LIO-SAM

COMPARISON BETWEEN THE OPEN LOOP AND CLOSED LOOP

Manish Kumar Peteri. **Holmes Joseph** **Thaarun Sivakumar** **Jay Menon**

mpeteri@asu.edu hjoseph@asu.edu Tsivaku1@asu.edu jmenon2@asu.edu

ABSTRACT

LiDAR-Inertial Odometry and Mapping (LIO-SAM) combines IMU, LiDAR, and GPS for mapping and navigation with precision. It guarantees high accuracy, robustness, and computational efficiency, adapting to various environments. This benefits autonomous vehicles, robotics, and drones. We're conducting a comparison between closed-loop and open-loop models to assess accuracy disparities.

1 INTRODUCTION

The imperative for robust localization and mapping solutions has intensified with the rapid advancements in autonomous vehicles, robotics, and unmanned aerial drones. Within this context, LiDAR-Inertial Odometry and Mapping (LIO-SAM) emerges as a formidable approach, facilitating precise localization and mapping critical for autonomous navigation and environmental interaction.

The framework integrates LiDAR and inertial measurements to estimate the robot's pose and construct a map of its surroundings that offers several advantages over traditional localization and mapping methods by allowing a multitude of relative and absolute measurements, including loop closures, to be incorporated from different sources as factors into the system. [1]

We establish a baseline using the Velodyne LiDAR in the Gazebo simulation environment. Through our study, we aim to provide valuable insights into the integration of sensor fusion techniques LiDAR sensors with LIO SAM. By advancing localization and mapping capabilities, we contribute to the broader goal of enabling safer and more efficient autonomous systems across various domains.

2 RELATED WORK

Iterative Closest Point (ICP) and Generalized Iterative Closest Point (GICP) stand out as prominent algorithms in the realms of computer vision and robotics. They are instrumental in aligning or registering two 3D point clouds, crucial for tasks such as state estimation, localization, and mapping. Despite their popularity, ICP exhibits limitations in generalization and is prone to higher error rates [2][3].

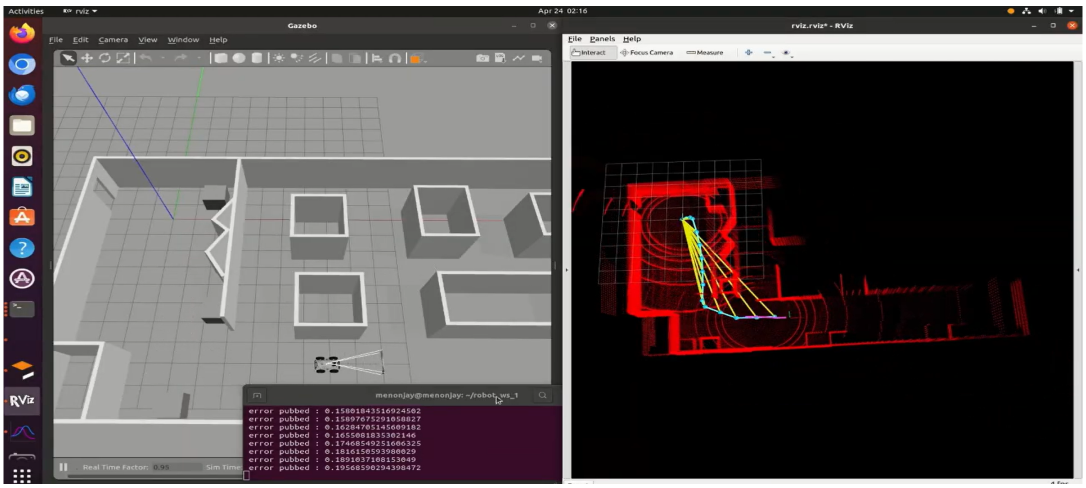
To facilitate state estimation, raw sensor data is often converted into point clouds [4], providing a basis for detecting the rotational and translational motion of the robot. An innovative approach proposed in [5] introduces a collar line-based method for odometry estimation. Here, line segments are randomly generated from the original point cloud and utilized for subsequent registration processes.

In the pursuit of fusion techniques, both loosely coupled and tightly coupled methodologies have been explored, each with its strengths and weaknesses. While loosely coupled fusion methods may offer simplicity, they often suffer from lower accuracies. Conversely, tightly coupled fusion systems, although more complex, typically yield superior accuracy [6], making them a focal point for sensor integration strategies.

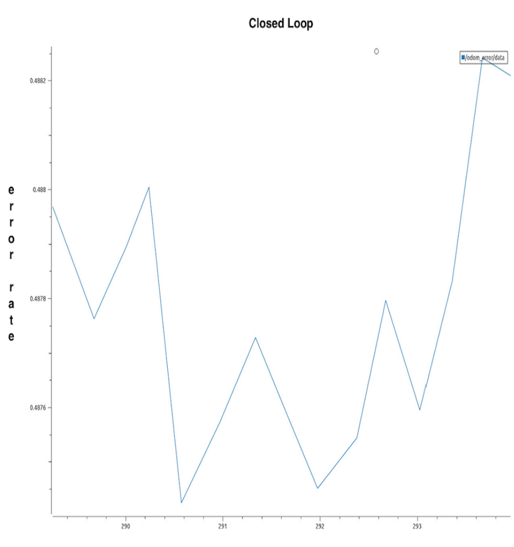
A notable contribution to tightly coupled sensor integration is the Robocentric Lidar-Inertial State Estimator (R-LINS), as introduced in [7]. Leveraging Kalman filtering techniques, R-LINS refines the robot's state estimate in a tightly coupled manner, emphasizing the significance of precise sensor fusion in robotics applications.

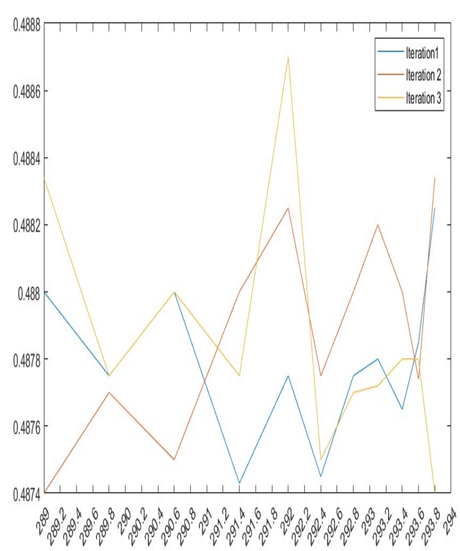
3 Baseline Results

The images illustrate that the LIO SAM system processes LiDAR data to control the UGV via keyboard-generated ROS messages. It constructs the map utilizing LIO SAM, although the accuracy of the building's representation, with the model overlooking several areas. Enhancements in IMU and LiDAR calibration could be explored to elevate the system's mapping accuracy and overall performance.



Error Graph:





Time	289	289.8	290.6	291.4	292	292.4	292.8	293.1	293.4	293.6	293.8
Y Error1	0.4880	0.48775	0.488	0.48743	0.48775	0.48745	0.48775	0.4878	0.48765	0.48785	0.48825
Error2	0.4874	0.48770	0.4875	0.4880	0.48775	0.48775	0.4880	0.4882	0.4880	0.48774	0.48834
Error 3	0.4880	0.48775	0.4887	0.48750	0.48770	0.48772	0.48780	0.4878	0.48775	0.48785	0.48815
Mean	0.4878	0.48773	0.4806	0.4876	0.48773	0.48764	0.48785	0.48793	0.4878	0.48781	0.48824

Table 1: The above table shows the error rates over 3 iterations using closed loop.

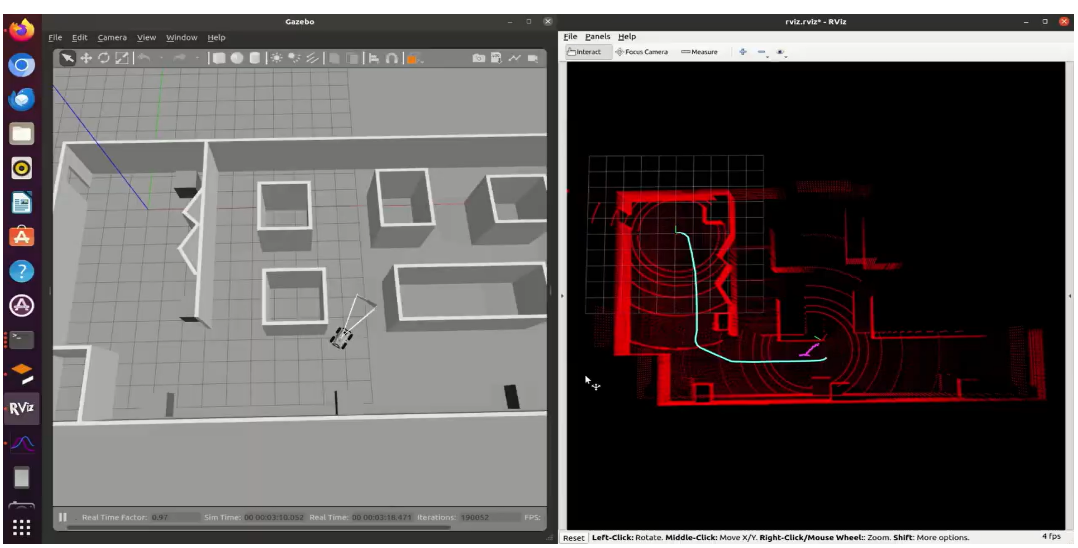
Proposal and Results:

The robotic system initially operated under manual control, receiving input via ROS messages. During this phase, the error rates primarily reflected human operational inaccuracies. To enhance the system's autonomy and environmental interaction, it was configured to operate autonomously, aiming to minimize collisions. This adjustment allowed the robot to map its surroundings with increased precision.

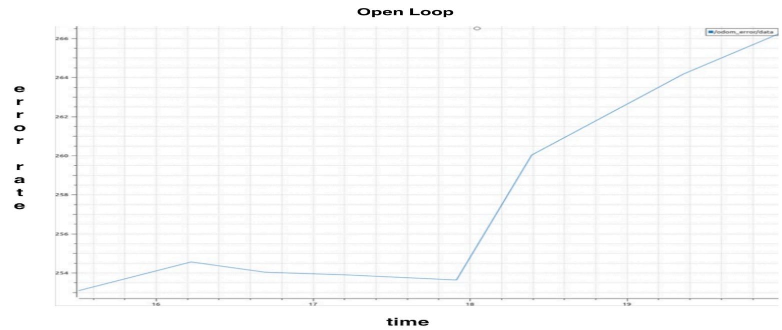
The initial implementation utilized a closed-loop control system, which demonstrated high accuracy levels. However, over time, a notable decline in accuracy was observed, following which there was an improvement. To further explore system capabilities and address the fluctuating accuracy, an open-loop control method was implemented, which lacks feedback mechanisms.

Initially, the open-loop system exhibited promising accuracy with notably low error rates. Nevertheless, as the system continued to operate over time, the error rates increased significantly. This observation suggests that while the open-loop method can effectively handle specific scenarios with initially low error rates, its lack of feedback mechanisms might lead to deteriorating performance as operational variables change or as the system encounters unforeseen environmental interactions.

Based on these findings, it is proposed that an integrated approach utilizing both open-loop and closed-loop methodologies could be beneficial. By switching between these systems based on real-time accuracy assessments, the robotic system could potentially optimize performance. Such a hybrid approach allows for the robustness and adaptive benefits of closed-loop control while also harnessing the simplicity and initial efficiency of open-loop control.



Error graph:



Even after some time, open loop system was more stable due to the environment's unchanging nature.

4 CONCLUSION & FUTURE WORK

In our work we used sensor integration of IMU, LiDAR and GPS data in a virtual environment and used the Kalman filter on the processed data in the virtual environment. We used route mean square error method to calculate the error in the computation and movement. The open loop results were not great compared to the closed loop, but we observe that the open loop error was minimum at certain less dense environments. Going forward we may explore a combination of both closed loop and open loop so the model can attain higher accuracy and efficiency rate. Also, techniques like the tightly coupled techniques can potentially improve the accuracy and reduce the error rates.

5 REFERENCES

[1] L. Huang, "Review on LiDAR-based SLAM Techniques," 2021 International Conference on Signal Processing and Machine Learning (CONF-SPML), Stanford, CA, USA, 2021, pp. 163-168, doi:10.1109/CONF-SPML54095.2021.00040.

[2] P.J. Besl and N.D. McKay, "A Method for Registration of 3D Shapes," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14(2): 239-256, 1992]

[3] GICP A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," Proceedings of Robotics: Science and Systems, 2009.

[4] W.S. Grant, R.C. Voorhies, and L. Itti, "Finding Planes in LiDAR Point Clouds for Real-time Registration," IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4347-4354, 2013

[5] M. Velas, M. Spanel, and A. Herout, "Collar Line Segments for Fast Odometry Estimation from Velodyne Point Clouds," IEEE International Conference on Robotics and Automation, pp. 4486-4495, 20

[6] C. Chen, H. Zhu, M. Li, and S. You, "A Review of Visual-Inertial Simultaneous Localization and Mapping from Filtering-Based and Optimization-Based Perspectives," Robotics, vol. 7(3):45, 2018.

[7] C. Qin, H. Ye, C.E. Pranata, J. Han, S. Zhang, and Ming Liu, "RLINS: A Robocentric Lidar-Inertial State Estimator for Robust and Efficient Navigation," arXiv:1907.02233, 2019.

Bibliography

https://github.com/TixiaoShan/LIO-SAM
https://github.com/1475015695/liosam_liauto
https://github.com/linzs-online/robot_gazebo
https://blog.csdn.net/weixin_40599145/article/details/126929222
https://arxiv.org/abs/2007.00258
https://www.scirp.org/reference/referencespapers?referenceid=1120390
https://www.scirp.org/reference/referencespapers?referenceid=1120390
http://vigir.missouri.edu/~gdesouza/Research/Conference_CDs/IEEE_IROS_2013/media/files/0337.pdf