**题目：将第二讲的仿真数据集（视觉特征，imu数据）接入我们的VINS代码，并运行出轨迹结果。**

**仿真数据集无噪声**

**仿真数据集有噪声（不同噪声设定时，需要配置vins中 imu noise大小。）**

第一步：为了将第二章的仿真数据接入VINS，仿照run\_enroc.cpp写simulation-test.cpp。

新建simulation-test.cpp代码，具体代码注释见源代码，在run\_enroc.cpp的基础上做了改动，主要改动是PubImuData()和PubImageData()。读取仿真数据需要注意仿真数据的格式。

第二步：将第二章的仿真数据放到当前文件夹下，便于simulation-test.cpp找到，主要是要读取imu\_pose.txt imu\_pose\_noise.txt cam\_pose.txt，还要利用cam\_pose\_tum,txt和结果作对比

第三步：在System.cpp中添加PubSimImageData()函数。(具体见代码)

第四步：config文件夹下euroc\_config.yaml参数修改，改为与param.h中参数一样。

第五步：使用evo工具评测SLAM算法性能并可视化结果

simulation-test生成的pose\_output.txt是TUM数据集格式，每行有8个元素：

timestamp tx ty tz qx qy qz qw

结尾没有空格，时间以秒为单位

**注意：**

1、第二章生成仿真数据要修改噪声设置，同时要修改第七章config文件夹下的噪声设置

2、第二章每次生成的仿真数据vio\_data\_simulation整个复制到第七章src文件夹下

3、第二章camera\_pose\_tum.txt数据文件存到第七章bin文件夹下

4、每次运行simulation-test要先删除上一次的pose\_output，否则数据会追加在后面

5、生成的pose\_output文件数据格式不是严格的TUM，利用bin下的python文件修改空格问题，生成pose\_output\_change.txt，每次也要删除上一次的文件，否则也会追加

6、simulation-test.cpp读取文件路径不要搞错

第二章仿真数据说明：

运动轨迹是一个xy平面椭圆运动，z正弦运动

house\_model中保存了原始设置的3d房屋关键点（一行两个点连成线，保存点前判断一下是否之前获取过）

运行后生成如下数据：

keyframe文件夹：每一帧相机观测到的特征点，代码中设定相机帧率为30Hz，仿真20s，对应一共产生600帧观测

all\_points:保存了所有特征点（对比房屋关键点可以有所增加）文件存储的是house模型的线特征，每行4个数，对应该线两端点在归一化平面的坐标，一共23行对应23条线段

cam\_pose:所有相机位姿， cam\_pose\_tum只是存储顺序和变量个数有所不同

imu\_pose:所有imu位姿， imu\_pose\_noise 带噪声

imu\_int\_pose:积分后得到的imu轨迹， imu\_int\_pose\_noise 带噪声

仿真数据集无噪声运行结果：（接入imu\_pose.txt）

许多鸟在飞

低可信度描述已自动生成

图形用户界面

描述已自动生成

图表, 雷达图

描述已自动生成

小噪声：(接入imu\_pose\_noise.txt)

在利用第二章代码生成imu\_pose\_noise.txt时的配置：（param.h）

文本

描述已自动生成

图片包含 图表

描述已自动生成

图形用户界面, 应用程序

描述已自动生成

图表

描述已自动生成

大噪声：(接入imu\_pose\_noise.txt)

噪声配置(param.h)

图形用户界面, 文本

描述已自动生成

地图上的鸟

低可信度描述已自动生成

图形用户界面

描述已自动生成

图表, 雷达图

描述已自动生成

分析：

每种情况运行结果和生成的数据集有很大关系，大噪声有时候偏的离谱。

可以看到设置的imu噪声越大，通过VINS估计出来的轨迹结果越差。