# IMU Mouse的滑动算法方案

总体思路：

1）屏幕上鼠标的移动

屏幕二维像素坐标系：s，s原点在屏幕左上角，s\_x向右，s\_y向下。鼠标在屏幕上的坐标为MS（MS\_x,MS\_y）。点MS在屏幕上的移动速度为VMS（VMS\_x,VMS\_y）

2）需要确定一个二维平面坐标系，假设指环在这个平面内滑动，将加计和陀螺的数据投影到这个二维坐标系中。确定一个滑动的起点，从这个起点开始积分计算速度和位置，再确定终点。

输出：

技术难点：

1. 这个二维平面坐标系的选取。
2. 起点的确定。
3. 终点的确定。
4. 积分计算速度和位置漂移的减小

## 计算过程

### （1）初始化

以初始时刻的本体系作为滑动参考坐标系s。设初始时刻的的姿态为，则。



### （2）实时解算

得到加速度计数据

#### （2．1）去除中的重力分量



#### （2．2）将投影到s系





#### （2．3）对进行积分

## 近似处理

1. 滑动二维平面坐标系的选取（slip\_frame）（简称s系）

**Neorun坐标系**



理论上slip\_frame是IMU滑动所在的平面，x向右，y向前，z向上，理论上y是向人的正前方。（**近似**）通常这个平面比较接近水平面，因此最简单的办法是用水平面替代这个平面。（slip\_frame = water\_frame）。（**近似**）以起点处IMU的y方向作为slip\_frame的y方向。

可能的优化方法：slip\_frame的特点是z方向的加速度为0，可在滑动一段时间（如2S后）拟合出更精确的（Z方向加速度最小的）slip\_frame。