ISSN: 2414-1895 DOI: 10.6919/ICJE.202209_8(9).0052

Overview of 2D Laser SLAM Algorithm Research

Dongna Xie, Feifan Zhao, and Guodong Xiu

Nanjing University of Science and Technology, Nanjing, China

Abstract

SLAM, or simultaneous localization and mapping, is primarily used to solve the problem of mobile robot positioning, navigation, and mapping in unknown environments. Among these, 2D laser SLAM has low cost and high accuracy in building drawings in an indoor environment. This paper describes the fundamental framework of 2D laser SLAM. Several popular SLAM algorithms are discussed. Finally, future developments in laser SLAM are anticipated.

Keywords

Inter-frame Matching; Loopback Detection; Back-end Optimization; Slam Algorithm.

1. Introduction

With the advancement of artificial intelligence, mobile robots and autonomous driving are becoming increasingly popular, which is inextricably linked to SLAM technology. Almost all navigation requires maps at the moment, but high-precision maps are expensive and not available in all environments. As a result, mapping, locating, and navigating in an unknown environment is becoming increasingly important. Laser SLAM development is nearing completion. This paper will introduce the framework and algorithm of 2D laser SLAM, as well as speculate on the future of laser SLAM.

2. SLAM Basic Framework

2.1 Sensor Data Processing

The distance of the mobile robot is calculated by the wheel odometer, and the point cloud data is obtained by the lidar scanning environment. The following two important sensors are introduced.

2.1.1 Laser Radar Motion Distortion Removal

The robot drifts during one round of low frame rate lidar scanning, resulting in lidar data distortion.

The ICP laser matching method does not account for laser motion distortion and laser data error. The VICP matching method assumes that the robot moves at the same speed when scanning a frame of data, calculates this speed, and then corrects the data distortion at each moment. However, because the assumption of average speed motion is not valid for low frame rate lasers, some researchers believe that sensors such as IMU and wheel odometer are required to aid in distortion processing.

2.1.2 Dometer Processing

According to the speed of the three wheels of the mobile robot, the speed (Vx,VY,Vθ) of the car in the two-dimensional coordinate system can be obtained, and then the displacement increment (dx,dy,dθ) can be obtained by integrating the speed. Then, we can convert this increment from the car body coordinate system to the world coordinate system to obtain the odometer of this section.

Because some fixed parameters of the wheels are not permitted at the factory, causing errors in track calculation, the odometer must be calibrated to remove system errors. We introduce the general linear method: given the lidar scan-match data as the true value b and the odometer data as A, and assuming the linear relationship: A*X=b, we can use the least square method to find the optimal transformation

ISSN: 2414-1895 DOI: 10.6919/ICJE.202209_8(9).0052

matrix X*, so that the odometer's pose can be transformed by the transformation matrix X* and the pose obtained by lidar inter-frame matching is the best match.

2.2 Front-end Registration Method

Front-end scanning matching is to estimate a pose difference so that two frames of laser point cloud data match as much as possible. Below, we will introduce several common matching algorithms: ICP matching, PL-ICP matching, optimization-based matching and correlation-based matching.

2.2.1 ICP Matching Method

Besl et al. [1] proposed in 1992 that ICP matching calculates the transformation matrix between two point clouds. Its principle is to find the corresponding points (pi, qi) in the target point cloud P and the source point cloud Q with matching, and then iteratively calculate the optimal matching parameters R (rotation transformation matrix) and T (translation transformation matrix), so that the aggregate matching degree of the two point clouds can be maximized, and the goal is to minimize the error function: $\mathrm{E}(R,t)=\frac{1}{n}\sum_{i=1}^{n}\| q_i-(Rp_i+t)\|^2$. ICP method depends on the initial value and converges in the first order, so the convergence speed is slow.

2.2.2 PL-ICP Matching Method

Because the ICP method uses the nearest neighbor points of two laser point clouds to find the corresponding points, it generates large random errors. Because Censi proposed an improved method based on this in 2008, PL-ICP finds the nearest two points to connect into a line and then takes the distance from point to line as the error. [2] PL-ICP has higher solution accuracy, second-order convergence, and faster convergence speed than ICP, but it is more sensitive to the initial value.

2.2.3 Matching Method based on Optimization

The optimization-based matching method is to obtain a robot's pose T so that the matching degree between all laser points and the world map is the greatest after they are projected into the world coordinate system via the pose T. (here, the matching degree is expressed by the probability and sum of laser points occupying the raster map). As a result, the maximum matching problem can be transformed into the objective function's extremum problem, which can then be solved by gradient descent using the nonlinear least square method. This method is sensitive to the starting value and is prone to falling into the local extremum.

2.2.4 Correlation Matching Method and Branch and Bound Acceleration

Olson[3] proposed Correlative Scan Matching in 2009, which is independent of the initial value. It is a violent search method, and the calculation amount is reduced by accelerating the strategy. The principle is that all new laser points are projected on the grid map using an estimated pose, and the current pose score is the sum of the logarithmic probability values of the grid where the laser points are located. The CSM method creates a search space near the estimated pose and speeds up the search using a branch and bound strategy. The final search result is the pose with the highest score. This method is used with Google's Cartographer algorithm.

2.3 Back-end Optimization

In SLAM front-end pose estimation, the pose estimation of the next frame needs to make use of the result of the previous frame, which will inevitably lead to the accumulation of errors frame by frame, so the back-end optimization is needed to reduce the accumulated errors. Back-end optimization methods can be divided into two categories: filtering-based methods and graph-based optimization methods.

2.3.1 Laser SLAM Method based on Filtering

The filtering method only estimates the state of the current moment; once an error in the previous moment occurs, it cannot be corrected. Although this method is not widely used at the moment, its Bayesian estimation and particle filtering theory are the foundations of SLAM development.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202209_8(9).0052

In Bayesian estimation, the predicted value xt of the current pose is obtained by adding the previous moment's pose estimation Xt-1 and the odometer ut, and then the predicted value xt is corrected by the current moment's observed data zt, allowing the pose estimation Xt to be obtained.

The Kalman filter [4] calculates the optimal estimation of the current pose by weighting the predicted and observed values, and the weighting coefficients of the predicted and observed values are constantly updated by their respective variances, allowing iteration to obtain the pose estimation of each moment.

The Kalman filter is used to analyze linear systems, while the extended Kalman filter (EKF) is used to analyze nonlinear systems. The basic idea behind EKF is to linearize nonlinear systems before applying the Kalman filter. [5] To linearize the system, Taylor expansion is required; however, if the problem is strongly nonlinear, linearization will result in a large error, and EKF may cause the filter to diverge.

Gordon et al. proposed a particle filter in 1993 to solve the aforementioned problems, which can deal with nonlinear situations and perform global positioning.

[6] Particle filtering approximates the probability distribution by using a series of particles. Its fundamental principle is to keep track of each particle's pose and weight. When the pose changes, each particle propagates its state and then resamples based on the weight, causing the particles to congregate near the area with the highest posterior probability. The particle filter will suffer from particle dissipation as a result of resampling, which is fatal for mapping. The amount of particle filter calculation increases exponentially as the dimensions increase.

2.3.2 Laser SLAM Method based on Graph Optimization

F. Lu et al. [7] proposed a graph optimization SLAM algorithm with global optimization to estimate every pose X0:t between time 0 and time t in 1997. If an error occurs at a specific point, it can be corrected in the next estimation.

As shown in Figure 1, graph optimization SLAM requires the construction of a pose graph. The graph's nodes represent the robot's pose, and the edges between them represent the spatial constraints of the two poses. The goal of graph optimization SLAM is to find an optimal configuration (robot posture in each node) using the nonlinear least square method, with the goal of minimizing the error between odometer prediction value and lidar observation value. In Figure 1, the odometer values from x1 to x4 are used as the predicted values of pose transformation; the loop was detected at x4, and then x1 and x4 were scanned-matched, with the result taken as the observed value. The error between the predicted value and the observed value of a loop can thus be calculated, and the optimal configuration can then be solved using the nonlinear least square method to minimize the sum of all loop errors.

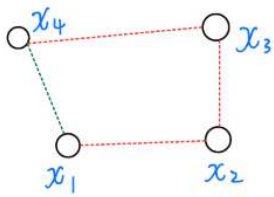

Figure 1. pose graph

2.4 Loop Detection

Loop detection is used to determine whether the robot has returned to its previous position using sensors and to correctly detect the loop, which is critical for building a global consistency map and repositioning the situation after a tracking failure. Loop detection is broadly classified into the following categories:

(1) Scan-to-Scan

ISSN: 2414-1895 DOI: 10.6919/ICJE.202209_8(9).0052

The interframe matching of point clouds is known as scan-to-scan. Because a single frame of laser data contains too little information, this type of matching is prone to errors and has been largely phased out for the time being.

(2) Scan-to-Map

Scan-to-map is a method of matching single-frame laser point cloud data with maps that is widely used today. The Cartographer algorithm uses the Scan-to-Submap matching method, and the submap contains n frames of data over a continuous period of time. [8] This method's matching accuracy is significantly higher than Scan-to-Scan.

(3) Map-to-Map

Map-to-Map is the matching of two sub-graphs to solve the problem of having less information in a single frame. The matching between sub-graphs, on the other hand, is computationally expensive and inefficient. Map-to-Map closed-loop detection was proposed by Wencheng et al. in 2018, and the matching candidate [9]set of closed-loop detection was optimized by locating, filtering, and compressing the table. The problems of matching accuracy and matching efficiency were effectively solved when creating large-scale maps.

(4) Other improved loop detection algorithms

Murphy et al. proposed a method of adding time information and online exploration and recognition, which improved the accuracy and loop rate [10] to some extent.

To speed up feature matching, Kawewong et al. used the K-means algorithm.

[11] Simonyan et al. proposed combining CNN model with spatial order filter to solve the problem of location recognition, which greatly improved feature extraction performance when combined with deep learning. [12].

2.5 Map Building

Moravec et al. proposed the grid map in 1985, which can distinguish between passable and impassable areas and can be used for path planning. Grid divides the entire environment into small grids, each of which has two states: occupied and unoccupied. Building a grid map entails estimating the best possible map given a given robot pose and observation data, that is, estimating whether each small grid is free or occupied.

The basic idea is that each grid keeps track of the occupied and idle probabilities, and the sum of the two is 1. The laser spot will be projected on different positions of the grid map for each frame of laser data based on the robot's pose. We update the grid's occupied probability based on the situation: the occupied probability of the grid in the laser click will increase. Finally, the value of each grid is compared to the given threshold to determine grid state.

3. SLAM Algorithm based on Lidar

3.1 Gmapping

Gmapping was proposed by Grisetti et al. in 2005, and improved on the basis of particle filter.[14] Because each particle contains its own raster map, in order to avoid memory explosion and keep the number of particles at a small value, Gmapping uses scan-match to optimize the pose and effectively reduce the number of particles. For the problem of particle dissipation caused by resampling, Gmapping sets a new resampling time, which greatly reduces the number of resampling. In 2018, Zheng Bing et al. proposed using firefly algorithm to improve Gmapping, aiming at the problem that Gmapping can't be accurately located due to particle dissipation. The particle set obtained after random sampling is moved to the high-likelihood distribution area, the distribution of particles is optimized, and the diversity of low-likelihood particles is increased..

3.2 Hector Slam

In 2011, Kohlbrecher et al. proposed a laser slam algorithm based on optimization method for inter-frame matching. This method does not depend on odometer, and can build a map according to laser

ISSN: 2414-1895 DOI: 10.6919/ICJE.202209_8(9).0052

information.[16] Its disadvantage is that the effect is not good when the radar frequency is low, and it is easy to be mismatched when turning quickly. Because there is no loop detection, the map has no adjustment ability when the accumulated error is large. Su Yiheng et al. proposed an optimization method for this in 2019: performing probabilistic Hough transform on lidar data, first reducing system noise and optimizing straight line data, and then replacing the original bilinear interpolation algorithm in Hector SLAM with bicubic interpolation, which effectively improved the accuracy of the map.

3.3 Cartographer

Cartographer is a set of SLAM algorithm based on graph optimization introduced by google.In the Local SLAM part, the odometer and IMU data are used to calculate the trajectory, and the attitude prediction value of small parking spaces is given. Then, a pose prediction is made by combining the matching results of lidar. After filtering, multiple frames of radar data are superimposed into sub-images, and multiple sub-images form a local map. Loop detection of scan-to-map is carried out in the Global SLAM part, and then back-end optimization is carried out, so that all sub-images form a complete map. Aiming at the problems of outliers and noises in multi-sensor data processing that affect the matching accuracy and the low accuracy of pose fusion algorithm, Shen Xin et al. proposed an improved Cartographer algorithm based on hybrid filtering algorithm and velocity integration pose fusion in 2021, which improved the quality of point cloud and the accuracy of point cloud matching.

3.4 Comparative analysis of algorithms

Table 1 will make a comparative analysis of three classical laser SLAM algorithms.

Table 1. Three Algorithms comparing

Laser SLAM algorithm	Advantages of algorithm	Algorithm shortcomings
Gmapping	Small amount of calculation and high accuracy in small scenes	Heavy reliance on odometer and no loop detection
Hector SLAM	No odometer required, high flexibility and expansibility.	High requirements for radar, sensitivity to initial value and no loop detection.
Cartographer	Low cumulative error	It takes up a lot of memory, the algorithm is bulky and the calculation cost is high.

4. The Future of Laser Slamming

The advantages of laser radar include accurate ranging, little interference from illumination and other environments, and stable operation. At the moment, laser slamming technology is more mature, and many of them are produced in industry. The camera is less expensive than the laser radar and can obtain rich texture information, but it has poor accuracy and more interference. As a result, fusion is the way of the future. At the moment, an increasing number of researchers are working on multi-sensor fusion. Furthermore, slamming is more closely combined with deep learning, which can not only extract feature points, improve matching accuracy and efficiency, but also abstract semantic features, which is more conducive to location and path planning.

Acknowledgments

Nanjing University of Science and Technology National Research Training Program Fund.

References

[1] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, pp. 239-256, Feb. 1992.

[2] A. Censi, "An ICP variant using a point-to-line metric," 2008 IEEE International Conference on Robotics and Automation, 2008, pp. 19-25.

[3] E. B. Olson, "Real-time correlative scan matching," 2009 IEEE International Conference on Robotics and Automation, 2009, pp. 4387-4393.

[4] Kalman, Rudolph Emil. "A new approach to linear filtering and prediction problems." (1960): 35-45.

[5] Julier, Simon J. and Jeffrey K. Uhlmann. “New extension of the Kalman filter to nonlinear systems.” Defense, Security, and Sensing (1997).

[6] N. Gordon, D. Salmond, and A. F. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” IEE Proc. F, Radar Signal Process. , vol. 140, pp. 107–113, 1993.

[7] F., Lu., Evangelos, E., Milios. (1997). Globally Consistent Range Scan Alignment for Environment Mapping. Autonomous Robots, 4(4):333-349.

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

[9] Wen Guocheng, Zeng Bi, Chen Yunhua, A closed-loop detection method for large-scale laser SLAM maps [J]. Computer Application Research, 2018, 35(6): 1724-1727, 1732.

[10] L. Murphy and G. Sibley, "Incremental unsupervised topological place discovery," 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 1312-1318.

[11] Kawewong, Aram, Noppharit Tongprasit, and Osamu Hasegawa. "A speeded-up online incremental vision-based loop-closure detection for long-term SLAM." Advanced Robotics 27.17 (2013): 1325-1336.

[12] Simonyan K, Zisserman A. Very Deep Convolutional networks for large-scale image recognition[J]. Computer Science, 2014.

[13] Moravec, Hans P. and Alberto Elfes. “High resolution maps from wide angle sonar.” Proceedings. 1985 IEEE International Conference on Robotics and Automation 2 (1985): 116-121.

[14] G. Grisetti, C. Stachniss and W. Burgard, "Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling," Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005, pp. 2432-2437.

[15] Zheng Bing, Chen Shili, Liu Rong. Research on Gmapping Based on Firefly Algorithm Optimization [J]. Computer Engineering, 2018, 44(09): 22-27.

[16] Kohlbrecher, Stefan , et al. “A flexible and scalable slam system with full 3d motion estimation.” 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics IEEE, 2011.

[17] Su Yiheng, Zhang Qizhi, Zhou Yali. An optimized Hector SLAM algorithm for low-end lidar [J]. Laboratory Research and Exploration, 2019, 38(09): 47-51.

[18] Shen Xin, Min Huasong. Improved Cartographer algorithm based on velocity integration and pose fusion [J]. Applied Laser, 2021, 41(05): 1063-1069.