

uEye プロバイダ IDS カメラ用プロバイダ

Version 1.0.2

ユーザーズ ガイド

May 15, 2020

【備考】

【改版履歴】

バージョン	日付	内容
1.0.0	2013-7-24	初版.
1.0.1	2015-8-3	IDS uEye Ver 4.50 に対応.
1.0.2	2020-5-15	GetEdgeEnhancementRange の誤記修正. SetGamma, GetGamma, SetTriggerDelay の処理修正.
	2021-10-28	AddController のオプションに Mono を追記.

目次

1. はじめに.....	5
2. プロバイダの概要.....	6
2.1. 概要.....	6
2.2. メソッド・プロパティ.....	6
2.2.1. CaoWorkspace::AddController メソッド.....	6
2.2.2. CaoController::Execute メソッド.....	7
2.2.3. CaoController::AddVariable メソッド.....	7
2.3. エラーコード.....	7
2.4. コマンドリファレンス.....	8
2.4.1. CaoController::Execute(“FreezeVideo”) コマンド.....	9
2.4.2. CaoController::Execute(“GetImage”) コマンド.....	9
2.4.3. CaoController::Execute(“GetImageAOI”) コマンド.....	10
2.4.4. CaoController::Execute(“SetImageAOI”) コマンド.....	10
2.4.5. CaoController::Execute(“GetEdgeEnhancementRange”) コマンド.....	10
2.4.6. CaoController::Execute(“GetEdgeEnhancement”) コマンド.....	11
2.4.7. CaoController::Execute(“SetEdgeEnhancement”) コマンド.....	11
2.4.8. CaoController::Execute(“GetExposureRange”) コマンド.....	12
2.4.9. CaoController::Execute(“GetExposure”) コマンド.....	12
2.4.10. CaoController::Execute(“SetExposure”) コマンド.....	12
2.4.11. CaoController::Execute(“SetGamma”) コマンド.....	13
2.4.12. CaoController::Execute(“GetGamma”) コマンド.....	13
2.4.13. CaoController::Execute(“SetTriggerDelay”) コマンド.....	13
2.4.14. CaoController::Execute(“GetTriggerDelay”) コマンド.....	14
2.4.15. CaoController::Execute(“GetTriggerDelayRange”) コマンド.....	14
2.4.16. CaoController::Execute(“SetHardwareGain”) コマンド.....	14
2.4.17. CaoController::Execute(“GetHardwareGain”) コマンド.....	15
2.4.18. CaoController::Execute(“SetSaturation”) コマンド.....	15
2.4.19. CaoController::Execute(“GetSaturation”) コマンド.....	16
2.4.20. CaoController::Execute(“GetCameraInfo”) コマンド.....	16
2.4.21. CaoController::Execute(“GetDLLVersion”) コマンド.....	17
2.4.22. CaoController::Execute(“GetError”) コマンド.....	17
2.4.23. CaoController::Execute(“GetImageInfoEx”) コマンド.....	17
2.4.24. CaoController::Execute(“GetSensorInfo”) コマンド.....	18

3. カメラの設定例	20
3.1. ソフトウェアトリガを使用する場合	20

1. はじめに

本書は IDS 社製 USB カメラの CAO プロバイダである, uEye プロバイダのユーザーズガイドです.

uEye プロバイダは USB 接続されたカメラを uEyeSDK のコマンドで接続し, カメラの制御や画像の取得を行います. 一部機種固有の関数は実装していません. 動作確認は以下のカメラで行っています.

UI-1480SE-M-GL

本プロバイダではフリーランモードはサポートしておりません. 必ずドライバソフトに付属しているコックピットパーツを用いてトリガモードに設定後, EEPROM に設定を書き込んでから使用してください. 詳しくは第 3 章を参照してください.

本プロバイダでは以下のカメラと IDS Software Suite バージョンについて動作確認をしています. バージョンが異なる場合, プロバイダが正しく機能しない可能性があります.

IDS Software Suite 4.50

2. プロバイダの概要

2.1. 概要

uEye プロバイダは、コマンドの実行方法として `CaoController::Execute` を実装しています。各 `Execute` コマンドでは uEye SDK の関数を実行しています。

PC で使用する場合には、IDS のドライバをインストールしてください。その際に `DirectShow` のフィルターはインストールしないでください。

2.2. メソッド・プロパティ

2.2.1. `CaoWorkspace::AddController` メソッド

uEye プロバイダでは `AddController` 時に、`is_InitCamera` 関数を使用しカメラの選択を行います。このときにカメラのトリガがフリーランモードの場合、`0x80110000` のエラーが返ります。

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

`bstrCtrlName` : [in] コントローラ名 任意
`bstrProvName` : [in] プロバイダ名 固定値 =”CaoProv.IDS.uEye”
`bstrPcName` : [in] プロバイダの実行マシン名
`bstrOption` : [in] オプション文字列

以下にオプション文字列に指定するリストを示します。

表 2-1 `CaoWorkspace::AddController` のオプション文字列

オプション	意味
<code>hCam[=<ID>]</code>	カメラ ID かデバイス ID を指定します。 省略または 0 を指定すると、先頭のカメラが選択されます。
<code>Mono=<True/False></code>	画像をモノクロで取得するかどうかを指定します。 True を指定するとモノクロで画像を取得します。 False を指定するとセンサがカラー対応の場合はカラーで画像を取得します。 省略時は False となります。

使用例

```
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. GaoController::Execute メソッド

ネイティブモードのコマンドの送受信を行います。第 1 引数にコマンド名、第 2 引数にコマンドのパラメータを指定します。各コマンドの詳細は 2.4 章コマンドリファレンスを参照してください。

書式 Execute (<bstrCommandName:VT_BSTR>,[<vntParam : VT_VARIANT>])

bstrCommandName: [in] コマンド名

vntParam : [in] パラメータ

Execute メソッド実行した際にエラーが発生した場合、uEyeSDK からの戻り値は HRESULT として返されます。エラーコードについては、2.3 章を参照してください。

2.2.3. GaoController::AddVariable メソッド

uEye プロバイダでは変数クラスを実装していません。

2.3. エラーコード

uEye プロバイダでは、uEyeSDK のエラーコードが下記のように取得できます。

ただし、IS_NO_SUCCESS は 0x80100000 で返ります。

0x80100000 | uEye SDK のエラーコード

例: IS_INVALID_PARAMETER (125)

0x80100000 | 0x7D = 8010007D

例: IS_NO_SUCCESS (-1)

0x80100000

また、下記エラーコードが定義されています。ORiN2 共通エラーについては、「ORiN2 プログラミングガイド」のエラーコードの章を参照してください。

表 2-2 独自エラーコード一覧

エラー名	エラー番号	説明
E_UEYE_PROV_ERROR	0x80110000 ~ 0x8011FFFF	各コマンド独自エラー

2.4. コマンドリファレンス

本章では CaoController::Execute メソッドの各コマンドについて解説します。各コマンドの詳細動作については IDS 社のマニュアルの関数についての説明を参照してください。

表 2-3 CaoController::Execute コマンド一覧

IDS 関数	コマンド	機能	
画像取得			
is_FreezeVideo	FreezeVideo	カメラから一個の画像を取得します	P9
is_GetImageInfo is_GetImageMemPitch is_GetImageMem is_CopyImageMem	GetImage	画像メモリから画像データを取得します。	P9
パラメータ設定			
Is_AOI	GetImageAOI	画像内の AOI (解析対象領域) のサイズと位置を取得します。	P10
	SetImageAOI	画像内の AOI (解析対象領域) のサイズと位置を設定します。	P10
is_EdgeEnhancement	GetEdgeEnhancementRange	エッジ強調の設定可能範囲を取得します。	P10
	GetEdgeEnhancement	設定されているエッジ強調の値を取得します。	P11
	SetEdgeEnhancement	エッジ強調の値を設定します。	P11
is_Exposure	GetExposureRange	露出の設定可能範囲を取得します。	P12
	GetExposure	露出の設定値を取得します。	P12
	SetExposure	露出の値を設定します。	P12
is_SetGamma	SetGamma	ガンマ補正値を設定します。	P13
	GetGamma	ガンマ補正値を取得します。	P13
is_SetTriggerDelay	SetTriggerDelay	トリガ遅延時間を設定します。	P13
	GetTriggerDelay	トリガ遅延時間を取得します。	P14
	GetTriggerDelayRange	トリガ遅延時間の設定可能範囲を取得します。	P14
is_SetHardwareGain	SetHardwareGain	センサーのゲインチャンネルを設定します。	P14
	GetHardwareGain	センサーのゲインチャンネルを取得します。	P15
is_SetSaturation	SetSaturation	ソフトウェアの色の彩度を設定します。	P15
	GetSaturation	ソフトウェアの色の彩度を取得します。	P16
情報取得			
is_GetCameraInfo	GetCameraInfo	EEPROM 内のデータを読み出します。	P16
is_GetDLLVersion	GetDLLVersion	ueye_api.dll のバージョンを返します。	P17
is_GetError	GetError	直近で発生したエラーを問い合わせ、関連	P17

		するエラーコードとメッセージを返します。	
Is_GetImageInfo	GetImageInfoEx	撮影した画像の追加情報を取得します。	P17
Is_GetSensorInfo	GetSensorInfo	カメラに搭載しているセンサーに関する情報を取得します。	P18

2.4.1. CaoController::Execute(“FreezeVideo”) コマンド

カメラから一個の画像を取得し、画像メモリに転送します。

書式 FreezeVideo ([<Wait>])

引数 : [in] Wait : 画像キャプチャのタイムアウト値 (VT_I4)
 0 : IS_DONT_WAIT
 1 : IS_WAIT (default)
 時間 t : タイムアウト時間

戻り値 : [out] なし

トリガモードをソフトウェアトリガに指定した場合、このコマンドでトリガをかけます。

転送された画像メモリは GetImage 関数で取得できます。

使用例

```
Dim bmp() as Byte
caoCtrl.Execute "FreezeVideo"
bmp = caoCtrl.Execute("GetImage")

bmp : 画像
```

2.4.2. CaoController::Execute(“GetImage”) コマンド

画像メモリから画像データを取得し、DIB 形式にして返します。

書式 GetImage ()

引数 : [in] なし
 戻り値 : [out] 画像データ (VT_UI1 | VT_ARRAY)
 ビットマップ

is_GetImageInfo と is_GetImageMemPitch で画像情報を取得し、DIB 用のメモリを確保します。

その後 is_GetImageMem で画像メモリへのポインタを取得後、is_CopyImageMem で DIB 用メモリへ画像データをコピーして返します。

使用例

```
Dim bmp() as Byte
caoCtrl.Execute "FreezeVideo"
bmp = caoCtrl.Execute("GetImage")

bmp : 画像
```

2.4.3. CaoController::Execute(“GetImageAOI”) コマンド

画像内の AOI (解析対象領域) のサイズと位置を取得します。

書式

GetImageAOI ()

< bstrSyntax > : [in] なし
戻り値 : [out] AOI (VT_UI4 | VT_ARRAY) (X, Y, Width, Height)
X : AOI の X 位置
Y : AOI の Y 座標
Width: AOI の幅
Height: AOI の高さ

使用例

```
Dim vntResult as Variant  
vntResult = caoCtrl.Execute("GetImageAOI")  
  
vntResult : 0, 0, 640, 480
```

2.4.4. CaoController::Execute(“SetImageAOI”) コマンド

画像内の AOI (解析対象領域) のサイズと位置を設定します。

書式

SetImageAOI (X, Y, Width, Height)

< bstrSyntax > : [in] AOI (VT_UI4 | VT_ARRAY) (X, Y, Width, Height)
X : AOI の X 位置
Y : AOI の Y 座標
Width: AOI の幅
Height: AOI の高さ
戻り値 : [out] なし

X,Y に 0x10000000 を論理和取ることで、メモリの絶対位置指定をすることができます。

使用例

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

2.4.5. CaoController::Execute(“GetEdgeEnhancementRange”) コマンド

エッジの強調範囲を返します。

書式

GetEdgeEnhancementRange ()

引数 : [in] なし
戻り値 : [out] 範囲(VT_UI4 | VT_ARRAY) (Min, Max, Inc)
Min: 最小値
Max: 最大値
Inc: インクリメント

使用例

```
Dim vntRet as Variant  
vntRet = Ctrl.Execute("GetEdgeEnhancementRange")  
  
vntRet : 0, 9, 1
```

2.4.6. CaoController::Execute ("GetEdgeEnhancement") コマンド

現在設定されているエッジ強調を返します。

書式

GetEdgeEnhancement()

引数 : [in] なし
戻り値 : [out] 設定されているエッジ強調の値 (VT_UI4)

使用例

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

2.4.7. CaoController::Execute ("SetEdgeEnhancement") コマンド

エッジ強調を設定します。

書式

SetEdgeEnhancement (< ulEdge >)

< ulEdge > : [in] 設定したい値(VT_UI4)
戻り値 : [out] なし

設定できる値の範囲は GetEdgeEnhancementRange コマンドで確認できます。

使用例

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

2.4.8. CaoController::Execute (“GetExposureRange”) コマンド

露出時間範囲を返します (ミリ秒).

書式

GetExposureRange ()

引数 : [in] なし

戻り値 : [out] 露出設定範囲(VT_R8 | VT_ARRAY) (Min, Max, Inc)

Min : 最小値

Max : 最大値

Inc : インクリメント

使用例

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

vntParams : 0.043668010001, 26.977273250525, 0.051400010001
```

2.4.9. CaoController::Execute (“GetExposure”) コマンド

現在の露出時間設定を返します (ミリ秒).

書式

GetExposure ()

引数 : [in] なし

戻り値 : [out] 露出時間 (VT_R8)

使用例

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

2.4.10. CaoController::Execute (“SetExposure”) コマンド

露出時間設定を設定します (ミリ秒).

書式

SetExposure (<dblExposure>)

<dblExposure> : [in] 露出時間 (VT_R8)

戻り値 : [out] 実際に設定された時間 (VT_R8)

使用例

```
Dim dExposure as Double
dExposure = caoCtrl.Execute("SetExposure", 12.12.23453)
```

dExposure : 12.225470380238

2.4.11. CaoController::Execute (“SetGamma”) コマンド

画像に適用されるガンマ特性の、デジタルガンマ補正値を設定します。

書式 SetGamma (<IGamma>)

<IGamma> : [in] ガンマ補正値 (VT_I4)
戻り値 : なし

使用例

```
caoCtrl.Execute “SetGamma”, 90
```

2.4.12. CaoController::Execute (“GetGamma”) コマンド

画像に適用されるガンマ特性の、デジタルガンマ補正値を取得します。

書式 GetGamma ()

引数 : [in] なし
戻り値 : [out] ガンマ補正値 (VT_I4)

使用例

```
Dim IGamma as Long  
IGamma = caoCtrl.Execute (“GetGamma”)
```

2.4.13. CaoController::Execute (“SetTriggerDelay”) コマンド

トリガ信号の到着と露出開始までの間の遅延時間を設定します⁽¹⁾。

書式 SetTriggerDelay (<IDelay>)

<IDelay> : [in] 遅延時間
戻り値 : なし

使用例

```
caoCtrl.Execute “SetTriggerDelay”, 20
```

¹ 32,768(0x8000)～32,771(0x8003)を指定することはできません。

2.4.14. GaoController::Execute (“GetTriggerDelay”) コマンド

トリガ信号の到着と露出開始までの間の遅延時間を取得します。

書式 GetTriggerDelay ()

引数 : [in] なし
戻り値 : [out] 遅延時間

使用例

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

2.4.15. GaoController::Execute (“GetTriggerDelayRange”) コマンド

トリガ信号の到着と露出開始までの間の遅延時間の設定範囲を取得します。

書式 GetTriggerDelayRange ()

引数 : [in] なし
戻り値 : [out] 遅延時間の設定範囲 (VT_I4 | VT_ARRAY)
 (Min, Max, Granularity)
Min : 調節可能な最小値
Max : 調節可能な最大値
Granularity : 調整可能な遅延時間の解像度

使用例

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

2.4.16. GaoController::Execute (“SetHardwareGain”) コマンド

センサーのゲインチャンネルを設定します。

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

引数 : [in] ゲイン係数 (VT_I4 | VT_ARRAY)
 Master : 全体のゲイン係数
 Red : レッドチャンネルのゲイン係数

Green : グリーンチャンネルのゲイン係数

Blue : ブルーチャンネルのゲイン係数

戻り値 : [out] なし

設定されている値を使用する場合は, IS_IGNORE_PARAMETER(-1)を指定してください.

使用例

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

2.4.17. CaoController::Execute ("GetHardwareGain") コマンド

センサーのゲインチャンネルを取得します.

書式

SetHardwareGain ()

引数 : [in] なし

戻り値 : [out] ゲイン係数 (VT_I4 | VT_ARRAY)

Master : 全体のゲイン係数

Red : レッドチャンネルのゲイン係数

Green : グリーンチャンネルのゲイン係数

Blue : ブルーチャンネルのゲイン係数

使用例

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

2.4.18. CaoController::Execute ("SetSaturation") コマンド

ソフトウェアの色の彩度を設定します.

書式

SetSaturation (< ChromU>, < ChromV>)

引数 : [in] 彩度 (VT_UI4 | VT_ARRAY) (ChromU, ChromV)

ChromU : U 彩度:100 を乗じた値

ChromV : V 彩度:100 を乗じた値

戻り値 : [out] なし

使用例

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

2.4.19. GaoController::Execute (“GetSaturation”) コマンド

ソフトウェアの色の彩度を取得します。

書式

GetSaturation ()

引数 : [in] なし
戻り値 : [out] 彩度 (VT_UI4 | VT_ARRAY) (ChromU, ChromV)
ChromU : U 彩度:100 を乗じた値
ChromV : V 彩度:100 を乗じた値

使用例

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

2.4.20. GaoController::Execute (“GetCameraInfo”) コマンド

ハードコーディングされた EEPROM 内のデータを読み出します。

書式

GetCameraInfo ()

引数 : [in] なし
戻り値 : [out] CAMINFO (VT_VARIANT | VT_ARRAY)
(SerNo, ID, Version, Date, Select, Type)
SerNo : カメラのシリアル番号 (VT_BSTR)
ID : カメラの製造会社 (VT_BSTR)
Version : ハードウェアバージョン (VT_BSTR)
Date : 最終品質検査のシステム日付 (VT_BSTR)
Select : カメラの ID (VT_UI1)
Type : カメラのタイプ (VT_UI1)

使用例

```
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”) コマンド

ueye_api.dll のバージョンを返します。

書式

GetDLLVersion ()

引数 : [in] なし
戻り値 : [out] Version (VT_I4 | VT_ARRAY) (major, minor, build)
major: メジャーバージョン
minor: マイナーバージョン
build: ビルドバージョン

使用例

```
Dim vntRet as Variant  
vntRet = caoCtrl.Execute("GetDLLVersion")  
  
vntRet : 4, 22, 11
```

2.4.22. CaoController::Execute (“GetError”) コマンド

直近で発生したエラーを問い合わせ、関連するエラーコードとメッセージを返します。

書式

GetError ()

引数 : [in] なし
戻り値 : [out] Error (VT_VARIANT | VT_ARRAY) (Errorcode, Error)

Error Code : エラーコード

Error : エラー内容

使用例

```
Dim vntRet as Variant  
vntRet = caoCtrl.Execute("GetError")  
  
vntRet : 0, "No error"
```

2.4.23. CaoController::Execute (“GetImageInfoEx”) コマンド

撮影した画像の追加情報を取得します。

書式

GetImageInfoEx ()

引数 : [in] なし
戻り値 : [out] Image Info(VT_VARIANT | VT_ARRAY) (Time, IoStatus)

Time : 撮影時間のタイムスタンプ (VT_UI2 | VT_ARRAY)
(year, month, day, hour, minute, second, milliseconds)

IoStatus : 撮像時のデジタル入出力の状態 (VT_UI4)

GetImage 時に is_GetImageInfo を実行した情報を返します。GetImage を実行していない場合はエラーになります。

使用例

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

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

2.4.24. CaoController::Execute ("GetSensorInfo") コマンド

カメラに搭載しているセンサーに関する情報を取得します。

書式

GetSensorInfo ()

引数 : [in] なし

戻り値 : [out] Image Info(VT_VARIANT | VT_ARRAY)

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

SensorID : センサーの種類を返します(VT_UI2)

SensorName : カメラのモデルを返します(VT_BSTR)

ColorMode : センサーのカラーモードを返します(VT_UI1)

MaxWidth : 画像の最大幅を返します(VT_UI4)

MaxHeight : 画像の最大高さを返します(VT_UI4)

MasterGain : センサーがアナログマスターゲインに対応しているかどうかを表示します(VT_BOOL)

RGain : センサーがアナログ レッドチャンネルゲインに対応しているかどうかを表示します(VT_BOOL)

GGain : センサーがアナログ グリーンチャンネルゲインに対応しているかどうかを表示します(VT_BOOL)

BGain : センサーがアナログ ブルーチャンネルゲインに対応しているかどうかを表示します(VT_BOOL)

GlobShutter : センサーがグローバルシャッターを搭載しているかどうかを表示します(VT_BOOL)

PixelSize : ピクセルサイズを μm で返します(VT_UI2)

使用例

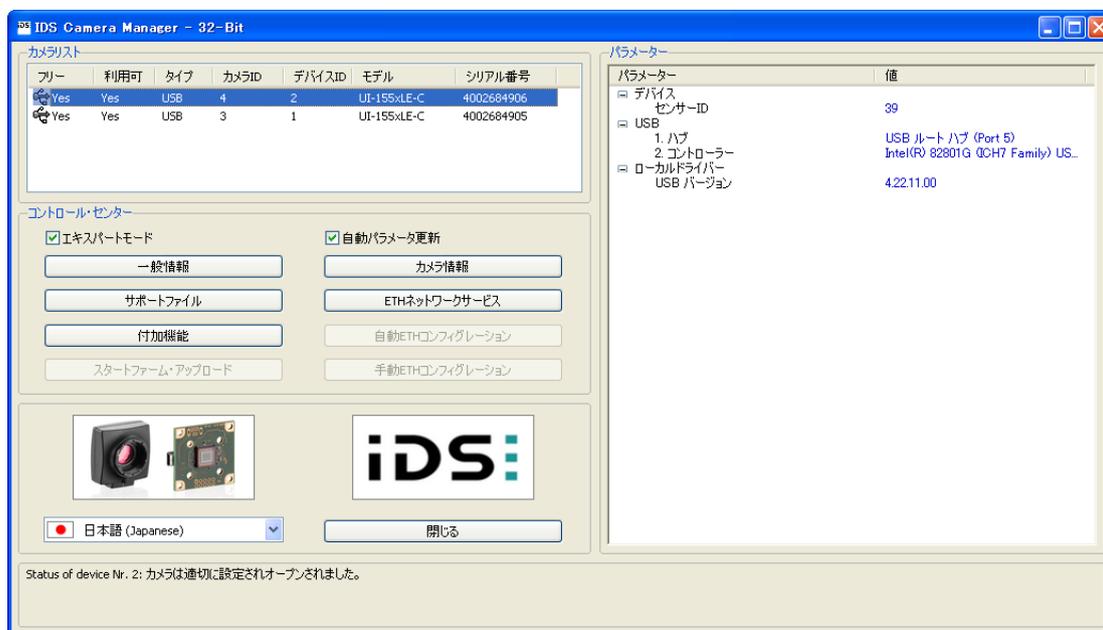
```
Dim vntRet as Variant
vntRet = caoCtrl.Execute("GetSensorInfo")

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

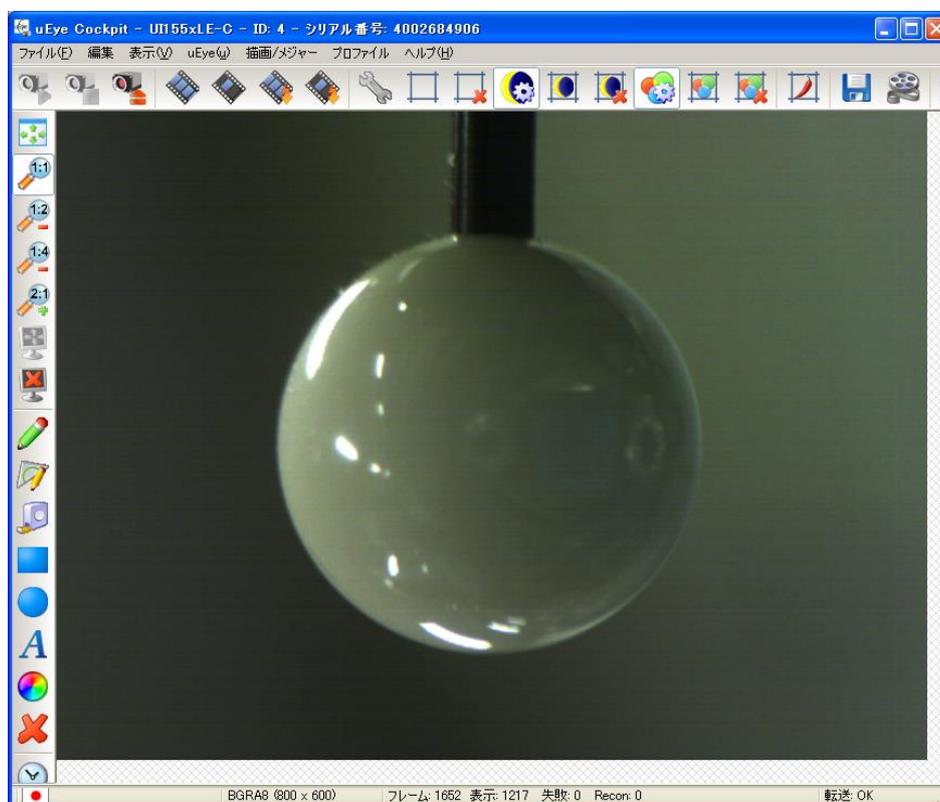
3. カメラの設定例

3.1. ソフトウェアトリガを使用する場合

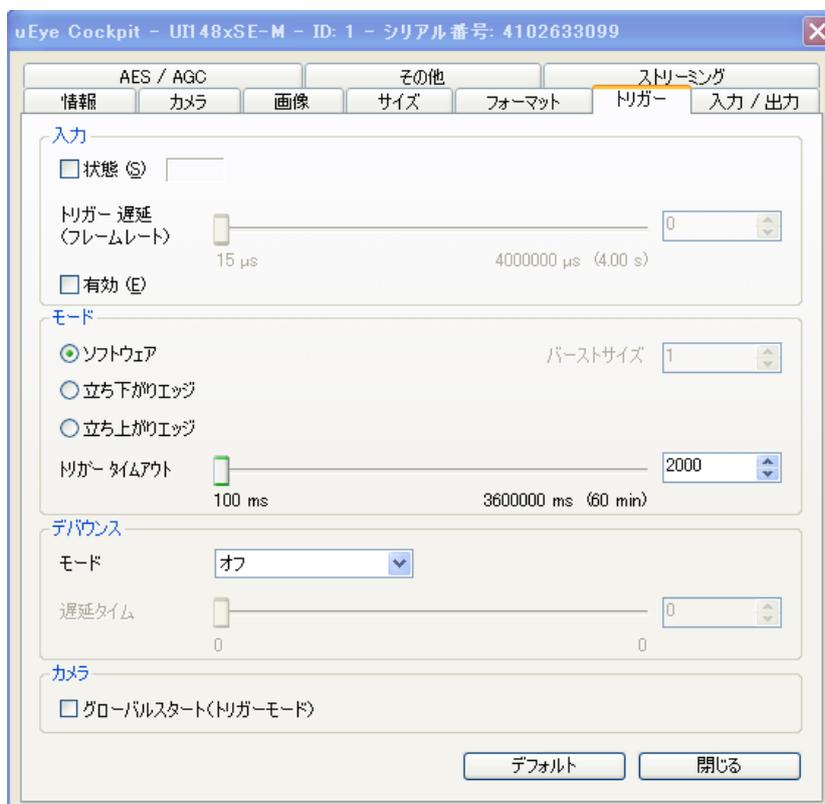
初めに「IDS Camera Manager」を立ち上げます。



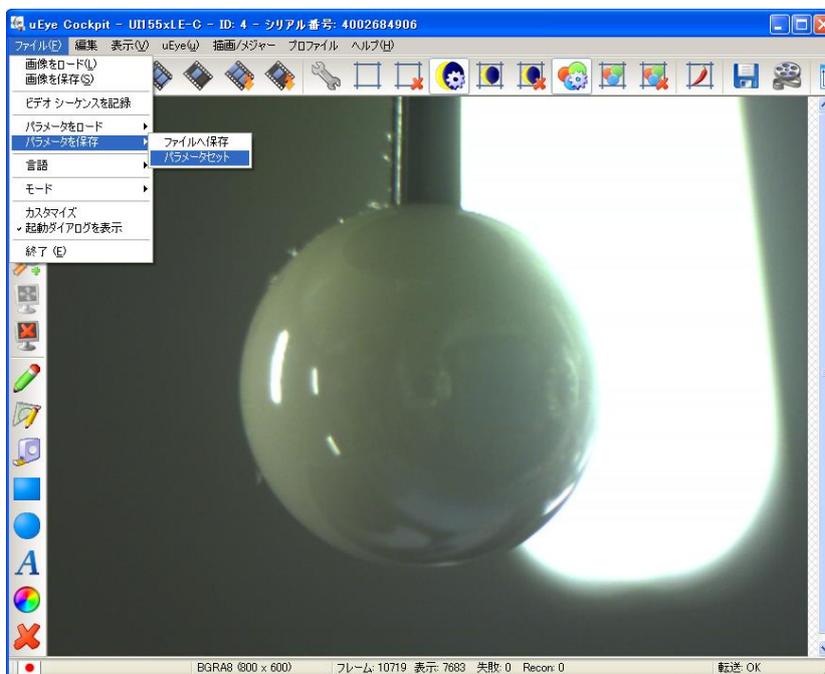
設定したいカメラを選択し、「uEye Cockpit」を立ち上げます。



立ち上げ時には連続取り込みモードになっていますので、一度トリガ取り込みボタンを押してから、設定ボタンを押して設定画面を立ち上げます。



ソフトウェアトリガにチェックが入っていることを確認後、パラメータセットに保存します。



画像を取得する場合には、「FreezeVideo」コマンドでソフトウェアトリガをかけて、「GetImage」コマンドで画像を取得します。

```
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")
```