**NOLO VR Android 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](#_Toc32275)**

[1.关于NOLO 3](#_Toc17820)

[2. 关于NOLO CV1 3](#_Toc26288)

[3.关于NOLO HOME 3](#_Toc14142)

[4.关于NOLO VR Android SDK 4](#_Toc2338)

[5.关于移动端SDK架构 4](#_Toc27643)

**[二、 接入准备 6](#_Toc4182)**

[1. 库文件及其导入 6](#_Toc31038)

[2. 权限注册 6](#_Toc18058)

**[三、 接口说明 6](#_Toc26830)**

[1. 基本信息 6](#_Toc15611)

[2. 定位信息 8](#_Toc17209)

**[四、 注意事项 10](#_Toc19744)**

[1.标定原点位置 10](#_Toc10934)

[2.设置AppKey 11](#_Toc27786)

[3. 获取 getInstance NoloVR 实例 11](#_Toc29175)

# 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

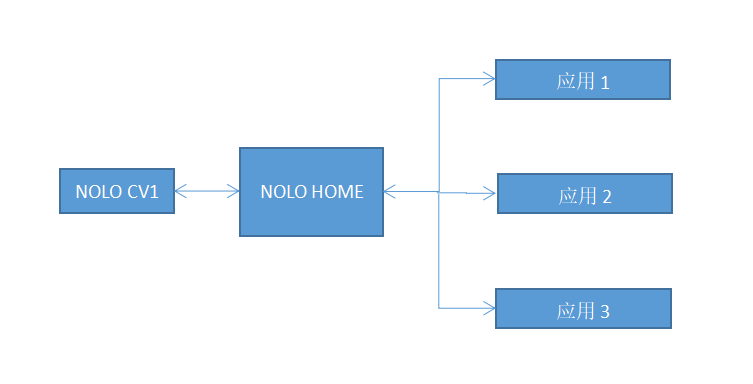
NOLO HOME 是 NOLO 品牌旗下的移动 VR 生态平台，也是全球首个基于手机的 6-DoF 移动 VR 应用平台。NOLO HOME + NOLO CV1 + 手机，用户可以随时随地、无线便捷得享受到全沉浸式 VR 体验，从而为普通消费者创造一种全新的娱乐游戏方式。目前，Android 用户率先可以下载安装，iOS 版将后续开放。

## 4.关于NOLO VR Android SDK

NOLO VR Android SDK是由NOLO Inc. 开发，便于Android 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。

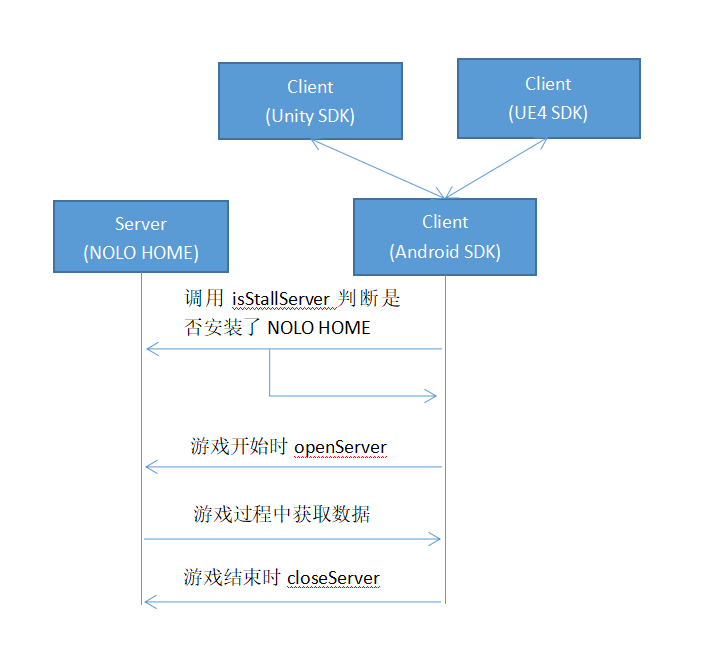
## 5.关于移动端SDK架构

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，如（图二）。



（图二）

# 接入准备

## 1. 库文件及其导入

Android直接导入 androidsdkclient-release.aar。

2. 权限注册

Android AndroidManifest.xml 注册如下权限：

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

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

<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />

<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" />

# 接口说明

## 基本信息

|  |  |
| --- | --- |
| **函数名称** | public static NoloVR getInstance(Context context)； |
| **功能描述** | 获取NoloVR |
| **参数** | context:必须传递 Activity对象，否则相关提示框无法正常显示。 |
| **返回值** | NoloVR对象 |
| **先决条件** |  |

|  |  |
| --- | --- |
| **函数名称** | public boolean isStallServer()； |
| **功能描述** | 判断NOLO HOME Server是否安装，所有数据基于安装Server。 |
| **参数** |  |
| **返回值** | bool |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | public void setAppKey(String appKey)； |
| **功能描述** | 设置appkey |
| **参数** | String 类型appKey，开发者平台获取该参数。 |
| **返回值** |  |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | public void openServer()； |
| **功能描述** | 绑定 NOLO HOME Server，Server 绑定成功后即可远程调用Server。 |
| **参数** |  |
| **返回值** |  |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | public void closeServer()； |
| **功能描述** | 解绑Server，游戏退出的时候解绑Server。 |
| **参数** |  |
| **返回值** | bool |
| **先决条件** | NoloVR getInstance(Context context) |

## 定位信息

|  |  |
| --- | --- |
| **函数名称** | NTrackedDevicePose getPoseByDeviceType(int type); |
| **功能描述** | 根据设备类型信息获取位置 |
| **参数** | 0：头盔定位器  1：左手柄  2：右手柄  3：基站 |
| **返回值** | NTrackedDevicePose |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | int getElectricityByDeviceType(int type) ; |
| **功能描述** | 根据设备类型获取设备电量信息 |
| **参数** | 0：头盔定位器  1：左手柄  2：右手柄  3：基站 |
| **返回值** | int |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | NControllerState getControllerStatesByDeviceType(int type); |
| **功能描述** | 按键信息 |
| **参数** | 0：头盔定位器  1：左手柄  2：右手柄  3：基站 |
| **返回值** | NControllerState |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | void SetHmdTrackingCenter(float x,float y,float z); |
| **功能描述** | 设置头盔旋转中心 （废弃） |
| **参数** |  |
| **返回值** |  |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | void triggerHapticPulse(int deviceType,int intensity); |
| **功能描述** | 震动 |
| **参数** | deviceType：  0：头盔定位器  1：左手柄  2：右手柄  3：基站  Intensity：0~100 |
| **返回值** |  |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | void setHmdType(int hmdType); |
| **功能描述** | 设置头盔类型 |
| **参数** | 0 :Gear VR  1:DayDream  2:CardBoard  3其他 |
| **返回值** |  |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | void reportError(String errorLog); |
| **功能描述** | 收集游戏的错误日志信息 |
| **参数** | String errorLog：游戏收集到的错误日志信息 |
| **返回值** |  |
| **先决条件** | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| **函数名称** | int add(int i,int j) |
| **功能描述** | 测试远程服务是否运行 |
| **参数** | 两个int类型 |
| **返回值** | int |
| **先决条件** | NoloVR getInstance(Context context) |

# 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。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。

游戏未上传NOLO HOME时，可以使用此公用Appkey进行开发测试。

公用Appkey：4e4f4c4f484f4d457eff82725bc694a5。

### 获取 getInstance NoloVR 实例

在调用绑定Server之前，请先调用 isStallServer来判断是否安装了NOLO HOME，在游戏开始的时候，调用openServer绑定Server，在游戏结束的时候，调用closeServer结束绑定。