined.
Load	Loads a point list from the file to the memory.
Save	Saves the point list in a file.

register information

Probe points



The probe command is one of the basis commands to extract characteristics out of an image for further processing. By a firm threshold or an adaptive threshold those points are determined which are involved in a gray transition.

General

- `Probe position`

  Position and size of the probe are configured, see also the section called “Probe Rectangle”

- `Tolerance`

  For more informations, see the section called “Evaluation (Tolerances)”.

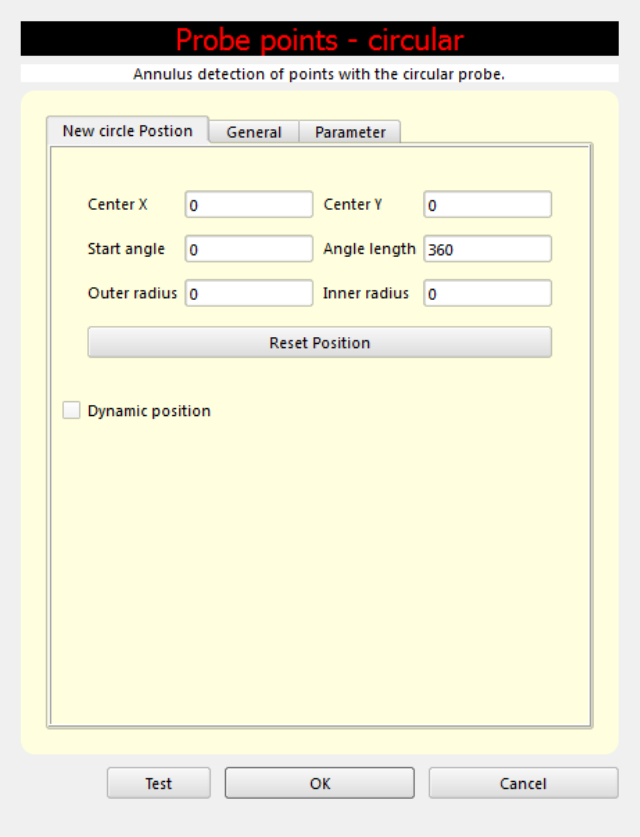
Parameter

- `Probe parameters`

  The parameter for edge determination can be specified. (see also the section called “Probe Parameter determination”

register information

Probe point (Circle probe)



In principle the circular probe works like the rectangular probe. The difference is that one has to determine an inner area and an outer area as well as an area of the circle, which should be probed.

New circle position

- `Probe position`

  Position and size of the probe are configured, see also the section called “Probe Rectangle”

General

- `Name`

  Here you can give the probe a user-defined name

- `Direction`

outwards	The search direction of the edge goes from circle center point outwards.
inwards	The search direction of the edge goes from circle center point inwards.
math. positive	The search direction runs from 0 degrees in mathematically positive direction.
math. negative	The search direction runs from 0 degrees in mathematically negative direction.

- `Tolerance`

For more informations, see the section called “Evaluation (Tolerances)”.

Parameter

- `Probe parameters`

  The parameter for edge determination can be specified. (see also the section called “Probe Parameter determination”)

register information

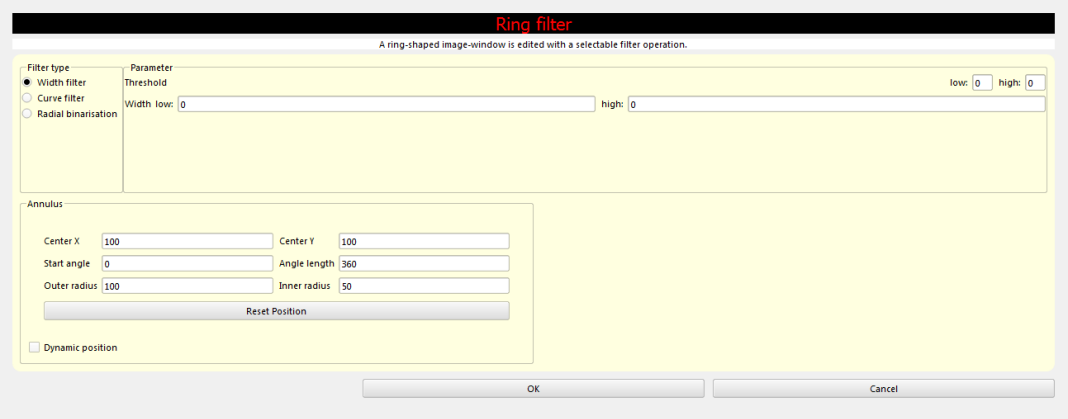
All the points detected are written in the point list.

From those detected points one point is selected depending on probe's parameters and written in the point register 0. This resulting point is the first, the last or the middle point starting from the center of the probing circle.

The amount points written in the point list is written in the value register 0.

Ring filter

With this command you can determine an angle either between two straight lines, three points or a vector between two points



Filter type

- `Width filter`

  The width filter checks the width of a ring. The width is displayed as the corresponding gray value. You must configure which gray values will be used (treshold) and how big the ring should be (Width). The segments of the ring which won't fit this configuration will be marked as to big (white) or to small (black). The whole teached in ring will be colored, not only the detected one.

  Parameter — The filter is applied to the area between the two circular rings. With the parameters the lower/ upper threshold and the minimum/ maximum height for the area between the two circular rings, is defined

  - Width

    Defines the minimum and maximum width for the ring between the two circular rings. The difference of this two will define the step size of the width color. Width color step size = 255 / (maximum - minimum).

- Threshold

  Defines the lower and upper threshold. All colors between this two values will be noted.

- `Curve filter`

  The curve filter is inspecting the configured ring on bends of the object. An object color (dark or bright) has to be determined and also if the contour is inside or outside of the configured ring. The filter will then find all segments of the ring, where the bend of the object lies inside the minimum and maximum angle (0°- 180°). Furthermore, you can specify whether the minimum and maximum angle should be inspected as absolute values or as the distance to the minimum and maximum angle.

Pixel distance	The distance between pixels can be defined.
Angle	The lowest and the highest angle can be defined.
Min. and Max. angle as	You can define whether minimum and maximum angle should be indicated as absolute value or as distance to the min./ max. bend.
Probing methode	You can choose if the brightness will be automatic adapted and some expert parameter (treshold, minimum contrast and contrast distance)
Object color	Define if the object color should be dark or bright.
Contour	You can choose between inside and outside of the ring.
Minimum size	You can enter a minimum size.

- `Radiale binarisation`

  A column vector is moved over the object. The values of the pixels inside are averaged. If the average is larger than the threshold, the pixels are colored with the entered grey value. For the binarisation enter the upper/ lower threshold as well as the color.

Threshold	Defines the lower and upper threshold.
Color	The color to color the ring if the averaged pixel value of the ring is bigger than the treshold.

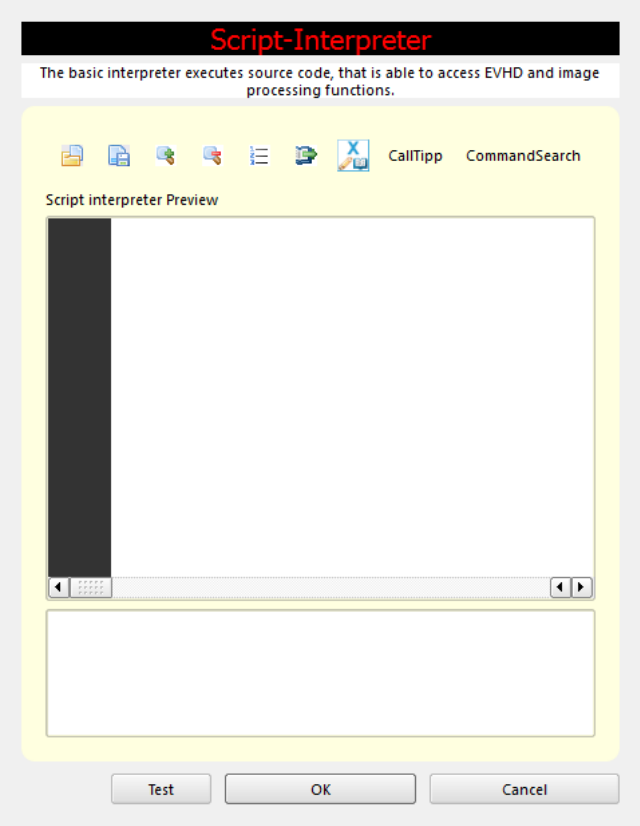
Annulus

- `Annulus element`

  For more informations, see chapter

register information

Script-Interpreter (BASIC)



The compatible VB6 BASIC interpreter is behind the Script-Interpreter. The user has access to the libraries of the EVT because of the interpreter. You can provide own image evaluation macros by calling up the numerous functions of the library. Furthermore the access to the image memory is possible with the interpreter, so that evaluations on the image data can be realized as well. You can also access to the functions of the operating system, that is why even the most different communication protocols can be realized thereby.

Select color channel

With this option you can select one of the three color levels, on which the gray scale image functions will work. If e.g. “green” is selected and then the caliper configured, the edge will be searched only in the green pixels.

- `Select color channel`

  Red — The red pixels are selected as basis for the evaluation.

  Green — The green pixels are selected as basis for the evaluation.

  Blue — The blue pixels are selected as basis for the evaluation.

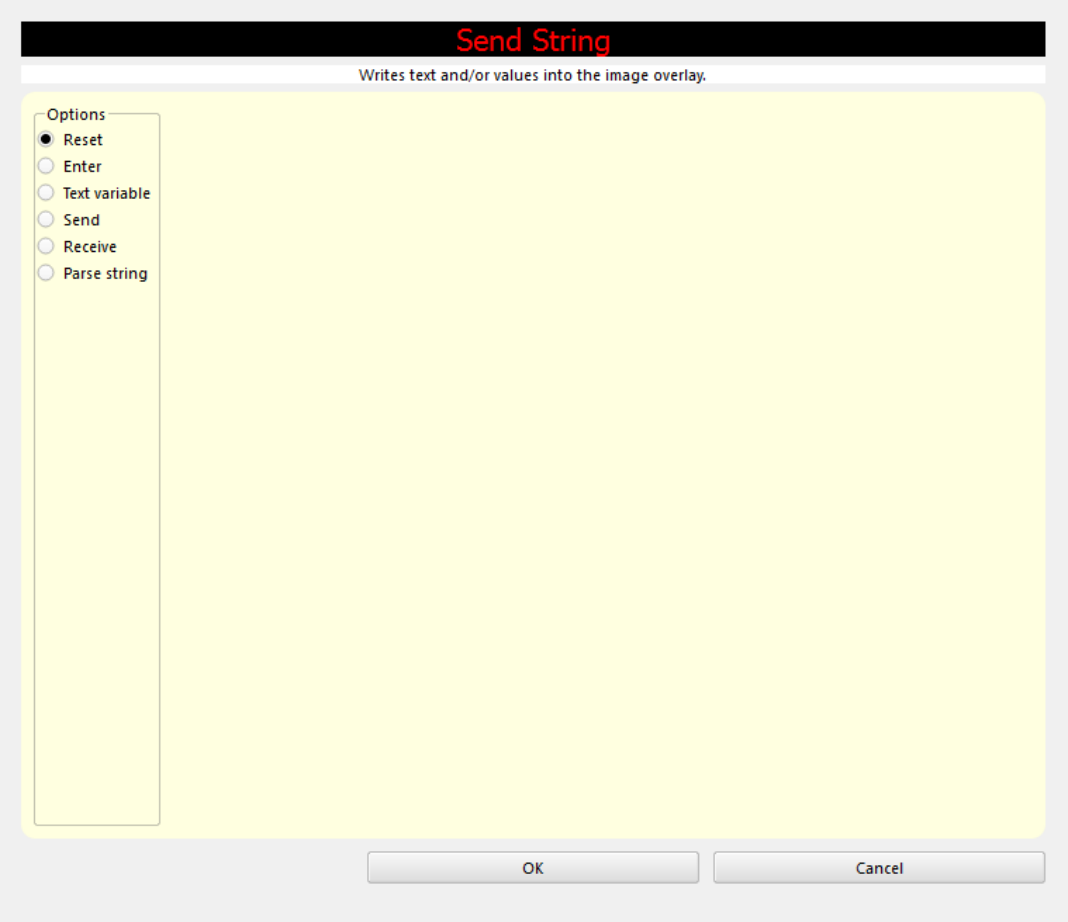
For a better understanding of the command you should get a brief idea how the color pixels are put down in the bit map memory. In a color image each color pixel is stored as a RGB (red, green, blue) value. Comparing an eight pixel color image line with a gray image line it looks as follows:

Color line [RGBRGBRGBRGBRGBRGBRGBRGB]
Gray line [GGGGGGGG]

With the color command a start value for the gray scale functions e.g. B and the offset between the color pixels is defined. Thus this function can work exactly like a gray scale image. The only difference is that the offset here is 3 and not 1 to get to the next B value.

Text string

With the command text string a global string buffer can be filled with a string, which then is transferred to the indicated target.



Options

Reset

The global string buffer is reset, afterwards there are no symbols in the buffer any more.

Enter

A text, a value, a date/time or a point are registered in the global string buffer. If there are several entries the string extends until the reset command empties the string buffer.

- Text

  Adds the given text at the end of the global string.

- Date/Time

  Adds the current date and time at the end of the global string.

- Value

  Adds a value chosen from the register at the end of the global string.

  Source Select the value from register or pickup list

Integer	The values are rounded up or down.
Floating point precision	Use own Format: Input the number of significants digits(Number of digits).

- Point

  Adds the coordinates of a point chosen from the register at the end of the global string.

Source	Select the point from register or pickup list
Floating point precision	Use own Format: Input the number of significants digits(Number of digits).

- Separator

List of separator	Space (" "); semicolon (";"); slash ("/"); underscore ("_"); hyphen ("-"); asterisk ("*"); comma (","); point ("."); tab ("->")
Separator at the start	You can choose if there is a separator at the beginning of the sting to send.

Text variable

A text variable can be chosen. Optionally the actual text string can be inserted into the text variable or the content of the text variable can be inserted into the actual text string.

- Source text variable

From text variable	Choose the variable from a pickup list which should be send.
Into text variable	Choose the variable from the pickup list where to input the string.

- Other settings

Concatenate at start	Puts the string at the start.
Concatenate at end	Puts the string at the end.
Replace	Replace the existing by the new string.

Send

The global string buffer which was filled with values by means of the option register is transferred to the selected target. Sending destination:

- In file

Destination	Direct From pickup list
String in quotes	Puts the sting into quotes (").

- COM-connection

  Selection of one of the available serial connections by means of which the string should be conveyed. COM 1-5

- Network (UDP)

  The UDP test server stored in the BIN folder of the EyeVision Software stands for the test.

IP- Address	Choose the IP-Adress of the reciever.
Port	Choose the port of the reciever.
Format	Choose between ASCII and HEX for the format.

- Network (TCP)

  You have to indicate an IP address and a network port where the files should be conveyed to via Ethernet.

Port	Choose the port of the reciever.
Type	Select if you are the server or the client.
Timeout || Close after communication	Choose how fast a timeout will occure and if you want to close after the communication.
Data format	Choose between ASCII and HEX for the format.

Note

You have to indicate the IP address of the target system where the data should be received and entered. The place where the data should be sent and received have to be identical. On the target system the same port number has to be used for reading and sending.

The installed UDP test server is translated for Windows, for other operating systems the source text is contained in C++ in the UDPTestServer.ZIP and can be translated on the favored target platform. If there are no data arriving at the target system, it must be checked if an activated FireWall prevents this. In this case, an exception for the UDPTest- Server should be established in the rules of the FireWall.

Receive

&oem.company; offers optional products for the automatic data acquisition into the standard applications of the Microsoft Office products as well as the OpenOffice products. Thus measurement and inspection results can be accepted and evaluated continuously in e.g. MS Excel or OpenOffice Calc. The reading of the values from a file is simple, because with the option Save in File it is possible to create a CSV file and therefore this files can be read automatically.

- Network (UDP)

IP- Address	Choose the IP-Adress of the sender.
Format	Choose between ASCII and HEX for the format.
Timeout	Choose how fast a timeout will occure.

- Network (TCP)

IP- Address	Choose the IP-Adress of the sender.
Port	Choose the port of the sender.
Type	Select if you are the server or the client.
Timeout || Close after communication	Choose how fast a timeout will occure and if you want to close after the communication.

Data format	Choose between ASCII and HEX for the format.

Parse String

The received string is separated into substrings according the specification set in the Format field.

- Format

  This field contains a great many of tokens, which determine the compound of the text string, e.g. “constant[CAPT] string value[3]” in the Format-field means that the text string is only excepted when it consists of three substrings, which are separated by separators. The first substring should then be “CAPT”, the second a user-defined character string and the third three-digit number.

- Separator

  A character or a character string, which is used to separate the substrings in the text string. Tip: separators inside substrings are not allowed.

- Remove separator at begin and at end

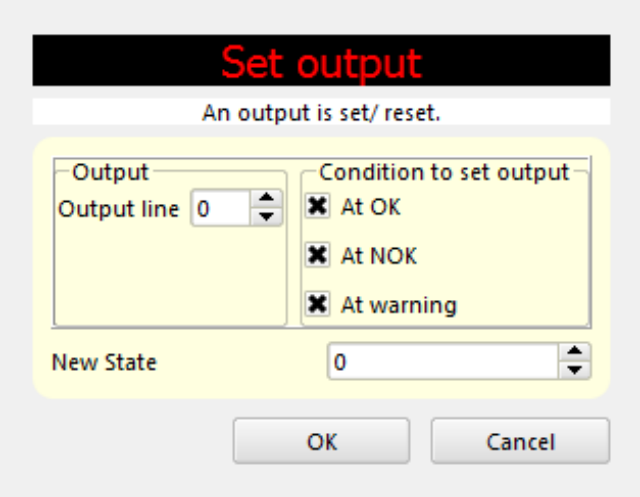
  This option determines if the separators at the beginning and at the end should be eliminated prior to the separation.

- Parser output

  As Output there are register, pickup-list, or a combination of both available.

  Register
  Pickup list
  Register + Pickup list

Set output



By the command set output all outputs that are attached to the sysem can be addressed.

- `Output`

Output line	Selection which output has to be addressed by the index 0 ... N (N = number of outputs)

- `Condition to set output`

  At OK — Output will be set if the result is OK. (For good part)

  At NOK — Output will be set if the result is NOK.(For bad part)

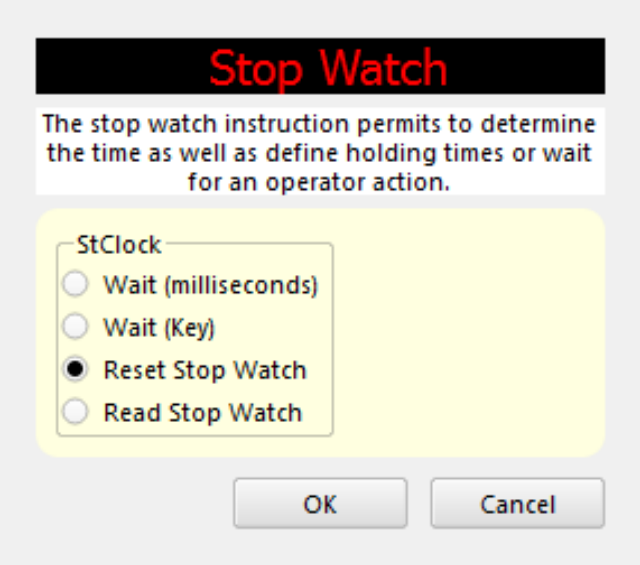
  At warning — Output will be set if the result is a warning.

- `New state`

  Input of a new state, which has to be adopted (0 or 1).

register information

Stop watch



The stop watch instruction permits to determine the time as well as define holding times or wait for an operator action.

- `Options`

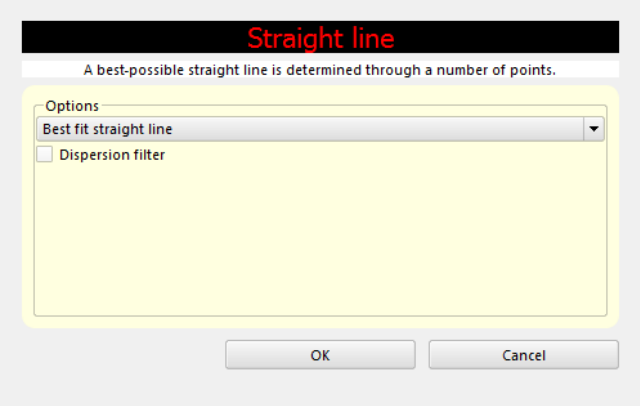
  Wait (milliseconds) — The execution of the test program stops for the adjusted value in milliseconds.

  Wait (key) — The test program stops until the user pushes a button.

  Reset stop watch — The stop watch is reset to the value 0 milliseconds and begins to run immediately.

  Read stop watch — The current value of the stop watch is read out and transferred to the value register.

Straight Line



Based on the point list or the register values a straight line will be computed.

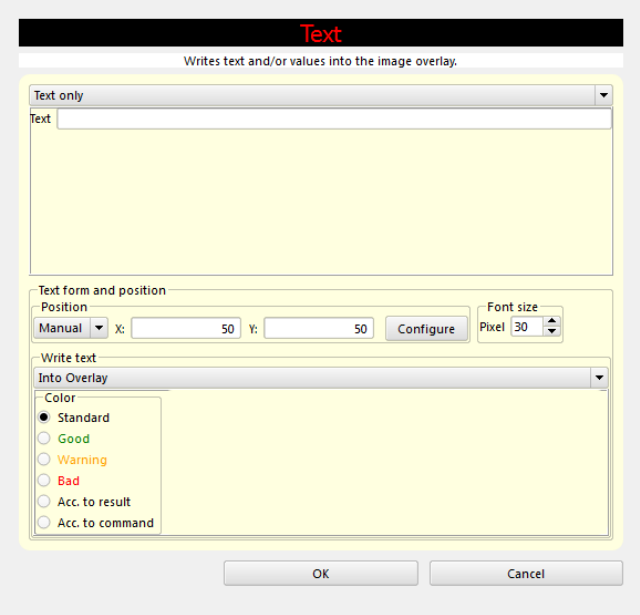
- Options

The following options are provided to define a straigth line. The reference lines are selected from a pickup list or a register.

Best fit straight line	The best straight line is based on the point list. Based on the LSR procedure, a straight line, which fits best though all points located on the point list, will be computed.
2 Points	A straight line, which fits either through two points taken from the point register or two points determined by the pick-up list, is defined.
Perpendicular line	Determines a perpendicular straight line to an existing straight line and a point.
Parallel straight line	Determines a parallel straight line to an existing straight line and a point.

register information

Text



Display of texts and values in the display range (Camera Viewer). This can be realized in different colors dependent on the hardware platform.

Typ of input

- `Only text`

  Displays only the text, which is written in the text box (white field).

  Text Here you can enter any text that should be displayed. For example when measuring an angle, click on value and type in the text “Angle”. In the Camera Viewer it is then displayed e.g. Angle: 65,00.

- `Value`

  A value of the value register is displayed.

  Order .Determine in which the value and the text have to be displayed in the camera viewer.

  Text Here you can enter any text that should be displayed. For example when measuring an angle, click on value and type in the text “Angle”. In the Camera Viewer it is then displayed e.g. Angle: 65,00.

  Source The source of Value will be selected from register or pickup list.

- `Point`

  A point of the value register is displayed.

  Order .Determine in which the value and the text have to be displayed in the camera viewer.

  Text Here you can enter any text that should be displayed. For example when measuring an angle, click on value and type in the text “Angle”. In the Camera Viewer it is then displayed e.g. Angle: 65,00.

  Source The source of the point will be selected from register or pickup list.

- `Straight line`

A straight line of the value register is displayed.

Order .Determine in which the value(s) and the text have to be displayed in the camera viewer.

Text Here you can enter any text that should be displayed. For example when measuring an angle, click on value and type in the text “Angle”. In the Camera Viewer it is then displayed e.g. Angle: 65,00.

Source The source of straight line will be selected from register or pickup list.

- `Circle`

A circle of the value register is displayed.

Order .Determine in which the value(s) and the text have to be displayed in the camera viewer.

Text Here you can enter any text that should be displayed. For example when measuring an angle, click on value and type in the text “Angle”. In the Camera Viewer it is then displayed e.g. Angle: 65,00.

Source The source of the cicle will be selected from register or pickup list.

- `Text string`

A global string buffer is displayed

Text form and position

- `Position`

The position of the text display is determined here either by direct numerical entry or by mouse click on the position of the Camera Viewer.

- `Font size`

By direct numerical input the font size of the text display can be determined.

- `Write text`

Into image The text is written down in the graphecon, for this reason the text is part of the picture.

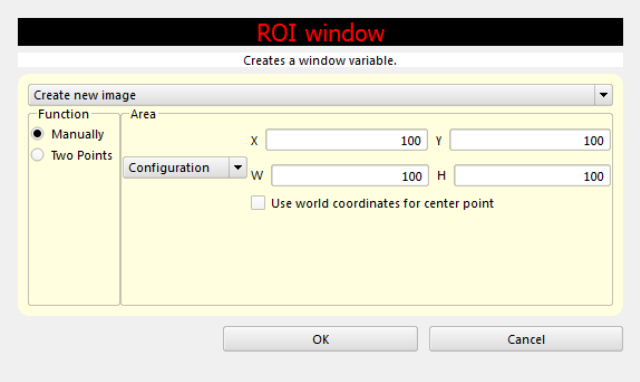
White
Black
Inverted
Grey value

into overlay The text is displayed in the overlay, if the hardware platform supports an overlay.

Standard
Good
Warning
Bad
According to result
According to command

register information

Window



This commands create a window element or change an existing window.

- `Options`

  The window could be created manually or the user can change the parameter of a window, that is saving in a pick up list.

Create new image	Two way are given to generate a new window: • *Manually*: the user can define the size the camera viewer (Configuration) or choose the whole image. • *Two points*: The first point define the upper right corner of window. The second is the bottom left corner.
Change existing window	The source window ist selected from the pickup list. The width and the heigh can be changed by input of a numeric value or by selection from register or pickup list.

register information

Chapter 8. Register informations for commands

In this chapter we describe the different commands of the EyeVision software concerning the seizure of different registers after the completion. Especially the seizure of registers with results is not evident for multifunctional commands

Command	Value register		Point register		Straight line register		Point list
Action	**Index**	**Description**	**Index**	**Description**	**Index**	**Description**	**Description**
 **Area check**							
	0	amount of detected pixels					
 **Access to variables**							
Back to register > Value	0	value from variable					
Back to register > Point			0	point from variable			
Back to register > Straight line					0	straight line from variable	
Back to register > Result	0	OK / NOK / WRN from variable					
Arithmetic > Value	0	the result					
Arithmetic > Point			0	the result			
 **Angle**							
	0	angle					
 **Caliper – hor./vert.**							
	0	distance	0	top-most point			
			1	right-most point			
			2	bottom-most point			
			3	left-most point			
 **Calliper free**							

Command	Value register		Point register		Straight line register		Point list
Action	**Index**	**Description**	**Index**	**Description**	**Index**	**Description**	**Description**
	0	distance					
 **Circle**							
	0	number of edges	0	first edge coord.			all edges
 **Color filter**							
	0	amount of detected pixels					
 **Color intensity**							
	0	amount of detected pixels					
	1	hue, average					
	2	hue, minimum					
	3	hue, maximum					
	4	saturation, average					
	5	saturation, minimum					
	6	saturation, maximum					
	7	brightness, average					
	8	brightness, minimum					
	9	brightness, maximum					
 **Contour check**							
	0	amount of points not detected					
 **Contour tracing**							
	0	amount of detected points					point coordinates
 **Count objects**							

Command	Value register		Point register		Straight line register		Point list
Action	**Index**	**Description**	**Index**	**Description**	**Index**	**Description**	**Description**
	0	amount of objects					
Save in point register			0-N point(s) of balance				
Save in point list							point(s) of balance
Cross section							
	0	distance	0-1	measurement points			
Distance							
	0	distance					
Edge counter line							
	0	number of edges	0	first edge coordinate		all edges	0
Image capture							
Load cyclic / save cyclic	0	index number of image					
Image information							
Brightness	0	average gray value					
	1	min. gray value					
	2	max. gray value					
Focus value	0	sharpness of image					
Image format	0	image horizontal resolution					
	1	image vertical resolution					
INI-file access							
Read INI-file > value	0	value from file					

Command	Value register		Point register		Straight line register		Point list
Action	**Index**	**Description**	**Index**	**Description**	**Index**	**Description**	**Description**
Read INI-file > point			0	value from file			
 **Measure gap width**							
	0	shortest distance	0	probe 1 shortest			all points
	1	longest distance	1	probe 2 shortest			
			2	probe 1 longest			
			3	probe 2 longest			
 **Point**							
	0	point coordinates					point coordinates
 **Points list**							
Reset							reset all to zero
Annulus value	0	amount of points found					
Object determination			0	point of balance			
			1	side of rectangle			
			2	1 . corner of rectangle			
			3	2. corner of rectangle			
			4	3. corner of rectangle			
			5	4. corner of rectangle			
Calculate distances	0	average distance					
	1	shortest distance					
	2	greatest distance					
Sort list							sorts all points
Geometrical center			0	point coordinates			
Load							points from file

Command	Value register		Point register		Straight line register		Point list
Action	**Index**	**Description**	**Index**	**Description**	**Index**	**Description**	**Description**
 **Probe points**							
	0	amount of detected points	0	First /last / middle point			
 **Probe point (Circle probe)**							
	0	amount of detected points	0	First / last / middle point			
 **Straight Line**							
					0	straight line coordinates	
 **Straight line distance**							
	0	distance					
 **Stop watch**							
	0	time in milliseconds					

Chapter 9. Implementation concepts

The aim of the following chapter is to demonstrate how to create robust solutions for the image processing in an easy and fast way by means of the functionality that are included in the EyeVision image processing software. Taking everything into account, it is very important to ascertain the basic parameters. For an efficient image processing solution, it has to abide by these parameters. It sounds trivial but the successful implementation of an image processing solution depends on these parameters.

A very important aspect of a successful implementation of an image processing topic is that approx. 30% - 40% of a successful solution depends on the best choice and adjustment of the optic and illumination. Then further components follow at which the software also plays an important role.

For this reason various tools, which allow the evaluation of the quality of the illumination and the optics, are integrated into the EyeVision software. Only if it is ensured by the means of these tools that the characteristics of the illumination and the optics are elaborated optimally, you should think about the analysis algorithm. The analysis algorithm recognizes or detects these characteristics as a basis for good or bad assertions which can be specified in the end.

The preliminary considerations of the image processing solution

Special circumstances

At requests for the feasibility concerning the image processing solution, the testing part always comes first. All analysis are executed in the laboratory and thereupon the choice of components is made.

Unfortunately it is often forgotten that the possibilities of the optics and illumination are limited. Therefore the first step is to find out if there is an arrangement by which it is possible to elaborate the characteristics which should be examined. Afterwards it should be examined immediately, whether this arrangement can be put in the test range. We would therefore recommend that the spacial circumstances are checked first in order to find out which arrangement for the optics and the illumination are possible.

If it is not possible to realize a transmitted light illumination e.g. because of the workholding device, specific measured values, which determine such a lighting arrangement, are required. Further analysis are then not necessary any more because the machine will never meet these requirements.

Another example for the spacial difficulties is, when having to establish a telecentric objective for a topic. The component is 20 mm in length but the space for the installation is non-essentially larger. In this case it is not possible to situate a lens. Therefore all analysis are unnecessary except a solution with the client, how to situate the components, is found.

Environmental conditions

We already addressed the problems concerning the environmental conditions before. These difficulties should under no condition, be ignored. Mostly it is concerning criteria which can not be changed. These criteria often influence the optic as well as the illumination.

Not only the optic and the lighting are influenced by the environmental conditions but also the components.

If it is about a precise optical measuring system the basic conditions are important to comprehend the measured values. If it is measured for example in micrometers the heat expansion of the work-piece and of the machine component

play a measurable role because there the image processing components are attached. Their linear extensions can be measured by means of image processing. If this heat expansion is not considered then a mistake will occur at the end, which in reality is a mistake caused by the environment or by a suboptimal construction.

The dust particles caused by the process also belongs to the environmental conditions. But mostly they are not existing in the sample parts which are allocated by the solution provider. The reason is that they disappear mostly during the transport or the storing. Oil drops are a typical example, they can always appear in a machine because of the process of machining. But they disappear completely when the sample part is sent. But contrasts are part of the image processing evaluation and the oil drop causes also a contrast which cannot be differentiated at all from the part. This leads to false results of measurement. Naturally the employer does not accept missing parts based on such oil drops and problems are bound to happen.

It is advisable to test the process for possible weak spots and to define what to do from the very first or whether these faults can be eliminated by a technical installation already before the check.

Test speed (process speed)

The test speed rarely causes problems nowadays. Because the computer science is developing so that most implementations now can be solved.

If the components are moving, the question is how to assimilate the component to illustrate it in the motion optimally so that on this basis the evaluations are able to take place. The maximally attainable accuracy of measurements depends directly on it.

In this context there are two different processes: the continuous process and the discrete process. For both the same problem is essential: it is necessary to indicate individual characteristics as sharp as possible. For the image processing for fast moving processes two solutions are available, which are also known from the photographic field. Either the picture is frozen by a flash or by a fast shutter. Modern sensors posses electronic catches, therefore the attainable shutter speeds are as fast as those of the flashes. It depends on the user which procedure is preferred.

The testing components

Concerning the testing components you should exercise caution, not that they are already “aged”, because the image processing orients itself always on the contrast characteristics of the testing parts.

Often the characteristics do not show directly after the production. Only during the storing the material changes in either a positive or negative way. If the characteristic, which should be recognized, is changing after a longer storage time, so that the characteristic can be detected only with difficulties. In this case it is a negative change. Then much time is spend to emerge the characteristic, although it has a highcontrast during the production.

It would be also very negative if the characteristic which should be recognized can be detected very simply but during the production you cannot recognize it at all or only with difficulties. The reason could be e.g. that the defect characteristics react with the atmospheric oxygen and therefore a contrast emerges. Suddenly a topic that was classified as very easy becomes not realizable.

Another problem emerges if the testing plant is used to control the incoming goods. In this case it is possible that a testing part which was allocated to define the feasibility could now have a completely different colouring e.g. by another surface texture. This inevitably leads to the case that the testing plant generates an increasing pseudo error. Therefore the variance of the testing parts should be defined and clarified exactly before.

Optics and illumination

The optics and illumination are essential factors, which also determine the way the process of an image processing machine works.

Unfortunately there is no rule of thumb for the appropriate optics. The intercept theorems are the first choice to determine the parameters for the wanted lens. In most cases it would be worth the trouble to consult an expert, because concerning lenses various parameters have to be taken into consideration.

If an assortment of lenses is available, it is advisable to test the lenses which come into consideration. By the way the test is also advisable for lenses which are calculated by an expert.

Concerning fast processes, which are photographed with a short shutter speed or short flash speed, the light intensity of the lenses should be checked.

Telecentric lenses	Concerning extremely precise measurement, these lenses are always advisable. The disadvantage of these lenses is mainly the price and the huge mounting form.

In resent years the illumination developed. The illumination with spiral-wound filament lamps decreased because of the more efficient LED's. Today the spiral-wound filament lamps are still used, where very high light power is demanded, because the light power of such a lamp is still far away from the power the LED has.

The disadvantage of this illumination is the short durability and the fast ageing process. Consequently the LED illumination is mostly preferred because of the durability and the robustness.

Another advantage of the LED illumination is the flexibility concerning the design. The LED illumination can be adjusted to almost each form. Solely a suitable plate has to be produced which serves as a carrier for several illuminant.

It is advisable to search for these illuminations on the Internet in order to get an impression of the different forms of LED illumination. If you enter LED in connection with „machine vision“ several hits are shown. Naturally you find also a well proven assortment of LED illuminations in all possible designs at EVT . The production of custom illumination is only one part of our assignment therefore we recommend you to visit our homepage.

Generating a test program

In this chapter we would like to show you how to generate a test program with the EyeVision. It is no problem to accomplish all program steps even if you possess only a demo version of the software. The image data bank which is integrated in EyeVision even makes possible to illustrate a realistic program scenario without a camera.

Different image data banks are installed automatically with the program installation. They permit to simulate the programming of realistic component parts.

We think you have already dealt with the most important structures of the software, therefore we do not describe all steps in detail any more , e.g. where you can find which file.

The file PC local in the device listing of the EyeVision software is optimally for sample programs, as well as their corresponding image data banks, sample files or character sets. Below you can find all described test programs and the corresponding pictures.

The test programs which are directly connected to this manual always begin with the prefix „Man “. If it concerns a more detailed example, then the letters Bsp are added on the above-named prefix, e.g. ManBspXxxxx.ckp. This would be the sample program Xxxxx from this manual.

Examples

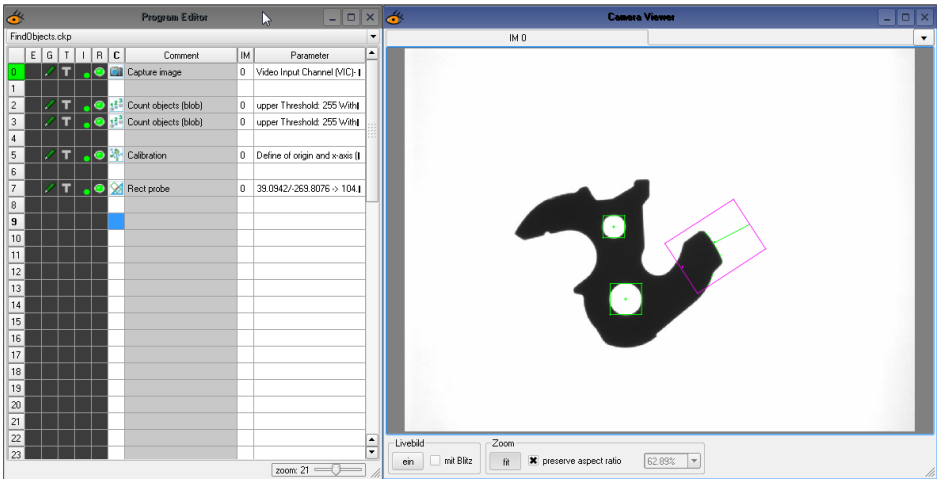
In this chapter you can work with inspection programs, which were saved by EVT. Do not be surprised, if your image in the Camera Viewer differs from the image of our screen shots.

Also it frequently happens, that you have to load the image once more by yourself. So, if you load an inspection program and no image shows up in the Camera Viewer, click onto the image capture icon in the program editor. In the field data you then can see the path of the image. Click on load data and select precisely the image, which is displayed in the data window. Confirm by clicking OK. Click onto the white field beneath the image capture icon and then the image should be shown and the inspection program should be running correctly.

Find objects and count holes.

The following example shows how to set the commands for detecting the same part on the identical object in various position. The object is used for fixing the coordinate-system. Both hole of the dark objects are detected, than they are used as coordinate system. A probe is set to found a point on the object. If the position of object is changed.

Figure 9.1. Counts objects



By means of the image capture you can choose the pictures from the file Image of the device PC local. (here Training_1_00000.jpg) is shown by loading a fixed image

The object counter is configured for the whole picture to find the smallest bright hole of the object.

The object counter is configured for the whole picture to find the biggest bright hole of the object.

Both points of the object of the object counter is used as coordinate system(Option: Origin and orientation) with the rotational direction CCW. Now the coordinate system for the following system is formed by the two bright points (or holes) of the dark object.

The rectangular probe can be set to find a edge. This probe will always test the same part of the object, if you select this object in various position with this example.

Chapter 10. Appendix

Edge-detection method of the EyeVision Software

Note

The following option is currently not available on the GUI

Depending on the graphical material, one of the four kinds of probe method should be selected.

Binary edge-detection (Method: Binary)

This method calculates, by means of linear interpolation, the position of the edge between the two pixels, where the threshold value is above or below target. The binary edge-detection is the fastest as well as the most exact method. However, the contrast conditions in the image have to be very steady, because the binary edge-detection works with a fixed threshold value. This means that a possible brightness offset leads to measuring errors. An example of use for the method is a metal part in a transmitted light.

Contrast edge-detection (Method: Minimum contrast)

That is the standard procedure in EyeVision , if the option **automatic brightness adaption** is activated.

This method does not analyze the absolute gray values, but compares the differences of the gray values. Therefore it is irrelevant if the edge ranges from gray value 0 to 100 or from gray value 50 to 150. The operation searches for a gray value sequence “plateau increase respectively plateau decrease”.

- Calculation of the nominal gray value:

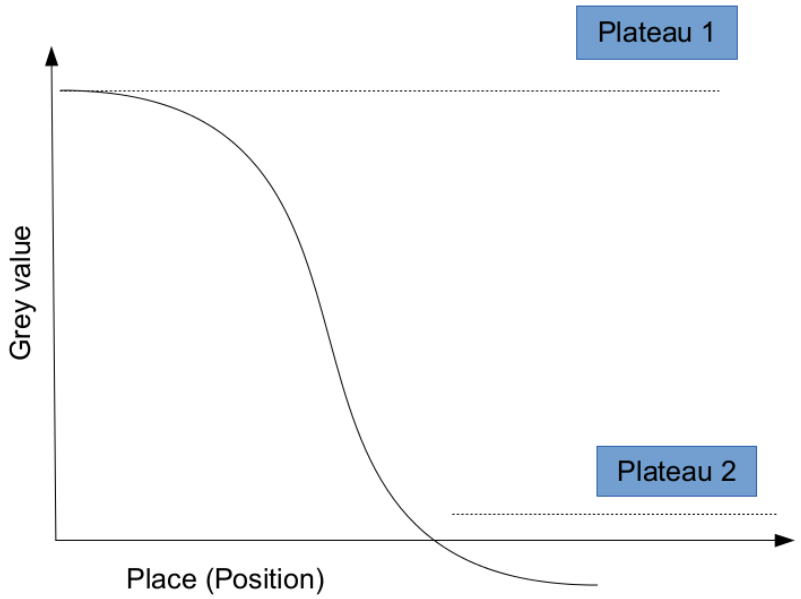
  Calculation of the nominal gray value: If the value 66% is set as edge symmetry, the set greyvalue = greyvalue of the first plateau +/- 66% of the greyvalue difference between plateau 1 and 2 (depending on the edge direction).

- Calculation of location:

  Linear interpolation between the pixel, where the greyvalue transition occurs (see drawing). This procedure is used where the binary procedure fails, that is to say that it is possible that the basic brightness varies. (example: knotholes in wood, the background can show strong differences in brightness).

Note

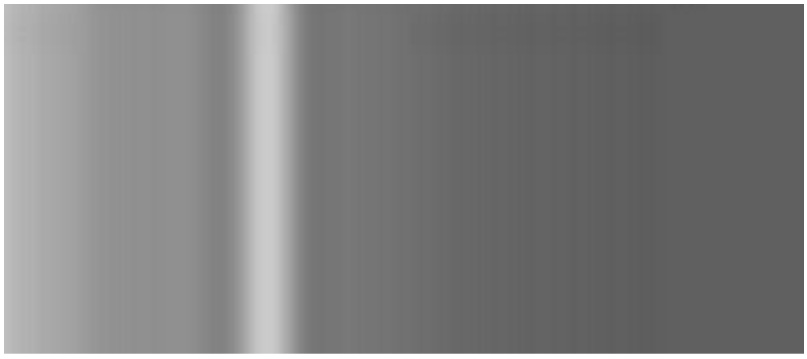
The both above mentioned procedures are used for most inspection tasks concerning the edge detection.



Contrast edge-detection (Method: Minimum contrast 2)

If the option **automatic brightness adaption** is activated in the detection command in EyeVision, this method correspond to the procedure **interference filter based on threshold value**.

This command works similar to the previous method. Although interferences are not determined according to its size but according to its greyvalue. At the detection direction bright-dark edges are rejected where interferences in plateau 2 occur which are brighter than plateau 1 minus minimum contrast. At the detection-direction dark-bright edges are rejected where interferences in plateau 2 occur which are darker than plateau 1 plus minimum contrast. The following synthetic image is used as example:



You can determine a bright-dark transition which shows a short, bright interference. The described method does not detect the interference as an edge.

Contrast edge-detection (Method: minimum contrast 3)

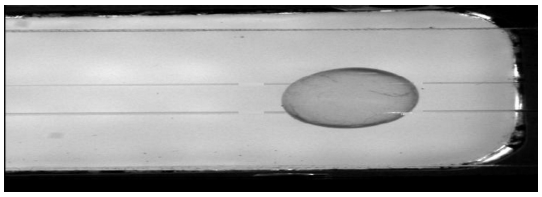
In EyeVision procedure **measuring threshold** if the option **automatic brightness adaption** is activated in the detection command

At some edge gradients the above mentioned two methods cannot find a second plateau, because the contrast conditions are met also after the edge, that should be detected (i.e. the difference in gray value from x and x+pixel distance is bigger than the minimum contrast). Two examples:

Figure 10.1. First picture

The image shows a dark-bright transition which is not followed by a plateau but a protracted rising ramp. The two methods described above cannot detect an edge. Another example out of experience: At the edges of the capsule is a good bright-dark transition which is followed by a longer dark-bright ramp.

Figure 10.2. Second picture



Active X interface

Note

The following option is currently not available on the GUI

The EyeVision software can also be integrated into your own software products. That is the reason why the software is provided with an ActiveX interface of Microsoft. For details consult the manual of EyeVision ActiveX.

File extensions

Table 10.1. Files extensions

Extension	Typ	Description
*.ckp Check Program	Check Program	The EyeVision inspection program is saved automatically with this file extension.
*.cgh	OCR sample file	In this data the samples for the optimal character reading, are filed.

Extension	Typ	Description
*.sbs	BASIC	Exported BASIC source text
*.pat	Correlation	Correlation sample
*.bmp	Image	An image of the Microsoft bitmap format.
*txt	Data	Various functions save the data as a free formatted ASCII text
*.ini	Structural text	Data, which are based on the Microsoft Windows INI file, are constituted and read.