

Multi-Constellation-Inspired Single-Shot Global LiDAR Localization

Tongzhou Zhang¹, Gang Wang^{1,2,3,4*}, Yu Chen², Hai Zhang⁵, Jue Hu⁵

¹College of Computer Science and Technology, Jilin University

²College of Software, Jilin University

³State Key Laboratory of Automotive Simulation and Control, Jilin University

⁴Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University

⁵National Key Laboratory of Science and Technology on Advanced Composites in Special Environments,
Harbin Institute of Technology

tzzhang20@mails.jlu.edu.cn, gangwang@jlu.edu.cn, chen_yu21@mails.jlu.edu.cn, {hai.zhang, juehundt}@hit.edu.cn

Abstract

Global localization is a challenging task for intelligent robots, as its accuracy directly contributes to the performance of downstream navigation and planning tasks. However, existing literature focus more on the place retrieval and the success rate of localization, with limited attention given to the metrics of position estimation. In this paper, a single-shot global LiDAR localization method is proposed with the ultimate goal of achieving high position accuracy, inspired by the positioning approach of multi-constellation localization systems. Initially, we perform coarse localization using global descriptors and select observation points along with their corresponding coordinates based on the obtained coarse localization results. Coordinates can be acquired from a pre-built map, GNSS, or other devices. Then, a lightweight LiDAR odometry method is designed to estimate the distance between the retrieved data and the observation points. Ultimately, the localization problem is transformed into an optimization problem of solving a system of multiple sphere equations. The experimental results on the KITTI dataset and the self-collected dataset demonstrate that our method achieves an average localization error (including errors in the z -axis) of 0.89 meters. In addition, it achieves retrieval efficiency of 0.357 s per frame on the former dataset and 0.214 s per frame on the latter one. Code and data are available at <https://github.com/jlurobot/multi-constellation-localization>.

Introduction

Global localization has a wide range of applications in autonomous navigation for robots, e.g., initial localization in autonomous driving (Li and Li 2021) and relocalization in the kidnapped robot problem (Yu et al. 2020). It can be regarded as a problem of determining a robot or vehicle’s initial pose from sensor data without a prior pose. Single-shot global localization (Ratz et al. 2020) is an efficient and effective method for initializing the pose, which utilizes a place recognition and pose estimation to establish the relationship between a single local lighting detection and ranging (LiDAR) frame and a pre-built map.

Single-shot global localization achieved solely through place recognition (Cop, Borges, and Dubé 2018; Kim and Kim 2018; Liu et al. 2019; Ma et al. 2022) is to retrieve

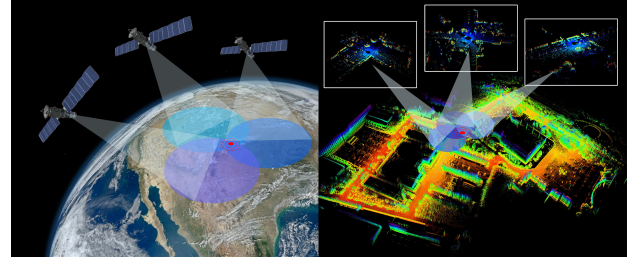


Figure 1: An illustration of the mechanism of multi-constellation localization and the proposed method.

the highest-probability keyframe (element of the pre-built map) using global descriptors, and assessing the similarity between the current frame and the keyframe. Typical methods include handcrafted feature-based methods such as Delight (2018) and Scan Context (2018), as well as deep learning based methods like LPD-Net (2019) and OverlapTransformer (2022). However, these methods are constrained by the aforementioned mechanism, as they treat the localization problem as a database retrieval task only, resulting in coarse localization results by providing the closest matching frame pose from the database.

If high-precision results are required, feature-based registration or similar methods is still necessary for matching current frame to retrieved keyframes. This category of method is known as “place recognition followed by local pose estimation,” employing a separate coarse-to-fine manner. Ratz et al. (2020) enhanced SegMap (Dubé et al. 2020) by training neural network and evaluated the localization accuracy using Iterative Closest Point (ICP) (Besl and McKay 1992). Shi et al. (Shi et al. 2021) proposed a variant of Scan Context for keyframe retrieval, then applied Normal Distributions Transform (NDT) (Biber and Straßer 2003) to get a precise initial pose. Luo et al. (Luo et al. 2022) proposed a descriptor called HOPN and refined the coarse pose using ICP. While these methods involve a refinement process, their primary focus still lies in place recognition. The registration methods they utilize, such as ICP and NDT, can be trapped in local optima due to unknown initial correspondences between the source and target frames, or fail to converge because of large translation between them. The cumu-

*Corresponding author

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

lative errors and the sparsity of the map further complicate the task of estimation, particularly in terms of estimating the z -axis. Consequently, it leads to considerable errors in localization. In addition, Yin et al.'s survey (Yin et al. 2023) only describes registration methods when discussing these two-stage methods, indicating limited innovation in this category of approaches. Notably, they argue that in classical scheme for autonomous mobile robots, a precise pose state is required for downstream planning and control module. Thus, place retrieval is not the ultimate of global localization. Instead, pose estimation metrics holds greater significance.

In this paper, inspired by multi-constellation localization systems, as shown in Fig. 1, a novel single-shot coarse-to-fine global localization method is proposed. Firstly, a global descriptor generated from input LiDAR data is used for coarse place retrieval, followed by selecting multiple adjacent point clouds as observation points. Subsequently, a lightweight LiDAR odometry algorithm is proposed for rapid registration between input data and various observed point clouds, as well as for calculating the distances between them. These distances are then used to construct a set of joint equations for multi-sphere localization. Finally, An iterative optimization method is proposed to solve this set of equations and determine the optimal position of the input data in the map. The contribution can be summarized as four-fold:

- A novel global localization strategy is proposed, which is less dependent on a pre-built map and effectively avoids errors introduced by mapping.
- A feature-based registration method is proposed, which enhances efficiency while maintaining insensitivity to initial values and large translations.
- An iterative optimization algorithm is proposed to solve the system of nonlinear equations in multi-sphere localization, ensuring accurate position estimation, particularly in the z -direction.
- Extensive experiments are carried out on different scenarios to verify the effectiveness of the proposed method, with a emphasis on location estimation metrics.

Related Work

Place Recognition Only. Place recognition only methods estimate the position based on the place retrieved through descriptor matching. Hence, designing highly discriminative and representative descriptors is crucial. Handcrafted feature based methods generally include point cloud compression approaches like histogramming, voxelization, segmentation, or projection. For example, Rusu et al. (Rusu, Blodow, and Beetz 2009) proposed the fast point feature histogram (FPFH), encoding relationships between neighbors of feature points into histograms. The similarity measure based on fast histogram, proposed by Röhling et al. (Röhling, Mack, and Schulz 2015), achieved place retrieval by statistically analyzing the distribution of point cloud ranges. Voxel-based representation learning (VBRL) (Siva, Nahman, and Zhang 2020) extracted multi-modal features from voxelized point clouds for place recognition. Fan et al. (Fan, He, and Tan 2020) introduced a descriptor Seed, enabling place recognition by encoding the topological infor-

mation of segmented objects. In addition to directly generating descriptors on point clouds and segments, there are various methods for mapping 3D point clouds to 2D planes, e.g., scan context and its variations. Kim et al. (2018) divided the point cloud into several bins based on radial and azimuth directions, and encoded the maximum height within each bin. In their later work, scan context++ (Kim, Choi, and Kim 2021), they further explored the descriptor's translational invariance. Building on their research, Wang et al. (Wang et al. 2020) introduced the concept of iris signatures and employed Fourier transform for similarity calculation. Kihara et al. (Kihara et al. 2022) combined Fourier transform and cross-correlation to enhance the matching efficiency between descriptors.

However, traditional manual methods struggle with invariance issues due to limitations in their descriptive capabilities. With the popularity of deep learning, some convolutional neural network models are exploited to address this issue. PointNetVLAD (Uy and Lee 2018), LPD-Net (2019), OverlapTransformer (2022), and other deep learning methods designed special structures to learn feature descriptors from large volumes of raw point cloud data. However, due to their reliance on a tremendous amount of training data, they struggle to generalize well across diverse scenarios or when faced with varying data acquisition conditions. Moreover, both manual methods and deep learning approaches aforementioned retrieve the nearest location from the database based on the current data, leading to inaccurate localization.

Place Recognition Followed by Local Pose Estimation. The approach of place recognition followed by local pose estimation is carried out in two separate stages: the first stage performs coarse place retrieval, and in the second stage, location estimation is refined through registering input data with map data attached to the retrieved place. The field of point cloud registration has made remarkable progress, with the emergence of several notable algorithms such as GICP (Segal, Haehnel, and Thrun 2009), GO-ICP (Yang et al. 2015), and KISS-ICP (Vizzo et al. 2023), in addition to various ICP variants. However, when it comes to the second stage of localization methods above, ICP and NDT remain the commonly used methods. Furthermore, research on two-stage localization methods is relatively limited. Shi et al. (2021) enhanced descriptor rotational invariance using principal component analysis (PCA) (Wold, Esbensen, and Geladi 1987) and achieved precise initial position by applying NDT. Luo et al. (2022) utilized point normals encoded histograms for coarse localization, with the option of using ICP refinement to optimize the global pose. Li et al. (Li et al. 2021) enhanced the scan context with semantic information and utilized a two-stage semantic ICP to estimate the pose. Chen et al. introduced OverlapNet (Chen et al. 2022), a dual-channel deep neural network trained on LiDAR data to estimate the overlap and yaw angle between point cloud pairs. Similarly, they applied ICP to refine the initial estimation.

Scant research exists on the second stage could be attributed to the challenges associated with handling two is-

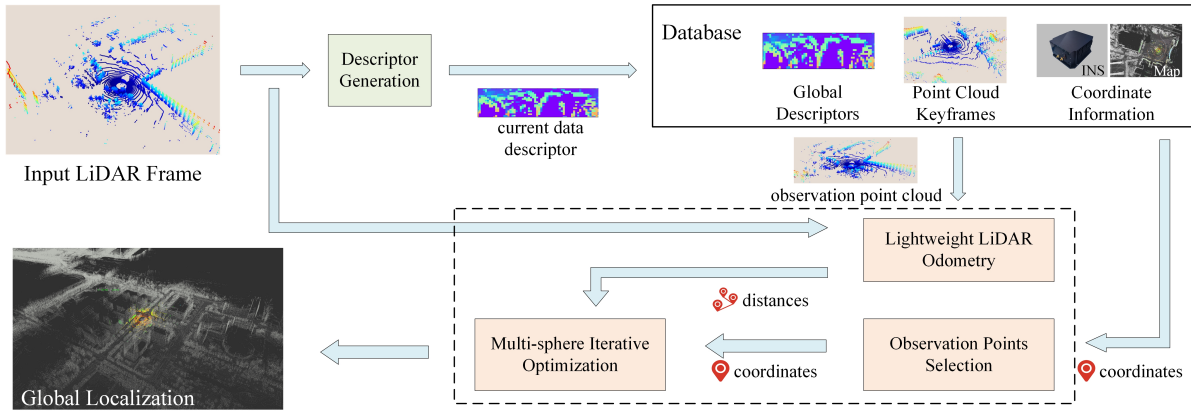


Figure 2: Overall pipeline of multi-constellation-inspired global localization. The content within the dashed rectangular box represents the core components of the proposed method in this paper.

sues in current methods. Firstly, the sparsity of keyframes leads to uncertainty in the translation and overlap between the retrieved data and the map data, subsequently affecting the accuracy and stability of registration algorithms. Secondly, point cloud registration is performed in the point cloud coordinate system, and transforming it to the ground truth coordinate system will inevitably introduce new errors. The motivation of this paper is to directly achieve position estimation in x , y , and z coordinates using multi-constellation localization mechanisms, thereby reducing the method's reliance on mapping and avoiding errors introduced by coordinate system transformation.

Methodology

Problem Definition

The mechanism of the proposed method bears resemblance to the multi-constellation localization system, which employs multiple satellites to establish three-dimensional spatial relationships for localization. In this paper, the global localization problem is converted into the establishment and optimization of a nonlinear equation system based on triangulation. Given n observation points o_i for $i = \{1, 2, \dots, n\}$, and the robot's position is denoted as s . Then the final optimal estimated position $\hat{s}$ of the robot is formalized as:

$$\hat{s} = \underset{s}{\operatorname{argmin}} \sum_{i=1}^n (\|s - o_i\| - d_i)^2, \quad (1)$$

where d_i is the estimated distance between each observation point and the robot.

System Overview

The preceding definition and the content formulated in Eq. (1) actually encompass three aspects that need to be addressed. Firstly, it entails the selection of observation points o_i , involving the decision on which data should be utilized as the observed position. Secondly, it deals with estimating the distances d_i between the observation points and the localization data. Lastly, it revolves around solving for the optimal position $\hat{s}$ that minimizes the final error.

Indeed, these aspects precisely correspond to the three crucial steps of the proposed method: (1) selection of observation points, (2) lightweight LiDAR odometry, and (3) multi-sphere iterative optimization. Firstly, a coarse localization is performed based on global descriptors, and observation points are then selected near that localization using a fixed strategy. The registration and distance calculation between the observed point cloud data and the query data are performed in the second step. In the final step, a multi-sphere nonlinear equation system is constructed using observed points' coordinates and distances from the query data, as to iteratively optimize the precise position of the query data. A visual overview of our multi-constellation-inspired localization method is depicted in Fig. 2.

Selection of Observation Points

The selection of observation points is essentially a database querying process. To facilitate retrieval, each element in the database is structured as a triplet $e_i = (k_i, g_i, o_i)$, comprising a point cloud keyframe k_i , a global descriptor g_i derived from the point cloud, and corresponding coordinate information o_i . $i \in \{1, 2, \dots, N\}$, N is the size of database. The keyframe plays a vital role in the second step of the proposed method, facilitating registration and distance calculation with the input data. The global descriptor serves as a keyword, allowing for the rapid retrieval of a coarse position. The coordinate information stores the relative 3D vector of each point cloud frame with respect to the first point cloud frame. This information can be provided by initial navigation system (INS), simultaneous localization and mapping (SLAM), or other positioning devices.

Given an arbitrary frame of input LiDAR data k_{query} during localization, the retrieval progress can be formulated as:

$$c = \underset{c \in \{1, 2, \dots, N\}}{\operatorname{argmax}} \operatorname{sim}(f(k_{query}), g_c), \quad (2)$$

where the function $f(\cdot)$ represents the process of converting input data into descriptors. The function $\operatorname{sim}(\cdot)$ returns the similarity between two global descriptors. If an index c is found, we obtain a coarse position o_c in e_c mentioned in

Algorithm 1: planar point downsampling**Input:** sorted planar point set Q , first point q_0 in set Q **Output:** the downsampled set Q'

```

1: Let  $Q' = \{q_0\}$ ,  $q = q_0$ .
2: for  $q_j$  in  $Q$  do
3:   if  $\|q - q_j\| > \sigma$  then
4:      $Q' = Q' \cup \{q_j\}$ 
5:      $q = q_j$ 
6:   end if
7: end for
8: return  $Q'$ 

```

the two-stage localization, which is also one of the observation points we want to select. To meet the requirement of at least three observed points for the equation system and ensure overlapping areas between point clouds for registration, the neighboring positions o_{c-u} to o_{c+v} , along with o_c , are selected as the observation points. u and v satisfy $u + v = n - 1$. n is the total number of observation points, including the one found through retrieval. u and v represent the number of observation points that precede and follow this found point, respectively. This completes the acquisition of observation position in Eq. (1).

Lightweight LiDAR Odometry

In the real multi-constellation localization problem, distances between the receiver and satellites are obtained using time of flight (TOF) measurements. However, the estimation of distances between the robot and observation points requires point cloud registration as the initial step. Due to the requirement for multiple registrations of various observation data with the query data, using NDT, ICP, or their variants would have a considerable impact on the real-time performance. Furthermore, considering the requirement for stability, we make modifications to the LiDAR odometry and mapping (LOAM) (Zhang and Singh 2014) to better align with the localization process proposed in this paper.

Firstly, lightweight is performed to reduce the number of feature points involved in registration, ensuring real-time performance. More specifically, we omit the edge point extraction part of the LOAM method and apply Algorithm 1 to filter the planar points. If the distance between the current point q and the subsequent point q_j is greater than a predefined threshold, the point q_j is added to a new set of planar points, and the current point q is updated. Otherwise, the iteration continues to traverse the remaining planar points.

Furthermore, the loss function is modified to mitigate the impact of initial values and large translations. The pose between the input point cloud k_{query} and the observation point cloud k_i is denoted by a , with its initial value $a_0 = [r_x \ r_y \ r_z \ t_x \ t_y \ t_z]^T$. The corresponding translation matrix is given by:

$$t = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

the rotation matrices around three axes are given by:

$$R_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(r_x) & -\sin(r_x) & 0 \\ 0 & \sin(r_x) & \cos(r_x) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4)$$

$$R_y = \begin{bmatrix} \cos(r_y) & 0 & \sin(r_y) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(r_y) & 0 & \cos(r_y) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5)$$

$$R_z = \begin{bmatrix} \cos(r_z) & -\sin(r_z) & 0 & 0 \\ \sin(r_z) & \cos(r_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6)$$

All points q_j in the planar point set Q' , which extracted from k_{query} , are transformed into the coordinate system of k_i , yields the following equation:

$$q'_j = t \cdot R_z \cdot R_y \cdot R_x \cdot q_j, \quad (7)$$

the points near q'_j in k_i is searched for constructing a plane equation $Ax + By + Cz + D = 1$, $A^2 + B^2 + C^2 = 1$. The distance from the point q'_j to the plane can be calculated as:

$$dis(q'_j, k_i) = |A \cdot x_{q'_j} + B \cdot y_{q'_j} + C \cdot z_{q'_j} + D|. \quad (8)$$

Thus, the cost function can be represented as:

$$loss = \sum_j dis(q'_j, k_i), \quad (9)$$

using the Gauss-Newton method for optimization, the Jacobian matrix is $J = \begin{bmatrix} \frac{\partial loss}{\partial r_x} & \frac{\partial loss}{\partial r_y} & \frac{\partial loss}{\partial r_z} & \frac{\partial loss}{\partial t_x} & \frac{\partial loss}{\partial t_y} & \frac{\partial loss}{\partial t_z} \end{bmatrix}^T$. Then the constructed linear system of equations is given as:

$$J \cdot J^T \cdot \Delta a = -J \cdot loss. \quad (10)$$

The linear system of equations is iteratively solved to obtain the parameter update Δa , which is then used to update the parameters a . The iteration continues until the objective function converges to its minimum or meets a specified convergence criterion. Once the parameters a is obtained, the distance d_i between the current data and the observed data can be calculated using its components t_x , t_y , and t_z :

$$d_i = \sqrt{t_x^2 + t_y^2 + t_z^2}. \quad (11)$$

Multi-sphere Iterative Optimization

After completing the two aforementioned steps, the coordinates of three observed points, namely o_i for $i = \{1, 2, 3\}$, along with their corresponding distances to the localization point (position of the robot), denoted as d_i for $i = \{1, 2, 3\}$ are obtained. Then, the localization point s can be determined by solving the following set of equations:

$$(\|s - o_i\|) - d_i = 0. \quad (12)$$

If Eq. (12) has solutions, the position of the robot can be represented as s_1 or s_2 . In a multi-constellation localization system, the distance between the receiver and satellites

is very large, making it easy to distinguish which solution lies on the Earth’s surface. The two solutions are close to each other in our method, considering that the registration requires a certain overlap between the point clouds. It requires solving at least 4 sets of Eq. (12) to find their shared solution. However, due to the errors introduced during registration, inaccurate estimation of d_i , finding a consistent solution becomes infeasible. Therefore, this problem needs to be transformed into an optimization problem, as illustrated in Eq. (1).

To address the aforementioned issue, a novel multi-sphere iterative optimization method is proposed for obtaining the optimal position. Firstly, the initial value $\hat{s}_0$ is set using the results obtained from the three-sphere localization as:

$$\hat{s}_0 = \frac{s_1 + s_2}{2}. \quad (13)$$

Then, the iterative process can be formulated as follows:

$$m_i = \|o_i - \hat{s}_l\|, \quad (14)$$

$$\hat{s}_{l+1} = \hat{s}_l + \sum_{i=1}^n \frac{(m_i - d_i) \cdot (o_i - \hat{s}_l)}{m_i} \cdot \rho, \quad (15)$$

$$\tau_{l+1} = \sum_{i=1}^n \|o_i - \hat{s}_{l+1}\| - d_i, \quad (16)$$

where n is the number of observed points, and n is greater than or equal to 4. ρ is theoretically a function that takes the registration error variance as input, but for simplification in this paper, it is set as a constant. τ_{l+1} refers to the iteration error. The iteration stops when $\tau_{l+1} - \tau_l < \varepsilon$ and $l < \gamma$. γ is an iterative termination condition.

Experiments

In this section, we will describe the dataset, experimental details, and a series of experiments which can demonstrate the performance of the proposed method. All experiments are conducted on a computer equipped with an Intel Core i7-1165G7 processor and 32GB of RAM. All methods are implemented in C++ and executed on Ubuntu Linux.

Dataset

KITTI Dataset. KITTI dataset is released in (Geiger, Lenz, and Urtasun 2012), which provides 3D point clouds generated by Velodyne-HDL64e LiDAR and ground truth provided by OxTS-RT3000. In this paper, we choose sequences “02”, “05” and “07” as benchmarks because of their trajectory lengths cover a diverse range of distances.

Self-collected Dataset. The self-collected dataset involves three sequences: two obtained using a passenger vehicle (Volkswagen Tiguan) in a campus scenario, and the other acquired with an off-road platform (Avenger) in hilly area. The data format for all sequences is consistent with KITTI. Point clouds is generated by Velodyne-HDL32e and VLP16¹,

while ground truth is obtained using a Npos220s². The data collection platforms are shown in Fig. 3.



Figure 3: Overview of data collection platforms, (a) Volkswagen Tiguan with Velodyne-HDL32e scanner, (b) Avenger with Velodyne-VLP16 scanner.

Evaluation Metric

We deploy the translation error and its standard deviation to evaluate the localization performance. The translation error err_i between each estimated position $\hat{s}_i$ and the ground truth gt_i is given by:

$$err_i = \|\hat{s}_i - gt_i\|, \quad (17)$$

and compute the standard deviation std is calculated as:

$$std = \sqrt{\frac{\sum_i (err_i - \bar{err})^2}{n}}, \quad (18)$$

where $\bar{err}$ is the average translation error of all estimated positions.

Experimental Setup

We adopt the strategy proposed by Shi et al. (Shi et al. 2021), employing the scan context for coarse localization in the first stage. Subsequently, comparative experiments are conducted using the methods introduced in this paper, alongside ICP, NDT, GICP, and KISS-ICP, in the second stage. For simplicity, these comparative methods are referred as Sc-Icp, Sc-Ndt, Sc-Gicp, and Sc-Kicp, respectively.

Because of the need for a pre-built map in these comparative methods, LIO-SAM (Shan et al. 2020) is employed in this paper, and global navigation satellite system (GNSS) factors are incorporated for mapping. Sequences “02”, “05”, and “07” are individually divided into two parts. Starting from the initial frame of each sequence, the point cloud is employed for mapping every 0.2 second, while the remaining data is dedicated to localization tasks. Moreover, the number of observed points in the proposed method is set to 5 (the preceding two and the subsequent two). To quantitatively analyze the experiments, we also align the mapping trajectory and GNSS ground truth trajectory using the NDT method. The average errors and standard deviations obtained by randomly selecting 80 points from the localization data.

¹<https://velodynelidar.com/products/puck/>

²<https://www.gpsolution.com/porduct-info/110.html>

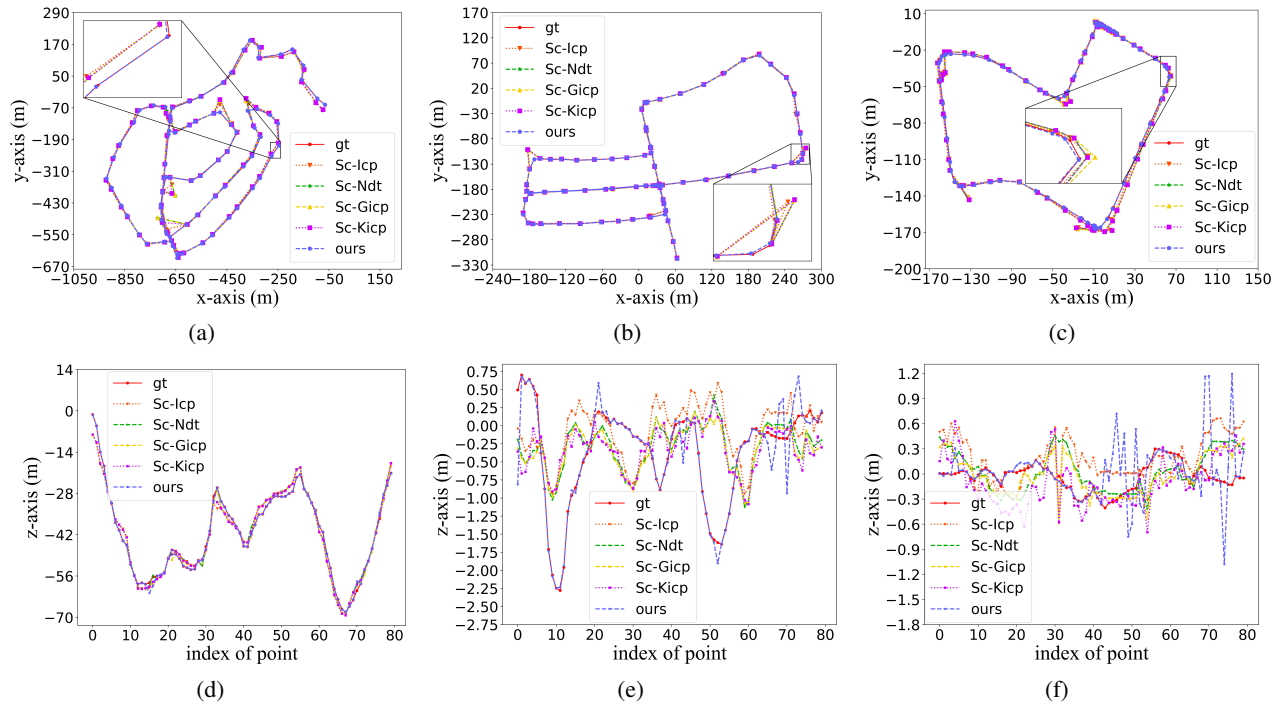


Figure 4: Visualization of localization results on the KITTI dataset: Figures (a), (b), and (c) present the horizontal comparison between 80 randomly sampled points estimated by each method and the ground truth for sequences 02, 05, and 07. Figures (d), (e), and (f) illustrate the vertical comparison for the same sequences.

Methods	KITTI “02”							KITTI “05”							KITTI “07”						
	avg			std				avg			std				avg			std			
	total	xoy	z	total	xoy	z	total	xoy	z	total	xoy	z	total	xoy	z	total	xoy	z	total	xoy	z
Sc-Icp	7.95	7.65	1.66	7.49	7.50	1.34	2.26	2.02	0.61	4.42	4.47	0.53	2.90	2.84	0.31	3.86	3.89	0.21			
Sc-Ndt	9.65	9.35	1.59	15.87	15.93	1.17	2.17	1.94	0.57	4.90	4.94	0.46	3.02	2.98	0.24	4.35	4.37	0.15			
Sc-Gicp	9.45	9.16	1.51	15.00	15.05	1.18	2.22	2.00	0.58	4.78	4.82	0.44	3.03	3.00	0.22	4.15	4.17	0.14			
Sc-Kicp	9.10	8.80	1.61	13.40	13.45	1.23	2.15	1.91	0.58	4.89	4.94	0.47	3.01	2.98	0.26	4.34	4.35	0.18			
Ours	1.28	1.16	0.25	0.65	0.45	0.68	0.77	0.74	0.09	0.35	0.35	0.21	0.90	0.83	0.17	0.32	0.32	0.31			

Table 1: Comparison of average errors and standard deviations for global localization on KITTI “02”, “05”, and “07”. “total” represents the metrics along the x , y , and z axes. “xoy” represents the metrics along the x and y axes. “z” represents the metrics along the z axis.

Test on KITTI Dataset

Due to the vital role of descriptors in ensuring the success rate of localization, the accuracy of localization, when successful and with correct position retrieval, is heavily reliant on the performance of the registration approach. Table 1 indicates the proposed method in this paper outperforms the competing methods by a considerable margin, owing to the implementation of a novel localization strategy. Averaging across all three sequences, our total localization error is 0.98 meters, 3.39 meters smaller than the error reported by Luo et al. for the ICP method in their paper, 3.97 meters smaller than NDT mentioned in Shi et al.’s paper. Compared to G-ICP and the latest KISS-ICP, the errors are reduced by 3.92 meters and 3.77 meters, respectively. In addition, our total standard deviation is 0.44 meters, reduced by 4.82 meters, 7.93 meters, 7.54 meters, and 7.1 meters compared to the comparative methods. The above results demonstrate that

our method not only achieves more accurate localization on the KITTI dataset but also exhibits remarkable stability.

To qualitatively showcase our results on the public dataset, we visualized the localization outcomes in both the horizontal and vertical directions, as depicted in Fig. 4. From Fig. 4 (a), (b), and (c), it is evident that the proposed multi-constellation inspired localization method exhibits a better alignment with the ground truth in the xoy plane when using the same descriptor and coarse positioning conditions. In contrast, other comparative methods exhibit larger localization errors when encountering significant turning maneuvers. In the vertical direction, as shown in Fig. 7(d), (e), and (f), the performance of all methods is relatively inferior compared to the horizontal direction. Our method shows z -values that are close to the ground truth in most localization points, with some minor fluctuations observed in a few cases. The fluctuation may be attributed to factors such

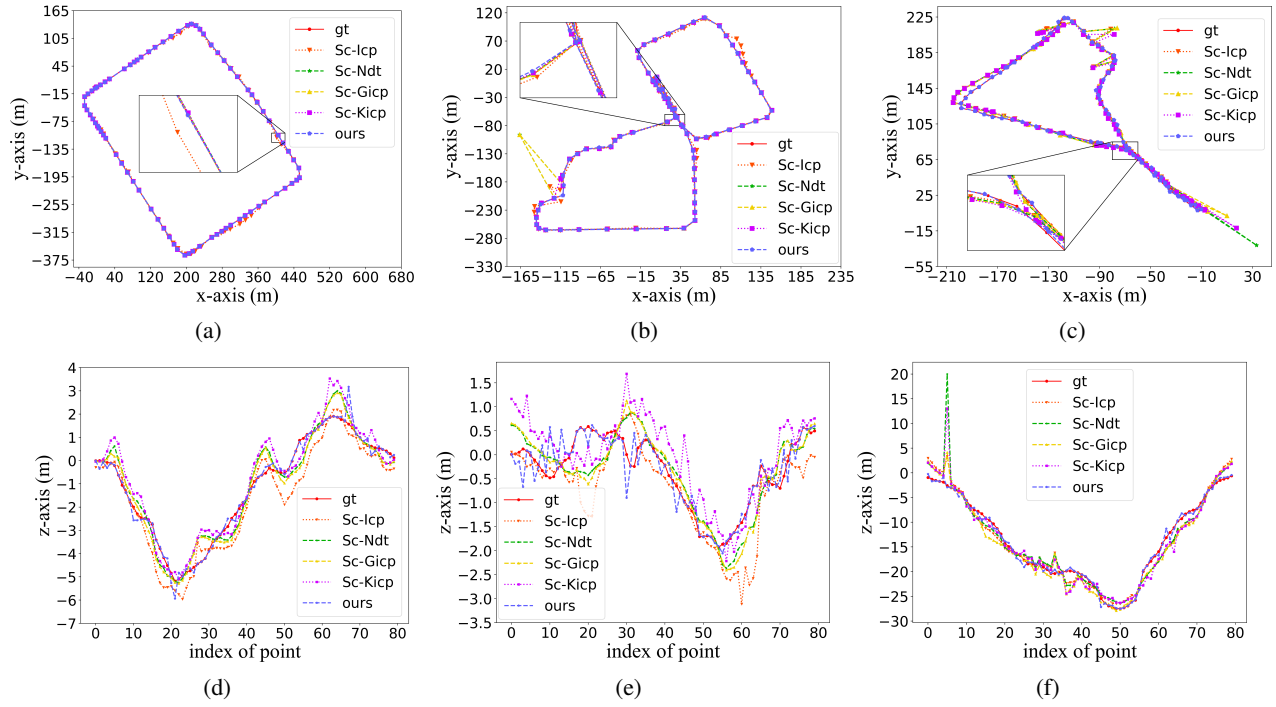


Figure 5: Visualization of localization results on the self-collected dataset. Figures (a), (b), (d), and (e) show results in the campus scenario, while the others are in wilderness.

Methods	Campus “01”						Campus “02”						Wilderness					
	avg			std			avg			std			avg			std		
	total	xoy	z	total	xoy	z	total	xoy	z	total	xoy	z	total	xoy	z	total	xoy	z
Sc-lcp	2.66	2.52	0.58	2.22	2.26	0.43	3.38	3.30	0.50	3.96	3.96	0.50	5.41	5.05	1.51	5.54	5.58	1.01
Sc-Ndt	0.78	0.60	0.42	0.33	0.28	0.33	2.41	2.25	0.38	13.30	13.32	0.27	6.38	6.04	1.61	11.59	11.40	2.48
Sc-Gicp	0.87	0.69	0.43	0.37	0.34	0.33	2.45	2.27	0.40	13.30	13.32	0.30	6.20	5.84	1.51	7.77	7.80	1.29
Sc-Kicp	0.89	0.62	0.55	0.41	0.31	0.43	1.44	1.20	0.53	3.27	3.30	0.36	6.88	6.61	1.60	9.00	8.86	1.86
Ours	0.61	0.57	0.14	0.33	0.30	0.22	0.64	0.59	0.17	0.41	0.38	0.23	1.13	0.96	0.41	1.77	1.77	0.41

Table 2: Comparison of average errors and standard deviations for global localization on self-collected dataset.

as occlusion and motion distortion, leading to suboptimal initial values provided by the registration. In contrast, the comparative methods deviate from the ground truth in the majority of cases and only occasionally estimate the correct z -values accurately.

Test on Self-collected Dataset

To further validate the effectiveness and generality of our proposed method, we conducted a series of experiments on self-collected data. Apart from increasing the keyframe selection interval from the original 0.2 seconds to 0.4 seconds, all other experimental settings remained unchanged. The average errors and standard deviations in campus and hilly area are shown in Table 2.

As with the previous experiments, we also visualized the localization trajectories, depicted in Fig. 5. The results from Table 2 and Fig. 5 support the conclusion that the proposed method outperforms the comparative approaches in terms of performance. On the two campus datasets, our average total localization error reaches the decimeter level, at 0.625

meters. The fluctuation is also relatively small, with an average total standard deviation of 0.37 meters. The results on hilly area data also show superior performance compared to the comparative methods. The total localization error is 1.13 meters, which is only 20.89% of the second-ranked method, and the total standard deviation is 1.77 meters, which is only 31.95% of the same method. It is worth noting that our proposed method also provides more accurate estimates in the z -axis, as demonstrated in the KITTI dataset’s sequence 02 (Fig. 4 (d)) and hilly area data (Fig. 5 (f)). In certain applications, such as mountainous rescue robots or autonomous driving in a viaduct scenario, having a more precise estimation in the z -axis becomes imperative. The improved accuracy can significantly enhance the performance of these systems.

The primary reason for obtaining such superior results is the implementation of our novel approach during the fine localization process. Compared to common registration-based approaches, our advantages mainly lie in two aspects. Firstly, registration-based methods rely on the relative po-

Scenario	Sc-Icp	Sc-Ndt	Sc-Gicp	Sc-Kicp	Ours
KITTI “02”	1.759	0.541	2.172	0.033	0.359
KITTI “05”	1.634	0.599	2.769	0.024	0.403
KITTI “07”	1.755	0.658	3.626	0.022	0.310
Campus “01”	1.418	0.509	3.178	0.044	0.139
Campus “02”	1.919	0.672	4.465	0.060	0.145
Wilderness	1.123	0.646	4.097	0.055	0.358

Table 3: Runtime of methods for processing one scan (s)

sition relationship between the current data and the reference data or the local map of the reference data for localization. The computation is performed only once, and if impacted by initial values or significant translations, the localization results lack redundant measures for rectification. In contrast, our method leverages multiple observations and employs optimization techniques to estimate the position. This approach helps mitigate errors caused by occasional incorrect observations to a considerable extent. Furthermore, our method directly estimates the coordinate positions, resulting in errors primarily arising from computations when compared to ground truth. Thus, the estimation of all three axes becomes more accurate, particularly the z -axis. In contrast, registration-based approaches introduce errors from map building, coordinate system transformations between ground truth and map coordinate systems, as well as computational errors.

Runtime Comparison

The runtime for each method is shown in Table 3. The running time is recorded for both coarse localization and fine registration steps. Sc-Kicp stands out as the fastest-performing method. Our method is ranked second, with an average processing speed of 0.357 s per frame for KITTI dataset and 0.214 s per frame for self-collected dataset. The above results demonstrate that our method can achieve real-time initial localization (excluding tracking), attributed to our lightweight registration method and concurrency.

Conclusion

In this paper, a novel single-shot global localization method is introduced, which leverages triangulation formed between retrieval data and multiple observed data based on their coordinates and distances. In comparison to conventional approaches, this method diminishes the dependency on point cloud maps and broadens its applicability. In addition, an improved LiDAR odometry method is proposed to estimate the distances. Its lightweight processing enhances the efficiency of the method. Lastly, utilizing the obtained coordinate and distance information, a system of multiple-sphere equations is constructed, and iterative optimization is employed to achieve accurate localization in the 3 degrees of freedom: x , y , and z . Extensive experiments validate the real-time capability, robustness, and generality of the proposed method. This is evident in localization times of under half a second (0.286 s), as well as achieving an average total localization error of approximately 1 meter (0.89 m)

in urban, campus, and wild environments. Additionally, the method excels in estimating elevation values.

Acknowledgments

This research is supported by the Jilin Scientific and Technological Development Program (20210401145YY), the Changsha Automobile Innovation Research Institute (CAIRIZT20220101), the National Key Research and Development Program of China (2023YFE0197800), and the science foundation of national key laboratory of science and technology on advanced composites in special environments (JCKYS2023603C014).

References

- Besl, P. J.; and McKay, N. D. 1992. Method for registration of 3-D shapes. In *Sensor fusion IV: control paradigms and data structures*, 586–606. Boston, MA, USA: Spie.
- Biber, P.; and Straßer, W. 2003. The normal distributions transform: A new approach to laser scan matching. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2743–2748. Las Vegas, NV, USA: IEEE.
- Chen, X.; Läbe, T.; Milioto, A.; Röhling, T.; Behley, J.; and Stachniss, C. 2022. OverlapNet: A siamese network for computing LiDAR scan similarity with applications to loop closing and localization. *Autonomous Robots*, 1–21.
- Cop, K. P.; Borges, P. V.; and Dubé, R. 2018. Delight: An efficient descriptor for global localisation using lidar intensities. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 3653–3660. Brisbane, QLD, Australia: IEEE.
- Dubé, R.; Cramariuc, A.; Dugas, D.; Sommer, H.; Dymczyk, M.; Nieto, J.; Siegwart, R.; and Cadena, C. 2020. SegMap: Segment-Based Mapping and Localization Using Data-Driven Descriptors. *Int. J. Rob. Res.*, 39(2–3): 339–355.
- Fan, Y.; He, Y.; and Tan, U.-X. 2020. Seed: A segmentation-based egocentric 3D point cloud descriptor for loop closure detection. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5158–5163. Las Vegas, NV, USA: IEEE.
- Geiger, A.; Lenz, P.; and Urtasun, R. 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition (CVPR)*, 3354–3361. Providence, RI, USA: IEEE.

- Kihara, H.; Kumon, M.; Nakatsuma, K.; and Furukawa, T. 2022. Fast Scan Context Matching for Omnidirectional 3D Scan. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 925–930. Kyoto, Japan: IEEE.
- Kim, G.; Choi, S.; and Kim, A. 2021. Scan context++: Structural place recognition robust to rotation and lateral variations in urban environments. *IEEE Transactions on Robotics*, 38(3): 1856–1874.
- Kim, G.; and Kim, A. 2018. Scan context: Egocentric spatial descriptor for place recognition within 3d point cloud map. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4802–4809. Madrid, Spain: IEEE.
- Li, L.; Kong, X.; Zhao, X.; Huang, T.; Li, W.; Wen, F.; Zhang, H.; and Liu, Y. 2021. SSC: Semantic scan context for large-scale place recognition. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2092–2099. Prague, Czech Republic: IEEE.
- Li, Y.; and Li, H. 2021. LiDAR-Based Initial Global Localization Using Two-Dimensional (2D) Submap Projection Image (SPI). In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 5063–5068. Xi'an, China: IEEE.
- Liu, Z.; Zhou, S.; Suo, C.; Yin, P.; Chen, W.; Wang, H.; Li, H.; and Liu, Y.-H. 2019. Lpd-net: 3d point cloud learning for large-scale place recognition and environment analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2831–2840. Seoul, Korea: IEEE.
- Luo, L.; Cao, S.-Y.; Sheng, Z.; and Shen, H.-L. 2022. LiDAR-based global localization using histogram of orientations of principal normals. *IEEE Transactions on Intelligent Vehicles*, 7(3): 771–782.
- Ma, J.; Zhang, J.; Xu, J.; Ai, R.; Gu, W.; and Chen, X. 2022. OverlapTransformer: An efficient and yaw-angle-invariant transformer network for LiDAR-based place recognition. *IEEE Robotics and Automation Letters*, 7(3): 6958–6965.
- Ratz, S.; Dymczyk, M.; Siegwart, R.; and Dubé, R. 2020. OneShot Global Localization: Instant LiDAR-Visual Pose Estimation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 5415–5421. Paris, France: IEEE.
- Röhling, T.; Mack, J.; and Schulz, D. 2015. A fast histogram-based similarity measure for detecting loop closures in 3-d lidar data. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, 736–741. Hamburg, Germany: IEEE.
- Rusu, R. B.; Blodow, N.; and Beetz, M. 2009. Fast point feature histograms (FPFH) for 3D registration. In *2009 IEEE international conference on robotics and automation (ICRA)*, 3212–3217. Kobe, Japan: IEEE.
- Segal, A.; Haehnel, D.; and Thrun, S. 2009. Generalized-icp. In *Robotics: science and systems*, 435. Seattle, WA.
- Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; and Rus, D. 2020. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, 5135–5142. Las Vegas, NV, USA: IEEE.
- Shi, X.; Chai, Z.; Zhou, Y.; Wu, J.; and Xiong, Z. 2021. Global place recognition using an improved scan context for lidar-based localization system. In *2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 498–503. Delft, Netherlands: IEEE.
- Siva, S.; Nahman, Z.; and Zhang, H. 2020. Voxel-based representation learning for place recognition based on 3d point clouds. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 8351–8357. Las Vegas, NV, USA: IEEE.
- Uy, M. A.; and Lee, G. H. 2018. Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 4470–4479. Salt Lake City, UT, USA: IEEE.
- Vizzo, I.; Guadagnino, T.; Mersch, B.; Wiesmann, L.; Behley, J.; and Stachniss, C. 2023. Kiss-icp: In defense of point-to-point icp—simple, accurate, and robust registration if done the right way. *IEEE Robotics and Automation Letters*, 8(2): 1029–1036.
- Wang, Y.; Sun, Z.; Xu, C.-Z.; Sarma, S. E.; Yang, J.; and Kong, H. 2020. Lidar iris for loop-closure detection. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5769–5775. Las Vegas, NV, USA: IEEE.
- Wold, S.; Esbensen, K.; and Geladi, P. 1987. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3): 37–52.
- Yang, J.; Li, H.; Campbell, D.; and Jia, Y. 2015. Go-ICP: A globally optimal solution to 3D ICP point-set registration. *IEEE transactions on pattern analysis and machine intelligence*, 38(11): 2241–2254.
- Yin, H.; Xu, X.; Lu, S.; Chen, X.; Xiong, R.; Shen, S.; Stachniss, C.; and Wang, Y. 2023. A Survey on Global LiDAR Localization: Challenges, Advances and Open Problems. arXiv:2302.07433.
- Yu, S.; Yan, F.; Zhuang, Y.; and Gu, D. 2020. A deep-learning-based strategy for kidnapped robot problem in similar indoor environment. *Journal of Intelligent & Robotic Systems*, 100: 765–775.
- Zhang, J.; and Singh, S. 2014. LOAM: Lidar odometry and mapping in real-time. In *Robotics: Science and systems*, 1–9. Berkeley, CA.