

EasyVideo ActiveX Ver 3.3.0.29 User Manual

<http://www.lht2000.com>

2005/05/28

This document can copy and distribute by anyone, but any modification must agreed by author.

Revision History

Initial Version(3.0)	11/19/2003	Hunter
Version 3.2 1) support NAT 2) support select video capture device, because it is use VFW now, so it can only support one usb video capture device now. 3) Check if the local port are available	11/25/2004	Hunter
Version 3.3 1) disable Full Screen 2) add Mothod ShowErrorDialog to disable error dialog appear. 3) Add Mothods SendFakeVideo to enable one-way NAT.	12/07/2004	Hunter
Version 3.3.0.29 1) support server receive and forward several stream(maximum is 30). 2) Support server write the video stream to file 3) The server can listen to one of clients	5/28/2005	Hunter

1、 Introduction

EasyVideo ActiveX Control allows easy integration of Video Conference or Web Conference capabilities into your applications. The control supports most VFW capable video devices like USB cameras or Video Capture Card. It allows you to capture video, compression, transport, decompression and show video. It supports MPEG-4 codec and supports bandwidth adjustment. It uses RTP(Realtime Transport Protocol) as transport protocol and supports multicast. EasyVideo will incorporate in any ActiveX hosting environment.

You can find follow files from EasyVideo.zip:

EasyVideo.doc: This document, EasyVideo ActiveX control User Manual.

EasyVideoTest Directory: The VC source code demonstrate how to use EasyVideo.ocx

EasyVideo.ocx : EasyVideo ActiveX control , When in Windows98 or Windows Me , you should copy this file to 'system' directory under Windows setup directory; When in Windows2000 or Windows XP/NT, you should copy this file to 'system32' directory under Windows setup directory. And then please run 'regsvr32 EasyVideo.ocx'.

What's new in version 3.1?

- (1) Methods **SetVideoInFormat** and **SetVideoOutFormat** are now support any video size such as "320*240" "640*480" "888*777".
- (2) Modified socket bug when stop receive video or stop send video but socket is not deleted.

What's new in version 3.3?

- (1) Disable full screen because web issue, you can control the size of ActiveX to realize the full screen.
- (2) Add Method ShowErrorDialog to disable error dialog appear.
- (3) Add Methods SendFakeVideo to enable one-way NAT..

2、 How to use EasyVideo.OCX

EasyVideo.ocx have no property and you only need to learn how to use methods of EasyVideo.ocx. The methods is as follow:

(1): void SetBandwidth(long lBandwidth, long lCodeOrDecode);

Set bandwidth and codec flag, this function must called first.

Return Value:

none

Parameters :

lBandwidth:

Bandwidth's unit is bps, range from 64000bps to 4096000bps;

lCodeOrDecode :

lCodeOrDecode is set to 1 or 2, 1 means compression, 2 means decompression.

(2): BOOL SetSendFrameRate(short sFrameRate);

Set frame rate of send video

Return Value:

TRUE means success and FALSE mean failed

Parameters:

sFrameRate:

frame rate of send video, range from 1 to 30 frames per seconds.

(3): BOOL SetUserName(LPCTSTR szUserName);

Set local user name. If you set user name, user name will paint to video which send to remote user; If you do not set user name, nothing will add to video.

Return Value:

TRUE means success and FALSE mean failed

Parameters:

szUserName:

User Name, terminate with '\0'.

(4): BOOL SetLanguageVersion(short nLanguageVersion);

Set language version.

Return Value:

TRUE means success and FALSE mean failed

Parameters:

nLanguageVersion:

1 means Chinese , 2 means English.

(5): BOOL SetCompressType(LPCTSTR szCompressType);

Set compression method.

Return Value:

TRUE means success and FALSE mean failed

Parameters:

szCompressType:

This parameter must be "MPEG4" because current version only support MPEG-4 codec.

(6): short SetVideoInFormat(LPCTSTR szVideoIn);

Set format of video capture.

Return Value:

Return 0 means success and other values mean failed

Parameters:

szVideoIn:

This parameter can be set to one the 4 values : "QCIF", "CIF", "4CIF", "16CIF".

QCIF	:176*144
CIF	: 352*288
4CIF	:704*576
16CIF	:1408*1152

This parameter can be set to as follow format “width*height”, such as :

“320*240”

“640*480”

“1024*768”

...

(7): short SetVideoOutFormat (LPCTSTR szVideoOut);

Set format of video display after decompression.

Return Value:

Return 0 means success and other values mean failed

Parameters :

szVideoOut:

This parameter can be set to one the 4 values : "QCIF", "CIF", "4CIF", "16CIF".

QCIF :176*144

CIF : 352*288

4CIF :704*576

16CIF :1408*1152

This parameter can be set to as follow format “width*height”, such as :

“320*240”

“640*480”

“1024*768”

...

(8): BOOL SetLocalIPAddress(LPCTSTR szIP, short usPort);

Set local IP address and port which is used to recv video packet, this function is only used to unicast. If you want multicast, please use function SetMulticastIPAddress

Return Value:

TRUE means success and FALSE mean failed

Parameters :

szIP:

IP address of local computer.

usPort

IP port of local computer to bind.

(9): long SetRemoteIPAddress(LPCTSTR szIP, short usPort, long hSock);

Set remote address and port which is used to send video packet, When used in unicast. You should set remote computer's IP and Port; When used in multicast, you should set multicast address to send .

Return Value:

Socket handle

Parameters :

szIP:

IP address of remote computer.

usPort

IP port of remote computer

hSock

socket handle, you can pass a socket handle to easyvideo, and then easyvideo will not create socket by szIP and usPort, hSock will used instead. This parameter is used to access socket handle which is not created by easyvideo activex control.

(9): BOOL SetMulticastIPAddress(LPCTSTR szIP, short usPort);

Set multicast address and port which is used to recv video packet.

Return Value:

TRUE means successs and FALSE mean failed

Parameters :

szIP:

multicast IP address

usPort

multicast port .

(10): BOOL HaveVideoCapture();

Judge if video capture device is installed correctly.

Return Value:

TRUE means have video capture device and FALSE mean have NOT video capture device.

Parameters :

none

(11): BOOL InitCapture(short DeviceType);

Initial capture device.

Return Value:

TRUE means successs and FALSE mean failed

Parameters :

DeviceType :

The device to initial, 0 means your first capture device. Please set to 0 if you do not know what device is using.

If it is set to -1, then a select video capture device dialog will open, you can select the video capture device you want to use.

(12): BOOL SendVideo();

Start send video to remote computer

Return Value:

TRUE means successs and FALSE mean failed

Parameters :

None

Notice:

If you want to send and receive video simultaneously, you must create two instance of EasyVideo ActiveX control, one is used to send and the other is used to receive video.

(13): BOOL StopSend();

Stop send video to remote computer

Return Value:

TRUE means successs and FALSE mean failed

Parameters :

None

(14): BOOL CloseCapture ();

Close video capture device

Return Value:

TRUE means successs and FALSE mean failed

Parameters :

none

(15): BOOL RecvVideo ();

Start receive video packet

Return Value:

TRUE means successs and FALSE mean failed

Parameters :

none

Notice:

If you want to send and receive video simultaneously, you must create two instance of EasyVideo ActiveX control, one is used to send and the other is used to receive video.

(16): BOOL StopRecv ();

Stop receive video packet

Return Value:

TRUE means successs and FALSE mean failed

Parameters :

none

(17): void ShowFullScreen ();

Show video in full screen

Return Value:

none

Parameters :

none

(18): short GetErrorID();

return last error id

Return Value:

last error id which means as follow:

1 : Connect video driver failed,please check your video device!

- 2 : Already connected video driver!
- 3 : Get audio format failed!
- 4 : Have NOT connect video driver!
- 6 : Get video format failed!
- 7 : The video format is NOT support!
- 8 : Have NOT send video!
- 9 : Already started receive video!

Parameters:

none

(19): void GetErrorMessage(short ErrorID);

Show dialog message box of error id.

Return Value:

none

Parameters:

ErrorID

Error id retruned by GetErrorID

(20): BOOL UseSameSocketForSendAndRecv((long nDeviceID)

you must set it when you want to use more than one receiver at a computer. nDeviceID is corresponding to the parameter of InitCapture. when set the same, they will use the same socket to send and recv.

Please note, when you use this method, you should use it in send and receive activex control simultaneity.

Return Value:

nDeviceID:

The device to initial, 0 means your first capture device. Please set to 0 if you do not know what device is using.

Parameters:

TRUE means successs and FALSE mean failed

(21): void ShowErrorDialog(BOOL bShowErrorDialog)

Let the error dialog pop up or not.

Return Value:

void

Parameters:

bShowErrorDialog : TRUE means pop up the error dialog.

FALSE means not pop up any error dialog.

(22): BOOL SendFakeVideo(short nPacketNum)

You must use this method when in one-way NAT environment. This method will send nPacketNum UDP packet to peer and fake the NAT device to open a pole for UDP communication.

Return Value:

TRUE means success and FALSE means failed

Parameters :

nPacketNum : the number of UDP packet to send.

(23): long AddClient(LPCTSTR szIP, short usPort, LPCTSTR szFileName, LPCTSTR szClientName);

Add one client

Return Value:

The client ID.

If return value < 0, then AddClient failed, please check the input param and the clients number. We only support 30 clients now.

If return value >= 0, then AddClient success. You should keep the return value which will be use in function RemoveClient, Forward, UnForward and ListenToClient.

Parameters :

szIP:

the IP address of Client.

usPort:

the IP port of Client.

szFileName:

the file name of save video stream, only support avi file format now. The file name must like *.avi.

if this param is NULL, then it will not write to any file..

szClientName:

NULL, reserved.

(24): BOOL RemoveClient(long lClientID);

Remove one client

Return Value:

TRUE means success and FALSE means failed

Parameters :

lClientID:

the client ID returned by AddClient.

(25): BOOL RemoveClientAll();

Remove All clients

Return Value:

TRUE means success and FALSE means failed

Parameters :

none

(26): BOOL Forward(long lSrcClient, long lDstClient);

Forward one audio stream received from one client to another client.

If you want to forward from one clients to several other clients, you need call this function

several times.

Return Value:

TRUE means success and FALSE means failed

Parameters:

ISrcClient:

The client which receives audio stream from. The client ID is returned by AddClient.

IDstClient

The client which sends audio stream to. The client ID is returned by AddClient.

(27): BOOL UnForward(long ISrcClient, long IDstClient);

UnForward one audio stream received from one client to another client.

If you want to unforward from one client to several other clients, you need call this function several times.

Return Value:

TRUE means success and FALSE means failed

Parameters:

ISrcClient:

The client which receives audio stream from. The client ID is returned by AddClient.

IDstClient

The client which sends audio stream to. The client ID is returned by AddClient.

(28): BOOL UnForwardAll();

Remove all forward relationship

Return Value:

TRUE means success and FALSE means failed

Parameters:

None

(29): BOOL ListenToClient(long IClientID)

Specify the server can hear one of the client's audio stream.

Return Value:

TRUE means success and FALSE means failed

Parameters:

IClientID:

The client ID which is returned by AddClient.

3、 Demo

The following VC code demonstrates how to send and receive video between two computers which IP is "10.110.0.10" and "10.110.0.20"

(1) Start send video to computer which IP is "10.110.0.20"

```
if( ! m_netvideo.HaveVideoCapture() )
```

```

{
    AfxMessageBox("Have no capture device!");
    return;
}
m_netvideo.SetLanguageVersion(2); //English
m_netvideo.SetUserName("Tom");

m_netvideo.SetBandwidth(400000, 1 ); //COMPRESS
m_netvideo.SetSendFrameRate(10);

m_netvideo.SetVideoInFormat("CIF");
m_netvideo.SetVideoOutFormat("CIF");

BOOL bRet = m_netvideo.InitCapture(0);
if( bRet == false )
{
    int nErr = m_netvideo.GetErrorID();
    m_netvideo.GetErrorMessage(nErr);
    return;
}

m_netvideo.SetLocalIPAddress( (LPCTSTR)"10.110.0.10", 3000 );
m_netvideo.SetRemoteIPAddress( (LPCTSTR)"10.110.0.20", 3000, 0);
bRet = m_netvideo.SendVideo();
if( bRet == false )
{
    int nErr = m_netvideo.GetErrorID();
    m_netvideo.GetErrorMessage(nErr);
    return;
}

```

- (2) Recv video from computer which IP is "10.110.0.10"

```

m_netvideo .SetBandwidth(400000, 2); // DECOMPRESS

m_netvideo.SetVideoInFormat("CIF");
m_netvideo.SetVideoOutFormat("CIF");

m_netvideo.SetLocalIPAddress( (LPCTSTR)"10.110.0.20", 3000 );
m_netvideo.SetRemoteIPAddress( (LPCTSTR)"10.110.0.10", 3000, 0 );
m_netvideo.RecvVideo();

```

- (3) Stop send to computer which IP is "10.110.0.20"

```

m_netvideo.StopSend();
m_netvideo.CloseCapture();

```

- (4) Stop receive from computer which IP is “10.110.0.10”
m_netvideo.StopRecv();

4、 How to buy

The release version of EasyVideo will costs 1000\$ and the source code of EasyVideo will costs 20000\$. If you want to buy, please go to <http://www.lht2000.com> or directly goto <http://secure.emetrix.com/order/product.asp?PID=14759354> .

The author's email is support@lht2000.com