Confidential Classification：   No.：

Skyworth Standalone VR SDK（Unity）

Development Document

Prepared by _____**Date** _____

Reviewed by _____**Date** _____

Approved by _____**Date** _____

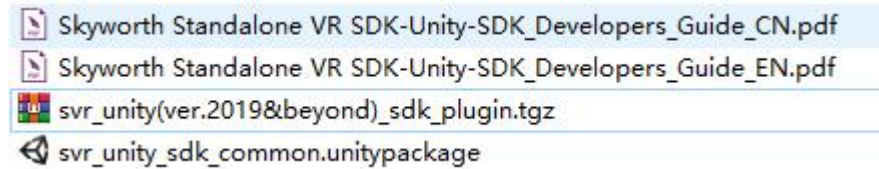**Shenzhen Skyworth New World Technology Co.,LTD**

Content

1 Introduction

1.1 SDK Introduction

The hardware equipment SDK supports： Skyworth Standalone VR V901 ,S1

SDK mainly provides: 3dof handle interaction support， Nolo6DOF， multifunctional interaction support， binocular stereo rendering, etc.

SDK content description：

"*Skyworth Standalone VR SDK-Unity-SDK_Developers_Guide_XX.pdf*" development guidance document

"*svr_unity_sdk_common.unitypackage*" basic Unity SDK, which shall be imported before development

"*svr_unity(ver.2019&beyond)_sdk_plugin.tgz*" is used for plug-ins for Unity 2019 and above versions, please refer to section 7 for use details.

1.2 Development Environment

1. Unity: Unity of LTS version is recommended to use， which is a long-term supported version of Unity, and more stable than other versions.

   2017: support above 2017.4.6

   2018: support above 2018.4.13

   2019: support above 2019.3.2

2. Android SDK: API Level 25 and above

3. JDK:jdk1.7.0_01 and above

1.3 SDK Brief Introduction

SDK is a UnityPackage, including necessary codes and resources. Import it into the project when using it. The specific use method will be introduced in the following chapters. It is recommended not to use it with other VR SDKs. In order to avoid conflicts, no SDKs from other manufacturers should be used in the project.

2 SDK Use Introduction

2.1 Create New Project

Open Unity, create a new project as follows:
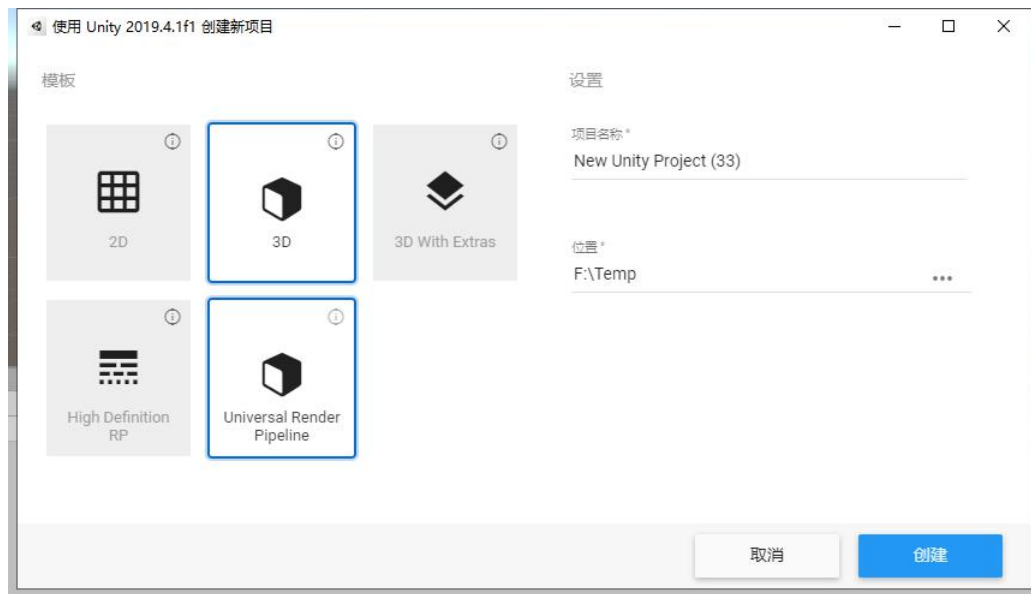
Picture 2.1 Create a New Project

2.2 Import UnityPackage

Choose menu item "Assets->Import Package->Custom Package…" as follows:

Picture 2.2 Import SDK
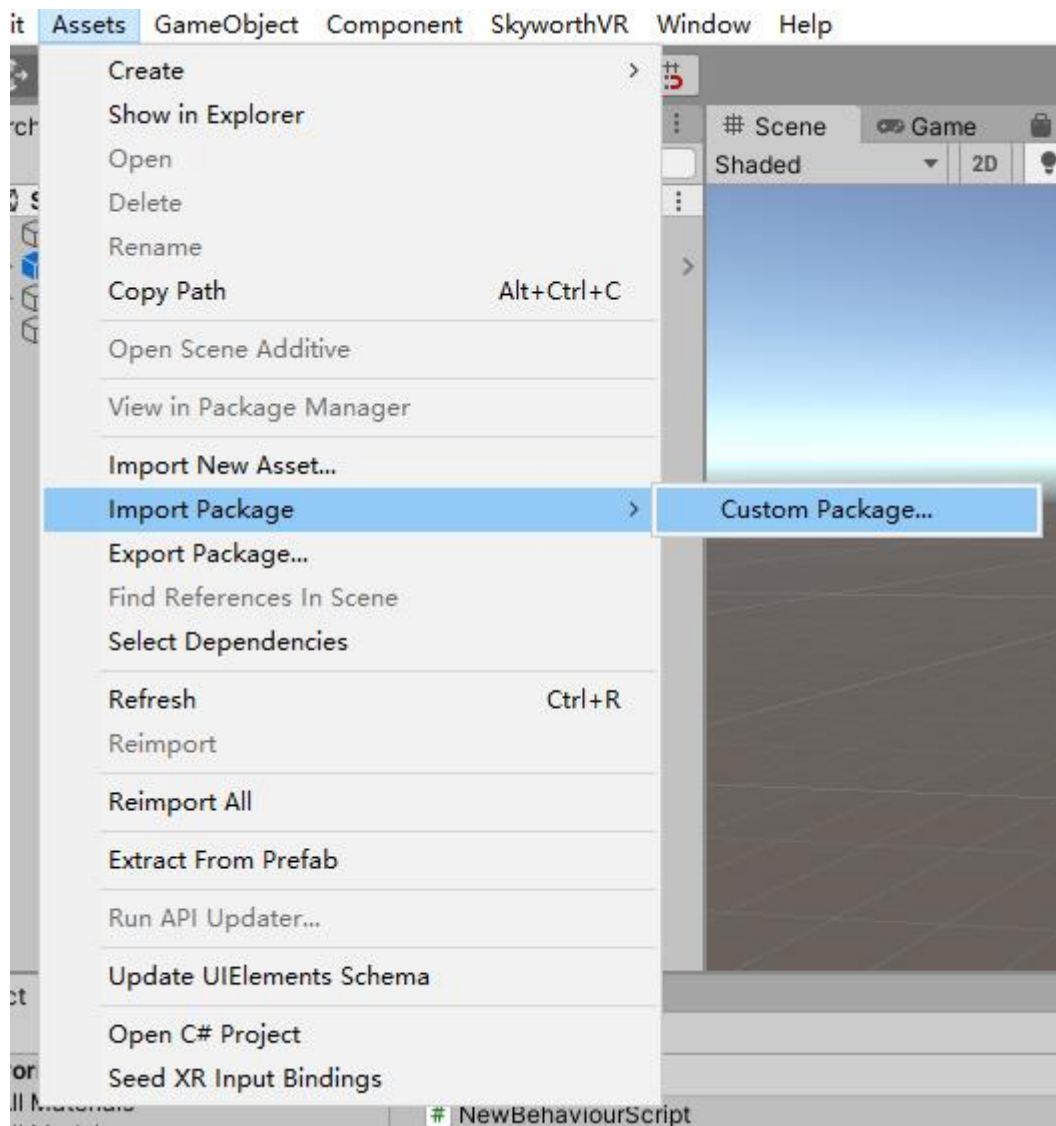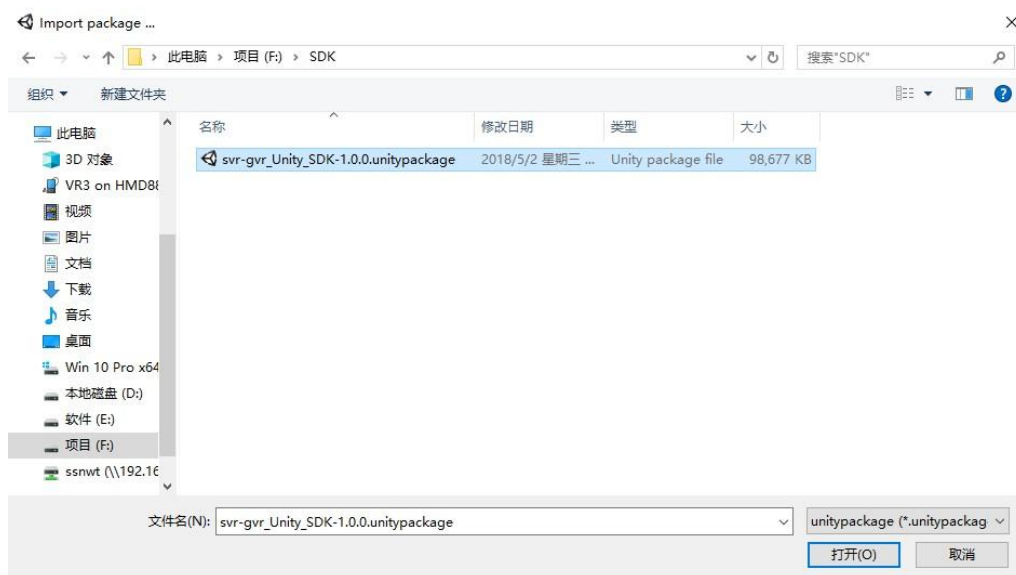
Then, Unity will pop up a file selection dialog, choose "svr_unity_sdk.unitypackage", then click to open:

Picture 2.3 Choose Unity Package

Click to return to the Unity window, the system will pop up the SDK directory level, please import what you need:
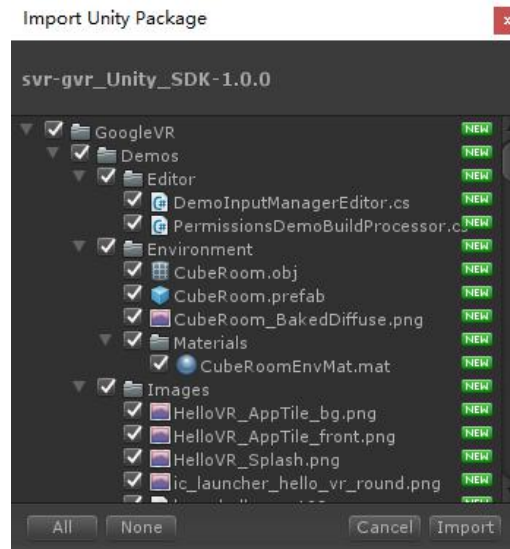
Picture 2.4 Import Options

2.3 Using SDK

Enter into Project option window, click the following in order: Assets->GoogleVR->Demos->Scenes, choose HelloVR scene, click Run button, you can see the following at Game window:
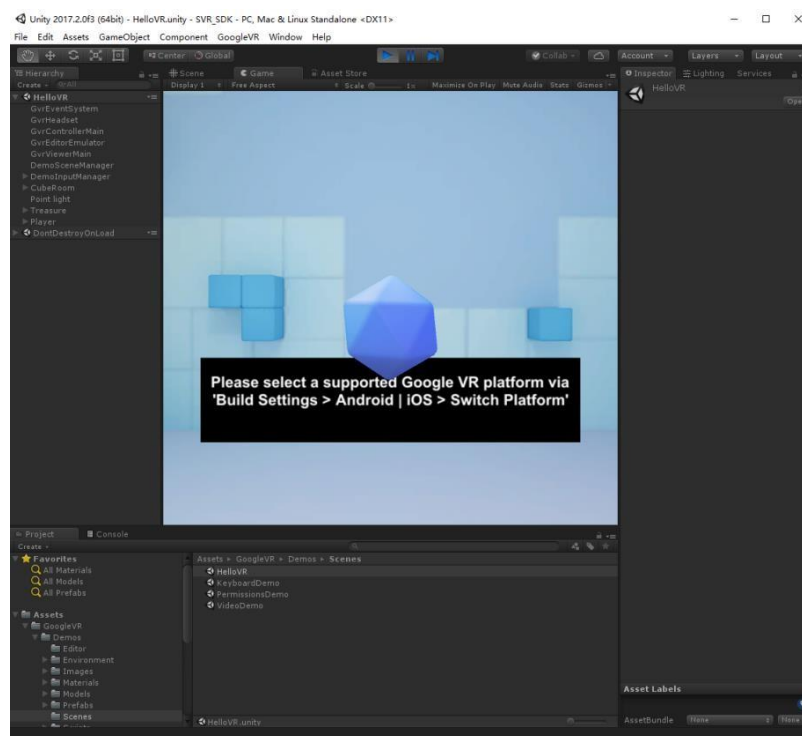
Picture 2.5 Simulation Running

Hold down Alt and move the mouse to turn the picture up, down, left and right; Hold down Ctrl and move the mouse to flip the picture up and down; hold down Shift and move the mouse to simulate handle operation.

2.5 Project Setting

After importing the UnityPackage, click SkyworthVR on the menu to open, there are "Using S1 V901 Setting", "Using S801 Setting", "Using S1 V901 Legacy Setting" in it. If your unity version is 2017 or 2018, Picture 2.6 is what you will see; if your unity version is 2019, then Picture 2.7.
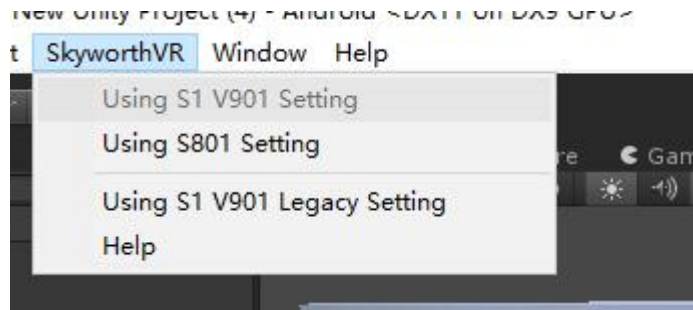
Picture 2.6 sdk Setting
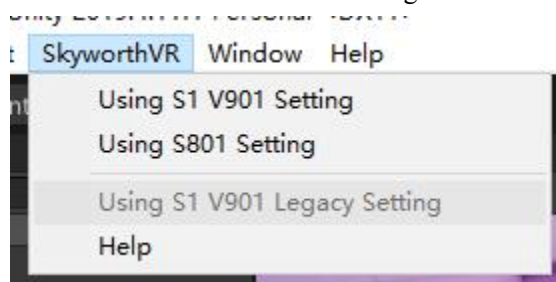
Picture 2.7 2019 Setting

2.5.1 Using S1 V901 Setting

This setting is used to export the apk running on s1 or V901, It will do the following settings for the project：

1. Blit Type:Always

Picture 2.8 bliy type

2. Scripting Define Symbols:SVR

Picture 2.9 Scriptng Define Symbols

3. XR Settings

Picture 2.10 XR Settings

2.5.2 Using S801 Setting

This setting is used to export the apk running on S801, It will do the following settings for the project:

1．Blit Type:Always
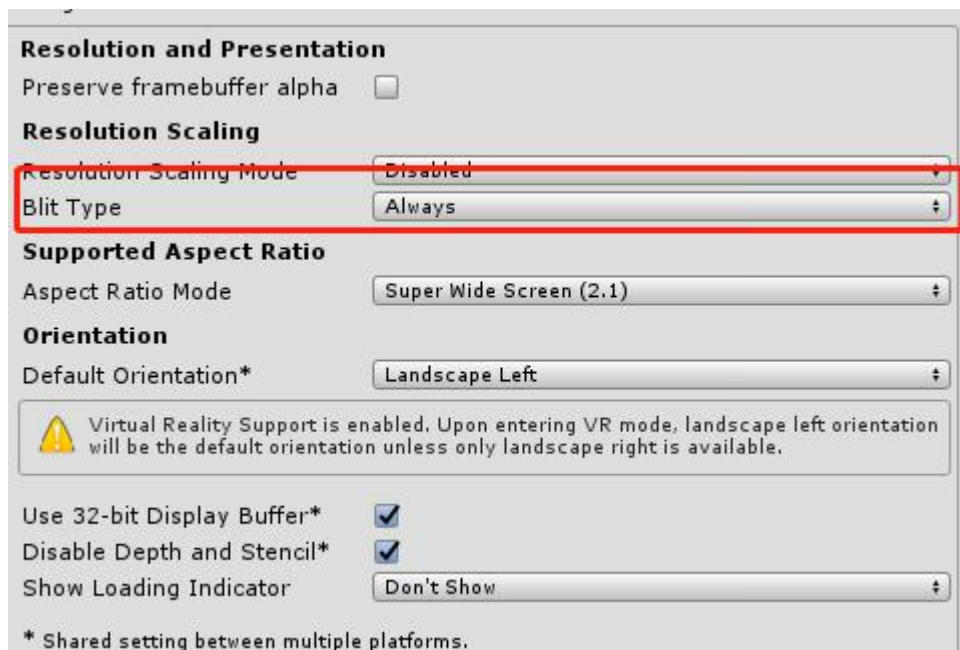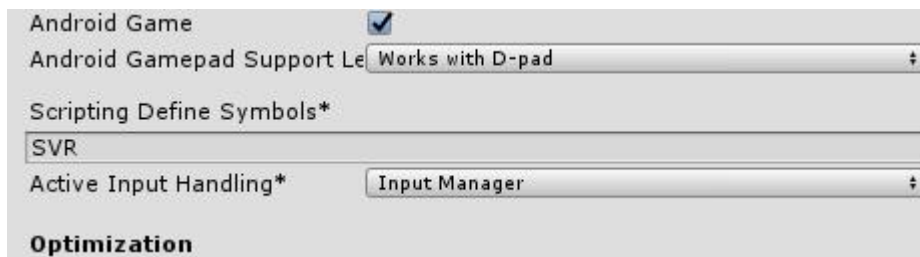

Picture 2.11 blit type

2. Script Define Symbols
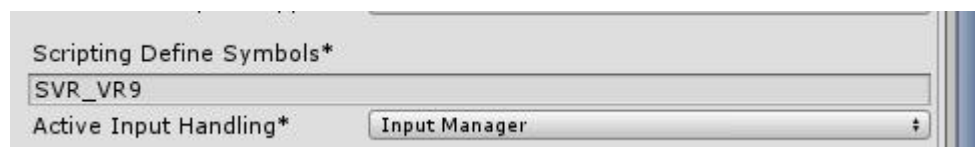
Picture 2.12

3. XR Settings
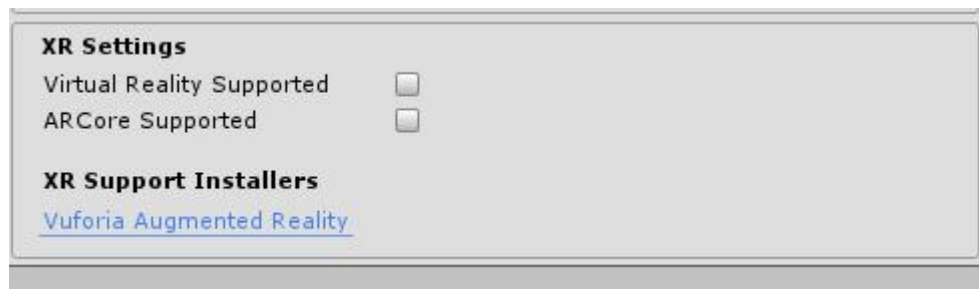
Picture 2.13 XR Settings

2.5.2 QualitySettings

As shown in Picture 2.6, "Levels" selects the "Medium" level of the Android platform by green ticks. The specific parameters are shown in Picture 2.6:
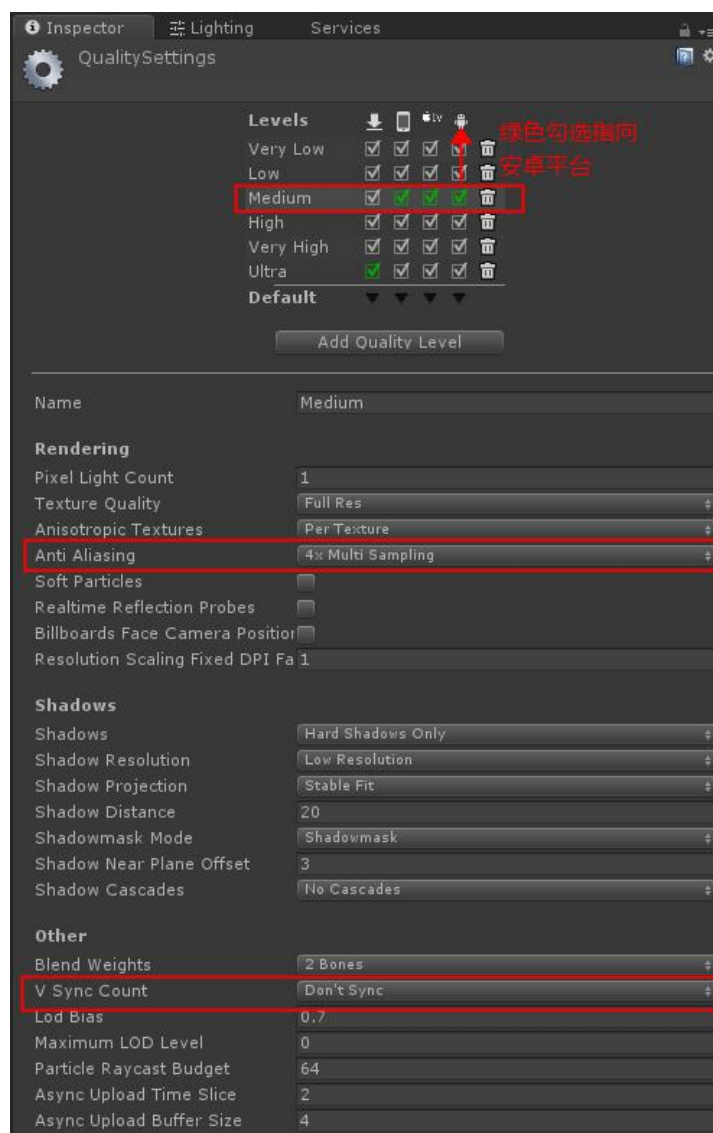
Picture 2.6 QualitySettings Setting Diagram

Note: In the Levels setting, first let the gray bar select the Android platform, i.e., the row with green ticks, as shown in Picture 2.6. Then set up Anti Aliasing and V Sync Count. If you do not select the Android platform first, the settings of these two items will not take effect in the packages exported from Android, causing a black screen when the application is running.

Anti Aliasing: It can be adjusted as needed, and "4 x Multi Sampling" is recommended.

V Sync Count: it's required to set to "Don't Sync"

2.5.3 PlayerSettings

1. On the "Resolution and Presentation" option window, set the export setting to "landscape". The specific parameters are shown in Picture 2.7:
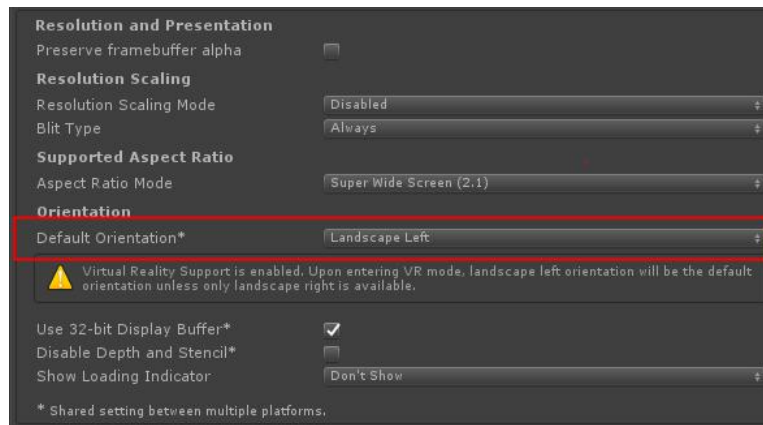
Picture 2.7 Export Setting Diagram

2. On the "Other Settings"option window, do not check "GPU Skinning"; for "API Level", select Android 7.1; for "Write Permission", select SDCard as your need. The specific parameters are shown in Picture 2.8:
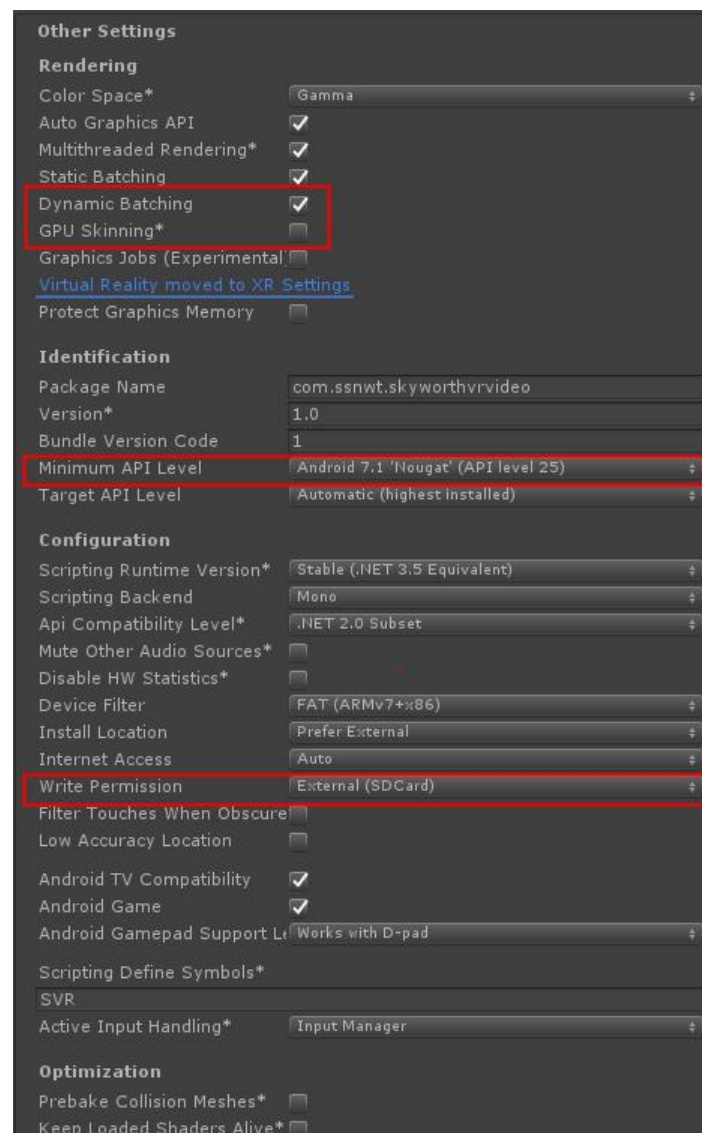
Picture 2.8 PlayerSettings Setting Diagram

2.5.4 Build Settings

Choose default platform "Android", for Build System, choose "Internal"compile method. The specific parameters are shown in Picture 2.10:
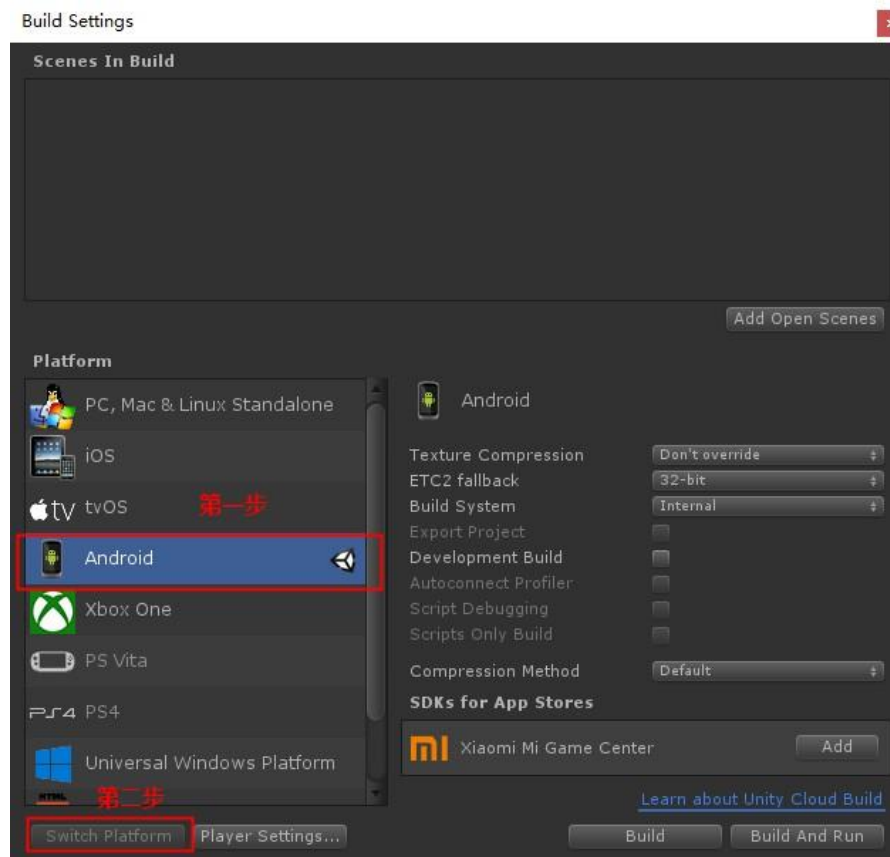
Picture 2.10 Build Settings Setting Diagram

2.5.5 AndroidManifest Setting

Set"Activity" to "com.ssnwt.sdk.MainActivity"

1.   If you use your own Manifest file in your project and also have configured your own Activity, then you need to have your Activity inherit from MainAcitivity.

2.   If your project is newly created, you should create directory "Plugins/Android/" in the project,then move "GoogleVR/Plugins/Android/AndroidManifest.xml" into "Plugins/Android/" directory you just create.

2.5.6 Blit Type Setting

There will be "Blit Type" setting option only in versions above Unity2017.3. **Here, you need to set up manually. For S801 platform, it shall be set to "Never", and for S8000 and 901 platforms, it shall be set to "Always".**
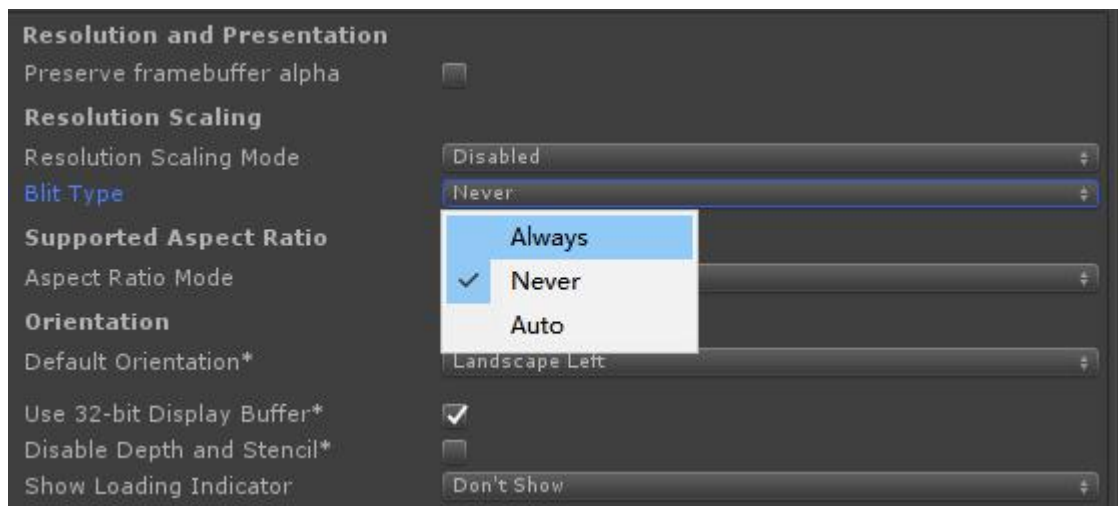
Picture 2.11 Blit Type Setting Diagram

2.6 Export to the equipment to run

1. Connect the equipment to the computer by USB. After connected to the computer, the battery icon on the Launcher window will change to charging status.

2. Click File->Build & Run, it's OK when the progress bar has finished!

3 API Interface Function

The SDK is extended based on GVR SDK v1.40.0. Here will introduce some of the interfaces, and for details of the rest of the interfaces, please go to the official website (https://developers.google.com/vr/unity/reference/).

3.1 GvrPointerInputModule Description
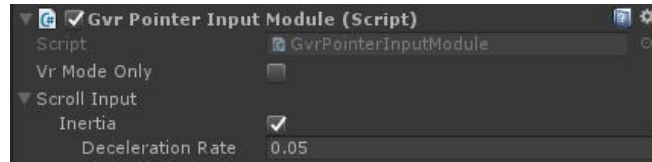
Picture 3.1 GvrPointerInputModule Setting Diagram

Function：

　　GvrPointerInputModule inherits from BaseInputModule, so, the use of this script can allow UI elements based on Canvas-based (UGUI) and 3D scene objects to interact in the application. Therefore, the event of UI elements selected by the Pointer or triggered by Trigger or Touching can be passed out.

Use:

　　Enter into Project option window,open the following in order: Assets->GoogleVR->Prefabs->UI, put the GvrEventSystem prefab into the scene to replace the original EventSystem.

3.2 GvrPointerPhysicsRaycaster Description

Picture 3.2 GvrPointerPhysicsRaycaster Setting Diagram

Function：

GvrPointerPhysicsRaycaster, which inherits from GvrBasePointerRaycaster, provides a collision detection used for GvrPointerInputModule.

Use:

Create an empty object in the scene and name it "Player". Drag the original "Main Camera" of the scene under the "Player". Add GvrPointerPhysicsRaycaster script for Main Camera.

3.3 GvrHead Description
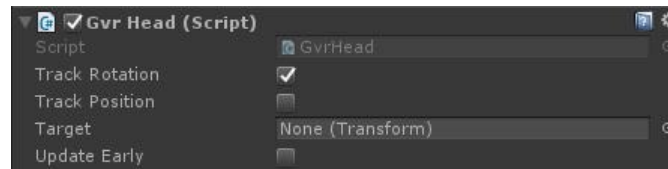
Picture 3.4 GvrHead Setting Diagram

Function：

The user head tracking simulation provides the head tracking data to the Camera, and the Transform property attached to it will also change synchronously during the head movement.。

Use:

Add GvrHead script for Main Camera.

4 3DoF Description of Handle and Helmet Buttons

4.1 Button Response of Handle or Helmet

1. Confirm button (helmet and handle)

GvrControllerInput.ClickButton

GvrControllerInput.ClickButtonDown

GvrControllerInput.ClickButtonUp

2. Return button (helmet)

Input.GetKey(KeyCode.Escape)

Input.GetKeyDown(KeyCode.Escape)

Input.GetKeyUp(KeyCode.Escape)

3. Home button (helmet and handle)

GvrControllerInput.HomeButton

GvrControllerInput.HomeButtonDown

GvrControllerInput.HomeButtonUp

4. Trigger button (handle)

GvrControllerInput.TriggerButton

GvrControllerInput.TriggerButtonDown

GvrControllerInput.TriggerButtonUp

5. App button （handle）

GvrControllerInput.AppButton

GvrControllerInput.AppButtonDown

GvrControllerInput.AppButtonUp

4.2　Interface Description

The SDK provides response events of buttons and touchpad of 3DoF handle controller. For calling handle related interfaces, please refer to the description of API interface function-GvrControllerInput in Chapter 3. Here introduces the corresponding relationship between interface parameters and physical handle buttons.

触摸板/确认
App键
Home键

Picture 5.1 3dof Handle Diagram

The corresponding relationship between the handle physical buttons and the parameters in the API interface functions is as follows:

Physical buttons	Relevant API interfaces
touchpad/confirm	ClickButton
	ClickButtonDown
	ClickButtonUp
App button	AppButton
	AppButtonDown
	AppButtonUp
Home button	HomeButtonDown
	HomeButtonState
Tigger button	TriggerButton
	TriggerButtonDown
	TriggerButtonUp

4.3　Shielding clicking handle Home button to return to Home

After the handle is connected, when the Home button is clicked, it will return to the Home window. To realize clicking the Home button does not return to Home, you need to add the following code into AndroidManifest.xml.

        <uses-feature android:name="android.software.vr.ignore.home" android:required="false" />

"True"indicates that clicking the Home button will not return to Home, "false"indicates that clicking the home button will return to Home, the default value is "false".

5 Gaze-click

The SDK provides the function of countdown to trigger click. When the handle operation is not needed, the gaze countdown can be used to trigger click events. This function needs to be used with gvrreticlepointer. (Note: the current 3DOF handle has become the standard interaction mode of VR standalone machine. For specific reasons, please use 3DOF handle as the default interaction mode.)

5.1 Use Instruction

In Assets->GoogleVR->Prefabs->UI, put the SvrReticleDownClick prefab under the GvrReticlePointer, then create a button in the scene and click it to run.When the anchor is hovered to the Button, a countdown will start.

It will work only when SvrReticleDownClick has checked objects that implement PointerClick. If you do not use PointerClick, it will have no effect.

By default, the countdown is 1 second, and we also provide the UICountDown component to control the countdown time.

5.2 UICountDown Component

When you need to control the countdown time of the button, you can plug the UICountDown.cs script in the corresponding PointerClick object, and then set the value of Count in seconds.

6 UI Configuration

GvrEventSystem is our own Eventsystem in the SDK. It supports click events in UGUI, highlight events 0. The handle supports drag events of scroll.

6.1 Create Canvas

Click "GameObject/UI/Canvas" to create a Canvas object in the scene. Set "Render Mode" to "World Space". Set position to (0 , 0 , 8) , set Scale to (0.008 , 0.008 , 0.008). Remove Graphic Raycast. Plug in GvrPointerGraphicRaycast script.
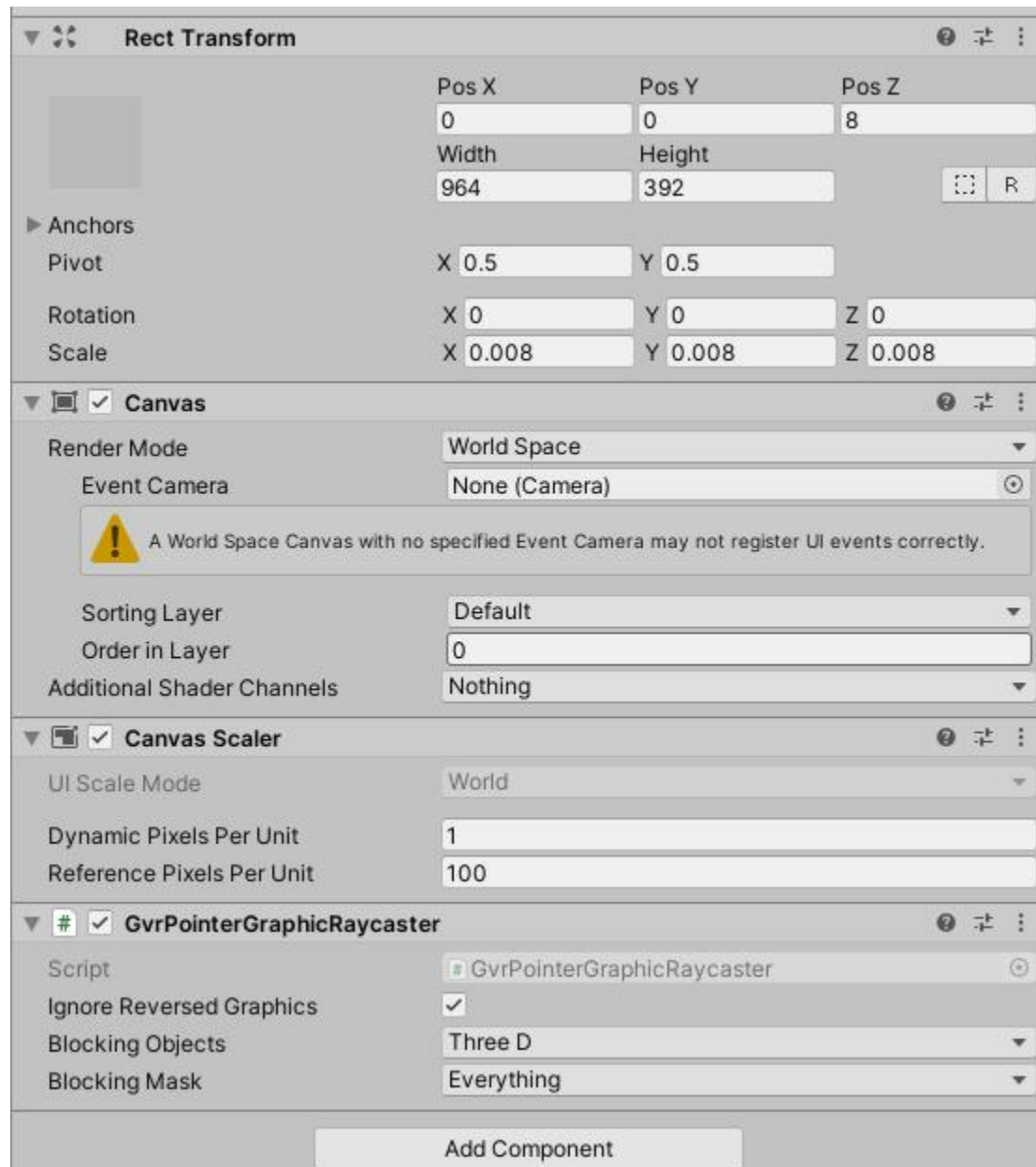
6.1 Canvas configuration

6.2 Available Event Range

The SDK's default available event range is within 20 meters. If it exceeds 20 meters, the event will not be triggered. In this case, you need to manually modify the maximum distance of the event.

Modify the value of Max Reticle Distance on Player/Main Camera/GvrReticlePointer to a proper size, and modify the value of Max Pointer Distance on Player/GvrControllerPointer/Laser to a proper size.

7. Using XR Plugin Management

Unity 2019.3 version launched XR Plugin Management, and we developed XRLoader for the XR framework. The file package is in "svr_unity(ver.2019&beyond)_sdk_plugin.tgz" in the SDK folder. First, select "SkyworthVR->Using S1 V901 Setting" in the status bar, and then click the "+" in the upper left corner of the Package Manager, then click "Add package from tarball... " to select "svr_unity(ver.2019&beyond)_sdk_plugin.tgz".

Note: If the version of Unity you are using is 2019.3.15 or above, it is recommended to use XRLoader, otherwise memory leaks may occur.

7.1 Using S1 V901 Setting

Need to switch the setting to S1 V901 setting. Only S1 and S901 support XR mode of Unity.
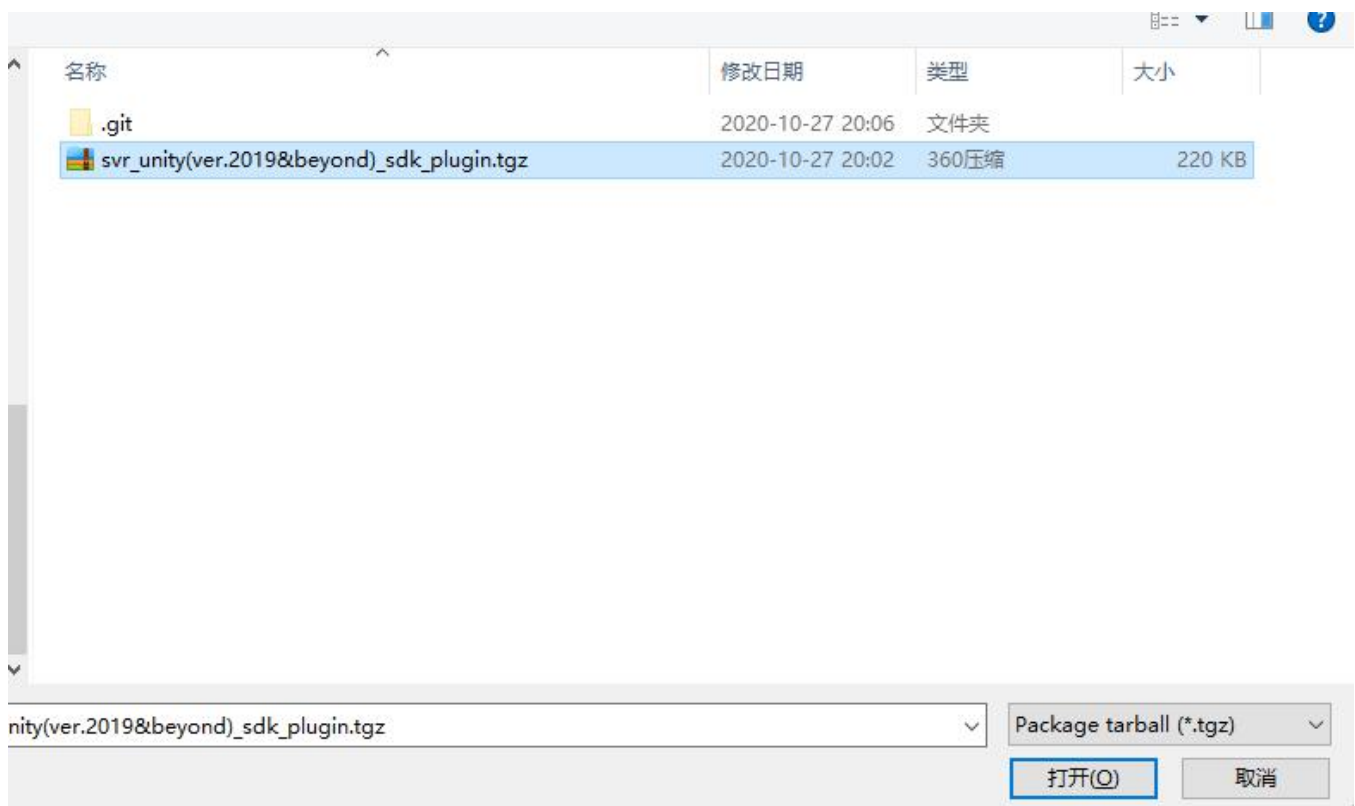
7.2 Import svr_unity(ver.2019&beyond)_sdk_plugin.tgz

Open window/Package Manager, click "+" in the upper left corner, click "Add package from tarball…"
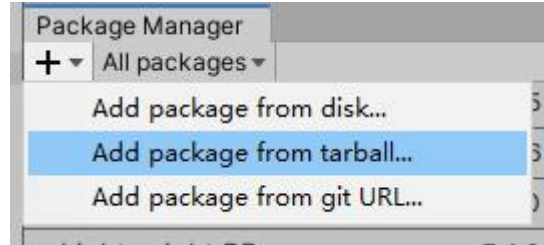
Picture 7.2.1 Import Button

Open the folder of package,choose "tgz" file, click it to open.

Picture 7.2.2 Choose package.json

Wait the completion of import of unity, then check if there is "Skyworth XR Plugin" in Package Manager。
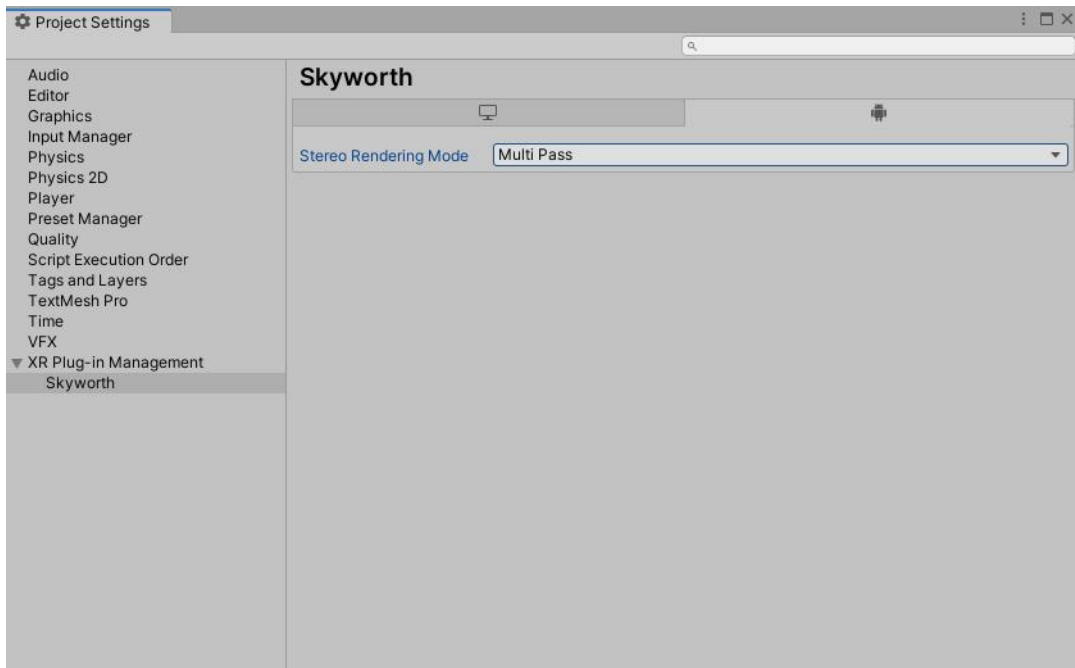
Picture 7.2.3 Package Manager

Open Editor/ProjectSetting, open "XR Plugin-in Management" in it，select"Android Setting"page, select "Skyworth" under "Plugin- in Providers".

The interface page of XR Plugin Management of 2019.3 version is different from that of 2019.4 version.
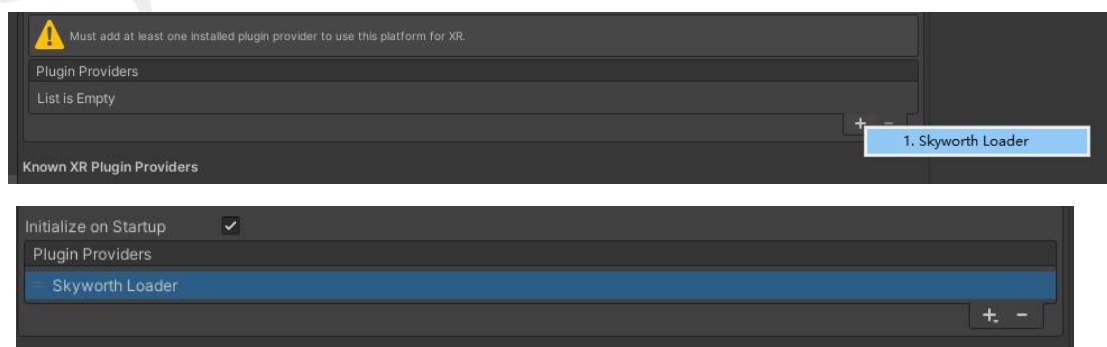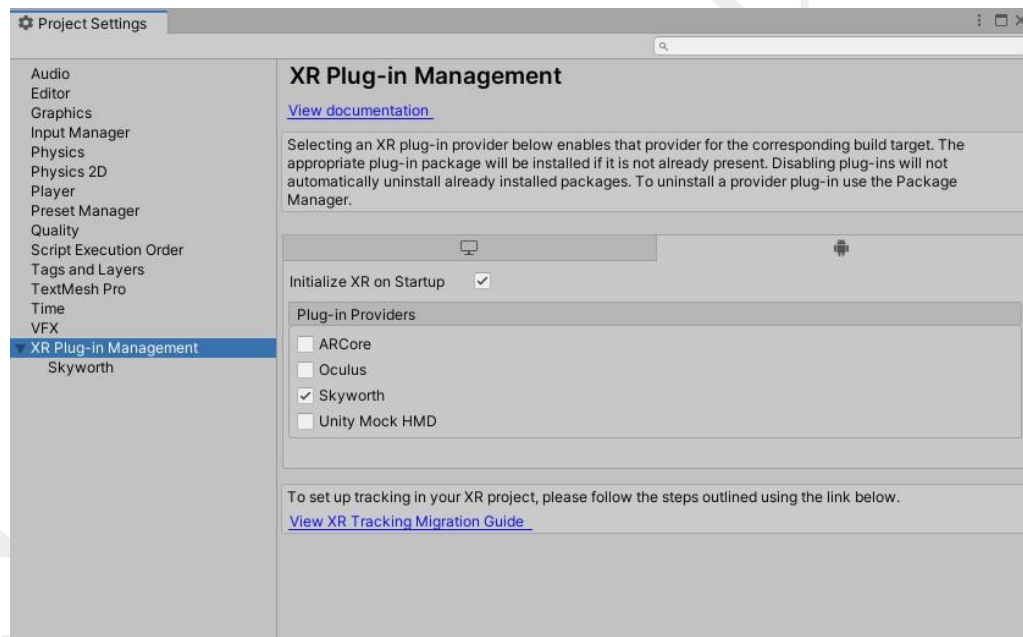
Picture 7.2.3 Select"Skyworth"

Then click "Skyworth" under XR Plugin-in Management to open "Android Settings", set "Stereo Rendering Mode" to "Multi Pass".

Picture 7.2.4 Set to "multi pass"

Note: for Unity 2019.4, it's required to update the XRPluginManager to version 3.2.16, otherwise,the following errors may occur:

Package Metadata store error. Check if there are any other errors in the console and make sure they are corrected before trying again.

Failure reason: Error installing package com.unity.xr.skyworth. ErrorCode: Failure Error Message: Unable to add package[com.unity.xr.skyworth]:

Package [com.unity.xr.skyworth] cannot be found UnityEditor.EditorApplication:Internal_CallUpdateFunctions()

8 QA

Q： How to conceal the anchor?

A： Call interface GvrControllerInput.GvrPointerEnable, set "true" to display anchor, set "false" to conceal anchor.

Q： How to remove the animation appearing when application starts up?

A： Open Assets/Plugins/Android/assets/splash.cfg, set "USE_SVR_SPLASH" to 0.