2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

A Point-to-distribution Degeneracy Detection Factor for LiDAR SLAM using Local Geometric Models

Sehua Ji[1,3], Weinan Chen[1*], Zerong Su[2], Yisheng Guan[1], Jiehao Li[4], Hong Zhang[5], Haifei Zhu[1]

***Abstract*— Limited by the working principles, LiDAR-SLAM systems suffer from the degeneration phenomenon in environments such as long corridors and tunnels, due to the lack of sufficient geometric features for frame-to-frame matching. The accuracy and sensitivity of existing degeneracy detection methods need to be further improved. In this paper, we propose a novel method for degeneracy detection using local geometric models based on point-to-distribution matching. To obtain an accurate description of local geometric models, an adaptive adjustment of voxel segmentation according to the point cloud distribution and density is designed. The codes of the proposed method is open-source and available at https://github.com/jisehua/Degenerate-Detection.git. Experiments with public datasets and self-build robots were conducted to evaluate the methods. The results exhibit that our proposed method achieves higher accuracy than the other existing approaches. Applying our proposed method is beneficial for improving the robustness of the LiDAR-SLAM systems.**

I. INTRODUCTION

LiDAR is a commonly used sensor for robot localization and mapping due to its exceptional capability to measure at long ranges and stability. A popular method for achieving data association in localization and mapping is by minimizing the matching of geometry features between consecutive frames. These algorithms rely on the geometrical information provided by LiDAR. Consequently, the ability to converge to an optimal solution depends on sufficient constraints in all directions [1]–[3]. However, in symmetrical environments like long corridors and tunnels, the constraints in specific directions become incomplete. This leads to anomalies in the optimization problem, which is referred to as the degeneracy phenomenon in LiDAR SLAM [4]. The degeneracy phenomenon in LiDAR SLAM remains a significant concern.

Degeneracy detection plays a crucial role in assessing localization confidence and enhancing the robustness of LiDAR SLAM. Consequently, degeneracy detection has garnered increasing attention in recent years. One popular approach is based on point-to-point matching, as demonstrated in [4] and [5]. This type relies on assessing the matching error of the point cloud between frames. However, considering point pairs exclusively leads to a significant impact of point noise on the detection results, causing the issue of "false detection" and inaccurate degeneracy judgment. In contrast, this paper proposes a point-to-distribution method that goes beyond considering point pairs alone.

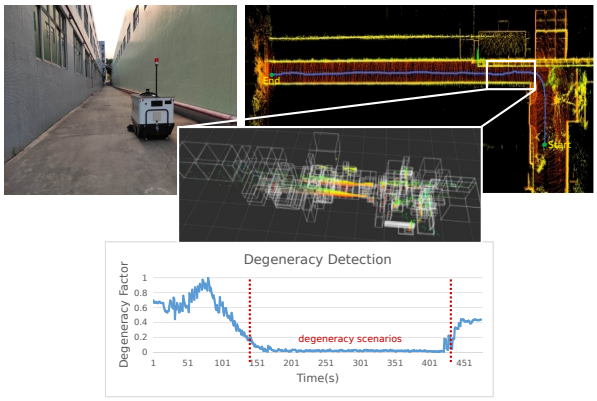

Fig. 1. The example of LiDAR SLAM in a degeneracy scene, with the zoomed image showcasing our adaptive voxel resizing. The lower image demonstrates the detection result obtained from our method, where the degeneracy factor decreases as the robot enters the channel.

To address the issue of "false detection" and enhance detection accuracy, this paper proposes a degeneracy detection method utilizing local geometric models base on point-to-distribution matching. Owing to the statistical nature of our proposed degeneracy detection factor, it effectively mitigates noise interference and mitigates the occurrence of the "false detection" problem. Moreover, to enhance the accuracy of the Probability Density Function (PDF) in describing the distribution, we introduce an adaptive voxel segmentation method that aligns with the Gaussian distribution assumption used in existing methods [3]. The main contributions of this work are the following points:

- We propose a robust degeneracy detection method using local geometric models based on point-to-distribution matching to overcome the "false detection" problem.
- We designing an adaptive voxel segmentation method for an accurate distribution description.

This work was supported in part by the Science and Technology Plan of Guangdong Province—Guangdong-Hong Kong-Macao Joint Innovation Project (Grant No. 2023A0505010016), the National Natural Science Foundation of China (62103179), the Foshan Science and Technology Innovation Team Project (Grant No. FS0AA-KJ919-4402-0098), and the Meituan Academy of Robotics ShenZhen.

*Corresponding author (chenwn@gdut.edu.cn).

[1]Biomimetic and Intelligent Robotics Lab (BIRL), School of Electromechanical engineer, Guangdong University of Technology, Guangzhou, Guangdong Province, China.

[2]Guangdong Key Laboratory of Modern Control Technology,Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, Guangzhou, Guangdong Province, China.

[3]JT-Innovation (Guangdong) Intelligent Technology Co., Ltd.

[4]College of Engineering, South China Agricultural University, China.

[5]Shenzhen Key Laboratory of Robotics and Computer Vision, Southern University of Science and Technology, China.

II. RELATED WORK

A. Degeneracy detection

Presently, three primary types of degeneracy detection methods exist. The first type is based on supplementary information. In [6] and [7], degeneracy detection is achieved by position information from UWB to validate the localization obtained from LiDAR SLAM. While this approach can determine the degeneracy state, it is susceptible to site constraints due to the necessity of UWB equipment. Likewise, [8] employs visual information from a depth camera for degeneracy detection. Although visual sensor is tasked with compensating for the absence of point cloud information in degeneracy scenarios, it also making it susceptible to changes in light intensity. [9] and [10] propose a known map-based method for localization confidence detection by solely utilizing LiDAR. Similarly, [11] utilizes raster maps for offline detection. However these methods are only effective in environments with priori information about the map.

The second type involves degeneracy detection methods based on deep learning. [12] and [13] propose predicting the uncertainty of the robot's pose by training on preprocessed sensor data. Additionally, [14] suggests training a degeneracy analysis network using raw LiDAR data to detect degeneracy in single-frame point clouds. However, deep learning-based methods suffer from limited generality and cannot be readily applied to diverse environments.

The third type involves eigenvalue-based methods. Masaki et al. propose the method assessing the degree of degeneracy using the Fisher information matrix [15]. In the presence of degeneracy in LiDAR SLAM, the maximum eigenvalue of the Fisher information matrix increases significantly. Similarly, [10] combines the local geometric model with the Fisher information matrix to assess localization confidence. Additionally, [16] introduces a PCA-based vector estimation method for detecting degeneracy in indoor scenes. However, these methods provide only a basic assessment of degeneracy status and do not account for the uncertainty in position estimation. To capture the uncertainty in position estimation, Ji Zhang et al. [4] propose utilizing the eigenvalues of the approximated Hessian matrix as a degeneracy factor to quantify the degree of degeneracy. Despite being validated in [17], this method remains vulnerable to noise interference due to its reliance on point-to-point matching. In [5] and [18], an enhanced point-to-point matching algorithm is proposed to improve the accuracy of degeneracy detection. However, despite the potential improvement in matching accuracy with the method proposed in [19] and [20], it remains susceptible to noise points and can lead to incorrect degeneracy detection results. The issue of "false detection" still remains unresolved.

To address the problem above, we propose a degeneracy factor utilizes local geometric models base on point-to-distribution matching, which provides additional reference information for degeneracy detection. We validate the effectiveness of our method in diverse environments, demonstrating its superiority over alternative approaches.

B. Voxel Segmentation

In point-to-distribution matching methods, voxel segmentation plays a crucial role in accurately describing the point cloud distributions. A common approach, as proposed in [3], is to divide the point clouds into voxel cells of fixed sizes. While this method is efficient, it does not take the distribution into account, often resulting in multi-peak point cloud distributions within the voxel. So violate the assumption of Gaussian distribution and lead to inaccurate local geometric models. Although an adaptive method [21] to accommodate different environmental dimensions, it is worth noting that when the point cloud distribution is scattered, relying solely on the eigenvalues of the covariance matrix can lead to inadequate representation of the voxel's geometric characteristics. In addition, the method [22] adapts voxel sizing for subsequent point cloud frames by computing the ratio between the number of scanned point clouds and the desired count. However, the size of the voxel still remains constant within a single frame. Consequently, it is fails to address the challenge by multi-peak point cloud distributions.

To address this limitation, researchers have explored alternative techniques. For instance, [23] proposes the k-means clustering method to enhance the accuracy of [3]. However, this method requires manual selection of the number of initial clusters, which can be subjective and time-consuming. Similarly, [24] employs a split greedy clustering method to improve 3D spatial matching accuracy. Nevertheless, the time consumption may not meet real-time requirements. Another approach [10] proposes a multi-constraint real-time point cloud clustering algorithm based on depth maps. However, this method necessitates adjusting the segmentation scale when deploying it on different LiDAR systems. Alternatively, deep learning-based methods [25], [26] have shown better to solve clustering problems. However, their real-time performance and generalizability remain significant limitations.

Given the discussion above, we design an adaptive segmentation method to adjust the voxel size by analyzing the distribution density of point clouds. The method divides the local geometric model into three kinds of geometric models to satisfy the Gaussian distribution assumption. Also, we adopt different segmentation methods for variant geometric models to build an accurate distribution description.

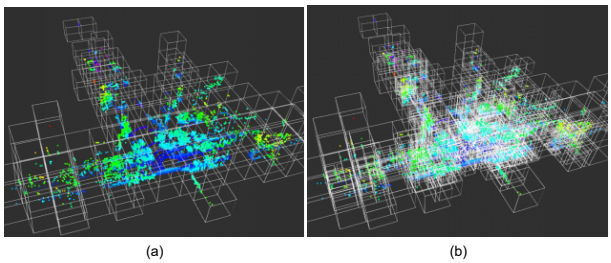

Fig. 2. Illustration of voxel segmentation. (a) is the result of using fixed size voxel. (b) is ours with adaptive voxel size for an accurate distribution description.

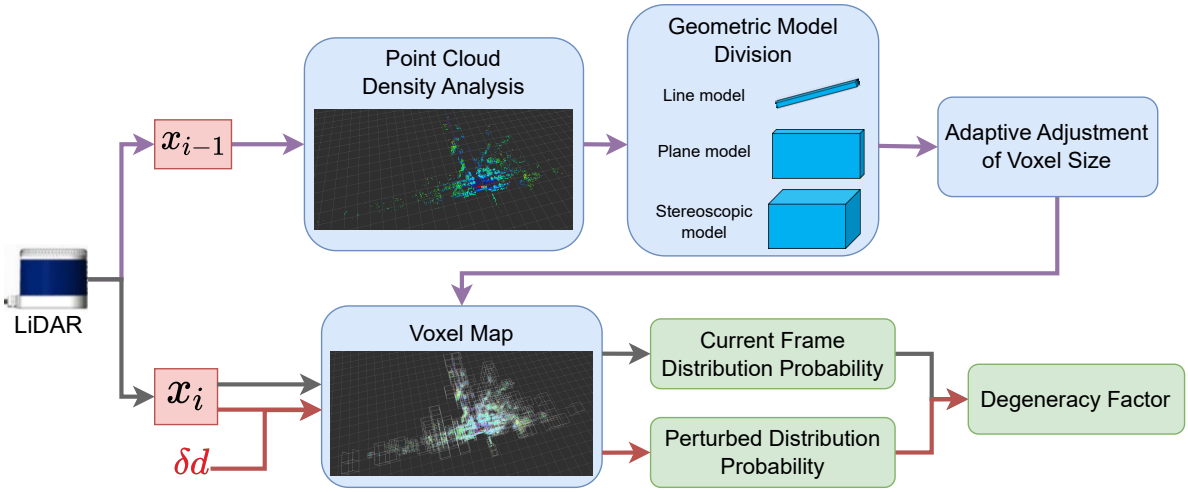

Fig. 3. System architecture of the method in this paper. The system receives information from the LiDAR of two consecutive frames, where the last frame $\boldsymbol{x_{i-1}}$ is segmented into geometric model according to the density analysis, which splits the model into voxels with adaptive sizes. The distribution probability of the current frame point cloud $\boldsymbol{x_i}$ in $\boldsymbol{x_{i-1}}$ is then calculated. The degeneracy state is judged by comparing the distribution probability change before and after adding the perturbation $\boldsymbol{\delta d}$.

III. Methodology

Fig. 3 illustrates the framework of our proposed method, which comprises two main modules. The first module focuses on adaptive tuning of voxel segmentation size. By analyzing the distribution density of the local point cloud, we classify voxels into distinct models. Subsequently, we employ different segmentation strategies to dynamically resize voxels based on the specific model, thereby enhancing the accuracy of local geometric model descriptions. The second module involves degeneracy detection using local geometric models base on point-to-distribution matching. The degeneracy state of the robot is determined by evaluating the change in distribution probability of the point cloud within the corresponding voxel between the inter-frame.

A. Adaptive Adjustment of Voxel Size

The comparison with existing methods [3] is visually illustrated in Fig. 2. The conventional method Fig. 2 (a) employs a fixed voxel size throughout the representation process, whereas our method Fig. 2 (b) adaptively adjusts the voxel size based on the density distribution of the point cloud. We perform voxel segmentation by identifying regions of the point cloud that exhibit sparsity or density. This segmentation approach not only enhances the accuracy of the local geometric model representation but also ensures the reasonableness of the Gaussian distribution assumption.

First, we partition the point clouds into line, surface, and stereoscopic models following the approach as described in [27]. Subsequently, specific strategies are applied to each model category to address their distinct characteristics.

In the case of stereoscopic models, the voxels are directly divided into eight equal parts due to the complex spatial distribution of the point cloud.

For the line model, the point cloud distribution is primarily concentrated in a single direction, making it necessary to focus only on the distribution along that specific direction.

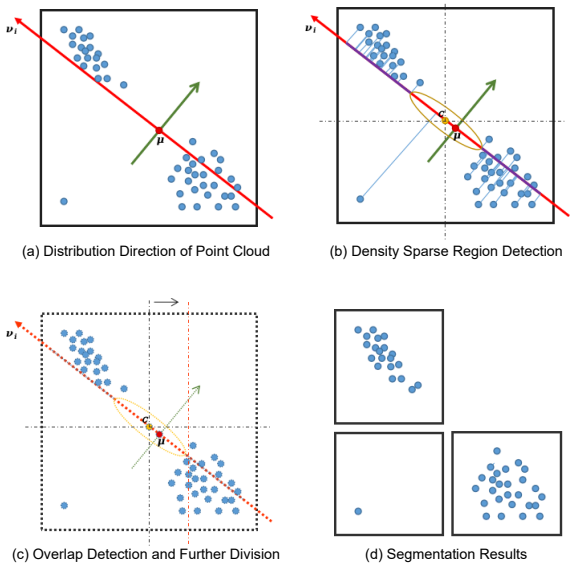

Fig. 4. An example of the adaptive voxel segmentation method for the line model is provided. Here, $\boldsymbol{\mu}$ denotes the mean value of the point cloud within the voxel, while $\boldsymbol{\nu_i}$ represents the dominant direction of the point cloud distribution. The ellipse delineates the range representing the sparsely distributed region of the point cloud, and $\boldsymbol{c}$ represents the segmentation point determined through density analysis.

As shown in Fig. 4(a), where $\boldsymbol{\mu}$ is the mean value within the voxel, $\boldsymbol{\nu_i}$ is the concentrated direction in which the point cloud distributes. First, all the point clouds are projected onto the vector $\boldsymbol{\nu_i}$, and the distribution density is determined based on the distance between the projected points. As show in 4 (b), the elliptical region indicates the sparse region, while the section of the purple line segment represents the dense region. Once the sparse region is identified, the center point $\boldsymbol{c}$ becomes the segmentation point, and the segmentation is performed along the direction of this point, as depicted by the black dashed line in the figure.

For the plane model, the point cloud distribution is primarily concentrated on a plane formed by two vectors. Therefore, our attention is focused on the density of the point cloud within that plane. The segmentation method is similar to that of line models, with the main difference being that we project the point clouds based on two vectors instead of just one.

As show in Fig. 4(c), if the vertical segmentation line overlaps with a significant number of point clouds, segmentation in that direction is omitted. Finally, the voxel is segmented into multiple smaller voxels as shown in Fig. 4(d), whose point cloud distribution is single-peak. In addition, a voxel with a small number of point clouds is defined as a sparse voxel. We fuse the sparse voxel with another nearby sparse voxel to form a new voxel. If a suitable voxel cannot be found, the point clouds within the sparse voxel are discarded as noise.

After the above voxel segmentation, the PDF is able to accurately describe the local geometric model. The proposed method satisfies the Gaussian distribution assumption and builds a voxel map for degeneracy detection.

B. Degeneracy Detection Factor

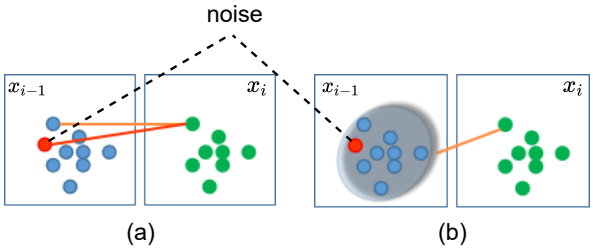

Fig. 5. The point cloud of the previous frame is represented as $\boldsymbol{x_{i-1}}$, while the current frame is denoted as $\boldsymbol{x_i}$. In Figure (a), the point-to-point method is used to establish correspondences between point pairs, where the orange line represents a correct match. However, as shown by the red line, errors occur when noisy red points interfere with the matching process. In (b), the point clouds are clustered for point-to-distribution matching, which effectively mitigates noise interference.

In this section, we introduce our degeneracy detection factor utilizing the previously constructed voxel map. Fig. 5(a) demonstrates that using a noisy point for inter-frame matching leads to errors and "false detectio" due to point-to-point matching [4]. In Fig. 5(b), by clustering the point clouds from the previous frame $\boldsymbol{x_{i-1}}$ for point-to-distribution matching, the interference of noise is mitigated. These clustered points are referred to as the local point clouds in our method. By considering the local point clouds as a unified entity, we can derive the local geometric information of the point set through its mean $\boldsymbol{\mu_{i-1}}$ and covariance matrix $\boldsymbol{\Sigma_{i-1}}$,

$$\mu_{i-1} = \frac{1}{n}\sum_{i=1}^{n} x_{i-1} \tag{1}$$

$$\Sigma_{i-1} = \frac{1}{n-1}\sum_{i=1}^{n}\left(x_{i-1}-\mu_{i-1}\right)\left(x_{i-1}-\mu_{i-1}\right)^T \tag{2}$$

where $\boldsymbol{n}$ denotes the number of point clouds, $\boldsymbol{x_{i-1}}\,\boldsymbol{(x,y,z)}$ is the point cloud position of the last frame. As a distribution description, we analyze the covariance matrix. As shown in Equation (3), each element of the covariance matrix $\boldsymbol{cov(x_i, x_j)}$ represents the degree of deviation between a random variable and its mean. Thus the mean and covariance matrices provide the feature of a local geometric model.

$$\Sigma = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix} \tag{3}$$

As a result of the adaptive voxel segmentation, the local point clouds are assumed to obey a Gaussian distribution. Consequently, the distribution probability $\boldsymbol{S_i(p)}$ of the current frame point cloud $\boldsymbol{x_i}$ within the matched area of $\boldsymbol{x_{i-1}}$ is represented by a probability density function (PDF):

$$S_i(p) = exp(-\frac{(x_i-\mu_{i-1})^T\Sigma_{i-1}^{-1}(x_i-\mu_{i-1})}{2}) \tag{4}$$

where $\boldsymbol{p}$ is the transformation of the position between the current point cloud and the matched local point cloud of $\boldsymbol{x_{i-1}}$. Taking logarithms on both sides of Equation (4), we can obtain

$$-2lnS_i(p) = (x_i-\mu_{i-1})^T\Sigma_{i-1}^{-1}(x_i-\mu_{i-1}) \tag{5}$$

let

$$e_i(p) = x_i - \mu_{i-1} \tag{6}$$

and substituting into Equation (5), there is

$$F_i(p) = -2lnS_i(p) = e_i^T(p)\Sigma_{i-1}^{-1}e_i(p) \tag{7}$$

where $F_i(p)$ is the distribution probability of x_i. Ideally, $\boldsymbol{p} = 0$ when there are sufficient geometric constraints, which means that the distribution probability of x_i in the voxel map is $F_i(0) = 1$. Adding a perturbation $\boldsymbol{\delta d}$ to Equation (7), the distribution probability of the perturbed point in the corresponding local geometric model can be denoted as:

$$\begin{aligned} F_i(p+\delta d) &= -2lnS_i(p+\delta d) \\ &= e_i^T(p+\delta d)\Sigma_{i-1}^{-1}e_i(p+\delta d) \end{aligned} \tag{8}$$

Based on the distribution probability changes of $\boldsymbol{x_i}$ before and after adding $\boldsymbol{\delta d}$, a degeneracy factor $\boldsymbol{D}$ is defined as follows

$$D = \Delta F_i = F_i(p+\delta d) - F_i(p) \tag{9}$$

In non-degenerate scenes exhibiting a diverse range of local geometric models, point-to-distribution based detection methods yield a multitude of distinct local geometric model descriptions. However, in the presence of a degenerate scene, as depicted in Fig. 1, where a single variation in local geometric models occurs, the impact of the matching error on our detection results becomes more pronounced. By utilising the differential responses observed in non-degraded and degraded scenes, our proposed degradation detection factor enables accurate determination of the degradation state.

To prove our proposed degeneracy detection factor, we further derive the degeneracy factor $\boldsymbol{D}$. Perform a Taylor series of $\boldsymbol{e_i(p+\delta d)}$ in Equation (8)

$$\begin{aligned} e_i(p+\delta d) &\approx e_i(p) + \frac{de_i}{dp}\delta d \\ &= e_i(p) + J_i\delta d \end{aligned} \tag{10}$$

Substituting Equation (10) into Equation (8):

$$F_i(p+\delta d) \approx (e_i + J_i\delta d)^T \Sigma_{i-1}^{-1}(e_i + J_i\delta d) \\ = e_i^T\Sigma_{i-1}^{-1}e_i + 2e_i^T\Sigma_{i-1}^{-1}J_i\delta d + \delta d^T J_i^T\Sigma_{i-1}^{-1}J_i\delta d \tag{11}$$

where $\boldsymbol{J_i}$ is a Jacobian matrix. In order to simplify Equation (11), let

$$b_i^T = e_i^T\Sigma_{i-1}^{-1}J_i \\ H_i = J_i^T\Sigma_{i-1}^{-1}J_i \tag{12}$$

According to Equation (7) and Equation (12), Equation (11) can be rewritten as

$$F_i(p+\delta d) = F_i(p) + 2b_i^T\delta d + \delta d^T H_i\delta d \tag{13}$$

therefore, $\boldsymbol{D}$ can be expressed by the following equation

$$D = F_i(p+\delta d) - F_i(p) = 2b_i^T\delta d + \delta d^T H_i\delta d \tag{14}$$

Substituting Equation (7) and Equation (8) into Equation (14):

$$D = [-2lnS_i(p+\delta d)] - [-2lnS_i(p)] \\ = 2ln\frac{S_i(p)}{S_i(p+\delta d)} \\ = 2b_i^T\delta d + \delta d^T H_i\delta d \tag{15}$$

Under the influence of $\boldsymbol{\delta d}$, the distribution probability of the point cloud located at the matched area of $\boldsymbol{x_{i-1}}$ decreases, so $\boldsymbol{S_i(p+\delta d)}$ also decreases. Thus $\boldsymbol{D}$ becomes larger when $\boldsymbol{\delta d}$ increases. Therefore the degeneracy factor is equivalent to calculating the following maximum value

$$D = arg\max(2b_i^T\delta d + \delta d^T H_i\delta d) \tag{16}$$

In Equation (16), $\boldsymbol{2b_i^T\delta d}$ is a 1×6 vector. Therefore, we only need to find the maximum value of $\boldsymbol{\delta d^T H_i\delta d}$:

$$D^* = arg\max \delta d^T H_i\delta d \\ = arg\max \frac{\delta d^T H_i\delta d}{\delta d^T\delta d} \times \delta d^T\delta d \tag{17}$$

The term to be maximized in $\frac{\boldsymbol{\delta d^T H_i\delta d}}{\boldsymbol{\delta d^T\delta d}}$ is a Rayleigh quotient [28]. Since $\boldsymbol{H_i}$ is a symmetric matrix, the maximum of the quotient is equal to the maximum eigenvalue of $\boldsymbol{H_i}$, namely $\boldsymbol{\lambda_{max}}$. Therefore, there are

$$D^* = \lambda_{max}\delta d^T\delta d \tag{18}$$

and

$$D = 2b_i^T\delta d + \lambda_{max}\delta d^T\delta d \tag{19}$$

Since $\boldsymbol{b_i^T}$ and $\boldsymbol{\delta d^T\delta d}$ are constant terms, they can be denoted as $\boldsymbol{a}$ and $\boldsymbol{b}$, respectively. Therefore, there is

$$D = a\delta d + b\lambda_{max} \tag{20}$$

According to the derivation, the degeneracy factor $\boldsymbol{D}$ is a $\mathbf{1\times 6}$ vector, which represents the degree of degeneracy in the direction where the perturbation occurs. The larger value of $\boldsymbol{D}$ indicates the larger degree of degeneracy and the eigenvector $\boldsymbol{\nu_i}$ corresponding to $\boldsymbol{\lambda_i}$ represents the direction of the degeneracy.

IV. EXPERIMENT

In this section, we performed a comprehensive evaluation of our method through a series of experiments. Firstly, we tested the influence of the environment on the degeneracy detection factor using a simulation system. Secondly, we conducted a comparative analysis of our method with existing approaches using the publicly datasets M2DGR [29] and KITTI [30], along with self-build robots experiments, to assess its robustness and accuracy.

For convenient comparison with other degeneracy detection factors, we perform min-max normalization in terms of the degeneracy detection factor. As show in Fig. 6, the degeneracy factor is below 0.2 when the robot enters the degeneracy scene. We eliminating any potential biases for the original data by normalizing the degeneracy factor values.

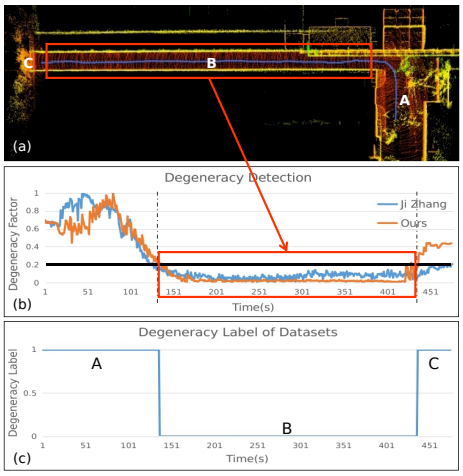

Fig. 6. (a) is the illustration of real-world experiments, where the white circle is the start position of the robot. (b) is the degeneracy factor after normalization in environment (a). As shown by the factor value, we set 0.2 as the threshold. (c) is the example of dataset labeling. a value of 1 is assigned to represent non-degeneracy, as observed in regions $\boldsymbol{A}$ and $\boldsymbol{C}$. On the other hand, a value of 0 is used to indicate degeneracy, as exemplified by region $\boldsymbol{B}$.

A. Feasibility Experiment

In the evaluation, we employed a simulation system to construct diverse environments, aiming to demonstrate the performance of our approach in accurately reflecting the geometric features present in these environments. Specifically, the geometric properties of each environment were quantified using the mean curvature of the point cloud. The results of the tests are presented in Table I, by comparing the mean values of the degeneracy detection factor across

TABLE I
EVALUATION IN THE ENVIRONMENTS WITH VARIANT GEOMETRY FEATURE

	Curvature	D_{ji}	D_{ours}
Environment 1	0.09079	0.52903	0.73440
Environment 2	2.49651	0.50635	0.60572
Difference		0.02268	**0.12868**

TABLE II

ROBUSTNESS EVALUATION ON PUBLIC DATASETS AND REAL-WORLD DATASETS INDICATED BY STANDARD DEVIATION AND FALSE DETECTION (THE LOWER THE BETTER).

	M2DGR [29]						KITTI [30]			Self-collected	
	street 01	street 02	street 03	street 04	street 05	street 06	01	03	06	city	park
Ji Zhang [4]	0.17026	0.15212	0.16409	0.21166	0.21495	0.19694	0.20392	0.23884	0.21433	0.197637	0.21071
Ours	0.14620	0.14827	0.15620	0.16977	0.19237	0.19253	0.19745	0.18926	0.15362	0.186840	0.20172
improve	**14.14%**	**2.53%**	**4.81%**	**19.79%**	**10.50%**	**2.24%**	**3.17%**	**20.76%**	**28.32%**	**5.46%**	**4.26%**
Number of False Detections											
Ji Zhang [4]	125	210	36	168	0	22	1	34	9	6	0
Ours	**10**	**40**	[7]	**3**	0	**7**	0	**5**	**6**	**1**	0

different environments, we observed significant variations in our degeneracy detection factor($\boldsymbol{D_{ours}}$), indicating its ability to effectively capture and reflect the distinct geometric features inherent in each environment. In contrast, the degradation factors($\boldsymbol{D_{ji}}$) obtained using the baseline method [4] demonstrate smaller variability.

B. System Performance Experiment

To facilitate the comparison of detection results, we labeled the dataset with degeneracy status based on specific rules. These rules involve calculating the angular change of the point cloud normal vector and comparing the error between odometry and the localization ground truth. If the angle change is less than 5 degrees and the localization error is greater than 5 meters, the frame is labeled as degeneracy. An example of the labeled results can be seen in Fig. 6 (c).

This section evaluates the robustness and accuracy of the proposed method from two perspectives. First, we measure the algorithm's robustness by calculating the standard deviation of the degeneracy factors across the entire scene. It is important to note that all the recorded sequences contain real-world sensor noise. The test results, presented in Table II, exclude the labeled degeneracy scenes from datasets $street04$, $street05$ and 01 for fair comparison. The results indicate that our method exhibits a smaller standard deviation, demonstrating its stability in non-degeneracy scenes and its ability to better resist noise influence. Additionally, the reduced number of false detections demonstrates our ability to mitigate the "false detection" issue.

Secondly, the algorithm's accuracy is assessed through a precision-recall test. Non-degeneracy ($\boldsymbol{P}$) and degeneracy ($\boldsymbol{N}$) frames are manually labeled beforehand. Subsequently, we compare the degeneracy detection factors ($\boldsymbol{D_{ours}}$ and $\boldsymbol{D_{ji}}$) with $\boldsymbol{P}$ and $\boldsymbol{N}$. Precision and recall are defined as follows: $\boldsymbol{TP}$ (true positives) represents degeneracy detections correctly predicting non-degeneracy state, $\boldsymbol{FP}$ (false positives) represents incorrect predictions of non-degeneracy state, and $\boldsymbol{FN}$ (false negatives) represents predictions of non-degeneracy into degeneracy state.

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} \\ Recall &= \frac{TP}{TP+FN}. \end{aligned} \tag{21}$$

According to equation (21), the precision-recall results are shown in Table III. Among them, all datasets contain degeneracy scenes. The results of "Precision" indicate that our method exhibits higher accuracy in the majority of scenes. The results of "Recall" demonstrate that our method is more effective in mitigating noise interference, thereby reducing "false detections".

TABLE III

PRECISION-RECALL EXPERIMENT.

		Precision		Recall	
		Ji Zhang [4]	*Ours*	*Ji Zhang* [4]	*Ours*
M2DGR [29]	street 04	0.9053	**0.9318**	0.9957	**0.9989**
	street 05	0.5890	**0.8833**	0.9970	**0.9985**
KITTI [30]	01	0.5969	**0.7763**	0.950	**0.9833**
Self-collected	corridors	0.9070	0.8757	0.7178	**0.9568**
	factory	0.8753	**0.8779**	0.8317	**0.9484**

Regarding the time consumption, the results in Table IV demonstrate that our method is able to satisfy the real-time requirement for on-road driving.

TABLE IV

THE TIME CONSUMPTION OF THE VOXEL SEGMENTATION METHOD.

Num. of point clouds	28040	16688	6331	4775	2783
Time(ms)	47.56	25.27	12.54	5.59	3.39

V. CONCLUSIONS

In this paper, a degeneracy detection method based on point-to-distribution matching is proposed. To achieve an accurate distribution description, we propose an adaptive voxel segmentation method that constructs a precise PDF for describing local geometric models. Furthermore, a degeneracy detection factor is proposed, which using local geometric models based on point-to-distribution matching. In contrast to relying on existing point-to-point matching, we refer to the local geometric models to overcome issues of noise interference and the problem of "false detection". The evaluation on a public and self-collected dataset demonstrates the superior robustness and accuracy of our method in detecting degradation compared to existing methods.

REFERENCES

[1] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, pp. 239–256, 1992. [Online]. Available: https://api.semanticscholar.org/CorpusID:21874346

[2] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," in *Robotics: Science and Systems*, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:18612391

[3] M. Magnusson, "The three-dimensional normal-distributions transform : an efficient representation for registration, surface analysis, and loop detection," 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:14692452

[4] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 809–816, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:14084603

[5] A. Hinduja, B.-J. Ho, and M. Kaess, "Degeneracy-aware factors with applications to underwater slam," *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1293–1299, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:208878454

[6] W. Zhen and S. A. Scherer, "Estimating the localizability in tunnel-like environments using lidar and uwb," *2019 International Conference on Robotics and Automation (ICRA)*, pp. 4903–4908, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:198910004

[7] H. Zhou, Z. Yao, and M. Lu, "Uwb/lidar coordinate matching method with anti-degeneration capability," *IEEE Sensors Journal*, vol. 21, pp. 3344–3352, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:226495681

[8] H. Cho, S. Yeon, H. Choi, and N. L. Doh, "Detection and compensation of degeneracy cases for imu-kinect integrated continuous slam with plane features †," *Sensors (Basel, Switzerland)*, vol. 18, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:4001275

[9] W. Zhen, S. Zeng, and S. Soberer, "Robust localization and localizability estimation with a rotating laser scanner," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6240–6245, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:474373

[10] Y. Liu, J. Wang, and Y. Huang, "A localizability estimation method for mobile robots based on 3d point cloud feature," *2021 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, pp. 1035–1041, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:237375838

[11] M. Kondo, M. Hoshi, Y. Hara, and S. Nakamura, "Localizability estimation based on occupancy grid maps," *2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 368–373, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:251847480

[12] W. Vega-Brown, A. Bachrach, A. Bry, J. Kelly, and N. Roy, "Cello: A fast algorithm for covariance estimation," *2013 IEEE International Conference on Robotics and Automation*, pp. 3160–3167, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:17504411

[13] D. Landry, F. Pomerleau, and P. Giguère, "Cello-3d: Estimating the covariance of icp in the real world," *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8190–8196, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:52916396

[14] J. Nubert, E. Walther, S. Khattak, and M. Hutter, "Learning-based localizability estimation for robust lidar localization," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 17–24, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:247411171

[15] M. Koizumi, K. Nonaka, and K. Sekiguchi, "Avoidance of singular localization environment using model predictive control for mobile robots," *2017 11th Asian Control Conference (ASCC)*, pp. 2866–2871, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:3441891

[16] W. Shi, S. Li, C. Yao, Q. Yan, C. Liu, and Q. Chen, "Dense normal based degeneration-aware 2-d lidar odometry for correlative scan matching," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–16, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:255037231

[17] M. Tranzatto, F. Mascarich, L. Bernreiter, C. Godinho, M. Camurri, S. Khattak, T. Dang, V. Reijgwart, J. Loeje, D. Wisth, S. Zimmermann, H. Nguyen, M. Fehr, L. Solanka, R. Buchanan, M. Bjelonic, N. Khedekar, M. Valceschini, F. Jenelten, M. Dharmadhikari, T. Homberger, P. D. Petris, L. Wellhausen, M. Kulkarni, T. Miki, S. Hirsch, M. Montenegro, C. Papachristos, F. Tresoldi, J. Carius, G. Valsecchi, J. Lee, K. Meyer, X. Wu, J. I. Nieto, A. P. Smith, M. Hutter, R. Y. Siegwart, M. W. Mueller, M. F. Fallon, and K. Alexis, "Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge," *ArXiv*, vol. abs/2201.07067, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:238034403

[18] T. Tuna, J. Nubert, Y. Nava, S. Khattak, M. H. R. S. Lab, E. Zurich, and A. A.G., "X-icp: Localizability-aware lidar registration for robust localization in extreme environments," *ArXiv*, vol. abs/2211.16335, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:254069896

[19] A. V. Segal, D. Hähnel, and S. Thrun, "Generalized-icp," in *Robotics: Science and Systems*, 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:231748613

[20] K. Koide, M. Yokozuka, S. Oishi, and A. Banno, "Voxelized gicp for fast and accurate 3d point cloud registration," *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11 054–11 059, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:211220621

[21] Z. Liu and F. Zhang, "Balm: Bundle adjustment for lidar mapping," *IEEE Robotics and Automation Letters*, vol. 6, pp. 3184–3191, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:223957253

[22] A. Reinke, M. Palieri, B. Morrell, Y. Chang, K. Ebadi, L. Carlone, and A. akbar Agha-mohammadi, "Locus 2.0: Robust and computationally efficient lidar odometry for real-time underground 3d mapping," *ArXiv*, vol. abs/2205.11784, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:249017934

[23] A. Das and S. L. Waslander, "Scan registration with multi-scale k-means normal distributions transform," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2705–2710, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:1685277

[24] A. Das, J. Servos, and S. L. Waslander, "3d scan registration using the normal distributions transform with ground segmentation and point cloud clustering," *2013 IEEE International Conference on Robotics and Automation*, pp. 2207–2212, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:206850248

[25] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, A. Trigoni, and A. Markham, "Randla-net: Efficient semantic segmentation of large-scale point clouds," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11 105–11 114, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:208290898

[26] D. Nie, R. Lan, L. Wang, and X. Ren, "Pyramid architecture for multi-scale processing in point cloud segmentation," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17 263–17 273, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:250552049

[27] S. Chen, H. Ma, C. Jiang, B. Zhou, W. Xue, Z. Xiao, and Q. Li, "Ndt-loam: A real-time lidar odometry and mapping with weighted ndt and lfa," *IEEE Sensors Journal*, vol. 22, pp. 3660–3671, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:245546902

[28] R. A. Horn and C. R. Johnson, "Matrix analysis, 2nd ed," 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:198169348

[29] J. Yin, A. Li, T. Li, W. Yu, and D. Zou, "M2dgr: A multi-sensor and multi-scenario slam dataset for ground robots," *IEEE Robotics and Automation Letters*, vol. 7, pp. 2266–2273, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:245502543

[30] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, pp. 1231 – 1237, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:9455111