

# **Nolo SDK for Unity 说明文档**

北京凌宇智控科技有限公司

2017 年 5 月

# 目录

一、简介.....	1
二、开发环境.....	1
三、版本说明.....	1
四、SDK 使用说明.....	1
4.1 导入.....	1
4.2 目录介绍.....	2
4.2.1 Example 文件夹.....	2
4.2.2 Icon 文件夹.....	2
4.2.3 Model 文件夹.....	2
4.2.4 Prefabs 文件夹.....	2
4.2.5 Scripts 文件夹.....	2
4.3 使用用例.....	2
4.4 Example 参考设计.....	3
4.4.1 General.....	3
4.4.2 InputTest.....	3
4.4.3 RotateSceneDemo.....	3
4.4.4 Teleport.....	4
五、SDK 功能模块.....	4
5.1 NoloVR_Manager.cs.....	4
5.2 NoloVR_TrackedDevice.cs.....	4
5.3 NoloVR_Controller.cs.....	5
5.4 NoloVR_PlayArea.cs.....	6
六、 安卓配置.....	6

## 一、简介

Nolo SDK for Unity 是凌宇智控针对其 NOLO CV1 产品为 Unity 开发者提供的开发包。使用本开发包可获取 NOLO 设备中头盔定位器和双手柄的定位数据，双手柄的姿态数据以及双手柄所有的按键信息，并可对双手柄进行振动控制，该 SDK 主要适用于安卓平台。

## 二、开发环境

Nolo SDK for Unity 开发环境为 Unity5.4.1(Win 64bit)，建议 Unity 版本为 5.4.1 及以上，JDK 版本为 jdk1.8.0\_101。

## 三、版本说明

版本号	版本内容
NoloVR_SDK_1.1.7	1.取消 NoloVR_Manager->automaticConnection 选项， 与 NOLO 设备的连接采用自动的方式。 2.新增速度和角速度，可通过 GetPose().vecVelocity 和 GetPose().vecAngularVelocity 获取使用。

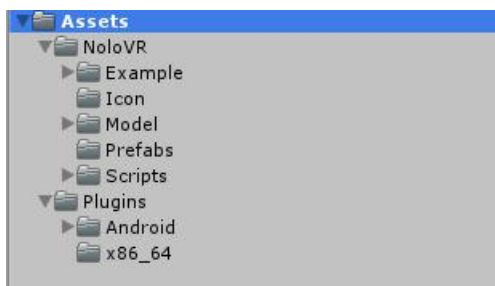
## 四、SDK 使用说明

### 4.1 导入

Nolo SDK for Unity 采用 unitypackage 形式，可通过 Assets->Import Package->Custom Package 的方式导入 unity 中进行开发。

## 4.2 目录介绍

Nolo SDK for Unity 的目录结构如下：



### 4.2.1 Example 文件夹

Example 文件夹包含 Nolo SDK 参考设计，将在 4.4 节中详细说明。

### 4.2.2 Icon 文件夹

Icon 文件夹包含 NOLO 的 Icon 素材，开发者可在其开发的 APP 的 icon 右上角添加 NOLO 图标，表示游戏支持 NOLO 设备。

### 4.2.3 Model 文件夹

Model 文件夹包含 NOLO 手柄模型材质。

### 4.2.4 Prefabs 文件夹

Prefabs 文件夹包含 NoloManager.prefab 预设体，使用其可以进行快速开发。

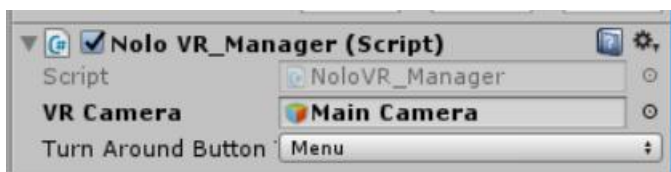
### 4.2.5 Scripts 文件夹

Scripts 文件夹为 NoloSDK 脚本文件夹。

## 4.3 使用用例

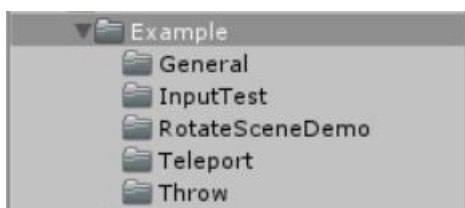
新建一个场景，将 Prefabs 文件夹下的 NoloManager.prefab 拖入场景中；将使用的 VR Camera，拖拽到 NoloManager->Hmd(camera)下，变成 Hmd(camera)的子物体，并将 position 和 rotation 都清零；然后在 NoloManager 上找到挂载的 NoloVR\_Manager.cs 脚本，将游戏运行时姿态真正发生变化的 Camera 物体拖至 VR Camera 处，即可完成设置。

设置完成后的界面如下图所示：



## 4.4 Example 参考设计

Example 参考设计的目录结构如下：



### 4.4.1 General

General 用于查看 NOLO 基本数据信息。

在 Test 场景中，UI\_Test.cs 脚本用于在 UI 界面上显示 NOLO 设备提供给 Unity 的数据信息，方便开发过程中调试。

### 4.4.2 InputTest

InputTest 用于测试 NOLO 按键功能。

在 InputTest 场景中，Input\_Test.cs 脚本用于测试 NOLO 设备两个手柄的按键状态，方便开发过程中调试。

### 4.4.3 RotateSceneDemo

RotateSceneDemo 实现的功能是同时按住 NOLO 两个手柄的 Grip 键（抓握键），可以对场景进行旋转、缩放以及移动。

在 NoloManager 上添加 NoloVR\_RotateScene.cs 脚本，将场景中需要改变的所有物

体放置在同一个父节点之下，将这个父节点添加到 NoloVR\_RotateScene.cs 脚本中的 Object Parents 位置，“Is change scale”表示是否需要修改缩放比例，“Is change rotation”表示是否修改旋转姿态。

#### 4.4.4 Teleport

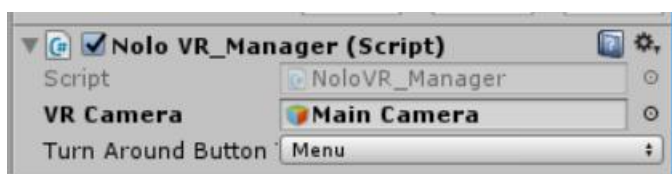
Teleport 用于实现传送功能。

在任意手柄 (Leftcontroller 或者 Rightcontroller) 上添加 NoloVR\_Teleport.cs 脚本即可。

## 五、SDK 功能模块

### 5.1 NoloVR\_Manager.cs

NoloVR\_Manager.cs 模块如下图所示：

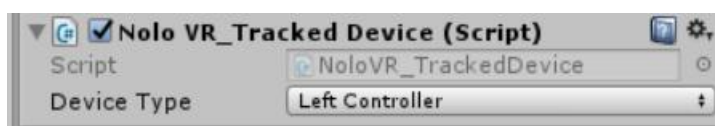


**VR Camera:** 获取场景中的 VR 摄像机，开发者可以根据自己使用的第三方 SDK，将对应的 VR Camera 赋值给它，但需要注意的是，这个 VR Camera 必须是拥有姿态数据的游戏物体，而不是一个简单的摄像机。

**Turn Around Button Type:** 一键转身热键设置，可以通过此设置来选择不同的转身按键或者关闭此功能。

### 5.2 NoloVR\_TrackedDevice.cs

NoloVR\_TrackedDevice.cs 模块如下图所示：



**Device Type:** 表示设备类型，分别是：Hmd(头盔定位器)，Left Controller(左手柄)，

Right Controller(右手柄) 和 Base station(基站)。

### 5.3 NoloVR\_Controller.cs

NoloVR\_Controller.cs 模块包括两个接口函数：

NoloVR\_Controller.GetDevice (NoloDeviceType deviceIndex);

NoloVR\_Controller.GetDevice (NoloVR\_TrackedDevice trackedobject);

用来获取想监听的 NOLO 组件的所有信息。

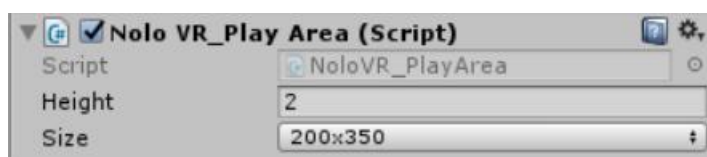
具体函数使用方法如下：

函数名	参数	返回值	说明
GetPose	无	Nolo_Transform	返回 NOLO 定位设备的位置和姿态
GetNoloButtonPressed	UInt buttonMask NoloButtonID button	bool	buttonMask: 1<<0 touchpad 1<<1 trigger 1<<2 menu 1<<3 system 1<<4 grip NoloButtonID: Touchpad, trigger, menu, system, grip
GetNoloButtonDown	UInt buttonMask NoloButtonID button	bool	buttonMask: 1<<0 touchpad 1<<1 trigger 1<<2 menu 1<<3 system 1<<4 grip NoloButtonID: Touchpad, trigger, menu, system, grip
GetNoloButtonUp	UInt buttonMask NoloButtonID button	bool	buttonMask: 1<<0 touchpad 1<<1 trigger 1<<2 menu 1<<3 system 1<<4 grip NoloButtonID: Touchpad, trigger, menu, system, grip
GetNoloTouchPressed	UInt touchMask NoloTouchID touch	bool	touchMask: 1<<0 touchpad

			NoloTouchID: Touchpad
GetNoloTouchDown	Uint touchMask NoloTouchID touch	bool	touchMask: 1<<0 touchpad NoloTouchID: Touchpad
GetNoloTouchUp	Uint touchMask NoloTouchID touch	bool	touchMask: 1<<0 touchpad NoloTouchID: Touchpad
GetAxis	NoloTouchID(默认为 touchpad, 其他无效)	Vector2	x 取值范围 (-1~1) y 取值范围 (-1~1)
GetTrackingStaus	无	NoloTrackingStatus	NoloTrackingStatus. NotConnect 未连接 NoloTrackingStatus. Normal 连接正常 NoloTrackingStatus. OutOfRange 被遮挡
TriggerHapticPulse	Int intensity 震动强度	无	手柄震动方法, 参数范围 (0~100), 值越大, 震动越剧烈。

## 5.4 NoloVR\_PlayArea.cs

NoloVR\_PlayArea.cs 模块主要用于在 Scene 中提示开发者用户可能的活动范围, 无其他作用。如下图所示:



Height: 高度。

Size: 长\*宽。

## 六、安卓配置

在 AndroidManifest.xml 中添加如下内容:



```
<uses-permission android:name="android.hardware.usb.host" />
<uses-feature android:name="android.hardware.usb.host" android:required="true"/>
```

在 main activity 中添加如下内容：

```
<intent-filter>
    <action android:name="android.intent.action.MAIN" />
    <category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
<intent-filter>
    <action android:name="android.hardware.usb.action.USB_DEVICE_DETACHED"/>
</intent-filter>
<intent-filter>
    <action android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED" />
</intent-filter>
```

具体可参考 SDK 中 AndroidManifest.xml 文件。