Fault-tolerant cross-modal pose realignment for heterogeneous air–ground localization

Zhuozhu Jian[1,†], Qixuan Li[1,†], An Yi[2], Shengtao Zheng[1], Lizhi Sun[4], Bin Zhang[3], Ziyi Zhou[3], Xueqian Wang[1], and Xinlei Chen[5,*]

Abstract Air–ground localization becomes brittle when visual–inertial drift or an abrupt state jump places a camera pose outside the attraction basin of local camera–LiDAR registration. We address this recovery problem with a coarse-to-fine cross-modal pose-realignment framework. Visual features are back-projected and evaluated against local LiDAR planes through point-to-plane residuals, avoiding learned appearance correspondences at run time. An association-quality check determines whether the propagated pose is sufficiently supported by local geometry. When the support is insufficient, a bounded particle swarm optimization (PSO) search selects a coarse pose hypothesis, which is subsequently refined by a visual–inertial–LiDAR factor graph. The proposed framework therefore couples implicit geometric association, bounded basin exploration, and local multimodal refinement to recover from large pose errors. Experiments use the EuRoC sequences with a static Leica scan and real indoor and outdoor air–ground platforms with UGV-provided maps. In the reported severe cases, the method reduces translational RMSE by 33.8% under a 0.5 m initial offset and reduces post-disturbance mean absolute position error by 55.2% after a 30° mid-sequence yaw jump, relative to the best successful baseline in each test. Indoor tests with one UGV and three UAVs further demonstrate 20 Hz operation in a shared LiDAR map frame.

Keywords Air–ground localization; cross-modal association; fault tolerance; map-assisted localization; point-to-plane constraints

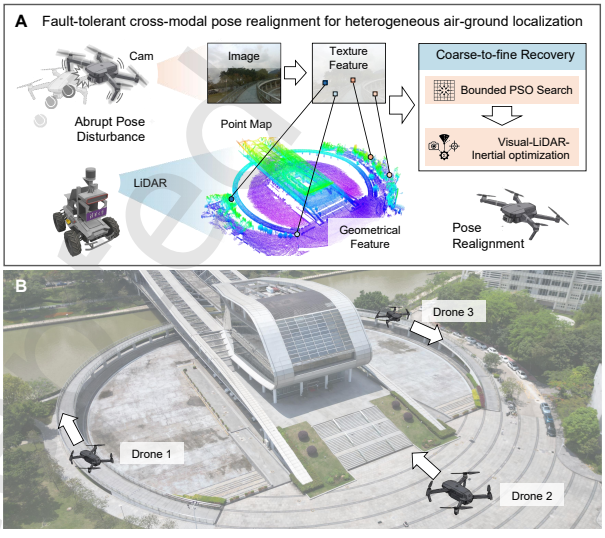


Fig. 1 LiDAR–vision localization in a shared air–ground map frame. A A LiDAR mounted on a UGV incrementally builds a dense point-cloud map. Multiple UAVs collect images and inertial measurements, and each UAV registers its observations to the shared map. The UAV estimators do not use direct inter-UAV measurements. **B** Outdoor multi-UAV experiments. The proposed method maintains accurate localization under camera jitter and feature degradation.

1 Introduction

Relative localization provides a common spatial reference for multi-robot perception [1–3], planning [4–6], and task execution [7, 8]. Multi-sensor fusion supports anchor-assisted ultra-wideband (UWB)–IMU positioning [9] and transport-mapping-based multi-radar tracking [10]. Heterogeneous air–ground systems are particularly challenging because cameras and LiDARs observe different physical quantities [11, 12]: a monocular camera measures appearance and bearing without direct metric depth, whereas LiDAR measures sparse metric geometry. Consequently, an image feature does not have an immediately available counterpart in a point cloud, especially when overlap is small or the relative pose is inaccurate.

Visual–inertial odometry (VIO) can accumulate gradual drift and can also experience abrupt state discontinuities after aggressive motion, occlusion, or tracking loss. When such an error becomes large, the current camera pose may leave the attraction basin of a local camera–LiDAR optimizer. The pose error then corrupts data association, while the corrupted

[1]Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China. E-mail: {jzz24, lqx23, st_zheng24}@mails.tsinghua.edu.cn; wang.xq@sz.tsinghua.edu.cn.
[2]School of Control Science and Engineering, Dalian University of Technology, Dalian 116024, China. E-mail: 1243394086@mail.dlut.edu.cn.
[3]Shenzhen Smart City Technology Development Group Co., Ltd., Shenzhen, China. E-mail: {zhangbin, zhouziyi}@smartcitysz.com.
[4]Tsinghua University, Beijing 100084, China. E-mail: slz23@mails.tsinghua.edu.cn.
[5]Shenzhen International Graduate School, Tsinghua University, Shenzhen, China; Pengcheng Laboratory, Shenzhen, China; and RISC-V International Open Source Laboratory, Shenzhen, China. E-mail: chen.xinlei@sz.tsinghua.edu.cn.
*Corresponding author: Xinlei Chen.
†Zhuozhu Jian and Qixuan Li contributed equally to this work.

JUST Accepted

associations further bias the optimizer. Recovery therefore requires two capabilities at the same time: an association model that remains informative under imperfect initialization, and a mechanism that can provide a new local-optimization seed when the current estimate is no longer reliable.

We propose a coarse-to-fine pose-realignment framework for camera–LiDAR localization under large pose uncertainty. First, visual bearing vectors with estimated depth are projected into the LiDAR frame and evaluated against local planar geometry. This produces implicit point-to-plane associations without requiring learned image–point descriptors at run time. Second, an association-quality check determines whether the propagated pose can directly initialize local refinement or requires a bounded PSO search. PSO acts as a stochastic basin selector that explores the pose region and supplies a coarse hypothesis for subsequent nonlinear least-squares (NLS) refinement.

At the system level, a UGV supplies either a pre-built or incrementally constructed LiDAR map, and each UAV independently estimates its state in that shared map frame. We refer to this as heterogeneous air–ground map-assisted localization. The shared map and coordinate frame enable spatial consistency across agents, but the formulation does not contain direct inter-UAV measurements or cross-UAV factors.

The contributions are summarized as follows:

(1) A geometry-based cross-modal pose-realignment formulation that evaluates back-projected visual features against local LiDAR planes. The resulting point-to-plane residuals provide a shared geometric residual model for coarse hypothesis evaluation and local factor-graph refinement.

(2) A coarse-to-fine pose-recovery cascade that combines bounded PSO-based basin selection with visual–inertial–LiDAR nonlinear refinement. The coarse stage enlarges the effective capture range of the local estimator under large initialization errors and abrupt state changes.

(3) Validation on EuRoC and real air–ground platforms, including a controlled ablation of the basin-entry stage, initial and mid-sequence pose disturbances, and three-UAV localization in a UGV-supplied map.

2 Related works

2.1 Visual–inertial and LiDAR–inertial estimation

Visual–inertial estimators [13] such as VINS-Mono [14] and OKVIS [15] combine visual reprojection factors with IMU pre-integration in a sliding-window or factor-graph formulation [16]. These methods provide accurate high-rate motion estimates, but monocular scale error, accumulated drift, and tracking loss can produce a poor initial pose for later map registration.

LiDAR–inertial odometry (LIO) systems, including FAST-LIO/FAST-LIO2 [17, 18], LeGO-LOAM [19], and LIO-SAM [20], commonly use point-to-plane geometry in either filtering or optimization. Unified LiDAR–visual–inertial estimators also use planar residuals to couple modalities [21–23]. These estimators are most effective when initialized within a suitable local attraction basin, leaving basin re-entry after large state errors as a critical unresolved problem. Our framework addresses this problem through geometry-based coarse pose realignment followed by local visual–inertial–LiDAR refinement.

2.2 Image-to-point-cloud association

Image-to-point-cloud registration must bridge dense appearance measurements and sparse 3-D geometry. Learning-based methods construct shared features or predict correspondences. DeepI2P [24] formulates frustum classification, P2-Net [25] learns pixel–point descriptors, and CoFiI2P [26] and 2D3D-MATR [27] use coarse-to-fine or transformer-based matching. RayI2P [28] and GraphI2P [29] further learn ray or graph representations. These methods can provide broad capture ranges, but their deployment cost and cross-domain behavior depend on the trained model and data distribution.

Geometry-based alternatives retain an explicit measurement model. Examples include uncertainty-aware PnP [30], pixel-level LiDAR–camera calibration [31], structural localization [32], Colmap-PCD [33], and graph-based registration to a prior map [34]. Such methods are interpretable and do not require task-specific feature training, but their local solvers remain sensitive to association quality and initialization.

Our method uses a shared point-to-plane residual model for coarse hypothesis evaluation and local camera-to-map refinement. The valid-association ratio assesses whether the propagated pose has sufficient geometric support for direct local refinement.

2.3 Coarse-to-fine pose recovery

Cross-modal pose estimation is non-convex when correspondences change with the pose. Local NLS estimators, including multi-sensor factor graphs such as R3LIVE [22] and Multi-LVI-SAM [23], are efficient near a suitable initialization but can converge to an incorrect local solution after a large state jump. RANSAC-based correspondence methods such as CorrI2P [35], probabilistic BEV alignment such as U-ViLAR [36], and coarse-to-fine learned registration such as

Just Accepted

CMRNext [37] address different aspects of the capture-range problem.

Building on the complementary capture ranges of global sampling and local refinement, our framework uses bounded PSO to generate a coarse pose hypothesis when the propagated initialization is unreliable. The selected hypothesis returns the estimator to a suitable attraction basin, after which the visual–inertial–LiDAR factor graph performs accurate local refinement.

3 Overview of the framework

3.1 Problem statement and notation

Let I_{cur} be the current image, $\mathcal{I} = \{I_i\}_{i=1}^{N_I}$ the image history, and $\mathcal{L} = \{\boldsymbol{l}_j\}_{j=1}^{N_L}$ the LiDAR map, where $\boldsymbol{l}_j \in \mathbb{R}^3$ is expressed in the LiDAR world frame L.

Problem statement: Given synchronized camera–IMU measurements and a LiDAR map supplied by a static scan or a mapping UGV, estimate the initial camera-to-map pose $\boldsymbol{T}_{LC_0} \in SE(3)$ and the sliding-window camera poses $\mathcal{T}_{LC} = \{\boldsymbol{T}_{LC_i}\}_{i=1}^{N_I}$. When the propagated pose is no longer geometrically consistent with the map, the estimator must recover a suitable coarse alignment before local refinement.

We use L for the LiDAR world frame, C_k for the camera frame at time k, and I_k for the corresponding IMU frame. The LiDAR world frame is initialized by LIO and is typically aligned with its first LiDAR frame. Throughout the paper, $\boldsymbol{T}_{AB} = (\boldsymbol{R}_{AB}, \boldsymbol{t}_{AB})$ maps coordinates from frame B to frame A:

$$\boldsymbol{p}^A = \boldsymbol{R}_{AB}\boldsymbol{p}^B + \boldsymbol{t}_{AB}. \tag{1}$$

A point expressed in L is denoted by $\boldsymbol{p}^L$, and the same point expressed in a camera frame is denoted by $\boldsymbol{p}^{C_k}$. After intrinsic calibration, a visual feature is represented by a unit bearing vector $\boldsymbol{x}^{C_k}$ on the normalized image plane. The coordinate frames are illustrated in Fig. 3.

3.2 System framework

As illustrated in Fig. 2, the framework contains three stages. First, point-to-plane association provides a geometric camera–LiDAR consistency score (Sec. 4.1). Second, the score determines whether the propagated pose can proceed directly to local refinement or requires bounded PSO sampling to obtain a coarse hypothesis (Sec. 4.2). Third, a local visual–inertial–LiDAR factor graph refines the state (Sec. 4.3). The optimization model remains local; large-error recovery is achieved through the additional coarse-realignment stage.

4 System design

4.1 Geometry-based point-to-plane association

In practical operation, aggressive motion, occlusion, or tracking loss can produce a pose discontinuity before a local estimator can recover. The resulting camera-to-map error can invalidate cross-modal associations and move NLS outside a useful attraction basin.

The association stage contains (i) projection of 2-D visual features (Sec. 4.1.1) and (ii) construction and gating of point-to-plane residuals (Sec. 4.1.2).

4.1.1 Visual feature projection

For the current camera frame C, a visual feature is represented by a unit bearing vector $\boldsymbol{x}^C$ on the normalized image plane, together with its inverse depth λ. The corresponding 3D point in the camera frame is obtained as

$$\boldsymbol{p}^C = \frac{1}{\lambda}\boldsymbol{x}^C. \tag{2}$$

Given the relative pose $\boldsymbol{T}_{LC} = (\boldsymbol{R}_{LC}, \boldsymbol{t}_{LC})$, the point is transformed into the LiDAR world frame as

$$\boldsymbol{p}^L = \mathbf{R}_{LC}\boldsymbol{p}^C + \boldsymbol{t}_{LC}. \tag{3}$$

4.1.2 Point-to-plane residuals

For each projected visual point, a KD-tree retrieves neighboring LiDAR points. A candidate association is accepted only if those neighbors pass the local planarity and distance gates; visual points without a valid local plane are excluded. We use the Hesse normal form [21], in which a plane is represented by a unit normal $\hat{\boldsymbol{n}} \in \mathbb{R}^3$ and offset d:

$$\boldsymbol{P} = \left\{ \langle \hat{\mathbf{n}}, d \rangle \in \mathbb{R}^4 \mid \hat{\mathbf{n}} \cdot \boldsymbol{x} + d = 0 \right\} \tag{4}$$

Given an accepted plane $\boldsymbol{P}_l$, the signed point-to-plane residual is

$$r_l = \hat{\boldsymbol{n}}_l \cdot \boldsymbol{p}_l^L + d_l \tag{5}$$

The coarse-alignment objective is the mean absolute residual over the N_v valid associations:

$$\rho(\boldsymbol{T}_{LC}) = \frac{1}{N_v}\sum_{l=1}^{N_v} |r_l|. \tag{6}$$

For local refinement, a pose perturbation is applied on $SE(3)$:

$$\boldsymbol{T} = \boldsymbol{T}_{LC} \oplus \delta\boldsymbol{\xi}, \quad \delta\boldsymbol{\xi} \in \mathfrak{se}(3), \tag{7}$$

where $\oplus$ denotes the pose update operator.

4.1.3 Robustness of point-to-plane association

To evaluate the robustness of point-to-plane association, we analyze its ability to support stable pose recovery under perturbations and texture interference.

We consider three representative geometric association methods: point-to-plane (P2Plane), point-to-line (P2Line),

Just Accepted

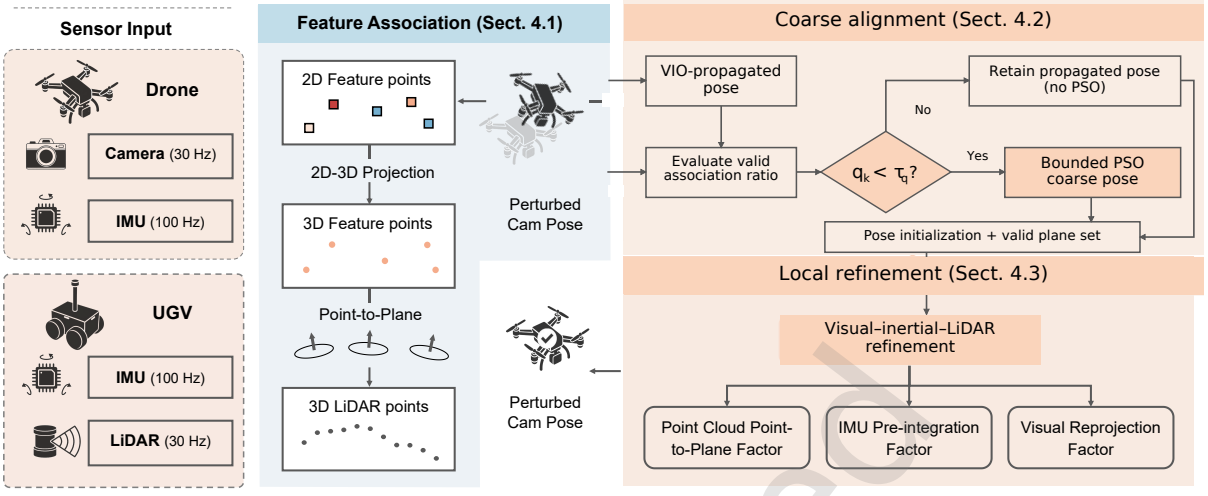


Fig. 2 **Coarse-to-fine camera–LiDAR pose-realignment framework.** Point-to-plane association supplies the geometric support ratio q_k. When $q_k < \tau_q$, bounded PSO generates a coarse pose hypothesis; otherwise, the propagated pose is retained. The selected pose and valid plane associations initialize the same local visual–inertial–LiDAR refinement, using point-to-plane, IMU pre-integration, and reprojection factors.

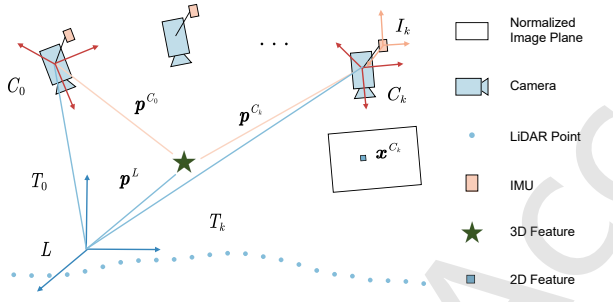


Fig. 3 An illustration of the notations and coordinate transformations.

and point-to-corner (P2Corner). As shown in Fig. 4(A), P2Line and P2Corner rely on explicit matching between structural or discrete features, requiring correspondences between 2D image features and 3D map features. In practical environments, these correspondences become increasingly unreliable under texture interference and pose perturbations, leading to incorrect associations.

In contrast, point-to-plane association based on continuous geometric constraints produces a smoother and better-conditioned curvature structure. As shown in Fig. 4(B), we evaluate the Hessian of the objective function with respect to yaw perturbations and translational perturbations along the x-axis. P2Plane exhibits a continuous and nearly unimodal curvature distribution, whereas P2Line and P2Corner present significant local variations and discontinuities, indicating non-smooth optimization behavior.

To further quantify robustness under texture interference, we evaluate the success rate (SR) of PSO-based pose recovery under different perturbation levels. As shown in Fig. 4(C), the proportion of texture-induced features is gradually increased to simulate increasing perceptual ambiguity. The results show that the success rates of P2Line and P2Corner decrease rapidly as texture interference increases. In contrast, P2Plane consistently maintains the highest success rate across all perturbation levels and exhibits the slowest performance degradation. These results indicate that point-to-plane association is less sensitive to correspondence ambiguity and maintains more stable optimization behavior under perturbations, thereby improving fault tolerance and pose recovery capability during abrupt state changes.

P2Plane and PSO address complementary parts of recovery. P2Plane evaluates a projected visual point against a local surface rather than requiring an exact match to a discrete corner or line feature. Under the tested perturbations, this association model remains more informative for evaluating candidate poses, as illustrated in Fig. 4. PSO uses the resulting geometric score to explore a bounded pose region when the propagated estimate is insufficiently supported. The selected hypothesis then initializes local visual–inertial–LiDAR refinement. P2Plane supplies the geometric evaluation model, while PSO supplies the coarse search.

4.2 Sampling-based coarse alignment

At estimation step k, the propagated camera pose is assessed using the valid-association ratio q_k defined in Eq. (10). If $q_k \geqslant \tau_q$, the propagated pose directly initializes local refinement; if $q_k < \tau_q$, bounded PSO is invoked to obtain a coarse pose hypothesis. Both routes use the same subsequent visual–inertial–LiDAR refinement stage.

Just Accepted

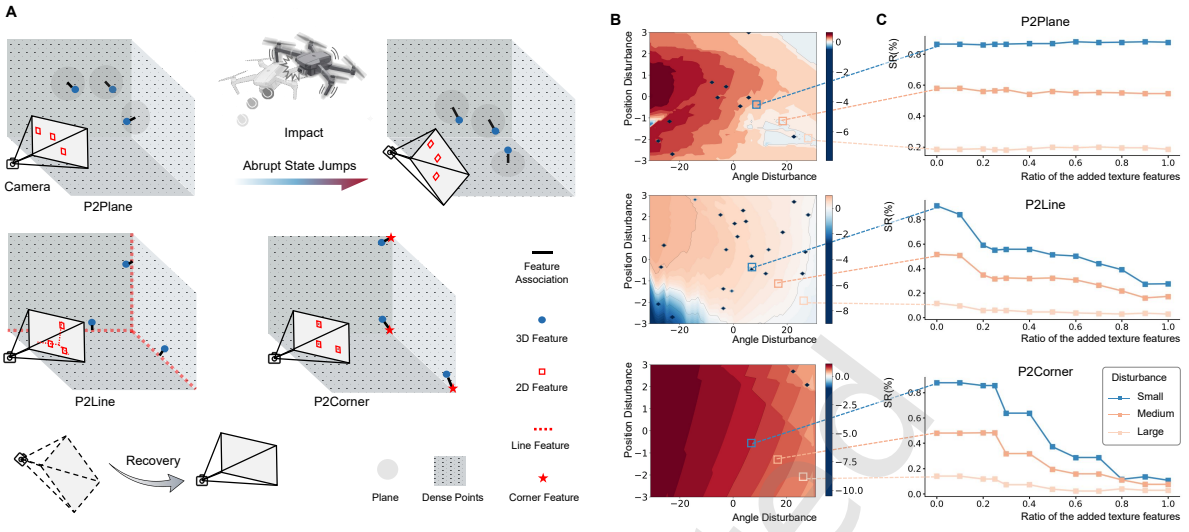


Fig. 4 Robustness analysis of point-primitive association and convergence behavior under perturbations. A Illustration of the association model. Image features are back-projected into sparse 3D points, which are associated with geometric primitives including planes, line segments, and corners (P2Plane, P2Line, P2Corner), enabling pose recovery under perturbations such as impact-induced disturbances. **B** Local curvature analysis. Heatmaps visualize the Hessian properties of the objective function under different pose perturbations, where the color intensity indicates the difficulty of optimization convergence for different methods. **C** Quantitative evaluation of attraction basin. Convergence performance is measured by the success rate of PSO. Experiments are conducted under varying perturbation scales, where texture-based feature interference is introduced to simulate perceptual ambiguity, enabling a comparative evaluation of robustness across methods.

The procedure consists of two steps: Initial Pose Propagation from VIO (Subsec.4.2.1), followed by PSO-based search over the induced uncertainty region (Subsec.4.2.2).

4.2.1 *Initial pose propagation*

Following the convention in Eq. (1), VIO supplies $\boldsymbol{T}_{C_{k-1}C_k}$, which maps coordinates from C_k to C_{k-1}. The propagated camera-to-LiDAR pose is therefore

$$\boldsymbol{T}_{LC_k} = \boldsymbol{T}_{LC_{k-1}} \boldsymbol{T}_{C_{k-1}C_k}. \tag{8}$$

Expanding Eq. (8) gives

$$\begin{aligned} \boldsymbol{R}_{LC_k} &= \boldsymbol{R}_{LC_{k-1}} \boldsymbol{R}_{C_{k-1}C_k}, \\ \boldsymbol{t}_{LC_k} &= \boldsymbol{t}_{LC_{k-1}} + \boldsymbol{R}_{LC_{k-1}} \boldsymbol{t}_{C_{k-1}C_k}. \end{aligned} \tag{9}$$

The camera–inertial frame has its x-, y-, and z-axes pointing forward, left, and upward, respectively. The propagated pose is the center of the subsequent sampling region.

4.2.2 *Bounded PSO coarse alignment*

PSO is adopted as the coarse sampling strategy due to its swarm-based search behavior, which provides natural diversity in candidate solutions and derivative-free exploration in highly non-convex pose spaces. In this work, PSO provides the pose initialization for subsequent local refinement.

Let TF_k be the number of projected visual features and VF_k the number that pass the planarity and residual gate $|r_l| < \tau_r$. We define the association-quality ratio

$$q_k = \frac{\mathrm{VF}_k}{\max(\mathrm{TF}_k, 1)}. \tag{10}$$

The ratio q_k determines whether the propagated pose can directly initialize local refinement. If $q_k \geqslant \tau_q$, the propagated pose is retained. Otherwise, bounded PSO search is performed to obtain a coarse initialization.

The single-threshold rule may lead to alternating PSO activation when q_k fluctuates around τ_q, particularly when few visual features are available. To state the expansion rule explicitly, let $\boldsymbol{b}^{\min}, \boldsymbol{b}^{\max} \in \mathbb{R}^6_+$ be the lower and upper translation–rotation half-widths and let $\boldsymbol{\kappa} \in \mathbb{R}^6_+$ control expansion. The bounded half-width is

$$\boldsymbol{b}(q_k) = \mathrm{clip}\big(\boldsymbol{b}^{\min} + \boldsymbol{\kappa}[\tau_q - q_k]_+,\, \boldsymbol{b}^{\min}, \boldsymbol{b}^{\max}\big), \tag{11}$$

where $[a]_+ = \max(a, 0)$ and clipping is element-wise. The sampling region centered at the propagated pose is

$$\mathcal{B}_k = \{\boldsymbol{T}_{LC_k} \oplus \delta\boldsymbol{\xi} \,|\, |\delta\xi_j| \leqslant b_j(q_k),\ j = 1, \ldots, 6\}. \tag{12}$$

The upper half-width $\boldsymbol{b}^{\max}$ bounds the search region around the propagated pose.

Thus, lower geometric consistency enlarges the search region but never beyond $\boldsymbol{b}^{\max}$. The PSO fitness is Eq. (6). Table 1 lists the PSO configuration. Unless otherwise stated, the same PSO configuration is used for all datasets, perturbation levels, and real-platform experiments.

Algorithm 1 summarizes the association-quality check,

Just Accepted

Table 1 PSO parameters

Parameter	Meaning	Value
m	Particle count	10
$G_{\max}$	Maximum iterations	15
w	Inertia weight	0.5
c_1	Cognitive coefficient	1.5
c_2	Social coefficient	1.5

Algorithm 1 Bounded PSO coarse alignment

Input: Visual features $\mathcal{F}$, LiDAR map $\mathcal{L}$, propagated pose $\boldsymbol{T}_{LC_k}$
Output: Pose initialization $\boldsymbol{T}^0_{LC_k}$

Compute q_k using Eq. (10);
if $q_k \geqslant \tau_q$ **then**
  return $\boldsymbol{T}_{LC_k}$;
end if
Construct $\mathcal{B}_k$ using Eqs. (11)–(12);
$\{\boldsymbol{p}_i, \boldsymbol{v}_i\}_{i=1}^m \leftarrow \text{InitializePSO}(\mathcal{B}_k)$;
$f(\boldsymbol{p}) \equiv \rho(\boldsymbol{T}_{LC_k} \oplus \boldsymbol{p})$; $\boldsymbol{p}$ is a 6-D perturbation;
$\{\boldsymbol{pbest}_i\}_{i=1:m} \leftarrow \{\boldsymbol{p}_i\}_{i=1:m}$;
$\boldsymbol{gbest} \leftarrow \arg\min_{\boldsymbol{p}_i} f(\boldsymbol{p}_i)$;
for $g = 1$ **to** $G_{\max}$ **do**
  for $i = 1$ **to** m **do**
    Recompute valid planes at $\boldsymbol{T}_{LC_k} \oplus \boldsymbol{p}_i$ and evaluate $f(\boldsymbol{p}_i)$;
    if $f(\boldsymbol{p}_i) < f(\boldsymbol{pbest}_i)$ **then**
      $\boldsymbol{pbest}_i \leftarrow \boldsymbol{p}_i$;
    end if
    if $f(\boldsymbol{p}_i) < f(\boldsymbol{gbest})$ **then**
      $\boldsymbol{gbest} \leftarrow \boldsymbol{p}_i$;
    end if
  end for
  for $i = 1$ **to** m **do**
    $r_1, r_2 \sim \mathcal{U}(0,1)$;
    $\boldsymbol{v}_i \leftarrow w\boldsymbol{v}_i + c_1 r_1(\boldsymbol{pbest}_i - \boldsymbol{p}_i) + c_2 r_2(\boldsymbol{gbest} - \boldsymbol{p}_i)$;
    $\boldsymbol{p}_i \leftarrow \boldsymbol{p}_i + \boldsymbol{v}_i$;
    Clip the perturbation components to $[-b_j(q_k), b_j(q_k)]$ if a boundary is crossed;
  end for
end for
return $\boldsymbol{T}^0_{LC_k} \leftarrow \boldsymbol{T}_{LC_k} \oplus \boldsymbol{gbest}$;

bounded search-region construction, boundary handling, and PSO updates. The particle state is a 6-D pose perturbation in the tangent space centered at $\boldsymbol{T}_{LC_k}$. Plane associations are recomputed for every particle because the valid local geometry depends on the hypothesized pose.

Let $\delta\boldsymbol{\xi}^\star_{\text{PSO}}$ denote the selected perturbation. The pose supplied to local refinement is

$$\boldsymbol{T}^0_{LC_k} = \begin{cases} \boldsymbol{T}_{LC_k}, & q_k \geqslant \tau_q, \\ \boldsymbol{T}_{LC_k} \oplus \delta\boldsymbol{\xi}^\star_{\text{PSO}}, & q_k < \tau_q. \end{cases}$$

The selected perturbation is composed with the propagated pose, and the resulting pose and valid plane associations initialize the visual–inertial–LiDAR estimator.

4.2.3 Basin-entry analysis

The visual–inertial–LiDAR objective is non-convex because the valid plane associations change with the pose. Local refinement is therefore sensitive to initialization.

PSO is used as a heuristic initializer that explores the bounded region in Eq. (12). It is a basin selector intended to increase the probability of entering a suitable local basin.

During sampling, plane associations are recomputed and Eq. (6) guides the search. After the coarse estimate is selected, its associations initialize NLS. Figure 5 compares PSO-based, local-optimization-based, and random initialization followed by local refinement.

Escape time (ET) is expressed in iteration counts. In Fig. 5D–E, boxes span the first to third quartiles, lines mark medians, whiskers reach the most extreme observations within 1.5 interquartile ranges (IQRs) of the box edges, and open circles indicate observations beyond the whiskers. "No Escape" denotes no finite observed ET and is displayed separately from the finite ET distribution.

Across 100 trials per method, PSO achieves a mean escape time of 15.5 iterations, compared with 59.3 for the local-optimization baseline (Fig. 5D). PSO also exhibits a lower median initialization error and a narrower interquartile range than the other methods (Fig. 5E). The corresponding basin-entry rates are 1.0, 0.7, and 0.5 for PSO, the local-optimization baseline, and random initialization, respectively (Fig. 5F).

PSO updates candidate poses using their individual best hypotheses and the swarm-level best hypothesis (Algorithm 1). Iteration counts in Fig. 5 are defined separately for each algorithm, with each PSO iteration evaluating multiple particles.

4.3 Optimization-based local refinement

The pose initialization from Sec. 4.2.2 initializes a bundle-adjustment formulation [38] that combines visual reprojection, IMU pre-integration, and LiDAR plane constraints. The coarse stage minimizes the mean absolute point-to-plane residual in Eq. (6), while local refinement minimizes the robust, weighted visual–inertial–LiDAR objective in Eq. (14). Under a large pose disturbance, pose-dependent plane associations can place a local solver in an incorrect attraction basin,

Just Accepted

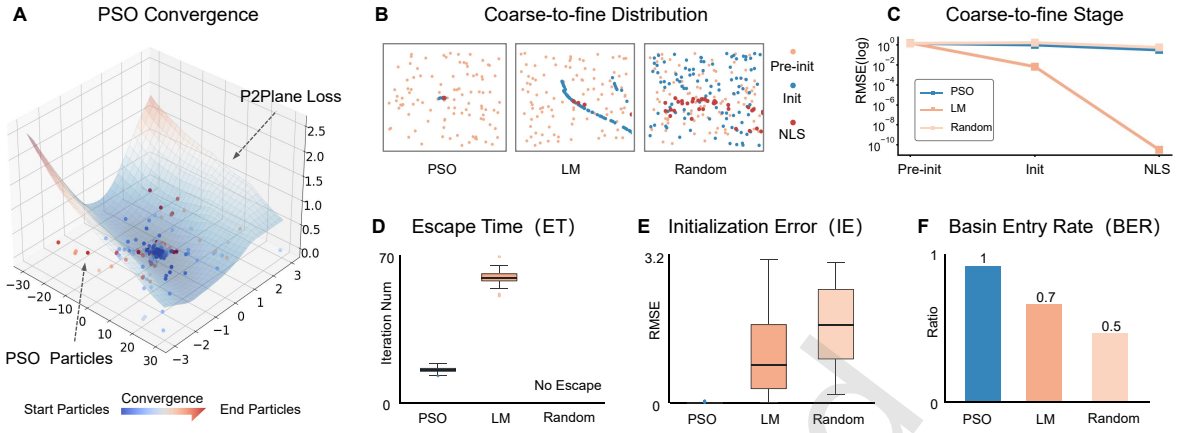


Fig. 5 Ablation of the basin-entry stage. A total of 100 trials per method are conducted on the same simulated non-convex point-to-plane registration problem. **A** PSO particles on one representative loss slice. **B** Pre-initialization, initialization, and NLS estimate distributions. **C** Error across the three stages. **D** Escape time (ET). **E** Initialization error (IE). **F** Basin-entry rate over 100 trials.

and bounded coarse alignment seeks an alternative starting point before local refinement.

The relative transformation between the camera trajectory and LiDAR map is included in the state: $\boldsymbol{x}_r = [\boldsymbol{R}_{LC}, \boldsymbol{t}_{LC}]$. The full system state is then written as:

$$\begin{aligned} \mathcal{X}_c &= \{\boldsymbol{x}_r, \boldsymbol{x}_c, \boldsymbol{x}_f\} \\ \boldsymbol{x}_r &= \{\boldsymbol{R}_{LC}, \boldsymbol{t}_{LC}\} \\ \boldsymbol{x}_c &= \{[\boldsymbol{R}_{C_0C_k}, \boldsymbol{t}_{C_0C_k}, \boldsymbol{v}_{C_0C_k}, \boldsymbol{b}_a, \boldsymbol{b}_g]_{k=[0,n_h]}\} \\ \boldsymbol{x}_f &= \{\lambda_0, ..., \lambda_{n_f}\} \end{aligned} \tag{13}$$

where C_0 is the initial VIO frame and $\{C_k\}_{k=0}^{n_h}$ are the camera frames in the sliding window. The state $\boldsymbol{x}_c$ contains camera poses, velocities, and accelerometer and gyroscope biases, while $\boldsymbol{x}_f$ contains the inverse depths used by the visual reprojection factors.

The factor graph minimizes the robust, weighted sum of the three residual families:

$$\begin{aligned} \mathcal{X}_c = \operatorname{argmin}_{\mathcal{X}_c} \Big[& \sum_{l=1}^{n_\mathcal{P}} \rho_H\big(\Omega_{\mathcal{P}_l} \boldsymbol{\mathcal{R}}^2_{\mathcal{P}_l}(\boldsymbol{x}_r, \bar{\lambda}_l, \boldsymbol{P}_l)\big) \\ & + \sum_{k=0}^{n_\mathcal{I}-1} \Big\| \boldsymbol{\mathcal{R}}_{\mathcal{I}_{k,k+1}}(\boldsymbol{x}_c, \mathcal{Z}_{I_k}^{I_{k+1}}) \Big\|^2_{\Sigma^{-1}_{\mathcal{I}_{k,k+1}}} \\ & + \sum_{i=0}^{n_\mathcal{F}} \sum_{j \in \mathcal{N}_i} \big\| \boldsymbol{\mathcal{R}}_{\mathcal{C}_{ij}}(\boldsymbol{x}_c, \boldsymbol{x}_f) \big\|^2_{\Sigma^{-1}_{\mathcal{C}_{ij}}} \Big] \end{aligned} \tag{14}$$

Here, ρ_H is the Huber loss, $\Omega_{\mathcal{P}_l}$ is the scalar point-to-plane information defined below, and $\bar{\lambda}_l$ is the inverse depth held fixed within that LiDAR factor. The visual factors continue to update $\lambda_l \in \boldsymbol{x}_f$. The three residuals denote point-to-plane distance, IMU pre-integration, and visual reprojection, respectively. The resulting problem is solved using Ceres Solver [39].

IMU Pre-Integration Factor

Between consecutive IMU frames, acceleration and angular-rate measurements are pre-integrated to constrain relative position, velocity, and orientation. We follow the standard bias-aware on-manifold formulation [40, 41]; $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$ denote the position, velocity, and rotation pre-integrates.

With gravity $\boldsymbol{g}$ expressed in C_0, the residual is

$$\boldsymbol{\mathcal{R}}_{\mathcal{I}_{k,k+1}} = \begin{bmatrix} \boldsymbol{r}_p^T & \boldsymbol{r}_v^T & \boldsymbol{r}_R^T & \boldsymbol{r}_{b_a}^T & \boldsymbol{r}_{b_g}^T \end{bmatrix}^T, \tag{15}$$

where

$$\begin{aligned} \boldsymbol{r}_p &= \boldsymbol{R}^T_{C_0I_k}\big(\boldsymbol{p}_{C_0I_{k+1}} - \boldsymbol{p}_{C_0I_k} - \boldsymbol{v}_{C_0I_k}\Delta t_k \\ &\qquad - \tfrac{1}{2}\boldsymbol{g}\Delta t_k^2\big) - \hat{\boldsymbol{\alpha}}^{I_k}_{I_{k+1}}, \\ \boldsymbol{r}_v &= \boldsymbol{R}^T_{C_0I_k}\big(\boldsymbol{v}_{C_0I_{k+1}} - \boldsymbol{v}_{C_0I_k} - \boldsymbol{g}\Delta t_k\big) - \hat{\boldsymbol{\beta}}^{I_k}_{I_{k+1}}, \\ \boldsymbol{r}_R &= 2\left[\left(\hat{\boldsymbol{\gamma}}^{I_k}_{I_{k+1}}\right)^{-1} \otimes \boldsymbol{q}^{-1}_{C_0I_k} \otimes \boldsymbol{q}_{C_0I_{k+1}}\right]_{xyz}, \\ \boldsymbol{r}_{b_a} &= \boldsymbol{b}_{a,I_{k+1}} - \boldsymbol{b}_{a,I_k}, \qquad \boldsymbol{r}_{b_g} = \boldsymbol{b}_{g,I_{k+1}} - \boldsymbol{b}_{g,I_k}. \end{aligned} \tag{16}$$

Here, $\boldsymbol{p}_{C_0I_k}$, $\boldsymbol{v}_{C_0I_k}$, and $\boldsymbol{q}_{C_0I_k}$ are the IMU position, velocity, and orientation in C_0; Δt_k is the integration interval; and $\otimes$ denotes quaternion multiplication.

The continuous-time IMU model uses additive white accelerometer and gyroscope measurement noise together with random-walk bias noise. The associated error-state transition and noise-injection matrices are discretized at each IMU sample, and the covariance is propagated recursively as

$$\boldsymbol{\Sigma}_{k+1} = \boldsymbol{F}_k\boldsymbol{\Sigma}_k\boldsymbol{F}_k^T + \boldsymbol{G}_k\boldsymbol{Q}_k\boldsymbol{G}_k^T, \tag{17}$$

where $\boldsymbol{Q}_k$ is formed from the sensor noise densities and bias random-walk densities. The resulting block covariance is used as $\boldsymbol{\Sigma}_{\mathcal{I}_{k,k+1}}$ in Eq. (14), following [41].

Visual Reprojection Factor

For a feature first observed in frame i and reobserved in

Just Accepted

frame j, the reprojection residual is

$$\begin{aligned}\boldsymbol{\mathcal{R}}_{\mathcal{C}_{ij}}(\boldsymbol{x}_c,\boldsymbol{x}_f) &= \boldsymbol{u}_j - \pi_c\left(\frac{\boldsymbol{p}_i^{C_j}}{\|\boldsymbol{p}_i^{C_j}\|}\right),\\ \boldsymbol{p}_i^{C_j} &= \boldsymbol{T}_{CI}\boldsymbol{T}_{C_0I_j}^{-1}\boldsymbol{T}_{C_0I_i}\boldsymbol{T}_{IC}\oplus\frac{1}{\lambda_i}\pi_c^{-1}(\boldsymbol{u}_i).\end{aligned} \tag{18}$$

Here, $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ are pixel observations, λ_i is the inverse depth in the first observation, π_c is the camera projection function, and $\boldsymbol{T}_{CI}$ is the fixed IMU-to-camera transform. The VIO estimate provides the propagated pose in Eq. (8); when coarse realignment is required, PSO replaces that initialization before joint refinement.

Point-to-Plane Factor

Let the plane associated with feature l be $\boldsymbol{P}_l = \langle \hat{\boldsymbol{n}}_l, d_l\rangle$ in frame L. The point-to-plane factor uses the inverse-depth estimate $\bar{\lambda}_l$ supplied by the current visual–inertial state and holds it fixed while this LiDAR factor is linearized:

$$\boldsymbol{\mathcal{R}}_{\mathcal{P}_l} = \hat{\boldsymbol{n}}_l^T\left(\frac{1}{\bar{\lambda}_l}\boldsymbol{R}_{LC}\boldsymbol{x}_l^C + \boldsymbol{t}_{LC}\right) + d_l. \tag{19}$$

For a right-multiplicative rotation perturbation and additive translation perturbation, the pose Jacobian is

$$\boldsymbol{\mathcal{J}}_{\mathcal{P}_l} = \begin{bmatrix} -\frac{1}{\bar{\lambda}_l}\hat{\boldsymbol{n}}_l^T\boldsymbol{R}_{LC}[\boldsymbol{x}_l^C]_\times & \hat{\boldsymbol{n}}_l^T\end{bmatrix}. \tag{20}$$

Because Eq. (19) uses the frozen value $\bar{\lambda}_l$, this factor has no inverse-depth Jacobian column. The visual reprojection factors still optimize λ_l, and the point-to-plane factor is relinearized using the updated value at the next outer iteration.

Residual weighting is defined in units of inverse squared distance. Let $\boldsymbol{\Sigma}_{p_l}^C$ be the 3-D point covariance propagated from bearing and inverse-depth uncertainty, $\sigma_{\text{fit},l}^2$ the mean squared orthogonal plane-fitting residual of the neighboring LiDAR points, and $q_l \in (0,1]$ their inlier fraction. The signed-distance variance and scalar information are

$$\begin{aligned}\sigma_{\mathcal{P}_l}^2 &= \hat{\boldsymbol{n}}_l^T\boldsymbol{R}_{LC}\boldsymbol{\Sigma}_{p_l}^C\boldsymbol{R}_{LC}^T\hat{\boldsymbol{n}}_l + \sigma_{\text{fit},l}^2 + \sigma_{\text{LiDAR}}^2,\\ \Omega_{\mathcal{P}_l} &= \frac{q_l}{\sigma_{\mathcal{P}_l}^2+\epsilon}.\end{aligned} \tag{21}$$

The plane-fitting threshold (0.05–0.20 m) is selected according to map resolution. A Huber loss in Eq. (14) limits the influence of remaining outliers.

The plane-fitting gate evaluates the consistency of neighboring LiDAR points, while τ_r tests the distance from a projected visual point to its associated plane. Equation (21) determines the weight of each accepted point-to-plane residual. A tighter fitting threshold rejects more candidate associations, whereas a looser threshold retains more candidates at the risk of less consistent local geometry.

A LiDAR-equipped UGV builds or supplies the map, and N_{UAV} camera-equipped UAVs enter the mapped region. Each UAV initializes and estimates its trajectory in the common LiDAR frame.

For UAV $a = 1,\dots,N_{\text{UAV}}$, the variables are

$$\begin{aligned}\mathcal{X}_c^a &= \{\boldsymbol{x}_r^a, \boldsymbol{x}_c^a, \boldsymbol{x}_f^a\},\\ \boldsymbol{x}_r^a &= \{\boldsymbol{R}_{LC}^a, \boldsymbol{t}_{LC}^a\},\\ \boldsymbol{x}_c^a &= \{[\boldsymbol{R}_{C_0C_k}^a, \boldsymbol{t}_{C_0C_k}^a, \boldsymbol{v}_{C_0C_k}^a, \boldsymbol{b}_a^a, \boldsymbol{b}_g^a]_{k=0}^{n_h}\},\\ \boldsymbol{x}_f^a &= \{\lambda_0^a,\dots,\lambda_{n_f}^a\}.\end{aligned} \tag{22}$$

The backend solves one map-assisted BA problem per UAV:

$$\begin{aligned}\mathcal{X}_c^a = \arg\min_{\mathcal{X}_c^a}\Bigg[&\sum_{l=1}^{n_{\mathcal{P}}}\rho_H\left(\Omega_{\mathcal{P}_l}\,\boldsymbol{\mathcal{R}}_{\mathcal{P}_l}^2(\boldsymbol{x}_r^a,\bar{\lambda}_l^a,\boldsymbol{P}_l)\right)\\ &+\sum_{k=0}^{n_{\mathcal{I}}-1}\left\|\boldsymbol{\mathcal{R}}_{\mathcal{I}_{k,k+1}}(\boldsymbol{x}_c^a,\mathcal{Z}_{I_k}^{I_{k+1}})\right\|_{\Sigma_{\mathcal{I}_{k,k+1}}^{-1}}^2\\ &+\sum_{r=0}^{n_{\mathcal{F}}}\sum_{j\in\mathcal{N}_r}\left\|\boldsymbol{\mathcal{R}}_{\mathcal{C}_{rj}}(\boldsymbol{x}_c^a,\boldsymbol{x}_f^a)\right\|_{\Sigma_{\mathcal{C}_{rj}}^{-1}}^2\Bigg].\end{aligned} \tag{23}$$

For a fixed UGV-provided map, Eq. (23) separates into one optimization problem per UAV. Each estimator uses its own camera–IMU observations and map associations, with no direct inter-UAV factors; the coupling is through the UGV-provided map and the shared coordinate frame. Its accuracy depends on both each UAV estimator and the quality of the shared LiDAR map.

5 Experiments

We evaluate the method in three settings. EuRoC provides synchronized camera–IMU data, a static Leica scan, and ground truth for controlled perturbation tests (Sec. 5.2). The indoor platform uses a UGV to construct the LiDAR map online and motion capture to evaluate three UAVs (Sec. 5.3). The outdoor platform evaluates operation under reduced feature counts using a UGV-built map and onboard LiDAR odometry as a reference (Sec. 5.4). The last reference is not independent ground truth, so the outdoor results are reported as estimator consistency rather than absolute accuracy.

5.1 Metrics and perturbation protocol

Let $\hat{\boldsymbol{T}}_{LC_k}$ and $\boldsymbol{T}_{LC_k}^{\text{gt}}$ be the estimated and reference poses in the same fixed map frame. We report translational absolute pose error (APE) and its root-mean-square value:

$$\begin{aligned}e_k^{\text{APE}} &= \left\|\text{trans}\left((\boldsymbol{T}_{LC_k}^{\text{gt}})^{-1}\hat{\boldsymbol{T}}_{LC_k}\right)\right\|_2,\\ \text{ATE}_{\text{RMSE}} &= \sqrt{\frac{1}{K}\sum_{k=1}^{K}(e_k^{\text{APE}})^2}.\end{aligned} \tag{24}$$

All values in Table 2 are in meters. Dataset calibration provides the common map frame; no per-trajectory similarity or rigid alignment is applied after estimation.

Just Accepted

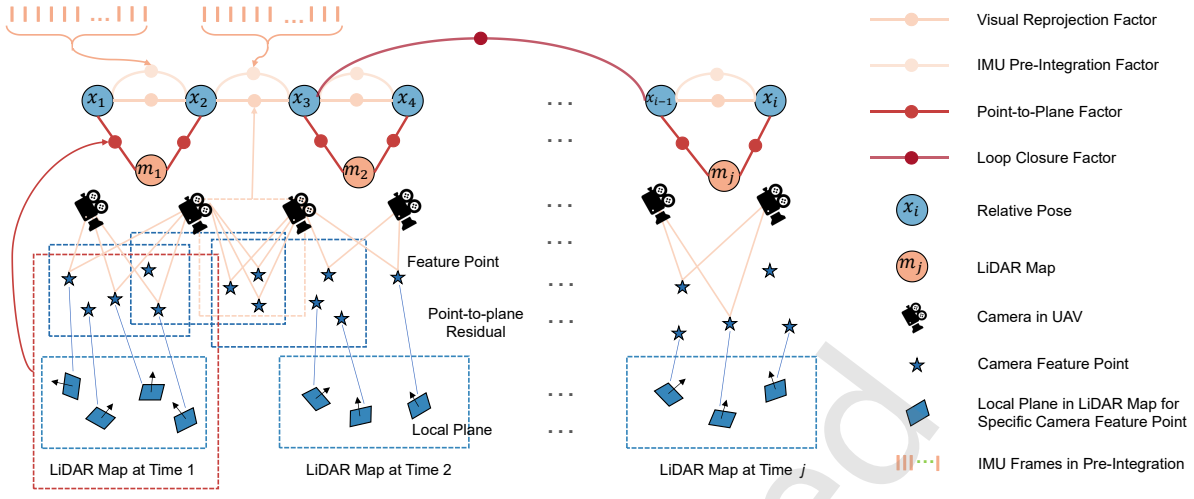


Fig. 6 System structure and factor graph. The estimator receives a LiDAR map, camera images, and IMU measurements. The local objective in Eq. (14) contains point-to-plane, visual reprojection, and IMU pre-integration factors. The loop-closure path shown in the VIO front end is inherited from VINS-Mono.

Table 2 Translational ATE RMSE (m) under the common perturbation protocol

Dataset	V1-03-difficult			V2-03-difficult			V2-02-medium		
	Raw	Ini-Perturb	Mid-Perturb	Raw	Ini-Perturb	Mid-Perturb	Raw	Ini-Perturb	Mid-Perturb
Line Corres	**0.21**	0.46	failed	0.89	1.00	4.66	0.58	1.82	failed
GMMLoc	1.97	1.97	failed	failed	failed	failed	2.00	2.00	failed
VINS-Mono (Loop)	0.44	0.74	0.51	0.80	0.87	2.89	0.40	0.58	0.71
Our Method	0.35	**0.38**	**0.37**	**0.39**	**0.37**	**0.48**	**0.38**	**0.37**	**0.34**

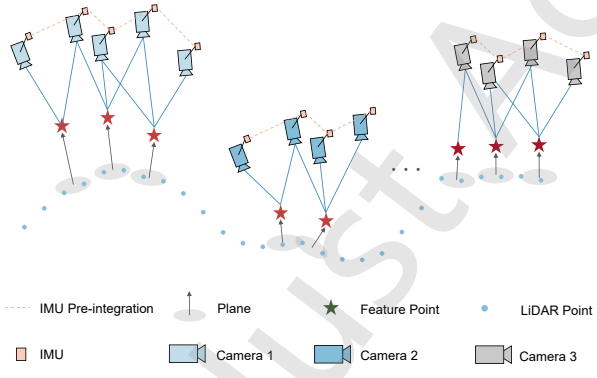


Fig. 7 Multi-UAV map-assisted localization. Each UAV uses its own visual–inertial state and point-to-plane factors to localize in the LiDAR map provided by the UGV. The UAV problems share the map frame but contain no direct inter-UAV factors.

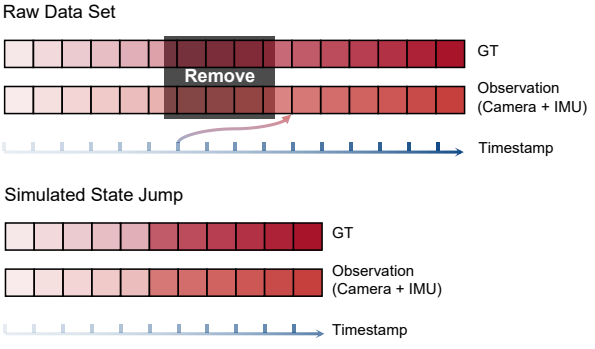


Fig. 8 Data modification procedure used to simulate state jumps on the EuRoC MAV Dataset. A contiguous segment of synchronized ground truth and sensor observations (camera and IMU) is removed and then reconnected along the original timestamp axis, preserving temporal continuity while introducing discontinuities in the underlying state trajectory.

Two protocols are reported. For Table 2, Ini-Perturb adds the same approximately 0.2 m translation bias to every method, while Mid-Perturb uses the same modified sensor stream obtained by removing and reconnecting a contiguous segment, creating an approximately 0.2 m state discontinuity (Fig. 8). The severe-case figures use a 0.5 m initial translation offset on V1-03-difficult and a 30° yaw jump at 53 s on V2-02-medium. The identical initialization or modified input is supplied to every applicable method. "Failed" means that a method did not return a finite, continuous trajectory over the evaluation interval.

5.2 Comparison on the EuRoC dataset

EuRoC contains stereo images at 20 Hz, IMU measurements at 200 Hz, ground-truth trajectories, and a Leica MS50 structural scan [42]. We use the left image stream, the IMU, and the static Leica scan as the LiDAR map. Experiments run on an Intel i7-12700 CPU with 16 GB RAM. We select V1-

Just Accepted

03-difficult, V2-03-difficult, and V2-02-medium because they contain aggressive motion and challenging visual conditions.

We compare with VINS-Mono [14], 2-D–3-D line-correspondence localization (Line Corres) [43], and GMM-Loc [44]. Line Corres and GMMLoc preprocess the same static map to extract 3-D lines or construct a Gaussian-mixture representation. Their initial camera pose is the ground-truth pose plus the protocol-specific perturbation. VINS-Mono does not use the LiDAR map and therefore measures the value of map correction relative to a visual–inertial baseline. These baselines test recovery behavior but do not cover recent learned image-to-point registration.

Table 2 reports the common protocol. Line Corres is sensitive to the modified mid-sequence input and reaches 4.66 m on V2-03-difficult; GMMLoc does not return a valid trajectory in most mid-sequence cases. VINS-Mono also degrades on V2-03-difficult, where its error reaches 2.89 m. The proposed method remains between 0.34 and 0.48 m in the nine reported EuRoC conditions.

Figures 9 and 10 report the two severe cases defined in Sec. 5.1. Under the 0.5 m initial offset, the proposed method obtains 0.51 m RMSE, compared with 0.77 m for the best baseline, a relative reduction of $(0.77-0.51)/0.77 = 33.8\%$.

After the 30° yaw jump at 53 s, Line Corres and GMMLoc diverge, whereas the proposed method retains a bounded APE. Over the post-disturbance interval, its mean APE is 55.2% lower than that of the only baseline that returns a complete trajectory, VINS-Mono.

With 10 particles and 15 iterations, an activated PSO call requires 17 ms on average on the i7-12700 platform. The $q_k \geqslant \tau_q$ branch bypasses PSO and directly initializes local refinement. The sampling cost depends on point-cloud density, and voxel downsampling is applied before neighborhood queries.

5.3 Evaluation in indoor scenarios

To evaluate the air–ground setting, we conduct experiments in an indoor environment equipped with an FZMotion motion-capture system. As shown in Fig. 11A, a UGV constructs the LiDAR map online. Each UAV carries an Intel RealSense D435i to collect synchronized image and IMU data. The measurements are processed on an Intel NUC 12 and transmitted to the UGV over WiFi. The UGV is equipped with a LiDAR and an IMU as its primary sensing modalities, and runs on an onboard industrial PC with an Intel Core i7-4790K processor and 32 GB RAM, as shown in Fig. 11C.

The LiDAR point-cloud map is incrementally constructed using FAST-LIO2 [18] and stored in a dynamically updated KD-tree. As illustrated in Fig. 11B, three UAVs are manually operated along different trajectories and enter the mapped region at different times. Each UAV receives only a coarse initial pose, after which its own estimator registers to the common LiDAR frame. This experiment validates online UGV map construction and simultaneous shared-frame operation of the three independently initialized UAV estimators.

For each UAV trajectory, we compare the proposed method with VINS and the ground truth. As shown in Fig. 11D, since VINS cannot localize within the LiDAR point cloud map, it is provided with an accurate initial pose, and only drift accumulation and pose jumps caused by motion disturbances are considered. It can be observed that under severe image jitter and motion blur, all three UAV trajectories suffer from significant scale estimation errors, with RMSE reaching 2.4 m, 1.36 m, and 1.13 m, respectively. In contrast, the proposed method reduces the error by 58.3%, 28.7%, and 46.0%, as illustrated in Fig. 11E–F. These results show 20 Hz operation with three UAV estimators and one UGV map server under the tested onboard-computation and wireless-communication conditions.

Fig. 12 illustrates the camera trajectory estimated by the proposed method in a dynamic indoor corridor environment. The colored point cloud is acquired by LiDAR and incrementally constructed in real time based on LIO. The blue points represent sparse feature points, while the green points correspond to matched camera feature points after visual–LiDAR association. The white curve denotes the estimated camera trajectory.

In corridor scenarios, the front end may occasionally fail to extract a sufficient number of reliable features. Under aggressive motion, pure visual–inertial odometry tends to exhibit noticeable drift, particularly at corners. In contrast, within our system, feature points are refined through visual–LiDAR matching, enabling accurate alignment with the surrounding geometric structure. Overall, the camera motion remains relatively smooth along the corridor, while bounded PSO search supplies a coarse realignment at challenging turning points before local refinement.

5.4 Evaluation in outdoor scenarios

We finally evaluate a plaza with fewer usable visual features. Compared with the indoor runs, the mean number of valid associations decreases from 134.08 to 50.72, while the standard deviation changes from 42.65 to 52.99 (Fig. 13C). The LiDAR map is built before the UAV evaluation runs. UAV-1 and UAV-3 traverse the left and right loops, where

Just Accepted

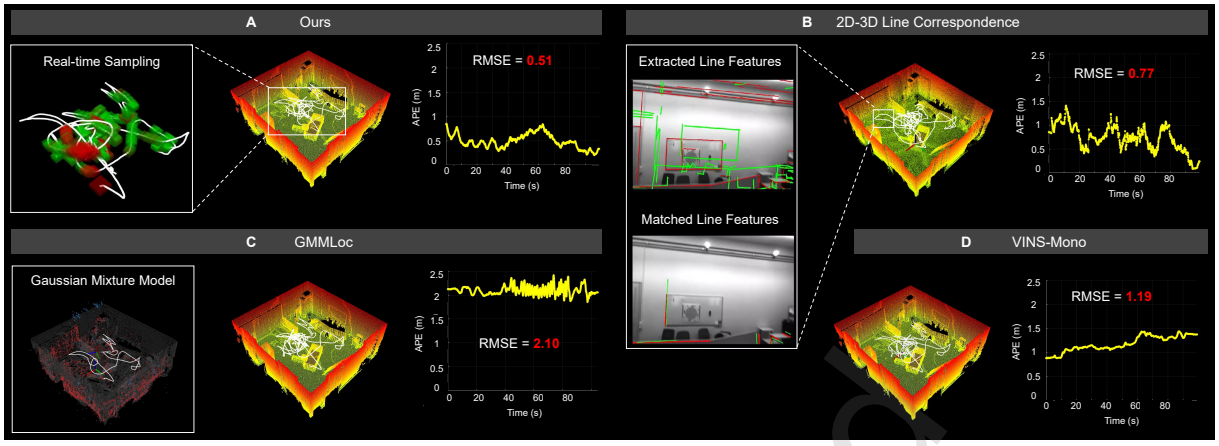


Fig. 9 Performance under 0.5 m initial pose perturbation. A-D show the results of the proposed method, Line Corres, GMMLoc, and VINS-Mono, respectively. The left column visualizes the sampling strategy or feature correspondences, the middle column presents top-view trajectory maps, and the right column plots the APE over time. The proposed method achieves the lowest RMSE (0.51 m), while baseline methods suffer from significant error growth due to failure in correcting the initial bias (Line Corres: 0.77 m, GMMLoc: 2.10 m, VINS-Mono: 1.19 m).

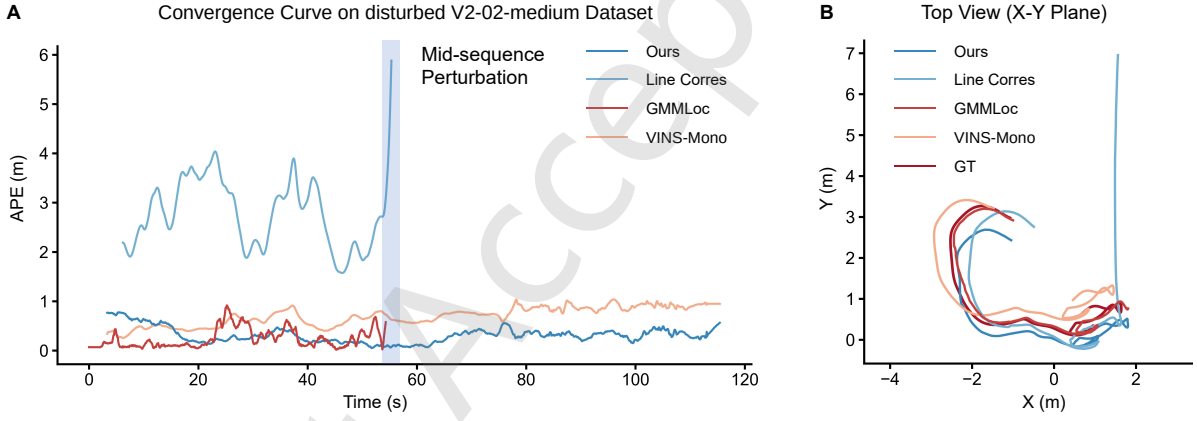


Fig. 10 Performance under mid-sequence 30° perturbation. A APE over time for different methods, where the shaded region indicates the disturbance injection. Line Corres and GMMLoc diverge rapidly after the perturbation, while the proposed method maintains low error. **B** Top-view trajectories in the X–Y plane over the interval 40-55 s.

associations concentrate on railings, while UAV-2 operates near the stairs (Fig. 13B).

Relative to onboard LiDAR odometry, the camera–LiDAR estimator obtains an X-position RMSE of 0.167 m and an X-velocity RMSE of 0.322 m/s (Fig. 13D–E).

6 Discussion

The behavior of the framework depends on the geometric information available for cross-modal alignment. Sparse LiDAR coverage, weak local planarity, and similarly oriented surfaces can reduce the constraints supplied by point-to-plane residuals. A tighter plane-fitting gate improves local consistency but may reject useful associations, whereas a looser gate retains more observations at the risk of admitting inconsistent geometry. Adapting association gates and residual weights to local geometric uncertainty is a promising direction for balancing these effects. Moving objects introduce a related challenge: their observations may be inconsistent with the static map and distort local plane estimates. Motion-aware association and map maintenance could help preserve reliable geometric constraints.

The bounded search region creates a tradeoff between recovery range and computational cost. During long unmapped intervals, the estimator relies on VIO, and accumulated drift may place the correct alignment outside the search region centered at the propagated pose. Expanding this region improves pose coverage but requires the search to explore more hypotheses. Within the region, swarm guidance promotes convergence toward promising candidates, while excessive particle concentration can reduce exploration of alternative basins. The PSO initializer is heuristic and does not guarantee global convergence. Future work could adapt the search extent

Just Accepted

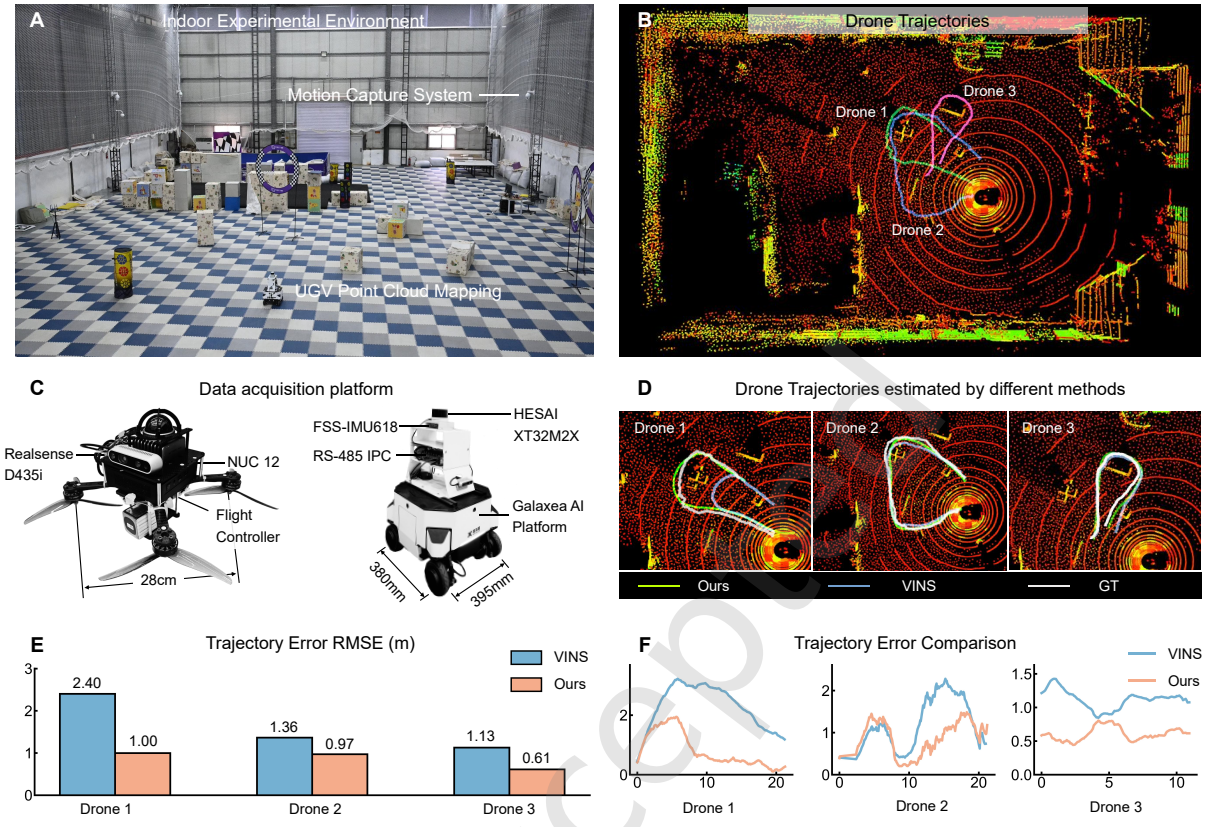


Fig. 11 Estimated trajectories in a multi-agent scenario. A Indoor experimental environment ($35 \times 25 \times 12$ m). A motion capture system (FZMotion) is used to obtain ground-truth trajectories of UAVs, while a UGV builds a dense point cloud map of the environment. **B** Estimated trajectories of UAVs using our method. The initial poses of UAVs are roughly initialized, and they enter the mapped area at different times. Multiple UAVs are manually operated and follow different flight paths. **C** Data acquisition platform. The UAV has a wheelbase of 280 mm and is equipped with a PX4-based flight controller (MicroAir), 7-inch tri-blade propellers. An Intel RealSense D435i is used to collect synchronized visual and IMU data. Motion capture markers are mounted on the UAV for ground-truth acquisition. The UGV is equipped with LiDAR and IMU sensors. The onboard computing platform is an industrial PC (Intel i7-4790K CPU, 32 GB RAM, Ubuntu 20.04 with ROS Noetic). **D** Trajectory comparison between our method, VINS, and ground truth (GT). The white, blue, and green curves represent trajectories estimated by our method, VINS, and ground truth, respectively. **E** RMSE comparison of trajectory estimation for UAVs between our method and VINS. **F** Temporal trajectory error comparison between our method and VINS for different UAVs.

to propagated pose uncertainty and use hierarchical search or particle-diversity control to balance recovery capability and computation.

The association-quality trigger also involves a tradeoff between responsiveness and temporal stability. When only a few visual features are available, individual association changes can cause the quality ratio to cross the activation threshold repeatedly. This may produce repeated coarse-search activation even though the local-refinement objective remains unchanged. Hysteresis, temporal filtering, or a persistence condition could stabilize activation, but may delay recovery after abrupt pose disturbances. An uncertainty-aware trigger that accounts for both association quality and pose confidence is a useful direction for managing this tradeoff.

A shared LiDAR map provides a common geometric reference for multiple UAVs, so map errors or outdated geometry can affect several trajectories through their common reference. Extending the factor graph with inter-UAV relative observations could provide additional constraints when map associations become weak. Such an extension would require consistent treatment of shared-map uncertainty, communication delays, and correlated information to avoid overconfident estimates. These considerations motivate future work on map maintenance and joint estimation across heterogeneous platforms.

7 Conclusion

This paper introduced a coarse-to-fine cross-modal pose-realignment framework for heterogeneous air–ground localization. The framework combines implicit point-to-plane association, bounded PSO-based basin selection, and visual–inertial–LiDAR factor-graph refinement. The association-

Just Accepted

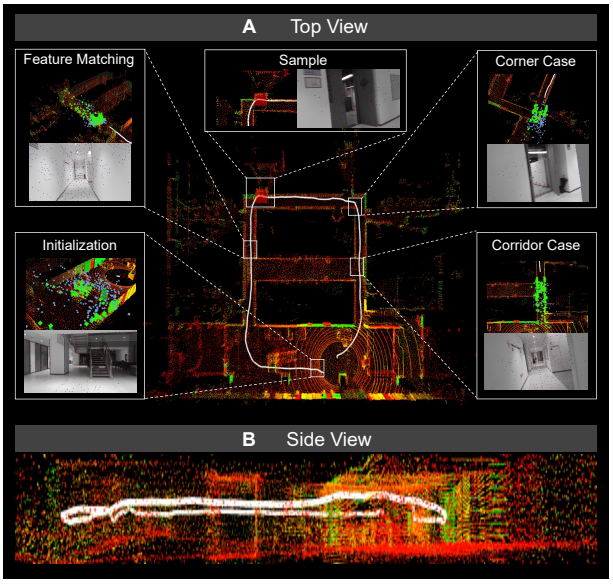


Fig. 12 Estimated trajectory in a corridor scenario. A Top view. Qualitative results of trajectory estimation, with zoom-in views showing initialization, feature matching, sampling, and representative challenging cases, including corridor and corner regions. **B** Side view. Side-view visualization of the estimated trajectory.

quality ratio determines whether the propagated pose can directly initialize local refinement or requires an additional coarse-search stage. In this way, the same geometric association model supports both coarse hypothesis evaluation and accurate local optimization.

Experiments on EuRoC and real indoor and outdoor air–ground platforms demonstrate improved basin-entry performance, recovery from controlled initial and mid-sequence pose disturbances, and stable shared-frame localization with UGV-supplied LiDAR maps. These results support the proposed coarse-to-fine framework as a practical solution for restoring cross-modal alignment after large pose errors.

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

Reference

[1] D. McGann and M. Kaess, imesa: Incremental distributed optimization for collaborative simultaneous localization and mapping, *arXiv preprint arXiv:2406.07371*, 2024.

[2] Y. Tian, Y. Chang, F. H. Arias, C. Nieto-Granda, J. P. How, and L. Carlone, Kimera-multi: Robust, distributed, dense metric-semantic slam for multi-robot systems, *IEEE transactions on robotics*, vol. 38, no. 4, 2022.

[3] P.-Y. Lajoie, B. Ramtoula, Y. Chang, L. Carlone, and G. Beltrame, Door-slam: Distributed, online, and outlier resilient slam for robotic teams, *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1656–1663, 2020.

[4] X. Zhou, X. Wen, Z. Wang, Y. Gao, H. Li, Q. Wang, T. Yang, H. Lu, Y. Cao, C. Xu *et al.*, Swarm of micro flying robots in the wild, *Science robotics*, vol. 7, no. 66, p. eabm5954, 2022.

[5] H. Sun, L. Dai, and B. Chen, Tube-based distributed model predictive control for heterogeneous vehicle platoons via convex optimization, in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 1122–1127.

[6] X. Zhang, L. Yan, T. L. Lam, and S. Vijayakumar, Task-space decomposed motion planning framework for multi-robot loco-manipulation, in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 8158–8164.

[7] Y. Liu, Q. Zheng, Y. Tu, Y. Gao, G. Liang, and T. L. Lam, Configuration-adaptive visual relative localization for spherical modular self-reconfigurable robots, in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 7356–7362.

[8] K. Kameyama, K. Horie, and K. Sekiyama, Optimal viewpoint assistance for cooperative manipulation using d-optimality, *Sensors*, vol. 25, no. 10, p. 3002, 2025.

[9] B. Bao, C. Luo, Y. Hong, Z. Chen, and X. Fan, High-precision UAV positioning method based on MLP integrating UWB and IMU, *Tsinghua Science and Technology*, vol. 30, no. 3, pp. 1315–1328, 2025.

[10] J. Song, Z. Shi, E. E. Kuruoğlu, X.-P. Zhang, and Z. Liu, Multi-source radar high-speed target tracking via transport mapping, *IEEE Transactions on Aerospace and Electronic Systems*, pp. 1–15, 2025, early Access.

[11] H. Xiang, R. Xu, and J. Ma, Hm-vit: Hetero-modal vehicle-to-vehicle cooperative perception with vision transformer, in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 284–295.

[12] Y. Hu, Y. Lu, R. Xu, W. Xie, S. Chen, and Y. Wang, Collaboration helps camera overtake lidar in 3d detection, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9243–9252.

[13] J. Zhu, H. Li, and T. Zhang, Camera, LiDAR, and IMU based multi-sensor fusion SLAM: A survey, *Tsinghua Science and Technology*, vol. 29, no. 2, pp. 415–429, 2024.

[14] T. Qin, P. Li, and S. Shen, Vins-mono: A robust and versatile monocular visual-inertial state estimator, *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[15] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, Keyframe-based visual–inertial odometry using nonlinear optimization, *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[16] M. Kaess, A. Ranganathan, and F. Dellaert, isam: Incremental smoothing and mapping, *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1365–1378, 2008.

[17] W. Xu and F. Zhang, Fast-lio: A fast, robust lidar-inertial

Just Accepted

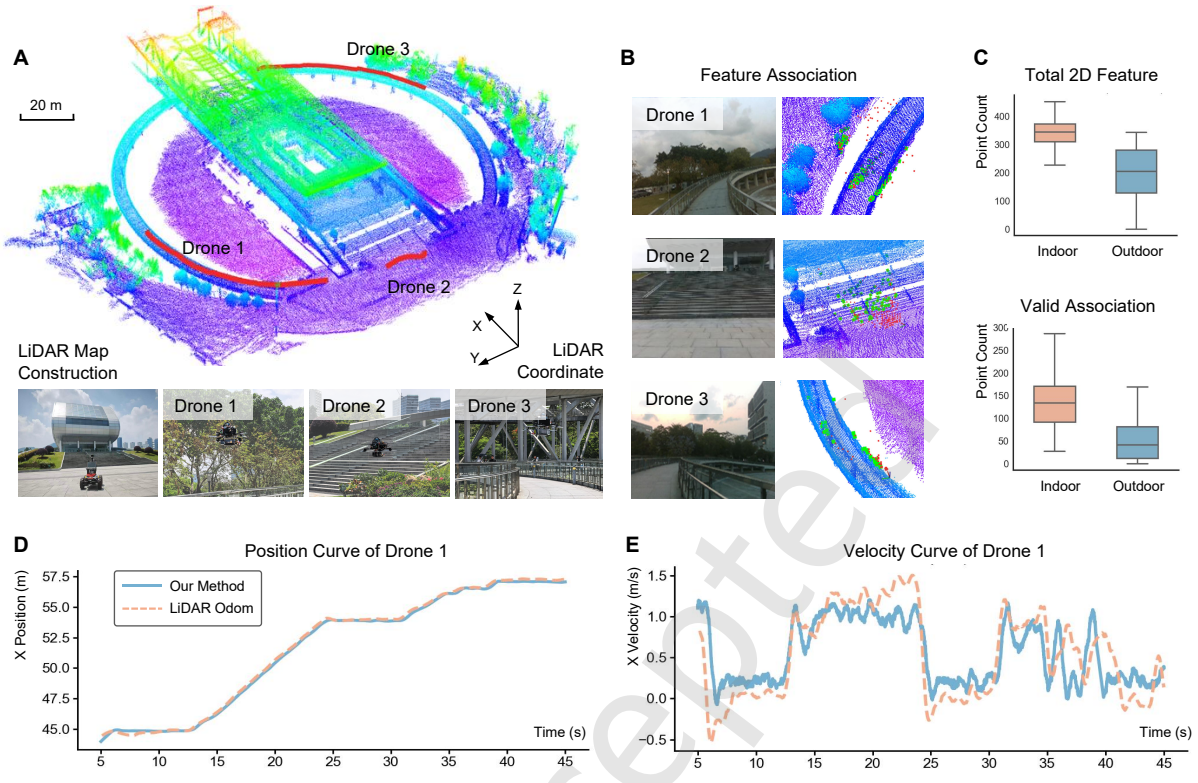


Fig. 13 Outdoor map-assisted localization. **A** Experimental setup: the UGV constructs the LiDAR map before the UAV evaluation runs. **B** Point-to-plane associations between UAV camera features and the 3-D map. **C** Total and valid feature counts indoors and outdoors. **D** X-axis position compared with onboard LiDAR odometry. **E** X-axis velocity compared with onboard LiDAR odometry. The LiDAR odometry is a reference estimator rather than independent ground truth.

Just Accepted

odometry package by tightly-coupled iterated kalman filter, *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3317–3324, 2021.

[18] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, Fast-lio2: Fast direct lidar-inertial odometry, *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.

[19] T. Shan and B. Englot, Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4758–4765.

[20] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and R. Daniela, Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[21] D. Wisth, M. Camurri, S. Das, and M. Fallon, Unified multi-modal landmark tracking for tightly coupled lidar-visual-inertial odometry, *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1004–1011, 2021.

[22] J. Lin and F. Zhang, R 3 live: A robust, real-time, rgb-colored, lidar-inertial-visual tightly-coupled state estimation and mapping package, in *2022 international conference on robotics and automation (ICRA)*. IEEE, 2022, pp. 10 672–10 678.

[23] X. Zhang, K. Huang, J. Zhao, Z. Yuan, and T. Feng, Multi-lvi-sam: A robust lidar-visual-inertial odometry for multiple fisheye cameras, *arXiv preprint arXiv:2509.05740*, 2025.

[24] J. Li and G. H. Lee, Deepi2p: Image-to-point cloud registration via deep classification, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 960–15 969.

[25] B. Wang, C. Chen, Z. Cui, J. Qin, C. X. Lu, Z. Yu, P. Zhao, Z. Dong, F. Zhu, N. Trigoni *et al.*, P2-net: Joint description and detection of local features for pixel and point matching, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 16 004–16 013.

[26] S. Kang, Y. Liao, J. Li, F. Liang, Y. Li, X. Zou, F. Li, X. Chen, Z. Dong, and B. Yang, Cofii2p: Coarse-to-fine correspondences-based image to point cloud registration, *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 10 264–10 271, 2024.

[27] M. Li, Z. Qin, Z. Gao, R. Yi, C. Zhu, Y. Guo, and K. Xu, 2d3d-matr: 2d-3d matching transformer for detection-free registration between images and point clouds, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 14 128–14 138.

[28] X. Li, W. Yang, Z. Cheng, J. Deng, F. Wang, C. Qian, and T. Zhang, Rayi2p: Learning rays for image-to-point cloud

registration, in *The Fourteenth International Conference on Learning Representations*, 2024.

[29] L. Bie, S. Pan, S. Li, Y. Zhao, and Y. Gao, Graphi2p: Image-to-point cloud registration with exploring pattern of correspondence via graph learning, in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 22 161–22 171.

[30] A. Vakhitov, L. Ferraz, A. Agudo, and F. Moreno-Noguer, Uncertainty-aware camera pose estimation from points and lines, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4659–4668.

[31] 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.

[32] J. Kim, C. Choi, H. Jang, and Y. M. Kim, Ldl: Line distance functions for panoramic localization, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 17 882–17 892.

[33] C. Bai, R. Fu, and X. Gao, Colmap-pcd: An open-source tool for fine image-to-point cloud registration, *arXiv preprint arXiv:2310.05504*, 2023.

[34] K. Koide, S. Oishi, M. Yokozuka, and A. Banno, Scalable fiducial tag localization on a 3d prior map via graph-theoretic global tag-map registration, in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 5347–5353.

[35] S. Ren, Y. Zeng, J. Hou, and X. Chen, Corri2p: Deep image-to-point cloud registration via dense correspondence, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 3, pp. 1198–1208, 2022.

[36] X. Li, Z. Xu, C. Wu, Z. Yang, Y. Zhang, J.-J. Liu, H. Yu, X. Ye, Y. Wang, S. Li *et al.*, U-vilar: Uncertainty-aware visual localization for autonomous driving via differentiable association and registration, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 24 889–24 898.

[37] D. Cattaneo and A. Valada, Cmrnext: Camera to lidar matching in the wild for localization and extrinsic calibration, *IEEE Transactions on Robotics*, 2025.

[38] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, Bundle adjustment—a modern synthesis, in *Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings*. Springer, 2000, pp. 298–372.

[39] S. Agarwal, K. Mierle, and T. C. S. Team, Ceres Solver, https://github.com/ceres-solver/ceres-solver, 10 2023, version 2.2.

[40] T. Lupton and S. Sukkarieh, Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions, *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 61–76, 2011.

[41] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation, in *Robotics: Science and Systems XI*, 2015.

[42] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, The euroc micro aerial vehicle datasets, *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[43] H. Yu, W. Zhen, W. Yang, J. Zhang, and S. Scherer, Monocular camera localization in prior lidar maps with 2d-3d line correspondences, in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4588–4594.

[44] H. Huang, H. Ye, Y. Sun, and M. Liu, Gmmloc: Structure consistent visual localization with gaussian mixture models, *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5043–5050, 2020.

Just Accepted