Manual

Toolmonitor Vision

Softline

Modline

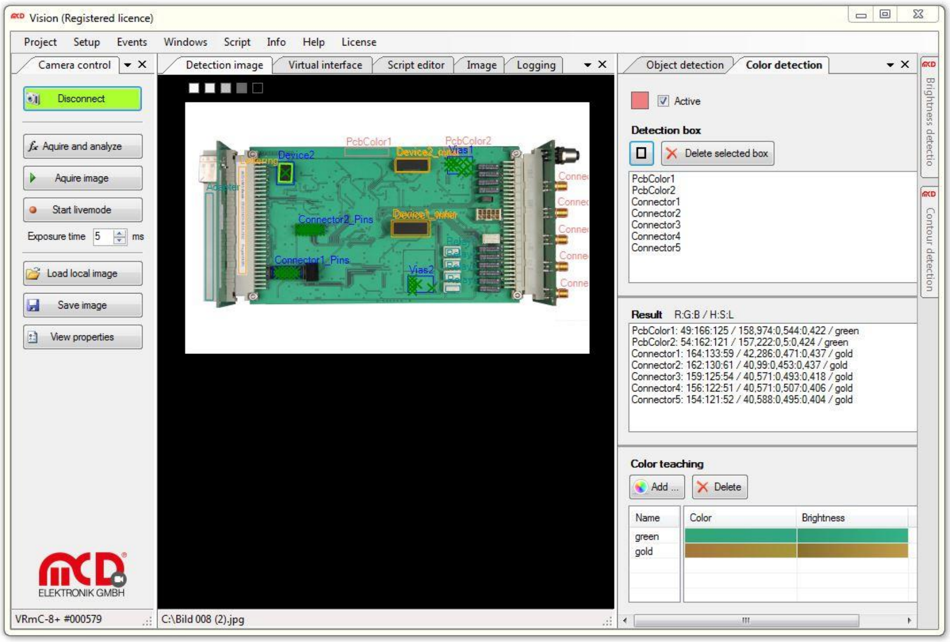
Conline

Boardline

Avidline

Pixline

Application



☑ Active

Detection box

PcbColor1
PcbColor2
Connector1
Connector2
Connector3
Connector4
Connector5

Result R:G:B / H:S:L

```
PcbColor1: 49:166:125 / 158,974:0,544:0,422 / green
PcbColor2: 54:162:121 / 157,222:0,5:0,424 / green
Connector1: 164:133:59 / 42,286:0,471:0,437 / gold
Connector2: 162:130:61 / 40,99:0,453:0,437 / gold
Connector3: 159:125:54 / 40,571:0,493:0,418 / gold
Connector4: 156:122:51 / 40,571:0,507:0,406 / gold
Connector5: 154:121:52 / 40,588:0,495:0,404 / gold
```

Color teaching

Name	Color	Brightness
green	🟩	🟩
gold	▨	▨

MCD Elektronik GmbH
Hoheneichstr. 52
75217 Birkenfeld / Germany
Fon +49 (0) 72 31/78 405-0
Fax +49 (0) 72 31/78 405-10
info@mcd-elektronik.de
www.mcd-elektronik.de
HQ : Birkenfeld
Managing CEO : Bruno Hörter
Register Court Mannheim
HRB 505692

Measurement Technology beyond the decimal point

Templates version: 2.0 / 2015-02-06
V1.0 2016-01-12

Table of Contents

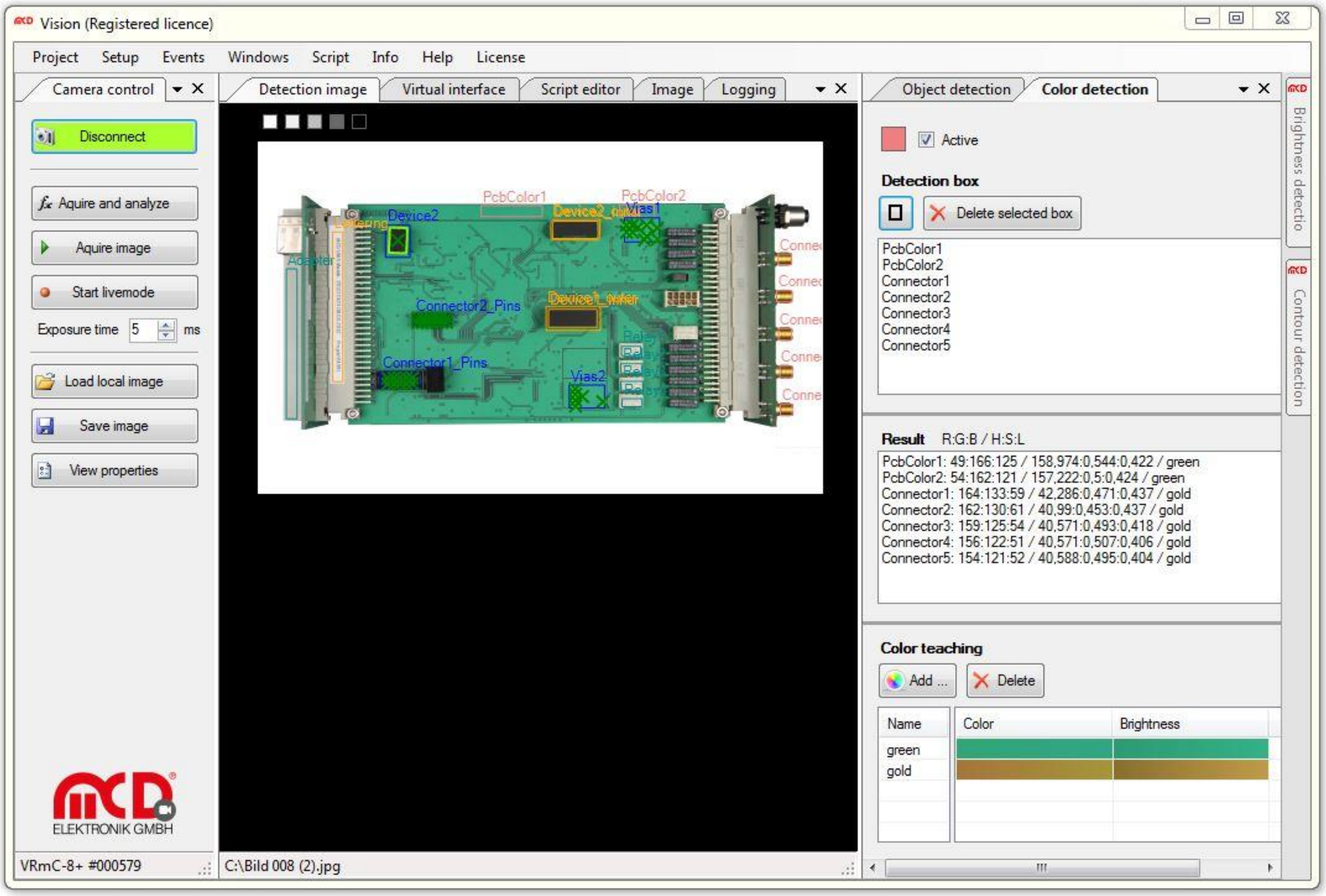
1. General

The main areas of this documentation are **Functions** and **Programming**. The descriptions of the individual software modules are available under **Functions**. The interfaces for the software user are listed in tabular form under **Programming**. The functions for Toolmonitor access are documented in the interface. The applicable handover parameters for these functions are provided in the respective modules.

1.1. Software Features

Industrial image processing tasks can be solved easily and cost-efficiently using the **Toolmonitor Vision**. The following tools are available:

- **Contour detection** - Number of contours for simple pattern recognition
- **Brightness detection** - 8-bit gray levels
- **Object detection** - Position and area in the search area
- **Color detection** - RGB, HSL, or taught colors



PcbColor1
PcbColor2
Connector1
Connector2
Connector3
Connector4
Connector5

Result R:G:B / H:S:L

PcbColor1: 49:166:125 / 158,974:0,544:0,422 / green
PcbColor2: 54:162:121 / 157,222:0,5:0,424 / green
Connector1: 164:133:59 / 42,286:0,471:0,437 / gold
Connector2: 162:130:61 / 40,99:0,453:0,437 / gold
Connector3: 159:125:54 / 40,571:0,493:0,418 / gold
Connector4: 156:122:51 / 40,571:0,507:0,406 / gold
Connector5: 154:121:52 / 40,588:0,495:0,404 / gold

Name	Color	Brightness
green	🟩	🟩
gold	🟫	🟫

The **Toolmonitor Vision** software enables the creation of different search areas, which can be placed and resized using a **graphical user interface**. The hardware interface supports USB cameras made by VRmagic. USB cameras of the VRmC + PRO design series are supported as color and monochrome cameras with different resolutions. It is important to select a camera from the + series, as they feature the required software trigger and a small form factor.

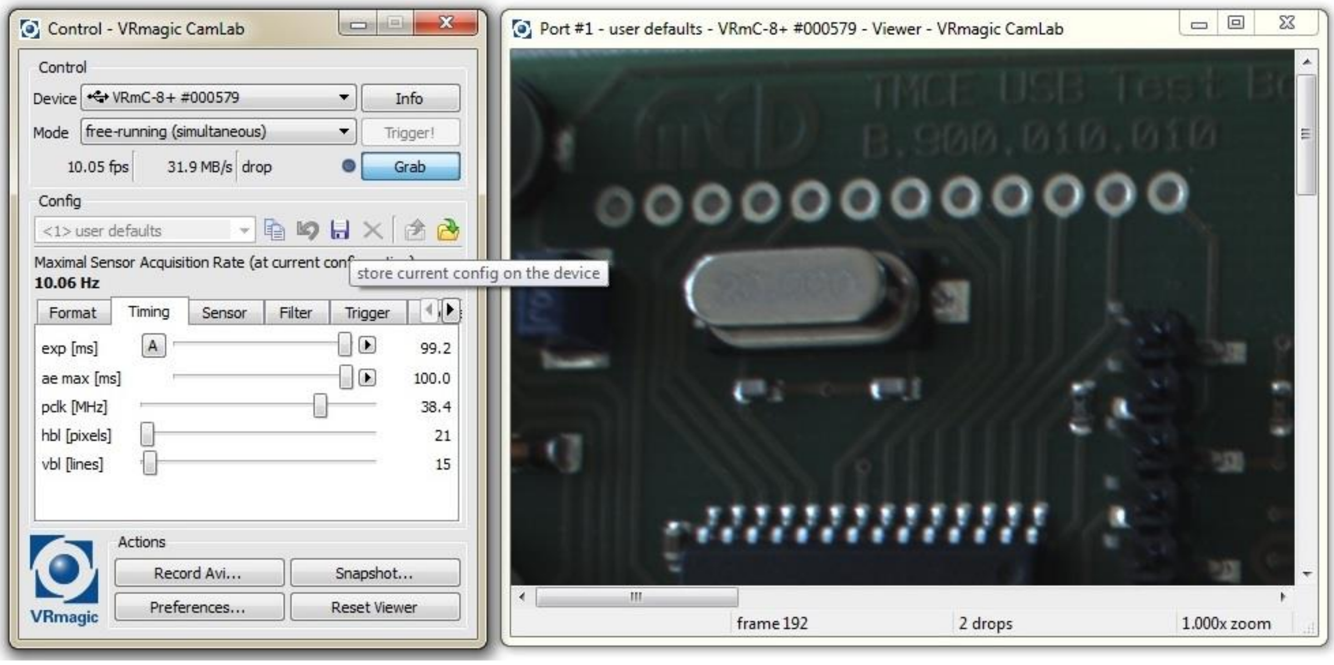
The **program interface** can be **freely arranged** to a large extent and adapted to the user requirements. Configurations created once can be saved in project files and reloaded as needed. Example projects can be found in the installation files.

All measurements and settings can be automatically performed using an **integrated script engine**. In connection with the **visualization module**, custom user interface can be created as well.

The Toolmonitor Vision can be fully **remote-controlled** using third-party software. COM/DCOM or a .Net – Assembly is used as **interface**. This way, the Toolmonitor can be integrated into a variety of applications (Microsoft Visual Studio ® (C#, C++, Visual Basic), Microsoft Office ® (e.g., Excel ®), Open Office ®, LabView ®, MCD TestManager CE, Standard Windows DLL).

1.2. Driver Installation for VRmagic Cameras

The **device driver** must be installed in order to integrate a VRmagic USB camera. This driver can be downloaded and installed from the VRmagic homepage using the VRmUsbCam DevKit. The current version (VRmUsbCam DevKit for Windows X86 x.x package) can be downloaded from http://www.vrmagic.com/de/imaging/downloads/. This version also contains the CamLab software, which can be used to parameterize the connected camera. All settings stored in the default setup are automatically loaded into Toolmonitor Vision.



Control - VRmagic CamLab

Control

Device: VRmC-8+ #000579

Mode: free-running (simultaneous)

10.05 fps | 31.9 MB/s | drop

Config

<1> user defaults

Maximal Sensor Acquisition Rate (at current conf

10.06 Hz

store current config on the device

exp [ms]	99.2
ae max [ms]	100.0
pclk [MHz]	38.4
hbl [pixels]	21
vbl [lines]	15

Port #1 - user defaults - VRmC-8+ #000579 - Viewer - VRmagic CamLab

frame 192 | 2 drops | 1.000x zoom

2. Installation of software, drivers

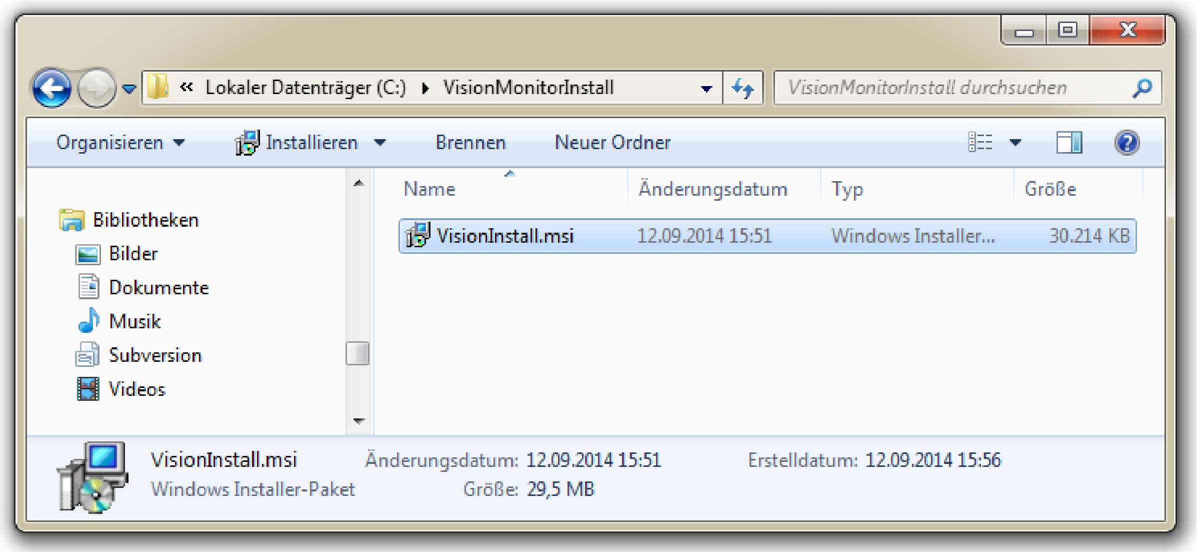
2.1. System requirements

- Operating Systems: Windows2000®, Windows XP®, Windows7®
- Architecture: 32- or 64-bit
- .Net Framework from version 3.5

2.2. Installing Software and Drivers

2.2.1. Installing MCD Software Toolmonitor Vision

The VisionMonitorInstall.msi installation program should be executed to install the software for the Toolmonitor Vision. To do this, open the Windows® file explorer and start the USBHubInstall.msi installation program.



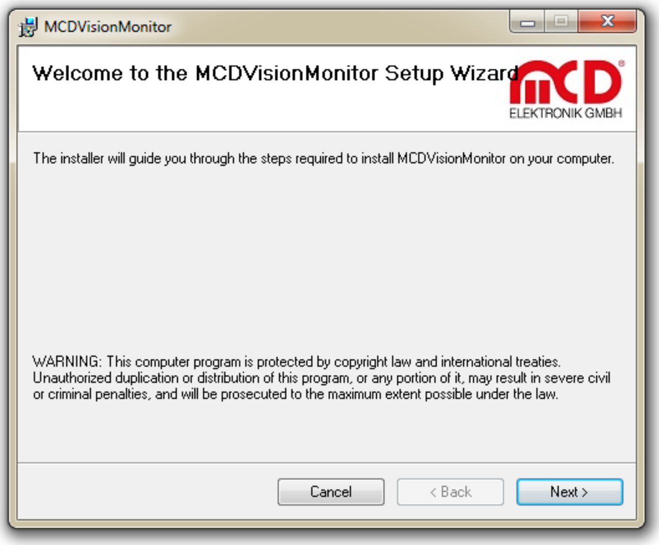
« Lokaler Datenträger (C:) ▸ VisionMonitorInstall

VisionMonitorInstall durchsuchen

Name	Änderungsdatum	Typ	Größe
VisionInstall.msi	12.09.2014 15:51	Windows Installer...	30.214 KB

VisionInstall.msi Änderungsdatum: 12.09.2014 15:51 Erstelldatum: 12.09.2014 15:56
Windows Installer-Paket Größe: 29,5 MB

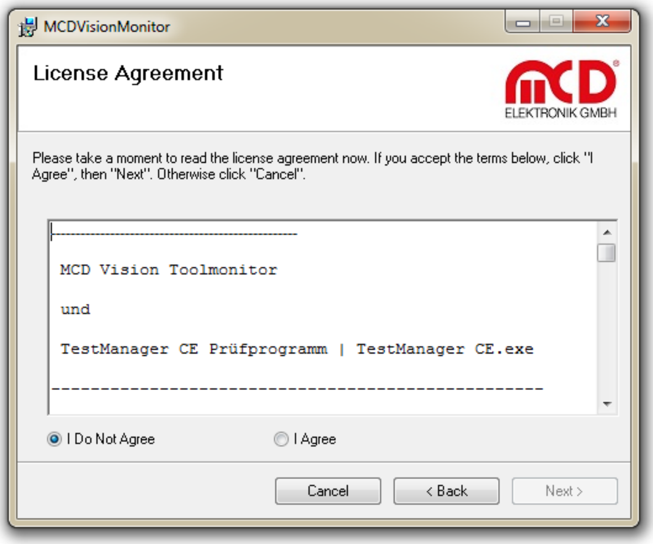
This program provides assistance during the installation of the software. Confirm the individual dialogs and adjust the installation path as needed.



Welcome to the MCDVisionMonitor Setup Wizard

The installer will guide you through the steps required to install MCDVisionMonitor on your computer.

WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.

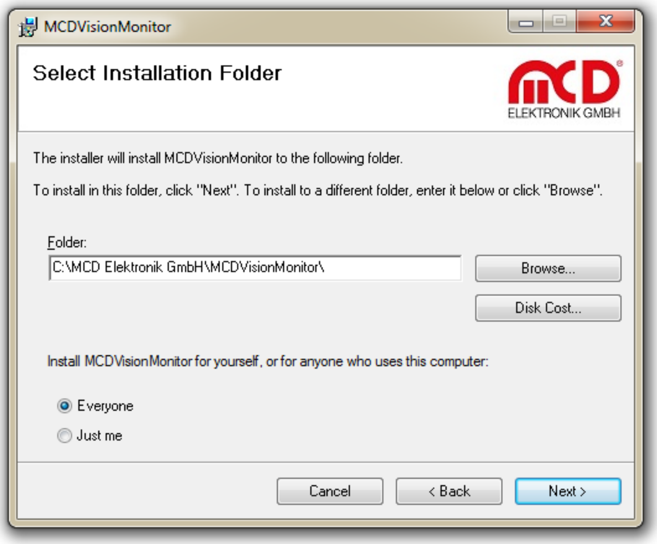


License Agreement

Please take a moment to read the license agreement now. If you accept the terms below, click "I Agree", then "Next". Otherwise click "Cancel".

```
----------------------------------------

MCD Vision Toolmonitor

und

TestManager CE Prüfprogramm | TestManager CE.exe

------------------------------------------------
```

● I Do Not Agree ○ I Agree



Select Installation Folder

The installer will install MCDVisionMonitor to the following folder.

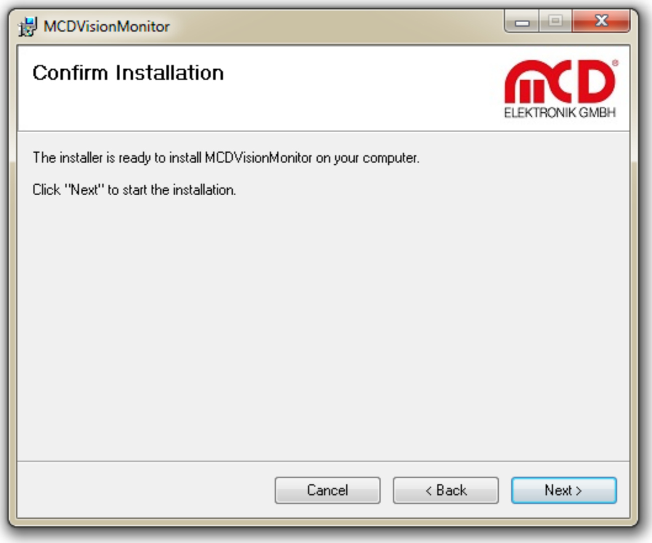
To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".

Folder: C:\MCD Elektronik GmbH\MCDVisionMonitor\

Install MCDVisionMonitor for yourself, or for anyone who uses this computer:

● Everyone

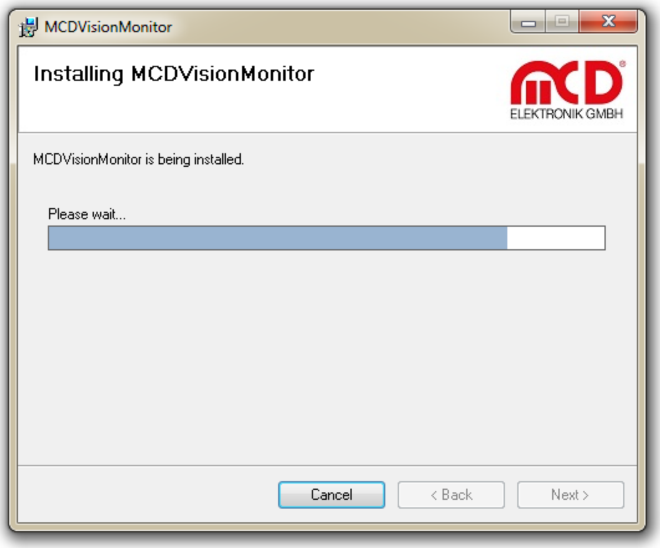
○ Just me



Confirm Installation

The installer is ready to install MCDVisionMonitor on your computer.

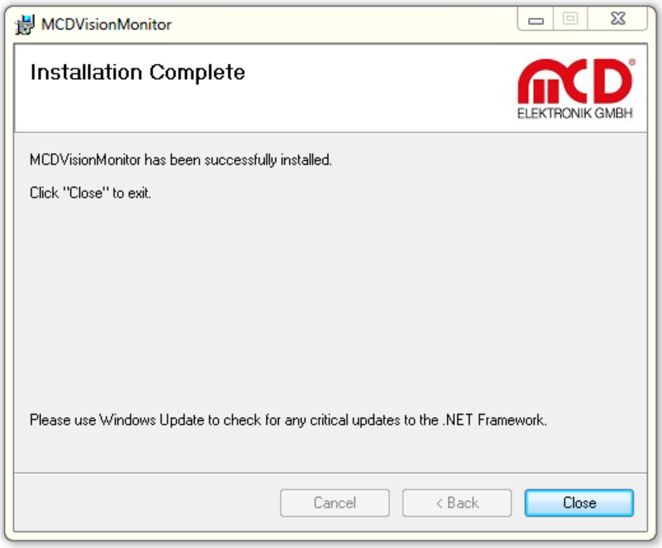
Click "Next" to start the installation.



Installing MCDVisionMonitor

MCDVisionMonitor is being installed.

Please wait...



Installation Complete

MCDVisionMonitor has been successfully installed.

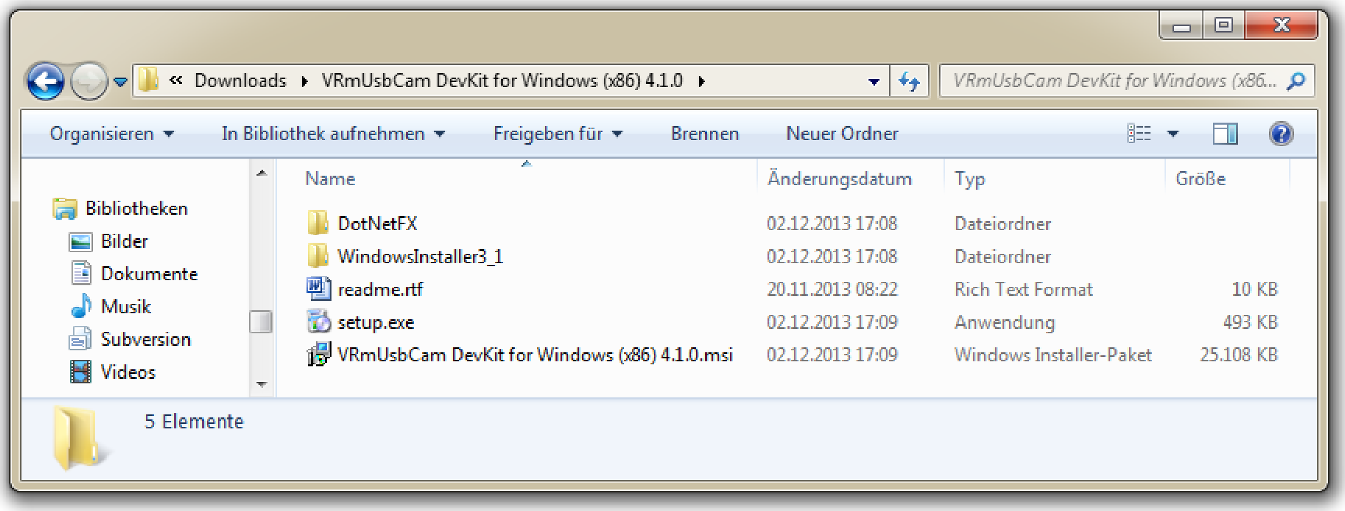
Click "Close" to exit.

Please use Windows Update to check for any critical updates to the .NET Framework.

2.2.2. Installing the VRmagic Driver

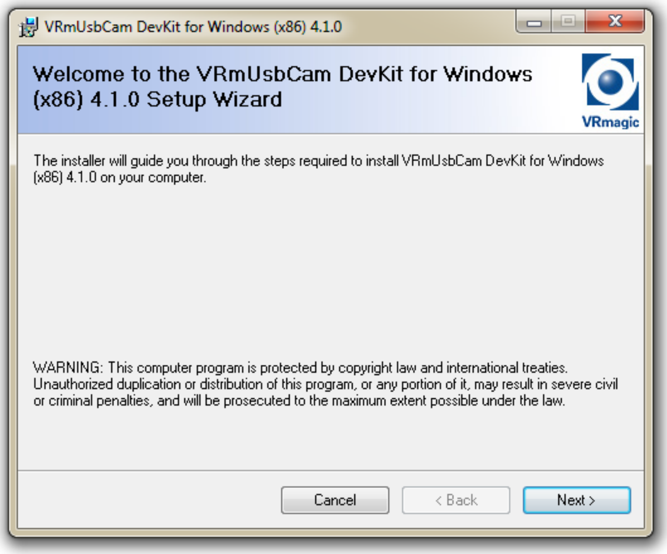
This device driver must be installed to access a VRmagic USB camera. It can be downloaded with the VRmUsbCam DevKit from the VRmagic home page and installed. The most recent version (VRmUsbCam DevKit for Windows X86 x.x Package) can be downloaded from the following page: http://www.vrmagic.com/de/imaging/downloads/. It will also include the CamLab software with which the settings for the connected camera can be made.

The VRmUsbCam DevKit for Windows (x86) 4.1.0.msi installation program should be executed to install the software for the MCD Vision Monitor. To do this, open the Windows® file explorer and start the VRmUsbCam DevKit for Windows (x86) 4.1.0.msi installation program.


« Downloads ▸ VRmUsbCam DevKit for Windows (x86) 4.1.0 ▸
VRmUsbCam DevKit for Windows (x86...
Organisieren ▾
In Bibliothek aufnehmen ▾
Freigeben für ▾
Brennen
Neuer Ordner
Bibliotheken
Bilder
Dokumente
Musik
Subversion
Videos
5 Elemente

Name	Änderungsdatum	Typ	Größe
DotNetFX	02.12.2013 17:08	Dateiordner	
WindowsInstaller3_1	02.12.2013 17:08	Dateiordner	
readme.rtf	20.11.2013 08:22	Rich Text Format	10 KB
setup.exe	02.12.2013 17:09	Anwendung	493 KB
VRmUsbCam DevKit for Windows (x86) 4.1.0.msi	02.12.2013 17:09	Windows Installer-Paket	25.108 KB

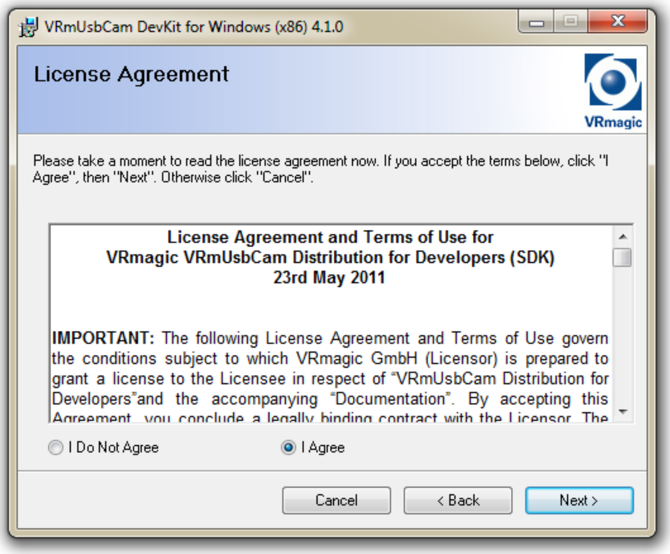
This program provides assistance during the installation of the software. Confirm the individual dialogs and adjust the installation path as needed.



Welcome to the VRmUsbCam DevKit for Windows (x86) 4.1.0 Setup Wizard

The installer will guide you through the steps required to install VRmUsbCam DevKit for Windows (x86) 4.1.0 on your computer.

WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.



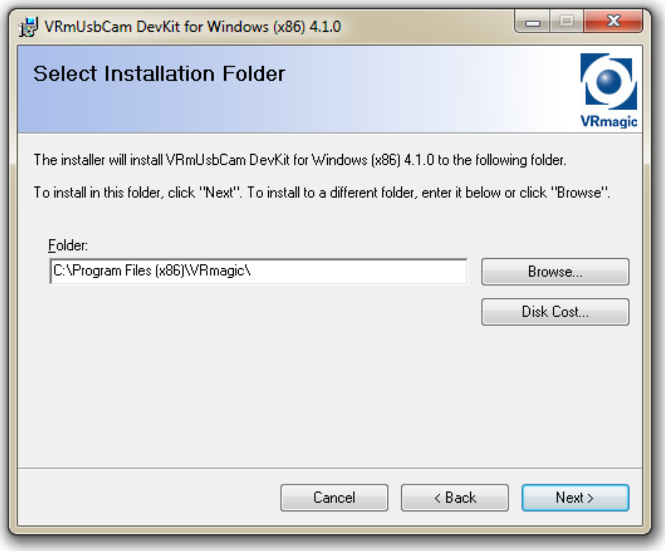
License Agreement

Please take a moment to read the license agreement now. If you accept the terms below, click "I Agree", then "Next". Otherwise click "Cancel".

License Agreement and Terms of Use for
VRmagic VRmUsbCam Distribution for Developers (SDK)
23rd May 2011

IMPORTANT: The following License Agreement and Terms of Use govern the conditions subject to which VRmagic GmbH (Licensor) is prepared to grant a license to the Licensee in respect of "VRmUsbCam Distribution for Developers"and the accompanying "Documentation". By accepting this Agreement, you conclude a legally binding contract with the Licensor. The

○ I Do Not Agree ● I Agree

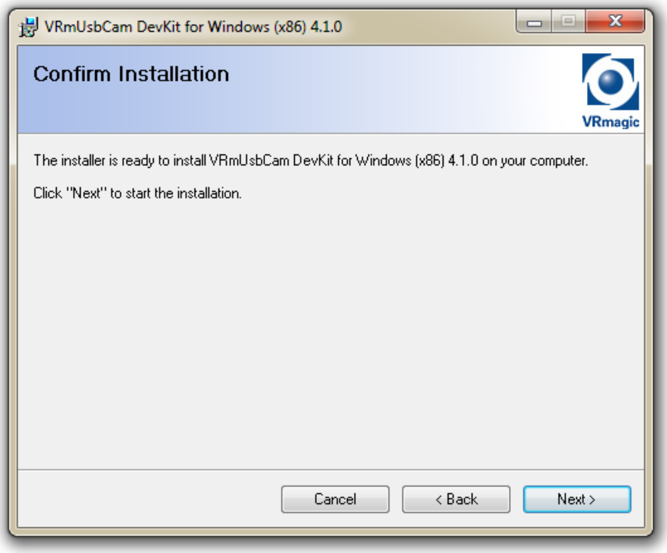


Select Installation Folder

The installer will install VRmUsbCam DevKit for Windows (x86) 4.1.0 to the following folder.

To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".

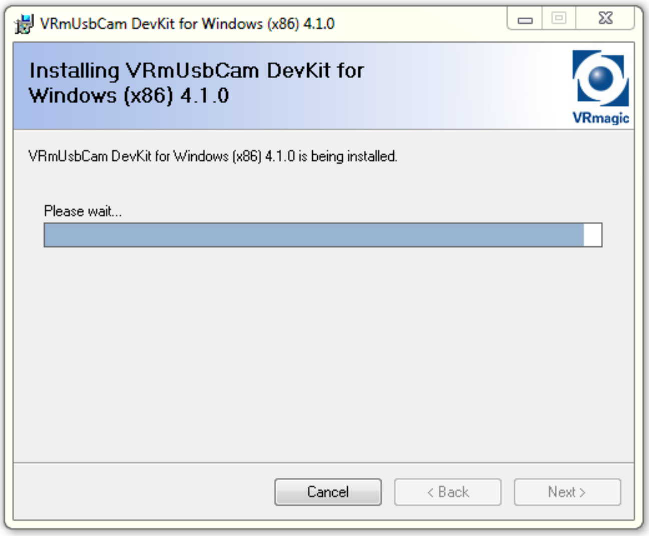
Folder: C:\Program Files (x86)\VRmagic\



Confirm Installation

The installer is ready to install VRmUsbCam DevKit for Windows (x86) 4.1.0 on your computer.

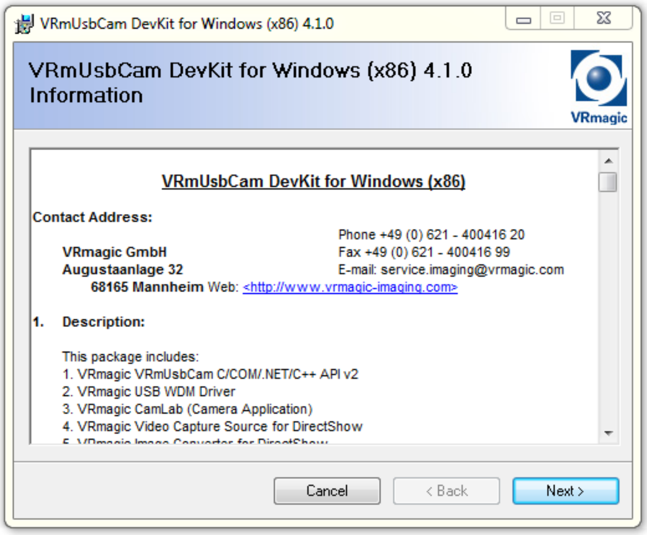
Click "Next" to start the installation.



Installing VRmUsbCam DevKit for Windows (x86) 4.1.0

VRmUsbCam DevKit for Windows (x86) 4.1.0 is being installed.

Please wait...



VRmUsbCam DevKit for Windows (x86) 4.1.0 Information

VRmUsbCam DevKit for Windows (x86)

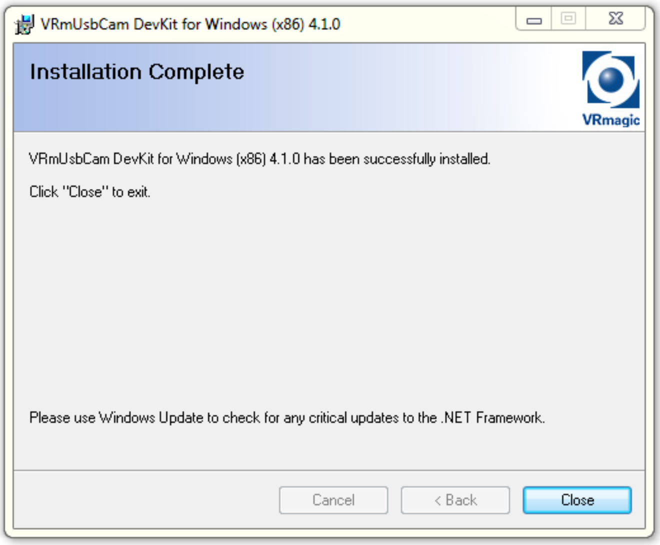
Contact Address:

VRmagic GmbH
Augustaanlage 32
68165 Mannheim

Phone +49 (0) 621 - 400416 20
Fax +49 (0) 621 - 400416 99
E-mail: service.imaging@vrmagic.com
Web: <http://www.vrmagic-imaging.com>

1. Description:

This package includes:
1. VRmagic VRmUsbCam C/COM/.NET/C++ API v2
2. VRmagic USB WDM Driver
3. VRmagic CamLab (Camera Application)
4. VRmagic Video Capture Source for DirectShow
5. VRmagic Image Converter for DirectShow



Installation Complete

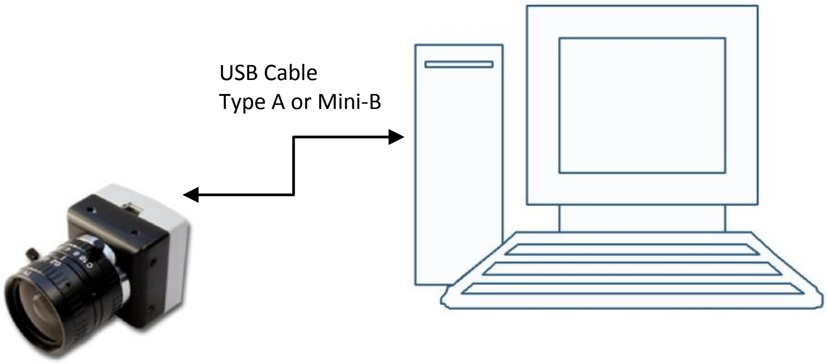
VRmUsbCam DevKit for Windows (x86) 4.1.0 has been successfully installed.

Click "Close" to exit.

Please use Windows Update to check for any critical updates to the .NET Framework.

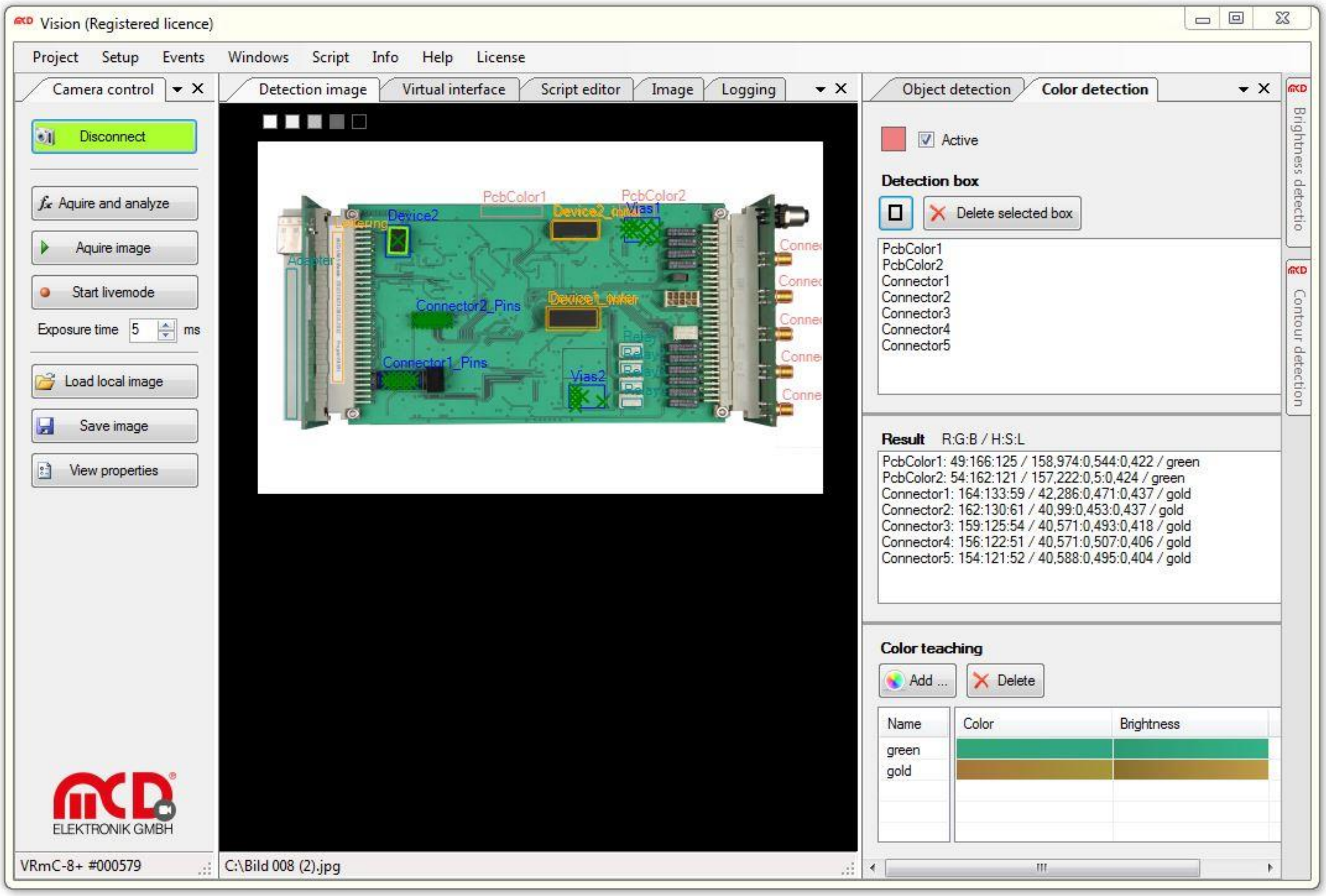
2.2.3. Connecting Hardware

The VRmagic USB camera used will draw its power through the USB connection. To use the camera with Toolmonitor Vision, connect it to the PC where the software has been installed using a USB cable (Type A or Mini-B).



2.2.4. Starting the Installed Software

From the Windows Start menu, find the MCD Elektronik folder. The Toolmonitor Vision can be started from the Vision sub-folder. The connection with the camera can be setup and image processing can work with Toolmonitor Vision.



Exposure time 5 ms

☑ Active

PcbColor1
PcbColor2
Connector1
Connector2
Connector3
Connector4
Connector5

Result R:G:B / H:S:L

```
PcbColor1: 49:166:125 / 158,974:0,544:0,422 / green
PcbColor2: 54:162:121 / 157,222:0,5:0,424 / green
Connector1: 164:133:59 / 42,286:0,471:0,437 / gold
Connector2: 162:130:61 / 40,99:0,453:0,437 / gold
Connector3: 159:125:54 / 40,571:0,493:0,418 / gold
Connector4: 156:122:51 / 40,571:0,507:0,406 / gold
Connector5: 154:121:52 / 40,588:0,495:0,404 / gold
```

Name	Color	Brightness
green	🟩	🟩
gold	🟫	🟫

VRmC-8+ #000579

C:\Bild 008 (2).jpg

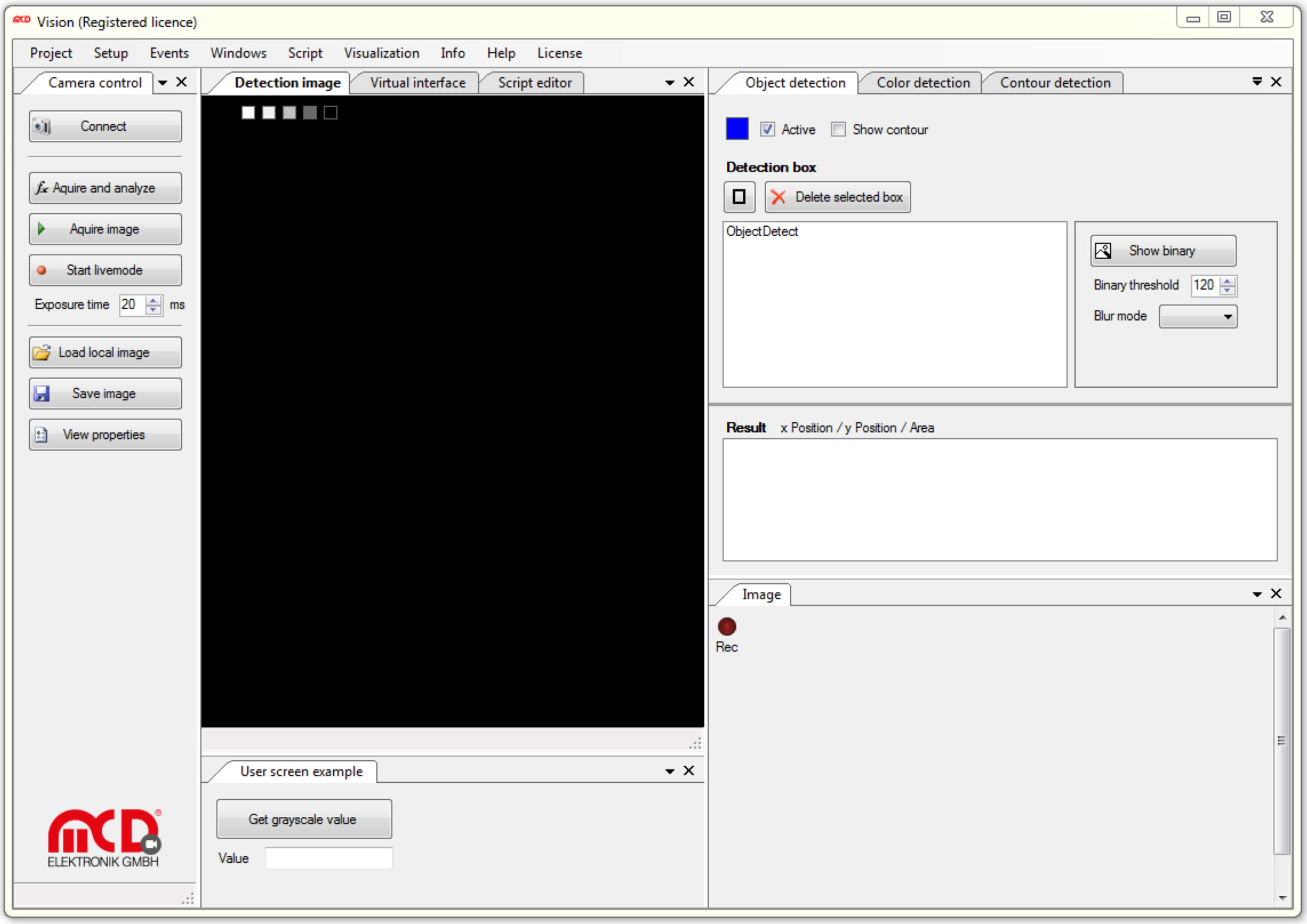
3. Introduction to the operation

3.1. Quick guide

3.2. Using Toolmonitor Vision without a Camera

1. Starting the Toolmonitor Vision

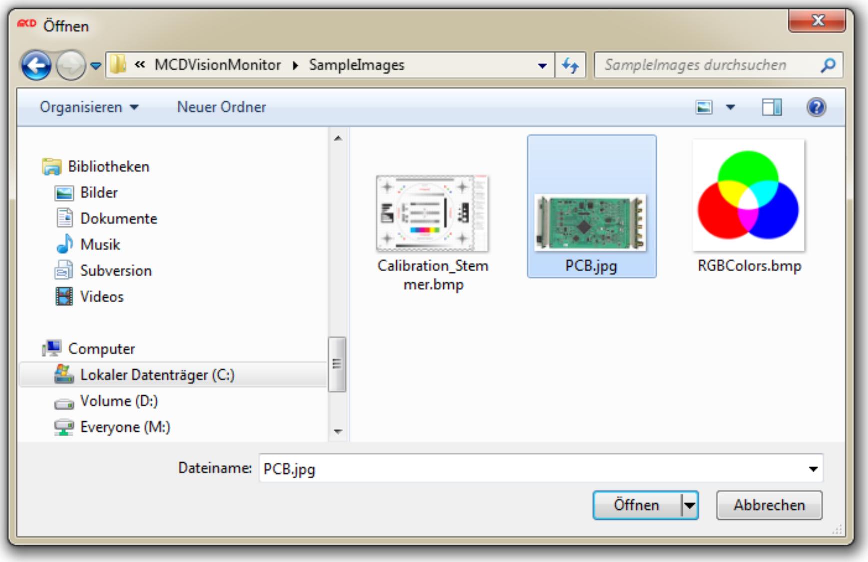
   Use the Windows Start menu to start MCD Toolmonitor Vision. You will then see the following screen.



You can change the viewing mode after starting the software from **Project -> Presets -> Default**, which will load the default settings.

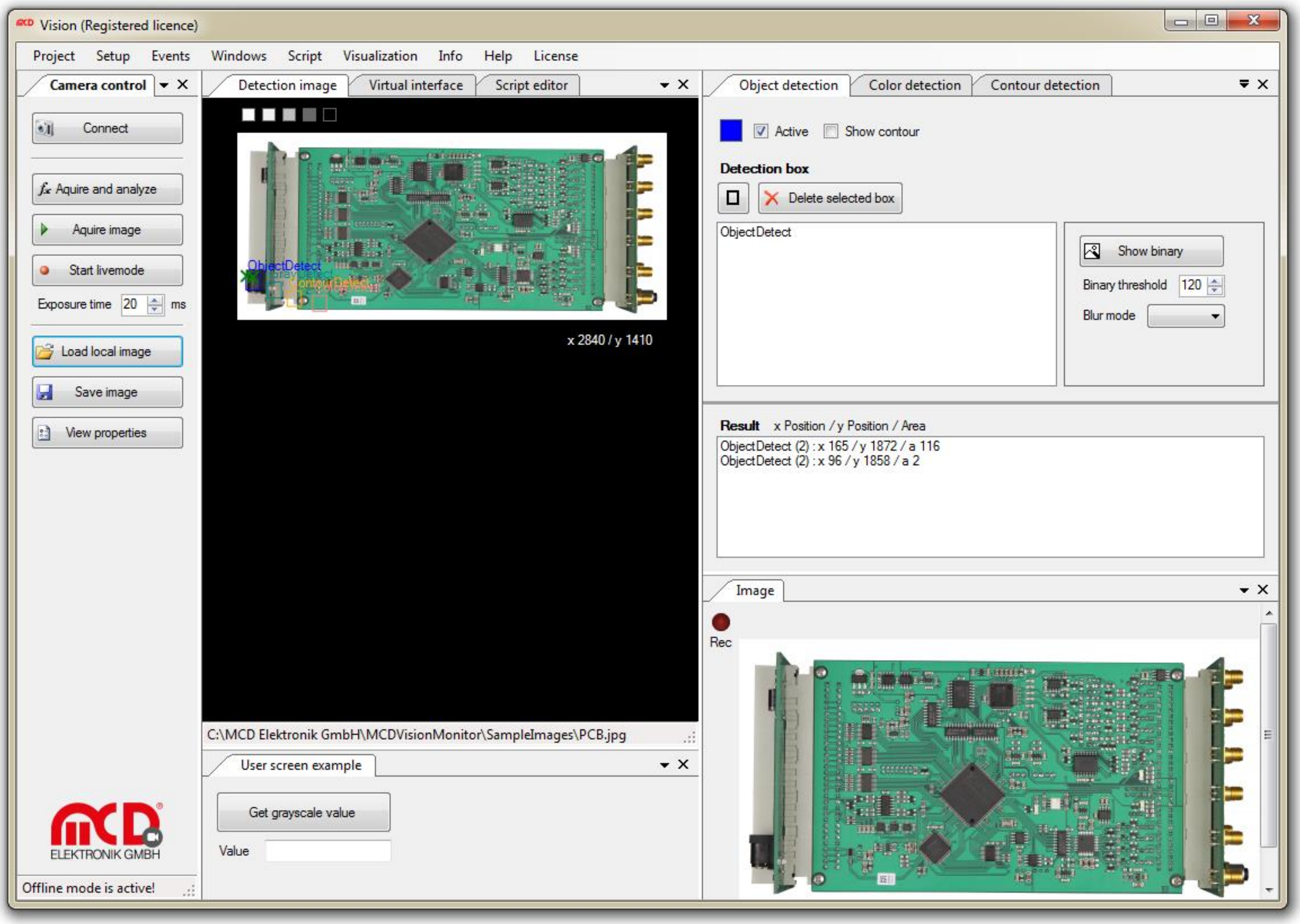
2. Loading a local picture from the PC

   To load an image from your PC for analysis, open the dialog for selecting an image file using the **Load Local Image** button on the Camera Control window.



Dateiname: PCB.jpg

You will see the following view once the desired image has been opened.



Exposure time 20 ms

x 2840 / y 1410

ObjectDetect

Binary threshold 120

ObjectDetect (2) : x 165 / y 1872 / a 116
ObjectDetect (2) : x 96 / y 1858 / a 2

C:\MCD Elektronik GmbH\MCDVisionMonitor\SampleImages\PCB.jpg

Offline mode is active!

You will find information about analyzing image in Section 3.4.

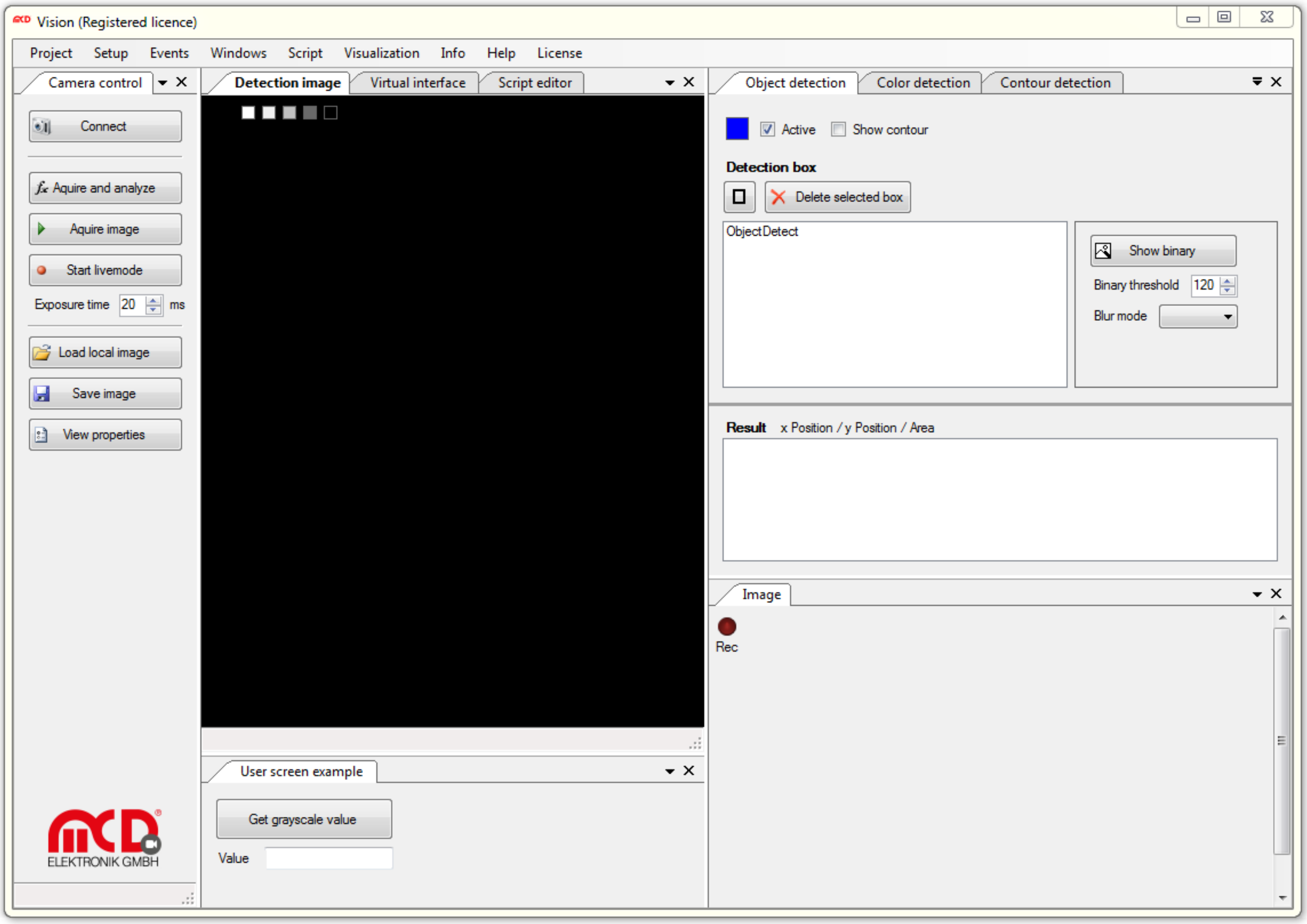
3.3. Using Toolmonitor Vision with a Camera

1. Installing the Toolmonitor Vision and VRmagic driver

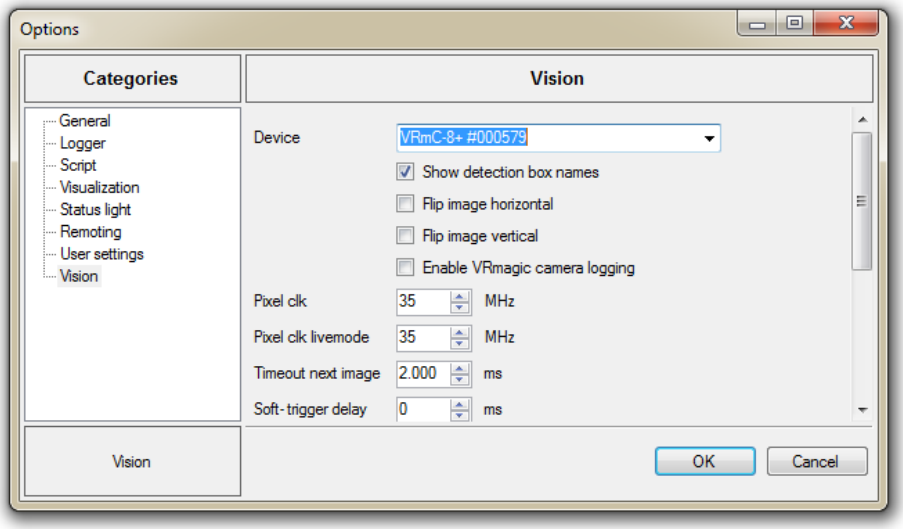
   Please install Toolmonitor Vision and the VRmagic driver as described in the InstallHardAndSoftware.doc file.

2. Starting the Toolmonitor Vision and selecting the Camera

   Use the Windows Start menu to start MCD Toolmonitor Vision. You will then see the following screen.



3. Next, choose a connected camera from the **Device** drop-down list on the **Setup -> Options** dialog and confirm the dialog by clicking **OK**.



Options

Categories

- General
- Logger
- Script
- Visualization
- Status light
- Remoting
- User settings
- Vision

Vision

Device: VRmC-8+ #000579

☑ Show detection box names

☐ Flip image horizontal

☐ Flip image vertical

☐ Enable VRmagic camera logging

Pixel clk: 35 MHz

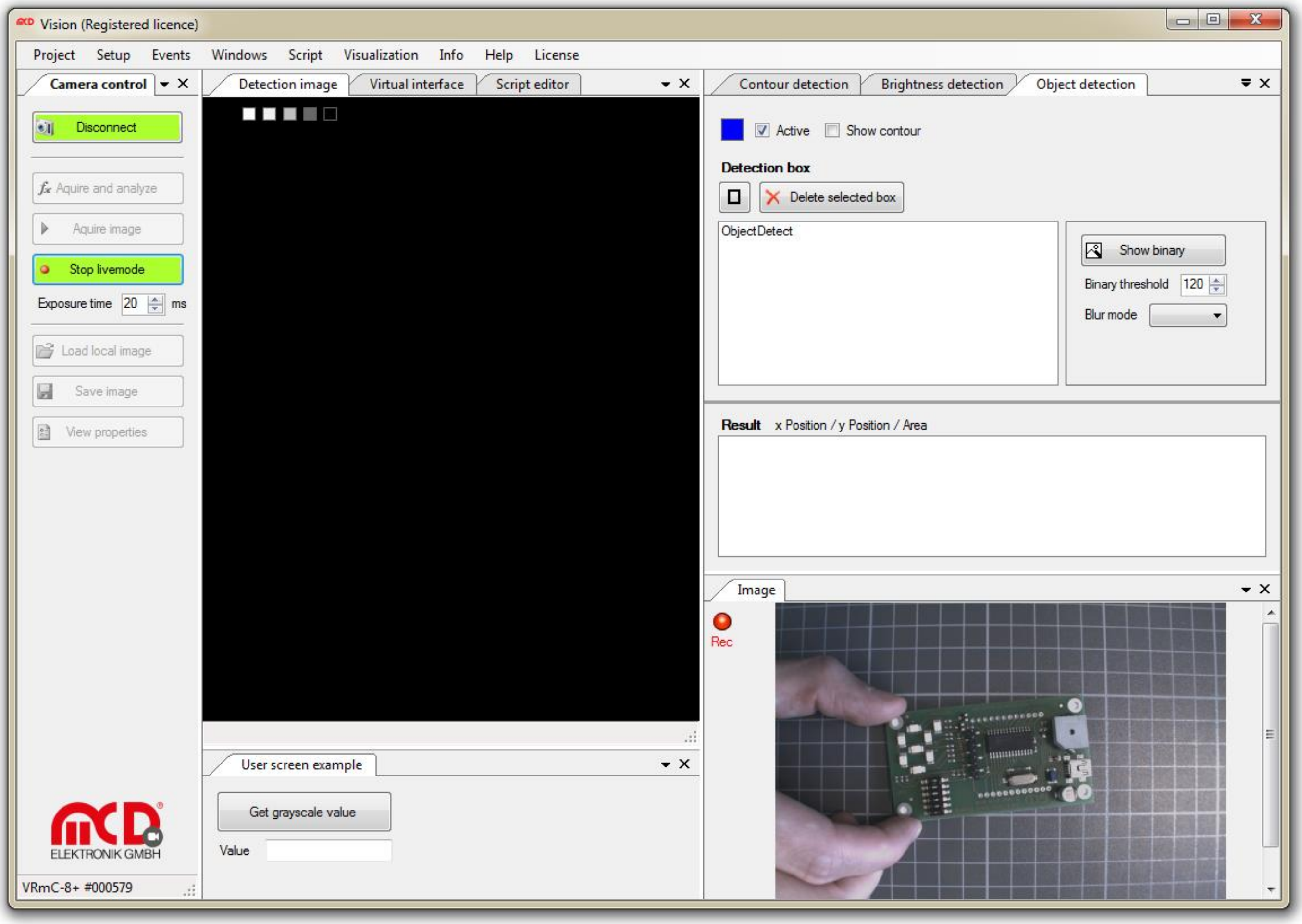
Pixel clk livemode: 35 MHz

Timeout next image: 2.000 ms

Soft-trigger delay: 0 ms

4. Using Live Mode to Setup

Once the camera has been selected from Setup, a connection can be made using the **Connect** button. Start the Live Mode using the **Start Live Mode** button to setup the camera. The actual image from the camera will be displayed in the **Image** window (Windows -> Image). The aperture and focus as well as the flash time can be set. Terminate the Live Mode using the **Stop Live Mode** button.



Vision (Registered licence)

Exposure time: 20 ms

☑ Active ☐ Show contour

ObjectDetect

Binary threshold: 120

Blur mode:

Result x Position / y Position / Area

Value:

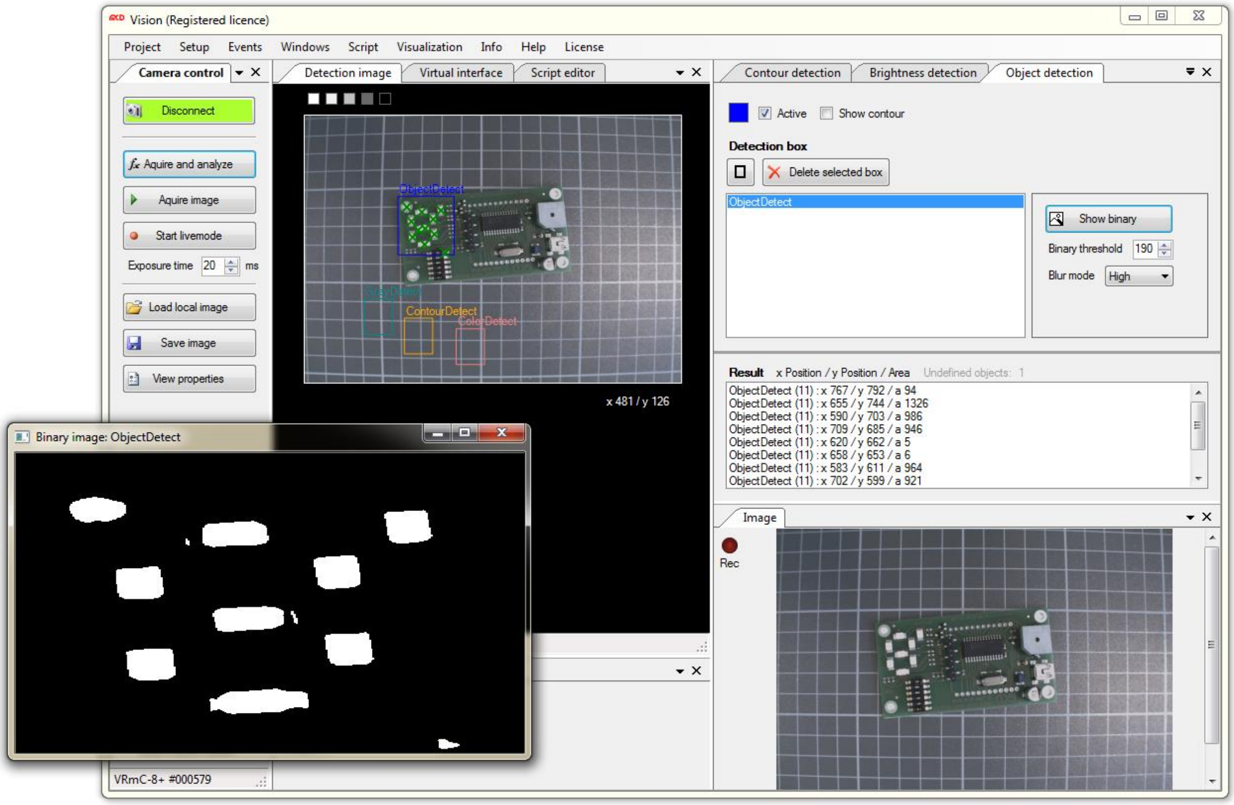
VRmC-8+ #000579

3.4. Conducting Analyses

1. Creating and Positioning the Search Area

   If the software has been successfully started with or without a camera, the search areas for the various tools can be established and the images can be analyzed.

   The various tools can be setup using the **Windows** menu. In the example, the search area will be created using the square cropping tool (□) on the Object Detection tab page. It will need a name and the parameters on the image should be set to the suggested values. Clicking the Detection Box checkbox in the list will start this and the size and position can be adjusted by dragging the lines or corners.



Exposure time 20 ms

Binary threshold 190

Blur mode High

Result x Position / y Position / Area Undefined objects: 1

```
ObjectDetect (11) : x 767 / y 792 / a 94
ObjectDetect (11) : x 655 / y 744 / a 1326
ObjectDetect (11) : x 590 / y 703 / a 986
ObjectDetect (11) : x 709 / y 685 / a 946
ObjectDetect (11) : x 620 / y 662 / a 5
ObjectDetect (11) : x 658 / y 653 / a 6
ObjectDetect (11) : x 583 / y 611 / a 964
ObjectDetect (11) : x 702 / y 599 / a 921
```

2. Conducting an Analysis

   A new analysis can be started by clicking the **Acquire and analyze** button. To change the Detection Box parameters, select them in the **Object Detection window**. The image will be processed for object detection using the specified parameters. The effects on the image will be displayed on the binary image (the **Show binary** button).

3. Accessing the Results

   The results will be displayed in the **Result** list for each tool.

4. Software manual

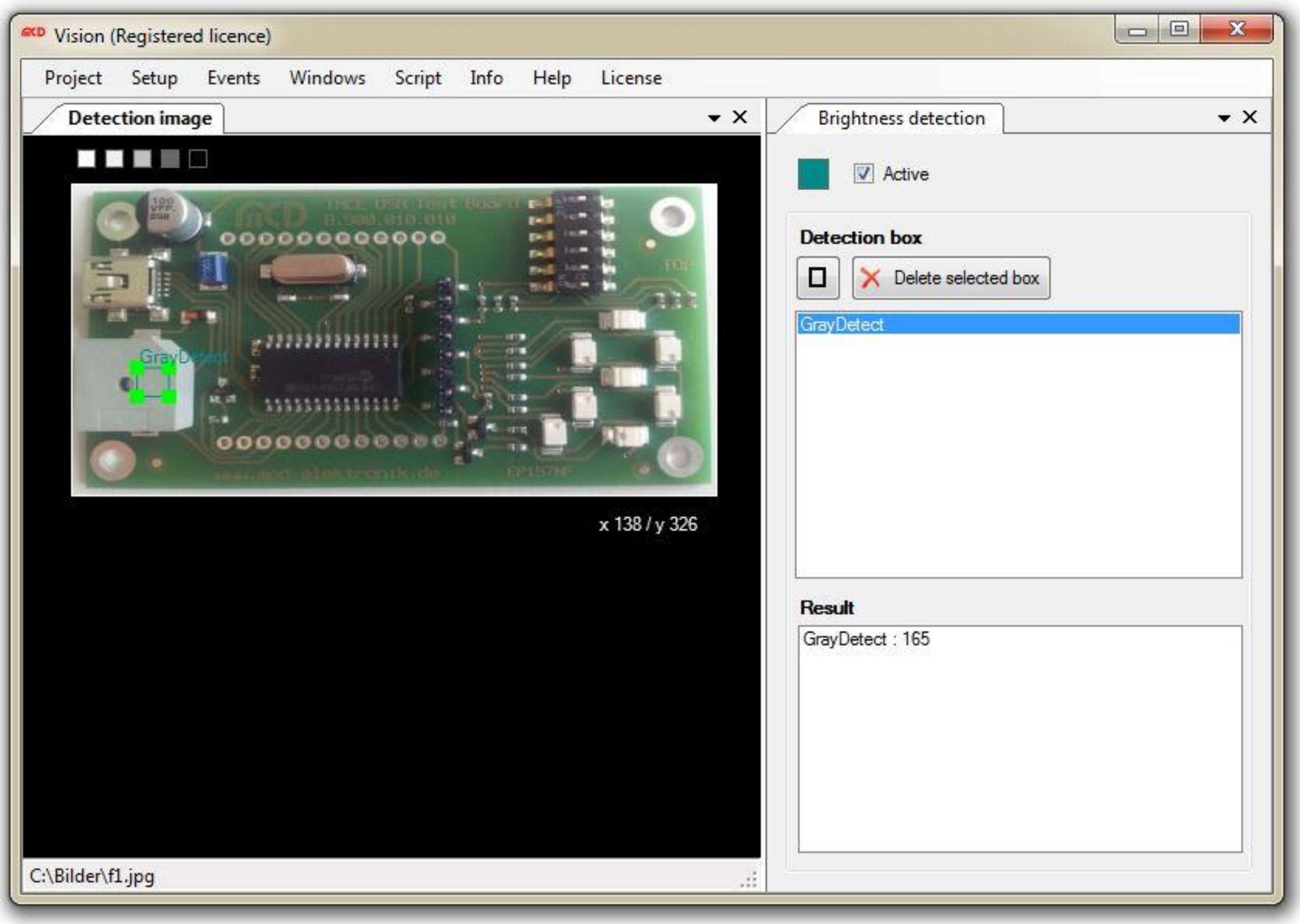
4.1. Windows

All tools and views can be opened via the Windows menu. The different views are explained in detail in the following chapters.

4.1.1. Brightness Detection

General Information

Using the Brightness detection image-processing tool, the mean gray value of a selected area (ROI - region of interest) can be determined. For this purpose, different regions of interest can be selected and checked using a detection box. The result is presented as 8-bit (0-255) gray value.



GrayDetect : 165

C:\Bilder\f1.jpg

Brightness detection with Toolmonitor Vision

User Interface

The gray level (brightness) detection image-processing tool can be activated or deactivated using the **Active** button. If regions of interest were created for analysis, the analysis can be activated or deactivated.

The user interface for gray level analysis consists of two areas. In the **Detection box** area, a new detection box (ROI) can be created or deleted using the rectangle button. The existing boxes are displayed. By selecting a box from the list, it is marked in the Detection image window and can be enlarged or shrunk using the corner markers. Selecting an edge of the box marks the entire box, which can then be moved.

The **Result** area displays the results for every detection box. The result comprises the determined gray level for every detection box.

Adding a Detection Box

Clicking the **rectangle button** opens a new dialog, in which a new detection box can be added. The name for the box can be assigned here. By clicking on the Add detection box button, the new box is then added. Box size and position can be adjusted in the Detection image window.

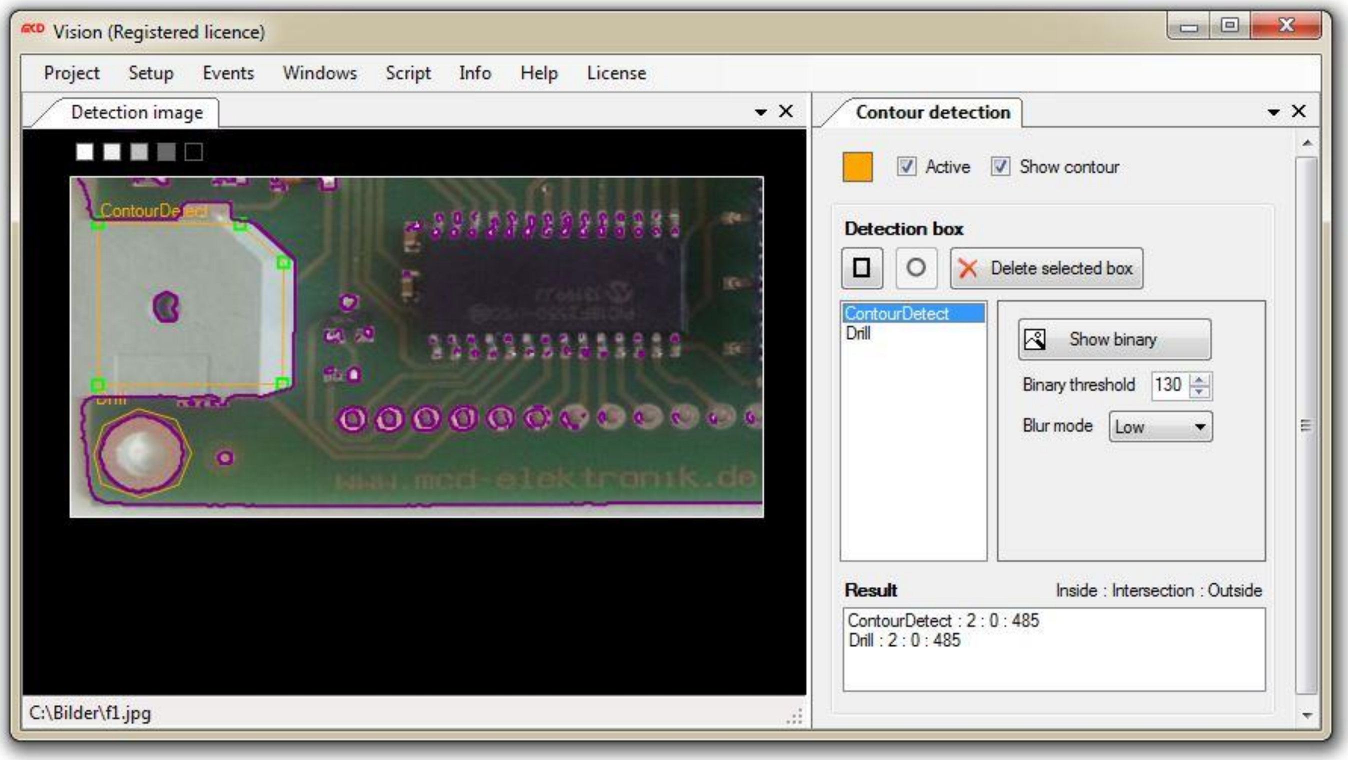
4.1.2. Contour Detection

General Information

The **Contour detection** image-processing tool enables the detection of contours in the captured image. Three types of contours can be detected:

- **Inside:** Contours, fully within the detection box
- **Intersection:** Contours, crossing the outer edges of the detection box
- **Outside:** Contours, fully outside of the detection box

The detected contours can be displayed in the **Detection image** window. For this purpose, the **Show contour** button in the Contour detection window must be activated. The contour display can be deactivated to save computing time in automatic mode.



☑ Active ☑ Show contour

ContourDetect
Drill

Binary threshold 130

Blur mode Low

Result Inside : Intersection : Outside

ContourDetect : 2 : 0 : 485
Drill : 2 : 0 : 485

C:\Bilder\f1.jpg

Contour detection with Toolmonitor Vision

User Interface

The contour detection image-processing tool can be activated or deactivated using the Active button. If regions of interest were created for analysis, the analysis can be activated or deactivated.

The user interface for contour detection consists of two areas. In the Detection box area, a new detection box (ROI) can be created or deleted using the rectangle button. The existing boxes are displayed. By selecting a box from the list, it is marked in the Detection image window and can be enlarged or shrunk using the corner markers. Selecting an edge of the box marks the entire box, which can then be moved. Using the context menu (right mouse button), additional markers can be added during contour detection to obtain a desired check window. In addition, the image preprocessing parameters can be adjusted. More detailed information regarding image preprocessing can be found in chapter Image Preprocessing / Binary Image.

The Result area displays the results for every detection box. The result comprises of the number of determined contours in the Inside / Outside / Intersection categories described above.

Image Preprocessing / Binary Image

Automatic image preprocessing is performed in Toolmonitor Vision to carry out the contour detection. This step can be parameterized in the Detection box area. First, a gray level image is generated for contour detection, followed by a binary image. The binary image resulting from the preprocessing step can be displayed in an additional window using the **Show binary** button. Using the **Binary threshold**, it can be determined, which gray levels in the image should be filtered as black and which as white. The next step is filtering using a **blur filter**, which blurs the image to filter out certain image capturing effects (e.g., noise). The filter can be set between None, Low, and High using the Blur mode. The parameters for binary threshold and blur mode are the same for all detection boxes during contour detection (different from object detection).

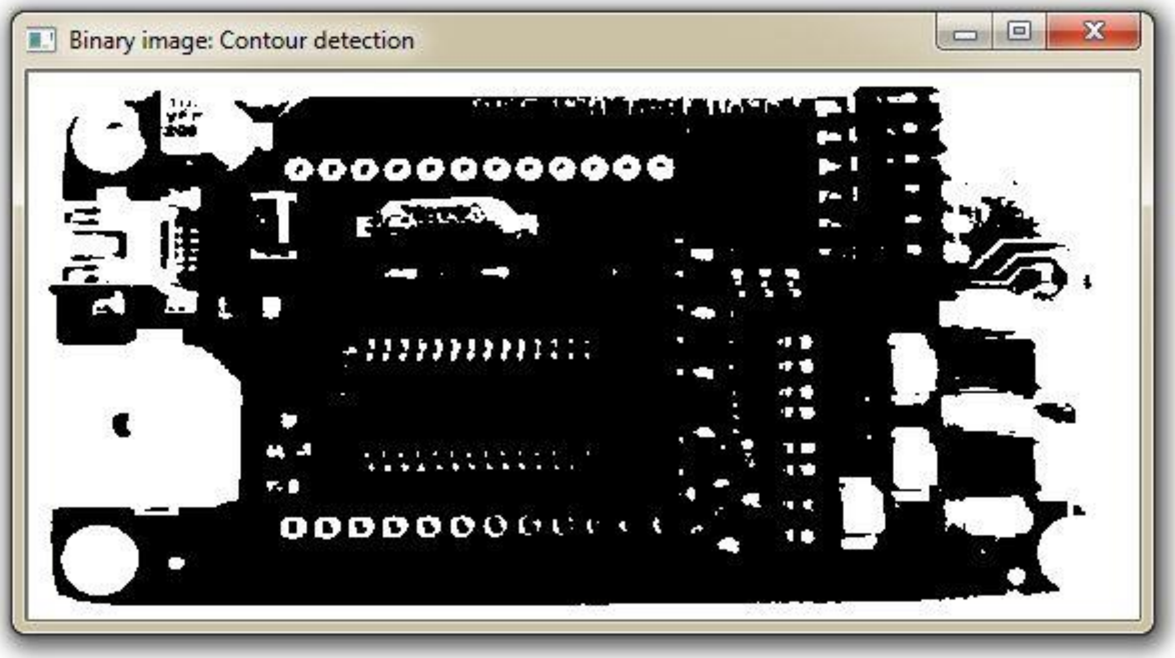


Image preprocessing: Binary image

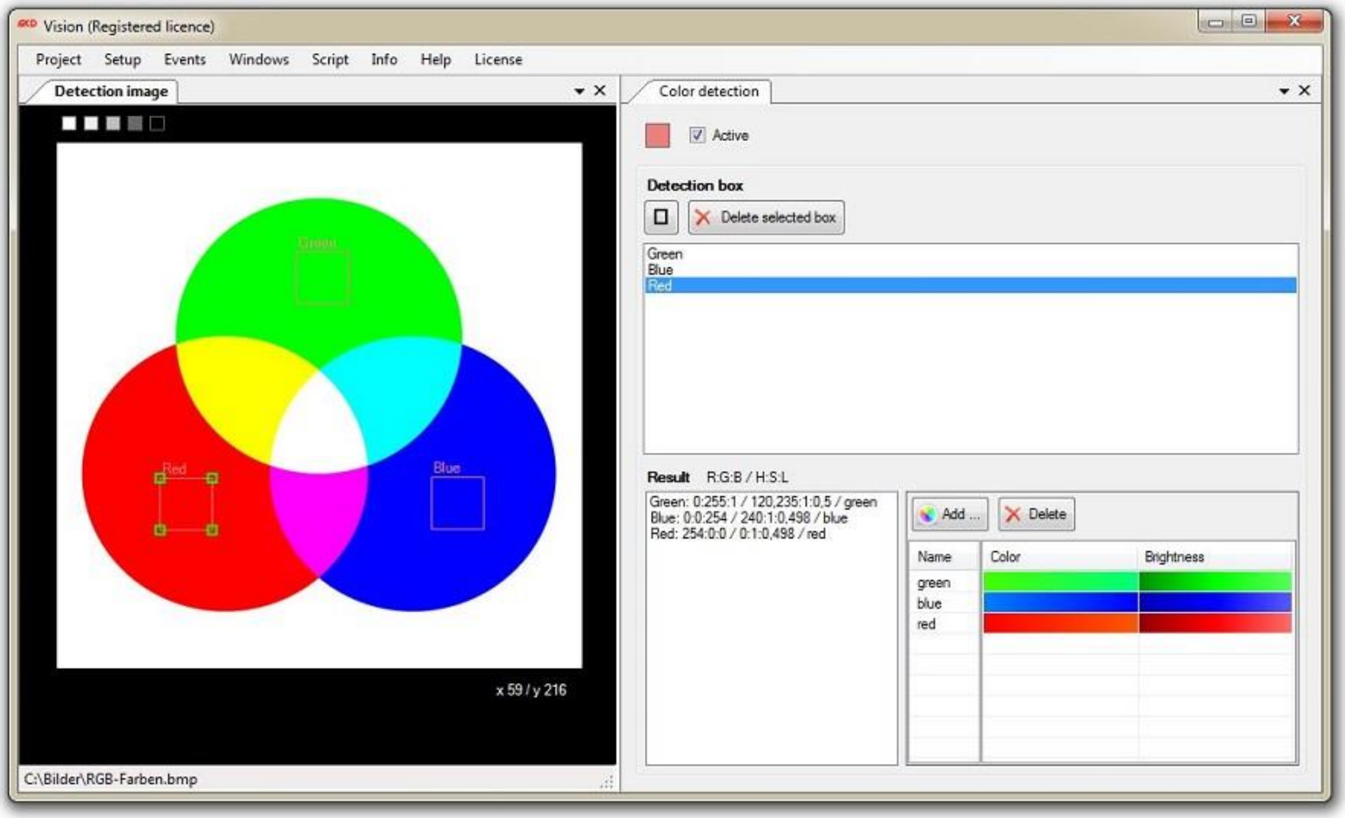
Adding a Detection Box

Clicking the **rectangle button** opens a new dialog, in which a new detection box can be added. The name for the box can be assigned here. By clicking on the Add detection box button, the new box is then added. Box size and position can be adjusted in the Detection image window.

4.1.3. Color Detection

General Information

Using the **Color detection** image-processing tool, the **mean color value** of a selected area (ROI - region of interest) can be determined. For this purpose, different regions of interest can be selected and checked using a detection box. The result is presented in the RGB and HSL color space. A **color list** can be **created** in addition. This list is used to teach colors and the set color and brightness limit values for this color. If a taught color is detected, the color name for the ROI can be analyzed in the result.



Green: 0:255:1 / 120,235:1:0,5 / green
Blue: 0:0:254 / 240:1:0,498 / blue
Red: 254:0:0 / 0:1:0,498 / red

Name	Color	Brightness
green	🟩	🟩
blue	🟦	🟦
red	🟥	🟥

Color detection with Toolmonitor Vision

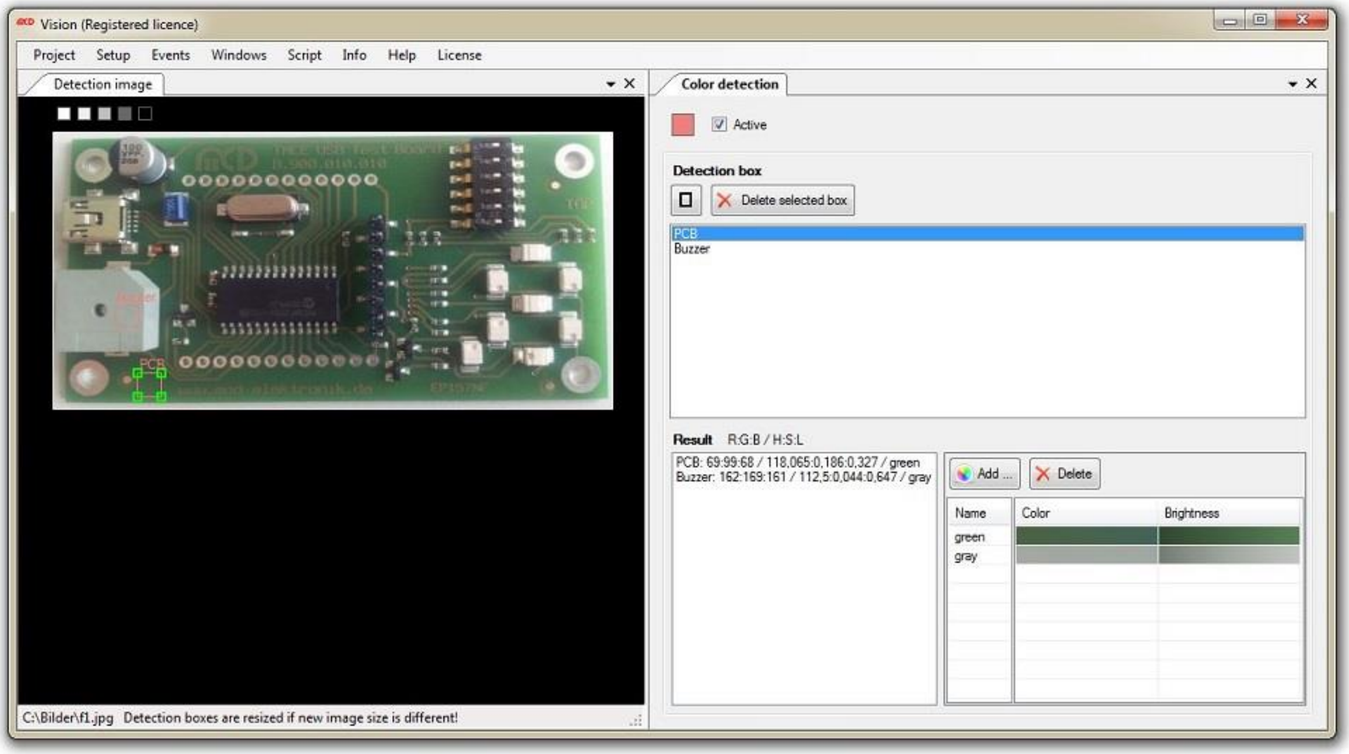
User Interface

The color detection image-processing tool can be activated or deactivated using the Active button. If regions of interest were created for analysis, the analysis can be activated or deactivated.

The user interface for color analysis consists of two areas. In the Detection box area, a new detection box (ROI) can be created or deleted using the rectangle button. The existing boxes are displayed. By selecting a box from the list, it is marked in the Detection image window and can be enlarged or shrunk using the corner markers. Selecting an edge of the box marks the entire box, which can then be moved.

The Result area displays the results for every detection box. The result comprises the RGB and HSL values. If a taught color was detected, the color name is displayed as well.

Colors can be taught and their limit values defined in the Color teaching area. Further information can be found in chapter Color Teaching.



☑ Active

PCB
Buzzer

Result R:G:B / H:S:L

```
PCB: 69:99:68 / 118,065:0,186:0,327 / green
Buzzer: 162:169:161 / 112,5:0,044:0,647 / gray
```

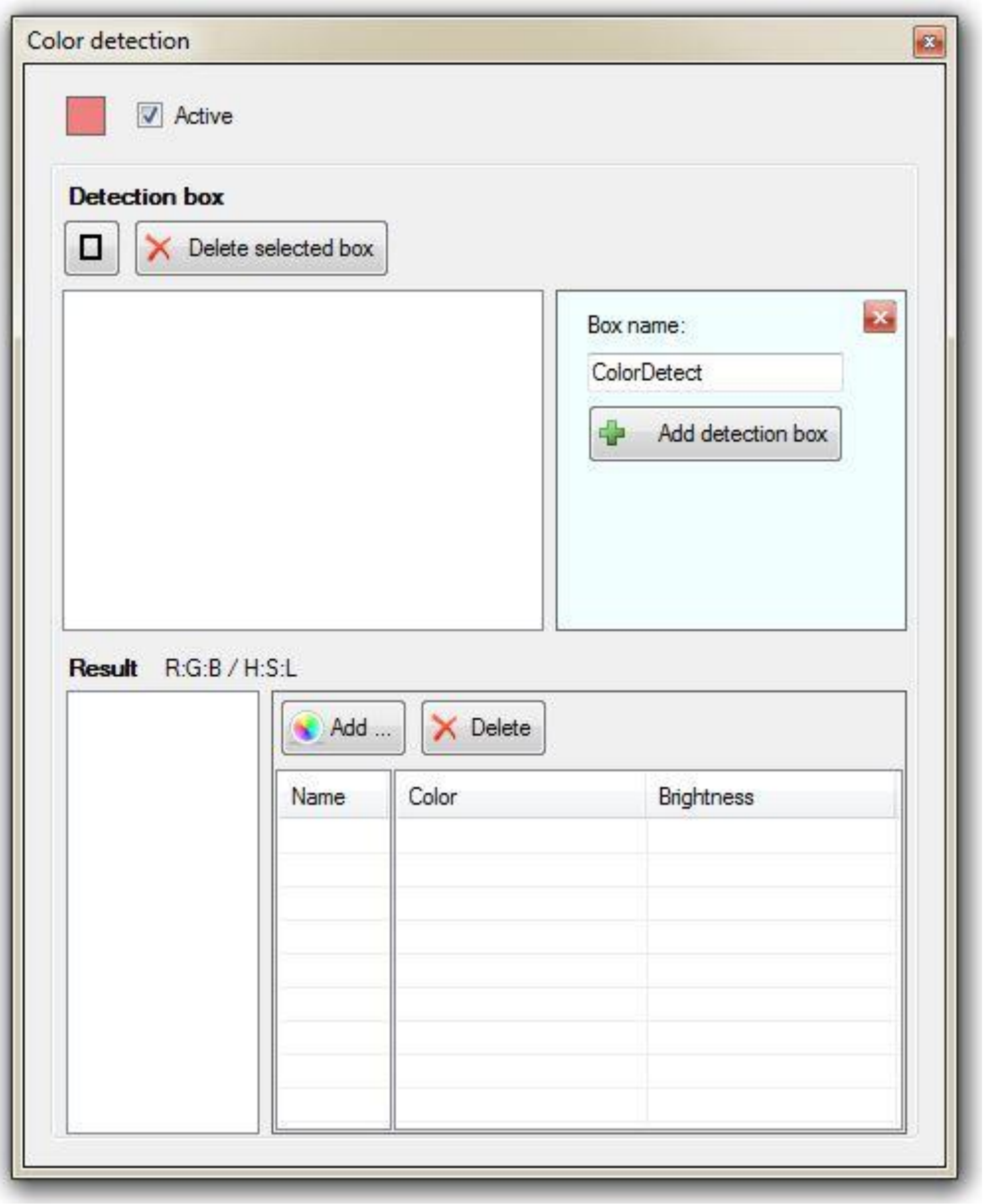
Name	Color	Brightness
green	🟩	🟩
gray	▨	▨

C:\Bilder\f1.jpg Detection boxes are resized if new image size is different!

Color detection with Toolmonitor Vision

Adding a Detection Box

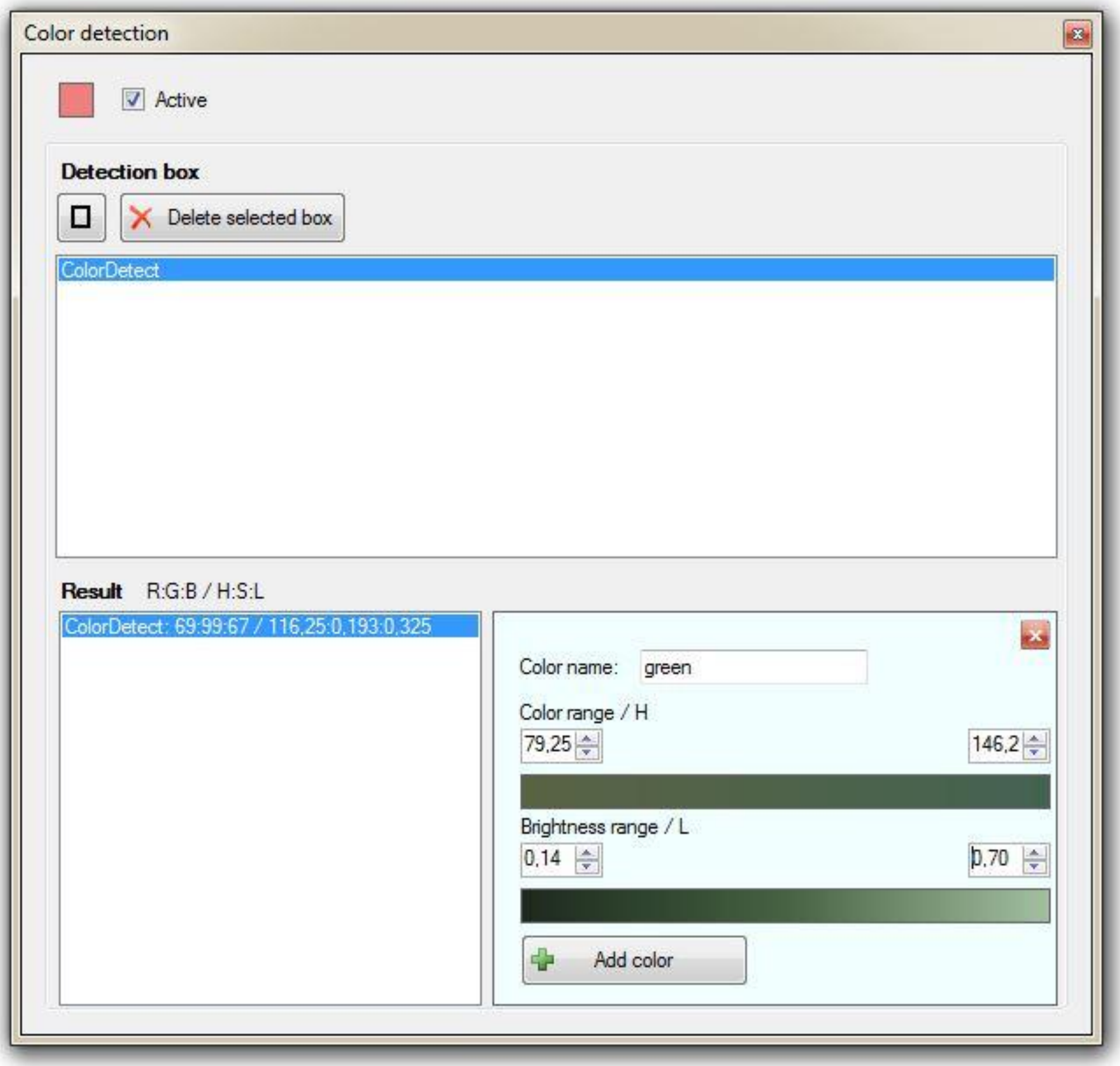
Clicking the rectangle button opens a new dialog, in which a new detection box can be added. The name for the box can be assigned here. By clicking on the Add detection box button, the new box is then added. Box size and position can be adjusted in the Detection image window.



Adding a detection box

Color Teaching

At least one detection box is required in the Color detection tool to teach a new color. If one box exists, the analysis must be carried out via **Acquire and analyze**. To teach the desired color, the result must be marked first in the **result list**. The dialog for creating the color is then opened using the **Add color** button. Here, a name can be entered for the color. An upper and lower limit for valid colors in the color spectrum can be entered for **Color range / H** (values 0-359). The valid colors are then presented as color gradient. The upper and lower brightness can be entered as limit for **Brightness range / L** (values 0.00 - 1.00). The range for the different permitted brightnesses is shown as color gradient as well. Once all settings were applied, the color can be added via **Add color**. Double-clicking on the color name allows **changes to the settings later on**. Now, a user-defined number of detection boxes can be created for the analysis to detect the taught color.



Color detection

☑ Active

Detection box

Delete selected box

ColorDetect

Result R:G:B / H:S:L

ColorDetect: 69:99:67 / 116,25:0,193:0,325

Color name: green

Color range / H

79,25 146,2

Brightness range / L

0,14 0,70

Add color

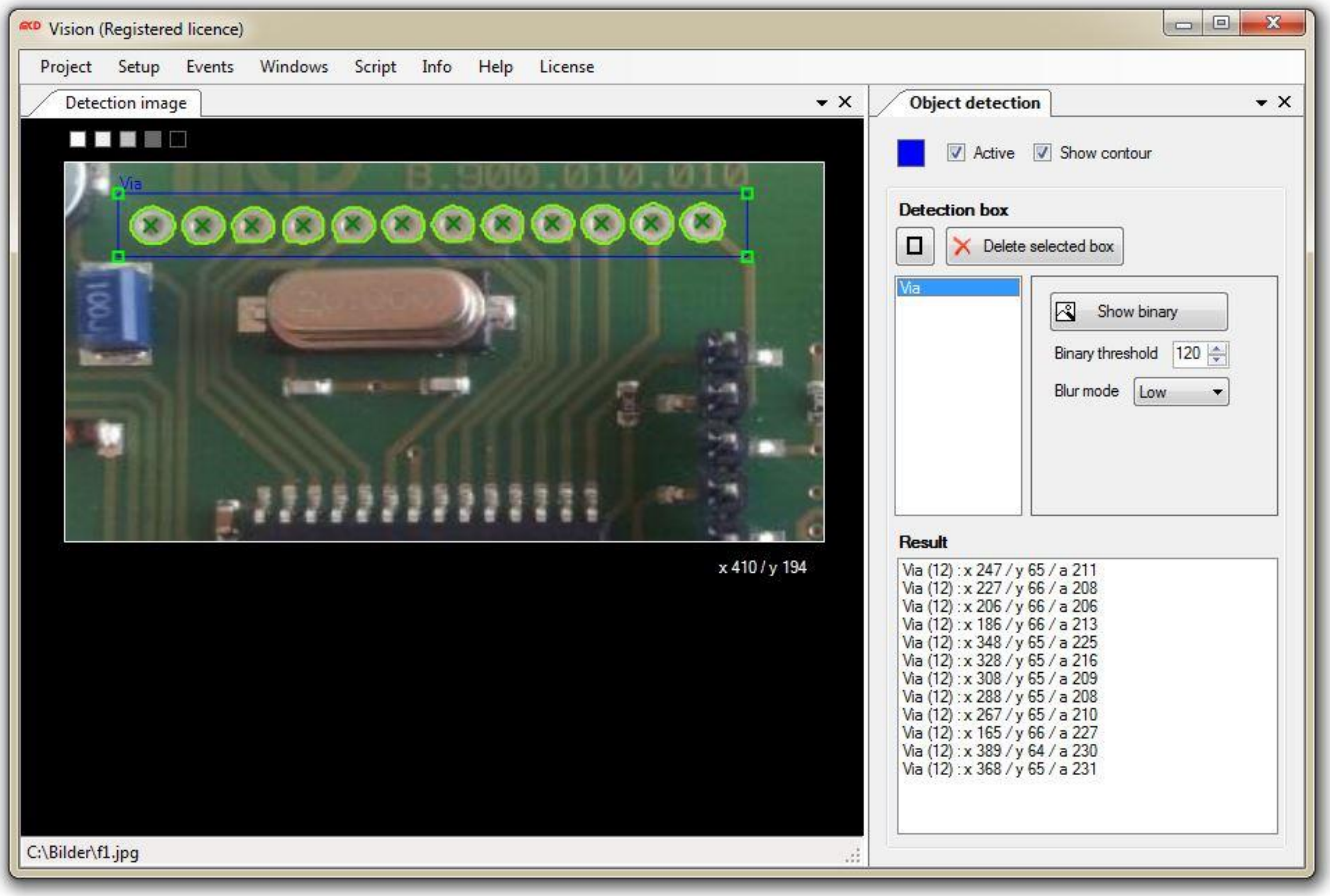
Adding a color

4.1.4. Object Detection

General Information

The **Object detection** image-processing tool enables the detection of objects via contour analysis. The **position in X/Y coordinates** (pixel) and the **contour area** (pixel) can be determined within a region of interest. The position of the object is calculated based on the centroid of the area.

The position of the determined objects is displayed in the **Detection image**. The display of the object contours can be activated via the **Show contour** button. The contour display can be deactivated to save computing time in automatic mode.



☑ Active ☑ Show contour

Binary threshold 120

Blur mode Low

```
Via (12) : x 247 / y 65 / a 211
Via (12) : x 227 / y 66 / a 208
Via (12) : x 206 / y 66 / a 206
Via (12) : x 186 / y 66 / a 213
Via (12) : x 348 / y 65 / a 225
Via (12) : x 328 / y 65 / a 216
Via (12) : x 308 / y 65 / a 209
Via (12) : x 288 / y 65 / a 208
Via (12) : x 267 / y 65 / a 210
Via (12) : x 165 / y 66 / a 227
Via (12) : x 389 / y 64 / a 230
Via (12) : x 368 / y 65 / a 231
```

Object detection with the Toolmonitor Vision

User Interface

The object detection image-processing tool can be activated or deactivated using the **Active** button. If regions of interest were created for analysis, the analysis can be activated or deactivated.

The user interface for contour detection consists of two areas. In the **Detection box** area, a new detection box (ROI) can be created or deleted using the **rectangle button**. The existing boxes are displayed. By selecting a box from the list, it is marked in the **Detection image** window and can be enlarged or shrunk using the corner markers. Selecting an edge of the box marks the entire box, which can then be moved. In addition, the **image preprocessing parameters** can be adjusted. More detailed information regarding image preprocessing can be found in chapter Image Preprocessing / Binary Image.

The **Result** area displays the results for every detection box. Here, the result comprises the results for every detection box. The number of detected objects per detection box is indicated in parentheses after the box name.

Image Preprocessing / Binary Image

Automatic image preprocessing is performed in Toolmonitor Vision to carry out the object detection. This step can be parameterized in the **Detection box** area. In the case of object detection, these parameters are individually created and saved for every detection box. First, a gray level image is generated for contour detection, followed by a binary image. The binary image of the selected detection box, resulting from the preprocessing step can be displayed in an additional window using the **Show binary** button. Using the **Binary threshold**, it can be determined, which gray levels in the image should be filtered as black and which as white. The next step is filtering using a **blur filter**, which blurs the image to filter out certain image capturing effects (e.g., noise). The filter can be set between None, Low, and High using the Blur mode. To adjust the parameters of already existing detection boxes later on, mark the box in the list of boxes. These parameters are automatically saved.

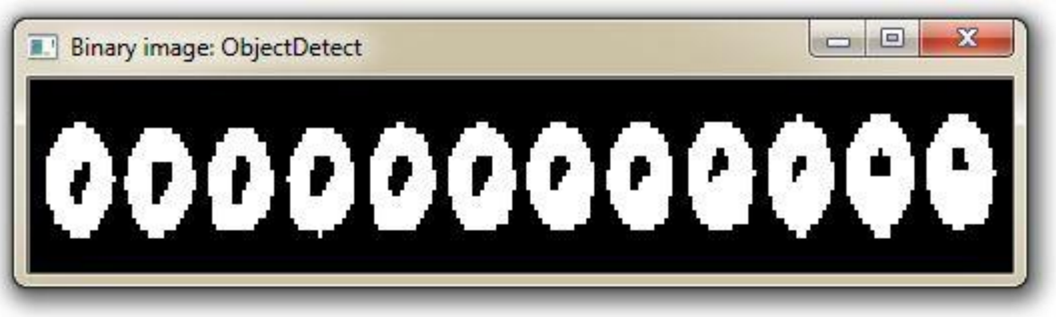


Image preprocessing: binary image of the ROI (region of interest)

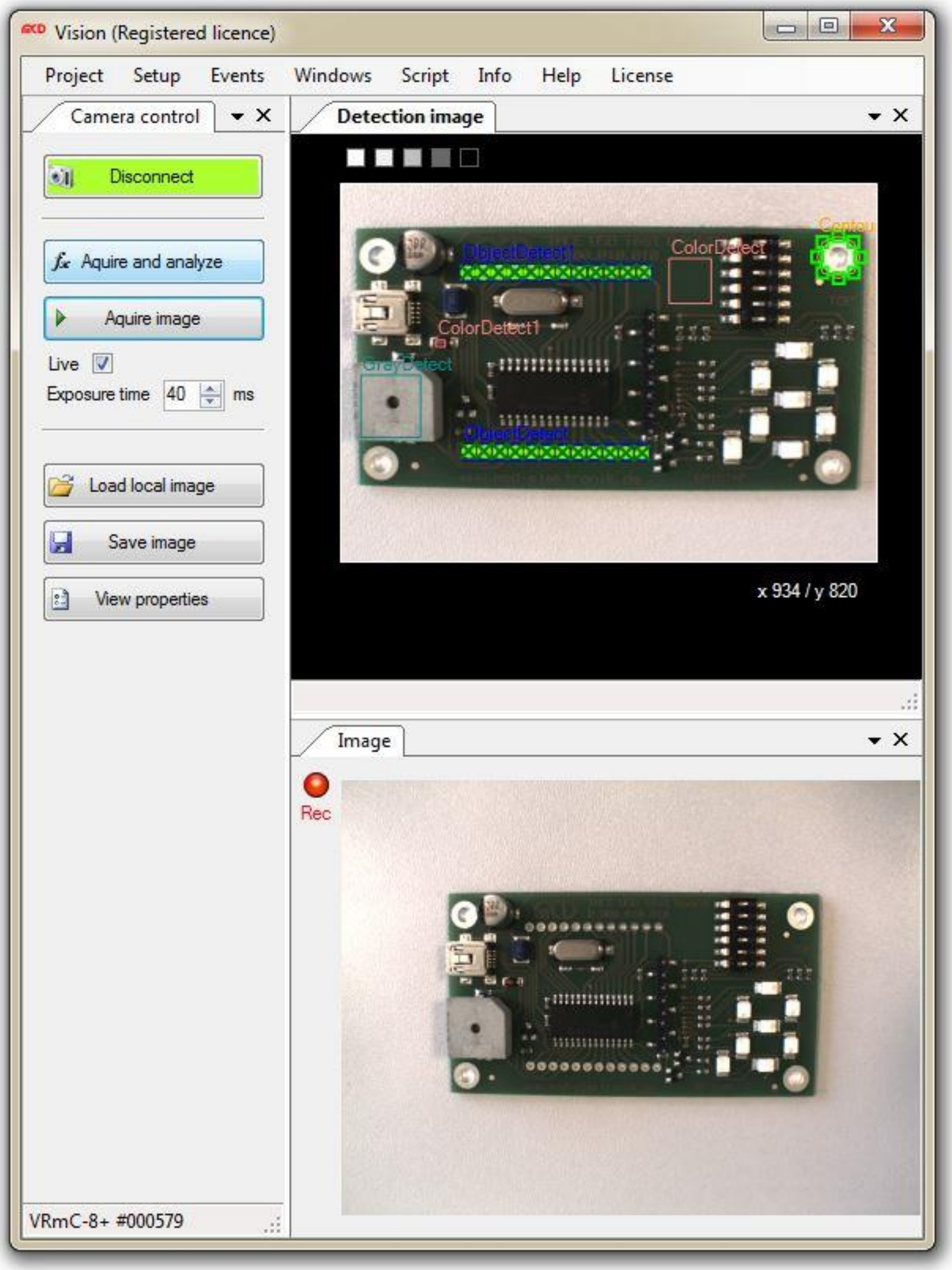
Adding a Detection Box

Clicking the **rectangle button** opens a new dialog, in which a new detection box can be added. The name for the box can be assigned here. By clicking on the Add detection box button, the new box is then added. Box size and position can be adjusted in the Detection image window.

4.1.5. Camera Control

General Information

The **Camera control** dialog is used for controlling the Toolmonitor Vision. In addition to **establishing a connection to a connected camera**, **images saved on PC can be loaded** or **captured images can be saved**. Furthermore, image capturing can be adjusted via exposure time, and configured using live mode.



MCD camera control with image in live mode

Connect / Disconnect

Clicking the Connect button establishes the connection to a connected camera. The camera can be selected first in the Options. Once the camera connection was established successfully, an image is captured and displayed automatically.

Once a connection to a camera is established, the Connect button is shown in green and changes automatically into the Disconnect button. Clicking the Disconnect button disconnects the connection to a connected camera. The button changes automatically into the Connect button.

Acquire and Analyze

Clicking the Acquire and Analyze button captures an image in online mode (camera connection is established). The activated image processing tools are executed. In offline mode (image was loaded from PC), all activated image processing tools are executed when clicking on the button.

Acquire image

A new image can be captured using the Acquire image button. This image is displayed in the Image window. The image processing tools are not executed. For example, this function can be used for system setup. The live mode is a further setup tool. Activating the Live checkbox and clicking the Acquire image button displays a live video of the connected camera in the Image window. This mode is very well suited for system setup. Deactivating the Live checkbox exits live mode.

Furthermore, the camera exposure time can be adjusted here as well. The exposure time can be individually adjusted depending on the capturing situation to achieve optimum image brightness. However, the aperture of the lens used should always be adjusted first to keep the exposure time as short as possible.

Load local image

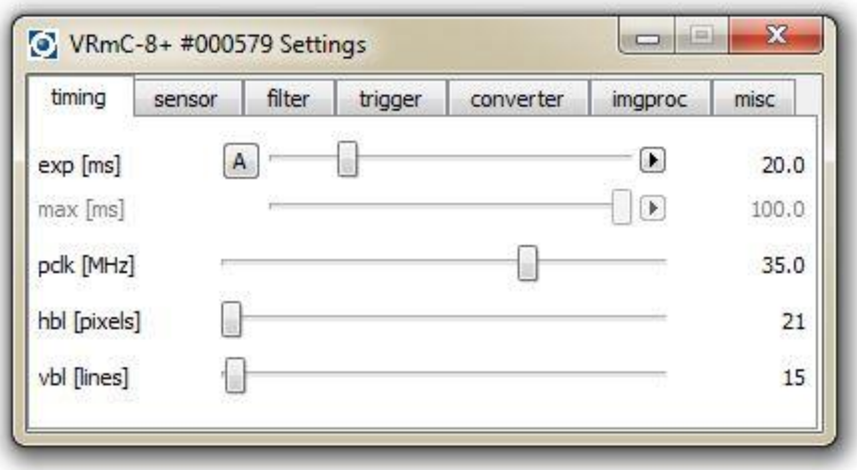
An image saved on PC can be loaded using the Load local image button. Using this function, the image processing tools can also be set up without the camera. Images are required only. Clicking on the button displays the standard file opening dialog. Any bitmap file can be loaded. After the images was loaded, the image processing tools can be executed as usual via the Acquire and analyze button.

Save Image

The Save image function enables saving of the current image in the Detection image window in online mode (with a camera connected). For this purpose, the standard save file dialog is displayed when clicking the button. The image can be saved under the desired name.

View properties

The standard properties of the connected camera can be adjusted in Toolmonitor Vision. All additional properties can be adjusted and saved in VRMagic CAM-Lab. All camera properties can be displayed in Vision Monitor by clicking on the View properties button. The properties window is shown in the figure.



VRmC-8+ #000579 Settings

exp [ms] 20.0

max [ms] 100.0

pclk [MHz] 35.0

hbl [pixels] 21

vbl [lines] 15

Displaying the camera settings (VRMagic)

4.1.6. Detection Image

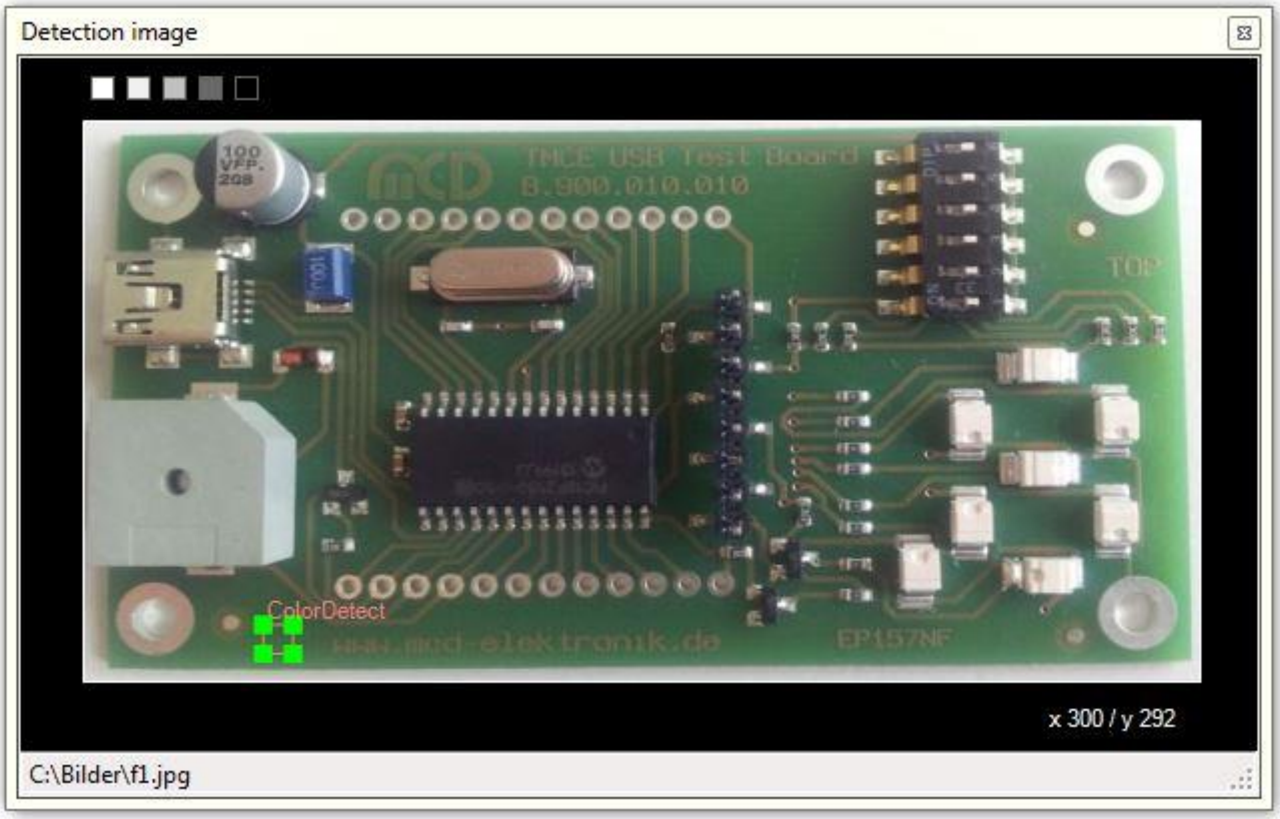
General Information

The Detection image window is the main display and main workspace for Toolmonitor Vision. The captured image is displayed here, the search areas (regions of interest) are created and changed, and determined contours displayed.

The background color can be adjusted to different gray levels via the small boxes in the upper left corner to provide for a more comfortable work environment.

The cursor position is displayed in the lower right corner. This is helpful when, e.g., searching for or analyzing objects.

If an image is loaded via the Load local image button, the path of the loaded image is displayed in the line at the bottom of the window. If an image were loaded with smaller dimensions than the image previously loaded, all created detection boxes that would now be outside of the new image, are adjusted to the new image size. This is indicated by message "Detection boxes are resized if new image size is different!" in the line at the bottom of the window.



Detection image display

4.1.7. Image

General Information

The Image window is used to display the raw image without boxes for analysis or contours. The image can be newly captured by clicking on the Acquire image button. If the Live button is activated, a live video can be displayed in the Image window via Acquire image.

The Image window can be used for setup and checking of the captured images. This is helpful if the captured image is overlaid with detection boxes or contours in the Detection image window.

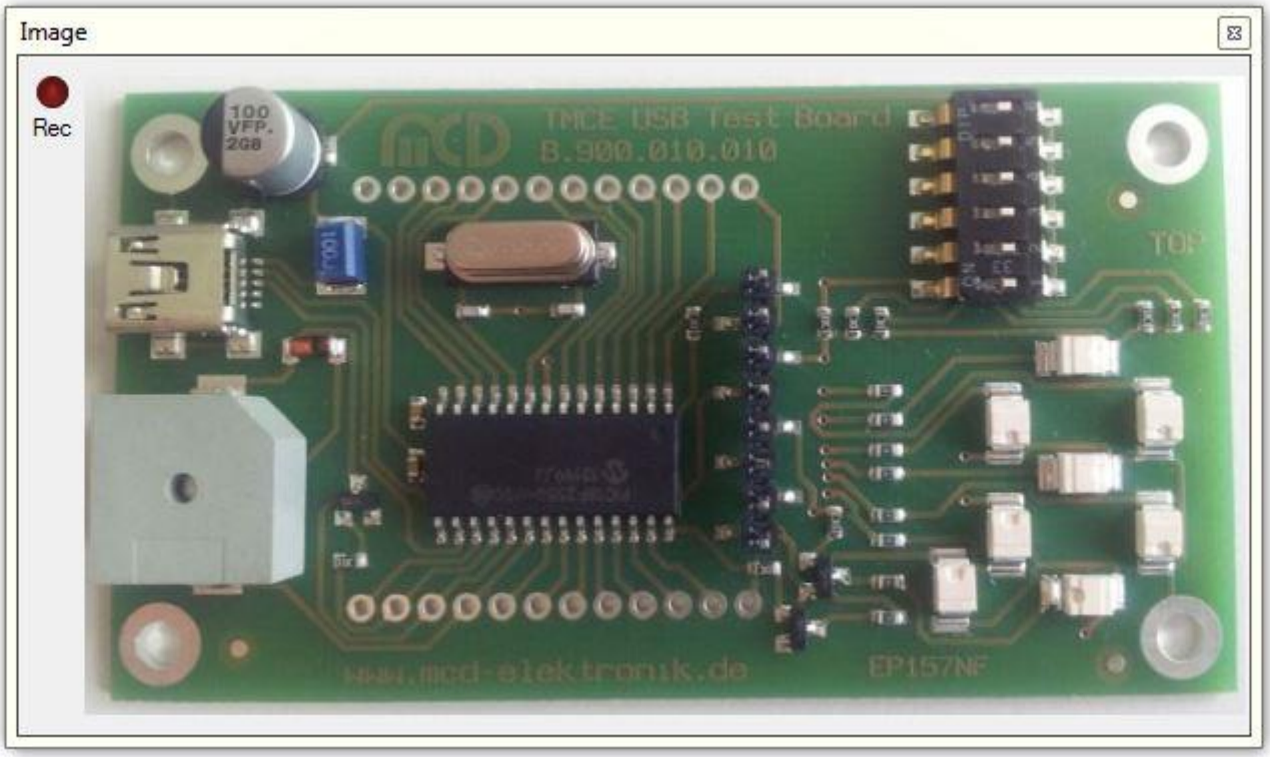
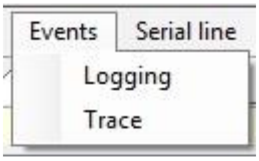


Image display

4.1.8. Events

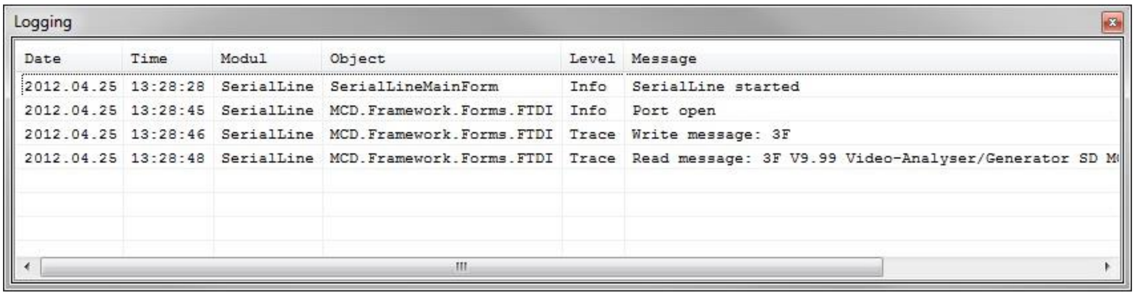
Here, the views for the log and trace messages can be accessed.



Events menu

Logging

Using this menu, the log messages view for general events, warnings, errors, etc. is called up.

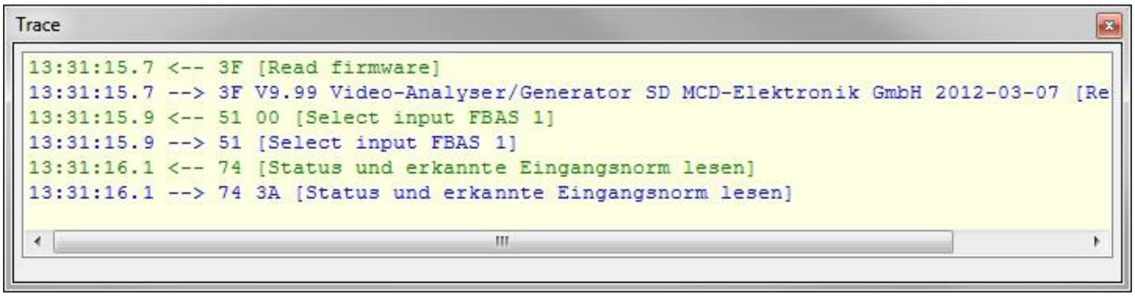


Date	Time	Modul	Object	Level	Message
2012.04.25	13:28:28	SerialLine	SerialLineMainForm	Info	SerialLine started
2012.04.25	13:28:45	SerialLine	MCD.Framework.Forms.FTDI	Info	Port open
2012.04.25	13:28:46	SerialLine	MCD.Framework.Forms.FTDI	Trace	Write message: 3F
2012.04.25	13:28:48	SerialLine	MCD.Framework.Forms.FTDI	Trace	Read message: 3F V9.99 Video-Analyser/Generator SD M

Log Monitor

Trace

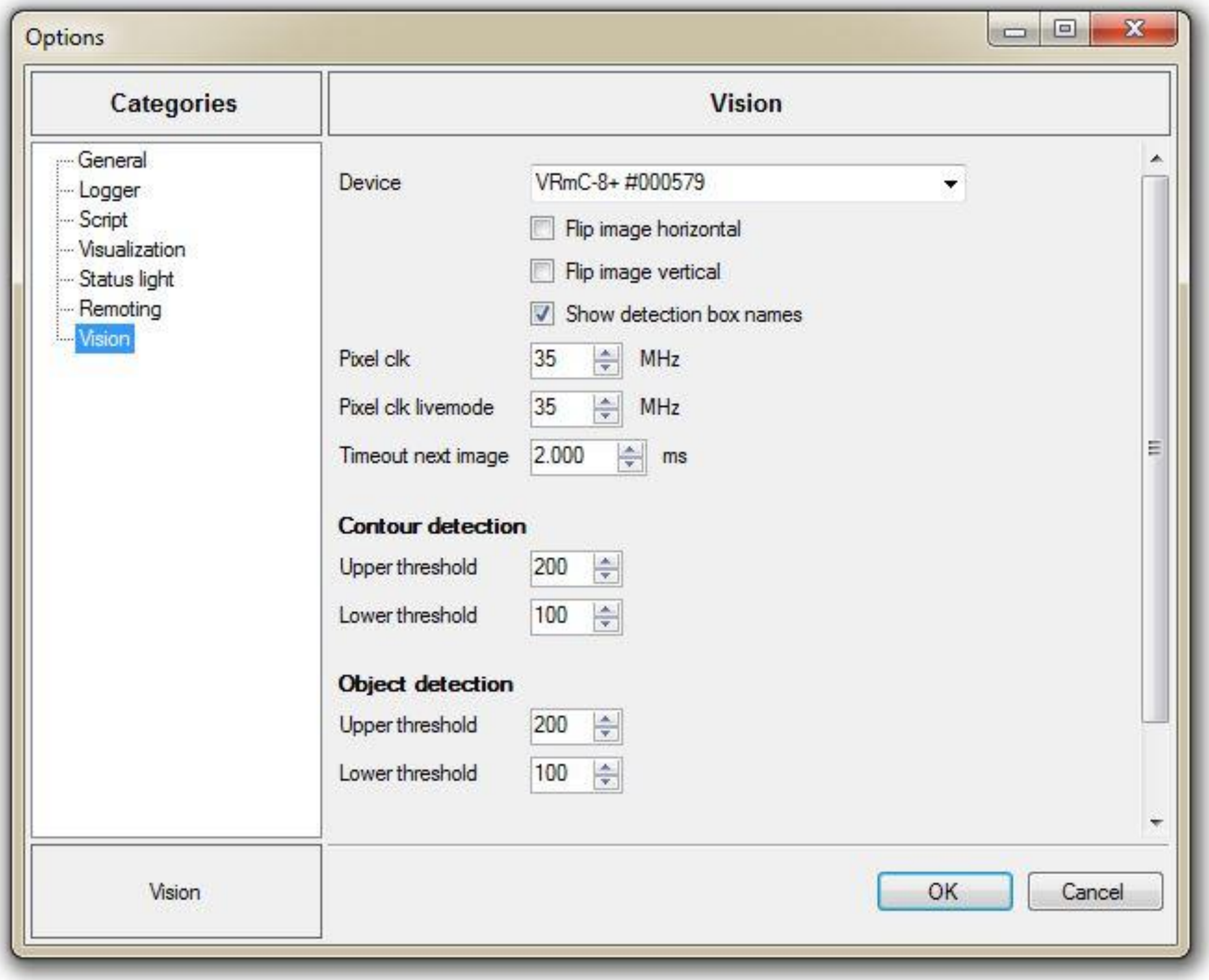
Using this menu, the trace messages view (sent and received messages) is called up.



```
13:31:15.7 <-- 3F [Read firmware]
13:31:15.7 --> 3F V9.99 Video-Analyser/Generator SD MCD-Elektronik GmbH 2012-03-07 [Re
13:31:15.9 <-- 51 00 [Select input FBAS 1]
13:31:15.9 --> 51 [Select input FBAS 1]
13:31:16.1 <-- 74 [Status und erkannte Eingangsnorm lesen]
13:31:16.1 --> 74 3A [Status und erkannte Eingangsnorm lesen]
```

Trace

4.2. Setup



Device: VRmC-8+ #000579

□ Flip image horizontal

□ Flip image vertical

☑ Show detection box names

Pixel clk: 35 MHz

Pixel clk livemode: 35 MHz

Timeout next image: 2.000 ms

Contour detection

Upper threshold: 200

Lower threshold: 100

Object detection

Upper threshold: 200

Lower threshold: 100

Options for Toolmonitor Vision

4.2.1. Device

All detected VRMagic cameras are listed in the Device selection window. The desired camera can be selected. If several cameras are connected to the PC, several presets can be created. The respective device should then be selected in every preset to work with the different cameras.

4.2.2. Flip image

Using the Flip image horizontal and Flip image vertical properties, the camera image can be mirrored horizontally or vertically. This function is helpful, when correcting an image rotated due to the installation position.

4.2.3. Show detection box names

Using the Show detection box names parameter, displaying the detection box names in the Detection image window can be activated and deactivated. For example, this function can be used for more clarity when using many boxes.

4.2.4. Pixelclock / Pixelclock livemode

Pixelclock indicates the speed, at which the individual sensor pixels are read into the camera. In the case of single image capturing, the Pixelclock parameter decisively defines the speed, at which an image is captured (depending on the image resolution). Distortions occur in live mode with a slow Pixelclock.

If cameras with a very high resolution (e.g., 3054x1542) are used, errors can occur during image transfer depending on the PC hardware. If an image cannot be transferred, an error is trigged and displayed in the Toolmonitor Vision. In this case, the Pixelclock parameter should be reduced until system stability is ensured

4.2.5. Timeout next image

This parameter indicates the waiting time for a new image from the camera. If no image can be captured/transferred by the camera, an exception is triggered and displayed in the Toolmonitor Vision.

4.2.6. Contour detection / Object detection

The parameters in this area are used for edge detection during contour and object detection. These parameters do not require any changes by the end user

5. Programming

5.1. Camera Control

5.1.1. Camera Parameters

Names for setting parameters for the camera control module.

Use these names to apply settings in the camera control module.
Example:

```
SetValue("Camera control.ExposureTime");
```

Enumerated values:

ExposureTime	Adjustment of the exposure time (0 ms - 100 ms)
SaveImage	Save recorded image (formats: bmp, jpg, png, and gif), handover path with file name, e.g.: C:/Pictures/picture.jpg.

5.1.2. Control Events

Names for controlling the Toolmonitor Vision.

Using these names, the system control actions are triggered. Events are accessed without handover parameters. For this purpose, the desired name of the Enum entry is handed over to the SetEvent () function as string.

Example:

```
SetEvent("Camera control.Connect");
```

Enumerated values:

Connect	Establish connection with the connected camera.
Disconnect	Disconnect connection with the connected camera.
AcquireImage	Acquire an image with the camera. The Live mode button should be deactivated in production mode.
AcquireAndAnalyze	Acquire an image and run all activated image-processing tools.

5.1.3. Control Values

Names for reading out status information of the camera control module.

Different status information can be read out using these names. For this purpose, the desired name of the Enum entry is handed over to the GetValue() function as string as shown in the example.

Getting measured values:

```
GetValue("Camera control.IsConnected");
```

Enumerated values:

IsConnected	Indicates, whether a connection was established with a camera.

5.2. Brightness Detection

5.2.1. Measurement Values

Names for reading out the results of the gray level detection.

The measured values can be read out using these names. For this purpose, the desired name of the Enum entry is handed over to the GetValue() function as string as shown in the example.

Getting measured values:

```
GetValue("BrightDetectBox.MeanGrayscale");
```

Enumerated values:

MeanGrayscale	8-bit gray value, mean gray value of the region of interest (0 - 255 / black - white)

5.2.2. Settings

Names for setting parameters for the gray level detection.

Use these names to apply settings for the gray level detection.
Example:

```
SetValue("BrightDetectBox.Color", "green");
```

Enumerated values:

Color	Using this name, the color of the desired polygon can be adjusted. (green/red/default)

5.3. Color Detection

5.3.1. Measurement Values

Names for reading out the results of the color detection.

The measured values can be read out using these names. For this purpose, the desired name of the Enum entry is handed over to the GetStream() function as string as shown in the example.

Getting measured values:

```
GetStream("ColorDetectBox.Rgb");
```

Enumerated values:

Rgb The determined RGB value is returned as array, index: 0 = R, 1 = G, 2 = B.

Hsl The determined HSL value is returned as array, index: 0 = H, 1 = S, 2 = L.

5.3.2. Measurement Value Strings

Names for reading out the results of the color detection.

The results of the color detection can be read out using these names. For this purpose, the desired name of the Enum entry is handed over to the GetValue() function as string as shown in the example.

Getting measured values:

```
GetValue("ColorDetectBox.ColorName");
```

Enumerated values:

ColorName Returns the color name of the color from the color list, which was detected for the respective detection box.

5.3.3. Settings

Names for setting parameters for the color detection.

Use these names to apply settings for the color detection.
Example:

```
SetValue("ColorDetectBox.Color", "green");
```

Enumerated values:

Color Using this name, the color of the desired polygon can be adjusted. (green/red/default)

5.4. Contour Detection

5.4.1. Measurement Values

Names for reading out the results of the contour detection.

The measured values can be read out using these names. For this purpose, the desired name of the Enum entry is handed over to the GetValue() function as string as shown in the example.

Getting measured values:

```
GetValue("ContourDetectBox.ContourInside");
```

Enumerated values:

ContourInside Number of contours found inside the desired detection box.

ContourOutside Number of contours found outside the desired detection box.

ContourIntersection Number of contours found that cross an edge of the desired detection box.

5.4.2. Settings

Names for setting parameters for the image analysis.

Use these names to apply settings for the object detection.
Example:

```
SetValue("ObjectDetectionBox1.Color", "green");
```

Enumerated values:

Color Using this name, the color of the desired polygon can be adjusted. (green/red/default)

5.5. Object Detection

5.5.1. Measurement Values

Names for reading out the results of the object detection.

The measured values can be read out using these names. For this purpose, the desired name of the Enum entry is handed over to the GetStream() function as string as shown in the example.

Getting measured values:

```
            GetStream("ObjectDetect.Position");
```

Enumerated values:

Position Read out the positions (centroid of the area) of the detected objects ([x0,y0,...,xn,yn])

Area Read out the area of the detected objects ([a1,a2,...,an])

UndefinedObjectCount Read out the number of all undefined objects (centroid of area outside of image or area)

ObjectCount Read out the number of all found and valid objects for a region of interest.

5.5.2. Settings

Names for setting parameters for the image analysis.

Use these names to apply settings for the object detection.
Example:

```
            SetValue("ObjectDetectBox.Color", "green");
```

Enumerated values:

Color Using this name, the color of the desired polygon can be adjusted. (green/red/default)