

DuLoc: Life-Long Dual-Layer Localization in Changing and Dynamic Expansive Scenarios

Haoxuan Jiang, Peicong Qian, Yusen Xie, Xiaocong Li, Ming Liu, and Jun Ma, *Senior Member, IEEE*

Abstract—LiDAR-based localization serves as a critical component in autonomous systems, yet existing approaches face persistent challenges in balancing repeatability, accuracy, and environmental adaptability. Traditional point cloud registration methods relying solely on offline maps often exhibit limited robustness against long-term environmental changes, leading to localization drift and reliability degradation in dynamic real-world scenarios. To address these challenges, this paper proposes DuLoc, a robust and accurate localization method that tightly couples LiDAR-inertial odometry with offline map-based localization, incorporating a constant-velocity motion model to mitigate outlier noise in real-world scenarios. Specifically, we develop a LiDAR-based localization framework that seamlessly integrates a prior global map with dynamic real-time local maps, enabling robust localization in unbounded and changing environments. Extensive real-world experiments in ultra unbounded port that involve 2,856 hours of operational data across 32 Intelligent Guided Vehicles (IGVs) are conducted and reported in this study. The results attained demonstrate that our system outperforms other state-of-the-art LiDAR localization systems in large-scale changing outdoor environments.

I. INTRODUCTION

High-precision life-long localization in large-scale environments faces fundamental challenges across various autonomous systems [1], [2], [3], [4]. While Real-Time Kinematic (RTK) localization has been widely adopted, its reliability is inherently limited by satellite signal availability and near metallic structures, with typical accuracy degradation from centimeters to meters in such scenarios. Map-based localization [5], [6] often encounters challenges due to environmental changes, leading to unstable and incorrect performance. These limitations drive the development of sensor-based localization approaches. Initial efforts focused on single-sensor solutions (e.g., LiDAR [7] or cameras [8]), but their susceptibility to environmental changes and perceptual degradation in feature-deprived areas proved inadequate for large-scale deployments [9], [10], [11]. The evolution to

This work was supported by the Guangdong provincial project under Grant 2023QN10Z006. (*Corresponding author: Jun Ma*)

Haoxuan Jiang and Yusen Xie are with Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511453, China (e-mail: hjiangax@connect.hkust-gz.edu.cn; yxie827@connect.hk-gz.edu.cn).

Peicong Qian and Ming Liu are with Shenzhen Unity Drive Innovation Technology Co., Ltd., Shenzhen 518063, China (e-mail: epsilonjohn9527@gmail.com; liu.ming.prc@gmail.com).

Xiaocong Li is with the College of Information Science and Technology, Eastern Institute of Technology, Ningbo, Ningbo 315200, China (e-mail: xiaocongli@eitech.edu.cn).

Jun Ma is with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511453, China, and also with the Division of Emerging Interdisciplinary Areas, The Hong Kong University of Science and Technology, Hong Kong SAR, China (e-mail: jun.ma@ust.hk).

multi-sensor fusion systems [12], [13], [14] address some shortcomings by combining complementary sensing modalities. However, these approaches still struggle with long-term consistency due to cumulative errors in odometry estimation and the absence of global constraints. Even state-of-the-art fusion methods exhibit gradual drift in kilometer-scale operations [9], [10], [11], particularly in dynamic environments where moving objects distort perception observations. This fundamental limitation underscores the need for persistent global references in large-scale localization.

To address these issues, in this paper, we propose a novel tightly-coupled framework that incorporates LiDAR-IMU odometry, integrates global offline map constraints, and leverages the constant velocity (CV) model, specifically designed to achieve accurate and robust localization in large-scale industrial environments. Firstly, we propose a tightly-coupled LiDAR-Inertial-CV odometry backbone based on an iterated Error-State Kalman Filter (iESKF) that ensures robust short-term tracking. Then, we design a dual-map architecture combining a static prior map with a dynamic local map, enabling simultaneous global consistency and local adaptability. Finally, we validate our approach in one of the most demanding real-world environments: automated port operations. Essentially, modern container ports present extreme localization challenges [15], including ultra large-scale unbounded scenes with persistent GPS-denial areas and severe multipath effects caused by dense metallic obstructions. The frequent cargo handling, vehicle movement, and machinery operations further intensify the highly dynamic nature of these settings, rendering precise localization exceptionally difficult. In addition, the challenges of mapping in such large areas are compounded by high mapping costs, difficulty in maintenance, high environmental repetitiveness, and the stringent accuracy requirements for localization. In our experiments, 2,856 hours of operational data across 32 Intelligent Guided Vehicles (IGVs) are involved. Our system demonstrates centimeter-level accuracy (mean: 8.3 cm, σ : 4.1 cm) while maintaining 99.98% availability, outperforming existing LiDAR-SLAM baselines [13], [14] that failed catastrophically within 30 minutes of operation due to misalignment of the map or odometry drift.

The main contributions of our work are summarized as follows:

- We present a highly efficient dual-map LiDAR localization framework integrated into a multi-sensor tightly-coupled localization system, specifically designed for real-world large-scale, dynamic, and rapidly changing environments.

- We import an optimized Kalman gain computation method to improve computational efficiency and enhance the accuracy of point cloud feature alignment during the filter measurement update process.
 - We propose a constant velocity (CV) model operating at up to 100 Hz, ensuring stable and continuous pose prediction even in case of sensor failures.
 - Our algorithm is deployed and validated in a port covering an area of approximately one million square meters, providing accurate and stable localization support for 32 IGVs; and this underscores the effectiveness of the proposed method in real-world settings.

II. RELATED WORK

A. LiDAR-Based SLAM

LiDAR-based SLAM methods are widely used for map construction and self-localization. LeGO-LOAM [16] and F-LOAM [14] both utilize ground planes for feature extraction and matching, with F-LOAM introducing a two-stage distortion compensation method. However, these methods struggle with handling dynamic environments and feature redundancy. SROM [13] uses a two-layer approach to estimate rotation and translation, refining the results with point-to-plane ICP [17], [18], though it can be computationally expensive. PFfilter [19], an extension of F-LOAM, filters invalid features to improve accuracy but may still suffer from high computational overhead in large-scale scenarios. FEVO-LOAM [20] optimizes feature extraction with enhanced ground segmentation and curvature definitions, yet it may face limitations in complex urban environments with varied terrain. RF-LOAM [21] uses FA-RANSAC to remove dynamic objects, but its effectiveness diminishes with extreme environmental changes. WiCRF2 [22] enhances motion observability and minimizes redundancy, but it may struggle in highly dynamic or unstructured environments. The method in [23] employs angle-based feature extraction and voxel-based feature matching but may be sensitive to LiDAR viewpoint variations. CDP-LOAM [24] introduces clustering-directed points for attitude estimation, yet it is computationally heavy. Light-LOAM [25] offers a two-stage correspondence strategy for reliable registration but may not scale well in large-scale dynamic environments.

B. Map-Based LiDAR Localization

Recent map-based LiDAR localization methods focus on balancing accuracy, efficiency, and adaptability. LOL [26] reduces cumulative drift via geometric place recognition but struggles with partially similar scenes, suggesting dynamic map updates. DLL [5] optimizes point-to-map distances without feature extraction, yet its computational intensity demands hierarchical optimization for scalability. For dynamic environments, ROLL [6] activates temporary mapping during global matching failures, though latency issues necessitate adaptive triggering, while DMLL [27] mitigates alignment errors through differential constraints but requires probabilistic error modeling to address complex uncertainties. Efficiency-driven approaches (Thakur [28] and Block Map

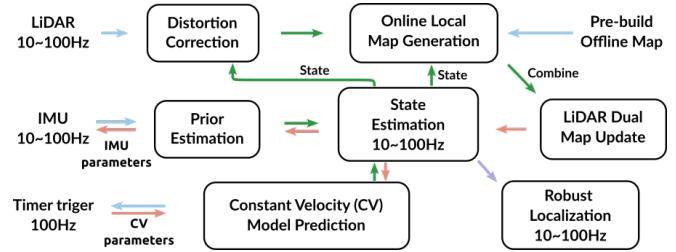


Fig. 1: The overall architecture of our proposed system, based on the iESKF filter, tightly integrates LiDAR, IMU data and constant velocity model for high-precision localization. The *blue* arrows indicate data input, the *green* arrows represent the forward process, the *orange* arrows depict the constraint and covariance propagation, and the *purple* arrows signify the output of the localization results.

[29]) leverage lightweight maps and block switching for large-scale applications, yet risk accuracy loss or transition misalignments, calling for multi-resolution maps and boundary refinement.

C. Multi-Sensor Fused Localization

Multi-sensor fusion methods combine LiDAR with inertial measurements to address motion distortion and environmental dynamics. Tightly-coupled frameworks dominate recent advancements, where LINS [30] and LIO-EKF [31] utilize iterated/extended Kalman filters with adaptive error compensation, though their computational load escalates in large-scale environments. Meanwhile, LIO-SAM [32] enhances real-time performance through factor graphs and IMU pre-integration, while the FAST-LIO series [12], [33] innovates with state-dependent Kalman gain and raw point-to-map registration (FAST-LIO2 [33]), significantly reducing feature extraction overhead at the cost of degraded accuracy during aggressive maneuvers. For high-dynamic scenarios, Point-LIO [34] introduces stochastic kinematic modeling to handle IMU saturation, yet its efficacy remains constrained by inertial sensor quality.

III. METHODOLOGY

As illustrated in Fig.1, our framework comprises several key modules: the system model in Sec. III-A, constant velocity prediction model in Sec. III-B, dual-map update strategy in Sec. III-C, IMU correction model in Sec. III-D, and additional implementation details including post-processing and LiDAR motion compensation in Sec. III-E.

A. System Description

Firstly, we define the state vector \mathbf{x} of our system as

$$\mathbf{x} \triangleq \begin{bmatrix} {}^W\mathbf{R}_I & {}^W\mathbf{p}_I & {}^W\mathbf{v}_I & \boldsymbol{\omega} & \mathbf{a} & \mathbf{b}_{\boldsymbol{\omega}} & \mathbf{b}_{\mathbf{a}} & {}^W\mathbf{g} & {}^I\mathbf{R}_L & {}^I\mathbf{p}_L \end{bmatrix} \quad (1)$$

where ${}^W\mathbf{R}_I$, ${}^W\mathbf{p}_I$, and ${}^W\mathbf{v}_I$ denote the IMU rotation, position, and velocity in the world frame W . \mathbf{b}_ω and \mathbf{b}_a are the IMU biases of angular velocity ω and linear acceleration a . ${}^W\mathbf{g}$ is the gravity vector in the world frame W . ${}^I\mathbf{R}_L$ and ${}^I\mathbf{p}_L$ are the extrinsic parameters between IMU frame and LiDAR frame.

Inertial Measurement Unit (IMU) provides motion information over short periods. When combined with inputs

from other sensors and advanced algorithms, it achieves impressive accuracy within a tightly-coupled multi-sensor fusion framework. The IMU noise \mathbf{w} is defined as:

$$\mathbf{w} \triangleq [\mathbf{n}_\omega \quad \mathbf{n}_a \quad \mathbf{n}_{b\omega} \quad \mathbf{n}_{ba}] \quad (2)$$

where ω and a are the IMU angular velocity and linear acceleration. \mathbf{n}_ω and \mathbf{n}_a denote the measurement noise of ω and a . $\mathbf{n}_{b\omega}$ and \mathbf{n}_{ba} are random walk process noises.

In our system, the state transition model at the sampling period Δt is defined as

$$\mathbf{x}_{i+1} = \mathbf{x}_i \boxplus (\Delta t \mathbf{f}(\mathbf{x}_i, \mathbf{w}_i)) \quad (3)$$

where the symbol \boxplus , as introduced in [12], represents the state transition of the system in the Lie algebra space. The function \mathbf{f} in forward process can be derived as

$$\mathbf{f}(\mathbf{x}, \mathbf{w}) \triangleq \begin{bmatrix} \omega - \mathbf{b}_\omega \\ {}^W \mathbf{R}_I (\mathbf{a} - \mathbf{b}_a) + {}^W \mathbf{g} \\ {}^W \mathbf{R}_I (\mathbf{a} - \mathbf{b}_a) + {}^W \mathbf{g} \\ \mathbf{n}_\omega \\ \mathbf{n}_a \\ \mathbf{n}_{b\omega} \\ \mathbf{n}_{ba} \\ \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (3)$$

B. Constant Velocity Model Prediction

The CV model is simple, robust, and reliable in environments with IMU data interruptions or sensor failures. Unlike the constant acceleration model (which risks instability), or machine-learning models (which depend heavily on data quality and training), the CV model avoids complexity and ensures stability. In the case of sensor failures, such as data interruptions from the IMU, we introduce a 100 Hz timer and switch to a CV model for prediction. This method ensures that even in the absence of IMU data, the localization system does not get stuck waiting for data synchronization or immediately diverge. The system assumes that the linear and angular velocity states remain constant between two consecutive LiDAR frames, allowing the entire positioning pipeline to degrade into a pure LiDAR-based localization method using the CV model. The prediction model, as applied during the filter prediction phase, can be expressed as:

$$\mathbf{x}_{i+1} = \mathbf{x}_i \boxplus (\Delta t \mathbf{f}(\mathbf{x}_i, \mathbf{0})) \quad (4)$$

where the covariance propagation follows a process similar to that of the iESKF algorithm.

C. LiDAR Dual-Map Updates

Compared to directly performing point cloud registration with a prior map, we transform the feature alignment process into the filter measurement update process within the iESKF framework based on point-to-plane constraints in the map \mathcal{M} :

$${}^\mathcal{M} \mathbf{R}_j(\mathbf{x}_i, {}^{L_i} \mathbf{p}_j, {}^{L_i} \mathbf{n}_j) = {}^\mathcal{M} \mathbf{u}_j^T ({}^\mathcal{M} \mathbf{T}_I {}^{I_i} \mathbf{T}_{L_i} ({}^{L_i} \mathbf{p}_j + {}^{L_i} \mathbf{n}_j) - {}^\mathcal{M} \mathbf{q}_j) \quad (6)$$

where ${}^{L_i} \mathbf{p}_j$ and ${}^{L_i} \mathbf{n}_j$ are the LiDAR point j in current i -th frame of point cloud and its noise, and we assume that this noise is affected by zero-mean Gaussian white noise. ${}^\mathcal{M} \mathbf{u}_j$ is the normal vector of the associated plane fitted using neighboring points of ${}^{L_i} \mathbf{p}_j$ in the map \mathcal{M} . ${}^\mathcal{M} \mathbf{T}_{I_i}$ is the transformation between the map frame \mathcal{M} and the IMU frame I_i . ${}^{I_i} \mathbf{T}_{L_i}$ is the transformation between the IMU frame I_i and the LiDAR frame L_i . ${}^\mathcal{M} \mathbf{q}_j$ is another point on the associated fitted plane in the map \mathcal{M} .

Then, a dual-map based localization scheme tailored for complex, expansive and dynamic environments is introduced. Unlike methods that solely rely on prior maps for pose estimation and correction, this approach also employs a tightly coupled LiDAR-inertial odometry integrated with a local online dynamic map. Meanwhile, instead of running a separate odometry system, we achieve tightly coupled feature-to-map matching by associating point cloud features with both the static prior map \mathcal{M}_{global} and the real-time local dynamic map \mathcal{M}_{local} within a single frame processing.

$$\mathbf{0} = {}^{\mathcal{M}_{global}} \mathbf{R}_j(\mathbf{x}_i, {}^{L_i} \mathbf{p}_j, {}^{L_i} \mathbf{n}_j) + {}^{\mathcal{M}_{local}} \mathbf{R}_j(\mathbf{x}_i, {}^{L_i} \mathbf{p}_j, {}^{L_i} \mathbf{n}_j) \quad (7)$$

This integration enables more precise, reliable, and robust localization.

Besides, to further accelerate computation, the new mathematically equivalent formula for calculating the Kalman gain is used, as proposed by FAST-LIO [12]. This reduces the computational complexity from the observation dimension to the state dimension, ensuring that the measurement update process remains computationally efficient and effective despite the large size of LiDAR point cloud observation dimension.

D. IMU Linear Acceleration and Angular Velocity Updates

In addition to performing the observation update using LiDAR data, the idea from Point-LIO [34] is incorporated, which treats IMU data as a measurement for filter correction. Specifically, assuming that the IMU measurement is affected by zero-mean Gaussian white noise, and considering that IMU's linear acceleration, angular velocity and their biases are included in the system state estimation, a measurement update can be performed upon receiving IMU data:

$$\omega = {}^I \omega - \mathbf{n}_\omega - \mathbf{b}_\omega - \mathbf{n}_{b\omega} \quad (8)$$

$$\mathbf{a} = {}^I \mathbf{a} - \mathbf{n}_a - \mathbf{b}_a - \mathbf{n}_{ba} \quad (9)$$

where ${}^I \omega$ and ${}^I \mathbf{a}$ are the measurements of IMU angular velocity and linear acceleration.

Then, for the i -th frame of IMU measurement data, we can denote the IMU constraints as:

$$\mathbf{R}(\mathbf{x}_i, {}^I \omega_i, \mathbf{n}_\omega) = {}^I \omega_i - \mathbf{n}_\omega - \mathbf{b}_\omega - \mathbf{n}_{b\omega} - \omega_i \quad (10)$$

$$\mathbf{R}(\mathbf{x}_i, {}^I \mathbf{a}_i, \mathbf{n}_a) = {}^I \mathbf{a}_i - \mathbf{n}_a - \mathbf{b}_a - \mathbf{n}_{ba} - \mathbf{a}_i \quad (11)$$

These IMU constraints can be integrated in the following form for the observation update:

$$\mathbf{0} = \mathbf{R}(\mathbf{x}_i, \omega_i^I, \mathbf{n}_\omega) + \mathbf{R}(\mathbf{x}_i, \mathbf{a}_i^I, \mathbf{n}_a) \quad (12)$$

E. Implement Details

LiDAR Motion Compensation. When the LiDAR is in motion, the points are sampled at different poses, which can introduce distortions in the data. To mitigate this, we adopt a motion compensation strategy similar to that used in Fast-LIO [12], leveraging filter state information. This approach involves maintaining a history of filter states over a specified period. For each point in the LiDAR frame, we interpolate this historical data to estimate the exact state at the moment the point was captured. By obtaining this state estimate, we can accurately project each point to a common reference frame, specifically the end of the point cloud scan. This process effectively compensates for distortions caused by the sensor's motion, ensuring that the resulting point cloud is coherent and provides an accurate representation of the environment.

Prior Map Construction and Local Map Updates. Constructing a LiDAR point cloud map is essential for high-precision localization systems. Initially, point cloud data often contains noise and redundancy. To address this, we use a simple yet effective interval sampling technique for denoising and downsampling, which improves both data quality and processing efficiency.

For the global prior map, we combine the Fast-LIO2 algorithm [33] with high-precision inertial navigation system (INS) data to accurately align point clouds from various poses, ensuring the creation of a consistent global map. In large-scale, dynamic environments, the presence of numerous moving objects and structures can cause ghosting and unreliable information. To deal with this issue, we apply post-processing to the global map, removing dynamic elements and only retaining static point cloud data to prevent localization drift. In the local dynamic map, we integrate the current frame's point cloud data using corrected LiDAR poses. Continuous updates from the iESKF filter help mitigate cumulative localization errors, maintaining the accuracy of the map. Outdated and distant data is also removed to optimize resource usage and ensure efficient output during local point cloud map registration in LiDAR-inertial odometry.

To enhance matching efficiency with both the local dynamic map and the global prior map, we employ the incremental k-d tree (ikd-Tree) data structure [33] for incremental map management. The ikd-Tree efficiently handles point insertion, deletion, dynamic rebalancing, and nearest neighbor searches, significantly improving the speed and accuracy of point cloud registration.

IV. EXPERIMENTS

To validate the effectiveness of the proposed method, we conduct comprehensive experiments on our private datasets. The experimental setup is detailed in Sec. IV-A, while the localization results are presented in Sec. IV-B.

A. Experiment Setup

Baselines and Metrics. Our baselines include LightLOAM [25], DLL [5], FAST-LIO2 [33], and Point-LIO [34], which represent state-of-the-art approaches in LiDAR

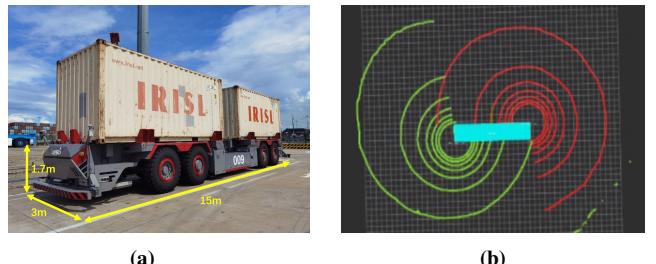


Fig. 2: Experimental Setup. (a) The IGV, with a size of $15 \text{ m} \times 3 \text{ m} \times 1.7 \text{ m}$. (b) The fused point clouds from two diagonally positioned 16-line LiDARs, where the blue rectangle represents the vehicle body. The green point cloud corresponds to the LiDAR located in the upper-left corner, while the red point cloud originates from the LiDAR in the lower-right corner.

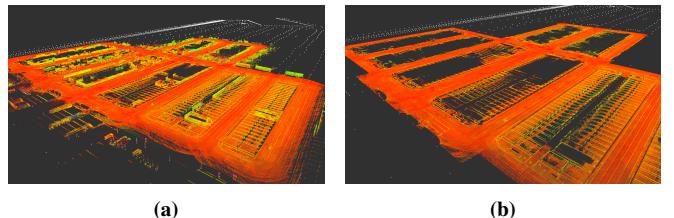


Fig. 3: Point cloud map. (a) Example of the point cloud map without dynamic obstacle removal. (b) Example of the point cloud map with dynamic obstacle removal. (c) The resulting point cloud output with a dimension of $1538 \text{ m} \times 596 \text{ m}$.

SLAM, localization, and odometry. We perform comparative experiments on these methods to assess the effectiveness of our proposed approach in various scenarios. To ensure a fair comparison, the results of these systems are obtained using the source code provided by their respective authors, with only minor adjustments made to the input data interfaces to adapt to our dataset. To assess the accuracy of SLAM, we use the Absolute Trajectory Error (ATE) of the pose trajectory as the evaluation metric and used the *evo* toolkit to compare and analyze the localization trajectories of different algorithms.

Datasets and Sensor Equipments. To evaluate the performance of our method in large-scale, unbounded, and dynamic environments, we conduct a detailed comparison with other state-of-the-art algorithms. We select a port environment as the testing scenario and manually created a private comprehensive dataset. The test area spans approximately 1 million square meters, with the dataset covering a total testing path of 6 km. Data was collected by driving an IGV throughout the port to capture diverse scenarios and conditions. The dataset includes high-quality 16-line LiDAR and 6-axis IMU

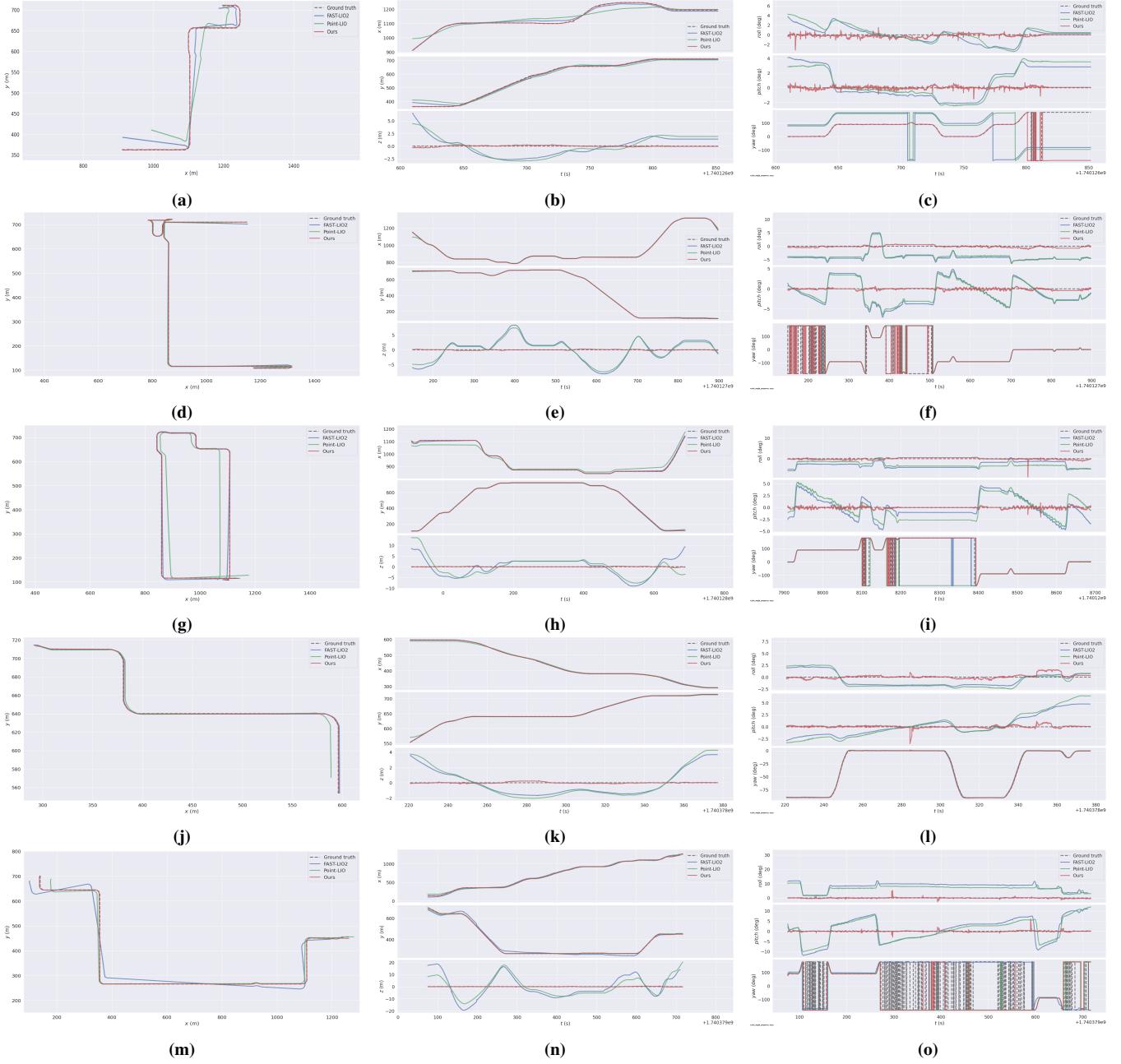


Fig. 4: Comparison results of the overall trajectories, incorporating both translational and rotational components, across various testing datasets. Each column represents the comparison results of different algorithms on each test dataset. (a), (d), (g), (j), and (m) represent the overall trajectories of each algorithm, (b), (e), (h), (k), and (n) represent the translation components of each algorithm, and (c), (f), (i), (l), and (o) represent the rotation components of each algorithm. Due to the excessive errors in DLL [5] and Light-LOAM [25], both methods failed in these scenarios, so we do not present their results.

data, with the LiDAR operating at a frequency of 10 Hz and the IMU at 100 Hz. Additionally, a higher-precision sensor fusion method with RTK, operating at 50 Hz, is provided as ground truth for quantitatively evaluating the performance of different localization systems. We design experiment trajectories in open areas with few tall mechanical structures to ensure GPS accuracy. To reduce potential LiDAR occlusions, the LiDAR data is generated from the fused point clouds of two 16-line LiDARs, which are time-synchronized and distortion-corrected. These LiDARs are strategically mounted at the upper-left and lower-right corners of the

test vehicle to maximize coverage. The test vehicle and the resulting LiDAR point cloud data are illustrated in Fig. 2.

Implementation Platform and Hardware. Our dataset evaluation and comparison experiments are implemented in C++ using the Robot Operating System (ROS). The experiments are conducted on a computer equipped with an Intel i7-8750H CPU running at 2.20 GHz, 32 GB of RAM, a GeForce GTX 1050 Ti GPU, and Ubuntu 20.04 as the operating system.

Global Prior Map. The port environment is highly dynamic, characterized by large IGVs, moving containers, mas-

TABLE I: Quantitative comparison for Light-LOAM [25], DLL [5], FAST-LIO2 [33], and Point-LIO [34], and our method on three datasets (corresponding to subfigures (d), (g), and (j) in Fig. 4)

Dataset	Case	Max absolute pose error(m)	Mean absolute pose error(m)	Max lateral error(m)	Mean lateral error(m)	Max longitudinal error(m)	Mean longitudinal error(m)
Fig. 4 (d)	DLL	451.693	207.601	231.120	63.017	444.835	100.318
	FAST-LIO2	8.229	2.753	8.053	1.879	2.845	0.274
	Light-LOAM	182.996	149.814	96.159	68.088	176.481	130.285
	Point-LIO	55.524	3.590	5.643	1.563	55.236	1.131
	Ours	0.742	0.165	0.236	0.031	0.704	0.030
Fig. 4 (g)	DLL	444.765	174.058	430.615	122.649	403.745	43.740
	FAST-LIO2	10.311	3.922	9.819	1.490	10.263	1.657
	Light-LOAM	20.374	17.793	20.029	11.800	20.366	8.228
	Point-LIO	39.408	20.741	36.930	14.726	38.253	4.734
	Ours	0.927	0.200	0.898	0.066	0.533	0.025
Fig. 4 (j)	DLL	140.157	68.741	133.230	22.116	139.585	15.966
	FAST-LIO2	2.161	0.921	1.290	0.404	2.158	0.169
	Light-LOAM	21.753	10.929	0.063	0.013	21.753	4.573
	Point-LIO	17.937	3.934	7.833	0.433	16.435	1.680
	Ours	0.553	0.262	0.511	0.116	0.472	0.075

sive gantry cranes, and rail cranes in constant motion. This ever-changing nature causes most environmental features to vary over time. Moreover, due to the port’s unbounded and expansive layout, LiDAR data predominantly captures ground-level information. To create a stable and static global prior map, we conduct extensive validation and fine-tuning in real-world scenarios. During the map construction process, we prioritize generating a point cloud map that primarily focuses on ground features while systematically removing all potentially dynamic objects. This approach ensures reliable and consistent pose constraints for the localization algorithm. The final map covers an area of approximately one million square meters as shown in Fig. 3.

B. Localization Results and Comparison

Localization Results. Through testing and validation, it is observed that Light-LOAM [25] and DLL [5] fail to operate effectively across all port test datasets, showing significant errors. Consequently, the following analysis focuses on the algorithms that functioned properly: our algorithm, FAST-LIO2 [33], and Point-LIO [34]. The complete test results are shown in Fig. 4.

Overall Trajectory Comparison. The trajectory generated by our algorithm closely aligns with the ground truth (as shown in subfigures (a), (d), (g), (j), and (m) in Fig. 4), with near-complete overlap, demonstrating exceptional accuracy in global localization. In contrast, on more complex paths (e.g., subfigures (a), (c), and (e) in Fig. 4), the trajectories of FAST-LIO2 [33] and Point-LIO [34] exhibit varying degrees of deviation. Notably, Point-LIO [34] shows more significant errors, especially in sharp turns or sections with complex motion, revealing its reduced robustness in dynamic scenarios. Our algorithm, however, maintains superior accuracy even under these challenging conditions, with minimal trajectory deviation, showcasing its strong adaptability and reliability.

Translational Comparison. The analysis of the translational component, examining accuracy in the x , y , and z directions, is presented in subfigures (b), (e), (h), (k), and (n) in Fig. 4. In the x and y directions, all algorithms perform well, with their estimated curves closely matching the ground truth, reflecting high translational accuracy. However, in the z direction (height), FAST-LIO2 [33] and Point-LIO [34] show noticeable deviations, as observed in subfigures (f), (h), and (j) in Fig. 4. In contrast, our algorithm exhibits

significantly better performance with smaller deviations. This discrepancy may stem from substantial sensor noise in height data or inadequate optimization of height estimation in the competing algorithms.

Rotation Comparison. The rotational component analysis, focusing on angle estimation around the x , y , and z axes, places particular emphasis on the yaw angle, as shown in subfigures (c), (f), (i), (l), and (o) in Fig. 4. Our algorithm demonstrates superior rotational accuracy across all datasets, with its estimated curves closely following the ground truth. While slight jitter is observed, it remains minimal and does not impact overall performance. In contrast, Point-LIO [34] and FAST-LIO2 [33] exhibit significant yaw estimation jitter in certain datasets (e.g., subfigures (k) and (o) in Fig. 4), especially near the end of the tests (the far-right sections of the figures), where error magnitudes increase considerably. This behavior is likely due to the inability of these algorithms to effectively correct accumulated angular errors.

Quantitative Comparison. We evaluated the performance of DLL [5], FAST-LIO2 [33], Light-LOAM [25], Point-LIO [34], and our approach on three datasets (Fig. 4 (d), (g), and (j)) using metrics such as maximum and mean absolute pose errors, lateral errors, and longitudinal errors. The results in TABLE I, consistently show that our method outperforms all other algorithms across all datasets and metrics. In Fig. 4 (d), our method achieved a maximum absolute pose error of 0.742 m, significantly lower than DLL [5] (451.693 m). Its mean absolute pose error was 0.165 m, far below FAST-LIO2 [33] (2.753 m) and Point-LIO [34] (3.590 m). Similarly, in Fig. 4 (g), our system maintained the smallest maximum (0.927 m) and mean (0.200 m) pose errors, with minimal lateral and longitudinal errors. In Fig. 4 (j), our algorithm continued to excel, achieving a maximum pose error of 0.553 m and a mean pose error of 0.262 m, both significantly better than the competing algorithms. Besides, DLL [5] and Light-LOAM [25] fail with large errors.

Overall, our framework consistently demonstrated the highest accuracy and robustness, with significantly lower errors across all datasets. These results highlight the effectiveness of its improved optimization strategies and robust error-handling mechanisms in complex environments.

V. CONCLUSION

In this paper, we present a robust and precise mapping and localization system designed for changing, dynamic outdoor environments. By tightly coupling a LiDAR-inertial odometry with map-based localization, the system addresses instability and drift caused by environmental changes, integrating offline global map registration with real-time dynamic mapping for reliable performance in large-scale scenarios. At its core, the system employs a multi-sensor fusion framework based on the iESKF, ensuring both robustness and accuracy. Extensive real-world experiments demonstrate that our method consistently outperforms state-of-the-art LiDAR localization algorithms, making it a reliable solution for applications like robotics and autonomous driving.

REFERENCES

- [1] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, “Towards Fully Autonomous Driving: Systems and Algorithms,” in *2011 IEEE Intelligent Vehicles Symposium (IV)*, 2011, pp. 163–168.
- [2] A. Dewan, A. Kumar, H. Singh, V. S. Solanki, and P. Kaur, “Advancement in SLAM Techniques and Their Diverse Applications,” in *2023 12th International Conference on System Modeling & Advancement in Research Trends (SMART)*, 2023, pp. 387–392.
- [3] D. Kumar and N. Muhammad, “A Survey on Localization for Autonomous Vehicles,” *IEEE Access*, vol. 11, pp. 115 865–115 883, 2023.
- [4] Z. Du, S. Ji, and K. Khoshelham, “3-D LiDAR-Based Place Recognition Techniques: A Review of the Past Ten Years,” *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–24, 2024.
- [5] F. Caballero and L. Merino, “DLL: Direct LiDAR Localization. A Map-based Localization Approach for Aerial Robots,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 5491–5498.
- [6] B. Peng, H. Xie, and W. Chen, “ROLL: Long-Term Robust LiDAR-based Localization With Temporary Mapping in Changing Environments,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 2841–2847.
- [7] K. Koide, M. Yokozuka, S. Oishi, and A. Banno, “GLIM: 3D Range-inertial Localization and Mapping with GPU-accelerated Scan Matching Factors,” *Robotics and Autonomous Systems*, vol. 179, p. 104750, 2024.
- [8] C. Campos, R. Elvira, J. J. Gómez, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM,” *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.
- [9] Y. Yu, P. Yun, B. Xue, J. Jiao, R. Fan, and M. Liu, “Accurate and Robust Visual Localization System in Large-Scale Appearance-Changing Environments,” *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 5222–5232, 2022.
- [10] H. Yin, X. Xu, S. Lu, X. Chen, R. Xiong, S. Shen, C. Stachniss, and Y. Wang, “A Survey on Global LiDAR Localization: Challenges, Advances and Open Problems,” *International Journal of Computer Vision*, vol. 132, no. 8, pp. 3139–3171, 2024.
- [11] B. Xue, X. Yan, J. Wu, J. Cheng, J. Jiao, H. Jiang, R. Fan, M. Liu, and C. Zhang, “Visual-Marker-Based Localization for Flat-Variation Scene,” *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–16, 2024.
- [12] W. Xu and F. Zhang, “FAST-LIO: A Fast, Robust LiDAR-Inertial Odometry Package by Tightly-Coupled Iterated Kalman Filter,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3317–3324, 2021.
- [13] N. Rufus, U. K. R. Nair, A. V. S. S. B. Kumar, V. Madiraju, and K. M. Krishna, “SROM: Simple Real-time Odometry and Mapping using LiDAR Data for Autonomous Vehicles,” in *2020 IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 1867–1872.
- [14] H. Wang, C. Wang, C.-L. Chen, and L. Xie, “F-LOAM : Fast LiDAR Odometry and Mapping,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 4390–4396.
- [15] U. N. C. on Trade and Development, *Review of Maritime Transport 2023*, 2023rd ed. United Nations, 2023. [Online]. Available: <https://www.un-ilibrary.org/content/books/9789213584569>
- [16] T. Shan and B. Englot, “LeGO-LOAM: Lightweight and Ground-Optimized LiDAR Odometry and Mapping on Variable Terrain,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4758–4765.
- [17] P. Besl and N. D. McKay, “A Method for Registration of 3-D Shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.
- [18] S. Rusinkiewicz and M. Levoy, “Efficient Variants of the ICP Algorithm,” in *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, 2001, pp. 145–152.
- [19] Y. Duan, J. Peng, Y. Zhang, J. Ji, and Y. Zhang, “PFilter: Building Persistent Maps through Feature Filtering for Fast and Accurate LiDAR-based SLAM,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 11 087–11 093.
- [20] Z. Wang, L. Yang, F. Gao, and L. Wang, “FEVO-LOAM: Feature Extraction and Vertical Optimized LiDAR Odometry and Mapping,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12 086–12 093, 2022.
- [21] J. Li, X. Zhang, Y. Zhang, Y. Chang, and K. Zhao, “RF-LOAM: Robust and Fast LiDAR Odometry and Mapping in Urban Dynamic Environment,” *IEEE Sensors Journal*, vol. 23, no. 23, pp. 29 186–29 199, 2023.
- [22] D. Chang, S. Huang, R. Zhang, M. Hu, R. Ding, and X. Qin, “WiCRF2: Multi-Weighted LiDAR Odometry and Mapping With Motion Observability Features,” *IEEE Sensors Journal*, vol. 23, no. 17, pp. 20 236–20 246, 2023.
- [23] N. Li, Y. Yao, X. Xu, Y. Peng, Z. Wang, and H. Wei, “An Efficient LiDAR SLAM With Angle-Based Feature Extraction and Voxel-Based Fixed-Lag Smoothing,” *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–13, 2024.
- [24] Y. Su, S. Zhang, C. Zhang, and Y. Wu, “A 3D LiDAR Odometry and Mapping Method Based on Clustering-Directed Feature Point,” *IEEE Transactions on Vehicular Technology*, vol. 73, no. 12, pp. 18 391–18 401, 2024.
- [25] S. Yi, Y. Lyu, L. Hua, Q. Pan, and C. Zhao, “Light-LOAM: A Lightweight LiDAR Odometry and Mapping Based on Graph-Matching,” *IEEE Robotics and Automation Letters*, vol. 9, no. 4, pp. 3219–3226, 2024.
- [26] D. Rozenberszki and A. L. Majdik, “LOL: LiDAR-only Odometry and Localization in 3D Point Cloud Maps,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 4379–4385.
- [27] Y. Wu, C. Zhao, and Y. Lyu, “DMLL: Differential-Map-Aided LiDAR-Based Localization,” *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–14, 2023.
- [28] A. Thakur and P. Rajalakshmi, “LiDAR-Based Optimized Normal Distribution Transform Localization on 3-D Map for Autonomous Navigation,” *IEEE Open Journal of Instrumentation and Measurement*, vol. 3, pp. 1–11, 2024.
- [29] Y. Feng, Z. Jiang, Y. Shi, Y. Feng, X. Chen, H. Zhao, and G. Zhou, “Block-Map-Based Localization in Large-Scale Environment,” 2024. [Online]. Available: <https://arxiv.org/abs/2404.18192>
- [30] C. Qin, H. Ye, C. E. Pranata, J. Han, S. Zhang, and M. Liu, “LINS: A Lidar-Inertial State Estimator for Robust and Efficient Navigation,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 8899–8906.
- [31] Y. Wu, T. Guadagnino, L. Wiesmann, L. Klingbeil, C. Stachniss, and H. Kuhlmann, “LIO-EKF: High Frequency LiDAR-Inertial Odometry using Extended Kalman Filters,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 13 741–13 747.
- [32] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, “LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 5135–5142.
- [33] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, “FAST-LIO2: Fast Direct LiDAR-Inertial Odometry,” *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.
- [34] D. He, W. Xu, N. Chen, F. Kong, C. Yuan, and F. Zhang, “Point-LIO: Robust High-Bandwidth Light Detection and Ranging Inertial Odometry,” *Advanced Intelligent Systems*, vol. 5, no. 7, p. 2200459, 2023.