

Multi-Scale Registration of LiDAR Odometry Based on Vertical Constraints in Dynamic Environments

Qifeng Wang, *Student Member, IEEE*, Weigang Li, *Member, IEEE*, Lei Nie, *Member, IEEE*, Xin Xu, *Senior Member, IEEE*, Wenping Liu, *Senior Member, IEEE*, Zhe Xu, *Member, IEEE*

Abstract—As a key technology for autonomous navigation and positioning of mobile robots, the Light Detection and Ranging (LiDAR) odometry is widely adopted in autonomous driving. However, most existing methods are single-scale registration: when the initial pose is unreliable or there are a large number of dynamic targets and occlusions in the scene, optimization is prone to converge to a local optimum, resulting in a decrease in registration accuracy. Moreover, the rotational 3D LiDAR has a limited number of channels in the pitch direction and a large angular interval, which leads to a low vertical resolution for long-distance and insufficient height observability, also causing a decrease in registration accuracy. To address these issues, this paper proposes a multi-scale registration LiDAR odometry based on vertical constraints in dynamic environments: using motion prediction as the initial pose, first completing frame-to-frame registration, then using the result as the initial value to perform frame-to-local-map registration to obtain the final pose, thereby enhancing the reliability of the initial value and reducing the risk of local optimum. To suppress height drift, vertical constraints are introduced in the Iterative Closest Point (ICP) iteration interior: setting thresholds and limits for the height translation increment within the iteration, and updating preferentially based on the predicted benefit; in the iteration exterior, frame-level cumulative height displacement is again limited. Experiments on various scenarios from the public KITTI dataset show that the proposed method achieves better trajectory accuracy than mainstream methods, verifying the effectiveness and robustness of the method in dynamic environments.

Index Terms—LiDAR odometry; multi-scale registration; vertical constraints; autonomous driving

I. INTRODUCTION

Light Detection and Ranging (LiDAR) provides accurate range sensing and 3D geometry under low light, strong illumination changes, and texture-sparse conditions, and has become a core sensor for autonomous vehicles and mobile robots [1], [2]. By directly exploiting depth, LiDAR odometry (LO) estimates the six-degree-of-freedom motion in real time and underpins localization and planning [3], [4].

This work was partially supported by the National Natural Science Foundation of China under Grant No. 51774219, the Hubei Provincial Science and Technology Innovation Talent Program under Grant No. 2024DJC077, and the Wuhan Key Research and Development Program under Grant No. 2025050102030008. Numerical calculations were supported by the High-Performance Computing Center of Wuhan University of Science and Technology. (*Corresponding author: Weigang Li.*)

Qifeng Wang, Weigang Li and Zhe Xu are with the School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan, China.

Lei Nie and Xin Xu are with the School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, China.

Wenping Liu is with the School of Information Management and Institute of Big Data and Digital Economy, Hubei University of Economics, Wuhan, China.

Among LO approaches, Iterative Closest Point (ICP)-based pipelines are attractive for their simplicity, controllable efficiency, and engineering maturity [5], [6]. ICP estimates pose by iteratively building correspondences and minimizing geometric residuals. However, two practical challenges remain prominent [7]. First, single-scale registration makes the solution sensitive to initialization: when the initial pose is unreliable, the optimizer can converge to local optima, causing alignment failures or accuracy degradation. Second, sensor geometry induces weak vertical observability: spinning 3D LiDARs have limited pitch channels with large angular spacing, so far-range vertical resolution is low; together with extended near-horizontal planes (e.g., roads) and small calibration/time-sync errors, this leads to amplified errors along the z -axis and noticeable z -drift [8], [9].

To address these challenges, this paper proposes a multi-scale registration LiDAR odometry based on vertical constraints in dynamic environments. By providing reliable initial poses through motion prediction, it first completes registration between adjacent frames, then uses the result as the initial value to perform frame-to-local-map registration to obtain the final pose. This enhances the reliability of the initial value and reduces the risk of local optimality. At the same time, addressing the root causes of weak z -axis observability and the tendency of misalignment leading to drift, this paper introduces vertical constraints in the ICP iteration both internally and externally: setting thresholds and limits on the translation increment in the height (z -axis) direction during iteration, and selecting more robust updates through predicted gain; again limiting the cumulative height displacement at the frame scale, directly suppressing z -axis drift, and improving vertical posture stability. To support efficient matching and online map maintenance, this paper uses voxel hashing to organize point clouds, establishing correspondences through parallel nearest neighbor search within a fixed neighborhood, and dynamically updating the map, balancing real-time performance and robustness in complex dynamic scenarios.

The main contributions of our work are as follows:

- To mitigate local-minimum failures, we use motion prediction for initialization, perform frame-to-frame (F2F) alignment, and then refine by frame-to-local-map (F2LM) registration to obtain the final pose. This two-stage schedule strengthens the initial guess and enlarges the basin of convergence.
- To tackle the weak observability of the vertical (z) axis with sensor geometry, we impose vertical constraints at two layers: within ICP iterations we gate and clamp the z -

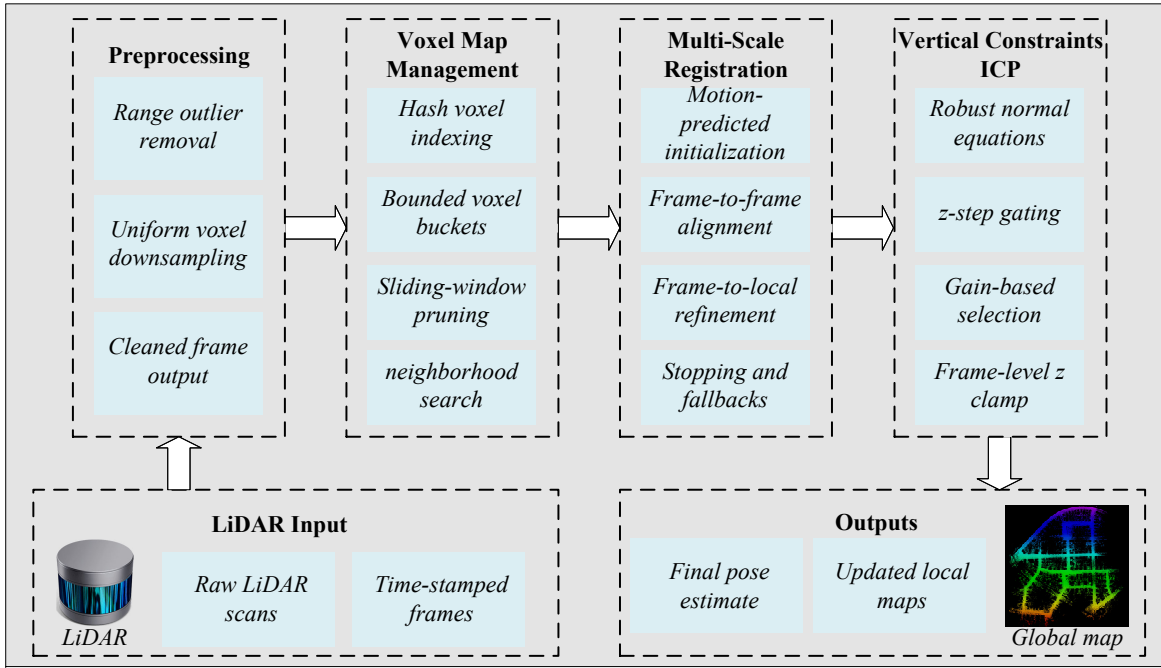


Fig. 1. Overview of the proposed pipeline with four components: preprocessing, voxel map management, vertical constraints ICP, and multi-scale registration.

translation increment and select the update by a predicted-gain criterion; at the frame scale we clip the net per-frame z displacement. Together, these constraints suppress z -drift while leaving planar motion largely unconstrained.

- To evaluate our design in real traffic, we benchmark against representative LO baselines. The results show consistently lower trajectory errors and improved stability, demonstrating the effectiveness of the proposed design in dynamic environments.

We release the code and evaluation scripts¹ that faithfully implement the method described in this paper.

II. RELATED WORK

LiDAR odometry, as the fundamental capability for autonomous navigation and positioning of mobile robots, has been widely adopted in scenarios such as autonomous driving, unmanned aerial vehicles, and service robots [10], [11], [12]. Generally, existing methods can be classified into two categories: feature-based odometry and ICP-based odometry [13], [14], [15].

A. Feature-Based LiDAR Odometry

Feature-based approaches estimate motion by detecting and associating salient structures (e.g., edges, planes) across scans [16], [17]. LOAM decouples the system into a high-rate odometry thread and a low-rate map-optimization thread, striking a balance between throughput and accuracy [18], [19]. LeGO-LOAM adds ground segmentation and clustering to improve feature quality and integrates loop closure to curb long-term drift [20]. F-LOAM revisits LOAM's engineering,

restructuring extraction and matching for much higher efficiency with comparable accuracy [21], [22]. Surfel-based mapping backends further stabilize long-horizon alignment in structured scenes [23], [24]. These methods work well when distinctive features are abundant, but performance can deteriorate in texture-sparse corridors, open fields, or under heavy occlusion.

In recent years, deep learning has been used to automatically learn more discriminative representations to assist odometry. LO-Net [25] learns features related to motion estimation through weighted geometric constraint losses; SDV-LOAM [26] embeds the learning module into the LOAM pipeline to enhance its robustness in complex environments. Although the learning methods have potential in perception representation, their training and deployment typically rely on large-scale data and high computing power, posing challenges to real-time systems' resource budget, and the interpretability of the model's decision process is relatively insufficient [27], [28].

B. ICP-Based LiDAR Odometry

Unlike feature-based pipelines, ICP-style methods operate directly on raw point sets, avoiding explicit feature extraction and remaining reliable when salient structures are sparse or unstable [29], [19]. Classical ICP estimates pose by iteratively building nearest-neighbor correspondences and minimizing point-point or point-plane residuals. On this foundation, many variants improve robustness and efficiency: GICP models local geometric uncertainty in the cost for better-conditioned updates [30]; Fast-gicp exploits voxel grids and parallelism to cut correspondence and linearization cost [6]; Faster-gicp further prunes redundant matches via acceptance-rejection sampling, increasing speed while preserving accuracy [7]. DLO shows that dense input with minimal preprocessing can deliver real-

¹<https://github.com/qfwang23/LO>

time, high-accuracy odometry in long deployments [5], and Kiss-icp distills a lightweight, robust ICP pipeline with simple multi-stage filtering and adaptive thresholds, achieving competitive accuracy with very low complexity [9].

Although ICP-based odometry offers a favorable accuracy–efficiency trade-off, two practical issues often undermine performance. First, when the initial pose is unreliable or the scene contains strong dynamics/occlusions, the objective landscape becomes multi-modal and the optimizer can settle in a local minimum, causing failures or degraded accuracy. Second, with spinning 3D LiDARs the elevation (pitch) channel count is small and beam spacing is wide, so long-range vertical resolution is weak; together with ground-dominant geometry (large near-horizontal planes), this yields poor observability along the z -axis. In such settings, small measurement noise, slight extrinsic or time-sync errors, and scan motion can be disproportionately expressed as vertical drift, harming trajectory stability [8], [31].

To counter these effects, we propose a LiDAR odometry system that couples multi-scale registration with explicit vertical constraints: a motion-predicted initialization followed by F2F alignment and then F2LM refinement to enlarge the basin of convergence, plus iteration- and frame-level z -clamping within ICP to suppress vertical drift while leaving planar motion largely unconstrained. This design improves convergence and robustness in complex, dynamic environments without sacrificing real-time performance.

III. OUR APPROACH

To improve alignment accuracy and robustness for LiDAR odometry in dynamic environments, we propose a multi-scale registration scheme with explicit vertical constraints. As illustrated in Fig. 1, the pipeline proceeds in four tightly coupled modules: (i) preprocessing that produces a cleaned, voxelized subset; (ii) hash-voxel mapping that maintains last-frame and local maps with bounded memory; (iii) a two-stage registration schedule guided by motion prediction; and (iv) a vertical constraints ICP that regularizes the poorly observed z -dimension without sacrificing planar motion. The detailed workflows of modules (iii) and (iv) are summarized in Algorithm 1.

A. Preprocessing

Let the raw point cloud be $\mathcal{P} = \{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^N$. We construct a compact subset $\tilde{\mathcal{P}}$ via two deterministic operators applied in sequence. (i) *Range masking*. Define $r(\mathbf{p}) = \|\mathbf{p}\|_2$ and retain points in a spherical band,

$$\mathcal{P}_r = \{\mathbf{p} \in \mathcal{P} \mid r_{\min} < r(\mathbf{p}) < r_{\max}\}, \quad (1)$$

where r_{\min}, r_{\max} depend on sensor mounting and scene scale. This removes near-field self-reflections/mount occlusions and far-field sparsity-dominated returns that contribute little to odometry. (ii) *Voxel downsampling*. Let $v > 0$ denote the voxel edge and

$$\mathbf{k}(\mathbf{p}) = \lfloor \mathbf{p}/v \rfloor \in \mathbb{Z}^3. \quad (2)$$

We select one representative per occupied voxel (first-hit or centroid) and denote the result as

$$\tilde{\mathcal{P}} = \mathcal{V}_v(\mathcal{P}_r). \quad (3)$$

This preserves scene-scale geometry, stabilizes nearest-neighbor queries, and renders the per-iteration ICP cost approximately linear in the number of accepted correspondences.

B. Voxel Map Management

We reuse the same voxel size v and indexing as in preprocessing to keep mapping and querying consistent. For any world-frame point $\mathbf{x} \in \mathbb{R}^3$,

$$\mathbf{k}(\mathbf{x}) = \lfloor \mathbf{x}/v \rfloor \in \mathbb{Z}^3. \quad (4)$$

a) *Hash voxel map and bounded buckets.*: A hash table stores, for each index \mathbf{k} , a bounded bucket $\mathcal{B}_{\mathbf{k}}$ with capacity P_{\max} . The bucket policy (first-in, reservoir-like, or incremental replacement) is implementation-specific; our choice keeps insertion $O(1)$ in expectation while ensuring the number of candidates per query remains bounded.

b) *Map layers and update flow.*: We maintain three complementary maps: a *last-frame* map $\mathcal{M}^{\text{last}}$ (built from the most recent frame and refreshed every step), a *local* map $\mathcal{M}^{\text{local}}$ (sliding window around the trajectory), and a *global* map (sparser insertion for visualization). Given the current pose $\mathbf{T} \in SE(3)$ and a downsampled point $\mathbf{p} \in \tilde{\mathcal{P}}$, compute $\mathbf{x} = \mathbf{T}\mathbf{p}$ and $\mathbf{k} = \lfloor \mathbf{x}/v \rfloor$; append \mathbf{x} into the corresponding bucket if $|\mathcal{B}_{\mathbf{k}}| < P_{\max}$. On each step, $\mathcal{M}^{\text{last}}$ is rebuilt from the current frame only (clearing old contents), whereas $\mathcal{M}^{\text{local}}$ accumulates points within a sliding radius.

$$\begin{aligned} \mathcal{B}_{\mathbf{k}} &\leftarrow \mathcal{B}_{\mathbf{k}} \cup \{\mathbf{x}\} \quad \text{if } |\mathcal{B}_{\mathbf{k}}| < P_{\max}, \\ \mathbf{k} &= \lfloor \mathbf{x}/v \rfloor, \quad \mathbf{x} = \mathbf{T}\mathbf{p}. \end{aligned} \quad (5)$$

c) *Sliding-window pruning.*: Let \mathbf{o} be the current origin (typically the latest pose) and R_{\max} the local radius. We periodically remove voxels whose stored points lie outside the sphere of radius R_{\max} centered at \mathbf{o} . This keeps memory bounded and ensures that registration uses a geometry-consistent, recently updated reference.

d) *27-neighborhood correspondences.*: During ICP, for a query point \mathbf{x} (already transformed by the current pose estimate), we examine the fixed $3 \times 3 \times 3$ neighborhood

$$\mathcal{N}_{27}(\mathbf{k}(\mathbf{x})) = \{\mathbf{k}(\mathbf{x}) + \boldsymbol{\delta} \mid \boldsymbol{\delta} \in \{-1, 0, 1\}^3\}. \quad (6)$$

The candidate set equals the union of buckets over \mathcal{N}_{27} ; we select the Euclidean nearest neighbor \mathbf{y}^* and accept $(\mathbf{x}, \mathbf{y}^*)$ only if the distance is below a gate τ_d , where τ_d is a fixed distance gate.

C. Multi-Scale Registration

We adopt a two-stage schedule guided by motion prediction. Let $\mathcal{S}_i \subset \tilde{\mathcal{P}}$ be the source subset for frame i . The last-frame and local voxel maps at step $i-1$ are denoted $\mathcal{M}_{i-1}^{\text{last}}$ and $\mathcal{M}_{i-1}^{\text{local}}$.

a) *Motion-predicted initialization.*: With the two most recent world poses $\mathbf{T}_{i-1}, \mathbf{T}_{i-2} \in SE(3)$,

$$\mathbf{T}_i^{\text{pred}} = \begin{cases} \mathbf{I}, & i < 2, \\ \mathbf{T}_{i-1}(\mathbf{T}_{i-2}^{-1}\mathbf{T}_{i-1}), & i \geq 2, \end{cases} \quad (7)$$

which extrapolates the most recent relative motion and provides a drift-aware initial guess.

b) *Stage 1: frame-to-frame.*: Using $\mathbf{T}_i^{\text{pred}}$ as the initial guess,

$$\mathbf{T}_i^{\text{ff}} = \text{Reg}(\mathcal{S}_i, \mathcal{M}_{i-1}^{\text{last}}; \mathbf{T}_i^{\text{pred}}, \sigma_{\text{ff}}), \quad (8)$$

where $\text{Reg}(\cdot)$ denotes our vertical constraints ICP (Sec. III-D) and σ_{ff} is a fixed robust scale for this stage. If $\mathcal{M}_{i-1}^{\text{last}}$ is empty (start-up), we set $\mathbf{T}_i^{\text{ff}} = \mathbf{T}_i^{\text{pred}}$.

c) *Stage 2: frame-to-local-map.*: Refine using the richer local map and the Stage 1 output as the initial guess,

$$\mathbf{T}_i = \text{Reg}(\mathcal{S}_i, \mathcal{M}_{i-1}^{\text{local}}; \mathbf{T}_i^{\text{ff}}, \sigma_{\text{lm}}), \quad (9)$$

with a fixed robust scale σ_{lm} .

This schedule first reduces relative-frame misalignment on an easy, geometry-consistent reference, then refines on a wider-coverage reference with higher redundancy, enlarging the basin of convergence and lowering the risk of local minima while preserving real-time performance.

d) *Stopping and fallbacks.*: Each stage stops on a small-step criterion or an iteration cap. If the accepted correspondence count falls below a threshold, we skip refinement and keep the best available estimate, which improves robustness in transient occlusions and heavy dynamics.

D. Vertical Constraints ICP

Let the correspondence set from the voxel map be $\mathcal{P}_{\text{corr}} = \{(\mathbf{p}_i, \mathbf{y}_i)\}$. We estimate $\mathbf{T} \in SE(3)$ that aligns source points to neighbors while explicitly constraining vertical motion.

a) *Linearization and normal equations.*: For $\mathbf{x}_i = \mathbf{T}\mathbf{p}_i$, define residual and Jacobian in compact form: $\mathbf{r}_i = \mathbf{x}_i - \mathbf{y}_i$, $\mathbf{J}_i = [\mathbf{I}_3 \quad -[\mathbf{x}_i]_{\times}]$. Using a twist increment $\delta\boldsymbol{\xi} = [\delta\mathbf{t}^T, \delta\boldsymbol{\theta}^T]^T \in \mathbb{R}^6$ and robust weights $w_i = w(\|\mathbf{r}_i\|_2; \delta_w)$, we accumulate $w_i \mathbf{J}_i^T \mathbf{J}_i$ and $w_i \mathbf{J}_i^T \mathbf{r}_i$ across correspondences to obtain \mathbf{H} and \mathbf{g} (omitting explicit summations for brevity), and solve with a small Levenberg–Marquardt damping:

$$\mathbf{r}_i = \mathbf{T}\mathbf{p}_i - \mathbf{y}_i. \quad (10)$$

$$(\mathbf{H} + \lambda \mathbf{I}) \delta\boldsymbol{\xi} = -\mathbf{g}. \quad (11)$$

b) *Iteration-level z gating with gain selection.*: Let δz be the z -component of $\delta\mathbf{t}$ and define a clamp operator $\mathcal{C}(x, a) = \min\{\max(x, -a), a\}$. If $|\delta z|$ is below a small gate Δz_{gate} , we apply the full update. Otherwise, we form three candidates: (i) full update; (ii) clamped update with $\delta z \leftarrow \mathcal{C}(\delta z, \Delta z_{\text{max}})$; (iii) zero- z update with $\delta z \leftarrow 0$.

$$\delta z^{\text{clp}} = \mathcal{C}(\delta z, \Delta z_{\text{max}}). \quad (12)$$

Using the local quadratic model

$$g(\delta\boldsymbol{\xi}) = -\mathbf{g}^T \delta\boldsymbol{\xi} - \frac{1}{2} \delta\boldsymbol{\xi}^T \mathbf{H} \delta\boldsymbol{\xi}, \quad (13)$$

we select the candidate with maximal $g(\cdot)$ and update the pose accordingly.

Algorithm 1: Two-Stage Registration with ICP under Vertical Constraints (VC-ICP)

Input : source cloud \mathcal{S}_i ; last-frame map $\mathcal{M}_{i-1}^{\text{last}}$; local map $\mathcal{M}_{i-1}^{\text{local}}$; prior poses $\mathbf{T}_{i-1}, \mathbf{T}_{i-2}$; robust scales $\sigma_{\text{ff}}, \sigma_{\text{lm}}$; distance gate τ_d ; min correspondences N_{min} ; iteration cap J_{max} ; step tol. ε ; gates $\Delta z_{\text{gate}}, \Delta z_{\text{max}}, \Delta z_{\text{frm}}^{\text{max}}$

Output: final pose \mathbf{T}_i

```

1 Stage A: motion-predicted init
2 if  $i < 2$  then  $\mathbf{T}_i^{\text{pred}} \leftarrow \mathbf{I}$ 
3 else  $\mathbf{T}_i^{\text{pred}} \leftarrow \mathbf{T}_{i-1}(\mathbf{T}_{i-2}^{-1}\mathbf{T}_{i-1})$ 
4 Stage B: frame-to-frame (F2F)
5 if  $\mathcal{M}_{i-1}^{\text{last}}$  empty then  $\mathbf{T}_i^{\text{ff}} \leftarrow \mathbf{T}_i^{\text{pred}}$ 
6 else  $\mathbf{T}_i^{\text{ff}} \leftarrow \text{VC\_ICP}(\mathcal{S}_i, \mathcal{M}_{i-1}^{\text{last}}, \mathbf{T}_i^{\text{pred}}, \sigma_{\text{ff}})$ 
7 Stage C: frame-to-local-map (F2LM)
8  $\mathbf{T}_i \leftarrow \text{VC\_ICP}(\mathcal{S}_i, \mathcal{M}_{i-1}^{\text{local}}, \mathbf{T}_i^{\text{ff}}, \sigma_{\text{lm}})$ 
9 Function  $\text{VC\_ICP}(\mathcal{S}, \mathcal{M}, \mathbf{T}_0, \sigma)$ 
10    $\mathbf{T} \leftarrow \mathbf{T}_0$ 
11   for  $j \leftarrow 1$  to  $J_{\text{max}}$  do
12     // Build correspondences via
13     // 27-neighborhood + distance gate
14      $\mathcal{P}_{\text{corr}} \leftarrow \emptyset$ 
15     for each  $\mathbf{p} \in \mathcal{S}$  do
16        $\mathbf{x} \leftarrow \mathbf{T}\mathbf{p}$ ;
17       find  $\mathbf{y}^*$  in the 27-neighborhood of  $\mathbf{k}(\mathbf{x})$  in  $\mathcal{M}$ 
18       if  $\|\mathbf{x} - \mathbf{y}^*\|_2 \leq \tau_d$  then
19         add  $(\mathbf{p}, \mathbf{y}^*)$  to  $\mathcal{P}_{\text{corr}}$ 
20       end
21     end
22     if  $|\mathcal{P}_{\text{corr}}| < N_{\text{min}}$  then
23       return  $\mathbf{T}_0$ 
24     end
25     // LM step (short key formula)
26     Solve  $(\mathbf{H} + \lambda \mathbf{I}) \delta\boldsymbol{\xi} = -\mathbf{g}$  with robust weights (scale  $\sigma$ )
27     Extract  $\delta z$  from the translation part  $\delta\mathbf{t}$  of  $\delta\boldsymbol{\xi}$ 
28     // Iteration-level  $z$  gating + gain selection
29     if  $|\delta z| \leq \Delta z_{\text{gate}}$  then
30        $\delta\boldsymbol{\xi}^* \leftarrow \delta\boldsymbol{\xi}$ 
31     end
32     else
33       make  $\delta\boldsymbol{\xi}^{\text{clp}}$  with  $\delta z \leftarrow \mathcal{C}(\delta z, \Delta z_{\text{max}})$ 
34       make  $\delta\boldsymbol{\xi}^{\text{zero}}$  with  $\delta z \leftarrow 0$ 
35       pick  $\delta\boldsymbol{\xi}^* \in \{\delta\boldsymbol{\xi}, \delta\boldsymbol{\xi}^{\text{clp}}, \delta\boldsymbol{\xi}^{\text{zero}}\}$  maximizing
36        $\mathcal{G}(\delta\boldsymbol{\xi}) = -\mathbf{g}^T \delta\boldsymbol{\xi} - \frac{1}{2} \delta\boldsymbol{\xi}^T \mathbf{H} \delta\boldsymbol{\xi}$ 
37     end
38      $\mathbf{T} \leftarrow \exp(\delta\boldsymbol{\xi}^*) \mathbf{T}$ 
39     if  $\|\delta\boldsymbol{\xi}^*\| < \varepsilon$  then
40       break
41   end
42   // Frame-level  $z$  clamp
43   Let  $\mathbf{t}_0, \mathbf{t}_{\text{out}}$  be translations of  $\mathbf{T}_0, \mathbf{T}$ ;
44    $t_{\text{out},z} \leftarrow t_{0,z} + \mathcal{C}((t_{\text{out}} - t_0)_z, \Delta z_{\text{frm}}^{\text{max}})$ 
45   return  $\mathbf{T}$ 

```

c) *Frame-level z clamp.*: We cap the per-frame vertical change by overwriting the output z with the initial z plus a clipped difference:

$$t_{\text{out},z} \leftarrow t_{0,z} + \mathcal{C}((t_{\text{out}} - t_0)_z, \Delta z_{\text{frm}}^{\text{max}}). \quad (14)$$

Here, $\mathcal{C}(u, a)$ denotes a symmetric clipping operator that

TABLE I
APE COMPARISON OF KITTI. METRICS: RMSE / MEAN / STD (LOWER IS BETTER).

Sequences	Fast-gicp [6]			Faster-gicp [7]			DLO [5]			Kiss-icp [9]			Ours		
	RMSE	Mean	Std	RMSE	Mean	Std	RMSE	Mean	Std	RMSE	Mean	Std	RMSE	Mean	Std
00	15.15	5.82	3.81	10.14	8.24	5.91	3.28	3.08	1.13	4.04	3.51	2.01	4.30	3.76	2.08
01	23.67	22.20	10.53	24.21	21.87	10.39	120.10	113.80	38.40	20.18	18.16	8.80	18.83	16.93	8.25
02	17.70	16.43	6.58	15.24	13.79	6.51	7.54	7.05	2.67	7.67	7.22	2.57	7.58	7.12	2.60
03	2.74	2.51	1.10	1.91	1.73	0.80	1.09	1.02	0.39	0.84	0.78	0.33	0.82	0.76	0.30
04	0.93	0.88	0.31	0.53	0.47	0.24	0.59	0.52	0.28	0.46	0.42	0.19	0.39	0.35	0.16
05	3.90	3.54	1.64	4.04	3.46	2.08	1.99	1.80	0.84	2.07	1.87	0.89	1.79	1.61	0.76
06	1.42	1.34	0.48	1.27	1.17	0.49	1.03	0.94	0.43	0.93	0.87	0.34	0.81	0.76	0.30
07	0.76	0.65	0.40	1.04	0.88	0.55	0.90	0.76	0.49	0.44	0.40	0.19	0.42	0.38	0.17
08	6.46	5.40	3.54	6.40	5.68	2.93	4.09	3.67	1.81	4.01	3.49	2.04	3.84	3.32	1.93
09	2.26	1.86	1.28	2.77	2.15	1.74	2.69	2.44	1.12	1.97	1.72	0.96	1.55	1.38	0.69
10	2.05	1.88	0.82	2.29	2.06	0.99	2.59	2.25	1.28	1.78	1.58	0.82	1.72	1.53	0.79
Average	7.00	5.68	2.77	6.35	5.59	2.97	13.26	12.49	4.44	4.04	3.64	1.74	3.82	3.45	1.64

limits u to the interval $[-a, a]$ (values above a are set to a , values below $-a$ are set to $-a$). In words: compute the vertical difference between the ICP output and the incoming initial guess, clip this difference to the per-frame bound, then add it to the initial vertical position to form the final output. This enforces a hard bound on each frame's vertical change and suppresses cumulative z drift under weak vertical observability.

IV. EXPERIMENTS

We evaluate the proposed odometry pipeline on the public KITTI Odometry benchmark to assess both accuracy and robustness under real traffic dynamics. Our method is compared against recent state-of-the-art LiDAR odometry approaches. In addition, we conduct ablation studies to isolate the effect of the two-stage registration schedule (F2F followed by F2LM) and the vertical constraints (iteration-level z gating and frame-level z clamping), as well as the hash-voxel map design. All experiments are performed on a desktop with an Intel i7-13700F CPU and 32 GB RAM.

A. Accuracy Evaluation and Comparison

The KITTI Odometry set provides ground-truth trajectories for 11 sequences (00–10), covering a broad spectrum of driving scenarios: dense urban streets (00, 06, 07, 08), rural roads (02, 03, 04, 05, 09, 10), and highway driving (01). This diversity introduces varying motion patterns, occlusions, and object dynamics, offering a comprehensive testbed for robustness and accuracy.

To assess the performance of our method, we compared it with leading LiDAR odometry approaches: Fast-gicp [6], Faster-gicp [7], DLO [5], and Kiss-icp [9]. Absolute Pose Error (APE) measures the direct difference between the estimated pose and the ground truth pose, providing a clear indication of method accuracy and global trajectory consistency. We utilized Root Mean Square Error (RMSE), Mean, and Standard Deviation (Std) of APE as evaluation metrics for method accuracy, as presented in Table I.

From Table I, Ours achieves the best average performance on KITTI with RMSE/Mean/Std of 3.82/3.45/1.64. Relative to the comparison methods, the average RMSE is reduced by 45.4% (Fast-gicp: 7.00 to 3.82), 39.8% (Faster-gicp: 6.35

to 3.82), 71.2% (DLO: 13.26 to 3.82), and 5.5% (Kiss-icp: 4.04 to 3.82). The Mean and Std also improve on average by 39.3%/40.8%, 38.3%/44.8%, 72.4%/63.1%, and 5.2%/5.8%, respectively, indicating that the proposed multi-scale registration with vertical constraints enhances both accuracy and trajectory stability.

At the per-sequence level, the largest gains appear where fast motion and weak vertical observability are prominent. On the highway sequence 01, the RMSE is 18.83 (Kiss-icp reports 20.18; Fast-gicp and Faster-gicp report 23.67 and 24.21), reflecting a clear advantage; on the urban turning sequence 07, the RMSE is 0.42 while the next-best method reports 0.44. In rural sequences 05/06/09/10, consistent improvements are obtained; for sequence 09, RMSE drops from 1.97 to 1.55 (about 21% reduction). Two exceptions are sequences 00 and 02, where DLO attains the lowest RMSE; nevertheless, our results remain competitive (e.g., 7.58 on 02 against 7.54) while maintaining low dispersion. Overall, the combination of motion-predicted two-stage registration and vertical constraints ICP enlarges the convergence basin, mitigates local optima under occlusions and dynamics, and suppresses z -drift across highway, urban, and rural scenarios.

For visualization, we select one representative KITTI sequence per scenario—07 (urban), 05 (rural), and 01 (highway). In Fig. 2, our trajectories are almost indistinguishable from the ground truth across urban, rural, and highway settings, with noticeably smaller deviations at sharp turns and over long distances. The advantages are consistent: (1) in dense urban scenes with frequent occlusions and rapid heading changes, our poses remain stable through intersections and tight corners, avoiding the lateral drift and corner overshoot seen in baselines; (2) on highways, we keep lane-level consistency and correct scale at sustained speeds, showing minimal bias on long curves where other methods gradually diverge; and (3) on rural roads with sparse structure, our cumulative drift grows much more slowly, yielding tighter end-point closure. These gains stem from the motion-predicted initialization and two-stage schedule (F2F then F2LM) that expand the convergence basin, plus the vertical constraints that suppress z -drift under weak pitch observability. The hash-voxel map with fixed neighborhood also provides stable correspondences in dynamic traffic, further reducing failure cases and ensuring accurate, smooth trajectories.

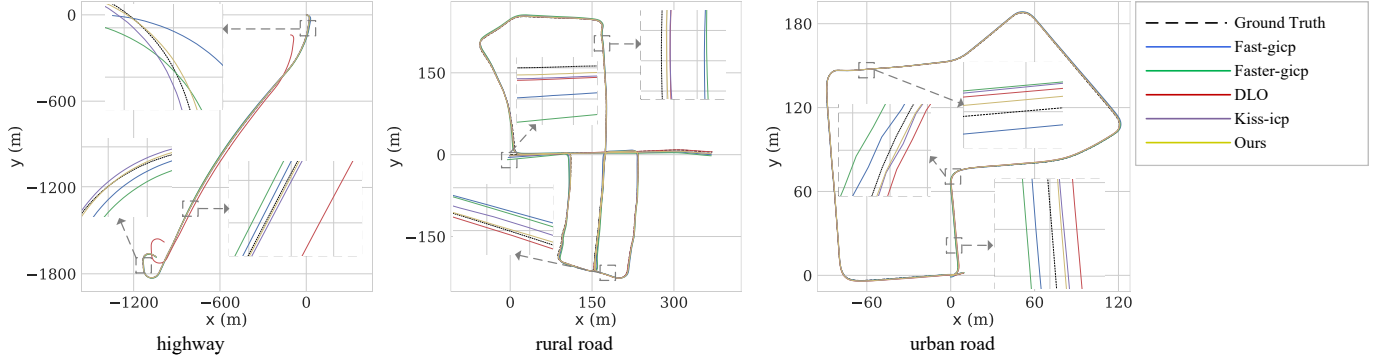


Fig. 2. Trajectory comparison on representative urban (07), rural (05), and highway (01) sequences. Estimated trajectories are compared against ground truth.

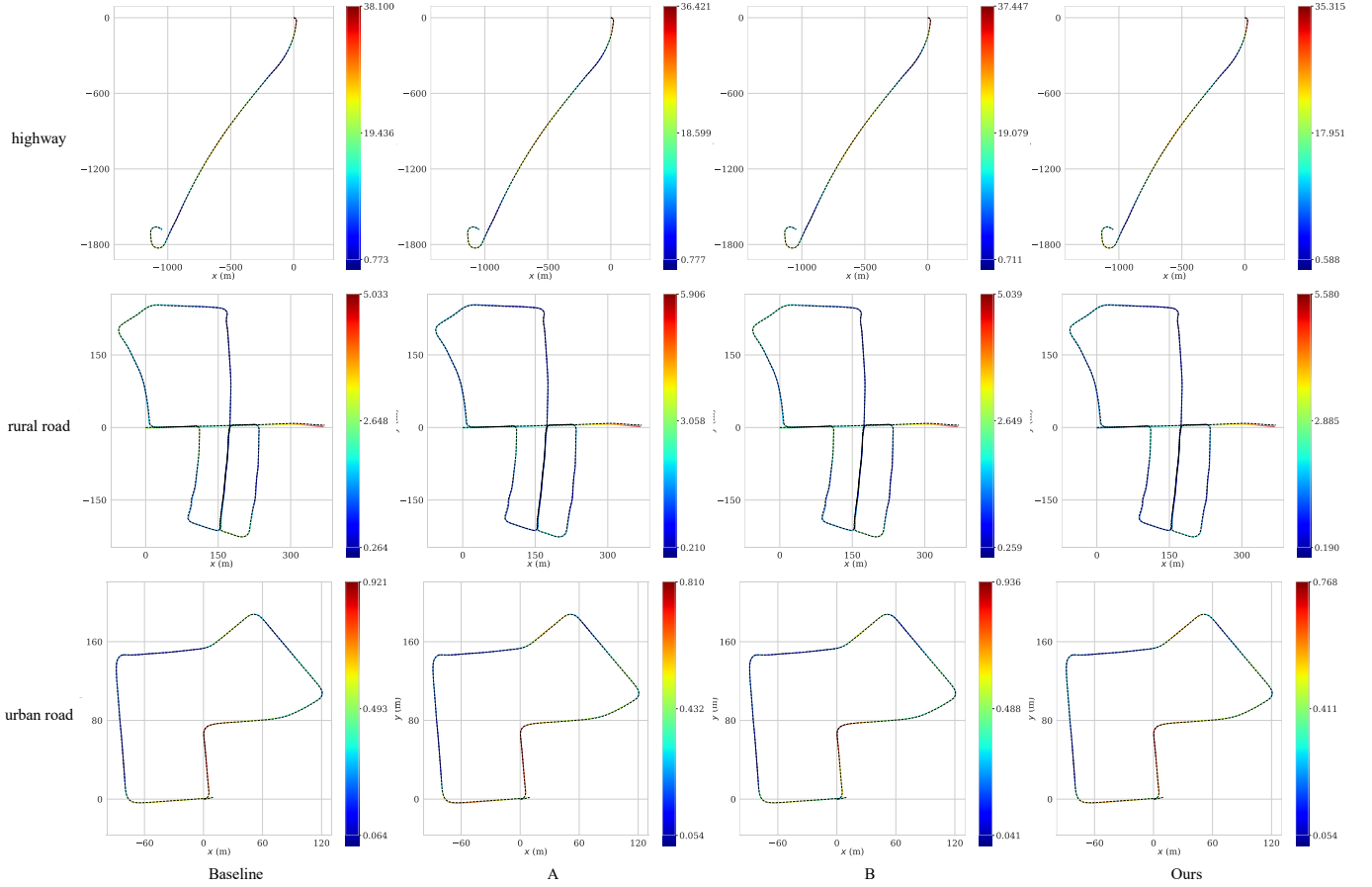


Fig. 3. Trajectory heatmap comparison for urban (00), highway (01), and rural (05). Color intensity encodes error magnitude (darker indicates larger error).

B. Ablation Study

To assess each component of our method, we conduct ablations on KITTI using a common preprocessing and hash-voxel map. We compare four configurations:

- Baseline: single-stage ICP to the local map (no motion-predicted initialization, no F2F stage, no vertical constraints).
- A (w/o F2F): our pipeline without the F2F pre-alignment; still uses motion-predicted initialization and F2LM refinement with vertical constraints ICP.
- B (w/o V-constraint): our pipeline without vertical constraints (standard ICP in both stages); retains motion-predicted initialization and the two-stage schedule.
- Ours: full system with motion-predicted initialization, two-stage (F2F and F2LM) schedule, and vertical constraints ICP.

Table II shows that Ours achieves the lowest Mean and RMSE on all three sequences. Relative to Baseline, the RMSE decreases from 20.216 to 18.830 on Seq. 01, from 1.981 to 1.786 on Seq. 05, and from 0.484 to 0.418 on Seq. 07; the corresponding Mean errors are reduced in the same order

TABLE II
ABLATION ON KITTI SEQUENCES 01/05/07. METRICS: MEAN / RMSE /
STD (LOWER IS BETTER).

Seq	Metric	Baseline	A (w/o F2F)	B (w/o V-const)	Ours
01	Mean	18.170	17.363	17.968	16.926
	RMSE	20.216	19.319	19.986	18.830
	Std	8.862	8.470	8.752	8.252
05	Mean	1.810	1.749	1.742	1.614
	RMSE	1.981	1.950	1.895	1.786
	Std	0.806	0.862	0.745	0.765
07	Mean	0.434	0.401	0.398	0.381
	RMSE	0.484	0.443	0.445	0.418
	Std	0.214	0.188	0.199	0.172

(18.170 to 16.926, 1.810 to 1.614, 0.434 to 0.381). Standard deviation also improves on Seq. 01 and Seq. 07; on Seq. 05, method B attains the lowest Std (0.745), while Ours remains competitive (0.765) and still yields the best Mean/RMSE.

Removing the F2F stage (A) consistently worsens accuracy compared with Ours, most clearly on the high-speed highway sequence (01), indicating that short-baseline pre-alignment stabilizes subsequent refinement. Removing the vertical constraint (B) degrades results across sequences, confirming that explicit control of the weakly observed z direction improves stability and alignment. The slight Std advantage of B on Seq. 05 suggests a different error distribution, but the overall accuracy (Mean/RMSE) remains best with the full design.

Fig. 3 compares trajectory heat maps in three representative scenarios. The vertical bars on the right indicate error ranges, where darker colors represent larger errors. Baseline exhibits dispersed and darker traces, especially at intersections and turning regions. Method A produces more compact paths, but noticeable drifts remain during sharp turns. Method B, benefiting from the two-stage schedule, further reduces errors but still shows slight vertical bias in some segments. In contrast, Ours consistently achieves the lightest and most compact overlays across all scenarios, closely following the ground truth with the smallest error ranges.

Overall, the ablations indicate that (i) motion-predicted initialization plus the two-stage schedule enlarges the convergence basin and mitigates local minima, and (ii) the vertical constraint effectively suppresses z -drift when vertical observability is weak. The full configuration combines these effects and delivers the most accurate and stable trajectories in dynamic driving environments.

V. CONCLUSION

LiDAR odometry is fundamental to autonomous navigation, yet conventional single-stage ICP pipelines are prone to local minima under unreliable initialization and struggle with weak vertical observability in spinning LiDARs, especially in dynamic scenes. These issues often manifest as unstable z updates and degraded global accuracy.

We presented a LiDAR-only odometry framework, Multi-scale Registration of LiDAR Odometry Based on Vertical Constraints in Dynamic Environments. The system leverages motion-predicted initialization and a two-stage schedule—first F2F alignment to rapidly correct short-baseline motion, then

frame-to-local-map refinement for accuracy. A hash-voxel map with bounded buckets and fixed 27-neighborhood search supports fast, stable correspondence queries. Experiments on KITTI demonstrate consistent reductions in mean and RMSE against mainstream baselines.

Future work will incorporate loop closure for long-term drift mitigation and explore lightweight fusion with IMU/vision to further strengthen robustness in highly dynamic, elevation-changing environments.

REFERENCES

- [1] 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.
- [2] 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*, 2024.
- [3] Y. Xu, C. Chen, B. Yang, W. Wu, S. Sun, Z. Wang, L. Li, and Q. Zou, "Atcm: Aerial-terrestrial lidar-based collaborative simultaneous localization and mapping," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 63, pp. 1–14, 2025.
- [4] M. Yokozuka, K. Koide, S. Oishi, and A. Banno, "Litamin2: Ultra light lidar-based slam using geometric approximation applied with kl-divergence," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 11 619–11 625.
- [5] K. Chen, B. T. Lopez, A.-a. Agha-mohammadi, and A. Mehta, "Direct lidar odometry: Fast localization with dense point clouds," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2000–2007, 2022.
- [6] K. Koide, M. Yokozuka, S. Oishi, and A. Banno, "Voxelized gicp for fast and accurate 3d point cloud registration," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 054–11 059.
- [7] J. Wang, M. Xu, F. Foroughi, D. Dai, and Z. Chen, "Fastergicp: Acceptance-rejection sampling based 3d lidar odometry," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 255–262, 2022.
- [8] S. Wang, J. Zhang, and X. Tan, "Pdli-lio: A precise and direct slam system toward large-scale environments with loop closures," *IEEE Transactions on Intelligent Transportation Systems*, 2023.
- [9] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "Kiss-icp: In defense of point-to-point icp—simple, accurate, and robust registration if done the right way," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 1029–1036, 2023.
- [10] L. He, W. Li, Y. Guan, and H. Zhang, "Ilgicp: Intensity and geometry enhanced lidar odometry," *IEEE Transactions on Intelligent Vehicles*, 2023.
- [11] Z. Shen, J. Wang, C. Pang, Z. Lan, and Z. Fang, "A lidar-imu-gnss fused mapping method for large-scale and high-speed scenarios," *Measurement*, vol. 225, p. 113961, 2024.
- [12] X. Zhang, Z. Fan, X. Tan, Q. Liu, and Y. Shi, "Spatiotemporal adaptive attention 3d multiobject tracking for autonomous driving," *Knowledge-Based Systems*, vol. 267, p. 110442, 2023.
- [13] 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*, pp. 1–1, 2024.
- [14] W. Wang, C. Wang, J. Liu, X. Su, B. Luo, and C. Zhang, "Hvl-slam: Hybrid vision and lidar fusion for slam," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–14, 2024.
- [15] G. P. C. Júnior, A. M. C. Rezende, V. R. F. Miranda, R. Fernandes, H. Azpúrua, A. A. Neto, G. Pessin, and G. M. Freitas, "Ekl-loam: An adaptive fusion of lidar slam with wheel odometry and inertial data for confined spaces with few geometric features," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 1458–1471, 2022.
- [16] 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.
- [17] D. Xu, J. Liu, Y. Liang, X. Lv, and J. Hyypä, "A lidar-based single-shot global localization solution using a cross-section shape context descriptor," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 189, pp. 272–288, 2022.

- [18] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," in *Robotics: Science and systems*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.
- [19] A. Tagliabue, J. Tordesillas, X. Cai, A. Santamaria-Navarro, J. P. How, L. Carlone, and A.-a. Agha-mohammadi, "Lion: Lidar-inertial observability-aware navigator for vision-denied environments," in *Experimental Robotics: The 17th International Symposium*. Springer, 2021, pp. 380–390.
- [20] 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.
- [21] 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.
- [22] S. Fang, H. Li, and M. Yang, "Lidar slam based multivehicle cooperative localization using iterated split cif," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 21 137–21 147, 2022.
- [23] J. Behley and C. Stachniss, "Efficient surfel-based slam using 3d laser range data in urban environments," in *Robotics: science and systems*, vol. 2018, 2018, p. 59.
- [24] Y. S. Park, H. Jang, and A. Kim, "I-loam: Intensity enhanced lidar odometry and mapping," in *2020 17th International Conference on Ubiquitous Robots (UR)*, 2020, pp. 455–458.
- [25] Q. Li, S. Chen, C. Wang, X. Li, C. Wen, M. Cheng, and J. Li, "Lonet: Deep real-time lidar odometry," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8473–8482.
- [26] Z. Yuan, Q. Wang, K. Cheng, T. Hao, and X. Yang, "Sdv-loam: semi-direct visual-lidar odometry and mapping," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 11 203–11 220, 2023.
- [27] B. Zhou, H. Mo, S. Tang, X. Zhang, and Q. Li, "Backpack lidar-based slam with multiple ground constraints for multistory indoor mapping," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–16, 2023.
- [28] Q. Zou, Q. Sun, L. Chen, B. Nie, and Q. Li, "A comparative analysis of lidar slam-based indoor navigation for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6907–6921, 2021.
- [29] C. Zheng, Q. Zhu, W. Xu, X. Liu, Q. Guo, and F. Zhang, "Fast-livo: Fast and tightly-coupled sparse-direct lidar-inertial-visual odometry," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 4003–4009.
- [30] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," in *Robotics: science and systems*, vol. 2, no. 4. Seattle, WA, 2009, p. 435.
- [31] C. Qian, Z. Xiang, Z. Wu, and H. Sun, "Rf-liv: Removal-first tightly-coupled lidar inertial odometry in high dynamic environments," *arXiv preprint arXiv:2206.09463*, 2022.