

Research on Cartographer Algorithm based on Low Cost Lidar

Cui Zhi

Tianjin University of Technology and Education
Institute of robotics and intelligent equipment
Tianjin , P. R.China

Shi Xiumin

Tianjin University of Technology and Education
Institute of robotics and intelligent equipment
Tianjin , P. R.China

Abstract—The positioning and navigation functions of mobile robots require lidar, but the existing low-cost lidars have low positioning accuracy and cannot achieve large-scale closed-loop detection. The good performance of laser radar improves the positioning of mobile robots in navigation technology. threshold. Therefore, this paper takes a low-cost laser radar indoor mobile robot as the research object, and proposes a method to optimize the Cartographer algorithm parameters based on low-cost laser radar. By analyzing the working principle of laser radar, the low-cost laser radar is constructed. The shortcomings of the aspect, and the parameter optimization method of the Cartographer algorithm is proposed for this shortcoming. The experimental results in the real scene show that the improved Cartographer algorithm can construct the grid map in real time, and the efficiency and precision of the construction are obviously improved.

Keywords— Lidar; Cartographer algorithm; positioning and navigation; map construction

INTRODUCTION

Lidar ranging has the advantages of long measuring distance and high measurement accuracy, and is widely used in mobile robot SLAM [1]. RPLIDAR A1 Lidar is a two-dimensional laser radar that uses the principle of triangulation to measure distance. With the self-developed high-speed visual acquisition and processing mechanism, it can perform ranging operations of more than 8000 times per second. Two-dimensional laser radar is widely used in mobile robot navigation, including environmental modeling, obstacle detection and autonomous positioning [4]. However, there are few studies on the performance characteristics of two-dimensional Lidar. In order to improve the accuracy of the mobile robot's detection of the object, it is necessary to test and analyze the radar measurement error used. Therefore, this paper uses the Cartographer algorithm to test the RPLIDAR A1 two-dimensional laser radar indoor construction, and analyzes the experimental data, and proposes a method to improve the algorithm parameters. Research results can support the construction of maps for mobile robots in indoor environments

I. HOW RPLIDAR LIDAR WORKS

The low-cost RPLIDAR A1 two-dimensional laser radar adopts laser triangulation, laser wavelength of 785nm (infrared band), laser power of 3MW, pulse time of 110μs, and high-speed visual acquisition and processing mechanism, which can perform up to 2,000 times per second. . During the ranging process, RPLIDAR will transmit a modulated infrared laser

signal. The reflected light generated by the laser signal after being irradiated to the target object will be received by the RPLIDAR visual acquisition system, and will be solved in real time by the DSP processor embedded in the RPLIDAR. The distance value of the target object RPLIDAR and the current angle information will be output from the communication interface. Its working principle is shown in Figure 1.

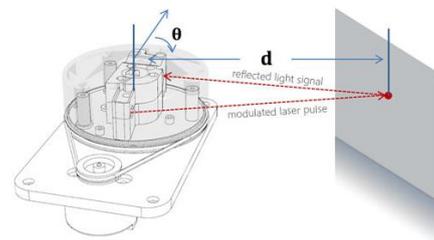


Fig 1 RPLIDAR A1 ranging schematic

II. CARTOGRAPHER ALGORITHM

Cartographer is Google's real-time indoor mapping project. The sensor is mounted on a backpack and can generate a 2D grid map with a resolution of 5 cm. Scan match is used to insert each frame of scan data obtained by the laser radar into a submap at the best estimated position, and scan matching is only related to the current submap. After generating a submap, a partial loop close is performed, using branch positioning and a pre-computed grid. After all submaps are completed, a global loopback is performed.

The entire system consists of two parts. The pose expression of the robot is k , and the pose of the robot is optimized by local optimization and global optimization respectively. The first two parameters represent translation, and the latter parameter represents rotation. The local optimization part is the matching of the lidar scanning frame and the sub-map, and the sub-map is locally optimized. The global optimization part is to perform global map optimization according to the pose relationship between the scan frames after finding the closed-loop scan frame.

A. local optimization

Local optimization is the process of matching the lidar scanning frame with the sub-map, and iteratively aligning the lidar scanning frame and the sub-map reference frame to construct the sub-map. The lidar scan is called a frame, the initial scan frame is at origin $(0,0)$, and the scan point is described as $H = \{h_k\}_{k=1,L,k}, h_k \in R$. In equation (1), the pose transformation of the scan frame to the submap is described as,

which can strictly map the scan points from the scan frame to the submap.

$$T_{\xi}p = \underbrace{\begin{pmatrix} \cos \xi_{\theta} & -\sin \xi_{\theta} \\ \sin \xi_{\theta} & \cos \xi_{\theta} \end{pmatrix}}_{R_{\xi}} p + \underbrace{\begin{pmatrix} \xi_x \\ \xi_y \end{pmatrix}}_{t_{\xi}} \quad (5)$$

Several iterative scan frames create a submap that takes the form of a probability grid with a resolution of r . For each grid point, we define a corresponding pixel. Each time a new scan is inserted into the probability grid, a set of hit grid points and disjoint miss grid points are calculated. For the hit point, we insert the adjacent grid points into the hit set. For the miss points, add all the relevant points on the scan center and scan point join rays to the miss set, except for the points in the hit set. For each point that has not been observed before, and if this has been observed before, then the point probability is updated according to equations (2) and (3).

$$odds(p) = \frac{P}{1-p} \quad (6)$$

$$M_{new}(x) = \text{clamp}\left(odds^{-1}\left(odds(M_{old}(x)) \cdot odds(p_{hit})\right)\right) \quad (7)$$

Before inserting the scan map into the sub-map, the scan frame pose is optimized with the current submap application ceres, and the mapping of the k scan points is superimposed by the nonlinear least squares optimization (4), and a total of k scan points hit Point, after transforming the pose and pairing with the probability value in the submap, each place corresponding to the display should be a big probability to be hit, then the match is better.

$$\underset{\xi}{\text{argmin}} \sum_{k=1}^K \left(1 - M_{smooth}(T_{\xi}h_k)\right)^2 \quad (8)$$

Since the least squares problem is a local optimal problem, a good initial value (the initial value of the pose) has a great influence on the solution. Therefore the IMU can be used to provide a scan-matched rotation variable for the initial pose value. In the absence of an IMU, it is necessary to increase the frequency or matching accuracy of the scan match.

B. global optimization

Global optimization is achieved through closed loop detection. Since each lidar scan frame is only matched with a submap containing some recent scan frames, the build error is slowly accumulated. To eliminate the accumulated error, a Sparse Pose Adjustment method is used to optimize all The pose of the scan and submap.

$$\underset{\Xi^m, \Xi^s}{\text{argmin}} \frac{1}{2} \sum_{ij} \rho \left(E^2 \left(\xi_i^m, \xi_j^s; \sum_{ij} \xi_{ij} \right) \right) \quad (9)$$

ρ is a loss function, and the purpose of using the loss function is to reduce the impact of outliers added to the optimization problem on the system. $\Xi^m = \{\xi_i^m\}_{i=1..m}$ and $\Xi^s = \{\xi_j^s\}_{j=1..n}$ are the poses of submap and scan in the world coordinate system, respectively, and are optimized under the constraints of the given relative pose ξ_{ij} and the associated covariance matrix Σ_{ij} . For paired Submap and scanj, ξ_{ij}

represents the relative pose of scan in the matched Submap coordinate system, and the covariance matrix Σ_{ij} can be estimated using the Ceres covariance estimation feature method.

The pose of the lidar scanning frame inserted into the sub-map is stored in the memory. When a sub-map is created, the corresponding scan frame and sub-map are taken into account for closed-loop detection. All scan matches are performed on the back end, and once a good closed-loop match is found, it is added to the global optimization.

III. EXPERIMENT AND RESULT ANALYSIS

This experiment selects the second floor of the electromechanical building as the experimental environment. The mobile robot used in the experiment is a two-wheel differential drive robot, as shown in Figure 1. The installed industrial computer uses the Intel i3-4030u processor. The differential drive robot is equipped with a laser radar model RPLIDAR A1, as shown in Figure 2. The radar has a measurement range of 0.15m to 12m and a measurement angle of 0 to 360°. The measurement accuracy of the measuring object within 1.5 meters is less than 5mm, and the measurement accuracy is less than 1% of the actual distance in all ranges; the resolution of the angle is approximately 1°, and the scanning frequency is 2.2 Hz.



Fig 2 RPLIDAR A1 Lidar

First, the bobac robot runs in the field, using the `revo_lds.lua` file built with the cartographer algorithm, and records the observed data with `roslabag`. The rest of the experiments use the recorded data to simulate to ensure the data is consistent during the test. The experiment uses 2D SLAM, the distance data can be processed in real time without additional information source, so IMU information is not needed, set: `TRAJECTORY_BUILDER_2D.use_imu_data=false`. The rest of the parameters use the default configuration, and the map created is shown in Figure 3.

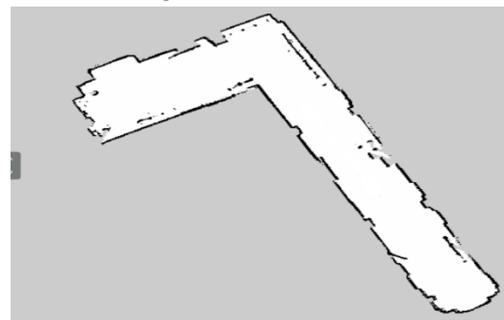


Fig.3 Map created by default parameters

Observing the map, it is found that the map has a large area of positional drift and motion distortion, so the algorithm parameters need to be optimized. The Cartographer 2D SLAM algorithm parameter optimization is divided into two parts, first local SLAM optimization, in order to generate better sub-maps, reduce the error generated when the scan data is inserted into the sub-map, and then perform global SLAM optimization on this basis. Better loopback constraints and build a more accurate global map.

a local SLAM optimization

When local SLAM constructs a submap, there is a problem that the submap slip and the map details are not clear. Therefore, the global SLAM needs to be turned off to avoid affecting the local SLAM when adjusting the parameters. Set `POSE_GRAPH.optimize_every_n_nodes=0` according to the cartographer algorithm tuning rules.

Submaps must be small enough to ensure drift below resolution, and on the other hand they need to be large enough to ensure that they are distinguishable when closed loop detection, so set `TRAJECTORY_BUILDER_2D.submaps_num_range_data=30`. The `realTimeCorrelativeScanMatcher` works by searching for similar scan data from the search window. According to the details in the map, it is not clear enough to modify the weight of the translation and rotation components of the search window or scanned data. Set `TRAJECTORY_BUILDER_2D.real_time_correlative_scan_matcher.linear_search_window=0.1`. Once the scan matcher finds the motion relationship between the two scan data, it will go to the motion filter. In order to avoid inserting too much scan data for each submap in the motion filter, by setting the motion of the mobile robot. The distance, angle or time threshold determines whether a scan data can be inserted into the current submap, so set: `TRAJECTORY_BUILDER_2D.motion_filter.max_range_radians=0.35`. Figure 5 shows the map created by closing the global SLAM after modifying the parameters.

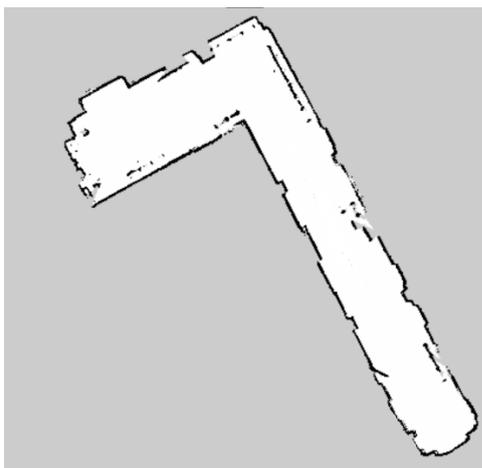


Fig.4 Close the map created by global SLAM after modifying the parameters

Observed that the inclination of the map 4 in the horizontal direction is reduced. Compared with the map 3 and the map 4, the optimized algorithm can better match the contour of the

obstacle without inputting important scan data under the input of the same data. .

b global SLAM optimization

On the basis of local SLAM optimization to obtain better quality sub-maps, continue global SLAM optimization. Since the submap is inserted into a certain number of track nodes, batch optimization is started. The size of the submap batch is adjusted according to the frequency of the algorithm running, and `POSE_GRAPH.optimize_every_n_nodes=35` is set. Node and submap build constraints are prioritized by the Fast Correlative Scan Matcher scan matcher for real-time closed-loop scan matching. Once the Fast Correlative Scan Matcher has a good enough proposal (beyond the minimum match score), the constructed constraints are sent to the Ceres scan matcher for pose refinement, so set: `POSE_GRAPH.constraint_builder.min_score=0.5`. Figure 5 is a map created after adding global SLAM optimization.

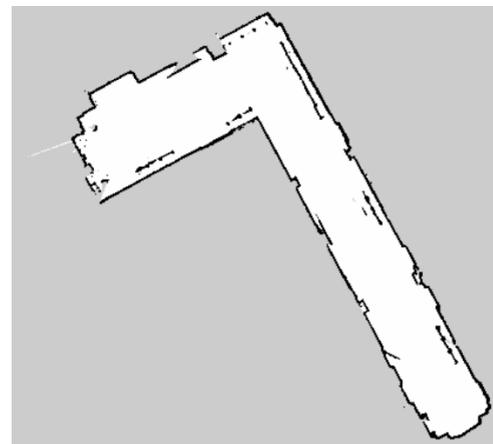


Fig.5 Map created after adding global SLAM optimization

IV. CONCLUSION

In this paper, the RPLIDAR A1 two-dimensional laser radar is tested by Cartographer algorithm and the accuracy of indoor mapping is successfully optimized. By analyzing the experimental data, the method of improving the algorithm parameters is proposed. The research results can support the construction of maps for mobile robots in indoor environment. By optimizing local SLAM, the error between scan data and sub-maps is repaired, the quality of sub-maps is improved, and the sub-map positions are reduced by optimizing global SLAM. The error makes the submap better coherent into a global map, fixing the map drift. The experimental results show that the optimized algorithm solves the problem of insufficient hardware configuration, fixes the motion distortion and map drift problem, improves the adaptability of the algorithm to complex environments, and uses the Cartographer 2D SLAM algorithm for RPLIDAR A1 two-dimensional laser mine. Provides an optimization method.

ACKNOWLEDGMENT

This paper is supported by The Graduate Innovation Fund Project of Research Department of Tianjin Vocational and Technical Normal University(YC19-08)

REFERENCES

- [1] Yu Jinxia, Cai Zixing, Zou Xiaobing, et al. Research on obstacle range of mobile robot based on lidar[J]. Journal of Sensors and Microsystems, 2006, 25(5): 31-33.
- [2] Yang Ming, Wang Hong. Environmental Modeling and Obstacle Avoidance of Mobile Robot Based on Lidar[J]. Journal of Tsinghua University: Natural Science, 2000, 40(7): 112-116.
- [3] Dissanayake G, Huang S, Wang Z, et al. A review of recent developments in Simultaneous Localization and Mapping[C]//6th International Conference on Industrial and Information Systems(ICIIIS), Kandy, Sir Lanka, 2011: 477-482.
- [4] Grisetti G, KuMmerle R, Stachniss C, et al. A Tutorial on Graph-Based SLAM[J]. IEEE Intelligent Transportation Systems Magazine, 2010, 2(4):31-43.
- [5] Luo Yuan, Yu Jiahang, Wang Longfeng, et al. Simultaneous localization and mapping of an improved RBPF based mobile robot[J].CAAI Transactions on Intelligent Systems, 2015, 10(3): 460-464.
- [6] Gutmann J S , Konolige K . Incremental mapping of large cyclic environments[C]// IEEE International Symposium on Computational Intelligence in Robotics and Automation. Piscataway, USA: IEEE, 1999: 318-325.
- [7] Wang Zhongli, Zhao Jie, Cai Hegao. A survey of back-end optimization method for graph-based SLAM under large-scale environment[J]. Journal of Harbin Institute of Technology, 2015, 47(01): 75-85.
- [8] Hess W , Kohler D , Rapp H , et al. Real-Time Loop Closure in 2D LIDAR SLAM[C]// IEEE International Conference on Robotics and Automation. IEEE, 2016:1271-1278.