RPLIDAR Mapping on the TurtleBot

Zijing Weng
Georgia Institute of Technology

Anson Chen
Georgia Institute of Technology

***Abstract*—This study compares the performance of a Kinect (depth camera) and RPLIDAR (2D LiDAR) sensor for Simultaneous Localization and Mapping (SLAM) using two state-of-the-art graph-based SLAM algorithms: Cartographer and SLAM Toolbox. The results demonstrate that RPLIDAR outperforms Kinect in both mapping and localization, offering more robustness and precision. However, Kinect can detect obstacles outside the RPLIDAR's field of view, illustrating the potential advantages of sensor fusion. A sensor fusion method was developed, though it presents challenges. This work highlights the trade-offs between the sensors and lays the groundwork for future research into sensor fusion for improved SLAM performance.**

I. Introduction

Simultaneous Localization and Mapping (SLAM) is a critical problem in robotics, where the objective is for a robot to navigate and map its environment while simultaneously estimating its own position. 2D LiDAR systems are widely employed in indoor SLAM due to their high accuracy in distance measurement, ability to generate detailed spatial maps, and 360° field of view. However, one significant drawback is their inability to detect low-lying obstacles, as 2D LiDAR sensors only scan the environment in a horizontal plane. Conversely, camera-based systems offer a broader field of view and depth information that enhances environmental perception. Nevertheless, these systems also face challenges, including sensitivity to varying lighting conditions and a restricted field of view. We conducted experiments to compare the mapping performance of the two sensors in various conditions, and proposed a method to integrate data from both a depth camera and LiDAR, with the goal of combining the strengths of both sensors while mitigating their weaknesses. For our experiment, we used a modified Turtlebot 2 with a Kinect depth camera and a RPLIDAR A1 2D LiDAR (Fig. 1).

II. Related Works

There are two main approaches to solving SLAM: filter-based and graph-based methods.

Filter-based SLAM approaches, such as Extended Kalman Filters or particle filters, aim to estimate the robot's state by continuously updating and refining measurements, minimizing uncertainty as the robot moves. These methods have been used for their simplicity and efficiency, but they struggle in larger, more complex environments where the accumulation of errors can degrade the mapping accuracy.

Graph-based SLAM, by contrast, models the problem as an optimization task over a graph, where nodes represent robot poses, and edges encode the spatial constraints between them. This allows for global optimization and has proven to deliver superior results in accuracy and scalability, as explained in the tutorial by Grisetti et al. [1]. Konolige et al. [2] further enhanced this approach with Sparse Pose Adjustment (SPA), an efficient method to handle large-scale SLAM problems in real-time, making it one of the leading optimization techniques in this domain.

Figure 1. The robot we used for experiments. It is a modified Turtlebot 2 equipped with a Jetson Nano computer running ROS2, a Kinect depth camera and a RPLIDAR A1 2D LiDAR.

Our project focuses on evaluating SLAM performance using two sensors, the Kinect RGB-D camera and the RPLIDAR laser scanner, mounted on a TurtleBot. The Kinect provides 3D depth data but is limited by its shorter range and limited field of view, while the RPLIDAR offers long-range, 360° field of view accurate 2D scans but cannot detect small, ground-level obstacles. We also seek to combine their strengths by taking data from both sensors, providing more robust mapping and localization performance.

Korkmaz et al. [3] conducted a comparative study of Kinect and RPLIDAR mapping performance using Gmapping, a filter-based SLAM algorithm. Their analysis provided qualitative insights but lacked quantitative metrics to comprehensively assess performance. Similarly, Mu et al. [4] and He et al. [5] explored the fusion of LiDAR and depth camera data, though their reliance on filter-based approaches limited the optimization potential when compared to graph-based techniques.

In our study, we aim to evaluate the performance of both sensors using graph-based SLAM frameworks, which offer enhanced accuracy through global optimization. We also propose a sensor fusion method to improve mapping results, particularly in complex indoor environments.

For our experiments, we employ two state-of-the-art graph-based SLAM algorithms. The first is Cartographer, developed by Google [6], known for its high-performance online SLAM and support for landmark-based features. The second is SLAM Toolbox by Macenski and Jambrecic [7], which is built on Sparse Pose Adjustment (SPA) and offers a versatile toolkit for SLAM in dynamic settings. The robust capabilities of both algorithms make them ideal for evaluating the Kinect and RPLIDAR on the TurtleBot platform.

At this stage, our practical implementation will focus on sensor integration, comparison and preliminary testing of the fusion results, leaving further exploration of performance improvements for future research.

III. Methodology

A. Setup

The experimental setup utilizes a modified Turtlebot 2 platform equipped with a Jetson Nano computer running ROS 2. The system integrates two primary sensors: an RPLIDAR mounted on the robot's second plate and a Kinect sensor in its original position on the chassis. To optimize performance, the RPLIDAR operates in "Boost" mode, having around 270-degree field of view (due to obstruction), a maximum range of 12 meters and a sampling rate of 8000 points per second. In contrast, the Kinect sensor only provides a 58-degree horizontal field of view and detects points up to 4 meters, with both RGB and depth image outputs.

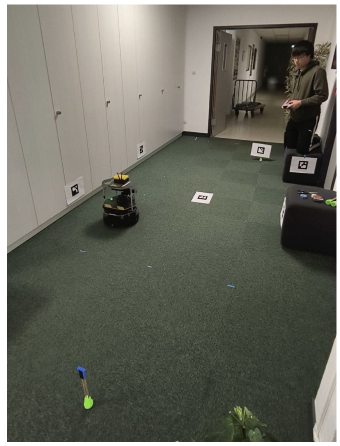
Preprocessing was necessary for data from both sensors before mapping. The Kinect's depth image was transformed into a point cloud, where only the points in the desired height range are kept. These points were then projected onto a horizontal plane, retaining only the closest points for each angle. Additionally, the Kinect's RGB image was processed using the ArUco library to detect the April tags placed in the environment, which served as landmarks for Cartographer. For the RPLIDAR, which had its field of view partially obstructed by the robot's mounting poles, we used a custom range filter package in ROS 2 to eliminate data from blocked regions.

B. Sensor Fusion

An online sensor fusion method is developed to integrate data from both sensors. The most recent measurements of both sensors are fused together by first calculating the coordinates of the points detected by Kinect in the RPLIDAR coordinate space, then interpolating the measurement by finding the closest angle measured by RPLIDAR. The closer points of the two sensors are retained in the merged data. However, this approach can have limitation when obstacles that can only be detected by Kinect move out of Kinect's field of view. The conflicting measurements from the RPLIDAR can override the accurate map of the obstacle, potentially compromising map consistency.

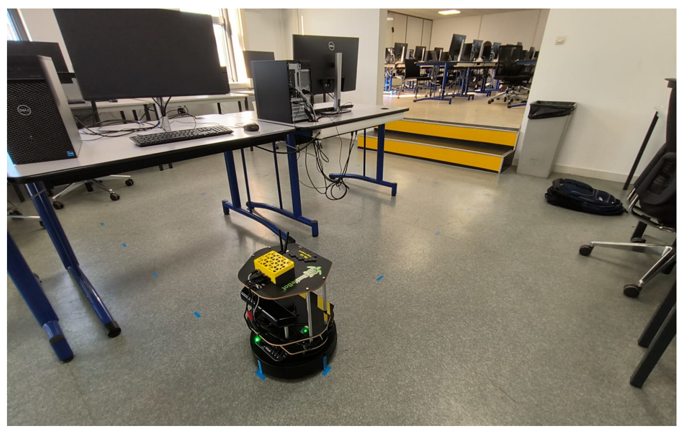
C. Experiments

Our experiments were conducted in three distinct environments (Fig. 2) designed to assess the performance of Kinect and RPLIDAR sensors, using both Cartographer and SLAM Toolbox. April tags are placed in the first two scenes as Cartographer landmarks.



(a) Rectangle

(b) Corridor



(c) Computer Lab

Figure 2. The robot testing environments

The first test, "Rectangle", took place in a compact room with several large blocks that served as easy features for mapping. The robot followed a rectangular trajectory, making a full loop plus a quarter to test the loop closure ability of graph-based SLAM. This setup allowed us to assess how well each sensor-library combination handled detailed mapping and turning in general, as well as how well our system handles a confined environment.

The second test, "Corridor", was a feature-scarce, open hallway, where the robot traveled in a straight path. This scenario posed a challenge due to the minimal presence of distinct features, which are critical for effective SLAM-based localization. The primary goal was to gauge the system's reliability and accuracy in a simplified and less informative environment, revealing potential limitations when environmental cues are sparse.

The third test, "Computer Lab", was conducted in a room filled with diverse obstacles, including low-lying objects that are undetectable by the RPLIDAR sensor. This test aimed to evaluate the robustness of the sensors in complex environments, emphasizing the advantages of sensor fusion and demonstrating the strengths and limitations of our approach.

To establish a ground truth, we pre-marked the intended driving path with tape and measured its dimensions. The robot was then driven directly along the taped route while its sensors collected data.

D. Evaluation

We recorded our experiments using ROS bags and later replayed the sensor measurements to run SLAM with different configurations on the robot. This approach ensured identical input data across all SLAM algorithms, eliminating the need to run multiple SLAM instances simultaneously, which could introduce performance issues and compromise evaluation accuracy.

For quantitative analysis, we assessed the precision of the robot's trajectories. The robot's relative position to the map was sampled at a frequency of 10 times per second. Given the absence of precise ground truth for the entire trajectory as we manually controlled the robot, we focused on key points for evaluation, calculating the squared error between the ground truth and the predicted locations of these key points. In our experiment, these key points correspond to the positions where the robot turned in place, and the final position. This method provided a more accurate assessment of sensor performance across various conditions, as well as the effectiveness of sensor fusion strategy.

IV. Results

In the first test, Rectangle, most sensor and algorithm combinations yielded accurate results in both mapping (Fig. 5) and localization (Fig. 3), with the exception of Kinect combined with SLAM Toolbox, where the system inaccurately detected a turn, leading to significant deviations in the final map and trajectory estimation. Results from Cartographer with and without tags were nearly identical, likely due to the Kinect's narrow field of view and low image resolution, which hindered reliable tag detection.

To determine error for each of the trajectories, we calculated the Root Mean Squared Error (RMSE):

$$\text{RMSE} = \sqrt{\frac{\sum(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}{N}}$$

The results (Table. I) showed that RPLIDAR provides better results in all scenarios.

In the second test (Corridor), Cartographer performed robustly in both mapping (Fig. 6) and localization (Fig. 4) with both Kinect and RPLIDAR. The generated maps matched the actual dimensions of the hallway (Table. II), and the system consistently identified the robot's position accurately.

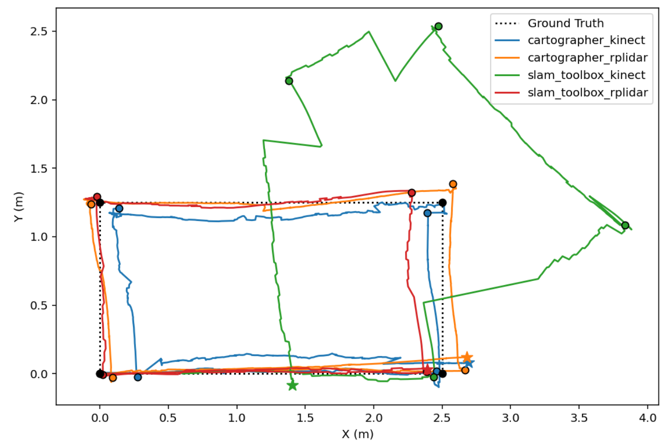


Figure 3. Trajectories calculated with different SLAM algorithms and sensor data on the "Rectangle" map. The trajectories calculated using tags are not plotted because they are nearly identical to those without using tags and they overlap on the graph. The ground truth is a rectangular path of 2.5*1.25 meters with uncertainty of 0.05 meters due to manual driving. All trajectories starts at (0,0) and follows a counterclockwise path. The highlighted dots are the localizations of the robot when it is turning at the vertices of the rectangle, and the final positions are marked with stars

"Rectangle" experiment	RMSE (m)
Cartographer with Kinect depth only	0.179
Cartographer with RPLIDAR only	0.151
Cartographer with Kinect depth and tags	0.180
Cartographer with RPLIDAR and tags	0.152
SLAM Toolbox with Kinect	1.859
SLAM Toolbox with RPLIDAR	**0.129**

Table I
ERROR OF KEY POINTS OF THE "RECTANGLE" MAP

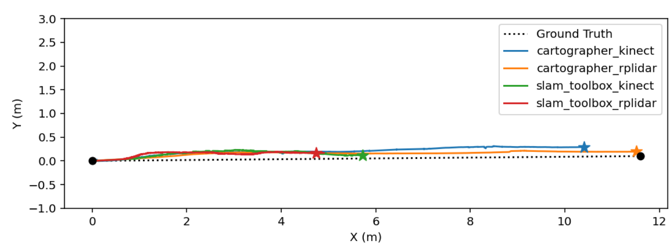
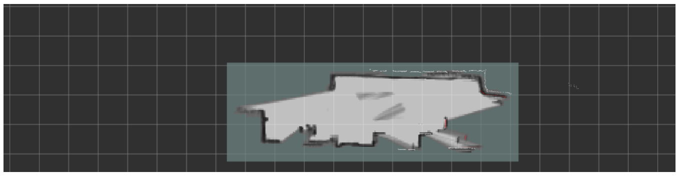


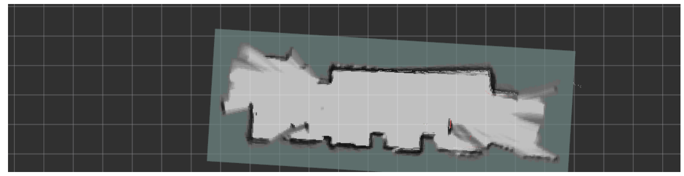
Figure 4. Trajectories calculated with different SLAM algorithms and sensor data on the "Corridor" map. The trajectories calculated using tags are not plotted because they are nearly identical to those without using tags and they overlap on the graph. The final positions are marked with stars. The robot traveled 11.6 meters with ground truth measurement that have uncertainty of 0.05 meters

"Corridor" experiment	Error (m)
Cartographer with Kinect depth only	1.214
Cartographer with RPLIDAR only	**0.126**
Cartographer with Kinect depth and tags	1.213
Cartographer with RPLIDAR and tags	**0.126**
SLAM Toolbox with Kinect	5.880
SLAM Toolbox with RPLIDAR	6.868

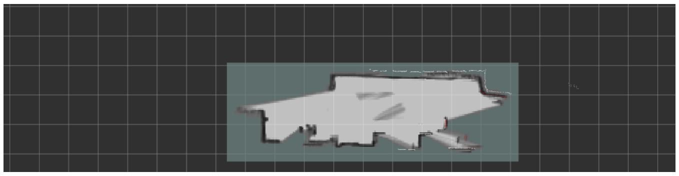
Table II
ERROR OF FINAL POSITION ON THE "CORRIDOR" MAP



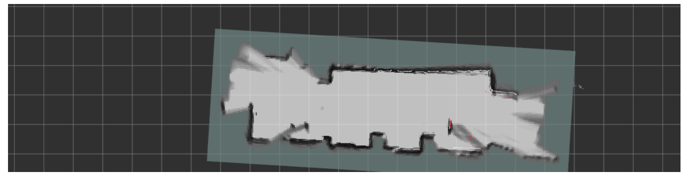
(a) Cartographer with Kinect depth only



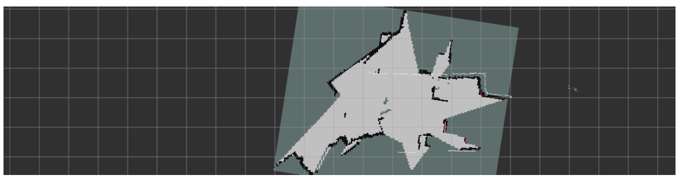
(b) Cartographer with RPLIDAR only



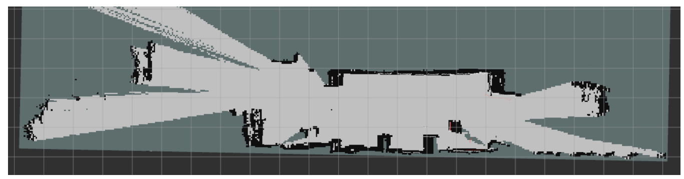
(c) Cartographer with Kinect depth and tags



(d) Cartographer with RPLIDAR and tags

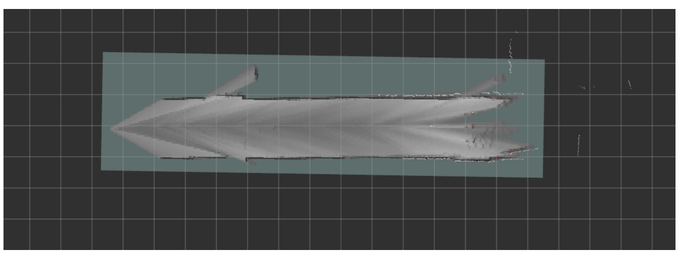


(e) SLAM Toolbox with Kinect

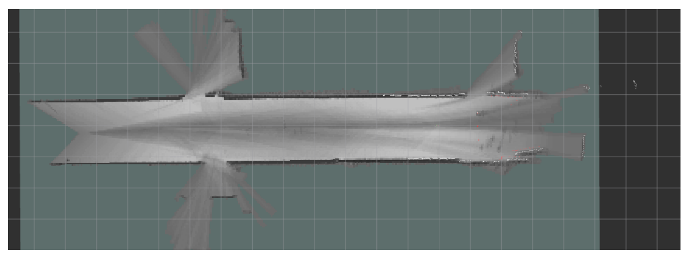


(f) SLAM Toolbox with RPLIDAR

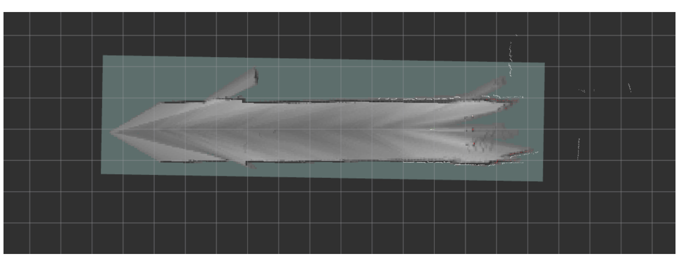
Figure 5. Maps generated by the SLAM algorithms in the first experiment ”Rectangle”. Black = occupied, white = unoccupied, gray = unknown



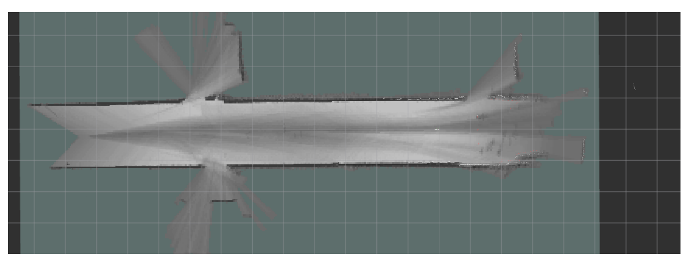
(a) Cartographer with Kinect depth only



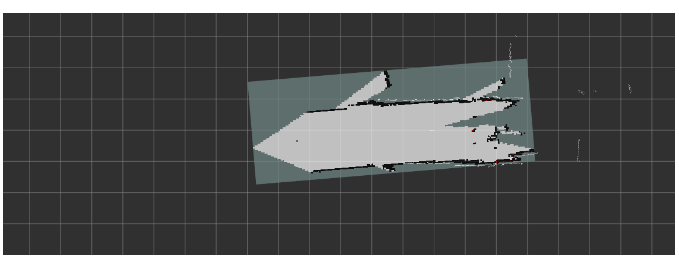
(b) Cartographer with RPLIDAR only



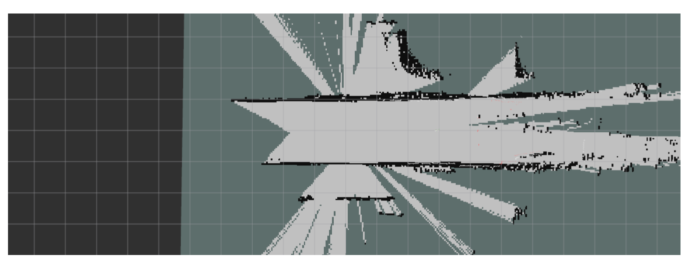
(c) Cartographer with Kinect depth and tags



(d) Cartographer with RPLIDAR and tags



(e) SLAM Toolbox with Kinect



(f) SLAM Toolbox with RPLIDAR

Figure 6. Maps generated by the SLAM algorithms in the second experiment ”Corridor”. Black = occupied, white = unoccupied, gray = unknown

However, SLAM Toolbox struggled in this feature-poor environment, producing maps significantly shorter than the hallway, indicating a considerable underestimation of traveled distance. This resulted in localization errors, as the robot's perceived position did not align with its true path. Across both environments, RPLIDAR outperformed Kinect, primarily due to its longer range and higher sensitivity to environmental features.

In the first two experiments, we only used a slice of Kinect data corresponding to the height detected by the RPLIDAR. This ensured mapping on the same plane, allowing for a fair comparison between the two sensors. However, for the "Computer Lab" test, we sought to illustrate scenarios where the RPLIDAR fails to detect certain obstacles due to its fixed height. In this test, we utilized all Kinect data points above the ground plane and below robot height, allowing the sensor to detect obstacles as long as they were within its field of view.

The mapping results (Fig. 7) clearly illustrate the advantages of the Kinect sensor in detecting a wider range of obstacles. Features such as the window on the left, stairs on the right, numerous table and chair legs, and even a dangling cable were all marked as non-traversable in the Kinect map but were completely missed by the RPLIDAR. The fusion data captured these features as well, although they are not very clear in

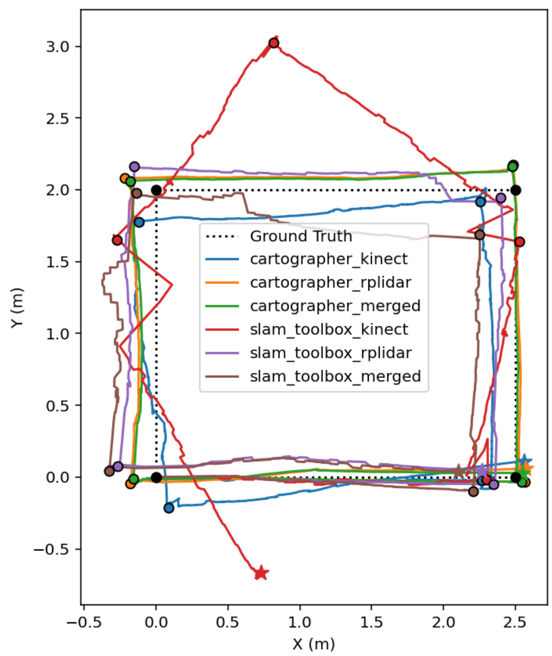
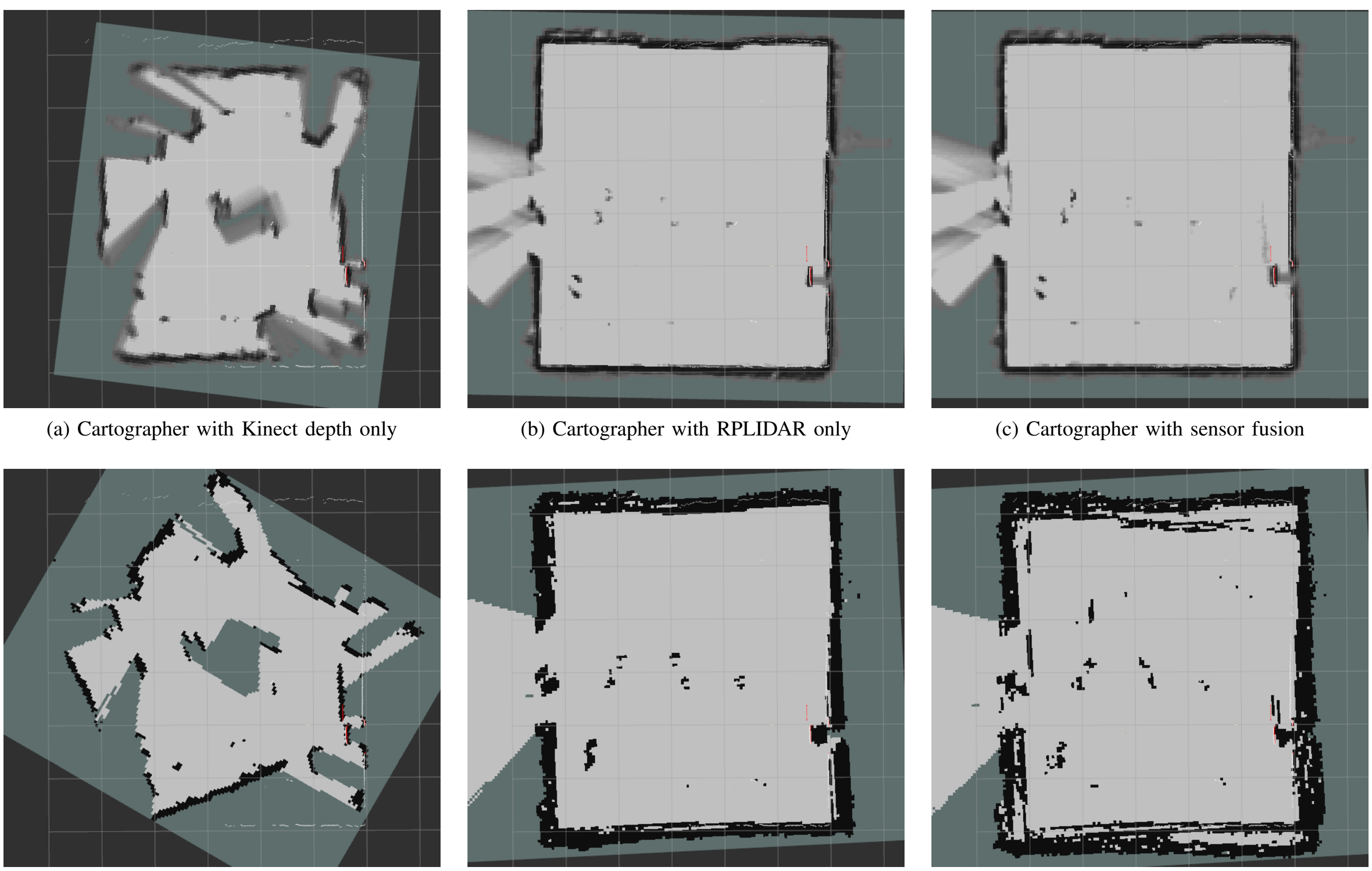


Figure 8. Trajectories calculated with different SLAM algorithms and sensor data on the "Computer Lab" map. The ground truth is a rectangular path of 2.5*2 meters with uncertainty of 0.05 meters



(a) Cartographer with Kinect depth only

(b) Cartographer with RPLIDAR only

(c) Cartographer with sensor fusion

(d) SLAM Toolbox with Kinect

(e) SLAM Toolbox with RPLIDAR

(f) SLAM Toolbox with sensor fusion

Figure 7. Maps generated by the SLAM algorithms in the third experiment "Computer Lab". Black = occupied, white = unoccupied, gray = unknown

the final map because RPLIDAR data gradually erased these obstacles as they moved out of the Kinect's field of view.

The trajectories (Fig.8) and errors (TableIII) were calculated using the same method as in the first experiment. The results were mixed: while Cartographer achieved similar localization precision using sensor fusion and RPLIDAR alone, SLAM Toolbox's precision was compromised by the higher uncertainty of Kinect measurements and the latency introduced by the additional processing required for the Kinect data.

"Computer Lab" experiment	RMSE (m)
Cartographer with Kinect depth only	0.225
Cartographer with RPLIDAR only	0.163
Cartographer with sensor fusion	**0.139**
SLAM Toolbox with Kinect	1.286
SLAM Toolbox with RPLIDAR	0.210
SLAM Toolbox with sensor fusion	0.329

Table III

ERROR OF KEY POINTS OF THE "COMPUTER LAB" MAP

V. DISCUSSION AND FUTURE WORKS

The experiments demonstrated that RPLIDAR consistently provides more robust and accurate mapping results than Kinect, making it a valuable upgrade for the Turtlebot 2 in this specific task. However, its inability to perceive anything outside a 2D plane limits its effectiveness in detecting obstacles or performing advanced tasks like autonomous navigation. This limitation underscores the potential value of sensor fusion.

This research faced several limitations due to time and equipment constraints, leaving room for further exploration in sensor performance and fusion strategies.

One major limitation was the inability to calibrate the Kinect's intrinsic parameters accurately. We relied on the default calibration, which introduced additional errors compared to a properly calibrated setup [8]. A more precise calibration could be achieved through stereo calibration using a large checkerboard pattern and an IR emitter [9].

Additionally, evaluating the error over the entire trajectory requires a more accurate ground truth. A vision-based system could provide precise pose tracking of the robot in an indoor environment, enhancing the accuracy of future comparisons.

Further tuning of SLAM algorithm parameters is also necessary to optimize results. Future studies could benefit from a comprehensive comparison of state-of-the-art filter-based and graph-based SLAM algorithms across varied environments and sensor configurations.

More work is needed to develop a robust sensor fusion solution. One idea involves using the lifelong mapping feature of SLAM Toolbox: creating an accurate base map using RPLIDAR and then augmenting it with additional features from Kinect data. Alternatively, local maps from Kinect data could be used for obstacle avoidance. However, this approach haven't been tested due to the time constraint of this research. More sophisticated sensor fusion methods, such as modifying SLAM algorithms to accommodate data from multiple sensors, also present an interesting direction for future research.

VI. CONCLUSION

In this study, we evaluated the performance of Kinect and RPLIDAR sensors using two graph-based SLAM algorithms: Cartographer and SLAM Toolbox. The experiments demonstrated that the RPLIDAR generally outperformed the Kinect sensor in mapping and localization due to its field of view, longer range, and higher precision in distance measurement. Cartographer emerged as the more robust algorithm, especially in feature-scarce environments like the "Corridor" test, accurately mapping and localizing the robot even with minimal environmental features. In contrast, SLAM Toolbox encountered difficulties, particularly with the Kinect sensor, leading to significant errors in both trajectory estimation and mapping. Future research could focus on improving sensor calibration and exploring advanced fusion strategies to further improve SLAM accuracy and robustness.

REFERENCES

[1] G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based slam," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, 2010.

[2] K. Konolige, G. Grisetti, R. Kümmerle, W. Burgard, B. Limketkai, and R. Vincent, "Efficient sparse pose adjustment for 2d mapping," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 22–29.

[3] M. Korkmaz, A. Durdu, and Y. Tusun, "Sensor comparison for a real-time slam application," *International Journal of Information and Electronics Engineering*, vol. 8, no. 1, pp. 1–4, 2018.

[4] L. Mu, P. Yao, Y. Zheng, K. Chen, F. Wang, and N. Qi, "Research on slam algorithm of mobile robot based on the fusion of 2d lidar and depth camera," *IEEE Access*, vol. 8, pp. 157 628–157 642, 2020.

[5] J. He, J. Fang, S. Xu, and D. Yang, "Indoor robot slam with multi-sensor fusion," *International Journal of Advanced Network, Monitoring and Controls*, vol. 9, pp. 10–21, 03 2024.

[6] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1271–1278.

[7] S. Macenski and I. Jambrecic, "Slam toolbox: Slam for the dynamic world," *Journal of Open Source Software*, vol. 6, no. 61, p. 2783, 2021. [Online]. Available: https://doi.org/10.21105/joss.02783

[8] J. Smisek, M. Jancosek, and T. Pajdla, "3d with kinect," *Consumer depth cameras for computer vision: Research topics and applications*, pp. 3–25, 2013.

[9] M. Maghoumi. (2017) Practical kinect stereo calibration for the highest accuracy. [Online]. Available: https://www.codefull.net/2017/04/practical-kinect-calibration/