**NOLO VR Unity SDK**

**接口说明**

北京凌宇智控科技有限公司

2018年3月

目录

[**一、 简介 3**](#_Toc11686)

[1.关于NOLO 3](#_Toc22796)

[2. 关于NOLO CV1 3](#_Toc28229)

[3.关于NOLO HOME 3](#_Toc7338)

[4.关于NOLO VR Unity SDK 4](#_Toc7869)

[5.关于移动端SDK架构 4](#_Toc10860)

[**二、 接入准备 5**](#_Toc25751)

[**三、 接口说明 6**](#_Toc9630)

[1. Button事件 6](#_Toc15971)

[2. Touch事件 6](#_Toc8506)

[3. 震动事件 7](#_Toc28465)

[4.定位数据 8](#_Toc26633)

[5.提交错误信息 8](#_Toc17083)

[**四、注意事项 8**](#_Toc23205)

[1.标定原点位置 8](#_Toc29652)

[2.设置AppKey 9](#_Toc15027)

[3.配置AndroidManifest.xml 9](#_Toc20287)

[4.修改camera参数 10](#_Toc13522)

[5. 标定方向 10](#_Toc15970)

[6. 一键转身 10](#_Toc18188)

# 简介

## 1.关于NOLO

NOLO 致力于将桌面级 VR 的优秀沉浸式体验与移动 VR 的便捷性之间架起一座桥梁，重新定义下一代移动 VR 的交互方式。

目前，NOLO 可兼容全球将近 8700 万市场存量的各类移动 VR 头盔，市场潜力巨大。不仅如此，NOLO 也已大规模展开与海内外众多 VR 一体机、PC VR、AR、服务机器人、无人机公司的合作。

## 关于NOLO CV1

NOLO CV1 是 NOLO Inc. 自主研发的全球首款全沉浸式 VR/AR 交互产品，包含 1 个定位基站、1 个头盔定位器及 2 个交互手柄。

NOLO CV1 能适配目前市面上主流的移动 VR 头盔、部分PC VR 和 AR 头盔，以及基于NibiruOS的一体机，为其提供空间定位与交互功能。用户首次使用时，仅需简单设置，即可实现在虚拟世界中真实的移动，以及通过交互手柄与虚拟世界中的事物进行多元化的互动。

## 3.关于NOLO HOME

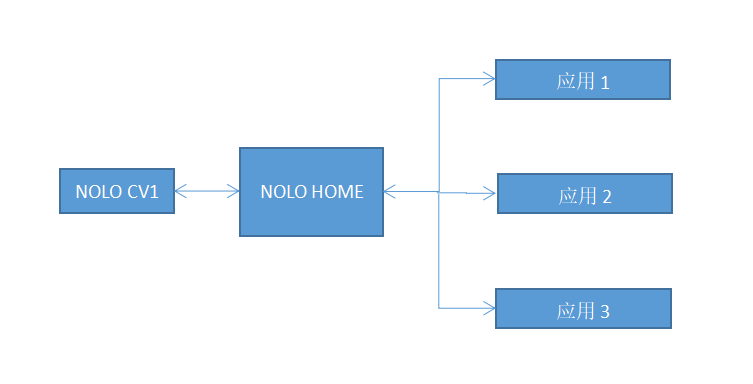
NOLO HOME 是 NOLO 品牌旗下的移动 VR 生态平台，也是全球首个基于手机的 6-DoF 移动 VR 应用平台。NOLO HOME + NOLO CV1 + 手机，用户可以随时随地、无线便捷得享受到全沉浸式 VR 体验，从而为普通消费者创造一种全新的娱乐游戏方式。目前，Android 用户率先可以下载安装，iOS 版将后续开放。

## 4.关于NOLO VR Unity SDK

NOLO VR Unity SDK是由NOLO Inc. 开发，便于Unity开发者获取NOLO设备数据，搭配Cardboard和GearVR 的SDK，开发移动端6-DoF VR游戏。

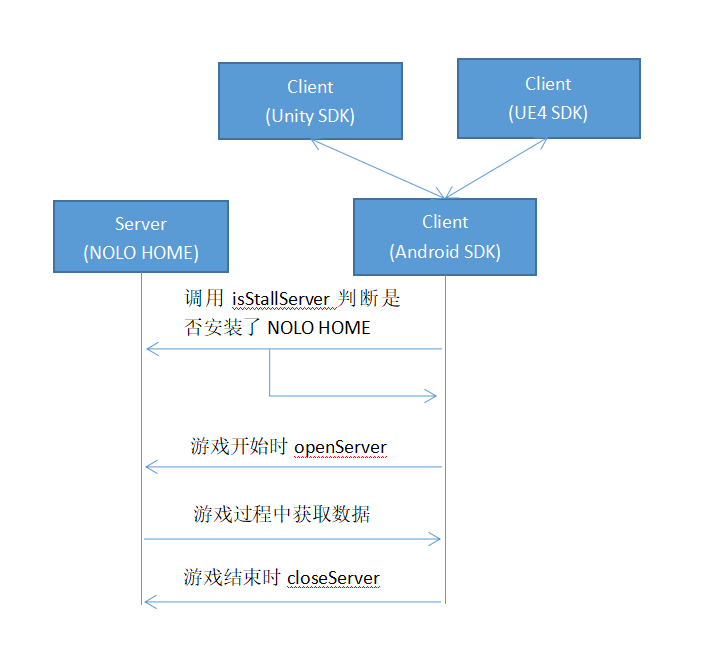
## 5.关于移动端SDK架构

NOLO SDK采用C/S架构设计，NOLO HOME为Server端，负责读取NOLO CV1设备数据，并进行处理。游戏为Client端，SDK通过AIDL将游戏与NOLO HOME建立通信，获取NOLO CV1设备数据。游戏本身不需要去读取USB设备数据，只需要与NOLO HOME建立连接来实现数据收发，如（图一）。



（图一）

NOLO VR Android SDK通过AIDL与NOLO HOME建立进程间通信，来获取设备数据。首先要通过getInstance()来获取NOLOVR的实例，游戏启动前调用isStallServer()判断用户是否安装了NOLO HOME，确认安装之后，调用openServer()绑定Server，在游戏结束后调用closeServer()结束绑定。NOLO VR Unity SDK和NOLO VR UE4 SDK是对NOLO VR Android SDK进行部分接口封装，开发者只需要通过对应接口获取定位和按键数据进行游戏开发即可，如（图二）。



（图二）

# 接入准备

开发者需要准备Unity5.6以上版本，NOLO VR Unity SDK。Windows调试需要下载NOLO Driver For Windows，手机上运行需要先安装NOLO HOME。

# 接口说明

## 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() |

## 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() |

## 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0~100 |
| **返回值** | void |
| **先决条件** | NoloVR\_Controller.GetDevice() |

## 4.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() |

## 5.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() |

# 四、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“NoloManager”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，AppKey是在NOLO开发者平台创建应用时自动生成的，请在游戏中添加NoloVR\_AppInfo脚本，并正确填写AppKey。

## 3.配置AndroidManifest.xml

请在AndroidManifest.xml中配置如下权限：

<uses-permission android:name="android.permission.BROADCAST\_STICKY" />

<uses-permission android:name="nolo.permission.ACCESS\_SERVER" />

<uses-permission android:name="android.permission.PACKAGE\_USAGE\_STATS" />

<uses-permission android:name="android.permission.SYSTEM\_ALERT\_WINDOW" />

<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE" />

<uses-permission android:name="android.permission.INTERNET" />

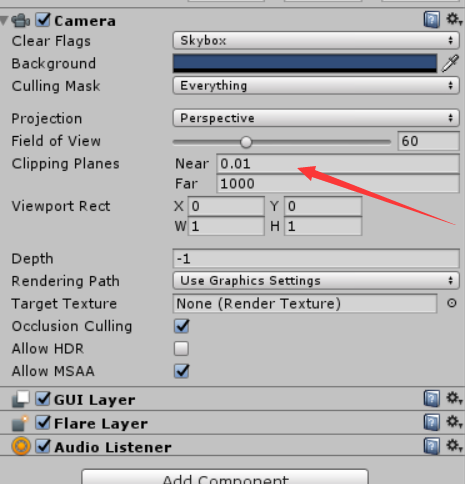
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

<uses-permission android:name="com.android.launcher.permission.WRITE\_SETTINGS" />

<uses-permission android:name="android.permission.WRITE\_APN\_SETTINGS" />

## 修改camera参数

如（图三）。



（图三）

## 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。

## 一键转身

修改下图值，可以将NOLO CV1某个按键设置为转身热键，可以在双击这个按键的情况下，实现原地旋转180度的操作，将游戏中背后的场景，转到身前，如（图四）。



（图四）