Article

Not peer-reviewed version

Research on a SLAM Method Based on Semantic Information

Ruo-huai SUN * , Xue ZHAO , Zhen-jun DU , Cheng-dong WU , Bin ZHAO *

Posted Date: 7 February 2024

doi: 10.20944/preprints202402.0437.v1

Keywords: SLAM; Semantic laser; Point cloud; Occupation probability

Article

Research on a SLAM Method Based on Semantic Information

Ruo-huai SUN [1,2,3], Xue ZHAO [4], Zhen-jun Du [2,3], Cheng-dong WU [1,3] and Bin ZHAO [1,2,3,*]

[1] College of Information Science and Engineering, Northeastern University, Shenyang 110819,China; sunruohuai@stumail.neu.edu.cn
[2] SIASUN Robot & Automation Co., Ltd., Shenyang, 110168, China; sunruohuai@siasun.com
[3] Faculty of Robot Science and Engineering, Northeastern University, Shenyang 110169, China; zhaobin@stumail.neu.edu.com.cn
[4] Daniel L. Goodwin College of Business, Benedict University, Chicago, IL 60601, USA; 20203734@stu.hebmu.edu.cn
* Correspondence: zhaobin@stumail.neu.edu.cn; Tel.: +8615998874918

Abstract: This paper proposes a solution for the problem of mobile robot navigation and trajectory interpolation in dynamic environments with large scenes. The solution combines a semantic laser SLAM system that utilizes deep learning and a trajectory interpolation algorithm. The paper first introduces some open-source laser SLAM algorithms and then elaborates in detail on the general framework of the SLAM system used in this paper. Secondly, the concept of voxel is introduced into the occupation probability map to enhance the ability of local voxel maps to represent dynamic objects. Then, in this paper, we propose a PointNet++ point cloud semantic segmentation network combined with deep learning algorithms to extract deep features of dynamic point clouds in large scenes and output semantic information of points on static objects. A descriptor of the global environment is generated based on its semantic information. Closed-loop completion of global map optimization is performed to reduce cumulative error. Finally, T-trajectory interpolation is utilized to ensure the motion performance of the robot and improve the smooth stability of the robot trajectory. The experimental results indicate that the combination of the semantic laser SLAM system with deep learning and the trajectory interpolation algorithm proposed in this paper yields better graph building and loop closure effects in large scenes at SIASUN large scene campus. The use of T-trajectory interpolation ensures vibration-free and stable transitions between target points.

Keywords: SLAM; semantic laser; point cloud; occupation probability

1. Introduction

The significance of simultaneous localization and mapping (SLAM) technology and trajectory interpolation for mobile robots and autonomous driving has been increasing due to the continuous development of artificial intelligence technology[1-6]. SLAM algorithms and trajectory interpolation have been successfully applied in various fields, including campus inspection, logistics and distribution, and unmanned driving. When dealing with large-scale outdoor environments such as factories, laser point clouds are less affected by factors such as weather and light, and can perceive a 360-degree range. However, it is important to note that their operation speed is faster. However, laser point clouds typically only contain the geometric information of the environment. In dynamic environments, they may generate residual shadows on the map, which can decrease the accuracy of laser mapping[7-8]. Loopback detection in laser SLAM currently relies on traditional features such as position and intensity. However, this method is unsatisfactory as it does not take into account the semantic information of the surrounding environment, which is crucial for human beings to recognize whether a place has been reached or not[9-10]. Unfortunately, there The significance of simultaneous localization and mapping (SLAM) technology and trajectory interpolation for mobile robots and autonomous driving has been increasing due to the continuous development of artificial intelligence technology[1-6]. SLAM algorithms and trajectory interpolation have been successfully applied in various fields, including campus inspection, logistics and distribution, and unmanned driving. When dealing with large-scale outdoor environments such as factories, laser point clouds are less affected by factors such as weather and light, and can perceive a 360-degree range. However, it is important to note that their operation speed is faster. However, laser point clouds typically only contain the geometric information of the environment. In dynamic environments, they may generate residual shadows on the map, which can decrease the accuracy of laser mapping[7-8]. Loopback detection in laser SLAM currently relies on traditional features such as position and intensity. However, this method is unsatisfactory as it does not take into account the semantic information of the surrounding environment, which is crucial for human beings to recognize whether a place has been reached or not[9-10]. Unfortunately, there are few studies in related papers that incorporate semantic information into the SLAM loopback detection process and address the related technical issues.

To address the issues, this paper explores a semantic laser SLAM system that incorporates deep learning and a trajectory interpolation algorithm. In comparison to existing methods, the SLAM system presented in this paper incorporates the concept of voxels into the occupation probability map, thereby improving the ability of local voxel maps to represent dynamic objects. This paper combines the deep learning semantic laser SLAM system with the trajectory interpolation algorithm research to solve the problem of the system in which the sensors cannot directly obtain the semantic information of the point cloud and recognize the points on the dynamic objects. The related algorithms of the deep learning point cloud semantic segmentation are also incorporated. The article proposes a PointNet++ network for point cloud semantic segmentation. This network is capable of recognizing points on dynamic objects and extracting deep features of the dynamic point cloud of a scene to output semantic information of points on static objects. A global environment descriptor containing semantic information is generated to identify loops and add loop constraints to the factor graph for optimization. By adding semantic information, dynamic points in the map can be filtered out to improve map building quality. Map building and trajectory interpolation experiments were conducted in SIASUN Campus. The experimental results were compared with satellite maps to demonstrate the algorithm's ability to build maps and localize in large scenes. The planning curves of the robot show that the use of T-trajectory optimization effectively ensures vibration-free and stable transitions between target points. The experiments demonstrate the feasibility of the proposed algorithm. The time-consuming analysis of the SLAM system shows that it can perform real-time computation, meeting the real-time localization requirements of mobile robots.

2. Mobile Robotic Systems Overview

2.1. Mobile Robot Navigation Technology Program

As depicted in Figure 1, several open-source laser SLAM algorithm frameworks are available[11-12].

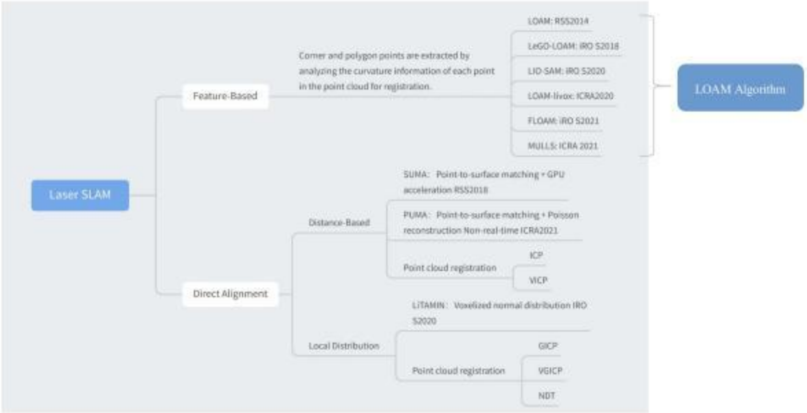

Laser SLAM
Feature-Based
Corner and polygon points are extracted by analyzing the curvature information of each point in the point cloud for registration.
LOAM: RSS2014
LeGO-LOAM: IRO S2018
LIO-SAM: IRO S2020
LOAM-livox: ICRA2020
FLOAM: IRO S2021
MULLS: ICRA 2021
LOAM Algorithm
Direct Alignment
Distance-Based
SUMA: Point-to-surface matching + GPU acceleration RSS2018
PUMA: Point-to-surface matching + Poisson reconstruction Non-real-time ICRA2021
Point cloud registration
ICP
VICP
Local Distribution
LiTAMIN: Voxelized normal distribution IRO S2020
Point cloud registration
GICP
VGICP
NDT

Figure 1. Laser SLAM system.

1.LOAM (Lidar Odometry and Mapping):

Front-end odometry: Scan to scan - feature point based alignment L-M nonlinear optimization.

Back-end optimization: Scan to map - map optimization.

2.LEGO-LOAM (Lightweight and Ground Optimized Lidar and Mapping):

Front-end odometry: Point cloud segmentation, scan to scan - two-stage L-M nonlinear optimization for feature point based alignment.

Back-end optimization: Scan to map - map optimization, graph optimization and loopback detection.

3.LIO-SAM:

Front-end odometer: Front-end odometer fused with IMU.

Back-end optimization: Graph optimization with added GPS factor and loopback detection.

2.2. General Framework of the SLAM System

SLAM algorithms are essential for enhancing the autonomy and intelligence of mobile robots. Recent research has identified two types of laser SLAM systems: feature-based alignment and direct alignment based on point cloud alignment methods. This paper proposes a framework for the SLAM system, as shown in Figure 2, which receives inputs from 3D LIDAR and outputs 6 DOF attitude estimates. The system is divided into three modules: front-end odometry, back-end nonlinear optimization, and loopback detection.

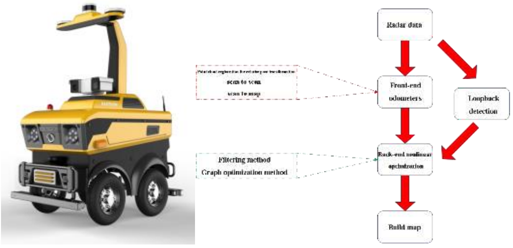


Figure 2. Robotic laser SLAM software platform.

The front-end odometers infer rough radar motion from adjacent frames of radar data and provide initial values for the back-end. Front-end alignment methods include ICP matching, NDT matching, PL-P matching, and CSM matching. ICP matching utilizes point cloud data to construct local geometric features, NDT matching constructs multidimensional variables based on normal distributions, PL-P matching approximates the actual surfaces using a segmented linear method, and CSM matching obtains the initial values through correlation scanning and least squares problems. The nonlinear optimization methods include gradient descent, Gaussian Newton method and L-M method, where the L-M method introduces the trust region based on Gaussian Newton method.

Loopback detection, also known as closed-loop detection, enables robots to recognize previously visited locations and achieve closed-loop capabilities for mapping. There are various methods for loopback detection, including feature matching, odometer position-based, and deep learning methods. Additionally, radar data can be matched using Scan-to-Scan, Scan-to-Map, and Map-to-Map methods.

Back-end optimization aims to improve the accuracy of estimating the robot's previous bit positions and waypoints in the presence of noise by reducing estimation errors in motion states and waypoints. The process of state prediction and measurement updating involves modeling the robot's motion and applying Kalman filtering. Additionally, graph-based optimization methods are used to represent the robot's poses as variables to be optimized and construct a graph of the error terms through the relationships between the poses. These methods are effective in improving the accuracy of robot localization and map construction in complex environments.

3. Map Organization and Update Strategy

doi:10.20944/preprints202402.0437.v1

3.1. Laser Odometry - Nonlinear Optimization Algorithm

Navigation systems use odometry data to estimate changes in robot position over time. In SLAM problems, both the front-end position optimization problem and the back-end graph optimization problem are modeled as nonlinear least squares problems. Therefore, nonlinear optimization algorithms are crucial for SLAM systems. A general nonlinear least squares problem can be defined with a minimization objective function.

$$\min_{x} F(x) = \min_{x} \frac{1}{2} \| f(x) \|^2 \qquad (1)$$

Where$x \in R^n, f(x)$ is a nonlinear function.

3.2. Voxel-based Local Map Construction and Updating

Compared to the 2D case, building and utilizing a 3D raster map for point cloud alignment requires significantly more computational effort, leading to increased burden on the system and reduced real-time performance. To address the issue of aligning 3D point clouds, we utilize NDT maps. These maps employ the distribution of points within a voxel to represent the entire voxel. The NDT algorithm divides the voxel into relatively large sections, treating the points within each voxel as sampling points of a single Gaussian distribution. The mean and covariance of the distribution of points within the voxel are then fitted. Additionally, rasters are utilized in the NDT algorithm to divide the map.

The paper introduces a voxel-based map representation for alignment, where local maps are voxelized and occupancy probability is incorporated to enhance the representation of dynamic objects. The environment is modeled using 1m3 square voxels to match the outdoor scene. Figure 3 illustrates the voxel mapping process. When aligning with the local map for the current frame, it is typically necessary to find the nearest neighbor of the point or voxel raster to establish the alignment relationship. Utilizing the hash algorithm for local voxels can expedite the voxel finding process and simplify voxel addition and deletion operations, resulting in lower algorithmic complexity.

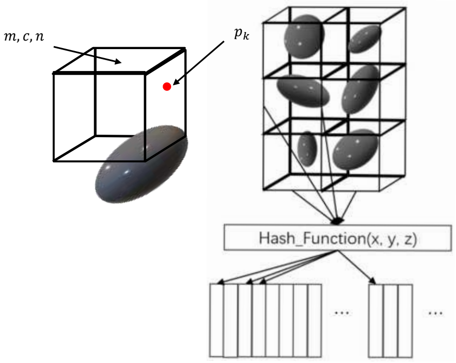


Figure 3. Schematic of voxel mapping.

When a new sampling point enters the voxel, the distribution of the voxel is corrected using an iterative update strategy. The traditional method of iterative updating each time is computationally expensive compared to this scheme, which greatly reduces computation. This correction scheme for map voxel information is a major research focus for maps. The original mean and covariance will not reflect the current new distribution within the voxel when a new point enters the voxel. Incremental corrections can be made to the existing mean and covariance through an iterative approach, rather than re-calculating them for all points. The data structure of the voxels in this system includes the mean, covariance, and occupancy probability as core parameters.

4. Combining Deep Learning for a Semantic Laser SLAM System

This paper enhances the network structure of previous research. Firstly, a multi-layer feature extraction module is used to achieve deep learning-based segmentation of dynamic object point cloud. Additionally, the output layer of the network is modified and retrained to output the semantic

doi:10.20944/preprints202402.0437.v1

categories of static object points. Secondly, the neural network's semantic results are used to create a global environment descriptor based on semantic information. Geometric and semantic similarity matching is used to identify loopback candidates, and then map-to-map matching is employed to precisely customize the loopbacks. This prevents the addition of erroneous constraints to the factor graph. Finally, in the outdoor environment with dynamic objects, we construct a pure 3D laser semantic SLAM algorithm to filter dynamic points based on semantic information and create a static semantic map of the point cloud. We generate a global environment descriptor containing semantic information and detect loopbacks using the loopback detection method, which are then optimized using the factor graph.

4.1. Segmentation Feature Extraction Based on Ground Constraints

Point cloud semantic segmentation is a crucial task in 3D applications as it provides high-precision localization information for SLAM systems to construct accurate maps. Additionally, it offers reference targets for buildings and man-made features in building information models.

A neural network framework-based scheme for semantic segmentation of 3D point clouds can determine the object categories in the point cloud data and provide a more comprehensive description of the environmental scene. This paper applies the PointNet++ point cloud semantic segmentation network to outdoor large scene point clouds with uneven density and large data volume. PointNet++ processes a set of points sampled in the metric space by building a multilayer neural network and extracts the features of the sampled points through multiple simplified PointNet. As illustrated in Figure 4, PointNet++ comprises multiple Set Abstraction (SA) layers. For each SA layer, the input vector is either the original point cloud or the local features extracted from the previous SA layer. The features of each layer are extracted using PointNet and then combined by a combination layer in the next SA layer to extract deeper features.

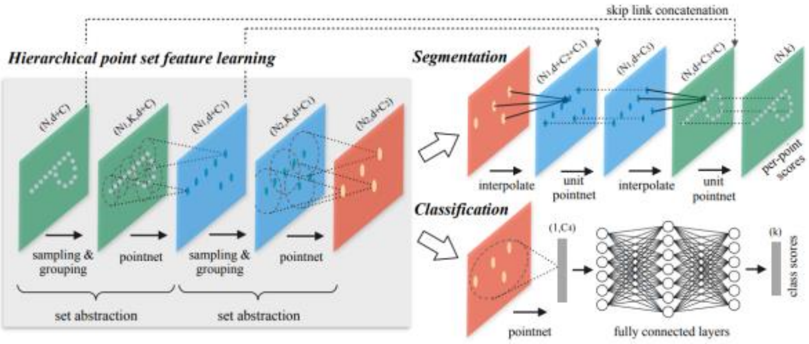


Figure 4. Schematic diagram of PointNet++ network structure.

The Set Abstraction Layer comprises three main components: the Sampling Layer, the Grouping Layer, and the PointNet Layer. The Sampling Layer selects a set of points from the input point set to serve as the center of the local neighborhood. The Grouping Layer constructs the local point set, which defines the local region of the centers. And the PointNet Layer uses mini PointNet to encode the local point set and obtain the feature vectors.

4.2. Closed-loop Detection and Position Optimization Flow

Closed-loop detection is a crucial component of the laser SLAM system. It ensures map consistency and eliminates accumulated errors during point cloud alignment, particularly when building maps for large scenes. The closed-loop detection strategy of the laser SLAM system can be divided into two algorithms: descriptor-based closed-loop detection and positional nearest neighbor-based closed-loop detection. The descriptor-based detection algorithm involves compressing high-dimensional point cloud data by extracting features from the point cloud. By comparing the low-dimensional descriptor data of two frames of the point cloud, it can be quickly determined whether the two frames of the point cloud may have been sampled from the same scene. The closed-loop nearest-neighbor detection algorithm compares the error values between the descriptors of the laser point cloud of the current frame and the descriptors of the point cloud of the historical laser keyframes. The worst and smallest historical point cloud descriptors are then selected to obtain the most probable closed-loop point of the current position.

The process of closed-loop detection, also known as scene recognition, involves generating a global environment descriptor that contains semantic information, which is then used for scene description and search. Once the closed loop is successfully detected, the bit positions in the global keyframes are optimized through graph optimization. The closed-loop detection thread can perform this step separately to complete global map optimization and reduce cumulative errors. The system extracts the local sub-map from the new global map and uses it to recreate the voxel map, completing the update operation of the old local map.

5. T-Trajectory Interpolation Strategy

The SLAM system enables the robot to accurately localize and build a map in unknown environments. Additionally, the trajectory interpolation feature generates smooth paths, ensuring the smooth motion of the robot. In the manipulation space, paths and poses are planned and interpolated separately. The resulting per-cycle positions are solved by inverse kinematics based on the model to obtain the corresponding joint angles for motion control. T-trajectory interpolation is designed to ensure that the robot exhibits smooth, accurate, and efficient motions when executing T-tracks. T-trajectory interpolation is the process of generating and optimizing T-trajectories in a robot, CNC machine, or other automation system. It involves inserting additional points into the path of the robot to ensure smooth, accurate, and efficient motion.

The objective of T-trajectory interpolation is to enable the robot to display desirable motion characteristics while executing T-trajectories using suitable mathematical algorithms and control strategies.

Acceleration:

$$A(t)=\begin{cases} A0\leqslant t<t_1 \\ 0t_1\leqslant t<t_2 \\ -At_2\leqslant t<t_3 \end{cases} \tag{2}$$

Speed :

$$V(t)=\begin{cases} A\tau_1 0\leq t<t_1 \\ AT_1\tau_2 t_1\leq t<t_2 \\ AT_1\tau_2-A\tau_3 t_2\leq t<t_3 \end{cases} \tag{3}$$

Distance:

$$S(t)=\begin{cases} S_s+\frac{1}{2}A\tau_1^2 0\leq t<t_1 \\ S_{01}+AT_1\tau_2 t_1\leq t<t_2 \\ S_{02}-\frac{1}{2}A\tau_3^2 t_2\leq t<t_3 \end{cases} \tag{4}$$

T-trajectory interpolation is a technique that helps to prevent robot instability when switching paths. Its key features include:

1. Smooth transitions: Ensure that the transitions of the robot between connecting target points are smooth to avoid erratic motion.

2. Trajectory Optimization: Interpolation algorithms can be used to generate T-trajectories that optimize the trajectory for a given motion condition, ensuring the shortest path, minimum acceleration/deceleration, and minimum mechanical stress.

3. Velocity Planning: The interpolation algorithm must consider the velocity changes in each part of the T-trajectory to maintain system stability by avoiding excessive speed or slowness.

6. Experimental Results and Analysis

A common use case involves a vast industrial complex located in SIASUN, comprising of fixed structures (static features), parked vehicles (semi-static features), pedestrians, and moving vehicles (dynamic features) in a typical dynamic environment. Figure 5 showcases an outdoor experimental vehicle system that verifies the proposed navigation architecture in this paper on the SIASUN Smart Park campus. The experiment is based on the outdoor security inspection robot developed by SIASUN. The robot is a four-wheeled ground mobile vehicle with Ackerman structure and is equipped with sensors such as LIDAR, camera, GPS, and IMU.

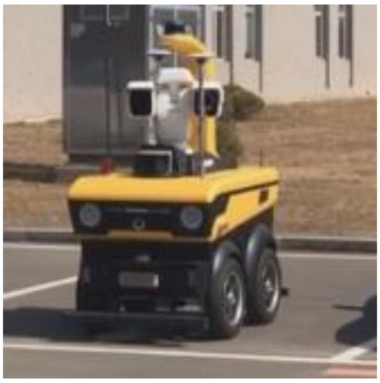


Figure 5. Outdoor experimental vehicle system.

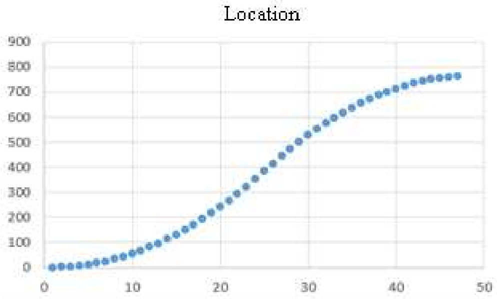
The study used a 3D LiDAR as a data source and an RTK-GPS system, which was constructed using GPS combined with a self-built base station, to provide trajectory truth data for the experiment. Table 1 displays the specifications of the equipped LiDAR model, the Sprint 16-line LiDAR RS-LiDAR-16.

Table 1. RS-LiDAR-16 parameter specifications.

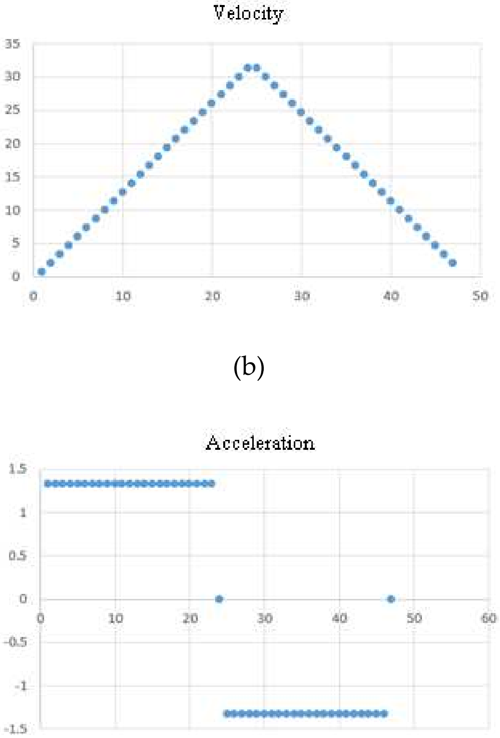
Parameters	Specification
Horizontal field of view	360°
Vertical field of view	30°
Horizontal Angular Resolution	0.1°/0.2°/0.4°
Frame Rate	5Hz/10Hz/20Hz
Ranging Capability	150m
Accuracy (typical)	±2cm

6.1. Trajectory Interpolation Test

Firstly, the curves of the mobile robot for trajectory interpolation are collected and tested to verify the key features such as smooth transition, trajectory optimization, and velocity planning.



(a)



(b)

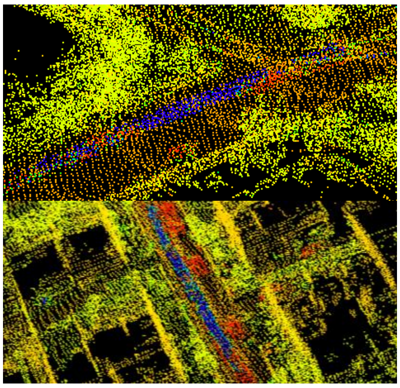
(c)

Figure 6. a). Location curve for T planning**.** b). Velocity curve for T planning. c). Acceleration curve for T planning.

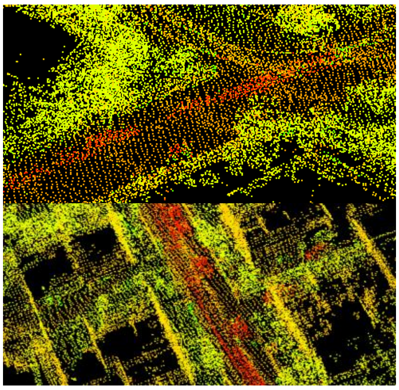
The aim of T-trajectory interpolation is to achieve optimal motion characteristics by utilizing appropriate mathematical algorithms and control strategies when executing T-trajectories. This improves the accuracy and efficiency of the automation system. Figure 6(a~c) demonstrates that T-trajectory interpolation results in a vibration-free and stable transition between target points, preventing robot instability during path switching. Trajectory optimization shapes the trajectory to meet specific motion conditions, considering factors such as shortest path, minimum acceleration/deceleration, and minimum mechanical stress. Velocity planning ensures appropriate velocity variations in different parts of the T-shaped trajectory to avoid excessively fast or slow movements, thereby improving system stability.

6.2. Semantic Maps and Closed-Loop Detection Experiments

The experiments on graph building were conducted using the ALOAM algorithm, Lego-LOAM algorithm, and our algorithm on the KITTI dataset sequence 05. The resulting trajectories were compared to the true values, and the results are presented in Figure 7a,b.



(a)



(b)

Figure 7. (a). Original semantic map without dynamic point filtering. (b). Static semantic map after dynamic point filtering.

Figure 7a displays the original semantic map without dynamic point filtering. It is evident that the blue section represents the residual shadow left by dynamic objects, which negatively impacts the quality of the map. In contrast, Figure 7b shows the map generated after dynamic object filtering during construction. The blue dynamic points have been filtered out, resulting in a reduction in the number of dynamic points in the map and an improvement in its quality. The conventional SLAM laser algorithm that uses feature point cloud building is unable to process dynamic points, which leads to residual shadows in the generated map.

Figure 8 shows the final global path of the robot, which has the same start and end points, indicating that the algorithm can effectively detect closed loops. Experimental tests on the SIASUN C1 building dataset demonstrate that the system can accurately identify closed-loop constraints and perform graph optimization.

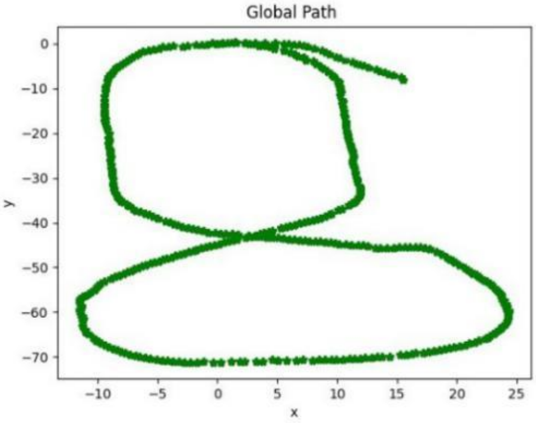
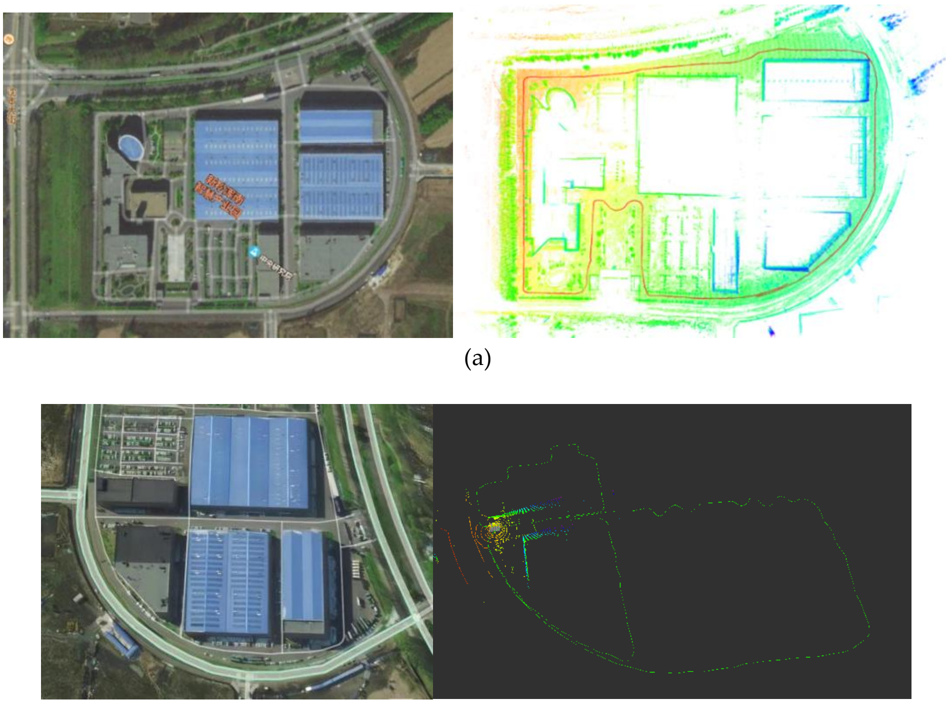


Figure 8. Closed-loop path trajectory.

The experiments demonstrate that the combination of a semantic laser SLAM system with deep learning and a trajectory interpolation algorithm can effectively utilize semantic information to identify loopbacks and perform global graph optimization, resulting in reduced cumulative errors and smoother robot trajectories.

6.3. Large-Scale Mapping Experiment for a Corporate Campus

The SIASUN outdoor mobile robot platform was used to extensively survey the periphery of the SIASUN campus. The effectiveness of whole-map building in a large scene was analyzed in this experiment. The first and last trajectories were connected to complete the experiment. An outdoor security inspection robot was used to patrol the periphery of the SIASUN campus, as depicted in Figure 9a. The route covered a total distance of approximately 1.8km, forming a long-distance loop. The environment consisted of typical outdoor features such as trees, buildings, and open roads. The robot's specific route is depicted by the red curve in Figure 9b. The global map, built by the robot, is shown in yellow-green, while the relative position of the trajectory calculated by the algorithm and the built map is shown in red.



(a)

(b)

Figure 9. a). Experimental scenarios and trajectories for the global map. Figure 9 (b) . Experimental scenarios and trajectories for the global map.

Figure 9a,b shows that the point cloud map aligns accurately with the satellite map, including the building edges, road edges, and tree shadows. The color change in the point cloud map reflects the height difference and confirms the color block change in the grayscale map. The results indicate that the proposed SLAM system achieves high positioning accuracy while maintaining good mapping efficiency.

7. Conclusion

This paper presents a complete solution for SLAM systems by combining a semantic laser SLAM system with deep learning and a trajectory interpolation algorithm. The work includes the following points:

(1) This paper proposes a general framework for a SLAM system based on open-source laser SLAM algorithms.

(2) The NDT algorithm is utilized to address the issue of aligning 3D point cloud alignments. It employs the feature point method for feature extraction and scan-to-map alignment of the point cloud to obtain the robot position with high accuracy. This enhances the ability of local voxel maps to represent dynamic objects.

(3) The semantic categories of the points are labeled as the point cloud is dynamically segmented using PointNet++. A global environment descriptor is generated based on the semantic information, and loopbacks are detected using a loopback detection method. The loopbacks are then optimized using a factor graph.

(4) The SLAM navigation algorithm employs T-trajectory interpolation for global and local planning to ensure the performance of the robot motion, resulting in a smooth and stable trajectory.

Experiments demonstrate that the semantic laser SLAM system can accurately recognize semantic information of points on both moving and static objects, meeting the basic requirements of the SLAM system in terms of operational speed. Combining the deep learning semantic laser SLAM system with the trajectory interpolation algorithm reduces cumulative error and provides a solid foundation for generating high-precision maps. The deep learning algorithm was tested on public datasets and compared with other SLAM algorithms. The results demonstrate that this algorithm satisfies the requirements of SLAM algorithms and is practical and feasible in outdoor scenes with dynamic objects.

8. Patents

Funding: This research was funded by the National Natural Science Foundation of China under Grants U20A20197 and the Provincial Key Research and Development for Liaoning under Grant 2020JH2/10100040.

References

1. Xie, H., Chen, W., Fan, Y., & Wang, J. (2021). Visual-inertial SLAM in featureless environments on lunar surface. Hang Kong Xue Bao, (1), 305.
2. Xing, Z., Zhu, X., & Dong, D. (2022). DE-SLAM: SLAM for highly dynamic environment. Journal of Field Robotics, 39(5), 528-542.
3. Chen, W., Wang, Y., Chen, H., & Liu, Y. (2022). eil-slam: Depth-enhanced edge-based infrared-lidar slam. Journal of Field Robotics, 39(2), 117-130. 117-130.
4. Zhang, Y., Song, J., Ding, Y., & Liu, J. (2023). Heterogeneous collaborative SLAM based on fisheye and RGBD cameras. Hang Kong Xue Bao, 44(10), 244.
5. Li, R., Qi, Y., Xie, H., & Han, X. (2023). Tightly coupled LiDAR SLAM method for unknown environment. Hongwai Yu Jiguang Gongcheng = Infrared and Laser Engineering, 52(9), 135.
6. Jiang, L., Liu, L., Zhou, A., Han, L., & Li, P. (2023). Improved ORB-SLAM algorithm based on motion prediction. Zhejiang Da Xue Xue Bao. Gong Xue Ban, 57(1), 170.
7. TIAN Ye, CHEN Hong-wei, WANG Fa-sheng, CHEN Xing-wen.(2021). Overview of SLAM algorithms for mobile robots. Ji Suan Ji Ke Xue, 48(9), 223-234.
8. Xiliang, S., Hongcan, G., Yanjun, S., Guangcai, X., & Qinghua, G. (2021). A tightly coupled SLAM method for precise urban mapping. Ce Hui Xue Bao, 50(11), 1585-1593.
9. Liu, S., Liu, Z., Hu, C., & Chen, H. (2023). Point-line feature adaptive fusion indoor SLAM algorithm. Xiaoxing Weixing Jisuanji Xitong = Journal of Chinese Computer Systems, 44(5), 1015.
10. Shen, Y., Chen, J., Li, X., Jiang, Q., Xie, O., Niu, X., & Zhu, Q. (2022). Dynamic visual SLAM based on unified geometric-semantic constraints. Shuju Caiji Yu Chuli = Journal of Data Acquisition and Processing, (3), 597.
11. Li, J., Pei, L., Zou, D., Xia, S., Wu, Q., Li, T., Sun, Z., & Yu, W. (2021). Attention-SLAM: A visual monocular SLAM learning from human gaze. IEEE Sensors Journal, 21(5), 6408-6420.
12. Tao, J., Fan, X., & Zhou, F. (2022). Point-line feature fusion in stereo visual SLAM algorithm. Xiaoxing Weixing Jisuanji Xitong = Journal of Chinese Computer Systems, 43(6), 1191.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.