uEye provider
IDS Camera

Version 1.0.2

User's guide

May 15, 2020

【Remarks】

【Revision history】

Version	Date	Content
1.0.0	2013-7-24	First edition.
1.0.1	2015-8-3	Supported IDS uEye Ver 4.50.
1.0.2	2020-5-15	Corrected error in GetEdgeEnhancementRange. Fixed processing of SetGamma, GetGamma, SetTriggerDelay.
	2021-10-28	Mono option was added to the AddController options.

Contents

1. Introduction

This book is a user's guide of the uEye provider that is the CAO provider for a USB camera made by IDS.

The uEye provider connects a USB-connected camera using a command of uEyeSDK, and then controls the camera and acquires an image. Some functions which are specific to models are not mounted. The operation check was done by the following camera.

UI-1480SE-M-GL

This provider does not support the free-run mode. Before use, be sure to set to the trigger mode by using the cockpit parts which is accompanied with the driver software, and then write the setting into EEPROM. For details, see Chapter 3.

This provider verifies the operation of the following cameras and IDS Software Suite versions. If the versions are different, the provider may not function properly.

IDS Software Suite 4.50

2. Outline of provider

2.1. Outline

The uEye provider is mounted CaoController::Execute as a command execution method. Each Execute command executes a function of uEye SDK

Please install IDS driver if you use this uEye provider with PC. In that case, please do not install the filter of DirectShow.

2.2. Method and Property

2.2.1. CaoWorkspace::AddController method

In uEye provider, camera is selected by means of is_InitCamera function at AddController execution. In this timing, an error 0x8011000 returns if the trigger mode of the camera is free-run mode.

Syntax AddController(<bstrCtrlName:VT_BSTR>,<bstrProvName:VT_BSTR>,
<bstrPcName:VT_BSTR > [,<bstrOption:VT_BSTR>])

bstrCtrlName : [in] Controller name (Arbitrary)
bstrProvName : [in] Provider name (Fixed value =”CaoProv.IDS.uEye”)
bstrPcName : [in] Execution machine name of provider
bstrOption : [in] Option character string

The table below is a list of option character strings.

Table 2-1 Option character string of CaoWorkspace::AddController

Option	Meaning
hCam[=<ID>]	Specify either Camera ID or device ID. The first camera is selected when it is omitted or 0 is specified.
Mono=<True/False>	Specifies whether the image is acquired in monochrome or not. If True, the image is acquired in monochrome. When False is specified, the image is acquired in color if the sensor supports color. The default value is false.

Example

```
Dim caoEng as CaoEngine
Dim IDSCam as CaoController

Set caoEng = New CaoEngine
Set IDSCam = caoEng.Workspaces(0).AddController("IDS", "caoProv.IDS.uEye", "", "")
```

2.2.2. CaoController::Execute method

Send and receive commands of the native mode. Specify a command name in the first argument, parameters of the command in the second argument. Please refer to 2.4 Command reference for details of each command.

Syntax Execute (<bstrCommandName:VT_BSTR>,[<vntParam : VT_VARIANT>])
bstrCommandName: [in] Command name
vntParam : [in] Parameter

If an error occurs at Execute method execution, uEyeSDK returns HRESULT. Please refer to Chapter 2.3 for the error code.

2.2.3. CaoController::AddVariable method

uEye provider does not mount variable class.

2.3. Error code

In uEye provider, you can acquire an error code of uEyeSDK as follows.
However, IS_NO_SUCCESS is returned as 0x80100000.
Error code of 0x80100000 | uEye SDK
Example: IS_INVALID_PARAMETER (125)
0x80100000 | 0x7D = 8010007D
Example: IS_NO_SUCCESS (-1)
0x80100000

In addition, the following error code is defined. Please refer to the chapter of the error code of "ORiN2 programming guide" for the ORiN2 commonness error.

Table 2-2 Original error code list

Error name	Error number	Explanation
E_UEYE_PROV_ERROR	0x80110000 – 0x8011FFFF	Original error of each command

2.4. Command reference

This chapter explains each command of the CaoController::Execute method. Please refer to the explanation of the function in the manual of IDS for detailed operation of each command.

Table 2-3 CaoController::Execute command list

IDS Function	Command	Function	
Image acquisition			
is_FreezeVideo	FreezeVideo	Acquire one image from the camera	P. 9
is_GetImageInfo is_GetImageMemPitch is_GetImageMem is_CopyImageMem	GetImage	Acquire image data from the image memory	P. 9
Parameter setting			
Is_AOI	GetImageAOI	Acquire the size and position of AOI (Are Of Interest) in the image	P. 10
	SetImageAOI	Set the size and position of AOI (Area Of Interest) in the image	P. 10
is_EdgeEnhancement	GetEdgeEnhancementRange	Acquire available setting range of the edge emphasis	P. 11
	GetEdgeEnhancement	Acquire currently configured setting value of the edge emphasis	P. 11
	SetEdgeEnhancement	Set the value of the edge emphasis	P. 11
is_Exposure	GetExposureRange	Acquire available setting range of the exposure time	P. 12
	GetExposure	Acquire currently configured setting value of the exposure time	P. 12
	SetExposure	Set the value of the exposure	P. 12
is_SetGamma	SetGamma	Set gamma correction value	P. 13
	GetGamma	Acquire gamma correction value	P. 13
is_SetTriggerDelay	SetTriggerDelay	Set the trigger delay time	P. 13
	GetTriggerDelay	Acquire the trigger delay time	P. 14
	GetTriggerDelayRange	Acquire available setting range of the trigger delay time	P. 14
is_SetHardwareGain	SetHardwareGain	Set the gain channel of the sensor	P. 15
	GetHardwareGain	Acquire the gain channel of the sensor	P. 15
is_SetSaturation	SetSaturation	Set the software color saturation	P. 15
	GetSaturation	Acquire the software color saturation	P. 16
Information acquisition			
is_GetCameraInfo	GetCameraInfo	Read out the data hard-corded in EEPROM	P. 16
is_GetDLLVersion	GetDLLVersion	Return the version of ueye_api.dll	P. 17

is_GetError	GetError	Request an error that occurred immediately before and then return the related error code and message	P. 17
Is_GetImageInfo	GetImageInfoEx	Acquire additional information of image captured	P. 18
Is_GetSensorInfo	GetSensorInfo	Acquire information about the sensor mounted in the camera	P. 18

2.4.1. CaoController::Execute("FreezeVideo") command

Acquire one image from the camera, and then send it to the image memory

Syntax FreezeVideo ([<Wait>])

Argument : [in] Wait: Time-out value of image capture (VT_I4)
0 : IS_DONT_WAIT
1 : IS_WAIT (default)
Time t: Timeout period

Return value : [out] None

When the software trigger is specified for the trigger mode, this command initiates a trigger.

The transmitted image memory can be acquired by GetImage function.

Example

```
Dim bmp() as Byte

caoCtrl.Execute "FreezeVideo"
bmp = caoCtrl.Execute("GetImage")

bmp : Image
```

2.4.2. CaoController::Execute("GetImage") command

Acquire image data from the image memory, and then return the data by DIB form.

Syntax GetImage ()

Argument : [in] None

Return value : [out] Image data (VT_UI1 | VT_ARRAY)
Bit map

Acquire image information with is_GetImageInfo and is_GetImageMemPitch, and then secure the memory for DIB.

Then, acquire the pointers for image memory by is_GetImageMem, and then copy the image data and paste it to the memory for DIB by is_CopyImageMem.

Example

```
Dim bmp() as Byte

caoCtrl.Execute "FreezeVideo"
bmp = caoCtrl.Execute("GetImage")

bmp : image
```

2.4.3. CaoController::Execute(“GetImageAOI”) command

Acquire the size and position of AOI (Area Of Interest) in the image

Syntax GetImageAOI ()

< bstrSyntax > : [in] None

Return value : [out] AOI (VT_UI4 | VT_ARRAY) (X, Y, Width, Height)

X: X-coordinates of AOI

Y: Y-coordinates of AOI

Width: Width of AOI

Height: Height of AOI

Example

```
Dim vntResult as Variant
vntResult = caoCtrl.Execute("GetImageAOI")

vntResult : 0, 0, 640, 480
```

2.4.4. CaoController::Execute(“SetImageAOI”) command

Set the size and position of AOI (Area Of Interest) in the image

Syntax SetImageAOI (X, Y, Width, Height)

< bstrSyntax > : [in] AOI (VT_UI4 | VT_ARRAY) (X, Y, Width, Height)

X: X-coordinates of AOI

Y: Y-coordinates of AOI

Width: Width of AOI

Height: Height of AOI

Return value : [out] None

By performing logical OR of X and Y in hexadecimal, this command specifies the absolute position of AOI and output image.

Example

```
caoCtrl.Execute("SetImageAOI", Array(100,100,320,240))
```

2.4.5. CaoController::Execute ("GetEdgeEnhancementRange") command

Return available setting range of the edge emphasis

Syntax GetEdgeEnhancementRange ()

Argument : [in] None

Return value : [out] Range (VT_UI4 | VT_ARRAY) (Min, Max, Inc)

Min: Minimum value

Max: The maximum value

Inc: Increment

Example

```
Dim vntRet as Variant
vntRet = Ctrl.Execute("GetEdgeEnhancementRange")

vntRet : 0, 9, 1
```

2.4.6. CaoController::Execute ("GetEdgeEnhancement") command

Return currently configured setting value of the edge emphasis

Syntax GetEdgeEnhancement()

Argument : [in] None

Return value : [out] Value of the currently configured edge emphasis (VT_UI4)

Example

```
Dim lParam as long
lParam  = caoCtrl.Execute("GetEdgeEnhancement")
```

2.4.7. CaoController::Execute ("SetEdgeEnhancement") command

Set the value of the edge emphasis

Syntax SetEdgeEnhancement (< ulEdge>)

< ulEdge > : [in] Value to be set(VT_UI4)

Return value : [out] None

The range of available setting value can be confirmed by GetEdgeEnhancementRange command.

Example

```
caoCtrl.Execute "SetEdgeEnhancement", 3
```

2.4.8. CaoController::Execute ("GetExposureRange") command

Return available setting range of the exposure time (millisecond).

Syntax GetExposureRange ()

Argument : [in] None

Return value : [out] Exposure setting range (VT_R8 | VT_ARRAY) (Min, Max, Inc)

Min: Minimum value

Max: The maximum value

Inc: Increment

Example

```
Dim vntParams as Variant
vntParams = caoCtrl.Execute("GetExposureRange")

vntParams : 0.043668010001,26.977273250525,0.051400010001
```

2.4.9. CaoController::Execute ("GetExposure") command

Return currently configured setting value of the exposure time (millisecond).

Syntax GetExposure ()

Argument : [in] None

Return value : [out] Exposure time (VT_R8)

Example

```
Dim dExposure as Double
dExposure = caoCtrl.Execute("GetExposure ")
```

2.4.10. CaoController::Execute ("SetExposure") command

Set the value of the exposure time (millisecond).

Syntax SetExposure (<dblExposure>)

<dblExposure> : [in] Exposure time (VT_R8)

Return value : [out] Exposure time actually set (VT_R8)

Example

```
Dim dExeposure as Double
dExposure = caoCtrl.Execute("SetExposure", 12. 12.23453)

dExposure : 12.225470380238
```

2.4.11. CaoController::Execute ("SetGamma") command

Set digital gamma correction value which applies a gamma characteristic to the image

Syntax SetGamma (<lGamma>)

<lGamma> : [in] Gamma correction value (VT_I4)

Return value : None

Example

```
caoCtrl.Execute "SetGamma", 90
```

2.4.12. CaoController::Execute ("GetGamma") command

Return digital gamma correction value which applied a gamma characteristic to the image

Syntax GetGamma ()

Argument : [in] None

Return value : [out] Gamma correction value (VT_I4)

Example

```
Dim lGamma as Long

lGamma = caoCtrl.Execute("GetGamma")
```

2.4.13. CaoController::Execute ("SetTriggerDelay") command

Set the delay time between the arrival of a trigger signal and the start of exposure[1]

Syntax SetTriggerDelay (<lDelay>)

[1] Cannot specify 32,768(0x8000) to 32,771(0x8003).

<lDelay> : [in] Delay time

Return value : None

Example

```
caoCtrl.Execute "SetTriggerDelay", 20
```

2.4.14. CaoController::Execute ("GetTriggerDelay") command

Acquire the delay time between the arrival of a trigger signal and the start of exposure

Syntax GetTriggerDelay ()

Argument : [in] None

Return value : [out] Delay time

Example

```
Dim lDelay as Long
lDelay  = caoCtrl.Execute("GetTriggerDelay")
```

2.4.15. CaoController::Execute ("GetTriggerDelayRange") command

Acquire currently set delay time between the arrival of a trigger signal and the start of exposure

Syntax GetTriggerDelayRange ()

Argument : [in] None

Return value : [out] Setting range of delay time (VT_I4 | VT_ARRAY)

(Min, Max, Granularity)

Min: The minimum adjustable value

Max: The maximum adjustable value

Granularity: Resolution of the adjustable delay time

Example

```
Dim vntDelay as Variant
vntDelay = caoCtrl.Execute("GetTriggerDelayRange")
vntDealy : 15, 4000000, 1
```

2.4.16. CaoController::Execute ("SetHardwareGain") command

Set the gain channel of the sensor

Syntax SetHardwareGain (<Master>, <Red>, <Green>, <Blue>)

Argument : [in] Gain factor (VT_I4 | VT_ARRAY)
Master: The entire gain factor
Red: Gain factor of red channel
Green: Gain factor of green channel
Blue: Gain factor of blue channel

Return value : [out] None

To use the values being set, specify IS_IGNORE_PARAMETER(-1).

Example

```
caoCtrl.Execute  "SetHardwareGain", Array (39, 10, -1, 10)
```

2.4.17. CaoController::Execute ("GetHardwareGain") command

Acquire the gain channel of the sensor

Syntax SetHardwareGain ()

Argument : [in] None

Return value : [out] Gain factor (VT_I4 | VT_ARRAY)
Master: The entire gain factor
Red: Gain factor of red channel
Green: Gain factor of green channel
Blue: Gain factor of blue channel

Example

```
Dim vntRet as Variant
vntRet  = caoCtrl.Execute("SetHardwareGain")
vntRet : 38, 0, 0, 0
```

2.4.18. CaoController::Execute ("SetSaturation") command

Set the software color saturation

Syntax SetSaturation (< ChromU>, < ChromV>)

Argument : [in] saturation (VT_UI4 | VT_ARRAY) (ChromU, ChromV)

ChromU: U saturation: value multiplied by 100

ChromV: V saturation : value multiplied by 100

Return value : [out] None

Example

```
caoCtrl.Execute "SetSaturation", Array(100, 50)
```

2.4.19. CaoController::Execute ("GetSaturation") command

Acquire the software color saturation

Syntax GetSaturation ()

Argument : [in] None

Return value : [out] Saturation (VT_UI4 | VT_ARRAY) (ChromU, ChromV)

ChromU: U saturation: Value multiplied by100

ChromV: V saturation : Value multiplied by 100

Example

```
Dim vntRet as Variant
vntRet = caoCtrl.Execute("GetSaturation")
vntRet:  100, 100
```

2.4.20. CaoController::Execute ("GetCameraInfo") command

Read out the data hard-coded in the EEPROM

Syntax GetCameraInfo ( )

Argument : [in] None

Return value : [out] CAMINFO (VT_VARIANT | VT_ARRAY)

(SerNo, ID, Version, Date, Select, Type)

SerNo: Serial number of camera (VT_BSTR)

ID: Manufacturer of the camera (VT_BSTR)

Version: Hardware version (VT_BSTR)

Date: System date of the final quality inspection (VT_BSTR)

Select: Camera ID (VT_UI1)

Type: Camera type (VT_UI1)

Example

```
Dim vntRet as Variant
vntRet = caoCtrl.Execute("GetCameraInfo")

vntRet : "4102633099", "IDS GmbH", "V14.1", "11.06.2013", 1, 64
```

2.4.21. CaoController::Execute ("GetDLLVersion") command

Return the version of uEye_api.dll

Syntax GetDLLVersion ( )

Argument : [in] None

Return value : [out] Version (VT_I4 | VT_ARRAY) (major, minor, build)

major: Majors version

minor: Minor version

build: Build version

Example

```
Dim vntRet as Variant
vntRet = caoCtrl.Execute("GetDLLVersion")

vntRet : 4, 22, 11
```

2.4.22. CaoController::Execute ("GetError") command

Request an error that occurred immediately before and then return the related error code and message

Syntax GetError ( )

Argument : [in] None

Return value : [out] Error (VT_VARIANT | VT_ARRAY) (Errorcode, Error)

Error Code: Error code

Error: Content of error

Example

```
Dim vntRet as Variant
vntRet = caoCtrl.Execute("GetDLLVersion")

vntRet : 0, "No error"
```

2.4.23. CaoController::Execute ("GetImageInfoEx") command

Acquire additional information of the captured image

Syntax GetImageInfoEx ( )

Argument : [in] None

Return value : [out] Image Info(VT_VARIANT | VT_ARRAY) (Time, IoStatus)

Time : Time stamp at the time of image capture (VT_UI2 | VT_ARRAY)

(year, month, day, hour, minute, second, milliseconds)

IoStatus : State of digital I/O at the time of image capture (VT_UI4)

Return the information of is_GetImageInfo execution at the timing of GetImage. When GetImage is not executed, an error occurs.

Example

```
Dim vntRet as Variant
vntRet = caoCtrl.Execute("GetImageInfoEx")

vntRet : Array(2013, 7, 22, 14, 45, 23, 757), 0
```

2.4.24. CaoController::Execute ("GetSensorInfo") command

Acquire information on the sensor installed in the camera

Syntax GetSensorInfo ( )

Argument : [in] None

Return value : [out] Image Info(VT_VARIANT | VT_ARRAY)

(SensorID, SensorName, ColorMode, MaxWidth, MaxHeight, MasterGain, RGain, GGain, BGain, GlobShutter, PixelSize)

SensorID: Return the sensor type (VT_UI2)

SensorName: Return the camera model (VT_BSTR)

ColorMode: Return the sensor color mode (VT_UI1)

MaxWidth: Return the maximum image width (VT_UI4)

MaxHeight: Return the maximum image height (VT_UI4)

MasterGain: Indicate whether the sensor provides analog mastering gain (VT_BOOL)

RGain: Indicate whether the sensor provides analog red channel gain (VT_BOOL)

GGain: Indicate whether the sensor provides analog green channel

gain (VT_BOOL)

BGain: Indicate whether the sensor provides analog blue channel gain (VT_BOOL)

GlobShutter: Indicate whether the sensor has a global shutter (VT_BOOL)

PixelSize: Return the pixel size in µm (VT_UI2).

Example

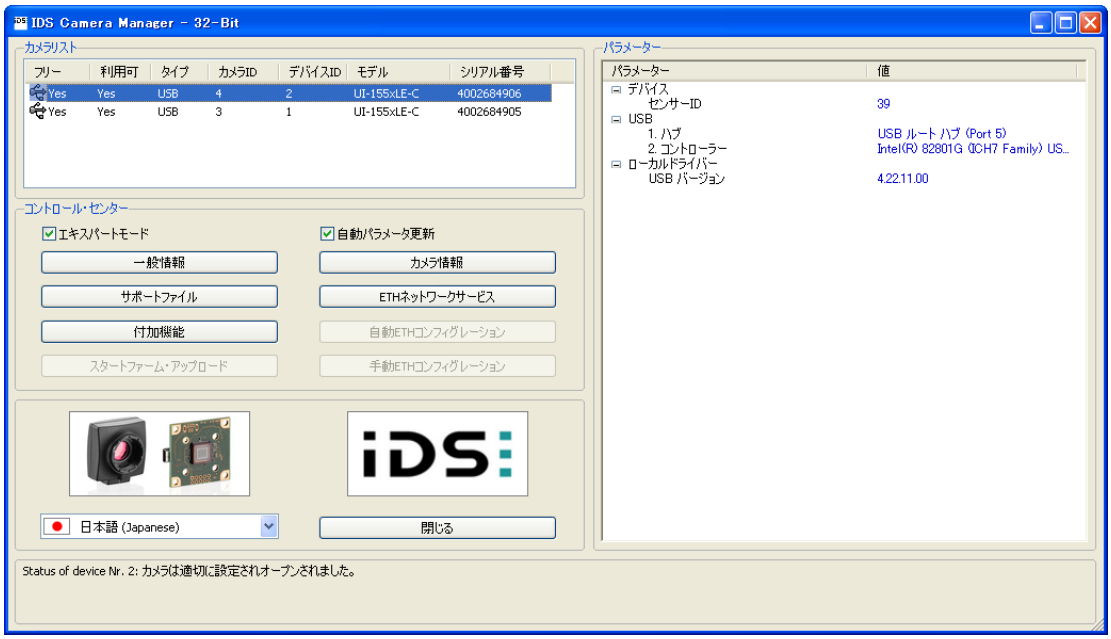
```
Dim vntRet as Variant
vntRet = caoCtrl.Execute("GetSensorInfo")

vntRet : 11, "UI148xSE-M", 1, 2560, 1920, 1, 0, 0, 0, 0, 220
```

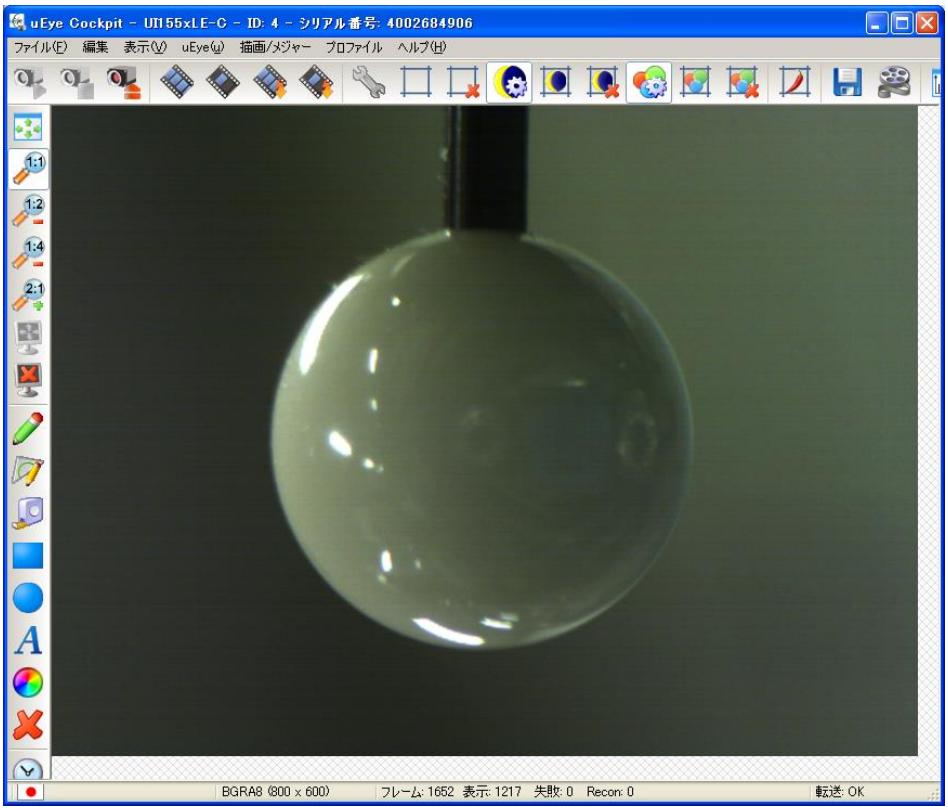
3. Example of setting camera

3.1. When you use the software trigger

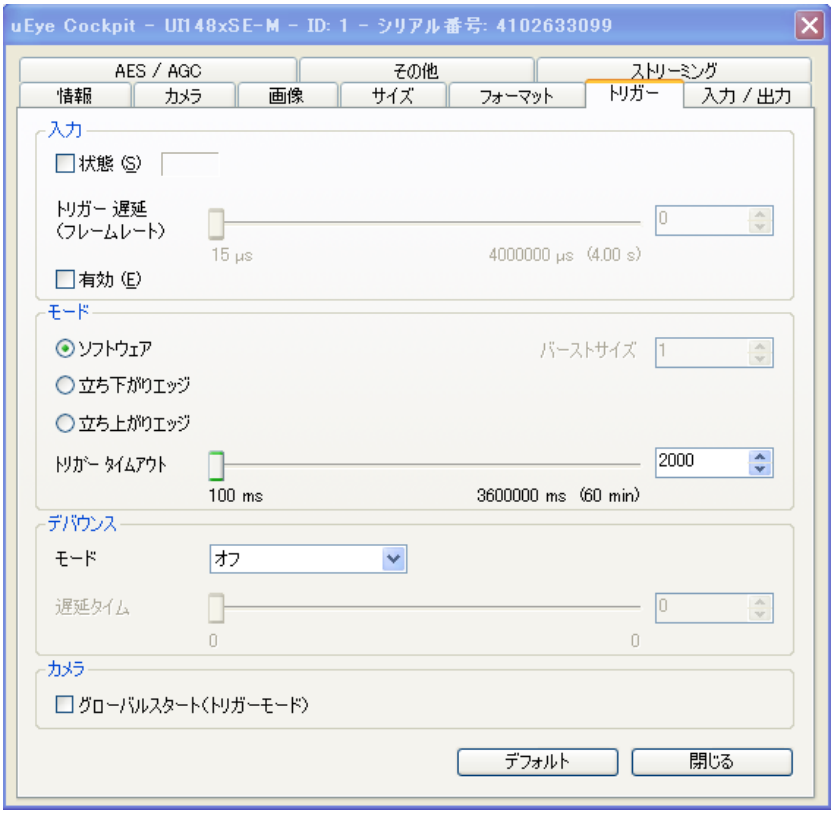
Start the "IDS Camera Manager"



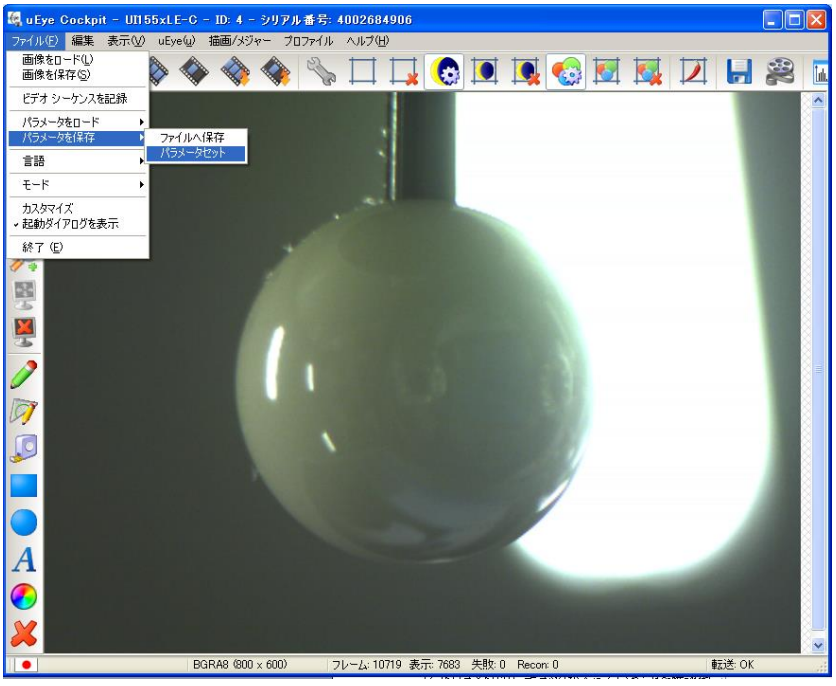
Select the camera that you intend to set, and then start "uEye Cockpit"



In the initial setting, the images are captured continuously. To change the setting, on the Toolbar, click [Start/Stop continuous trigger capture] button. Once the dialog is displayed, select [Software] from the [Mode] area.



Check that the checkbox of the software trigger is selected, and then save it to the parameter set



To capture an image, initiate a software trigger by FreezeVideo command and then capture the image by GetImage command.

```
Dim bmp() As Byte
Set caoEng = New CaoEngine
Set caoCtrlIDS = caoEng.Workspaces(0).AddController("IDS", "caoProv.IDS.uEye", "", "")
Call caoCtrlIDS.Execute("FreezeVideo", 1)
bmp = caoCtrlIDS.Execute("GetImage")
```