Academic Journal of Science and Technology
ISSN: 2771-3032 | Vol. 14, No. 3, 2025

Enhancing Inter-frame Registration Accuracy Based on Adaptive Grid Projection

Zhenhao Xing[1,*], Hongmei Qu[1], Xiayu Zhao[1]

[1] School of Surveying & Land Information Engineering, Henan Polytechnic University, Jiaozuo 454000, China

[*] Corresponding author: Zhenhao Xing (Email: xzh123@home.hpu.edu.cn)

Abstract: The high-precision positioning of multi-line LiDAR in urban environments relies on the accuracy of inter-frame point cloud registration. However, dynamic objects and changes in local geometric features in complex scenes can affect the stability of traditional registration methods. This paper proposes an inter-frame registration method based on adaptive grid projection to improve registration accuracy. The proposed method first preprocesses the point cloud by extracting feature points for registration and applying Euclidean clustering to generate point cloud clusters. Each corresponding feature cluster is labeled, and an adaptive grid projection based on the incident angle is employed to enhance pole-like features. Additionally, low-stability regions are removed to improve the reliability of matching points. Subsequently, inter-frame registration is performed using pole-like features to achieve high-precision pose estimation. Experimental results demonstrate that the proposed method effectively enhances registration accuracy and robustness in urban environments, providing reliable technical support for LiDAR-based autonomous localization in GNSS-denied scenarios.

Keywords: Adaptive Grid Projection, Autonomous Localization, Inter-frame Registration, Multi-line LiDAR.

1. Introduction

Multi-line LiDAR has been widely used in autonomous localization and navigation tasks in urban environments due to its high precision and strong resistance to environmental interference. Compared to traditional GNSS/INS integrated navigation[1], LiDAR-based positioning methods can still provide stable pose estimation even when GNSS signals are obstructed or unavailable. However, the accuracy of inter-frame LiDAR registration directly impacts localization reliability, making it a critical research challenge to enhance registration accuracy in complex urban environments.

Traditional inter-frame registration methods, such as point-to-point ICP (Iterative Closest Point) and point-to-plane ICP, rely on local geometric feature matching. However, urban environments contain numerous dynamic objects (e.g., pedestrians and vehicles) and structurally complex scenes (e.g., glass curtain walls of buildings and occlusions caused by vegetation), making traditional methods susceptible to local feature variations, thereby reducing registration accuracy. Furthermore, the high-dimensional nature of point cloud data and computational complexity limit real-time performance. Therefore, effectively utilizing stable environmental features and integrating efficient registration strategies to enhance inter-frame registration accuracy and robustness remains a key research focus.

To address these challenges, this paper proposes an inter-frame registration method based on adaptive grid projection[2,3] to improve registration accuracy. The proposed method preprocesses point clouds through adaptive grid projection, adjusting the projection strategy based on the incident angle to enhance the prominence of pole-like features, thereby improving the stability of point cloud matching. During registration, pole-like features are utilized for matching, and least-squares optimization is employed to estimate the pose transformation, further enhancing registration accuracy. The main contributions of this paper include:

1. An adaptive grid projection method is proposed, adjusting the projection scale based on the incident angle to effectively enhance the representation of pole-like features and improve the stability of point cloud matching.
2. An optimized pole-like feature matching strategy is introduced, eliminating low-stability regions to enhance the reliability of inter-frame registration point pairs and improve pose estimation accuracy.

Experimental results demonstrate that the proposed method significantly improves inter-frame registration accuracy in complex urban environments while maintaining strong robustness, providing reliable technical support for high-precision LiDAR-based autonomous localization.

2. Related Work

In recent years, multi-line LiDAR has been widely applied in high-precision positioning and navigation, with inter-frame registration being a core technology that has attracted continuous research attention. Traditional ICP and its variants are commonly used for point cloud registration but often fail in dynamic environments or under sparse point cloud conditions. To address challenges in urban environments, various studies have proposed feature-based methods, local descriptors, probabilistic models, and deep learning approaches to improve registration accuracy and robustness.

Point cloud registration is a key method for obtaining the pose transformation between adjacent frames by finding the correspondence between features in adjacent frames and calculating the accurate transformation matrix. Currently, the popular methods include ICP[4] (Iterative Closest Point) registration, NDT[5,6] (Normal Distribution Transform) registration, and various variants of ICP, such as point-to-line ICP[7,8], point-to-plane ICP[9], and Normal Iterative Closest Point (NICP)[10].

The ICP algorithm selects the nearest points from the source and target point clouds as correspondences and solves for the rotation and translation matrices. The algorithm

iterates until the registration error reaches a threshold or the maximum number of iterations is reached. It is highly sensitive to noise and outliers. While it converges well when a good initial guess is provided, it is prone to getting stuck in local minima. The Generalized Iterative Closest Point (GICP) algorithm[11,12] combines traditional ICP and point-to-plane methods within a probabilistic framework. It considers the local plane structures of the two scans, improving registration accuracy while maintaining the simplicity and speed of ICP. Koide et al.[13] proposed a voxelized GICP algorithm, which estimates the voxel distribution by aggregating the distribution of points within each voxel. This method maintains the accuracy of nearest neighbor search while avoiding the large resource consumption typically associated with it. The algorithm is as accurate as GICP and faster in terms of processing speed. Zhao et al.[14] proposed a point cloud edge extraction method based on Principal Component Analysis (PCA) to optimize the GICP algorithm. By calculating the point cloud normal vectors and their tangent planes, a local coordinate system is constructed. The angles between adjacent vectors and the local coordinate system are calculated to identify the feature points for the edge contour. The principal components are derived from the covariance matrix of the point set. The pose transformation parameters and rigid body transformation matrix are solved using the maximum likelihood estimation method for refined GICP registration, resulting in improved accuracy and convergence speed.

The NDT algorithm divides the point cloud to be registered into grids of specified size. It constructs a probability distribution function for each grid using a normal distribution and solves for the optimal transformation parameters that maximize the probability density distribution of the source point cloud. This algorithm is less dependent on the initial value and does not require feature matching, but it has high requirements for voxel grid size and offers slightly lower registration accuracy compared to ICP. It is not suitable for registering low-density point clouds. Zhou et al.[15] used the NDT algorithm to compress the original dense point cloud into a lightweight representation. They employed the NDT-Transformer network to learn global descriptors from the NDT cells and retrieve these descriptors for 3D point cloud localization. This method offers high computational efficiency and is suitable for large-scale point cloud-based localization.

The registration algorithm based on deep learning extracts the features of point clouds and registers them by training the neural network model. It has a good effect on large-scale registration tasks, and can adapt to different scenes by changing the training set, but it needs a lot of labeled data for training, and the training time is long. Dubé et al.[16] used machine learning methods for feature point recognition and matching based on 3D segmentation. Li et al.[17] proposed the DMLO deep learning matching framework, which extracts high-confidence matching pairs from LiDAR scans using deep learning methods and minimizes the distance between the matching pairs to solve the transformation matrix.

3. Materials and Methods

In this study, the rod-shaped feature point clouds are projected onto a grid map, and the overlap of feature points between two adjacent frames of point clouds is calculated. For the rod-shaped point clouds in adjacent frames that have been processed using least squares fitting, the KD-tree based Euclidean clustering algorithm is used to generate point cloud clusters. Corresponding rod-shaped point cloud clusters in adjacent frames are labeled, with points representing the same rod-shaped ground objects in both frames receiving the same label. A grid map of size 32×1000 is created, where 32 represents the number of laser beams and 1000 represents the number of point clouds obtained by one laser beam after completing a full rotation. The rod-shaped feature point clouds with the same label are projected onto the grid map based on their laser beam and incident angle. After the point clouds are projected onto the grid, their grid values are updated, and their 3D coordinates are saved.

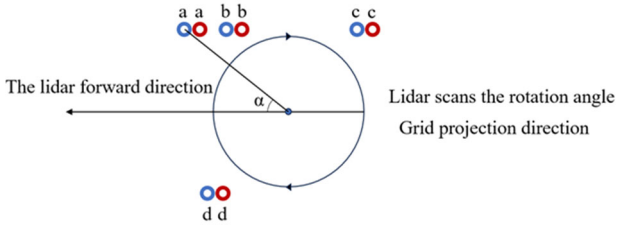

Figure 1. Schematic of the adaptive transformation based on the average incident angle

As shown in Figure 1, the red circles represent the current frame's marked rod-shaped point clouds, while the blue circles represent the marked rod-shaped point clouds from the previous frame. For each pair of marked point clouds, the average incident angle α is calculated during projection. From the Figure 1, it can be seen that, in the forward direction of the LiDAR, the scanning rotation angle ranges from 0° to 360°, and the grid projection starts at 0° and proceeds along the rotation angle. When the incident angle α is between 0° and 180°, the blue circle appears in front of the red circle on the grid image. When α is between 180° and 360°, the red circle appears in front of the blue circle. By determining the average incident angle α, the point cloud clusters in the same marked adjacent frames are projected onto the grid. One grid image is fixed, and the other is column-shifted by one grid unit at a time, calculating the number of overlapping points. When the average incident angle α is between 0° and 180°, the shift direction is set to leftward, and the grid representing the red circle's point cloud cluster projection is shifted. When the average incident angle α is between 180° and 360°, the shift direction remains unchanged, and the grid image of the blue circle's point cloud cluster projection is shifted. By determining the number of overlapping grids, the point cloud corresponding to the moment with the maximum overlap is saved. Figure 2 shows the schematic diagram of the grid overlap calculation.

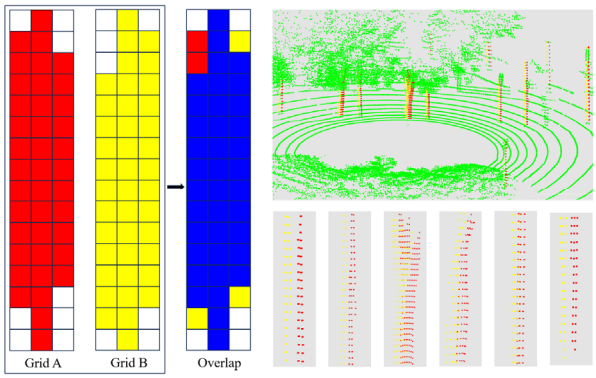

Figure 2. Schematic diagram of grid overlap calculation

4. Experimental Results

The experimental system is based on Windows 11 with Visual Studio 2017, utilizing the Point Cloud Library (PCL) 1.13.1 and OpenCV 4.9.0. The experimental data was collected using a Hesai 32-line multi-line LiDAR (XT32M2X) configured in dual-echo mode. The point cloud acquisition rate is 1,280,000 points per second, with a scanning frame rate of 20 Hz, a horizontal angular resolution of 0.36°, and a vertical angular resolution of 1.3°. The point cloud data is shown in Figure 3.

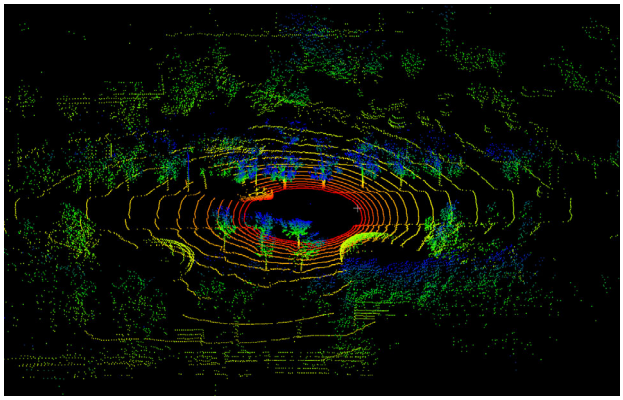

Figure 3. Point cloud data

According to the empirical information of the LiDAR, this study focuses on the ground points extracted from the 24th to the 32nd laser beams of the 32-line lidar, with the ground point cloud range defined by an angle of 3° to the left and right of the vehicle's forward and rearward centerline.

In Figure 4, the original point cloud is shown in (a). In (b), the light blue points represent the extracted rod-shaped features, and the white points represent the extracted ground points within the front and rear range of the vehicle. In (c), the light blue points represent the rod-shaped feature points, and the white points represent the extracted ground points. In (d), the red points represent the rod-shaped feature points and ground points from the previous frame of the adjacent frames, and the yellow points represent the rod-shaped feature points and ground points from the subsequent frame of the adjacent frames.

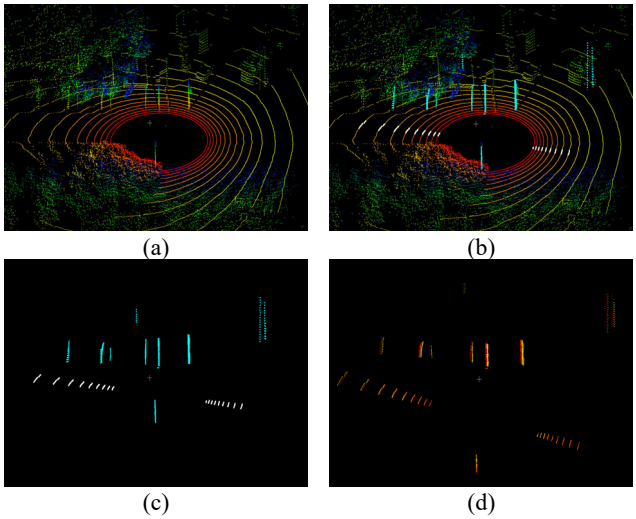

Figure 4. Feature extraction process in a noisy environment. (a) Original point cloud; (b) Features in the point cloud environment; (c) Point cloud features; (d) Corresponding features in adjacent frames

Figure 5 shows a comparison of the results before and after grid projection processing. The top row displays the rod-shaped point cloud clusters before grid projection, where many mismatched points can be observed. This mismatch is due to changes in the surrounding environment along the forward direction, causing occlusions and other issues that affect subsequent registration and computational accuracy. The bottom row shows the point cloud clusters after grid projection processing, with a one-to-one correspondence to the clusters in the top row. Visually, the correspondence between the processed blue and green point clouds is clearly superior to that between the red and yellow point clouds. From the data perspective, after grid projection processing, the number of rod-shaped point clouds in adjacent frames is equal, and the correspondence is significantly improved.

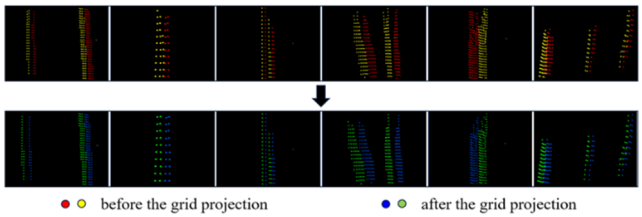

Figure 5. Comparison of grid projection processing before and after

Table 1 presents a comparison of the feature registration accuracy before and after grid projection processing. Before grid projection, the average RMSE of feature registration was 0.082m, and the average overlap rate of feature point clouds was 93.23%. After grid projection processing, the average RMSE of feature registration decreased to 0.046m, and the average overlap rate of feature point clouds increased to 96.58%.

Table 1. Comparison of feature registration accuracy before and after grid projection processing

Area	RMSE/m		Overlap	
	Before	After	Before	After
Road 1	0.056	0.037	96.46%	98.46%
Road 2	0.103	0.044	94.60%	97.09%
Road 3	0.085	0.050	90.21%	96.33%
Road 4	0.085	0.051	91.49%	94.77%
Road 5	0.081	0.048	92.20%	95.55%
Road 6	0.079	0.045	94.40%	97.29%
Average	0.082	0.046	93.23%	96.58%

5. Conclusion

This study proposes an adaptive grid projection processing method to refine the extracted feature point clouds from adjacent frames, ensuring one-to-one correspondence between feature points in consecutive frames and enhancing inter-frame registration accuracy. The average registration error after grid projection processing of the extracted rod-shaped feature point clouds is very small (average RMSE is approximately 0.046m). Experimental results show that after grid projection processing, the average RMSE error of point cloud feature registration is reduced by approximately 43.90%, and the average point cloud overlap rate is increased to 96.58%, compared to the unprocessed grid projection case. This indicates that the rod-shaped point clouds extracted by this method can be effectively registered between frames.

References

[1] R. B. Langley, P. J. G. Teunissen, and O. Montenbruck, “Introduction to GNSS,” in *Springer Handbook of Global Navigation Satellite Systems*, P. J. G. Teunissen and O. Montenbruck, Eds. Cham: Springer International Publishing, 2017, pp. 3–23.

[2] P. K. Agarwal, L. Arge, and A. Danner, “From point cloud to grid DEM: A scalable approach,” in *Progress in Spatial Data Handling: 12th International Symposium on Spatial Data Handling*, A. Riedl, W. Kainz, and G. A. Elmes, Eds. Berlin, Heidelberg: Springer, 2006, pp. 771–788.

[3] L. Guo, M. Yang, B. Wang, and C. Wang, “Occupancy grid based urban localization using weighted point cloud,” in *Proc. 2016 IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Rio de Janeiro, Brazil, Nov. 2016, pp. 60–65.

[4] 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.

[5] P. Biber and W. Strasser, “The normal distributions transform: A new approach to laser scan matching,” in *Proc. 2003 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Las Vegas, NV, USA, 2003, vol. 3, pp. 2743–2748.

[6] M. Magnusson, *The Three-Dimensional Normal-Distributions Transform: An Efficient Representation for Registration, Surface Analysis, and Loop Detection*. Örebro, Sweden: Örebro University, 2009.

[7] A. Censi, “An ICP variant using a point-to-line metric,” in *Proc. 2008 IEEE Int. Conf. Robot. Autom. (ICRA)*, 2008, pp. 19–25.

[8] Q. Wang and J. Zhang, “Point cloud registration algorithm based on combination of NDT and PLICP,” in *Proc. 2019 15th Int. Conf. Comput. Intell. Security (CIS)*, Macao, China, Dec. 2019, pp. 132–136.

[9] K.-L. Low, “Linear least-squares optimization for point-to-plane ICP surface registration,” 2004.

[10] J. Serafin and G. Grisetti, “NICP: Dense normal based point cloud registration,” in *Proc. 2015 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Hamburg, Germany, Sep. 2015, pp. 742–749.

[11] A. Segal, D. Hähnel, and S. Thrun, “Generalized-ICP,” in *Proc. Robotics: Sci. Syst.*, Jun. 2009.

[12] J. Servos and S. L. Waslander, “Multi-channel generalized-ICP: A robust framework for multi-channel scan registration,” *Robot. Auton. Syst.*, vol. 87, pp. 247–257, 2017.

[13] K. Koide, M. Yokozuka, S. Oishi, and A. Banno, “Voxelized GICP for fast and accurate 3D point cloud registration,” in *Proc. 2021 IEEE Int. Conf. Robot. Autom. (ICRA)*, Xi’an, China, May 2021, pp. 11054–11059.

[14] W. Zhao, D. Zhang, D. Li, Y. Zhang, and Q. Ling, “Optimized GICP registration algorithm based on principal component analysis for point cloud edge extraction,” *Meas. Control*, vol. 57, pp. 77–89, 2024.

[15] Z. C. Zhou, C. Zhao, D. Adolfsson, S. Z. Su, Y. Gao, T. Duckett, and L. Sun, “NDT-transformer: Large-scale 3D point cloud localisation using the normal distribution transform representation,” in *Proc. 2021 IEEE Int. Conf. Robot. Autom. (ICRA)*, Xi'an, China, 2021, pp. 5654–5660.

[16] R. Dubé, D. Dugas, E. Stumm, J. Nieto, R. Siegwart, and C. Cadena, “SegMatch: Segment based loop-closure for 3D point clouds,” in *Proc. 2017 IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 5266–5272.

[17] Z. Li and N. Wang, “DMLO: Deep matching LiDAR odometry,” in *Proc. 2020 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Las Vegas, NV, USA, Oct. 2020, pp. 6010–6017.