
2015年12月13日

基于RMCP协议的远程鼠标遥控程序

13307130228

周凯文

RMCP协议

目的

RMCP(RemoteMouseControlProtocol)协议是一个应用层协议，基于TCPSocket实现，默认使用端口号12345，支持客户端(Android端)对服务端(PC端)进行鼠标的远程控制，支持的操作方式有触控板操作、重力感应操作以及线性加速度操作。

总体操作

客户端与服务端建立TCP连接，向服务端发送RMCP消息，服务端接收到RMCP消息，根据消息内容控制服务端鼠标的位置、状态。

协议规范

在RMCP消息头部需包含以下参数，均使用空格隔开：

- RMCP协议的名称及其版本号
 - 选择的操作方式代号（TouchPad/Gravity/LinearAcceleration）
 - 两个控制参数域（鼠标位移参数或者鼠标滚轮参数）
 - 五个触发操作域（鼠标左键、右键、中键的触发标志，包括触发鼠标滚轮滚动的触发标志）
-

客户端(ANDROID端)

兼容环境

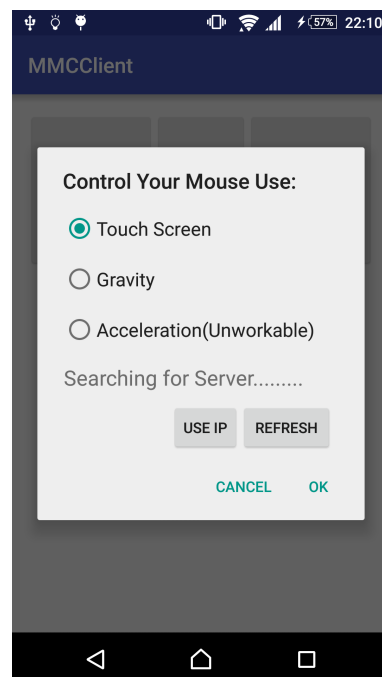
Android 4.0.3 及以上，配有重力传感器及线性加速度传感器。

界面

选择操控服务端鼠标的方式：

- Touch Screen：使用触控板的方式控制鼠标
- Gravity：使用重力感应的方式控制鼠标
- Acceleration：使用线性加速度控制鼠标（难以进行光标控制，但具体保留原因见后文）

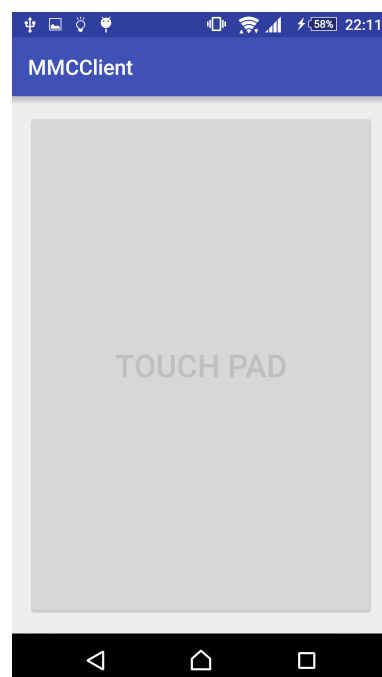
应用支持通过UDP广播查找同一网段下的服务端，单击Refresh按钮可以重新发送UDP广播，成功连接会显示“Connected to /*服务端IP地址*/”。但在iFudan.1x等网络下需使用USE IP输入服务端显示的IP地址进行连接。



Touch screen 界面

在TouchPad上支持的操作有：

- 单指在屏幕上移动：光标移动
- 单指点击屏幕：鼠标左键单击
- 单指双击屏幕：鼠标左键双击
- 双指点击屏幕：鼠标右键单击
- 双指在屏幕上移动：鼠标滚轮上下移动

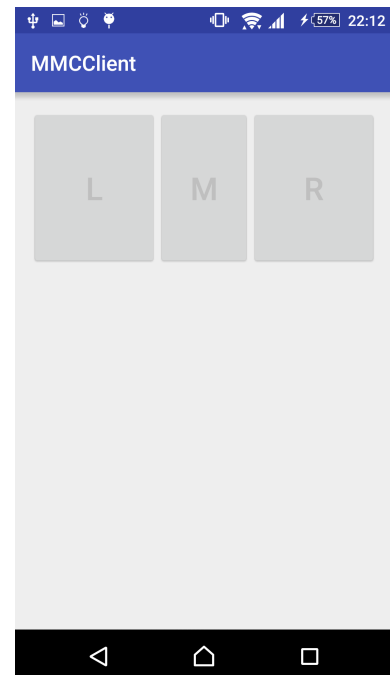


2015年12月13日

Gravity及Acceleration界面

在Gravity及Acceleration界面上的操作方式：

- 通过Gravity或者LinearAcceleration传感器进行鼠标控制
- L, M, R三个键分别代表鼠标左键，鼠标中键以及鼠标右键
- 在鼠标中键上滑动可以模拟鼠标滚轮操作



保留Acceleration操作方式的含义

这个PJ最初的目的是想用手机的加速度传感器来控制光标移动（使用体验就像一般鼠标那样），但是由于手机传感器精度问题，还有作者个人能力问题无法实现，为了纪念作者在这个上面浪费的时间，故特此保留这个功能。

服务端(PC端)

兼容环境

装有Java(TM) SE Runtime Environment (build 1.8.0_20-b26)及以上版本的Windows或者Mac OS X系统

（由于此应用对网络要求较高，建议使用手机开WIFI热点，PC端连接此WIFI热点以获得最佳体验）

2015年12月13日

其他说明

开发环境

操作系统：Mac OS X 10.11.1

IDE: Android Studio 1.5.1, IntelliJ IDEA 15.0.1

软件测试

Android端在Sony Xperia Z2(Android 5.1.1 API22)上测试通过。

PC端在Windows 10及Mac OS X 10.11.1上均测试通过。
