

OpenCV provider  
DENSO robot vision  
(DENSO Robot Imaging Library)

Version 1.5.4

User's guide

February 25, 2015

**ATTENTION:**

DENSO WAVE doesn't assume the responsibility of any problems caused by the mistranslation of this document.

**【Revision History】**

| Version | Date       | Content                                                                                                                           |
|---------|------------|-----------------------------------------------------------------------------------------------------------------------------------|
| 1.0.0.0 | 2007-01-30 | First edition.                                                                                                                    |
| 1.0.1.0 | 2007-04-06 | Added several OpenCV commands; OcvTester bugs fixed.                                                                              |
| 1.1.0.0 | 2007-08-06 | Added several new commands for pattern matching; the start image ID was changed from 0 to 1.                                      |
| 1.2.0.0 | 2007-11-21 | Added triangulation functions and several OpenCV commands.                                                                        |
| 1.2.1.0 | 2007-12-05 | Improve the camera setting functions; Added blob commands.                                                                        |
| 1.2.2.0 | 2008-02-06 | Added put/get commands of a video control mode, 'Database' option, and Error table.                                               |
| 1.3.0.0 | 2008-02-29 | Added Inner and Outer product commands.                                                                                           |
| 1.3.1.0 | 2008-05-07 | Added Blob commands. Improve MatchTemplate functions.                                                                             |
| 1.3.2.0 | 2008-09-17 | Added Trim commands. Improve MatchTemplate2, MatchShapes2 and Undistort2 functions.                                               |
| 1.3.3.0 | 2009-01-20 | Added Blob commands, calibration wizard to OcvTester, message transfer function and default camera function.                      |
| 1.3.4.0 | 2009-06-03 | Added several commands of the CaoController::Execute method.                                                                      |
| 1.3.5.0 | 2009-10-07 | Added various commands, unify the rotation direction for all commands, eliminate CamShit command, error code, and sample program. |
| 1.4.0.0 | 2010-7-01  | Mounting of CaoController::get_VariableNames.                                                                                     |
| 1.4.1.0 | 2010-11-02 | Append a camera ID parameter to camera calibration commands.                                                                      |
| 1.4.1.1 | 2011-06-13 | Added the capture screen.                                                                                                         |

|         |            |                                                                                                                                           |
|---------|------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| 1.4.2.0 | 2012-01-11 | Added various commands. Correct SetRobCalDat.                                                                                             |
| 1.4.2   | 2012-07-17 | Document versioning rule was changed.                                                                                                     |
| 1.4.3   | 2012-09-06 | Added the recovery function of the database file (mdb).                                                                                   |
|         | 2012-09-20 | Added direction for "SetCameraCtrl".                                                                                                      |
| 1.4.4   | 2012-10-23 | Added commands.<br>"GetCameraFormatList", "GetCameraFormat", "SetCameraFormat"                                                            |
|         | 2012-12-05 | [Bug Fix] Memory leak in message transfer function.                                                                                       |
| 1.4.5   | 2013-03-13 | Added various commands.<br>SetCameraFrameRate, GetCameraFrameRate, IsUpdated, ClearUpdated                                                |
|         | 2013-04-01 | Added Original Error Code.<br>Mounting of CaoFile::get_Attribute.                                                                         |
|         | 2013-06-26 | Add CaoWorkspace::AddController method option<br>FrameRate                                                                                |
| 1.5.0   | 2013-07-22 | Added FrameRate option and extended camera function<br>Change name: OriN Vision > DENSO Robot Imaging Library<br>Correct BlobMatchShapes. |
|         | 2013-09-24 | Added CARD commands.                                                                                                                      |
| 1.5.1   | 2013-11-18 | Added extended camera commands.                                                                                                           |
| 1.5.2   | 2014-05-27 | Added CARD commands.<br>Added extended camera commands.<br>Canon WebView Livescope camera correspondence.                                 |
| 1.5.3   | 2014-10-01 | Added CARD commands.                                                                                                                      |
| 1.5.4   | 2015-02-25 | [Bug Fix] ADO<br>Change error code.                                                                                                       |

**【Hardware】**

| Model | Version | Notes |
|-------|---------|-------|
|       |         |       |
|       |         |       |
|       |         |       |

**【Attention】**

**Additional license for " DENS0 Robot Imaging Library " is required to use this provider.**

## Contents

|                                                         |    |
|---------------------------------------------------------|----|
| 1. Introduction.....                                    | 8  |
| 1.1. Installing license.....                            | 9  |
| 2. Outline of provider .....                            | 10 |
| 2.1. Outline.....                                       | 10 |
| 2.1.1. Image memory.....                                | 13 |
| 2.1.2. Calibration .....                                | 13 |
| 2.1.3. Triangulation.....                               | 19 |
| 2.1.4. Message transfer function .....                  | 21 |
| 2.2. Method and Property.....                           | 22 |
| 2.2.1. CaoWorkspace::AddController method .....         | 22 |
| 2.2.2. CaoController::AddCommand method.....            | 23 |
| 2.2.3. CaoController::AddFile method .....              | 23 |
| 2.2.4. CaoController::AddVariable method .....          | 23 |
| 2.2.5. CaoController::Execute .....                     | 23 |
| 2.2.6. CaoController::get_ VariableNames property ..... | 24 |
| 2.2.7. CaoCommand::Execute method.....                  | 24 |
| 2.2.8. CaoCommand::put_ Parameter property.....         | 24 |
| 2.2.9. CaoCommand::get_ Parameter property.....         | 24 |
| 2.2.10. CaoCommand::get_ Result property .....          | 24 |
| 2.2.11. CaoFile::Execute method .....                   | 24 |
| 2.2.12. CaoFile::get_ Attribute property.....           | 24 |
| 2.2.13. CaoFile::put_ ID property .....                 | 25 |
| 2.2.14. CaoFile::get_ ID property .....                 | 25 |
| 2.2.15. CaoFile::get_ DateLastModified property .....   | 25 |
| 2.2.16. CaoFile::Get_ Size property .....               | 25 |
| 2.2.17. CaoFile::put_ Value property .....              | 25 |
| 2.2.18. CaoFile::get_ Value property .....              | 25 |
| 2.2.19. CaoFile::get_ Help property.....                | 25 |
| 2.2.20. CaoController::OnMessage event .....            | 25 |
| 2.3. Variable list.....                                 | 26 |
| 2.3.1. Controller class.....                            | 26 |
| 2.3.2. File class .....                                 | 26 |
| 2.4. Error code .....                                   | 27 |
| 3. Sample program.....                                  | 28 |
| 3.1. CaoScript sample program .....                     | 28 |
| 3.2. Other sample programs .....                        | 28 |

---

|                                                                 |            |
|-----------------------------------------------------------------|------------|
| <b>4. Command Reference .....</b>                               | <b>30</b>  |
| 4.1. Controller class .....                                     | 35         |
| 4.1.1. Video setting.....                                       | 35         |
| 4.2. File class .....                                           | 46         |
| 4.2.1. General.....                                             | 46         |
| 4.2.2. Edit .....                                               | 53         |
| 4.2.3. Filter .....                                             | 58         |
| 4.2.4. Mask.....                                                | 73         |
| 4.2.5. Draw .....                                               | 79         |
| 4.2.6. Contours.....                                            | 86         |
| 4.2.7. Blob .....                                               | 92         |
| 4.2.8. Histogram .....                                          | 99         |
| 4.2.9. Matching.....                                            | 102        |
| 4.2.10. CARD .....                                              | 109        |
| 4.2.11. CAL.....                                                | 112        |
| 4.2.12. Misc. ....                                              | 128        |
| 4.3. Command class .....                                        | 138        |
| 4.3.1. Triangulation.....                                       | 138        |
| <b>5. OcvTester .....</b>                                       | <b>144</b> |
| 5.1. Outline.....                                               | 144        |
| 5.2. Main screen .....                                          | 145        |
| 5.2.1. Object window .....                                      | 145        |
| 5.2.2. Log window .....                                         | 145        |
| 5.2.3. Menu .....                                               | 146        |
| 5.3. Image window .....                                         | 148        |
| 5.4. DENSO Robot connection window .....                        | 151        |
| 5.5. Camera Settings window .....                               | 153        |
| 5.6. Triangulation window.....                                  | 153        |
| 5.7. Calibration Wizard.....                                    | 155        |
| 5.7.1. Overview .....                                           | 155        |
| 5.7.2. Step 0: Select calibration target.....                   | 156        |
| 5.7.3. Step 1 : Set camera calibration parameter .....          | 157        |
| 5.7.4. Step 2 : Acquire chessboard image.....                   | 158        |
| 5.7.5. Step 3 : Map world coordinate and robot coordinate ..... | 159        |
| 5.7.6. Step 4 : Complete Wizard.....                            | 160        |
| 5.8. Lookup table editor .....                                  | 160        |
| 5.9. Image sampling window.....                                 | 161        |

---

---

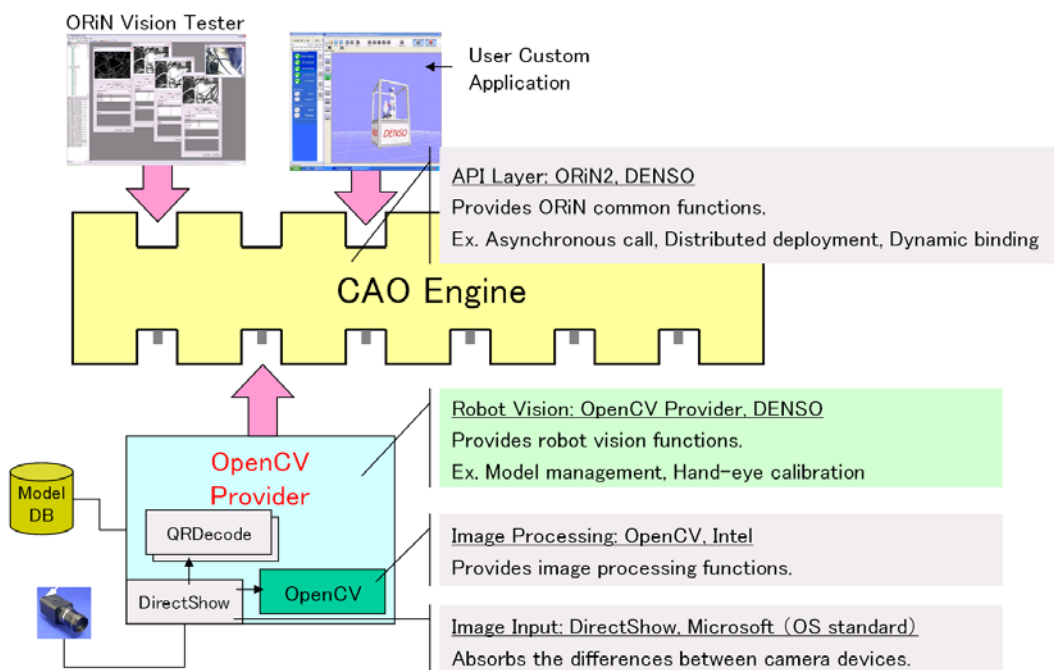
|                                                                                   |     |
|-----------------------------------------------------------------------------------|-----|
| 5.10. Haar training window.....                                                   | 163 |
| Appendix A. OpenCV method implementation list.....                                | 166 |
| Appendix B. uVision21 equivalent OpenCV method .....                              | 181 |
| Appendix C. Intel License Agreement For Open Source Computer Vision Library ..... | 185 |

## 1. Introduction

The OpenCV provider is a provider that uses OpenCV image processing library. The image-processing library is developed by Intel Corporation.

DENSO Robot Imaging Library(RIL) is a tool set combining ORiN (Open Robot interface for the Network) and OpenCV (Open Source Computer Vision Library)<sup>1</sup> as a platform of Hand-eye system. The platform is not only just combining those two technologies but also fusing it into one ORiN provider. Therefore users can make a robot vision application program based on ORiN programming model easily. In addition, common functions such as model management are implemented in the provider module. This means that RIL provides the good balance of high speed processing and usability.

The goal of RIL is to provide the environment which an application engineer can make a hand-eye system easily with low-cost. To achieve this goal, RIL consists of only software, and it can use many cameras on the market. The architecture of RIL is shown below.



**Figure 1-1 Architecture of RIL**

<sup>1</sup> OpenCV is a vision library developed by Intel, and it's opened under the License Agreement (Appendix D).



## 1.1. Installing license

To use OpenCV Provider, you need to install ORiN2 SDK, and also need to input "DENSO Robot Imaging Library" license information. If you would like to install it for evaluation, please use the following license.

**CVG3-MZPB-7W2G-L43Q** (valid for 3 months)

How to add the license is as follows.

1. Run the CaoConfig tool from the [Start] menu, and select the [Cao Provider] tab.
2. Select the [OpenCV CAO Provider] item on the provider list.
3. Click the [...] button of the license input box.
4. Click the [Add] button in the "ORiN2 License Manager" window.
5. Input a license key, and click the [OK] button.
6. Click the [Close] button to exit.

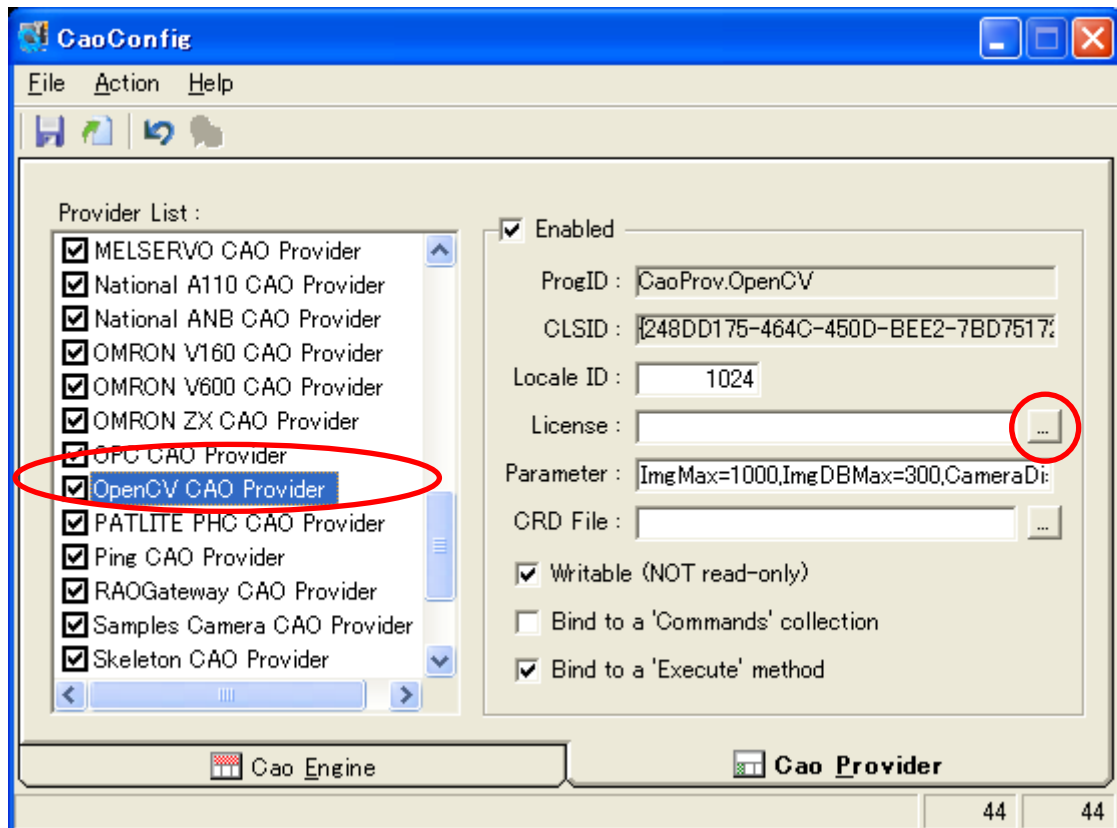
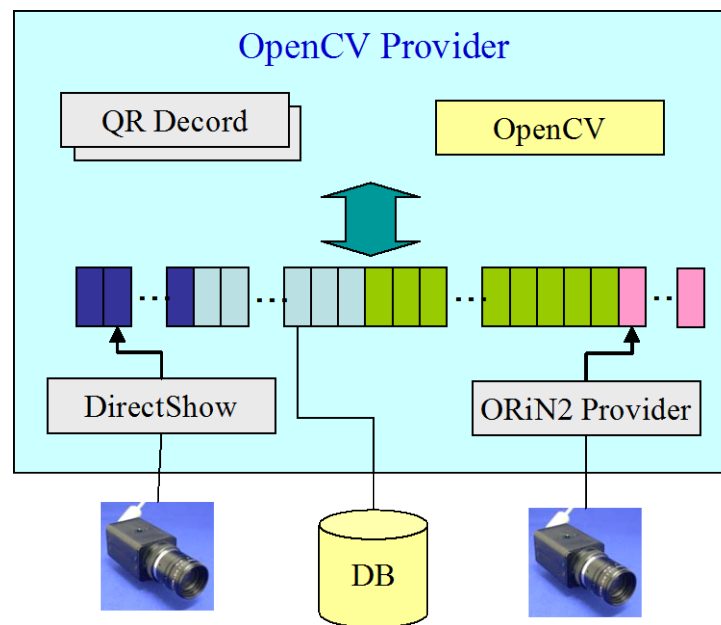


Figure 1-2 Installing 'DENSO Robot Imaging Library' license

## 2. Outline of provider

### 2.1. Outline

The OpenCV provider acquires the image from the imaging device using DirectShow. OpenCV processes the acquired image according to the user application command. Therefore many kinds of imaging device supporting DirectShow on the market can be used with this provider. The processed image is stored in the image memory. Details of the image memory are described later.



**Figure 2-1 System Configuration**

Certain types of cameras can be used as extended cameras. ORiN2 provider dedicated for each camera can configure the detailed setting of extended cameras.

ORiN2 provider used by extended camera may require the manufacturer's driver. For detailed information, refer to manuals of each provider. When you install the driver, do not install the driver of DirectShow.. <sup>2</sup>

The following two types of cameras can be recognized as extended camera..

**Table 2-1 List of compatible models for extended camera**

|                   |                                                               |
|-------------------|---------------------------------------------------------------|
| Extended camera 1 | Basler GigE camera<br>ORiN2¥CAO¥ProviderLib¥Basler¥Pylon¥GigE |
| Extended camera 2 | IDS uEye camera<br>ORiN2¥CAO¥ProviderLib¥IDS¥uEye             |

<sup>2</sup> If DirectShow driver is installed, the camera will not be recognized as an extended camera

|                   |                                                                       |
|-------------------|-----------------------------------------------------------------------|
| Extended camera 3 | Canon webView Livescope camera<br>ORiN2¥CAO¥ProviderLib¥Canon¥WebView |
|-------------------|-----------------------------------------------------------------------|

The file format of OpenCV provider is DLL (Dynamic Link Library); that is automatically loaded from CAO engine when it is used. To use OpenCV provider, install ORiN2SDK or manually register the registry in reference with the following table.

**Table 2-1 OpenCV provider**

|                                    |                               |
|------------------------------------|-------------------------------|
| File name                          | CaoProvOpenCV.dll             |
| ProgID                             | CaoProv.OpenCV                |
| Registry registration <sup>3</sup> | Regsvr32 CaoProvOpenCV.dll    |
| Registry unregistration            | Regsvr32 /u CaoProvOpenCV.dll |

OpenCV provider uses registry to store configurations of image memory area and cameras. The configuration is changed with CaoConfig software. The following table shows configuration items.

---

<sup>3</sup> Manual registration/unregistration is not necessary if the provider is installed by ORiN SDK.

**Table 2-2 CaoConfig Parameter strings**

| Parameter                                                                                                | Meaning                                                                                                                                                                                                                                    |
|----------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Database=<Database file name>                                                                            | Specify an absolute file path of ACCESS database. If omitted, the system uses the default database. That is 'opencv_en.mdb' stored in the bin directory. Please use the following file as a template.<br><OpenCV>%Bin%opencv_master_en.mdb |
| DefaultCamera=<default camera ID>                                                                        | Specify default camera ID (default: 1).<br>If ID option is omitted for AddFile, Default Camera ID is used for file object creation.                                                                                                        |
| ImgMax=< image memory size >                                                                             | Specify the entire image memory size. (default: 200)                                                                                                                                                                                       |
| ImgDBMax=< database area size >                                                                          | Specify the image memory database area size. (default: 100)<br>If the specified value is larger than ImgMax, all image memory is assumed to be stored in database.                                                                         |
| CameraDisable<br>[=<disabled camera>]                                                                    | Specify the disabled cameras. Bit0 corresponds to camera ID=0, and Bit0 to Bit9 is specified. (default: 0x00)<br>big ON: camera disabled<br>bit OFF: camera enabled                                                                        |
| FormatType<br>=<camera0 display size>:<br>[<camera1 display size>:<br>...<br>[<camera9 display size>: ]] | Specify the display size of each camera.<br>(default: 0: 0: 0: 0: 0: 0: 0: 0: 0)<br>-1:default<br>The value range depends on the camera. (0~)<br>If an invalid value was specified, the default value is used instead.                     |
| FrameRate<br>=<camera0 frame rate>:<br>[<camera1 frame rate>:<br>...<br>[<camera9 display size>: ]]      | Specify the frame rate of each camera.<br>(default: 0: 0: 0: 0: 0: 0: 0: 0: 0)<br>If an invalid value was specified, the default value is used instead.                                                                                    |
| ExtCamera<br>[=<Basler GigE camera count >:<br>[<IDS uEye camera count>]]                                | Specify the number of extended camera.<br>(default: 0:0)                                                                                                                                                                                   |

### 2.1.1. Image memory

OpenCV provider can store the image data in the two types of data area, database area and memory area. The data stored in the database area remains even if OpenCV provider unloaded (non-volatile), but the access speed is very slow. Therefore please don't use database area for a temporally use. On the other hand, the access speed of the memory area is fast, but the data stored in the area is cleared at the end (Volatile).

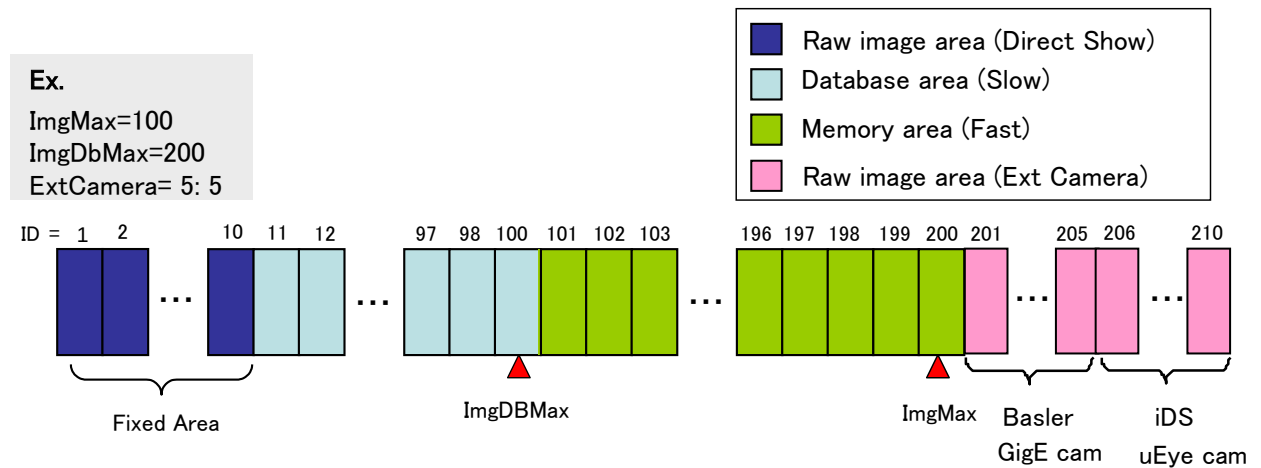


Figure 2-2 Image storage area

### 2.1.2. Calibration

The calibration commands of OpenCV provider can be divided into two categories, camera calibration and robot-camera calibration. Those commands are as follows:

- Camera calibration:

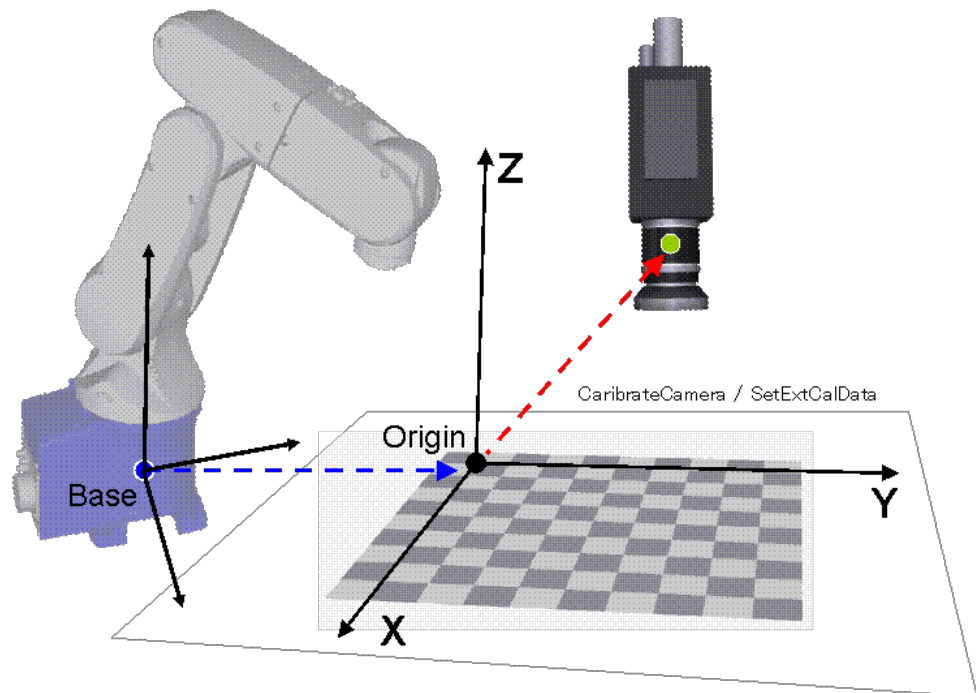
CalibrateCamera, GetCamCalData, SetCamCalData, GetCamCalExtData, SetCamCalExtData, GetPosFromCam, GetCamPos, Undistort2

- Robot-calibration:

CalibrateRobot, GetRobCalData, SetRobCalData, GetPosFromRob, GetRobPos

The relation among three coordinates, the world coordinate, camera coordinates and robot coordinates, are shown below: where

- The origin of the world coordinate is "Origin".
- The origin of a robot coordinate is "Base".



**Figure 2-3 Robot -Camera Calibration**

The triangulation commands of CaoCommand class need to be done the camera calibration. The procedure is as follows.

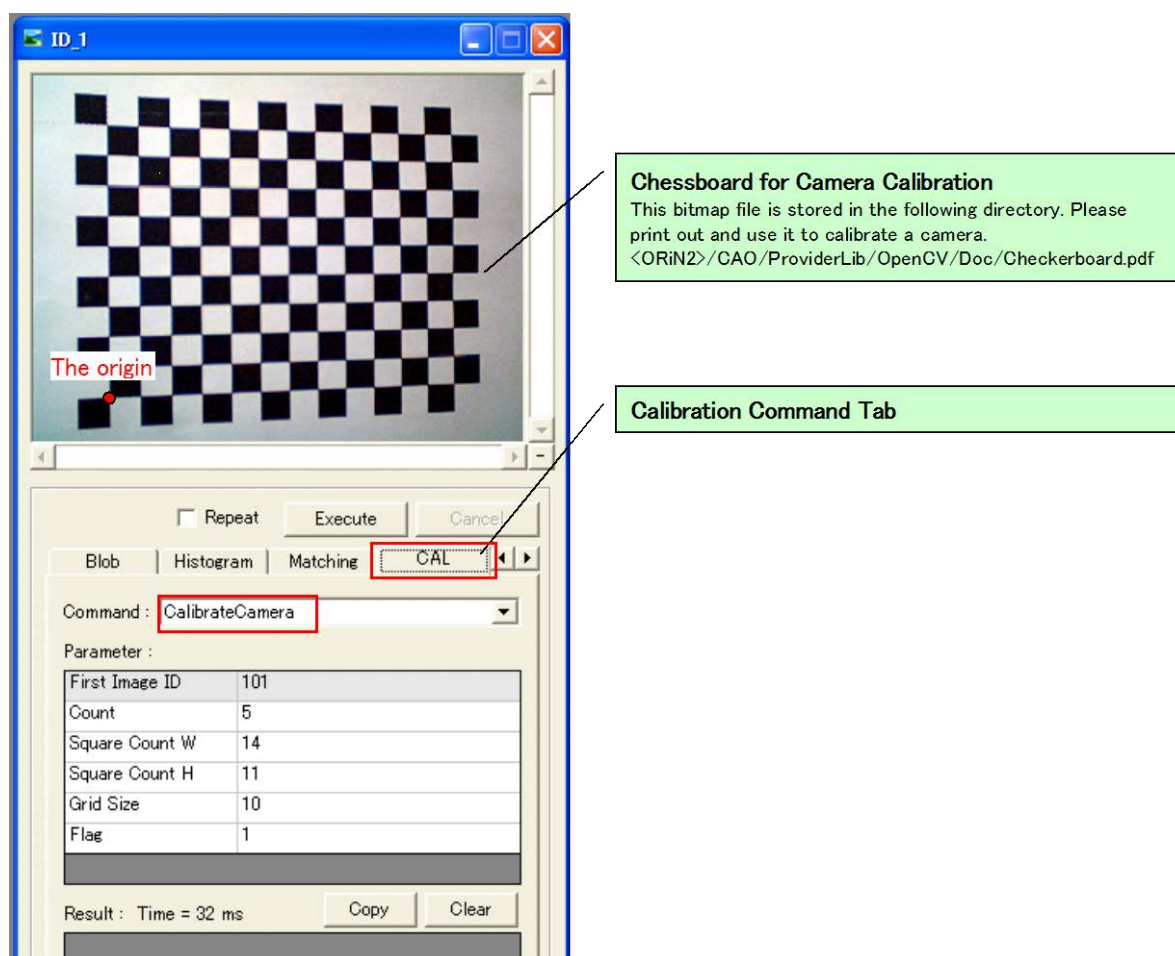
#### **2.1.2.1. Camera calibration**

OpenCV Provider provides a simple calibration method using a chessboard to calibrate the intrinsic parameters and the extrinsic parameters of the camera. The procedure is as follows.

##### **[Step 1] Prepare chessboard image**

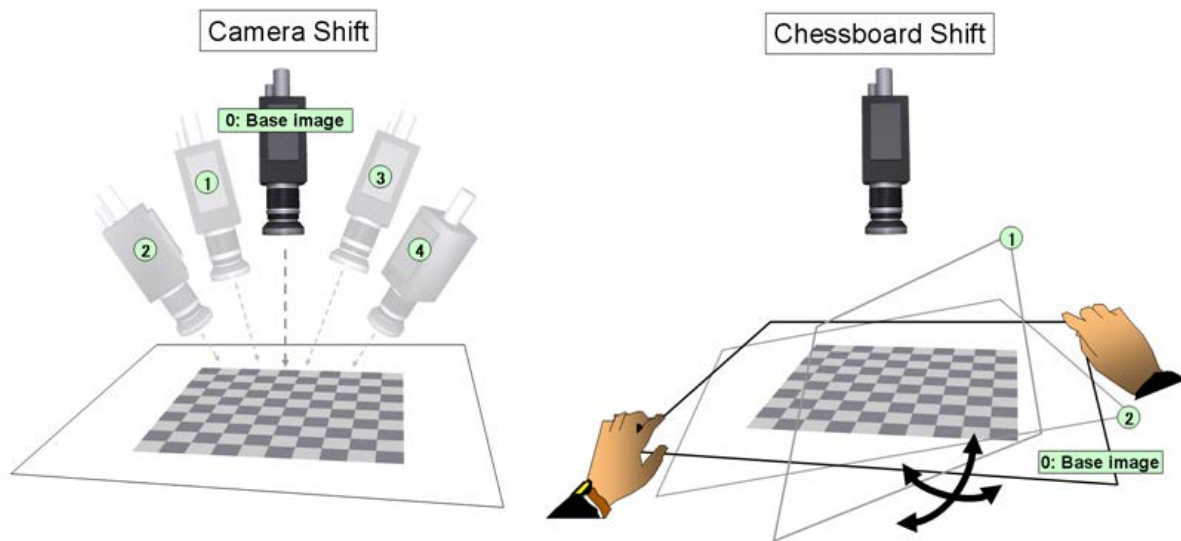
Each camera requires at least five chessboard images. Store these images into appropriate consecutive image area, e.g. 101-105. The first image is used as a reference image to calculate external parameter. Input the reference image ID into 'Input ID' parameter of 'CalibrateCamera' command. The plane on which the chessboard is placed when picturing the reference image is called as reference plane(Figure 2-4). The following directory of ORiN2 SDK stores the chessboard image that can be used for calibration.

<ORiN2>/CAO/ProviderLib/OpenCV/Doc/Chessboard.pdf

**Figure 2-4 Calibration Dialogue**

There are two ways to take 5 images, a camera shift and a chessboard shift (Figure 2-5). In both cases, the right-top corner of the left-bottom "**black box**" becomes the origin.<sup>4</sup>

<sup>4</sup> In case of a chessboard which there is no black box in the left-bottom corner, the origin may become the right-top corner. Please confirm the origin by GetCamPos after executing CalibrateCamera.



**Figure 2-5 Two ways to take chessboard images**

[Step 2] Calculate camera parameters

Use 'CalibrateCamera' command to calculate camera parameters. When you use chessboard file image in SDK, please specify the values for 'Square count W', 'Square count H' and 'Grid size (mm)' as written on each image.

After setting these values, press [Execute] button (if you use OpenCV tester), or invoke 'CalibrateCamera' command (if you directly call the command) for camera parameter calculation. The calculated data is automatically stored in database. Use 'GetCamCalDat' command to get the stored parameter value.

[Step 3] Confirm the result

You can get the correct value (X, Y, Z) in the world coordinate after calibration. Please perform 'GetPosFromCam' command with a reasonable value (Xc, Yc) in the camera coordinate, and then (Xw, Yw, Zw) corresponding (Xc, Yc) returns. In case of two points on the world coordinate, the distance of two points can be calculated correctly.

In addition, by using 'Undistort2' command, a distorted image can be undistorted. Please check whether the linearity of the image was improved.

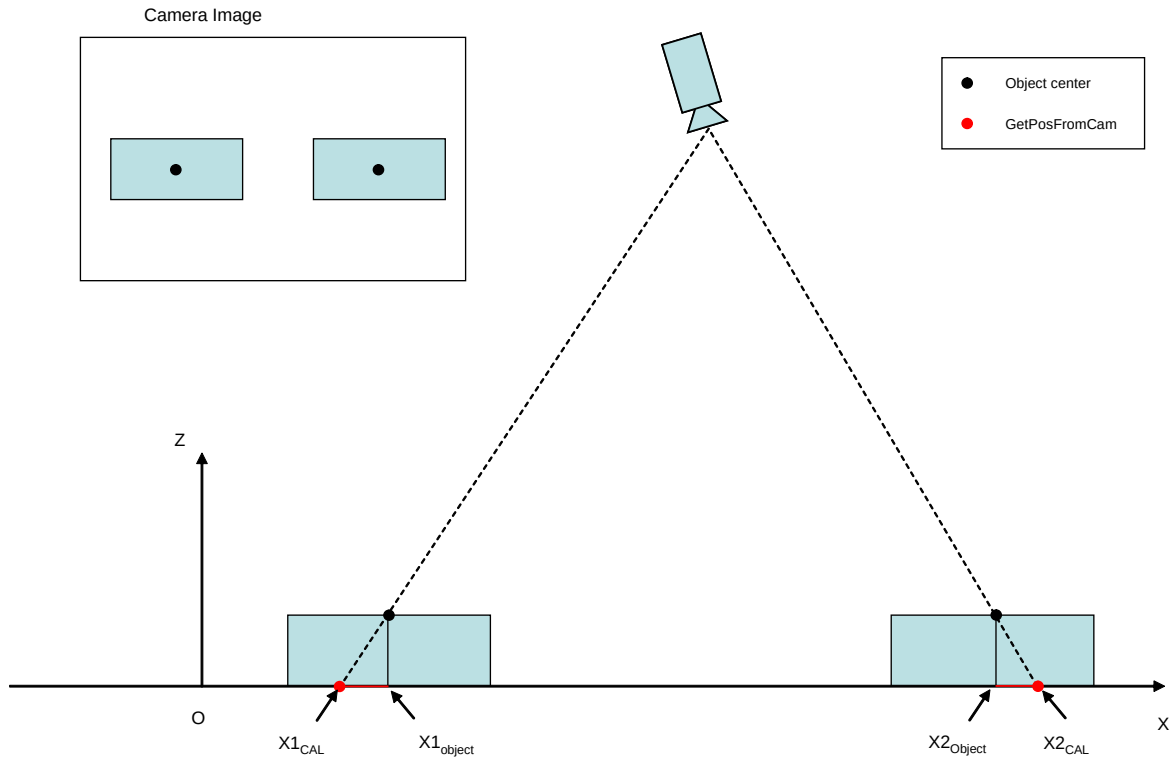
[Caution]

GetPosFromCam function assumes that the target point is on a plane where  $Z=0$  in the world coordinate system. Therefore, positioning error may occur if the camera detected target point is not on the  $Z=0$  plane (e.g.  $X1obj - X1cal$  in Figure 2-6 shows the positioning error.)

If the target object is a three-dimensional object with height, calibration reference plane should be



aligned to the target object height. For measuring accurate 3D coordinate position, please use triangulation method command.



**Figure 2-6 Camera image and actual position error in camera calibration**

#### 2.1.2.1.1. Distortion correction

The shot image of the camera has distortion, because camera lens also has distortion. Camera calibration calculates the parameters to correct this type of distortion.

‘Undistort’ command transforms distorted image to distortion-corrected image.

‘GetPosFromCam’ command transforms camera coordinate to world coordinate. By assigning distortion correction flag of this command as TRUE, distorted image coordinate is directly converted to world coordinate.

Transformation from distorted image to undistorted image is supported, but undistorted image to distorted image is not supported.

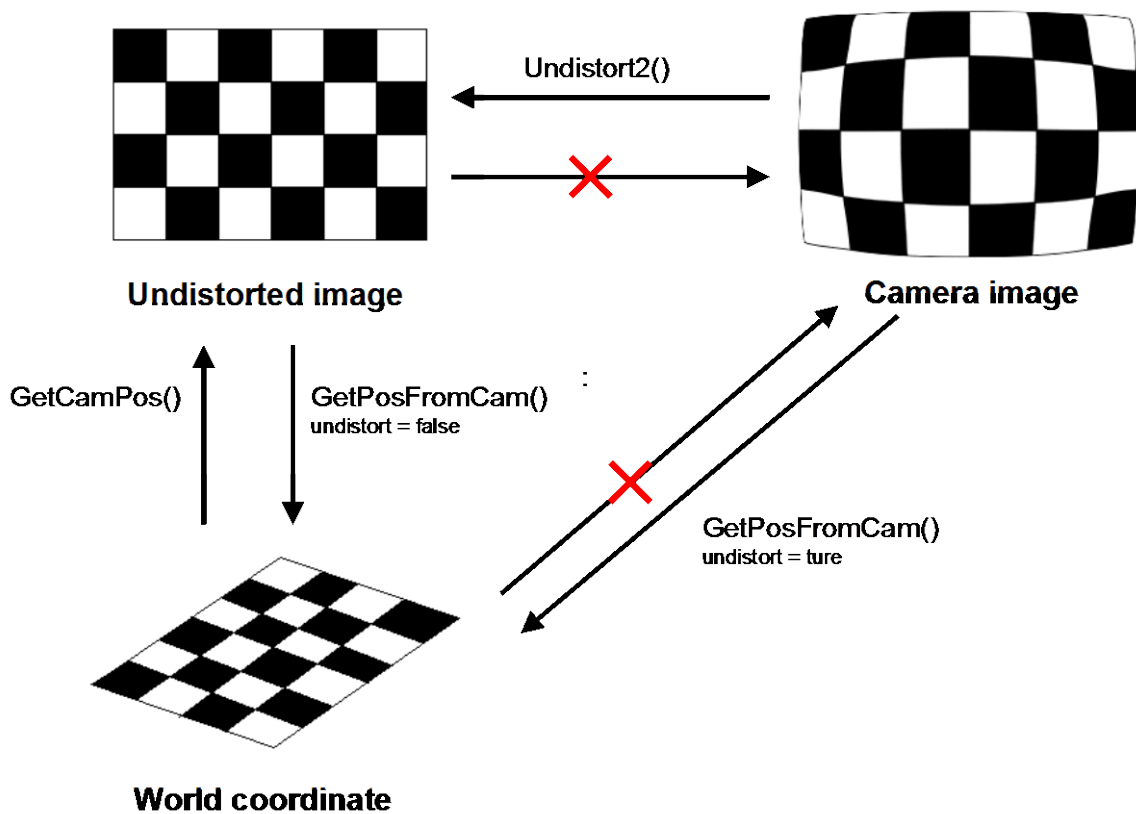


Figure 2-7 Distortion correction commands

### 2.1.2.2. Robot calibration

There are two ways to connect the world coordinate and a robot coordinate:

1. Calculate the robot parameters between the world coordinate and a robot coordinate by using OpenCV provider functions.
2. Set the world coordinate as a work coordinate of the robot.

In this section, the 1st procedure is shown. Regarding the 2nd procedure, please refer to the robot manual. If you don't have to connect the world coordinate and a robot coordinate, the following procedure is not required.

[Step 1] Calculate the robot parameters

Calculate the robot parameters by using 'CalibrateRobot' command. This command requires the following 3 points:

- The origin of the world coordinate.
- One point on the X axis of the world coordinate.
- One point on the X-Y plane of the world coordinate.

Please note that those three points should be the points of a robot coordinate.

After performing the command, the robot parameters are stored in the database. To get the parameters from the database, use 'GetRobCalDat' command.

[Step 2] Confirm the result

You can get the correct value (X, Y, Z) in the world coordinate after calibration. Please perform 'GetPosFromRob' command with a reasonable value (Xr, Yr, Zr) in the robot coordinate, and then (Xw, Yw, Zw) corresponding (Xr, Yr, Zr) returns.

### 2.1.3. Triangulation

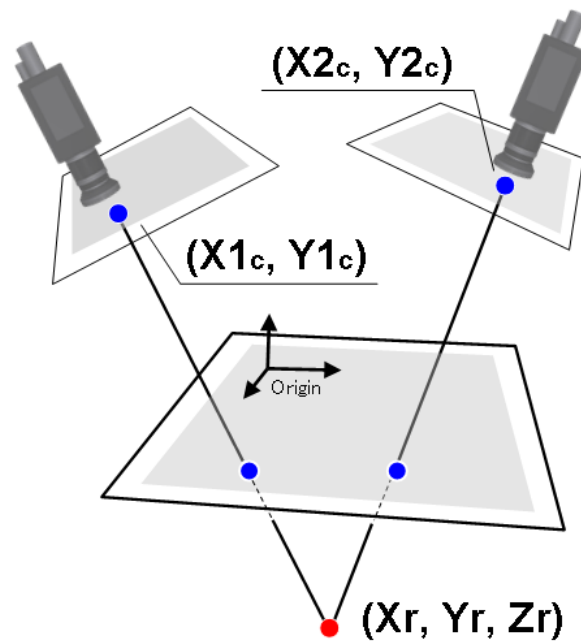
OpenCV provider has triangulation function using two cameras. To use the function, the above described camera calibration and robot-camera calibration need to be completed. There are three commands for triangulation.

Triangulation, TriHaarDetect, TriMatchShapes, TriMatchTemplate

For 'Triangulation' command, two camera image coordinates, (X1c,Y1c) and (X2c,Y2c), are specified for triangulation calculation. (Figure 2-8). For other commands, HaarDetect, MatchShapes2 and MatchTemplate2 commands are used to detect corresponding points automatically, and then 'Triangulation' command uses the detected points. Please notice that the return values of (X, Y) of these three commands have different meanings. For HaarDetect, the adjustment is performed as following.

- HaarDetect: Add half of W and H to the first result of (X, Y), i.e.  $(X + W/2, Y + H/2)$
- MatchShapes2: Use result as it is.
- MatchTemplate2: Use result as it is.

In the same way, you may develop your original triangulation command by combining your original corresponding point detection algorithm and 'Triangulation' command. In the actual application, corresponding point detection can be optimized for each application. If the position accuracy or detection speed is not enough, we recommend to develop original routine for position detection.

**Figure 2-8 Triangulation**

Triangulation command returns the coordinate position from the origin of the reference image. Camera position is also represented using the same origin, and you can easily calculate the distance between the camera and the target object using Distance command.(Figure 2-9).

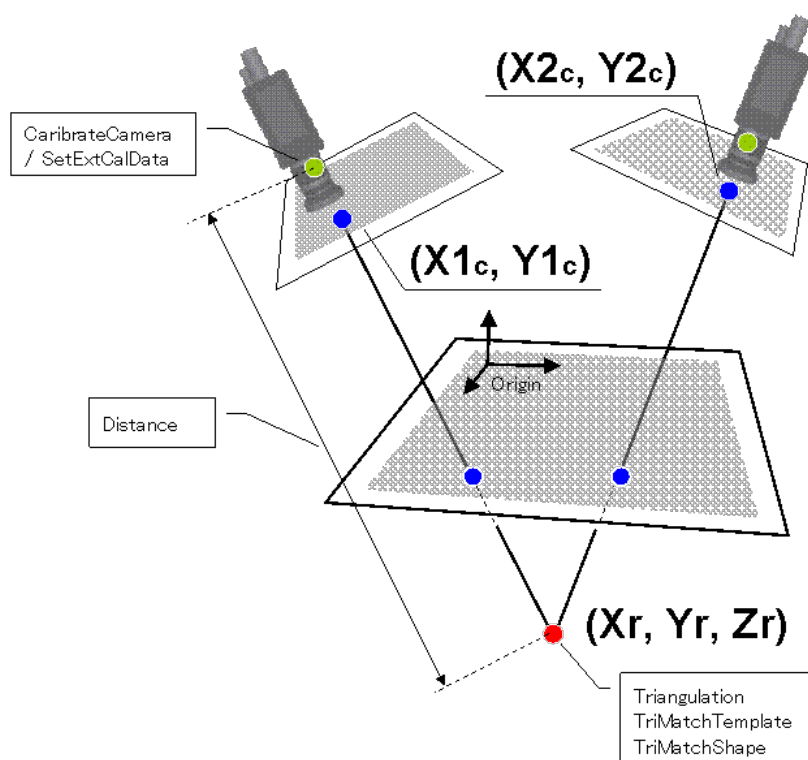


Figure 2-9 Distance calculation

#### 2.1.4. Message transfer function

By using message transfer function of CAO engine, image data stored in received message can be transferred to the specified image ID.

Transfer destination is specified by `MsgDestID` option of `AddController()`. To transfer images from plural sources to different image ID destinations, create plural `CaoController` objects and specify different destination image ID in `AddController()`.

When transfer message data is not bitmap file, the data is not stored.

## 2.2. Method and Property

### 2.2.1. CaoWorkspace::AddController method

The OpenCV provider searches camera and performs connection process at AddController.

If you don't use the option character, it uses setting which registered in registry. (refer 2.1)

**Format.** AddController( <bstrCtrlName: BSTR>,<bstrProvName: BSTR>,  
<bstrPcName: BSTR > [,<bstrOption: BSTR>] )

**bstrCtrlName** : [in] controller-name

```

bstrProvName      : [in] provider name. "CaoProv.OpenCV fixed value ="

```

bstrPcName : [in] provider process execution machine name

bstrOption : [in] option character string

### Table 2-3 option character string table

| Option                                   | Meaning                                                                                                                                                                                      |
|------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| QREnabled=True/False                     | "QRDecode" Command Enable. Default=False                                                                                                                                                     |
| OCREnabled =True/False                   | " OCRead " Command Enable. Default=False                                                                                                                                                     |
| MsgDestID=<image ID>                     | Specify destination image ID for message transfer.                                                                                                                                           |
| FormatType=t1:t2:t3:t4:t5:t7:t8:t9:t10   | Specify the display size of each camera.<br>If an invalid value was specified, the default value is used instead.<br>Ex. Use No2 camera format to camera2.<br>FormatType=0:2:0:0:0:0:0:0:0:0 |
| FrameRate=f1:f2:f3:f4:f5:f6:f7:f8:f9:f10 | Specify the frame rate of each camera.<br>If an invalid value was specified, the default value is used instead.<br>Ex. Use 30 frame rate to camera2.<br>FormatType=0:30:0:0:0:0:0:0:0:0      |

When AddController failed, it might be caused by the following problems.

- Camera device failure

A camera device might not be working properly. Please check the camera with "amcap.exe" program included in the DirectX samples.

- Image database failure

A database file might be broken. Please delete “opencv.mdb” file located in the

“CAO/ProviderLib/OpenCV/Bin” directory. A new database file will be generated automatically at the next startup. But all image data stored in the deleted file are discarded.

### 2.2.2. CaoController::AddCommand method

Create CaoCommand for triangulation.

**Format** AddCommand( <bstrName: BSTR > [,<bstrOption: BSTR>] )

bstrName : [in] command name

bstrOption : [in] option character string(unused)

Refer 4.3.1 for available commands.

### 2.2.3. CaoController::AddFile method

Create a file object to access camera device and the image memory.

**Format** AddFile( <bstrName: BSTR > [,<bstrOption: BSTR>] )

bstrName : [in] arbitrary name

bstrOption: : [in] option character string

**Table 2-4 Option character string of CaoWorkspace::AddFile**

| Option              | Meaning                                                                                                                                                                                                  |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ID[=<image number>] | Initially connected image memory number<br>(default: default camera ID)<br><br>When this option is omitted, file object is connected to the default camera ID specified in “DefaultCamera of Table 2-2 . |

### 2.2.4. CaoController::AddVariable method

Creates variable object of camera device and image memory information. Only variable names on 2.3.1 can be used for this method.

**Format** AddVariable( <bstrName: BSTR > [,<bstrOption: BSTR>] )

bstrName : [in] Arbitrary name

bstrOption : [in] Option character string

### 2.2.5. CaoController::Execute

Execute specified command.

Refer to 4.1 for details of each commands.

### 2.2.6. CaoController::get\_VariableNames property

The variable list is acquired. Please refer to 2.3.1 for the acquired variable.

### 2.2.7. CaoCommand::Execute method

Execute specified command.

Refer to 4.3 for details of each commands.

### 2.2.8. CaoCommand::put\_Parameter property

Set parameters for a command.

Refer to 4.3 for details of each commands. This property does not check illegal parameters.

### 2.2.9. CaoCommand::get\_Parameter property

Get parameters set by 2.2.8 . When the parameter is not set, VT\_EMPTY is Return valued.

### 2.2.10. CaoCommand::get\_Result property

Get latest execution result of 2.2.7. Refer to 4.3.1 for result of each commands.

### 2.2.11. CaoFile::Execute method

Execute image processing or the arithmetic processing specified by the command name.

The arguments of the Execute method are specified by BSTR for command and VARIANT array for parameters.

**Format** [<vntRet: VARIANT> = ] Execute( <bstrCmd: BSTR > [,<vntParam: VARIANT>] )

|          |                      |
|----------|----------------------|
| bstrCmd  | : [in] command       |
| vntParam | : [in] parameter     |
| vntRet   | : [out] Return value |

Refer to “4 Command Reference” for details of each command.

### 2.2.12. CaoFile::get\_Attribute property

Get the kind of image memory.

|        |                         |
|--------|-------------------------|
| 0x0002 | Camera                  |
| 0x0003 | Database area           |
| 0x0004 | Memory area             |
| 0x0100 | Basler GigE Camera area |



|        |                           |
|--------|---------------------------|
| 0x0101 | IDS uEye Camera area      |
| 0x0102 | Canon WebView Camera area |

### 2.2.13. CaoFile::put\_ID property

Change referred image memory.

### 2.2.14. CaoFile::get\_ID property

Get ID of the currently referring image memory.

### 2.2.15. CaoFile::get\_DateLastModified property

Get last modified date of the currently referring image memory.

VT\_EMPTY will be returned if the memory does not have image.

### 2.2.16. CaoFile::Get\_Size property

Get file size of the referring image memory.

### 2.2.17. CaoFile::put\_Value property

Overwrite BMP format image to the currently referring image memory.

The image is overwritten as a color image.

### 2.2.18. CaoFile::get\_Value property

Get BMP format image from the currently referring image memory.

Color image is acquired as 24bits bitmap image, and grayscale image is acquired as 8bits bitmap image.

### 2.2.19. CaoFile::get\_Help property

Get character strings that have been set by the PutHelp command.

If camera area is specified, a camera name will be obtained.

### 2.2.20. CaoController::OnMessage event

Updating image data generates OnMessage event of CaoController class. With this event, Message::Number property is set to 1, and Message::Value property is set to the image number.

## 2.3. Variable list

### 2.3.1. Controller class

**Table 2-5 Controller class system variable list**

| Variable identifier                 | Data type           | Explanation                                                                                                                     | Attribute |     |
|-------------------------------------|---------------------|---------------------------------------------------------------------------------------------------------------------------------|-----------|-----|
|                                     |                     |                                                                                                                                 | get       | put |
| @IMG_MAX                            | VT_I4               | Size of the entire image memory                                                                                                 | √         | -   |
| @IMGDB_MAX                          | VT_I4               | Size of data base area of image memory                                                                                          | √         | -   |
| @CAM_COUNT                          | VT_I4               | Number of connected cameras                                                                                                     | √         | -   |
| @VERSION<br>[V1.3.5 or later]       | VT_BSTR             | Provider version                                                                                                                | √         | -   |
| @EVENT_ENABLED<br>[V1.3.5 or later] | VT_BOOL             | CAO message event generation setting                                                                                            | √         | √   |
| @EXT_CAM_COUNT<br>[V1.4.6 or later] | VT_I4  <br>VT_ARRAY | Number of extended camera.<br><Number of Basler GigE camera><br><Number of IDS uEye camera><br><Number of Canon WebView camera> | √         | -   |

### 2.3.2. File class

**Table 2-2 File class system variable list**

| Variable identifier         | Data type            | Explanation                                                                                                                                        | Attribute |     |
|-----------------------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|-----------|-----|
|                             |                      |                                                                                                                                                    | get       | put |
| @VALUE<br>[V1.3.5 or later] | VT_UI1  <br>VT_ARRAY | Image data in the image memory<br>Referring and assigning the variable has same effect as executing CaoFile::get_Value() and CaoFile::put_Value(). | √         | √   |

## 2.4. Error code

Open CV provider defines following specific error codes. For common error code for ORiN2, please refer to the error code section of "[ORiN2 Programming guide](#)".

**Table 2-3 Specific error code for OpenCV provider**

| Error name           | Error code | Explanation                                                                                                               |
|----------------------|------------|---------------------------------------------------------------------------------------------------------------------------|
| E_CAOP_NO_LICENSE    | 0x80100000 | There is no license. Please purchase an additional license.                                                               |
| E_CAOP_DB_RESTORE    | 0x80100001 | The database file (mdb) was crashed. Because it was recovered automatically from the last mdb, restartd your program.     |
| E_CAOP_INITTERM      | 0x80100002 | The another process is initializing or terminating. Please wait a minute.                                                 |
| E_CAOP_NOIMAGE       | 0x80100003 | No image.                                                                                                                 |
| E_CAOP_LOCK_IMAGE    | 0x80100004 | The another program is using the image.                                                                                   |
| Original error code. | 0x801010xx | The original error code depending on a command.<br>Please refer to the chapter 4.                                         |
| OpenCV API Error     | 0x8011xxxx | OpenCV API error number will be assigned at "xxxx" part of the error code. Please refer OpenCV reference for the details. |

### 3. Sample program

RIL programs can be developed with various programming languages (C/C++, VB, etc.) The easiest way is to use RIL is to use CaoScript, a VB Script based scripting language. CaoScript is included in ORiN2 SDK. Section 3.1 explains CaoScript sample program. For other samples, refer to section 3.2

#### 3.1. CaoScript sample program

This sample program is to detect target position (target image is stored in ID101), and move DENSO robot to the detected position.

```
' Create CAO object
Set rc = Cao.AddController("rc", "CaoProv.DENSO.NetwoRC", "", "Conn=eth:192.168.0.1")
Set robo = rc.AddRobot("vp")
Set vis = Cao.AddController("cv", "CaoProv.OpenCV", "", "")
Set rawImg = vis.AddFile("cam1", "ID=1")
Set tmpImg = vis.AddFile("mem1", "ID=101")
' Search and trace target by using pattern matching
OldX = -1: OldY = -1
Do
    ' Calculate threshold level by discriminant analysis method
    iT = rawImg.AutoThreshDiscrim(rawImg.CalcHistEx(255))
    ' Binarization & B/W inversion(1)
    rawImg.ThresholdEx 101, iT, 255, 1
    ' Shape matching
    res = tmpImg.MatchShapes2(11, 2, 0.2)
    ' Calculate position shift length and move robot
    If (OldX <> -1) Then
        v = "V(" & (OldX - res(0)) & ", " & (OldY - res(1)) & ", 0)"
        robo.Draw 1, v, "next"
    End If
    OldX = res(0): OldY = res(1)
Loop
```

#### 3.2. Other sample programs

Other RIL sample programs are located in the following directory.

<ORiN2>\¥CAO¥ProviderLib¥OpenCV ¥Samples

**Table 3-1 Sample program list**

| Program    | Contents                                                                                                                                                  | Language          |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| 3DTracking | Robot motion by pattern matching and triangulation.                                                                                                       | Visual Basic 6    |
| Benchmark  | Short test programs for benchmark.                                                                                                                        | Excel VBA         |
| CutImage   | Cut camera 0 image from coordinate point (0,0) to 100 in width and 100 in height, and display the cut image.<br>The cut image is preserved in memory #11. | Visual Basic 2005 |

|                 |                                                                                                                                                                                                  |                   |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| DENSO NetwoRC   | Search for a specified target image in the camera image, and store the detected coordinate into a variable of the controller with IP address "10.6.235.60".                                      | Visual Basic 6    |
| Filter          | Display the image from camera in the following four patterns. <ul style="list-style-type: none"><li>• raw image</li><li>• gray-scale image</li><li>• binary image</li></ul> Canny filtered image | Visual Basic 2005 |
| FindCountoursEx | Execute FindCountoursEx command.                                                                                                                                                                 | C                 |
| Histogram       | Generate a histogram of camera 1 image.                                                                                                                                                          | Visual Basic 2005 |
| Others          | Robot motion by pattern matching.                                                                                                                                                                | CaoScript         |
| SaveImage       | Save camera 1 image in image memory #11.                                                                                                                                                         | Visual Basic 2005 |

## 4. Command Reference

This chapter shows the details of each OpenCV provider command. Regarding the detailed behavior of the commands depending on OpenCV library deeply, please refer the OpenCV manual like the followings. And regarding the used OpenCV functions in the OpenCV provider commands, please see 5.10. Appendix A.

[OpenCV Japanese manual]      <http://opencv.jp/opencv-1.0.0/document/>  
 [OpenCV English manual]      [http://opencv.jp/opencv-1.0.0\\_org/docs/index.htm](http://opencv.jp/opencv-1.0.0_org/docs/index.htm)

**Table 4-1 Controller class command list**

| Category      | Command name                          | Function                                  |      |
|---------------|---------------------------------------|-------------------------------------------|------|
| Video Setting |                                       |                                           |      |
|               | <del>SetFormat</del> <sup>5</sup>     | <del>Set a video format</del>             |      |
|               | <del>GetFormatList</del> <sup>5</sup> | <del>Get a video format list</del>        |      |
|               | OpenFileterProperty                   | Open a filter property window             | P.35 |
|               | OpenPinProperty                       | Open an output Pin property window        | P.35 |
|               | SetCtrlMode                           | Set a video control mode                  | P.35 |
|               | GetCtrlMode                           | Get a video control mode                  | P.36 |
|               | GetRangeCameraCtrl                    | Get a parameter range of a camera control | P.36 |
|               | GetCameraCtrl                         | Get a parameter of a camera control       | P.37 |
|               | SetCameraCtrl                         | Set a parameter of a camera control       | P.37 |
|               | GetRangeVideoProcAmp                  | Get a parameter range of a video control  | P.38 |
|               | GetVideoProcAmp                       | Get a parameter of a video control        | P.39 |
|               | SetVideoProcAmp                       | Set a parameter of a video control        | P.40 |
|               | GetCameraFormatList                   | Get camera format list.                   | P.41 |
|               | GetCameraFormat                       | Get camera format ID.                     | P.41 |
|               | SetCameraFormat                       | Set camera format ID.                     | P.41 |
|               | ExtExecSoftTrigger                    | Execute software trigger                  | P.42 |
|               | ExtRefreshImage                       | Refresh extended camera's image           | P.42 |
|               | ExtInvoke                             | Execute extended camera's command         | P.42 |
|               | ExtConnect                            | Connect extended camera.                  | P.43 |
|               | ExtDisconnect                         | Dicconnect extended camera.               | P.43 |
|               | ExtIsConnected                        | Connection check of extended camera       | P.44 |
|               | ExtGetConnectOption                   | Get extended camera's connection option   | P.44 |
|               | ExtSetConnectOption                   | Set extended camera's connection option   | P.44 |

<sup>5</sup> This command was integrated into the OpenPinProperty command.

**Table 4-2 File class command list**

| Category       | Command name        | Function                                  |       |
|----------------|---------------------|-------------------------------------------|-------|
| <b>General</b> |                     |                                           |       |
|                | SetROI              | Set a ROI (Region Of Interest)            | P.46  |
|                | GetROI              | Get current ROI.                          | P.46  |
|                | ResetROI            | Reset current ROI setting.                | P.47  |
|                | PutColor            | Put color                                 | P.47  |
|                | GetColor            | Get color                                 | P.48  |
|                | SearchPoint         | Search point                              | P.48  |
|                | Trim                | Trimming                                  | P.49  |
|                | ImageSize           | Get image size                            | P.50  |
|                | IsColor             | Color image flag                          | P.50  |
|                | IsEmpty             | Detemine whether an image data is empty   | P.51  |
|                | IsUpdated           | Detemine whether an image data is updated | P.51  |
|                | ClearUpdated        | Clear IsUpdated flag                      | P.51  |
|                | Distance            | Measure distance                          | P.51  |
|                | InnerProduct        | Inner product of two vectors              | P.52  |
|                | OuterProduct        | Outer product of two vectors              | P.52  |
|                | PutHelp             | Set character strings                     | P.53  |
| <b>Edit</b>    |                     |                                           |       |
|                | Copy                | Copy image                                | P.53  |
|                | Cut                 | Cut image                                 | P.54  |
|                | Paste               | Paste image                               | P.54  |
|                | Rotate              | Rotate image                              | P.55  |
|                | Flip                | Flip image                                | P.56  |
|                | Resize              | Resize image                              | P.56  |
|                | Split               | Split color space                         | P.57  |
|                | Merge               | Merge color space                         | P.58  |
| <b>Filter</b>  |                     |                                           |       |
|                | ConvertGray         | Convert to gray scale                     | P.58  |
|                | ThresholdEx         | Threshold process                         | P.59  |
|                | Threshold2          | Applies fixed-level threshold             | P.60  |
|                | AdaptiveThresholdEx | Adaptive threshold process                | P.61  |
|                | Smooth              | Smoothing                                 | P.62  |
|                | Sobel               | Sobel filter                              | P.63  |
|                | Laplace             | Laplace filter                            | P.64  |
|                | CannyEx             | Canny filter                              | P.65  |
|                | WarpAffine          | Affine transformation                     | P.65  |
|                | WarpPerspective     | Perspective transformation                | P.66  |
|                | PreCornerDetectEx   | Corner detector                           | P.68  |
|                | CornerHarrisEx      | Harris edge detector                      | P.68  |
|                | CalcBackProjectEx   | Calculate back projection                 | P.69  |
|                | Inpaint             | Inpainting                                | P.69  |
|                | Erode               | Erode image                               | P.70  |
|                | Dilate              | Dilate image                              | P. 71 |

|           |                       |                                                      |        |
|-----------|-----------------------|------------------------------------------------------|--------|
|           | PyrDown               | Down sampling                                        | P.71   |
|           | PyrUp                 | Upsampling                                           | P.72   |
| Mask      |                       |                                                      |        |
|           | NOT                   | Bit inversion                                        | P.73   |
|           | AND                   | Logical AND                                          | P.73   |
|           | OR                    | Logical OR                                           | P.74   |
|           | XOR                   | Logical Exclusive OR                                 | P.74   |
|           | ADD                   | Addition                                             | P.75   |
|           | SUB                   | Subtraction                                          | P.75   |
|           | MAXEx                 | Maximum value                                        | P.76   |
|           | MINEx                 | Minimum value                                        | P.76   |
|           | ABS                   | Absolute value                                       | P.77   |
|           | LUT                   | Lookup table translation                             | P.77   |
|           | SetLUT                | Set lookup table                                     | P.78   |
|           | GetLUT                | Get lookup table                                     | P.78   |
| Draw      |                       |                                                      |        |
|           | Line                  | Draw a line (between two specified points)           | P.79   |
|           | Line2                 | Draw a line (length specified)                       | P.79   |
|           | Rectangle             | Draw a square                                        | P.80   |
|           | Circle                | Draw a circle                                        | P.81   |
|           | Ellipse               | Draw an ellipse                                      | P.82   |
|           | Sector                | Fill eclipse sector                                  | P.83   |
|           | Cross                 | Cross drawing                                        | P.84   |
|           | Text                  | Display character string                             | P.85   |
| Contours  |                       |                                                      |        |
|           | FindContoursEx        | Detect contour.                                      | P.86   |
|           | CopyContours          | Copy contour image                                   | P.87   |
|           | ContoursNumber        | Retrieve contour ID                                  | P.88   |
|           | PointPolygonTest      | Check the position relation of a point and a contour | P.88   |
|           | BoundingRect          | Find a rectangle bounding a contour                  | P.89   |
|           | FitEllipse            | Get minimum ellipse bounding the specified contour   | P.89   |
|           | ArcLength             | Get contour boundary length                          | P.90   |
|           | CheckContourConvexity | Check shape convexity                                | P.90   |
|           | DrawContours          | Draw contours                                        | P.90   |
| Blob      |                       |                                                      |        |
|           | FindBlobs             | Find blobs                                           | P.92   |
|           | BlobsFilter           | Filter the result of FindBlobs                       | P.92   |
|           | BlobResult            | Get the blob information by ID                       | P.95   |
|           | BlobResults           | Get all blob information                             | P.95   |
|           | BlobEllipse           | Get an ellipse fitting the blob                      | P.96   |
|           | BlobMatchTemplate     | Detection blob template matching                     | P.96   |
|           | BlobMatchShapes       | Detection blob shape matching                        | P.98   |
| Histogram |                       |                                                      |        |
|           | CalcHistEx            | Calculate histogram                                  | P.99   |
|           | NormalizeHistEx       | Normalize histogram                                  | P.99   |
|           | ThreshHistEx          | Threshold histogram                                  | P.100  |
|           | EqualizeHistEx        | Equalize histogram                                   | P.100  |
|           | GetMinMaxHistValue    | Get maximum and minimum value of histogram           | P. 100 |



|          |                       |                                                                |       |
|----------|-----------------------|----------------------------------------------------------------|-------|
|          | HistAve               | Calculate average luminance.                                   | P.101 |
|          | AutoThreshPTile       | Calculate threshold value using P-tile method                  | P.101 |
|          | AutoThreshMode        | Calculate threshold value using mode method                    | P.101 |
|          | AutoThreshDiscrim     | Calculate threshold value using discrimination analysis method | P.102 |
| Matching |                       |                                                                |       |
|          | MatchTemplate         | Template matching                                              | P.102 |
|          | MatchShapesEx         | Contour matching                                               | P.104 |
|          | MatchTemplate2        | Extended template matching                                     | P.104 |
|          | MatchShapes2          | Extended shape matching                                        | P.106 |
|          | CamShift              | Object tracking                                                | P.108 |
|          | HaarDetect            | Haar matching                                                  | P.109 |
| CARD     |                       |                                                                |       |
|          | CARDInit2             | Register the template image for CARD                           | P.109 |
|          | CARDRun2              | Excute CARD command                                            | P.110 |
| CAL      |                       |                                                                |       |
|          | CalibrateCamera       | Calibrate camera                                               | P.112 |
|          | CalibrateRobot        | Calibrate robot                                                | P.113 |
|          | FindChessBoardCorners | Find corner of the chessboard                                  | P.114 |
|          | DrawChessBoardCorners | Draw the corners of the chessboard                             | P.115 |
|          | DrawXYAxes            | Draw XY axis                                                   | P.115 |
|          | SetCamCalDat          | Set calibration data                                           | P.116 |
|          | GetCamCalDat          | Get calibration data                                           | P.117 |
|          | SetCamCalExtDat       | Set external variables of camera calibration                   | P.118 |
|          | GetCamCalExtDat       | Get external variables of camera calibration                   | P.119 |
|          | ModifyCamCalExtDat    | Modify camera calibration external data                        | P.120 |
|          | SetRobCalDat          | Set robot calibration data                                     | P.121 |
|          | GetRobCalDat          | Get robot calibration data                                     | P.122 |
|          | SetCameraDescription  | Set camera calibration description                             | P.122 |
|          | GetCameraDescription  | Get camera calibration description                             | P.123 |
|          | SetRobotDescription   | Set robot calibration description                              | P.123 |
|          | GetRobotDescription   | Get robot calibration description                              | P.123 |
|          | GetPostFromCam        | Convert from camera image coordinate to world coordinate       | P.124 |
|          | GetCamPos             | Convert from world coordinate to camera image coordinate       | P.125 |
|          | GetPostFromRob        | Convert from robot coordinate to world coordinate              | P.125 |
|          | GetRobPos             | Convert from world coordinate to robot coordinate              | P.126 |
|          | GetRobPosFromCam      | Conver from camera image coordinate to robot coordinate        | P.126 |
|          | GetCamPosFromRob      | Conver from robot coordinate to camera image coordinate        | P.127 |
|          | Undistort2            | Image distortion compensation                                  | P.128 |
| Misc.    |                       |                                                                |       |
|          | GoodFeatureToTrackEx  | Determines corners on image                                    | P.128 |
|          | FindCornerSubPixEx    | Refine corner detection result                                 | P.129 |
|          | MomentsEx             | Calculate moment                                               | P.130 |
|          | MeasureInfo           | Calculate area size, gravity center, and principal axis angle  | P.131 |

|                  |                                                                   |       |
|------------------|-------------------------------------------------------------------|-------|
| HoughLine        | Find lines using Hough transform                                  | P.131 |
| HoughCircles     | Find circles using Hough transform                                | P.132 |
| DFTEx            | Fourier transform                                                 | P.133 |
| IDFT             | Inverse Fourier transform                                         | P.133 |
| OpticalFlowEx    | Calculate optical flow for two images                             | P.134 |
| OpticalFlowPyrEx | Optical flow using image pyramid                                  | P.134 |
| BoxPoints        | Calculate the four corner positions of the specified rectangular. | P.135 |
| FindHomography   | Calculate projection matrix                                       | P.136 |
| QRDecode         | Decode several types of two dimensional code such as QRCode       | P.137 |
| OCRead           | Character recognition                                             | P.137 |

**Table 4-3 Command class command list**

| Category      | Command name     | Function                          |       |
|---------------|------------------|-----------------------------------|-------|
| Triangulation |                  |                                   |       |
|               | Triangulation    | Triangulation                     | P.138 |
|               | TriMatchTemplate | Template matching + Triangulation | P.138 |
|               | TriMatchShapes   | Shape matching + Triangulation    | P.140 |
|               | TriHaarDetect    | Haar matching + Triangulation     | P.142 |

## 4.1. Controller class

### 4.1.1. Video setting

# OpenFilterProperty

|              |                                                                                              |
|--------------|----------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> OpenFilterProperty <Camera ID>, <Window Handle>                               |
| Parameters   | <Camera ID> = VT_I4: Camera ID6<br><Window Handle> = VT_I4: Handle to parent or owner window |
| Return value | None                                                                                         |
| Explanation  | Open a camera filter property window.                                                        |

# OpenPinProperty

|              |                                                                                             |
|--------------|---------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> OpenPinProperty(<Camera ID>, <Window Handle>)                                |
| Parameters   | <Camera ID> = VT_I4: Camera ID<br><Window Handle> = VT_I4: Handle to parent or owner window |
| Return value | <Format ID> = VT_I4: Camera format ID                                                       |
| Explanation  | Open an output Pin property window.                                                         |

# SetCtrlMode

|              |                                                                                                                                                                                                                                                                                                                                                                                                 |                             |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------|---------------------------------|-----------------|---|-------------------------------|---------------|---|----------------------------------------|-------------------------|---|--------------------------|-----------------------------|--|
| Format       | <i>object.</i> SetCtrlMode(<Camera ID>, <Mode>)                                                                                                                                                                                                                                                                                                                                                 |                             |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
| Parameters   | <Camera ID> = VT_I4: Camera ID<br><Mode> = VT_I4: Mode                                                                                                                                                                                                                                                                                                                                          |                             |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
|              | <table border="1"> <tr> <td>1</td><td>VideoControlFlag_FlipHorizontal</td><td>Horizontal flip</td></tr> <tr> <td>2</td><td>VideoControlFlag_FlipVertical</td><td>Vertical flip</td></tr> <tr> <td>4</td><td>VideoControlFlag_ExternalTriggerEnable</td><td>External trigger enable</td></tr> <tr> <td>8</td><td>VideoControlFlag_Trigger</td><td>External trigger simulation</td></tr> </table> | 1                           | VideoControlFlag_FlipHorizontal | Horizontal flip | 2 | VideoControlFlag_FlipVertical | Vertical flip | 4 | VideoControlFlag_ExternalTriggerEnable | External trigger enable | 8 | VideoControlFlag_Trigger | External trigger simulation |  |
| 1            | VideoControlFlag_FlipHorizontal                                                                                                                                                                                                                                                                                                                                                                 | Horizontal flip             |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
| 2            | VideoControlFlag_FlipVertical                                                                                                                                                                                                                                                                                                                                                                   | Vertical flip               |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
| 4            | VideoControlFlag_ExternalTriggerEnable                                                                                                                                                                                                                                                                                                                                                          | External trigger enable     |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
| 8            | VideoControlFlag_Trigger                                                                                                                                                                                                                                                                                                                                                                        | External trigger simulation |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
| Return value | None                                                                                                                                                                                                                                                                                                                                                                                            |                             |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |
| Explanation  | Set a video control mode.<br>Please refer to IAMVideoControl::SetMode() with MSDN for the details.<br>[Caution] This command may not be executed properly depending on a camera driver                                                                                                                                                                                                          |                             |                                 |                 |   |                               |               |   |                                        |                         |   |                          |                             |  |

<sup>6</sup> The number 1 to 10 are called 'Camera ID' for convenience.

used.

Related item     GetCtrlMode

---

## GetCtrlMode

---

Format            *object. GetCtrlMode* (<CameraID>)

Parameters       <CameraID> = VT\_I4: Camera ID

Return value     <Mode> = VT\_I4: Mode

|   |                                        |                             |
|---|----------------------------------------|-----------------------------|
| 1 | VideoControlFlag_FlipHorizontal        | Horizontal flip             |
| 2 | VideoControlFlag_FlipVertical          | Vertical flip               |
| 4 | VideoControlFlag_ExternalTriggerEnable | External trigger enable     |
| 8 | VideoControlFlag_Trigger               | External trigger simulation |

Explanation      Get a video control mode.

Please refer to IAMVideoControl::SetMode() with MSDN for the details.

[Caution] This command may not be executed properly depending on a camera driver used.

Related item      SetCtrlMode

---

## GetRangeCameraCtrl

---

Format            *object. GetRangeCameraCtrl* (<CameraID>, <Property>)

Parameters       <CameraID> = VT\_I4: Camera ID

<Property> = VT\_I4: Property ID

|   |                        |                    |
|---|------------------------|--------------------|
| 0 | CameraControl_Pan      | Pan (degree)       |
| 1 | CameraControl_Tilt     | Tile (degree)      |
| 2 | CameraControl_Roll     | Roll (degree)      |
| 3 | CameraControl_Zoom     | Zoom (mm)          |
| 4 | CameraControl_Exposure | Exposure (2n Sec.) |
| 5 | CameraControl_Iris     | Iris (fstop * 10)  |
| 6 | CameraControl_Focus    | Focus (mm)         |

Return value     <Min> = VT\_I4: Min value

<Max> = VT\_I4: Max value

<Step> = VT\_I4: Step

<Default> = VT\_I4: Default value

<Flag> = VT\_I4: Flag

|   |                            |                   |
|---|----------------------------|-------------------|
| 1 | CameraControl_Flags_Auto   | Automatic Control |
| 2 | CameraControl_Flags_Manual | Manual Control    |

**Explanation** Get a parameter range of a camera control.

Please refer to IAMCameraControl::GetRange() with MSDN for the details.

[Caution] This command may not be executed properly depending on a camera driver used.

**Related item** GetCameraCtrl, SetCameraCtrl

---

## GetCameraCtrl

---

**Format** *object. GetCameraCtrl* (<CameraID>, <Property>)

**Parameters** <CameraID> = VT\_I4: Camera ID

<Property> = VT\_I4: Property ID

|   |                        |                    |
|---|------------------------|--------------------|
| 0 | CameraControl_Pan      | Pan (degree)       |
| 1 | CameraControl_Tilt     | Tilt (degree)      |
| 2 | CameraControl_Roll     | Roll (degree)      |
| 3 | CameraControl_Zoom     | Zoom (mm)          |
| 4 | CameraControl_Exposure | Exposure (2n Sec.) |
| 5 | CameraControl_Iris     | Iris (fstop * 10)  |
| 6 | CameraControl_Focus    | Focus (mm)         |

**Return value** <Value> = VT\_I4: Value

<Flag> = VT\_I4: Flag

|   |                            |                   |
|---|----------------------------|-------------------|
| 1 | CameraControl_Flags_Auto   | Automatic control |
| 2 | CameraControl_Flags_Manual | Manual control    |

**Explanation** Get a parameter of a camera control.

Please refer to IAMCameraControl::Get() with MSDN for the details.

[Caution] This command may not be executed properly depending on a camera driver used.

**Related item** GetRangeCameraCtrl, SetCameraCtrl

---

## SetCameraCtrl

---

**Format** *object. SetCameraCtrl* (<CameraID>, <Property>, <Value>, <Flag>)

Parameters      **<CameraID>** = VT\_I4: Camera ID  
                      **<Property>** = VT\_I4: Property ID

|   |                        |                    |
|---|------------------------|--------------------|
| 0 | CameraControl_Pan      | Pan (degree)       |
| 1 | CameraControl_Tilt     | Tile (degree)      |
| 2 | CameraControl_Roll     | Roll (degree)      |
| 3 | CameraControl_Zoom     | Zoom (mm)          |
| 4 | CameraControl_Exposure | Exposure (2n Sec.) |
| 5 | CameraControl_Iris     | Iris (fstop * 10)  |
| 6 | CameraControl_Focus    | Focus (mm)         |

**<Value>** = VT\_I4: Value

**<Flag>** = VT\_I4: Flag

|       |                              |                   |
|-------|------------------------------|-------------------|
| 0x001 | CameraControl_Flags_Auto     | Automatic control |
| 0x002 | CameraControl_Flags_Manual   | Manual control    |
| 0x000 | CameraControl_Flags_Absolute | Absolute values   |
| 0x010 | CameraControl_Flags_Relative | Relative values   |

Return value      None

Explanation      Set a parameter of a camera control.

Please refer to IAMCameraControl::Set() with MSDN for the details.

[Caution] This command may not be executed properly depending on a camera driver used.

Related item      GetRangeCameraCtrl, GetCameraCtrl

---

## GetRangeVideoProcAmp

---

Format            *object*. GetRangeVideoProcAmp (<CameraID>, <Property>)

Parameters      **<CameraID>** = VT\_I4: Camera ID  
                      **<Property>** = VT\_I4: Property ID

|   |                          |                                  |
|---|--------------------------|----------------------------------|
| 0 | VideoProcAmp_Brightness  | Brightness                       |
| 1 | VideoProcAmp_Contrast    | Contrast (gain * 100)            |
| 2 | VideoProcAmp_Hue         | Hue (degree * 100)               |
| 3 | VideoProcAmp_Saturation  | Saturation                       |
| 4 | VideoProcAmp_Sharpness   | Sharpness                        |
| 5 | VideoProcAmp_Gamma       | Gamma (gamma * 100)              |
| 6 | VideoProcAmp_ColorEnable | Color enabled<br>(0: OFF, 1: ON) |

|   |                                    |                                           |
|---|------------------------------------|-------------------------------------------|
| 7 | VideoProcAmp_WhiteBalance          | White balance                             |
| 8 | VideoProcAmp_BacklightCompensation | Backlight compensation<br>(0: OFF, 1: ON) |
| 9 | VideoProcAmp_Gain                  | Gain                                      |

Return value    **<Min>** = VT\_I4: Min value  
                   **<Max>** = VT\_I4: Max value  
                   **<Step>** = VT\_I4: Step  
                   **<Default>** = VT\_I4: Default value  
                   **<Flag>** = VT\_I4: Flag

|   |                            |                   |
|---|----------------------------|-------------------|
| 1 | CameraControl_Flags_Auto   | Automatic control |
| 2 | CameraControl_Flags_Manual | Manual control    |

Explanation    Get a parameter range of a video control.  
                   Please refer to IAMVideoProcAmp::GetRange() with MSDN for the details.  
                   [Caution] This command may not be executed properly depending on a camera driver used.

Related item    GetVideoProcAmp, SetVideoProcAmp

## GetVideoProcAmp

Format            *object*.GetVideoProcAmp(<CameraID>, <Property>)

Parameters       **<CameraID>** = VT\_I4: Camera ID  
                   **<Property>** = VT\_I4: Property ID

|   |                                    |                                           |
|---|------------------------------------|-------------------------------------------|
| 0 | VideoProcAmp_Brightness            | Brightness                                |
| 1 | VideoProcAmp_Contrast              | Contrast (gain * 100)                     |
| 2 | VideoProcAmp_Hue                   | Hue (degree * 100)                        |
| 3 | VideoProcAmp_Saturation            | Saturation                                |
| 4 | VideoProcAmp_Sharpness             | Sharpness                                 |
| 5 | VideoProcAmp_Gamma                 | Gamma (gamma * 100)                       |
| 6 | VideoProcAmp_ColorEnable           | Color enabled<br>(0: OFF, 1: ON)          |
| 7 | VideoProcAmp_WhiteBalance          | White balance                             |
| 8 | VideoProcAmp_BacklightCompensation | Backlight compensation<br>(0: OFF, 1: ON) |
| 9 | VideoProcAmp_Gain                  | Gain                                      |

Return value <Value> = VT\_I4: Value

<Flag> = VT\_I4: Flag

|   |                            |                   |
|---|----------------------------|-------------------|
| 1 | CameraControl_Flags_Auto   | Automatic control |
| 2 | CameraControl_Flags_Manual | Manual control    |

Explanation Get a parameter of a video control.

Please refer to IAMVideoProcAmp::Set() with MSDN for the details.

[Caution] This command may not be executed properly depending on a camera driver used.

Related item GetRangeVideoProcAmp, SetVideoProcAmp

## SetVideoProcAmp

Format *object*. SetVideoProcAmp(<CameraID>, <Property>, <Value>, <Flag>)

Parameters <CameraID> = VT\_I4: Camera ID

<Property> = VT\_I4: Property ID

|   |                                    |                                           |
|---|------------------------------------|-------------------------------------------|
| 0 | VideoProcAmp_Brightness            | Brightness                                |
| 1 | VideoProcAmp_Contrast              | Contrast (gain * 100)                     |
| 2 | VideoProcAmp_Hue                   | Hue (degree * 100)                        |
| 3 | VideoProcAmp_Saturation            | Saturation                                |
| 4 | VideoProcAmp_Sharpness             | Sharpness                                 |
| 5 | VideoProcAmp_Gamma                 | Gamma (gamma * 100)                       |
| 6 | VideoProcAmp_ColorEnable           | Color enabled<br>(0: OFF, 1: ON)          |
| 7 | VideoProcAmp_WhiteBalance          | White balance                             |
| 8 | VideoProcAmp_BacklightCompensation | Backlight compensation<br>(0: OFF, 1: ON) |
| 9 | VideoProcAmp_Gain                  | Gain                                      |

<Value> = VT\_I4: Value

<Flag> = VT\_I4: Flag

|   |                            |                   |
|---|----------------------------|-------------------|
| 1 | CameraControl_Flags_Auto   | Automatic control |
| 2 | CameraControl_Flags_Manual | Manual control    |

Return value None

Explanation Set a parameter of a video control.

Please refer to IAMVideoProcAmp::Get() with MSDN for the details.

[Caution] This command may not be executed properly depending on a camera driver



used.

Related item    GetRangeVideoProcAmp, GetVideoProcAmp

---

## GetCameraFormatList

---

|              |                                                                                                                                                                                                                                                       |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> <b>GetCameraFormatList</b> (<CameraID>)                                                                                                                                                                                                |
| Parameters   | <CameraID> = VT_I4: Camera ID                                                                                                                                                                                                                         |
| Return value | <Lists> = VT_VARIANT VT_ARRAY:FormatList (<List1>, <List2>, ...)<br><Listn> = VT_I4 VT_ARRAY:Format (<Format ID>, <Width>, <Height>)<br><Format ID> = VT_I4: Camera format ID (0~)<br><Width> = VT_I4: X resolution<br><Height> = VT_I4: Y resolution |
| Explanation  | Get camera format list.<br>-1 : Can't use this Format ID.<br>[Caution] This command may not be executed properly depending on a camera driver used.                                                                                                   |
| Related item | GetCameraFormat, SetCameraFormat                                                                                                                                                                                                                      |

---

## GetCameraFormat

---

|              |                                                                                                          |
|--------------|----------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> <b>GetCameraFormat</b> (<CameraID>)                                                       |
| Parameters   | <CameraID> = VT_I4: Camera ID                                                                            |
| Return value | <Format ID> = VT_I4: Camera format ID (0~)                                                               |
| Explanation  | Get camera format ID.<br>[Caution] Whether this command correctly works is depending on a camera driver. |
| Related item | GetCameraFormatList, SetCameraFormat                                                                     |

---

## SetCameraFormat

---

|            |                                                                             |
|------------|-----------------------------------------------------------------------------|
| Format     | <i>object.</i> <b>SetCameraFormat</b> (<CameraID>, <Format ID>)             |
| Parameters | <CameraID> = VT_I4: Camera ID<br><Format ID> = VT_I4: Camera format ID (0~) |

---

|              |                                                                                                          |
|--------------|----------------------------------------------------------------------------------------------------------|
| Return value | None                                                                                                     |
| Explanation  | Set camera format ID.<br>[Caution] Whether this command correctly works is depending on a camera driver. |
| Related item | GetCameraFormatList, GetCameraFormat                                                                     |

## ExtExecSoftTrigger

[V1.4.6or later]

|              |                                                                                                                                                                                                                                       |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object. ExtExecSoftTrigger</i> (<CameraID>)                                                                                                                                                                                        |
| Parameters   | <CameraID> = VT_I4: Camera ID                                                                                                                                                                                                         |
| Return value | None                                                                                                                                                                                                                                  |
| Explanation  | Execute software trigger of camera.<br>This command is available only for extended camera.<br>This command runs “OCV_ExecSoftTrigger” command in CaoController::Execute() on ORiN2 provider which is compatible with extended camera. |

## ExtRefreshImage

[V1.4.6or later]

|              |                                                                                                                                                                                                                                                                                           |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object. ExtRefreshImage</i> (<CameraID>)                                                                                                                                                                                                                                               |
| Parameters   | <CameraID> = VT_I4: Camera ID                                                                                                                                                                                                                                                             |
| Return value | None                                                                                                                                                                                                                                                                                      |
| Explanation  | Update the image of extended camera<br>This command is available only for extended camera.<br>This command runs OCV_GetImage command in CaoController::Execute() on ORiN2 provider which is compatible with the extended camera, and then update the internal buffer with obtained image. |

## ExtInvoke

[V1.4.6 or later]

|            |                                                               |
|------------|---------------------------------------------------------------|
| Format     | <i>object. ExtInvoke</i> (<CameraID>, <Command>, <Parameter>) |
| Parameters | <CameraID> = VT_I4: Camera ID                                 |

<Command> = VT\_BSTR: Command name

<Parameter> = VT\_VARIANT: Parameter

Return value <Result> = VT\_VARIANT: Return value

**Explanation** Execute the command of extended camera  
 This command is available only for extended camera.  
 This command runs CaoController::Execute() on ORiN2 provider which is compatible with extended camera.  
 For the command name that can be specified by <Command>, or the contents of <Parameter> or <Result>, refer to the user's guide of ORiN2 provider corresponding with the extended camera.

---

## ExtConnect

[V1.5.1 or later]

Format *object. ExtConnect* (<CameraID>)

Parameters <CameraID> = VT\_I4: Camera ID

Return value <Connected> = VT\_VARIANT: Result

|       |                   |
|-------|-------------------|
| TRUE  | Already connected |
| FALSE | New connection    |

**Explanation** Connect with the extended camera specified by <CameraID>  
 When an extended camera has not been connected, this command connects with an extended camera. The return value is FALSE when it succeeds.  
 When the specified extended camera has been already connected, this command succeeds without processing. The return value is TRUE.

Related item ExtDisconnect, ExtIsConnected

---

## ExtDisconnect

[V1.5.1 or later]

Format *object. ExtDisconnect* (<CameraID>)

Parameters <CameraID> = VT\_I4: Camera ID

Return value None

**Explanation** Disconnect the extended camera specified by <CameraID>

This command always succeeds regardless of the connection state with an extended camera.

Related item    ExtConnect, ExtIsConnected

---

## ExtIsConnected

[V1.5.1 or later]

Format            *object.* ExtIsConnected(<CameraID>)

Parameters       <CameraID> = VT\_I4: Camera ID

Return value     <Result> = VT\_VARIANT: Result

|       |                              |
|-------|------------------------------|
| TRUE  | Communication is possible.   |
| FALSE | Communication is impossible. |

Explanation      Check the communication state with the extended camera specified by <CameraID>. This command always succeeds regardless of the connection state with an extended camera. When an extended camera has been connected and the communication is available, this command returns TRUE. FALSE is returned in other status.

Related item      ExtConnect, ExtDisconnect

---

## ExtGetConnectOption

[V1.5.2 or later]

Format            *object.* ExtGetConnectOption(<CameraID>)

Parameters       <CameraID> = VT\_I4: Camera ID

Return value     <Parameter> = VT\_BSTR: Connection Option

Explanation      Get extended camera's connection option.

Related item      ExtConnect, ExtDisconnect

---

## ExtSetConnectOption

[V1.5.2 or later]

Format            *object.* ExtSetConnectOption(<CameraID>, <Parameter>)

Parameters       <CameraID> = VT\_I4: Camera ID

<Parameter> = VT\_BSTR: Connection Option

Return value     None

**Explanation**     Set extended camera's connection option.

**Related item**     ExtConnect, ExtDisconnect

## 4.2. File class

### 4.2.1. General

# SetROI

|              |                                                                                                                                                                                         |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.SetROI</i> <ROI>                                                                                                                                                              |
| Parameters   | <ROI>= VT_I4 VT_ARRAY:ROI Information (<X>, <Y>, <W>, <H>)<br><X>= VT_I4: Start point X coordinates<br><Y>= VT_I4: Start point Y coordinates<br><W>= VT_I4: Width<br><H>= VT_I4: Height |
| Return value | None                                                                                                                                                                                    |
| Explanation  | Set region of interest(ROI).<br>After executing this command, the input and result about coordinates depend on <X> and <y>.                                                             |
| Related item | GetROI, ResetROI                                                                                                                                                                        |
| Example      | [VB6]<br><pre>vntParam = Array(0, 0, 200, 100) caoFile.Execute "SetROI", vntParam ' Set a ROI (0,0) - (200,100)</pre>                                                                   |

# GetROI

|              |                                                                                                                                                                                        |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.GetROI</i> ()                                                                                                                                                                |
| Parameters   | None                                                                                                                                                                                   |
| Return value | <ROI>= VT_I4 VT_ARRAY:ROI Information(<X>, <Y>, <W>, <H>)<br><X>= VT_I4: Start point X coordinates<br><Y>= VT_I4: Start point Y coordinates<br><W>= VT_I4: Width<br><H>= VT_I4: Height |
| Explanation  | Get value of ROI. VT_EMPTY returns when ROI is not set up.                                                                                                                             |
| Related item | SetROI, ResetROI                                                                                                                                                                       |
| Example      | [VB6]<br><pre>vntRet = caoFile.Execute( "GetROI" ) ' Get current ROI x = vntRet(0) ' &lt;X&gt; y = vntRet(1) ' &lt;Y&gt;</pre>                                                         |

```
w = vntRet(2) ' <W>
h = vntRet(3) ' <H>
```

---

## ResetROI

---

|              |                                                 |
|--------------|-------------------------------------------------|
| Format       | <i>object.ResetROI ()</i>                       |
| Parameters   | None                                            |
| Return value | None                                            |
| Explanation  | Reset the parameter that set by SetROI command. |
| Related item | SetROI, GetROI                                  |
| Example      | [VB6]<br>caoFile.Execute "ResetROI"             |

---

## PutColor

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.PutColor</i> <Output ID>, <X>, <Y>, <R>, <G>, <B>                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><X> = VT_I4: X coordinates<br><Y> = VT_I4: Y coordinates<br><R> = VT_I4: Red density<br><G> = VT_I4: Green density<br><B> = VT_I4: Blue density                                                                                                                                                                                                                                                                                                |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Explanation  | Set color at the specified coordinate point.<br>For grayscale image, the point is changed to the value of <B>.<br>If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.<br>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image. |
| Related item | GetColor, SearchPoint                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

## Example

[VB6]

```
'Draw a point in red at XY position (100,200) and output it to 101st image.
vntParam = Array(101, 100, 200, 255, 0, 0)
caoFile.Execute "PutColor", vntParam
```

---

## GetColor

---

## Format

***object.GetColor ( <X>, <Y> )***

## Parameters

&lt;X&gt; = VT\_I4: X coordinates

&lt;Y&gt; = VT\_I4: Y coordinates

## Return value

&lt;Value&gt;= VT\_I4 or VT\_I4|VT\_ARRAY: Color density(&lt;R&gt;, &lt;G&gt;, &lt;B&gt;)

&lt;R&gt; = VT\_I4: Red density

&lt;G&gt; = VT\_I4: Green density

&lt;B&gt; = VT\_I4: Blue density

## Explanation

Get color at the specified coordinate point.

Color picture: Color density(VT\_I4 | VT\_ARRAY)

Grayscale picture: Brightness(VT\_I4)

## Related item

PutColor, SearchPoint

## Example

[VB6]

```
vntParam = Array(100, 200)
' Get the color at XY position (100,200).
vntRet = caoFile.Execute( "GetColor", vntParam )
r = vntRet(0) ' <R>
g = vntRet(1) ' <G>
b = vntRet(2) ' <B>
```

---

## SearchPoint

---

## Format

***object.SearchPoint( <StartX>, <StartY>, <Direction>, <Search value>, <Condition> )***

## Parameters

&lt;StartX&gt; = VT\_I4: Start point X coordinate

&lt;StartY&gt; = VT\_I4: Start point Y coordinate

&lt;Direction&gt; = VT\_I4: Search direction

|   |       |
|---|-------|
| 0 | Left  |
| 1 | Right |
| 2 | Up    |
| 3 | Down  |



<Search value> = VT\_I4: Search value

<Condition> = VT\_I4: Search condition

|   |              |                               |
|---|--------------|-------------------------------|
| 0 | equal        | [Point data] = <Search value> |
| 1 | greater than | [Point data] > <Search value> |
| 2 | less than    | [Point data] < <Search value> |

Return value <SerchPoint> = VT\_I4|VT\_ARRAY: Searched coordinate

<X> = VT\_I4: Searched X coordinate

<Y> = VT\_I4: Searched Y coordinate

Explanation Search point.

Color image is converted to gray scale before searching.

Returns the first coordinate point that meets the specified condition. When no point meets the condition, (-1, -1) is returned.

Related item PutColor, GetColor

Example [VB6]

```
'Search the point with the value of 255 at the right of XY position (10,20).
vntParam = Array(10, 20, 1, 255, 0)
vntRet = caoFile.Execute( "SearchPoint", vntParam )
x = vntRet(0) ' <X>
y = vntRet(1) ' <Y>
```

## Trim

Format *object.Trim( <Threshold>, <Condition> )*

Parameters <Threshold> = VT\_I4: Threshold

<Condition> = VT\_I4: Condition

|   |              |                            |
|---|--------------|----------------------------|
| 0 | greater than | [point data] > <Threshold> |
| 1 | less than    | [point data] < <Threshold> |

Return value <Area> = VT\_I4|VT\_ARRAY: Area of trimming

<X> = VT\_I4: X coordinate

<Y> = VT\_I4: Y coordinate

<W> = VT\_I4: Width

<H> = VT\_I4: Height

Explanation Trim the area which fulfills the argument condition.

Color image is changed to grayscale image.

Return (-1, -1, -1, -1) value when area is wrong..

Related item      SearchPoint, SetROI

Example            [VB6]

```

' Trim the area with threshold value condition of more than 128.
vntParam = Array(128, 0)
vntRet = caoFile.Execute( "Trim", vntParam )
x = vntRet(0) ' <X>
y = vntRet(1) ' <Y>
w = vntRet(2) ' <W>
h = vntRet(3) ' <H>

```

---

## ImageSize

---

Format            *object. ImageSize()*

Parameters        None

Return value      <Size> = VT\_I4|VT\_ARRAY:Size of image.  
                      <W> = VT\_I4: Width of image  
                      <H> = VT\_I4: Height of image

Explanation       Get image size.

Example            [VB6]

```

vntRet = caoFile.Execute( "ImageSize" )
w = vntRet(0) ' <W>
h = vntRet(1) ' <H>

```

---

## IsColor

---

[V1.3.5 or later]

Format            *object. IsColor()*

Parameters        None

Return value      <IsColor> = VT\_BOOL: image color

|       |                  |
|-------|------------------|
| TRUE  | Color image      |
| FALSE | Gray scale image |

Explanation       Determine whether image is colored

## IsEmpty

[V1.4.0 or later]

Format *object.IsEmpty()*

Parameters None

Return value  $\langle \text{IsEmpty} \rangle = \text{VT\_BOOL}$ : Empty or not

|       |           |
|-------|-----------|
| TRUE  | Empty     |
| FALSE | Not empty |

Explanation Determine whether an image data is empty.

## IsUpdated

[V1.4.0 or later]

Format *object.IsUpdated()*

Parameters None

Return value  $\langle \text{IsUpdated} \rangle = \text{VT\_BOOL}$ : The state of update of image data.

|       |                  |
|-------|------------------|
| TRUE  | Updated(Default) |
| FALSE | nonupdated       |

Explanation Determine whether an image data is updated.  
Execute "ClearUpdated" command, in order to change to nonupdate. An initial value is Updated. Please use in combination with "ClearUpdated".

## ClearUpdated

[V1.4.0 or later]

Format *object.ClearUpdated()*

Parameters None

Return value None

Explanation Clear the flag of update.

## Distance

Format *object.Distance<Point1>, <Point2>*

Parameters  $\langle \text{Point1} \rangle = \text{VT\_I4}|\text{VT\_ARRAY}$ : Coordinate point 1 ( $\langle X \rangle$ ,  $\langle Y \rangle$ ,  $\langle Z \rangle$ )  
 $\langle X \rangle = \text{VT\_I4}$ : X coordinate

|              |                                                                                                                                                                                                                                                                                                                                                                           |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | $\langle Y \rangle$ = VT_I4: Y coordinate<br>$\langle Z \rangle$ = VT_I4: Z coordinate<br>$\langle \text{Point2} \rangle$ = VT_I4 VT_ARRAY: Coordinate point 2 ( $\langle X \rangle$ , $\langle Y \rangle$ , $\langle Z \rangle$ )<br>$\langle X \rangle$ = VT_I4: X coordinate<br>$\langle Y \rangle$ = VT_I4: Y coordinate<br>$\langle Z \rangle$ = VT_I4: Z coordinate |
| Return value | $\langle \text{Distance} \rangle$ = VT_R8: Distance between two points                                                                                                                                                                                                                                                                                                    |
| Explanation  | Measure the distance between $\langle \text{Point1} \rangle$ and $\langle \text{Point2} \rangle$ .                                                                                                                                                                                                                                                                        |

---

## InnerProduct

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> InnerProduct $\langle \text{Vector1} \rangle$ , $\langle \text{Vector2} \rangle$                                                                                                                                                                                                                                                                                                                                                                                |
| Parameters   | $\langle \text{Vector1} \rangle$ = VT_I4 VT_ARRAY: Vector 1 ( $\langle X \rangle$ , $\langle Y \rangle$ , $\langle Z \rangle$ )<br>$\langle X \rangle$ = VT_I4: X<br>$\langle Y \rangle$ = VT_I4: Y<br>$\langle Z \rangle$ = VT_I4: Z<br>$\langle \text{Vector2} \rangle$ = VT_I4 VT_ARRAY: Vector 2 ( $\langle X \rangle$ , $\langle Y \rangle$ , $\langle Z \rangle$ )<br>$\langle X \rangle$ = VT_I4: X<br>$\langle Y \rangle$ = VT_I4: Y<br>$\langle Z \rangle$ = VT_I4: Z |
| Return value | $\langle \text{Inner Product} \rangle$ = VT_R8: Inner product                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Explanation  | Calculate the inner product between $\langle \text{Vector1} \rangle$ and $\langle \text{Vector2} \rangle$ .                                                                                                                                                                                                                                                                                                                                                                    |

---

## OuterProduct

---

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format     | <i>object.</i> OuterProduct $\langle \text{Vector1} \rangle$ , $\langle \text{Vector2} \rangle$                                                                                                                                                                                                                                                                                                                                              |
| Parameters | $\langle \text{Vector1} \rangle$ = VT_I4 VT_ARRAY: vector 1 ( $\langle X \rangle$ , $\langle Y \rangle$ , $\langle Z \rangle$ )<br>$\langle X \rangle$ = VT_I4: X<br>$\langle Y \rangle$ = VT_I4: Y<br>$\langle Z \rangle$ = VT_I4: Z<br>$\langle \text{Vector2} \rangle$ = VT_I4 VT_ARRAY: vector 2 ( $\langle X \rangle$ , $\langle Y \rangle$ , $\langle Z \rangle$ )<br>$\langle X \rangle$ = VT_I4: X<br>$\langle Y \rangle$ = VT_I4: Y |

|              |                                                                                                                                                                                                                                                        |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | $\langle Z \rangle = VT\_I4: Z$                                                                                                                                                                                                                        |
| Return value | $\langle \text{Outer product} \rangle = VT\_R8 VT\_ARRAY$ : Outer product ( $\langle X \rangle$ , $\langle Y \rangle$ , $\langle Z \rangle$ )<br>$\langle X \rangle = VT\_I4: X$<br>$\langle Y \rangle = VT\_I4: Y$<br>$\langle Z \rangle = VT\_I4: Z$ |
| Explanation  | Calculate the outer product between $\langle \text{Vector1} \rangle$ and $\langle \text{Vector2} \rangle$ .                                                                                                                                            |

---

## PutHelp

---

|              |                                                                                                                                   |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.PutHelp</i> $\langle \text{strHelp} \rangle$                                                                            |
| Parameters   | $\langle \text{strHelp} \rangle = VT\_BSTR$ : String                                                                              |
| Return value | None                                                                                                                              |
| Explanation  | Set a character string that can be obtained by <code>CaoFile::get_Help</code> .<br>This is not available for the raw image areas. |

### 4.2.2. Edit

---

## Copy

---

|              |                                                                                                                                                                                                                                                                                                                                                                                          |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.Copy</i> $\langle \text{Output ID} \rangle$                                                                                                                                                                                                                                                                                                                                    |
| Parameters   | $\langle \text{Output ID} \rangle = VT\_I4$ : Output Image ID                                                                                                                                                                                                                                                                                                                            |
| Return value | $\langle \text{Image} \rangle = VT\_UI1 VT\_ARRAY$ : Changed Image                                                                                                                                                                                                                                                                                                                       |
| Explanation  | Copy image.<br>If Output Image ID=0, return value is changed image data. If Output Image Id $\neq 0$ , the change image is stored in the specified ID image memory, and return value is Empty.<br>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image. |
| Related item | Cut, Paste                                                                                                                                                                                                                                                                                                                                                                               |

---

## Cut

---

|              |                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.Cut</i> <Output ID>, <X>, <Y>, <W>, <H>                                                                                                                                                                                                                                                                                                                                  |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><X> = VT_I4: X coordinates<br><Y> = VT_I4: Y coordinates<br><W> = VT_I4: Width<br><H> = VT_I4: Height                                                                                                                                                                                                                                      |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                           |
| Explanation  | Cut image.<br>If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.<br>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image. |
| Related item | Copy, Paste                                                                                                                                                                                                                                                                                                                                                                        |

---

## Paste

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.Paste</i> <Output ID>, <Input ID>, <X>, <Y>                                                                                                                                                                                                                                                                                                                                                                                                           |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><Input ID> = VT_I4: Putting Image ID<br><X> = VT_I4: X Coordinates<br><Y> = VT_I4: Y Coordinates                                                                                                                                                                                                                                                                                                                        |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Explanation  | The image which be specified by <Input ID> is stuck on the image which be specified by <Output ID>.<br>If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.<br>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bits color bitmap image, and grayscale output format is 8bits |

bitmap image.

Related item    Copy, Cut

## Rotate

Format    ***object.Rotate*** <Output ID>, <X>, <Y>, <Angle>, <Flag>

Parameters    <Output ID> = VT\_I4: Output Image ID  
                   <X> = VT\_I4: X Coordinates  
                   <Y> = VT\_I4: Y Coordinates  
                   <Angle> = VT\_I4: Rotating angle (degree)  
                   <Flag> = VT\_I4: Flag (<Warp>|<Interpolation>)

<Interpolation> =

|   |                 |                                                                                                                                                                          |
|---|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | CV_INTER_NN     | nearest-neighbor interpolation.                                                                                                                                          |
| 1 | CV_INTER_LINEAR | bilinear interpolation.                                                                                                                                                  |
| 2 | CV_INTER_AREA   | resampling using pixel area relation. It is preferred method for image decimation that gives moire-free results. In case of zooming it is similar to CV_INTER_NN method. |
| 3 | CV_INTER_CUBIC  | bicubic interpolation.                                                                                                                                                   |

<Warp> =

|    |                       |                                                                                                                                                                                                       |
|----|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8  | CV_WARP_FILL_OUTLIERS | Fill all the destination image pixels. If some of them correspond to outliers in the source image, they are set to 0.                                                                                 |
| 16 | CV_WARP_INVERSE_MAP   | Indicates that matrix is inverse transform from destination image to source and, thus, can be used directly for pixel interpolation. Otherwise, the function finds the inverse transform from matrix. |

Return value    <Image> = VT\_UI1|VT\_ARRAY: Changed Image

Explanation    Rotate image.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard.

Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

[Note] From Version 1.3.5, rotation direction is changed to clockwise.

Related item     Resize, Flip

---

## Flip

---

Format            ***object.Flip*** <Output ID>, <Type>

Parameters       <Output ID> = VT\_I4: Output Image ID

<Type> = VT\_I4: Type

|   |                       |
|---|-----------------------|
| 0 | Flip around Y axis    |
| 1 | Flip around X axis    |
| 2 | Flip around both axes |

Return value      <Image> = VT\_UI1|VT\_ARRAY: Changed Image

Explanation       Flip image.

If Output Image ID=0, return value is changed image data. If Output Image ID <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bits color bitmap image, and grayscale output format is 8bits bitmap image.

Related item       Resize, Rotate

---

## Resize

---

Format            ***object.Resize*** <Output ID>, <W>, <H>, <Interpolation>

Parameters       <Output ID> = VT\_I4: Output Image ID

<W> = VT\_I4: Width

<H> = VT\_I4: Height

<Interpolation> = VT\_I4: Interpolation method

|   |                 |                                                       |
|---|-----------------|-------------------------------------------------------|
| 0 | CV_INTER_NN     | nearest-neighbor interpolation.                       |
| 1 | CV_INTER_LINEAR | bilinear interpolation.                               |
| 2 | CV_INTER_AREA   | resampling using pixel area relation. It is preferred |



|   |                |                                                                                                                    |
|---|----------------|--------------------------------------------------------------------------------------------------------------------|
|   |                | method for image decimation that gives moire-free results. In case of zooming it is similar to CV_INTER_NN method. |
| 3 | CV_INTER_CUBIC | bicubic interpolation.                                                                                             |

**Return value**     <Image> = VT\_UI1|VT\_ARRAY: Changed Image

| Explanation | Resize image.                                                                                                                                                                          |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.             |
|             | The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image. |

|              |              |
|--------------|--------------|
| Related item | Rotate, Flip |
|--------------|--------------|

## Split

|        |                                                                      |
|--------|----------------------------------------------------------------------|
| Format | <i>object</i> . Split <Output ID(R)>, <Output ID(G)>, <Output ID(B)> |
|--------|----------------------------------------------------------------------|

|            |                                                                              |
|------------|------------------------------------------------------------------------------|
| Parameters | $\langle \text{Output ID (R)} \rangle = \text{VT\_I4: Red Output Image ID}$  |
|            | $\langle \text{Output ID (G)} \rangle = \text{VT\_I4: GreenOutput Image ID}$ |
|            | $\langle \text{Output ID (B)} \rangle = \text{VT\_I4: Blue Output Image ID}$ |

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return value | $\langle \mathbf{Images} \rangle = \text{VT\_VARIANT VT\_ARRAY: Splited image}$<br>$(\langle \mathbf{Image (R)} \rangle, \langle \mathbf{Image (G)} \rangle, \langle \mathbf{Image (B)} \rangle)$<br>$\langle \mathbf{Image (R)} \rangle = \text{VT\_UI1 VT\_ARRAY: Red image}$<br>$\langle \mathbf{Image (G)} \rangle = \text{VT\_UI1 VT\_ARRAY: Green image}$<br>$\langle \mathbf{Image (B)} \rangle = \text{VT\_UI1 VT\_ARRAY: Blue image}$ |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

| Explanation                                                                                                                                                                | Remarks |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| Separate color image into three elements of RGB, and each element is output to <Output ID(R)>, <Output ID(G)>, <Output ID(B)> respectively as greyscale images.            |         |
| If input image is grayscale, an error is returned.                                                                                                                         |         |
| If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty. |         |
| The changed image data is output by the 8-bit bitmap file format of the Windows standard.                                                                                  |         |

Related item Merge

---

## Merge

---

Format *object.Merge* <Output ID>, <InputID(R)>, <InputID(G)>, <InputID(B)>

Parameters  
 <Output ID> = VT\_I4: Output Image ID  
 <InputID(R)> = VT\_I4: Red Output Image ID  
 <InputID(G)> = VT\_I4: Green Output Image ID  
 <InputID(B)> = VT\_I4: Blue Output Image ID

Return value <Image> = VT\_UI1|VT\_ARRAY: Changed Image

Explanation Merge three gray scale images that correspond to R, G and B components of the color image, and output a color image. The three component images are <InputID(R)>, <InputID(G)>, <InputID(B)> and merged image output is <Output ID>.  
 If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.  
 The changed image data is output by the 24-bit bitmap file format of the Windows standard.

Related item Split

### 4.2.3. Filter

---

## ConvertGray

---

Format *object.ConvertGray* <Output ID>

Parameters <Output ID> = VT\_I4: Output Image ID

Return value <Image> = VT\_UI1|VT\_ARRAY: Changed Image

Explanation Convert to gray scale.  
 If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.  
 The changed image data is output by the 8-bit bitmap file format of the Windows standard.

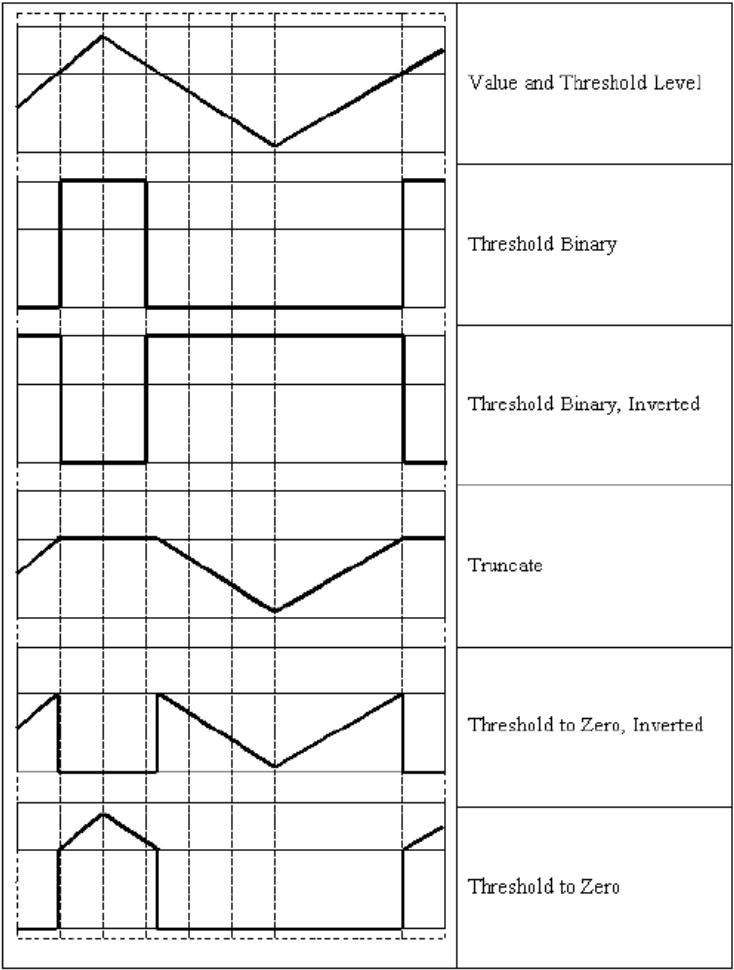
Example [VB6]  
caoFile.Execute “ConvertGray”, 101 ‘ Output to the 101st image.

# ThresholdEx

Format *object*.ThresholdEx <Output ID>, <Threshold>, <Max>, <Mode>

Parameters <Output ID> = VT\_I4: Output Image ID  
<Threshold> = VT\_I4: Threshold  
<Max> = VT\_I4: Maximum Value  
<Mode> = VT\_I4: Threshold process type

|   |                      |
|---|----------------------|
| 0 | CV_THRESH_BINARY     |
| 1 | CV_THRESH_BINARY_INV |
| 2 | CV_THRESH_TRUNC      |
| 3 | CV_THRESH_TOZERO     |
| 4 | CV_THRESH_TOZERO_INV |



|              |                                                                                                                                                                                                                                                                                                                                                                                      |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return value | <b>&lt;Image&gt;</b> = VT_UI1 VT_ARRAY : Changed Image                                                                                                                                                                                                                                                                                                                               |
| Explanation  | <p>Threshold process.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> |
| Related item | Threshold2, AdaptiveThresholdEx                                                                                                                                                                                                                                                                                                                                                      |

---

## Threshold2

---

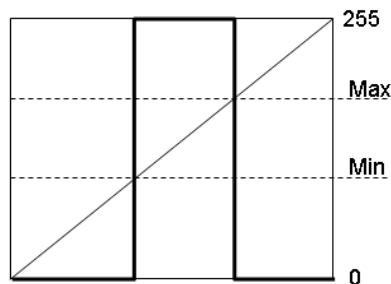
Format ***object*.Threshold2** <Output ID>, <Min>, <Max>

Parameters

**<Output ID>** = VT\_I4: Output Image ID

**<Min>** = VT\_I4: Lower bound threshold

**<Max>** = VT\_I4: Upper bound threshold



|              |                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return value | <b>&lt;Image&gt;</b> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                              |
| Explanation  | <p>Applies fixed-level threshold to array elements.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> |
| Related item | ThresholdEx, AdaptiveThresholdEx                                                                                                                                                                                                                                                                                                                                                                                   |

# AdaptiveThresholdEx

**Format** `object.AdaptiveThresholdEx <Output ID>, <Max value>, <Method>, <Type>, <Block size>, <Parameter>`

**Parameters**

**<Output ID>** = VT\_I4: Output Image ID

**<Max value>** = VT\_I4: Maximum Value

**<Method>** = VT\_I4: Adaptive threshold algorithm type

|   |                               |
|---|-------------------------------|
| 0 | CV_ADAPTIVE_THRESH_MEAN_C     |
| 1 | CV_ADAPTIVE_THRESH_GAUSSIAN_C |

**<Type>** = VT\_I4: Threshold process type

|   |                      |
|---|----------------------|
| 0 | CV_THRESH_BINARY     |
| 1 | CV_THRESH_BINARY_INV |

**<Block size>** = VT\_I4: The size of a pixel neighborhood.(3,5,7,...)

**<Parameter>** = VT\_R8: The method-dependent parameter

**Return value** **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation** Adaptive threshold process.

Color image is automatically converted to grayscale image.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the 8-bit bitmap file format of the Windows standard.

Method option determines threshold calculation method, as shown in the table below.

|                               |                                                                                                               |
|-------------------------------|---------------------------------------------------------------------------------------------------------------|
| CV_ADAPTIVE_THRESH_MEAN_C     | It is a mean of <Block size> × <Block size> pixel neighborhood, subtracted by <Parameter>.                    |
| CV_ADAPTIVE_THRESH_GAUSSIAN_C | It is a weighted sum (gaussian) of <Block size> × <Block size> pixel neighborhood, subtracted by <Parameter>. |

**Related item** ThresholdEx, Threshold2

# Smooth

**Format** *object.Smooth* <Output ID>, <Type>, <Parameter1>, <Parameter2>, <Parameter3>, <Parameter4>

**Parameters** <Output ID> = VT\_I4: Output Image ID

<Type> = VT\_I4: Smoothing type

|   |                                                   |                                                                                                                           |
|---|---------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| 0 | CV_BLUR_NO_SCALE<br>(simple blur with no scaling) | summation over a pixel <Parameter1> × <Parameter2> neighborhood.                                                          |
| 1 | CV_BLUR<br>(simple blur)                          | summation over a pixel <Parameter1> × <Parameter2> neighborhood with subsequent scaling by 1/(<Parameter1>•<Parameter2>). |
| 2 | CV_GAUSSIAN<br>(gaussian blur)                    | convolving image with <Parameter1> × <Parameter2> Gaussian kernel.                                                        |
| 3 | CV_MEDIAN<br>(median blur)                        | finding median of <Parameter1> × <Parameter1> neighborhood (i.e. the neighborhood is square).                             |
| 4 | CV_BILATERAL<br>(bilateral filter)                | applying bilateral 3x3 filtering with color sigma=<Parameter1>and space sigma=<Parameter2>.                               |

<Parameter1> = VT\_I4: Parameter 1

<Parameter2> = VT\_I4: Parameter 2

In case of simple scaled/non-scaled and Gaussian blur if <Parameter2> is zero, it is set to <Parameter1>.

<Parameter3> = VT\_I4: Parameter 3

In case of Gaussian kernel this parameter may specify Gaussian sigma (standard deviation). If it is zero, it is calculated from the kernel size:

$$\sigma = \left( \frac{n}{2} - 1 \right) \times 0.3 + 0.8$$

where n=<Parameter1>for horizontal kernel,

n=<Parameter2>for vertical kernel.

With the standard sigma for small kernels (3×3 to 7×7) the performance is better. If param3 is not zero, while param1 and param2 are zeros, the kernel size is calculated from the sigma (to provide accurate enough operation).

<Parameter4> = VT\_I4: Parameter 4

In case of non-square Gaussian kernel the parameter may be used to specify a

different (from param3) sigma in the vertical direction.

**Return value**    **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation**    Smoothing. Parameters 1-4 have different meanings according to the conversion Type. If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

Smoothing without scaling only supports gray scale images.

Other types of smoothing supports both grayscale images and color images.

---

## Sobel

---

**Format**    *object. Sobel* **<Output ID>**, **<X order>**, **<Y order>**, **<Aperture>**

**Parameters**    **<Output ID>** = VT\_I4: Output Image ID  
**<X order>** = VT\_I4: X direction order  
**<Y order>** = VT\_I4: Y direction order  
**<Aperture>** = VT\_I4: Aperture size

Size of the extended Sobel kernel, must be 1, 3, 5 or 7. In all cases except 1, <Aperture> × <Aperture> separable kernel will be used to calculate the derivative. For aperture\_size=1 3x1 or 1x3 kernel is used (Gaussian smoothing is not done). There is also special value CV\_SCHARR (=1) that corresponds to 3x3 Scharr filter that may give more accurate results than 3x3 Sobel. Scharr aperture is:

$$\begin{pmatrix} -3 & 0 & 3 \\ -10 & 0 & 10 \\ -3 & 0 & 3 \end{pmatrix}$$

for x-derivative or transposed for y-derivative.

**Return value**    **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation**    Sobel filter.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bits color bitmap image, and grayscale output format is 8bit bitmap image.

This command calculates the image derivative by convolving the image with the appropriate kernel:

$$dst(x, y) = \frac{d^{XOrder+YOrder} src}{dx^{XOrder} \cdot dy^{YOrder}} \Big|_{(x,y)}$$

---

## Laplace

---

Format `object.Laplace <Output ID>, <Aperture>`

Parameters `<Output ID>` = VT\_I4: Output Image ID

`<Aperture>` = VT\_I4: Aperture size

Size of the extended Sobel kernel, must be 1, 3, 5 or 7. (it has the same meaning as in Sobel command).

Return value `<Image>` = VT\_UI1|VT\_ARRAY: Changed Image

Explanation Laplace filter.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

This command calculates Laplacian of the source image by summing second x- and y-derivatives calculated using Sobel operator:

$$dst(x, y) = \frac{d^2 src}{dx^2} + \frac{d^2 src}{dy^2}$$

Specifying `<Aperture>=1` gives the fastest variant that is equal to convolving the image with the following kernel:

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$



## CannyEx

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.CannyEx</i> <Output ID>, <Threshold1>, <Threshold2>, <Aperture>                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Parameters   | <p>&lt;Output ID&gt; = VT_I4: Output Image ID</p> <p>&lt;Threshold1&gt; = VT_I4: Threshold 1</p> <p>&lt;Threshold2&gt; = VT_I4: Threshold 2</p> <p>&lt;Aperture&gt; = VT_I4: Aperture size</p> <p>Size of the extended Sobel kernel, must be 3, 5 or 7. (it has the same meaning as in Sobel command).</p>                                                                                                                                                                                                                         |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Explanation  | <p>Canny filter.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> <p>The smallest of &lt;Threshold1&gt; and &lt;Threshold2&gt; is used for edge linking, the largest - to find initial segments of strong edges.</p> |

## WarpAffine

|            |                                                                                                                                                                                                                                                                                                                                                                 |                                 |             |                                 |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|-------------|---------------------------------|
| Format     | <i>object.WarpAffine</i> <Output ID>, <Ax>, <Bx>, <Dx>, <Ay>, <By>, <Dy>, <Flag>                                                                                                                                                                                                                                                                                |                                 |             |                                 |
| Parameters | <Output ID> = VT_I4: Output Image ID<br><Ax> = VT_I4: Affine transformation matrix<br><Bx><br><Dx><br><Ay><br><By><br><Dy><br>$\begin{pmatrix} Ax & Bx & Dx \\ Ay & By & Dy \end{pmatrix}$<br><Flag> = VT_I4: Flag (<Warp> <Interpolation>)<br><Interpolation> = <table><tr><td>0</td><td>CV_INTER_NN</td><td>nearest-neighbor interpolation.</td></tr></table> | 0                               | CV_INTER_NN | nearest-neighbor interpolation. |
| 0          | CV_INTER_NN                                                                                                                                                                                                                                                                                                                                                     | nearest-neighbor interpolation. |             |                                 |

|   |                 |                                                                                                                                                                          |
|---|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | CV_INTER_LINEAR | bilinear interpolation.                                                                                                                                                  |
| 2 | CV_INTER_AREA   | resampling using pixel area relation. It is preferred method for image decimation that gives moire-free results. In case of zooming it is similar to CV_INTER_NN method. |
| 3 | CV_INTER_CUBIC  | bicubic interpolation.                                                                                                                                                   |

**<Warp> =**

|    |                       |                                                                                                                                                                                                       |
|----|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8  | CV_WARP_FILL_OUTLIERS | Fill all the destination image pixels. If some of them correspond to outliers in the source image, they are set to 0.                                                                                 |
| 16 | CV_WARP_INVERSE_MAP   | Indicates that matrix is inverse transform from destination image to source and, thus, can be used directly for pixel interpolation. Otherwise, the function finds the inverse transform from matrix. |

**Return value**    **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation**    Affine transformation.

If Output Image ID=0, return value is changed image data. If Output Image ID <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

---

## WarpPerspective

---

**Format**    *object. WarpPerspective***<Output ID>, <Extrinsic matrix>, <Flag>**

**Parameters**    **<Output ID>** = VT\_I4: Output Image ID

**<Extrinsic matrix>** = VT\_R8|VT\_ARRAY: Transformation matrix

(**<r11>**, **<r21>**, **<r31>**, **<r12>**, **<r22>**, **<r32>**, **<r13>**, **<r23>**, **<r33>**)

**<r11>** = VT\_R8:

**<r21>** = VT\_R8:

**<r31>** = VT\_R8:

**<r12>** = VT\_R8:

<r22> = VT\_R8:

<r32> = VT\_R8:

<r13> = VT\_R8:

<r23> = VT\_R8:

<r33> = VT\_R8:

$$\begin{pmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{pmatrix}$$

<Flag> = VT\_I4: Flag (<Warp>|<Interpolation>)

<Interpolation> =

|   |                 |                                                                                                                                                                          |
|---|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | CV_INTER_NN     | nearest-neighbor interpolation.                                                                                                                                          |
| 1 | CV_INTER_LINEAR | bilinear interpolation.                                                                                                                                                  |
| 2 | CV_INTER_AREA   | resampling using pixel area relation. It is preferred method for image decimation that gives moire-free results. In case of zooming it is similar to CV_INTER_NN method. |
| 3 | CV_INTER_CUBIC  | bicubic interpolation.                                                                                                                                                   |

<Warp> =

|    |                       |                                                                                                                                                                                                       |
|----|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8  | CV_WARP_FILL_OUTLIERS | Fill all the destination image pixels. If some of them correspond to outliers in the source image, they are set to 0.                                                                                 |
| 16 | CV_WARP_INVERSE_MAP   | Indicates that matrix is inverse transform from destination image to source and, thus, can be used directly for pixel interpolation. Otherwise, the function finds the inverse transform from matrix. |

**Return value** <Image> = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation** Calculate perspective transformation.

When output image number is 0, the transformed image is output to return value. When output image number is not 0, the transformed image is output to the specified number ID image, and return value becomes Empty.

The transformed image output data is Windows standard bitmap file format. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

---

## PreCornerDetectEx

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> PreCornerDetectEx <Output ID>, <Aperture>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Parameters   | <p>&lt;Output ID&gt; = VT_I4: Output Image ID</p> <p>&lt;Aperture&gt; = VT_I4: Aperture size</p> <p>Size of the extended Sobel kernel, must be 3, 5 or 7. (it has the same meaning as in Sobel command).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Explanation  | <p>Corner detector.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> <p>This command calculates the function</p> $D_x^2 D_{yy} + D_y^2 D_{xx} - 2 D_x D_y D_{xy}$ <p>where D? denotes one of the first image derivatives and D?? denotes a second image derivative. The corners can be found as local maximums of the function.</p> |

---

## CornerHarrisEx

---

|              |                                                                                                                                                                                                                                                                                                    |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> CornerHarrisEx <Output ID>, <Block size>, <Aperture>, <K>                                                                                                                                                                                                                           |
| Parameters   | <p>&lt;Output ID&gt; = VT_I4: Output Image ID</p> <p>&lt;Block size&gt; = VT_I4: Block size</p> <p>&lt;Aperture&gt; = VT_I4: Aperture size</p> <p>Size of the extended Sobel kernel, must be 3, 5 or 7. (it has the same meaning as in Sobel command).</p> <p>&lt;K&gt; = VT_R8: free variable</p> |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                           |

|             |                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Explanation | <p>Harris edge detector.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

---

## CalcBackProjectEx

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                      |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> . CalcBackProjectEx <Output ID>, <Input ID>                                                                                                                                                                                                                                                                                                                                            |
| Parameters   | <p>&lt;Output ID&gt; = VT_I4: Output Image ID</p> <p>&lt;Input ID&gt; = VT_I4: Input image ID</p>                                                                                                                                                                                                                                                                                                    |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                             |
| Explanation  | <p>Calculate back projection.</p> <p>Creates histogram from input image, and calculate back projection.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> |

---

## Inpaint

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                            |                                |               |                             |   |                  |                                |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------|---------------|-----------------------------|---|------------------|--------------------------------|
| Format       | <i>object</i> . Inpaint <Output ID>, <MaskID>, <Range>, <Flag>                                                                                                                                                                                                                                                                                                                                             |                                |               |                             |   |                  |                                |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><MaskID> = VT_I4: Mask Image ID<br>gray scale. Non-zero pixels indicate the area that needs to be inpainted.<br><Range> = VT_I4: Adjacent area<br><Flag> = VT_I4: Repair method<br><table><tr><td>0</td><td>CV_INPAINT_NS</td><td>Navier-Stokes based method.</td></tr><tr><td>1</td><td>CV_INPAINT_TELEA</td><td>The method by Alexandru Telea.</td></tr></table> | 0                              | CV_INPAINT_NS | Navier-Stokes based method. | 1 | CV_INPAINT_TELEA | The method by Alexandru Telea. |
| 0            | CV_INPAINT_NS                                                                                                                                                                                                                                                                                                                                                                                              | Navier-Stokes based method.    |               |                             |   |                  |                                |
| 1            | CV_INPAINT_TELEA                                                                                                                                                                                                                                                                                                                                                                                           | The method by Alexandru Telea. |               |                             |   |                  |                                |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                   |                                |               |                             |   |                  |                                |

**Explanation**

Repair image.

As for the mask data, the brightness of < MaskID > image is made from the value of one or more.

Color image is automatically converted to grayscale image.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

---

**Erode**

---

**Format**

*object.Erode* <Output ID>, <Iterations>, <Cols>, <Rows>, <AnchorX>, <AnchorY>, <Shape>

**Parameters**

<Output ID> = VT\_I4: Output image number

<Iterations> = VT\_I4: Number of times erosion is applied

<Cols> = VT\_I4: Columns of structuring element

<Rows> = VT\_I4: Rows of structuring element

<Anchor X> = VT\_I4: Horizontal relative offset of anchor point

<Anchor Y> = VT\_I4: Vertical relative offset of anchor point

<Shape> = VT\_I4: Structuring element shape

|   |                  |                        |
|---|------------------|------------------------|
| 0 | CV_SHAPE_RECT    | A rectangular element  |
| 1 | CV_SHAPE_CROSS   | A cross-shaped element |
| 2 | CV_SHAPE_ELLIPSE | An elliptic element    |

**Return value**

<Image> = VT\_UI1|VT\_ARRAY: Converted image

**Explanation**

Erodes the image using the specified structuring element that determines the shape of a pixel neighborhood over which the minimum is taken

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

Related item     Dilate

## Dilate

Format     *object.Dilate* <Output ID>, <Iterations>, <Cols>, <Rows>, <AnchorX>, <AnchorY>, <Shape>

Parameters     <Output ID> = VT\_I4: Output image number  
                   <Iterations> = VT\_I4: Number of times erosion is applied  
                   <Cols> = VT\_I4: Columns of structuring element  
                   <Rows> = VT\_I4: Rows of structuring element  
                   <Anchor X> = VT\_I4: Horizontal relative offset of anchor point  
                   <Anchor Y> = VT\_I4: Vertical relative offset of anchor point  
                   <Shape> = VT\_I4: Structuring element shape

|   |                  |                        |
|---|------------------|------------------------|
| 0 | CV_SHAPE_RECT    | A rectangular element  |
| 1 | CV_SHAPE_CROSS   | A cross-shaped element |
| 2 | CV_SHAPE_ELLIPSE | An elliptic element    |

Return value     <Image> = VT\_UI1|VT\_ARRAY: Converted image

Explanation     Dilates the image using the specified structuring element that determines the shape of a pixel neighborhood over which the maximum is taken.  
                   If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.  
                   The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

Related item     Erode

## PyrDown

Format     *object.PyrDown* <Output ID>

Parameters     <Output ID> = VT\_I4: Output image number

Return value     <Image> = VT\_UI1|VT\_ARRAY: Converted image

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Explanation  | <p>Performs downsampling step of Gaussian pyramid decomposition. First it convolves source image with the specified filter and then downsamples the image by rejecting even rows and columns.</p> <p>The width and height of output image becomes half of input image.</p> <p>When output image number is 0, transferred image is output to return value.</p> <p>When output image is not 0, transferred image is output to the specified number, and return value.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | PyrUp                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

---

## PyrUp

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <code>object. PyrUp &lt;Output ID&gt;</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Parameters   | <code>&lt;Output ID&gt;</code> = VT_I4: Output image number                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Return value | <code>&lt;Image&gt;</code> = VT_UI1 VT_ARRAY: Converted image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Explanation  | <p>performs up-sampling step of Gaussian pyramid decomposition. First it upsamples the source image by injecting even zero rows and columns and then convolves result with the specified filter multiplied by 4 for interpolation. So the destination image is four times larger than the source image.</p> <p>The width and height of output image is doubled from the input image.</p> <p>When output image number is 0, transferred image is output to return value.</p> <p>When output image is not 0, transferred image is output to the specified number, and return value.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | PyrDown                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |



#### 4.2.4. Mask

## NOT

|              |                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object. NOT</i> <Output ID>                                                                                                                                                                                                                                                                                                                                                                              |
| Parameters   | <Output ID> = VT_I4: Output Image ID                                                                                                                                                                                                                                                                                                                                                                        |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                    |
| Explanation  | <p>Bit inversion.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | AND, OR, XOR, ADD, SUB, MAXEx, MINEx, ABS                                                                                                                                                                                                                                                                                                                                                                   |

## AND

|              |                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object. AND</i> <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                                                 |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                                                 |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                  |
| Explanation  | <p>Logical AND.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | NOT, OR, XOR, ADD, SUB, MAXEx, MINEx, ABS                                                                                                                                                                                                                                                                                                                                                                 |

---

## OR

---

|              |                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object. OR</i> <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                            |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                           |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                            |
| Explanation  | Logical OR.<br>If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.<br>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image. |
| Related item | NOT, AND, XOR, ADD, SUB, MAXEx, MINEx, ABS                                                                                                                                                                                                                                                                                                                                          |

---

## XOR

---

|              |                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object. XOR</i> <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                                     |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                                     |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                      |
| Explanation  | Logical Exclusive-OR.<br>If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.<br>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image. |
| Related item | NOT, AND, OR, ADD, SUB, MAXEx, MINEx, ABS                                                                                                                                                                                                                                                                                                                                                     |

---

## ADD

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                        |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.ADD</i> <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                                               |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                                              |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                               |
| Explanation  | <p>Addition.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | NOT, AND, OR, XOR,, SUB, MAXEx, MINEx, ABS                                                                                                                                                                                                                                                                                                                                                             |

---

## SUB

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.SUB</i> <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                                                  |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                                                 |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                  |
| Explanation  | <p>Subtraction.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | NOT, AND, OR, XOR, ADD, MAXEx, MINEx, ABS                                                                                                                                                                                                                                                                                                                                                                 |

---

## MAXEx

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> MAXEx <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                                                                                                                     |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Explanation  | <p>Maximum value.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | NOT, AND, OR, XOR, ADD, SUB, MINEx, ABS                                                                                                                                                                                                                                                                                                                                                                                                                                       |

---

## MINEx

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> MINEx <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                                                                                                                     |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Explanation  | <p>Minimum value.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | NOT, AND, OR, XOR, ADD, SUB, MAXEx, ABS                                                                                                                                                                                                                                                                                                                                                                                                                                       |

---

## ABS

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.ABS</i> <Output ID>, <InputID>                                                                                                                                                                                                                                                                                                                                                                     |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><InputID> = VT_I4: Input Image ID                                                                                                                                                                                                                                                                                                                                    |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                     |
| Explanation  | <p>Absolute value.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | NOT, AND, OR, XOR, ADD, SUB, MAXEx, MINEx                                                                                                                                                                                                                                                                                                                                                                    |

---

## LUT

---

[V1.3.5 or later]

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.LUT</i> <Output ID>, <LUT ID>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Parameters   | <Output ID> = VT_I4: Output Image ID<br><LUT ID> = VT_I4: Lookup table number                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Explanation  | <p>Apply lookup table conversion to &lt;InputID&gt; image, and the converted image is output to &lt;Output ID&gt; or return value.</p> <p>When &lt;InputID&gt; image is color, each hue is converted using corresponding lookup table.</p> <p>When &lt;InputID&gt; image is grayscale, the image is converted using blue hue table.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |

|              |                |
|--------------|----------------|
| Related item | SetLUT, GetLUT |
|--------------|----------------|

## SetLUT

[V1.3.5 or later]

|        |                                                                 |
|--------|-----------------------------------------------------------------|
| Format | <i>object</i> .SetLUT <LUT ID>, <Table R>, <Table G>, <Table B> |
|--------|-----------------------------------------------------------------|

|            |                                                     |
|------------|-----------------------------------------------------|
| Parameters | <LUT ID> = VT_I4: Lookup table number               |
|            | <Table R> = VT_UI1 VT_ARRAY: Red hue lookup table   |
|            | <Table G> = VT_UI1 VT_ARRAY: Green hue lookup table |
|            | <Table B> = VT_UI1 VT_ARRAY: Blue hue lookup table  |

Return value     **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

|             |                                                                                                 |
|-------------|-------------------------------------------------------------------------------------------------|
| Explanation | Setup the specified lookup table.                                                               |
|             | Each hue table requires 256 pixels.                                                             |
|             | When hue table is not specified and VT_EMPTY is used instead, the table content is not changed. |

|              |             |
|--------------|-------------|
| Related item | LUT, GetLUT |
|--------------|-------------|

## GetLUT

[V1.3.5 or later]

|        |                                |
|--------|--------------------------------|
| Format | <i>object</i> .GetLUT <LUT ID> |
|--------|--------------------------------|

|            |                                       |
|------------|---------------------------------------|
| Parameters | <LUT ID> = VT_I4: Lookup table number |
|------------|---------------------------------------|

**Return value**     <LUT> = VT\_VARIANT|VT\_ARRAY: Lookup table  
                              (<Table R>, <Table G>, <Table B>)

<**Table R**> = VT\_UI1|VT\_ARRAY: Red hue lookup table  
<**Table G**> = VT\_UI1|VT\_ARRAY: Green hue lookup table  
<**Table B**> = VT\_UI1|VT\_ARRAY: Blue hue lookup table

|             |                                 |
|-------------|---------------------------------|
| Explanation | Get the specified lookup table. |
|-------------|---------------------------------|

|              |             |
|--------------|-------------|
| Related item | LUT, SetLUT |
|--------------|-------------|

## 4.2.5. Draw

## Line

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |     |                   |   |                   |    |                   |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-------------------|---|-------------------|----|-------------------|
| Format       | <i>object.Line</i> <Output ID>, <StartX>, <StartY>, <End X>, <End Y>, <R>, <G>, <B>, <Thick>, <Type>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |     |                   |   |                   |    |                   |
| Parameters   | <p>           &lt;Output ID&gt; = VT_I4: Output Image ID<br/>           &lt;StartX&gt; = VT_I4: Start point X coordinates<br/>           &lt;StartY&gt; = VT_I4: Start point Y coordinates<br/>           &lt;End X&gt; = VT_I4: End point X coordinates<br/>           &lt;End Y&gt; = VT_I4: End point Y coordinates<br/>           &lt;R&gt; = VT_I4: Red density<br/>           &lt;G&gt; = VT_I4: Green density<br/>           &lt;B&gt; = VT_I4: Blue density<br/>           &lt;Thick&gt; = VT_I4: Thickness<br/>           &lt;Type&gt; = VT_I4: Line type         </p> <table border="1"> <tbody> <tr> <td>0,8</td><td>8-connected line.</td></tr> <tr> <td>4</td><td>4-connected line.</td></tr> <tr> <td>16</td><td>antialiased line.</td></tr> </tbody> </table> | 0,8 | 8-connected line. | 4 | 4-connected line. | 16 | antialiased line. |
| 0,8          | 8-connected line.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |     |                   |   |                   |    |                   |
| 4            | 4-connected line.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |     |                   |   |                   |    |                   |
| 16           | antialiased line.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |     |                   |   |                   |    |                   |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |     |                   |   |                   |    |                   |
| Explanation  | <p>Draw a line (between two specified points).</p> <p>For grayscale image, the drawing brightness is set to the value of &lt;B&gt;.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p>                                                                                                                                                                                                                                                |     |                   |   |                   |    |                   |
| Related item | Line2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |     |                   |   |                   |    |                   |

## Line2

|        |                                                                                                         |
|--------|---------------------------------------------------------------------------------------------------------|
| Format | <i>object.Line2</i> <Output ID>, <StartX>, <StartY>, <Length>, <Rotate>, <R>, <G>, <B>, <Thick>, <Type> |
|--------|---------------------------------------------------------------------------------------------------------|

Parameters      <Output ID> = VT\_I4: Output Image ID  
                     <StartX> = VT\_I4: Start point X coordinates  
                     <StartY> = VT\_I4: Start point Y coordinates  
                     <Length> = VT\_I4: Length  
                     <Angle> = VT\_I4: Rotating angle (degree)  
                     <R> = VT\_I4: Red density  
                     <G> = VT\_I4: Green density  
                     <B> = VT\_I4: Blue density  
                     <Thick> = VT\_I4: Thickness  
                     <Type> = VT\_I4: Line type

|     |                   |
|-----|-------------------|
| 0,8 | 8-connected line. |
| 4   | 4-connected line. |
| 16  | antialiased line. |

Return value    <Image> = VT\_UI1|VT\_ARRAY: Changed Image

Explanation    Draw a line (length specified).  
                     For grayscale image, the drawing brightness is set to the value of <B>.  
                     If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.  
                     The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.  
                     [Note] From Version 1.3.5, rotation direction is changed to clockwise.

Related item    Line

---

## Rectangle

---

Format            *object. Rectangle* <Output ID>, <StartX>, <StartY>, <End X>, <End Y>, <R>, <G>, <B>, <Thick>, <Type>

Parameters      <Output ID> = VT\_I4: Output Image ID  
                     <StartX> = VT\_I4: Start point X coordinates  
                     <StartY> = VT\_I4: Start point Y coordinates  
                     <End X> = VT\_I4: End point X coordinates  
                     <End Y> = VT\_I4: End point Y coordinates



<R> = VT\_I4: Red density

<G> = VT\_I4: Green density

<B> = VT\_I4: Blue density

<Thick> = VT\_I4: Thickness

<Type> = VT\_I4: Line type

|     |                   |
|-----|-------------------|
| 0,8 | 8-connected line. |
| 4   | 4-connected line. |
| 16  | antialiased line. |

**Return value** <Image> = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation** Draw a rectangle.

For grayscale image, the drawing brightness is set to the value of <B>.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

---

## Circle

---

**Format** *object.Circle* <Output ID>, <X>, <Y>, <Radius>, <R>, <G>, <B>, <Thick>, <Type>

**Parameters** <Output ID> = VT\_I4: Output Image ID

<X> = VT\_I4: center X coordinates

<Y> = VT\_I4: center Y coordinates

<Radius> = VT\_I4: radius

<R> = VT\_I4: red density

<G> = VT\_I4: green density

<B> = VT\_I4: blue density

<Thick> = VT\_I4: thickness

<Type> = VT\_I4: line type

|     |                   |
|-----|-------------------|
| 0,8 | 8-connected line. |
| 4   | 4-connected line. |
| 16  | antialiased line. |

**Return value**     **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation**     Draw a circle.

For grayscale image, the drawing brightness is set to the value of <B>.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

## Ellipse

**Format**     *object.Ellipse* <Output ID>, <X>, <Y>, <XRadius>, <YRadius>, <Angle>, <Start angle>, <End angle>, <R>, <G>, <B>, <Thick>, <Type>

**Parameters**     **<Output ID>** = VT\_I4: Output Image ID  
**<X>** = VT\_I4: Center point X coordinates  
**<Y>** = VT\_I4: Center point Y coordinates  
**<XRadius>** = VT\_I4: Radius of X axis  
**<YRadius>** = VT\_I4: Radius of Y axis  
**<Angle>** = VT\_I4: Rotate angle (degree)  
**<Start angle>** = VT\_I4: Start angle (degree)  
**<End angle>** = VT\_I4: End angle (degree)  
**<R>** = VT\_I4: red density  
**<G>** = VT\_I4: green density  
**<B>** = VT\_I4: blue density  
**<Thick>** = VT\_I4: thickness  
**<Type>** = VT\_I4: line type

|     |                   |
|-----|-------------------|
| 0,8 | 8-connected line. |
| 4   | 4-connected line. |
| 16  | antialiased line. |

**Return value**     **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation**     Draw an ellipse.

For grayscale image, the drawing brightness is set to the value of <B>.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0,

the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

Please see the Ellipse function in the OpenCV reference for the details.

[Note] From Version 1.3.5, rotation direction is changed to clockwise..

## Sector

**Format** *object.Sector* <Output ID>, <X>, <Y>, <XRadius>, <YRadius>, <Angle>, <Start angle>, <End angle>, <R>, <G>, <B>, <Thick>, <Type>

**Parameters**

- <Output ID> = VT\_I4: Output Image ID
- <X> = VT\_I4: Center point X coordinates
- <Y> = VT\_I4: Center point Y coordinates
- <XRadius> = VT\_I4: Radius of X axis
- <YRadius> = VT\_I4: Radius of Y axis
- <Angle> = VT\_I4: Rotate angle degree
- <Start angle> = VT\_I4: Start angle degree
- <End angle> = VT\_I4: End angle degree
- <R> = VT\_I4: red density
- <G> = VT\_I4: green density
- <B> = VT\_I4: blue density
- <Thick> = VT\_I4: thickness
- <Type> = VT\_I4: line type

|     |                   |
|-----|-------------------|
| 0,8 | 8-connected line. |
| 4   | 4-connected line. |
| 16  | antialiased line. |

**Return value** <Image> = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation** Fill eclipse sector.

For grayscale image, the drawing brightness is set to the value of <B>.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard.

Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

[Note] From Version 1.3.5, rotation direction is changed to clockwise.

## Cross

**Format** *object.Cross* <Output ID>, <X>, <Y>, <XSize>, <YSize>, <Angle>, <R>, <G>, <B>, <Thick>, <Type>

**Parameters** <Output ID> = VT\_I4: Output Image ID

<X> = VT\_I4: Start point X coordinates

<Y> = VT\_I4: Start point Y coordinates

<XSize> = VT\_I4: Size of X axis

<YSize> = VT\_I4: Size of Y axis

<Angle> = VT\_I4: Rotate angle degree

<R> = VT\_I4: red density

<G> = VT\_I4: green density

<B> = VT\_I4: blue density

<Thick> = VT\_I4: thickness

<Type> = VT\_I4: line type

|     |                   |
|-----|-------------------|
| 0,8 | 8-connected line. |
| 4   | 4-connected line. |
| 16  | antialiased line. |

**Return value** <Image> = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation** Cross drawing.

For grayscale image, the drawing brightness is set to the value of <B>.

If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

[Note] From Version 1.3.5, rotation direction is changed to clockwise.

# Text

**Format** *object.Text* <Output ID>, <X>, <Y>, <Text>, <R>, <G>, <B>, <Font>, <HScale>, <VScale>, <Shear>, <Thick>

**Parameters** <Output ID> = VT\_I4: Output Image ID

<X> = VT\_I4: Start point X coordinates

<Y> = VT\_I4: Start point Y coordinates

<Text> = VT\_BSTR: Displayed text

<R> = VT\_I4: Red density

<G> = VT\_I4: Green density

<B> = VT\_I4: Blue density

<Font> = VT\_I4: Font type

|   |                                |                                                                         |
|---|--------------------------------|-------------------------------------------------------------------------|
| 0 | CV_FONT_HERSHEY_SIMPLEX        | normal size sans-serif font                                             |
| 1 | CV_FONT_HERSHEY_PLAIN          | small size sans-serif font                                              |
| 2 | CV_FONT_HERSHEY_DUPLEX         | normal size sans-serif font (more complex than CV_FONT_HERSHEY_SIMPLEX) |
| 3 | CV_FONT_HERSHEY_COMPLEX        | normal size serif font                                                  |
| 4 | CV_FONT_HERSHEY_TRIPLEX        | normal size serif font (more complex than CV_FONT_HERSHEY_COMPLEX)      |
| 5 | CV_FONT_HERSHEY_COMPLEX_SMALL  | smaller version of CV_FONT_HERSHEY_COMPLEX                              |
| 6 | CV_FONT_HERSHEY_SCRIPT_SIMPLEX | hand-writing style font                                                 |
| 7 | CV_FONT_HERSHEY_SCRIPT_COMPLEX | more complex variant of CV_FONT_HERSHEY_SCRIPT_SIMPLEX                  |

<HScale> = VT\_R8: Width ratio

If equal to 1.0f, the characters have the original width depending on the font type.

If equal to 0.5f, the characters are of half the original width.

<VScale> = VT\_R8: Height ratio

If equal to 1.0f, the characters have the original height depending on the font type.

If equal to 0.5f, the characters are of half the original height.

<Shear> = VT\_R8: Relative angle from perpendicular line

Zero value means a non-italic font, 1.0f means  $\approx 45^\circ$  slope, etc. thickness

Thickness of lines composing letters outlines. The function `cvLine` is used for drawing letters.

**<Thick>** = VT\_I4: Thickness

**Return value** **<Image>** = VT\_UI1|VT\_ARRAY: Changed Image

**Explanation** Display character string.

For grayscale image, the drawing brightness is set to the value of **<B>**.

**< Shear >** is 0 degrees by 0.0, and 45 degrees by 1.0.

If Output Image ID=0, return value is changed image data. If Output Image Id  $\neq$  0, the change image is stored in the specified ID image memory, and return value is Empty.

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

#### 4.2.6. Contours

## FindContoursEx

**Format** *object*.FindContoursEx( **<Mode>**, **<Method>** )

**Parameters** **<Mode>** = VT\_I4: Retrieval mode

|   |                  |                                                                                                                                                                             |
|---|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | CV_RETR_EXTERNAL | retrive only the extreme outer contours                                                                                                                                     |
| 1 | CV_RETR_LIST     | retrieve all the contours and puts them in the list                                                                                                                         |
| 2 | CV_RETR_CCOMP    | retrieve all the contours and organizes them into two-level hierarchy: top level are external boundaries of the components, second level are bounda boundaries of the holes |
| 3 | CV_RETR_TREE     | retrieve all the contours and reconstructs the full hierarchy of nested contours                                                                                            |

**<Method>** = VT\_I4: Approximation method

|   |                      |                                                                                                       |
|---|----------------------|-------------------------------------------------------------------------------------------------------|
| 0 | CV_CHAIN_CODE        | output contours in the Freeman chain code. All other methods output polygons (sequences of vertices). |
| 1 | CV_CHAIN_APPROX_NONE | translate all the points from the chain code into points;                                             |
| 2 | CV_CHAIN_APPROX_S    | compress horizontal, vertical, and diagonal                                                           |

|   |                           |                                                                                                                                                                |
|---|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | IMPLE                     | segments, that is, the function leaves only their ending points;                                                                                               |
| 3 | CV_CHAIN_APPROX_TC89_L1   | apply one of the flavors of Teh-Chin chain approximation algorithm.                                                                                            |
| 4 | CV_CHAIN_APPROX_TC89_KCOS | apply one of the flavors of Teh-Chin chain approximation algorithm.                                                                                            |
| 5 | CV_LINK_RUNS              | use completely different contour retrieval algorithm via linking of horizontal segments of 1's. Only CV_RETR_LIST retrieval mode can be used with this method. |

**Return value**     **<Count>** = VT\_I4: Detection outline number

**Explanation**     Detect contour.  
Please refer to the descriptions of FindContours on OpenCV reference for details of the mode and the method.  
Color image is automatically converted to grayscale image.  
Detected contours are numbered from 0.

---

## CopyContours

---

**Format**             *object*. CopyContours <Output ID>, <Contour ID>

**Parameters**        **<Output ID>** = VT\_I4: Output memory ID  
**<Contour ID>** = VT\_I4: Outline ID

**Return value**       **<Image>** = VT\_UI1|VT\_ARRAY: Outline extraction image

**Explanation**       Copy contour image.  
If Output Image ID=0, return value is changed image data.  
If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.  
The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image. This command returns error if "FindContoursEx" is not executed beforehand.

**Error**                0x80101001     :    The contours is not detectable.  
Please execute the "FindContoursEx" command.

About the another error, please refer to the chapter 2.4.

---

## ContoursNumber

---

|              |                                                                                                                                                                                     |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> .ContoursNumber ( <X>, <Y> )                                                                                                                                          |
| Parameters   | <X> = VT_I4: X coordinates<br><Y> = VT_I4: Y coordinates                                                                                                                            |
| Return value | <Contour ID> = VT_I4: Outline ID                                                                                                                                                    |
| Explanation  | Retrieve Contour ID.<br>If the specified coordinate does not correspond to Contour ID, 0 is returned.<br>This command returns error if “FindContoursEx” is not executed beforehand. |
| Error        | 0x80101001 : The contours is not detectable.<br>Please execute the "FindContoursEx" command.<br>About the another error, please refer to the chapter 2.4.                           |

---

## PointPolygonTest

---

|              |                                                                                                                                                                                                                                                                                                                                                                                             |   |                        |     |                  |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|------------------------|-----|------------------|
| Format       | <i>object</i> .PointPolygonTest ( <Contour ID>, <X>, <Y>, <Measure distance> )                                                                                                                                                                                                                                                                                                              |   |                        |     |                  |
| Parameters   | <Contour ID> = VT_I4: Contour ID<br><X> = VT_I4: X coordinates<br><Y> = VT_I4: Y coordinates<br><Measure distance> = VT_I4: Distance measurement flag <table border="1"> <tr> <td>0</td><td>Don't measure distance</td></tr> <tr> <td>0&lt;&gt;</td><td>Measure distance</td></tr> </table>                                                                                                 | 0 | Don't measure distance | 0<> | Measure distance |
| 0            | Don't measure distance                                                                                                                                                                                                                                                                                                                                                                      |   |                        |     |                  |
| 0<>          | Measure distance                                                                                                                                                                                                                                                                                                                                                                            |   |                        |     |                  |
| Return value | <Distance> = VT_R8: Measured distance                                                                                                                                                                                                                                                                                                                                                       |   |                        |     |                  |
| Explanation  | Check the position relation of a point and a contour.<br>If the return value 'Distance' is negative, the point is at the inside of the polygon. If positive, it is at the outside. If 0, the point is on the contour of the polygon.<br>If Measuring the distance, it means the nearest neighborhood outline.<br>This command returns error if “FindContoursEx” is not executed beforehand. |   |                        |     |                  |
| Error        | 0x80101001 : The contours is not detectable.<br>Please execute the "FindContoursEx" command.                                                                                                                                                                                                                                                                                                |   |                        |     |                  |



About the another error, please refer to the chapter 2.4.

---

## BoundingRect

---

|              |                                                                                                                                                                                                                                                    |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> . BoundingRect ( <Contour ID> )                                                                                                                                                                                                      |
| Parameters   | <Contour ID> = VT_I4: Contour ID                                                                                                                                                                                                                   |
| Return value | <Rectangle> = VT_I4 VT_ARRAY: Rectangle which connots outline. (<X>, <Y>, <W>, <H>)<br><X> = VT_I4: X coordinate of rectangle left up corner<br><Y> = VT_I4: Y coordinate of rectangle left up corner<br><W> = VT_I4: Width<br><H> = VT_I4: Height |
| Explanation  | Find a rectangle bounding a contour.<br>This command returns error if “FindContoursEx” is not executed beforehand.                                                                                                                                 |
| Error        | 0x80101001 : The contours is not detectable.<br>Please execute the "FindContoursEx" command.<br>About the another error, please refer to the chapter 2.4.                                                                                          |

---

## FitEllipse

---

|              |                                                                                                                                                                                                                                           |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> . FitEllipse ( <Contour ID> )                                                                                                                                                                                               |
| Parameters   | <Contour ID> = VT_I4: Contour ID                                                                                                                                                                                                          |
| Return value | <Ellipse> = VT_VARIANT VT_ARRAY: 輪郭にフィットする最良楕円<br>(<X>, <Y>, <W>, <H>, <Angle>)<br><X> = VT_I4: Center X coordinates<br><Y> = VT_I4: Center Y coordinates<br><W> = VT_I4: Width<br><H> = VT_I4: Height<br><Angle> = VT_I4: Rotating angle |
| Explanation  | Get minimum ellipse bounding the specified contour.<br>This command returns error if “FindConoursEx” is not executed beforehand.<br>[Note] Return value is different from argument of Ellipse.                                            |

**Error**                    0x80101001 : Contour is not found. Please execute "FindContoursEx" command.

About another error, please refer to 2.4.

---

## ArcLength

---

**Format**                *object.ArcLength( <Contour ID> )*

**Parameters**        <Contour ID> = VT\_I4: Contour ID

**Return value**       <Length> = VT\_R8: Contour boundary length

**Explanation**       Get contour boundary length.  
This command returns error if "FindContoursEx" is not executed beforehand.

**Error**                    0x80101001 : Contour is not found. Please execute "FindContoursEx" command.

About another error, please refer to 2.4.

---

## CheckContourConvexity

---

**Format**                *object.CheckContourConvexity( <Contour ID> )*

**Parameters**        <Contour ID> = VT\_I4: Contour ID

**Return value**       <Convexity> = VT\_I4: Convexity check result

**Explanation**       Check shape convexity.  
If the contour is concave, the return value = 0; if the contour is convex, the return value = 1.  
This command returns error if "FindContoursEx" is not executed beforehand.

**Error**                    0x80101001 : Contour is not found. Please execute "FindContoursEx" command.

About another error, please refer to 2.4.

---

## DrawContours

---

**Format**                *object.DrawContours <Output ID>, <InputID>, <Contour ID>, <External R>, <External G>, <External B>, <Hole R>, <Hole G>, <Hole B>, <Max level>,*

---

**<Thick>, <Type>, <Offset X>, <Offset Y>**

**Parameters**

**<Output ID>** = VT\_I4: output image number  
**<Input ID>** = VT\_I4: input image number  
**<Contour ID>** = VT\_I4: contour number  
**<External R>** = VT\_I4: Red element of external contour  
**<External G>** = VT\_I4: Green element of external contour  
**<External B>** = VT\_I4: Blue element of external contour  
**<Hole R>** = VT\_I4: Red element of internal hole  
**<Hole G>** = VT\_I4: Green element of internal hole  
**<Hole B>** = VT\_I4: Blue element of internal hole  
**<Max level>** = VT\_I4: Maximum level of contour drawing

|     |                                                                                                                      |
|-----|----------------------------------------------------------------------------------------------------------------------|
| 0   | Only contour is drawn.                                                                                               |
| 0 < | Draw outlines with same level as contour, and child outline of contour until the level of abs(max_level)-1.          |
| 0 > | Draw child outline of contour until the level of abs(max_level)-1. Outlines with same level as contour is not drawn. |

**<Thick>** = VT\_I4: Thickness  
**<Type>** = VT\_I4: Line type  
**<Offset X>** = VT\_I4: X direction offset  
**<Offset Y>** = VT\_I4: Y direction offset

**Return value**

**<Image>** = VT\_UI1|VT\_ARRAY : Draw image

**Explanation**

Draw external contour or hole (internal) contour of the current image, and the drawn image is output to <Output ID> or the return value.

For grayscale image, the drawing brightness is set to the value of <B>.

When output image number is 0, the drawn image is output as a return value. When output image number is not 0, the drawn image is output the specified number, and returns Empty.

The image is drawn in the Windows standard bitmap file format. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

**Error**

0x80101001 : Contour is not found. Please execute "FindContoursEx" command.

About another error, please refer to 2.4.

## 4.2.7. Blob

## FindBlobs

|              |                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <code>object.FindBlobs( &lt;Mask ID&gt;, &lt;Threshold&gt;, &lt;Moments&gt; )</code>                                                                                                                                                                                                                                                                                                                    |
| Parameters   | <p><b>&lt;Mask ID&gt;</b> = VT_I4: Mask image ID</p> <p><b>&lt;Threshold&gt;</b> = VT_I4: Threshold</p> <p><b>&lt;Moments&gt;</b> = VT_BOOL: Flag whether it calculates moments or not</p>                                                                                                                                                                                                              |
| Return value | <b>&lt;Count&gt;</b> = VT_I4: Count of blobs                                                                                                                                                                                                                                                                                                                                                            |
| Explanation  | <p>Finds blobs.</p> <p>A color-scale image is converted to gray-scale image automatically.</p> <p>If the mask image ID is 0, then masking is not done.</p> <p>If &lt;Moments&gt; is true, then it calculates moment for each blob. Please note this takes more time.</p> <p>Found blobs are numbered from 0.</p> <p>An error is returned when the detected blob number exceeds a limit value (100).</p> |
| Error        | <p>0x80101002 : The detected blob number is over the limit value.</p> <p>About another error, please refer to 2.4.</p>                                                                                                                                                                                                                                                                                  |
| Related item | BlobsFilter, BlobResult, BlobResults, BlobEllipse, BlobMatchTemplate, BlobMatchShapes                                                                                                                                                                                                                                                                                                                   |

## BlobsFilter

|            |                                                                                                                                                                                                                     |   |         |   |         |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---------|---|---------|
| Format     | <code>object.BlobsFilter &lt;Action&gt;, &lt;Evaluator&gt;, &lt;Condition&gt;, &lt;Low Limit&gt;, &lt;High Limit&gt;</code>                                                                                         |   |         |   |         |
| Parameters | <p><b>&lt;Action&gt;</b> = VT_I4: Filter action</p> <table border="1"> <tr> <td>0</td><td>Include</td></tr> <tr> <td>1</td><td>Exclude</td></tr> </table> <p><b>&lt;Evaluator&gt;</b> = VT_I4: Evaluation items</p> | 0 | Include | 1 | Exclude |
| 0          | Include                                                                                                                                                                                                             |   |         |   |         |
| 1          | Exclude                                                                                                                                                                                                             |   |         |   |         |

|    |                          |    |                   |
|----|--------------------------|----|-------------------|
| 0  | Area                     | 1  | AreaEllipseRatio  |
| 2  | AxisRatio                | 3  | Breadth           |
| 4  | Compactness              | 5  | DiffX             |
| 6  | DiffY                    | 7  | DistanceFromPoint |
| 8  | Elongation               | 9  | Exterior          |
| 10 | ExternHullPerimeterRatio | 11 | ExternPerimeter   |
| 12 | ExternPerimeterRatio     | 13 | HullArea          |
| 14 | HullPerimeter            | 15 | Length            |
| 16 | MajorAxisLength          | 17 | MaxX              |
| 18 | MaxXatMaxY               | 19 | MaxY              |
| 20 | MaxYatMinX               | 21 | Mean              |
| 22 | MinorAxisLength          | 23 | MinX              |
| 24 | MinXatMinY               | 25 | MinY              |
| 26 | MinYatMaxX               | 27 | Moment            |
| 28 | Orientation              | 29 | OrientationCos    |
| 30 | Perimeter                | 31 | Roughness         |
| 32 | StdDev                   | 33 | XCenter           |
| 34 | XYInside                 | 35 | Ycenter           |

**<Condition>** = VT\_I4: Filter condition

|   |                  |   |               |
|---|------------------|---|---------------|
| 0 | Equal            | 1 | Not Equal     |
| 2 | Greater          | 3 | Less          |
| 4 | Greater or equal | 5 | Less or equal |
| 6 | Inside           | 7 | Outside       |

**<Low Limit>** = VT\_I4: Lower bound

**<High Limit>** = VT\_I4: Upper bound

**Return value** **<Count>** = VT\_I4: Count of blobs after filtering

**Explanation** Filter the blob list retrieved by FindBlobs. If FindBlobs was not done, then error occur.  
In case of no upper bound, please set 0 to <High Limit>.

**Error** 0x80101001 : Blob is not found. Please execute "FindContoursEx" command.  
About another error, please refer to 2.4.

**Related item** FindBlobs

# BlobResult

## Format

*object.* BlobResult( <Blob ID>, <Evaluator>, <Parameter1>, <Parameter2>)

## Parameters

<Blob ID> = VT\_I4: Blob ID

<Evaluator> = VT\_I4: Evaluation items

|    |                          |    |                   |
|----|--------------------------|----|-------------------|
| 0  | Area                     | 1  | AreaEllipseRatio  |
| 2  | AxisRatio                | 3  | Breadth           |
| 4  | Compactness              | 5  | DiffX             |
| 6  | DiffY                    | 7  | DistanceFromPoint |
| 8  | Elongation               | 9  | Exterior          |
| 10 | ExternHullPerimeterRatio | 11 | ExternPerimeter   |
| 12 | ExternPerimeterRatio     | 13 | HullArea          |
| 14 | HullPerimeter            | 15 | Length            |
| 16 | MajorAxisLength          | 17 | MaxX              |
| 18 | MaxXatMaxY               | 19 | MaxY              |
| 20 | MaxYatMinX               | 21 | Mean              |
| 22 | MinorAxisLength          | 23 | MinX              |
| 24 | MinXatMinY               | 25 | MinY              |
| 26 | MinYatMaxX               | 27 | Moment            |
| 28 | Orientation              | 29 | OrientationCos    |
| 30 | Perimeter                | 31 | Roughness         |
| 32 | StdDev                   | 33 | XCenter           |
| 34 | XYInside                 | 35 | Ycenter           |

<Parameter1> = VT\_I4: Parameter 1

<Parameter2> = VT\_I4: Parameter 2

## Return Value

<Value> = VT\_I4: Value of the specified item

## Explanation

Get value of the specified item from the result of FindBlobs.

Error occurs if this command is called before FindBlobs.

<Parameter1> and <Parameter2> depend on <Evaluator> as follows.

Table 4-4 Parameters of BlobResult

| Evluador             | Parameter1        | Parameter2        |
|----------------------|-------------------|-------------------|
| 7: DistanceFromPoint | X coordinates     | Y coordinates     |
| 27: Moment           | Number of X diff. | Number of Y diff. |
| 34: XYInside         | X coordinate      | Y coordinate      |

|        |     |     |
|--------|-----|-----|
| Others | n/a | n/a |
|--------|-----|-----|

Error 0x80101001 : The blob is not detectable.  
Please execute "FindBlobs" command.  
About the another error, please refer to the capter 2.4.

|              |           |
|--------------|-----------|
| Related item | FindBlobs |
|--------------|-----------|

## BlobResults

|        |                                         |
|--------|-----------------------------------------|
| Format | <i>object</i> .BlobResults( <Blob ID> ) |
|--------|-----------------------------------------|

Parameters      &lt;Blob ID&gt; = VT\_I4: Blob ID

Return value     **<Result>** = VT\_VARIANT|VT\_ARRAY: Blob result.

                  (**<Label>**, **<Exterior>**, **<Perimeter>**, **<External perimeter>**,  
                  **<Parent>**, **<M00>**, **<M10>**, **<M01>**, **<M20>**, **<M11>**, **<M02>**,  
                  **<Min X>**, **<Max X>**, **<Min Y>**, **<Max Y>**, **<Mean >**, **<StdDev>**)

**<Label>** = VT\_I4: Label of the blob.

**<Exterior>** = VT\_I4: True for extern blobs.

**<Perimeter>** = VT\_R8: Blob perimeter.

**<External perimeter>** = VT\_R8: Amount of blob perimeter which is exterior.

**<Parent>** = VT\_I4: Label of the parent blob.

**<M00>** = VT\_R8: Moments.

**<M10>**

**<M01>**

**<M20>**

**<M11>**

**<M02>**

**<Min X>** = VT\_R8: Bounding rect.

**<Max X>**

**<Min Y>**

**<Max Y>**

**<Mean>** = VT\_R8: Mean of the grey scale values of the blob pixels.

**<StdDev>** = VT\_R8: Standard deviation of the grey scale values of the blob pixels.

|                    |                                                                                                     |
|--------------------|-----------------------------------------------------------------------------------------------------|
| <b>Explanation</b> | Get the results of specified one blob from blob ID.<br>If FindBlobs was not done, then error occur. |
|--------------------|-----------------------------------------------------------------------------------------------------|

|              |                                                                                                                                      |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Error        | 0x80101001 : Blob is not found. Please execute "FindContoursEx" command.<br>About the another error, please refer to the capter 2.4. |
| Related item | FindBlobs                                                                                                                            |

---

## BlobEllipse

---

|              |                                                                                                                                                        |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> .BlobEllipse( <Blob ID> )                                                                                                                |
| Parameters   | <Blob ID> = VT_I4:Blob ID                                                                                                                              |
| Return value | <X> = VT_I4: Center X coordinates<br><Y> = VT_I4: Center Y coordinates<br><W> = VT_I4: Width<br><H> = VT_I4: Height<br><Angle> = VT_I4: Rotating angle |
| Explanation  | Get an ellipse fitting the blob.<br>If FindBlobs was not done, then error occur.                                                                       |
| Error        | 0x80101001 : Blob is not found. Please execute "FindContoursEx" command.<br>About the another error, please refer to the capter 2.4.                   |
| Related item | FindBlobs                                                                                                                                              |

---

## BlobMatchTemplate

---

|            |                                                                                                                                                                                                                                                                                                                                                                                                                 |                                                |              |                                                |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|--------------|------------------------------------------------|
| Format     | <i>object</i> .BlobMatchTemplate( <Input ID>, <Method>, <Threshold>, <Start angle>, <End angle>, <Step angle>, <Down sizing>, <Max count>, <Min distance> )                                                                                                                                                                                                                                                     |                                                |              |                                                |
| Parameters | <div><div>&lt;Input ID&gt; = VT_I4: Template Image ID</div><div>&lt;Method&gt; = VT_I4: Matching method</div><div><div>(I denotes image, T - template, R - result. The summation is done over template and/or the image patch: x'=0..w-1, y'=0..h-1)</div><table><tr><td>0</td><td>CV_TM_SQDIFF</td><td><math display="block">R(x,y)=\sum_{x',y'}[T(x',y')-I(x+x',y+y')]^2</math></td></tr></table></div></div> | 0                                              | CV_TM_SQDIFF | $R(x,y)=\sum_{x',y'}[T(x',y')-I(x+x',y+y')]^2$ |
| 0          | CV_TM_SQDIFF                                                                                                                                                                                                                                                                                                                                                                                                    | $R(x,y)=\sum_{x',y'}[T(x',y')-I(x+x',y+y')]^2$ |              |                                                |



|   |                     |                                                                                                                                                                                                                                                                  |
|---|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | CV_TM_SQDIFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') - I(x + x', y + y')]^2}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$                                                                                                                     |
| 2 | CV_TM_CCORR         | $R(x, y) = \sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]$                                                                                                                                                                                                    |
| 3 | CV_TM_CCORR_NORMED  | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$                                                                                                                   |
| 4 | CV_TM_CCOEFF        | $R(x, y) = \sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]$ <p>where</p> $T'(x', y') = T(x', y') - \frac{\sum_{x'', y''} T(x + x'', y + y'')}{(w \cdot h)}$ $I'(x + x', y + y') = I(x + x', y + y') - \frac{\sum_{x'', y''} I(x + x'', y + y'')}{w \cdot h}$ |
| 5 | CV_TM_CCOEFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}}$                                                                                                               |

<Threshold> = VT\_R8: Threshold

<Start angle> = VT\_I4: Start angle degree

<End angle> = VT\_I4: End angle degree

<Step angle> = VT\_I4: Step angle degree

<Down sizing> = VT\_I4: Down sizing count

<Max count> = VT\_I4: Number of detections

<Min distance> = VT\_I4: Minimum distance (0: Same as template image size)

Return value

<Points> = VT\_VARIANT|VT\_ARRAY: Detected point list

(<Point1>, <Point2>, ...)

<Pointn> = VT\_I4|VT\_ARRAY: Detected point (<X>, <Y>, <Value>)

<X> = VT\_I4: X coordinates

<Y> = VT\_I4: Y coordinates

<Angle> = VT\_R8: Rotating angle

<Value> = VT\_R8: Correlation value

**Explanation** Extended template matching between each detected blob and <Input ID> image.  
The process same as MatchTemplate2 command is used for extended template matching.

This command returns error before executing FindBlobs command.

[Note] From Version 1.3.5, rotation direction is changed to clockwise.

**Error** 0x80101001 : Blob is not found. Please execute "FindContoursEx" command.  
About the another error, please refer to the chapter 2.4.

**Related item** FindBlobs, MatchTemplate2

**Hint for high speed search**

- Use BlobsFilter to reduce the number of search target detect Blob number.

## BlobMatchShapes

**Format** *object*. BlobMatchShapes ( <Input ID>, <Method>, <Min scale>, <Similarity>, <Max count> )

**Parameters** <Input ID> = VT\_I4: Template Image ID

<Method> = VT\_I4: Matching method

A means original image and B means template image in the table below.

|   |                     |                                                                             |
|---|---------------------|-----------------------------------------------------------------------------|
| 0 | CV_CONTOUR_MATCH_I1 | $I_1(A, B) = \sum_{i=1}^7 \left  \frac{1}{m_i^A} - \frac{1}{m_i^B} \right $ |
| 1 | CV_CONTOUR_MATCH_I2 | $I_2(A, B) = \sum_{i=1}^7  m_i^A - m_i^B $                                  |
| 2 | CV_CONTOUR_MATCH_I3 | $I_3(A, B) = \sum_{i=1}^7 \frac{ m_i^A - m_i^B }{ m_i^A }$                  |

where

$$m_i^A = \sin(h_i^A) \cdot \log(h_i^A)$$

$$m_i^B = \sin(h_i^B) \cdot \log(h_i^B)$$

$h_i^A$ ,  $h_i^B$  are Hu moments of A and B, respectively.



|              |                                                                                                             |
|--------------|-------------------------------------------------------------------------------------------------------------|
| Parameters   | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram<br><b>&lt;Factor&gt;</b> = VT_R8: Normalization factor |
| Return value | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram                                                        |
| Explanation  | Normalize histogram.                                                                                        |
| Related item | CalcHistEx                                                                                                  |

---

## ThreshHistEx

---

|              |                                                                                                           |
|--------------|-----------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> ThreshHistEx( <b>&lt;Histogram&gt;</b> , <b>&lt;Threshold&gt;</b> )                        |
| Parameters   | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram<br><b>&lt;Threshold&gt;</b> = VT_R8: Threshold level |
| Return value | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram                                                      |
| Explanation  | This command clears histogram bins that are below the specified threshold.                                |
| Related item | CalcHistEx                                                                                                |

---

## EqualizeHistEx

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> EqualizeHistEx <b>&lt;Output ID&gt;</b>                                                                                                                                                                                                                                                                                                                                                                                   |
| Parameters   | <b>&lt;Output ID&gt;</b> = VT_I4: Output image ID                                                                                                                                                                                                                                                                                                                                                                                        |
| Return value | <b>&lt;Image&gt;</b> = VT_UI1 VT_ARRAY: Converted image                                                                                                                                                                                                                                                                                                                                                                                  |
| Explanation  | <p>The command normalizes brightness and increases contrast of the image.</p> <p>Color image is automatically converted to grayscale image.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> |

---

## GetMinMaxHistValue

---

|        |                                                               |
|--------|---------------------------------------------------------------|
| Format | <i>object.</i> GetMinMaxHistValue( <b>&lt;Histogram&gt;</b> ) |
|--------|---------------------------------------------------------------|

|              |                                                                                                                                                                                                                                                              |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Parameters   | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram                                                                                                                                                                                                         |
| Return value | <b>&lt;Min value&gt;</b> = VT_R4: Minimum value of histogram<br><b>&lt;Max value&gt;</b> = VT_R4: Maximum value of histogram<br><b>&lt;Min index&gt;</b> = VT_I4: Minimum value of luminance<br><b>&lt;Max index&gt;</b> = VT_I4: Maximum value of luminance |
| Explanation  | Get maximum and minimum value of histogram.                                                                                                                                                                                                                  |

---

## HistAve

---

|              |                                                      |
|--------------|------------------------------------------------------|
| Format       | <i>object.HistAve</i> ( <b>&lt;Histogram&gt;</b> )   |
| Parameters   | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram |
| Return value | <b>&lt;Average&gt;</b> = VT_R4: Average              |
| Explanation  | Calculate the average of histogram.                  |
| Related item | CalcHistEx                                           |

---

## AutoThreshPTile

---

|              |                                                                                                                                                                                                                                                                                                                |      |                |       |                 |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|----------------|-------|-----------------|
| Format       | <i>object.AutoThreshPTile</i> ( <b>&lt;Histogram&gt;</b> , <b>&lt;Rate&gt;</b> , <b>&lt;Forward&gt;</b> )                                                                                                                                                                                                      |      |                |       |                 |
| Parameters   | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram<br><b>&lt;Rate&gt;</b> = VT_R8: Area rate<br><b>&lt;Forward&gt;</b> = VT_BOOL: Search direction <table border="1" data-bbox="475 1458 842 1561"> <tr> <td>TRUE</td><td>Forward search</td></tr> <tr> <td>FALSE</td><td>Backward search</td></tr> </table> | TRUE | Forward search | FALSE | Backward search |
| TRUE         | Forward search                                                                                                                                                                                                                                                                                                 |      |                |       |                 |
| FALSE        | Backward search                                                                                                                                                                                                                                                                                                |      |                |       |                 |
| Return value | <b>&lt;Threshold&gt;</b> = VT_I4: Threshold value                                                                                                                                                                                                                                                              |      |                |       |                 |
| Explanation  | Calculate the threshold by Percentile Method.                                                                                                                                                                                                                                                                  |      |                |       |                 |
| Related item | CalcHistEx, AutoThreshMode, AutoThreshDiscrim                                                                                                                                                                                                                                                                  |      |                |       |                 |

---

## AutoThreshMode

---

|        |                                                           |
|--------|-----------------------------------------------------------|
| Format | <i>object.AutoThreshMode</i> ( <b>&lt;Histogram&gt;</b> ) |
|--------|-----------------------------------------------------------|

|              |                                                      |
|--------------|------------------------------------------------------|
| Parameters   | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram |
| Return value | <b>&lt;Threshold&gt;</b> = VT_I4: Threshold value    |
| Explanation  | Calculate the threshold by mode method.              |
| Related item | CalcHistEx, AutoThreshPTile, AutoThreshDiscrim       |

---

## AutoThreshDiscrim

---

|              |                                                              |
|--------------|--------------------------------------------------------------|
| Format       | <i>object.</i> AutoThreshDiscrim( <b>&lt;Histogram&gt;</b> ) |
| Parameters   | <b>&lt;Histogram&gt;</b> = VT_R8 VT_ARRAY: Histogram         |
| Return value | <b>&lt;Threshold&gt;</b> = VT_I4: Threshold value            |
| Explanation  | Calculate the threshold by discriminant analysis method.     |
| Related item | CalcHistEx, AutoThreshMode, AutoThreshPTile                  |

### 4.2.9. Matching

---

## MatchTemplate

---

|            |                                                                                                                |
|------------|----------------------------------------------------------------------------------------------------------------|
| Format     | <i>object.</i> MatchTemplate( <b>&lt;Input ID&gt;</b> , <b>&lt;Method&gt;</b> , <b>&lt;Result points&gt;</b> ) |
| Parameters | <b>&lt;Input ID&gt;</b> = VT_I4: Template Image ID<br><b>&lt;Method&gt;</b> = VT_I4: Matching method           |

(I denotes image, T - template, R - result. The summation is done over template and/or the image patch:  $x'=0..w-1$ ,  $y'=0..h-1$ )

|   |                     |                                                                                                                                              |
|---|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | CV_TM_SQDIFF        | $R(x, y) = \sum_{x', y'} [T(x', y') - I(x + x', y + y')]^2$                                                                                  |
| 1 | CV_TM_SQDIFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') - I(x + x', y + y')]^2}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$ |
| 2 | CV_TM_CCORR         | $R(x, y) = \sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]$                                                                                |

|   |                     |                                                                                                                                                                                                                                                                  |
|---|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3 | CV_TM_CCORR_NORMED  | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$                                                                                                                   |
| 4 | CV_TM_CCOEFF        | $R(x, y) = \sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]$ <p>where</p> $T'(x', y') = T(x', y') - \frac{\sum_{x'', y''} T(x + x'', y + y'')}{(w \cdot h)}$ $I'(x + x', y + y') = I(x + x', y + y') - \frac{\sum_{x'', y''} I(x + x'', y + y'')}{w \cdot h}$ |
| 5 | CV_TM_CCOEFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}}$                                                                                                               |

<Max count> = VT\_I4: Number of detection

Return value <Points> = VT\_VARIANT|VT\_ARRAY: Detected point list  
 (<Point1>, <Point2>, ...)  
 <Pointn> = VT\_I4|VT\_ARRAY: Detected point (<X>, <Y>, <Value>)  
 <X> = VT\_I4: X coordinates  
 <Y> = VT\_I4: Y coordinates  
 <Value> = VT\_R8: Correlation value

Explanation Template matching. Compares template against overlapped image regions.  
 Return values are the center point of a detected image which has the highest correlative values up to <Max count>.  
 [Note] The specification of the arguments and return value were changed from 1.3.1.

Related item MatchTemplate2, MatchShapesEx, MatchShapes2

Hint for high speed search • If the search target image is in a specific area, specify ROI to limit search area for faster search

## MatchShapesEx

**Format** `object.MatchShapesEx( <Input ID>, <Method> )`

**Parameters** `<Input ID>` = VT\_I4: Template Image ID

`<Method>` = VT\_I4: Matching method

A means original image and B means template image in the table below.

|   |                     |                                                                             |
|---|---------------------|-----------------------------------------------------------------------------|
| 0 | CV_CONTOUR_MATCH_I1 | $I_1(A, B) = \sum_{i=1}^7 \left  \frac{1}{m_i^A} - \frac{1}{m_i^B} \right $ |
| 1 | CV_CONTOUR_MATCH_I2 | $I_2(A, B) = \sum_{i=1}^7  m_i^A - m_i^B $                                  |
| 2 | CV_CONTOUR_MATCH_I3 | $I_3(A, B) = \sum_{i=1}^7 \frac{ m_i^A - m_i^B }{ m_i^A }$                  |

where

$$m_i^A = \sin(h_i^A) \cdot \log(h_i^A)$$

$$m_i^B = \sin(h_i^B) \cdot \log(h_i^B)$$

$h_i^A$ ,  $h_i^B$  are Hu moments of A and B, respectively.

**Return value** `<Similarity>` = VT\_R8: Correlation value of contours

**Explanation** Compares two shapes.

Perform shape matching between current image and <Input ID> image.

**Related item** MatchTemplate, MatchTemplate2, MatchShapes2

## MatchTemplate2

**Format** `object.MatchTemplate2( <Input ID>, <Method>, <Threshold>, <Start angle>, <End angle>, <Step angle>, <Down sizing>, <Max count>, <Min distance> )`

**Parameters** `<Input ID>` = VT\_I4: Template Image ID

`<Method>` = VT\_I4: Matching method

(I denotes image, T - template, R - result. The summation is done over template and/or the image patch: x'=0..w-1, y'=0..h-1)



|   |                     |                                                                                                                                                                                                                                                                  |
|---|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | CV_TM_SQDIFF        | $R(x, y) = \sum_{x', y'} [T(x', y') - I(x + x', y + y')]^2$                                                                                                                                                                                                      |
| 1 | CV_TM_SQDIFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') - I(x + x', y + y')]^2}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$                                                                                                                     |
| 2 | CV_TM_CCORR         | $R(x, y) = \sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]$                                                                                                                                                                                                    |
| 3 | CV_TM_CCORR_NORMED  | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$                                                                                                                   |
| 4 | CV_TM_CCOEFF        | $R(x, y) = \sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]$ <p>where</p> $T'(x', y') = T(x', y') - \frac{\sum_{x'', y''} T(x + x'', y + y'')}{(w \cdot h)}$ $I'(x + x', y + y') = I(x + x', y + y') - \frac{\sum_{x'', y''} I(x + x'', y + y'')}{w \cdot h}$ |
| 5 | CV_TM_CCOEFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}}$                                                                                                               |

<Threshold> = VT\_R8: Threshold

<Start angle> = VT\_I4: Start angle degree

<End angle> = VT\_I4: End angle degree

<Step angle> = VT\_I4: Step angle degree

<Down sizing> = VT\_I4: Down sizing count

<Max count> = VT\_I4: Number of detections

<Min distance> = VT\_I4: Minimum distance (0: Same as template image size)

Return value <Points> = VT\_VARIANT|VT\_ARRAY

: Detected point list (<Point1>, <Point2>, ...)

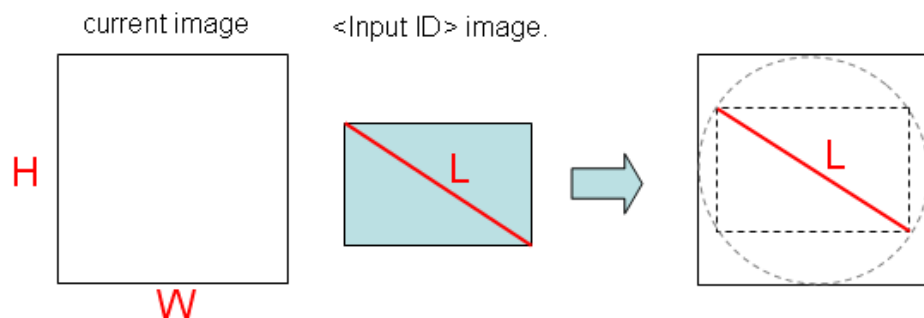
<Pointn> = VT\_I4|VT\_ARRAY: Detected point (<X>, <Y>, <Value>)

$\langle X \rangle$  = VT\_I4: X coordinates  
 $\langle Y \rangle$  = VT\_I4: Y coordinates  
 $\langle \text{Angle} \rangle$  = VT\_I4: Rotating angle  
 $\langle \text{Value} \rangle$  = VT\_R8: Correlation value

## Explanation

Perform extended template matching between current image and  $\langle \text{Input ID} \rangle$  image. Return values are the center point of a detected image which has the highest correlative values up to  $\langle \text{Max count} \rangle$ . If correlative value is below  $\langle \text{Threshold} \rangle$  or the center point is very near the highest correlative point, then the values are filtered. When either current image or template image is grayscale, another image is converted to grayscale first, and then template matching is executed. A current image and  $\langle \text{Input ID} \rangle$  image must satisfy the following conditions.

$$\min(W, H) \geq L$$



### [Note]

From Version 1.3.2, the specification of the arguments and return value were changed.

From Version 1.3.5, rotation direction is changed to clockwise.

## Related item

MatchTemplate, MatchShapesEx, MatchShapes2

## Hint for high speed search

- If the search target image is in a specific area, specify ROI to limit search area for faster search
- If the rotation angle is in a specific range, appropriately specify the start angle and end angle of matching search.
- To search a round shape, specify start angle and end angle to 0.

# MatchShapes2

Format `object.MatchShapes2( <Input ID>, <Method>, <Min scale>, <Similarity>, <Max count> )`

**Parameters**      **<Input ID>** = VT\_I4: Template Image ID

**<Method>** = VT\_I4: Matching method

A means original image and B means template image in the table below.

|   |                     |                                                                             |
|---|---------------------|-----------------------------------------------------------------------------|
| 0 | CV_CONTOUR_MATCH_I1 | $I_1(A, B) = \sum_{i=1}^7 \left  \frac{1}{m_i^A} - \frac{1}{m_i^B} \right $ |
| 1 | CV_CONTOUR_MATCH_I2 | $I_2(A, B) = \sum_{i=1}^7  m_i^A - m_i^B $                                  |
| 2 | CV_CONTOUR_MATCH_I3 | $I_3(A, B) = \sum_{i=1}^7 \frac{ m_i^A - m_i^B }{ m_i^A }$                  |

where

$$m_i^A = \sin(h_i^A) \cdot \log(h_i^A)$$

$$m_i^B = \sin(h_i^B) \cdot \log(h_i^B)$$

$h_i^A$ ,  $h_i^B$  are Hu moments of A and B, respectively.

**<Min scale>** = VT\_R8: Minimum scale

**<Similarity>** = VT\_R8: Correlation value of contours

**<Max count>** = VT\_I4: Number of detections

**Return value**      **<Points>** = VT\_VARIANT|VT\_ARRAY: Detected point list

(**<Point1>**, **<Point2>**, ...)

**<Pointn>** = VT\_I4|VT\_ARRAY : Detected point (**<X>**, **<Y>**, **<Angle>**, **<Similarity>**)

**<X>** = VT\_I4: X coordinates

**<Y>** = VT\_I4: Y coordinates

**<Angle>** = VT\_I4: Rotating angle

**<Similarity>** = VT\_R8: Correlation value of contours

**Explanation**

Perform the extended shape matching between current image and <Input ID> image.

The shape of the contour extracted from current image is matched with <Input ID> image, and the following information of the minimum ellipse that circumscribes the best matching contour, i.e., center coordinate, rotation angle and contour correlation value.

This function searches from detected contours whose correlation value is less than <Similarity>.

If current image or <Input ID> image is not binary image, contour cannot be correctly detected. Contour is extracted from the white colored part of the binary image. Please use binary images in which search target is white colored.

If two or more cantors are extracted from <Input ID> image, matching may fail. Therefore, <Input ID> image should be selected so that return value of the FindContours for the image is 1.

Small images whose size is less than “size of <Input ID> image” x “<Min scale>” are not searched by this function.

[Note] The specification of the auguments and return value were changed from 1.3.2.

Related item    MatchTemplate, MatchTemplate2, MatchShapesEx

---

## CamShift

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> CamShift( <Position>, <Max loop>, <Precision> )                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Parameters   | <p>&lt;Position&gt; = VT_I4 VT_ARRAY: Search start position (&lt;X&gt;, &lt;Y&gt;, &lt;W&gt;, &lt;H&gt;)</p> <p>&lt;X&gt; = VT_I4: X coordinates</p> <p>&lt;Y&gt; = VT_I4: Y coordinates</p> <p>&lt;W&gt; = VT_I4: Width</p> <p>&lt;H&gt; = VT_I4: Height</p> <p>&lt;Max loop&gt; = VT_I4: Number of occurrence</p> <p>&lt;Precision&gt; = VT_R8: Precision</p>                                                                                                                                                                                                                                                                                                                             |
| Return value | <p>&lt;Window&gt; = VT_I4 VT_ARRAY: Detection window (&lt;X&gt;, &lt;Y&gt;, &lt;W&gt;, &lt;H&gt;)</p> <p>&lt;X&gt; = VT_I4: X coordinates</p> <p>&lt;Y&gt; = VT_I4: Y coordinates</p> <p>&lt;W&gt; = VT_I4: Width</p> <p>&lt;H&gt; = VT_I4: Height</p> <p>&lt;Area&gt; = VT_R8: Sum of all pixels in the search window.</p> <p>&lt;Rectangle&gt; = VT_I4 VT_ARRAY: Circumscribing rectangle</p> <p style="text-align: right;">(&lt;X&gt;, &lt;Y&gt;, &lt;W&gt;, &lt;H&gt;, &lt;Angle&gt;)</p> <p>&lt;X&gt; = VT_I4: X coordinates</p> <p>&lt;Y&gt; = VT_I4: Y coordinates</p> <p>&lt;W&gt; = VT_I4: Width</p> <p>&lt;H&gt; = VT_I4: Height</p> <p>&lt;Angle&gt; = VT_I4: Rotating angle</p> |

|             |                  |
|-------------|------------------|
| Explanation | Object tracking. |
|-------------|------------------|

# HaarDetect

|              |                                                                                                                                                                                                                                                               |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> . HaarDetect ( <Path>, <Scale>, <MinNeighbors> )                                                                                                                                                                                                |
| Parameters   | <Path> = VT_BSTR: Haar file path<br><Scale> = VT_R8: scale<br><MinNeighbors> = VT_I4: Minimum neighbors number                                                                                                                                                |
| Return value | <Points> = VT_VARIANT VT_ARRAY: Detected point list<br>(<Point1>, <Point2>, ...)<br><Pointn> = VT_I4 VT_ARRAY: Detected point ( <X>, <Y>, <W>, <H> )<br><X> = VT_I4: X coordinates<br><Y> = VT_I4: Y coordinates<br><W> = VT_I4: Width<br><H> = VT_I4: Height |

|                    |                                                                                                                                                                                                                 |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Explanation</b> | <p>Perform Haar matching to the current image, and return the list of the detected objects.</p> <p>The function return Empty if no object is detected. In this case, the result of the function is S_FALSE.</p> |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

#### 4.2.10. CARD

# CARDInit2

[V1.5.0 or later]

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <b><i>object.</i> CARDInit</b> (<Input ID>, <X>, <Y> )                                                                                                                                                                                                                                                                                                                                                                                            |
| Parameters   | <p>&lt;Input ID&gt; = VT_I4: Image ID of the template image.</p> <p>&lt;X&gt; = VT_I4: The X coordinate of the detecting point in the template image.</p> <p>&lt;Y&gt; = VT_I4: The Y coordinate of the detecting point in the template image.</p>                                                                                                                                                                                                |
| Return value | <Count> = VT_I4: The number of the feature points of the template image.                                                                                                                                                                                                                                                                                                                                                                          |
| Explanation  | <p>A template image is registered as initialization of CARD.</p> <p>For &lt;X&gt; and &lt;Y&gt;, specify two points that are detected as a result of CARDRun1.</p> <p>When -1 is specified, set these points to the center of the template image automatically.</p> <p>&lt;Count&gt; returns the number of feature points of the template image returned by CARDRun2. When you change the template image, please execute this function again.</p> |



For the template image, specify an image which size is between 2000 pixels and three million pixels.

For <X> and <Y>, specify respective point that will be detected as a execution result of CARDRunMulti.

<Count> returns the number of the feature points of the template image used by CARDRunMulti. To change the template image, execute the command again.

If "0" is specified to <Intermediate ID>, a processing image is not created. If you prefer high-speed operation, set this item to "0".

|       |            |   |                                                                                                       |
|-------|------------|---|-------------------------------------------------------------------------------------------------------|
| Error | 0x80100005 | : | The template image size is too large. Reduce the template image size to three million pixels or less. |
|       | 0x80101102 | : | The template image size is too small. Set the template image size to 2000 pixels or more.             |

For about other errors, refer to 2.4.

## CARDRunMulti

[V1.5.3 or later]

|              |                                                                                                                                                                                                                                                                                                                                               |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> CARDRunMulti ( <Threshold>, <Count>, <Min distance>, <Intermediate ID> )                                                                                                                                                                                                                                                       |
| Parameters   | <Threshold> = VT_R8: Threshold value<br><Count> = VT_R8: Detected number<br><Min distance> = VT_I4: Minimum distance (default : -1)<br><Intermediate ID> = VT_R8: Output destination ID that stores a processing image at CARD detection (default:0)                                                                                          |
| Return value | <Points> = VT_VARIANT VT_ARRAY: Detection point list (<Point1>, <Point2>, ...)<br><Point1> = VT_R8 VT_ARRAY: Detection point<br>(<X>, <Y>, <Angle>, <Scale>, <Value>)<br><X> = VT_R8: X-coordinates<br><Y> = VT_R8: Y-coordinates<br><Angle> = VT_R8: Rotating angle (degree)<br><Scale> = VT_R8: Scale<br><Value> = VT_R8: Correlation value |
| Explanation  | Perform image detection by CARD.<br>CARDInitMulti needs to be performed before executing this command.                                                                                                                                                                                                                                        |

The size of detection target image must be three million pixels or less.

An object whose correlation value is smaller than the value specified by <Threshold> will be excluded from the detection result.

If the distance between the center of detected objects is less than the value specified in <Min distance>, the one with smaller correlation value will be excluded from the searching result, that prevents to count identical object two times or more.

If "-1" is specified to <Min distance>, half the size of width or heights of template image whichever the smaller will be applied. (Example: 640x480 > <Min distance> = 240)

If "0" is specified to <Min distance>, the exclusion based on the distance between objects will not be done.

The detection result will be stored with the decending order of the correlation value.

If there is no detected point, "Empty" will be returned.

If "0" is specified to <Intermediate ID>, a processing image is not created. If you prefer high-speed operation, set this item to "0".

## Error

0x80101001 : CARD is not initilized.

Execute CARDInitMulti command

0x80100005 : The template image size is too large. Reduce the template image size to three million pixels or less.

For about other errors, refer to 2.4.

### 4.2.11. CAL

## CalibrateCamera

**Format** *object*. CalibrateCamera <Input ID>, <Count>, <Square count W>, <Square count H>, <Grid Size>, <Flag>, <Camera CAL ID>

**Parameters** <Input ID> = VT\_I4: First chessboard image(reference image)

<Count> = VT\_I4: Number of chessboard images

<Square count W> = VT\_I4: Number of squares (Horizontal)

<Square count H> = VT\_I4: Number of squares (Vertical)

<Grid Size> = VT\_R8: Grid size

<Flag> = VT\_I4: Flag

|   |                             |                                                                                                                                                    |
|---|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | CV_CALIB_CB_ADAPTIVE_THRESH | Use adaptive thresholding to convert the image to black-n-white, rather than a fixed threshold level (computed from the average image brightness). |
|---|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|



|   |                             |                                                                                                                                                        |
|---|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2 | CV_CALIB_CB_NORMALIZE_IMAGE | Normalize the image using cvNormalizeHist before applying fixed or adaptive thresholding.                                                              |
| 4 | CV_CALIB_CB_FILTER_QUADS    | Use additional criteria (like contour area, perimeter, square-like shape) to filter out false quads that are extracted at the contour retrieval stage. |

<Camera CAL ID> = VT\_I4: Camera calibration ID

Return value    None

**Explanation**    Calibrate camera parameters.

The function calculates the intrinsic camera parameters from the specified chessboard images, and calculates the extrinsic camera parameters from <Input ID> image. It requires 5 chessboard images or more. <Square Count> is a number of boxes. The results are stored in the database automatically.

**Related item**    FindChessBoardCorners, SetCamCalDat, GetCamCalDat, SetCamCalExtDat, GetCamCalExtDat, ModifyCamCalExtDat, GetPosFromCam, GetCamPos, GetRobPosFromCam, GetCamPosFromRob, Undistort2

---

## CalibrateRobot

---

**Format**    *object.* CalibrateRobot <Robot CAL ID>, <Point1>, <Point2>, <Point3>

**Parameters**    <Robot CAL ID> = VT\_I4: Robot calibration ID

<Points> = VT\_VARIANT|VT\_ARRAY : World – Robot correspondence point list  
(<Point1>, <Point2>, ...)

<Pointn> = VT\_VARIANT|VT\_ARRAY : World – Robot correspondence point  
(<World Point>, <Robot Point>)

<World Point> = VT\_R8|VT\_ARRAY : World coordinate (<X>, <Y>, <Z>)

<X> = VT\_R8 : X coordinate

<Y> = VT\_R8 : Y coordinate

<Z> = VT\_R8 : Z coordinate

<Robot Point> = VT\_R8|VT\_ARRAY : Robot coordinate (<X>, <Y>, <Z>)

<X> = VT\_R8 : X coordinate

<Y> = VT\_R8 : Y coordinate

<Z> = VT\_R8 : Z coordinate

|              |                                                                                                                                           |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| Return value | None                                                                                                                                      |
| Explanation  | Perform robot calibration.<br>Specify arbitraly numbers of World – Robot coordinate corresponding points, and calculate calibration data. |
| Related item | SetRobCalDat, GetRobCalDat, GetPosFromRob, GetRobPos, GetRobPosFromCam, GetCamPosFromRob                                                  |

## FindChessBoardCorners

|        |                                                                                 |
|--------|---------------------------------------------------------------------------------|
| Format | <i>object</i> .FindChessBoardCorners <Square Count W>, <Square Count H>, <Flag> |
|--------|---------------------------------------------------------------------------------|

Parameters      **<Square Count W>** = VT\_I4: Number of squares (Horizontal)  
                      **<Square Count H>** = VT\_I4: Number of squares (Vertical)  
                      **<Flag>** = VT\_I4: Flag

|   |                             |                                                                                                                                                        |
|---|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | CV_CALIB_CB_ADAPTIVE_THRESH | Use adaptive thresholding to convert the image to black-n-white, rather than a fixed threshold level (computed from the average image brightness).     |
| 2 | CV_CALIB_CB_NORMALIZE_IMAGE | Normalize the image using cvNormalizeHist before applying fixed or adaptive thresholding.                                                              |
| 4 | CV_CALIB_CB_FILTER_QUADS    | Use additional criteria (like contour area, perimeter, square-like shape) to filter out false quads that are extracted at the contour retrieval stage. |

**Return value**

- <**Pattern was found**> = VT\_BOOL: Detection result (0: Fail, <>0: Success)
- <**Points**> = VT\_VARIANT|VT\_ARRAY: Detected point list
  - (<**Point1**>, <**Point2**>, ...)
- <**Pointn**> = VT\_I4|VT\_ARRAY: Detected point (<**X**>, <**Y**>)
  - <**X**> = VT\_I4: X Coordinate
  - <**Y**> = VT\_I4: Y Coordinate

|              |                                            |
|--------------|--------------------------------------------|
| Explanation  | Find corners of chessboard from the image. |
| Related item | CalibrateCamera, DrawChessBoardCorners     |

## DrawChessBoardCorners

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |   |      |     |         |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|------|-----|---------|
| Format       | <i>object</i> . DrawChessBoardCornersEx <Output ID>, <Square Count W>, <Square Count H>, <Pattern was found > ,<Points>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |   |      |     |         |
| Parameters   | <p>&lt;Output ID&gt; = VT_I4: Output image number</p> <p>&lt;Square Count W&gt; = VT_I4: Number of squares (Horizontal)</p> <p>&lt;Square Count H&gt; = VT_I4: Number of squares (Vertical)</p> <p>&lt;Pattern was found&gt; = VT_BOOL: Detection result</p> <table border="1"> <tr> <td>0</td><td>Fail</td></tr> <tr> <td>&lt;&gt;0</td><td>Success</td></tr> </table> <p>&lt;Points&gt; = VT_VARIANT VT_ARRAY : Detected point list<br/>(&lt;Point1&gt;, &lt;Point2&gt;, ...)</p> <p>&lt;Pointn&gt; = VT_I4 VT_ARRAY: Detected point (&lt;X&gt;, &lt;Y&gt;)</p> <p>&lt;X&gt; = VT_I4: X Coordinate</p> <p>&lt;Y&gt; = VT_I4: Y Coordinate</p> | 0 | Fail | <>0 | Success |
| 0            | Fail                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |   |      |     |         |
| <>0          | Success                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |   |      |     |         |
| Return value | <Image> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |   |      |     |         |
| Explanation  | <p>Draw chessboard corner detection result.</p> <p>If corners are completely detected, colored corners are displayed by connected line. If complete detection was failed, failed corners are shown in red circle.</p> <p>If Output Image ID=0, return value is changed image data.</p> <p>If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p>                              |   |      |     |         |
| Related item | FindChessBoardCorners                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |   |      |     |         |

## DrawXYAxes

|            |                                                                                                                                             |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| Format     | <i>object</i> . DrawChessBoardCornersEx <Output ID>, <Camera CAL ID>, <R>, <G> , <B>                                                        |
| Parameters | <p>&lt;Output ID&gt; = VT_I4: Output image number</p> <p>&lt;Camera CAL ID&gt; = VT_I4: Camera ID</p> <p>&lt;R&gt; = VT_I4: Red density</p> |

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | <p><b>&lt;G&gt;</b> = VT_I4: Green density</p> <p><b>&lt;B&gt;</b> = VT_I4: Blue density</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Return value | <b>&lt;Image&gt;</b> = VT_UI1 VT_ARRAY: Changed Image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Explanation  | <p>Draw X and Y axis which is based on calibration data.</p> <p>For grayscale image, the drawing brightness is set to the value of <b>&lt;B&gt;</b>.</p> <p>If Output Image ID=0, return value is changed image data. If Output Image Id &lt;&gt; 0, the change image is stored in the specified ID image memory, and return value is Empty.</p> <p>The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.</p> |
| Related item | CalibrateCamera                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |

---

## SetCamCalDat

---

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format     | <i>object.</i> <b>SetCamCalDat</b> <b>&lt;Intrinsic matrix&gt;</b> , <b>&lt;Distortion coeffs&gt;</b> , <b>&lt;Extrinsic matrix&gt;</b> , <b>&lt;Camera CAL ID&gt;</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Parameters | <p><b>&lt;Intrinsic matrix&gt;</b> = VT_R8 VT_ARRAY: Internal parameters<br/>(<b>&lt;fx&gt;</b>, <b>&lt;fy&gt;</b>, <b>&lt;cx&gt;</b>, <b>&lt;cy&gt;</b>)</p> <p><b>&lt;fx&gt;</b> = VT_R8: Focus distance X</p> <p><b>&lt;fy&gt;</b> = VT_R8: Focus distance Y</p> <p><b>&lt;cx&gt;</b> = VT_R8: Center coordinate X</p> <p><b>&lt;cy&gt;</b> = VT_R8: Center coordinate Y</p> <p><b>&lt;Distortion coeffs&gt;</b> = VT_R8 VT_ARRAY: Distortion Coeffs<br/>(<b>&lt;k1&gt;</b>, <b>&lt;k2&gt;</b>, <b>&lt;p1&gt;</b>, <b>&lt;p2&gt;</b>)</p> <p><b>&lt;k1&gt;</b> = VT_R8: Distortion coefficients in radius direction</p> <p><b>&lt;k2&gt;</b> = VT_R8: Distortion coefficients in radius direction</p> <p><b>&lt;p1&gt;</b> = VT_R8: Distortion coefficients in circumference direction</p> <p><b>&lt;p2&gt;</b> = VT_R8: Distortion coefficients in circumference direction</p> <p><b>&lt;Extrinsic matrix&gt;</b> = VT_R8 VT_ARRAY: External parameters<br/>(<b>&lt;r11&gt;</b>, <b>&lt;r21&gt;</b>, <b>&lt;r31&gt;</b>, <b>&lt;r12&gt;</b>, <b>&lt;r22&gt;</b>, <b>&lt;r32&gt;</b>, <b>&lt;r13&gt;</b>, <b>&lt;r23&gt;</b>, <b>&lt;r33&gt;</b>, <b>&lt;dx&gt;</b>, <b>&lt;dy&gt;</b>, <b>&lt;dz&gt;</b>)</p> <p><b>&lt;r11&gt;</b> = VT_R8: Rotation vector</p> <p><b>&lt;r21&gt;</b> = VT_R8:</p> <p><b>&lt;r31&gt;</b> = VT_R8:</p> |



$\langle k2 \rangle$  = VT\_R8: Radius direction distortion coefficients  
 $\langle p1 \rangle$  = VT\_R8: Circumference direction distortion coefficients  
 $\langle p2 \rangle$  = VT\_R8: Circumference direction distortion coefficients  
 $\langle \text{Extrinsic matrix} \rangle$  = VT\_R8|VT\_ARRAY: External parameters  
 $(\langle r11 \rangle, \langle r21 \rangle, \langle r31 \rangle, \langle r12 \rangle, \langle r22 \rangle, \langle r32 \rangle, \langle r13 \rangle, \langle r23 \rangle, \langle r33 \rangle, \langle dx \rangle, \langle dy \rangle, \langle dz \rangle)$   
 $\langle r11 \rangle$  = VT\_R8: Rotation vector  
 $\langle r21 \rangle$  = VT\_R8:  
 $\langle r31 \rangle$  = VT\_R8:  
 $\langle r12 \rangle$  = VT\_R8:  
 $\langle r22 \rangle$  = VT\_R8:  
 $\langle r32 \rangle$  = VT\_R8:  
 $\langle r13 \rangle$  = VT\_R8:  
 $\langle r23 \rangle$  = VT\_R8:  
 $\langle r33 \rangle$  = VT\_R8:  
 $\langle dx \rangle$  = VT\_R8: Translational vector  
 $\langle dy \rangle$  = VT\_R8:  
 $\langle dz \rangle$  = VT\_R8:

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} r11 & r12 & r13 & dx \\ r21 & r22 & r23 & dy \\ r31 & r32 & r33 & dz \end{pmatrix} \times \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

**Explanation** Get internal and external parameters and distortion coefficients from database.

**Related item** CalibrateCamera, SetCamCalDat, SetCamCalExtDat, GetCamCalExtDat, ModifyCamCalExtDat, GetPosFromCam, GetCamPos, GetRobPosFromCam, GetCamPosFromRob, Undistort2

---

## SetCamCalExtDat

---

**Format** *object*.SetCamCalExtDat  $\langle \text{Extrinsic matrix} \rangle$ ,  $\langle \text{Camera CAL ID} \rangle$

**Parameters**  $\langle \text{Extrinsic matrix} \rangle$  = VT\_R8|VT\_ARRAY: External parameter  
 $(\langle r11 \rangle, \langle r21 \rangle, \langle r31 \rangle, \langle r12 \rangle, \langle r22 \rangle, \langle r32 \rangle, \langle r13 \rangle, \langle r23 \rangle, \langle r33 \rangle, \langle dx \rangle, \langle dy \rangle, \langle dz \rangle)$   
 $\langle r11 \rangle$  = VT\_R8: Rotation vector  
 $\langle r21 \rangle$  = VT\_R8:  
 $\langle r31 \rangle$  = VT\_R8:  
 $\langle r12 \rangle$  = VT\_R8:

$\langle r22 \rangle = VT\_R8$   
 $\langle r32 \rangle = VT\_R8$   
 $\langle r13 \rangle = VT\_R8$   
 $\langle r23 \rangle = VT\_R8$   
 $\langle r33 \rangle = VT\_R8$   
 $\langle dx \rangle = VT\_R8$ : Translational vector  
 $\langle dy \rangle = VT\_R8$   
 $\langle dz \rangle = VT\_R8$

$\langle \text{Camera CAL ID} \rangle = VT\_I4$ : Camera calibration ID

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} r11 & r12 & r13 & dx \\ r21 & r22 & r23 & dy \\ r31 & r32 & r33 & dz \end{pmatrix} \times \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Return value    None

Explanation    Set external parameters to database

Related item    CalibrateCamera,    SetCamCalDat,    GetCamCalDat,    GetCamCalExtDat,  
 ModifyCamCalExtDat,    GetPosFromCam,    GetCamPos,    GetRobPosFromCam,  
 GetCamPosFromRob, Undistort2

---

## GetCamCalExtDat

---

Format    *object.* GetCamCalExtDat(  $\langle \text{Inverse} \rangle$ ,  $\langle \text{Camera CAL ID} \rangle$  )

Parameters     $\langle \text{Inverse} \rangle = VT\_BOOL$  : Inverse matrix flag

Return value     $\langle \text{Extrinsic matrix} \rangle = VT\_R8|VT\_ARRAY$ : Extrinsic parameter  
 $(\langle r11 \rangle, \langle r21 \rangle, \langle r31 \rangle, \langle r12 \rangle, \langle r22 \rangle, \langle r32 \rangle, \langle r13 \rangle, \langle r23 \rangle, \langle r33 \rangle, \langle dx \rangle, \langle dy \rangle, \langle dz \rangle)$   
 $\langle r11 \rangle = VT\_R8$ : Rotation vector  
 $\langle r21 \rangle = VT\_R8$ :  
 $\langle r31 \rangle = VT\_R8$ :  
 $\langle r12 \rangle = VT\_R8$ :  
 $\langle r22 \rangle = VT\_R8$ :  
 $\langle r32 \rangle = VT\_R8$ :  
 $\langle r13 \rangle = VT\_R8$ :  
 $\langle r23 \rangle = VT\_R8$ :  
 $\langle r33 \rangle = VT\_R8$ :

<dx> = VT\_R8: Translation vector

<dy> = VT\_R8:

<dz> = VT\_R8:

<Camera CAL ID> = VT\_I4: Camera calibration ID

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} r11 & r12 & r13 & dx \\ r21 & r22 & r23 & dy \\ r31 & r32 & r33 & dz \end{pmatrix} \times \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

**Explanation** Get extrinsic parameters from database.

If <Inverse> is TRUE, inverse of extrinsic matrix is returned.

**Related item** CalibrateCamera, SetCamCalDat, GetCamCalDat, SetCamCalExtDat, ModifyCamCalExtDat, GetPosFromCam, GetCamPos, GetRobPosFromCam, GetCamPosFromRob, Undistort2

## ModifyCamCalExtDat

**Format** *object.* ModifyCamCalExtDat <Input ID>, <Square count W>, <Square count H>, <Grid Size>, <Flag>, <Camera CAL ID>

**Parameters** <Input ID> = VT\_I4: Chess board image

<Square Count W> = VT\_I4: Number of squares (Horizontal)

<Square Count H> = VT\_I4: Number of squares (Vertical)

<Flag> = VT\_I4: Flag

|   |                             |                                                                                                                                                        |
|---|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | CV_CALIB_CB_ADAPTIVE_THRESH | Use adaptive thresholding to convert the image to black-n-white, rather than a fixed threshold level (computed from the average image brightness).     |
| 2 | CV_CALIB_CB_NORMALIZE_IMAGE | Normalize the image using cvNormalizeHist before applying fixed or adaptive thresholding.                                                              |
| 4 | CV_CALIB_CB_FILTER_QUADS    | Use additional criteria (like contour area, perimeter, square-like shape) to filter out false quads that are extracted at the contour retrieval stage. |

<Camera CAL ID> = VT\_I4: Camera calibration ID

**Return value** None



**Explanation** Update external parameter using the specified image.

**Related item** CalibrateCamera, FindChessBoardCorners, SetCamCalDat, GetCamCalDat, SetCamCalExtDat, GetCamCalExtDat, GetPosFromCam, GetCamPos, GetRobPosFromCam, GetCamPosFromRob, Undistort2

---

## SetRobCalDat

---

**Format** *object*. SetRobCalDat <Robot CAL ID>, <r11>, <r21>, <r31>, <r12>, <r22>, <r32>, <r13>, <r23>, <r33>, <dx>, <dy>, <dz>

**Parameters** <Robot CAL ID> = VT\_I4: Robot calibration ID  
(<r11>, <r21>, <r31>, <r12>, <r22>, <r32>, <r13>, <r23>, <r33>, <dx>, <dy>, <dz>):

Homogeneous translation matrix

<r11> = VT\_R8: Rotation Vector

<r21> = VT\_R8:

<r31> = VT\_R8:

<r12> = VT\_R8:

<r22> = VT\_R8:

<r32> = VT\_R8:

<r13> = VT\_R8:

<r23> = VT\_R8:

<r33> = VT\_R8:

<dx> = VT\_R8: Translation Vector

<dy> = VT\_R8:

<dz> = VT\_R8:

$$\begin{pmatrix} Xr \\ Yr \\ Zr \\ 1 \end{pmatrix} = \begin{pmatrix} r11 & r12 & r13 & dx \\ r21 & r22 & r23 & dy \\ r31 & r32 & r33 & dz \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} Xw \\ Yw \\ Zw \\ 1 \end{pmatrix}$$

**Return value** None

**Explanation** Set robot calibration data to database.

**Related item** CalibrateRobot, GetRobCalDat, GetPosFromRob, GetRobPos, GetRobPosFromCam, GetCamPosFromRob

---

## GetRobCalDat

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> .GetRobCalDat( <Robot CAL ID>, <Inverse> )                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Parameters   | <Robot CAL ID> = VT_I4: Robot calibration ID<br><Inverse> = VT_BOOL : Inverse matrix flag                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Return value | <Matrix> = VT_R8 VT_ARRAY: Homogeneous translation matrix<br>(<r11>, <r21>, <r31>, <r12>, <r22>, <r32>, <r13>, <r23>, <r33>, <dx>, <dy>, <dz>)<br><r11> = VT_R8: Rotation Vector<br><r21> = VT_R8:<br><r31> = VT_R8:<br><r12> = VT_R8:<br><r22> = VT_R8:<br><r32> = VT_R8:<br><r13> = VT_R8:<br><r23> = VT_R8:<br><r33> = VT_R8:<br><dx> = VT_R8: Translation Vector<br><dy> = VT_R8:<br><dz> = VT_R8:<br>$\begin{pmatrix} Xr \\ Yr \\ Zr \\ 1 \end{pmatrix} = \begin{pmatrix} r11 & r12 & r13 & dx \\ r21 & r22 & r23 & dy \\ r31 & r32 & r33 & dz \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} Xw \\ Yw \\ Zw \\ 1 \end{pmatrix}$ |
| Explanation  | Get robot calibration data from database.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Related item | CalibrateRobot, SetRobCalDat, GetPosFromRob, GetRobPos, GetRobPosFromCam, GetCamPosFromRob                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

---

## SetCamDescription

---

|            |                                                                                        |
|------------|----------------------------------------------------------------------------------------|
| Format     | <i>object</i> .SetCamDescription <Camera CAL ID>, <Description>                        |
| Parameters | <Camera CAL ID> = VT_I4: Camera calibarion ID<br><Description> = VT_BSTR : Description |

|              |                                                                                                                                                                                  |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return value | None                                                                                                                                                                             |
| Explanation  | Write camera calibration description into the database.                                                                                                                          |
| Related item | CalibrateCamera, SetCamCalDat, GetCamCalDat, SetCamCalExtDat,<br>GetCamCalExtDat, GetCamDescription, GetPosFromCam, GetCamPos,<br>GetRobPosFromCam, GetCamPosFromRob, Undistort2 |

---

## GetCamDescription

---

|              |                                                                                                                                                                                  |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> GetCamDescription(<Camera CAL ID>)                                                                                                                                |
| Parameters   | <Camera CAL ID> = VT_I4: Camera calibarion ID                                                                                                                                    |
| Return value | <Description> = VT_BSTR : Description                                                                                                                                            |
| Explanation  | Read camera calibration description from the database.                                                                                                                           |
| Related item | CalibrateCamera, SetCamCalDat, GetCamCalDat, SetCamCalExtDat,<br>GetCamCalExtDat, SetCamDescription, GetPosFromCam, GetCamPos,<br>GetRobPosFromCam, GetCamPosFromRob, Undistort2 |

---

## SetRobDescription

---

|              |                                                                                                                                |
|--------------|--------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.</i> SetRobDescription <Robot CAL ID>, <Description>                                                                 |
| Parameters   | <Robot CAL ID> = VT_I4: Robot calibration ID<br><Description> = VT_BSTR : Description                                          |
| Return value | None                                                                                                                           |
| Explanation  | Write robot calibration description into the database.                                                                         |
| Related item | CalibrateRobot, SetRobCalDat, GetRobCalDat, GetRobDescription, GetPosFromRob,<br>GetRobPos, GetRobPosFromCam, GetCamPosFromRob |

---

## GetRobDescription

---

|            |                                                  |
|------------|--------------------------------------------------|
| Format     | <i>object.</i> GetRobDescription(<Robot CAL ID>) |
| Parameters | <Robot CAL ID> = VT_I4: Robot calibration ID     |

|              |                                                                                                                             |
|--------------|-----------------------------------------------------------------------------------------------------------------------------|
| Return value | <b>&lt;Description&gt;</b> = VT_BSTR : Description                                                                          |
| Explanation  | Read robot calibration description from the database.                                                                       |
| Related item | CalibrateRobot, SetRobCalDat, GetRobCalDat, SetRobDescription, GetPosFromRob, GetRobPos, GetRobPosFromCam, GetCamPosFromRob |

---

## GetPosFromCam

---

Format `object.GetPosFromCam( <Xc>, <Yc>[, <Zoffset>, <Camera CAL ID>, <Undistort>] )`

Parameters

**<Xc>** = VT\_R8: X on camera coordinate

**<Yc>** = VT\_R8: Y on camera coordinate

**<Zoffset>** = VT\_R8: Z on world coordinate

**<Camera CAL ID>** = VT\_I4: Camera calibration ID (Default: 0)

**<Undistort>** = VT\_BOOL: Undistortion(Default: False)

Return value

**<Xw>** = VT\_R8: X on world coordinate

**<Yw>** = VT\_R8: Y on world coordinate

**<Zw>** = VT\_R8: Z on world coordinate (Same value as Zoffset parameter)

Explanation

Convert camera coordinate to world coordinate on the plane where Z=0.  
Calibration data of <Camera ID> is used to convert.  
When Camera ID=<Zoffset>, following value is used.

| Image ID        | Used camera ID |
|-----------------|----------------|
| Camera(1~10)    | Image ID       |
| Not camera(10~) | 1              |

If <Undistort> is TRUE, the command first undistorts the image, and it converts undistorted image coordinate to world coordinate. If <Undistort> is FALSE, the image coordinate is directly converted to world coordinate, without correcting distortion.

Related item

CalibrateCamera, SetCamCalDat, GetCamCalDat, SetCamCalExtDat, GetCamCalExtDat, ModifyCamCalExtDat, GetCamPos, GetRobPosFromCam, Undistort2

## GetCamPos

**Format** `object.GetCamPos( <Xw>, <Yw>, <Zw>, <Camera CAL ID> )`

**Parameters**

- <Xw> = VT\_R8: X on world coordinate
- <Yw> = VT\_R8: Y on world coordinate
- <Zw> = VT\_R8: Z on world coordinate
- <Camera CAL ID> = VT\_I4: Camera calibration ID (Default: 0)

**Return value**

- <Xc> = VT\_R8: X on camera coordinate
- <Yc> = VT\_R8: Y on camera coordinate

**Explanation**

Convert world coordinate to camera coordinate  
 Calibration data of <Camera ID> is used to convert.  
 When Camera ID=0, following value is used.

| Image ID        | Used camera ID |
|-----------------|----------------|
| Camera(1~10)    | Image ID       |
| Not camera(10~) | 1              |

**Related item** CalibrateCamera, SetCamCalDat, GetCamCalDat, SetCamCalExtDat, GetCamCalExtDat, ModifyCamCalExtDat, GetPosFromCam, GetCamPosFromRob, Undistort2

## GetPosFromRob

**Format** `object.GetPosFromRob( <Robot CAL ID>, <Xr>, <Yr>, <Zr> )`

**Parameters**

- <Robot CAL ID> = VT\_I4: Robot calibration ID
- <Xr> = VT\_R8: X on robot coordinate
- <Yr> = VT\_R8: Y on robot coordinate
- <Zr> = VT\_R8: Z on robot coordinate

**Return value**

- <Xw> = VT\_R8: X on world coordinate
- <Yw> = VT\_R8: Y on world coordinate
- <Zw> = VT\_R8: Z on world coordinate

**Explanation**

Convert robot coordinate to world coordinate  
 Calibration data of <Robot ID> is used to convert.

**Related item** CalibrateRobot, SetRobCalDat, GetRobCalDat, GetRobPos, GetCamPosFromRob

---

## GetRobPos

---

|              |                                                                                                                                                                                                                                          |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <code>object.GetRobPos( &lt;Robot CAL ID&gt;, &lt;Xw&gt;, &lt;Yw&gt;, &lt;Zw&gt; )</code>                                                                                                                                                |
| Parameters   | <p><b>&lt;Robot CAL ID&gt;</b> = VT_I4: Robot calibration ID</p> <p><b>&lt;Xw&gt;</b> = VT_R8: X on world coordinate</p> <p><b>&lt;Yw&gt;</b> = VT_R8: Y on world coordinate</p> <p><b>&lt;Zw&gt;</b> = VT_R8: Z on world coordinate</p> |
| Return value | <p><b>&lt;Xr&gt;</b> = VT_R8: X on robot coordinate</p> <p><b>&lt;Yr&gt;</b> = VT_R8: Y on robot coordinate</p> <p><b>&lt;Zr&gt;</b> = VT_R8: Z on robot coordinate</p>                                                                  |
| Explanation  | <p>Convert world coordinate to robot coordinate.</p> <p>Calibration data of &lt;Robot ID&gt; is used to convert.</p>                                                                                                                     |
| Related item | CalibrateRobot, SetRobCalDat, GetRobCalDat, GetPosFromRob, GetRobPosFromCam                                                                                                                                                              |

---

## GetRobPosFromCam

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <code>object.GetRobPosFromCam( &lt;Xc&gt;, &lt;Yc&gt; [, &lt;ZOffset&gt;, &lt;Camera CAL ID&gt;, &lt;Robot CAL ID&gt;, &lt;Undistort&gt;])</code>                                                                                                                                                                                                                                                                     |
| Parameters   | <p><b>&lt;Xc&gt;</b> = VT_R8: X on camera coordinate</p> <p><b>&lt;Yc&gt;</b> = VT_R8: Y on camera coordinate</p> <p><b>&lt;ZOffset&gt;</b> = VT_R8: Z on world coordinate</p> <p><b>&lt;Camera CAL ID&gt;</b> = VT_I4: Camera calibration ID (Default: 0)</p> <p><b>&lt;Robot CAL ID&gt;</b> = VT_I4: Robot calibration ID (Default: 1)</p> <p><b>&lt;Undistort&gt;</b> = VT_BOOL: Undistortion (Default: False)</p> |
| Return value | <p><b>&lt;Xw&gt;</b> = VT_R8: X on Robot coordinate</p> <p><b>&lt;Yw&gt;</b> = VT_R8: Y on Robot coordinate</p> <p><b>&lt;Zw&gt;</b> = VT_R8: Z on Robot coordinate (Same value as Zoffset parameter)</p>                                                                                                                                                                                                             |
| Explanation  | <p>Convert camera coordinate to robot coordinate as followings.</p> <p>Camera coordinate → World coordinate → Robot coordinate</p> <p>Where: when converting camera coordinate to world coordinate, the point is on a Z=&lt;ZOffset&gt; plane.</p>                                                                                                                                                                    |

Calibration data specified by <CameraID> and <RobotID> are used for the conversion.  
When Camera ID=0, following value is used.

| Image ID        | Used camera ID |
|-----------------|----------------|
| Camera(1~10)    | Image ID       |
| Not camera(10~) | 1              |

Set <Undistort> to TRUE for a distorted image.

**Related item** CalibrateCamera, CalibrateRobot, SetCamCalDat, GetCamCalDat, SetCamCalExtDat, GetCamCalExtDat, ModifyCamCalExtDat, SetRobCalDat, GetRobCalDat, GetPosFromCam, GetCamPos, GetPosFromRob, GetRobPos, GetCamPosFromRob, Undistort2

---

## GetCamPosFromRob

---

**Format** `object.GetCamPosFromRob( <Xw>, <Yw>, <Zw>[, <Camera CAL ID>, <Robot CAL ID>] )`

**Parameters** <Xw> = VT\_R8: X on Robot coordinate  
 <Yw> = VT\_R8: Y on Robot coordinate  
 <Zw> = VT\_R8: Z on Robot coordinate  
 <Camera CAL ID> = VT\_I4: Camera calibration ID (Default: 0)  
 <Robot CAL ID> = VT\_I4: Robot calibration ID(Default:1)

**Return value** <Xc> = VT\_R8: X on camera coordinate  
 <Yc> = VT\_R8: Y on camera coordinate

**Explanation** Convert robot coordinate to camera coordinate as followings.  
 Robot coordinate → World coordinate → Camera coordinate  
 Calibration data specified by <CameraID> and <RobotID> are used for the conversion.  
 When Camera ID=0, following value is used.

| Image ID        | Used camera ID |
|-----------------|----------------|
| Camera(1~10)    | Image ID       |
| Not camera(10~) | 1              |

**Related item** CalibrateCamera, CalibrateRobot, SetCamCalDat, GetCamCalDat, SetCamCalExtDat, GetCamCalExtDat, ModifyCamCalExtDat, SetRobCalDat, GetRobCalDat, GetPosFromCam, GetCamPos, GetPosFromRob, GetRobPos, GetRobPosFromCam, Undistort2

## Undistort2

Format `object.Undistort2 <Output ID>, <Camera CAL ID>`

Parameters `<Output ID>` = VT\_I4: Output Image ID  
`<Camera CAL ID>` = VT\_I4: Camera calibration ID (Default: 0)

Return value `<Image>` = VT\_UI1|VT\_ARRAY: Changed image

Explanation Adjust distortion.  
 If Output Image ID=0, return value is changed image data. If Output Image Id <> 0, the change image is stored in the specified ID image memory, and return value is Empty.  
 Parameters for the specifed camera ID are used for distortion adjustment.  
 When Camera ID=0, following value is used.

| Image ID        | Used camera ID |
|-----------------|----------------|
| Camera(1~10)    | Image ID       |
| Not camera(10~) | 1              |

The changed image data is output by the bitmap file format of the Windows standard. Color output format is 24bit color bitmap image, and grayscale output format is 8bit bitmap image.

[Note] The specification of the auguments and return value were changed from 1.3.2.

Related item CalibrateCamera, SetCamCalDat, GetCamCalDat

### 4.2.12. Misc.

## GoodFeaturesToTrackEx

Format `object.GoodFeaturesToTrackEx( <Max count>, <Quality>, <Distance>, <Block size> )`

Parameters `<Max count>` = VT\_I4: Maximum number of corner detection  
`<Quality>` = VT\_R8: Quality  
`<Distance>` = VT\_I4: Minimum distance  
`<Block size>` = VT\_I4: Size of the averaged block

Return value `<Points>` = VT\_VARIANT|VT\_ARRAY: Detected corners coordinates list  
 (`<Point1>`, `<Point2>`, ...)



**<Pointn>** = VT\_I4|VT\_ARRAY: Coordinates list (<X>, <Y>)

**<X>** = VT\_I4: X coordinates

**<Y>** = VT\_I4: Y coordinates

**Explanation** The command finds corners with big eigenvalues in the image.

**Related item** FindCornerSubPixEx

---

## FindCornerSubPixEx

---

**Format** *object.FindCornerSubPixEx*( <Points>, <Win X>, <Win Y>, <Zero X>, <Zero Y>, <Term type>, <Max iteration>, <Epsilon> )

**Parameters** **<Points>** = VT\_VARIANT|VT\_ARRAY: Detected corners coordinates list

**<Win X>** = VT\_I4: X direction half sizes of the search window

**<Win Y>** = VT\_I4: Y direction half sizes of the search window

**<Zero X>** = VT\_I4:

X direction half size of the dead region in the middle of the search zone

**<Zero Y>** = VT\_I4:

Y direction half size of the dead region in the middle of the search zone

**<Term type>** = VT\_I4: Repetition end condition type

|   |                        |
|---|------------------------|
| 1 | Periodic duty          |
| 2 | Precision attained use |

**<Max iteration>** = VT\_I4: Maximum number of occurrences

**<Epsilon>** = VT\_R8: Precision attained

**Return value** **<Points>** = VT\_VARIANT|VT\_ARRAY: Detected corners coordinates list  
(<Point1>, <Point2>, ...)

**<Pointn>** = VT\_I4|VT\_ARRAY: Coordinates list (<X>, <Y>)

**<X>** = VT\_I4: X coordinates

**<Y>** = VT\_I4: Y coordinates

**Explanation** Refine corner detection result.

The result of command DetermineCorners is used for Points of the parameter.

The size of the retrieval area and the exclusion area should specify half the size of the target.

**Related item** GoodFeaturesToTrackEx

# MomentsEx

Format `object.MomentsEx( <Contour ID> )`

Parameters <Contour ID> = VT\_I4: Contour ID

|      |                       |
|------|-----------------------|
| -1   | The entire screen     |
| <>-1 | The specified contour |

Return value <Moments> = VT\_VARIANT|VT\_ARRAY: Moment  
(<Spatial Moments>, <Central Moments>, <inv\_sqrt\_m00>)

<Spatial moments> = VT\_R8|VT\_ARRAY: Spatial moments

<M00> = VT\_R8:

<M10> = VT\_R8:

<M01> = VT\_R8:

<M20> = VT\_R8:

<M11> = VT\_R8:

<M02> = VT\_R8:

<M30> = VT\_R8:

<M21> = VT\_R8:

<M12> = VT\_R8:

<M03> = VT\_R8:

<Central moments> = VT\_R8|VT\_ARRAY: Central moments

<M20> = VT\_R8:

<M11> = VT\_R8:

<M02> = VT\_R8:

<M30> = VT\_R8:

<M21> = VT\_R8:

<M12> = VT\_R8:

<M03> = VT\_R8:

<inv\_sqrt\_m00> = VT\_R8:1/sqrt(M00)

Explanation Calculate moment.

Execute command FindContoursEx beforehand when you specify Contour ID.

Color image is automatically converted to grayscale image.

Error 0x80101001 : Contour is not found. Please execute "FindContoursEx" command.

About the another error, please refer to the capter 2.4.

## MeasureInfo

|              |                                                                                                                                                                                                                                                                                  |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object.MeasureInfo</i> ( <b>&lt;Moments&gt;</b> )                                                                                                                                                                                                                             |
| Parameters   | <b>&lt;Moments&gt;</b> = VT_R8 VT_ARRAY: Moment<br><b>&lt;M00&gt;</b> = VT_R8:<br><b>&lt;M10&gt;</b> = VT_R8:<br><b>&lt;M01&gt;</b> = VT_R8:<br><b>&lt;M20&gt;</b> = VT_R8:<br><b>&lt;M11&gt;</b> = VT_R8:<br><b>&lt;M02&gt;</b> = VT_R8:                                        |
| Return value | <b>&lt;Area&gt;</b> = VT_R8: Area<br><b>&lt;Center of gravity X&gt;</b> = VT_R8: Center of gravity point X coordinates<br><b>&lt;Center of gravity Y&gt;</b> = VT_R8: Center of gravity point Y coordinates<br><b>&lt;Principal axis angle&gt;</b> = VT_R8: Principal axis angle |
| Explanation  | Calculate area size, gravity center, and principal axis angle.<br>Specify the execution result of MomentsEx command for Moments argument.<br>Please refer to the descriptions of MeasureInfo on OpenCV reference for details of the mode and the method.                         |

## HoughLines

Format

*object.HoughLines*( **<Method>**, **<Rho>**, **<Theta>**, **<Threshold>**, **<Para1>**, **<Para2>** )

Parameters

**<Method>** = VT\_I4: The Hough transform variant

|   |                        |                                                                                                                                                                                                                                                         |
|---|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | CV_HOUGH_STANDARD      | classical or standard Hough transform. Every line is represented by two floating-point numbers ( $\rho$ , $\theta$ ), where $\rho$ is a distance between (0,0) point and the line, and $\theta$ is the angle between x-axis and the normal to the line. |
| 1 | CV_HOUGH_PROBABILISTIC | probabilistic Hough transform (more efficient in case if picture contains a few long linear segments). It returns line segments rather than the whole lines. Every                                                                                      |

|   |                      |                                                                                                               |
|---|----------------------|---------------------------------------------------------------------------------------------------------------|
|   |                      | segment is represented by starting and ending points.                                                         |
| 2 | CV_HOUGH_MULTI_SCALE | multi-scale variant of classical Hough transform. The lines are encoded the same way as in CV_HOUGH_STANDARD. |

$$\langle \text{Rho} \rangle = \text{VT\_R8: } \rho(\text{Rho})$$
 $\langle \text{Theta} \rangle = V_T \quad R8: \theta (\text{Theta})$ 

**<Threshold>** = VT I4: Threshold

**<Para1>** = VT R8: Parameter1

**<Para2>** = VT R8: Parameter2

[illegible]

**<Linen>** = VT\_I4|ARRAY : Straight line list  
(**<StartX>**, **<StartY>**, **<End X>**, **<End Y>**)

**<StartX>** = VT I4: Start X-Coordinates

**<StartY>** = VT I4: Start Y-Coordinates

**<End X>** = VT\_I4: End X-Coordinates

**<End Y>** = VT I4: End Y-Coordinates

|                    |                                   |
|--------------------|-----------------------------------|
| <b>Explanation</b> | Find lines using Hough transform. |
|--------------------|-----------------------------------|

Related item    [HoughCircles](#)

## HoughCircles

|        |                                                                                                                                                                 |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format | <code>object.HoughCircles( &lt;dp&gt;, &lt;Min distance&gt;, &lt;Canny threshold&gt;, &lt;Center threshold&gt;, &lt;Min radius&gt;, &lt;Max radius&gt; )</code> |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|

|            |                                                                    |
|------------|--------------------------------------------------------------------|
| Parameters | <dp> = VT_R8: Calculation resolution                               |
|            | <Min distance> = VT_R8: Minimum distance between center coordinate |
|            | <Canny threshold> = VT_R8: Higher threshold used in Canny          |
|            | <Center threshold> = VT_R8: Center detection calculation threshold |
|            | <MinRadius> = VT_I4: Minimum radius                                |
|            | <MaxRadius> = VT_I4: Maximum radius                                |

**Return value**      <**Circles**> = VT\_VARIANT|VT\_ARRAY : Detected circle list  
                                                                                                          (<**Circle1Circle2
                                                                                                          <**Circlen**> = VT\_R4|VT\_ARRAY: Circle (<**CenterXCenterYRadius****

<CenterX> = VT\_R4: Center X coordinates

<CenterY> = VT\_R4: Center Y coordinates

<Radius> = VT\_R4: Radius

**Explanation** Find circles using Hough transform.

**Related item** HoughLines

---

## DFTE<sub>x</sub>

---

**Format** *object*. DFTE<sub>x</sub> <Output ID>, <Output ID(R)>, <Output ID(I)>

**Parameters** <Output ID> = VT\_I4: Output image ID

<Output ID(R)> = VT\_I4: Output real part image ID

<Output ID(I)> = VT\_I4: Output imaginary part image ID

**Return value** <Image> = VT\_UI1|VT\_ARRAY: Converted image

**Explanation** Perform DFT(Discrete Fourier Transform).

If <Output ID> is 0, then the transformed image is returned. If <Output ID> is not 0, then the transformed image is stored in the specified image memory area, and VT\_EMPTY is returned.

The database area can not be used as <Real image ID> or <Imaginary Image ID>. Use memory area instead.

A color-scale image is converted to the gray-scale image automatically.

The changed image data is output by the 8-bit bitmap file format of the Windows standard.

**Related item** IDFT

---

## IDFT

---

**Format** *object*. IDFT <Output ID>, <Input ID(R)>, <Input ID(I)>

**Parameters** <Output ID> = VT\_I4: Output image ID

<Input ID(R)> = VT\_I4: Input real part image ID

<Input ID(I)> = VT\_I4: Input imaginary part image ID

**Return value** <Image> = VT\_UI1|VT\_ARRAY: Converted image

|              |                                                                                                                                                                                                                                                                                                                                                                      |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Explanation  | <p>Perform Inverse DFT(Discrete Fourier Transform).</p> <p>If &lt;Output ID&gt; is 0, then the transformed image is returned. If &lt;Output ID&gt; is not 0, then the transformed image is stored in the specified image memory area, and VT_EMPTY is returned.</p> <p>The changed image data is output by the 8-bit bitmap file format of the Windows standard.</p> |
| Related item | DFTE <sub>x</sub>                                                                                                                                                                                                                                                                                                                                                    |

---

## OpticalFlowEx

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <i>object</i> .OpticalFlowEx( <Input ID>, <X size>, <Y size> )                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Parameters   | <p>&lt;Input ID&gt; = VT_I4: Comparison image ID</p> <p>&lt;X size&gt; = VT_I4: Unit of X axis measurement</p> <p>&lt;Y size&gt; = VT_I4: Unit of Y axis measurement</p>                                                                                                                                                                                                                                                                                                                                                                                         |
| Return value | <p>&lt;Points&gt; = VT_VARIANT VT_ARRAY : Point result list</p> <p style="padding-left: 40px;">(&lt;Point1&gt;, &lt;Point2&gt;, ...)</p> <p style="padding-left: 40px;">&lt;Pointn&gt; = VT_I4 VT_ARRAY: Position and Variation (&lt;X&gt;, &lt;Y&gt;, &lt;dX&gt;, &lt;dY&gt;)</p> <p style="padding-left: 80px;">&lt;X&gt; = VT_I4: X-Coordinates</p> <p style="padding-left: 80px;">&lt;Y&gt; = VT_I4: Y-Coordinates</p> <p style="padding-left: 80px;">&lt;dX&gt; = VT_I4: X variation</p> <p style="padding-left: 80px;">&lt;dY&gt; = VT_I4: Y variation</p> |
| Explanation  | <p>Perform OpticalFlowEx process between the current image and &lt;Input ID&gt; image, and return the point list.</p> <p>For details, please refer to CalcOpticalFlowLK of OpenCV reference.</p>                                                                                                                                                                                                                                                                                                                                                                 |

---

## OpticalFlowPyrEx

---

|            |                                                                                                                                                                                                                                                                                                                                                                                              |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format     | <i>object</i> .OpticalFlowPyrEx( <Input ID>, <Points>, <Win X>, <Win Y>, <Level> )                                                                                                                                                                                                                                                                                                           |
| Parameters | <p>&lt;Input ID&gt; = VT_I4: comparison image number</p> <p>&lt;Points&gt; = VT_VARIANT VT_ARRAY: search point list</p> <p style="padding-left: 40px;">(&lt;Point1&gt;, &lt;Point2&gt;, ...)</p> <p style="padding-left: 40px;">&lt;World Point&gt; = VT_R8  VT_ARRAY : search point coordinate(&lt;X&gt;, &lt;Y&gt;)</p> <p style="padding-left: 80px;">&lt;X&gt; = VT_R8: X coordinate</p> |

<Y> = VT\_R8: Y coordinate

<Win X> = VT\_I4: search window size (X)

<Win Y> = VT\_I4: search window size (Y)

<Level> = VT\_I4: Pyramid level maximum value

|    |                                                         |
|----|---------------------------------------------------------|
| 0  | Pyramid not used (single level). Level is set to 2      |
| 1  | Set pyramid level to 2                                  |
| >2 | Specified value is set to the maximum level of pyramid. |

**Return value** <Points> = VT\_VARIANT|VT\_ARRAY: point list (<Point1>, <Point2>, ... )  
 <Pointn> = VT\_I4|VT\_ARRAY: point after move (<X>, <Y>)  
 <X> = VT\_I4: X coordinate  
 <Y> = VT\_I4: Ycoordinate

**Explanation** Calculates optical flow for a sparse feature set using iterative Lucas-Kanade method in pyramids.  
 The function CalcOpticalFlowPyrLK calculates the optical flow between two images for the given set of points in <Input ID> image. The function finds the flow with sub-pixel accuracy.  
 For details, please refer to cvCalcOpticalFlowPyrLK of OpenCV reference.

---

## BoxPoints

---

**Format** *object.BoxPoints( <X>, <Y>, <W>, <H>, <Angle> )*

**Parameters** <X> = VT\_I4: Center X coordinates  
 <Y> = VT\_I4: Center Y coordinates  
 <W> = VT\_I4: Width  
 <H> = VT\_I4: Height  
 <Angle> = VT\_I4: Rotating angle

**Return value** <Points> = VT\_VARIANT|VT\_ARRAY: Four corner point list  
 (<Point1>, <Point2>, <Point3>, <Point4>)  
 <Pointn> = VT\_I4|VT\_ARRAY: Position (<X>, <Y>)  
 <X> = VT\_I4: X-Coordinates  
 <Y> = VT\_I4: Y-Coordinates

**Explanation** Calculate the four corner point coordinate of the specified rectangular.  
 [Note] From Version 1.3.5, rotation dillection is changed to crockwise..

# FindHomography

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Format       | <code>object.FindHomography( &lt;Point1&gt;, &lt;Point2&gt;, &lt;Point3&gt; )</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Parameters   | <p><b>&lt;Points&gt;</b> = VT_VARIANT VT_ARRAY:<br/>Projection transformation corresponding point list (<b>&lt;Point1&gt;</b>, <b>&lt;Point2&gt;</b>, ...)</p> <p><b>&lt;Pointn&gt;</b> = VT_VARIANT  VT_ARRAY:<br/>Corresponding points before and after the projection transformation<br/>(<b>&lt;Before Point&gt;</b>, <b>&lt;After Point&gt;</b>)</p> <p><b>&lt;Before Point&gt;</b> = VT_R8  VT_ARRAY : Point before transformation (<b>&lt;X&gt;</b>, <b>&lt;Y&gt;</b>)<br/> <b>&lt;X&gt;</b> = VT_R8: X coordinate<br/> <b>&lt;Y&gt;</b> = VT_R8: Y coordinate</p> <p><b>&lt;After Point&gt;</b> = VT_R8  VT_ARRAY : Point after transformation (<b>&lt;X&gt;</b>, <b>&lt;Y&gt;</b>)<br/> <b>&lt;X&gt;</b> = VT_R8: X coordinate<br/> <b>&lt;Y&gt;</b> = VT_R8: Y coordinate</p> |
| Return value | <p><b>&lt;Matrix&gt;</b> = VT_R8 VT_ARRAY : Homography matrix<br/> (<b>&lt;r11&gt;</b>, <b>&lt;r12&gt;</b>, <b>&lt;r13&gt;</b>, <b>&lt;r21&gt;</b>, <b>&lt;r22&gt;</b>, <b>&lt;r23&gt;</b>, <b>&lt;r31&gt;</b>, <b>&lt;r32&gt;</b>, <b>&lt;r33&gt;</b>)</p> <p><b>&lt;r11&gt;</b> = VT_R8:<br/> <b>&lt;r12&gt;</b> = VT_R8:<br/> <b>&lt;r13&gt;</b> = VT_R8:<br/> <b>&lt;r21&gt;</b> = VT_R8:<br/> <b>&lt;r22&gt;</b> = VT_R8:<br/> <b>&lt;r23&gt;</b> = VT_R8:<br/> <b>&lt;r31&gt;</b> = VT_R8:<br/> <b>&lt;r32&gt;</b> = VT_R8:<br/> <b>&lt;r33&gt;</b> = VT_R8:</p> $\begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$                                                                                                 |
| Explanation  | <p>Calculate projection transformation matrix.</p> <p>Specify arbitrary numbers of corresponding points for projection transformation, and calculate projection transformation matrix.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Related item | WarpPerspective                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |



## QRDecode

Format *object. QRDecode ( <Code> )*

Parameters <Code> = Code type

|   |                 |                                                                                                                     |
|---|-----------------|---------------------------------------------------------------------------------------------------------------------|
| 0 | CODE_QR         | Decode QR code Model 1 or Model 2                                                                                   |
| 1 | CODE_MICROQR    | Decode Micro QR                                                                                                     |
| 2 | CODE_DATAMATRIX | Decode DataMatrix                                                                                                   |
| 3 | CODE_PDF417     | Decode PDF417                                                                                                       |
| 4 | CODE_BARCODE    | Decode barcode ()<br>(UPC/EAN , CODE39 , CODABAR(NW-7) ,<br>Interleaved 2 of 5(ITF), CODE128, EAN-128,<br>RSS)をデコード |
| 5 | CODE_MICROPDF   | Decode MicroPDF417                                                                                                  |
| 6 | CODE_COMPOSITE  | Decode EAN.UCC Composite                                                                                            |

Return value <Data> = VT\_BSTR: Decoded data

<Decode info> = VT\_VARIANT|VT\_ARRAY: Information about decoding

Explanation Decode several types of two dimensional code such as QRCode.

Please see the <Code> and <Decode info> information in the QRdecoder manual for the details.

Error 0x80101001 : QR Decoder is not initialized. Please use "Qrabled=True" Option in AddController..

About the another error, please refer to the capter 2.4.

## OCRead

Format *object. OCRead ()*

Parameters None

Return value <Data> = VT\_BSTR: recognized string

Explanation Recognize a character string from image. But only an alphanumeric character string is recognizable.

The image should be converted to binary image to improve recognition.

Error 0x80101001 : OCR is not initialized. Please use "OCREnabled=True " Option in AddController..

About the another error, please refer to the capter 2.4.

## 4.3. Command class

### 4.3.1. Triangulation

# Triangulation

Format *object.Triangulation( <Camera1 CAL ID>, <Camera2 CAL ID>, <Camera3 CAL ID>, <Xc1>, <Yc1>, <Xc2>, <Yc2>, <Xc3>, <Yc3>, <Tru-Method> )*

Parameters

- <Camera1 CAL ID> = VT\_I4: Camera 1 calibration number
- <Camera2 CAL ID> = VT\_I4: Camera 2 calibration number
- <Camera3 CAL ID> = VT\_I4: Camera 3 calibration number(0: Not used)
- <Xc1> = VT\_R8: Camera 1 X coordinate
- <Yc1> = VT\_R8: Camera 1 Y coordinate
- <Xc2> = VT\_R8: Camera 2 X coordinate
- <Yc2> = VT\_R8: Camera 2 Y coordinate
- <Xc3> = VT\_R8: Camera 3 X coordinate
- <Yc3> = VT\_R8: Camera 3 Y coordinate
- <Tru-Method> = VT\_I4: Triangulation method

|   |          |                         |
|---|----------|-------------------------|
| 0 | Liner    | Singular value analysis |
| 1 | Midpoint | Center point analysis   |

Return value

- <X> = VT\_R8: X coordinates
- <Y> = VT\_R8: Y coordinates
- <Z> = VT\_R8: Z coordinates

Explanation

Perform triangulation using two or three cameras.

Camera calibration and camera position setup need to be performed before executing this command.

If <Camera3 ID> is 0, then it calculates from two camera data.

Related item

CalibrateCamera, SetCamCalDat, GetPosFromCam

# TriMatchTemplate

Format *object.TriMatchTemplate( <Camera1 ID>, <Camera2 ID>, <Camera3 ID>,*

<Input ID>, <Method>, <Threshold>, <Start angle>, <End angle>, <Step angle>, <Down sizing>, <Undistorted>, <Tru-Method> )

## Parameters

<Camera1 ID> = VT\_I4: Camera 1 image number

<Camera2 ID> = VT\_I4: Camera 2 image number

<Camera3 ID> = VT\_I4: Camera 3 image number(0: Not used)

<Input ID> = VT\_I4: Template image number

<Method> = VT\_I4: Matching method

|                                                                                                                                     |                     |                                                                                                                                                                                                                                                                  |
|-------------------------------------------------------------------------------------------------------------------------------------|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (I denotes image, T - template, R - result. The summation is done over template and/or the image patch: $x'=0..w-1$ , $y'=0..h-1$ ) |                     |                                                                                                                                                                                                                                                                  |
| 0                                                                                                                                   | CV_TM_SQDIFF        | $R(x, y) = \sum_{x', y'} [T(x', y') - I(x + x', y + y')]^2$                                                                                                                                                                                                      |
| 1                                                                                                                                   | CV_TM_SQDIFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') - I(x + x', y + y')]^2}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$                                                                                                                     |
| 2                                                                                                                                   | CV_TM_CCORR         | $R(x, y) = \sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]$                                                                                                                                                                                                    |
| 3                                                                                                                                   | CV_TM_CCORR_NORMED  | $R(x, y) = \frac{\sum_{x', y'} [T(x', y') \cdot I(x + x', y + y')]}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$                                                                                                                   |
| 4                                                                                                                                   | CV_TM_CCOEFF        | $R(x, y) = \sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]$ <p>where</p> $T'(x', y') = T(x', y') - \frac{\sum_{x'', y''} T(x + x'', y + y'')}{(w \cdot h)}$ $I'(x + x', y + y') = I(x + x', y + y') - \frac{\sum_{x'', y''} I(x + x'', y + y'')}{w \cdot h}$ |
| 5                                                                                                                                   | CV_TM_CCOEFF_NORMED | $R(x, y) = \frac{\sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}}$                                                                                                               |

<Threshold> = VT\_R8: Threshold

<Start angle> = VT\_I4: Search start angle (degree)

<End angle> = VT\_I4: Search end angle(degree)

<Step angle> = VT\_I4: Step angle degree

<Down sizing> = VT\_I4: Down sizing count

<Undistorted> = VT\_BOOL: Distortion compensation flag

|       |                                 |
|-------|---------------------------------|
| True  | Enable distortion compensation  |
| False | Disable distortion compensation |

<Tru-Method> = VT\_I4: Triangulation method

|   |          |                         |
|---|----------|-------------------------|
| 0 | Liner    | Singular value analysis |
| 1 | Midpoint | Center point analysis   |

Return value <X> = VT\_R8: X coordinates

<Y> = VT\_R8: Y coordinates

<Z> = VT\_R8: Z coordinates

**Explanation** Perform template matching for two or three camera images, and use the detected position for triangulation.

Camera calibration and camera position setup need to be performed before executing this command.

If <Camera3 ID> is 0, then it calculates from two camera data.

[Note] From Version 1.3.5, rotation dillection is changed to crockwise.

**Related item** CalibrateCamera, SetCamCalDat, GetPosFromCam, MatchTemplate2

**Example** [VB6]

```
Set caoCommand = caoController.AddCommand("TriMatchTemplate")
caoCommand.Parameters= Array(1, 2, 0, 11, 3, 0.8, 0, 0, 3, False)
' Detect image in #11 from camera image #1 and #2. and perform triangulation.
' Camera image rotation detect: None(0° ~0° ), Search depth: 3,
' Distortion adjustment: none
caoCommand.Execute 0
vntRet = caoCommand.Result
x = vntRet(0) ' <X>
y = vntRet(1) ' <Y>
angle = vntRet(2) ' <Angle>
```

## TriMatchShapes

**Format** *object*.TriMatchShapes( <Camera1 ID>, <Camera2 ID>, <Camera3 ID>, <Input ID>, <Threshold>, <Type>, <Method>, <Min scale>, <Similarity>, <Undistorted>, <Tru-Method> )

## Parameters

&lt;Camera1 ID&gt; = Camera 1 image number

&lt;Camera2 ID&gt; = VT\_I4: Camera 2 image number

&lt;Camera3 ID&gt; = VT\_I4: Camera 3 image number(0: Not used)

&lt;Input ID&gt; = VT\_I4: Template image number

&lt;Threshold&gt; = VT\_I4: Threshold

&lt;Type&gt; = VT\_I4: Threshold type

|   |                      |
|---|----------------------|
| 0 | CV_THRESH_BINARY     |
| 1 | CV_THRESH_BINARY_INV |

&lt;Method&gt; = VT\_I4: Matching method

A means original image and B means template image in the table below.

|   |                     |                                                                             |
|---|---------------------|-----------------------------------------------------------------------------|
| 0 | CV_CONTOUR_MATCH_I1 | $I_1(A, B) = \sum_{i=1}^7 \left  \frac{1}{m_i^A} - \frac{1}{m_i^B} \right $ |
| 1 | CV_CONTOUR_MATCH_I2 | $I_2(A, B) = \sum_{i=1}^7  m_i^A - m_i^B $                                  |
| 2 | CV_CONTOUR_MATCH_I3 | $I_3(A, B) = \sum_{i=1}^7 \frac{ m_i^A - m_i^B }{ m_i^A }$                  |

where

$$m_i^A = \sin(h_i^A) \cdot \log(h_i^A)$$

$$m_i^B = \sin(h_i^B) \cdot \log(h_i^B)$$

 $h_i^A$ ,  $h_i^B$  are Hu moments of A and B, respectively.

&lt;Min scale&gt; = VT\_R8: Minimum scale

&lt;Similarity&gt; = VT\_R8: Contour similarity

&lt;Undistorted&gt; = VT\_BOOL: Distortion compensation flag

|       |                                 |
|-------|---------------------------------|
| True  | Enable distortion compensation  |
| False | Disable distortion compensation |

&lt;Tri-Method&gt; = VT\_I4: Triangulation method

|   |          |                         |
|---|----------|-------------------------|
| 0 | Liner    | Singular value analysis |
| 1 | Midpoint | Center point analysis   |

## Return value

&lt;X&gt; = VT\_R8: X coordinate

&lt;Y&gt; = VT\_R8: Y coordinate

<Z> = VT\_R8: Z coordinate

**Explanation** Compare object shape using two or three cameras, and perform triangulation using the detected coordinate.

Each camera and its position need to be calibrated before using this function.

Camera image is converted to binary image using Canny filter. Therefore, specify images converted by Canny for <Input ID> image.

If two or more cantors are extracted from <Input ID> image, matching may fail. Therefore, <Input ID> image should be selected so that return value of the FindContours for the image is 1.

If <Camera3 ID> is 0, then it calculates from two camera data.

**Related item** CalibrateCamera, SetCamCalDat, GetPosFromCam, MatchShapes2

---

## TriHaarDetect

---

**Format** *object.TriHaarDetect*( <Camera1 ID>, <Camera2 ID>, <Camera3 ID>, <XML Path>, <Scale>, <Min Neighbors>, <Undistorted>, <Tru-Method> )

**Parameters**

<Camera1 ID> = VT\_I4: Camera 1 image number  
 <Camera2 ID> = VT\_I4: Camera 2 image number  
 <Camera3 ID> = VT\_I4: Camera 3 image number(0: Not used)  
 <Path> = VT\_BSTR: Path to Haar file  
 <Scale> = VT\_R8: Scale  
 <MinNeighbors> = VT\_I4: Minimum neighbor number  
 <Undistorted> = VT\_BOOL: Distortion adjustment flag

|       |                                 |
|-------|---------------------------------|
| True  | Enable distortion compensation  |
| False | Disable distortion compensation |

<Tru-Method> = VT\_I4: Triangulation method

|   |          |                         |
|---|----------|-------------------------|
| 0 | Liner    | Singular value analysis |
| 1 | Midpoint | Center point analysis   |

**Return value**

<X> = VT\_R8: X coordinates  
 <Y> = VT\_R8: Y coordinates  
 <Z> = VT\_R8: Z coordinates

**Explanation** Perform Haar matching for two or three camera image, and use the detected position for triangulation.

Camera calibration and camera position setup need to be performed before executing this command.

If <Camera3 ID> is 0, then it calculates from two camera data.

**Related item**     CalibrateCamera, SetCamCalDat, GetPosFromCam, HaarDetect

## 5. OcvTester

### 5.1. Outline

OcvTester is an application using OpenCV provider to perform image-processing process interactively.

With OcvTester, you can perform image processing step by step, and show the processed images on different windows. In addition, OcvTester can record and output the performed image processing procedures in CaoScript scripting language.

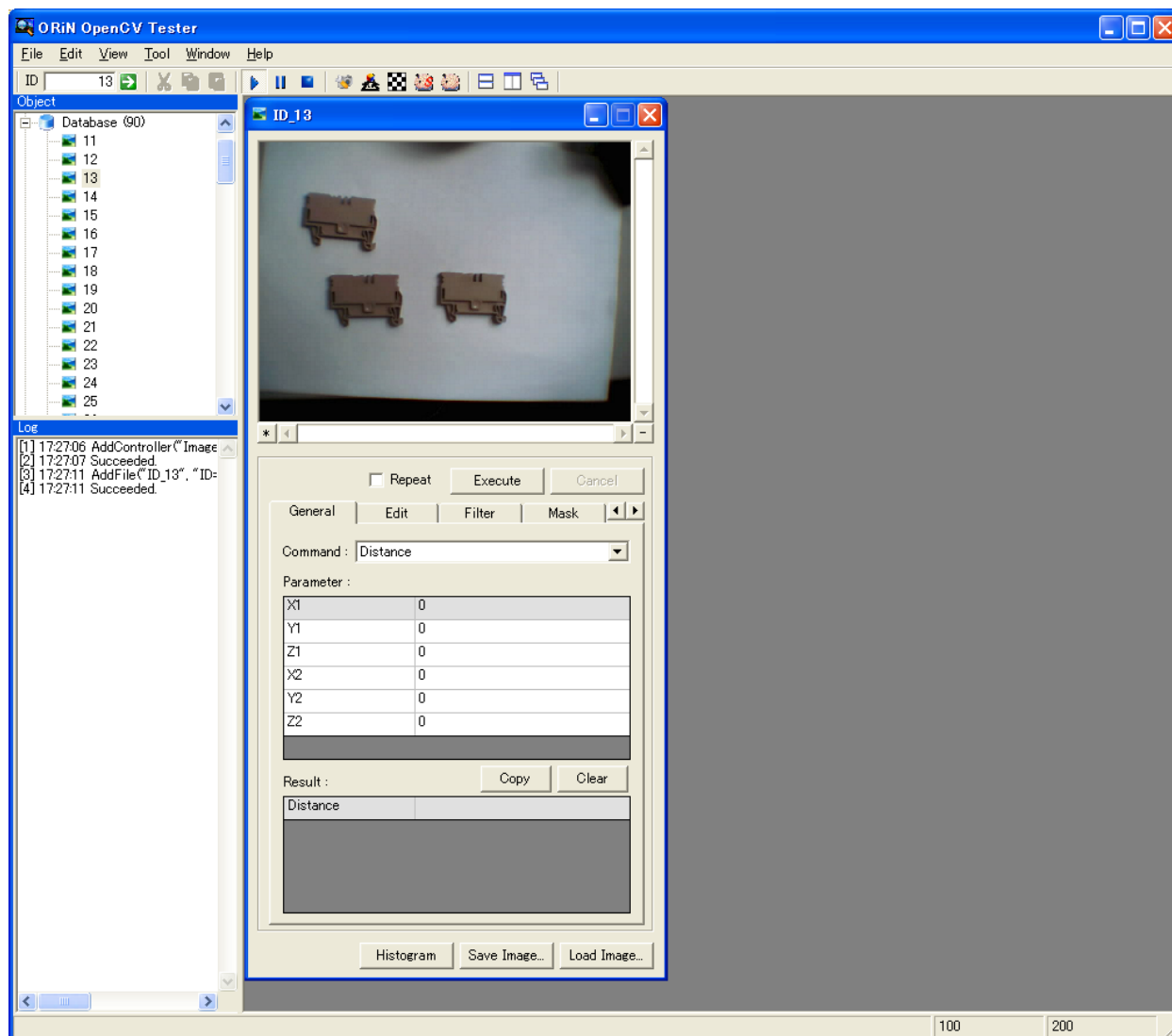


Figure 5-1 OcvTester screen image

Double-clicking the tree view on the left above of the main window displays image window corresponding to the selected node. Image window is used for command execution, and command execution log is displayed in the log window at the left below of the main window.



## 5.2. Main screen

Main screen is for operations like setup OpenCV provider, manage each windows and file input/output.

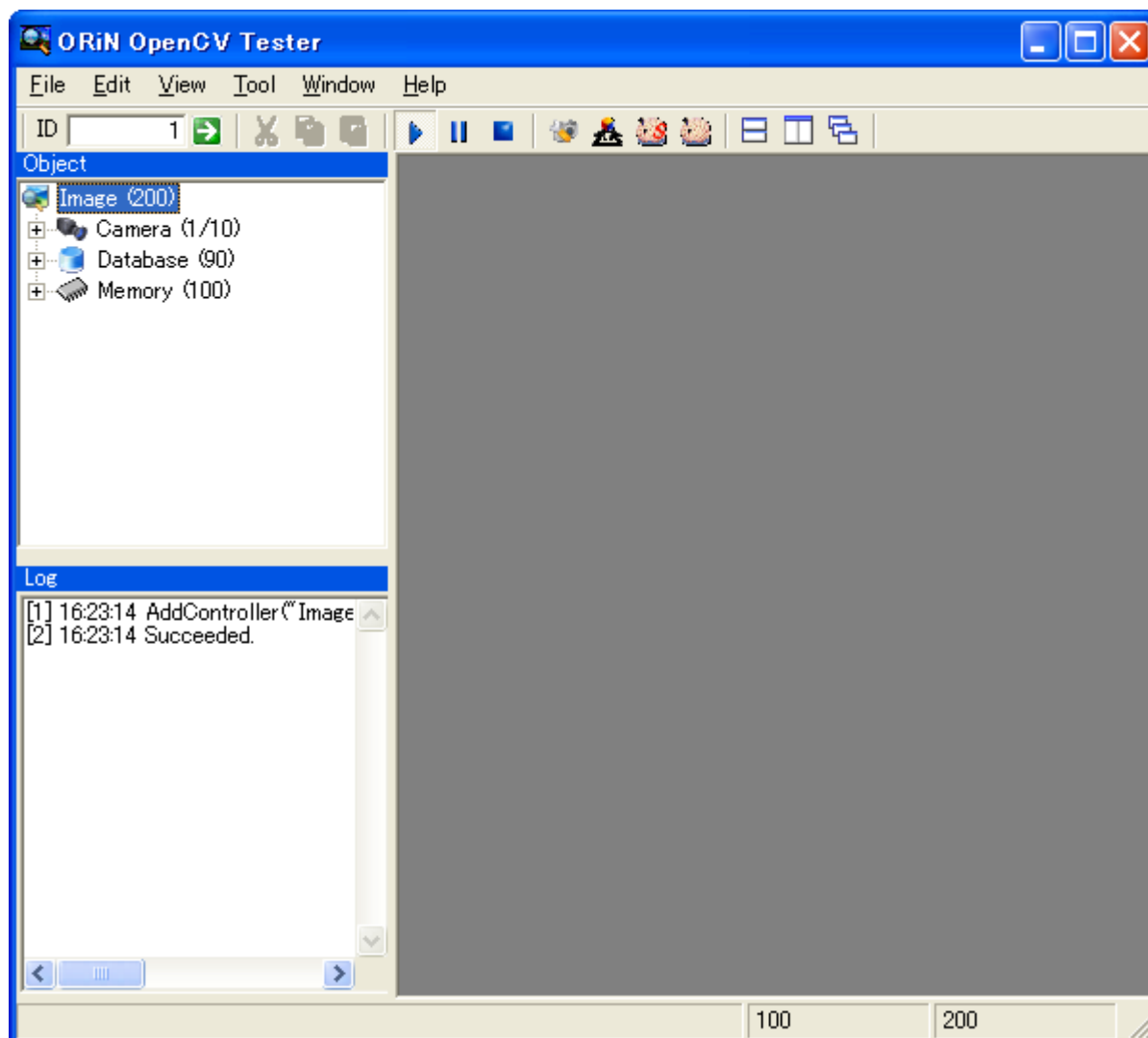


Figure 5-2 OcvTester screen

### 5.2.1. Object window

Object window is for camera and memory image management.

By double-clicking the object, image window is displayed.

### 5.2.2. Log window

Command execution log is output on this window.

### 5.2.3. Menu

#### 5.2.3.1. File menu

This menu is for saving files.

**[Export Log]**

Output log to a file.

**[Export Script]**

Output recorded command execution procedure as CaoScript file.

**[Connect]**

Connect to DENSO robot. When connecting, DENSO robot connection window is displayed. For details, please refer to 5.4.

**[Disconnect]**

Disconnect communication with DENSO robot.

**[Exit]**

Exit OcvTester.

#### 5.2.3.2. Edit menu

This menu is for editing log output and script record.

**[Output Log]**

Output command execution log to log window.

**[Clear Log]**

Clear log window contents.

**[Copy]**

Copy text to clipboard.

**[Cut]**

Cut text and copy to clipboard.

**[Paste]**

Paste contents of clipboard to text.

#### 5.2.3.3. Display menu

This menu is for screen display setting.

**[Object Window]**

Display and hide object window.

**[Log Window]**

Display and hide log window.

**[Script Viewer]**

Display CAO Script Viewer.

**[Main Screen]**

Display and hide main screen.

#### 5.2.3.4. Tool Menu

This menu is for tool menu display.

##### **[Camera Setting]**

Display camera setting tool. For details, please refer to 5.5

##### **[Triangulation]**

Displays triangulation window. For details, please refer to 5.6

##### **[Calibration]**

Display calibration wizard. For details, please refer to 5.6.

##### **[Image Samples]**

Displays image sampling creation tool. For details, please refer to 5.9

##### **[Haar Training]**

Displays Haar training tool. For details, please refer to 5.10

#### 5.2.3.5. Script menu

This menu is for CAO script automatic generation function setting.

##### **[Start Recording]**

Start command execution recording.

##### **[Pause Recording]**

Pause command execution recording. When the recording is resumed, the command record is added to the paused command record.

##### **[Stop Recording]**

Stop command execution recording. When the recording is started next time, previous command record is destructed.

##### **[Copy Script]**

Copy currently recorded CAO script to clipboard.

##### **[Clear Script]**

Clear currently recorded CAO script.

##### **[Script Manager]**

Start CAO script manager.

#### 5.2.3.6. Window menu

This menu is for window display setting.

##### **[Horizontal]**

Align child windows horizontally.

##### **[Vertical]**

Align child windows vertically.

**[Cascade]**

Cascade child windows.

**[Arrange Icon]**

Arrange minimized window icons.

**[Close All]**

Close all child windows.

### 5.2.3.7. Help Window

This window is to display help information.

**[Version]**

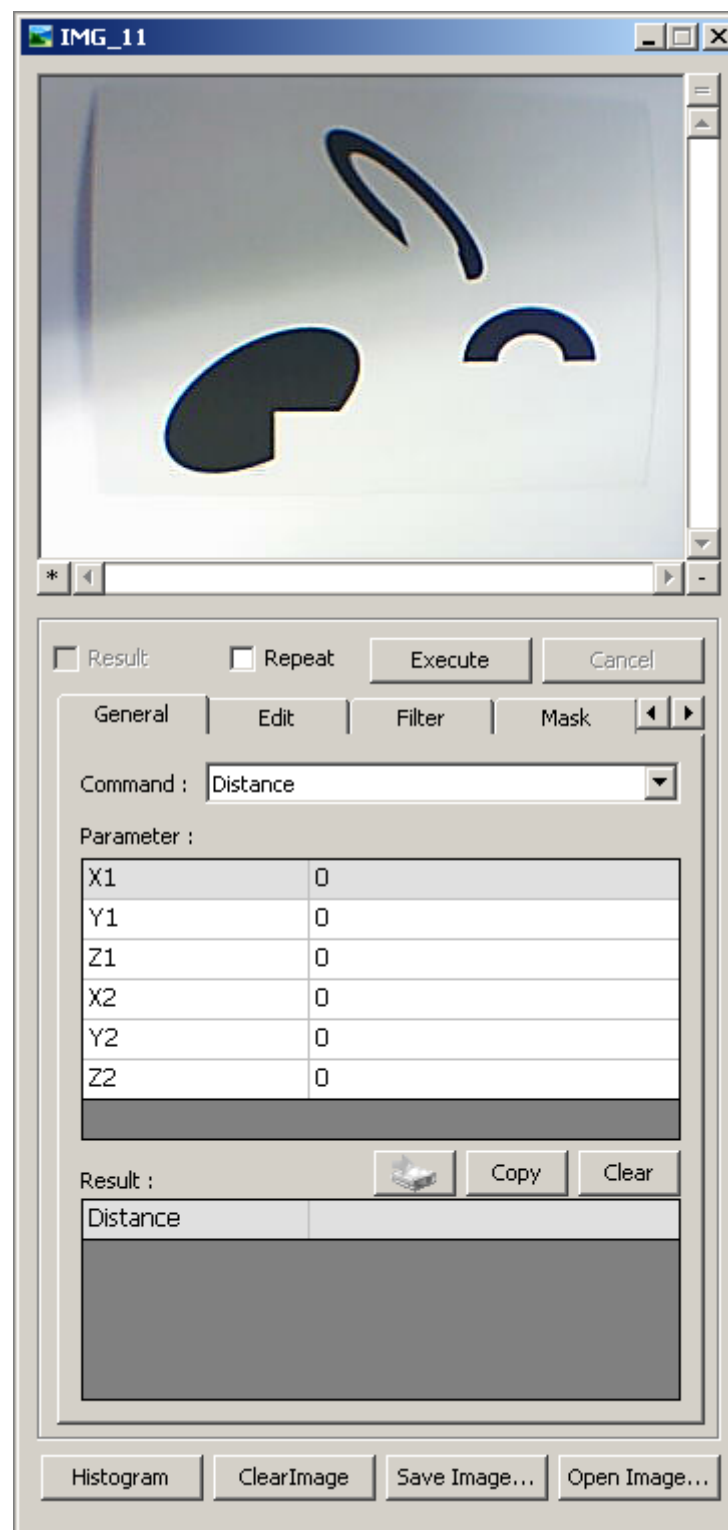
Display version information.

## 5.3. Image window

Image window is to display images and perform image-processing command.

Command execution procedure is as following.

1. Select command type from tab.
2. Select command from Command combo-box.
3. Set executed command parameter.
4. Press Execute button to execute command.
5. Result displays acquired value. (Some commands don't acquire value.) Executed result is shown in the log window at the left bottom part of the main screen.

**Figure 5-3 Image window**

[Image monitor]

Display image to image window.

When left below [\*] button is ON, the image is output to main screen.

The capture screen is displayed by [=] button in the upper right corner. This function is valid only for the memory area image.

Right below [+] button and [-] button is to display and hide image monitor.

In some commands, parameter setting by dragging on an image is possible. Following is the list of the command supporting area assignment.

**Table 5-1 List of commands supporting area assignment**

| Shape     | Command          | Operation                                                                      |
|-----------|------------------|--------------------------------------------------------------------------------|
| Rectangle | SetROI           | Move      5pixels : [ ↑ ] [ ↓ ] [←][→]                                         |
|           | Cut              | 1pixel : [Ctrl] + [ ↑ ] [ ↓ ] [←][→]                                           |
|           | Rectangle        | Resize      5pixels : [Shift] + [ ↑ ] [ ↓ ] [←][→]                             |
|           | CamShift         | 1pixel : [Ctrl] + [Shift] + [ ↑ ] [ ↓ ] [←][→]                                 |
|           |                  | Automatic fitting      [Shift]+[Enter]                                         |
| Line      | Line             | Move      5pixels : [ ↑ ] [ ↓ ] [←][→]                                         |
|           |                  | 1pixel : [Ctrl] + [ ↑ ] [ ↓ ] [←][→]                                           |
|           |                  | Resize      5pixels : [Shift] + [ ↑ ] [ ↓ ] [←][→]                             |
|           |                  | 1pixel : [Ctrl] + [Shift] + [ ↑ ] [ ↓ ] [←][→]                                 |
| Point     | PutColor         | Move      5pixels : [ ↑ ] [ ↓ ] [←][→]<br>1pixel : [Ctrl] + [ ↑ ] [ ↓ ] [←][→] |
|           | GetColor         |                                                                                |
|           | SearchPoint      |                                                                                |
|           | Paste            |                                                                                |
|           | Rotate           |                                                                                |
|           | Line2            |                                                                                |
|           | Circle           |                                                                                |
|           | Ellipse          |                                                                                |
|           | Sector           |                                                                                |
|           | Cross            |                                                                                |
|           | Text             |                                                                                |
|           | ContoursNumber   |                                                                                |
|           | PointPolygonTest |                                                                                |
|           | GetPosFromCam    |                                                                                |
|           | BoxPoints        |                                                                                |

**[Repeat]**

Repeat command execution. To stop repeated command execution, uncheck Repeat, or click Cancel button. Repeated execution outputs large amount of logs.

**[Execute]**

Execute a command. Execute command selected from Command combo. Values set in Parameter property box is used for execution parameter. The execution result is displayed in Result property box. Execution result is output to log window.

When Repeat is checked, the command execution is repeated until Cancel button is clicked.

**[Cancel]**

Stop repeatedly executing command.

**[Command Type]**

Select command type.

**[Command]**

Select execution command.

**[Parameter]**

Set parameters for command execution.

**[Result]**

Display acquired data of command execution result.

Execution result is stored to clipboard by clicking Copy button.

Clicking Clear button clears currently displayed execution result.

Clicking the data transfer button sends result to DENSO robot.

**[Histogram]**

Display image histogram. Histogram is updated when "CalcHistEx" command is executed.

**[Save Image]**

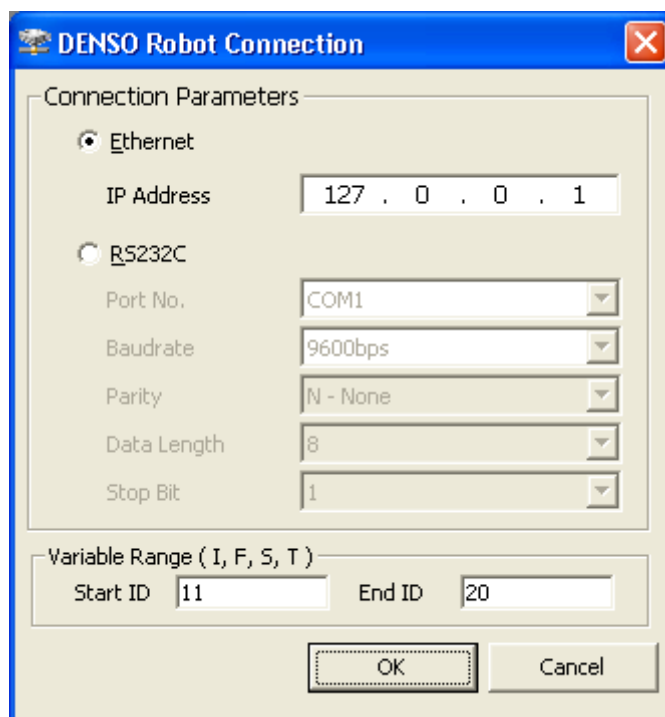
Save image displayed on image monitor as bitmap.

**[Load Image]**

Load bitmap and display on image monitor. The button cannot be used in image window for camera images.

## 5.4. DENSO Robot connection window

The window is to setup conditions to connect to DENSO Robot.

**[Ethernet]**

Use Ethernet to connect to robot controller.

**[IP Address]**

Specify robot controller IP address.

This parameter is available only when Ethernet connection is selected.

**[RS232C]**

Use RS232C to connect to robot controller.

**[Port No.]**

Specify COM port number to communicate to robot controller.

This parameter is available only when RS232C connection is selected.

**[Baudrate]**

Specify baudrate to communicate to robot controller.

This parameter is available only when RS232C connection is selected.

**[Parity]**

Specify parity setting to communicate to robot controller.

Only “No parity” is available for this item, and cannot be changed.

**[Data Length]**

Specify data length to communicate to robot controller.

Only “8bit” is available for this item, and cannot be changed.

**[Stop Bit]**

Specify stop bit to communicate to robot controller.



Only "1bit" is available for this item, and cannot be changed.

**[Variable Range]**

Set variable range for DENSO robot result output execution.

## 5.5. Camera Settings window

Set camera and image memory size.

The setting is effective after restarting OcvTester.

The setting is registered in registry. (refer 2.1)

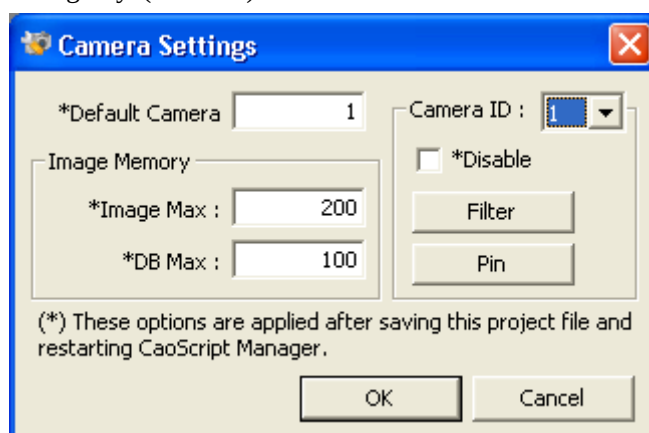


Figure 5-4 Camera Setting window

**[Camera ID]**

Specify camera ID to setup.

**[Disable]**

Checking this item disables camera operation.

**[Filter]**

Open a camera filter property window.

**[Pin]**

Open a output Pin property window.

**[Image Max]**

Set image memory max index number.

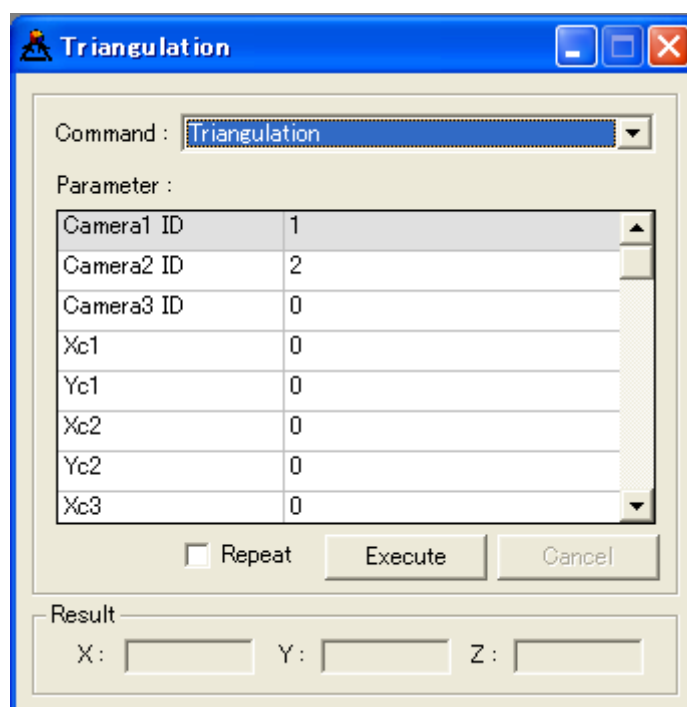
**[DB Max]**

Set image database max index number.

## 5.6. Triangulation window

Calculate coordinate using triangulation.

To use this function, two or three calibrated camera is necessary.

**Figure 5-5 Triangulation window****[Command]**

Select execution command.

**[Parameter]**

Set command execution parameters.

**[Repeat]**

Repeat command execution. To stop repeated command execution, uncheck Repeat, or press Cancel button. Repeated execution outputs large amount of logs.

**[Execute]**

Execute command. Execute a command selected from Command combo. Values set in Parameter property box is used for execution parameter. The execution result is displayed in Result property box. Execution result is output to log window.

When Repeat is checked, the command execution is repeated until Cancel button is clicked.

**[Cancel]**

Stop repeatedly executing command.

**[Result]**

Display acquired data of command execution result.

## 5.7. Calibration Wizard

### 5.7.1. Overview

The wizard is for camera calibration and robot calibration.

Calibration wizard supports following four types of calibrations.

#### [Camera]

Calibrate camera.

CalibrateCamera is called.

#### [Robot]

Calibrate robot.

CalibrateRobot is called.

#### [Camera attached on a cell & Robot]

Calibrate attached camera on a cell and robot.

CalibrateCamera and CalibrateRobot is called.

Calibration Wizard performs calibration in the following six steps.

[Step 0] Specify calibration target

[Step 1] Set camera calibration parameter

[Step 2] Acquire chessboard image

[Step 3] Calculate mapping between world coordinate and robot coordinate

[Step 4] Show completion message

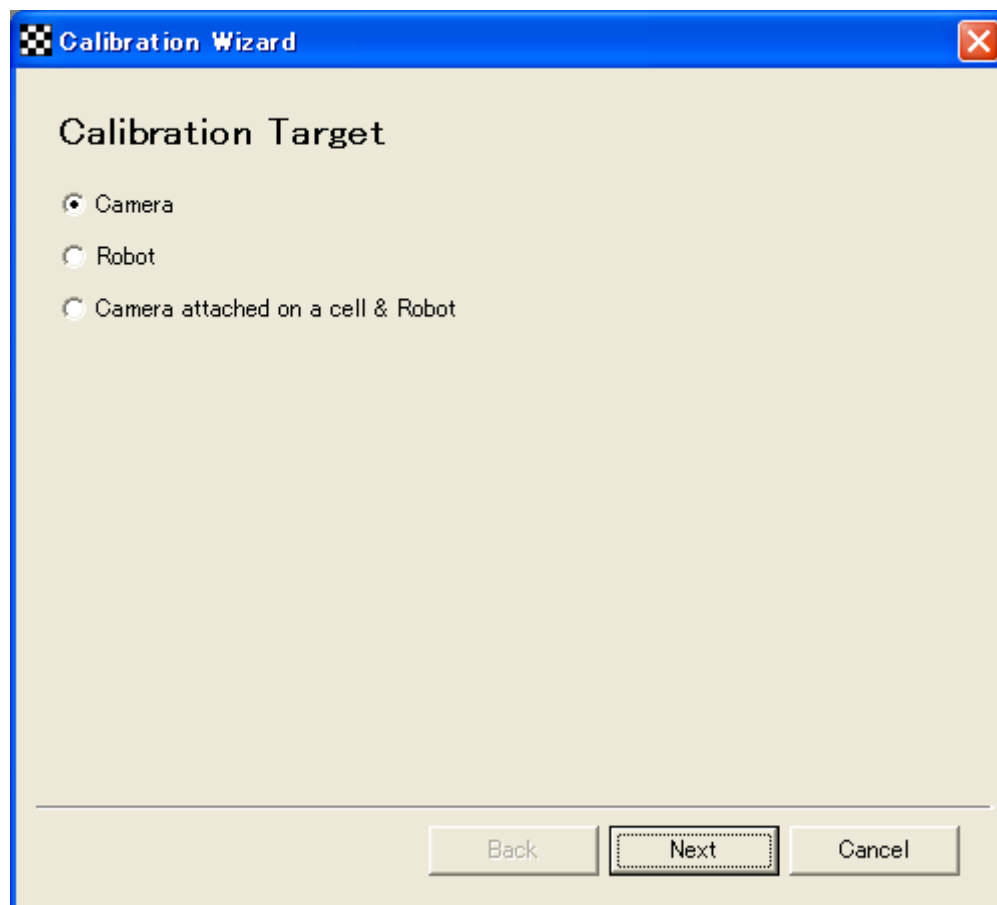
Detail of each step is described later in this manual.

Executed step is different depending on the calibration target. Following table shows a list of executed steps for each calibration target.

**Table 5-2 Execution steps for each calibration target**

| Calibration target                | Step 0 | Step 1 | Step 2 | Step 3 | Step 4 |
|-----------------------------------|--------|--------|--------|--------|--------|
| Camera                            | ○      | ○      | ○      | →      | ○      |
| Robot                             | ○      | →      | →      | ○      | ○      |
| Camera attached on a cell & Robot | ○      | ○      | ○      | ○      | ○      |

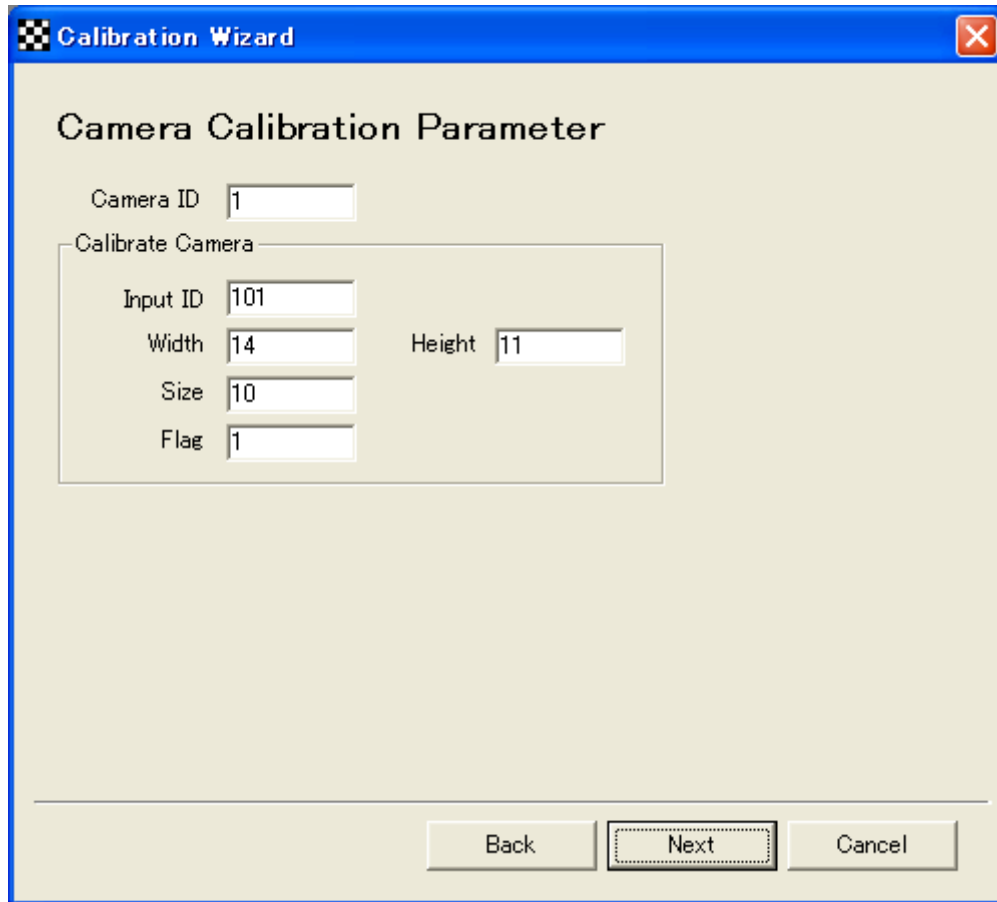
### 5.7.2. Step 0: Select calibration target



**Figure 5-6 Step 0 : Calibration target selection**

This screen is to select calibration target. For details of each calibration target, please refer 5.6.1

### 5.7.3. Step 1 : Set camera calibration parameter



The screenshot shows a Windows-style dialog box titled "Calibration Wizard" with a close button (X) in the top right corner. The main title inside the dialog is "Camera Calibration Parameter". Below this title, there is a "Camera ID" field with the value "1". Underneath, there is a section titled "Calibrate Camera" which contains several input fields: "Input ID" (101), "Width" (14), "Height" (11), "Size" (10), and "Flag" (1). At the bottom of the dialog, there are three buttons: "Back", "Next" (which is highlighted with a dashed border), and "Cancel".

**Figure 5-7 Step 1 : Camera calibration parameter setting**

**[Camera ID]**

Specify calibrated camera ID.

**[Input ID]**

Specify chessboard image storing destination ID for camera calibration. The images are stored sequentially from the specified ID.

**[Width]**

Specify the width (number of columns) of the calibration chessboard.

**[Height]**

Specify the height (number of rows) of the calibration chessboard.

**[Size]**

Specify the square size of the calibration chessboard.

**[Flag]**

Specify the flag of camera calibration

#### 5.7.4. Step 2 : Acquire chessboard image

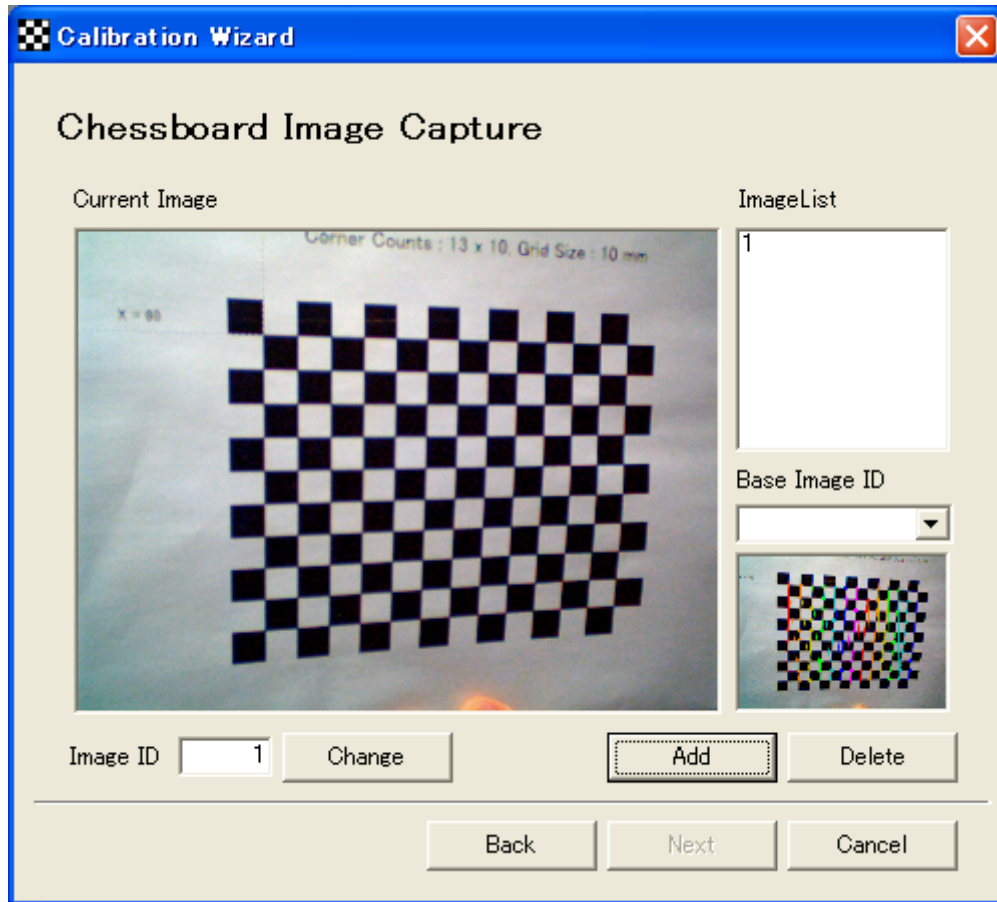


Figure 5-8 Step 3 : Chess board image acquisition

**[Current Image]**

Display image of the specified ImageID.

**[Image ID]**

Specify image ID to store acquired image.

**[Image List]**

Display a list of acquired images. Clicking an item from the list will display a thumbnail image.

**[BaseImage]**

Specify the image ID that is used as a reference (base) image for camera calibration

**[Add]**

Add current image to the list. Only images recognized as chessboard image are added to the image list. The chessboard check result is displayed on the thumbnail.

**[Delete]**

Delete currently selected image from image list.

### 5.7.5. Step 3 : Map world coordinate and robot coordinate

**Calibration Wizard**

## World-Robot Coordination Mapping

Points

| World Point | Robot Point |
|-------------|-------------|
| (0,0,0)     | (10,10,10)  |
|             |             |

World Point

X

Y

Z

Robot Point

X

Y

Z

**Figure 5-9 Step 4 : World coordinate and robot coordinate mapping**

**[Points]**

Display mapping list of world coordinate and robot coordinate

The selected mapping point list is displayed in input item of world coordinate and robot coordinate.

**[World Point]**

Specify a point of world coordinate. [NOTE]

**[Robot Point]**

Specify a point of robot coordinate. [NOTE]

**[GetCurrentPos]**

Get current position from robot controller, and set to each item of robot coordinate.

**[Add]**

Add input world coordinate and robot coordinate to list as corresponding points.

**[Edit]**

Edit list item of selected world coordinate and corresponding robot coordinate.

**[Delete]**

Delete selected mapping information form the list.

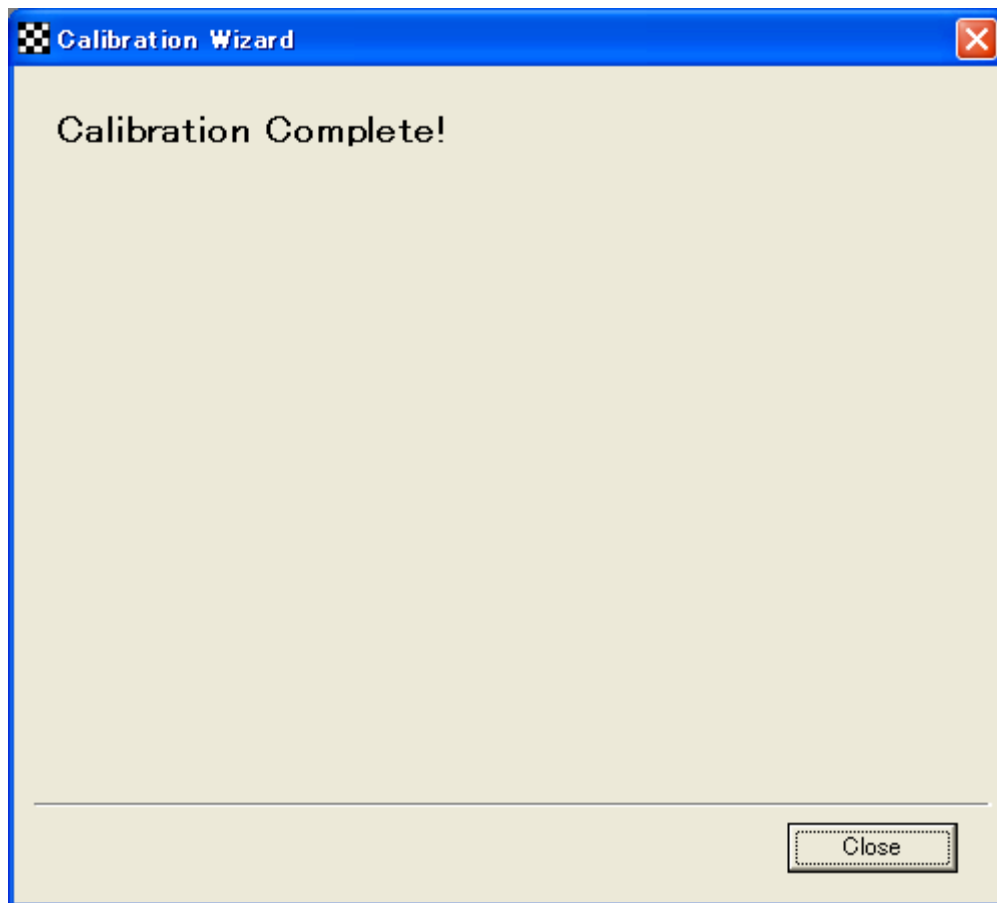
---

[NOTE] In case of 'Camera + Robot' Calibration

The left-bottom of the base image (Chessboard) used in the camera calibration is the world origin (0, 0, 0). In case of 'Camera + Robot' calibration, it is easy to use the chessboard for specifying a world point and a robot point. To put it concretely, any points (X, Y, 0) on the chessboard can be specified as a world point. And next, move the robot to the point, and click [GetCurrentPos] to get the current robot point. Repeat those steps more than 3 times. We recommend attaching a teaching pointer like a ball point pen.

---

#### 5.7.6. Step 4 : Complete Wizard



**Figure 5-10 Step 5 : Completion screen**

This screen is displayed when calibration is finished. Click Close button to exit calibration wizard.

### 5.8. Lookup table editor

Edit lookup table.



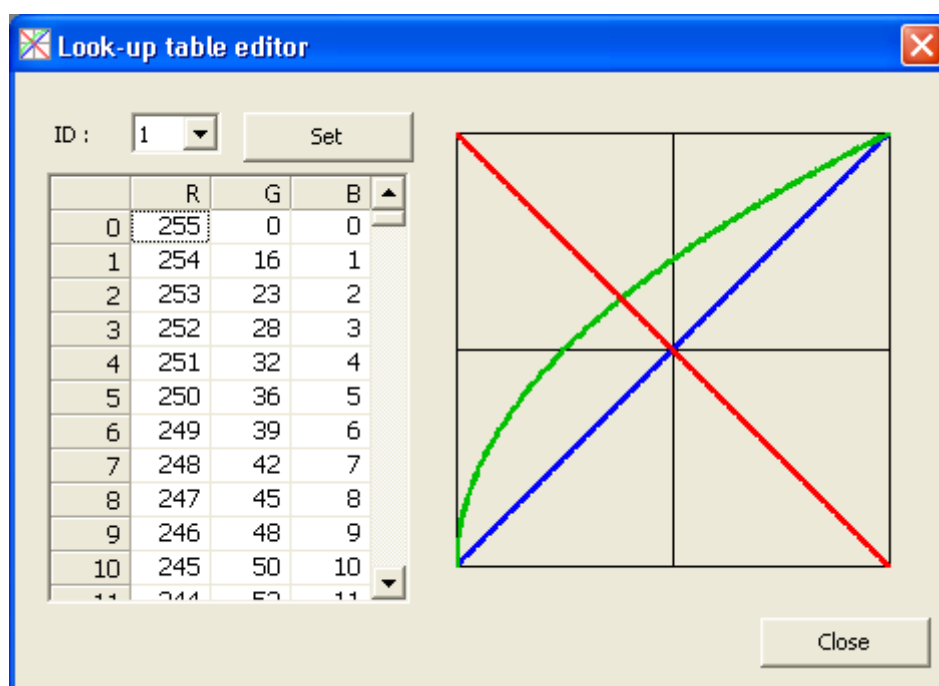


Fig 5-1 Lookup table editor

**[ID]**

Specify edited lookup table ID.

**[Look-up table]**

Display and edit current lookup table.

R,G, B shows hue.

**[Graph]**

Show the graph of each hue of current lookup table.

**[Set]**

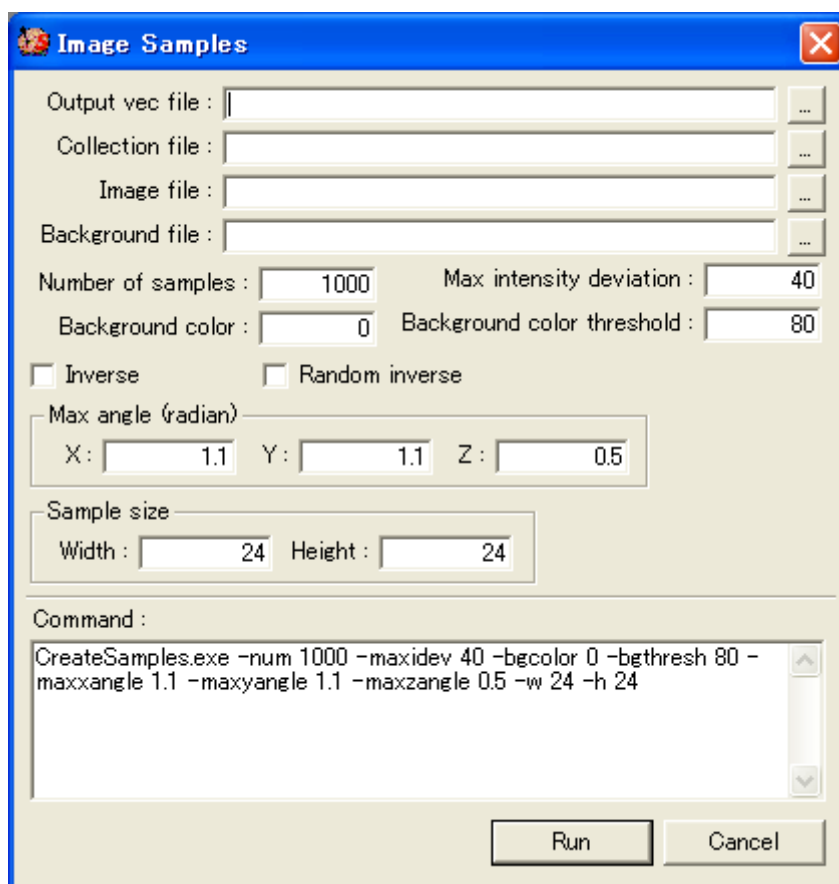
Set edited lookup table.

**[Close]**

Exit lookup table editor.

## 5.9. Image sampling window

Create positive image used for Haar training, by using CreateSamples.exe command of OpenCV.



**Figure 5-11 Image sampling window**

**[Output vec file]**

Specify created positive file output destination.

**[Collection file]**

Specify collection file. Collection file describes a list of image files containing detected images. For details of the file format, please refer to the document for CreateSamples.exe.

**[Image file]**

Specify detected image file.

**[Background file]**

Specify background image collection file.

**[Number of samples]**

Specify output positive image sample number.

**[Max intensity deviation]**

Specify front image maximum intensity deviation.

**[Background color]**

Change specified background color to transparent.

**[Background color threshold]**

Change areas of “Background color” plus/minus “background color threshold” to be transparent.

**[Inverse]**

Inverse color.

**[Random imverse]**

Randomly inverse color.

**[Max angle]**

Specify maximum rotation angle of detected image.

**[Sample size]**

Specify created positive image size.

**[Command]**

Show executed command in the format of CreateSamples.exe command line.

**[Run]**

Execute command displayed in Command.

**[Cancel]**

Cancel command execution and close window.

## 5.10. Haar training window

Create XML file used in “HaarDetect” command by using HaarTraining.exe command of OpenCV.

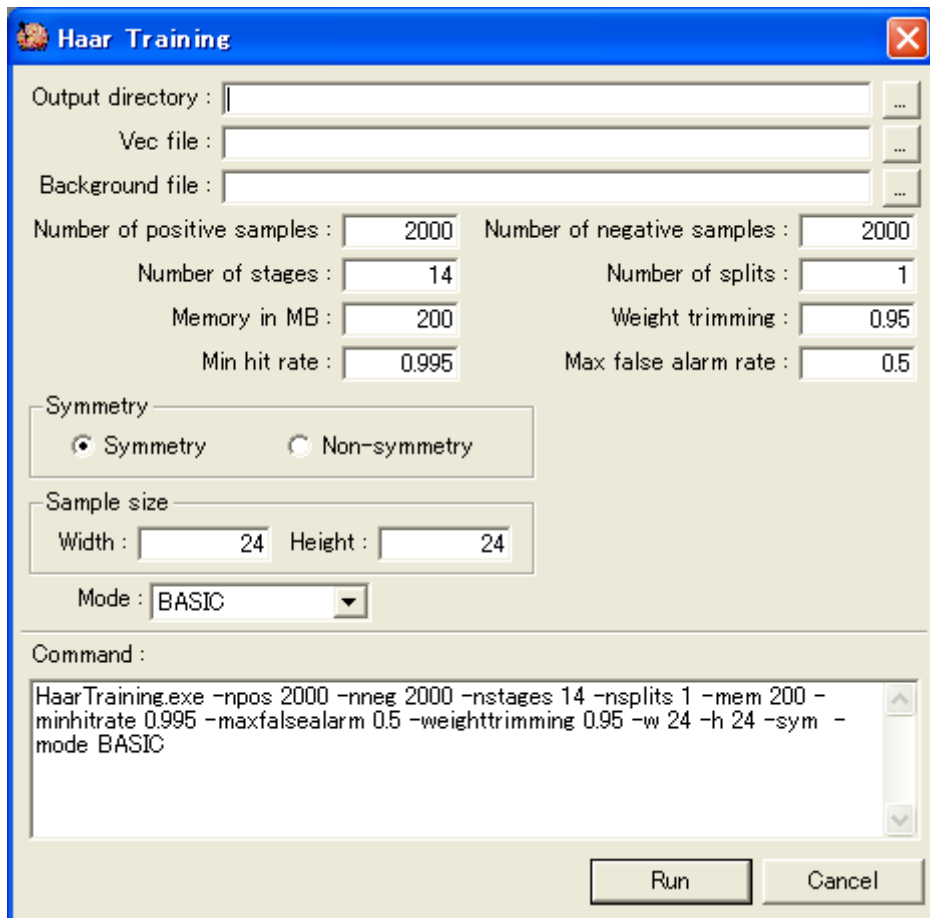


Figure 5-12 Image sampling window

**[Output directory]**

Specify Haar training result output directory.

**[Vec file]**

Specify positive imag4e file.

**[Background file]**

Specify background file.

**[Number of positive samples]**

Specify the number of positive samples to be used for stage training of each classifier.

**[Number of negative samples]**

Specify the number of negative samples to be used for stage training of each classifier.

**[Number of stage]**

Specify the training stage number.

**[Number of split]**

Specify number of weak classifiers used in classifier stage. If 1 is specified, simple stump classifier

is used. If 2 or more is specified, CART classifier with “Number of split” internal branch node is used.

**[Memory in MB]**

Specify the size of memory to be used for calculation.

**[Weight trimming]**

Specify the degree of using weight trimming.

**[Min hit rate]**

Specify minimum hit rate necessary for each stage classifier.

**[Max flase alarm rate]**

Specify maximum false alarm rate required for classifiers of each stages.

**[Symmetry]**

In the training, specify whether the object is symmetry to vertical axis.

**[Sample size]**

Specify positive image size.

**[Mode]**

Select Haar characteristics set type used for training.

**[Command]**

Show executed command in the format of HaarTraining.exe command line.

**[Run]**

Execute command displayed in Command.

**[Cancel]**

Cancel command execution and close window.

## Appendix A. OpenCV method implementation list

**Table A-1 penCV method implementation list**

| CaoFile::Execute command name | Implemented OpenCV method                                                       |
|-------------------------------|---------------------------------------------------------------------------------|
| SetROI                        | cvRect<br>cvSetImageROI                                                         |
| GetROI                        | cvGetImageROI                                                                   |
| ResetROI                      | cvResetROI                                                                      |
| PutColor                      | cvSet2D                                                                         |
| GetColor                      | cvGet2D                                                                         |
| ImageSize                     | -                                                                               |
| Trim                          | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvReleaseImage<br>cvGet2D |
| SearchPoint                   | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvReleaseImage<br>cvGet2D |
| Distance                      | -                                                                               |
| InnerProduct                  | -                                                                               |
| OuterProduct                  | -                                                                               |
| Copy                          | cvCreateImage<br>cvGetSize<br>cvCopy<br>cvReleaseImage                          |
| Cut                           | cvReleaseImage<br>cvRect<br>cvSetImageROI<br>cvCreateImage<br>cvGetSize         |

|             |                                                                                                                   |
|-------------|-------------------------------------------------------------------------------------------------------------------|
|             | cvCopy                                                                                                            |
| Paste       | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvRect<br>cvSetImageROI<br>cvCopy<br>cvConvert<br>cvReleaseImage      |
| Rotate      | cvCreateImage<br>cvGetSize<br>cvCreateMat<br>cv2DRotationMatrix<br>cvWarpAffine<br>cvReleaseImage<br>cvReleaseMat |
| Flip        | cvCreateImage<br>cvGetSize<br>cvFlip                                                                              |
| Resize      | cvCreateImage<br>cvSize<br>cvResize                                                                               |
| Split       | cvCreateImage<br>cvGetSize<br>cvSplit<br>cvReleaseImage                                                           |
| Marge       | cvCreateImage<br>cvGetSize<br>cvMerge                                                                             |
| ConvertGray | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvReleaseImage                                              |
| ThresholdEx | cvCreateImage<br>cvGetSize                                                                                        |

|                     |                                                                                                |
|---------------------|------------------------------------------------------------------------------------------------|
|                     | cvCvtColor<br>cvCopy<br>cvThreshold<br>cvReleaseImage                                          |
| Threshold2          | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvThreshold<br>cvReleaseImage            |
| AdaptiveThresholdEx | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvAdaptiveThreshold<br>cvReleaseImage    |
| Smooth              | cvCreateImage<br>cvGetSize<br>cvSmooth<br>cvReleaseImage                                       |
| Sobel               | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvSobel<br>cvConvert<br>cvReleaseImage   |
| Laplace             | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvLaplace<br>cvConvert<br>cvReleaseImage |
| CannyEx             | cvCreateImage<br>cvGetSize                                                                     |



|                   |                                                                                                                     |
|-------------------|---------------------------------------------------------------------------------------------------------------------|
|                   | cvCvtColor<br>cvCopy<br>cvCanny<br>cvReleaseImage                                                                   |
| WarpAffine        | cvCreateImage<br>cvGetSize<br>cvCreateMat<br>cvSetReal2D<br>cvWarpAffine<br>cvReleaseImage<br>cvReleaseMat          |
| WarpPerspective   | cvCreateImage<br>cvCreateMat<br>cvSetReal2D<br>cvWarpPerspective<br>cvReleaseMat                                    |
| PreCornerDetectEx | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvConvert<br>cvPreCornerDetect<br>cvConvert<br>cvReleaseImage |
| CornerHarrisEx    | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvConvert<br>cvCornerHarris<br>cvReleaseImage                 |
| CalcBackProjectEx | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvCreateHist                                                  |

|         |                                                                                          |
|---------|------------------------------------------------------------------------------------------|
|         | cvCalcHist<br>cvCalcBackProject                                                          |
| Inpaint | cvCreateImage<br>cvGetSize<br>cvInpaint                                                  |
| Erode   | cvCreateStructuringElementEx<br>cvCreateImage<br>cvErode<br>cvReleaseStructuringElement  |
| Dilate  | cvCreateStructuringElementEx<br>cvCreateImage<br>cvDilate<br>cvReleaseStructuringElement |
| PyrDown | cvGetSize<br>cvCreateImage<br>cvPyrDown                                                  |
| PyrUp   | cvGetSize<br>cvCreateImage<br>cvPyrUp                                                    |
| NOT     | cvCreateImage<br>cvGetSize<br>cvNot                                                      |
| AND     | cvCreateImage<br>cvGetSize<br>cvAnd<br>cvReleaseImage                                    |
| OR      | cvCreateImage<br>cvGetSize<br>cvOr<br>cvReleaseImage                                     |
| XOR     | cvCreateImage<br>cvGetSize<br>cvXor<br>cvReleaseImage                                    |
| ADD     | cvCreateImage<br>cvGetSize                                                               |

|           |                                                                               |
|-----------|-------------------------------------------------------------------------------|
|           | cvAdd<br>cvReleaseImage                                                       |
| SUB       | cvCreateImage<br>cvGetSize<br>cvSub<br>cvReleaseImage                         |
| MAXEx     | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvMax<br>cvReleaseImage |
| MINEx     | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvMin<br>cvReleaseImage |
| ABS       | cvCreateImage<br>cvGetSize<br>cvAbsDiff<br>cvReleaseImage                     |
| Line      | cvCreateImage<br>cvGetSize<br>cvCopy<br>cvLine<br>cvPoint<br>cvReleaseImage   |
| Line2     | cvCreateImage<br>cvGetSize<br>cvCopy<br>cvLine<br>cvPoint<br>cvReleaseImage   |
| Rectangle | cvCreateImage<br>cvGetSize                                                    |

|         |                                                                                                    |
|---------|----------------------------------------------------------------------------------------------------|
|         | cvCopy<br>cvRectangle<br>cvPoint<br>cvReleaseImage                                                 |
| Circle  | cvCreateImage<br>cvCopy<br>cvCircle<br>cvPoint<br>cvReleaseImage                                   |
| Ellipse | cvCreateImage<br>cvGetSize<br>cvCopy<br>cvEllipse<br>cvPoint<br>cvSize<br>cvReleaseImage           |
| Sector  | cvCreateImage<br>cvGetSize<br>cvCopy<br>cvLine<br>cvEllipse<br>cvPoint<br>cvSize<br>cvReleaseImage |
| Cross   | cvCreateImage<br>cvGetSize<br>cvCopy<br>cvLine<br>cvPoint<br>cvReleaseImage                        |
| Text    | cvCreateImage<br>cvGetSize<br>cvFlip<br>cvInitFont<br>cvPutText<br>cvPoint                         |

|                       |                                                                                                                                                                                   |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                       | cvReleaseImage                                                                                                                                                                    |
| FindContoursEx        | cvCreateImage<br>cvGetSize<br>cvClearMemStorage<br>cvFindContours<br>cvReleaseImage                                                                                               |
| CopyContours          | cvBoundingRect<br>cvReleaseImage<br>cvRect<br>cvSetImageROI<br>cvCreateImage<br>cvGetSize<br>cvCopy                                                                               |
| ContoursNumber        | cvPointPolygonTest                                                                                                                                                                |
| PointPolygonTest      | cvPointPolygonTest                                                                                                                                                                |
| BoundingRect          | cvBoundingRect                                                                                                                                                                    |
| FitEllipse            | cvFitEllipse2                                                                                                                                                                     |
| ArcLength             | cvArcLenth                                                                                                                                                                        |
| CheckContourConvexity | cvCheckContourConvexity                                                                                                                                                           |
| FindBlobs             | -                                                                                                                                                                                 |
| BlobsFilter           | -                                                                                                                                                                                 |
| BlobResult            | -                                                                                                                                                                                 |
| BlobResults           | -                                                                                                                                                                                 |
| BlobEllipse           | -                                                                                                                                                                                 |
| BlobMatchTemplate     | cvCloneImage<br>cvCreateImage<br>cvCopy<br>cvGetSize<br>cvResize<br>cvSet<br>cvSetImageROI<br>cvResetImageROI<br>cvCreateMemStorage<br>cvCreateSeq<br>cvPoint2D32f<br>cvSize2D32f |

|                 |                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                 | <div>cvCreateMat</div> <div>cv2DRotationMatrix</div> <div>cvWarpAffine</div> <div>cvBoxPoints</div> <div>cvMatchTemplate</div> <div>cvMinMaxLoc</div> <div>cvSeqPush</div> <div>cvSeqSort</div> <div>cvGetSeqElem</div> <div>cvClearSeq</div> <div>cvReleaseImage</div> <div>cvReleaseMat</div> <div>cvReleaseMemStorage</div>                                                           |
| BlobMatchShapes | <div>cvCloneImage</div> <div>cvCreateMemStorage</div> <div>cvCreateImage</div> <div>cvCvtColor</div> <div>cvCopy</div> <div>cvClone</div> <div>cvFindContours</div> <div>cvBoundingRect</div> <div>cvSetImageROI</div> <div>cvClearMemStorage</div> <div>cvFindContours</div> <div>cvMatchShapes</div> <div>cvFitEllipse2</div> <div>cvReleaseImage</div> <div>cvReleaseMemStorage</div> |
| CalcHistEx      | <div>cvCreateImage</div> <div>cvGetSize</div> <div>cvCvtColor</div> <div>cvCopy</div> <div>cvCreateHist</div> <div>cvCalcHist</div> <div>cvQueryHistValue_1D</div> <div>cvReleaseHist</div>                                                                                                                                                                                              |

|                    |                                                                                        |
|--------------------|----------------------------------------------------------------------------------------|
|                    | cvReleaseImage                                                                         |
| NormalizeHistEx    | cvCreateHist<br>cvSetReal1D<br>cvNormalizeHist<br>cvQueryHistValue_1D<br>cvReleaseHist |
| ThreshHistEx       | cvCreateHist<br>cvSetReal1D<br>cvThreshHist<br>cvQueryHistValue_1D<br>cvReleaseHist    |
| EqualizeHistEx     | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvEqualizeHist                   |
| GetMinMaxHistValue | cvCreateHist<br>cvSetReal1D<br>cvGetMinMaxHistValue<br>cvReleaseHist                   |
| HistAve            | -                                                                                      |
| AutoThreshPTile    | -                                                                                      |
| AutoThreshMode     | -                                                                                      |
| AutoThreshDiscrim  | -                                                                                      |
| MatchTemplate      | cvCreateImage<br>cvSize<br>cvMatchTemplate<br>cvMinMaxLoc<br>cvReleaseImage            |
| MatchShapesEx      | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvMatchShapes<br>cvReleaseImage  |
| CamShift           | cvRect                                                                                 |

|                |                                                                                                                                                                                                                                                                                                                                                                                          |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                | cvTermCriteria<br>cvCamShift                                                                                                                                                                                                                                                                                                                                                             |
| MatchTemplate2 | cvCreateImage<br>cvCopy<br>cvGetSize<br>cvResize<br>cvSet<br>cvSetImageROI<br>cvResetImageROI<br>cvCreateMemStorage<br>cvCreateSeq<br>cvPoint2D32f<br>cvSize2D32f<br>cvCreateMat<br>cv2DRotationMatrix<br>cvWarpAffine<br>cvBoxPoints<br>cvMatchTemplate<br>cvMinMaxLoc<br>cvSeqPush<br>cvSeqSort<br>cvGetSeqElem<br>cvClearSeq<br>cvReleaseImage<br>cvReleaseMat<br>cvReleaseMemStorage |
| MatchShapes2   | cvCreateMemStorage<br>cvCreateImage<br>cvCvtColor<br>cvCopy<br>cvClone<br>cvFindContours<br>cvBoundingRect<br>cvSetImageROI<br>cvClearMemStorage<br>cvFindContours                                                                                                                                                                                                                       |



|                       |                                                                                                                                                                                                                                               |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                       | cvMatchShapes<br>cvFitEllipse2<br>cvReleaseImage<br>cvReleaseMemStorage                                                                                                                                                                       |
| HaarDetect            | cvLoad<br>cvCreateMemStorage<br>cvHaarDetectObjects<br>cvGetSeqElem<br>cvReleaseMemStorage                                                                                                                                                    |
| CalibrateCamare       | cvCreateMat<br>cvCalibrateCamera2<br>cvRodrigues2<br>cvMat<br>cvMatMul<br>cvInvert<br>cvSetReal2D<br>cvGetReal2D<br>cvSet2D<br>cvReleaseMat<br>cvCreateImage<br>cvFindChessboardCorners<br>cvCvtColor<br>cvFindCornerSubPix<br>cvReleaseImage |
| CalibrateRobot        | cvCreateMat<br>cvmSet<br>cvSolve<br>cvReleaseMat                                                                                                                                                                                              |
| FindChessBoardCorners | cvFindChessboardCorners                                                                                                                                                                                                                       |
| DrawChessBoardCorners | cvCreateImage<br>cvCopy<br>cvDrawChessboardCorners                                                                                                                                                                                            |
| DrawXYAxes            | cvCreateImage<br>cvGetSize<br>cvCopy<br>cvLine                                                                                                                                                                                                |

|                 |                                                                                |
|-----------------|--------------------------------------------------------------------------------|
|                 | cvPoint<br>cvReleaseImage<br>cvFlip<br>cvInitFont<br>cvPutText                 |
| SetCamCalDat    | -                                                                              |
| GetCamCalDat    | -                                                                              |
| SetCamCalExtDat | -                                                                              |
| GetCamCalExtDat | -                                                                              |
| SetRobCalDat    | -                                                                              |
| GetRobCalDat    | -                                                                              |
| GetPosFromCam   | cvCreateMat<br>cvmSet<br>cvmGet<br>cvMatMul<br>cvReleaseMat                    |
| GetCamPos       | cvCreateMat<br>cvmSet<br>cvmGet<br>cvMatMul<br>cvReleaseMat                    |
| GetPosFromRob   | cvInitMatHeader<br>cvCreateMat<br>cvmSet<br>cvmGet<br>cvMatMul<br>cvReleaseMat |
| GetRobPos       | cvInitMatHeader<br>cvCreateMat<br>cvmSet<br>cvMatMul<br>cvmGet<br>cvReleaseMat |
| Undistort2      | cvInitMatHeader<br>cvCreateImage<br>cvUndistort2                               |

|                      |                                                                                                                        |
|----------------------|------------------------------------------------------------------------------------------------------------------------|
| GoodFeatureToTrackEx | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvGoodFeaturesToTrack<br>cvReleaseImage                          |
| FindCornerSubPixEx   | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvFindCornerSubPix<br>cvSize<br>cvTermCriteria<br>cvReleaseImage |
| MomentsEx            | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvMoments<br>cvReleaseImage                                      |
| MeasureInfo          | -                                                                                                                      |
| HoughLine            | cvCreateMemStorage<br>cvHoughLines2<br>cvGetSeqElem<br>cvReleaseImage<br>cvReleaseMemStorage                           |
| HoughCircles         | cvCreateMemStorage<br>cvHoughCircles<br>cvGetSeqElem<br>cvReleaseImage<br>cvReleaseMemStorage                          |
| DFTEEx               | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvConvart                                                        |

|                |                                                                                             |
|----------------|---------------------------------------------------------------------------------------------|
|                | cvZero<br>cvMerge<br>cvDFT<br>cvPow<br>cvAdd<br>cvReleaseImage                              |
| IDFT           | cvCreateImage<br>cvMerge<br>cvDFT<br>cvSplit<br>cvConvert<br>cvReleaseImage                 |
| OpticalFlowEx  | cvCreateImage<br>cvGetSize<br>cvCvtColor<br>cvCopy<br>cvCalcOpticalFlowLK<br>cvReleaseImage |
| BoxPoints      | cvBoxPoints                                                                                 |
| FindHomography | cvCreateMat<br>cvmSet<br>cvFindHomography<br>cvmGet<br>cvReleaseMat                         |
| QRDecode       | -                                                                                           |
| OCRead         | -                                                                                           |

## Appendix B. uVision21 equivalent OpenCV method

**Table B-1 uVision21 function equivalent OpenCV method list**

| Vision function identifier | Function                                                              | OpenCV provider equivalent method                                                                                                                                                    |
|----------------------------|-----------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CAMIN                      | Store image from camera in image memory (processing screen).          | File::put_Value = File::get_Value                                                                                                                                                    |
| CAMMODE                    | Set functions for storing camera image in image memory.               | Not implemented<br>(Camera setting should be changed)                                                                                                                                |
| CAMLEVEL                   | Set camera image input level.                                         | Not implemented<br>(Camera setting should be changed)                                                                                                                                |
| VISCAMOUT                  | Display camera image on the monitor.                                  | File::get_Value(ID=0~9)                                                                                                                                                              |
| VISPLNOUT                  | Display image memory on the monitor.                                  | File::get_Value(ID>10)                                                                                                                                                               |
| VISOVERLAY                 | Display information in the drawing screen on the monitor.             | Not implemented<br>(Image representation method depends on the client program.)                                                                                                      |
| VISDEFTABLE                | Set camera image lookup table data for camera image input and output. | Not implemented<br>(Use commands of File::Execute for image processing. )                                                                                                            |
| VISREFTABLE                | Refer the lookup table data.                                          | Not implemented                                                                                                                                                                      |
| WINDMAKE                   | Specify the image processing range.                                   | Partially implemented<br>(Similar process can be achieved by cutting out the image with Cut command of File::Execute. However, cutting methods except rectangle is not implemented.) |
| WINDCLR                    | Delete window information.                                            | Not implemented                                                                                                                                                                      |
| WINDCOPY                   | Copy window information.                                              | Not implemented                                                                                                                                                                      |
| WINDREF                    | Get window information.                                               | Not implemented                                                                                                                                                                      |
| WINDDISP                   | Display specified window.                                             | Not implemented                                                                                                                                                                      |
| WINDMAKE                   | Specify image processing area                                         | Implemented partially, only rectangle.<br>File::Execite SetROI command                                                                                                               |
| WINDCLR                    | Clear window setup information.                                       | File::Execite ResetROI command                                                                                                                                                       |
| WINDCOPY                   | Copy window information.                                              | Get the data by File::Execute GetROI command, and set the data by SetROI                                                                                                             |

|             |                                                                                               |                                                                                                                                                                                                        |
|-------------|-----------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             |                                                                                               | command.                                                                                                                                                                                               |
| WINDREF     | Get window information                                                                        | File::Execute GetROI command                                                                                                                                                                           |
| WINDDISP    | Draw the setup window                                                                         | Execute SetROI then execute Copy command.                                                                                                                                                              |
| VISSCREEN   | Specify drawing screen.                                                                       | Because two or more of File object can be generated, the output image can be arbitrarily set with the client.                                                                                          |
| VISBRIGHT   | Specify the luminance for drawing.                                                            | Not implemented<br>(Brightness can be set on camera side.)                                                                                                                                             |
| VISCLS      | Clear the specified screen by painting out the screen specified with the specified luminance. | The image is cleared by setting EMPTY to File::put_Value.<br>The screen is painted out by specifying the entire screen in Rectangle command of File::Execute, and set the thickness of the line as -1. |
| VISPUTP     | Draw a point on the screen.                                                                   | PutPoint command of File::Execute                                                                                                                                                                      |
| VISLINE     | Draw a straight line on the screen.                                                           | Line2 command of File::Execite                                                                                                                                                                         |
| VISPTP      | Draw a straight line connecting two points on the screen.                                     | Line command of File::Execite                                                                                                                                                                          |
| VISRECT     | Draw a rectangle on the screen.                                                               | Rectangle command of File::Execite                                                                                                                                                                     |
| VISCIRCLE   | Draw a circle on the screen.                                                                  | Circle command of File::Execite                                                                                                                                                                        |
| VEISELLIPSE | Draw an ellipse on the screen.                                                                | Ellipse command of File::Execite                                                                                                                                                                       |
| VISSECT     | Draw a sector on the screen.                                                                  | Sector command of File::Execite                                                                                                                                                                        |
| VISCROSS    | Draw a cross mark on the screen.                                                              | Cross command of File::Execite                                                                                                                                                                         |
| VISLOC      | Specify the position to draw character.                                                       | Text command of File::Execite                                                                                                                                                                          |
| VISDEFCHAR  | Specify the size and the display method of the character.                                     | Text command of File::Execite                                                                                                                                                                          |
| VISPRINT    | Display characters and numbers on the screen.                                                 | Text command of File::Execite                                                                                                                                                                          |
| VISWORKPLN  | Specify the processing image memory.                                                          | File::ID                                                                                                                                                                                               |
| VISGETP     | Get the luminance of specified coordinates from the image memory (processing screen).         | GetP command of File::Execite                                                                                                                                                                          |
| VISHIST     | Get histogram of the screen (intensity distribution).                                         | CalcHistEx command of File::Execite                                                                                                                                                                    |
| VISREFHIST  | Read histogram result.                                                                        | CalcHistEx command of File::Execite                                                                                                                                                                    |
| VISLEVEL    | Select binary level from the histogram result.                                                | Following command in File: Execute                                                                                                                                                                     |

|             |                                                                                            |                                                                                                                      |
|-------------|--------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
|             |                                                                                            | AutoThreshPTile<br>AutoThreshMode<br>AutoThreshDiscrim                                                               |
| VISBINA     | Binalize the screen.                                                                       | Following command in File: Execute<br>ThresholdEx<br>Threshold2                                                      |
| VISBINAR    | Display the binarized screen image.                                                        | Following command in File: Execute<br>ThresholdEx<br>Threshold2                                                      |
| VISFILTER   | Apply filter on the image.                                                                 | Smooth command of File::Execite                                                                                      |
| VISMASK     | Mask operations                                                                            | Mask system command of File::Execite (AND and OR, etc.)                                                              |
| VISCOPY     | Copy screen.                                                                               | File::put_Value = File::get_Value                                                                                    |
| VISMEASURE  | The feature (area, center of gravity, and principal axis angle) in the window is measured. | Combination of the commands of File::Execite Moments, MeasureInfo, ImageSize                                         |
| VISPROJ     | Measure the projection data in the window.                                                 | Not implemented                                                                                                      |
| VISEGE      | Measure the edge in the window.                                                            | Filter system command of File::Execite (Canny and Laplace, etc.)                                                     |
| VISREADQR   | Read QR code                                                                               | File::Execite QRDecode command                                                                                       |
| BLOB        | Labeling                                                                                   | FindContoursEx command of File::Execite                                                                              |
| BLOBMEASURE | Measure the feature of the object specified by label number.                               | Combination of the following commands of File::Execite FindContoursEx, CopyContours, Moments, MeasureInfo, ImageSize |
| BLOBLABEL   | Get the label number of specified coordinate.                                              | Following command in File: Execute<br>ContoursNumber<br>PointPolygonTest                                             |
| BLOBCOPY    | Copy object label number.                                                                  | CopyContours command of File::Execite                                                                                |
| SHDEFMODEL  | Register search model                                                                      | Put_Value                                                                                                            |
| SHREFMODEL  | Refer the registered model data.                                                           | Combination of the following commands of File::Execite MomentsEx, MeasureInfo, ImageSize                             |
| SHCOPYMODEL | Copy registered model data.                                                                | File::put_Value = File::get_Value                                                                                    |

|             |                                                                      |                                                                                                                                                                                                        |
|-------------|----------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SHCLRMODEL  | Clear registered model data.                                         | The image is cleared by setting EMPTY to File::put_Value.<br>The screen is painted out by specifying the entire screen in Rectangle command of File::Execute, and set the thickness of the line as -1. |
| SHDISPMODEL | Display registered model on the screen.                              | Get_Value                                                                                                                                                                                              |
| SHMODEL     | Search model                                                         | Following command in File: Execute<br>TemplateMatch<br>HaarDetect<br>MatchEx                                                                                                                           |
| SHDEFCORNER | Set condition for corner search.                                     | HoughLine command of File: Execute                                                                                                                                                                     |
| SHCORNER    | Search corners.                                                      | HoughLine command of File: Execute                                                                                                                                                                     |
| SHDEFCIRCLE | Set condition for circle search.                                     | HoughCircles command of File: Execute                                                                                                                                                                  |
| SHCIRCLE    | Search circle.                                                       | HoughCircles command of File: Execute                                                                                                                                                                  |
| VISGETNUM   | Get image processing result from image memory.                       | Return value of Execute<br>Get_Value(For the image. )                                                                                                                                                  |
| VISGETSTR   | Get code recognition result.                                         | Return value of File::Execute<br>QRDecode command                                                                                                                                                      |
| VISPOSX     | Get image processing result (X coordinates) from the storage memory. | Return value of File::Execute                                                                                                                                                                          |
| VISPOSY     | Get image processing result (Y coordinates) from the storage memory. | Return value of File::Execute                                                                                                                                                                          |
| VISSTATUS   | Refer the processing result of each instruction.                     | Return value of File::Execute                                                                                                                                                                          |
| VISREFCAL   | Get CAL (vision - robot coordinate transformation) data.             | CAL relation                                                                                                                                                                                           |



## **Appendix C. Intel License Agreement For Open Source Computer Vision Library**

Copyright © 2000, Intel Corporation, all rights reserved. Third party copyrights are property of their respective owners.

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

- Redistribution's of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistribution's in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- The name of Intel Corporation may not be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall Intel or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

All information provided related to future Intel products and plans is preliminary and subject to change at any time, without notice.