A LiDAR SLAM Algorithm Considering Dynamic Extraction of Feature Points in Underground Coal Mine

Xiaohu Lin [1], Wanqiang Yao [1], Tao Yan [1]

[1] College of Geomatics, Xi’an University of Science and Technology, Xi’an 710054, China - xhlin214@xust.edu.cn, sxywq@xust.edu.cn, yantao@stu.xust.edu.cn

Keywords: Underground Coal Mine, LiDAR SLAM, Dynamic Threshold, Feature Extraction, Degeneration Environment.

Abstract

To address the absence of GNSS signals in underground coal mines and the susceptibility of mainstream LiDAR SLAM to degradation due to insufficient feature constraints, this paper proposes a tightly-coupled SLAM algorithm incorporating LiDAR and IMU for use in such environments. The paper initially introduces a dynamic feature point extraction strategy, where the number of corner feature points can be dynamically adjusted based on the occurrence of degradation in the underground coal mine environment. This approach constructs a constraint matrix with rich and robust feature information, enhancing pose estimation accuracy in environments with inadequate feature constraints, mitigating degradation effects, and reducing global cumulative error. Subsequently, the consistent building of the underground coal mine environment is achieved through back-end factor graph optimization. Finally, to validate the effectiveness of the method, experimental analysis is conducted in an underground coal mine. The results demonstrate that LeGO_LOAM exhibits poor robustness in underground coal mines and fails to construct a globally consistent pose estimation and map. Conversely, the pose estimation error of the proposed method in this paper is 50.93% lower in the plane direction and 42.13% lower in the elevation direction compared to LIO_SAM. This underscores the method's significance as crucial technical support for the intelligent perception and positioning of robots in underground coal mines.

1. Introduction

Coal mine intelligence is one of the key research and development directions advocated by the state, and it is also the key to realize the transformation of the coal mine industry and solve the problem of coal mine safety mining. The realization of coal mine intelligence requires three aspects of technology support: intelligent perception, intelligent decision-making, and automatic execution (RHE Haitao et al., 2021; Wang Guofa, 2022). Simultaneous Localization and Mapping (SLAM) is one of the key technologies to construct maps and localization of complex environments and realize unmanned autonomy for underground coal mine operation vehicles (Wang Guofa et al., 2019).

In recent years, domestic scholars have proposed unmanned mining, intelligent mining, and other related definitions through SLAM-related research. Zhao Yuan et al. (2021) believe that the key technology of mining intelligent interconnected new energy trackless auxiliary transportation system lies in the vehicle scheduling system, vehicle intelligent driving system, and its technical barriers lie in the real-time accurate position estimation of the vehicle and the intelligent perception of the environment, and put forward the use of 5G Internet of things technology to promote the application of intelligent driving technology in the trackless auxiliary system. Zhang Chao et al. (2021) and others proposed a cantilevered digging position estimation method based on binocular vision for the complex working conditions of low illumination, high dust, and high water mist in the coal mine underground, which utilizes the image composed of infrared LED light source to measure the target and artificially constructs the image features to solve the problem of missing feature information. Bao Wenliang (2021) discusses the applicability of laser ranging sensors by analyzing the underground scene in coal mine. Finally, the applicability of the Monte Carlo localization method based on laser-ranging sensors in the roadway environment is verified through simulation experiments.

Foreign scholars have conducted a lot of research on SLAM algorithms based on vision (Forster, C. et al., 2014; Engel, J. et al., 2018; Qin, T. et al., 2018; Campos, C. et al., 2021) and LiDAR (Rusu, R. et al., 2010; Fernandes, D. et al., 2021; Zhao, S. et al., 2019; Shan, T. et al., 2020; Liu, Z. et al., 2021). Because LIDAR has the ability of intuitive map representation, high ranging accuracy, is not easily affected by changes in lighting and viewing angle, and operates under all-weather conditions (Zhou, Z. et al., 2021), it is widely used in unmanned fields (Hu, Y. et al., 2021), and is more suitable for SLAM in the complex and variable coal mine environment with poor lighting conditions. Huber and Vandapel (2006) used a high-precision LIDAR scanner to establish a high-precision coal mine 3D geological model, but this method requires post-processing of the point cloud, which cannot meet the real-time localization and mapping requirements and has a high cost. Ren et al. (2019) investigated a lightweight closed-loop detection and optimization algorithm based on the generalized ICP (GICP) (Segal, A. et al., 2009), and proposed an optimization method of SLAM based on the 3D point cloud alignment of GICP, but its accuracy still needs to be improved. Considering the real-time and accuracy of localization and map building, there are still many problems to be solved for SLAM in coal mine underground environment.

Aiming at the real-time positioning and mapping needs of coal mine underground, a laser SLAM algorithm method with dynamic extraction of features is proposed, which evaluates the degree of degradation online and adaptively adjusts the number of feature points extracted according to the actual environment of the coal mine roadway to make full use of the feature point information in degraded environments and improve the accuracy of the point cloud alignment. An adaptive key frame selection method is adopted in the factor graph optimization, and the keyframes are selected according to the bit position difference between the two frames before and after. The

effectiveness of the algorithm in coal mine underground applications is verified by experiments.

2. System Overview

The SLAM algorithm based on the tight coupling of LiDAR and IMU proposed by the author is illustrated in Figure 1. It employs a dynamic feature point extraction algorithm to increase the number of edge feature points in areas with limited feature constraints. Additionally, an adaptive keyframe selection scheme is used in factor graph optimization, enhancing the algorithm's real-time performance by setting a positional transformation threshold between two frames of the point cloud and adaptively selecting keyframes. The key steps of the algorithm are as follows:

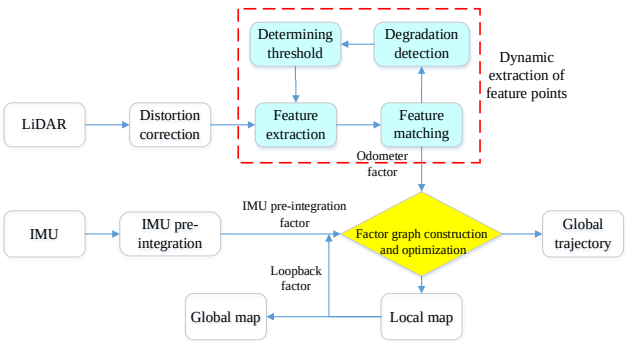


Figure 1. Flowchart of the method

(1) IMU Pre-Integration: Integrate angular velocity and linear acceleration using the IMU motion equations at a certain sampling interval to obtain the position of the LiDAR point cloud at the corresponding moment.

(2) Aberration Correction and Feature Point Extraction: Correct motion distortion in the LiDAR point cloud using the relative motion position obtained from IMU pre-integration, and extract edge and plane points using the depth image.

(3) Feature Matching and Pose Estimation: Perform matching using the scan-map matching method, compose LiDAR odometry factors, and optimize them with IMU pre-integrated factors by adding them to local map matching, correcting the pose of the LiDAR odometry solution in real-time.

(4) Factor Graph Construction and Optimization: Optimize the attitude estimation using the LiDAR odometer factor, IMU pre-integration factor, and closed-loop factor to update all keyframe factors.

(5) Closed-Loop Detection: Perform scan-map optimization, use the ICP matching method to obtain the optimized position, delay updating the current frame's position, generate the closed-loop factor, and update the factor graph.

(6) Global Mapping: Receive the 3D point cloud corresponding to the optimized positions from the factor graph, and stitch the point clouds together to create a globally consistent trajectory and map.

3. Front-End Odometers

The estimation of the relative position between frames of the LiDAR point cloud relies on IMU pre-integration information to eliminate displacement and rotational distortion in the point cloud. The corrected point cloud is used for feature point extraction, combined with edge and planar features for matching, using the predicted value of the IMU pre-integrated pose as the initial value for matching. The pose is then determined using the scan-map matching algorithm.

3.1 Dynamic Feature Point Extraction

With the dynamic threshold feature point extraction method, the threshold can be adaptively adjusted according to the richness of environmental features to obtain more high-quality edge feature points. This approach provides more feature information in feature-rich environments, thereby improving the robustness of local pose estimation.

(1) Projection to Depth Image: The LiDAR point cloud set acquired by single-frame scanning is projected into the depth image, where each depth point has a corresponding pixel position in the depth image. The image depth value is the Euclidean distance from the point cloud to the origin of the LiDAR coordinate system.

(2) Curvature Calculation: The curvature of a point cloud is calculated using the difference in depth between the left and right 5 points located on the same scan line as the point to minimize computational overhead.

$$c_i = \left\| \sum_{j \in S, j \neq i} r_j - r_i \right\| \tag{1}$$

where c_i denotes the curvature of the i point, r denotes the depth of the point, S is the set of points consisting of 5 points taken from the left and right sides of the point, and r_j denotes the depth of the points in the set of points. When the feature point is located in the plane, the distance of the points on the plane from the far point is approximately the same, then the sum of the squares of the curvature difference will be a very small value, on the contrary, when the feature point is located in the position of the protrusion of the plane, then the depth difference between the point in the position of the protrusion and the surrounding points is larger, then the sum of the squares of the depth difference is a larger value.

(3) Dynamic Extraction of Feature Points: To ensure uniform distribution of extracted feature points, the point cloud acquired by the VLP-16 LiDAR is divided into 6 parts, each covering a 60° range and containing up to 300 points. The point cloud is traversed and sorted according to the curvature value, and the curvature thresholds T_{corner} for edge points and T_{surf} for plane points are set. The points with curvature less than T_{surf} are extracted as candidate planar feature points. Considering the large number of planar feature points in the environment, the number of planar feature points is not limited here, and all the candidate points with curvature less than the threshold are subjected to voxel filtering, which is used as the final extracted planar feature points.

The points with curvature greater than T_{corner} are extracted as candidate edge feature points N , and the maximum number of

extracted edge feature points involved in feature matching is M . According to Zhang et al. (2016) degradation judgment theory in the LIO-SAM algorithm based on the additional online degradation judgment, the size of the degree of degradation can be judged by the degradation factor λ (λ smaller represents the more serious degradation). As shown in Figure 2, with an empirically set degradation occurrence threshold $\lambda_{th}=100$. If $\lambda \geq \lambda_{th}$, 50% curvature greater will be selected from the candidate edge feature points as edge feature points (i.e., $M=0.5N$, 0.5 is the empirical threshold). Otherwise, the number of candidate edge feature points with 80% greater curvature will be used as the maximum extraction number threshold.

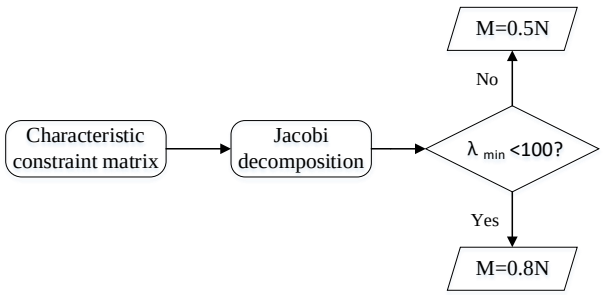


Figure 2. Dynamic threshold flowchart

By performing the above process for each scan line for each copy of the point cloud, uniformly distributed feature points are obtained. The set of edge feature points is denoted as F_c ; the set of planar feature points is denoted as F , and the number of planar feature points is not limited, which tends to be more than the number of edge points.

3.2 Feature Matching-Based Positional Solving

After extracting feature points, feature association is necessary to construct the residual function for position optimization. The author employs a Principal Component Analysis (PCA)-based algorithm for line and surface feature fitting. The feature-based matching algorithm optimizes the positional transformation by minimizing the sum of squares of the distances from point to line and point to surface.

Residual function construction: Let the translation-rotation parameter of the solution be x . The optimization objective function constructed from the point-to-line and point-to-surface distances can be expressed as:

$$f(p_E, x) = d_E \tag{2}$$

$$f(p_S, x) = d_S \tag{3}$$

where p_E denotes the set of edge points, d_E is the distance from its corresponding point to the line, p_S denotes the set of plane points, and d_S is the distance from its corresponding point to the plane. This can be further expressed as:

$$f(p, x) = d \tag{4}$$

where p denotes all the feature points and d denotes their corresponding point-to-line and point-to-plane distances. This formula represents the objective function of the final optimization.

By constructing the residual functions of the distances from points to lines and points to surfaces as described above, they are optimized to solve the bitmap transformation. The Jacobian matrix for solving the coordinate transformation is computed using the Levenberg-Marquardt (L-M) two-step method. First, plane points are used to solve for the vertical direction change, roll angle, and pitch angle. Then, edge feature points are used to solve for the lateral direction change, forward direction change, and heading angle. Finally, the positional transformation between adjacent frames is obtained through joint two-step optimization.

4. Back-End Optimization and Map Building

To improve the efficiency of back-end computing, only representative keyframes are selected for optimization estimation, without optimizing the entire map point cloud. Keyframes are selected based on the principle that the positional change exceeds a certain threshold, rather than relying on a fixed distance or a fixed time interval.

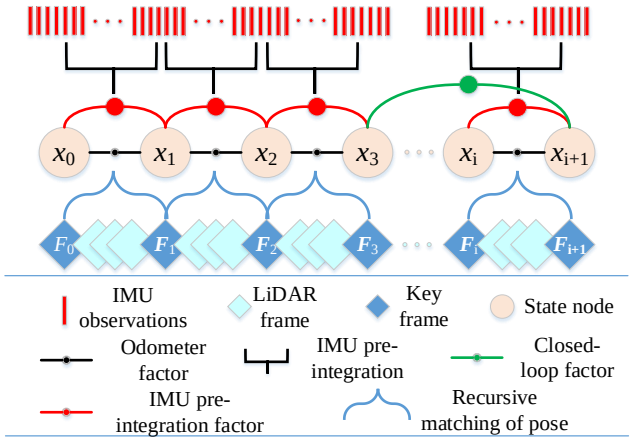


Figure 3. Factor graph optimization model

As shown in Figure 3, the LiDAR odometry factor, the IMU pre-integration factor, and the closed-loop factor are used as edge update state nodes. Each time a new node is inserted, the entire factor graph is optimized to update the current position estimate. The created position graph is optimized using the g2o optimization library. Once optimization is complete, the 3D point clouds corresponding to the position nodes are stitched together to form a globally consistent trajectory and map.

5. Experiments

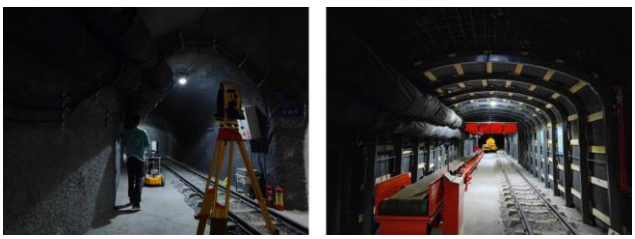
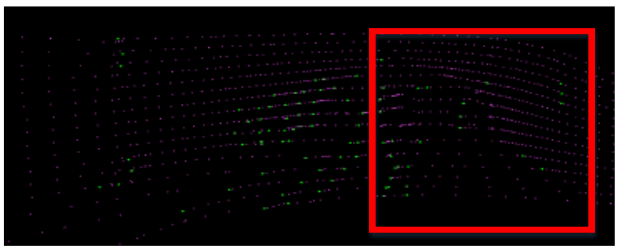


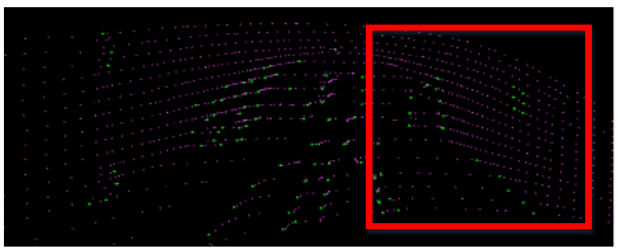
Figure 4. Roadway environment

To verify the effectiveness of the algorithm, data was collected from an underground coal mine using the Autolabor robotic trolley platform integrated with a Velodyne VLP-16 LiDAR and an Ellipse2-N IMU, as shown in Figure 4. The roadway forms a small loop with a total length of about 280 meters and a width of about 3 meters. Due to the complexity of the roadway environment, the robotic trolley had to travel close to the side of the roadway. There is no GNSS signal in the roadway, so seven checkpoints were observed using a total station to serve as reference points for evaluating trajectory accuracy.

5.1 Feature Point Extraction Experiments

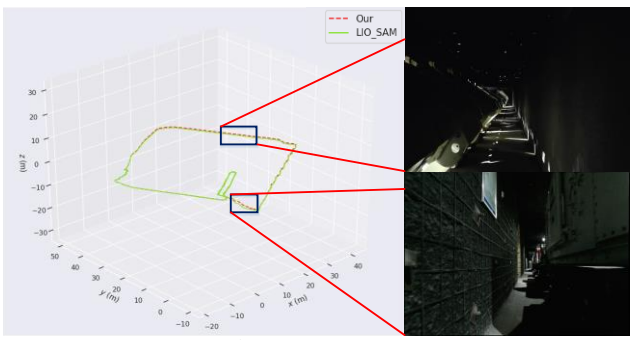


（a）LIO-SAM

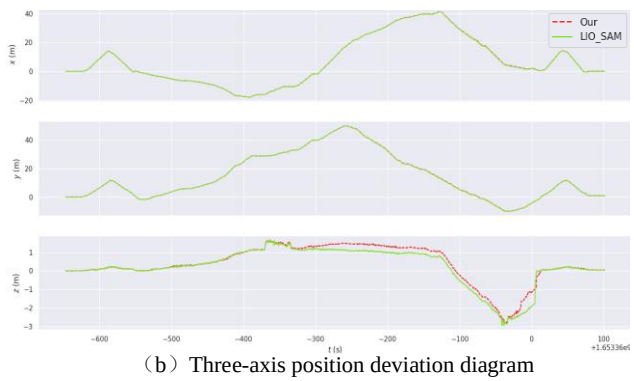


（b）Our

Figure 5. Comparison of feature point extraction algorithms



（a）Experimental scene



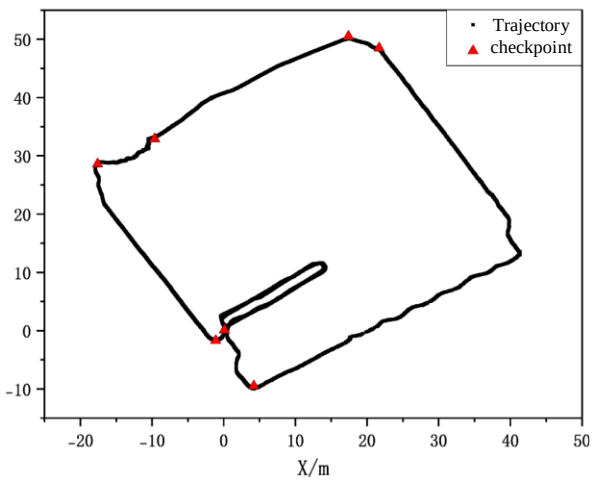
（b）Three-axis position deviation diagram

Figure 6. Pose deviation diagram between the experimental scene and the forward direction

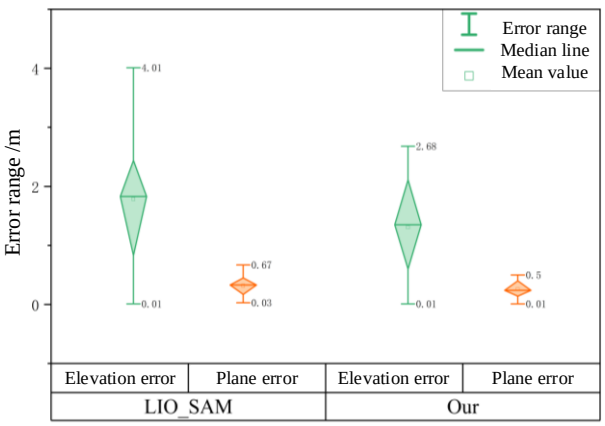
The dynamic feature point extraction algorithm was compared with the original algorithm by analyzing data from the alleyway. The number of feature points extracted by both methods was compared in the same scene and at the same moment. As shown in Figure 5, the dynamic feature point extraction method adjusts the number of edge feature points based on scene changes, allowing for the extraction of more high-quality edge feature points in feature-rich scenes. Compared to the fixed threshold extraction method, the dynamic method improves both the number and quality of local LiDAR point cloud edge feature points, providing better feature information for local map pose estimation.

5.2 Pose Estimation Accuracy Analysis

Due to the failure of the LeGO-LOAM algorithm, the trajectory accuracy of the proposed algorithm is evaluated only against the LIO-SAM algorithm. Figure 6 shows the comparison of trajectories, scenarios where deviations occur, and the deviation along each coordinate axis. In Environments 1 and 2, the width of the travel paths is less than 1 meter. Consequently, the scanning range on both sides of the robot trolleys is significantly restricted, reducing the extracted feature information and resulting in considerable discrepancies between the two algorithms in both planar and vertical directions.



（a）Checkpoint distribution



（b）Comparison of elevation and plane error

Figure 7. Comparison of control point distribution and algorithm error

Because there is no GNSS signal in the roadway, accurate traveling trajectories cannot be obtained. Therefore, seven checkpoints are deployed at roadway turns with good observation conditions to verify the accuracy of the output trajectory results (Figure 7). The experimental results show that the in-plane error of the proposed algorithm is 0.212 meters, and the elevation error is 0.820 meters. The cumulative errors of both algorithms in the vertical direction are significant, with the maximum deviation of the proposed algorithm being 2.683 meters and that of LIO-SAM being 4.012 meters (Tables 1 and 2). The proposed algorithm demonstrates reduced cumulative deviation in both planar and elevation directions compared to the LIO-SAM algorithm, verifying its improved accuracy for SLAM pose estimation.

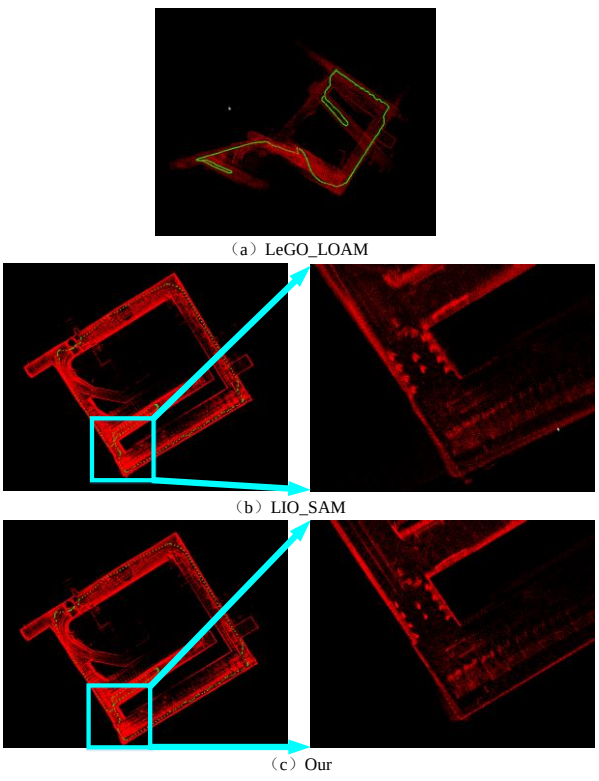
Algorithm	Min/m	Max/m	Mean/m	RMSE/m
LeGO_LOAM	N/A	N/A	N/A	N/A
LIO_SAM	0.041	0.530	0.382	0.432
Our	0.052	0.460	0.236	0.212

Table 1. Statistical results of plane error

Algorithm	Min/m	Max/m	Mean/m	RMSE/m
LeGO_LOAM	N/A	N/A	N/A	N/A
LIO_SAM	0.013	4.012	0.787	1.417
Our	0.012	2.683	0.474	0.820

Table 2. Statistical results of elevation error

5.3 Mapping Analysis



（a）LeGO_LOAM

（b）LIO_SAM

（c）Our

Figure 8. Mapping comparison

As shown in Figure 8, the LeGO_LOAM algorithm exhibited obvious breaks in the lane traveling, resulting in failed map building. In contrast, both the proposed LiDAR SLAM algorithm and the LIO_SAM algorithm successfully constructed a complete map. Through local zoom-in, it can be observed that the map output from the LIO_SAM algorithm is more cluttered, with more noise and a thicker point cloud layer. Thus, the proposed algorithm demonstrates higher accuracy in map building.

6. Conclusions

(1) By assessing the degradation of the underground coal mine environment, the number of feature points extracted is dynamically adjusted. This approach ensures real-time performance while obtaining a larger number of high-quality feature points that constitute the feature constraint matrix, providing a basis for accurate local pose estimation.

(2) The LiDAR SLAM algorithm, which incorporates dynamic feature point extraction, fully utilizes feature constraint information even in degraded scenes with missing features. This improves the accuracy of local pose estimation, effectively reduces the accumulation of global error, and enhances the accuracy of global pose estimation and map building.

(3) Compared to mainstream LiDAR SLAM algorithms, the proposed algorithm improves the accuracy of pose estimation and map building. It also better handles degraded scenarios, providing a valuable reference for the intelligent perception of robots in underground coal mines.

References

Bao, W. 2021. Monte Carlo Localization for autonomous auxiliary transport vehicles used in coal mine. *Coal Science and Technology*, 49(11):211-217.

Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M., Tardós, J.D. 2021. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. *IEEE Trans. Robot.*, 37, 1874–1890.

Engel, J., Koltun, V., Cremers, D. 2018. Direct sparse odometry. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40, 611–625.

Fernandes, D., Afonso, T., Girão, P., Gonzalez, D., Silva, A., Névoa, R., Novais, P., Monteiro, J., Melo-Pinto, P. 2021. Real-Time 3D Object Detection and SLAM Fusion in a Low-Cost LiDAR Test Vehicle Setup. *Sensors*, 21, 8381.

Forster, C., Pizzoli, M., Scaramuzza, D. 2014. SVO: Fast semi-direct monocular visual odometry. *In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA 2014)*, Hong Kong, China, 31 May–7 June 2014; pp. 15–22.

He, H. 2022. Key technology of intelligent mining system in fully -mechanized mining face. *Coal Science and Technology*, 50(1):1-27.

Hu, Y., Wang, X., Hu, J., Gong, J., Wang, K., Li, G., Mei, C. 2021. An overview on unmanned vehicle technology in off-road environment. *Trans. Beijing Inst. Technol.*, 41, 1137–1148.

Huber, D.F., Vandapel, N. 2006. Automatic three-dimensional underground mine mapping. *Int. J. Robot. Res.*, 25, 7–17.

Zhang, J., Kaess, M., Singh, S. 2016. On degeneracy of optimization-based state estimation problems. *In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA)*, 809-816.

Liu, Z., Zhang, F. 2021. Balm: Bundle adjustment for lidar mapping. *IEEE Robot. Autom. Lett.*, 6, 3184–3191.

Qin, T., Li, P., Shen, S. 2018. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.*, 34, 1004–1020.

Ren, Z., Wang, L., Bi, L. 2019. Robust GICP-based 3D LiDAR SLAM for underground mining environment. *Sensors*, 19, 2915.

Rusu, R., Bradski, G., Thibaux, R., Hsu, J. 2010. Fast 3d recognition and pose using the viewpoint feature histogram. *In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, 18–22 October 2010; pp. 2155–2162.

Segal, A., Haehnel, D., Thrun, S. 2009. Generalized-icp. *In Proceedings of the Robotics: Science and Systems V*, University of Washington, Seattle, WA, USA, 28 June–1 July 2009.

Shan, T., Englot, B., Meyers, D., Wang, W., Ratti, C., Rus, D. 2020. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. *In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA, 24 October–24 January 2021; pp. 5135–5142.

Wang, G., Zhao, G., Ren, H. 2019. Analysis on key core technologies of smart coal mine and intelligent mining. *Chinese Journal of Coal*, 44(01):34-41.

Wang, G. 2022. Discussion on the latest technological progress and problems of coal mine intelligence. *Coal Science and Technology*, 50(01):1-27.

Zhang, C., Zhang, X., Du, Y., et al. 2021. Measuring technique of cantilever roadheader position based on binocular stereo vision. *Coal Science and Technology*, 49(11):225-235.

Zhao, Y., Ji, Q., Wang, T. 2021. Current status and prospects of intelligent trackless auxiliary transportation technology in coal mines. *Coal Science and Technology*, 49(12):209 − 216.

Zhao, S., Fang, Z., Li, H., Scherer, S. 2019. A robust laser-inertial odometry and mapping method for large-scale highway environments. *In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, 3– 8 November 2019; pp. 1285–1292.

Zhou, Z., Cao, J., Di, S. 2021. Overview of 3D Lidar Slam Algorithms. *Chin. J. Sci. Instrum.*, 42, 13–27.