DirectShow provider

Get images (DirectShow)

Version 1.2.1

User's Guide

June 9, 2020

[Remark]

[Revision History]

Version	Date	Description
1.0.0.0	2009-01-19	First edition
1.0.1.0	2009-06-15	Various commands were added.
1.0.2.0	2009-09-15	System variables, events added to the controller class.
1.0.2.1	2010-03-10	Error code added, GetImageSize added to File.
1.0.2.2	2010-11-22	Added descriptions of properties of IsEmpty implementation. File classes.
1.0.2	2012-07-17	Modify document version rules.
1.0.3	2012-10-22	GetCameraFormatList command and SetCameraFormat command are added to the controller class.
1.1.1	2013-07-01	AddController optional added Add SetCameraFrameRate, GetCameraFrameRate Commands
1.1.2	2014-08-26	Adding System Variables to a File Class
1.2.0	2018-12-26	Add JPEG Format to Get Images AddController optional additional ImageType, JpegQuality
1.2.1	2020-06-09	OpenFilterProperty commands, change OpenPinProperty command parameters, add your own error codes, improve behavior, and correct phrases

[Compatible device]

Model	Version	Notes

[Operation Check device]

Model	Firmware Version	Driver Version
Logicool Qcam Orbit/Sphere AF	2.96.6009	13.80.853.0

Table of Contents

1. Introduction

This user's guide is for DirectShow providers, a CAO provider that is a ORiN application that uses DirectShow to retrieve images from capture devices. Fig. 1-1 shows the overall configuration of this provider and the device.

DirectShow providers use DirectShow, one of the APIs included in Microsoft's Windows Multimedia-Enhanced API Group DirectX, to get images from capture devices.

DirectShow providers pass images from the capture device to the client as requested by the client.

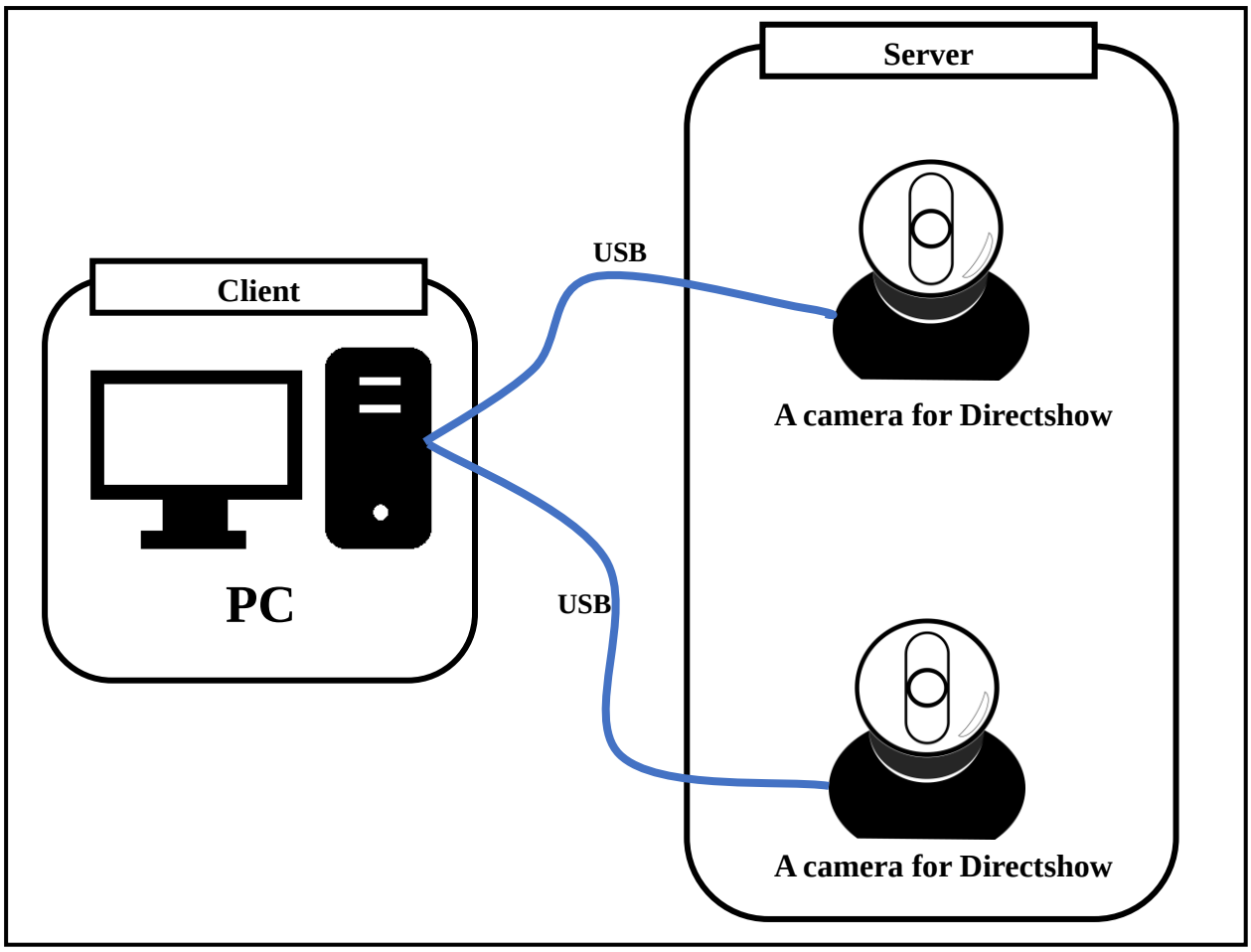

Fig. 1-1 Configuration Diagram

Fig. 1-2 shows the correspondence between this provider and each device.

(※An example. It does not represent everything.)

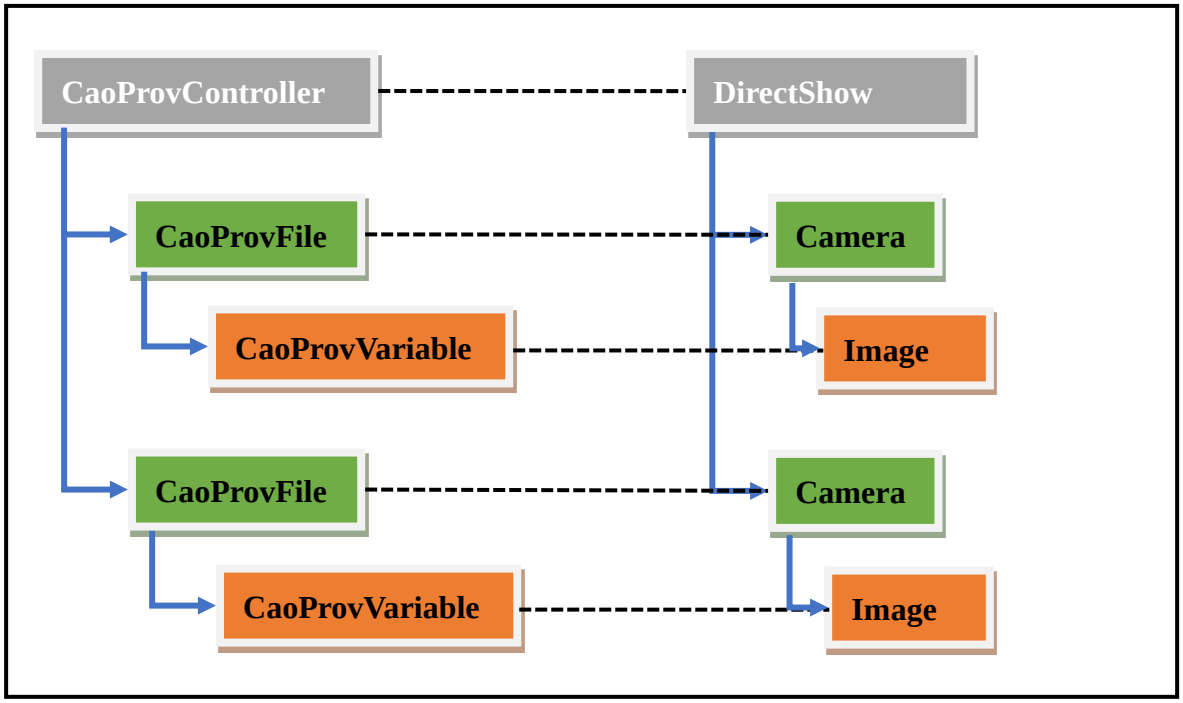

Fig. 1-2 Provider configuration and device information

2. Provider Overview

2.1. Introduction

DirectShow providers use file classes to get USB-camera images. In addition, variable classes can be used to obtain and set the response sensitivity when reporting image change events.

DirectShow providers have a DLL (Dynamic Link Library) file format that is loaded dynamically when used from CAO engine. To use DirectShow providers, you must install ORiN2SDK or manually register the registry by referring to Table 2-1.

Table 2-1 File Formats for DirectShow Provider

File name	CaoProvDirectShow.dll
ProgID	CaoProv.DirectShow
Registry registration	Regsvr32 CaoProvDirectShow.dll
Unregistering the Registry	Regsvr32 /u CaoProvDirectShow.dll

3. Command Reference

3.1. Method/Property List

Category	Methods/Properties[1]		Function	Page
CaoWorkspace				
	AddController	M	Connected to controller	P. 8
CaoController				
	AddFile	M	Adding a File/Folder Object	P.10
	VariableNames	P	Get a list of variable names that can be connected	P.11
	AddVariable	M	Adding Variable Objects	P.11
	Execute	M	Execute Extended Commands	P.11
	OnMessage	E	Message reception event	P.11
CaoFile				
	VariableNames	P	Get a list of variable names that can be connected	P.12
	AddVariable	M	Adding Variable Objects	P.12
	ID	P	ID get/set	P.12
	DateLastModified	P	Get the update time of the currently referenced camera	P.13
	Help	P	Get the device name of the currently referenced camera	P.13
	Value	M	Acquiring image data from the camera	P.13
	Execute	M	Execute Extended Commands	P.14
CaoVariable				
	Value	P	Get/set value	P.14

[1] M: Indicates methods, P: properties, and E: events, respectively.

3.2. Method properties

3.2.1. CaoWorkspace classes

3.2.1.1. AddController method

DirectShow providers search for cameras and connect to them during AddController. The following are the specifics of AddContoller method:

Format

AddController

```
(
        "<Controller name>",                    // Controller name (optional)
        "CaoProv.Directshow",                   // Provider name (fixed)
        "<Machine name>",                       // Provider execution machine name (unused)
        "<Option>"                              // Option character string (optional)
)
```

Option

The following options are specified in the option string: The option string is a string consisting of the following options separated by a comma (,).

Option	Required	Description	Range	Default
CameraDisable [=<Specify unused camera>]	--	Specifies the bit of the camera that is not to be used. Bit 0 corresponds to ID=0 and is specified up to bit 9. Bit ON: Not used. Bit OFF: Used. Example: Do not use Camera 3 CameraDisable=4	-32768 ~ 32767	0
FormatType[= t1:t2:t3:t4:t5: t6:t7:t8:t9:t10]	--	Specifies the camera format at startup. If you do not specify a value or a value that cannot be set, the camera starts with the default camera settings. The settable values can be obtained by GetCameraFormatList command after AddController. Example: Starting the Second Camera	Device dependent	0

Option	Required	Description	Range	Default
		with the Second Camera Setting FormatType=0:2:0:0:0:0:0:0:0:0 * When you specify the format number of the camera built in the notebook, you may be able to connect to the camera. Refer to 3.5.1 for details.		
FrameRate[= f1:f2:f3:f4:f5: f6:f7:f8:f9:f10]	--	Specifies the camera frame rate at startup. If it is not set or cannot be set, it starts at the default frame rate. Example: Starting the Frame Rate of the Second Camera at 20 FrameRate=0:20:0:0:0:0:0:0:0:0	Device dependent	Device dependent
ImageType [=<image type>]	--	Specify one of the following as the image type when acquiring an image. 1: BMP format 2: JPEG format[2] If it is not set or cannot be set, it starts in the default BMP format. Example: Starting an Image Type in BMP Format ImageType=1	1 ~ 2	1
JpegQuality [=<JPEG>]	--	Specifies JPEG quality when image acquisition is specified in JPEG format. 0 (Low Quality) to 100 (High Quality) Example: Starting JPEG Quality at 60 JpegQuality =60	0 ~ 100	80

[2] If you are using a version earlier than WindowsXP, please specify the BMP format. If you specify JPEG conversion and repeat AddController and Delete, memory leak may occur.

Example (VB)

```
Dim caoEng As CaoEngine                 ' Engine
Dim caoWs As CaoWorkspace               ' Workspace
Dim caoCtrl As CaoController            ' Controller

' Create CaoEngine
Set caoEng = new CaoEngine
' Create CaoWorkspace
Set caoWs = caoEng.Workspaces.Item(0)
' Create CaoController
Set caoCtrl = caoWs.AddController("Directshow", _
                                  "CaoProv.DirectShow", _
                                  "", _
                                  " CameraDisable = 0, FormatType = 0:0:0:0:0:0:0:0:0:0, FrameRate =
0:0:0:0:0:0:0:0:0:0, ImageType = 2, JpegQuality = 40")
```

3.2.2. CaoController classes

3.2.2.1. AddFile method

Create a file object to access the video capture device.

Format

AddFile

```
(
        "<Filename>",           // File name (optional)
        "<Option>"              // Option character string
)
```

Option

The following options are specified in the option string:

Option	Required	Description	Range	Default
ID	--	Number of the initially connected image memory	1 - 10	1

Example (VB)

```
' File
Dim caoFile As File
' Create CaoFile
Set caoFile = caoCtrl.AddFile("WebCamera1","ID=1")
```

3.2.2.2. VariableNames Properties

Gets a list of variable names. The variable name obtained by this property can be used as the first argument of the AddVariable method described later.

Example (VB)

```
' Get variable name list
Dim variableNames() As String
variableNames = caoCtrl.variableNames
```

3.2.2.3. AddVariable method

Creates a variable object for acquiring camera information and provider information settings. You can only use variables from 3.4.1 in variable names. AddVariable is specifi ed as follows.

Format

AddVariable

```
(
        "<variable name>",          // Variable name
        "<Option>"                  // Option character string (optional)
)
```

3.2.2.4. Execute method

Execute CaoController extended command. For the extended commands that can be specified with Execute, only the commands shown in 3.3.1 can be executed. The specifi cations of Execute are shown below.

Format

Execute

```
(
        "<extension command name>",     // Extended command name
        "<Option>"                      // Option character string (optional)
)
```

3.2.2.5. OnMessage event

When the image data is updated, a CaoController class-wide OnMessage event occurs. At this time, Message::Number property is set to 1, and Message::Value property is set to the image number of the camera where the image data has been updated.

3.2.3. CaoFile classes

3.2.3.1. VariableNames Properties

Gets a list of variable names. The variable name obtained by this property can be used as the first argument of the 3.2.3.2. AddVariable method described later.

Example (VB)

```
' Get variable name list
Dim variableNames() As String
variableNames = caoFile.VariableNames
```

3.2.3.2. AddVariable method

Create a variable object to get the camera image. You can only use variables from 3.4.2 for variable names.

Format

AddVariable

```
(
        "<variable name>",          // Variable name
        "<Option>"                  // Option character string (optional)
)
```

3.2.3.3. ID property

Gets/sets the currently referenced camera ID.

Data Type

Type	Description
VT_I4	Camera ID: 1-10

Example (VB)

```
' Get ID
Dim id As Integer
id = caoFile.id
' ID setting
CaoFile.id = 3
```

3.2.3.4. DateLastModified Properties

Gets the update time of the currently referenced camera. If the camera is not connected, get VT_EMPTY.

Example (VB)

```
' Get update time
If Not IsEmpty(caoFile.DateLastModified) Then
        Dim dateStr As String
        dateStr = caoFile.DateLastModified
End If
```

3.2.3.5. Help Properties

Gets the device name of the currently referenced camera. If the camera is not connected, get VT_EMPTY.

Example (VB)

```
' Get the device name
Dim name As String
name = caoFile. Help
```

3.2.3.6. Value Properties

Gets the currently referenced images in BMP or JPEG format.

Data Type

Type Description	
VT_ARRAY | UI1	Image in the image memory

Example (VB)

```
' Get Camera image data
Dim bArray() As Byte
bArray = caoFile.Value
```

3.2.3.7. Execute method

Execute CaoFile extended command. For the extended commands that can be specified with Execute, only the commands indicated in can be executed. The specifi cations of Execute are shown below.

Format

Execute

```
(
        "<Extension command name>",          // Extended command name
        "<Option>"                           // Option character string (optional)
)
```

3.2.4. CaoVariable classes

3.2.4.1. Value Properties

Gets/sets data from the connected camera. The behavior depends on the variable name. For details, see section , Variable List.

3.3. Extended command list

Defines the list of extended commands available for each class.

3.3.1. CaoController classes

Table 3-1 List of Extended Commands for CaoController Classes

Command	Description	Page
OpenFilterProperty	Displays the Camera Filter Properties window.	P.16
OpenPinProperty	Displays the Camera Output Pin Properties window.	P.17
GetCameraCount	Returns the number of the currently connected cameras.	P.18
GetCameraFormatList	Gets a list of formats that can be set for the camera	P.18
GetCameraFormat	Obtains the index number of the format set in the camera.	P.19
SetCameraFormat	Set the camera resolution, etc.	P.20
GetCameraFrameRate	Gets the frame rate set for the camera.	P.20
SetCameraFrameRate	Sets the frame rate of the camera.	P.21

3.3.1.1. OpenFilterProperty Commands

Displays the properties window for the camera filter with the specified camera ID. The camera filter property window can control camera control, brightness, contrast, etc. as shown in Fig. 3-1. The Properties window that is displayed depends on the device to which you are connected.

You can also specify a parent window handle to display it as a child window of the specified parent window.

Fig. 3-1 Camera Filter Properties Window Example

If you do not specify a parent window:

Item	Type Description	
Argument	VT_I4	Camera ID: 1 to 10
Return Value	None	

[3] for specifying the parent window:[3]

Item	Type Description	
Argument	VT_ARRAY | VT_I4	
	0	Camera ID: 1 to 10
	1	Handle of the parent window or owner window
Return Value	None	

Example (VB)

```
' Properties Window Display (Independent Window)
Call caoCtrl.Execute("OpenFilterProperty", 1)
' Display Properties window (specify parent window)
Dim param As Variant
param = Array(1, 36574396)
Call caoCtrl.Execute("OpenFilterProperty", param)
```

3.3.1.2. OpenPinProperty Commands

Displays the Output Pin Properties window for the specified camera ID. The Output Pin Properties window allows you to control the output size, frame rate, and so on, as shown in Fig. 3-2. You can also specify a parent window handle to display it as a child window of the specified parent window.

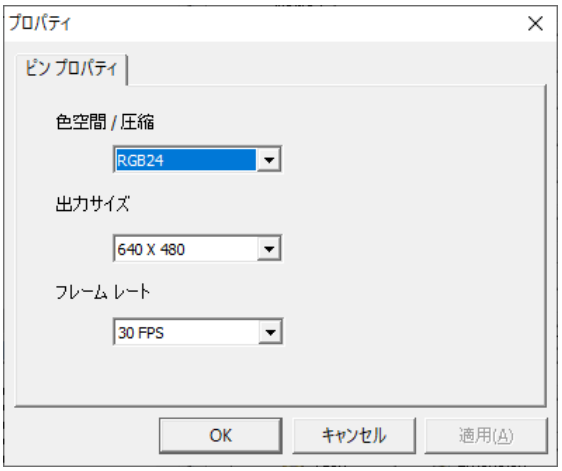

Fig. 3-2 Example of the Output Pin Property Window

The following are the arguments and return values:

If you do not specify a parent window:

Item	Type Description	
Argument	VT_I4	Camera ID: 1 to 10
Return Value	VT_I4	Camera format number set in the properties window

To specify a parent window:[3]

Item	Type Description	
Argument	VT_ARRAY | VT_I4	
	0	Camera ID: 1 to 10
	1	Handle of the parent window or owner window
Return Value	VT_I4	Camera format number set in the properties window

[3] You cannot operate the parent window until you close the properties window. (Modal)
You may not be able to set the parent window handle in some properties windows.
You can use Microsoft's "Microsoft Spy ++" to check the data.
If an error code "0x80040217" occurs, see 3.5.1.

Example (VB)

```
' Output Pin Properties Window Display (Independent Window)
Dim index As Integer
index = caoCtrl.Execute("OpenPinrProperty", 1)
' Output Pin Properties Window Display (Specify Parent Window)
Dim param As Variant
param = Array(1, 36574396)
index = caoCtrl.Execute("OpenPinProperty", param)
```

3.3.1.3. GetCameraCount Commands

Gets the number of currently connected cameras. The return values are shown below.

Item	Type Description	
Argument	None	
Return Value	VT_I4	Number of connected cameras

Example (VB)

```
' Get the number of camera connections
Dim cameraCount As Integer
cameraCount = caoCtrl.Execute("GetCameraCount")
```

3.3.1.4. GetCameraFormatList Commands

Gets a list of formats that can be set for the camera. The following are the arguments and return values:

Item	Type Description		
Argument	VT_I4	Camera ID: 1 to 10	
Return Value	VT_ARRAY | VT_VARIANT		
	I	VT_ARRAY | VT_I4	Format information
		0	Index number
		1	Resolution (horizontal)
		2	Resolution (Vertical)

※ **i : Number of formats that can be set**

Example (VB)

```
' Get a list of formats that can be set for the camera
Dim value As Variant
value = caoCtrl.Execute("GetCameraFormatList", 1)

If Not IsEmpty(value) Then
        Dim i As Integer
        For i = LBound(value) To UBound(value)
                ' Each format information
                Dim formatInfo As Variant
                formatInfo = value(i)
                If Not IsEmpty(formatInfo) Then
                        ' Index number
                        Dim index As Integer
                        index = formatInfo(0)
                        ' Resolution (horizontal)
                        Dim width As Integer
                        width = formatInfo(1)
                        ' Resolution (Vertical)
                        Dim height As Integer
                        height = formatInfo(2)
                End If
        Next i
End If
```

3.3.1.5. GetCameraFormat Commands

Obtains the index number of the format set in the camera. The settings correspond to the settings obtained in 3.3.1.4. GetCameraFormatList. The following are the arguments and return values.

Item	Type Description	
Argument	VT_I4	Camera ID: 1 to 10
Return Value	VT_I4	Index number of the format set for the camera

Example (VB)

```
' Get current format
Dim format As Integer
format = caoCtrl.Execute("GetCameraFormat", 1)
```

3.3.1.6. SetCameraFormat Commands

Set the camera resolution, etc. For settable index numbers, etc., you can set the value obtained by 3.3.1.4. GetCameraFormatList command.

Item	Type Description	
Argument	VT_ARRAY | VT_I4	
	0	Camera ID: 1 to 10
	1	Index number
Return Value	None	

※If an error code "0x80040217" occurs, see 3.5.1.

Example (VB)

```
' Set the format
Dim param As Variant
param = Array(1, 10)
Call caoCtrl.Execute("SetCameraFormat", param)
```

3.3.1.7. GetCameraFrameRate Commands

Gets the frame rate set for the camera. The following are the arguments and return values:

Item	Type Description	
Argument	VT_I4	Camera ID: 1 to 10
Return Value	VT_I4	Frame Rate

Example (VB)

```
' Get current frame rate
Dim frameRate As Integer
frameRate = caoCtrl.Execute("GetCameraFrameRate", 2)
```

3.3.1.8. SetCameraFrameRate Commands

Sets the frame rate of the camera. If you specify a value that cannot be set, the behavior depends on the driver specifications. The following arguments are provided.

Item	Type Description	
Argument	VT_ARRAY | VT_I4	
	0	Camera ID: 1 to 10
	1	Frame Rate
Return Value	None	

Example (VB)

```
' Set Frame Rate
Dim param As Variant
param = Array(2, 10)
Call caoCtrl.Execute("SetCameraFrameRate", param)
```

3.3.2. CaoFile classes

Table 3-2 List of Extended Commands for CaoFile Classes

Command	Description	Page
GetRangeCameraCtrl	Gets the range and default values of the properties for the specified camera.	P.23
GetCameraCtrl	Gets the current settings for camera properties.	P.25
SetCameraCtrl	Sets the properties of the specified camera.	P.26
GetRangeVideoProcAmp	Gets the minimum, maximum, and default values for a configuration property.	P.27
GetVideoProcAmp	Gets the video quality of the specified properties.	P.28
SetVideoProcAmp	Sets the video quality for the specified properties.	P.29
GetImageSize	Gets the width and height of the camera image.	P.30
IsEmpty	Check that the camera is connected properly.	P.30

3.3.2.1. GetRangeCameraCtrl Commands

Retrieves the range and default values for the specified camera properties. Some properties are not supported by the device. If you specify a property that is not supported, an error code "0x80070490" occurs. If the error code "0x80070490" is generated by this command, the GetCameraCtrl command and SetCameraCtrl command generate an error in the same way. The following are the arguments and return values.

<table>
<tr><td>Item</td><td colspan="3">Type Description</td></tr>
<tr><td>Argument</td><td>VT_I4</td><td colspan="2">Camera properties. Specify one of the following:
0 : Pan
1 : Tilt
2 : Role
3 : Zoom
4 : Exposure
5 : Aperture
6 : Focus</td></tr>
<tr><td rowspan="6">Return Value</td><td colspan="2">VT_ARRAY | VT_I4</td><td>Property Information</td></tr>
<tr><td>0</td><td colspan="2">Minimum value</td></tr>
<tr><td>1</td><td colspan="2">Maximum value</td></tr>
<tr><td>2</td><td colspan="2">Step Size</td></tr>
<tr><td>3</td><td colspan="2">Default Value</td></tr>
<tr><td>4</td><td colspan="2">Flag. It consists of the following formats:
[Operation]
0x00 : Absolute operation. Operation based on absolute value
0x10 : Relative operation. Operation based on current value
[Control method]
0x01 : Auto control
0x02 : Manual control
The value obtained by performing the And operation of either of the above operations and the control method is set.
Example 1-0x01 Absolute Operation & Auto Control
Example 2-0x12 Relative Operation & Manual Control
EXAMPLE 3-0x00 ABSOLUTE OPERATION (NO CONTROL)
Example 4-0x03 Absolute Operation & Auto Control & Manual Control
→ In Example 4, both controls are accepted.</td></tr>
</table>

※ See IAMCameraControl::GetRenge() in Microsoft Developer Network(MSDN for details.

Example (VB)

```
' Get Camera Property Range and Default Values
Dim values As Variant
values = caoFile.Execute("GetRangeCameraCtrl", 0)

If Not IsEmpty(values) Then
        ' Minimum value
        Dim minLimit As Integer
        minLimit = values(0)

        ' Maximum value
        Dim maxLimit As Integer
        maxLimit = values(1)

        ' Step Size
        Dim stepSize As Integer
        stepSize = values(2)

        ' Default Value
        Dim default As Integer
        default = values(3)

        ' Flag
        Dim flag As Integer
        flag = values(4)
End If
```

3.3.2.2. GetCameraCtrl Commands

Retrieves the current settings for the specified camera properties. For properties that failed with GetRangeCameraCtrl command, the error code "0x80070490" is generated in the same way. The following are the arguments and return values.

<table>
<tr><th>Item</th><th colspan="3">Type Description</th></tr>
<tr><td>Argument</td><td>VT_I4</td><td colspan="2">Camera properties. Specify one of the following:
0 : Pan
1 : Tilt
2 : Role
3 : Zoom
4 : Exposure
5 : Aperture
6 : Focus</td></tr>
<tr><td rowspan="3">Return Value</td><td colspan="2">VT_ARRAY | VT_I4</td><td>Property Information</td></tr>
<tr><td>0</td><td colspan="2">Current value</td></tr>
<tr><td>1</td><td colspan="2">Flag. For more information, see section, Flags for GetRangeCameraCtrl Commands.</td></tr>
</table>

※ For more information, see IAMCameraControl::Get() in MSDN.

Example (VB)

```
' Get current value of camera property
Dim values As Variant
values = caoFile.Execute("GetCameraCtrl", 0)

If Not IsEmpty(values) Then
        ' Current value
        Dim value As Integer
        value = values(0)

        ' Flag
        Dim flag As Integer
        flag = values(1)
End If
```

3.3.2.3. SetCameraCtrl Commands

Sets the setting value of the specified camera property. For properties that failed with GetRangeCameraCtrl command, the error code "0x80070490" is generated in the same way. The following arguments are provided.

Item	Type	Description
Argument	VT_ARRAY | VT_I4	
	0	Camera properties. Specify one of the following: 0 : Pan 1 : Tilt 2 : Role 3 : Zoom 4 : Exposure 5 : Aperture 6 : Focus
	1	Setting value. Specify a value within the range gotten by GetRangeCameraCtrl command.
	2	Flag. Specify one of the following: [Control method] 1 : Automatic control 2 : Manual control If automatic control is specified, the set value is ignored. Some models and properties can be controlled or some cannot be controlled.
Return Value	None	

※ See IAMCameraControl::Set() in MSDN for more information.

※ Operation varies depending on the manufacturer and model. Be careful.

Example (VB)

```
' Set Camera Properties
Dim param As Variant
param = Array(0, 10, 2)
Call caoFile.Execute("SetCameraCtrl", param)
```

3.3.2.4. GetRangeVideoProcAmp Commands

Retrieves the range and default values for the specified video properties. Some properties are not supported by the device. If you specify a property that is not supported, an error code "0x80070490" occurs. If an error code "0x80070490" occurs in this command, an error occurs in 3.3.2.5. GetVideoProcAmp command and 3.3.2.6. SetVideoProcAmp command in the same way. The following are the arguments and return values.

<table>
<tr><th>Item</th><th colspan="3">Type Description</th></tr>
<tr><td>Argument</td><td>VT_I4</td><td colspan="2">Video properties. Specify one of the following:
0 : Brightness level
1 : Contrast
2 : Hue
3 : Saturation
4 : Sharpness
5 : Gamma
6: Color enable setting
7 : White Balance
8 : Backlight compensation setting
9 : Gain adjustment</td></tr>
<tr><td rowspan="6">Return Value</td><td colspan="2">VT_ARRAY | VT_I4</td><td>Property Information</td></tr>
<tr><td>0</td><td colspan="2">Minimum value</td></tr>
<tr><td>1</td><td colspan="2">Maximum value</td></tr>
<tr><td>2</td><td colspan="2">Step Size</td></tr>
<tr><td>3</td><td colspan="2">Default Value</td></tr>
<tr><td>4</td><td colspan="2">Flag. For more information, see section 0, Flags for GetRangeCameraCtrl Commands.</td></tr>
</table>

※ See IAMVideoProcAmp::GetRenge() section of MSDN for more information.

Example (VB)

```
' Get video property ranges and default values
Dim values As Variant
values = caoFile.Execute("GetRangeVideoProcAmp", 7)

If Not IsEmpty(values) Then
        ' Minimum value
        Dim minLimit As Integer
        minLimit = values(0)

        ' Maximum value
        Dim maxLimit As Integer
        maxLimit = values(1)

        ' Step Size
```

```
        Dim stepSize As Integer
        stepSize = values(2)

        ' Default Value
        Dim default As Integer
        default = values(3)

        ' Flag
        Dim flag As Integer
        flag = values(4)
End If
```

3.3.2.5. GetVideoProcAmp Commands

Gets the current settings for the specified video properties. For properties that failed with GetRangeVideoProcAmp command, the error code "0x80070490" is generated in the same way. The following are the arguments and return values.

Item	Type Description	
Argument	VT_I4	Video properties. Specify one of the following: 0 : Brightness level 1 : Contrast 2 : Hue 3 : Saturation 4 : Sharpness 5 : Gamma 6: Color enable setting 7 : White Balance 8 : Backlight compensation setting 9 : Gain adjustment
Return Value	VT_ARRAY | VT_I4	Property Information
	0	Current value
	1	Flag. For more information, see section Flags for GetRangeCameraCtrl Commands.

※ For more information, see IAMVideoProcAmp::Get() in MSDN.

Example (VB)

```
' Get current value of video property
Dim values As Variant
values = caoFile.Execute("GetVideoProcAmp", 7)

If Not IsEmpty(values) Then
        ' Current value
        Dim value As Integer
        value = values(0)
```

```
        ' Flag
        Dim flag As Integer
        flag = values(1)
End If
```

3.3.2.6. SetVideoProcAmp Commands

Sets the settings for the specified video properties. For properties that failed with GetRangeVideoProcAmp command, the error code "0x80070490" is generated in the same way. The following arguments are provided.

Item	Type Description	
Argument	VT_ARRAY | VT_I4	
	0	Video properties. Specify one of the following: 0 : Brightness level 1 : Contrast 2 : Hue 3 : Saturation 4 : Sharpness 5 : Gamma 6: Color enable setting 7 : White Balance 8 : Backlight compensation setting 9 : Gain adjustment
	1	Setting value. Specify a value within the range acquired by GetRangeVideoProcAmp command.
	2	Flag. Specify one of the following: [Control method] 1 : Automatic control 2 : Manual control If automatic control is specified, the set value is ignored. Some models and properties can be controlled or some cannot be controlled.
Return Value	None	

※ See IAMVideoProcAmp::Set() in MSDN for more information.

※ Operation varies depending on the manufacturer and model. Be careful.

Example (VB)

```
' Set Video Properties
Dim param As Variant
param = Array(7, 10, 2)
```

```
Call caoFile.Execute("SetVideoProcAmp", param)
```

3.3.2.7. GetImageSize Commands

Gets the width and height of the camera image. The return values are shown below.

Item	Type Description	
Argument	None	
Return Value	VT_ARRAY | VT_I4	Image size
	0	Width
	1	Height

Example (VB)

```
' Get camera image width and height
Dim values As Variant
values = caoFile.Execute("GetImageSize")

If Not IsEmpty(values) Then
        ' Width
        Dim width As Integer
        width = values(0)
        ' Height
        Dim height As Integer
        height = values(1)
End If
```

3.3.2.8. IsEmpty Commands

Verify that the camera is connected. The return values are shown below.

Item	Type Description	
Argument	None	
Return Value	VT_BOOL	True: Camera connection incomplete False : Camera-connected

Example (VB)

```
' Connection confirmation
Dim value As Boolean
value = caoFile.Execute("IsEmpty")
```

3.4. Variable list

3.4.1. CaoController classes

Table 3-3 CaoController Class System Variable List

Variable name	Description	Attribute		Page
		Get	Put	
@CAM_COUNT	Number of connected cameras	○	-	P.31
@VERSION	Provider version	○	-	P.31
@EVENT_ENABLED	Switches the occurrence of a CAO message event.	○	○	P.32

3.4.1.1. @CAM_COUNT

Gets the total number of currently connected cameras.

Data Type

Item	Type Description	
Acquisition	VT_I4	Total number of connected cameras

Example (VB)

```
' Add Variable
Dim var As CaoVariable
Set var = caoCtrl.AddVariable("@CAM_COUNT")
' Acquisition of Values
Dim val As Integer
val = var.value
```

3.4.1.2. @VERSION

Get the provider version.

Data Type

Item	Type Description	
Acquisition	VT_BSTR	Provider version *.*.*

Example (VB)

```
' Add Variable
Dim var As CaoVariable
Set var = caoCtrl.AddVariable("@VERSION")
' Acquisition of Values
Dim val As String
val = var.value
```

3.4.1.3. @EVENT_ENABLED

3.2.2.5. Switches OnMessage event generation for CaoController.

Data Type

Item	Type Description	
Get/Set	VT_BOOL	True : Generates a message event. False: Do not generate a message event.

Example (VB)

```
' Add Variable
Dim var As CaoVariable
Set var = caoCtrl.AddVariable("@EVENT_ENABLED")
' Acquisition of Values
Dim val As Boolean
val = var.value
' Value setting
var.value = true
```

3.4.2. CaoFile classes

Table 3-4 CaoFile Class System Variable List

Variable name	Description	Attribute		Page
		Get	Put	
@VALUE	Image in the image memory	○	-	P.33

3.4.2.1. @VALUE

Acquires the currently referenced images in BMP or JPEG format.

Data Type

Item	Type Description	
Acquisition	VT_ARRAY | VT_UI1	Image in the image memory

Example (VB)

```
' Add Variable
Dim var As CaoVariable
Set var = caoFile.AddVariable("@VALUE")
' Acquisition of Values
Dim val() As Byte
val = var.value
```

3.5. Error code

The provider has the following unique error codes: (Refer to the proprietary error code table in Table 3-5.)
For ORiN2 common errors, refer to the Error Codes section of ORiN2 Programming Guide.

Table 3-5 Unique Error Codes

Error Number	Description
0x80100001	The value specified for the ID option is incorrect. Specify an ID in the range of 1 to 10.
0x80100002	The camera is not connected or is not in use. Check the connection with the camera.
0x80100003	Image data does not exist. Perform GetValue after a while.

3.5.1. Notebook PC built-in camera

If you specify a specific index number such as the resolution of the laptop's built-in camera, one of the following may occur.

1. **When specified by FormatType**

   If FormatType option is used to specify the format of the laptop built-in camera during AddController, the connection will fail if a certain value is set. Specify a different formatting for AddContoller.

2. **When specified by OpenPinProperty command or SetCameraFormat**

   The error code "0x80040217" may occur when the command is executed. The same error code will occur even if another index number is specified. If this happens, perform AddContoller again. Also, do not specify a format in which an error occurs.

4. Sample program

Please refer to Visual Basic example programs in the following folders.

<install folder>¥ORiN2¥CAO¥ ProviderLib¥DirectShow¥Samples¥Capture¥VB2010

5. API list

CaoWorkspace::AddController

API function name
ICreateDevEnum::CreateClassEnumerator
IEnumMoniker::Next
IMoniker::BindToStorage
IMoniker::BindToObject
IPropertyBag::Read
IBaseFilter::QueryInterface
IBaseFilter::EnumPins
IGraphBuilder::QueryInterface
IGraphBuilder::AddFilter
IGraphBuilder::Connect
ISampleGrabber::QueryInterface
ISampleGrabber::SetBufferSamples
ISampleGrabber::SetOneShot
ISampleGrabber::SetMediaType
ISampleGrabber::SetCallback
IEnumPins::Next
IPin::QueryDirection
IPin::QueryInterface
IAMStreamConfig::GetNumberOfCapabilities
IAMStreamConfig::GetStreamCaps
IAMStreamConfig::SetFormat
IAMVideoControl::GetCaps
IMediaControl::GetState
IMediaControl::Run
IMediaControl:: Stop

CaoController::Execute

Command name	API function name
OpenFilterProperty	IBaseFilter::QueryInterface ISpecifyPropertyPages::GetPages
OpenPinProperty	IPin::QueryInterface IAMStreamConfig::QueryInterface ISpecifyPropertyPages::GetPages IGraphBuilder::Connect IGraphBuilder::Disconnect ISampleGrabber::SetCallback
GetCameraFormatList	IPin::QueryInterface IAMStreamConfig::GetNumberOfCapabilities IAMStreamConfig::GetStreamCaps
GetCameraFormat	IPin::QueryInterface IAMStreamConfig::GetFormat
SetCameraFormat	IPin::QueryInterface IGraphBuilder::Connect IGraphBuilder::Disconnect ISampleGrabber::SetCallback IAMStreamConfig::GetNumberOfCapabilities IAMStreamConfig::GetStreamCaps IAMStreamConfig::SetFormat
GetCameraFrameRate	IPin::QueryInterface IAMStreamConfig::GetFormat
SetCameraFrameRate	IPin::QueryInterface IGraphBuilder::Connect IGraphBuilder::Disconnect ISampleGrabber::SetCallback IAMStreamConfig::GetFormat IAMStreamConfig::SetFormat

CaoFile::Execute

Command name	API function name
GetRangeCameraCtrl	IAMCameraControl::GetRange
GetCameraCtrl	IAMCameraControl::Get
SetCameraCtl	IAMCameraControl::Set

Command name	API function name
GetRangeVideoProcAmp	IAMVideoProcAmp::GetRange
GetVideoProcAmp	IAMVideoProcAmp::Get
SetVideoProcAmp	IAMVideoProcAmp::Set

CaoFile::get_Value (when Jpeg format is specified)

For WindowsVista or higher

API function name
IWICImagingFactory::CreateBitmapFromHBITMAP
IWICImagingFactory::CreateEncoder
IWICBitmapEncoder::Initialize
IWICBitmapEncoder::CreateNewFrame
IWICBitmapEncoder::Commit
IPropertyBag2::Write
IWICBitmapFrameEncode::Initialize
IWICBitmapFrameEncode::WriteSource
IWICBitmapFrameEncode::Commit

For WindowsXP and earlier

API function name
Gdiplus::DllExports::GdipCreateBitmapFromGdiDib
Gdiplus::DllExports::GdipSaveImageToStream
Gdiplus::DllExports::GdipDisposeImage

CaoVariable::get_Value (when Jpeg format is specified)

For WindowsVista or higher

Variable name	API function name
@VALUE	IWICImagingFactory::CreateBitmapFromHBITMAP IWICImagingFactory::CreateEncoder IWICBitmapEncoder::Initialize IWICBitmapEncoder::CreateNewFrame IWICBitmapEncoder::Commit IPropertyBag2::Write IWICBitmapFrameEncode::Initialize IWICBitmapFrameEncode::WriteSource IWICBitmapFrameEncode::Commit

For WindowsXP and earlier

Variable name	API function name
@VALUE	Gdiplus::DllExports::GdipCreateBitmapFromGdiDib Gdiplus::DllExports::GdipSaveImageToStream Gdiplus::DllExports::GdipDisposeImage