AVPro Live Camera
Unity Plugin

Version 1.8 - Released 27 August 2013

Brings high-resolution cameras and video capture cards to Unity.

RENDERHEADS
DIGITALLY PERFORMED

Contents

1. Introduction
2. Installation
3. Features
4. Demos
5. Components
6. Scripting
7. Tips
8. FAQ
9. Version History
10. Support
11. About RenderHeads Ltd

1. Introduction

AVPro Live Camera is a plugin that brings high quality cameras and video capture devices to Unity.

2. Installation

System Requirements:

- Unity Pro 3.5 and above.
- The plugin only currently supports Microsoft Windows.
- Windows XP SP3 and higher.

Installation Steps:

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

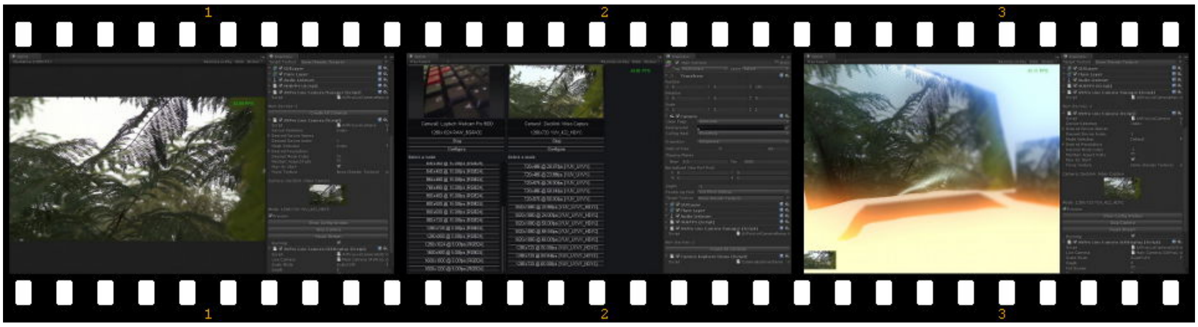
3. Features

- *NEW* Unity 4.0 support, including fast DirectX texture updates
- High performance.
- Easy to use.
- Supports high-end cameras & formats.
- Support for Blackmagic Design’s DeckLink capture cards.
- Support for TV capture cards (eg AVerMedia)
- Supports Unity 3.5 and above.
- Works in the editor and also in stand-alone builds.
- 3 demos included.
- No scripting required (for most usage).

Useful for:

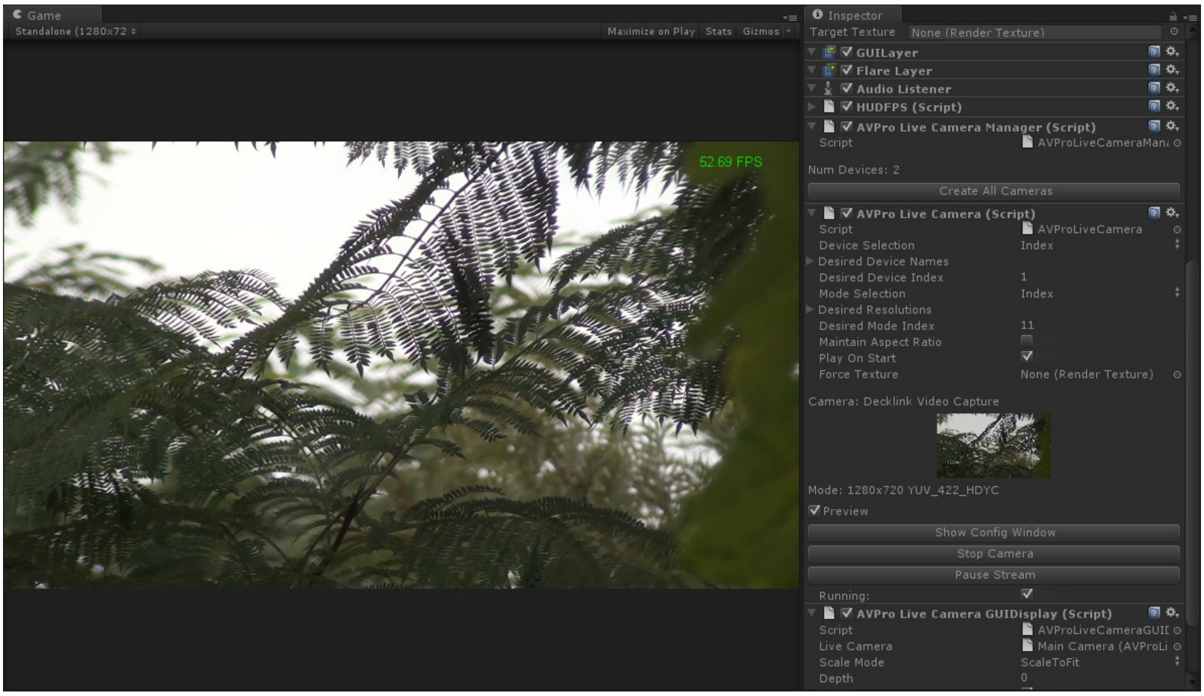
- Interactive installations.
- Scientific research
- Serious games.
- Training and simulation
- Kiosks.
- Video apps.

4. Demos



The plugin includes several demos, some which require scripting and some which don’t.

4.1 Default Camera Demo



Target Texture None (Render Texture)

☑ GUILayer

☑ Flare Layer

☑ Audio Listener

☑ HUDFPS (Script)

☑ AVPro Live Camera Manager (Script)

Script AVProLiveCameraMan:

Num Devices: 2

☑ AVPro Live Camera (Script)

Script AVProLiveCamera
Device Selection Index
Desired Device Names
Desired Device Index 1
Mode Selection Index
Desired Resolutions
Desired Mode Index 11
Maintain Aspect Ratio □
Play On Start ☑
Force Texture None (Render Texture)

Camera: Decklink Video Capture

Mode: 1280x720 YUV_422_HDYC

☑ Preview

Running: ☑

☑ AVPro Live Camera GUIDisplay (Script)

Script AVProLiveCameraGUII
Live Camera Main Camera (AVProLi
Scale Mode ScaleToFit
Depth 0

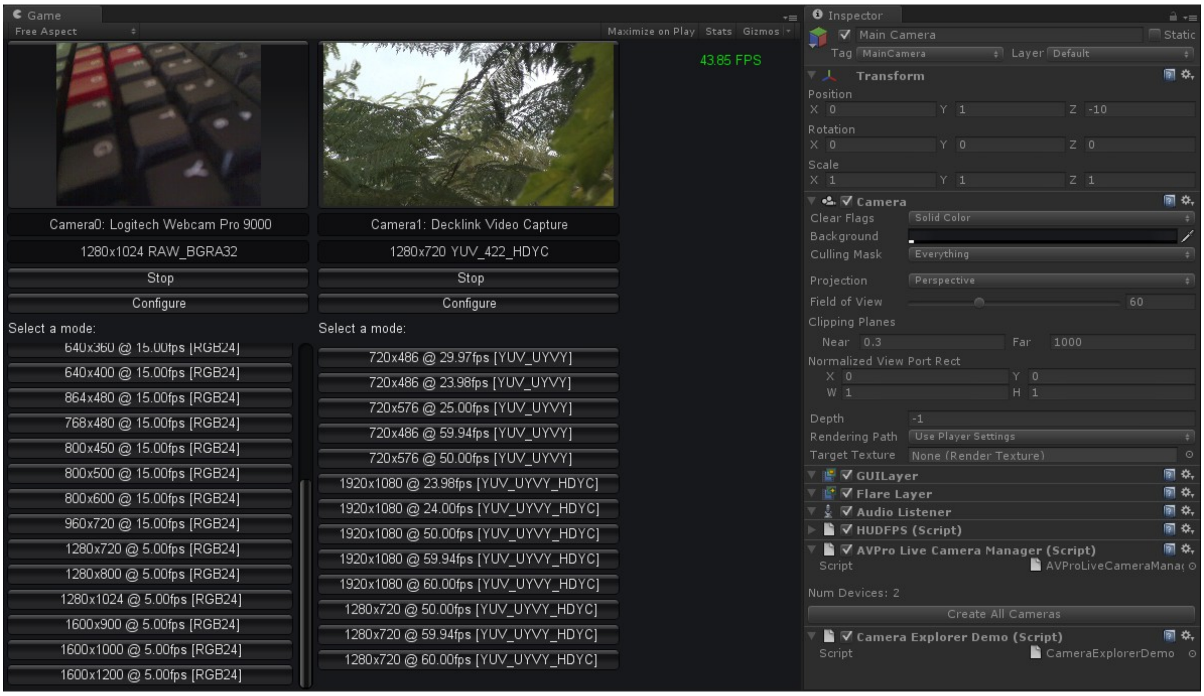
This is the most basic demo as it requires no scripting at all. The demo simply draws the default camera to the screen. To create the demo 3 components were used:

1) **AVProLiveCameraManager** - This component is always required.

2) **AVProLiveCamera -** This component represents the default camera.

3) **AVProLiveCameraGUIDisplay -** Actually draws the camera to the screen via Unity’s GUI system. You could use multiple of these components to render the camera image to multiple places on the screen.

4.2 Camera Explorer Demo


Game
Free Aspect
Maximize on Play
Stats
Gizmos
43.85 FPS
Camera0: Logitech Webcam Pro 9000
1280x1024 RAW_BGRA32
Stop
Configure
Select a mode:
640x360 @ 15.00fps [RGB24]
640x400 @ 15.00fps [RGB24]
864x480 @ 15.00fps [RGB24]
768x480 @ 15.00fps [RGB24]
800x450 @ 15.00fps [RGB24]
800x500 @ 15.00fps [RGB24]
800x600 @ 15.00fps [RGB24]
960x720 @ 15.00fps [RGB24]
1280x720 @ 5.00fps [RGB24]
1280x800 @ 5.00fps [RGB24]
1280x1024 @ 5.00fps [RGB24]
1600x900 @ 5.00fps [RGB24]
1600x1000 @ 5.00fps [RGB24]
1600x1200 @ 5.00fps [RGB24]
Camera1: Decklink Video Capture
1280x720 YUV_422_HDYC
Stop
Configure
Select a mode:
720x486 @ 29.97fps [YUV_UYVY]
720x486 @ 23.98fps [YUV_UYVY]
720x576 @ 25.00fps [YUV_UYVY]
720x486 @ 59.94fps [YUV_UYVY]
720x576 @ 50.00fps [YUV_UYVY]
1920x1080 @ 23.98fps [YUV_UYVY_HDYC]
1920x1080 @ 24.00fps [YUV_UYVY_HDYC]
1920x1080 @ 50.00fps [YUV_UYVY_HDYC]
1920x1080 @ 59.94fps [YUV_UYVY_HDYC]
1920x1080 @ 60.00fps [YUV_UYVY_HDYC]
1280x720 @ 50.00fps [YUV_UYVY_HDYC]
1280x720 @ 59.94fps [YUV_UYVY_HDYC]
1280x720 @ 60.00fps [YUV_UYVY_HDYC]
Inspector
Static
Create All Cameras

Main Camera
Tag MainCamera Layer Default

Transform
Position X 0 Y 1 Z -10
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Camera
Clear Flags Solid Color
Background
Culling Mask Everything
Projection Perspective
Field of View 60
Clipping Planes Near 0.3 Far 1000
Normalized View Port Rect X 0 Y 0 W 1 H 1
Depth -1
Rendering Path Use Player Settings
Target Texture None (Render Texture)

GUILayer
Flare Layer
Audio Listener
HUDFPS (Script)

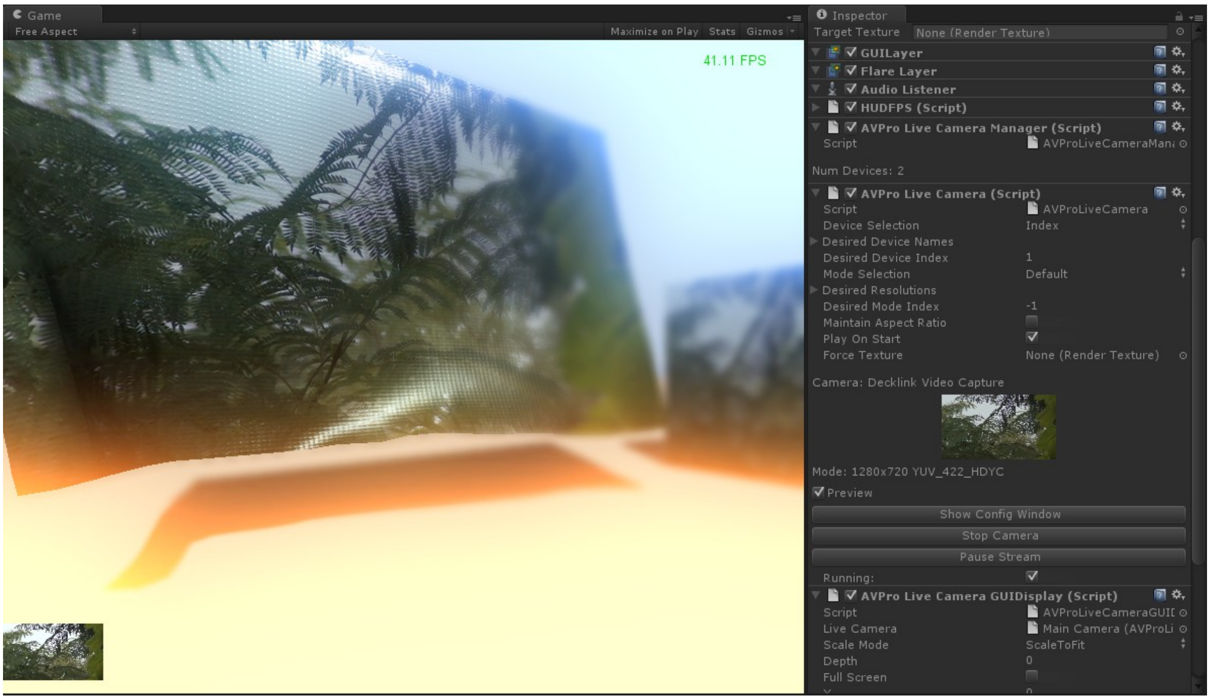
AVPro Live Camera Manager (Script)
Script AVProLiveCameraMana(
Num Devices: 2

Camera Explorer Demo (Script)
Script CameraExplorerDemo

This demo doesn’t use any of the drag ‘n drop components (except the required **AVProLiveCameraManager** component) and demonstrates how easily the a demo can be written using a few lines of scripting. The demo is created by a single 120 line script file, most of which is GUI related.

The demo allows you to explore all of the cameras connected to the system.

4.3 Material Mapping Demo



```
Target Texture          None (Render Texture)
☑ GUILayer
☑ Flare Layer
☑ Audio Listener
☑ HUDFPS (Script)
☑ AVPro Live Camera Manager (Script)
  Script                        AVProLiveCameraMan:
Num Devices: 2
☑ AVPro Live Camera (Script)
  Script                        AVProLiveCamera
  Device Selection              Index
  Desired Device Names
  Desired Device Index          1
  Mode Selection                Default
  Desired Resolutions
  Desired Mode Index            -1
  Maintain Aspect Ratio         □
  Play On Start                 ☑
  Force Texture                 None (Render Texture)
Camera: Decklink Video Capture
Mode: 1280x720 YUV_422_HDYC
☑ Preview
Show Config Window
Stop Camera
Pause Stream
  Running:                      ☑
☑ AVPro Live Camera GUIDisplay (Script)
  Script                        AVProLiveCameraGUII
  Live Camera                   Main Camera (AVProLi
  Scale Mode                    ScaleToFit
  Depth                         0
  Full Screen                   □
```

This demo shows how to integrate AVProLiveCamera into your 3D scene. No scripting is required as you can just use either of the 2 included components:

AVProLiveCameraMaterialApply - Replaces the main texture of a material with the texture from the camera.

AVProLiveCameraMeshApply - Replaces the main texture of all of the materials on a mesh with the texture from the camera.

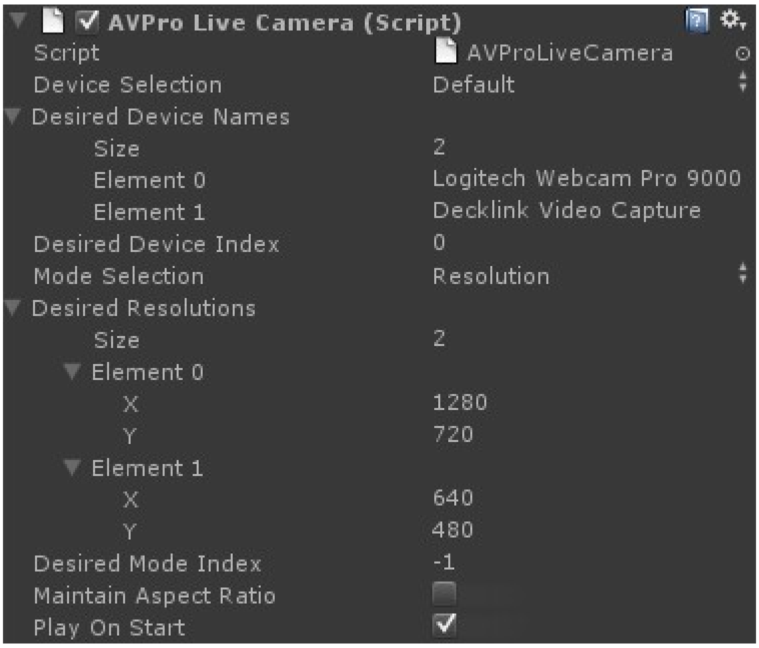
5. Components

This asset includes a number of Unity script components that allow use of the asset without any scripting.

5.1 AVProLiveCameraManager

There must always be exactly one **AVProLiveCameraManager** in your scene when you use this plugin. It is also important that this component starts before the other **AVProLiveCamera** components. There is nothing to configure in this component.

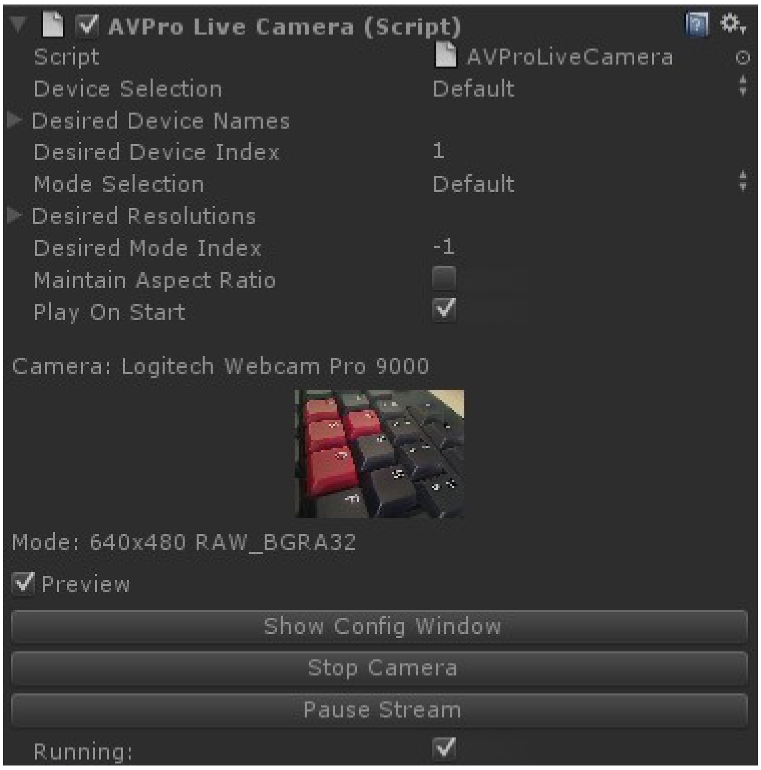
5.2 AVProLiveCamera



☑ AVPro Live Camera (Script)	
Script	AVProLiveCamera
Device Selection	Default
▼ Desired Device Names	
Size	2
Element 0	Logitech Webcam Pro 9000
Element 1	Decklink Video Capture
Desired Device Index	0
Mode Selection	Resolution
▼ Desired Resolutions	
Size	2
▼ Element 0	
X	1280
Y	720
▼ Element 1	
X	640
Y	480
Desired Mode Index	-1
Maintain Aspect Ratio	□
Play On Start	☑

This component represents a single camera device. It has options to allow you to set which camera it uses. The default is just to load the default camera in the default mode, however you can also search cameras by name or by index on your system. The resolution of the camera can also be set either as the default, from a list of resolutions or by index. When using a list of device names or resolutions, it will try to find the closest match, prioritising those at the top of the list.

When the scene is playing this component also previews the currently running camera and allows you to control the stream which is useful for debugging:



AVPro Live Camera (Script)

Script	AVProLiveCamera
Device Selection	Default
▸ Desired Device Names	
Desired Device Index	1
Mode Selection	Default
▸ Desired Resolutions	
Desired Mode Index	-1
Maintain Aspect Ratio	□
Play On Start	☑

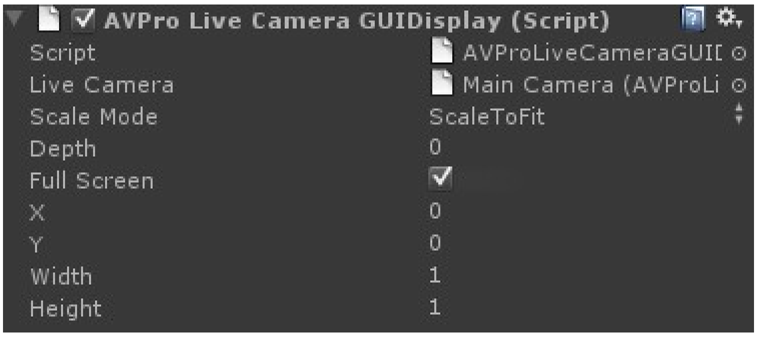
Camera: Logitech Webcam Pro 9000

Mode: 640x480 RAW_BGRA32

☑ Preview

Running: ☑

5.3 AVProLiveCameraGUIDisplay

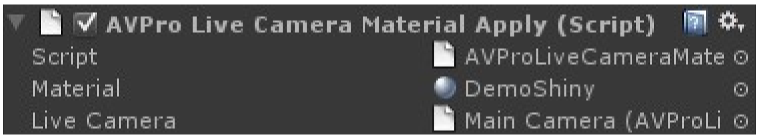


AVPro Live Camera GUIDisplay (Script)

Script	AVProLiveCameraGUII
Live Camera	Main Camera (AVProLi
Scale Mode	ScaleToFit
Depth	0
Full Screen	☑
X	0
Y	0
Width	1
Height	1

This component displays an AVProLiveCamera on the screen using Unity's GUI system. Simply select the **AVProLiveCamera** component you want to display in the "Live Camera" option. Next you can set the placement of the item on the screen or use the fullscreen default.

5.4 AVProLiveCameraMaterialApply

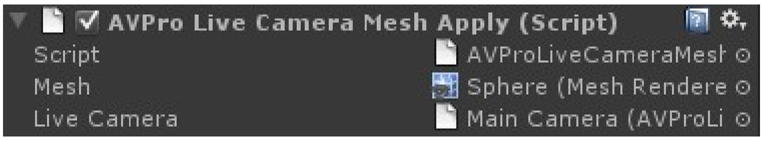


AVPro Live Camera Material Apply (Script)

Script	AVProLiveCameraMate
Material	DemoShiny
Live Camera	Main Camera (AVProLi

Use this component to apply an **AVProLiveCamera** to a material in your scene.

5.5 AVProLiveCameraMeshApply



☑ AVPro Live Camera Mesh Apply (Script)

Script AVProLiveCameraMesh

Mesh Sphere (Mesh Rendere

Live Camera Main Camera (AVProLi

Use this component to apply an **AVProLiveCamera** to all of the materials used by a mesh in your scene.

5.6 AVProLiveCameraGrabber

This component demonstrates how you can grab the camera texture as a Color32 array. This can be useful for passing the data to a 3rd party library, for example for Augmented Reality (AR).

6. Scripting

The scripting can easily be understood by examining the script in the **CameraExplorerDemo** scene which uses the Unity Wrapper Layer**.**

6.1 Layers

The plugin scripts are split into 3 logical layers of abstraction:

1. Low-Level
2. Unity Wrapper
3. Unity Components

6.2 Low-Level Layer

The low-level aspect of this plugin consists of a single static class AVProLiveCameraPlugin that wraps the native DLL functions, exposing it to C# scripting. Most of the plugin functionality is derived from functions in this class. Is it possible to use this class alone however we have built 2 more layers on top to make it easier to use.

6.3 Unity Wrapper Layer

This layer consists of classes with high-level functionality that are built from the low-level layer and expose functionality in a more easy to understand manner. These classes are Unity specific as they use certain Unity classes, eg Texture2D. It is possible to use these scripts in your own scripts or you can use the higher level Unity Components Layer. Classes include:

- AVProLiveCameraDevice
- AVProLiveCameraDeviceMode
- AVProLiveCameraFormatConverter
- AVProLiveCameraPixelBuffer

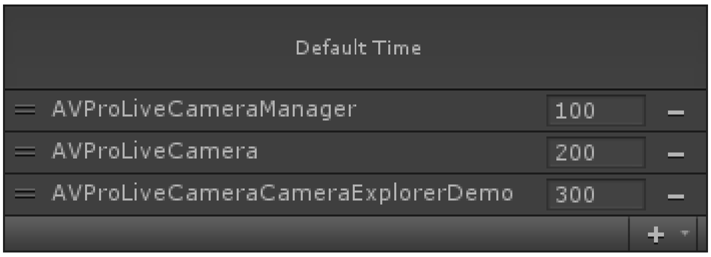
6.4 Unity Components Layer

These scripts are all based on Unity’s MonoBehaviour class and are thus drag-and-drop type components. The scripts are built on the middle wrapper layer classes and allow use of the plugin without having to write scripts. Some of these scripts also have Editor components to customise their appearance. Classes include:

- AVProLiveCameraManager
- AVProLiveCamera
- AVProLiveCameraGUIDisplay
- AVProLiveCameraGrabber
- AVProLiveCameraMaterialApply
- AVProLiveCameraMeshApply

6.5 Script Order

Sometimes the script execution order is important and we recommend this order for our component scripts:



Default Time

AVProLiveCameraManager	100
AVProLiveCamera	200
AVProLiveCameraCameraExplorerDemo	300

The most important is the Manager script which should always be one of the first in your list. Any of our own scripts that refer to the AVPro Live Camera scripts may have to have their script order explicitly set so they run after the AVPro Live Camera scripts.

7. Tips

For best results we recommend:

1. If using a version of Unity before 4.0, then run your application or Unity with “-force-opengl” switch as this plugin runs much faster in OpenGL.

8. FAQ (Frequently Asked Questions)

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

You need to move/copy the “Plugins” folder from your “AVProLiveCamera” 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 the Unity Asset Store handles plugins.

2. How do I make a 64-bit Windows build?

This asset comes with a 32-bit (default) and 64-bit DLL. Unfortunately Unity doesn’t handle 32-bit and 64-bit DLLs automatically so you have to copy the files manually. When you have created a 64-bit Windows build you need to replace the AVProLiveCamera.dll in your ”{YourBuildName}_Data/Plugins” folder with the 64-bit DLL. The 64-bit DLL needs to be renamed to AVProLiveCamera.dll

3. The plugin isn’t playing back smoothly, what could the cause be?

The first thing to check is your hardware to make sure it’s suitable. You’ll need a decent GPU and CPU with the specifications related to the resolution of camera you’re trying to play back.

If your video plays smoothly in the editor but is jerky when you make a build then disabling Unity’s multi-threaded rendering can help. Go to Player Settings, switch the Inspector to Debug disable and disable “MT Rendering”.

You should also check the Asset Store to make sure you’re using the latest version of this plugin, and using the latest version of Unity.

4. The camera display frame rate is lower than expected, what could the cause be?

You should check the settings of the camera to make sure it isn’t doing unnecessary processing which reduces the frame rate. Disable options like “low light compensation”, disable auto exposure and reduce exposure settings. You can access camera settings via the CameraExplorer demo (click “Configure”), or usually via the camera setup application on the Start Menu.

5. 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.

9. Version History

- **Version X**
  - ← Your suggestion here.
  - Add vsync locked updating?
  - Add dropped frame counter?
  - Faster format conversion path for native formats?
  - Support planar pixel formats?

- **Version 1.8 - 29 August 2013**
  - Added support for MJEPG camera mode
  - Added support for RGB24 camera mode
  - Added support for 8-bit mono camera mode
  - Added support for colour convertable camera modes
  - 64-bit builds no longer require DLL copying
  - Fixed yuy2 bug in frame grabber

- **Version 1.7 - Tuesday 28 May 2013**
  - Improved Camera Explorer demo
  - Fixed bug affecting IDS-Imaging cameras
  - Optimised frame conversion
  - High-speed cameras (>60fps) now supported.
  - Improved internal frame buffering
  - Added grabbing from internal frame buffer
  - Added counter for captured and displayed FPS
  - Added documentation about layers of abstraction

- **Version 1.62 - Monday 29 April 2013**
  - Fixed DemoInfo script error
  - Fixed lost device when dragging/resizing window
  - Fixed 64-bit crash bug when dragging/resizing window

- **Version 1.6 - Monday 15 April 2013**
  - Added deinterlacing
  - Added Unity 4.1 non-pow2 texture support - optimisation
  - Added support for hot-swapping camera devices
  - Added mirror display to GUI Display component
  - Added Script Order to documentation

- **Version 1.52 - Monday 18 March 2013**
  - Added Unity 4.1 support
  - Fixed some platform #if issues

- **Version 1.5 - Monday 4 March 2013**
    - Added Color32 grabber component
    - Optimised pixel format conversion
    - Fixed conflict with other AVPro plugins by using unique GL.IssuePluginEvent()
    - Fixed some shaders display half the texture

- **Version 1.4 - Saturday 15 December 2012**
    - Unity 4.0 support added.
    - Unity 4.0 native DirectX texture updates supported, boosting performance.
    - Various minor improvements

- **Version 1.2 - Monday 10 September 2012**
    - Added support for devices with crossbars including TV-cards and capture cards with multiple inputs (eg AVerMedia).
    - Faster device enumeration (startup).
    - Added colour to AVProLiveCameraGUIDisplay component
    - Inspector UI improvements.
    - Fixed a texture size bug in OpenGL mode.

- **Version 1.12 - Thursday 12 July 2012**
    - Fixed minor null reference bug.
    - Minor tweaks to file names to make them unique.

- **Version 1.1 - Wednesday 4 July 2012**
    - Added Windows 64-bit support.
    - Renamed files to prevent future name collisions.

- **Version 1.0 - Thursday 20 June 2012**
    - Initial release submitted to Asset Store.

10. 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 (screenshots, logs, scripts etc) so that we may remedy the problem as fast as possible.

11. About RenderHeads Ltd

RenderHeads is an award winning team of creative and technical people that has been designing and building cutting edge technology solutions since 2006. We specialise in developing bespoke audio-visual interactive installations.

11.1 Services

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

11.2 Our Unity Plugins

AVPro QuickTime

AVPro Windows Media

AVPro Movie Capture

AVPro Live Camera