HP²-SLAM: Adaptive Hybrid ICP for Robust and Efficient LiDAR SLAM

Nam Tran[1,†], Thu Tran[1,†], Hieu Phan[1,†], Thai Luu[1,†], Toan Nguyen[2], William J. Beksi[3], and Tuan Dang[1,*]

***Abstract*— Achieving robustness, accuracy, and efficiency simultaneously remains a central challenge in light detection and ranging (LiDAR) simultaneous localization and mapping (SLAM). While learning-based approaches deliver strong benchmark performance, they often require extensive training, substantial computational resources, and struggle to generalize to unseen or degenerate environments. Geometry-based methods are efficient and interpretable, yet their performance degrades in planar or repetitive scenes due to limitations of standard iterative closest point (ICP) formulations. We present HP²-SLAM , a minimalist yet robust LiDAR SLAM framework built around a neighborhood-size adaptive hybrid ICP. Our key insight is a planarity-aware adaptive threshold that dynamically classifies correspondences based on local geometric structure and density, thereby enabling a principled balance between point-to-plane and point-to-point residuals. This formulation stabilizes alignment in both structured and degenerate environments without feature engineering, learning modules, or dataset-specific tuning. Integrated into a complete SLAM pipeline with submap management, loop closure detection, and pose graph optimization, HP²-SLAM consistently outperforms strong geometry-based baselines across publicly available datasets while maintaining real-time performance on commodity hardware. Our results demonstrate that carefully designed geometric adaptation can achieve strong generalization and robustness without sacrificing simplicity or efficiency.**

I. Introduction

Simultaneous localization and mapping (SLAM) is a fundamental capability for autonomous robots operating in unknown environments. Learning-based SLAM systems such as DeepFactors [1], DROID-SLAM [2], and NICE-SLAM [3] have recently demonstrated strong benchmark performance. However, these methods often require extensive training data, high computational resources, and careful domain adaptation. Their generalization to unseen environments remains limited, and their black-box nature complicates deployment on resource-constrained robotic platforms.

Geometry-based light detection and ranging (LiDAR) SLAM remains attractive due to its efficiency, interpretability, and cross-domain robustness. Classical systems such as LOAM [4], A-LOAM [5], and LeGO-LOAM [6] rely on feature extraction and scan-to-map registration. More recently, minimalist approaches such as KISS-ICP [7] and KISS-SLAM [8] show that competitive performance can be achieved without handcrafted features or learning modules.

[1]Department of Electrical Engineering and Computer Science, University of Arkansas, Fayetteville, AR, USA; [2]VinRobotics; [3]Department of Computer Science, University of Texas at Dallas, Richardson, TX, USA. [†]Contributed to this work as summer interns at the Cognitive Robotics Lab, University of Arkansas. [*]Corresponding author: Tuan Dang (`tuand@uark.edu`).

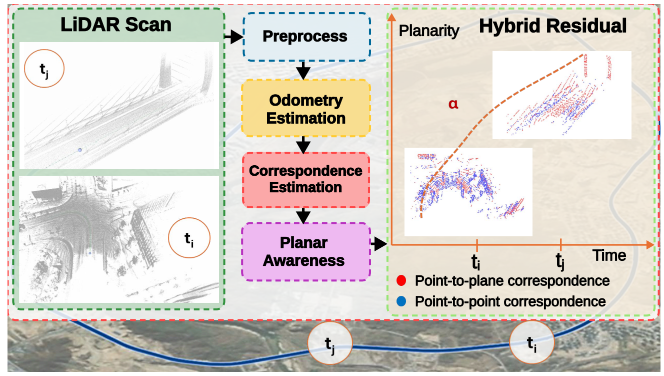

Fig. 1: HP²-SLAM conceptualization. Each LiDAR scan is registered to a local map using a density-aware planarity criterion that separates planar and non-planar correspondences. The hybrid ICP weight α is adaptively updated at each scan based on the observed planar ratio, increasing in structured environments and decreasing in less planar settings.

Nevertheless, a fundamental limitation persists: ICP degeneracy in planar or repetitive environments. On highways, bridges, and long corridors, point-to-point ICP [9] becomes ill-conditioned, causing drift. Point-to-plane ICP [10] improves convergence in structured regions, but is sensitive to noise and unreliable when geometric support is weak.

Various hybrid strategies have attempted to address the issue of ICP degeneracy but have introduced other drawbacks. Methods such as MULLS [11] and GICP [12] combine residual formulations but often require complex modeling or sensitive parameter tuning. GenZ ICP [13] introduces adaptive residual weighting based on degeneracy metrics and achieves strong odometry performance. However, its adaptation is driven by global degeneracy indicators rather than explicit correspondence-level geometric classification, and it focuses on front-end odometry without integration into a complete SLAM framework.

This motivates a central question: *Can residual balancing be derived directly from local geometric structure in a density-aware and principled manner, while preserving minimalist design?* To this end, we propose HP²-SLAM, a lightweight LiDAR SLAM framework built around a neighborhood-size adaptive hybrid ICP. As shown in Fig. 1, instead of fixed thresholds or heuristic weighting, we introduce a density-aware adaptive planarity criterion that scales with local neighborhood size. Correspondences are classified into planar and non-planar sets, and the balance between point-to-plane and point-to-point residuals emerges directly from real-time geometric observations. This reduces parameter sensitivity and improves stability in both structured and degenerate environments.

Building on KISS-SLAM [8], we integrate this adaptive hybrid ICP into a complete pipeline that includes submap management, loop closure detection, and pose graph optimization. This pipeline helps HP^2-SLAM achieve not only accurate odometry but global consistency as well. Our main contributions are as follows.

- A **neighborhood-size–adaptive planarity criterion** for density-aware correspondence classification.
- An **adaptive hybrid ICP formulation** that dynamically balances residuals based on observed geometry.
- A **minimalist full-SLAM integration** that achieves improved global consistency while preserving real-time efficiency.

Extensive experiments on public SLAM datasets show that HP^2-SLAM consistently outperforms strong geometry-based baselines, including KISS-SLAM [8], MULLS [11], CT-ICP [14], and GenZ-ICP [13], while maintaining real-time performance and robustness across different types of LiDAR sensors. We will release our source code to foster reproducibility and validation, and to contribute to the growth of the research community.

II. Related Work

A. LiDAR-Based SLAM and ICP-Based Odometry

LiDAR-based SLAM is widely used for large-scale robotic navigation due to its accurate and illumination-insensitive range sensing. Classical systems such as LOAM [4] and its variants [5], [6] rely on handcrafted geometric features and careful parameter tuning for scan-to-map registration. More recent minimalist approaches, like KISS-ICP [7] and KISS-SLAM [8], demonstrate that competitive performance can be achieved using pure geometric registration without feature extraction or auxiliary sensors. Most of these methods rely on the ICP algorithm, typically adopting either a point-to-point or point-to-plane formulation. The point-to-point formulation is efficient but becomes ill-conditioned in planar or repetitive environments. The point-to-plane formulation [15], [16] improves convergence in structured scenes but demands reliable normal estimation and sufficient geometric support. Recent analyses [17] highlight the complementary strengths and weaknesses of these residuals under degeneracy. The point-to-plane formulation was originally introduced by Chen and Medioni [18], constraining optimization along the normal plane to address slow convergence, while Felix et al. [19] further refined alignment by minimizing the distance from source points to the tangent planes of corresponding target points within a hybrid-ICP framework. However, existing systems usually employ a single residual globally or combine them using fixed heuristics, limiting their ability to adapt to changing geometric conditions along a trajectory. This limitation motivates our work, in which we introduce a neighborhood-size–adaptive planarity criterion that classifies correspondences based on local geometric structure and density. Defining a planarity criterion enables dynamic balancing between point-to-point and point-to-plane residuals within a unified SLAM framework, thereby improving robustness while preserving computational simplicity.

B. Hybrid Alignment Strategies

To address the limitations of individual error metrics, researchers introduced hybrid ICP methods that integrate multiple alignment strategies. For instance, MULLS [11] jointly optimizes point-to-point, point-to-plane, and point-to-line residuals in a least-squares framework, but requires complex feature extraction and tuning. Generalized-ICP (GICP) [12] introduces a probabilistic formulation that unifies point-to-point and point-to-plane ICP under local planarity assumptions, improving accuracy over single-metric approaches, and has been further enhanced in subsequent works [20], [21], [22], with widespread applications in large-scale and underground environments [23], [24]. More recently, GenZ-ICP [13] adapts the weighting between point-to-point and point-to-plane terms based on local geometry. This adaptive strategy improves robustness in degenerate environments, such as long corridors, and enhances performance in diverse real-world settings. This survey [25] highlights the growing trend toward hybrid metrics that balance efficiency and robustness. Unlike prior hybrid formulations that rely on fixed or heuristic thresholds for planarity classification, our method introduces an adaptive planarity threshold that dynamically adjusts with local density. This design enables us to separate planar and non-planar correspondences more reliably, allowing the balance between point-to-plane and point-to-point residuals to be derived directly from real-time observations. In contrast to MULLS, we avoid explicit feature extraction and complex parameter tuning while keeping the design minimalist. Unlike GenZ-ICP, which is limited to odometry, our method integrates hybrid ICP with an adaptive planarity threshold into a complete SLAM framework that combines local mapping, loop closure, and global optimization.

III. System Overview

We build HP^2-SLAM on top of KISS-SLAM, while replacing its original KISS-ICP module (point-to-point ICP) with a novel adaptive ICP algorithm. The objective is to preserve the "small and simple" design philosophy of KISS-SLAM, while improving robustness in geometrically degenerate environments, such as long corridors or large planar surfaces. An overview of our system is shown in Fig. 2 and described below.

LiDAR odometry (adaptive ICP). Following the original KISS-SLAM pipeline, each incoming LiDAR scan is first *deskewed* to compensate for motion distortion and *downsampled* to reduce point density. A constant-velocity motion model provides an initial pose guess. The main difference of our approach lies in the refinement stage. Instead of relying solely on point-to-point ICP as in KISS-ICP, we introduce an adaptive ICP mechanism. This mechanism classifies correspondences into two categories: (i) planar-based methods, optimized using point-to-plane, and (ii) non-planar-based methods, optimized using point-to-point residuals. An adaptive weight α is determined based on the ratio of

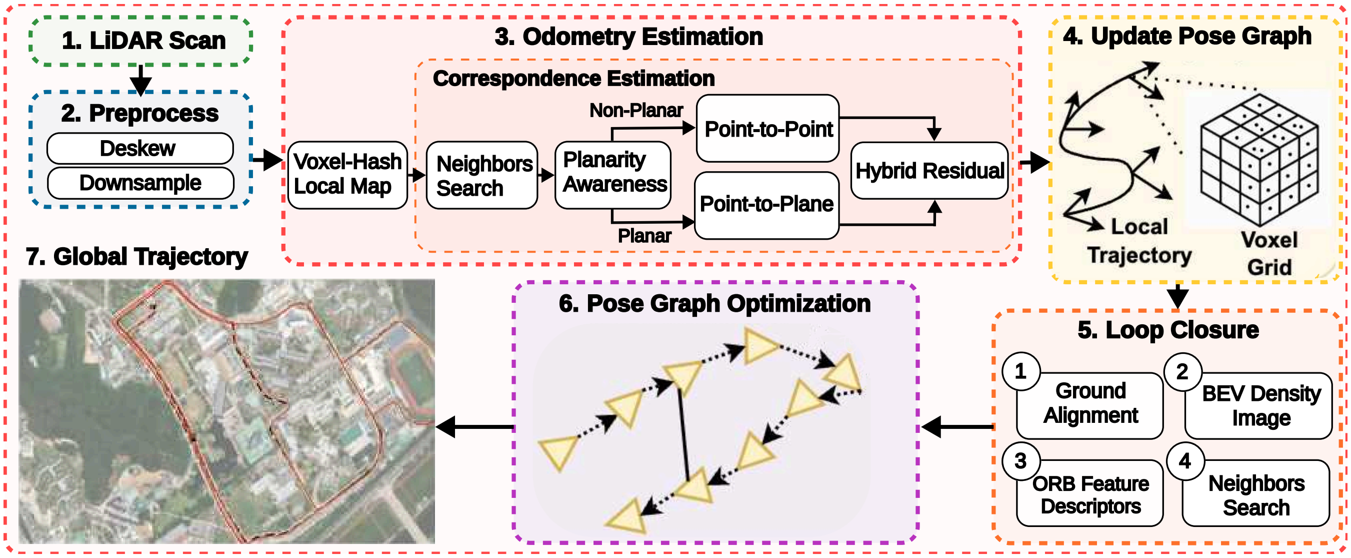

Fig. 2: An overview of HP^2-SLAM. Our system includes LiDAR preprocessing, odometry estimation, local mapping, loop-closure detection using ground-aligned BEV and ORB with RANSAC verification, and pose-graph optimization for global consistency. Odometry is improved by a planarity-aware hybrid residual that applies point-to-plane in planar regions and point-to-point in the non-planar areas with adaptive weighting. This hybrid approach stabilizes odometry and reduces longitudinal drift when mapping degenerate environments, such as long corridors, roads, and bridges.

planar to non-planar correspondences, enabling the system to balance the two types of constraints automatically. After convergence, a downsampled version of the registered scan is integrated into the current local map.

Local mapping. As in KISS-SLAM, we exploit short-term odometry consistency by splitting the global map into *submaps*. Each submap M_k is associated with a *keypose* T_k, a voxel grid V_k, and a local odometry trajectory. Deskewed and downsampled scans are continuously integrated into the active submap. Once the traveled distance exceeds a predefined threshold β, a new submap is initialized at the current keypose while the previous submap is finalized and stored in the *pose graph*. In the pose graph, nodes represent keyposes and edges encode odometry constraints.

Loop closure and pose graph optimization. To reduce accumulated odometry drift, HP^2-SLAM incorporates a loop closure detection module. Following the method of Gupta et al. [26], candidate loop closures are identified by aligning the current local map point cloud with the xy-plane of the keypose coordinate system. All points are then projected into a bird's-eye-view (BEV) representation by computing the density of 3D LiDAR points within each 2D grid cell. Binary ORB descriptors [27] are extracted from these BEV images and matched against previously stored descriptors. To enable efficient loop-candidate matching, these ORB descriptors are stored and queried in a Hamming distance embedding binary search tree [28]. If a potential loop candidate is found, then geometric verification is performed using RANSAC to provide 2D alignment between the density images.

To further ensure robustness, we additionally validate loop closures in 3D by checking the overlap consistency between the registered submaps. Only candidates with sufficient overlap are kept and added to the pose graph, reducing the risk of false loop constraints. For global consistency, we optimize the pose graph using the approach of Grisetti et al. [29]. In this graph, keyposes are kept fixed while new nodes and edges are introduced from odometry and validated loop closures. The optimization is performed at a fine-grained level, redistributing small odometry errors among scan poses within local trajectories. This step effectively reduces drift and ensures a globally consistent map.

IV. Methodology

A. Problem Formulation

The LiDAR SLAM problem is defined as follows. Given a sequence of 3D point clouds $\{P_t\}_{t=1}^{T}$, where each $P_t \in \mathbb{R}^3$ is a point cloud observed at time t, the robot state at t is represented by a matrix $T_t \in SE(3)$, with $T_t = [R_t|t_t]$, $R_t \in SO(3)$ and $t_t \in \mathbb{R}^3$. The goal of LiDAR SLAM is to jointly estimate the trajectory $X = \{T_t\}_{t=1}^{T}$ and a global map M.

KISS-SLAM estimates the pose of the current scan by registering it to a voxelized local map via a point-to-point ICP formulation. Concretely, given a set of correspondences $\{(p_i, q_i)\}$ between the target and source point clouds, the point-to-point residual is

$$e_{pp}(p_i, q_i; R, t) = p_i - (Rq_i + t). \tag{1}$$

The pose $(R, t) \in SE(3)$ is obtained by minimizing the sum of squared residuals

$$\min_{R,t} \sum_i \| p_i - (Rq_i + t) \|^2. \tag{2}$$

As discussed in Sec. II, the point-to-point objective is computationally attractive, but can be ill-conditioned in predominantly planar scenes. A complementary formulation is the point-to-plane residualthat leverages local surface normals to enforce orthogonal consistency, i.e.,

$$e_{pl}(p_i, q_i, R, t) = n_i^\top (p_i - (Rq_i + t)), \tag{3}$$

where n_i is the normal vector at p_i. In this work, we define a hybrid alignment objective as a convex combination of point-to-point and point-to-plane residuals with adaptive weight α. To achieve this, we first analyze the local geometric structure of each correspondence and use it to balance the two residual terms. For each candidate correspondence, we form a local map neighborhood $\mathcal{N}$ of size $n = |\mathcal{N}|$ around the target point and compute the covariance matrix C. More formally, let the eigenvalues of C be ordered as $\lambda_1 \geq \lambda_2 \geq \lambda_3$. We define a local planarity score,

$$s \;=\; \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3}, \tag{4}$$

such that smaller values indicate a more planar structure.

Rather than employing a fixed cutoff, we define the following neighborhood-size–adaptive decision boundary that scales inversely with the local neighborhood size:

$$\tau(n) \;=\; \tau_{\mathrm{ref}} \frac{N_{\mathrm{ref}}}{n}, \qquad \tau_{\min} \leq \tau(n) \leq \tau_{\max}. \tag{5}$$

A correspondence is classified as planar if the local planarity score s falls below $\tau(n)$, and non-planar otherwise. The evaluation is performed only when the neighborhood contains at least $n_{\min}$ points, ensuring that the covariance estimate is statistically meaningful. By construction, $\tau(n)$ decreases with increasing density, enforcing stricter conditions in well-sampled regions. Conversely, in sparse neighborhoods it relaxes, allowing structural cues to be retained even under limited observations. We denote the resulting planar and non-planar sets as $\mathcal{P}$ and $\mathcal{Q}$, respectively.

To ensure practicality, we introduce a reference threshold τ_{ref}, which specifies the nominal decision boundary at a neighborhood size N_{ref}. In practice, τ_{ref} acts only as a coarse scaling factor and can be kept within a narrow range from 0.01 to 0.2 across datasets. Although lowering τ_{ref} may yield further improvements in certain cases, the adaptive formulation remains far less sensitive to this choice than fixed-threshold schemes. This property substantially reduces the need for dataset-specific retuning and strengthens the robustness of our pipeline across diverse sensors and environments.

We formulate the hybrid alignment objective as

$$\begin{aligned} \underset{R,t}{min}[\alpha \sum_{(p_i,q_i)\in\mathcal{P}} \rho(e_{\mathrm{pl}}(p_i, q_i, R, t) + \\ (1-\alpha) \sum_{(p_i,q_i)\in\mathcal{Q}} \rho(e_{\mathrm{pp}}(p_i, q_i, R, t)], \end{aligned} \tag{6}$$

where $\rho(\cdot)$ is a robust loss function. The mixing weight $\alpha \in [0, 1]$ is set adaptively based on the current planar ratio, i.e.,

$$\alpha = \frac{|\mathcal{P}|}{|\mathcal{P}|+|\mathcal{Q}|}. \tag{7}$$

In our work, we use the Geman-McClure estimator to suppress the influence of outliers and dynamic structures. Concretely,

$$\rho(e) = \frac{e^2}{e^2 + c^2}, \tag{8}$$

where e is the residual and c is a scale parameter that controls the degree of down-weighting for significant errors.

Unlike the standard least-squares loss, which penalizes large residuals quadratically and can be overly sensitive to outliers, the Geman-McClure estimator saturates as $|e|$ grows large. Specifically, when e is small, $\rho(e) \approx e^2/c^2$ behaves similarly to a squared error, ensuring good convergence for inlier correspondences. However, for large $|e|$, $\rho(e)$ approaches 1, effectively down-weighting the contribution of these residuals in the optimization. This property makes the loss function robust by reducing the influence of outliers or dynamic structures that would otherwise dominate the objective and skew the estimated pose.

In order to avoid manual tuning, we set $c = \sigma$, where σ is estimated online from recent motion prediction errors between the constant-velocity prior and the registered pose. More formally,

$$\mathrm{err}_k = \|\delta t_k\| + 2 r_{\max} \sin(\theta_k/2), \tag{9}$$

where δt_k is the translational error, $r_{\max}$ denotes the maximum LiDAR sensing range, and θ_k is the rotational error angle. The term $2r_{\max}\sin(\theta_k/2)$ converts the rotational deviation into an equivalent displacement at the farthest observed point, ensuring consistent units with the ICP residuals. The robust scale is then computed as the root mean square of the accumulated motion errors over a sliding window of size N,

$$\sigma = \sqrt{\frac{1}{N}\sum_{k=1}^{N} \mathrm{err}_k^2}. \tag{10}$$

This motion-adaptive scale enlarges the inlier region under aggressive motion and tightens it during slow motion, maintaining consistent robustness across varying motion regimes.

B. Implementation Details

HP^2-SLAM is built on KISS-ICP and extended into a KISS-SLAM pipeline with a hybrid alignment mechanism and adaptive planar classification. Each incoming LiDAR scan is first deskewed using a constant-velocity motion model to compensate for sensor motion,

$$P_t^{\mathrm{deskew}} = e^{(s_i - 1)\omega P_t}, \tag{11}$$

where $s_i \in [0, 1]$ is the normalized timestamp within the scan and $\omega = log(\Delta T) \in \mathfrak{se}(3)$, $\Delta T \in SE(3)$ denotes the relative transform from start to end in the scan. After deskewing, the scan is voxelized and downsampled. We maintain a voxel-hashed local map to support fast nearest-neighbor queries for ICP registration and efficient map updates.

For each candidate correspondence, we retrieve a local neighborhood from the voxel map and compute the covariance matrix via principal component analysis. From its

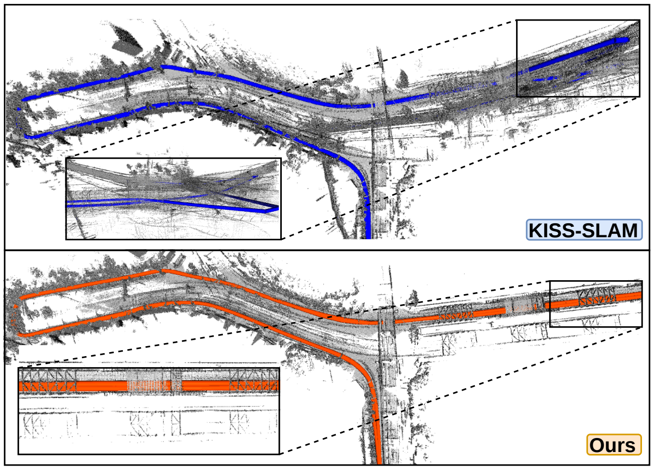

Fig. 3: Mapping results on the HeLiPR Bridge (Avia04) sequence. Our method demonstrates stable odometry and consistent map reconstruction across long bridge structures, which are challenging due to repetitive geometric patterns and limited features.

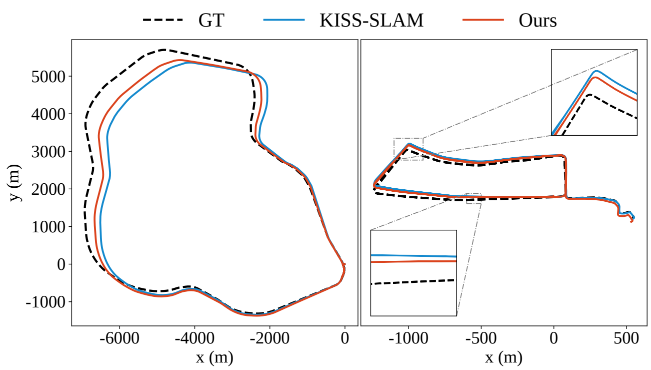

Fig. 4: Trajectory overlays on Sejong01 (left) and Riverside02 (right). Our approach aligns more closely with the ground truth (GT), particularly in challenging curved and loop regions, as shown in the zoomed-in views.

eigenvalues, we compute the planarity score s as in Equation (4) and compare it with the adaptive threshold $\tau(n)$ in Equation (5). This adaptive decision boundary increases tolerance in sparse regions while enforcing stricter flatness conditions in dense areas. As a result, the correspondences are automatically categorized into planar or non-planar classes.

Planar correspondences are aligned with a point-to-plane residual, whereas non-planar correspondences rely on point-to-point distances. Both sets are processed in parallel, and robust kernels are employed to mitigate the influence of outliers. The hybrid alignment objective is minimized in a Gauss-Newton framework based on the Lie algebra $\mathfrak{se}(3)$, updating the pose incrementally until convergence. The weighting factor between planar and non-planar residuals is computed adaptively from the current ratio of classified correspondences, balancing geometric constraints with robustness.

After alignment, the local map is updated with the registered scan. To maintain global consistency, we integrate a loop-closure module that uses density-preserving BEV projections. Each local map is projected onto the ground-aligned plane, from which ORB descriptors are extracted and stored in a database. Candidate matches are validated through RANSAC-based 2D alignment before being inserted as constraints into the global pose graph. This back-end optimization corrects long-term drift and enforces map consistency across revisits.

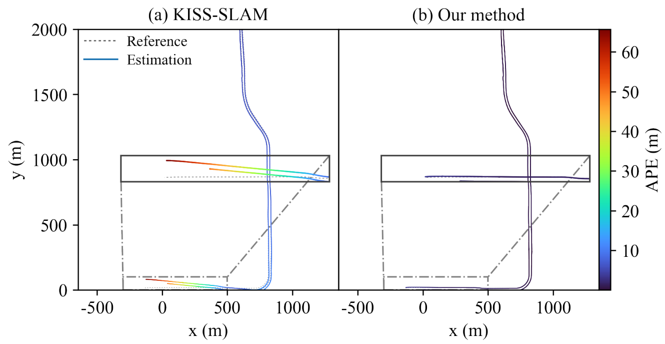

Fig. 5: An evaluation on the MathildaAVE sequence. Our methodology consistently yields lower error than the baseline.

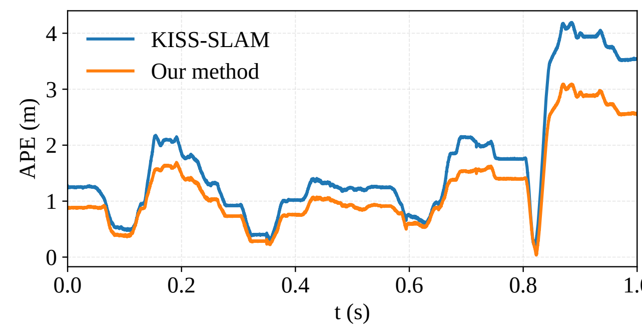

Fig. 6: The APE of raw trajectories on the SanJose-Downtown dataset. HP^2-SLAM maintains consistently smaller drift in the raw trajectory error, especially at segments with large error spikes.

Method	KAIST ATE (m)	DCC ATE (m)	Riverside ATE (m)	Sejong ATE (m)
KISS-SLAM [8]	2.96	3.68	8.34	<u>178.34</u>
PIN-SLAM [30]	**2.42**	<u>3.57</u>	<u>7.55</u>	782.96
CT-ICP [14]	2.76	4.03	8.37	-
MULLS [11]	34.21	27.30	66.60	1082.55
HP^2-SLAM (ours)	<u>2.71</u>	**3.56**	**7.50**	**146.51**

TABLE I: A performance comparison of HP^2-SLAM and the baselines on the Mulran dataset. A lower ATE indicates better localization accuracy. The best and second best performing methods are reported in **bold** and <u>underline</u>, respectively.

V. Evaluation

In this section, we present an extensive experimental evaluation of the performance of HP^2-SLAM. The goal is to examine both odometry accuracy and mapping consistency under diverse sensing setups and motion conditions. We validate our methodology on four representative benchmarks: KITTI [31], Apollo [32], MulRan [33], and HeLiPR [34],

KITTI-corrected	00 ATE (m)	01 ATE (m)	02 ATE (m)	03 ATE (m)	04 ATE (m)	05 ATE (m)	06 ATE (m)	07 ATE (m)	08 ATE (m)	09 ATE (m)	10 ATE (m)	AVG ATE (m)
KISS-SLAM [8]	0.89	3.11	1.92	0.67	0.38	0.67	0.32	0.33	2.55	1.25	0.71	1.16
GenZ-ICP [13]	5.19	25.47	11.19	3.27	0.90	1.50	0.86	0.67	4.46	3.07	2.32	5.36
HP^2-SLAM (ours)	**0.85**	**2.75**	**1.89**	**0.51**	**0.24**	**0.56**	**0.31**	**0.31**	**2.40**	**1.15**	**0.69**	**1.06**

TABLE II: Quantitative evaluations on the KITTI-corrected benchmark using the ATE in meters (m). Compared to KISS-SLAM and GenZ-ICP, HP^2-SLAM yields lower average error and improved performance across the majority of sequences.

Method	Bridge Aeva	Bridge Avia	Bridge Ouster	Round Aeva	Round Avia	Round Ouster	Town Aeva	Town Avia	Town Ouster
KISS-SLAM [8]	98.61	148.88	19.47	6.06	3.84	1.18	14.44	12.01	1.99
PIN-SLAM [30]	-	365.72	19.23	-	7.02	1.47	41.19	11.40	2.55
CT-ICP [14]	-	41.47	576.62	10.04	3.36	1.81	65.29	63.72	-
MULLS [11]	365.06	321.87	52.65	19.08	16.39	2.65	39.82	14.93	4.45
HP^2-SLAM (ours)	**57.37**	**26.77**	**17.51**	**5.77**	**3.17**	**1.08**	**11.65**	**10.45**	**1.91**

TABLE III: Quantitative results on the Bridge, Roundabout and Town sequences of the HeLiPR dataset using the ATE in meters (m). The best and second best performing methods are reported in **bold** and underline, respectively.

Method	BTS 2018-10-12	CP 2018-10-11	H237 2018-10-12	MAVE 2018-10-12	SB 2018-10-03	SJD 2018-10-11 (1)	SJD 2018-10-11 (2)
KISS-SLAM [8]	3.07	0.78	25.02	12.7	2.40	1.96	0.73
PIN-SLAM [30]	6.10	**0.64**	97.11	7.06	2.23	1.86	1.15
CT-ICP [14]	10.81	0.97	258.87	61.39	667.28	**0.78**	1.09
MULLS [11]	104.14	47.03	354.21	182.59	-	13.38	11.41
HP^2-SLAM (ours)	**2.63**	**0.64**	**24.41**	**4.01**	**2.07**	1.46	**0.63**

TABLE IV: Quantitative results on the Apollo dataset using the ATE in meters (m). The best and second best performing methods are reported in **bold** and underline, respectively.

Method	0 ATE (m)	1 ATE (m)	2 ATE (m)	3 ATE (m)	4 ATE (m)	5 ATE (m)	6 ATE (m)	7 ATE (m)	8 ATE (m)	9 ATE (m)	10 ATE (m)
GenZ-ICP [13]	4.72	17.21	15.41	3.55	0.64	1.62	1.16	0.73	5.62	3.64	2.16
Point-to-point only	3.50	3.15	**6.46**	0.52	**0.31**	1.49	0.42	0.31	2.20	1.31	0.71
Point-to-plane only	3.96	**2.49**	6.63	0.60	2.41	1.45	**0.40**	0.39	2.41	**0.88**	0.79
Adaptive hybrid (ours)	**3.46**	2.70	6.64	**0.51**	**0.31**	**1.32**	0.41	**0.30**	**2.19**	1.14	**0.67**

TABLE V: A front-end odometry ablation study on KITTI sequences. A lower ATE indicates better localization accuracy. The best and second best performing methods are reported in **bold** and underline, respectively.

KITTI-corrected	Neighbor Search	PCA	ICP	Backend	**Total**
Sequence 00	5.92	1.69	0.91	8.72	17.24
Sequence 01	8.68	2.50	1.07	11.36	23.61
Sequence 02	5.70	1.65	0.85	9.48	17.68
Sequence 03	8.37	2.4	1.06	9.91	21.74
Sequence 04	7.01	1.98	0.97	11.24	21.20
Sequence 05	5.95	1.71	0.86	9.56	18.08
Sequence 06	8.18	2.35	1.03	12.04	23.60
Sequence 07	7.12	2.04	0.97	7.24	17.37
Sequence 08	7.64	2.19	1.02	10.41	21.26
Sequence 09	6.97	2.03	0.96	10.23	20.19
Sequence 10	5.06	1.47	0.81	9.00	16.34
Average	6.96	2.00	0.96	9.93	19.85

TABLE VI: A runtime (ms) breakdown of the front-end and back-end modules on the KITTI sequences.

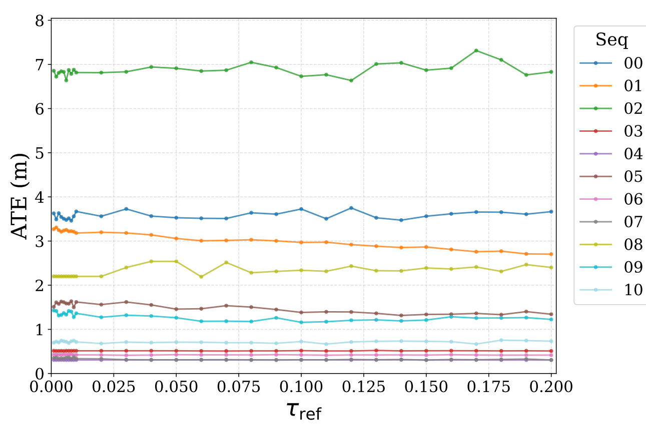

Fig. 7: The sensitivity of τ_{ref} on the KITTI sequences. The ATE remains stable over a wide parameter range, indicating low sensitivity.

across various LiDAR types including handheld and unmanned aerial vehicle (UAV) scenarios. The results demonstrate that HP^2-SLAM maintains low drift, exhibits strong adaptability across sensors with minimal or no parameter tuning, and produces globally consistent maps that can be directly used for robotic navigation and inspection tasks.

A. Experimental Setup

We report odometry accuracy using the absolute trajectory error (ATE), which is computed after performing a global $SE(3)$ alignment of the estimated trajectory to the ground truth via the *evo* package [35]. Additionally, we evaluate using the KITTI odometry drift metric. Ground-truth poses are taken from the official releases of each dataset. We start from a single default configuration and allow limited tuning to improve robustness. In clearly degenerate cases, we allow at most one generic adjustment to the base planarity threshold. The adaptive-planarity threshold $\tau(n)$ is clamped to $[\tau_{\min}, \tau_{\max}]$ and only these bounds may be modified. All experiments were executed on a machine with an Intel i7 CPU with 32 GB of RAM and an NVIDIA RTX 3060 GPU.

B. Qualitative Results

We evaluate our approach against the KISS-SLAM baseline across multiple challenging datasets. On the HeLiPR Bridge (Avia04) sequence, highly regular patterns and sparse UAV LiDAR data cause severe geometric degeneracy. Since KISS-SLAM relies solely on point-to-point ICP, it accumulates noticeable drift and map misalignment. By contrast, our hybrid point-to-point and point-to-plane alignment, combined with adaptive-planarity classification, successfully exploits limited structural cues to preserve tight trajectory alignment and map consistency, as highlighted in Fig. 3.

Fig. 4 demonstrates our method's tighter ground-truth alignment on the Sejong01 and Riverside02 datasets, successfully mitigating trajectory deviations during curved loops, sharp turns, and loop closures. Absolute pose error (APE) evaluations on the MathildaAVE dataset in Fig. 5 show that our pipeline prevents the gradual drift typical of KISS-SLAM along extended straight-line segments. Finally, Fig. 6 shows the raw APE profiles from the SanJose-Downtown sequence and confirms that HP^2-SLAM yields a significantly lower overall error magnitude. Even during sudden error spikes, our trajectory stabilizes and recovers faster, proving its resilience in environments characterized by strong rotational motion and repetitive structures.

C. Quantitative Results

We first evaluate our method and compare our performance against state-of-the-art pipelines on the MulRan benchmark. As shown in Table I, HP^2-SLAM achieves an average ATE competitive with CT-ICP and PIN-SLAM, while substantially outperforming MULLS and KISS-SLAM, especially on the high-speed Sejong sequence. Highly dynamic downtown areas and featureless riverside routes (DCC and Riverside sequences) remain challenging for purely geometric odometry due to ICP degeneracy. Nevertheless, allowing minor adjustments to the base planarity threshold is sufficient to stabilize the optimization in these structure-scarce environments, entirely avoiding the need for extensive dataset-specific tuning.

We then evaluate our approach on the KITTI dataset to assess global trajectory consistency via the ATE. As reported in Table II, HP^2-SLAM consistently yields a lower average ATE than the KISS-SLAM baseline. Although absolute error reduction is moderate, the proposed hybrid alignment and adaptive-planarity threshold clearly improve robustness across diverse geometric conditions without sacrificing accuracy. We also compare against GenZ-ICP, a front-end-only system that naturally accumulates higher global drift due to the lack of loop closure. Unlike GenZ-ICP, our pipeline is explicitly designed to minimize global trajectory errors while preserving the standard ICP efficiency.

Table III highlights the pipeline's robustness across different sensors and motion diversity. It consistently ranks in the top tier across sequences, particularly with Ouster LiDARs. This indicates that it can handle sparse scans and rapid rotations without sensor-specific overfitting. On the large-scale Apollo dataset in Table IV, our method maintains high accuracy under dense traffic, on wide multi-lane roads, and across cross-day traversals. This stability indicates that the hybrid formulation effectively mitigates the effects of transient dynamic objects and adapts well to appearance changes. Notably, we achieve these results by limiting the base planarity threshold during degenerate sequences, confirming our system's practicality for real-world urban deployments.

D. Ablation Study and Sensitivity Analysis

Table V presents the front-end ablation study on KITTI sequences 00–10. While single geometric constraints are highly environment-dependent (point-to-plane excels in sequences 01, 06, and 09, and point-to-point in sequences 02 and 04), our adaptive hybrid method exhibits superior robustness. It achieves the best and second-best ATE in 7 and 3, respectively, out of 11 sequences. The method also notably reduces sequence 03's ATE to 0.51 m, compared with 3.55 m for the GenZ-ICP baseline. These results confirm that dynamically weighting loss functions based on local geometry effectively mitigates drift across complex trajectories.

We observe in Fig. 7 that varying τ_{ref} over a broad interval results in only minor and smooth changes in ATE. No sharp performance collapse is observed, even away from the optimal setting. This suggests that the adaptive formulation mitigates the strong parameter sensitivity typically associated with fixed planarity thresholds and maintains stable behavior across different environments.

E. Runtime Analysis

We assess the runtime performance of HP^2-SLAM on the KITTI sequence. Our method achieves an average runtime of 19 ms per scan, compared to 13 ms for KISS-SLAM. However, the system still operates above 50 Hz, thereby enabling real-time applications. The runtime breakdown indicates that the proposed front-end is highly computationally efficient, as shown in Table VI. The per-frame processing time remains below the scan period of typical LiDAR sensors, confirming that integrating adaptive-planarity estimation and hybrid alignment preserves real-time performance.

VI. Conclusion

In this paper, we presented HP^2-SLAM, a LiDAR SLAM framework that provides both a minimalist philosophy for real-time applications and a novel adaptive hybrid ICP. Our approach aims to dynamically balance point-to-point and point-to-plane via a neighborhood-size adaptive planarity threshold, achieving stable alignment in both structured and degenerate environments without relying on complex feature engineering. Integrated with loop closure and pose graph optimization, this front-end enables globally consistent mapping over long trajectories. This research demonstrates that HP^2-SLAM outperforms or matches strong baselines across widely used benchmarks. Moreover, HP^2-SLAM achieves notable robustness in degenerate scenarios, allows for cross-sensor generalization without requiring dataset-specific tuning, and provides real-time efficiency on commodity hardware. Future work will explore pairing this adaptive hybrid ICP with complementary sensing modalities and test operation in long-term autonomy scenarios.

REFERENCES

[1] J. Czarnowski, T. Laidlow, R. Clark, and A. J. Davison, "Deepfactors: Real-time probabilistic dense monocular slam," IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 721–728, 2020.

[2] Z. Teed and J. Deng, "Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras," in Proceedings of the Advances in Neural Information Processing Systems, vol. 34, 2021, pp. 16 558–16 569.

[3] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 12 786–12 796.

[4] J. Zhang, S. Singh, et al., "Loam: Lidar odometry and mapping in real-time." in Proceedings of Robotics: Science and Systems, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

[5] A.-L. Contributors, "A-loam: Advanced implementation of loam," https://github.com/HKUST-Aerial-Robotics/A-LOAM, 2018.

[6] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2018, pp. 4758–4765.

[7] 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.

[8] T. Guadagnino, B. Mersch, S. Gupta, I. Vizzo, G. Grisetti, and C. Stachniss, "Kiss-slam: A simple, robust, and accurate 3d lidar slam system with enhanced generalization capabilities," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2025, pp. 5363–5370.

[9] P. Li, R. Wang, Y. Wang, and W. Tao, "Evaluation of the icp algorithm in 3d point cloud registration," IEEE Access, vol. 8, pp. 68 030–68 048, 2020.

[10] K.-L. Low, "Linear least-squares optimization for point-to-plane icp surface registration," Chapel Hill, University of North Carolina, vol. 4, no. 10, pp. 1–3, 2004.

[11] Y. Pan, P. Xiao, Y. He, Z. Shao, and Z. Li, "Mulls: Versatile lidar slam via multi-metric linear least square," in Proceedings of the IEEE International Conference on Robotics and Automation, 2021, pp. 11 633–11 640.

[12] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," in Proceedings of Robotics: Science and Systems, vol. 2, no. 4. Seattle, WA, 2009, p. 435.

[13] D. Lee, H. Lim, and S. Han, "Genz-icp: Generalizable and degeneracy-robust lidar odometry using an adaptive weighting," IEEE Robotics and Automation Letters, vol. 10, no. 1, pp. 152–159, 2024.

[14] P. Dellenbach, J.-E. Deschaud, B. Jacquet, and F. Goulette, "Ct-icp: Real-time elastic lidar odometry with loop closure," in Proceedings of the IEEE International Conference on Robotics and Automation, 2022, pp. 5580–5586.

[15] A. Makovetskii, S. Voronin, V. Kober, and A. Voronin, "A point-to-plane registration algorithm for orthogonal transformations," in Proceedings of the Applications of Digital Image Processing XLI, vol. 10752. SPIE, 2018, pp. 828–836.

[16] X. Wei, J. Lv, J. Sun, E. Dong, and S. Pu, "Gclo: Ground constrained lidar odometry with low-drifts for gps-denied indoor environments," in Proceedings of the IEEE International Conference on Robotics and Automation, 2022, pp. 2229–2235.

[17] J. Hatleskog and K. Alexis, "Probabilistic degeneracy detection for point-to-plane error minimization," IEEE Robotics and Automation Letters, vol. 9, no. 12, pp. 11 234–11 241, 2024.

[18] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," Image and vision computing, vol. 10, no. 3, pp. 145–155, 1992.

[19] D. M. K. Felix, X. Yu, J. Zhang, S. Ha, Z. Zou, and T. Westerlund, "Lidar variability: A novel dataset and comparative study of solid-state and spinning lidars," arXiv preprint arXiv:2507.04321, 2025.

[20] E. Koukoulis, G. Arvanitis, and K. Moustakas, "Unleashing the power of generalized iterative closest point for swift and effective point cloud registration," in Proceedings of the IEEE International Conference on Image Processing, 2024, pp. 3403–3409.

[21] S. Choi, H.-W. Chae, Y. Jeung, S. Kim, K. Cho, and T.-w. Kim, "Fast and versatile feature-based lidar odometry via efficient local quadratic surface approximation," IEEE Robotics and Automation Letters, vol. 8, no. 2, pp. 640–647, 2022.

[22] E. Koukoulis, G. Arvanitis, and K. Moustakas, "Leveraging anisotropic error for robust point cloud registration," in Proceedings of the International Conference on Digital Signal Processing. IEEE, 2025, pp. 1–5.

[23] B. Kim, C. Jung, D. H. Shim, and A.-a. Agha-mohammadi, "Adaptive keyframe generation based lidar inertial odometry for complex underground environments," in Proceedings of the IEEE International Conference on Robotics and Automation, 2023, pp. 3332–3338.

[24] T. Tuna, J. Nubert, Y. Nava, S. Khattak, and M. Hutter, "X-icp: Localizability-aware lidar registration for robust localization in extreme environments," IEEE Transactions on Robotics, vol. 40, pp. 452–471, 2023.

[25] D. Lee, M. Jung, W. Yang, and A. Kim, "Lidar odometry survey: Recent advancements and remaining challenges," Intelligent Service Robotics, vol. 17, no. 2, pp. 95–118, 2024.

[26] S. Gupta, T. Guadagnino, B. Mersch, N. Trekel, M. V. Malladi, and C. Stachniss, "Efficiently closing loops in lidar-based slam using point cloud density maps," arXiv preprint arXiv:2501.07399, 2025.

[27] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2011, pp. 2564–2571.

[28] H. Lim, H. N. Chu, and C. Yim, "Hierarchical binary search tree for packet classification," IEEE Communications Letters, vol. 11, no. 8, pp. 689–691, 2007.

[29] G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based slam," IEEE Intelligent Transportation Systems Magazine, vol. 2, no. 4, pp. 31–43, 2011.

[30] Y. Pan, X. Zhong, L. Jin, L. Wiesmann, M. Popović, J. Behley, and C. Stachniss, "Pings: Gaussian splatting meets distance fields within a point-based implicit neural map," arXiv preprint arXiv:2502.05752, 2025.

[31] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," The International Journal of Robotics Research, vol. 32, no. 11, pp. 1231–1237, 2013.

[32] T. Bakogiannis, I. Giannakopoulos, D. Tsoumakos, and N. Koziris, "Apollo: A dataset profiling and operator modeling system," in Proceedings of the International Conference on Management of Data, 2019, pp. 1869–1872.

[33] G. Kim, Y. S. Park, Y. Cho, J. Jeong, and A. Kim, "Mulran: Multimodal range dataset for urban place recognition," in Proceedings of the IEEE International Conference on Robotics and Automation, 2020, pp. 6246–6253.

[34] M. Jung, W. Yang, D. Lee, H. Gil, G. Kim, and A. Kim, "Helipr: Heterogeneous lidar dataset for inter-lidar place recognition under spatiotemporal variations," The International Journal of Robotics Research, vol. 43, no. 12, pp. 1867–1883, 2024.

[35] M. Grupp, "evo: Python package for the evaluation of odometry and slam," 2017. [Online]. Available: https://github.com/MichaelGrupp/evo