2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

UCA-SLAM: Tightly Coupled Visual-LiDAR SLAM with DoF-wise Uncertainty-driven Constraint Analysis

Shizhuo Yu[1], Wenbin Zhu[1], Jing Yuan[1] *Member, IEEE*, and Yuanxi Gao[1]

***Abstract*— Single sensor (visual or LiDAR) simultaneous localization and mapping (SLAM) is fragile in the complex environment, which makes visual-LiDAR fusion a mainstream in SLAM research. However, most existing fusion methods omit explicit modeling of feature uncertainties and do not quantify each feature's constraint strength on each degree of freedom (DoF) of the 6-DoF pose, thereby hindering the full exploitation of the complementary information across different sensors. In this paper, a tightly coupled visual-LiDAR SLAM method termed UCA-SLAM is proposed, which integrates the closed-form uncertainty propagation and the DoF-wise constraint analysis. Specifically, UCA-SLAM maintains uncertainties for visual map points and LiDAR voxel planes, and computes DoF-wise constraint strength for each feature. In the front-end tracking, the DoF-wise constraints of features are comprehensively analyzed, which provides an adaptive fusion mechanism for pose estimation, and an explicit uncertainty propagation from feature measurements to the 6-DoF pose is derived. The resultant feature and pose uncertainties are then used to weight the cost function in local bundle adjustment (BA) optimization of UCA-SLAM to improve the accuracy of the system. Extensive experiments conducted on public datasets and in real-world environments demonstrate that UCA-SLAM outperforms state-of-the-art visual-LiDAR fusion SLAM methods. UCA-SLAM is open-sourced to benefit the community.**

I. INTRODUCTION

Simultaneous localization and mapping (SLAM) plays a key role in robot navigation, autonomous driving, and augmented reality (AR) [1]. Common SLAM systems include visual SLAM [2] [3] and LiDAR SLAM [4] [5]. However, single sensor SLAM degrades markedly under weak texture, low illumination, or structural degeneracy. Accordingly, visual-LiDAR fusion methods have become a mainstream in recent SLAM research [6] [7] [8] [9].

Existing visual-LiDAR fusion SLAM methods are commonly divided into two categories: depth-aided visual methods and tightly coupled fusion methods. Depth-aided visual methods integrate the LiDAR depth into visual odometry or SLAM to improve the pose estimation accuracy and reduce sensitivity to illumination changes. Depth-aided visual methods comprise two subtypes: feature-based methods and direct methods. Feature-based methods assign the LiDAR depth to the matched visual feature points [6] [7], while the direct methods use the LiDAR depth in photometric tracking [8] [9]. However, depth-aided visual methods typically use only the subset of LiDAR data that can be assigned as the depth to image features, leaving most LiDAR measurements unused.

To fully exploit multi sensor measurements, tightly coupled fusion methods combine visual and LiDAR constraints within a single optimization framework, improving overall accuracy and robustness. For example, TVL-SLAM [10] and GAC-Mapping [11] jointly leverage visual and LiDAR measurements across odometry and loop closure, leading to accurate and reliable pose estimates. Despite these advances, the existing tightly coupled visual-LiDAR methods still lack two key elements: (i) explicit modeling of feature uncertainties from different sensors (e.g., visual points and LiDAR planes) and (ii) constraint strength analysis of each feature on each DoF. Without these, it is difficult to fully exploit cross-sensor and cross-feature complementarity for designing a theoretically grounded adaptive fusion strategy.

In this paper, we propose UCA-SLAM, a tightly coupled monocular visual–LiDAR SLAM framework. It models feature uncertainties and performs DoF-wise, uncertainty-driven constraint analysis. The main contributions are summarized as follows.

1) A DoF-wise, uncertainty-driven adaptive fusion mechanism for visual-LiDAR SLAM is proposed, which explicitly models feature uncertainties and quantifies how strongly each feature constrains the 6-DoF pose. Unlike the geometry analysis that provides a binary verdict for a specific feature along a particular degree of freedom, our constraint analysis yields continuous, soft weights. Based on the estimated uncertainties and constraint strengths, the uncertainty-informed weights are assigned to visual and LiDAR residuals, leading to more stable optimization and improved pose accuracy.
2) A tightly coupled visual-LiDAR SLAM method termed UCA-SLAM is implemented based on the above mechanism. In the front end, LiDAR voxel planes and visual points are fused via the constraint strength, and keyframes are selected using propagated measurement uncertainties. In the back end, local BA performs uncertainty-aware feature fusion, and loop closing jointly optimizes visual and LiDAR constraints in a tightly coupled pose graph, improving overall SLAM accuracy and robustness under diverse conditions.
3) Extensive experiments are performed on public datasets and in real-world scenes. The results demonstrate that

This work was supported in part by the National Key Research and Development Program of China under Grants 2024YFB4708900, in part by the Natural Science Foundation of China under Grant 62473208 and U21A20486 (Corresponding author: Wenbin Zhu).

[1]The authors are with the College of Artificial Intelligence, also with the Tianjin Key Laboratory of Intelligent Robotics, and also with the Engineering Research Center of Trusted Behavior Intelligence, Ministry of Education, Nankai University, Tianjin 300350, China.

The source code of UCA-SLAM and supplementary materials will be made available at: https://github.com/NKU16y/UCA_SLAM_open.

UCA-SLAM outperforms existing systems in both accuracy and robustness. Also, UCA-SLAM will be open sourced to benefit the community.

The rest of this paper is organized as follows. Section II reviews related work. Section III provides an overview of UCA-SLAM. Section IV details the method. Section V reports experimental results. Section VI concludes the paper.

II. Related Work

A. Uncertainty Modeling in SLAM

Accurately modeling and quantifying uncertainties in SLAM enables reliable and consistent information fusion, helping mitigate measurement noise. Prior work has investigated uncertainties at the feature and pose levels. Voxelmap [4] models point measurement noise and pose estimation error to construct a probabilistic map representation. MAC-VO [12] leverages a deep learning model to estimate 2-D keypoint uncertainties and, together with the depth uncertainties, propagates them to 3-D landmark uncertainties. Wang *et al.* [13] estimate uncertainties for 6-DoF object poses and derive confidence regions within a direct method framework. ORB-TEDM [14] explicitly derives uncertainties for camera poses and 3-D point features in the ORB-SLAM2 framework. However, these methods typically focus on a single sensor and a single type of feature. A unified multi-sensor approach that jointly estimates feature and pose uncertainties remains underexplored, especially one with closed-form uncertainty estimation and explicit feature-to-pose propagation.

B. Constraint Analysis in SLAM

Constraint analysis is essential to ensure pose estimation is well constrained along all DoFs. Most existing methods analyze pose constraints mainly through feature geometry. Zhang *et al.* [15] assess degeneracy via the condition number of the approximate Hessian matrix in an optimization framework, while Hinduja *et al.* [16] detect degeneracy by thresholding its eigenvalues. These approaches can indicate underconstrained, but they do not localize it to specific pose DoFs. STING SLAM [17] explicitly characterizes unconstrained DoFs by constructing a plane induced pose constraint subspace. Building on this, Plane-Edge SLAM [18] augments planes with edge points to complement plane degenerate directions and enable seamless plane-edge fusion. Nevertheless, geometry-based analyses often provide a binary verdict, *i.e.*, whether a specific feature constrains a particular DoF, without capturing constraint quality or strength across DoFs. In contrast, an uncertainty-driven constraint analysis assigns continuous, quality-aware constraint strengths, offering a principled basis for tightly coupled multi-sensor fusion.

C. Tightly coupled Visual-LiDAR SLAM

Tightly coupled visual-LiDAR SLAM fuses visual and LiDAR constraints within a single optimization framework to enhance accuracy and robustness. A key challenge is how to balance the heterogeneous residuals across the entire SLAM system. However, most existing methods seldom address this issue explicitly. TVL-SLAM [10] and GAC-Mapping[11] treat visual reprojection residuals and LiDAR geometric residuals equally, with no modality specific or adaptive balancing. HVL-SLAM [19] models residual influence at the feature type level using the inverse variance weights and an eigenvalue-based LiDAR quality term, but does not propagate per feature uncertainties to the pose or analyze per DoF contributions. Shu *et al.* [20] apply depth uncertainty informed weights to visual residuals while leaving LiDAR residuals unadjusted. In summary, designing a fusion mechanism that jointly accounts for feature uncertainties and DoF-wise constraint strength is still insufficiently addressed in current methods.

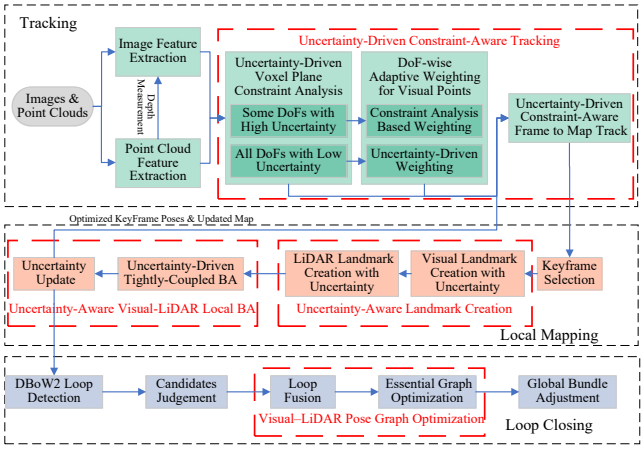

Fig. 1. The framework of UCA-SLAM.

III. System Overview

The overall system framework of UCA-SLAM is illustrated in Fig. 1. UCA-SLAM comprises three key modules: tracking, local mapping and loop closing. A monocular camera and a 3-D LiDAR provide images and point clouds as input. In tracking, we perform a DoF-wise constraint analysis using the uncertainties of visual map points and LiDAR voxel planes, producing a per-feature constraint strength. Guided by these strengths, we fuse visual and LiDAR residuals in real time and solve a unified optimization problem to estimate the current pose. The pose covariance is also obtained to quantify uncertainty along each DoF. In local mapping, we construct visual landmarks (map points) and LiDAR landmarks (voxel planes) while explicitly modeling their uncertainties. These per-feature uncertainties are then used to weight residuals in local bundle adjustment. In loop closing, we run a visual–LiDAR fusion pose-graph optimization to correct drift and improve global consistency.

To facilitate the presentation of the proposed method, the notations used in this paper are defined as follows. The world frame $\{W\}$ is denoted by ${}^{W}(\cdot)$, the camera frame $\{C\}$ is denoted by ${}^{C}(\cdot)$, and the LiDAR frame $\{L\}$ is denoted by ${}^{L}(\cdot)$. In our system, the LiDAR-camera extrinsic parameters are known. The transformation from $\{W\}$ to $\{C\}$, *i.e.*, the current camera pose, is represented as ${}^{C}_{W}\boldsymbol{\xi}$, and the corresponding transformation matrix is given by

${}^{C}_{W}\mathbf{T} = \exp({}^{C}_{W}\boldsymbol{\xi}^{\wedge})$, where $\exp(\cdot)$ is the exponential map that relates the Lie algebra $\mathfrak{se}(3)$ to its Lie group $\mathrm{SE}(3)$. For convenience, we omit the superscripts and subscripts in the notation, and use $\boldsymbol{\xi}$ to represent the current pose in the following sections. Notations with hat $\hat{(\cdot)}$ stand for estimates, vectors with bar $\bar{(\cdot)}$ stand for homogeneous coordinates, and without bar $(\cdot)$ stand for inhomogeneous coordinates.

IV. Method

A. Uncertainties of Visual Point and LiDAR Plane Features

Following standard depth-aided methods [6] [7], we lift 2-D image features to 3-D using the LiDAR depth. Since both the visual points (via depth-aided) and LiDAR voxel planes (via local points fitting) derive from LiDAR points, their uncertainties originate from LiDAR measurement noises. Therefore, we first model the LiDAR point uncertainties.

The uncertainty of a LiDAR point ${}^{L}\boldsymbol{Q}_i$ in $\{L\}$ consists of the ranging uncertainty and the bearing direction uncertainty [4]. Let ${}^{L}\boldsymbol{\omega}_i \in \mathbb{S}^2$ be the bearing direction, $\boldsymbol{\delta}_{{}^{L}\hat{\boldsymbol{\omega}}_i} \sim \mathcal{N}(\mathbf{0}_{2\times1}, \Sigma_{{}^{L}\hat{\boldsymbol{\omega}}_i})$ be its noise, ${}^{L}d_i$ be the depth measurement and $\delta_{{}^{L}\hat{d}_i} \sim \mathcal{N}(0, \sigma^2_{{}^{L}\hat{d}_i})$ be its noise. The uncertainty $\boldsymbol{\delta}_{{}^{L}\hat{\boldsymbol{Q}}_i}$ on ${}^{L}\boldsymbol{Q}_i$ is formulated by $\boldsymbol{\delta}_{{}^{L}\hat{\boldsymbol{Q}}_i} \sim \mathcal{N}\Big(\mathbf{0}_{3\times1}, \boldsymbol{\Sigma}_{{}^{L}\hat{\boldsymbol{Q}}_i}\Big)$, and $\boldsymbol{\Sigma}_{{}^{L}\hat{\boldsymbol{Q}}_i}$ is calculated by

$$\boldsymbol{\Sigma}_{{}^{L}\hat{\boldsymbol{Q}}_i} = {}^{L}\mathbf{A}_i \begin{bmatrix} \sigma^2_{{}^{L}\hat{d}_i} & \mathbf{0}_{1\times2} \\ \mathbf{0}_{2\times1} & \Sigma_{{}^{L}\hat{\boldsymbol{\omega}}_i} \end{bmatrix} {}^{L}\mathbf{A}_i^{T}, \tag{1}$$

where the detailed derivation of ${}^{L}\mathbf{A}_i$ is given in [4] and [21]. The coordinates ${}^{C}\boldsymbol{Q}_i$ of ${}^{L}\boldsymbol{Q}_i$ in $\{C\}$ are given by

$${}^{C}\boldsymbol{Q}_i = {}^{C}_{L}\mathbf{R}\,{}^{L}\boldsymbol{Q}_i + {}^{C}_{L}\boldsymbol{t}. \tag{2}$$

With the given extrinsic parameters between $\{C\}$ and $\{L\}$ (*i.e.,* ${}^{C}_{L}\mathbf{T}$), the uncertainty $\boldsymbol{\delta}_{{}^{C}\hat{\boldsymbol{Q}}_i}$ of ${}^{C}\boldsymbol{Q}_i$ is formulated by $\boldsymbol{\delta}_{{}^{C}\hat{\boldsymbol{Q}}_i} \sim \mathcal{N}\Big(\mathbf{0}_{3\times1}, \boldsymbol{\Sigma}_{{}^{C}\hat{\boldsymbol{Q}}_i}\Big)$, and $\boldsymbol{\Sigma}_{{}^{C}\hat{\boldsymbol{Q}}_i}$ is calculated by

$$\boldsymbol{\Sigma}_{{}^{C}\hat{\boldsymbol{Q}}_i} = {}^{C}_{L}\mathbf{R}\boldsymbol{\Sigma}_{{}^{L}\hat{\boldsymbol{Q}}_i}{}^{C}_{L}\mathbf{R}^{T}. \tag{3}$$

As with ${}^{C}\boldsymbol{Q}_i$ and ${}^{L}\boldsymbol{Q}_i$, all subsequent uncertain variables are modeled as zero-mean Gaussian perturbations. For brevity, in what follows we quantify uncertainty using the covariance matrix. For a 3-D visual point feature ${}^{C}\boldsymbol{P}_i$ associated with ${}^{C}\boldsymbol{Q}_i$, according to the camera projection model, it can be computed by

$${}^{C}\boldsymbol{P}_i = {}^{C}\, d_i\mathbf{K}^{-1}\bar{\boldsymbol{p}}_i, \tag{4}$$

where $\mathbf{K}$ is the camera intrinsic matrix. The uncertainty $\boldsymbol{\Sigma}_{{}^{C}\hat{\boldsymbol{P}}_i}$ of ${}^{C}\hat{\boldsymbol{P}}_i$ arises from the noises of the pixel $\hat{\boldsymbol{p}}_i$ and the depth ${}^{C}\hat{d}_i$, which is formulated by

$$\boldsymbol{\Sigma}_{{}^{C}\hat{\boldsymbol{P}}_i} = \frac{\partial {}^{C}\boldsymbol{P}_i}{\partial \boldsymbol{p}_i}\boldsymbol{\Sigma}_{\hat{\boldsymbol{p}}_i}\left(\frac{\partial {}^{C}\boldsymbol{P}_i}{\partial \boldsymbol{p}_i}\right)^{T} + \frac{\partial {}^{C}\boldsymbol{P}_i}{\partial {}^{C}d_i}\boldsymbol{\Sigma}_{{}^{C}\hat{d}_i}\left(\frac{\partial {}^{C}\boldsymbol{P}_i}{\partial {}^{C}d_i}\right)^{T}, \tag{5}$$

where $\Sigma_{{}^{C}\hat{d}_i}$ can be obtained from $\boldsymbol{\Sigma}_{{}^{C}\hat{\boldsymbol{Q}}_i}$.

Furthermore, the uncertainty of ${}^{C}\hat{\boldsymbol{P}}_i$ is propagated to that of ${}^{W}\hat{\boldsymbol{P}}_i$. ${}^{W}\boldsymbol{P}_i$ is computed by

$${}^{W}\boldsymbol{P}_i = {}^{W}_{C}\mathbf{R}\,{}^{C}\boldsymbol{P}_i + {}^{W}_{C}\boldsymbol{t}. \tag{6}$$

The uncertainty of ${}^{W}\hat{\boldsymbol{P}}_i$ is affected by both ${}^{C}\hat{\boldsymbol{P}}_i$ and the transform from $\{C\}$ to $\{W\}$, which is computed by

$$\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{P}}_i} = {}^{W}_{C}\mathbf{R}\boldsymbol{\Sigma}_{{}^{C}\hat{\boldsymbol{P}}_i}\left({}^{W}_{C}\mathbf{R}\right)^{T} + \frac{\partial {}^{W}\boldsymbol{P}_i}{\partial {}^{W}_{C}\boldsymbol{\xi}}\boldsymbol{\Sigma}_{{}^{W}_{C}\hat{\boldsymbol{\xi}}}\left(\frac{\partial {}^{W}\boldsymbol{P}_i}{\partial {}^{W}_{C}\boldsymbol{\xi}}\right)^{T}, \tag{7}$$

where ${}^{W}_{C}\boldsymbol{\xi}$ denotes the transformation from $\{C\}$ to $\{W\}$. The uncertainty of ${}^{W}_{C}\hat{\boldsymbol{\xi}}$ is computed by

$$\boldsymbol{\Sigma}_{{}^{W}_{C}\hat{\boldsymbol{\xi}}} = \mathrm{Ad}_{{}^{W}_{C}\mathbf{T}}\,\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}}\,\mathrm{Ad}^{T}_{{}^{W}_{C}\mathbf{T}}, \tag{8}$$

where Ad represents the adjoint map [22]. The calculation of the pose uncertainty $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}}$ will be provided in Section IV-C.

Similarly, the uncertainty of ${}^{W}\boldsymbol{Q}_i$, *i.e.*, $\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{Q}}_i}$ can be calculated by (7). Assume that a LiDAR voxel plane feature ${}^{W}\boldsymbol{\pi}_j$ consists of a group of LiDAR points ${}^{W}\boldsymbol{Q}_{j_k}(k = 1, \cdots, N_j)$, each with an uncertainty $\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{Q}}_{j_k}}$. ${}^{W}\boldsymbol{\pi}_j$ is denoted by ${}^{W}\boldsymbol{\pi}_j = [{}^{W}\boldsymbol{n}_j^{T}, {}^{W}\tilde{\boldsymbol{Q}}_j^{T}]^{T}$, where ${}^{W}\boldsymbol{n}_j$ is the normal and ${}^{W}\tilde{\boldsymbol{Q}}_j = \frac{1}{N_j}\sum_{k=1}^{N_j}{}^{W}\boldsymbol{Q}_{j_k}$ denotes a point on ${}^{W}\boldsymbol{\pi}_j$. ${}^{W}\boldsymbol{\pi}_j$ can be denoted as a function h of ${}^{W}\boldsymbol{Q}_{j_k}$, which is given by

$${}^{W}\boldsymbol{\pi}_j = h\left({}^{W}\boldsymbol{Q}_{j1}, {}^{W}\boldsymbol{Q}_{j2}, \cdots, {}^{W}\boldsymbol{Q}_{j\,N_j}\right), \tag{9}$$

and the uncertainty of ${}^{W}\boldsymbol{\pi}_j$ is computed by

$$\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{\pi}}_j} = \sum_{k=1}^{N_j}\frac{\partial h}{\partial {}^{W}\boldsymbol{Q}_{j_k}}\,\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{Q}}_{j_k}}\,\left(\frac{\partial h}{\partial {}^{W}\boldsymbol{Q}_{j_k}}\right)^{T}, \tag{10}$$

where $\frac{\partial h}{\partial {}^{W}\boldsymbol{Q}_{j_k}} = [\frac{\partial {}^{W}\boldsymbol{n}_j}{\partial {}^{W}\boldsymbol{Q}_{j_k}}, \frac{\partial {}^{W}\tilde{\boldsymbol{Q}}_j}{\partial {}^{W}\boldsymbol{Q}_{j_k}}]^{T}$, the detailed derivation is given in [4].

B. DoF-wise Uncertainty-driven Constraint Analysis

For a visual point feature ${}^{W}\boldsymbol{P}_i$, the reprojection error is defined by

$$\boldsymbol{e}_{p_i} = \bar{\boldsymbol{p}}_i - \frac{1}{{}^{C}d_i}\mathbf{K}\exp(\boldsymbol{\xi}^{\wedge})\,{}^{W}\bar{\boldsymbol{P}}_i, \quad {}^{C}d_i = \left(\exp(\boldsymbol{\xi}^{\wedge})\,{}^{W}\bar{\boldsymbol{P}}_i\right)_z. \tag{11}$$

For a LiDAR voxel plane feature ${}^{W}\boldsymbol{\pi}_j$, the residual is defined by

$$e_{\,\pi_j} = \sum_{k=1}^{N_j}{}^{W}\boldsymbol{n}_j{}^{T}({}^{W}\boldsymbol{Q}_{j_k} - {}^{W}\tilde{\boldsymbol{Q}}_j), \tag{12}$$

where

$${}^{W}\bar{\boldsymbol{Q}}_{j_k} = \exp(\boldsymbol{\xi}^{\wedge})^{-1}\,{}^{C}\bar{\boldsymbol{Q}}_{j_k}, \qquad {}^{W}\boldsymbol{Q}_{j_k} = \left({}^{W}\bar{\boldsymbol{Q}}_{j_k}\right)_{1:3}. \tag{13}$$

We define the tracking cost function as $F(\boldsymbol{\xi})$. Its explicit form is given in Section IV-C. With the covariance propagation and implicit differentiation, the pose uncertainties induced by the visual point ${}^{W}\mathbf{P}_i$ and the LiDAR plane ${}^{W}\boldsymbol{\pi}_j$ are computed by

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_{\,P_i}} = \mathbf{H}_1^{-1}(\mathbf{H}_{2i}^{T}\boldsymbol{\Sigma}_{\hat{\boldsymbol{p}}_i}\mathbf{H}_{2i} + \mathbf{H}_{3i}^{T}\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{P}}_i}\mathbf{H}_{3i})\mathbf{H}_1^{-T} \tag{14}$$

and

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_{\,\pi_j}} = \mathbf{H}_1^{-1}(\mathbf{H}_{4j}^{T}\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{\pi}}_j}\mathbf{H}_{4j} + \sum_{k=1}^{N_j}\mathbf{H}_{5j_k}^{T}\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{Q}}_{j_k}}\mathbf{H}_{5j_k})\mathbf{H}_1^{-T} \tag{15}$$

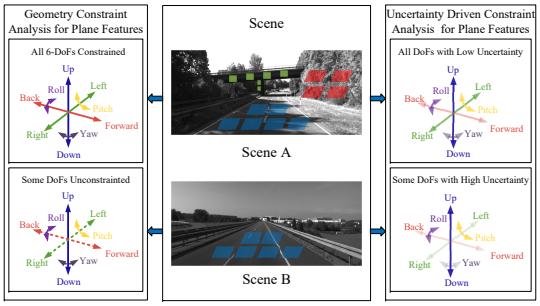

Fig. 2. The difference between two constraint analysis methods.

respectively, where $\mathbf{H}_1 = \frac{\partial F^2(\boldsymbol{\xi})}{\partial \boldsymbol{\xi} \partial \boldsymbol{\xi}^T}$, $\mathbf{H}_{2i} = \frac{\partial F^2(\boldsymbol{\xi})}{\partial \boldsymbol{\xi} \partial \boldsymbol{p}_i^T}$, $\mathbf{H}_{3i} = \frac{\partial F^2(\boldsymbol{\xi})}{\partial \boldsymbol{\xi} \partial {}^W\boldsymbol{P}_i^T}$, $\mathbf{H}_{4j} = \frac{\partial F^2(\boldsymbol{\xi})}{\partial \boldsymbol{\xi} \partial {}^W\boldsymbol{\pi}_j^T}$, and $\mathbf{H}_{5j_k} = \frac{\partial F^2(\boldsymbol{\xi})}{\partial \boldsymbol{\xi} \partial {}^C\boldsymbol{Q}_{j_k}^T}$. The lth diagonal entry of $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_{P_i}}$ and $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_{\pi_j}}$ $(l = 1, 2, \cdots, 6)$ denote the pose uncertainty contributed by ${}^W\mathbf{P}_i$ and ${}^W\boldsymbol{\pi}_j$ on the lth DoF, respectively.

Compared with cameras, LiDAR typically provides more accurate geometric constraints for pose estimation. Therefore, we first analyze the constraints from LiDAR measurements. Let N_π be the number of associated planes. The pose uncertainty induced by LiDAR measurements $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_\pi}$ is given by Eq. (15).

By inspecting the diagonal entries of $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_\pi}$, we quantify the LiDAR-induced pose uncertainty for each DoF, while larger diagonal values imply higher uncertainty, *i.e.*, weaker constraints along the corresponding DoF, yielding a per-DoF uncertainty-driven constraint analysis.

Unlike geometry-based constraint analysis, which provides only a binary verdict for a feature along a given DoF and cannot reflect constraint quality, our method outputs continuous per-DoF constraint strengths for each feature. As shown in Fig. 2, two representative autonomous-driving scenes illustrate the difference between conventional geometry-based analysis and our uncertainty-driven analysis. Dashed arrows denote unconstrained DoFs, while higher transparency indicates lower constraint strength. In Scene A, voxel planes align with three dominant surfaces, including the road (blue), the front face of the overpass (green), and an approximately planar shrub region (red). A purely geometry-based analysis would mark all six DoFs as constrained. However, the overpass lies far above the sensor, amplifying LiDAR measurement noise, while the shrub region shows weak coplanarity, resulting in high plane-fitting uncertainty. Our method converts these effects into per-DoF pose uncertainty, where directions supported by the road exhibit low uncertainty (low transparency), whereas the others remain highly uncertain (high transparency). In Scene B, voxel planes concentrate on the road, where geometry-based analysis reports three constrained DoFs (solid) and three unconstrained (dashed). Consistently, our uncertainty-driven analysis yields low uncertainty for the constrained DoFs and much higher uncertainty for the rest. Hence, beyond detecting under-constrained cases, our approach quantifies per-DoF constraint strength and guides the incorporation of complementary visual cues. This analysis further motivates adaptive fusion of LiDAR and visual residuals, as detailed in Section IV-C.

To further verify that our analysis agrees with geometry-based constraint analysis, we evaluate the proposed uncertainty-driven analysis on KITTI sequence 01. An overview of the scene is shown in Fig. 3, where voxel planes are predominantly extracted from the ground. According to the geometry-based analysis in [18], this configuration is nearly degenerate in the rotation ϕ_y about the y-axis and the translations ρ_x and ρ_z along the x- and z-axes. We compute the pose uncertainty under (i) LiDAR-only constraints and (ii) joint LiDAR–visual constraints. The results are reported in Table I. As expected, the LiDAR-only uncertainties are substantially higher along the LiDAR-degenerate DoFs. After incorporating visual constraints, the uncertainties on these DoFs drop markedly. These results indicate that the proposed uncertainty-driven constraint analysis explicitly highlights the DoFs that geometry-based analysis identifies as (nearly) unconstrained, where the corresponding uncertainties become pronounced. Moreover, for the DoFs that are considered constrained by geometry-based analysis, our uncertainty measure further quantifies the constraint strength, enabling DoF-wise adaptive fusion of visual and LiDAR residuals.

Fig. 3. Degeneration case in KITTI sequence 01

TABLE I
PER DOF UNCERTAINTY OF THE CURRENT POSE.

	LiDAR constraints	LiDAR+Visual constraints
ϕ_x(rad)	0.0132	0.0124
ϕ_y(rad)	**0.2189**	**0.0133**
ϕ_z(rad)	0.0525	0.0521
ρ_x(m)	**0.2770**	**0.0483**
ρ_y(m)	0.0551	0.0431
ρ_z(m)	**0.8782**	**0.4102**

C. Front-end Tracking

In front-end tracking, each residual in Eqs. (11) and (12) is weighted by the uncertainty and the constraint analysis in Section IV-B. The cost function is formulated as

$$F(\boldsymbol{\xi}) = f(\boldsymbol{\xi}) + g(\boldsymbol{\xi}). \tag{16}$$

Specifically, the two terms correspond to point and plane features, respectively:

$$f(\boldsymbol{\xi}) = \sum_{i=1}^{N_p} \rho_p\left(w_{p_i}\, \boldsymbol{e}_{p_i}^T \boldsymbol{e}_{p_i}\right) \tag{17}$$

and

$$g(\boldsymbol{\xi}) = \sum_{j=1}^{N_\pi} \rho_\pi \Big(w_{\pi_j} \, e_{\pi_j}^2 \Big), \tag{18}$$

where $\boldsymbol{e}_{p_i}$ is defined in (11) and e_{π_j} is defined in (12). N_p and N_π denote the numbers of associated point and plane features, respectively. $\rho_p(\cdot)$ and $\rho_\pi(\cdot)$ are the Huber loss functions for point and plane features, respectively, and w_{p_i} and w_{π_j} are the weights assigned to the point feature ${}^W\boldsymbol{P}_i$ and the plane feature ${}^W\boldsymbol{\pi}_j$.

Based on the analysis in Section IV-B, pose estimation is primarily constrained by LiDAR measurements, with visual measurements playing a supporting role in front-end tracking. Therefore, LiDAR residuals are weighted by their measurement uncertainties, while weights for visual residuals are determined by the per-DoF constraint analysis. Specifically, the weight w_{π_j} is computed as

$$w_{\pi_j} = \mathrm{tr}\Big(\boldsymbol{\Sigma}_{\hat{e}_{\pi_j}}^{-1} ({}^W\boldsymbol{\pi}_j) \Big), \tag{19}$$

where $\mathrm{tr}(\cdot)$ denotes the trace of a matrix and $\boldsymbol{\Sigma}_{\hat{e}_{\pi_j}}({}^W\boldsymbol{\pi}_j)$ is the uncertainty of $\hat{e}_{\pi_j}$ induced by ${}^W\boldsymbol{\pi}_j$.

For numerical stability and consistency, all weights are further normalized to $[0, 1]$ within each frame

$$\tilde{w}_{\pi_j} = \frac{w_{\pi_j} - w_\pi^{\min}}{w_\pi^{\max} - w_\pi^{\min} + \epsilon}, \tag{20}$$

where $w_\pi^{\min}$ and $w_\pi^{\max}$ are the minimum and maximum values of $\{w_{\pi_j}\}$, and ϵ is a small constant.

For the point feature ${}^W\boldsymbol{P}_i$, we assign a higher weight when it offers strong constraints along DoFs that are weakly constrained by LiDAR. Specifically, first, eigendecomposition is performed on $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_\pi}$, *i.e.,*

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_\pi} = \mathbf{E}_\pi \, \mathrm{diag}(\boldsymbol{\lambda}_\pi) \, \mathbf{E}_\pi^T, \tag{21}$$

where $\mathbf{E}_\pi = [\boldsymbol{e}_{\pi 1}, \cdots, \boldsymbol{e}_{\pi 6}]$ and $\boldsymbol{\lambda}_\pi = [\lambda_{\pi 1}, \cdots, \lambda_{\pi 6}]^T$. Next, $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_{P_i}}$ is projected onto the principal directions of $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_\pi}$

$$\lambda_{p_i l} = \boldsymbol{e}_{\pi l}^T \, \boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}_{P_i}} \, \boldsymbol{e}_{\pi l}, \quad l = 1, 2, \cdots, 6. \tag{22}$$

Then, the uncertainties are converted to directional information by

$$I_{\pi l} = 1/\lambda_{\pi l}, I_{p_i l} = 1/\lambda_{p_i l}, \tag{23}$$

and a directional fusion ratio r_l is defined by

$$r_l = \frac{I_{p_i l}}{I_{p_i l} + I_{\pi l}} \in (0, 1), \tag{24}$$

which increases when the point contributes more information along LiDAR weak directions. Finally, w_{p_i} is formulated as the normalized average of these ratios

$$w_{p_i} = \frac{1}{6} \sum_{l=1}^{6} r_l. \tag{25}$$

When the pose exhibits low uncertainty across all DoFs under LiDAR constraints, w_{p_i} is primarily governed by the pose uncertainty induced by ${}^W\boldsymbol{P}_i$.

Based on the covariance propagation and implicit differentiation, the pose uncertainty after the optimization in front-end tracking can be computed by

$$\begin{aligned} \boldsymbol{\Sigma}_{\hat{\boldsymbol{\xi}}} = \mathbf{H}_1^{-1} (& \sum_{i=1}^{N_p} w_{p_i} (\mathbf{H}_{2i}^T \boldsymbol{\Sigma}_{\hat{\boldsymbol{p}}_i} \mathbf{H}_{2i} + \mathbf{H}_{3i}^T \boldsymbol{\Sigma}_{W\hat{\boldsymbol{P}}_i} \mathbf{H}_{3i}) \\ & + \sum_{j=1}^{N_\pi} w_{\pi_j} (\mathbf{H}_{4j}^T \boldsymbol{\Sigma}_{W\hat{\boldsymbol{\pi}}_j} \mathbf{H}_{4j} + \sum_{k=1}^{N_j} \mathbf{H}_{5j_k}^T \boldsymbol{\Sigma}_{W\hat{\boldsymbol{Q}}_{j_k}} \mathbf{H}_{5j_k})) \mathbf{H}_1^{-T} \end{aligned} \tag{26}$$

D. Local Mapping

In local mapping, keyframe poses and their associated features are refined to improve localization accuracy and map quality. A hybrid keyframe selection strategy that leverages visual cues, LiDAR geometry and pose uncertainty is adopted. For the visual component, we follow ORB-SLAM2 [2] and apply lightweight temporal spacing together with a tracking-quality gate to avoid oversampling and to filter out unreliable visual measurements. In addition, UCA-SLAM monitors LiDAR geometric cues and the estimated pose uncertainty. A new keyframe is inserted when either (i) novel voxel-plane normals are detected and validated via a D-optimality information-gain test, or (ii) the maximum uncertainty among the six DoFs exceeds a predefined threshold. Together, these criteria yield a robust and information-efficient keyframe selection policy.

To support uncertainty-aware fusion of visual point and LiDAR plane features while accounting for pose uncertainties, the local BA is formulated a weighted least-squares problem over visual and LiDAR residuals. Since Tracking has already exploited the complementarity between visual and LiDAR constraints, local BA assigns weights directly from the corresponding measurement uncertainties. The cost function is given by

$$F_{BA} = \sum_{k \in \mathbb{K}} \Big(f_{BA}(\boldsymbol{\xi}_k, \{{}^W\boldsymbol{P}_i\}) + g_{BA}(\boldsymbol{\xi}_k) \Big). \tag{27}$$

where $\mathbb{K}$ denotes the set of keyframes involved in the local BA, $\{{}^W\boldsymbol{P}_i\}$ denotes the set of point features involved in the local BA. For point features, the residuals are weighted according to their uncertainties, *i.e.*,

$$f_{BA}\left(\boldsymbol{\xi}_k, \{{}^W\boldsymbol{P}_i\}\right) = \sum_{i=1}^{N_{pk}} w_{p_{ki}} \boldsymbol{e}_{p_{ki}}{}^T \boldsymbol{e}_{p_{ki}}, \tag{28}$$

where $\boldsymbol{e}_{p_{ki}}$ denotes the reprojection error of ${}^W\boldsymbol{P}_i$ in the kth keyframe, and N_{pk} is the number of point features associated with the kth keyframe. The weight is computed by

$$w_{p_{ki}} = \mathrm{tr}\left(\left[\boldsymbol{\Sigma}_{\hat{\boldsymbol{e}}_{p_{ki}}}(\boldsymbol{\xi}_k) + \boldsymbol{\Sigma}_{\hat{\boldsymbol{e}}_{p_{ki}}}({}^W\boldsymbol{P}_i) \right]^{-1} \right), \tag{29}$$

where $\boldsymbol{\Sigma}_{\hat{\boldsymbol{e}}_{p_{ki}}}(\boldsymbol{\xi}_k)$ and $\boldsymbol{\Sigma}_{\hat{\boldsymbol{e}}_{p_{ki}}}({}^W\boldsymbol{P}_i)$ denote the uncertainty of $\hat{\boldsymbol{e}}_{p_{ki}}$ induced by $\boldsymbol{\xi}_k$ and ${}^W\boldsymbol{P}_i$, respectively.

For LiDAR planes, the residuals are weighted by

$$g_{BA}(\boldsymbol{\xi}_k) = \sum_{j=1}^{N_{\pi k}} w_{\pi_{kj}} \, e_{\pi_{kj}}^2, \tag{30}$$

where $N_{\pi k}$ denotes the number of plane features associated with the kth keyframe, and $e_{\pi_{kj}}$ is the scalar residual induced by ${}^{W}\boldsymbol{\pi}_j$ in the kth keyframe. The detailed derivation of $e_{\pi_{kj}}$ can be found in [23]. The weight $w_{\pi_{kj}}$ is computed by

$$w_{\pi_{kj}} = \mathrm{tr}\left(\left[\boldsymbol{\Sigma}_{\hat{e}_{\pi_{kj}}}(\boldsymbol{\xi}_k) + \boldsymbol{\Sigma}_{\hat{e}_{\pi_{kj}}}({}^{W}\boldsymbol{\pi}_j)\right]^{-1}\right), \tag{31}$$

where $\boldsymbol{\Sigma}_{\hat{e}_{\pi_{kj}}}(\boldsymbol{\xi}_k)$ and $\boldsymbol{\Sigma}_{\hat{e}_{\pi_{kj}}}({}^{W}\boldsymbol{\pi}_j)$ denote the uncertainty of $\hat{e}_{\pi_{kj}}$ induced by $\boldsymbol{\xi}_k$ and ${}^{W}\boldsymbol{\pi}_j$, respectively. For numerical stability and consistent scaling, the point weights $\{w_{p_{ki}}\}$ and the plane weights $\{w_{\pi_{kj}}\}$ are each normalized to $[0, 1]$ as (20) within the local BA window.

After local BA, the uncertainties of poses and map features are updated. For the kth keyframe, the pose uncertainty is updated by

$$\begin{aligned}\boldsymbol{\Sigma}'_{\hat{\boldsymbol{\xi}}_k} =& \mathbf{H}_6^{-1}\sum_{i=1}^{N_{pk}}\left(\mathbf{H}_{7ki}^{T}\,\boldsymbol{\Sigma}_{\hat{\boldsymbol{p}}_{ki}}\,\mathbf{H}_{7ki} + \mathbf{H}_{8ki}^{T}\,\boldsymbol{\Sigma}_{{}^{W}\hat{\boldsymbol{P}}_i}\,\mathbf{H}_{8ki}\right)\mathbf{H}_6^{-T} \\ &+ \mathbf{H}_6^{-1}\sum_{j=1}^{N_{\pi k}}\mathbf{H}_{9kj}^{T}\,\boldsymbol{\Sigma}_{\hat{\mathbf{C}}_{jk}}\,\mathbf{H}_{9kj}\,\mathbf{H}_6^{-T},\end{aligned} \tag{32}$$

where $\boldsymbol{p}_{ki}$ denotes the pixel coordinates of ${}^{W}\boldsymbol{P}_i$ in the kth keyframe, and $\mathbf{C}_{jk}$ denotes the point cluster of ${}^{W}\boldsymbol{\pi}_j$ in the kth keyframe (see [23] for details). Here, $\mathbf{H}_6 = \frac{\partial^2 F_{BA}(k)}{\partial \boldsymbol{\xi}_k \partial \boldsymbol{\xi}_k^T}$, $\mathbf{H}_{7ki} = \frac{\partial^2 F_{BA}(k)}{\partial \boldsymbol{\xi}_k \partial \boldsymbol{p}_{ki}^T}$, $\mathbf{H}_{8ki} = \frac{\partial^2 F_{BA}(k)}{\partial \boldsymbol{\xi}_k \partial {}^{W}\boldsymbol{P}_i^T}$, $\mathbf{H}_{9kj} = \frac{\partial^2 F_{BA}(k)}{\partial \boldsymbol{\xi}_k \partial \mathbf{C}_{jk}^T}$, and $F_{BA}(k)$ denotes the kth term of F_{BA}.

For the map point ${}^{W}\boldsymbol{P}_i$, the corresponding terms are

$$\bar{f}({}^{W}\boldsymbol{P}_i) = \sum_{k\in\mathbb{K}} w_{p_{ki}}\,\boldsymbol{e}_{p_{ki}}^{T}\boldsymbol{e}_{p_{ki}}, \tag{33}$$

and the updated uncertainty of ${}^{W}\boldsymbol{P}_i$ is computed by

$$\boldsymbol{\Sigma}'_{{}^{W}\hat{\boldsymbol{P}}_i} = \mathbf{H}_{10}^{-1}\sum_{k\in\mathbb{K}}\left(\mathbf{H}_{11k}^{T}\boldsymbol{\Sigma}_{\hat{\boldsymbol{p}}_{ki}}\mathbf{H}_{11k} + \mathbf{H}_{12k}^{T}\boldsymbol{\Sigma}'_{\hat{\boldsymbol{\xi}}_k}\mathbf{H}_{12k}\right)\mathbf{H}_{10}^{-T}, \tag{34}$$

where $\mathbf{H}_{10} = \frac{\partial^2 \bar{f}({}^{W}\boldsymbol{P}_i)}{\partial {}^{W}\boldsymbol{P}_i \partial {}^{W}\boldsymbol{P}_i^T}$, $\mathbf{H}_{11k} = \frac{\partial^2 \bar{f}({}^{W}\boldsymbol{P}_i)}{\partial {}^{W}\boldsymbol{P}_i \partial \boldsymbol{p}_{ki}^T}$, and $\mathbf{H}_{12k} = \frac{\partial^2 \bar{f}({}^{W}\boldsymbol{P}_i)}{\partial {}^{W}\boldsymbol{P}_i \partial \boldsymbol{\xi}_k^T}$. The uncertainties of LiDAR plane features are updated following [23].

E. Loop Closing

To mitigate the inevitable odometry drift over long trajectories, we adopt the loop-closing pipeline of ORB-SLAM2 and augment it with LiDAR measurements. Loop candidates are retrieved using DBoW2 [24]. Once a candidate is verified, the relative pose between the current keyframe and the loop keyframe is estimated via a unified point-plane ICP that fuses visual point correspondences with LiDAR plane correspondences. The resulting $\mathrm{SE}(3)$ constraint is then inserted into the pose-graph optimization to reduce drift and improve global consistency. Finally, a global BA is performed to further refine keyframe poses and visual points, following ORB-SLAM2 [2].

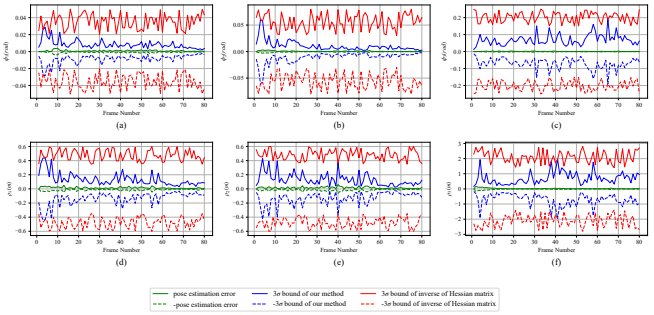

Fig. 4. Uncertainty on the poses in the first 80 frames on KITTI 00 sequence. (a) ϕ_1. (b) ϕ_2. (c) ϕ_3. (d) ρ_1. (e) ρ_2. (f) ρ_3.

V. Experiments and Results

In this section, extensive experiments are conducted on public datasets and in real-world environments to evaluate the accuracy and robustness of UCA-SLAM. We first validate the accuracy of our uncertainty estimation method. Then, we evaluate UCA-SLAM on the KITTI dataset [25], comparing both odometry performance and full-SLAM performance against several state-of-the-art systems. Finally, UCA-SLAM is tested on a mobile-robot dataset collected in two real-world outdoor scenes, with comparisons to LOAM [5] and voxel-map [4]. All experiments are conducted on an industrial PC equipped with an AMD Ryzen 5 4600H CPU (3.00 GHz), 16 GB RAM, running Ubuntu 18.04.

A. Evaluation on Uncertainty Estimation

The proposed uncertainty estimation method is evaluated on the KITTI dataset using the 3σ bounds. Our results are compared with the widely used Hessian-inverse approximation [18], [26], [27]. Fig. 4 shows the results on KITTI sequence 00, where the 3σ bounds of the pose covariance estimated by the two methods are shown in blue (ours) and red (Hessian-inverse), respectively. For both rotations ϕ_i $(i = 1, 2, 3)$ and translations ρ_i $(i = 1, 2, 3)$, all estimates produced by our method fall within the 3σ bounds. By contrast, although the Hessian-inverse bounds also cover the estimates, they are overly conservative and do not faithfully reflect the system's actual uncertainty. Overall, our 3σ bounds are tighter and more informative, demonstrating the advantage of the proposed method.

B. Evaluation on Odometry and SLAM

The KITTI odometry benchmark [25] is a widely used standard for evaluating visual, LiDAR, and visual–LiDAR SLAM. It provides long driving sequences with accurate ground-truth trajectories across diverse scenes for odometry and SLAM assessment.

To thoroughly evaluate UCA-SLAM, we conduct two sets of experiments. First, we evaluate odometry on five sequences, comparing UCA-odometry with seven state-of-the-art odometry methods and performing ablation studies to isolate the contributions of key components. Second, we evaluate full-SLAM on the remaining six sequences, benchmarking UCA-SLAM against seven representative SLAM

TABLE II
COMPARISON OF DIFFERENT ODOMETRY IN TERMS OF $t_{\mathrm{rel}}(\%)$ / r_{rel} (DEG/100 M)

Sequence	UCA-SLAM(-)	UCA-SLAM(-) (w/o adaptive in Tracking)	UCA-SLAM(-) (w/o adaptive in Local BA)	HVL-SLAM(-)	LAMV-SLAM(-)	GAC-Mapping(-)	DV-LOAM(-)	CamVox(-)	DEMO	DVL-SLAM(-)
KITTI 01	1.23/**0.39**	1.44/0.42	1.42/0.41	1.74/**0.39**	**1.02**/–	15.22/0.87	1.74/0.43	–	1.87/–	1.47/–
KITTI 03	**0.65/0.22**	0.79/0.27	0.82/0.30	0.81/0.39	1.01/–	1.94/1.10	0.85/0.48	0.84/0.31	0.99/–	0.92/–
KITTI 04	**0.30/0.30**	0.38/0.36	0.42/0.38	0.34/0.32	0.67/–	2.74/1.04	1.39/0.42	0.70/0.71	1.23/–	0.67/–
KITTI 07	**0.51**/0.33	0.55/0.36	0.58/0.42	0.53/**0.29**	0.84/–	0.88/0.78	**0.51**/0.33	0.61/0.38	1.16/–	1.26/–
KITTI 10	**0.55/0.21**	0.62/0.33	0.73/0.36	0.70/0.28	**0.55**/–	3.14/1.37	0.87/0.47	0.76/0.37	1.14/–	0.70/–
avg	**0.65/0.29**	0.76/0.35	0.79/0.37	0.84/0.34	0.82/–	4.58/0.83	1.17/0.43	0.73/0.44	1.28/–	1.01/–

Best results are highlighted in **bold**, (-) indicates that loop closing is disabled and "–" denotes that this result is not provided in the original paper.

systems. Together, these experiments assess accuracy and robustness in both odometry and SLAM settings.

In the odometry experiments, UCA-odometry is compared against seven visual-LiDAR fusion odometry methods including HVL-SLAM [19], LAMV-SLAM [29], GAC-Mapping [11], DV-LOAM [30], CamVox [7], DEMO [31], and DVL-SLAM [8]. All methods are run with loop closing disabled. Performance is measured by the average relative translation error t_{rel} and rotation error r_{rel} following [25]. Results are reported in Table II. We also conduct ablations by disabling the adaptive weighting in tracking and in local BA, respectively.

Overall, UCA-odometry achieves the best localization accuracy on four of five sequences, demonstrating the effectiveness of the proposed DoF-wise uncertainty-driven feature fusion. On sequence 01 (highway driving), strong illumination changes degrade visual matching. Consequently, UCA-odometry yields a slightly higher t_{rel} than LAMV-SLAM [29], which mitigates illumination effects via an online photometric calibration thread.

In the SLAM experiments, UCA-SLAM is compared with seven representative visual-LiDAR fusion SLAM systems, including HVL-SLAM [19], LAMV-SLAM [29], the method in [28], GAC-Mapping [11], DV-LOAM [30], CamVox [7], and RGB-L [32]. Evaluation follows the KITTI odometry benchmark, using the average relative translation error t_{rel} and rotation error r_{rel} as in the odometry experiments. Results are reported in Table III.

Results show that UCA-SLAM achieves the best localization accuracy on all six sequences and the lowest rotation error on five of the six sequences. These results demonstrate the effectiveness of the proposed uncertainty-driven adaptive fusion strategy, which enables UCA-SLAM to maintain high accuracy across diverse environments.

To demonstrate the real-time capability of UCA-SLAM, we report the computational cost of the tracking and keyframe creation over the six KITTI sequences in Table IV. The results show that the average front-end time remains within 100 ms, which satisfies the real-time requirements.

C. Experiments in Real-World Environments

To evaluate the accuracy and robustness of the proposed method in real-world environments, it is deployed as a visual-LiDAR fusion odometry system on a Husky robot equipped with a custom sensor suite consisting of a Velodyne 16-beam LiDAR and a Kinect V2 camera. Experiments are conducted in two challenging outdoor environments, where the robot follows loop trajectories and returns to the starting point.

We compare UCA-odometry against LOAM [5] and voxel-map [4] in terms of start-to-end drift. Both methods are executed using their official open-source implementations and are run with identical parameter settings for a fair comparison. The quantitative results are reported in Table V, with the trajectory length shown in parentheses. On both sequences, the plane-based LiDAR odometry voxel-map fails to produce a valid trajectory, whereas UCA-odometry attains lower start-to-end drift than LOAM, a point-based LiDAR odometry based on ICP.

It is worth noting that our experimental setup is particularly challenging for plane-based LiDAR SLAM for two reasons: (i) we use a 16-beam LiDAR, which produces sparse point clouds and makes reliable plane extraction difficult, and (ii) the scenes are predominantly open fields, where planar structures are largely limited to the ground, resulting in weak geometric constraints and reduced pose observability. Consequently, the plane-based LiDAR odometry voxel-map fails to produce a valid trajectory on both sequences. By explicitly modeling feature uncertainties and performing a DoF-wise constraint analysis, UCA-odometry tightly fuses visual features and LiDAR measurements, achieving robust localization and improved accuracy under these challenging conditions.

VI. CONCLUSIONS

This paper presented UCA-SLAM, a tightly coupled visual-LiDAR SLAM system that integrates closed-form uncertainty propagation with the DoF-wise constraint analysis. By explicitly modeling feature uncertainties and quantifying constraint strength per DoF, UCA-SLAM enables uncertainty-driven adaptive fusion, improving robustness and accuracy across diverse environments. Experiments on KITTI and real-world outdoor datasets demonstrated clear advantages over existing methods. Moreover, the proposed analysis provides a principled foundation for future extensions, such as integrating thermal imaging for low-visibility navigation, supporting long-term autonomous navigation under changing environments, and enabling cross-modal high-precision mapping.

REFERENCES

[1] W. Zhu, J. Yuan, X. Zhang, and F. Chen, "Bridging the gap between semantics and geometry in SLAM: A semantic–geometric tight-coupling monocular visual object SLAM system," *IEEE Transactions on Robotics*, vol. 41, pp. 3078–3098, 2025.

[2] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[3] M. B. Azhari and D. H. Shim, "DINO-VO: A feature-based visual odometry leveraging a visual foundation model," *IEEE Robotics and Automation Letters*, vol. 10, no. 9, pp. 9152–9159, 2025.

TABLE III

COMPARISON OF DIFFERENT SLAM METHODS IN TERMS OF $t_{\mathrm{rel}}(\%)$ / r_{rel} (DEG/100 M)

Sequence	UCA-SLAM	HVL-SLAM	LAMV-SLAM	[28]	GAC-Mapping	DV-LOAM	CamVox	RGB-L
KITTI 00	**0.62/0.19**	0.64/0.22	0.83/–	0.99/–	1.68/0.78	0.66/0.30	0.77/0.35	0.66/0.28
KITTI 02	**0.78/0.22**	0.83/0.21	0.97/–	1.38/–	1.98/0.77	0.84/0.36	0.89/0.34	0.84/0.31
KITTI 05	**0.36/0.16**	**0.36/0.16**	0.57/–	0.72/–	1.13/0.52	0.54/0.30	0.59/0.34	0.44/0.21
KITTI 06	**0.30/0.15**	0.34/0.16	0.49/–	0.61/–	1.19/0.58	0.70/0.33	0.89/0.48	0.79/0.44
KITTI 08	**0.92**/0.30	1.07/0.34	1.19/–	1.27/–	1.69/0.87	0.93/**0.27**	2.24/0.83	1.20/0.37
KITTI 09	**0.68/0.22**	0.72/0.24	0.80/–	1.06/–	1.56/0.80	0.75/0.27	0.79/0.37	0.79/0.27
avg	**0.61/0.21**	0.66/0.22	0.81/–	1.01/–	1.54/0.71	0.74/0.31	1.19/0.47	0.79/0.31

Best results are highlighted in **bold** and "–" denotes that this result is not provided in the original paper.

TABLE IV

AVERAGE TIME USED IN TRACKING AND KEFRAME CREATION

Sequence	00	01	03	05	07	10
Tracking Time(ms)	51.54	53.25	51.31	50.42	51.39	52.83
keyframe Creation Time(ms)	21.58	23.42	24.12	26.13	24.73	22.35
Total Time(ms)	73.12	76.67	75.43	76.55	76.12	75.18

TABLE V

START-TO-END DRIFT IN THE REAL-WORLD EXPERIMENTS

Dataset(Length)	UCA-odometry	LOAM	voxel-map
Park(434.19m)	8.314m(1.9%)	16.767m(2.7%)	–
Forest(302.47m)	15.214m(5.0%)	31.783m(10.5%)	–

[4] C. Yuan, W. Xu, X. Liu, X. Hong, and F. Zhang, "Efficient and probabilistic adaptive voxel mapping for accurate online LiDAR odometry," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8518–8525, 2022.

[5] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robotics: Science and Systems (RSS)*, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:18612391

[6] J. Graeter, A. Wilczynski, and M. Lauer, "LIMO: LiDAR-monocular visual odometry," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2018, pp. 7872–7879.

[7] Y. Zhu, C. Zheng, C. Yuan, X. Huang, and X. Hong, "CamVox: A low-cost and accurate LiDAR-assisted visual SLAM system," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2021, pp. 5049–5055.

[8] Y.-S. Shin, Y. S. Park, and A. Kim, "Direct visual SLAM using sparse depth for camera–LiDAR system," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2018, pp. 5144–5151.

[9] K. Huang, J. Xiao, and C. Stachniss, "Accurate direct visual–laser odometry with explicit occlusion handling and plane detection," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2019, pp. 1295–1301.

[10] C.-C. Chou and C.-F. Chou, "Efficient and accurate tightly coupled visual–LiDAR SLAM," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 509–14 523, 2022.

[11] J. He, Y. Zhou, L. Huang, Y. Kong, and H. Cheng, "Ground and aerial collaborative mapping in urban environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 1, pp. 95–102, 2021.

[12] Y. Qiu, Y. Chen, Z. Zhang, W. Wang, and S. Scherer, "MAC-VO: Metrics-aware covariance for learning-based stereo visual odometry," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2025, pp. 3803–3814.

[13] J. Wang, Z. Li, Z. Wang, B. Guan, Y. Shang, and Q. Yu, "Deterministic object pose confidence region estimation," in *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, 2025, pp. 14 866–14 875.

[14] J. Yuan, S. Zhu, K. Tang, and Q. Sun, "ORB-TEDM: An RGB-D SLAM approach fusing ORB triangulation estimates and depth measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–15, 2022.

[15] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2016, pp. 809–816.

[16] A. Hinduja, B.-J. Ho, and M. Kaess, "Degeneracy-aware factors with applications to underwater SLAM," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2019, pp. 1293–1299.

[17] Q. Sun, J. Yuan, X. Zhang, and F. Sun, "RGB-D SLAM in indoor environments with STING-based plane feature extraction," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 3, pp. 1071–1082, 2018.

[18] Q. Sun, J. Yuan, X. Zhang, and F. Duan, "Plane-edge-SLAM: Seamless fusion of planes and edges for SLAM in indoor environments," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 4, pp. 2061–2075, 2021.

[19] 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.

[20] C. Shu and Y. Luo, "Multi-modal feature constraint based tightly coupled monocular visual–LiDAR odometry and mapping," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 5, pp. 3384–3393, 2023.

[21] C. Yuan, X. Liu, X. Hong, and F. Zhang, "Pixel-level extrinsic self-calibration of high-resolution LiDAR and camera in targetless environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7517–7524, 2021.

[22] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017.

[23] Z. Liu, X. Liu, and F. Zhang, "Efficient and consistent bundle adjustment on LiDAR point clouds," *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4366–4386, 2023.

[24] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[25] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3354–3361.

[26] R. Gomez-Ojeda, F.-A. Moreno, D. Zuniga-Noël, D. Scaramuzza, and J. Gonzalez-Jimenez, "PL-SLAM: A stereo SLAM system through the combination of points and line segments," *IEEE Transactions on Robotics*, vol. 35, no. 3, pp. 734–746, 2019.

[27] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proc. European Conf. on Computer Vision (ECCV)*. Springer, 2014, pp. 834–849.

[28] S.-S. Huang, Z.-Y. Ma, T.-J. Mu, H. Fu, and S.-M. Hu, "LiDAR-monocular visual odometry using point and line features," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2020, pp. 1091–1097.

[29] J. Yin, D. Luo, F. Yan, and Y. Zhuang, "A novel LiDAR-assisted monocular visual SLAM framework for mobile robots in outdoor environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–11, 2022.

[30] W. Wang, J. Liu, C. Wang, B. Luo, and C. Zhang, "DV-LOAM: Direct visual LiDAR odometry and mapping," *Remote Sensing*, vol. 13, no. 16, p. 3340, 2021.

[31] J. Zhang, M. Kaess, and S. Singh, "A real-time method for depth enhanced visual odometry," *Autonomous Robots*, vol. 41, no. 1, pp. 31–43, 2017.

[32] F. Sauerbeck, B. Obermeier, M. Rudolph, and J. Betz, "RGB-L: Enhancing indirect visual SLAM using LiDAR-based dense depth maps," in *Proc. Int. Conf. on Computer, Control and Robotics (ICCCR)*, 2023, pp. 95–100.