AVPro Movie Capture
Unity Plugin

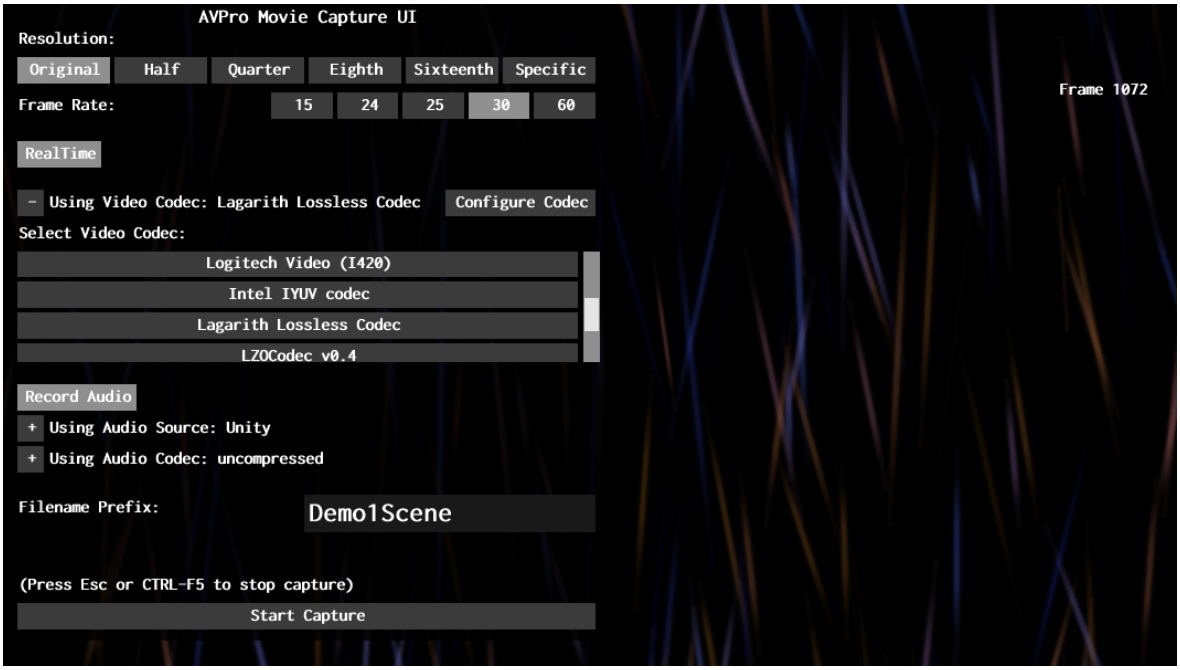
Version 2.52 - Released 27 February 2014

Real-time capture of Unity Camera to AVI/MP4 file.

Contents

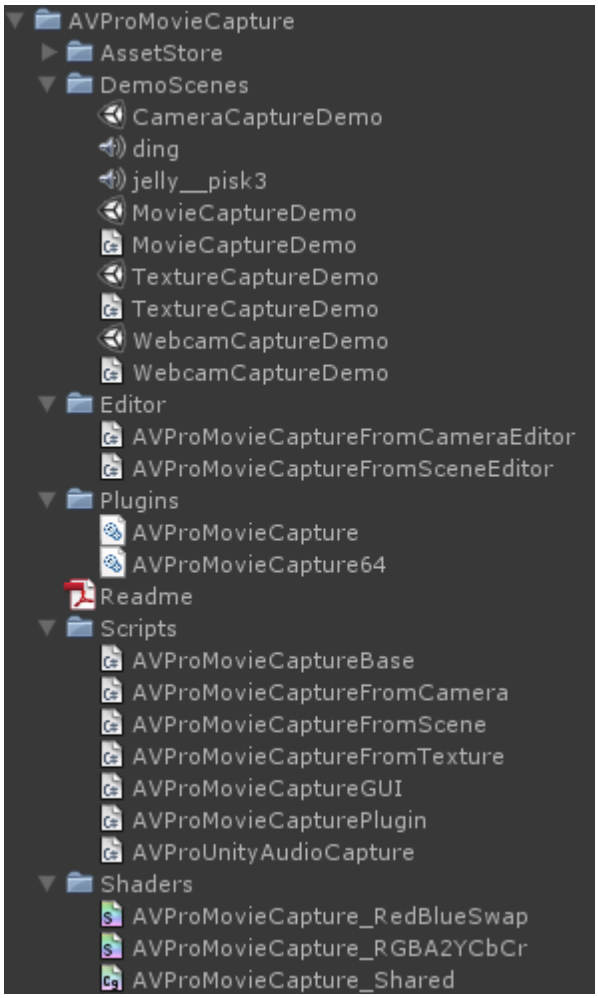
1. Introduction

“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:



- DemoScenes
    - **CameraCaptureDemo.unity** - A simple demo showing how to use the AVProMovieCaptureFromCamera component.
    - **MovieCaptureDemo.unity** - A simple demo scene showing how to use the AVProMovieCaptureFromScene component.
    - **TextureCaptureDemo.unity** - A simple demo scene showing how to use the AVProMovieCaptureFromTexture component.
    - **WebcamCaptureDemo.unity** - A simple demo scene showing how to use the AVProMovieCaptureFromTexture component.
- Plugins
    - **AVProMovieCapture.dll** - The main plugin DLL that talks to DirectShow.
    - **AVProMovieCapture64.dll** - The main 64-bit plugin DLL that talks to DirectShow.
- Scripts
    - **AVProMovieCapturePlugin.cs** - Wrapper interface to access capture functions in the DLL.
    - **AVProMovieCaptureBase.cs** - Base class
    - **AVProMovieCaptureGUI.cs** - Helper component that displays a GUI exposing the capture options of AVProMovieCaptureBase
    - **AVProMovieCaptureFromCamera.cs** - Drag ‘n drop component to allow

easy capturing from a camera but not GUI.
    - **AVProMovieCaptureFromScene.cs** - Drag ‘n drop component to allow easy capturing of the entire scene including GUI.
    - **AVProMovieCaptureFromTexture.cs** - Drag ‘n drop component to allow easy capturing of a dynamic texture.
    - **AVProUnityAudioCapture.cs** - Drag ‘n drop component for capturing Unity audio into a buffer for saving to the AVI file.
- Shaders
    - **AVProMovieCapture_RedBlueSwap.shader** - Internal shader used to swap red and blue channels.
    - **AVProMovieCapture_RGBA2YCbCr.shader** - Internal shader used to convert RGBA to YCbCr YUY2 format.

2. Features

- High performance.
- Easy to use.
- Use any video codec you want.
- Real-time capture and offline rendering.
- Works in the editor and also in stand-alone builds.
- Can capture alpha channel for creating transparent videos.
- Records audio directly from Unity or from a Windows recording device.
- Unicode file name support.

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.

3. Installation

System Requirements:

- Unity Pro 3.5 and above.
- The plugin only supports Microsoft Windows (32 and 64-bit builds)
- 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. See Codec Setup below.

4. Codec Setup

There are many codecs out there, each with their own pros and cons. Some codecs are great for real-time encoding, some are lossless and some achieve tiny file sizes so it is important to consider which codec you use and configure them correctly for your needs.

Codecs we recommend:

Codec	Real-time	File Size	Alpha Channel
Lagarith	Yes	Large	Yes (after tweaking)
x264	Yes (after tweaking)	Small	No
Xvid	Yes	Medium	No

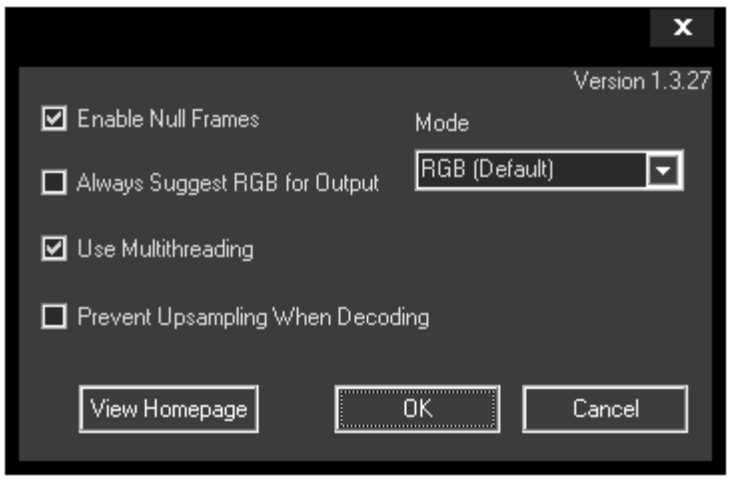
4.1 Lagarith Codec

website: http://lags.leetcode.net/codec.html

Lagarith is a great general purpose codec. It's fast enough for real-time (due to great multi-threading) and lossless. Naturally the files it generates are very large and not suitable for sharing over the net. We use Lagarith as an intermediate codec for real-time capturing and then re-encode to something else like MP4 offline.

Always check your codec settings. You can do this directly in the plugin via the Configure button, or in 3rd party software like Virtualdub (video>compression menu). .Recommended settings:

- Enable "Null Frames"
- Enable "Use Multithreading"
- Set Mode to RGB or RGBA if you need to capture alpha channel



Recommended Lagarith Settings

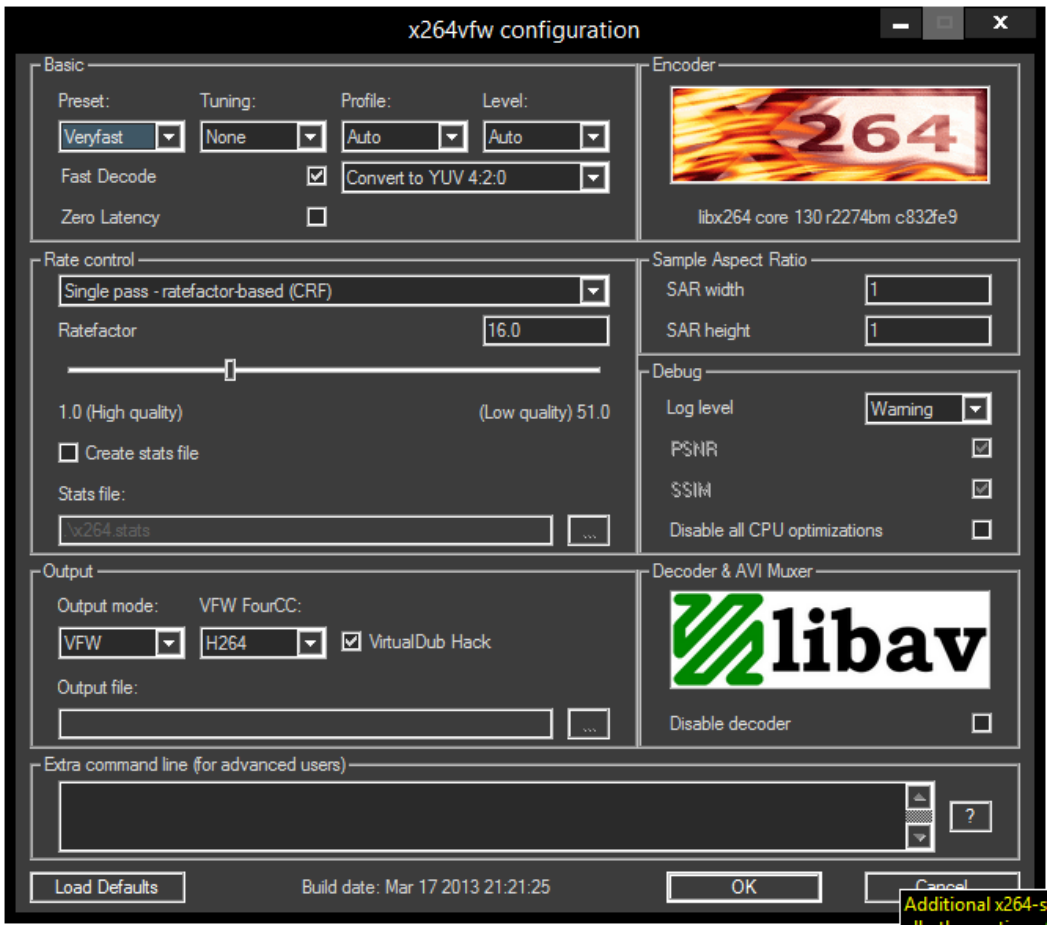
4.2 x264 Codec

website: http://sourceforge.net/projects/x264vfw/

x264 is a highly tunable codec and suits almost any need. By default it's set up for off-line processing which produces tiny files but generally uses way too much CPU for real-time capture. We tweak x264 for real-time capture and use it to directly generate video files suitable for sharing. x264 can also be used with an MP4 muxer to generate MP4 files instead of AVI files (see FAQ).

Always check your codec settings. You can do this directly in the plugin via the Configure button, or in 3rd party software like Virtualdub (video>compression menu). .Recommended settings:

- Preset: Fast/Veryfast
- Enable "Fast Decode"
- Set "Ratefactor" to your desired quality level
- Enable "VirtualDub Hack"

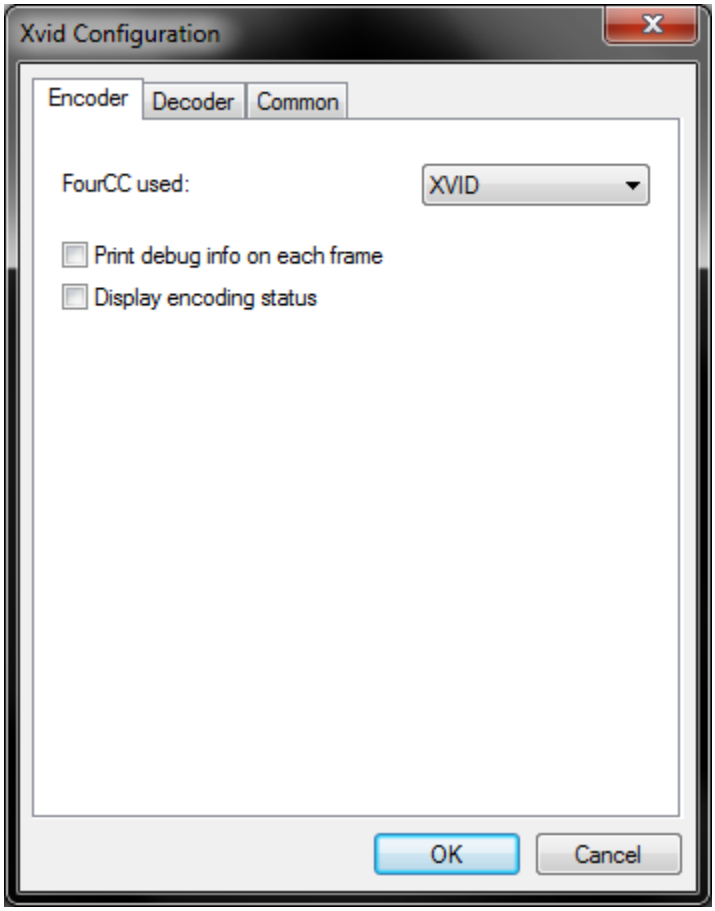


Recommended x264 settings for real-time capture

4.3 Xvid Codec

website: http://www.koepi.info/xvid.html

If you want to use Xvid make sure to disable the option “Display encoding status” under “Other options”.

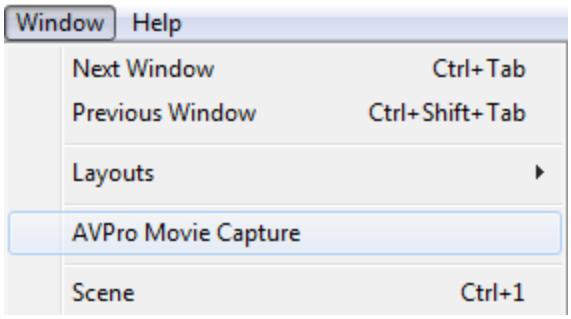


Recommended Xvid settings for real-time capture

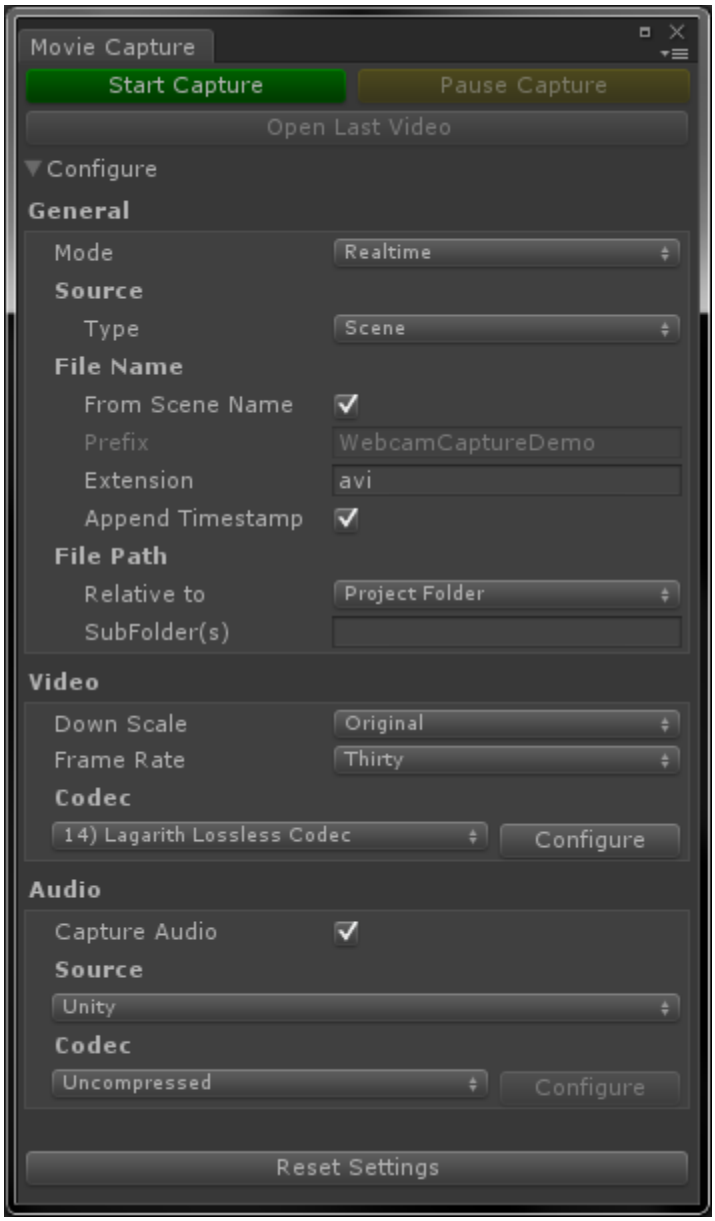
5. Usage

5.1 In-Editor Capture

The editor components allow you to quickly and easily capture videos directly from inside the editor without modifying your scene. Simply open the AVPro Movie Capture editor window:



You can now add this UI panel to your editor layout to allow one-click video captures, or open and close it as needed.



This panel allows you to configure your recording options and codecs. All settings are remembered between sessions so once it has been configured once you only need to press

the “Start Capture” button.

Caveat:
Currently selecting “Unity” as the audio source requires you to manually add the AVProUnityAudioCapture component to the main camera of your scene (the one with the main Audio Listener) before hitting Play on your scene.

5.2 In-Game Capture

You can add one of the components (eg AVProMovieCaptureFromScene / AVProMovieCaptureFromCamera) to your scene and trigger them to record directly from your game. This works in the editor and standalone.

5.3 Offline Rendering

For visuals that don’t require real-time input (cut-scenes, procedural animations, input playback systems) you have the option to record in offline/non-realtime mode. This mode allows you to capture every frame of animation even if the playback runs very slowly and allows you to capture to any target frame rate.

For example you may have a sequence that runs at 5fps at the highest quality. You can use offline recording to render this to a 60fps video.

You may also have an animation that renders at 500fps and you need to compress it to video as quickly as possible. With offline recording it will record as fast as the video compressor will run.

6. Components

There are 2 main components that come with this plugin.

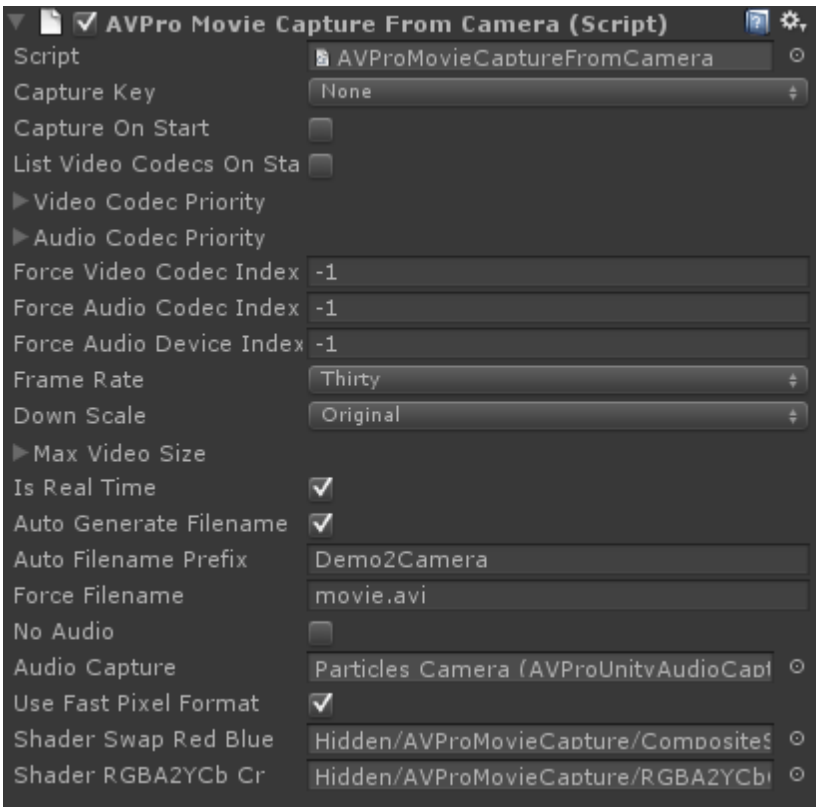
AVProMovieCaptureFromCamera & **AVProMovieCaptureFromScene**

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

AVProMovieCaptureFromCamera Component

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

select it from the “AVPro Movie Capture” components menu. Make sure “AVProMovieCaptureFromCamera” component is the last component on your camera.



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.

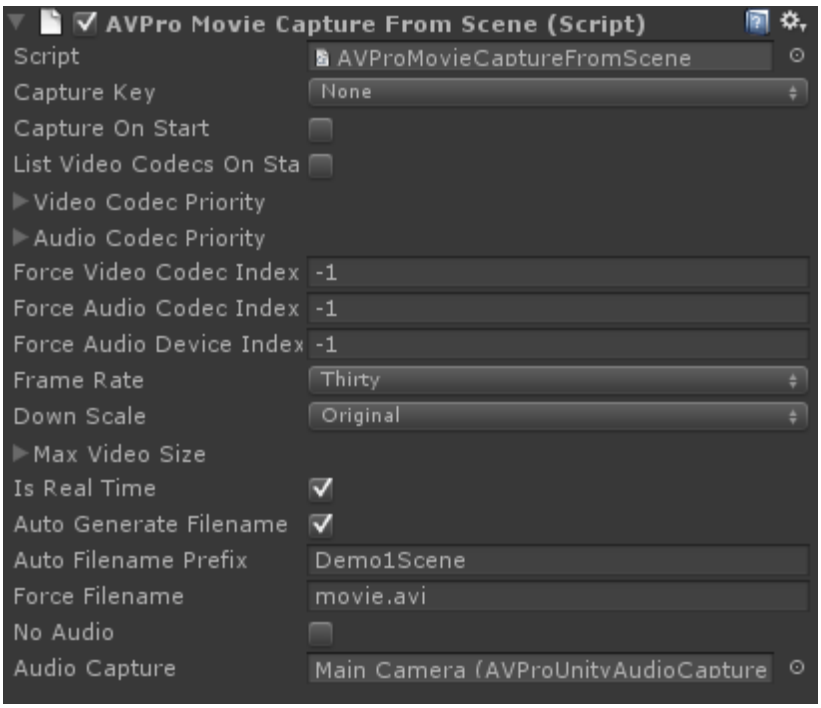
The option “List video codecs on start” will print the list of available video codecs installed on the machine to the console. This is useful to be able to know which codecs you can use.

“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.

AVProMovieCaptureFromScene Component

This is the preferred component for capturing as it grabs directly from the GPU buffers bypassing Unity (requires Unity 3.5) and is thus faster. Unlike AVProMovieCaptureFromCamera this component captures the entire scene including the GUI. It must be added to a camera object in the scene.



The options are the same as AVProMovieCaptureFromCamera .

Components Comparison

Component	Native	Unity GUI	YUV
AVProMovieCaptureFromScene	Yes	Yes	No
AVProMovieCaptureFromCamera	No	No	Yes

Native - Support for native GPU capturing as opposed to using Unity's Texture2D.GetPixels32(). Native is the fastest method of capturing.

Unity GUI - Whether the capture will include the Unity GUI

YUV - Captures usually happen in RGBA32 mode. Having YUV support allows the image to be converted to YUV 4:2:2 format (half the bandwidth of RGBA32) prior to encoding. This can be faster depending on the codec.

7. Custom Usage

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

The DLL has the following functions which are wrapped in AVProMovieCapturePlugin.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, bool isTopDown, 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.

void Pause(int handle);
Pauses recording.

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.

8. 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. You need to configure the Lagarith codec and enable multi-threading and null frames.

2. If you need to convert videos from one codec to another use VirtualDub or FFMPEG from the command-line.

3. Install and use the X264 VFW video codec. It is fairly fast and produces videos of a very small file size. If you're recording at a high resolution though you'll need a very fast CPU for this codec.

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

5. For best performance create a build and do your captures from the build running in full-screen mode.

6. Capture a resolution that has a width that is a multiple of 16. This should allow best cache usage and SIMD instructions to be used during memcpy.

9. FAQ (Frequently Asked Questions)

1. **How do I update to the latest version of this plugin?**

   If you have purchased this plugin from the Unity Asset Store then you simply login to the store and check if there is a new version to update to.

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

   Yes, if you use the AVProUnityAudioCapture component you can record audio directly from Unity.

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

   You need to move/copy the “Plugins” folder from your “AVProMovieCapture” 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. **How do I fix the error: “DLLNot FoundException” where it’s trying to load the 64-bit DLL in the editor?**

   Sometimes Unity gets confused and will try to load the 64-bit DLL in the editor (which is only 32-bit). To fix this:

   - Open Build Settings
   - Select Web Player platform and press Switch Platform
   - Select PC and Mac Standalone (with Target platform set to Windows not Windows 64-bit) and press Switch Platform

5. **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 location for a capture.

6. **How do I record in-game audio and microphone audio at the same time?**

   Currently this plugin only supports recording audio from a single Windows audio device. There is a “trick” you can use though. In Windows 7 (and perhaps Vista) you may be able to set your microphone to play through the speakers by going to:

   Control Panel -> Sound -> Recording -> Select your microphone -> right click -> Properties -> Listen -> check "Listen to this device".

   You should then hear your microphone recording through your speakers. It’s recommended to use headphones during recording to prevent feedback from the speakers into the microphone.

7. **How do I get this plugin working with 64-bit Windows builds?**

   Currently in Unity there is no way to automatically support both 32-bit and 64-bit

plugins at build time so you need to rename the plugin file manually once you've made a build. In your build "plugins" folder simply delete "AVProMovieCapture.dll" and rename "AVProMovieCapture64.dll" to "AvProMovieCapture.dll".

8. **How do I prevent Unity from freezing after doing a recording using the Xvid MPEG4 codec?**

   Open the configuration for the Xvid MPEG4 codec and go to the "other options" page and make sure "Display encoding status" is not selected.

9. **I'm using Autodesk ScaleForm and it's glitching, how do I get it to record properly?**

   ScaleForm doesn't seem to like it when render settings (window size, vsync count) are changed while the app/game is running. This plugin removes vsync during recordings which breaks the ScaleForm rendering. Just run your app/game using a quality settings that doesn't have vsync.

10. **How can I encode to MP4 container file instead of AVI?**

    You need to download the following codec: http://www.gdcl.co.uk/mpeg4/
    Install it from the command-line using "regsvr32 mp4mux.dll" command via an Administrator command-prompt. In the AVPro Movie Capture plugin specify your target file name ending with ".mp4"

    Note that the MP4 container only support certain video and audio codecs (unlike AVI). We recommend using x264vfw video codec and MPEG 3 / AC3 audio codec. You may need to disable audio recording if you aren't using the recommended audio codec.

11. **I have compiled the scripts into a DLL and am now experiencing some unexpected behaviour.**

    Some of our scripts have Unity version-specific preprocessor defines which determine how they compile (eg UNITY_4_0). Usually when you build an external DLL these defines are missing and so the incorrect version of the code can be compiled. You need to add the appropriate compiler defines to your build.

12. **My videos aren't capturing correctly, or they appear upside-down**

    This can happen when Unity is using OpenGL emulation in the editor (which can happen when your built target is set to other platforms such as Android, iOS etc). Check your Edit > Graphics Emulation settings to make sure no emulation is being used. Check your File > Build Settings and for best results set this to "PC, Mac & Linux Standalone" then press "Switch Platform" and restart Unity.

13. **The plugin crashes when it starts, what could be causing this?**

    The first thing the plugin does is enumerates all of the video and audio codecs and audio recording devices installed on your system. It is possible that there's a problem with one of these. Try uninstalling all 3rd party codecs you have installed. Once you have it working you can being reinstalling the codecs again. We have found that the professional version of the Cinepak codec can caused this issue.

14. **How can I create transparent videos with an alpha channel?**

    - Select a video codec that supports alpha channels:
        i. Uncompressed
        ii. Lagarith
        iii. Cineform
    - If you use Lagarith make sure it's configured to Mode=RGBA and enable "Use Multithreading".
    - Use the AVProMovieCaptureFromCamera component, or if you use the Editor mode GUI select "Camera" as the source type. Disable the "fast pixel format" option as this removes the alpha channel.

10. Version History

- **Version 2.x - ?**
    - ← Your suggestion here
    - Add support for specifying regions?
    - Improve code documentation?
    - Support non-stall GPU grabs?
    - Recompile with newer version of VS?
    - Motion blur for offline capture?
    - Super-resolution for offline capture?
    - DX11 resizer?
    - Fix H.264 initial freeze bug?
    - Support floating point audio to integer conversion?
    - Lower recording frame rate uses too much CPU?
    - Add audio delay support?

- **Version 2.52 - Thursday 27 February 2014**
    - Fixed unnamed scene recording bug
    - Removed usage of deprecated Unity functions
    - Added display of output frame resolution to UI
    - Video file now deleted before recording to prevent AVI bloat

- **Version 2.5 - Monday 24 February 2014**
    - Added new trial version
    - Added script to automatically copy the DLL files during installation
    - Added better output path options

    - Added support for more YCbCr formats
    - Improved in-editor GUI to allow capturing from specific camera
    - Various minor fixes to the demos
    - Fixed some script issues on non Windows platforms

- **Version 2.48 - Wednesday 15 January 2014**
    - Fixed crash bug when system has no codecs or devices
    - Minor improvements to UI

- **Version 2.46 - Monday 8 January 2014**
    - Added recording stats (file size and time)
    - Improved recording stats GUI
    - Fixed editor launch crash bug when using non Windows built settings
    - Added OpenGL emulation notes to FAQ for captures appearing upside-down

- **Version 2.44 - Monday 6 January 2014**
    - Fixed editor error message when launching configure dialogs

- **Version 2.42 - Monday 30 December 2013**
    - Fixed y-flip bug
    - Fixed camera capture bug using wrong texture size
    - Fixed DX11 viewport capture bug

- **Version 2.4 - Friday 27 December 2013**
    - Added codec configuration dialog support detection
    - Reduced CPU usage of new EditorWindow widget
    - Improved codec lists in new EditorWindow widget
    - EditorWindow can now play the scene to start recording
    - Improved documentation

- **Version 2.3 - Wednesday 4 December 2013**
    - Added proper fast native DX11 support
    - Scene capture no longer requires a camera
    - Added new EditorWindow widget for easier capturing from Editor
    - Improved documentation for best codec settings

- **Version 2.2 - Thursday 22 August 2013**
    - Added support for writing to MP4 container
    - Added support for Unity 4.2
    - Added automatic DLL swap for 64-bit builds
    - Fixed some material leaks
    - Fixed DX11 colour swap and flip bug

- **Version 2.02 - Monday 18 March 2013**
    - Added Unity 4.1 support
    - Fixed some platform #if issues

- **Version 2.0 - Monday 12 March 2013**
    - Added audio recording directly from Unity
    - Fixed GL.IssuePluginEvent() conflict bug with other AVPro plugins
    - Fixed DX11 recording in Unity 4.0
    - Fixed bug in audio codec listing
    - Renamed and restructured code

- **Version 1.8 - Tuesday 18 December 2012**
    - Added audio codec enumeration
    - Added Unity 4.0 support
    - GUI improved
    - Added more demos

- **Version 1.6 - Thursday 6 September 2012**
    - Added ability to pause movie capture
    - Scene capture resolution can differ from Screen resolution
    - Inspector: displays capture rate and has buttons to control capture
    - Less CPU usage
    - Optimisation: removed software RB channel swap
    - Optimisation: removed per-frame memcpy
    - Optimisation: removed vertical flip
    - GUI layout improved
    - Lots of source cleaning up

- **Version 1.5 - Monday 6 June 2012**
    - Improved smoothness of captures significantly.
    - 64-bit Windows support added.
    - Added GUI to easily set up recordings (taken from previous demo scene).
    - Added code to detect dropped frames during encoding.

- **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 help codec compatibility.
    - 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.

11. Support

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

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

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

1. Unity version
2. Operating system
3. GPU model
4. Rendering API (DX9 / DX11 / OpenGL)
5. AVPro Movie Capture plugin version
6. Screenshot of the bug
7. Output log
8. Which codec you are using
9. Which plugin component you are using (eg AVProMovieCaptureFromScene)
10. A copy of the captured video file (if relevant)

12. About RenderHeads Ltd

RenderHeads is an award winning creative and technical company that has been designing and building cutting edge technology solutions since its formation in 2006.

12.1 Services

- Unity plugin development
- Unity game / interaction / augmented reality development
- Unity consulting

12.2 Our Unity Plugins

AVPro QuickTime

AVPro Windows Media

AVPro Movie Capture

AVPro Live Camera