

NOLO VR Unity SDK

接口说明

北京凌宇智控科技有限公司

2020 年 9 月

目录

一、 简介.....	3
1.NOLO VR Unity SDK.....	3
二、 接入准备.....	3
三、 使用说明.....	4
1. 快速入门.....	4
2. NOLO Prefab.....	6
3. 调试说明.....	9
四、 接口说明.....	10
1. Button 事件.....	10
2. Touch 事件.....	11
3. 震动事件.....	12
4. 定位数据.....	12
5. 提交错误信息.....	12
6. 设备连接状态.....	13
7. 设备电量信息.....	13
五、 注意事项.....	13
1.标定原点位置.....	13
2.设置 AppKey.....	14
3.标定方向.....	14
4.射线检测.....	14
5.NOLO 提供的 Demo 场景说明.....	17
6.Universal RP(通用渲染管线)配置说明.....	18
7.NOLO SDK 支持 Armv8.....	19
8.打包规范.....	19
9.联系方式.....	20

一、简介

1.NOLO VR Unity SDK

NOLO VR Unity SDK 是由 NOLO Inc. 开发，方便 Unity 开发者开发移动端、6-DoF 的 VR 游戏。

二、接入准备

软件需求：Unity5.6 以上版本

NOLO HOME PC 版本下载地址：

http://download.nolovr.com/download/NOLO_home_PC.html

请联系 dev@nolovr.com 获取 Appkey，并填写到你的 Unity 项目中，调试阶段可以使用公用 Appkey，正式上线的时候请修改为正式的 Appkey，公用 Appkey：4e4f4c4f484f4d457eff82725bc694a5。

项目测试包名：com.nolo.xr.unitydemo，apk 包名与 NibiruSDKKey.bin 文件绑定。如需修改包名，详见 Nolo SDK 包名修改说明 3.1 文档

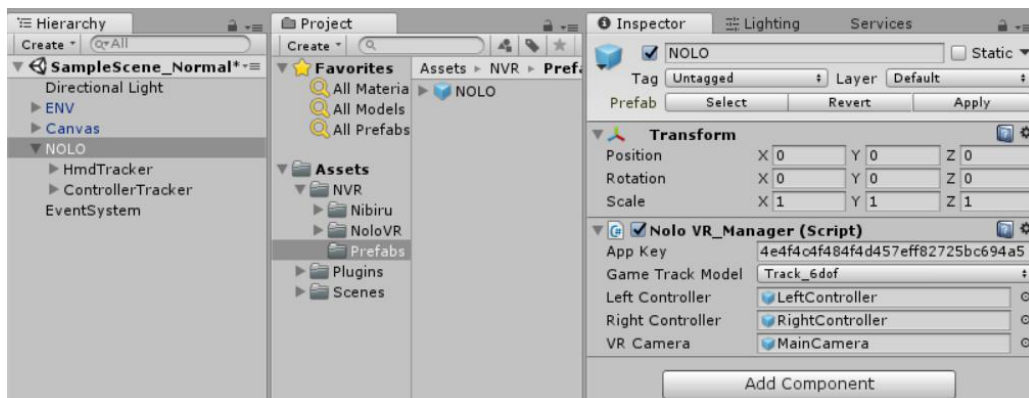
硬件需求：NOLO X1 一体机

三、使用说明

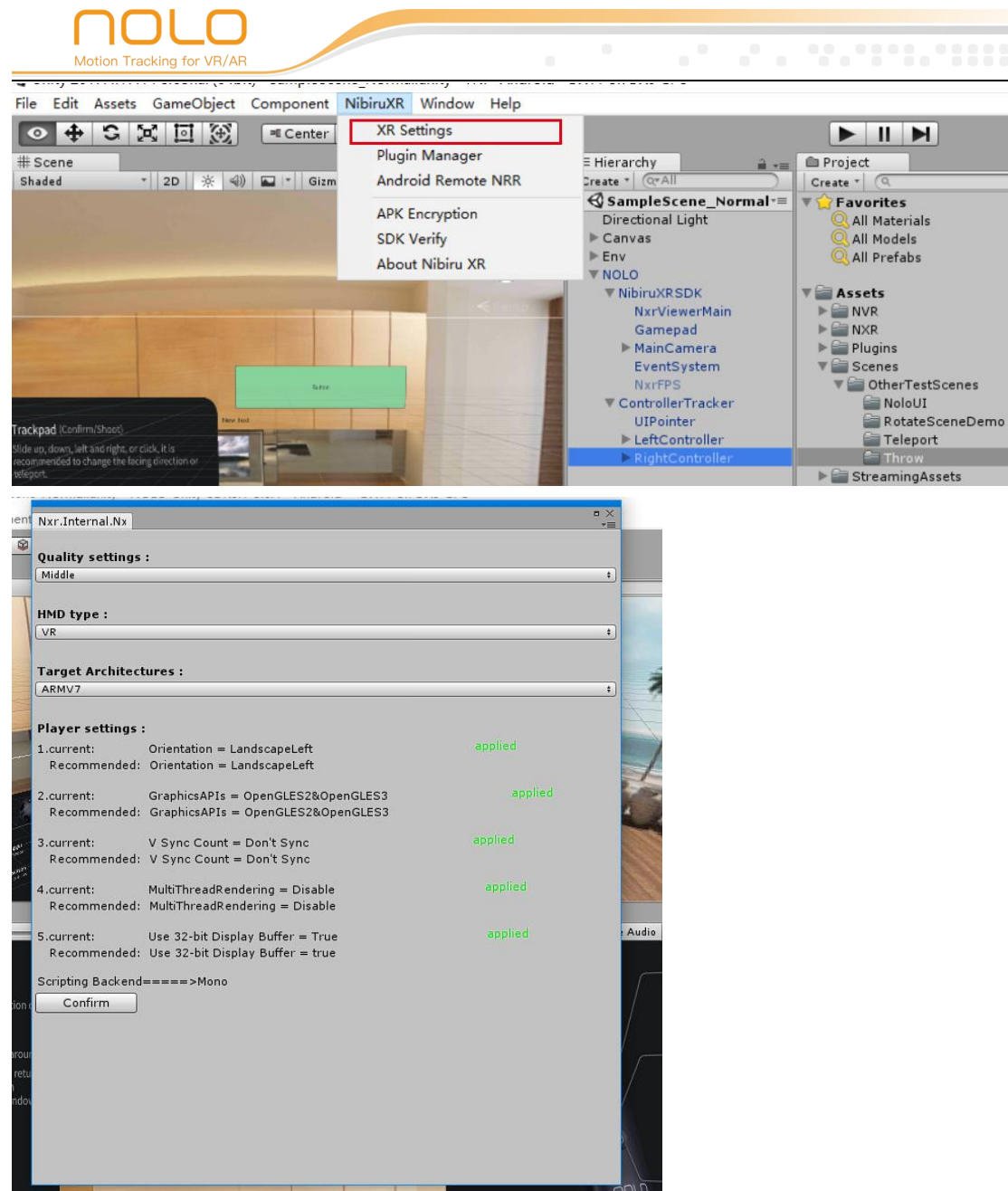
1. 快速入门

(1) 一体机工程

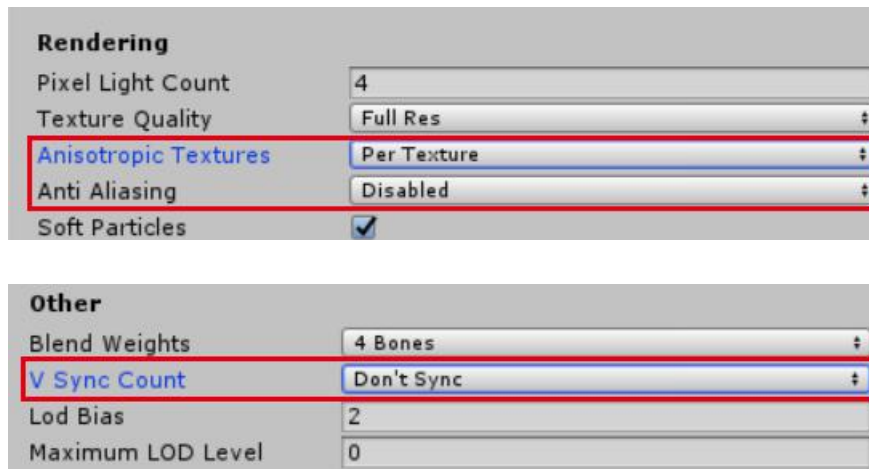
- 1) 新建一个 Unity 工程，将 NOLO VR Unity SDK 导入工程中。
- 2) 新建一个场景，将 NVR/Prefabs/NOLO 放入场景中，并保存。
- 3) 将 Appkey 填写到如下位置



- 4) Player Settings 设置：点击菜单 NibiruXR->XR Settings,
Quality Settings: 对应 Unity 中的阴影设置 Shadow Resolution
HMD type: 默认 VR 类型
Target Architectures: 分别对应 32 位, 64 位
点击 Confirm 完成环境配置



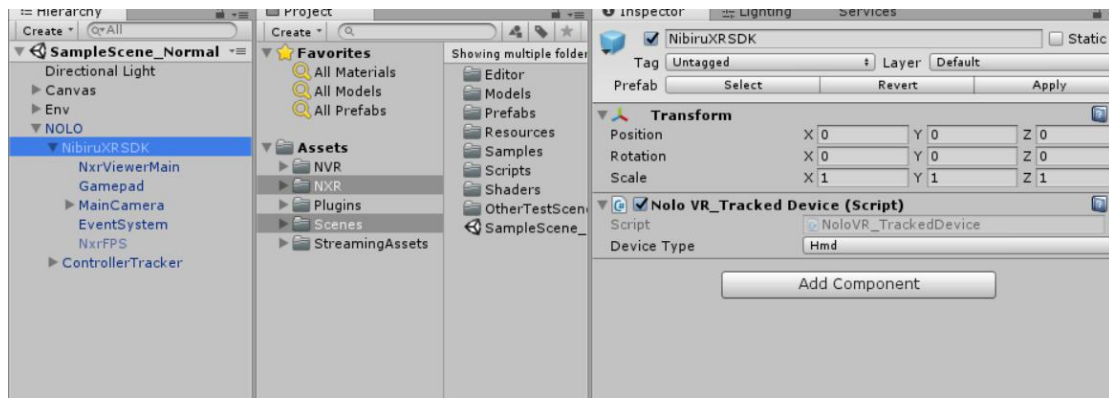
5) Quality Settings 设置： 'Rendering' 中， 'Anisotropic Textures' 设置为 'Per Texture' ， 'Anti Aliasing' 设置为'Disabled' 。 'Other' 中 'Sync Count' 设置为'Don't Sync' 。

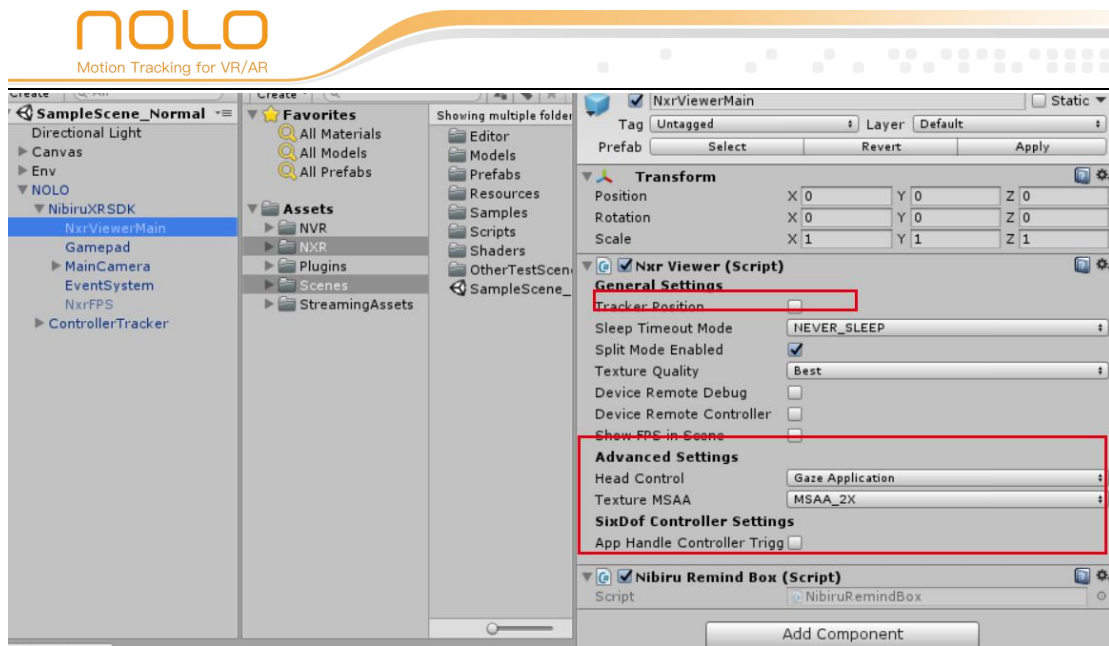


6) 填写正确的包名信息，即可打包到 NOLO X1 一体机上运行

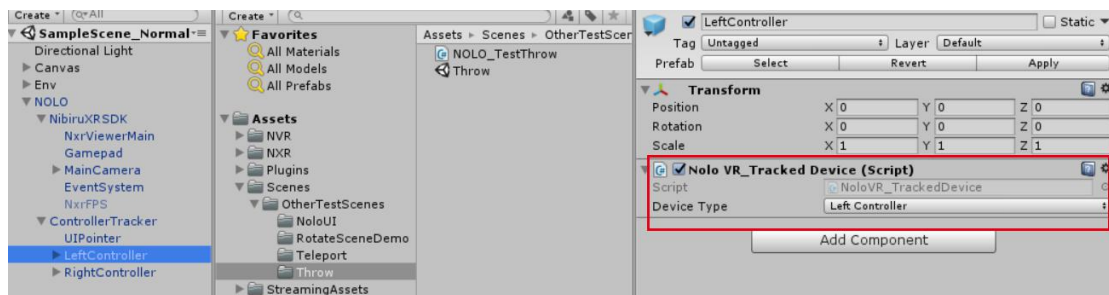
2.NOLO Prefab

1) 头部组件，包含头部定位信息，摄像机信息

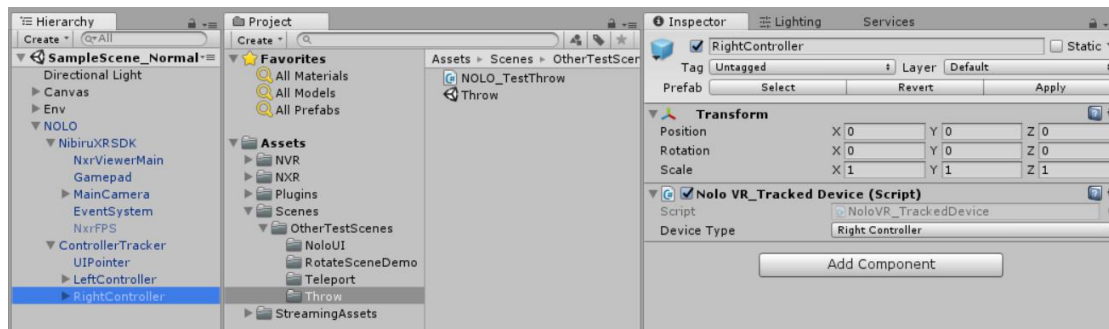




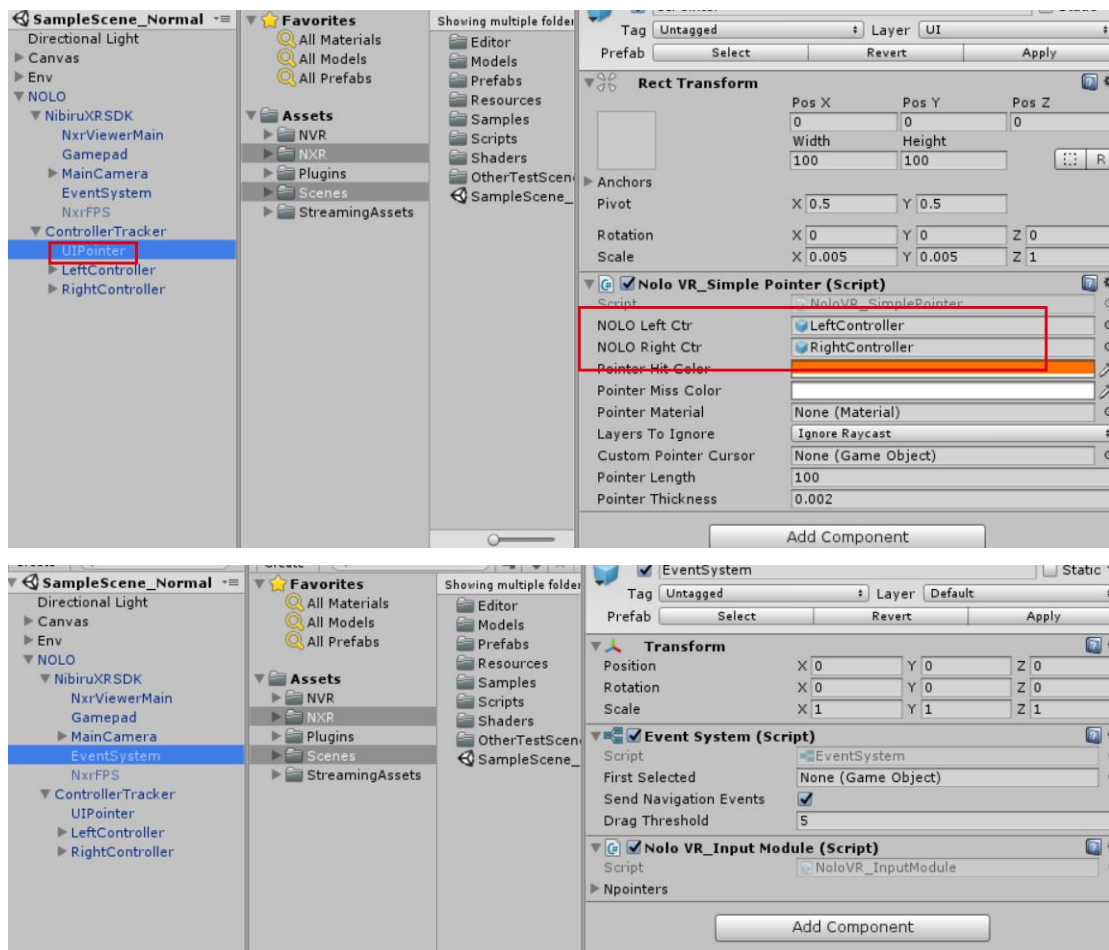
2) 左手组件，包含左手定位信息

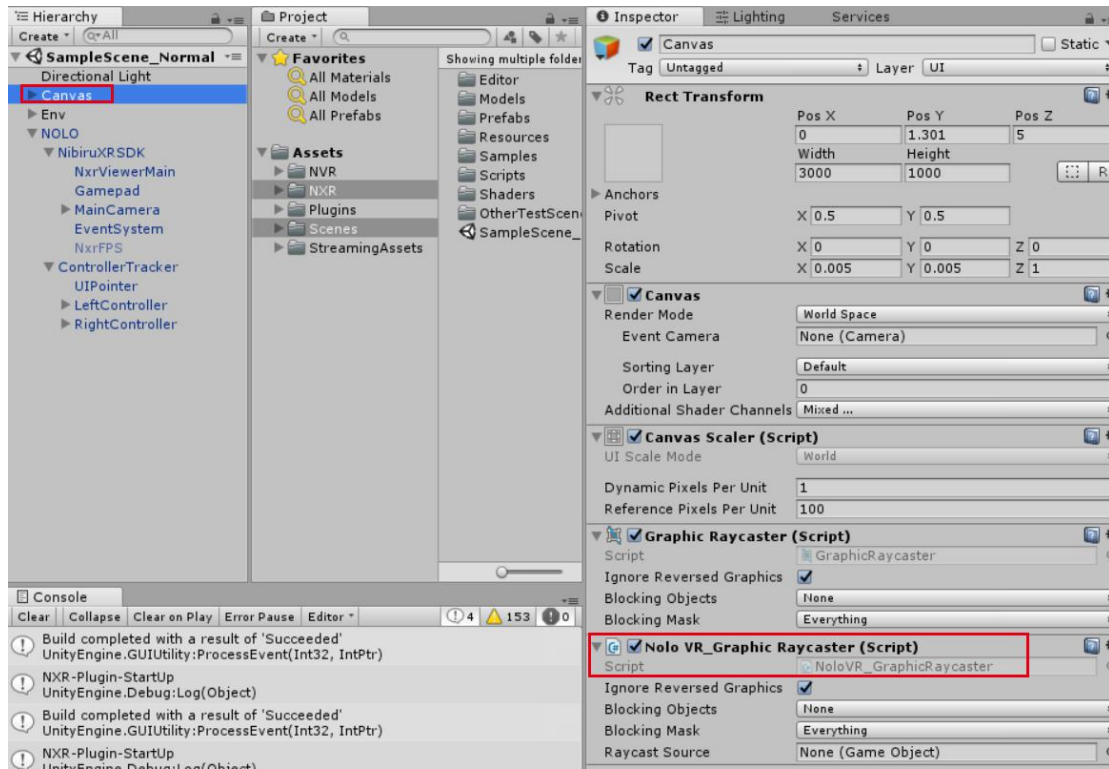


3) 右手组件，包含右手定位信息



4) 射线组件，用来切换左右手柄射线





3. 调试说明

Unity Editor 中调试: 请将且仅将 NOLO 头盔定位器用 usb 线连接至电脑, 开启 NOLO HOME PC 版本, 其他 NOLO 设备打开, 待 NOLO HOME 上显示 NOLO 所有设备的电量信息后, 点击 Unity 的 Run 按钮, 即可在 Unity Editor 进行调试。

安卓端调试: 请在移动设备上安装 NOLO HOME, 在项目中填写正确的 Appkey, 在未审核之前, 可以使用测试 key 进行调试。同样用 OTG 线, 将 NOLO 头盔定位器连接到一体机上, 如果提示“是否运行 NOLO HOME 访问 USB 设备”时, 点击确定, 即可在你的 APP 中获取到 NOLO 数据。

四、接口说明

1.Button 事件

函数名称	bool GetNoloButtonPressed()
功能描述	获取按键长按状态
参数	Enum NoloButtonID
返回值	bool
先决条件	NoloVR_Controller.GetDevice()

函数名称	bool GetNoloButtonDown()
功能描述	获取按键从抬起到按下状态
参数	Enum NoloButtonID
返回值	bool
先决条件	NoloVR_Controller.GetDevice()

函数名称	bool GetNoloButtonUp()
功能描述	获取按键从按下到抬起状态
参数	Enum NoloButtonID
返回值	bool
先决条件	NoloVR_Controller.GetDevice()

2.一体机按键

按键图标	Unity 键值
OK	<i>KeyCode.JoystickButton0 or KeyCode.Joystick2Button0</i>



KeyCode.Escape

3.Touch 事件

函数名称	bool GetNoloTouchPressed()
功能描述	获取触摸板触摸状态
参数	Enum NoloTouchID
返回值	bool
先决条件	NoloVR_Controller.GetDevice()

函数名称	bool GetNoloTouchDown()
功能描述	获取触摸板从非触摸状态到触摸状态
参数	Enum NoloTouchID
返回值	bool
先决条件	NoloVR_Controller.GetDevice()

函数名称	bool GetNoloTouchUp()
功能描述	获取触摸板从触摸状态到非触摸状态
参数	Enum NoloTouchID
返回值	bool
先决条件	NoloVR_Controller.GetDevice()

函数名称	Vector2 GetAxis()
功能描述	获取触摸板触摸点的坐标数据
参数	Enum NoloTouchID:默认值为触摸板，其他参数无效。（见附录）
返回值	Vector2

先决条件	NoloVR_Controller.GetDevice()
------	-------------------------------

4. 震动事件

函数名称	void TriggerHapticPulse()
功能描述	调用手柄震动
参数	int:震动强度, 范围 50~100
返回值	void
先决条件	NoloVR_Controller.GetDevice()

5. 定位数据

函数名称	Nolo_Transform GetPose()
功能描述	获取设备定位信息
参数	Null
返回值	Nolo_Transform
先决条件	NoloVR_Controller.GetDevice()

6. 提交错误信息

函数名称	void ReportError ()
功能描述	收集游戏错误信息
参数	string
返回值	void
先决条件	NoloVR_Playform.GetInstance()

6. 设备连接状态

函数名称	bool GetNoloConnectStatus()
功能描述	获取 NOLO 设备连接状态信息
参数	int/NoloDeviceType
返回值	bool
先决条件	NoloVR_Plugins.GetNoloConnectStatus()

7. 设备电量信息

函数名称	int GetElectricity()
功能描述	获取 NOLO 设备电量信息
参数	int/NoloDeviceType
返回值	int, 范围 (0~5)
先决条件	NoloVR_Plugins.GetElectricity()

五、注意事项

1. 标定原点位置

NOLO 正常运行时，将头盔定位器放置在地面上，按一下头盔定位器上的按钮，此时，这个位置就会成为游戏中的初始位置，即引擎中“NOLO”所在的位置。标定原点操作是会被记录到设备中的，只有在基站位置发生很大变化时，重新执行一次标定操作即可。

2.设置 AppKey

游戏必须设置 AppKey，才能正常运行，请在游戏 NOLO 对象上 NoloVR_Manager 脚本中正确填写 AppKey。

请联系 dev@nolovr.com 获取 Appkey，或者使用公用 Appkey 进行开发测试。

公用 Appkey: 4e4f4c4f484f4d457eff82725bc694a5。

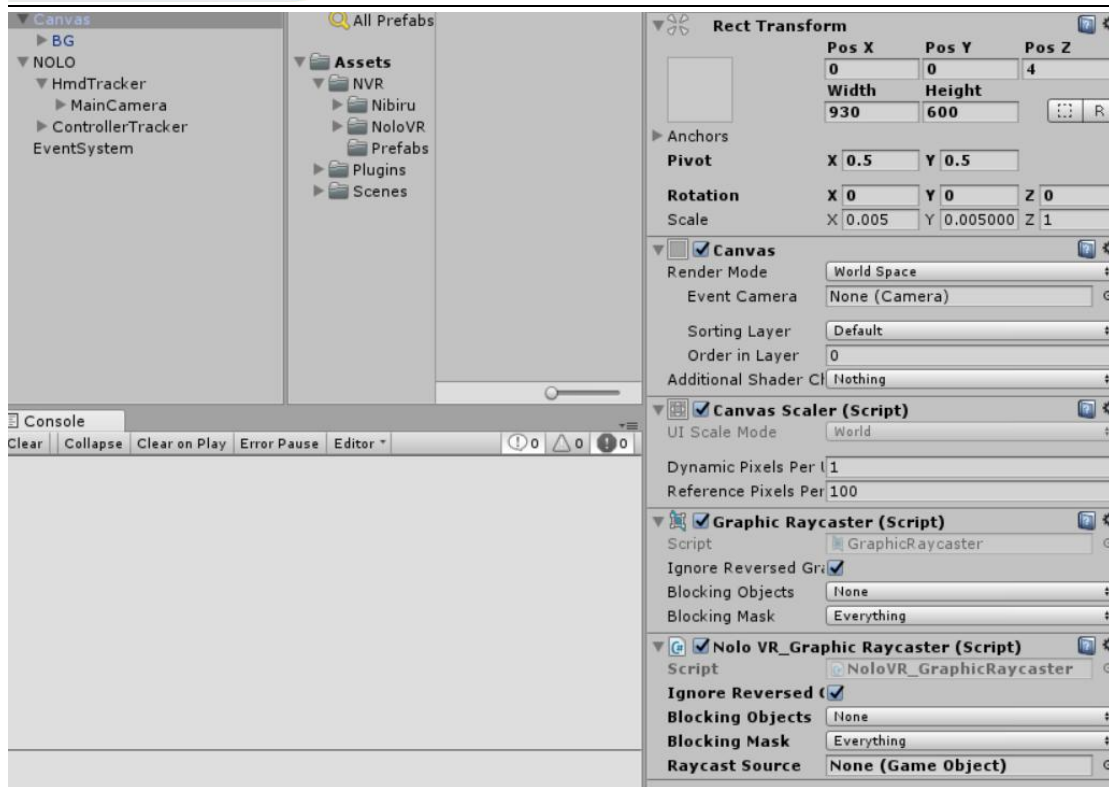
3.标定方向

游戏启动之后，如果游戏正方向不是基站所在方向，需要面对基站双击其中一个手柄的电源键来标定正方向，同样，如果手柄的方向不对，也请面对基站，双击这个手柄的电源键标定正方向。

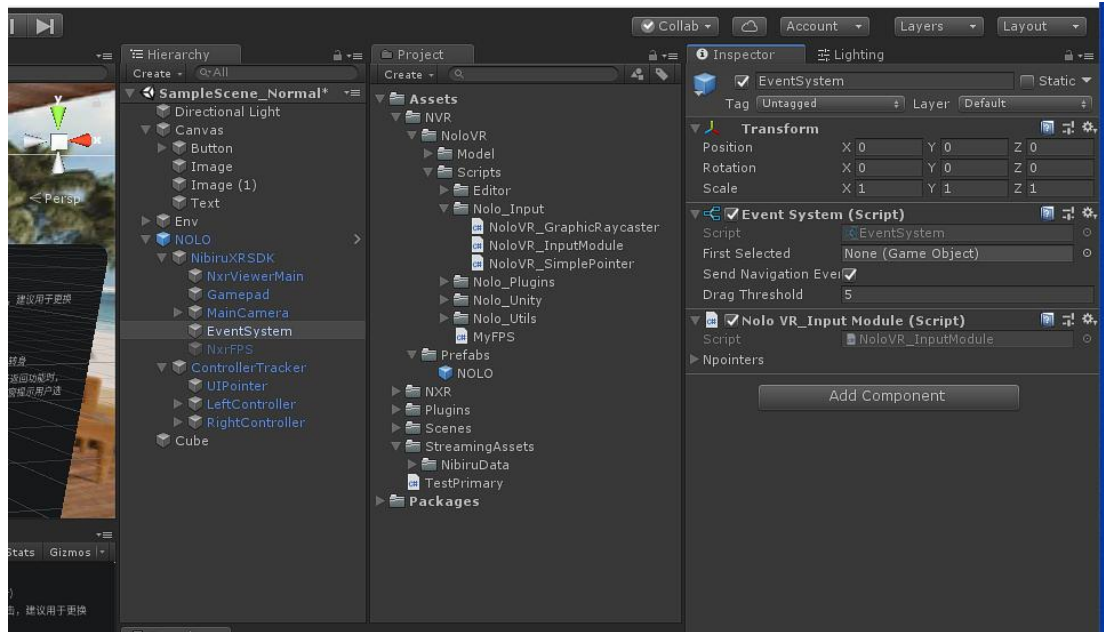
4.射线检测

NOLO 提供一套可以 UGUI 射线触发方案：

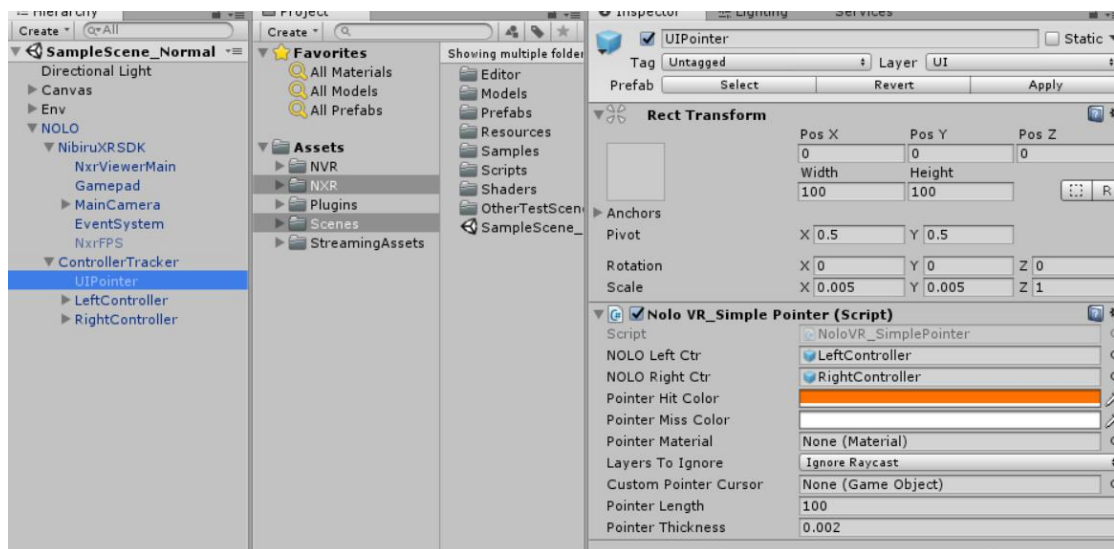
1) 在 UGUI 的 Canvas 上面添加 NoloVR_GraphicRaycaster.cs 脚本



2) 在 UGUI 上的 EventSystem 上面添加 NoloVR_InputModule.cs 脚本
与 3D 交互示例详见场景 SampleScene_Normal 场景



3) 创建一个 GameObject 并添加 NoloVR_SimplePointer.cs 脚本，通过扳机按钮进行切换手柄射线



5.NOLO 提供的 Demo 场景说明

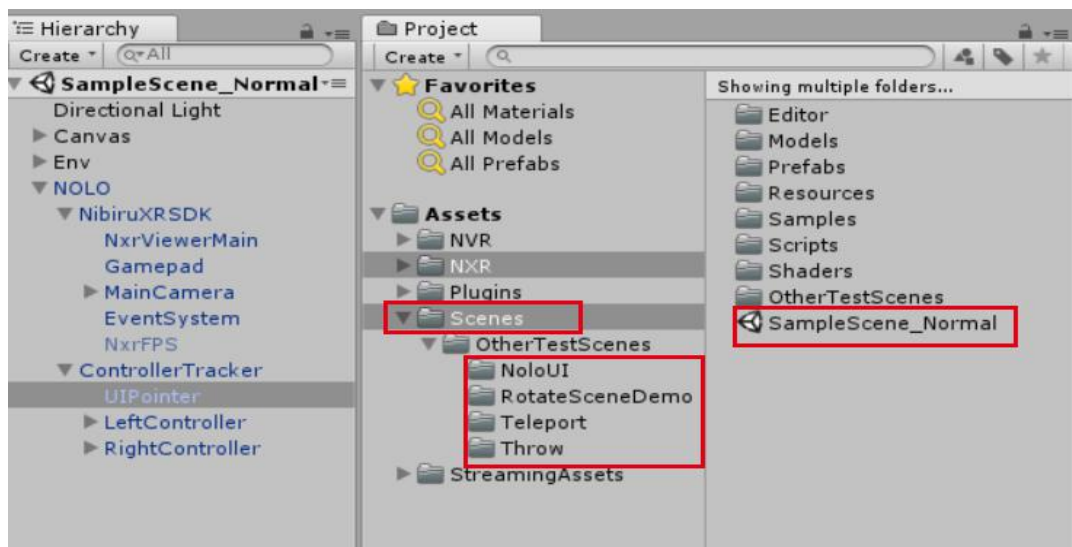
1) NOLO Unity SDK 提供了不同的 Demo Scene 以供了解 NOLO SDK 的使用

SampleScene_Normal Scene: 此场景用来描述如何使用 NOLO 手柄触发 UI, 显示 NOLO 设备的定位信息, 按键信息和按键功能介绍

Throw Scene: 此场景展示了如何使用 NOLO 手柄进行仍物体

TelePort Scene: 此场景展示了如何使用 NOLO 手柄进行瞬移

RotateSceneDemo: 此场景展示了如何使用 NOLO 手柄进行旋转和缩放整个游戏场景

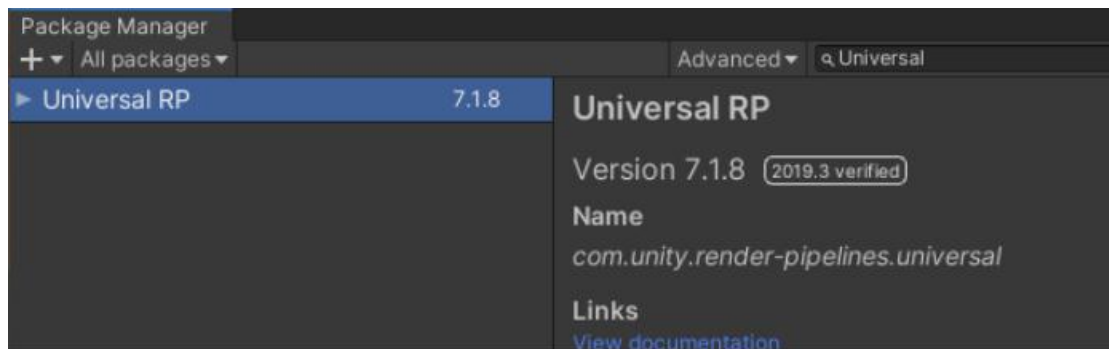


6.Universal RP(通用渲染管线)配置说明

Unity 版本: Unity2019.3.6

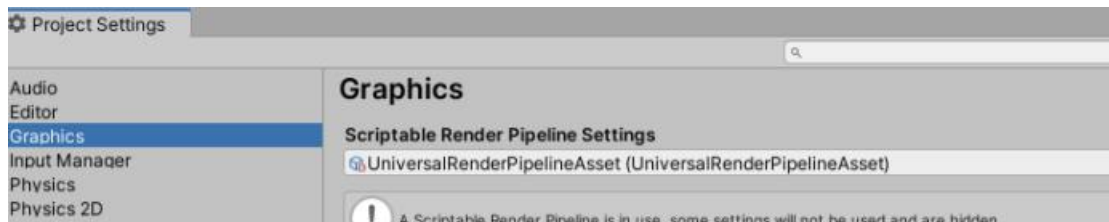
URP 版本: V7.1.8

1) 打开 Window/Package Manager, 搜索 Universal RP, 点击 Install。

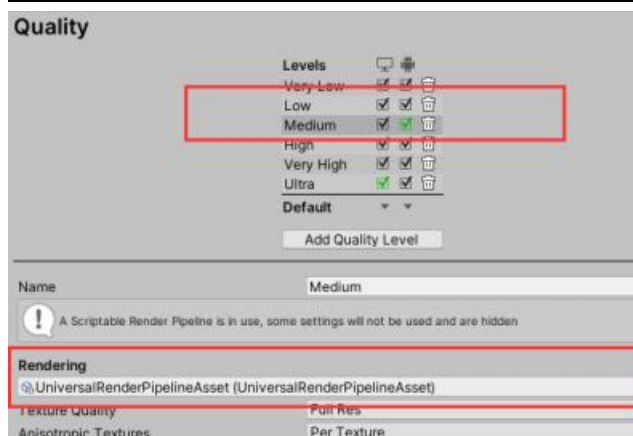


2) 安装完成, 创建 UniversalRenderPipelineAsset,
Assets/Create/Rendering/Universal Render Pipeline/Pipeline
Asset

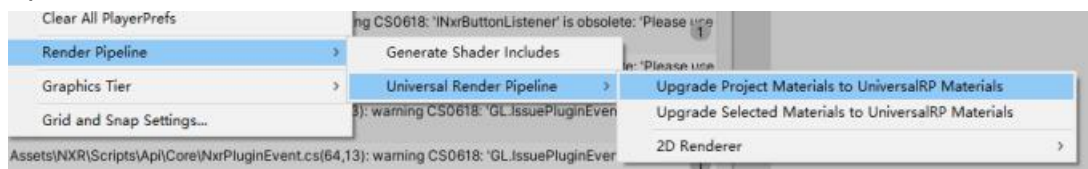
3) 打开 Edit/Project Settings/Graphics/, 在 Scriptable Render
Pipeline Settings 中选择已创建的 PipelineAsset。



4) 打开 Edit/Project Settings/Quality 界面, 选择 Android 类别, 在
Rendering 中选择已创建的 PipelineAsset。



5) 升级工程材质



注：取消 UniversalRenderPipelineAsset 中的 SPR Batcher/Dynamic Batching

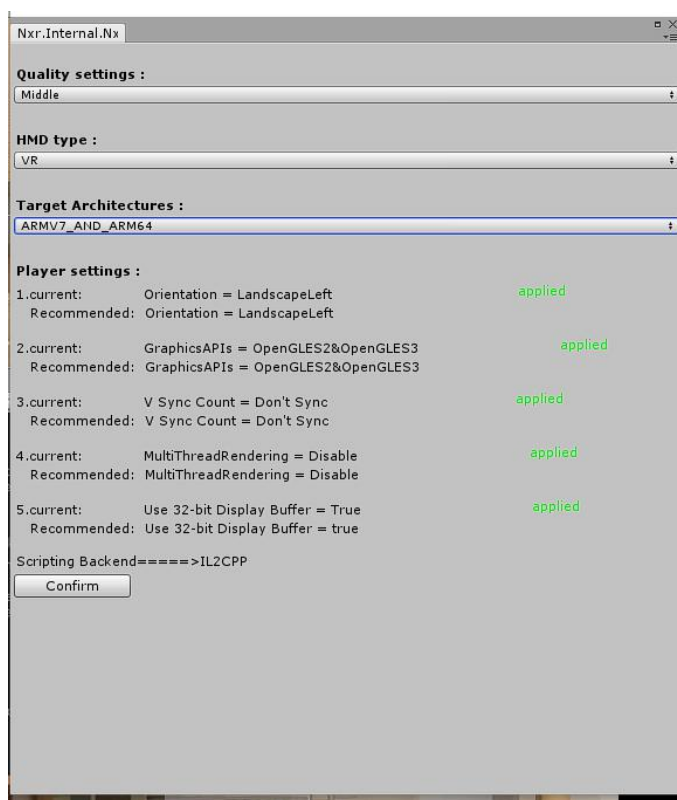
否则会出现内存泄露问题。



7.NOLO SDK 支持 Armv8

1) 打包设置点击菜单 NibiruXR-->XR Settings, 弹出设置面板, 只要按下图

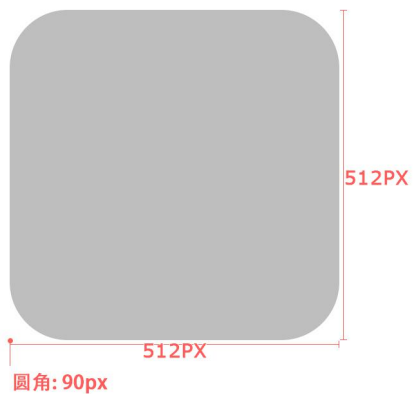
操作即可



8 打包规范

游戏 icon 512 x 512, png 格式, 四周圆角 90px, 小于 200kb, icon 添加

“NOLO_HOME” 角标, 参见如下模板



角标

9.联系方式

有任何问题或者建议欢迎联系我们：developer@nolovr.com