

# Documentation

Summary of the SmartVMD methods and properties:

## Video Sources

Property NumOfVideoSources: (int, read only)

Get the number of video sources.

Property VideoSource (int)

Select the video source to use / Get the active video source.

## Video Display

Method StartMonitoring(BOOL) - Start the video display.

Method StopMonitoring() - stop the video display.

IsMonitoring() - return TRUE if the video display is on.

## Motion detection

Property DetectMotion (BOOL)

Set motion detection on and off.

Get motion detection on /off status

Changing Motion detection parameters:

Motion detection parameters can be changed in the properties dialog or directly by setting the appropriate methods:

Method SetDefaultParm()

Set all VMD parameters to default values.

Method ImageScaling (int , legal range: 0 - 2)

Set the downscaling of the video image for the motion detection algorithm. The default value is 1 so that the image is scaled one time down.

More down scaling means less CPU load and less detection sensitivity. It is always best to choose downscaling if the VMD sensitivity is adequate after the downscaling.

Method MinNumOfDetectorsForAlert (int, legal range:1 - 12)

Sets the minimal number of detectors that should be on before motion detection alert is activated. As this number is decreased, smaller objects will be detected. (In the property dialog it appears as Min object size)

Method MinimalContrastToDetection (int, legal range: 0 - 10)

Sets the minimal object to background contrast for detection. As this number is increased, the detection sensitivity will be reduced.

Method MinimalVelocityForDetection (int, legal range:1- 20)

Set the minimal object velocity for detection. Higher value mean more sensitivity for slow objects.

Method AntiFlickerFilter (Boolean)

Set / Reset the anti flicker filter. This filter reduce dramatically the sensitivity for video flickering and other noise sources in the video. However, it may reduce the detection in cases of moving objects with a very high speed and low contrast to background.

{mospagebreak title=Save Frame/s to file}Save video or snapshot to file:

Method SaveCurrentImage(FileName As String)

Save a snapshot of the video source to a filename in BMP format.

Method SaveCurrentImageWithAutoFileName(FileName As String \*)

Save a snapshot of the video source to a filename. The name is given automatically by the controller and it is composed of the current date and time. This method can be used as a response to the motion detection event. In this case, every motion detection event will yield a snapshot BMP file where its name contains the date and time. The method returns the video file name created by the controller.

Save video clip to AVI file.

Method StartVideoSaving(FileName As String)

Start recording the video to FileName. The video will be recorded in AVI format. The compression is set in the properties dialog. If it was not set before, A dialog box will appear and let the user choose the compression method. This setting will be kept until the user change it from the properties dialog.

Method StartVideoSavingWithAutoFileName(FileName As String \*)

Start saving the video stream to an AVI file. The name is given automatically by the controller and it is composed of the current date and time. This method can be used as a response to motion detection event. In this case, every motion detection event will yield a video clip. The length of the video clip can be changed. see the VideoClipLen property.

The method returns the video file name created by the controller.

Method StopVideoSaving()

Stop saving the video stream. The saving is stopped automatically when the recording time exceeded the value of VideoClipLen property. However, you can stop video saving anytime using the StopVideoSaving() method.

Property VideoClipLen (int)

Controls the length in second of the video clip that will be saved to file. When StartVideoSaving or StartVideoSavingWithAutoFileName is invoked, a video clip will be saved to disk. The recording will be automatically ended after the number of seconds in VideoClipLen is exceeded.

{mospagebreak title=Motion Detection mask}

Using motion detection mask

video mask is used to determine which areas in the video image will be masked from motion detection.

The mask is composed of cells . Every cell can be masked on and off.

You can use an existing mask, create a new one, load and save masks to disk.

Upon creating a new mask, the cell size is determined by the user.

Property ShowMask (Boolean)

Determine whether the mask is visible or not.

Method CreateMask(int sizeX, int sizeY)

Create a new mask with cell size of (sizeX, sizeY).

Method ResetMask()

Reset all the cells in the mask.

Method SetMask(int X, int Y, int val)

Set or reset the cell (X,Y)

Method SaveMaskToFile(FileName As String)

Save the mask to file specified by FileName..

Method LoadMaskFromFile(FileName As String)

Load the mask from file specified by FileName.

{mospagebreak title=Dialog box and Events}Open dialog box

Method OpenCameraPropertiesDialog()

Open the camera properties dialog allowing the user to control camera parameters.

Method OpenPropertiesDialog.

Opens the main properties dialog. The dialog can be opened as well during the development phase like any standard properties dialog.

Properties changed from the development environment will be automatically saved in the project.

Property CurrentDirectory(DirName As String)

Set and Get the working directory for saving snapshot and video clip.

Property VideoRate(int)

Set and Get the video frame rate operation of the system.

As this value is decreased, less CPU resources are used.

Property VideoFormat(BITMAPINFOHEADER \* as VARIANT)

returns a pointer to BITMAPINFOHEADER structure containing the image format of the video source.

Events created by the SmartVMD control

-

OnMotionDetection(int x1, int y1, int x2, int y2)

This event is fired when motion is detected. The parameters defines the moving object delimiters.

-

AfterMotionDetection()

This event is fired at the end of motion detection.

-

AfterSavingVideoToDisk()

Fired when video clip recording was finished

-

NextFrameReady(unsigned char \* as VARIANT)

Fired after processing a video frame. The event transfers a pointer to the current video frame.

With this method, the programmer can access the video source images. The format of the images is give in the VideoFormat Property.  
{mospagebreak title=Version 1.1 additions} Version 1.1 additions  
Audio

The ActiveX can now record also Audio in the AVI file.

You can choose which device to record from (Web cam mic, sound card mic etc&hellip;)

Property RecordVoiceInAvi (Boolean)

Determine whether to record the audio.

Property NumOfAudioSources (int, read only)

Returns the number of audio devices on the current computer.

## Property AudioSource (int)

Get the current audio device selected.

Set which audio device should be used.

## Image

Method GetImageInformation (ByRef width As Integer, ByRef height As Integer, ByRef color\_depth As Integer)

Return the size and the color depth of the current frame.

## Mask

Method GetMaskCellSize (ByRef cx As Integer, ByRef cy As Integer)

Return the width and height of the current mask base unit.

## Detection

Property DelayBeforeNewDetectionEvent (int)

Determines the minimal silent time before raising a new detection event. (in seconds).

This sets a delay of x seconds before the ActiveX closes an alert (AfterMotionDetection event).

This Property is intended to avoid creating a few alerts from close events.

## ActiveX Activation

Property ActivationStatus (int, read only)

Return the status of the ActiveX activation.

The property values are:

0 = Bad Activation

1 = Expired version

2 = Trail version

3 = Single channel version.

4 = Multi channel version.

## Update method parameter

In the OpenPropertiesDialog function added parameter that determines the properties tab to be shown. OpenPropertiesDialog(OpenPage parameter) 0 &ndash; Show All tabs. Other - Show the Register tab only.