engineering proceedings

Proceedings

Visual SLAM method for point, line and surface feature fusion

Xuyang Zhang [1] and Xin Zhang [1,2,3,*]

[1] School of Mechanical Engineering, Shenyang Ligong University, Shenyang 110159, China
[2] Shenyang Institute of Computing Technology Co., Ltd., Chinese Academy of Sciences, Shenyang 110168, China
[3] Software College, Northeastern University, Shenyang 110169, China
* Correspondence: zhangx@sylu.edu.cn; Tel.: +86-186-40218600

Abstract: Aiming at the problems of target initialization and target tracking failure in images, a visual SLAM (Simultaneous Localization and Mapping) algorithm for point-line-plane feature fusion is proposed to improve the accuracy and robustness of automatic localization and map creation in mobile robots. Firstly, a suitable algorithm is selected to extract point features, line features, and planar features, respectively; secondly, a structural constraint model for feature fusion is constructed to build a point, line, and plane fusion visual odometry and a loopback detection module; finally, a structural constraint model is constructed for fusing point, line, and planar features, fusing the data information between frames, realizing the estimation of the camera poses, constructing a global consistency map, and realizing the back-end nonlinear optimization. Compare with the ORB SLAM (Oriented FAST and Rotated BRIEF SLAM) and LSD SLAM (Large-Scale Direct monocular SLAM) methods, and verify the accuracy and effectiveness of the proposed method in this paper through the TUM dataset. The experimental results show that the plp SLAM (point-line-plane SLAM) method proposed in this paper reduces the average value of the root mean square error of the absolute trajectory by about 0.6 and 20, respectively, compared with the ORB SLAM and LSD SLAM methods, and is able to realize the motion trajectory in an unknown environment, which sufficiently verifies that the plp SLAM method proposed in this paper is feasible and effective.

Keywords: Feature points; Feature fusion; Positioning and mapping; Feature matching

Citation: To be added by editorial staff during production.

Academic Editor: Firstname Lastname

Published: date

Copyright: © 2023 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

When low texture or texture is single, it is difficult to effectively extract a large number of images in the image using conventional point feature methods, which leads to the loss of image information in the image, making the initialization of the image as well as the loss of the target. Although the direct method can reduce the correlation of feature values in the image to a certain extent, dense or semi-dense has the disadvantages of complex arithmetic and great influence by the light environment, which makes it difficult to close the loop detection and restricts the real-time SLAM creation. Traditional VSLAM (Visual SLAM) algorithms, such as PTAM (Parallel Tracking and Mapping) [5], ORB-SLAM [6], VINS-Mono [7], and MonoSLAM [8], all adopt point-based VSLAM algorithms, which are less robust to single-point features in weakly textured and structured scenes.

There are a large number of SLAM methods based on planar features. In 2006, Weingarten et al. utilized the laser digitizing scanning technique to acquire the surrounding environment, used the SP model to delineate the planar surface, and applied it to the SLAM system [1]. Moreno et al. proposed in 2014 to extract 2D features based on depth and color maps and used them to construct dense 2D maps [2]. Based on this, an EM method was utilized for optimal control of camera pose and surface characteristics [3]. In 2017, Hsiao et al. introduced the KDP-SLAM system, which is capable of obtaining 2D information in depth maps and integrating photometric and geometric methods to solve keyframes with planar markers of the pose [4].

In order to solve the problem of poor target feature extraction and tracking failure visual SLAM in complex scenes such as sparse texture or dynamic objects, this paper proposes a visual SLAM method based on ORB_SLAM2 with point, line and plane feature fusion; firstly, the algorithm's image data is collected and inputted for point, line and plane feature extraction; secondly, the fusion of point, line and plane feature geometric constraints, minimize the feature constraint function and add more constraint relations to achieve SLAM front-end motion estimation; finally, the unknown environment is mapped by loopback detection and back-end (nonlinear) optimization; the algorithm proposed in this paper is able to improve the accuracy and robustness of the SLAM system.

2. Visual SLAM method for fusing point-line-plane features

In order to realize SLAM in weakly textured or structured scenes, firstly, the point-line-plane features in the image are extracted; secondly, combining the three kinds of features: point, line, and plane, constructing the point-line-plane fusion visual odometry, and constructing the loopback detection module; finally, constructing the structural constraints model for the fusion of the point, line, and plane features to achieve local and global optimization, creating a global consistency map, and realizing the backend nonlinear optimization; and the proposed point-line-plane feature fusion with the plp SLAM method is shown in Figure 1.

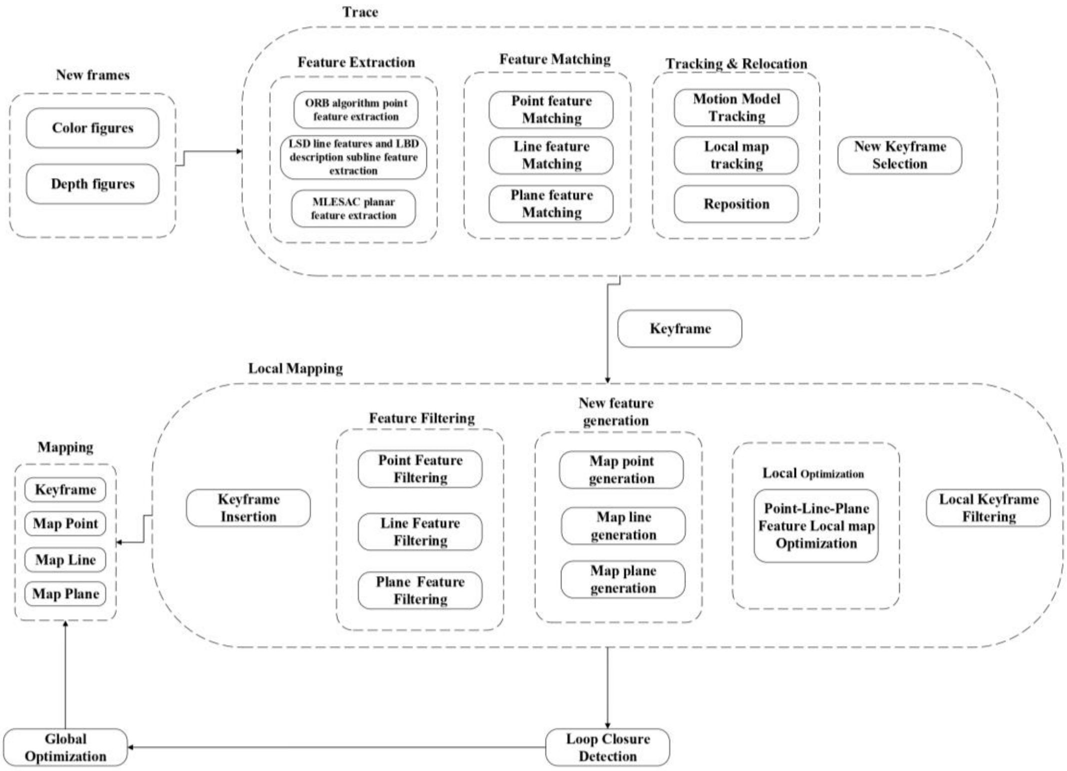


Figure 1. A plp SLAM method based on point-line-plane feature fusion.

The method mainly carries out tracking, local mapping and loopback detection three major threads of simultaneous processing, when the RGB-D image is acquired for point, line and plane feature extraction and feature matching, and then through the motion model tracking and local maps for target tracking and selecting new keyframes for local mapping; will be above the characteristics obtained by filtering and selecting the better point, line, and plane features to generate new maps points, lines, and planes for local optimization; will be screened for the local keyframes, and the map will be constructed through the loopback detection as well as the global optimization of the unknown environment.

2.1. Extraction of point-line-plane features

The front-end is also known as visual odometry, and its algorithms are mainly divided into two categories: feature point method and direct method [9], selected ORB algorithm [10] for point feature extraction, LSD line features and LBD (Line Band Descriptor) descriptor for line feature extraction, and Random Sample Consensus (RANSAC) algorithm for planar feature extraction.

2.2. Point, Line and Plane Feature Fusion Odometry

Construct a structural constraint model for feature fusion and utilize the merging of data information between image frames to establish structural constraints between image frames for camera pose estimation. In this process, the most important aspects are feature extraction, feature matching and pose estimation. As shown in Figure 2.

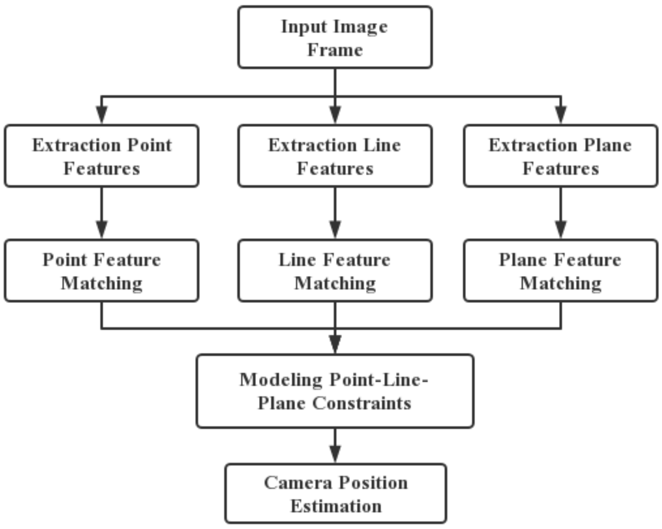


Figure 2. Camera Position Estimation.

First, after the depth camera collects the image information of unknown environment, ORB algorithm, LSD Line feature and LBD descriptor and RANSAC algorithm are used to extract point, line and plane features respectively, and corresponding features are matched in turn. Then the point, line and plane constraint model is constructed by point constraint, line constraint and plane constraint. Finally, the camera pose is estimated according to the point, line and plane constraint model.

2.3. Structural constraint model for point-line-plane feature fusion

The structural constraint model for feature fusion is shown in Figure 3. Where T_i is the bit pose of the key frame, P_i is the point marker, l_i is the line marker and π_i is the plane marker. The camera pose is estimated using this model and optimized.

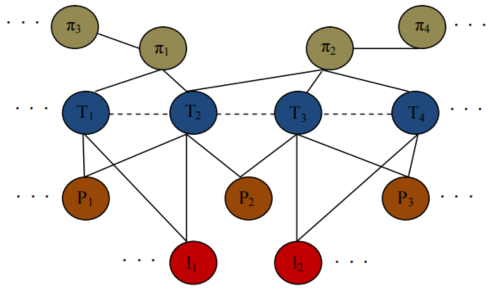


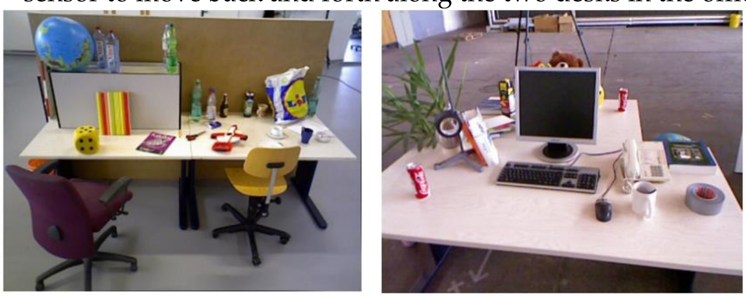
Figure 3. Structural constraint modeling of point-line plane features.

where the structural constraint function for point-line feature fusion is:

$$T_{cw} = \arg_{T_{cw}} \min\left(\sum H_P\left(f_p\right) + \sum H_{plane-p}\left(f_{plane-p}\right) + \sum H_l\left(f_l\right) + \Sigma H_{\parallel}\left(f_{\parallel}\right) + \sum H_{\perp}\left(f_{\perp}\right)\right) \tag{1}$$

3. Experimental results and analysis

In this paper, the proposed algorithm will be examined using the TUM dataset to compare it with the classical ORB SLAM and LSD SLAM methods.The TUM dataset is a generalized dataset for SLAM collected by the Foundation University in Munich (Germany) [11]. The fr3_long_office sequence was performed in two different environments, home and office, using the Lenovo G50 sensor, with a total length of 21.455 m. The duration was 87.05 s. The fr2_desk sequence was performed using the Kinect sensor to move back and forth along the two desks in the office scenario, with a total length of 18.880 m, and a duration of 99.30 s. The fr2_desk sequence was performed using the Lenovo G50 sensor to move back and forth along the two desks in the office scenario.



(a) fr3_long_office Sequence (b) fr2_desk Sequence

Figure 4. TUM datasets.

Figures 5 and 6 show the comparison between the estimated trajectories and the real trajectories of point feature SLAM (point SLAM) and point line plane feature SLAM (plp SLAM) for the fr3_long_office and fr2_desk sequences, respectively. Where the black solid line is the realistic trajectory, the green dashed line is the point SLAM estimated trajectory, and the red dashed line is the plp SLAM estimated trajectory.

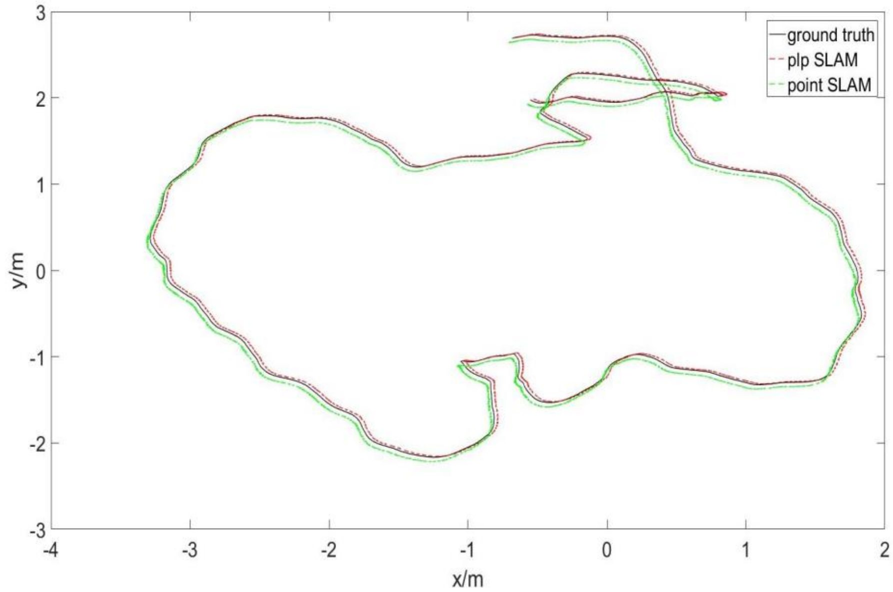


Figure 5. Trajectory estimation results (fr3_long_office sequence).

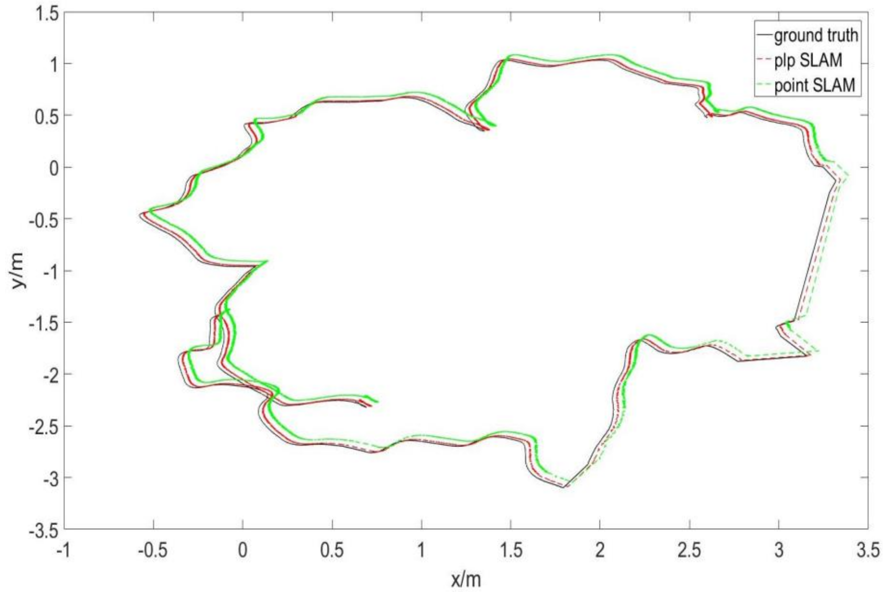


Figure 6. Trajectory estimation results (fr2_desk sequence).

To prove that the plp SLAM algorithm proposed in this paper has accuracy the algorithm is compared with ORB SLAM method and LSD SLAM algorithm. The predicted trajectories are compared with the real trajectories using Absolute Trajectory Error (ATE) method.

Assume that the position sequence of the predicted trajectory robot is M and the position sequence of its realistic trajectory robot is N. Then the relative trajectory deviation at the ith moment is:

$$F_i = N_i^{-1} M_i \tag{2}$$

Then the Root Mean Square Error (RMSE) at any point in time is:

$$(F_{i:n})_{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left\|F_{i,trans}\right\|^2} \tag{3}$$

where $F_{i,trans}$ is the translational component of F_i .

Table 1. Comparison of Absolute Trajectory Root Mean Square Error (cm).

	plp SLAM	ORB SLAM	LSD SLAM
fr3_long_office	2.54	4.05	38.50
fr2_desk	2.42	2.15	6.23

Table 1 shows that the accuracy of plp SLAM is better than other methods in texture sparse and large scenes. plp SLAM can extract the point-line structure constraints in the scene, which is more accurate than ORB SLAM and LSD SLAM for trajectory prediction. The plp SLAM method is adaptable in large scenes.

4. Conclusion

Aiming at the difficulty of initialization and target tracking in sparse scenes, a VSLAM algorithm based on point-line-face feature fusion is proposed, and a point, line, and plane feature fusion model is established that is suitable for structurally complex and more direct expression of geometric constraint relations in rich scenes. From the experimental results, it can be seen that the proposed plp SLAM algorithm has higher accuracy and better environment adaptation ability for low texture or single texture backgrounds.

Author Contributions: Methodology, X.Z. and X.Z.; software X.Z.; investigation, X.Z. and X.Z.; writing, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Project of the Education Department of Liaoning Province (LJKZ0258); the Doctoral Research Foundation of Liaoning Science and Technology Department (2022-BS-187).

Informed Consent Statement: Not applicable.

Acknowledgments: The authors thank the reviewers for their valuable comments on the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Weingarten, J.; Siegwart, R. 3D SLAM using planar segments[C]//2006 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2006: 3062-3067.
2. Salas-Moreno, R.F.; Glocken, B.; Kelly, P.H.J.; et al. Dense planar SLAM[C]//2014 IEEE international symposium on mixed and augmented reality (ISMAR). IEEE, 2014: 157-164.
3. Ma, L.; Kerl, C.; Stückler, J.; et al. CPA-SLAM: Consistent plane-model alignment for direct RGB-D SLAM[C]//2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2016: 1285-1291.
4. Hsiao, M.; Westman, E.; Zhang, G.; et al. Keyframe-based dense planar SLAM[C]//2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017: 5110-5117.
5. Klein, G.; Murray, D. Parallel Tracking and Mapping for Small AR Workspaces[C]// IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007, 63(4): 225-234.
6. Mur-Artal R , Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A Versatile and Accurate Monocular SLAM System[J]. IEEE Transactions on Robotics, 2015, 31(5): 1147-1163.
7. Tong, Qin, Peiliang, et al. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator[J]. IEEE Transactions on Robotics, 2018, 34(4): 1004-1020.
8. Davison, A.J.; Reid, I.D.; Molton, N.D.; et al. MonoSLAM: real-time single camera SLAM[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29 (6): 1052-1067.
9. M Bolaños, Dimiccoli, M.; Radeva,P. Toward Storytelling From Visual Lifelogging: An Overview[J]. IEEE Transactions on Human-Machine Systems, 2017, pp. 1-14.
10. Rosten, E.; Porter, R.; Drummond, T. Faster and better: A machine learning approach to corner detection[J]. IEEE transactions on pattern analysis and machine intelligence, 2008, 32(1): 105-119.
11. Sturm, J.; Engelhard, N.; Endres, F.; et al. A benchmark for the evaluation of RGB-D SLAM systems[C]//Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012: 573-580.