

# 作业整体说明

## 修改点：

完成本作业的目的是加深对算法框架的理解和提高算法应用能力。搜索 CF 可以查到所有在原框架上修改的点。

## 外参不准确说明：

UrbanNav 数据集 1 给定的外参  $\text{body\_T\_LiDAR}$ （即  $\text{imu\_to\_lidar}$ ）不准确（旋转角度接近  $1.73\text{rad}$ ，而不是给定的接近  $1.57\text{rad}$ ），这导致 lidar 位姿先后通过  $\text{imu\_to\_lidar}$  的逆和组合导航位姿转换到 map 坐标系下的结果与实际不符，考虑到建图时的后端优化是在 map 坐标系下完成的，该外参对仿真结果必然存在影响。本次作业的主要目的是在新的数据集下跑通原来的代码框架，所以忽略这个问题。

## 工作量：

本作业只用新的数据集 1（没有用数据集 2 和 3）跑通原来的算法，没有分析结果，没有对比不同的仿真参数的仿真效果。主要出于如下三点原因：

1) 大作业中的特殊说明里提到“在融合的时候使用组合导航的位置代替 RTK 作为先验观测，这会使融合比预期的好，并且会使精度分析环节得到的结论略有偏差”，在这种情况下精度分析的标准比较模糊。

2) UrbanNavDataset 数据集的说明材料比较少，有一些关键信息没有明确说明，还有待完善。

3) 自己学艺不精，进步空间还很大。

# 及格

## 2.1.1 测试基于“激光里程计+RTK 位置”的建图(第 4 讲)

对应代码文件：10-mapping-and-matching

## 程序修改要点说明：

1) 数据订阅说明

主要对程序 `data_pretreat_flow.cpp` 进行修改

原 rosbag 的信息如下图：

```
topics:      /kitti/oxts/gps/fix          4544 msgs      : sensor_msgs/NavSatFix
             /kitti/oxts/gps/fix/extract 45826 msgs    : sensor_msgs/NavSatFix
             /kitti/oxts/gps/vel         4544 msgs      : geometry_msgs/TwistStamped
             /kitti/oxts/gps/vel/extract 45826 msgs    : geometry_msgs/TwistStamped
             /kitti/oxts/imu             4544 msgs      : sensor_msgs/Imu
             /kitti/oxts/imu/extract     45826 msgs    : sensor_msgs/Imu
             /kitti/velo/pointcloud      4544 msgs      : sensor_msgs/PointCloud2
             /tf                         45826 msgs    : tf2_msgs/TFMessage
             /tf_static                  45826 msgs    : tf2_msgs/TFMessage
root@fa0d0edad550:/workspace/data/kitti#
```

新的 rosbag 的信息如下图：

```
topics:      /camera/image_color      4874 msgs      : sensor_msgs/Image
             /imu/data                 48754 msgs     : sensor_msgs/Imu
             /navsat/fix                488 msgs      : sensor_msgs/NavSatFix
             /navsat/odom                487 msgs      : nav_msgs/Odometry
             /navsat/origin              1 msg         : geometry_msgs/Pose
             /novatel_data/bestpos       488 msgs      : novatel_msgs/BESTPOS
             /novatel_data/corrimudata   488 msgs      : novatel_msgs/CORRIMUDATA
             /novatel_data/inscov        487 msgs      : novatel_msgs/INSCOV
             /novatel_data/inspvax       487 msgs      : novatel_msgs/INSPVAX
             /ublox_node/aidalm          488 msgs      : ublox_msgs/AidALM
             /ublox_node/aideph          488 msgs      : ublox_msgs/AidEPH
             /ublox_node/fix             488 msgs      : sensor_msgs/NavSatFix
             /ublox_node/monhw           488 msgs      : ublox_msgs/MonHW
             /ublox_node/navclock        488 msgs      : ublox_msgs/NavCLOCK
             /ublox_node/navpvt          488 msgs      : ublox_msgs/NavPVT
             /ublox_node/navsat           24 msgs      : ublox_msgs/NavSAT
             /ublox_node/navstatus       488 msgs      : ublox_msgs/NavSTATUS
             /ublox_node/rxmraw          1950 msgs     : ublox_msgs/RxmRAWX
             /velodyne_points            4871 msgs     : sensor_msgs/PointCloud2
root@fa0d0edad550:/workspace/data/UrbanNav-HK-Data20190428#
```

点云数据的 topic 由/kitti/velo/pointcloud 替换为/velodyne\_points。

imu 数据的 topic 由/kitti/oxts/imu 替换为/imu/data。

速度数据虽然按照原来方式订阅，但是由于新的 rosbag 中没有相应的 topic，所以不会订阅到，相关的使用速度数据的代码也做了修改。

位置数据的 topic 由/kitti/oxts/gps/fix 替换为/navsat/fix。

lidar 和 imu 之间的位姿变换 topic 按照原来方式订阅，同样得不到有效数据，将对应的解读数据的函数内容替换成新的数据集统的 lidar 和 imu 的外参。

```

102 Eigen::Matrix4f imu_to_lidar;//CF
103
104 imu_to_lidar << 2.67949e-08,   -1.0,    0.0,    0.0,
105                  1.0,    2.67949e-08,   0.0,    0.0,
106                  0.0,    0.0,    1.0,   -0.28,
107                  0.0,    0.0,    0.0,    1.0;//origin in github CF
108
109
110 lidar_to_imu_ = imu_to_lidar.inverse();//CF
111
112 calibration_received = true; //CF

```

## 2) 时间同步策略

原 kitti 数据集在数据预处理 data\_pretreat\_flow.cpp 中使用的点云，gnss 和 imu 数据的频率都是 10Hz。

UrbanNav 数据集用于数据预处理的组合导航数据频率为 1Hz，点云数据为 10Hz，IMU 为 100Hz。

按照对原来代码框架的理解，时间同步的关键参数有两个，一个是 data\_pretreat\_flow.cpp 中函数 ValidData()里判断多个传感器数据是否同步的参数 0.05，这里选用的是同步基准点云数据周期 0.1s 的一半，不需要修改。另一个参数是 gnss\_data.cpp 和 imu\_data.cpp 中数据同步函数 SyncData 中的参数 0.2，这里的物理意义是相应数据周期的 2 倍，所以分别修改为 2 和 0.02，对应于可以容忍的数据丢失个数。

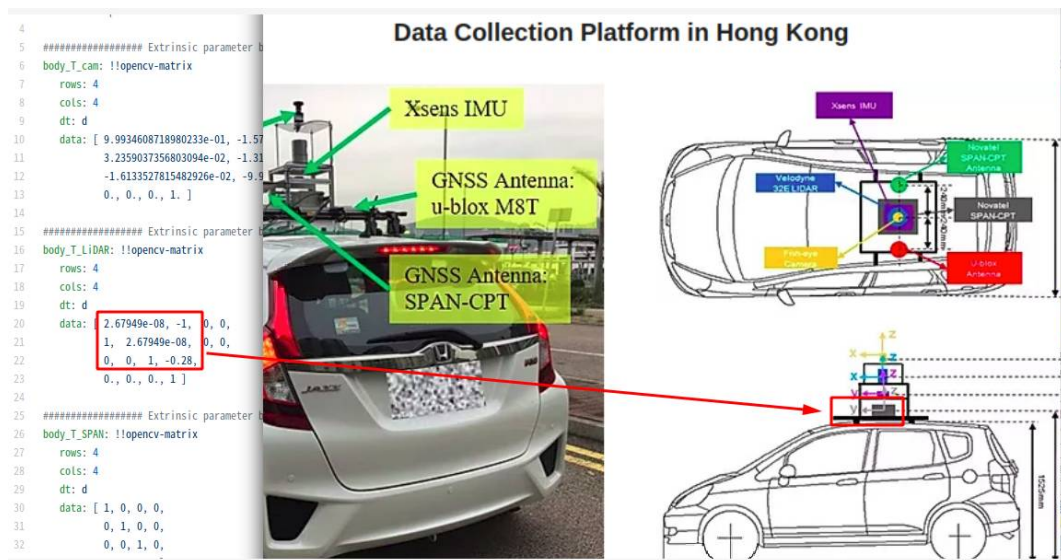
## 3) 安装新的 rosmmsg 类型 novatel msgs

novatel msgs 相关问题 用下面命令安装，注意修改（kinetic）为你电脑对应的 ros 版本，需要先 sudo apt update

```
sudo apt-get install ros-kinetic-novatel-msgs
```

## 4) 雷达方向说明

从仿真结果和外参含义来看，雷达 x 轴的方向和车前进的方向是相反的。所以轨迹箭头和前进方向相反，不过，这并不影响建图等算法，只是一个体现在外参中的一个设定。



这个信息也得到了 weisong 的证实。

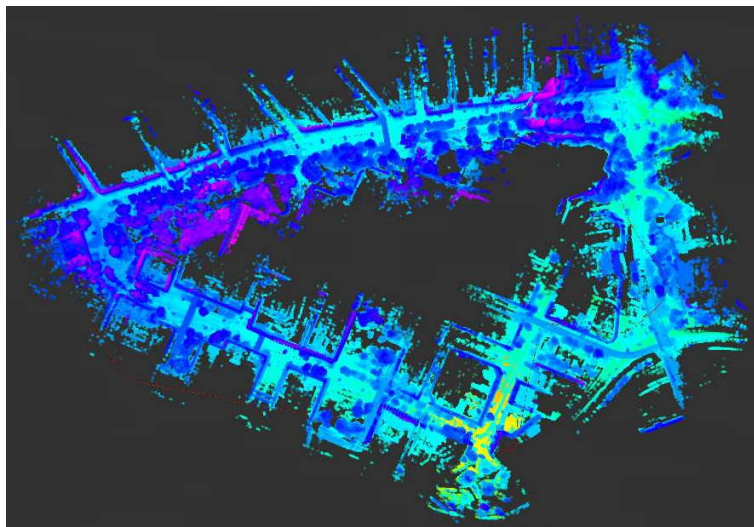
## 5) 没有去除运动畸变

如前所述，没有用到速度信息，所以没有做运动畸变去除。

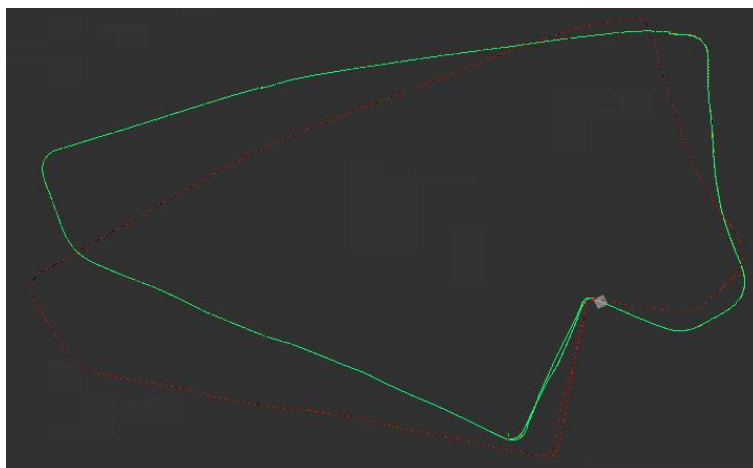
助教老师说单纯去畸变，可以按照 loam 的思路，先 scan 2 scan，然后那这个位姿去畸变，或者对相邻两帧的 GPS 数据差分获得速度。自己还没有尝试实现。

## 仿真结果：

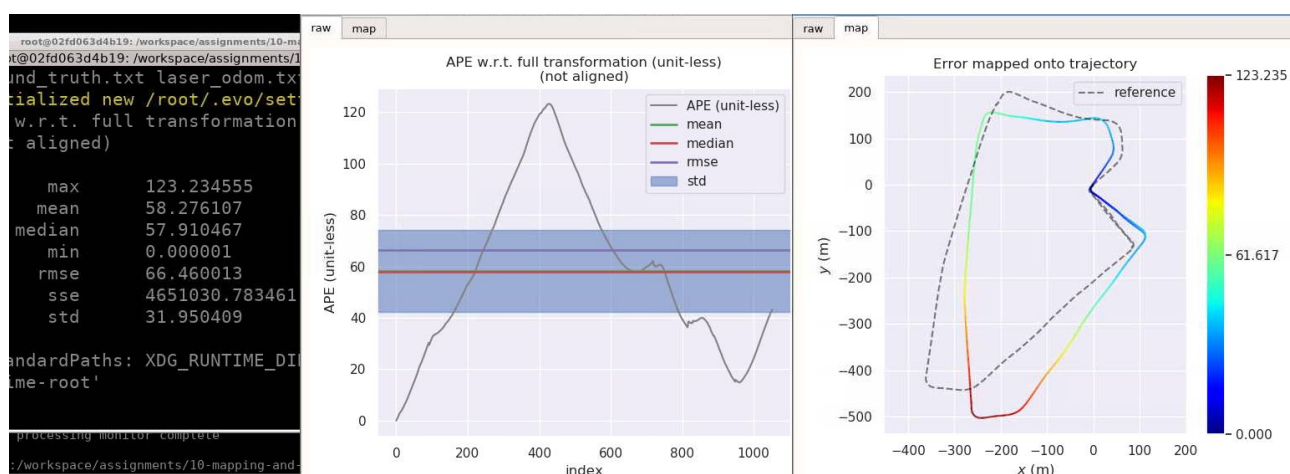
优化后的点云效果：



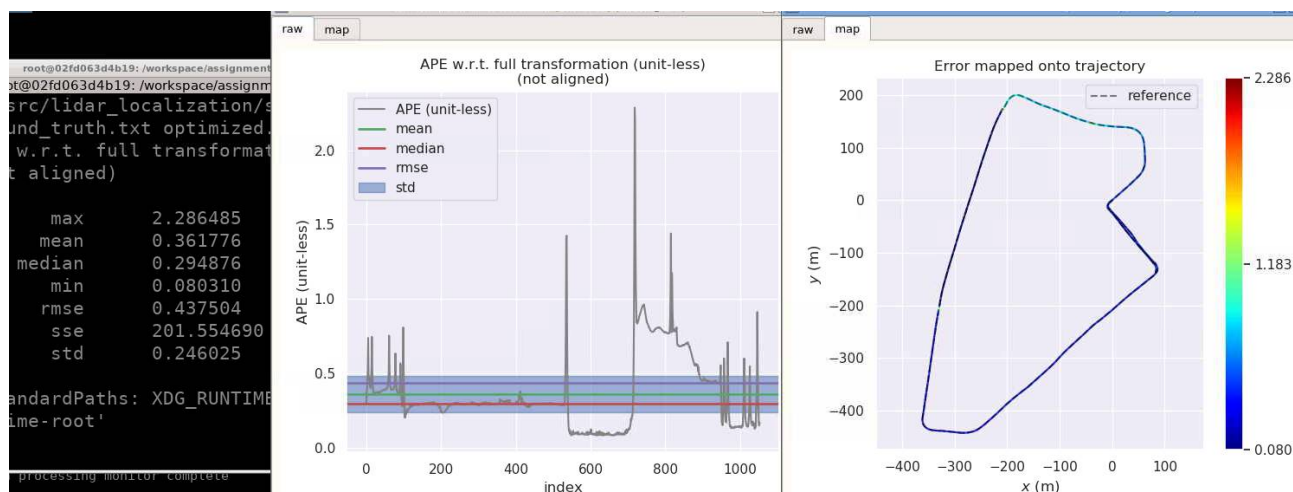
优化后轨迹效果图：



对 laser\_odom 进行评估:



对 optimized\_odom 进行评估:





## 2.2.1 测试不加融合的定位(第 4 讲)

对应代码文件：10-mapping-and-matching

### 程序修改要点说明：

使用 2.1.1 中生成的 map 和 scan context 文件，修改 matching.yaml 中的两个路径：

# a. scan context:

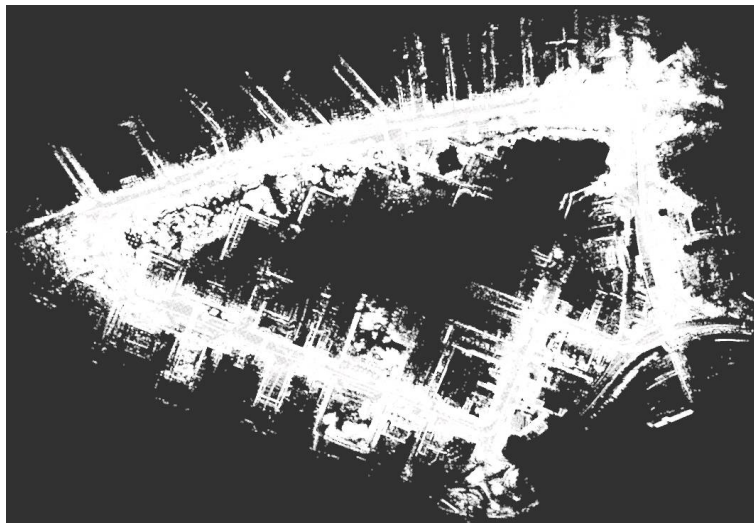
```
scan_context_path: /workspace/assignments/10-mapping-and-matching/src/lidar_localization/  
slam_data/scan_context
```

# b. global map:

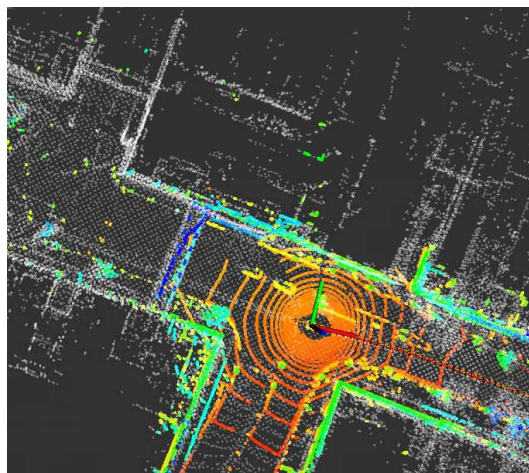
```
map_path: /workspace/assignments/10-mapping-and-matching/src/lidar_localization/  
slam_data/map/filtered_map.pcd
```

### 仿真结果：

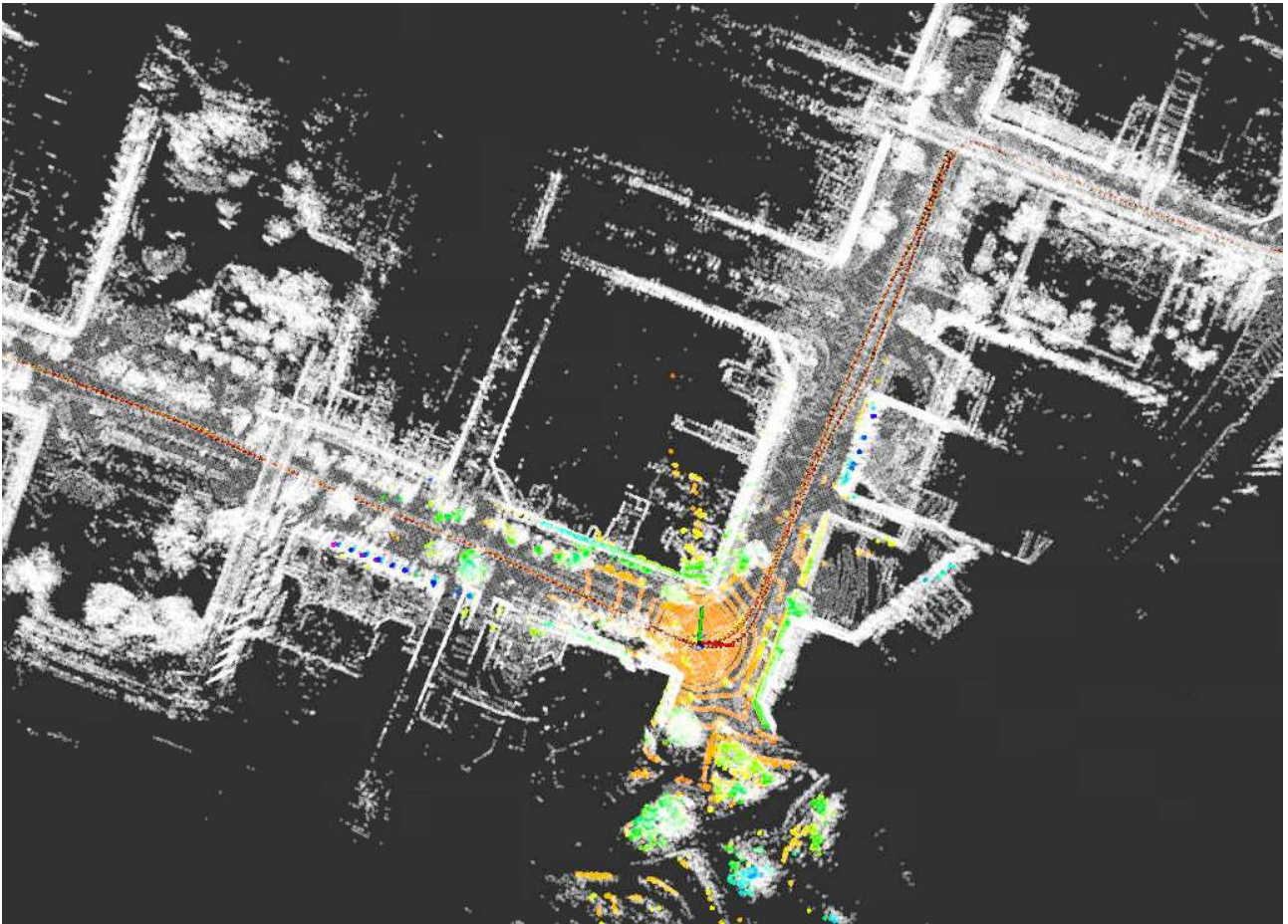
地图加载效果图：



定位过程图：



数据运行完后的结果图：



## 2.2.2 测试基于滤波的定位(第 7 讲)

对应代码文件：10-filtering-advanced

### 程序修改要点说明：

- 1) 修改重力参数：在 <https://www.sensorsone.com/local-gravity-calculator/> 中计算了香港 (Latitude 22°26'N，即 22.43°，Height 0) 的重力参数为  $9.78785 \text{ ms}^{-2}$ 。修改了文件 “kitti\_filtering.yaml” 中 error\_state\_kalman\_filter 空间下的 gravity\_magnitude 和 latitude。
- 2) 惯性器件参数只考虑了加速度计和陀螺仪的白噪声，没有考虑加速度计和陀螺仪的随机游走对应的白噪声。考虑到噪声参数本身需要调参，所以噪声参数采用原来代码框架中默认的参数，也没有比较不同参数下的仿真结果，只是跑通了算法。

3) 对融合算法解决同时使用原始 imu 数据 current\_imu\_raw\_data 和时间同步后的 imu 数据 current\_imu\_synced\_data\_所造成的“timestamp chaos”问题所对应的代码进行了修改，即对 kitti\_filtering\_flow.cpp 中的 Run()函数做了如下修改：

```
94 // if ( HasIMUData() && ValidIMUData() ) {
95 //     UpdateLocalization(); //CF
96 // }
97
98 if ( current_imu_raw_data_.time <= current_cloud_data_.time ) {
99     if (HasIMUData() && ValidIMUData() ) {
100         UpdateLocalization();
101     }
102 } else {
103     break;
104 }
```

修改之前，对于新的数据集，使用订阅到的 current\_imu\_raw\_data 进行更新后总是导致其时间戳“远大于”最新订阅到的点云时间戳，从而不能正常进行卡尔曼滤波更新，频繁输出提示"ESKF Correct: Observation is not synced with filter. Skip, " 如下图：

```
filter. Skip, 1556456768 <-- 1556456769 @ -0.577888
I0421 08:26:04.336586 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456768 <-- 1556456769 @ -0.47778
I0421 08:26:04.394915 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456769 @ -0.377713
I0421 08:26:04.456202 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456769 @ -0.277615
I0421 08:26:04.506812 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456769 @ -0.177532
I0421 08:26:04.569655 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456769 @ -0.077456
I0421 08:26:05.185979 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456770 @ -0.867399
I0421 08:26:05.221248 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456770 @ -0.787246
I0421 08:26:05.261180 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456770 @ -0.687162
I0421 08:26:05.305053 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456770 @ -0.587069
I0421 08:26:05.351145 21456 error_state_kalman_filter.cpp:278] ESKF Correct: Observation is not synced with
filter. Skip, 1556456769 <-- 1556456770 @ -0.487009
```

修改之后，增加了启用 current\_imu\_raw\_data 进行里程计更新的判断条件，避免了上述问题。

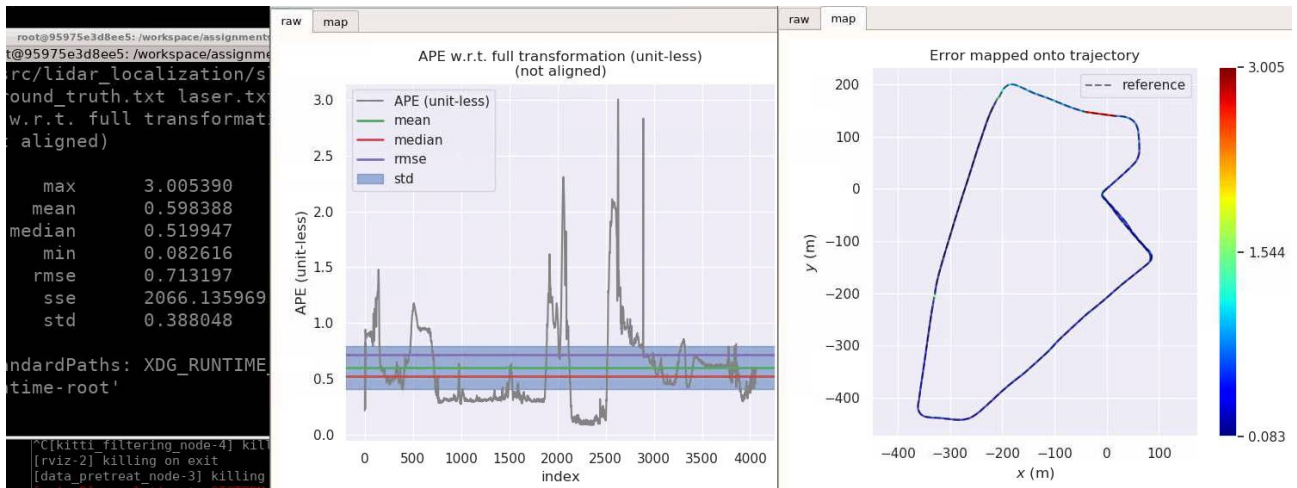
#### 4) 地图与匹配所需文件

将 lidar\_localization/workspace/assignments/09-sliding-slam\_data 下的 map 和 scan\_context 全部换使用 2.1.1 代码框架下对应的文件。

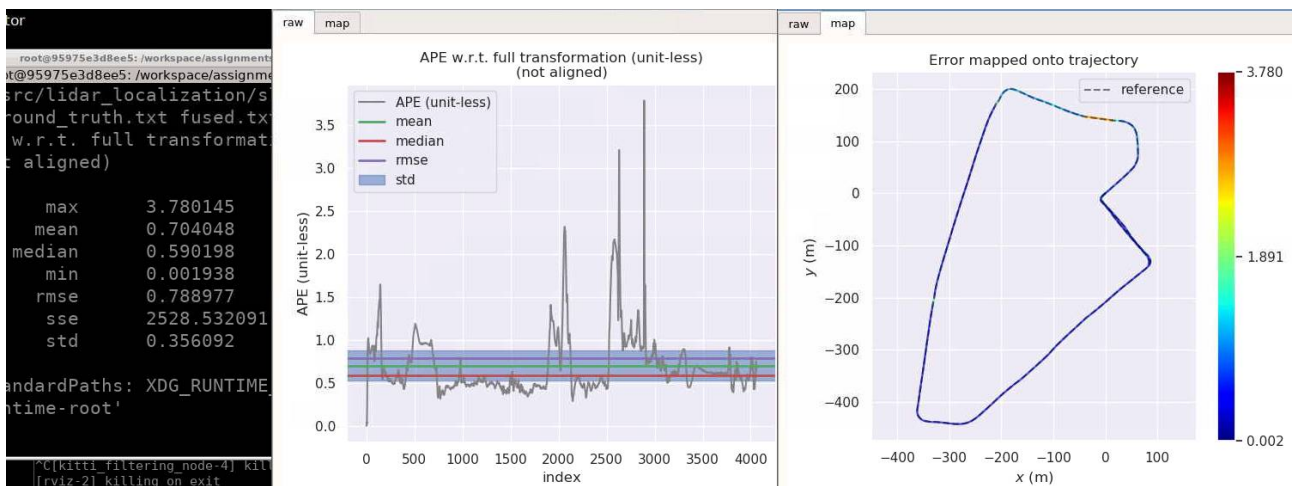
## 仿真结果：

laser:





fused:



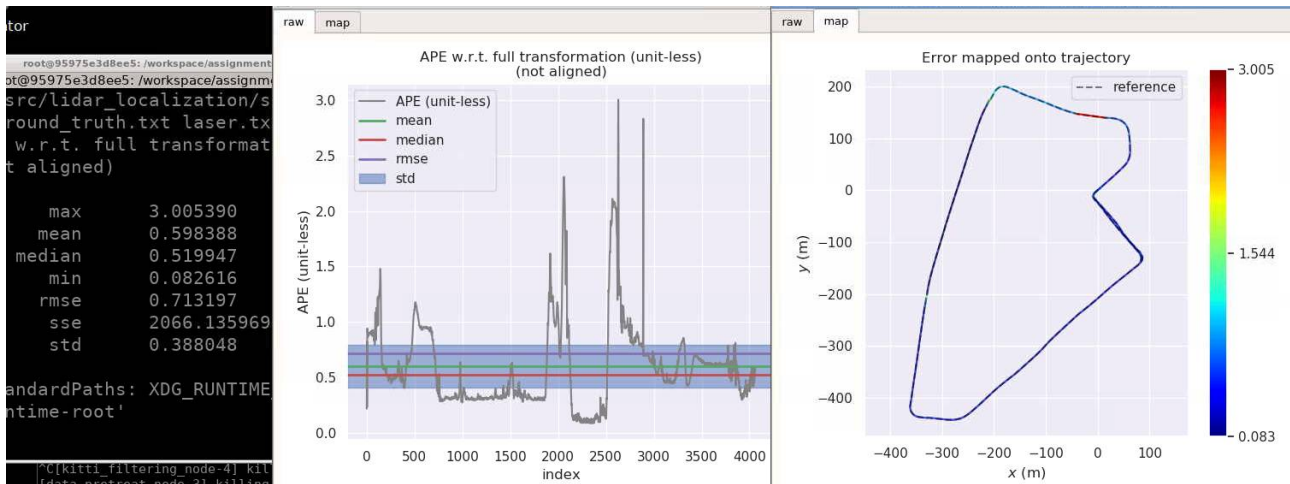
## 2.2.3 测试基于滤波+运动约束的定位(第 8 讲)

对应代码文件: 10-filtering-advanced

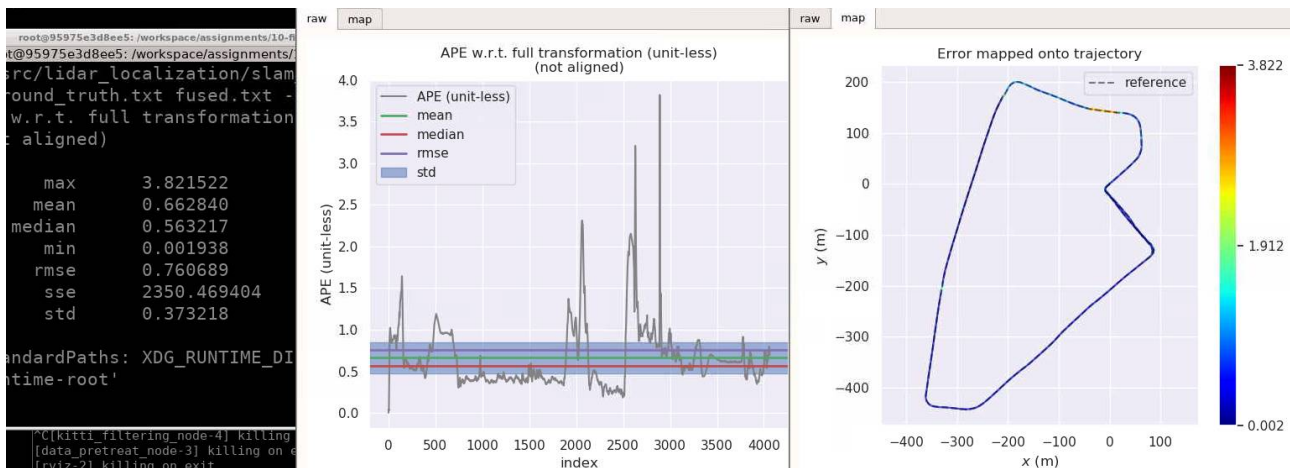
算法框架与 2.2.2 相同, 只需要修改 kitti\_filtering.yaml 中的 fusion\_strategy 为 pose\_velocity\_constraint 即可。

仿真结果:

laser:



fused:



良好

## 2.1.2 测试基于“使用预积分的融合”的建图(第 9 讲)

抱歉，这部分内容没有实践，对于预积分的测试，下面的 2.2.4 中会涉及到。

## 2.2.4 测试基于滑动窗口的定位(第 10 讲)

对应代码文件：10-sliding-window

### 程序修改要点说明：

- 1) 地图与匹配所需文件

将 lidar\_localization/workspace/assignments/09-sliding-slam\_data 下的 map 和 scan\_context 全部换使用 2.1.1 代码框架下对应的文件。

修改文件 matching.yaml 中的参数：

# a. global map for relative pose estimation using lidar frontend:

map\_path: /workspace/assignments/09-sliding-window/src/lidar\_localization/slam\_data/map/  
filtered\_map.pcd

# b. scan context index for map matching:

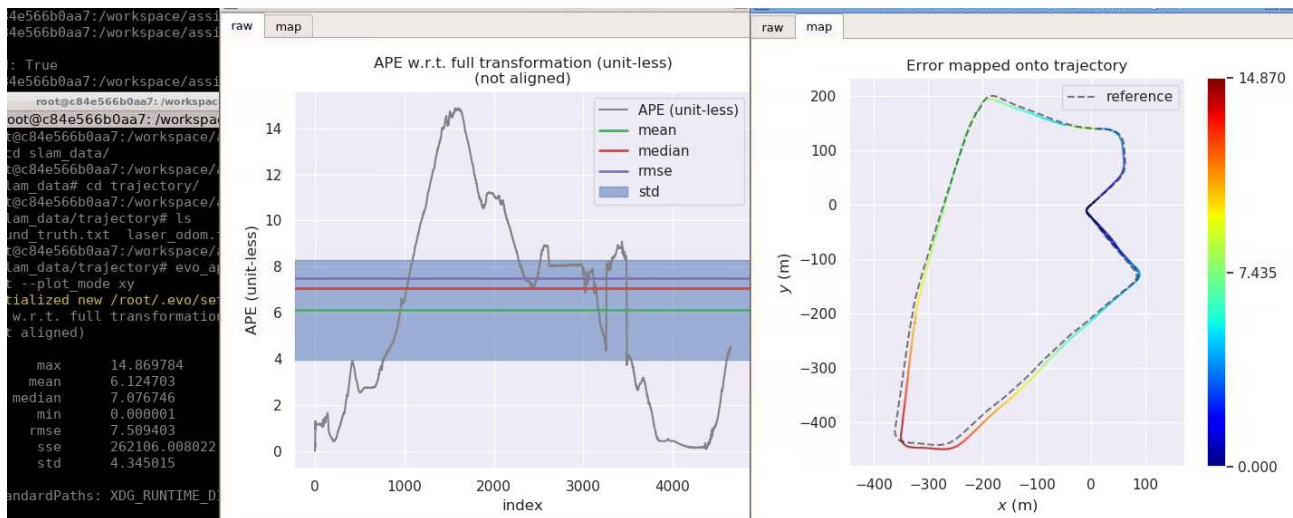
scan\_context\_path:  
/workspace/assignments/09-sliding-window/src/lidar\_localization/slam\_data/scan\_context

sliding\_window\_size: 20

gravity\_magnitude: 9.80943

## 仿真结果：

laser:



optimized:

```
c84e566b0aa7:/workspace/as
c84e566b0aa7:/workspace/as
d: True
c84e566b0aa7:/workspace/as
root@c84e566b0aa7:/works
root@c84e566b0aa7:/works
mean 6.124703
median 7.076746
min 0.000001
rmse 7.509403
sse 262106.0080
std 4.345015

StandardPaths: XDG_RUNTIME_
t@c84e566b0aa7:/workspac
lam_data/trajectory# evo
--plot_mode xy
w.r.t. full transformat
t aligned)

max 5.573107
mean 0.506484
median 0.340700
min 0.000001
rmse 0.895721
sse 3729.167349
std 0.738777

StandardPaths: XDG_RUNTIME_
```

