









P2d-DO: Degeneracy Optimization for LiDAR SLAM With Point-to-Distribution Detection Factors

Weinan Chen , Senior Member, IEEE, Sehua Ji , Member, IEEE, Xubin Lin , Zhi-Xin Yang , Member, IEEE, Wenzheng Chi , Senior Member, IEEE, Yisheng Guan , Haifei Zhu , Member, IEEE, and Hong Zhang , Life Fellow, IEEE

Abstract—Although the LiDAR SLAM technique has been already widely deployed on various robots, it may still suffer from degeneracy caused by inadequate constraints in scenes with sparse geometric features. If the degeneracy is not detected and properly processed, the accuracy of localization and mapping will significantly decrease. In this letter, we propose the P2d-DO method, which consists of a point-to-distribution degeneracy detection algorithm and a point cloud-weighted degeneracy optimization algorithm, to relieve the negative impact of degeneracy. The degeneracy detection algorithm outputs factors that characterize the degeneracy state by observing changes in the distribution probabilities within a local region. Factors reflecting the confidence of the point clouds are then fed to the degeneracy optimization algorithm, enabling the system to prioritize reliable point clouds by assigning larger weights during the matching process. Comprehensive experiments validate the effectiveness of our method, demonstrating significant improvements in both degeneracy detection and pose estimation in terms of accuracy and robustness.

Index Terms—LiDAR SLAM, degeneracy detection, degeneracy optimization, accuracy and robustness improvement.

Received 25 August 2024; accepted 9 December 2024. Date of publication 25 December 2024; date of current version 6 January 2025. This article was recommended for publication by Associate Editor T. Stoyanov and Editor J. Civera upon evaluation of the reviewers' comments. This work was supported in part by the Science and Technology Plan of Guangdong Province—Guangdong-Hong Kong-Macao Joint Innovation Project under Grant 2023A0505010016, in part by the National Natural Science Foundation of China under Grant 62103179, in part by the Shenzhen Science and Technology Plan under Grant SZXJP20230703093206015, and in part by the Science and Technology Development Fund, Macau SAR, under Grant 0092/2024/AFJ, Grant 0075/2023/AMJ, Grant 0003/2023/RIB1, and Grant 001/2024/SKL. (Corresponding author: Haifei Zhu.)

Weinan Chen, Yisheng Guan, and Haifei Zhu are with the State Key Laboratory of Precision Electronic Manufacturing Technology and Equipment, Guangdong University of Technology, Guangzhou 510006, China (e-mail: hfzhu@gdut.edu.cn).

Sehua Ji is with the State Key Laboratory of Precision Electronic Manufacturing Technology and Equipment, Guangdong University of Technology, Guangzhou 510006, China, and also with the JT-Innovation (Guangdong) Intelligent Technology Company Ltd., Foshan 528251, China.

Xubin Lin is with the Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, Guangzhou 510070, China.

Zhi-Xin Yang is with the State Key Laboratory of Internet of Things for Smart City, and Department of Electromechanical Engineering, University of Macau, Macau SAR, China.

Wenzheng Chi is with the Robotics and Microsystems Center, School of Mechanical and Electric Engineering, Soochow University, Suzhou 215137, China.

Hong Zhang is with the Shenzhen Key Laboratory of Robotics and Computer Vision, Southern University of Science and Technology, Shenzhen 518055, China.

This letter has supplementary downloadable material available at <https://doi.org/10.1109/LRA.2024.3522839>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2024.3522839

I. INTRODUCTION

THE current mainstream LiDAR SLAM heavily rely on LiDAR's ability to detect geometric structural features in the environment. Consequently, these methods suffer from degeneracy when applied in scenes with sparse geometric features, such as tunnels, long straight corridors or open areas [1]. In such degeneracy scenes, the pose estimation accuracy of LiDAR SLAM is significantly reduced due to the lack of constraints in certain directions.

For the degeneracy issue, various approaches [2], [3], [4] have been proposed to address the lack of constraints in degeneracy scenes by fusing heterogeneous sensor data. However, these methods don't adequately analyze the characteristics of the degeneracy environment or the system's degeneracy state, resulting in less robust algorithms and, at times, unpredictable results. Consequently, extensive research has been conducted on degeneracy detection, aiming to provide valuable information for degeneracy optimization. For example, the prevailing method [5] relies on point-to-point matching. However, this matching-based method is susceptible to false detections owing to its low noise resistance.

In this letter, we propose the P2d-DO method, which comprises two key components: a point-to-distribution (P2d) algorithm that leverages local geometric features to enhance degeneracy detection and a point cloud-weighted degeneracy optimization (DO) algorithm that improves pose estimation performance in the presence of degeneracy. To ensure that local geometric features adhere to the Gaussian distribution assumption, which is essential for accurate degeneracy detection, an adaptive voxel segmentation (AVS) algorithm is integrated with the P2d algorithm. The degeneracy factors output by the P2d algorithm, as illustrated in Fig. 1, are then used in the DO algorithm to mitigate the impact of low-confidence point clouds, thereby enhancing the accuracy and robustness of LiDAR SLAM pose estimation. The main contributions are as follows:

- 1) An adaptive voxel segmentation algorithm based on distribution density is presented to guarantee the quality of local geometric features.
- 2) A point-to-distribution degeneracy detection algorithm is proposed for effectively reducing noise interference and avoiding false detections by incorporating local geometric features.
- 3) A point cloud-weighted degeneracy optimization algorithm is proposed to relieve the impact of degenerate point clouds in pose estimation of LiDAR SLAM.

The rest of this letter is organised as follows. Related work on degeneracy optimization, degeneracy detection, and local

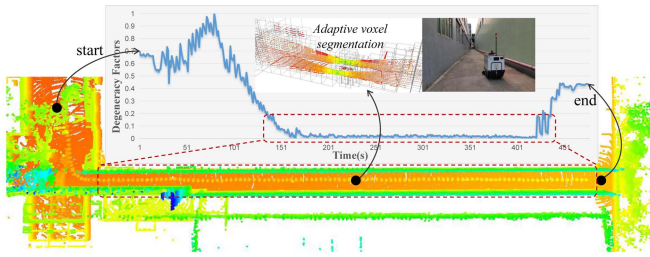


Fig. 1. Example of LiDAR SLAM in the corridor considering degeneracy, where the degeneracy factors decreases as the robot progresses into the degeneracy scene.

geometric feature extraction is presented in Section II. The details of the proposed P2d-DO method are described in Section III. Extensive experiments to verify and evaluate our proposed method are presented in Section IV. Finally, the conclusion and the future work are outlined in Section V.

II. RELATED WORK

A. Pose Estimation in Degeneracy Scenes

Extensive research has led to the development of numerous methods addressing degeneracy in LiDAR SLAM. These methods can be classified into two categories. As degeneracy arises from insufficient constraints, the first category is to directly integrate additional prior information. For instance, a method incorporating a visual-inertial system (VIS) and a LiDAR-inertial system (LIS) ensures proper functioning even if one of the systems fails [3]. However, the method is limited by the performance of other sensors, for example, visual sensors may suffer from decreased accuracy and robustness due to variant light intensity. By analyzing the point cloud reflectivity, a method fusing point cloud intensity was introduced in [6] to enhance robustness in degeneracy. However, this method is limited to intensity feature-rich scenes. Similarly, a multi-robot collaborative approach was proposed in [7] to compensate for the lack of constraints. Although an IMU-based backpropagation process was proposed in [4] to compensate for motion distortion in LiDAR sampling, the lack of analysis of the environment or states prevents appropriate adjustments and optimizations under degeneracy conditions.

Alternatively, the second category provides optimization directions and reliable information for suppressing negative impacts by detecting the system's degeneracy state, as exemplified in [1], [5] and [8]. The effectiveness of these methods relies significantly on the accuracy of degeneracy detection. However, existing degeneracy detection methods still encounter challenges such as insufficient accuracy, limited robustness, and false detections, which inevitably introduce new problems to LiDAR SLAM.

B. Degeneracy Detection

The degeneracy state can be detected with prior data of other modalities. For example, a degeneracy representation model that utilizes prior point cloud maps to estimate locatability was proposed in [9]. However, map-based methods require pre-modeling of the environment, making them inadequate for the challenge of localization in unknown environments.

Additionally, although the degeneracy state of LiDAR SLAM can be detected by aligning data from UWB and LiDAR in [8], this method requires advanced deployment in the environment, limiting its applicability. While the usage of geomagnetic sensors was proposed in [10], new sources of noise may be introduced by metallic objects in the environment.

Deep learning-based degeneracy detection is another promising solution. For 2D LiDAR, a method for improving the estimation accuracy and reliability by extending the traditional fixed covariance Gaussian measurement model was described in [11]. Extending this idea to 3D, a sensor data-driven learning method was proposed in [12] to estimate uncertainty in the form of covariance within iterative closest point registration. However, these methods filter out features, thereby limiting the utilization of other available data. To address this limitation, a method that trains and analyzes degeneracy states based on raw LiDAR data to improve the detection accuracy and robustness by learning higher-level features was presented in [13]. Nevertheless, deep learning-based approaches face significant challenges in terms of generalizability and real-time performance, making them difficult to adapt to complex and dynamic unknown environments.

Eigenvalue analysis-based degeneracy detection has gained wide popularity for its universality and extensibility [14]. Utilizing the Fisher matrix, an eigenvalue-based method was proposed in [15] to determine the degeneracy state. Furthermore, a localization capability evaluation method that combines local geometric models with Fisher matrices was proposed in [16]. Similarly, a vector estimation method based on principal component analysis was proposed in [17]. However, these methods only provide a simple assessment of degeneracy without accurately quantifying the uncertainty of pose estimation in each direction. To address this issue, in [5], Zhang et al. proposed a degeneracy factor that effectively reflects the uncertainty of pose estimation based on the Hessian matrix. A similar approach was presented in [18]. However, these methods rely on point-to-point matching, making them susceptible to false detection caused by noise. Although an improved matching method was proposed in [1] to enhance degeneracy detection accuracy, distance residuals between feature points still rely upon judging the degeneracy state, leaving the issue of false detection unresolved.

A common approach to addressing the noise issue involves translating the association of feature points into an analysis of the local observation model, as demonstrated in [19]. However, these methods overlook the consideration of whether the point cloud data meet the Gaussian distribution during the extraction of local features. Using Gaussian models for non-Gaussian distributed point cloud data can lead to information loss and inaccurate results. Therefore, it is crucial to ensure that the local geometric features comply with the Gaussian assumption.

C. Extracting Local Geometric Features

Local geometric features can be extracted through point cloud segmentation. A common method is to segment the point cloud into uniform voxel grids, as mentioned in [20]. However, this method has the drawback of multiple peaks in the point clouds distribution within voxels, which breaks the Gaussian distribution assumption. To overcome this limitation, a novel adaptive voxelization method relying solely on covariance matrix information was proposed in [21]. However, the risk of misclassification increases when extracting plane or edge features. Additionally, an adaptive voxel filter was proposed in [22],

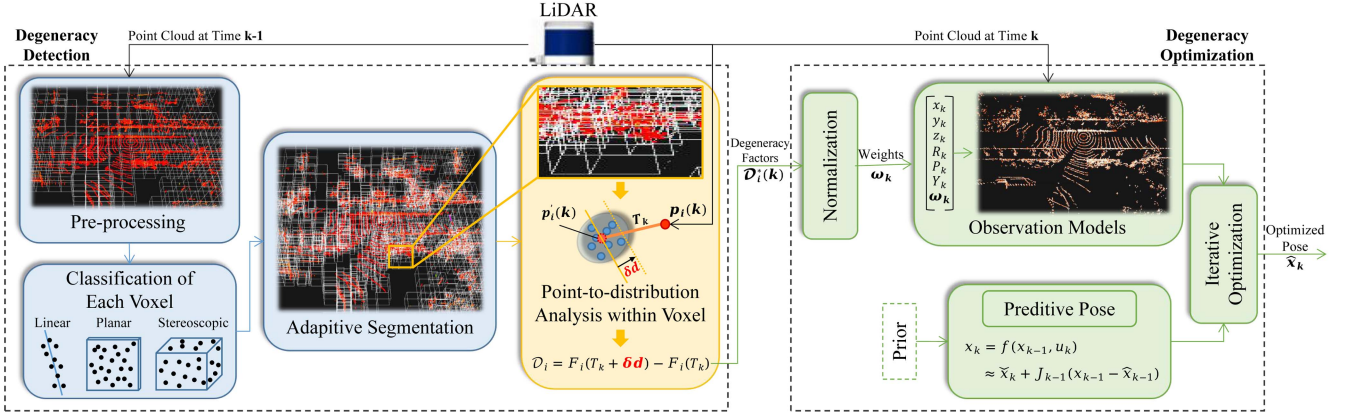


Fig. 2. Framework of our proposed P2d-DO method. It comprises three key modules: AVS, P2d and DO, distinguished in blue, yellow and green, respectively. The AVS and P2d modules together form the degeneracy detection algorithm.

which addresses the computational load of the system but does not resolve the issue of multiple peaks.

The local geometric features can also be obtained through point cloud clustering. The most common method, k-means, was proposed in [23]. However, this method requires pre-setting the value of k , leading to significant uncertainty in unknown scenes. An enhanced version, k-means++, was presented in [24]. But it demands higher computational resources as the dimensionality of the point cloud data increases. To improve the instability of k-means, a split greedy clustering method was proposed in [25], and a region growth-based method that prioritizes spatial continuity was presented in [26]. Nevertheless, these methods often fail to meet real-time requirements when handling large-scale point cloud data in SLAM systems.

Deep learning-based methods have demonstrated strong performance in addressing the issue of multi-peak distributions. For example, the RandLA-Net proposed in [27] showcases an efficient and lightweight neural network architecture capable of analyzing and inferring semantic information from large-scale point clouds. Similarly, a point cloud segmentation network was introduced in [28] by analyzing the interaction and fusion between different scales. However, significant limitations in terms of real-time performance and generalization capabilities were found and reported in these learning methods.

III. METHODOLOGY

A. Overview of P2d-DO

Fig. 2 depicts the overview of the proposed P2d-DO method for addressing the degeneracy issues in LiDAR SLAM. Its framework comprises three modules: AVS, P2d, and DO, distinguished in blue, yellow and green, respectively. The AVS module discretizes the point cloud and adaptively adjusts voxel size based on point cloud distribution density. This process enhances the accuracy of local geometric features by refining spatial distribution within each voxel. The P2d module uses the extracted local geometric features to observe changes in the probability distribution of point clouds between consecutive frames within a local region. Consequently, the P2d module outputs the degeneracy factors that quantify the degeneracy degree of point clouds, providing important basis for subsequent optimization processing. The DO module uses degeneracy

factors to assess the confidence level of all point clouds. Point clouds with higher degeneracy are down-weighted so that their impacts on the overall pose estimation are weakened, and accordingly, the accuracy and robustness of the pose estimation of LiDAR SLAM is enhanced.

B. AVS Module

Our voxel segmentation algorithm adapts to environmental changes in real time and consists of three steps. *Step 1*: the point cloud is discretized into uniform voxels. *Step 2*: following the approach proposed in [29], the resulted voxels are classified into three geometric models: linear, planar, and stereoscopic. *Step 3*: sparse regions are detected, and segmentation strategies are dynamically adjusted based on the distribution density of different geometric models. Then segmentations are performed to obtain new voxels. The above three steps are repeated until no sparse region is identified.

Considering that the stereoscopic model is difficult for analysis due to its complex spatial distribution, we simplify the process by dividing the voxel into eight equal parts and then returning to *Step 2* to reclassify the segmented voxel. Consequently, our approach only considers linear and planar models. The feature of linear model is that point clouds primarily distributes in a specific direction. This direction can be determined by analyzing the covariance matrix of the voxels, as depicted in Fig. 3(a), where the vector ν_i represents the principal direction of point cloud distribution. Subsequently, all points are projected onto the principal direction, as shown in Fig. 3(b). The density of point cloud distribution in this direction is evaluated by measuring the distance between each projected point and its two neighboring points. In Fig. 3(b), the blue segments in the principal direction indicate the dense regions, while the in-between purple ellipse (or the yellow segment) separated by the boundaries of dense regions is considered as the sparse region. The centroid c of the sparse region is identified as the segmentation point, with segmentation lines aligned along the x , y , and z axes and go through this point. Before segmentation, overlap detection is performed by measuring the distance between points and segmentation lines. As shown by the red dotted line in Fig. 3(c), if more than five points in any given direction are found to be near the segmentation line, the corresponding segmentation is discarded. Fig. 3(d) illustrates the AVS result, which is a more concentrated

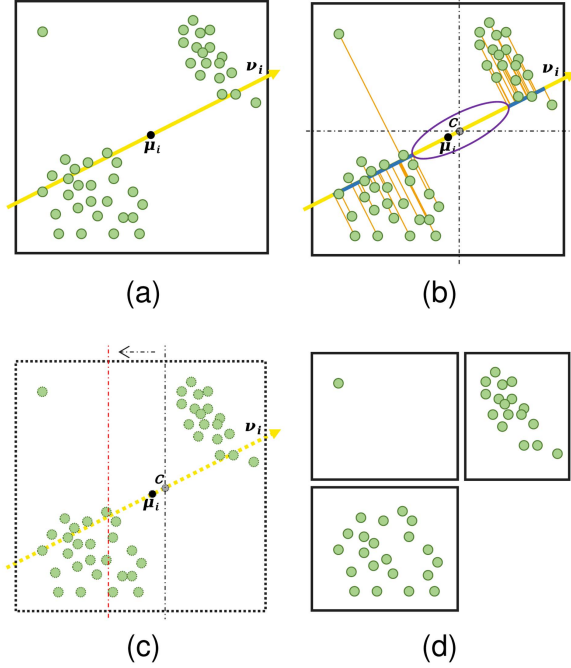


Fig. 3. Schematic diagram of the AVS module, where μ_i and ν_i represent the mean and the principal direction of the point cloud, and c is the centroid of the sparse region. (a) Distribution direction. (b) Density sparse region detection. (c) Overlap detection. (d) Illustration of AVS Results.

distribution of points within each voxel and complies with the Gaussian distribution assumption.

The segmentation operation for planar models is similar to that for linear models, with the key difference being that the voxels of the planar model are projected onto a plane defined by two principal directions. Sparse regions are then extracted from the projected plane to perform voxel segmentation.

C. P2d Module

Degeneracy scenes are typically characterized by structural symmetry, resulting in geometrical features extracted from LiDAR point clouds remain nearly invariant. Motivated by this observation, we hypothesize that the accuracy of degeneracy detection can be improved by leveraging the inherent characteristics of degeneracy scenes. Hence, our proposed P2d module integrates variations in local geometric models with registration residuals. Assuming that the mean μ and the covariance matrix Σ represent the distributional properties of the point cloud within the local geometric model, the distributional probability of each point within the voxel at its corresponding location can be calculated as:

$$S_i(T_k) = \frac{1}{\sqrt{2\pi}\sqrt{|\Sigma|}} \exp\left(-\frac{(p'_i - \mu)^T \Sigma^{-1} (p'_i - \mu)}{2}\right), \quad (1)$$

where T_k denotes the transformation increment between the source point cloud and the target point at time k , and p'_i represents the i -th transformed point cloud. By applying logarithmic operations and omitting the constant term, (1) can be converted into:

$$F_i(T_k) = e_i^T(T_k) \Sigma^{-1} e_i(T_k), \quad (2)$$

where $e_i(T_k) = p'_i - \mu$ and $F_i(T_k) = -2\ln S_i(T_k)$.

To simulate the degeneracy state, a perturbation $\delta d = (d_x, d_y, d_z, d_R, d_P, d_Y)$ is introduced to this model. Therefore, the distribution probability considering the perturbation δd becomes:

$$\begin{aligned} F_i(T_k + \delta d) &= -2\ln S_i(T_k + \delta d) \\ &= e_i^T(T_k + \delta d) \Sigma^{-1} e_i(T_k + \delta d). \end{aligned} \quad (3)$$

Using the Taylor expansion, (3) can be transformed as:

$$\begin{aligned} F_i(T_k + \delta d) &\approx (e_i(T_k + \delta d) + J_i \delta d)^T \Sigma^{-1} (e_i(T_k + \delta d) + J_i \delta d) \\ &= e_i^T \Sigma^{-1} e_i + 2e_i^T \Sigma^{-1} J_i \delta d + \delta d^T J_i^T \Sigma^{-1} J_i \delta d, \end{aligned} \quad (4)$$

where J_i denotes the Jacobian matrix reflecting the correlation between the observation model and the position. The (4) can be further simplified as:

$$\begin{aligned} F_i(T_k + \delta d) &= e_i^T \Sigma^{-1} e_i + 2b_i^T \delta d + \delta d^T H_i \delta d \\ &= F_i(T_k) + 2b_i^T \delta d + \delta d^T H_i \delta d, \end{aligned} \quad (5)$$

where $b_i^T = e_i^T \Sigma^{-1} J_i$ and $H_i = J_i^T \Sigma^{-1} J_i$. The approximate Hessian matrix H_i , constructed from the Jacobian matrix, is symmetric and at least semi-positive definite. The changes in the distribution probability after adding the perturbation can be calculated as:

$$\begin{aligned} D_i &= F_i(T_k + \delta d) - F_i(T_k) \\ &= 2\ln \frac{S_i(T_k)}{S_i(T_k + \delta d)} = 2b_i^T \delta d + \delta d^T H_i \delta d. \end{aligned} \quad (6)$$

It is implied by (6) that larger perturbations result in greater uncertainty, which decreases the distribution probability $S_i(T + \delta d)$ and consequently increases D_i . In this study, we define the degeneracy factor as the maximum of D_i :

$$D_i^* = \max_{\delta d} (2b_i^T \delta d + \delta d^T H_i \delta d), \quad (7)$$

where $2b_i^T \delta d$ is a linear term with respect to δd , while $\delta d^T H_i \delta d$ is a convex quadratic term which dominates in solving D_i^* . Ignoring the linear term in (7) yields:

$$\begin{aligned} D_i^* &= \max_{\delta d} (\delta d^T H_i \delta d) \\ &= \max_{\delta d} \frac{\delta d^T H_i \delta d}{\delta d^T \delta d} \delta d^T \delta d. \end{aligned} \quad (8)$$

According to the Rayleigh quotient [30], as H_i is a symmetric matrix, the maximum of the fraction in (8) is equal to the largest eigenvalue of H_i . Therefore, the degeneracy factors can be calculated as:

$$D_i^* = a_i \lambda_i^{\max}, \quad (9)$$

where $a_i = \delta d^T \delta d$ represents the constant term and λ_i^{\max} denotes the largest eigenvalue of H_i . a_i can be calculated from the registration error of the interframe point cloud in practice. Consequently, the j -th eigenvalue λ_i^j determines the degree of degeneracy in the direction represented by the corresponding eigenvector v_i^j of H_i , where $j = 1, 2, \dots, 6$.

Accordingly, the degeneracy state of the system containing n point clouds at time k can be defined as:

$$D(k) = \sum_{i=1}^n D_i^*. \quad (10)$$

We can determine whether the system encounters degeneracy problems by checking $D(k)$. In this letter, we preset a fixed threshold D_{thre} for detecting system degeneracy.

D. DO Module

Suppressing the function of point clouds in degeneracy states should be beneficial for reducing the pose estimation errors of LiDAR SLAM systems. Adjusting the weights associated with point clouds provides a straightforward way. Let D'_i denote the degeneracy confidence of the i -th point. It can be obtained by applying the max-min normalization to the degeneracy factors at time k , such as:

$$D'_i = \frac{D_i^* - D_{\min}}{D_{\max} - D_{\min}}, \quad (11)$$

where D_{\min} and D_{\max} denote the minimum and maximum degeneracy factors, respectively. After normalization, as high degeneracy confidence corresponds to small significance weights, the weighting factor of each point at time k can be assigned as:

$$\omega_k = \{1 - D'_i | i = 1, 2, \dots, n\}. \quad (12)$$

Integrated into the Extended Kalman Filter [31], the weighting factors ω_k are applied to the observation model in real time. This operation is implemented by two steps: prediction and update.

1) *Prediction*: The prediction step involves estimating the system state x_k and the associated uncertainty \tilde{P}_k at time k , based on the previous state estimate x_{k-1} . As the state estimation is non-linear, it is necessary to linearize the non-linear equations using the first-order Taylor expansion to predict the positional information, with the form as,

$$x_k = f(x_{k-1}, u_k) \approx \tilde{x}_k + J_{k-1}(x_{k-1} - \hat{x}_{k-1}), \quad (13)$$

where \hat{x}_{k-1} and \tilde{x}_k denote the optimal pose of the previous frame and the predicted pose of the current frame, respectively, J_{k-1} represents the Jacobian matrix of the state equation for the previous frame and u_k is the acceleration or steering angle of the robot.

In addition, predicting the uncertainty of the next pose is crucial for better guiding the update direction of the filter. The error covariance \tilde{P}_k , representing the state uncertainty, can be derived by utilizing J_{k-1} as follows:

$$\tilde{P}_k = F_{k-1} \hat{P}_{k-1} J_{k-1}^T + Q_k, \quad (14)$$

where F_{k-1} describes the state evolution of the system from time step $k-1$ to k , and Q_k is the covariance matrix of the measurement noise.

2) *Update*: The state estimates and associated uncertainties are refined using the observation from LiDAR at time k . Similarly, the nonlinear observation equation for the point cloud is linearized using the Taylor expansion:

$$y_k = g(x_k, n_k) \approx \omega_k \tilde{y}_k + H_k(x_k - \tilde{x}_k), \quad (15)$$

where n_k denotes the observation noise, \tilde{y}_k represents the predicted observation obtained from motion model and prior map matching, and H_k denotes the Jacobian matrix of the observation

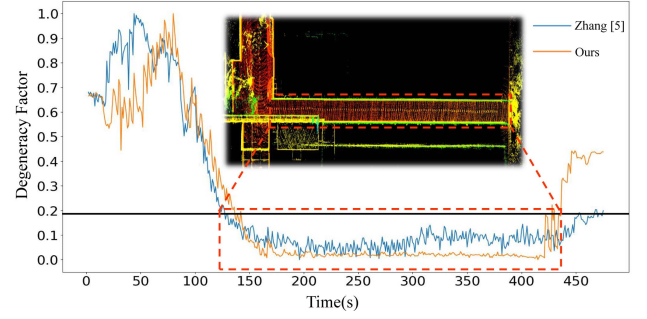


Fig. 4. Results of normalized degeneracy factors in the corridor scenario.

equation. In (15), the weighting factors ω_k designate the role that each point can play in the matching process, prioritizing points with low degeneracy confidence, while suppressing those with high degeneracy confidence.

The Kalman gain can be calculated as:

$$K_k = \tilde{P}_k H_k^T (H_k \tilde{P}_k H_k^T + R_k)^{-1}, \quad (16)$$

where R_k represents the noise matrix. The Kalman gain K_k and the difference between the actual observations y_k and the weighted predicted observations $\omega_k \tilde{y}_k$ are used to update the state estimates, as follows:

$$\hat{x}_k = \tilde{x}_k + K_k(y_k - \omega_k \tilde{y}_k). \quad (17)$$

Simultaneously, the error covariance matrix \hat{P}_k is updated to reflect the state estimate improved:

$$\hat{P}_k = (I - K_k H_k) \tilde{P}_k. \quad (18)$$

It is worth noting that our proposed method can be extended to multi-sensor fusion SLAM systems with LiDAR, as additional sensor data can be integrated in the prediction step.

IV. EXPERIMENTS

To validate and evaluate our proposed method, we have conducted two groups of experiments, one for degeneracy detection and another for degeneracy optimization. The source codes of our algorithms can be downloaded at <https://github.com/jisehua/Degenerate-Detection.git>.

A. Degeneracy Detection

1) *Hyper-Parameter Setting*: In this group of experiments, we used [5] as the baseline, whose keynote is that “the smaller the factor, the greater the degeneracy,” which is opposite to our method. Therefore, we adjusted the output of our P2d module by taking its opposite ($-D_i^*$) to align with [5]. Subsequently, the results of both methods were normalized for fair comparison. Fig. 4 depicts the degeneracy factors output by the two methods corresponding to a corridor scene. From the figure, the degeneracy factors are typically below 0.2 when the robot enters the corridor where degeneracy occurs. Consequently, D_{thre} was set as 0.2 for degeneracy detection in the subsequent experiments.

2) *Adaptive Voxel Segmentation*: To verify the effectiveness of the AVS module in degeneracy detection, we selected M2DGR street03 [32] and KITTI 03 [33] with rich geometrical features in campus and urban and conducted tests with uniform voxel segmentation [20] and AVS (proposed), respectively.

TABLE I
FALSE DETECTION RESULTS REPORTED IN TESTS RUNNING WITH AND WITHOUT AVS MODULE

	M2DGR street03	KITTI 03	False detection rate
Without AVS [20]	26	48	9.02%
With AVS	9	6	1.83%

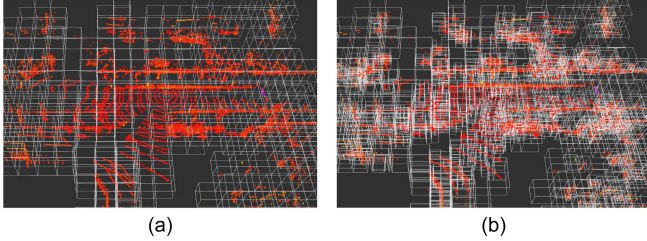


Fig. 5. Comparison of the effect of voxel segmentation. (a) Uniform voxel segmentation described in [20]. (b) AVS module proposed in this letter.

TABLE II
DATASET CLASSIFICATION

Datasets	Contains degeneracy	Non-degeneracy
M2DGR street	02	01
	04	03
	05	06
KITTI	01	03
	04	05
	06	-
Self-collected	Corridor Factory	Residential Park

Theoretically, there should be not any degeneracy detected in these scenes. From the Table I, the false detection rate is 9.02% without AVS, and decreases to 1.83% with AVS. Fig. 5 shows the different effects of voxel segmentation with and without AVS, affirming the positive contribution of the AVS module. When the number of point clouds in a frame is 28040 and the framerate is 10 Hz, the AVS processing time is 47.56 ms, which satisfies the real-time requirement.

3) *Performance in Non-Degeneracy Scenes*: To access the robustness and false detection rate in non-degeneracy scenes, we conducted dedicated tests on self-annotating datasets including M2DGR street and KITTI, and self-collected datasets. When labeling the dataset, a frame is annotated as degeneracy if the angular change of the point cloud's normal vector is below 5° and the odometry error of SLAM exceeds a threshold distance determined by the robot's speed (to filter out misjudgments caused by robots being stationary). The classification outcomes are presented in Table II.

The standard deviation σ was employed as a quantitative metric for evaluating the robustness in non-degeneracy scenes. A smaller standard deviation indicates more concentrated data and resistant to noise interference. Additionally, the number of false detections (FDs) identified based on D_{thre} was counted. Table III summarizes the test results. Compared to the baseline Zhang's method [5], our proposed P2d module achieved a reduction in standard deviation ranging from 9.01% to 21.83%, indicating an average improvement of 15.86% across seven non-degeneracy scenes. Moreover, the number of false detections

was significantly reduced with the P2d module, from 75.00% to 93.6% (excluding cases of no false detections), demonstrating its effectiveness in mitigating noise interference. These results show that our proposed P2d module achieved notable robustness improvements over the baseline without introducing new issues in non-degeneracy scenes.

4) *P-R Test in Degeneracy Scenes*: To assess the performance in degeneracy scenes, we conducted Precision-Recall comparison tests following [34]. In this test, non-degeneracy is designated as the positive class and degeneracy as the negative class. Therefore, precision reflects the algorithm's accuracy, while recall assesses its capability to identify degeneracy states. High precision and recall indicate that the algorithm achieves comprehensive and reliable results. The test results are summarized in Table IV. As shown in the table, the precision and recall of our proposed P2d module were 0.918% and 0.981% on average, improved by 7.61% and 6.34%, respectively, compared with the baseline. These findings indicate that the P2d module demonstrated a stronger ability to identify degeneracy states and detect them accurately.

In the first group of experiments, our algorithm demonstrated outstanding accuracy and robustness, effectively reducing false detections and offering superior resistance to noise.

B. Degeneracy Optimization

1) *Localization Accuracy*: To verify the effectiveness of the DO module, the entire P2d-DO method was integrated into LOAM [35] and LEGO-LOAM [36], resulting in DO-LOAM and DO-LEGO-LOAM. It is worth noting that these systems are typical LiDAR-only SLAM systems that rely solely on LiDAR data as input, allowing for an intuitive verification of the advantages of our algorithm. In the experiment, different methods were tested on public dataset and their localization accuracies were recorded. The Root Mean Square Error (RMSE) calculated based on the Absolute Trajectory Error (ATE) was used to access the localization accuracy.

Table V shows the experimental results. From the results, it can be observed that applying the P2d-DO method enables both LOAM and LEGO-LOAM to achieve better accuracy, with an average improvement of 3.92% over the original methods. This improvement demonstrates that the proposed P2d-DO method effectively suppresses interference caused by observation information in degeneracy states, thereby enhancing the system's localization performance.

Furthermore, the P2d-DO method was also integrated with LIO-SAM [2] and FAST-LIO2 [4] to assess its generalization. The experimental results presented in Table VI confirm that the localization accuracy of the multi-sensor SLAM systems improved after integrating the P2d-DO method. Notably, our method was applied to four mainstream algorithms and demonstrated improvements in all six tested scenes, improving localization accuracy by an average of **5.38%**. These findings highlight the strong generalization capability of the proposed P2d-DO method across various LiDAR SLAM systems.

2) *Real-Vehicle Experiment*: In this experiment, we deployed the DO-LIO-SAM on our self-developed mobile robot, and investigated its performance. Although the localization ground truth with our self-developed mobile robot in outdoor (as shown in Fig. 6) is difficult to obtain, the performance can still be identified by analyzing the mapping quality.

TABLE III
RESULTS OF ROBUSTNESS TESTS IN NON-DEGENERACY SCENES (σ : STANDARD DEVIATION, FDS: NUMBER OF FALSE DETECTIONS)

	M2DGR street						KITTI				Self-collected			
	01		03		06		03		05		Residential		Park	
	σ	FDS	σ	FDS	σ	FDS	σ	FDS	σ	FDS	σ	FDS	σ	FDS
Zhang [5]	0.170	125	0.164	36	0.197	22	0.239	34	0.178	0	0.198	6	0.211	0
Ours	0.151	8	0.133	9	0.154	7	0.194	6	0.149	0	0.168	1	0.192	0
Improvement	11.18%	93.60%	18.95%	75.00%	21.83%	68.18%	18.83%	82.35%	16.08%	0	15.15%	83.33%	9.01%	0

TABLE IV
RESULTS OF PRECISION-RECALL COMPARISON TESTS

		Precision		Recall	
		Zhang [5]	Ours	Zhang [5]	Ours
M2DGR street	02	1.000	0.946	0.931	0.997
	04	0.905	0.934	0.996	0.997
	05	0.589	0.896	0.997	0.998
KITTI	01	0.597	0.774	0.950	0.994
	04	0.912	0.969	1.000	1.000
	06	1.000	1.000	0.927	0.958
Self-collected	Corridor	0.907	0.932	0.718	0.960
	Factory	0.875	0.893	0.832	0.945

TABLE V
RMSE OF LOAM [35], LEGO-LOAM [36] AND THEIR VARIANTS INTEGRATING P2D-DO

	M2DGR street			KITTI		
	02	04	05	01	04	06
LOAM	23.576	0.915	0.361	21.198	0.231	5.124
DO-LOAM	22.721	0.853	0.352	20.148	0.216	5.050
LEGO-LOAM	3.175	1.002	0.351	94.674	0.719	3.430
DO-LEGO-LOAM	3.073	0.969	0.342	90.152	0.675	3.387

TABLE VI
RMSE OF LIO-SAM [2] AND FAST-LIO2 [4] AND THEIR VARIANTS INTEGRATING P2D-DO

	M2DGR street			KITTI		
	02	04	05	01	04	06
LIO-SAM	3.415	0.993	0.342	64.145	0.575	17.928
DO-LIO-SAM	3.307	0.854	0.336	53.636	0.574	14.811
FAST-LIO2	2.289	0.457	0.314	69.155	0.712	12.160
DO-FAST-LIO2	2.223	0.413	0.308	59.734	0.709	12.053

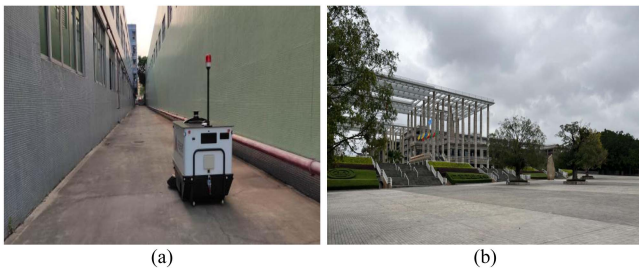


Fig. 6. Scenes for real-vehicle experiments. (a) Corridor, unidirectional degeneracy. (b) Plaza, multidirectional degeneracy.

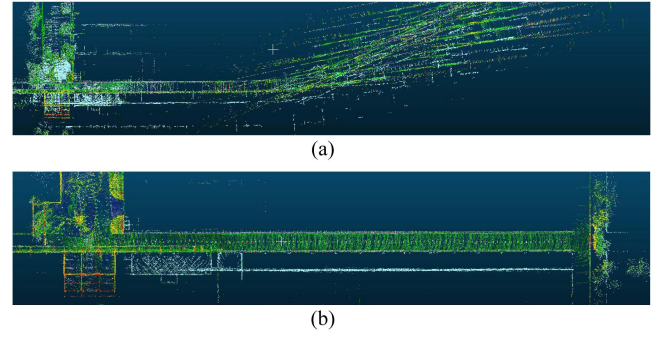


Fig. 7. Mapping results in a corridor scene. (a) LIO-SAM. (b) DO-LIO-SAM (integrated with P2d-DO).

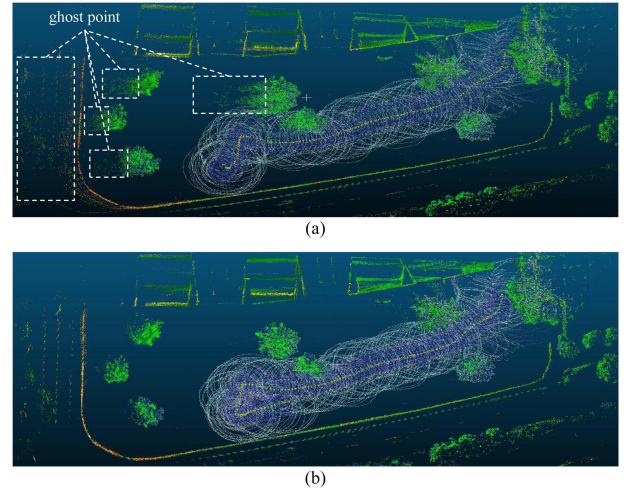


Fig. 8. Mapping results in an open plaza scene. (a) LIO-SAM. (b) DO-LIO-SAM (integrated with P2d-DO).

Fig. 7 presents the mapping results in the corridor scene. As demonstrated in Fig. 7(b), DO-LIO-SAM effectively improved the mapping stability of LiDAR SLAM. In contrast, Fig. 7(a) showcases that LIO-SAM suffered from significant localization errors, leading to mapping failure. In the plaza scene, as depicted in Fig. 8(a), the mapping suffered from noticeable ghosting, which severely impacted the robot's localization and navigation, if the degeneracy state was not properly processed. However, after integrating our method, as shown in Fig. 8(b), the mapping quality is significantly improved.

In the second group of experiments, our P2d-DO method demonstrated its contribution in enhancing the SLAM accuracy and mapping quality, particularly in degeneracy scenes.

V. CONCLUSION

In this study, we proposed a LiDAR SLAM degeneracy optimization method, the P2d-DO, with degeneracy state awareness to mitigate the impact of degeneracy on localization and mapping accuracy. Specifically, the P2d method reduces noise interference by incorporating local geometric features with the adaptive voxel segmentation technique, and outputs detection factors. The experiments demonstrated significant advancements in robustness for non-degeneracy scenes and detection accuracy for degeneracy scenes. Based on the detection factors, the DO module accesses the degeneracy confidence of point clouds, and controls their impacts by assigning proper weightings. Experimental results verified that the P2d-DO method improves localization accuracy by an average of 5.38%. However, the P2d module still relies on a preset threshold to detect the system degeneracy state, which may limit its generalization. In the future, we will focus on developing adaptive threshold adjustment methods to further enhance the P2d's adaptability and accuracy. Additionally, we plan to explore more efficient multi-sensor fusion methods integrating P2d-DO to further improve the performance and stability of the LiDAR SLAM systems.

REFERENCES

- [1] T. Tuna et al., "X-ICP: Localizability-aware LiDAR registration for robust localization in extreme environments," *IEEE Trans. Robot.*, vol. 40, pp. 452–471, 2024.
- [2] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled LiDAR inertial odometry via smoothing and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5135–5142.
- [3] T. Shan, B. Englot, C. Ratti, and D. Rus, "LVI-SAM: Tightly-coupled LiDAR-visual-inertial odometry via smoothing and mapping," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 5692–5698.
- [4] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.
- [5] J. Zhang, K. Michael, and S. Sanjiv, "On degeneracy of optimization-based state estimation problems," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 809–816.
- [6] Y. Zhang et al., "RI-LIO: Reflectivity image assisted tightly-coupled LiDAR-inertial odometry," *IEEE Robot. Automat. Lett.*, vol. 8, no. 3, pp. 1802–1809, Mar. 2023.
- [7] Y. Chang et al., "LAMP 2.0: A robust multi-robot slam system for operation in challenging large-scale underground environments," *IEEE Robot. Automat. Lett.*, vol. 7, no. 4, pp. 9175–9182, Oct. 2022.
- [8] H. Zhou, Z. Yao, and M. Lu, "UWB/LiDAR coordinate matching method with anti-degeneration capability," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3344–3352, Feb. 2021.
- [9] W. Zhen, S. Zeng, and S. A. Scherer, "Robust localization and localizability estimation with a rotating laser scanner," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 6240–6245.
- [10] Z. Wu, Y. Yue, M. Wen, J. Zhang, G. Peng, and D. W. Wang, "MSTSL: Multi-sensor based two-step localization in geometrically symmetric environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 5245–5251.
- [11] W. Vega-Brown, A. Bachrach, A. Bry, J. Kelly, and N. Roy, "CELLO: A fast algorithm for covariance estimation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2013, pp. 3160–3167.
- [12] D. Landry, F. Pomerleau, and P. Giguère, "CELLO-3D: Estimating the covariance of ICP in the real world," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 8190–8196.
- [13] J. Nubert, E. Walther, S. Khattak, and M. Hutter, "Learning-based localizability estimation for robust LiDAR localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 17–24.
- [14] S. Ji et al., "A point-to-distribution degeneracy detection factor for LiDAR SLAM using local geometric models," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 12283–12289.
- [15] M. Koizumi, K. Nonaka, and K. Sekiguchi, "Avoidance of singular localization environment using model predictive control for mobile robots," in *Proc. Asian Control Conf.*, 2017, pp. 2866–2871.
- [16] Y. Liu, J. Wang, and Y. Huang, "A localizability estimation method for mobile robots based on 3D point cloud feature," in *Proc. IEEE Int. Conf. Real-time Comput. Robot.*, 2021, pp. 1035–1041.
- [17] W. Shi, S. Li, C. Yao, Q. Yan, C. Liu, and Q. Chen, "Dense normal based degeneration-aware 2D LiDAR odometry for correlative scan matching," *IEEE Trans. Instrum. Meas.*, vol. 72, 2023, Art. no. 9500516.
- [18] J. Lv, X. Zuo, K. Hu, J. Xu, G. Huang, and Y. Liu, "Observability-aware intrinsic and extrinsic calibration of LiDAR-IMU systems," *IEEE Trans. Robot.*, vol. 38, no. 6, pp. 3734–3753, Dec. 2022.
- [19] M. Kondo, M. Hoshi, Y. Hara, and S. Nakamura, "Localizability estimation based on occupancy grid maps," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatron.*, 2022, pp. 368–373.
- [20] M. Magnusson, "The three-dimensional normal-distributions transform: An efficient representation for registration, surface analysis, and loop detection," Ph.D. dissertation, Örebro Univ., Örebro, Sweden, 2009.
- [21] Z. Liu and F. Zhang, "BALM: Bundle adjustment for LiDAR mapping," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 3184–3191, Apr. 2021.
- [22] A. Reinke et al., "LOCUS 2.0: Robust and computationally efficient LiDAR odometry for real-time 3D mapping," *IEEE Robot. Automat. Lett.*, vol. 7, no. 4, pp. 9043–9050, Oct. 2022.
- [23] J. A. Hartigan and M. A. Wong, "A k-means clustering algorithm," *J. Roy. Stat. Society, Ser. C. (Applied Statistics)*, vol. 28, pp. 100–108, 1979.
- [24] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, 2007, pp. 1027–1035.
- [25] J. S. Arun Das and S. L. Waslander, "Scan registration using the normal distributions transform with ground segmentation and point cloud clustering," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2013, pp. 2207–2212.
- [26] A. Das and S. L. Waslander, "Scan registration using segmented region growing NDT," *Int. J. Robot. Res.*, vol. 33, pp. 1645–1663, 2014.
- [27] Q. Hu et al., "RandLA-net: Efficient semantic segmentation of large-scale point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11105–11114.
- [28] D. Nie, R. Lan, L. Wang, and X. Ren, "Pyramid architecture for multi-scale processing in point cloud segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17263–17273.
- [29] S. Chen et al., "NDT-LOAM: A real-time LiDAR odometry and mapping with weighted NDT and LFA," *IEEE Sensors J.*, vol. 22, no. 4, pp. 3660–3671, Feb. 2022.
- [30] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
- [31] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proc. 1995 Amer. Control Conf. - ACC'95*, 1995, vol. 3, pp. 1628–1632.
- [32] J. Yin, A. Li, T. Li, W. Yu, and D. Zou, "M2DGR: A multi-sensor and multi-scenario SLAM dataset for ground robots," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2266–2273, Apr. 2022.
- [33] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, pp. 1231–1237, 2013.
- [34] D. M. Powers, "Evaluation: From precision, recall and f-measure to ROC, informedness, markedness and correlation," *J. Mach. Learn. Tech.*, vol. 2, no. 1, pp. 37–63, 2011.
- [35] J. Zhang and S. Sanjiv, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot., Sci. Syst.*, 2014, vol. 2, pp. 1–9.
- [36] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized LiDAR odometry and mapping on variable terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 4758–4765.