

秘别:

编号:



Skyworth Standalone VR SDK (Unity)

开发文档

拟制 _____ 日期 _____

审核 _____ 日期 _____

批准 _____ 日期 _____

深圳创维新世界科技有限公司

1 简介	3
1.1 SDK 介绍	3
1.2 开发环境	3
1.3 SDK 简介	3
2 SDK 的使用说明	3
2.1 新建工程	3
2.2 导入 UnityPackage	3
2.3 使用 SDK	5
2.5 项目设置	6
2.5.1 Using S1 V901 Setting	6
2.5.2 Using S801 Setting	7
2.5.2 QualitySettings 设置	8
2.5.3 PlayerSettings 设置	9
2.5.4 Build Settings 设置	10
2.5.5 AndroidManifest 设置	11
2.5.6 Blit Type 设置	11
2.6 导出到设备运行	12
3 API 接口函数	13
3.1 GvrPointerInputModule 说明	13
3.2 GvrPointerPhysicsRaycaster 说明	13
3.3 GvrHead 说明	13
4 3DoF 手柄与头盔按键说明	15
4.1 手柄或头盔的按键响应	15
4.2 接口说明	16
4.3 屏蔽手柄 Home 键返回到 Home	16
5 凝视点击	17
5.1 使用说明	17
5.2 组件 UICountDown	17
6 UI 配置	17
6.1 创建 Canvas	17
6.2 可用事件范围	18
7. 使用 XR Plugin Management	18
7.1 Using S1 V901 Setting	19
7.2 导入 com.unity.xr.skyworth@1.0.0	19
8 QA	21

1 简介

1.1 SDK 介绍

SDK 支持硬件设备：创维 VR 一体机 V901 , S1

SDK 主要提供：3dof 手柄交互支持，NoI6D0F，多功能交互支持，双目立体渲染等功能

1.2 开发环境

1. Unity: 推荐使用 LTS 版本的 Unity，这是 Unity 长期支持的版本，相对其他版本更加稳定。
2017: 支持 2017.4.6 以上
2018: 支持 2018.4.13 以上
2019: 支持 2019.3.2 以上
2. Android SDK: API Level 25 及以上。
3. JDK: jdk1.7.0_01 及以上。

1.3 SDK 简介

SDK 是一个 UnityPackage，包含必要的代码和资源。使用时将其导入工程中，具体使用方法下面的章节会介绍。建议不要和其他的 VR SDK 一起使用，工程中不要有其他厂商的 SDK，避免出现冲突。

2 SDK 的使用说明

2.1 新建工程

打开 Unity，新建工程如下：

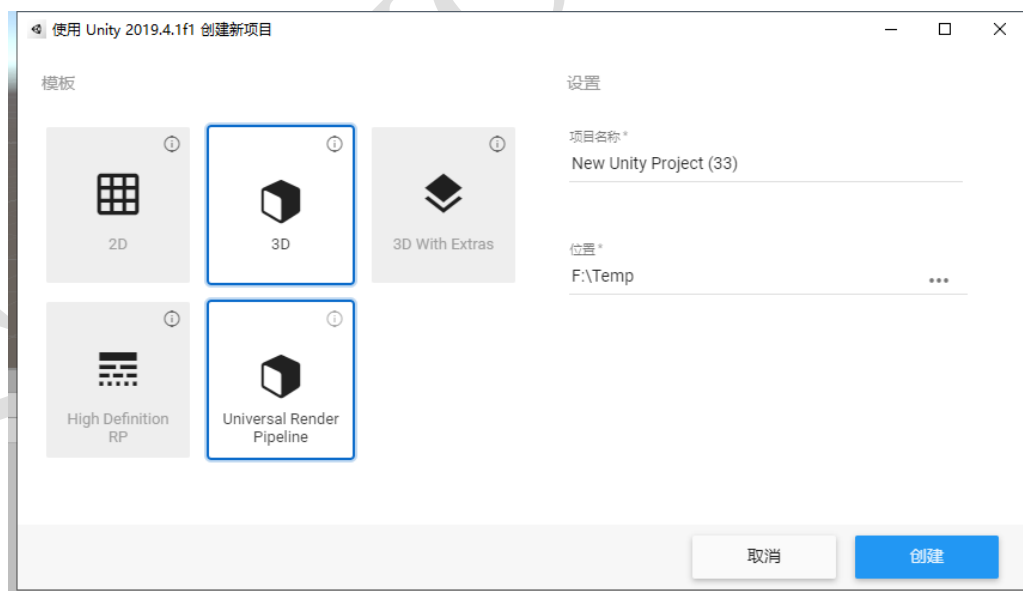


图 2.1 新建工程

2.2 导入 UnityPackage

选择菜单 Assets->Import Package->Custom Package...如下：

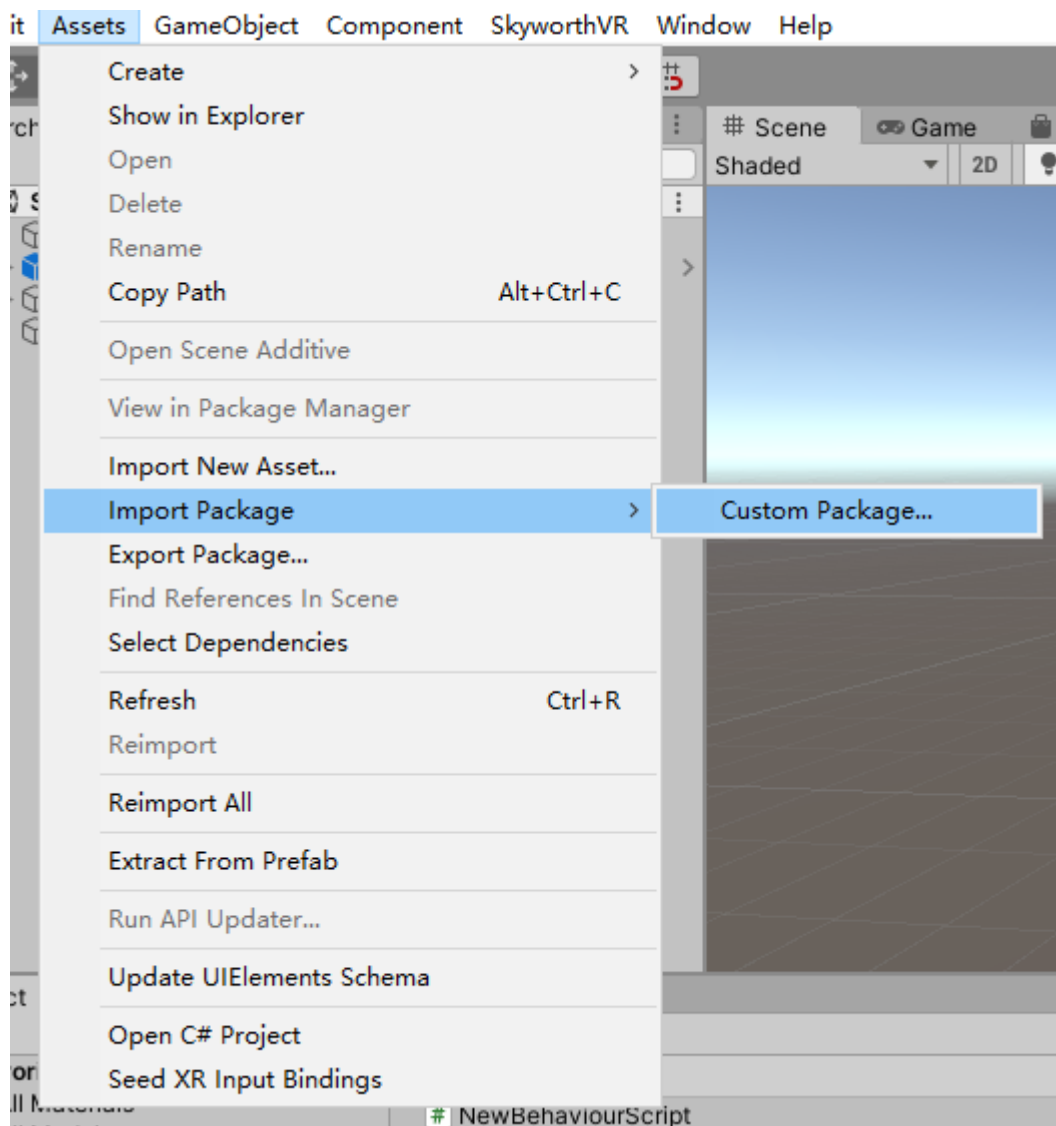


图 2.2 导入 SDK

此时 Unity 会弹出文件选择对话框，选择“svr_unity_sdk.unitypackage”后点击打开如下：

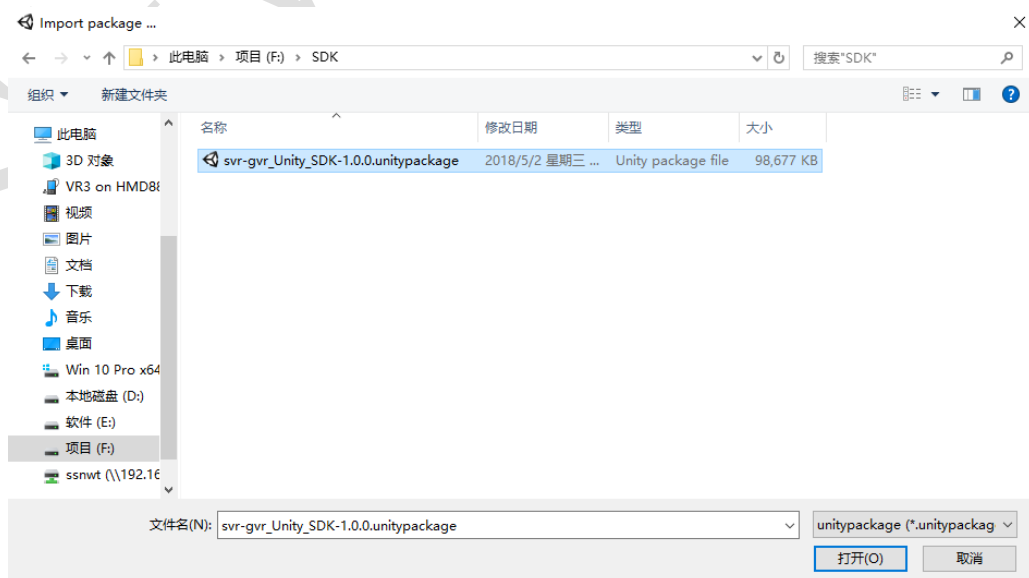


图 2.3 选择 Unity Package

点击后返回至 Unity 界面，系统弹出 SDK 目录层级，请根据需要导入：

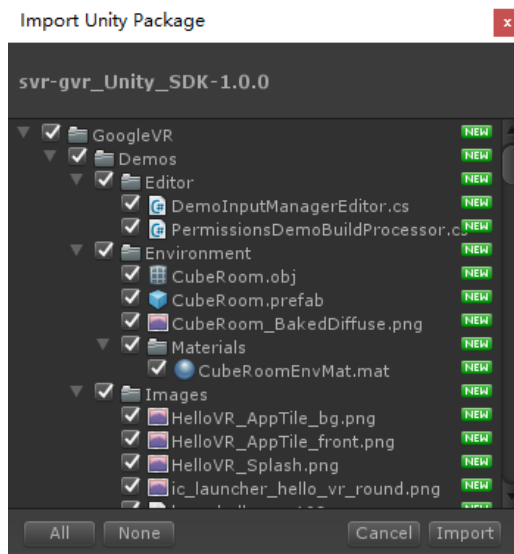


图 2.4 导入选项

2.3 使用 SDK

进入 Project 选项卡，依次展开 Assets->GoogleVR->Demos->Scenes，选择 HelloVR 场景，点击运行按钮，在 Game 窗口中可看到如下：

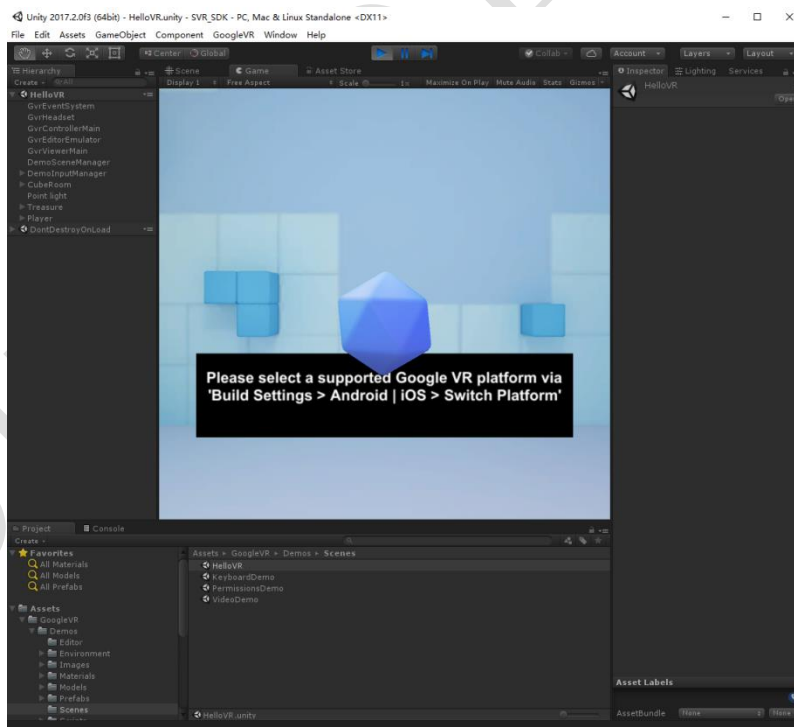


图 2.5 模拟运行

按住 **Alt** 移动鼠标，可上下左右转动画面；按住 **Ctrl** 移动鼠标，可上下翻转画面；按住 **Shift** 移动鼠标，可模拟手柄操作。

2.5 项目设置

导入 UnityPackage 后，点开菜单栏上的 SkyworthVR，里面有 Using S1 V901 Setting，Using S801 Setting，Using S1 V901 Legacy Setting. 如果你的 unity 版本是 2017 或者 2018 的版本看到的是图 2.6，如果是 2019 版本是图 2.7。

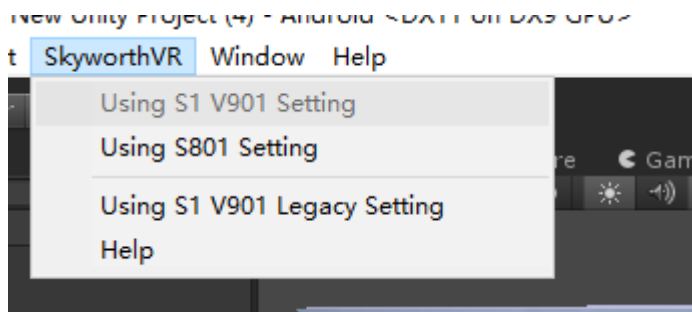


图 2.6 sdk 设置

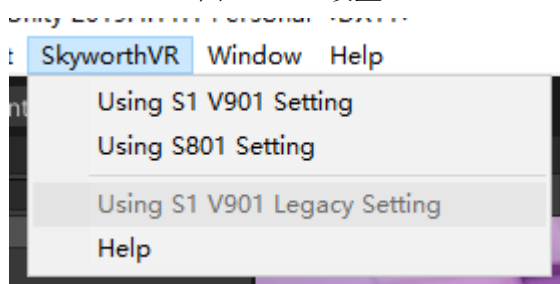


图 2.7 2019 设置

2.5.1 Using S1 V901 Setting

该设置用于导出在 S1 或者 V901 上运行的 apk，会对工程做以下设置：

1. Blit Type: Always

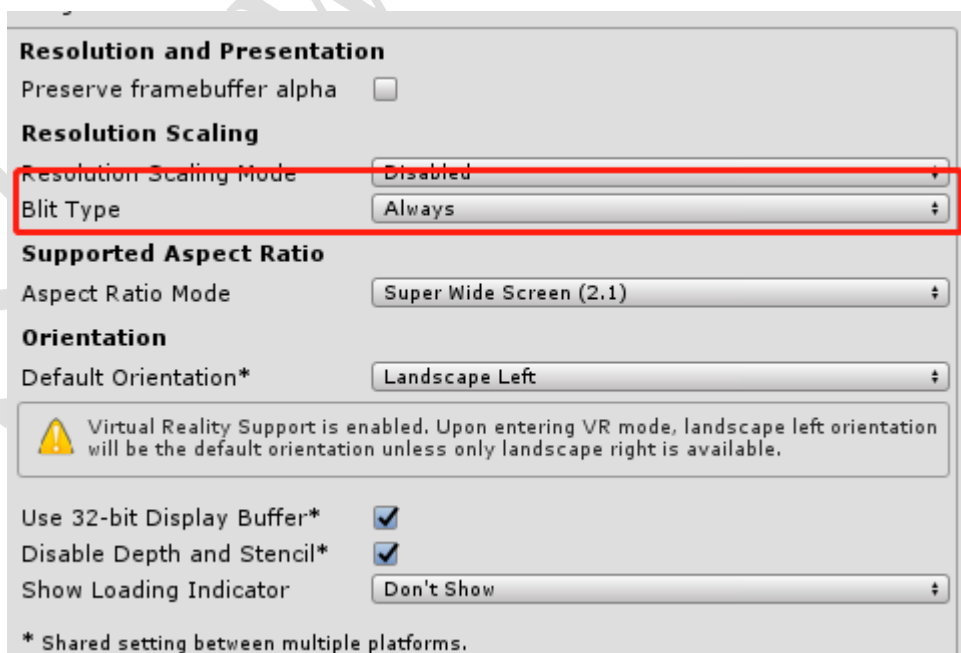


图 2.8 blit type

2. Scripting Define Symbols: SVR

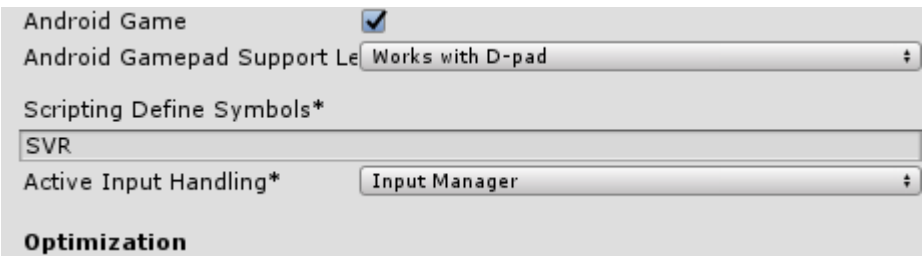


图 2.9 Scriptng Define Symbols

3. XR Settings

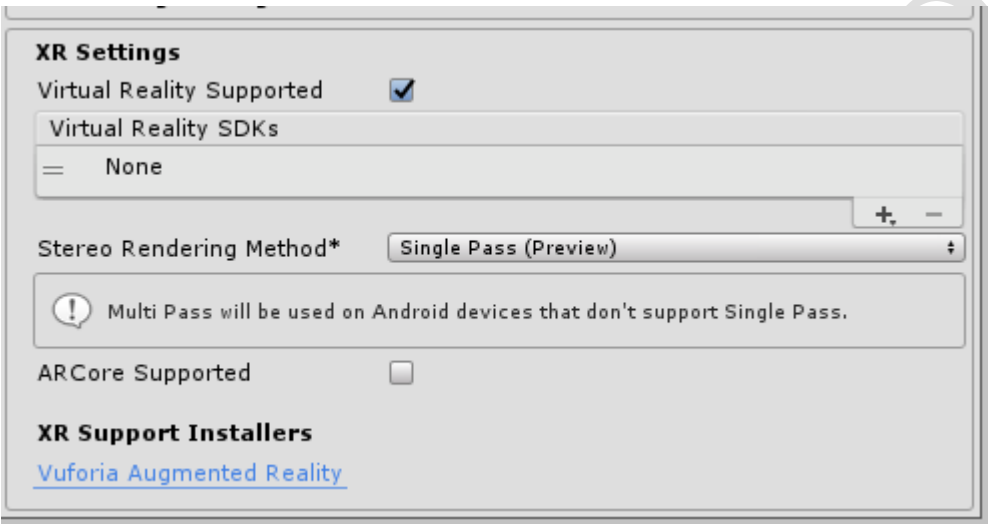


图 2.10 XR Settings。

2.5.2 Using S801 Setting

该设置用于导出在 S801 上运行的 apk，使用后会对工程做以下设置：

1. Blit Type:Always

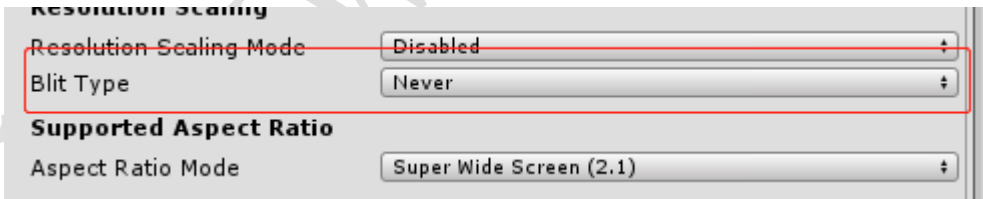


图 2.11 blit type

2. Script Define Symbols

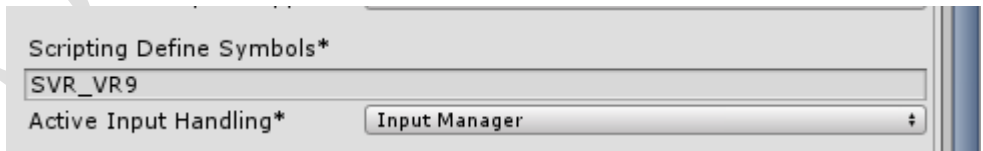


图 2.12

3. XR Settings



图 2.13 XR Settings

2.5.2 QualitySettings 设置

如图 2.6 中 Levels 选择安卓平台绿色勾选项的“Medium”等级。具体参数如下图 2.6 所示：



图 2.6 QualitySettings 设置示意图

注意：Levels 设置中要先让灰色条选中到 Android 平台一列打绿色勾的行，如图 2.6 所示意。然后再设置 Anti Aliasing 和 V Sync Count 两项。如果不先选择到 Android 平台，这两项目设置就不会在 Android 导出包中

生效，导致应用运行是黑屏的。

Anti Aliasing: 可以根据需要进行调整，推荐使用 4 倍抗锯齿。

V Sync Count: 一定要设置为 Don't Sync。

2.5.3 PlayerSettings 设置

1. Resolution and Presentation 选项卡中，导出设置为横屏。具体参数如下图 2.7 所示：

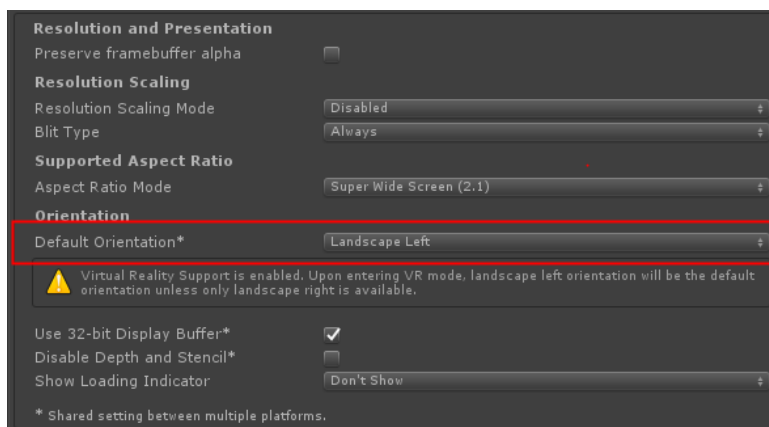


图 2.7 导出设置示意图

2. Other Settings 选项卡中，不要勾选 GPU Skinning, API Level 为 Android 7.1, 读写权限根据需要选择 SDCard。具体参数如下图 2.8 所示：

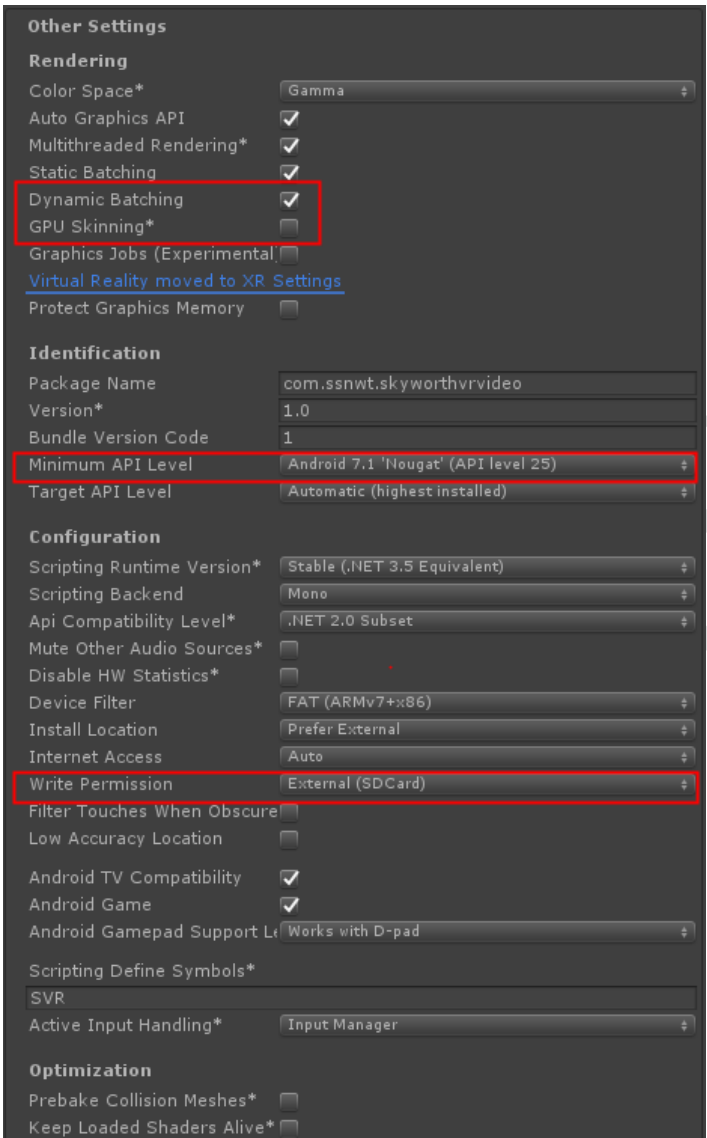


图 2.8 PlayerSettings 设置示意图

2.5.4 Build Settings 设置

选择默认平台“Android”，构建系统(Build System)选择“Internal”编译方式。具体参数如下图 2.10 所示：

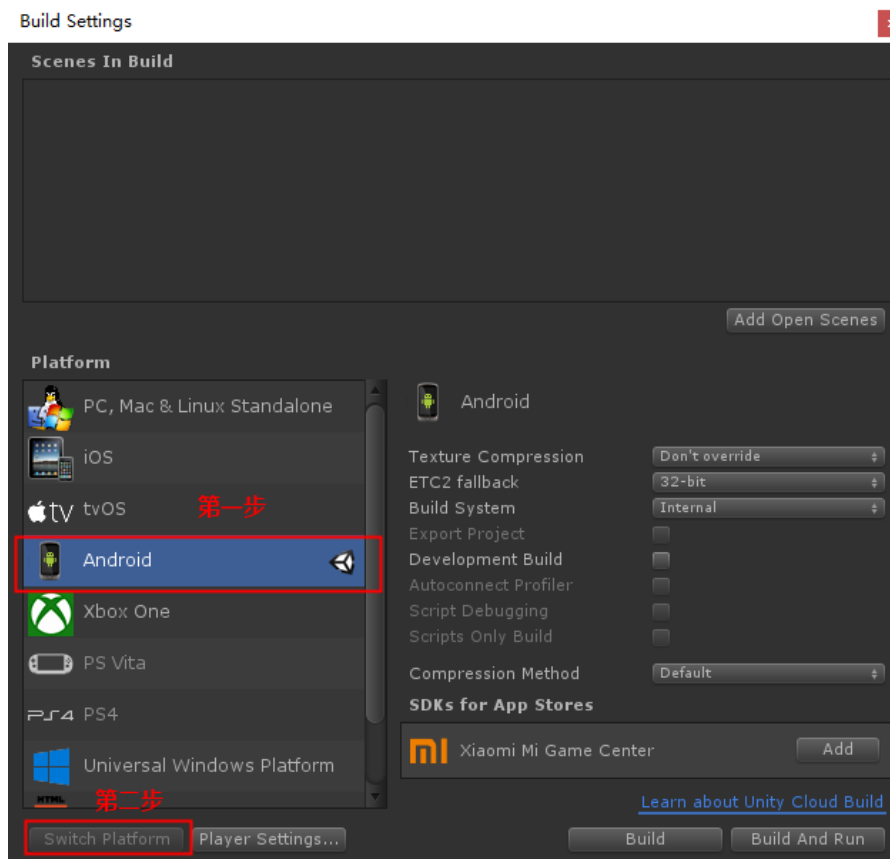


图 2.10 Build Settings 设置示意图

2.5.5 AndroidManifest 设置

将 Activity 配置为 “com.ssnwt.sdk.MainActivity”

1. 如果你的工程里使用了自己的 Manifest 文件并也配置了自己的 Activity，那么需要将你的 Activity 继承自 MainActivity。

2. 如果你是新建的工程，需要在工程里创建 Plugins/Android/ 目录，然后将 GoogleVR/Plugins/Android/AndroidManifest.xml 文件移动到刚才创建的 Plugins/Android/ 目录。

2.5.6 Blit Type 设置

在 Unity2017.3 以上的版本才会有 Blit Type 的设置项。这里需要手动设置，S801 平台需要设置成 Never，S8000 和 901 平台需要设置成 Always。

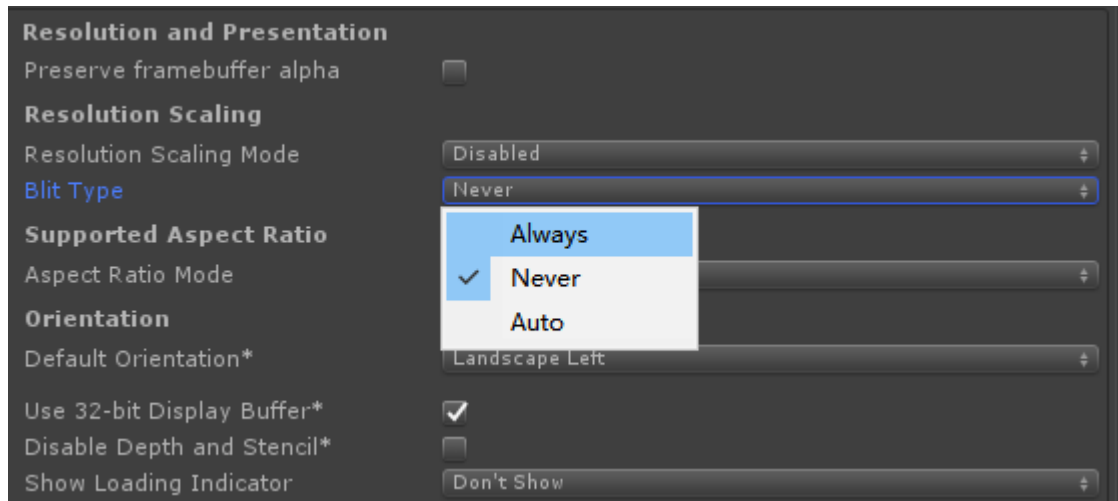


图 2.11 Blit Type 设置示意图

2.6 导出到设备运行

- 1.将设备通过 USB 连接到电脑，当连上电脑之后，Launcher 界面的电池图标会变成充电状态。
- 2.点击 File->Build & Run，等待进度条结束就 OK 了！

3 API 接口函数

该 SDK 以 GVR SDK v1.40.0 为基础拓展，下面介绍部分接口，其余接口请前往官方网站 (<https://developers.google.com/vr/unity/reference/>) 获取详情。

3.1 GvrPointerInputModule 说明

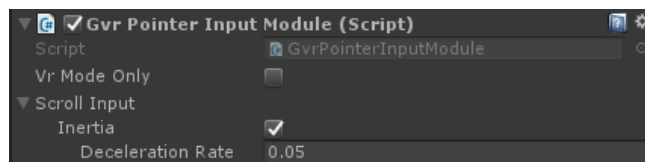


图 3.1 GvrPointerInputModule 设置示意图

功能:

GvrPointerInputModule 继承自 BaseInputModule, 使用此脚本可以让基于 Canvas 的(UGUI)UI 元素和 3D 场景对象在应用程序中进行交互。所以 UI 元素在被 Pointer 选中或 Trigger, Touching 触发的事件, 可以传递出来。

使用:

进入 Project 选项卡, 依次展开 Assets->GoogleVR->Prefabs->UI, 将 GvrEventSystem 预制体放入场景替换原有的 EventSystem。

3.2 GvrPointerPhysicsRaycaster 说明

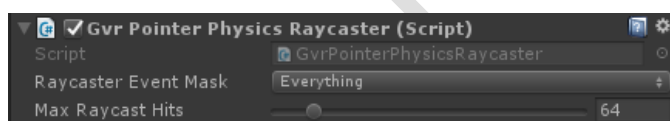


图 3.2 GvrPointerPhysicsRaycaster 设置示意图

功能:

GvrPointerPhysicsRaycaster 继承自 GvrBasePointerRaycaster, 提供了一个用于 GvrPointerInputModule 的碰撞检测。

使用:

在场景中新建一个空对象, 为其命名为 Player。将场景原有 Main Camera 拖动至 Player 之下。为 Main Camera 添加 GvrPointerPhysicsRaycaster 脚本。

3.3 GvrHead 说明

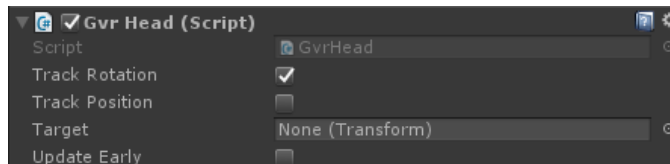


图 3.4 GvrHead 设置示意图

功能:

用户头部跟踪模拟, 提供头部跟踪的数据给 Camera, 在头部运动过程中它附着的 Transform 属性也会同

步发生改变。

使用：

为 Main Camera 添加 GvrHead 脚本。

SKYworthVR

4 3DoF 手柄与头盔按键说明

4.1 手柄或头盔的按键响应

1.确认键（头盔和手柄）

GvrControllerInput.ClickButton

GvrControllerInput.ClickButtonDown

GvrControllerInput.ClickButtonUp

2.返回键（头盔）

Input.GetKey(KeyCode.Escape)

Input.GetKeyDown(KeyCode.Escape)

Input.GetKeyUp(KeyCode.Escape)

3.Home 键（头盔和手柄）

GvrControllerInput.HomeButton

GvrControllerInput.HomeButtonDown

GvrControllerInput.HomeButtonUp

4.Trigger 键（手柄）

GvrControllerInput.TriggerButton

GvrControllerInput.TriggerButtonDown

GvrControllerInput.TriggerButtonUp

5.App 键（手柄）

GvrControllerInput.AppButton

GvrControllerInput.AppButtonDown

GvrControllerInput.AppButtonUp

4.2 接口说明

SDK 提供 3DoF 手柄控制器的按键与触摸板响应事件，调用手柄相关接口请参照第三章 API 接口函数——GvrControllerInput 说明，这里介绍接口参数与物理手柄按键的对应关系。

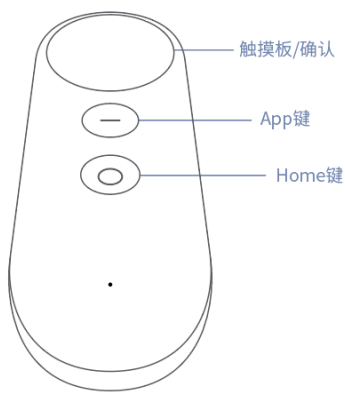


图 5.1 3dof 手柄示意图

手柄物理按键与 API 接口函数中参数的对应关系如下：

物理按键	API 相关接口
触摸板/确认	ClickButton ClickButtonDown ClickButtonUp
App 键	AppButton AppButtonDown AppButtonUp
Home 键	HomeButtonDown HomeButtonState
Tigger键	TriggerButton TriggerButtonDown TriggerButtonUp

4.3 屏蔽手柄 Home 键返回到 Home

当连接手柄后点了 Home 按键会返回的 Home 界面，如果想实现点 Home 键不返回到 Home 需要在 AndroidManifest.xml 添加一下代码。

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

“true”表示点击 Home 不返回到 Home，“false”表示点击 Home 键会返回到 Home，默认是 false。

5 凝视点击

SDK 中提供了倒计时出发点击的功能，当不需要手柄操作时可以使用凝视倒计时来触发点击事件。该功能需要和 GvrReticlePointer 一起使用。（注：当期 3DoF 手柄已成为 VR 一体机标准交互方式，如特定的原因，请默认使用 3DoF 手柄作为交互方式。）

5.1 使用说明

在 Assets->GoogleVR->Prefabs->UI 中将 SvrReticleDownClick 预制体放在 GvrReticlePointer 下面，然后在场景中创建一个 Button，点击运行。当锚点 Hover 到 Button 上时会出现倒计时效果。

SvrReticleDownClick 只会检查实现了 PointerClick 的物体才起作用，如果你不是使用的 PointerClick 将没有任何效果。

默认情况下倒计时是 1 秒，我们也提供了 UICountDown 组件来控制倒计时时间。

5.2 组件 UICountDown

当需要控制按钮倒计时的时间时可以将 UICountDown.cs 脚本挂在相应 PointerClick 的物体上，然后设置 Count 的值，单位是秒。

6 UI 配置

在 SDK 里面有自己的 Eventsystem 是 GvrEventSystem。支持 UGUI 里面的点击事件，高亮事件 0，手柄支持 scroll 的拖动事件。

6.1 创建 Canvas

点击 GameObject/UI/Canvas，会在场景中创建一个 Canvas 对象。将 Render Mode 设置成 World Space。将位置设置成 (0, 0, 8)，缩放设置成 (0.008, 0.008, 0.008)。移除 Graphic Raycast。挂上 GvrPointerGraphicRaycast 脚本。



6.1 Canvas 设置

6.2 可用事件范围

SDK 默认可用事件是 20 米以内，如果超出 20 米会导致无法触发事件的问题，这时就需要手动修改事件的最大距离。

修改 Player/Main Camera/GvrReticlePointer 上面的 Max Reticle Distance 的值到合适大小，修改 Player/GvrControllerPointer/Laser 上面的 Max Pointer Distance 的值到合适大小。

7. 使用 XR Plugin Management

Unity2019.3 版本推出了 XR Plugin Management，我们针对 XR 框架开发出了 XRLoader。文件在 SDK 文件夹里面的 com.unity.xr.skyworth@1.0.0.tgz，将它解压后，通过 PackageManager 导入到工程。

7.1 Using S1 V901 Setting

需要将设置切换到 S1 V901 设置，只有 S1 和 S901 支持 Unity 的 XR 模式。

7.2 导入 com.unity.xr.skyworth@1.0.0

打开 window/Package Manager，点击左上角的“+”符号，点击“Add package from disk...”

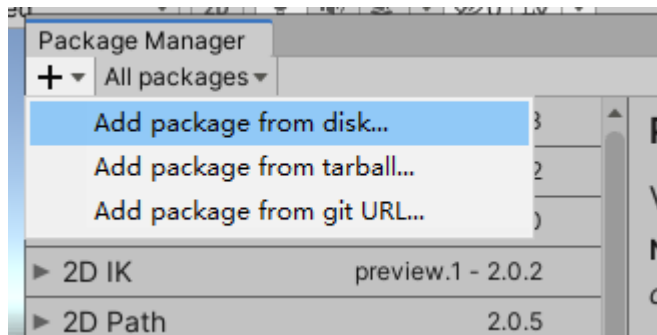


图 7.2.1 导入按钮

打开包文件夹，选中 package.json，点击打开。

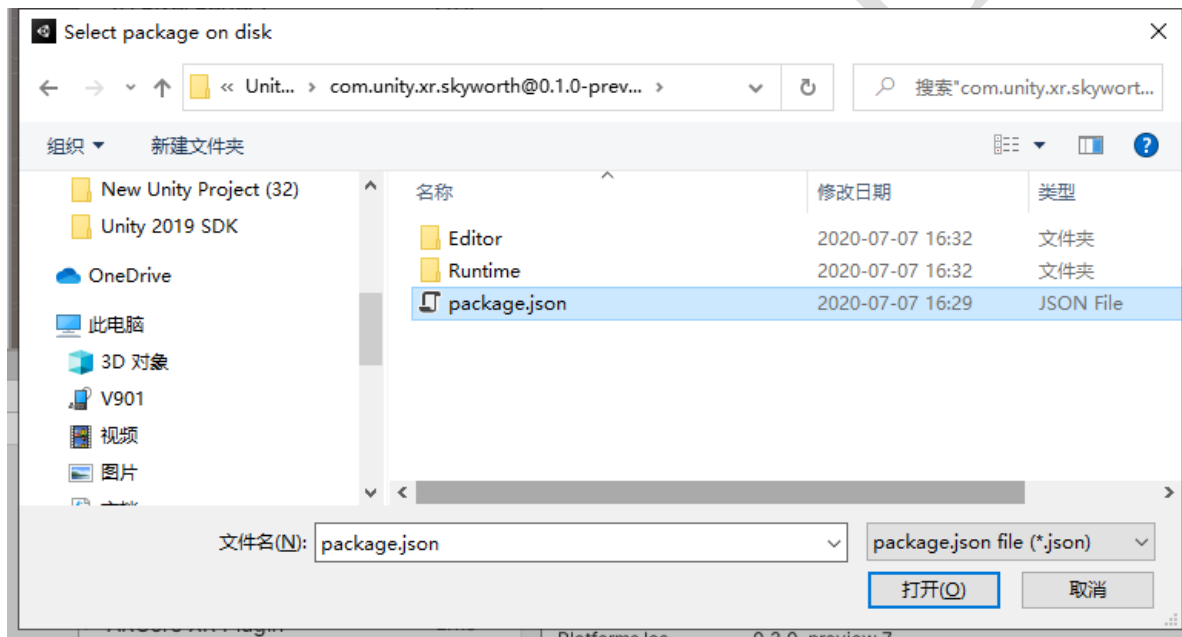


图 7.2.2 选中 package.json

等待 unity 导入，然后在 Package Manager 里面看是否有“Skyworth XR Plugin”。

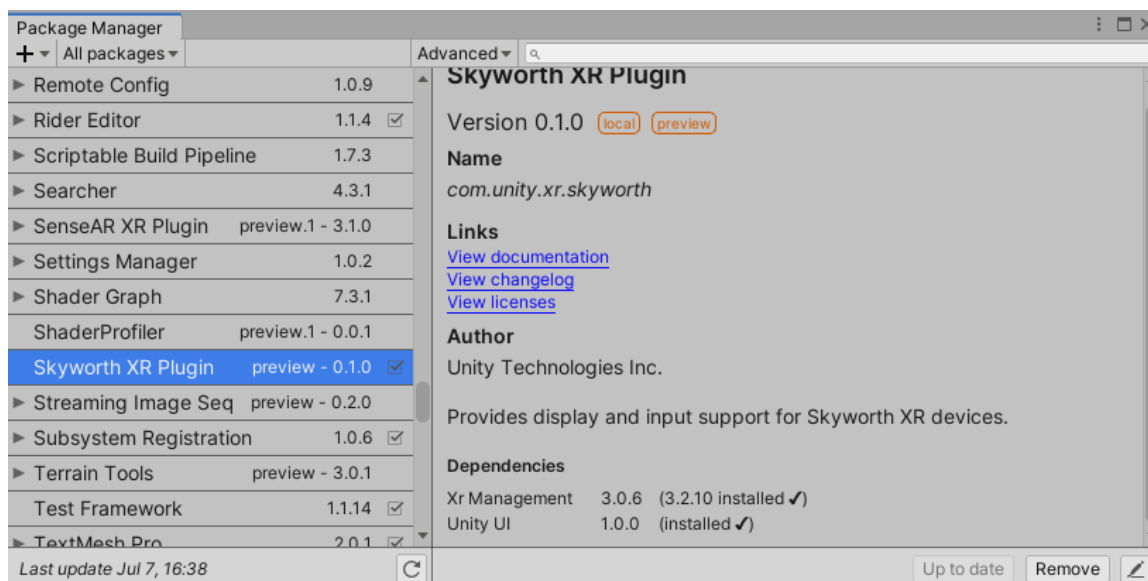


图 7.2.3 Package Manager

打开 Editor/ProjectSetting, 打开里面的 XR Plugin-in Management, 选中 Android Setting 页, 在 Plugin-in Providers 中选中 Skyworth。

在 2019.3 版本里面的 XR Plugin Management 界面和 2019.4 版本不同。

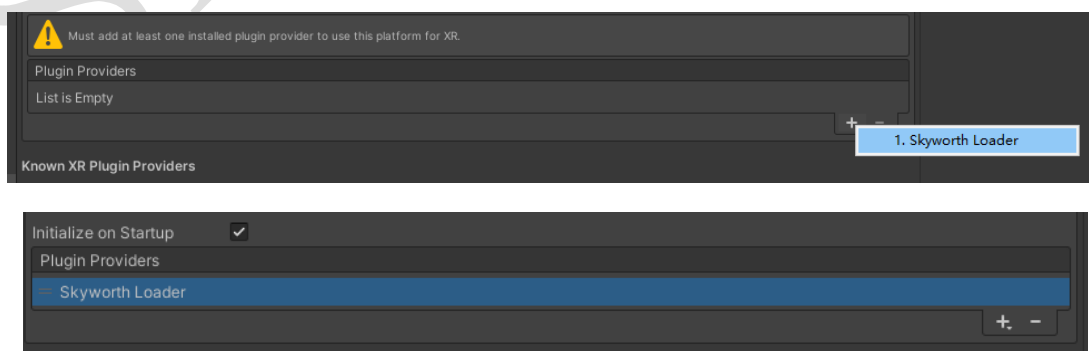
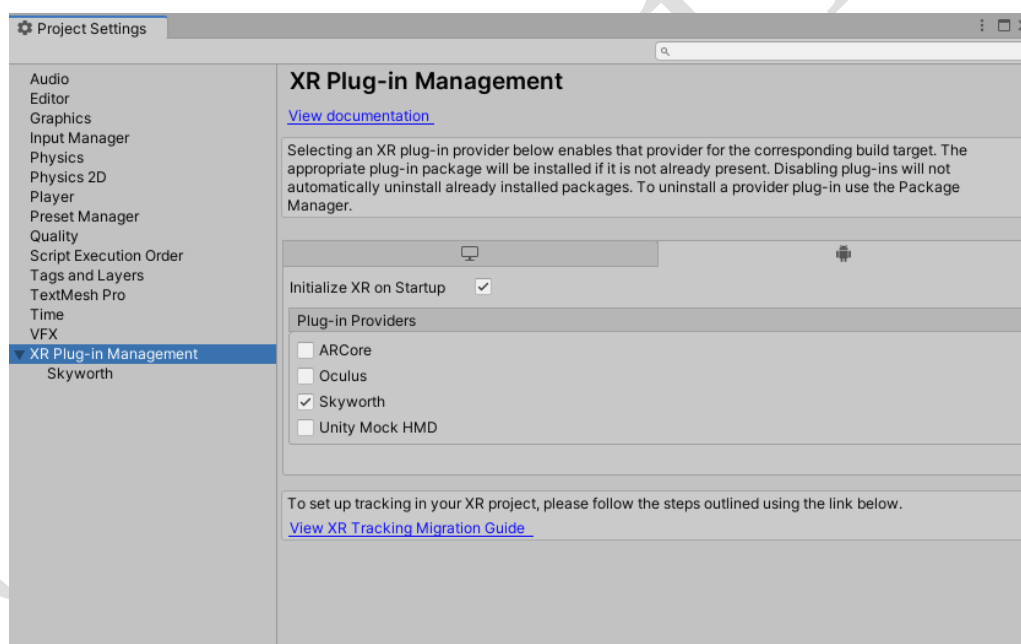


图 7.2.3 选中 Skyworth

然后点击 XR Plugin-in Management 下面的 Skyworth, 打开 Android Settings, 将 Stereo Rendering Mode 设置成 Multi Pass。

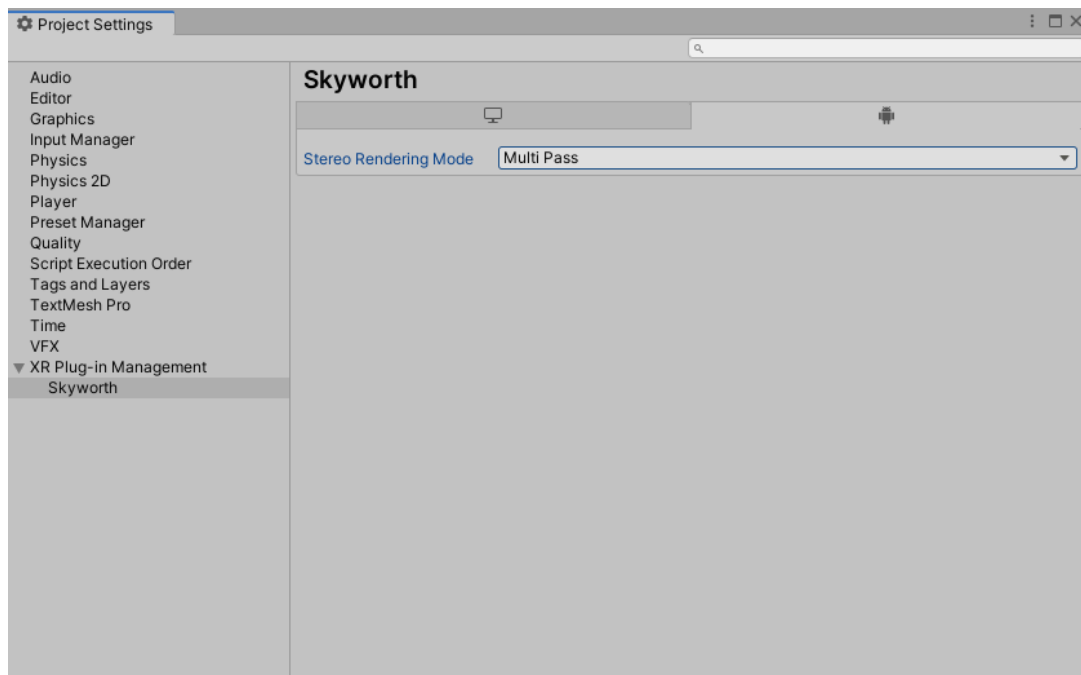


图 7.2.4 设置成 multi pass

8 QA

Q: 如何隐藏锚点?

A: 调用接口 `GvrControllerInput.GvrPointerEnable`, `true` 显示锚点, `false` 隐藏锚点。

Q: 如何去掉应用启动的动画?

A: 打开 `Assets/Plugins/Android/assets/splash.cfg`。将 `USE_SVR_SPLASH` 设置成 0。