## 3.1\_b A-LOAM for 3D LiDAR SLAM

A-LOAM (Advanced Lidar Odometry And Mapping) is an improved framework from LOAM, using Eigen and Ceres Solvers to simplify code structure. A-LOAM use two parallel algorithms: a low-precision high-speed odometer and a low-speed high-precision odometer, combining the two odometers to obtain real-time map updates. The feature matching method adopted improves the accuracy and operation efficiency of the algorithm.

In this task, we refer this repo[1] to help do the task and directly generate odometry file for **EVO** evaluation. The process of algorithm implementation is: 1) Develop the Aloam from repo[1] with modifications needed. 2) Create a folder named *‘txt’* with ‘*00.txt’* and empty *‘aloam.txt’*. 3) Launch the Aloam and then meanwhile play *‘<3dlidar.bag>’*. 4) Evaluate the performance of the algorithm with EVO*.*

Notice that in step 1, there are some modifications we did which the README.md do not mention.

* Firstly, uncomment the *“//generate the KITTI format trajectory result”* part in the *laserMapping.cpp,* so that theAloam lidar odometry result will be stored in *‘aloam.txt’.*
* Secondly, the ‘*ceres::LocalParameterization’* and ‘*ceres::EigenQuaternionParameterization’* are deprecated in the latest release of Ceres Solver (v 2.1.0) and need to be replaced to *‘ceres::Manifold’* and *‘ceres::QuaternionManifold()‘.*

The result of the EVO evaluation of this 3D SLAM task and the screenshot of algorithm running in rviz is shown in Figure 3.1.b.1.

|  |
| --- |
| 黑暗中的灯光  中度可信度描述已自动生成绿色的灯光  低可信度描述已自动生成 |
| Figure 3.1.b.1 3D SLAM algorithm running in rviz and performance |

From the result, it can be seen that the max error value is , the mean error value is and the min error value is . Thus, the performance of this algorithm is not bad.

To further improve the performance, we tried tuning some parameters (‘DISTANCE\_SQ\_THRESHOLD’ from to and ‘SCAN\_PERIOD’ from to ) in *LaserOdometry.cpp* and successfully reduced the maximum error from toas shown in Figure 3.1.b.2.

|  |
| --- |
|  |
| Figure 3.1.b.2 3D SLAM task result after tuning |

The disadvantage of A-LOAM is that it does not do back-end closed-loop detection, so perhaps we can seek to correct motion estimation drift through closed-loop detection. Alternatively, the output can be added to a filter, such as a Kalman filter, to further reduce motion estimation drift.

## Reference

[1] nuslde (2023) aloam\_lidar\_odom\_result\_generate [Source Code] https://github.com/nuslde/aloam\_lidar\_odom\_result\_generate