1. 补充去除激光雷达运动畸变模块的代码；(6 )

代码：

    void Lidar\_MotionCalibration(

        tf::Stamped<tf::Pose>*frame\_base\_pose*,

        tf::Stamped<tf::Pose>*frame\_start\_pose*,

        tf::Stamped<tf::Pose>*frame\_end\_pose*,

        std::vector<double>&*ranges*,

        std::vector<double>&*angles*,

        int *startIndex*,

        int&*beam\_number*)

    {

*//TODO*

*// frame\_base\_pose.inverse;*

        tf::Quaternion start\_q = frame\_start\_pose.getRotation();

        tf::Quaternion end\_q = frame\_end\_pose.getRotation();

        tf::Vector3 start\_xy(frame\_start\_pose.getOrigin().getX(), frame\_start\_pose.getOrigin().getY(), 1);

        tf::Vector3 end\_xy(frame\_end\_pose.getOrigin().getX(), frame\_end\_pose.getOrigin().getY(), 1);

        for (size\_t i = startIndex; i < startIndex + beam\_number; i++) {

            tf::Vector3 mid\_xy = start\_xy.lerp(end\_xy, (i - startIndex) / (beam\_number - 1));

            tf::Quaternion mid\_q = start\_q.slerp(end\_q, (i - startIndex) / (beam\_number - 1));

            tf::Transform mid\_frame(mid\_q, mid\_xy);

            double x = ranges[i] \* cos(angles[i]);

            double y = ranges[i] \* sin(angles[i]);

            tf::Vector3 calib\_point = frame\_base\_pose.inverse() \* mid\_frame \* tf::Vector3(x, y, 1);

            ranges[i] = sqrt(calib\_point[0] \* calib\_point[0] + calib\_point[1] \* calib\_point[1]);

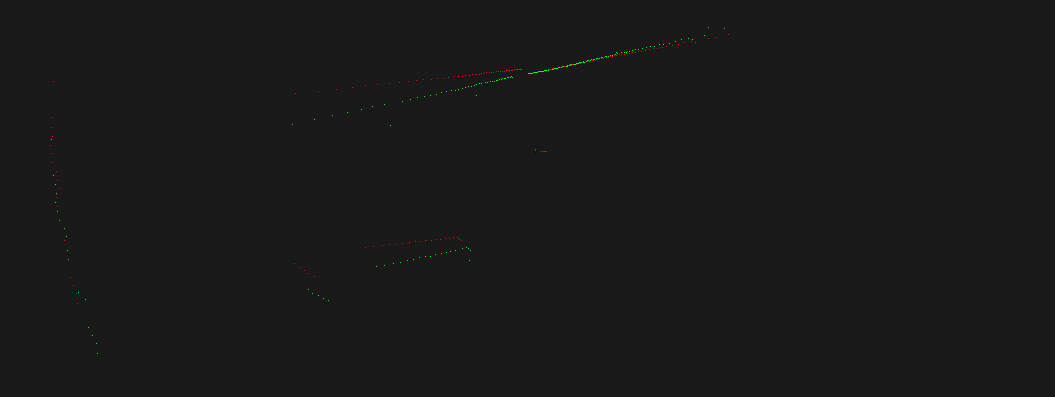
            angles[i] = atan2(calib\_point[1], calib\_point[0]);

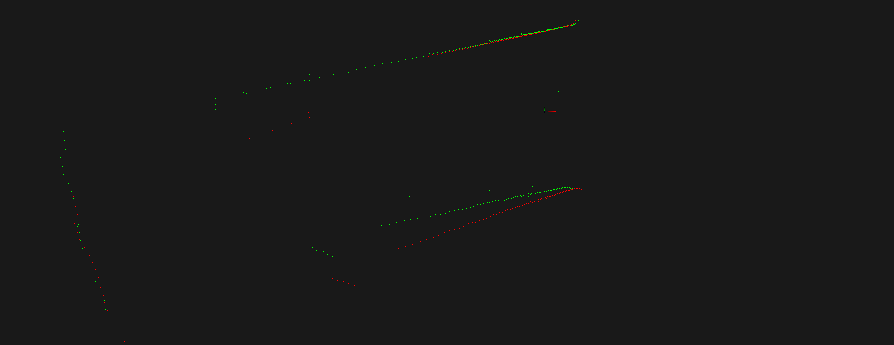
        }

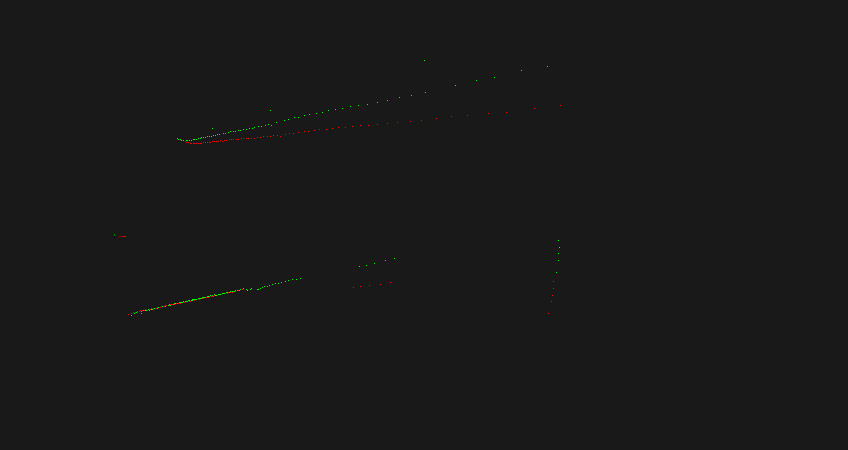
*//end of TODO*

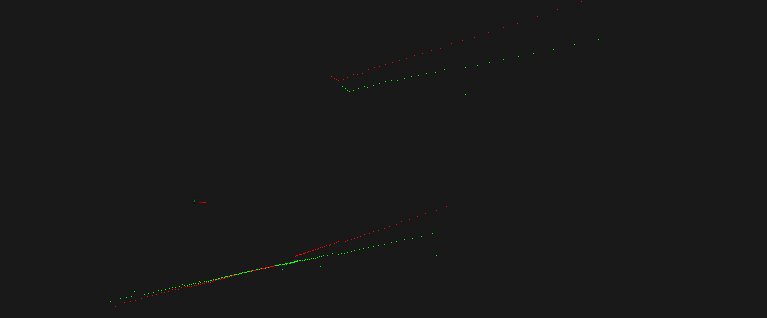
    }

运行效果：





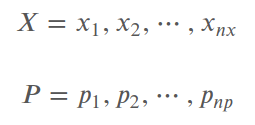




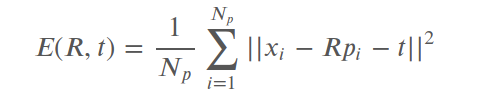
1. 阅读论文 Least-Squares Fitting of Two 3-D Points Sets，推导并证明已知对应点的 ICP 求解方法；

推导：

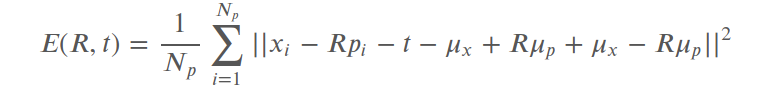
对于两组点云：



求解R和t，使得下式最小：

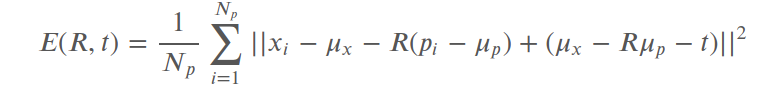


上式进行变换：

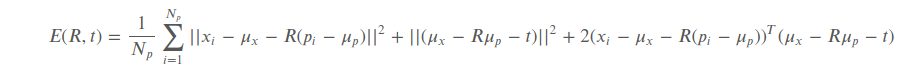


其中，цx 和 цp分别为X点云和P点云的质心。

转换为去除质心的形式：



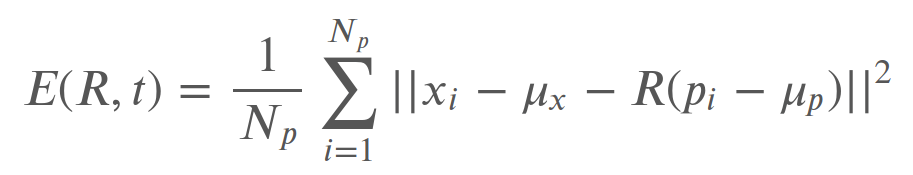
展开得到



由于后面两项都有

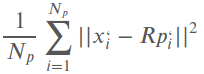


我们可以令该项等于0，然后式子转化为

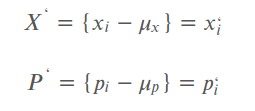


该式子仅与R有关，可以寻找可以最小化上式的R，然后根据=0求解t。

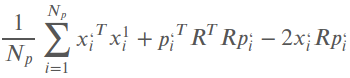
上式进一步转化，得到：



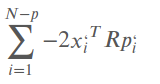
其中，X’ 和 P‘ 是原始点云减去对应质心的点点云，x’和p’是其中的点。



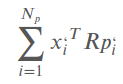
展开得到：



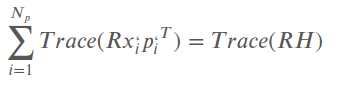
其中旋转矩阵R因为是正交矩阵，所以RTR=1，因此求上式的最小值等价于求最后一项的最小值：



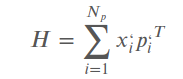
即该式的最大值：



进行转化，用矩阵的迹进行表示：



其中，



对H进行SVD分解得到：



设定一个变量X为



得到：



此时得到的XH是一个正定对称矩阵。

根据定理：

假设矩阵A为正定对称矩阵，则对于任意的正交矩阵B，都有



类比到我们的推导这里，我们准备求解可以让Trace(RH) 最大的R值，我们发现，当R=VUT的时候，RH的结果是一个正定对称矩阵，而对于任意一个正定对称矩阵来说，根据定理可以知道，我们再乘以任何一个正交矩阵（旋转矩阵是正交矩阵），都会导致Trace值下降，所以说明，此时的R=VUT取值可以让Trace(RH)取得最大值。

因此：





3. 阅读论文 Precise indoor localization for mobile laser scanner 前两章，回答问题。（2 分）

（1）根据第二章内容，简述激光雷达测距原理；

（2）简要介绍一下右图的含义。

(1)

激光雷达测距原理分为两种：一种是根据三角法测量距离，另一种是测量飞行时间，即TOF。

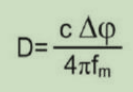
后者又分为两种测量方式，分别是直接测量飞行时间和测量相位差。

公式如下所示：

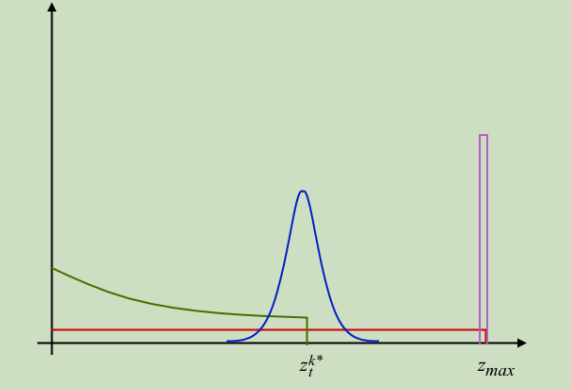
直接测量飞行时间：



测量相位差：



（2）



右图是激光雷达光束模型的联合概率分布图。

其中ZtK表示测量返回的距离值，Zmax表示激光雷达可能测量得到的最大数值。

由四种概率分布组成：

蓝色是高斯概率分布，代表实际测量距离及其不确定性。

绿色是指数分布，代表检测到动态障碍物的可行性。

红色代表随机噪声。

粉色代表错误测量的可能性。

4. 简答题，开放性答案：设计使用 IMU 去除激光雷达运动畸变的方法并回答问题。

（1）仅用 IMU 去除运动畸变可能会有哪些不足之处？

（2）在仅有 IMU 和激光雷达传感器的情况下，你会如何设计运动畸变去除方案(平移+旋转)，达到较好的畸变去除效果?

（1）不足之处：

IMU对于线加速度进行二次积分可能误差较大

1. 设计：

在每一个激光雷达数据帧内部，使用imu进行积分，得到IMU里程计。

通过时间同步，然后确定每一个激光雷达点束的在IMU里程计坐标系下面的位置

将每一个激光雷达点束都矫正到该帧起始位姿下面。

对于n时刻矫正好的激光雷达数据，使用icp算法，将n时刻与n-1时刻的矫正好的激光雷达帧进行匹配，得到激光雷达匹配的位姿。

使用这个位姿纠正imu的累计误差，然后继续矫正下一枕数据。