LiDAR-Based 3D SLAM for Indoor Mapping

Teng Hooi Chan
Department of Aerospace and Geodesy
Technical University of Munich
Germany
tenghooi91@gmail.com

Henrik Hesse
Aerospace Sciences Division
University of Glasgow (Singapore)
Singapore
Henrik.Hesse@glasgow.ac.uk

Song Guang Ho
ST Engineering-NTU Corporate Lab
Nanyang Technological University
Singapore
hosongguang@gmail.com

***Abstract*—Aiming to develop methods for real-time 3D scanning of building interiors, this work evaluates the performance of state-of-the-art LiDAR-based approaches for 3D simultaneous localisation and mapping (SLAM) in indoor environments. A simulation framework using ROS and Gazebo has been implemented to compare different methods based on LiDAR odometry and mapping (LOAM). The featureless environments typically found in interiors of commercial and industrial buildings pose significant challenges for LiDAR-based SLAM frameworks, resulting in drift or breakdown of the processes. The results from this paper provide performance criteria for indoor SLAM applications, comparing different room topologies and levels of clutter. The modular nature of the simulation environment provides a framework for future SLAM development and benchmarking specific to indoor environments.**

***Index Terms*—SLAM, Lidar Odometry, Indoor Mapping**

I. Introduction

With autonomous vehicles becoming more popular for a range of commercial and industrial applications, such as autonomous driving or autonomous inspection, it is critical to develop robust solutions for simultaneous localisation and mapping (SLAM). In the past two decades, the robotics community has put a great effort into developing SLAM solutions. Among these solutions, vision-based and LiDAR-based methods are most popular. With many SLAM frameworks available in the public domain, this work aims to evaluate the latest open source LiDAR-based 3D SLAM frameworks specifically for the application in indoor environments.

In GPS-denied indoor environments, localisation based on GPS is not feasible. Autonomous mobile robots such as unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs) require dedicated localisation methods such as SLAM to navigate the indoor setting and perform autonomous tasks. Current visual-based SLAM frameworks such as [5] and [6] rely on monocular or RGB-D cameras for real time mapping of the environment. These methods have advantages in loop-closure detection, however as they are sensitive to illumination and range, they are unreliable when used for autonomous tasks without fusion of additional sensors. On the other hand, LiDAR is insensitive to illumination change. High resolution and long range of 3D LiDARs permit the capture of fine details of the environment which makes them favourable for large scale mapping tasks. In particular, in this work we consider the automated inspection of large indoor environments with UAVs or UGVs. This task requires the vehicles to localise themselves accurately within the space and build 3D maps of large infrastructures. For these applications we favour a LiDAR-based SLAM approach which is able to perform building inspection with high resolution and detailed depth information.

When LiDAR is used to generate 3D maps of the surrounding through a 3D SLAM approach, the sensor odometry estimation and map optimization are performed by processing consecutive scans. Due to the amount of data collected, it usually requires high computational resources and hence poses a limitation on real time applications. Moreover, indoor environments are considered as degraded environments as there are significantly less features compared to outdoor feature-rich environments. Featureless environments strongly impair the performance of LiDAR-based SLAM frameworks, resulting in large drift or breakdown of the process.

Many LiDAR-based 3D SLAM frameworks have been proposed in the past. The LiDAR odometry and mapping (LOAM) method [10] is among the first real-time LiDAR-based 3D SLAM methods that achieves state-of-the-art results. Among the latest 3D SLAM frameworks, a lightweight and ground-optimized LiDAR odometry and mapping (LeGO-LOAM) method [8] and LiDAR inertial odometry via smoothing and mapping (LIO-SAM) method [9] were developed based on the underlying LOAM concept proposed in [10].

The aim of this paper is twofold. Firstly, we propose a series of simulation environments which allows the LiDAR-based SLAM methods to be tested against different geometries and pinpoint challenges in indoor SLAM operation. These geometries include empty square rooms, long narrow corridors, and circular rooms, which are representatives of challenging indoor scenarios with lack of distinct features and can occur in modern building designs. We further populate these environments with everyday objects to compare the behaviour of the SLAM frameworks in featureless against feature-rich environments. The latter has been demonstrated for a realistic warehouse setting where the SLAM methods provide excellent results. Secondly, we aim to implement and evaluate the performance of the two state-of-the-art LiDAR-based 3D SLAM frameworks for indoor environments. The two frameworks namely LeGO-LOAM and LIO-SAM were selected because of their availability in the public domain and have been tested extensively in outdoor settings that gave state-of-the-art results as mentioned in [8] and [9], respectively.

II. Related Work

Among the many LiDAR-based 3D SLAM methods, LOAM [10] is a common real-time LiDAR odometry estimation and mapping framework that uses LiDAR data and optionally inertial measurement unit (IMU) for pose estimation. This method achieves real-time performance by separating the SLAM problem into odometry estimation and mapping optimization algorithms. The odometry estimation algorithm runs at high frequency with low fidelity while the mapping optimization algorithm runs at an order of magnitude lower frequency with high accuracy for scan-matching. Since its publication, LOAM has remained top ranked in the odometry category of the KITTI Vision Benchmark Suite [2]. LOAM has since been commercialized and its framework is no longer available in the public domain.

LeGO-LOAM [8] is similar to LOAM whereby two algorithms running at different frequencies are implemented. It separates the ground plane and performs cloud segmentation using a range image before feature extraction to further lighten the computational load. A two-step Levenberg-Marquardt optimization method is used to perform a six-degree-of-freedom (DOF) pose estimation. LeGO-LOAM achieves similar or better results than LOAM when evaluated using the same KITTI datasets.

A more recent framework is LIO-SAM [9] which utilizes factor graphs to incorporate multiple measurement factors for odometry estimation and global map optimization. The framework incorporates an IMU preintegration to deskew the incoming point cloud data and performs fast computation using key frames rather than consecutive scans. The IMU preintegration factor serves as an initial guess as well. LIO-SAM allows the incorporation of a GPS factor as an option for additional key factor.

Both LeGO-LOAM and LIO-SAM use incremental smoothing and mapping (iSAM2) [3] to perform factor graph optimization. The same iterative closest point (ICP) [1] method is implemented in these two frameworks for loop closure.

III. System Overview

In this paper, we utilized the Robot Operating System (ROS) [7] and Gazebo simulator [4] to perform various simulation scenarios. ROS is an open source robot operating system that provides a structured communication layer on top of the host operating systems. Gazebo is a well-known open source robot simulation tool that allows accurate and efficient simulation of robots in complex indoor and outdoor environments.

For each test case a Gazebo world model is generated. A "Turtlebot3 Waffle" mobile robot model (as shown in Fig. 1) equipped with a Velodyne VLP-16 3D-LiDAR and a 9-DOF IMU sensor travels around the environment to collect measurements. These measurements are then published to the mentioned frameworks running in ROS for LiDAR odometry and mapping optimization process. The estimated trajectory from the SLAM frameworks can then be used to compare with the ground truth of the trajectory of the mobile robot extracted from Gazebo.

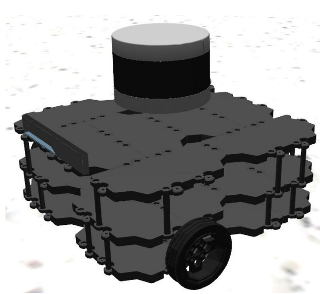
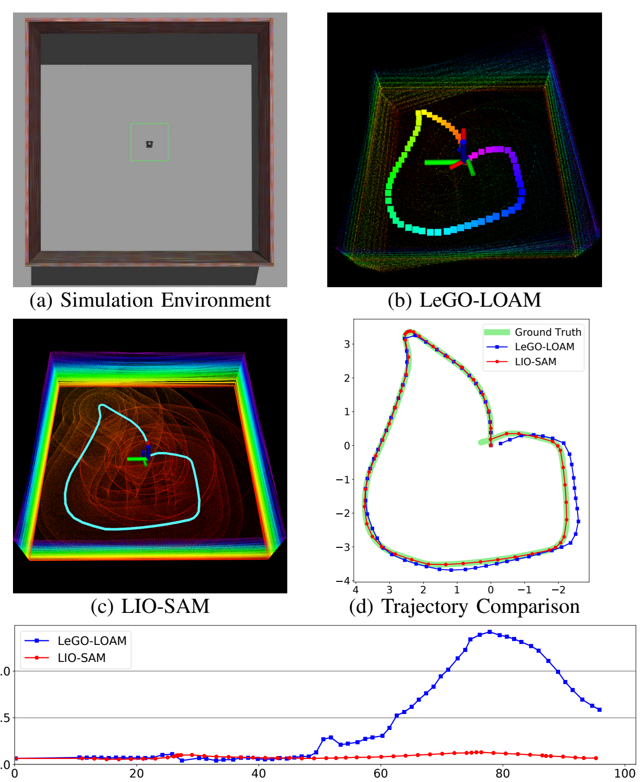


Fig. 1. A "Turtlebot3 Waffle" Gazebo model by Open Robotics is equipped with a Velodyne VLP-16 LiDAR, an Intel Realsense RGB-D camera and a 9-DOF IMU. The RGB-D camera was not used in our simulations.

IV. Simulation

This section describes a series of simulations to qualitatively and quantitatively analyze LeGO-LOAM and LIO-SAM in different indoor environments. We compare different basic room geometries, namely square room and long narrow corridor, typically found in indoor environments. We further populated these environments with everyday objects to compare the result of featureless against feature-rich environments. We further considered additional more complex scenarios but for brevity of the paper present results only for an L-shaped, a closed square-loop and a complex warehouse environment. The estimated trajectories from the two frameworks are then



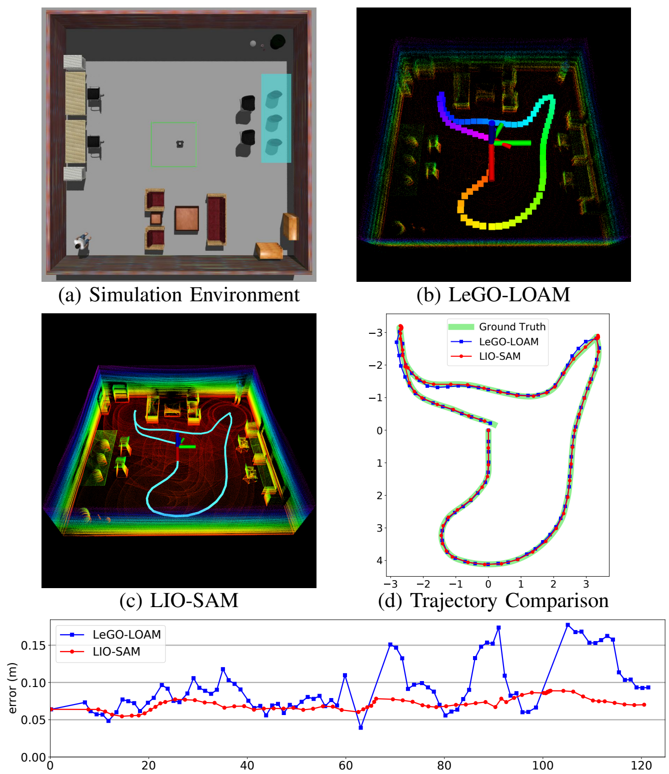
(a) Simulation Environment (b) LeGO-LOAM (c) LIO-SAM (d) Trajectory Comparison (e) Absolute Trajectory Error

Fig. 2. Simulation results for square room.

plotted against the ground truth trajectory. The absolute translation errors from both frameworks are also plotted to show the variation of the errors throughout the simulation. The root mean square errors of each trajectory estimated by both frameworks are tabulated at Table I at the end of this section. Complete visualizations of all simulations, can be found at the link below.[1]

A. Square Room Environment

This simulation sets in a square room which is encountered in almost every indoor environment. In featureless setting, both LeGO-LOAM and LIO-SAM are able to estimate trajectories close to the ground truth. With LeGO-LOAM showed inaccurate orientation estimation resulting in angular misalignment of the map. Both frameworks performed well in the feature-rich setting. The estimated trajectories of both frameworks for featureless and feature-rich square room are shown in Figs. 2-3 along with ground truth.

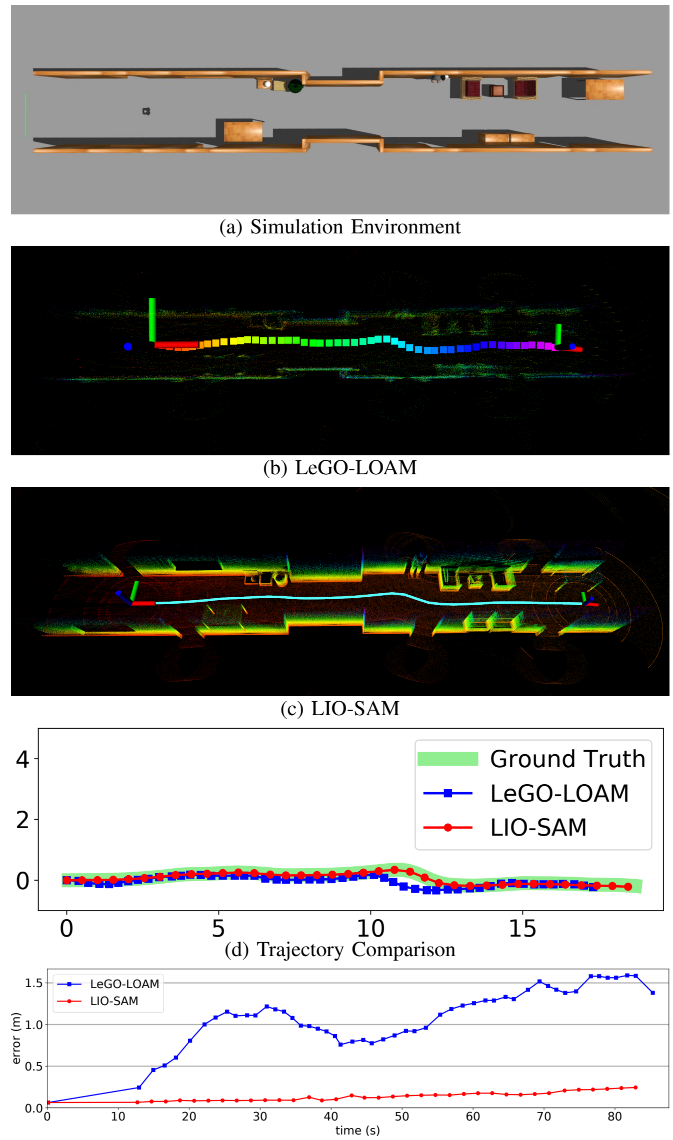


(a) Simulation Environment (b) LeGO-LOAM (c) LIO-SAM (d) Trajectory Comparison (e) Absolute Trajectory Error

Fig. 3. Simulation results for square room with features.

B. Corridor Environment

The corridor environment is created to simulate the "endless corridor" scenario. The "endless corridor" is a well-known highly degraded environment in the SLAM community. In such featureless environments, only planar features are detected and both LeGO-LOAM and LIO-SAM fail to produce meaningful results. With the introduction of objects along the corridor, both methods able to estimate the trajectory with LeGO-LOAM exhibiting drift at the bottleneck section. The estimated trajectories of both frameworks are shown in Fig. 4 along with ground truth.



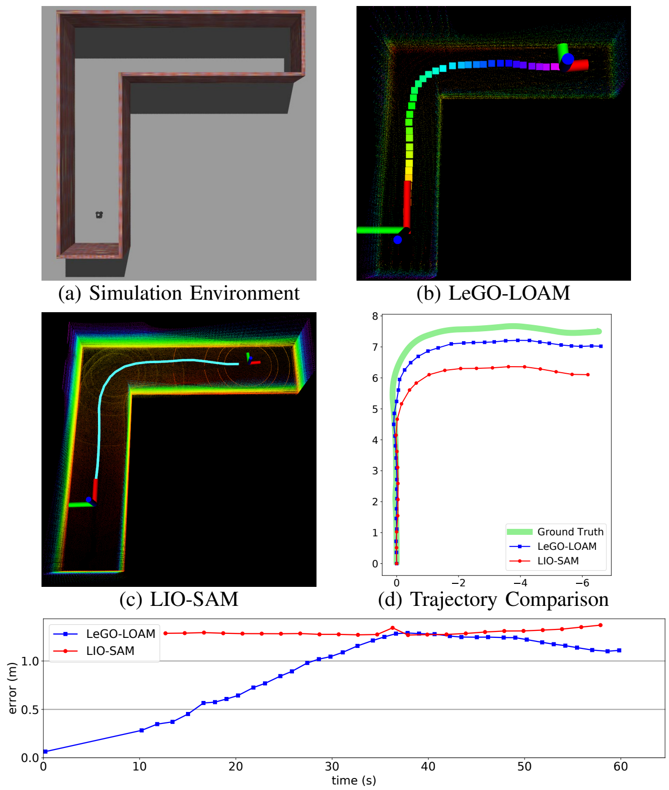
(a) Simulation Environment (b) LeGO-LOAM (c) LIO-SAM (d) Trajectory Comparison (e) Absolute Trajectory Error

Fig. 4. Simulation results for corridor with features.

C. L-Shape and Square Loop Environment

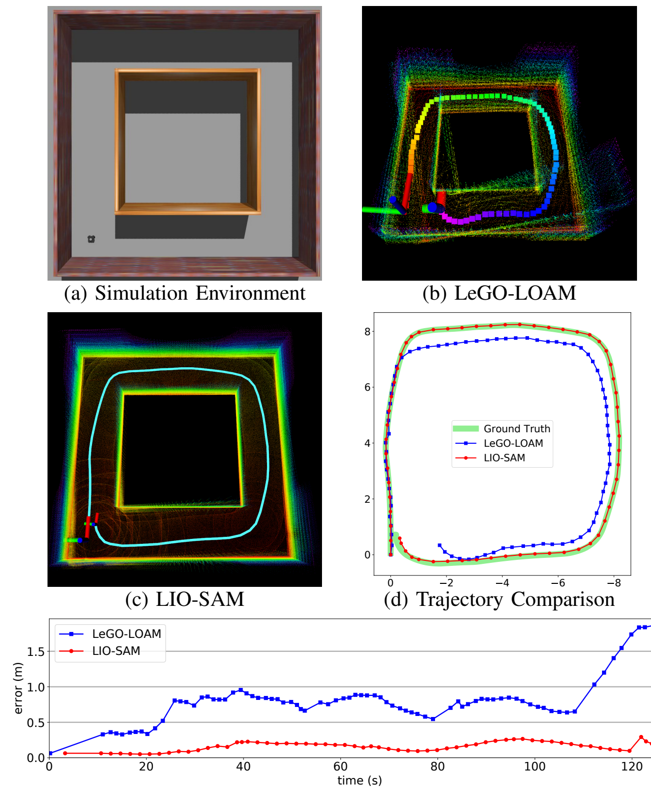
This series of simulations demonstrate the behaviour and accuracy of the frameworks in an L-shape and a closed loop environment. In featureless settings, both LeGO-LOAM and LIO-SAM struggle to estimate an accurate trajectory. For L-shaped environment, LIO-SAM suffered from poor LiDAR odometry estimation. Once the robot starts moving, the mapping process becomes stable (represented by consistent absolute trajectory errors) as the IMU preintegration factor improves the pose estimation. LeGO-LOAM suffered the most during the closed loop simulation scenario. The drift is most severe at the middle of each side where distinctive features

[1]https://youtu.be/aH4dnWnwABs



(a) Simulation Environment (b) LeGO-LOAM
(c) LIO-SAM (d) Trajectory Comparison
(e) Absolute Trajectory Error

Fig. 5. Simulation results for L-shape layout.



(a) Simulation Environment (b) LeGO-LOAM
(c) LIO-SAM (d) Trajectory Comparison
(e) Absolute Trajectory Error

Fig. 6. Simulation results for closed square loop.

TABLE I
ROOT MEAN SQUARE ERROR (METER)

Environments	**LeGO-LOAM**	**LIO-SAM**
Square room	0.667	0.089
Square with features	0.010	0.072
Corridor with features	1.144	0.147
L-shape	1.291	0.599
Closed loop	0.861	0.171
Small warehouse	0.121	0.074
Average	**0.682**	**0.192**

of other sides are blocked. The estimated trajectories of both frameworks are shown in Figs. 5-6 along with ground truths.

D. Small Warehouse Environment

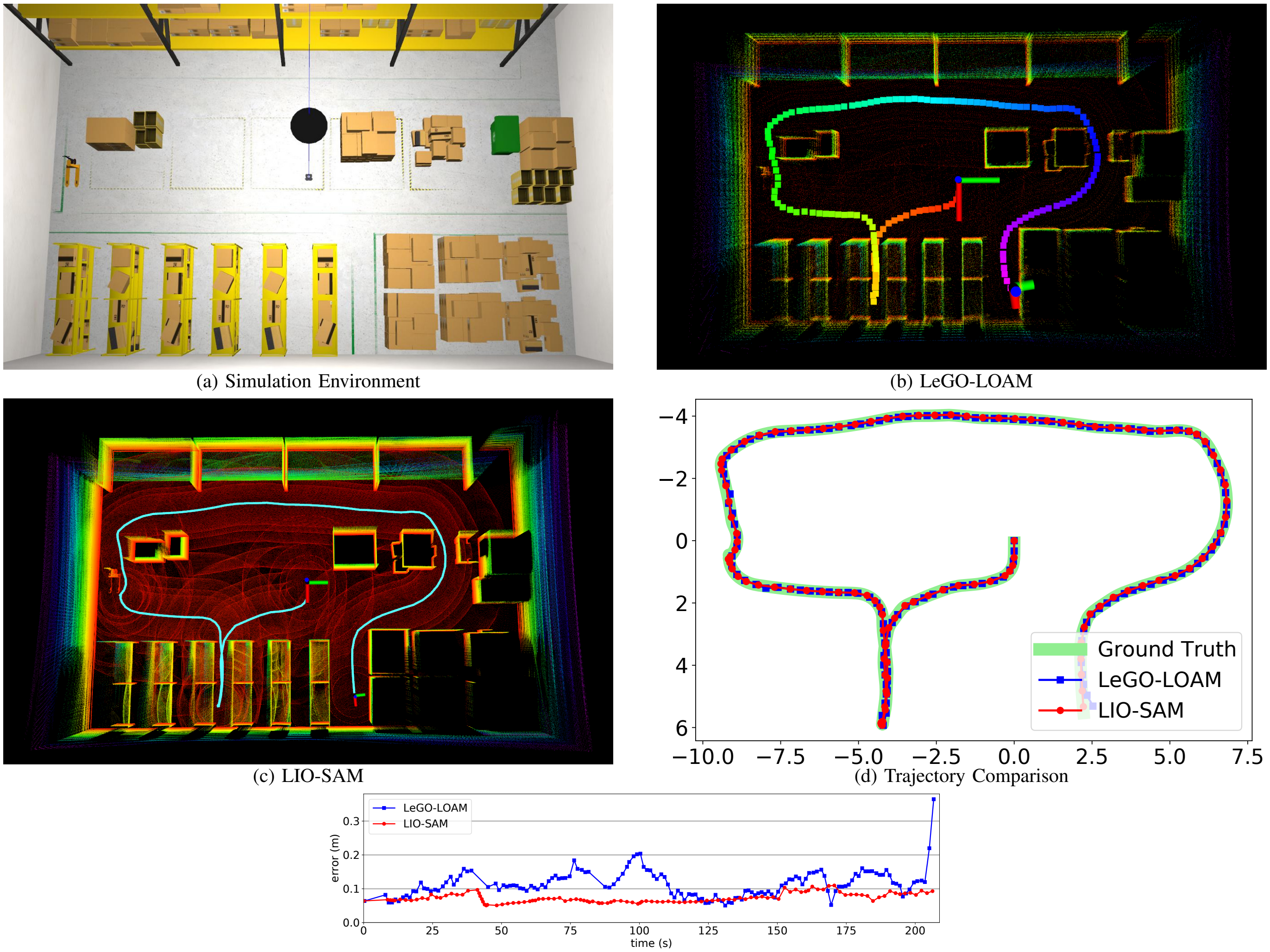
This Gazebo world model is created by AWS Robotics and is freely available for download on GitHub. This simulation demonstrates the accuracy of the frameworks for mobile robots in a complex warehouse environment. At such a feature-rich environment, both LIO-SAM and LeGO-LOAM are able to perform seamlessly and accurately. The estimated trajectories of both frameworks are shown in Fig. 7 along with ground truth.

E. Trajectory Root Mean Square Error

The performance of the SLAM frameworks were measured by the root mean square error of the estimated trajectories for each of the above scenarios. The root mean square values are tabulated in Table I. Overall, LIO-SAM demonstrated higher accuracy as compared to LeGO-LOAM for indoor environment mapping.

V. CONCLUSION

This work evaluates the performance of state-of-the-art 3D approaches for SLAM in indoor environments. Typical SLAM applications, such as autonomous driving, focus on feature-rich outdoor environments. However, most indoor scenarios lack features which pose a challenge for 3D SLAM methods. With the development of a modular simulation framework using ROS and Gazebo in this work, we have shown that the lack of features can lead to significant drift even in state-of-the-art LiDAR odometry and mapping (LOAM) methods. The evaluation focused on different environment topologies and interior arrangements, such as furniture and shelves. The additional clutter in indoor settings has been shown for a representative warehouse environment to result in very accurate 3D point clouds. However, this work has also demonstrated that scanning of empty interiors, usually found in surveying of newly built office buildings, can provide poor results using common LOAM methods for 3D mapping. Using the simulation environment developed in this work provides a framework for benchmarking SLAM approaches in indoor environments and isolates detrimental performance issues specific to featureless environments. These developments will lead to improved 3D LiDAR-based SLAM methods tailored towards indoor applications.



(a) Simulation Environment

(b) LeGO-LOAM

(c) LIO-SAM

(d) Trajectory Comparison

(e) Absolute Trajectory Error

Fig. 7. Simulation results for a small warehouse.

REFERENCES

[1] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[2] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[3] Michael Kaess, Hordur Johannsson, Richard Roberts, Viorela Ila, John J Leonard, and Frank Dellaert. iSAM2: Incremental smoothing and mapping using the bayes tree. *The International Journal of Robotics Research*, 31(2):216–235, 2012.

[4] Nathan Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 3, pages 2149–2154. IEEE, 2004.

[5] Mathieu Labbé and François Michaud. RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of Field Robotics*, 36(2):416–446, 2019.

[6] Raul Mur-Artal and Juan D Tardós. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017.

[7] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. ROS: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

[8] Tixiao Shan and Brendan Englot. LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4758–4765. IEEE, 2018.

[9] Tixiao Shan, Brendan Englot, Drew Meyers, Wei Wang, Carlo Ratti, and Rus Daniela. LIO-SAM: Tightly-coupled lidar inertial odometry via smoothing and mapping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5135–5142. IEEE, 2020.

[10] Ji Zhang and Sanjiv Singh. LOAM: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*, volume 2, 2014.