Mapping performance comparison of 2D SLAM algorithms based on different sensor combinations

Pengtao Qu1, Chen Su1, Hang Wu1, Xinxu Xu1, Sheng Gao1 and Xiuguo Zhao1*

1Institute of Medical Support, Institute of Systems Engineering, Academy of Military Sciences, Tianjin, 300161, China
*Corresponding author’s e-mail: michaelzhao@tjyysbxjs.onexmail.com

Abstract. Currently, some existing SLAM algorithms were widely applied in the field of mobile robots. However, most SLAM algorithms were tested either by simulation or using offline bags storing messages, which ignored other contributing factors influencing mapping including the changes of position and speed of the robot during it autonomous movement. The aim of this study is to investigate the solution of constructing more accurate map with high quality sensors based on freely available SLAM algorithms. Comparisons and analyses of three 2D SLAM algorithms (i.e. Gmapping, Hector SLAM and Cartographer) available in ROS were conducted to map different environments in our work using different sensor combinations of a LIDAR, a Stereo camera and an IMU, respectively. The research showed that the Cartographer algorithm has advantage over two other algorithms on constructing well-defined and informative environmental maps with minimal errors. Also, the LIDAR scan is more suitable than the ZED scan as the input of mapping algorithms, and the Cartographer combining the LIDAR scan, ZED odometry and IMU is the best solution to map in all combinations.

1. Introduction

In recent years, mobile robots have increasingly become a hot topic of most interest to researchers. The wide application of mobile robots, such as including space exploration, military missions, agricultural automation, intelligent handling and so on, has accelerated the development of society and has made great contributions to the welfare of modern society [1]. As is known, the process when a robot acquires an environment map while simultaneously localizing itself relatively to this map is called SLAM (Simultaneous Localization And Mapping). The solution for SLAM is considered to be an effective way to make a robot truly autonomous [2]. At present, various forms of SLAM algorithms were developed in terms of sensor types, mathematical solutions, and working environment. Therefore, it is a great challenge to identify an effective and reasonable SLAM solution from the available methods for given unknown environment [3].

In [4], a wheeled mobile robot based on Arduino was designed. Two classic algorithms including Gmapping and Hector SLAM were used for mapping and navigation comparison. In [5], a study of several 2D slam techniques available was conducted based on the algorithms Cartographer, Karto SLAM and Gmapping respectively. Among them, Cartographer has the minimal error for both a simple scene with missing features and a large complex scene. In [6], they used some SLAM techniques available in ROS to make 2D maps with different operations. The results showed that the best performance was obtained by Gmapping while Hector SLAM failed due to low frequency and small FOV(Field Of View) of Kinect Depth Camera. In [7], the experiment showed the laser-based Hector SLAM is slightly better than Gmapping.
From above studies, it can be found that different SLAM algorithms have different performance under different configurations. However, most studies were still based on simulation and offline packages, which ignore the real factors for robot’s working caused by sudden movements or speed changes of robots. In this paper we focus on the algorithmic mapping performance of constructing physical space, and discuss the mapping differences between different SLAM techniques using different sensors. The purpose is to find out the better SLAM mapping solution based on the existing methods in a practical situation and to provide a useful reference for researchers who adopt different sensor solutions.

2. Materials and Methods
This section mainly described the preparation and application methods of the experiment in our work.

2.1. Robot and System
The robot used in research was shown in figure 1. The robot was equipped with a ZED stereo camera, a LIDAR sensor and a screen for visual adjustment. In addition, the omni-directional Meachum structure allowed the operators in the process of experiment flexibly and randomly to control the robot position and orientation. The experiment was performed in NVIDIA-XAVIER embedded processor which had 384 GPU cores and 8GB of RAM, and Ubuntu 18.04 installed with ROS Melodic.

![Figure 1](image1.png)

Figure 1. The robot equipped with LIDAR, ZED, screen and encoder.

2.2. Sensor types
Both the ZED camera and Mid-40 LIDAR, as the 3D device, were used to collect 3D point clouds in experiments by tuning each wrapper package in ROS. In order to satisfy the input forms specified by algorithms[8], it was necessary to slice the point cloud data and then convert it to laser scan data. Fortunately, this process could be quickly implemented by the `pointcloud_to_laserscan` package in ROS. As a ROS node, it converted a 3D Point Cloud into a 2D laser scan which is useful for making devices like ZED appear like a laser scanner for 2D-based algorithms. In ROS, once the parameters such as `min_height`, `max_height`, `angle_min`, `angle_max`, `target_frame` and `concurrency_level` were configured, the node can continuously and steadily subscribed to the `/cloud_in(sensor_msgs/PointCloud2)` topic and published the `/scan(sensor_msgs/LaserScan)` topic. The FOV(field of view) of ZED scan and LIDAR scan and two scan displayed in RVIZ were showed in figure 2.

![Figure 2](image2.png)
2.3. Algorithms
This section briefly described three 2D SLAM algorithms: Gmapping, Hector SLAM and Cartographer, respectively.

Gmapping was a RBPF-based SLAM method proposed by the literature [9], which solved two thorny problems of particle filter. First, it presented adaptive techniques for reducing number of particles in a Rao-Blackwellized particle filter for learning grid maps. Then, it proposed the selective resampling operation to reduce the seriously problem of particle depletion.

Hector SLAM was a system for fast online learning of occupancy grid maps requiring low computational resources [10]. It combined a robust scan matching approach using a LIDAR system with a 3D attitude estimation system based on inertial sensing. This algorithm did not use any odometry information, which lacked the loop detection, resulting in the sensitivity to rotational movement.

Cartographer was a system that provided real-time simultaneous localization and mapping in 2D and 3D across multiple platforms and sensor configurations [11]. The most significant contribution of it was to use the scan_to_submap matching idea to find out the exact pose of the scan, which reduced the cumulative error of the long-term iterative pose estimation. The accurate mapping performance was achieved by loop detection for the multiple submaps and scans constructed with the help of brand and bound acceleration strategy.

3. Experiments & Results
3.1. Comparison of two odometry
Before mapping, the experiments on pose estimation using different odometers were carried out in robot, which had the aim of not only selecting the odometry with less noise and more accurate pose data based on the existing odometer model (Wheel odometry and ZED odometry), but also improving the performance of the algorithm to estimate the exact pose of the robot. The cumulative drift error of each odometry was calculated by collecting the output of each /odom topic in ROS, while the robot controlled by remote control moved around the room in a series of experiments. The trajectory and the cumulative trajectory error generated by each odometry was showed in figure 3.
Figure 3. Trajectories error for each odometry.
(a) The cumulative trajectory error caused by Wheel odometry; (b) The cumulative trajectory error caused by ZED odometry; (c) The cumulative trajectories measured by wheel and ZED odometry in experiment.

From the above figures, it could be seen that the cumulative trajectory errors caused by both odometry were concentrated. The mean cumulative error of the wheel odometry in both X and Y directions were greater than 1m, while that of ZED odometry in both directions were less than 0.05m. During experiment, the robot trajectory calculated by the ZED odometry matched the actual robot pose with a cumulative error less than 0.01m, and that by the Wheel odometry had a severely drifts with a cumulative error greater than 0.5m. The mechanism of the Wheel odometry to calculate the robot position was obtained by integrating the speed against time, which was easy to cause errors. The ZED odometry calculated the position increments by matching adjacent images, so it was independent of drive structure and ground conditions. Also, the robot trajectories calculated from two odometers demonstrated the ZED odometry had the better mapping performance of compared to the Wheel odometry due to excellent loop detection of the ZED odometry.

3.2. Mapping with different algorithms using ZED scan
In figure 4, the snapshot of experiment scenario was displayed, namely Room and Corridor. The Room was a single room with various features such as desks, chairs and so on. The Corridor was a long corridor with many similar doors but lacking rich features. Both scenarios allowed the robot to move freely.
All algorithms were built and run in ROS, and the data obtained in the experiment were collected under the default configuration of the algorithms. In addition, the trajectory of robot in the Room was different from that in the Corridor, which depended on the space of the scenario and the FOV of sensor used. For example, the robot was easy to perceive all information of the Corridor only once round trip due to the narrow of the Corridor and the wide FOV of ZED scan. The key purpose of testing two scenarios with big difference was to identify the adaptability of each algorithm facing distinctly different levels of complexity of the environment. In the experiment, the robot speed was approximately 1m/s, and the angular velocity was about 0.6 radians. The maps obtained by each algorithm in the Room in live time was showed in figures 5.

As shown from the above figures, the maps created by the Gmapping and Cartographer algorithms more closely represented the scenario of Room than the Hector SLAM algorithm. The severe overlaps and drifts of the map generated by Hector SLAM may resulted from the use of inappropriate sensor and algorithmic extreme sensitivity to rotation motion. The phenomenon could be explained by literature[10], in which the laser scan frequency was close to 40HZ, much higher than the ZED scan frequency in above experiments.

The maps built by each algorithm in the Corridor were shown in figure 6. The figures illustrated that the map built by Hector SLAM had more severe drifts and distortions in the Corridor than in the Room, which showed the Hector SLAM algorithm was impossible to generate the realistic maps with the main reason it was prone to losing the robot pose due to low frequency sensors and sensitivity to rotational movements.
Figure 6. Maps generated in Corridor in live time. (a) Gmapping; (b) Hector SLAM; (c) Cartographer.

When it comes to the Gmapping, the mapping performance in Corridor by it is slightly less capable than in Room. The poor performance of Gmapping facing larger scenarios was potentially influenced by the mechanism of algorithm such as selecting the optimal particle poses at current time, ignoring the valid information from historical moments, which resulting in the algorithm not being able to obtain the global optimal solution with limited number of particles.

In spite of the extension and complexity of the environments, Cartographer was still able to generate approximately globally consistent map of the Corridor due to the Cartographer combining correlative scan matching with sparse pose adjustment, as verified based on well-defined contours of map featuring the corridors.

During the experiments, the comparison of length of the maps provided by Gmapping, Hector SLAM and Cartographer relative to ground truth was presented in table 2. The result showed that the Cartographer has the minimal mapping errors in all experiments, which suggested that the Cartographer algorithm has advantage over both Gmapping and Hector SLAM for 2D SLAM solutions at present.

Table 2. Map comparison: error of the length for SLAM methods relative to the ground truth.

| Environments | Algorithms      | Measurement(m) | Truth(m) | Error(m) |
|--------------|----------------|----------------|----------|----------|
| Room         | Gmapping       | 5.52           | 5.90     | 0.38     |
|              | Hector SLAM    | Invalid        |          |          |
|              | Cartographer   | 5.65           | 5.90     | 0.25     |
|              | Gmapping       | 61.28          | 67.98    | 6.70     |
| Corridor     | Hector SLAM    | Invalid        |          |          |
|              | Cartographer   | 67.40          | 67.98    | 0.58     |

3.3. Mapping with different sensors under Cartographer

The type of sensor largely affected the performance of SLAM algorithm. From above discussions, the Cartographer was the better solution for 2D mapping than other algorithms. In this section, the mapping performance of Cartographer employing different sensor combinations were compared. The aim was on the one hand to determine the optimal Cartographer-based sensor combinations for mapping and on the other hand to fully understand the map-built differences between the ZED scan and LIDAR scan.

The maps generated by six different sensor combinations were shown in figure 7. From the perspective of visualization, each map could be closely matched to the real environment. The severe drifts and blurring of walls appeared in maps generated by single sensor-based algorithm, which was aggravated in the ZED scan-based algorithm. Fortunately, with the addition of the odometry and IMU, the maps created by the algorithm had significantly reduced drifts and blurring of the map.
Figure 7. Comparisons of map constructed from ZED scan(down) and LIDAR scan(top).
(a)LIDAR scan; (b)LIDAR scan + ZED odom; (c) LIDAR scan + ZED odom + IMU;
(d)ZED scan; (e)ZED scan + ZED odom; (f)ZED scan + ZED odom + IMU.

As is known, the quality of the map depends heavily on the initial pose of AGV. The odometry was recognized as one of the methods that could provide high precision pose estimation in a short period of time. Similarly, the accurate orientation of AGV could also be provided by IMU. Therefore, the maps built by algorithms combining the odometry and IMU are superior to those built by single sensor-based algorithms.

An intuitive observation of above maps did not help us determine which was the best choice. Next, the lengths of maps generated above were also compared with the ground truth, and the metric, calculating the proportion of occupied cells in map as the sum of occupied cells divided by number of free cells (see, equation 1), as described in the paper[3] were used to compare the quality of above maps. Generally, the proportion of occupied cells correspond to the quality of the map: the higher this proportion – the lower quality of the map. And it should be noted that the metric calculating the proportion of occupied cells should be used only as a part of a complex analysis.

\[
\text{prop(occupied)} = \frac{\text{Number(occupied\ cells)}}{\text{Number(free\ cells)}}
\]

The experimental results were classified by different sensor combinations applied, which were presented in table 3. The algorithm combing the LIDAR, ZED odometry and IMU showed the best mapping performance in all combinations with a minimum cumulative mapping error of 0.074. The mapping performance of the algorithm using LIDAR combinations with mean cumulative error around 0.11 was better than the algorithm using ZED combinations mean cumulative error with around 0.32. Next, from the last column in table 3, it could be seen that the mean proportion of occupied cells of ZED combinations was almost twice that of LIDAR combinations, implying that the maps generated by algorithms based on LIDAR combinations had little blurry effect.

Also, it was easily found that the algorithm combining LIDAR scan, ZED odometry and IMU had advantage over other combinations. Moreover, both ZED odometry and IMU played a very important role in improving mapping performance in Cartographer only based on single scan(LIDAR scan or ZED scan).
scan) due to the accurate position provided by the odometry and the orientation information provided by IMU. The advantage of Cartographer based on LIDAR combinations largely depended on the small measurement variance of LIDAR.

| Combinations                  | Measure(m) | Truth(m) | Error(m) | Proportion(%) |
|-------------------------------|------------|----------|----------|---------------|
| LIDAR combinations            |            |          |          |               |
| LIDAR scan                    | 5.722      | 5.901    | 0.179    | 13.6          |
| LIDAR scan + ZED odom         | 5.808      | 5.901    | 0.093    | 10.0          |
| LIDAR scan + ZED odom + IMU   | 5.827      | 5.901    | 0.074    | 10.7          |
| ZED scan                      | 5.405      | 5.901    | 0.496    | 19.9          |
| ZED scan + ZED odom           | 5.655      | 5.901    | 0.246    | 19.6          |
| ZED scan + ZED odom + IMU     | 5.667      | 5.901    | 0.234    | 20.4          |

4. Conclusions
This work provides insights into the mapping performance of the three representative SLAM algorithms: Gmapping, Hector SLAM and Cartographer. The difference in the quality of the map constructed based on different sensor combinations is investigated in detail. Through the above discussions, the conclusions are drawn as following: 1) Compared to the Gmapping and the Hector SLAM, the Cartographer showed the better robustness in different environments and the maps constructed by Cartographer were the most accurate. 2) The mapping performance of LIDAR scan-based Cartographer is preferred to that of ZED scan-based Cartographer and the Cartographer combining the LIDAR scan, ZED odometry and IMU outperformed other combinations in mapping. 3) The algorithms with addition of an odometry or IMU improves the mapping performance of algorithms based on single sensor. Another thing is to note is that, the ZED scan-based Cartographer generates the maps with the approximate global consistency and low error drifts, indicating that the camera-based laser emulator can also be useful in specific environments.

The comparison of different SLAM algorithms is really a challenging task because of the performance of a SLAM technique greatly influenced by the parameter settings, the complexity of environment and the sensor type. Nevertheless, this work still give advice on the choice of an appropriate SLAM approach for the intended application. Based on the results of this paper, the future work should focus on the study on improving the map quality by deeply coupled data and advancing the three algorithms in unknown scenarios.

References
[1] Tzaferstas, S. G. (2018) Mobile Robot Control and Navigation: A Global Overview. J. Journal of Intelligent & Robotic Systems. 91: 35-58.
[2] Durrant-Whyte, H.Balley, T. (2006) Simultaneous localization and mapping: Part I. J. IEEE Robotics Automation Magazine. 13: 99-110.
[3] Filatov, A., Filatov, A., Krinkin, K., Chen, B.Molodan, D. (2017) 2D SLAM quality evaluation methods. In:IEEE. Helsinki.pp. 120-126.
[4] Cheng, Y.Wang, G. Y. (2018) Mobile robot navigation based on lidar. In:2018 Chinese Control And Decision Conference (CCDC). Shenyang.pp. 1243-1246.
[5] Fan, X., Wang, Y.Zhang, Z. (2021) An evaluation of Lidar-based 2D SLAM techniques with an exploration mode. J. Journal of Physics: Conference Series. 1905: 1-7.
[6] Rojas-Fernandez, M., Mujica-Vargas, D., Matuz-Cruz, M.Lopez-Borreguero, D. (2018) Performance comparison of 2D SLAM techniques available in ROS using a differential drive robot. In:2018 International Conference on Electronics, Communications and Computers (CONIELECOMP). Cholula.pp. 50-58.
[7] Kamarudin, K., Manduh, S. M., Yeon, A. S. A., Visvanathan, R., Shakaff, A. Y. M., Zakaria, A., Kamarudin, L. M.Rahim, N. A. (2015) Improving performance of 2D SLAM methods by
complementing Kinect with laser scanner. In: IEEE International Symposium on Robotics & Intelligent Sensors. Langkawi. pp. 278-283.

[8] Joseph, L. (2015) Mastering ROS for Robotics Programming. Packt Publishing, Birmingham.

[9] Grisetti, G., Stachniss, C., Burgard, W. (2007) Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. J. IEEE Transactions on Robotics. 23: 34-46.

[10] Kohlbrecher, S., Stryk, O. V., Meyer, J., Klingauf, U. (2011) A flexible and scalable SLAM system with full 3D motion estimation. In: IEEE International Symposium on Safety. Kyoto. pp. 155-160.

[11] Hess, W., Kohler, D., Rapp, H., Andor, D. (2016) Real-time loop closure in 2D LIDAR SLAM. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). Stockholm. pp. 1271-1278.