

180°

360°

3D



VR Video Player



oculus

Samsung
Gear^{VR}



VIVE



Cardboard

VR Video Player

v2.1.0

Created by [Evereal](#)
Contact dev@evereal.com

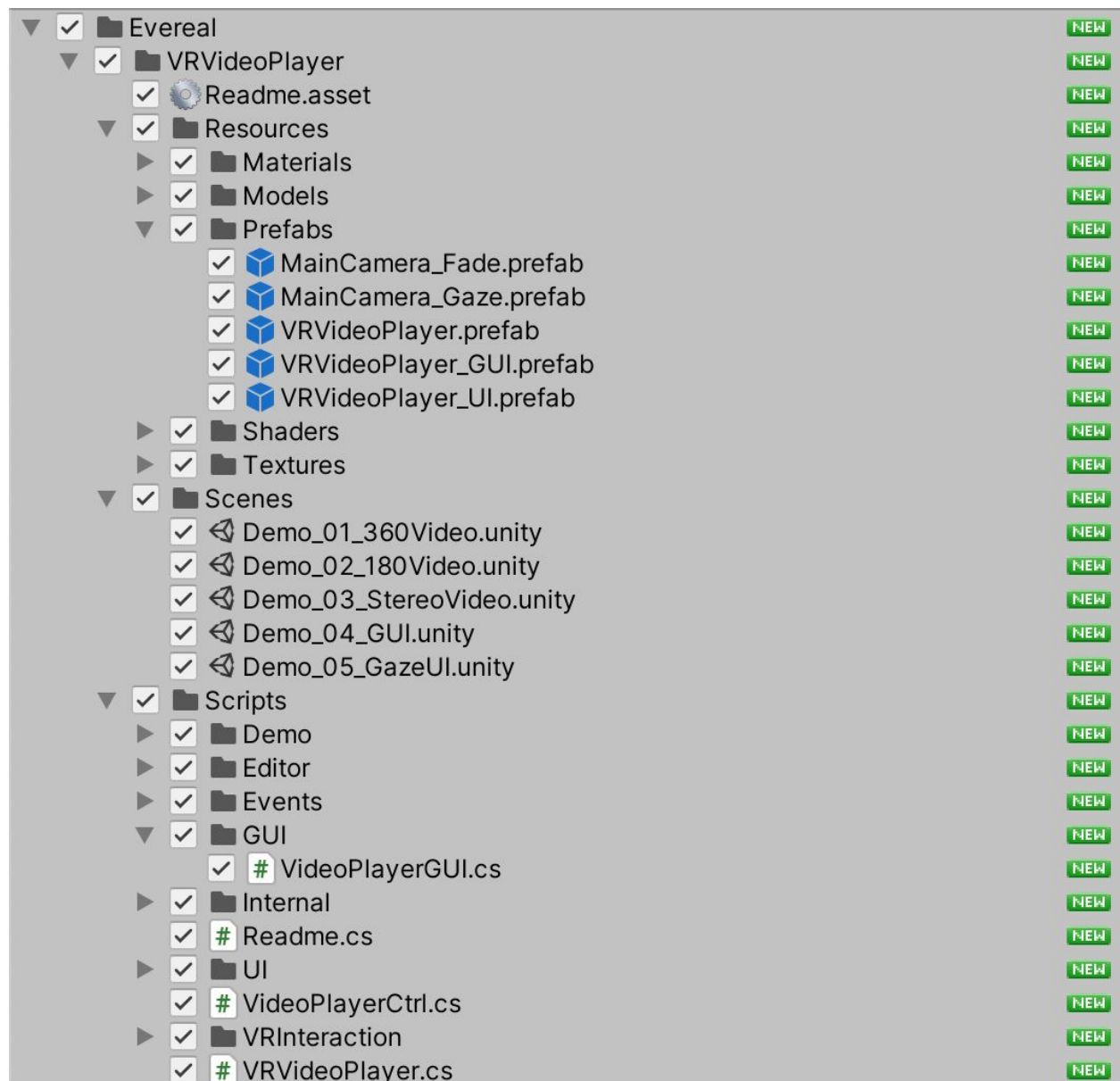
Table of Contents

1. Overview	3
2. Getting Started	4
3. Prefabs	5
4. Integrations	6
4.1 Integrate with Gaze UI	6
4.1.1 Integrate with Oculus	7
4.1.2 Integrate with SteamVR	8
4.1.3 Integrate with Cardboard	10
4.2 Integrate with Controller UI	11
4.2.1 Integrate with Oculus	11
4.2.2 Integrate with SteamVR	12
4.3 Integrate with Video Player	13
4.4 Integrate with AVPro Video	15
4.5 Integrate with Easy Movie Texture	16
5. Events	17
6. Properties	18
7. Public Methods	18
8. Feedback	18

1. Overview

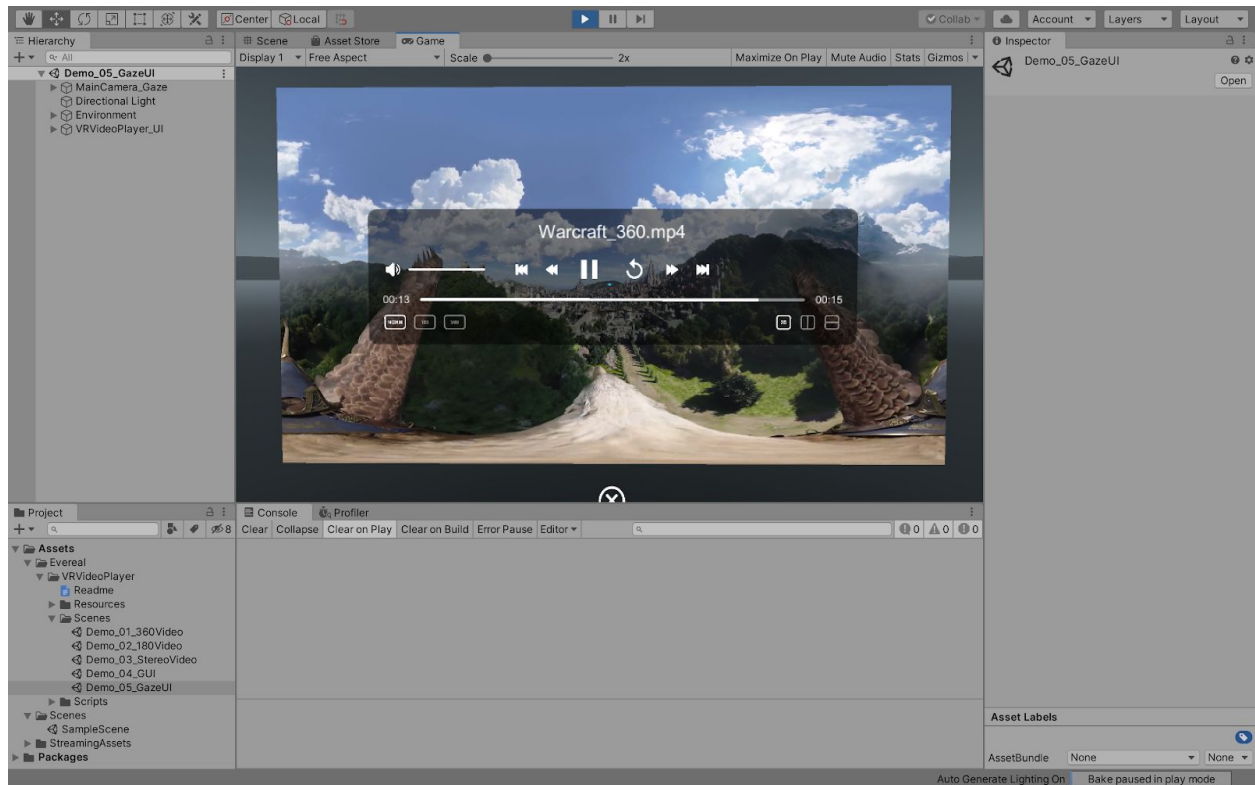
VR Video Player assets will help you to set up a virtual reality video experience in your Unity app just in a few seconds, it can support 360, 180 and stereo video play.

When you import VR Video Player plugin into your Unity project, the following assets will add:



2. Getting Started

VR Video Player comes with several demos to help you to understand the functionality of the plugin quickly. Start by creating a new project and importing all VR Video Player package assets including demo scenes files.



All demo scenes located in [Assets/Everyreal/VRVideoPlayer/Scenes/](#) folder:

[Demo_01_360Video](#) plays a mono 360 video.

[Demo_02_180Video](#) plays a mono 180 video.

[Demo_03_StereoVideo](#) plays a stereo top bottom 360 video.

[Demo_04_GUI](#) demonstrates VR video player function with simple GUI.

[Demo_05_GazeUI](#) is a complete gaze interaction demo play video with time scrub and controls UI components.

3. Prefabs

VR Video Player assets containing useful prefabs can be dragged and dropped into your scene without any code.

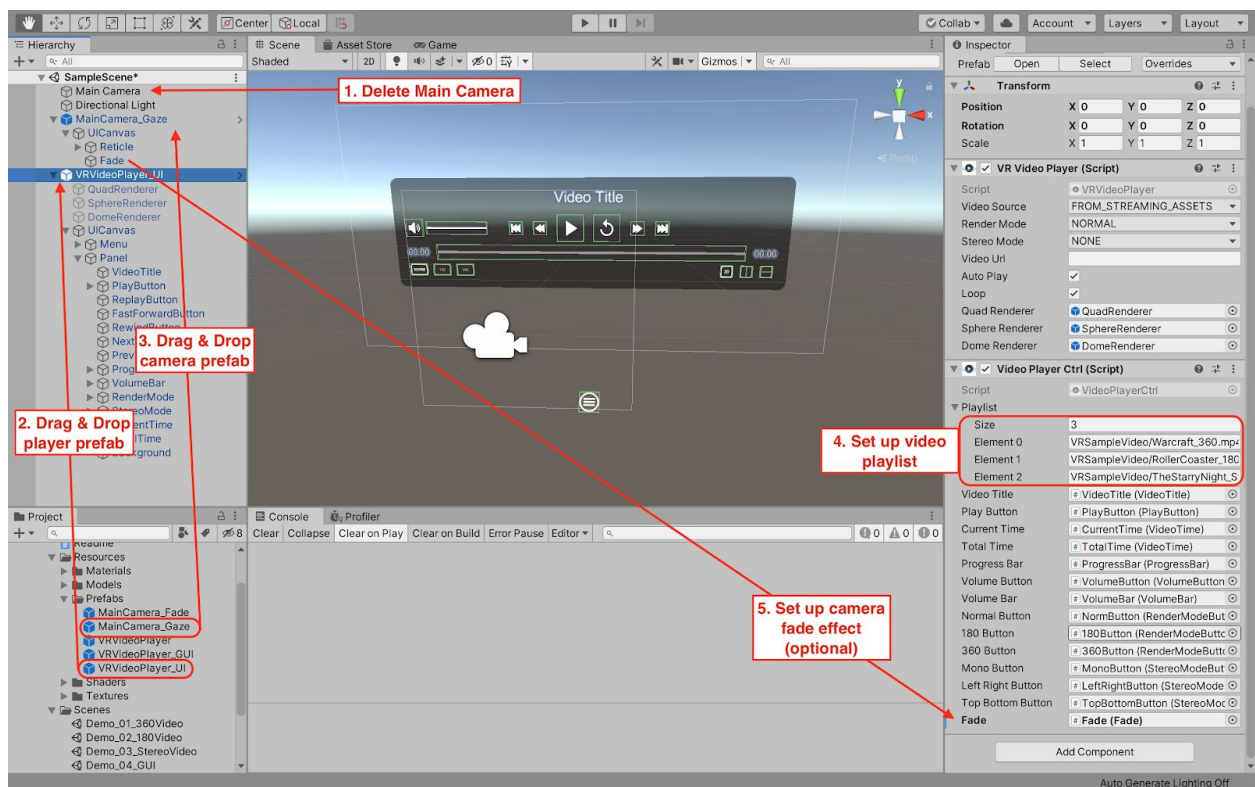
Prefab	Description
<i>VRVideoPlayer</i>	Main VR video player prefab you can use to switch normal/180/360 render mode, mono/stereo mode.
<i>VRVideoPlayer_GUI</i>	Same with VRVideoPlayer but built with a simple GUI you can test the video player.
<i>VRVideoPlayer_UI</i>	Same with VRVideoPlayer but built with a complete VR UI you can test the video player.
<i>MainCamera_Fade</i>	Add the fade effect to the camera.
<i>MainCamera_Gaze</i>	Add gaze interaction to the camera.
<i>UICanvas_Gaze</i>	Gaze interaction for VR integration.

4. Integrations

You can integrate your Unity project with VR Video Player based on your requirements.

4.1 Integrate with Gaze UI

You can choose this option if you want full integration with the UI. Including features such as play, pause, replay, time scrubber and adjust volume, etc. And you can also easily customize the UI as you like. This asset supports Gaze UI interaction and Controller interaction, let's start with gaze interaction.



Step 1: Delete the original Main Camera in the scene.

Step 2: Drag & Drop the **VRVideoPlayer_UI** prefab ([Assets/Everreal/VRVideoPlayer/Resources/Prefabs/VRVideoPlayer_UI](#)) prefab into your game scene.

Step 3: Drag & Drop the **MainCamera_Gaze** prefab ([Assets/Everreal/VRVideoPlayer/Resources/Prefabs/MainCamera_Gaze](#)) prefab into your game scene.

This prefab will add gaze interaction support, controller interaction support is coming soon.

Step 4: Set up video playlist files, default the files located in the StreamingAssets folder.

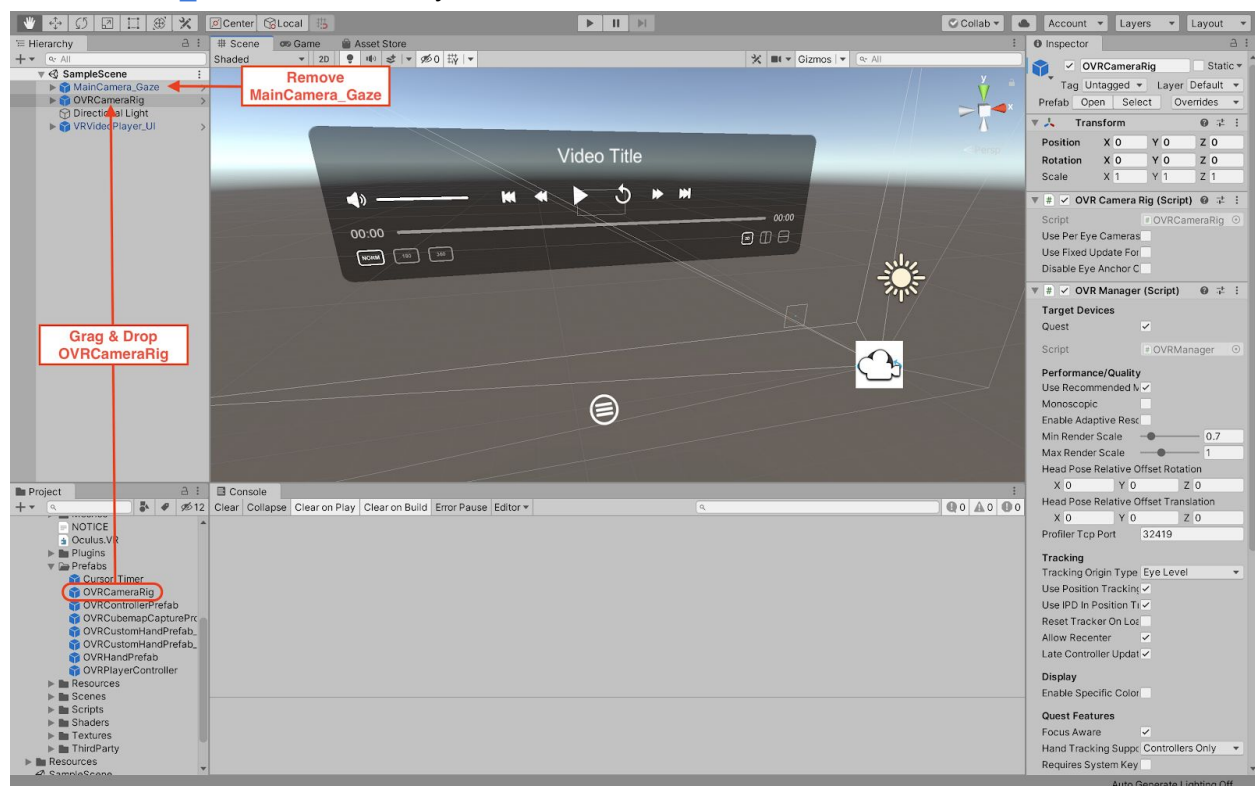
Step 5 (Optional): If you want to add a camera fade effect for switching between video, you can attach the *Fade* object into the VideoPlayerCtrl script.

And you can also check out *Demo_05_GazeUI* which is already set up complete and comes with a beautiful background scene.

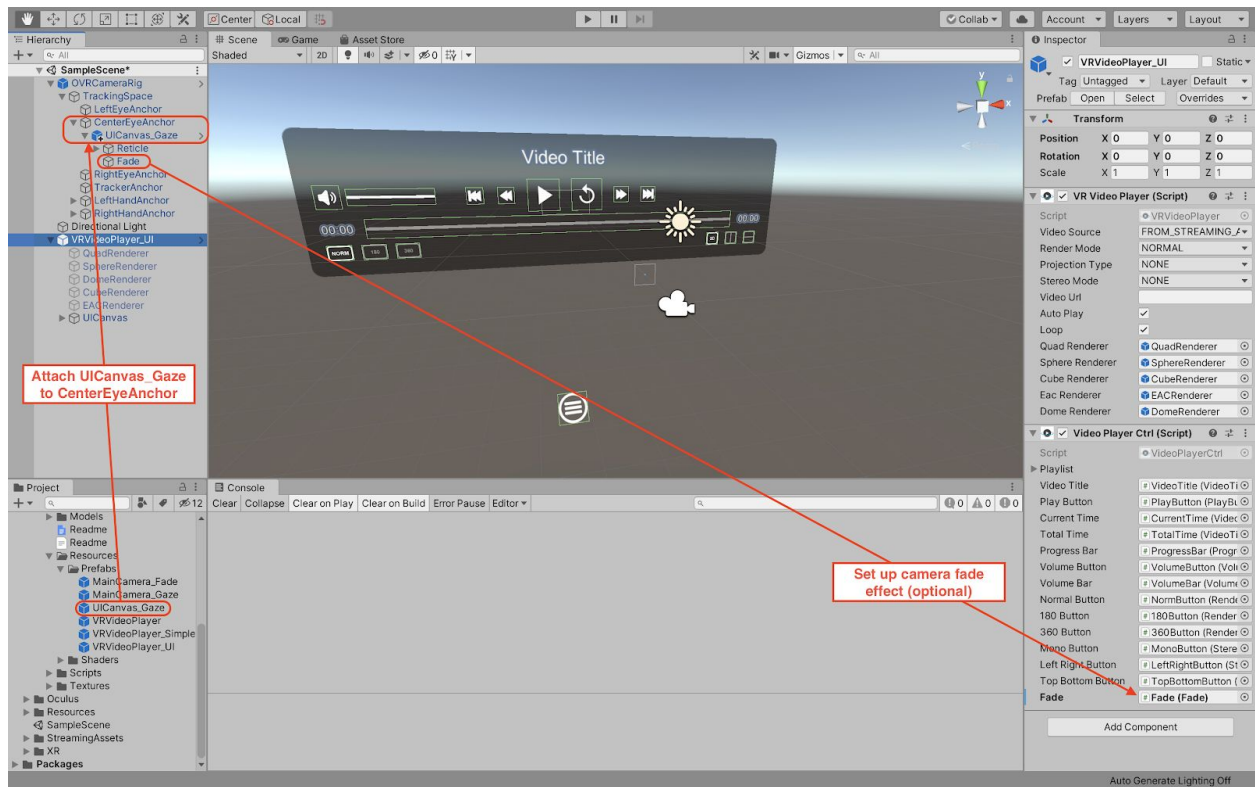
4.1.1 Integrate with Oculus

If you want to integrate with Oculus, you will need a few more steps. First, integrate Oculus with Unity, please see: <https://developer.oculus.com/documentation/unity/book-unity-gsg/>

Step 1: Drag & Drop the *OVRCameraRig* prefab into the scene, and remove *MainCamera_Gaze* if it exists in your scene.



Step 2: Attach **UICanvas_Gaze** prefab into **CenterEyeAnchor** of **OVRCameraRig** prefab.



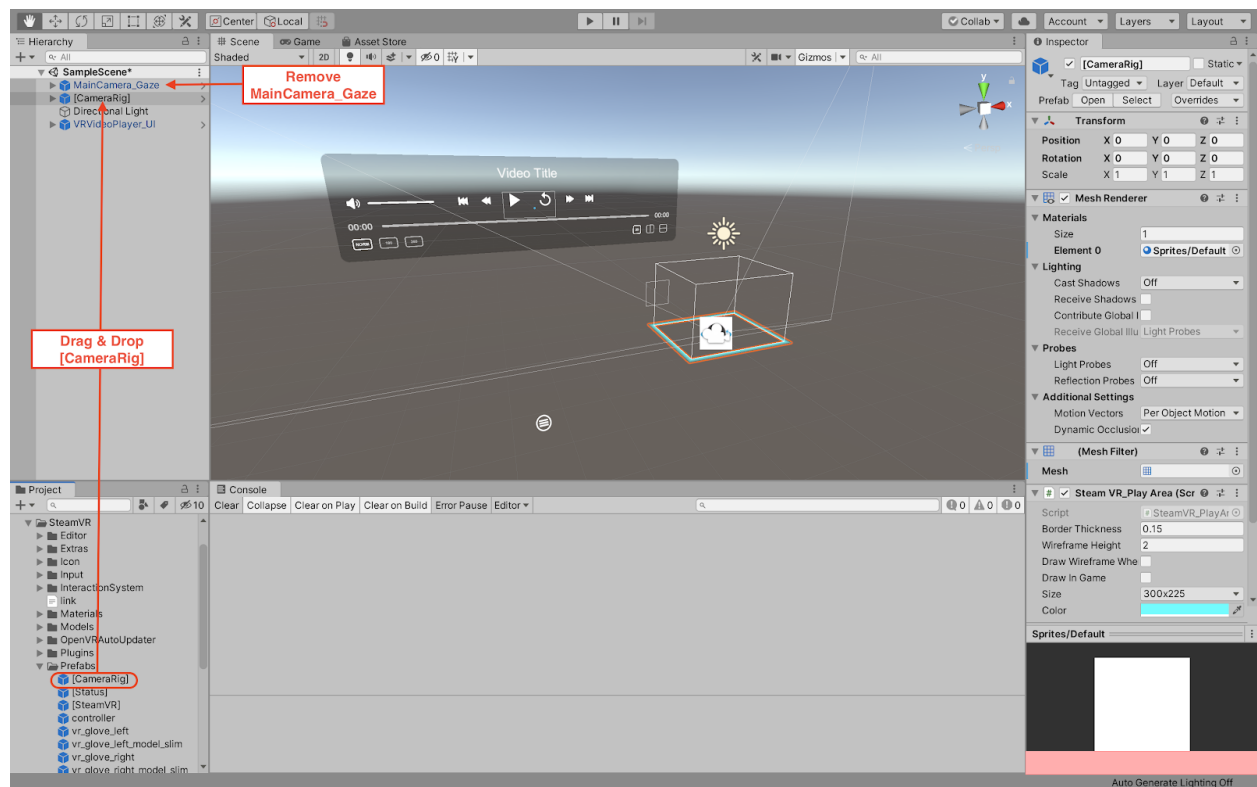
Step 3 (Optional): If you want to add a camera fade effect for switching between video, you can attach the **Fade** object into the **VideoPlayerCtrl** script.

4.1.2 Integrate with SteamVR

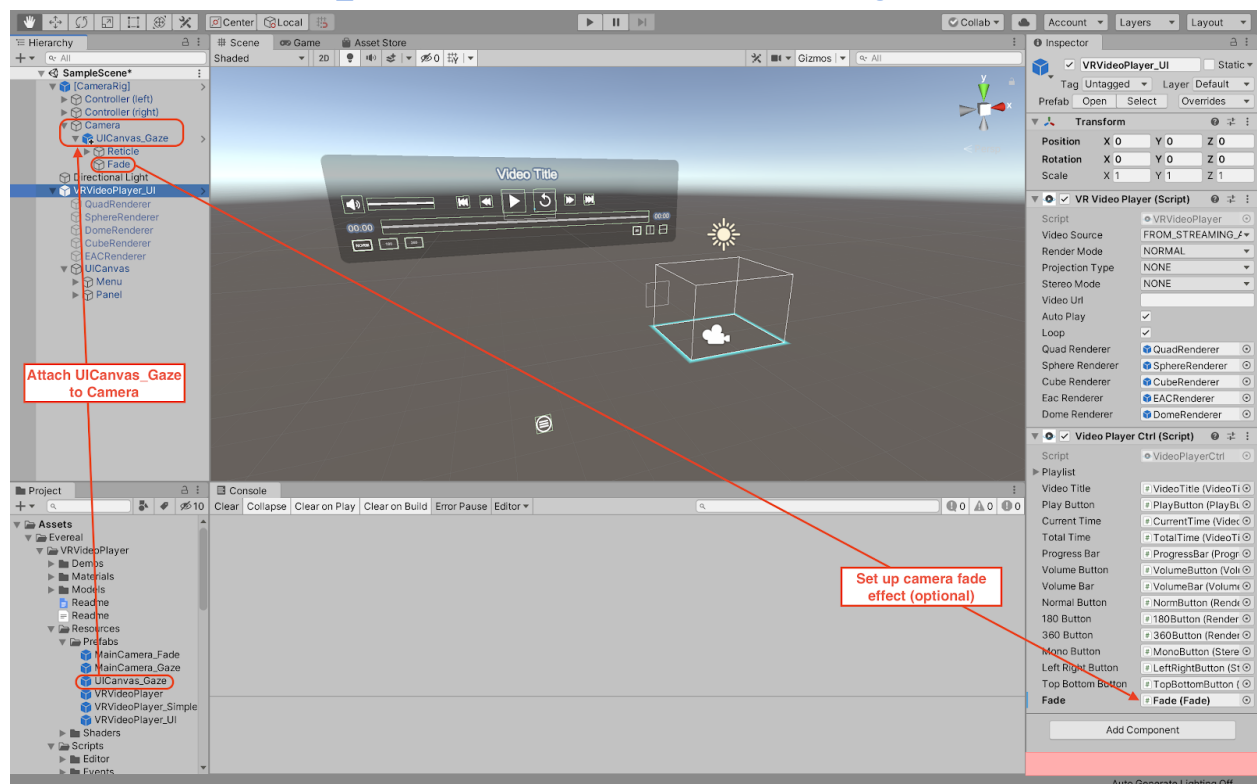
If you want to integrate with SteamVR, you will need a few more steps. First, integrate SteamVR with Unity, please see:

https://valvesoftware.github.io/steamvr_unity_plugin/articles/Quickstart.html

Step 1: Drag & Drop the **[CameraRig]** prefab into the scene, and remove **MainCamera_Gaze**.



Step 2: Attach **UICanvas_Gaze** prefab into **Camera** of **[CameraRig]** prefab.



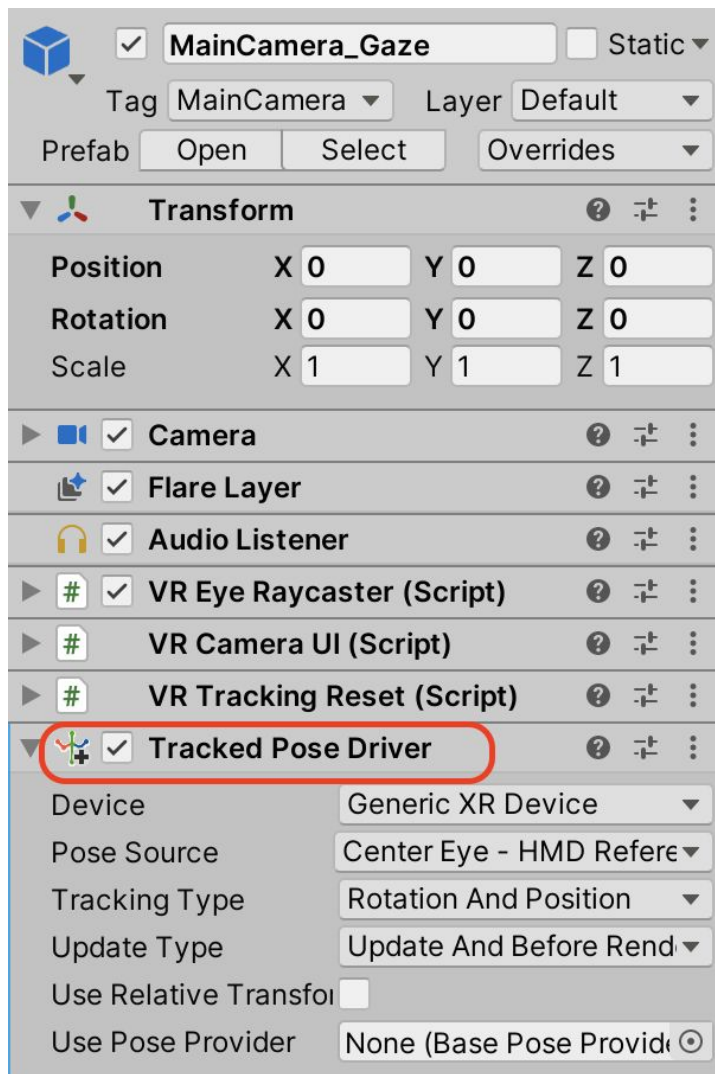
Step 3 (Optional): If you want to add a camera fade effect for switching between video, you can attach the [Fade](#) object into the VideoPlayerCtrl script.

4.1.3 Integrate with Cardboard

If you want to integrate with Cardboard, you will need a few more steps. First, integrate Google Cardboard with Unity, please see:

<https://developers.google.com/cardboard/develop/unity/quickstart>

Step 1: Attach [TrackedPoseDriver.cs](#) to the prefab [MainCamera_Gaze](#).



4.2 Integrate with Controller UI

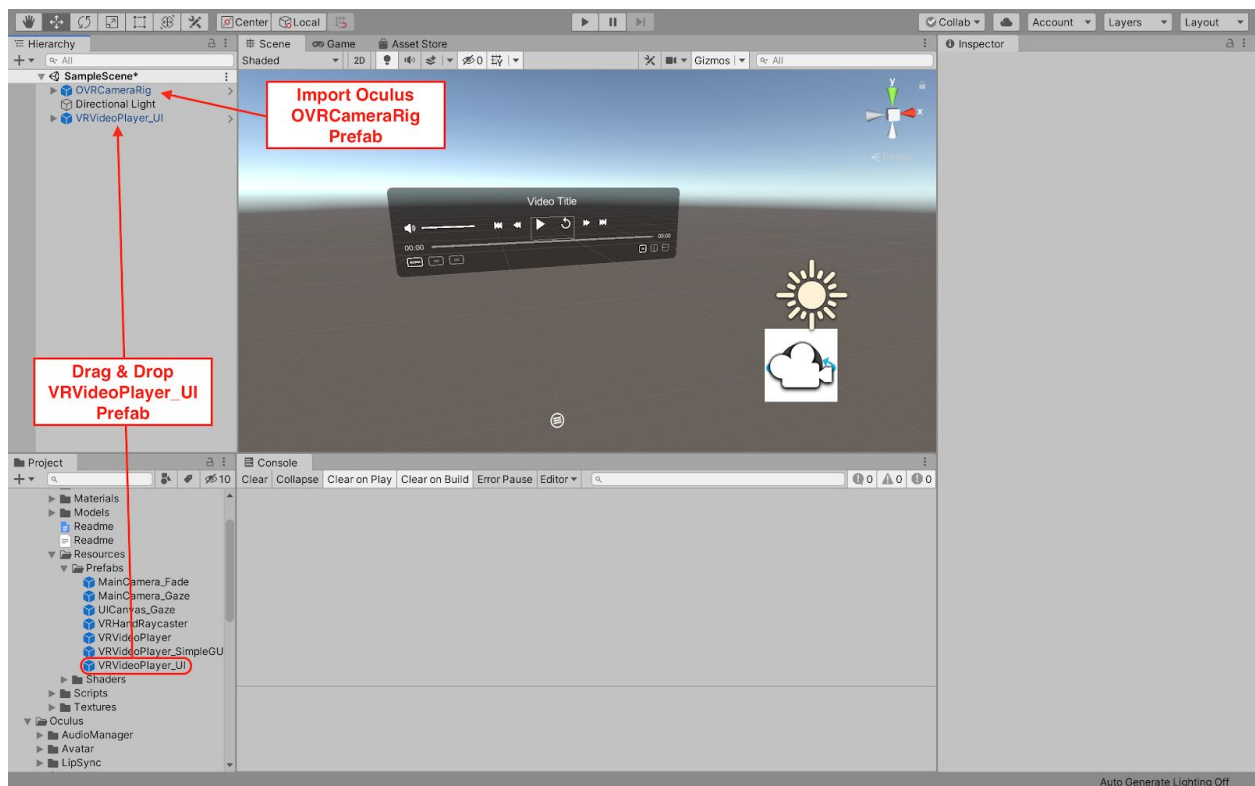
This part will introduce how to integrate Controller interaction.

4.2.1 Integrate with Oculus

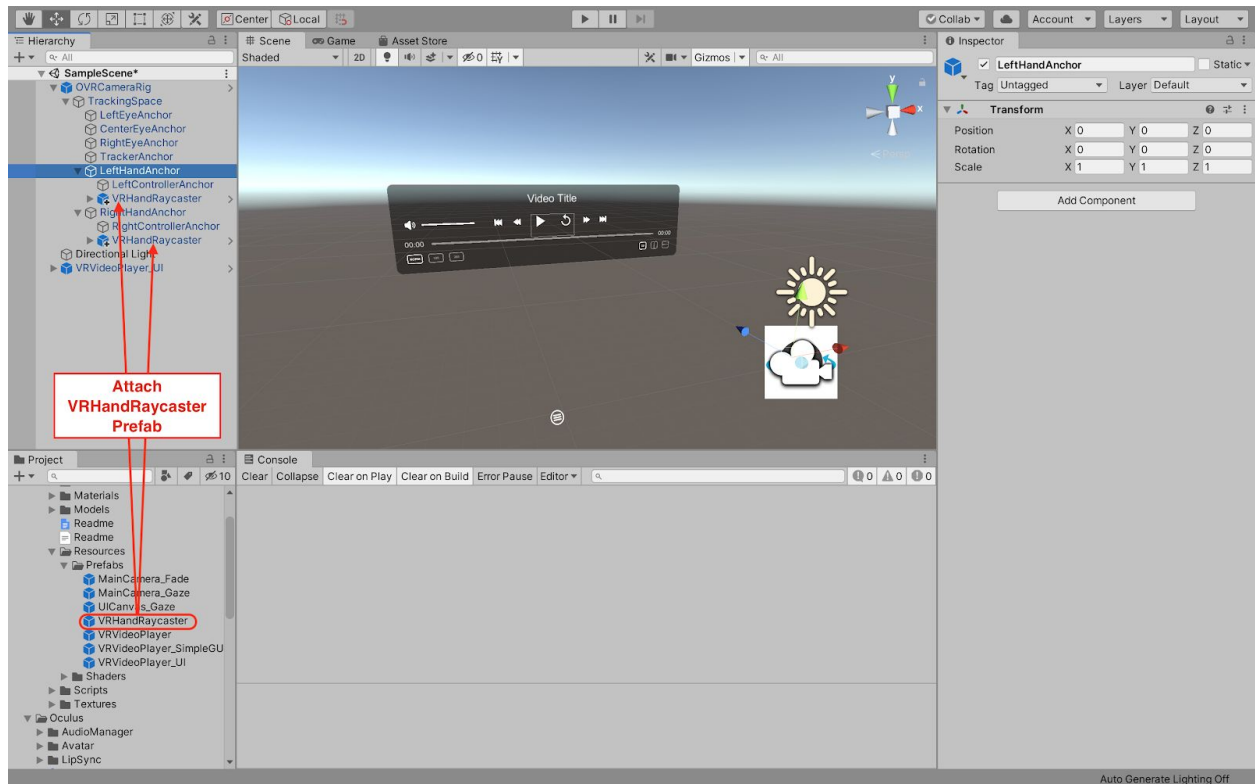
For integrating Oculus with Unity, please see:

<https://developer.oculus.com/documentation/unity/book-unity-gsg/>

Step 1: Drag & Drop the **VRVideoPlayer_UI** and **OVRCameraRig** prefab into the scene, and remove the **Main Camera** if it exists in your scene.



Step 2: Attach **VRHandRaycaster** prefab into **LeftHandAnchor** and **RightHandAnchor** for enabling controller raycaster interaction.



Step 3: Enable Oculus VR input by selecting **Tools -> Evereal -> VRVideoPlayer -> VR -> Oculus**.

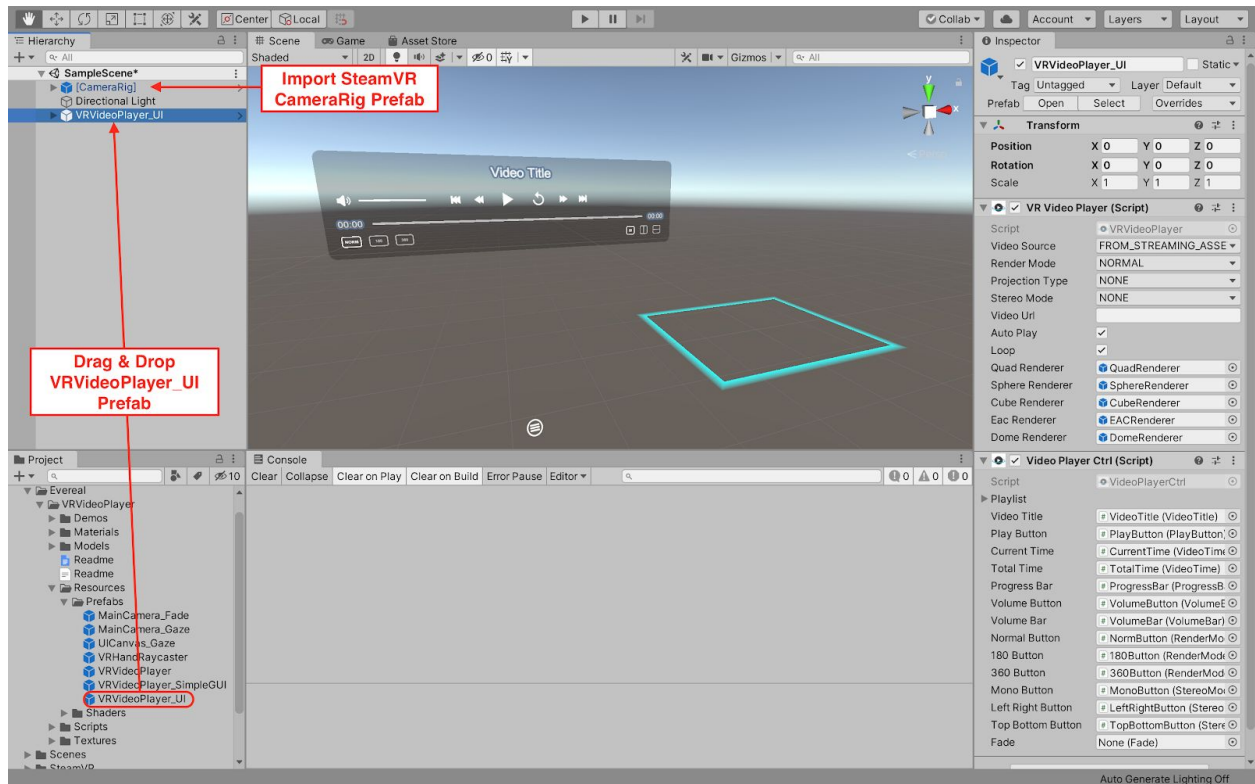


4.2.2 Integrate with SteamVR

For integrating SteamVR with Unity, please see:

https://valvesoftware.github.io/steamvr_unity_plugin/articles/Quickstart.html

Step 1: Drag & Drop the **VRVideoPlayer_UI** and **[CameraRig]** prefab into the scene, and remove the **Main Camera** if it exists in your scene.



Step 2: Attach **VRHandRaycaster** prefab into **Controller (left)** and **Controller (right)** for enabling controller raycaster interaction.

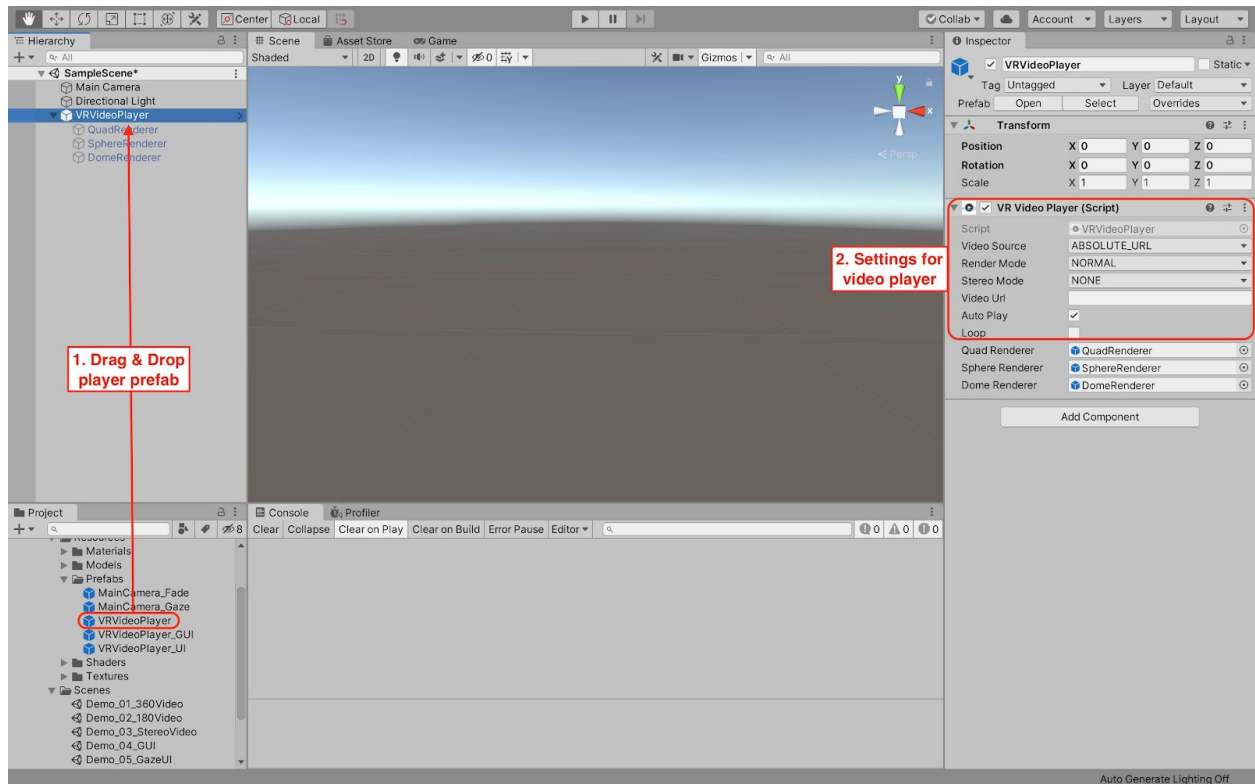
Step 3: Enable Oculus VR input by selecting **Tools -> Evereal -> VRVideoPlayer -> VR -> SteamVR**.



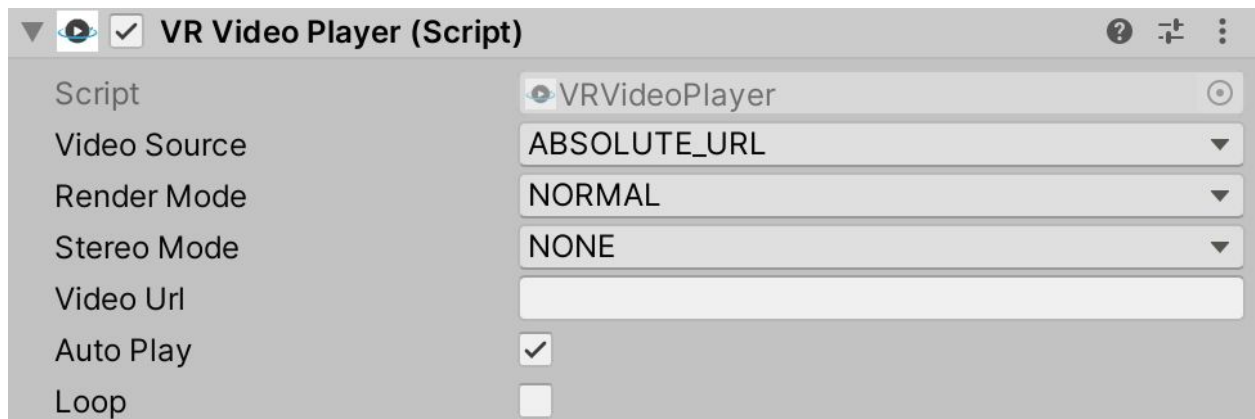
4.3 Integrate with Video Player

You can choose this integration option If you only require the plugin just play a VR video without any interaction.

Step 1: Drag & Drop the **VRVideoPlayer** prefab.



Step 2: Configuring **VRVideoPlayer** component's properties based on your requirement.



Video Source - Video file source type, supports Absolute Path, Streaming Assets and Project Folder.

Render Mode - Video render mode type, supports Normal, 360 and 180 video.

Stereo Mode - Video stereo mode type, supports Mono, Side by Side and Top Bottom.

Video Url - Set your video path url based on video source.

Auto Play - Start video playback as soon as the game starts or new video url loaded.

Loop - Start video playback at the beginning when the end is reached.

4.4 Integrate with AVPro Video

AVPro Video is a powerful plugin for Unity that gives developers an easy-to-use video playback solution on multiple platforms.



AVPro Video asset versions:

- All-in-one: <https://assetstore.unity.com/packages/tools/video/avpro-video-56355>
- Android: <https://assetstore.unity.com/packages/tools/video/avpro-video-android-52745>
- iOS: <https://assetstore.unity.com/packages/tools/video/avpro-video-ios-56352>
- Windows: <https://assetstore.unity.com/packages/tools/video/avpro-video-windows-57969>
- OS X: <https://assetstore.unity.com/packages/tools/video/avpro-video-macos-56353>
- tvOS: <https://assetstore.unity.com/packages/tools/video/avpro-video-tvos-56354>

VR Video Player supports switching video player backend to AVPro Video instead of Unity VideoPlayer if you think it is lacking features you need for your project.

The integration steps is extreaming easy.

Step 1: Download the AVPro Video asset into your project, you can purchase it on Asset Store or using a [Free Trial](#) version.

Step 2: From menu, select **Tools** -> **Everreal** -> **VRVideoPlayer** -> **MediaPlayer** -> **AVPro Video** and click.



Then the video playback system should be switched to AVPro Video instead of Unity VideoPlayer.

4.5 Integrate with Easy Movie Texture

Easy Movie Texture has great support for mobile devices and streaming features.



Easy Movie Texture asset link:

<https://assetstore.unity.com/packages/tools/video/easy-movie-texture-video-texture-10032>

VR Video Player supports switching video player backend to Easy Movie Texture instead of Unity VideoPlayer if you think it is lacking features you need for your project.

The integration steps is extreaming easy.

Step 1: Download the Easy Movie Texture asset into your project, you can purchase it on Asset Store.

Step 2: From menu, select **Tools -> Everreal -> VRVideoPlayer -> MediaPlayer -> Easy Movie Texture** and click.



Then the video playback system should be switched to Easy Movie Texture instead of Unity VideoPlayer.

5. Events

VR Video Player wrapped Unity VideoPlayer events, for more details, please visit:

<https://docs.unity3d.com/ScriptReference/Video.VideoPlayer.html>

6. Properties

VR Video Player wrapped Unity VideoPlayer properties, for more details, please visit:
<https://docs.unity3d.com/ScriptReference/Video.VideoPlayer.html>

7. Public Methods

VR Video Player wrapped Unity VideoPlayer public methods, for more details, please visit:
<https://docs.unity3d.com/ScriptReference/Video.VideoPlayer.html>

8. Feedback

If you have any feedback on the **VR Video Player** plugin, please email us through dev@everreal.com directly, your advice will be very valuable to us.

You are also very welcome to rate our plugin and leave your comment on the Unity Asset Store.