AVPro Movie Capture
Unity Plugin

Version 1.4 - Released 15 March 2012

Real-time capture of Unity Camera to AVI file.

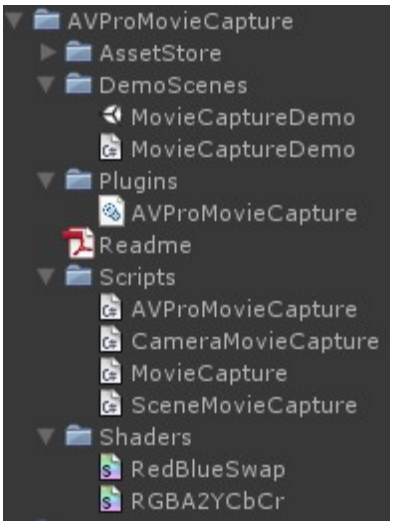
Contents

1. Introduction

“AVPro Movie Capture” is a plugin for Unity that allows recording video directly to disk as an AVI file.

Whether we’re just testing out an idea for a new effect, playing around with some parameters or producing demos for our clients, we often find it useful to be able to quickly and easily capture a video from within Unity. Previously we used screenshots and captured videos using tools like Fraps, however we wanted something completely integrated into Unity and so AVPro Movie Capture was born.

The asset package consists of the following elements:



- **MovieCaptureDemo.unity** - A simple demo scene showing how to use the MovieCapture component.
- **AVProMovieCapture.dll** - The main plugin DLL that talks to DirectShow.
- **AVProMovieCapture.cs** - Wrapper interface to access capture functions in the DLL.
- **MovieCapture.cs** - Base class
- **CameraMovieCapture** - Drag ‘n drop component to allow easy capturing from a camera but not GUI.
- **SceneMovieCapture** - Drag ‘n drop component to allow easy capturing of the entire scene including GUI.
- **RedBlueSwap.shader** - Internal shader used to swap red and blur channels.
- **RGBA2YCbCr.shader** - Internal shader used to convert RGBA to YCbCr YUY2 format.

2. Installation

System Requirements:

- Unity Pro 3.4 and above.
- The plugin only supports Microsoft Windows.
- Windows XP SP3 and higher.
- Codecs for any video formats you want to record to.

Installation Steps:

1. Import the unitypackage file into your Unity project.
2. Move the “Plugins” folder into the root of your project.

Note: Ensure you have the relevant video codecs installed for formats you want to record to.

3. Features

- High performance.
- Easy to use.
- Use any video codec you want.
- Can capture alpha channel for creating transparent videos.
- Records audio from windows recording device.
- Unicode file name support.
- Works in the editor and also in stand-alone builds.

Useful for:

- Games - recording gameplay.
- Development - easily record videos for clients or to share online.
- Testing - create videos of bugs to aid debugging.
- Interactive Installations - making videos of each user session.

4. Components

There are 2 components that come with this plugin.

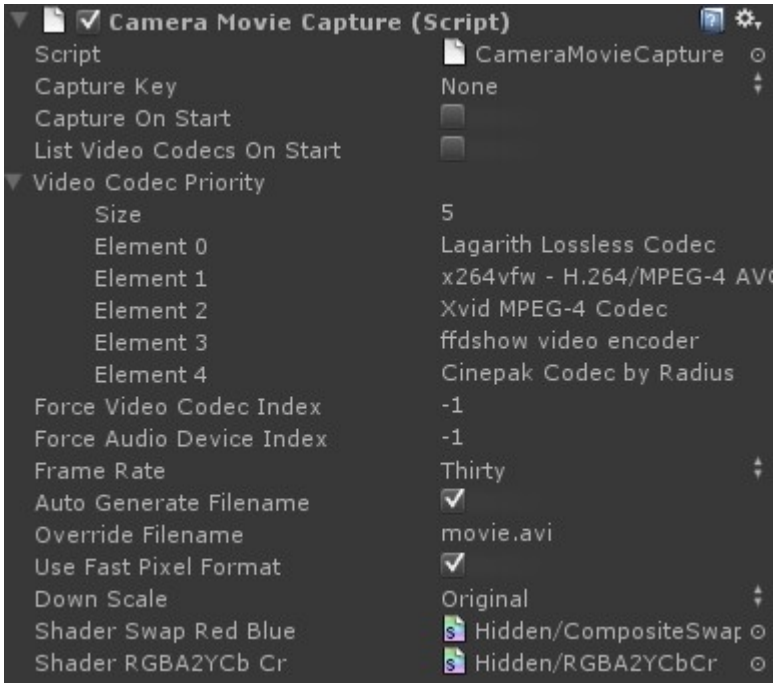
CameraMovieCapture & SceneMovieCapture

In Unity 3.5 and above SceneMovieCapture is faster than CameraMovieCapture as it accesses the graphics API directly. For older versions of Unity, CameraMovieCapture has the best performance, however it cannot capture the GUI.

CameraMovieCapture Component

This component is attached to a camera and captures the 3D output from that capture. It cannot capture GUI (for this use the SceneMovieCapture component). Simply drag the “CameraMovieCapture” script to the camera you want to capture or select it from

the “AVPro” components menu. Make sure “CameraMovieCapture” component is the last component on your camera.



The camera component
The component can be set to start/stop recording when a specific key is pressed, or it can be set to start recording when the application starts.

The option “Use Fast Pixel Format” translates internally to using 1 of 2 pixel formats: RGBA32 (standard) or YCrCb422_YUY2 (fast).

Using YCrCb422_YUY2 is usually faster to encode as it is half the size of RGBA32 and also usually the native format for most encoders. RGBA32 should be used when compatibility with a codec is needed and when lossless encoding is required.

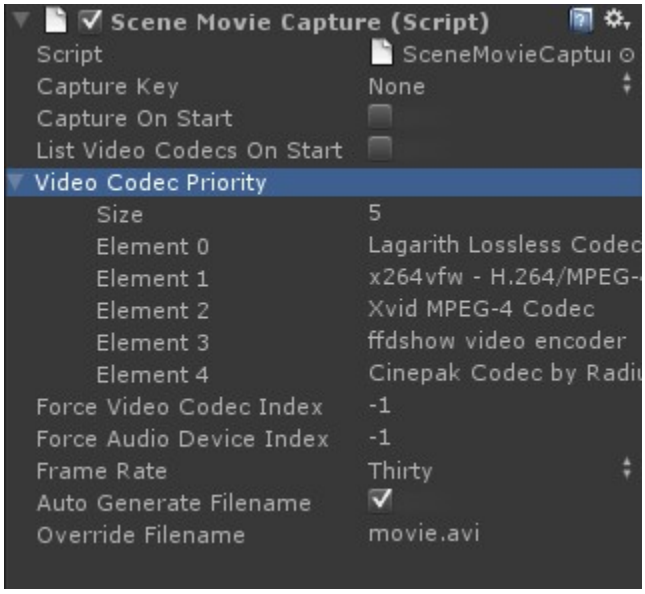
The option “List video codecs on start” will print the list of available video codecs installed on the machine to the console. This is useful to be able to know which codecs you can use.

“Video codec priority” is an array of strings the user is free to edit. Each string is the name of a codec. When the component runs it will try to select the codec from the the list that it finds first on the system.

“Force video codec index” will override the “Video codec priority” list and allows the user to easily select a codec from the list of system codecs. This index is not an index into the above list of codec names but into the list returned in the console. This value is default to “-1”
which means ignore and use the priority codec list.

SceneMovieCapture Component

Unlike CameraMovieCapture this component captures the entire scene including GUI. It must be added to a camera object in the scene.



The options are the same as CameraMovieCapture except some of the options like “Down Sample” and “Use Fast Mode Format” are missing. This is simply due to the way Unity allows us to capture data.

5. Tips

For best results we recommend:

1. Install and use the Lagarith video codec. It is a free lossless codec with excellent performance. It does produce quite large files though so you may need to convert it to another format before sharing/uploading.

2. If you need to convert videos from one codec to another use VirtualDub.

3. Install and use the X264 VFW video codec. It’s fast and produces videos of a very small file size.

4. If a specified codec couldn’t be found, a warning is generated and uncompressed video will be produced.

6. Custom Usage

If you want to go beyond the MovieCapture component you can access the functionality of the AVProMovieCapture DLL directly or edit MovieCapture.cs to make your changes.

The DLL has these functions which are wrapped in AVProMovieCapture.cs:

bool Init();
Global initialisation for the plugin. Returns false if unsuccessful.

void Deinit();
Global deinitialisation for the plugin.

int GetNumAVIVideoCodecs();
Returns the number of video codecs on the system.

bool GetAVIVideoCodecName(int index, StringBuilder name);
Returns true if successful. The name of the system codec at index is returned as a StringBuilder. StringBuilder should be created with size 512.

int GetNumAVIAudioInputDevices();
Returns the number of audio input devices on the system.

bool GetAVIAudioInputDeviceName(int index, StringBuilder name);
Returns true if successful. The name of the system audio inptu device at index is returned as a StringBuilder. StringBuilder shoudl be created with size 512.

int CreateRecorderAVI(string filename, uint width, uint height, int format, int videoCodecIndex, int audioInputDeviceIndex);
Creates a recorder instance to generating AVI files. An integer is returned which is a unique value specific to this instance of the recorder.

void Start(int handle);
Starts recording. Handle is the handle of the recorder instance.

bool IsNewFrameDue(int handle);
Let's us know whether the encoder is ready for another frame.

bool EncodeFrame(int handle, System.IntPtr data);
Sends frame to be encoded. "data" points to an array of widthxheight and with a bitdepth of 32 for RGBA32 videos and 16 for YUY2 videos.

void Stop(int handle);
Stops recording. Handle is the handle of the recorder instance.

void FreeRecorder(int handle);
Releases the instance of the recorder.

7. FAQ (Frequently Asked Questions)

1. **Does this plugin record the audio from Unity?**

   This is only supported if your soundcard supports exposing your speakers as a recording device. You can check this in Windows 7 by going to "Sound" in the Control Panel. Go to the Recording panel and right-click. Make sure all devices are set to show. If you see something like "Speakers" in the list then you will be able to record the audio from Unity.

2. **Why isn't the speed as fast as FRAPS or another screen capture tools?**

   Our plugin is pretty fast but we have to use Unitys scripting engine to grab the bytes for the video frames and this takes about 80% of our capture time. We hope in an upcoming version of Unity we'll be able to capture the data directly in native code.

3. **How do I fix the error: "DLLNotFoundException"?**

   You need to move/copy the "Plugins" folder from your "AVProQuickTime" folder into the root of your folder structure. This means the "Plugins" folder should be moved to your "Assets" folder. Unfortunately this is a limitation in the way Unitys Asset Store handles plugins.

4. **Where are my movie captures stored?**

   By default the components auto-generate a filename each time you run a capture. These files are stored in the root of your project (the folder above "Assets"). You can always disable auto-generation of filenames in the component and specify your own file name and locatin for a capture.

8. Version History

Version 1.4 - Thursday 15 March 2012

- Much faster capturing due to new Unity 3.5 native API features.

Version 1.3 - Friday 17 February 2012

- Added audio for testing audio recording.
- Autodetection of loopback audio devices.

Version 1.2 - Saturday 4 February 2012

- Added ability to capture GUI.
- Added audio capture.
- Added resizing to half, quarter, eighth resolution.
- Improved capture performance and smoothness by only preparing the frame capture data when the encoder requires it.
- Automatic disable of vsync helps performance.
- Rounding to multiple of 4 resolution to hep codec compatability.
- Added ability to set target frame rate (15, 24, 30, 60).
- Added ability to set own file name.
- Video codecs can now be configured.
- Fixed various minor bugs.

Version 1.1 - Tuesday 24 Janurary 2012

- Removed Vista/Win7 dependency (WMV).

Version 1.0 - Thursday 17 January 2012

- Initial release submitted to Asset Store.

9. Support

If you are in need of support or have any comments/suggestions regarding this product please contact us.

Website: http://www.renderheads.com/contact/
Email: contact@renderheads.com

If you are reporting a bug please include any relevant files so that we may remedy the problem as fast as possible.
