

A Robust and Fast GNSS-Inertial-LiDAR Odometry With INS-Centric Multiple Modalities by IESKF

Jixin Gao^{ID}, Jianjun Sha^{ID}, Hongwen Li^{ID}, and Yanheng Wang^{ID}, *Student Member, IEEE*

Abstract—Global navigation satellite system (GNSS), inertial measurement unit (IMU), and light detection and ranging (LiDAR), as common sensors in navigation systems, are widely used in fusion simultaneous localization and mapping (SLAM) to improve the performance. However, strong robustness and high efficiency are not received much attention, posing challenges for ubiquitous application of robotics systems. This article presents a robust and fast fusion odometry with inertial navigation system (INS)-centric multiple modalities. Specifically, our framework is based on two subsystems: the GNSS-IMU subsystem estimates coarse states and provides global constraints, the LiDAR-IMU subsystem refines states and constructs the 3-D map. The state estimation is performed by iterated error state Kalman filter without interruption even if partial sensor fails. Moreover, a fast and dynamic frame transformation method and distributed backward propagation are designed for linking world-frame states with global measurements and coping with asynchronism data. Extensive qualitative and quantitative experiments are also conducted on public datasets and in real-world scenarios. The results show that our system exhibits competitive accuracy and efficiency, with strong robustness against sensor degradation and failure.

Index Terms—Iterated error state Kalman filter (IESKF), multiple modalities, multisensor simultaneous localization and mapping (SLAM), robust navigation system, sensor failure.

NOMENCLATURE

$i(\cdot)$, $l(\cdot)$, and $g(\cdot)$	Inertial measurement unit (IMU), light detection and ranging (LiDAR), and global navigation satellite system (GNSS) frame.
$b(\cdot)$ and $w(\cdot)$	Body frame and the world frame.
$E(\cdot)$ and $N(\cdot)$	Earth-centered, Earth-fixed (ECEF) frame and the East-North-up (ENU) frame.
${}^g\mathbf{T}_l$, ${}^g\mathbf{T}_i$, and ${}^l\mathbf{T}_i$	Extrinsic between different sensors.
t_k	Time k for measurements.
t_ρ	Time ρ for forward propagation.

Manuscript received 24 September 2023; revised 28 November 2023; accepted 20 December 2023. Date of publication 9 January 2024; date of current version 29 January 2024. This work was supported by the National Key Laboratory on Electromagnetic Environmental Effects and Electro-Optical Engineering under Grant KY3240020001. The Associate Editor coordinating the review process was Dr. Jing Yuan. (*Corresponding author: Jianjun Sha.*)

Jixin Gao, Jianjun Sha, and Hongwen Li are with the Qingdao Innovation and Development Base, Harbin Engineering University, Qingdao 266000, China (e-mail: gaojixin@hrbeu.edu.cn; shajianjun_hh@163.com; hongwen@hrbeu.edu.cn).

Yanheng Wang is with the College of Intelligent System Science and Engineering, Harbin Engineering University, Harbin 150001, China (e-mail: b220040023@hrbeu.edu.cn).

Digital Object Identifier 10.1109/TIM.2024.3351253

$\mathbf{p}_l$	Points sampled by LiDAR.
$\mathbf{p}_g$ and $\mathbf{p}_i$	Positions of GNSS and IMU.
$\mathbf{x}$, $\hat{\mathbf{x}}$, and $\bar{\mathbf{x}}$	Ground-true, predicted, and updated states.
$\delta\mathbf{x}$ and $\hat{\mathbf{x}}^k$	Error state and updated state for the k th iteration.

I. INTRODUCTION

IN RECENT years, simultaneous localization and mapping (SLAM) has become a key technology for the navigation of autonomous robots. It aims to achieve real-time self-localization and map construction in unknown environments, providing the essential information for autonomous exploration [1], [2], path planning [3], [4], [5], and unmanned driving [6], [7]. SLAM can be implemented relying on various sensors, among which vision camera and LiDAR are two distinct basic sensors. Visual SLAM is often camera-based, offering a low-cost advantage but limited by the lighting conditions. LiDAR is considered to be more reliable as it can measure environmental information actively and is not affected by illumination. However, LiDAR SLAM with a single sensor at the core still suffers from many challenges when it works with complex situations such as aggressive motion, open scenes, and featureless tunnels.

To improve the performance of navigation systems, related research has focused on the multisensor fusion approaches, especially LiDAR and IMU [8], [9], [10], [11], [12]. IMU compensates for point cloud distortion with high-frequency measurements, while LiDAR can reduce the accumulated error caused by IMU. Therefore, the integration is natural, effective, and increasingly popular in the deployment of autonomous robotic systems. However, it is difficult for LiDAR-inertial SLAM or LiDAR-inertial odometry (LIO) to achieve stable localization and mapping anytime and anywhere. For the reason that the real environment is complex and unpredictable, any sensor degradation or failure will cause the navigation system to be inoperative. It is hazardous and unacceptable for autonomous robots. Responding to unforeseen situations and challenging environments, a solution is to combine additional sensors or multiple modalities to improve robustness against sensor degradation and failures [13], [14], [15], [16]. Most related multimodal frameworks tend to the front-end/back-end architecture based on optimizer, which optimizes multi-source, multimoment states at the cost of huge computational resources. In contrast, the filter-based framework for state estimation is more lightweight and efficient. However, it has

the typical limitation of inevitable error accumulation, which reduces the system long-term performance and accuracy.

Motivated by the above issues, we concern with the following features of the navigation systems: 1) the strong robustness that tolerates sensor degradation and failure; 2) high computational efficiency for practical implementation; and 3) long-term performance and competitive accuracy. To deliver these requirements, we pay attention to the GNSS and the iterated error state Kalman filter (IESKF). GNSS is a widely equipped sensor in navigation systems and measures the absolute position in space. In addition, GNSS/inertial navigation system (INS), as the typical integrated navigation method, can provide reliable and accurate position outdoors. Due to the complementarity and consistency with LIO, which is relatively localized and is also the common navigation system, the fusion between the sensors is necessary to improve the system robustness and accuracy. For IESKF in state estimation, the error state and iterative processing can improve the linearization [17], [18] and solving ability [19], respectively. Compared to the optimizer, it is lightweight and efficient, reducing the computational resources occupied by the navigation system.

Therefore, we contribute a fusion SLAM framework based on integrating GNSS, IMU, and LiDAR by IESKF. Benefiting from the designed architecture of GNSS-IMU subsystem and LiDAR-IMU subsystem, the proposed system exhibits strong robustness, thus fully utilizing multiple heterogeneous information when the sensors are working and ensuring system operation in the case of sensor degradation or failure. Multisource state estimation is entirely on IESKF, making the system fast and lightweight for easy practical deployment. The integration of GNSS in LIO enables the proposed system to achieve competitive accuracy by improving long-term performance. Moreover, we put forward the corresponding solutions in this article to the problems of absolute/relative frame transformation and asynchronous multisource measurements. The main contributions are as follows.

- 1) A new GNSS-inertial-LiDAR fusion SLAM framework with INS-centric multiple modalities by IESKF is proposed that not only achieves fast state estimation under multiple constraints with competitive accuracy, but also has strong robustness to tolerate sensor degradation and failure.
- 2) A fast and dynamic frame transformation method based on double sliding window for our filter-based navigation framework is developed that online aligns the global measurements from GNSS and the local world states of the robot.
- 3) A distributed backward propagation for motion compensation is designed to avoid point cloud fragmentation and underutilization, which makes full use of heterogeneous sensor data at different frequencies to ensure the system accuracy.

The remaining sections of this article are organized as follows. In Section II, related work is presented. Section III shows the system overview of the proposed method. The multisensor state estimation is given in Section IV. The solutions for sensor degradation and failure are described in Section V. The comprehensive experiments on public datasets and in the real

world are conducted in Section VI, followed by the conclusion in Section VII.

II. RELATED WORK

A. LiDAR-Inertial Odometry

LOAM [20] is a representative 3-D LiDAR SLAM proposed in 2014, and it is divided into frontend (solving high-frequency odometry by point cloud registration) and backend (optimizing and mapping in low frequency), facilitating the development of subsequent LiDAR (inertial) systems. LIO mapping [8] inherits the structure and couples IMU and LiDAR tightly in sliding window optimizer. LINS [10] is also derived from LOAM but achieving lightweight state estimator in the frontend by IESKF. FAST-LIO [11] is entirely based on IESKF without refining map points or states by optimizer. On this basis, FAST-LIO2 [21] registers points to the map without feature extraction, in order to make full use of environmental information and adapt to emerging LiDAR.

B. GNSS Fusion in LIO

GNSS with the global information is usually fused with LIO to improve the navigation system performance. LIO-SAM [22] is a LiDAR and IMU tightly coupled method based on factor graph optimization (FGO), and the GPS factor and loop closure factor are also designed to improve the global consistency. LIO-SAM has been widely applied and considered as the state-of-the-art (SOTA) fusion system, facilitating many studies with FGO as the core framework. In [23], LiDAR, IMU, wheel encoder, and GNSS are fused by sliding window FGO, where GNSS is also used to assist in refining accumulated mapping errors. Based on the similar framework, Zhong et al. [24] add outlier elimination for GNSS and LIO before fixed-time sliding window optimization. Compared to the optimization framework, filter-based fusion methods receive less attention in relative research. In [25], INS and GNSS are integrated with grid-based SLAM by Kalman filter and output 2-D pose, reducing the system drift by refreshing map strategy and increasing the reliability by cross validation of multisource data. P³-LINS [26] designed three-phase filter-based estimator and two initialization methods for frame transformation in frontend to fuse GNSS, INS, and LiDAR, but the back-end mapping is still based on optimization.

C. Multimodal SLAM

To be robust in complex real-world environments, the multimodal framework for SLAM has been focused on related research. Super Odometry [16] is an IMU-centric state estimator that incorporates multimodal data from LiDAR and vision to improve the system performance in perceptually degraded environments. Shu and Luo [27] take full use of multiple constraints from LiDAR and visual features, and introduce a multimodal mapping with different features, proving its effectiveness in the case of blurred images or LiDAR degradation. Compared to utilizing multimodality constraints to improve stability in harsh environments, the GPS-visual-inertial fusion [28] takes into account GPS data interruptions and compensates for cumulative drift by closed-loop-like manner.

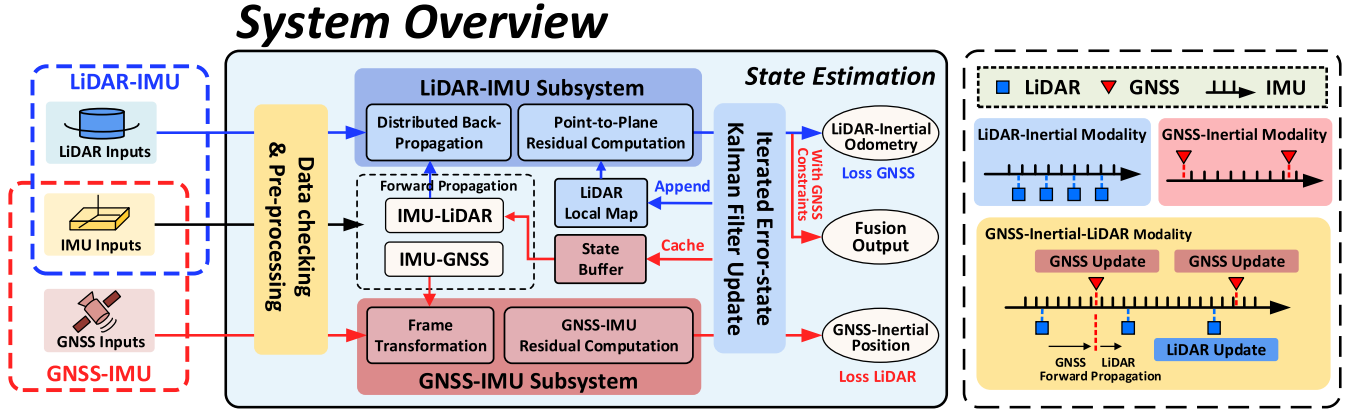


Fig. 1. System overview and the processing of each modality.

The failure or degradation of core sensors is also considered in some studies. VMO [13] based on RGB, thermal, and hyperspectral camera achieves motion estimation, and the multimodal approach can work in the presence of partial camera failures. MIMOSA [15] utilizes LiDAR points, visible-light and thermal vision fusion odometry, and IMU module to construct frontend, the versatile backend tracks the healthy modalities against the failure of partial sensors. LVI-SAM [29] and R²LIVE [30] design visual-inertial subsystem and LiDAR-inertial subsystem in a tightly coupled manner based on the FGO as the backend. Failure detection is developed in them to decouple subsystems in the case of LiDAR or camera degradation or failure. LIO-Fusion [31] utilizes GNSS, prior map, and wheel odometry to reinforce LIO, with the health checking module in FGO to evaluate potential failures of the sensors. Similarly, the healthy monitor and priority queue are designed in LiDAR odometry, which is integrated with multiple modal odometry and raw IMU data, exhibiting high accuracy and strong robust at the cost of huge computation [14]. The GNSS-LiDAR fusion system [32] develops two modes based on optimization for open scenes and GNSS-denied environment, taking into account the degradation of GNSS but neglecting LiDAR. Unlike the approaches of nonlinear optimization, Li et al. [33] implement multimodal grid-based SLAM based on extend Kalman filter (EKF), and GNSS/INS and LiDAR/INS are switched outdoors and indoors, but the global constraint from GNSS and sensor failure problems are not involved.

III. SYSTEM OVERVIEW

The pipeline of the proposed system is shown in Fig. 1. The sensor measurements from GNSS, IMU, and LiDAR are simultaneously input to the system at respective frequencies. After corresponding preprocessing and necessary checking, we take IMU as the core sensor and divide the overall framework into two subsystems for coarse-to-fine state estimation based on IESKF: GNSS-IMU subsystem and LiDAR-IMU subsystem. When the sensors are healthy, GNSS-IMU subsystem achieves coarse state estimation at GNSS frequency, improving the stability by adding global constraints. The results of the coarse estimation are not directly available as the system output but are temporarily cached. The updated states of GNSS are taken as the input of LiDAR-IMU subsystem, which further refines the system states. Finally, the system

states with GNSS constraints are output at the frequency of LiDAR, and the 3-D map with dense points is constructed.

Benefiting from the framework of alternative state update by the two subsystems, our system is featured with strong robustness. In the case of sensor failure or degradation, whether LiDAR or GNSS, the system can continue to operate as different modalities, ensuring that state estimation is not interrupted.

IV. MULTISENSOR STATE ESTIMATION

A. Definitions

The sensor frame, the ECEF frame, and the ENU frame are involved in this article. The sensor frame is used to describe the spatial location of the equipped sensors, including IMU frame $^i(\cdot)$, LiDAR frame $^l(\cdot)$, and GNSS frame $^g(\cdot)$. The first IMU frame is considered as the world frame $^w(\cdot)$, which is fixed after system operation, and the odometry and mapping output are also represented in the world frame. In contrast, the IMU frame is constantly changing as the system body moves in the world frame, also can be called the body frame $^b(\cdot)$. The ECEF frame $^E(\cdot)$ is a Cartesian coordinate system and fixed with the Earth, which is used to host the measurements of GNSS. The point of origin is located at the center of the Earth mass, the x -axis is aligned with the prime meridian, the z -axis is perpendicular to the Earth's equatorial plane toward the geographic North pole, and the y -axis follows the right-hand rule. In this article, we use the representation of WGS-84 in the ECEF frame. In addition, the ENU frame $^N(\cdot)$ is introduced to convert measurements from the ECEF frame to the world frame, in which the origin is based on the body and x -axis, y -axis, and z -axis are pointing to the East, North, and up directions, respectively.

For convenience, we assume the IMU is coincided with the system body and attached to other sensors rigidly with known extrinsic. The state vector $\mathbf{x}$, input $\mathbf{u}$, and noise $\mathbf{n}$ in this article are defined as

$$\begin{aligned} \mathcal{M} &= \text{SO}(3) \times \mathbb{R}^{15} \\ \mathbf{x} &= [{}^w\mathbf{R}_i^T {}^w\mathbf{p}_i^T {}^w\mathbf{v}_i^T {}^w\mathbf{b}_o^T {}^w\mathbf{b}_a^T {}^w\mathbf{g}^T]^T \in \mathcal{M} \\ \mathbf{u} &= [\mathbf{a}_m^T \boldsymbol{\omega}_m^T]^T \\ \mathbf{n} &= [\mathbf{n}_o^T \mathbf{n}_a^T \mathbf{n}_{ba}^T \mathbf{n}_{ba}^T]^T \end{aligned} \quad (1)$$

where ${}^w\mathbf{R}_l$ is the rotation matrix on $\text{SO}(3)$, ${}^w\mathbf{p}_i$ and ${}^w\mathbf{v}_i$ are the position and velocity, respectively, $\mathbf{b}_\omega$ and $\mathbf{b}_a$ are the bias from the gyroscope and accelerometer of the IMU, respectively, ${}^w\mathbf{g}$ is the gravity vector in the world frame that needs to be estimated [34], $\mathbf{a}_m$ and ω_m are the input from the acceleration and angular velocity of IMU, respectively, and $\mathbf{n}_\omega$, $\mathbf{n}_a$ and $\mathbf{n}_{b\omega}$, $\mathbf{n}_{ba}$ are the Gaussian white noise, affecting the measurements and bias, respectively.

In addition, we use the encapsulation operators $\boxplus$ and $\boxminus$ defined in [35] to parameterize the state error, which is on a compound manifold $\mathcal{M} = \text{SO}(3) \times \mathbb{R}^n$ with the following expression:

$$\begin{bmatrix} \mathbf{R} \\ \mathbf{a} \end{bmatrix} \boxplus \begin{bmatrix} \mathbf{r} \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{R}\text{Exp}(\mathbf{r}) \\ \mathbf{a} + \mathbf{b} \end{bmatrix}; \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{a} \end{bmatrix} \boxminus \begin{bmatrix} \mathbf{R}_2 \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \text{Log}(\mathbf{R}_2^T \mathbf{R}_1) \\ \mathbf{a} - \mathbf{b} \end{bmatrix} \quad (2)$$

where $\mathbf{r} \in \mathbb{R}^3$, $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$, $\text{Exp}(\mathbf{r})$ is the exponential map in [35], and $\text{Log}(\cdot)$ is its inverse map.

B. Forward Propagation

When the measurements from GNSS or LiDAR are received, the accumulated IMU data are propagated forward until it is synchronized with the measurements. The forward propagation of IMU can be represented by its discrete-time kinematic model, in which the noise is set to zero

$$\hat{\mathbf{x}}_{\rho+1} = \begin{bmatrix} {}^w\mathbf{R}_{i,\rho}\text{Exp}[(\omega_{m,\rho} - \mathbf{b}_{\omega,\rho})\Delta t] \\ {}^w\mathbf{p}_{i,\rho} + {}^w\mathbf{v}_{i,\rho}\Delta t + \frac{1}{2}[{}^w\mathbf{R}_{i,\rho}\mathbf{a}'_{m,\rho} + {}^w\mathbf{g}]\Delta t^2 \\ {}^w\mathbf{v}_{i,\rho} + {}^w\mathbf{R}_{i,\rho}\mathbf{a}'_{m,\rho}\Delta t + {}^w\mathbf{g}\Delta t \\ \mathbf{b}_{\omega,\rho} \\ \mathbf{b}_{a,\rho} \\ {}^w\mathbf{g}_\rho \end{bmatrix} \quad (3)$$

where $\hat{\mathbf{x}}_{\rho+1}$ is the predicted state at $t_{\rho+1}$ between the updated states, ${}^w\mathbf{R}_{i,\rho}$, ${}^w\mathbf{p}_{i,\rho}$, and ${}^w\mathbf{v}_{i,\rho}$ are the rotation, position, and velocity of the body at t_ρ in the world frame, $\mathbf{a}'_{m,\rho}$ is the $\mathbf{a}_{m,\rho}$ without $\mathbf{b}_{a,\rho}$ and $\mathbf{n}_a$, and Δt is the time between t_ρ and $t_{\rho+1}$.

We implement our fusion SLAM framework entirely based on IESKF; thus, the kinematic model of the error state $\tilde{\mathbf{x}}_{\rho+1}$ and covariance $\hat{\mathbf{P}}_{\rho+1}$ are used to drive the process of state estimation and can be described as

$$\begin{aligned} \tilde{\mathbf{x}}_{\rho+1} &= \mathbf{x}_{\rho+1} \boxminus \hat{\mathbf{x}}_{\rho+1} \\ &= \mathbf{F}_{\tilde{\mathbf{x}}}\tilde{\mathbf{x}}_\rho + \mathbf{F}_w\mathbf{n}_\rho \\ \hat{\mathbf{P}}_{\rho+1} &= \mathbf{F}_{\tilde{\mathbf{x}}}\hat{\mathbf{P}}_\rho\mathbf{F}_{\tilde{\mathbf{x}}}^T + \mathbf{F}_w\mathbf{Q}_\rho\mathbf{F}_w^T \end{aligned} \quad (4)$$

where $\mathbf{F}_{\tilde{\mathbf{x}}}$ and $\mathbf{F}_w$ are the transition matrices of error state and noise, the readers can refer to [11] for more details.

C. GNSS-Inertial Subsystem

1) *Online Frame Transformation:* The measurements of GNSS are sampled in the ECEF frame while the states are estimated in the world frame; thus, online frame transformation needs to be performed before updating states. We dynamically maintain the transformation during system operation at a low time cost, rather than fixing it after initialization. The ENU

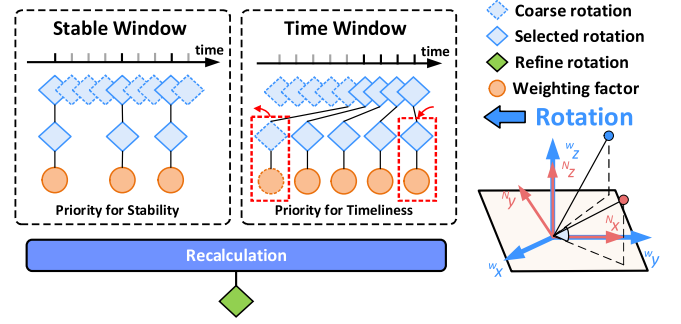


Fig. 2. Rotation refinement based on double sliding window.

frame is introduced to bridge the ECEF frame and the world frame [22], [36]. The ENU station center is set according to the predicted state, and the rotation (yaw offset) between the world frame and the ENU frame is calculated coarsely by subsequent GNSS measurements and predicted states

$$\begin{aligned} {}^w\mathbf{P}_{N,k+1} &= {}^w\mathbf{P}_{N,k} = {}^w\hat{\mathbf{p}}_{i,g0} \\ {}^w\mathbf{R}_{N,k+1}^c &= \mathbf{I} + \|\mathbf{p}_{c,k}\| [\mathbf{p}_{c,k}]_{\wedge} + (1 - \mathbf{p}_{d,k})[\mathbf{p}_{c,k}]_{\wedge}^2 \\ \mathbf{p}_{c,k} &= {}^w\hat{\mathbf{p}}_{i,k,\text{proj}}^n \times {}^N\mathbf{p}_{g,k,\text{proj}}^n \\ \mathbf{p}_{d,k} &= {}^w\hat{\mathbf{p}}_{i,k,\text{proj}}^n \cdot {}^N\mathbf{p}_{g,k,\text{proj}}^n \end{aligned} \quad (5)$$

where ${}^w\mathbf{P}_{N,k+1}$ and ${}^w\mathbf{R}_N^c$ are the position and coarse rotation from the ENU frame to the world frame, respectively, ${}^w\hat{\mathbf{p}}_{i,g0}$ is the predicted position at the time of first GNSS measurement in the world frame, $[\mathbf{p}_{c,k}]_{\wedge}$ is the skew-symmetric matrix of $\mathbf{p}_{c,k}$, and ${}^w\hat{\mathbf{p}}_{i,k,\text{proj}}^n$ and ${}^N\mathbf{p}_{g,k,\text{proj}}^n$ are the predicted position and GNSS measurements in the world frame after projection and normalization, respectively.

Then, we construct a double sliding window based on the coarse rotation for further refinement. As shown in Fig. 2, sliding windows are organized into two types: stable window and time window. The stable window holds the n_s rotations solved from the most stable measurements, and the time window holds the recent n_t rotations ($n_s \leq 3$ and $n_t \leq 5$ in this article). The stability of measurements is determined by the covariance and the angle offsets

$$\begin{aligned} s_c &= \frac{1}{(e_x + e_y + e_z)/3} \\ s_d &= \left| \text{acos} \frac{{}^N\mathbf{v}_k \cdot {}^N\mathbf{v}_{k-1}}{\|{}^N\mathbf{v}_k\| \|{}^N\mathbf{v}_{k-1}\|} - \text{acos} \frac{{}^w\mathbf{v}_k \cdot {}^w\mathbf{v}_{k-1}}{\|{}^w\mathbf{v}_k\| \|{}^w\mathbf{v}_{k-1}\|} \right| \\ {}^N\mathbf{v}_k &= {}^N\mathbf{p}_{g,k} - {}^N\mathbf{p}_{g,k-1}, \quad {}^w\mathbf{v}_k = {}^w\hat{\mathbf{p}}_k - {}^w\hat{\mathbf{p}}_{k-1} \end{aligned} \quad (6)$$

where s_c is the confidence of the coarse rotation, e_x is the eigenvalue of the GNSS covariance on x -axis, s_d is the angle offsets between vector differences, and ${}^N\mathbf{p}_{g,k}$ and ${}^w\hat{\mathbf{p}}_k$ are the positions at time t_k in the ENU frame from GNSS measurements and in the world frame from the predicted states.

Based on s_c and s_d , we compute the weight factor w_i of each rotation in double sliding window. The refined rotation ${}^w\mathbf{R}_{N,k}^r$, which is computed by ${}^w\mathbf{R}_{N,i}^c$ and the respective w_i , is used to transform the sequent GNSS measurements in the ENU frame. Notice that ${}^w\mathbf{R}_{N,k}^r$ will be fixed until the difference between the world-frame GNSS and the predicted state is beyond a

threshold. Then, we can get the GNSS measurements $^w \mathbf{p}_k$ in the world frame. This process can be summarized as

$$\begin{aligned} w_i &= \frac{S_{c,i}}{2 \sum_{j=1}^n S_{c,j}} + \frac{S_{d,i}}{2 \sum_{j=1}^n S_{d,j}} \\ ^w \mathbf{R}_{N,k}^r &= \sum_{i=1}^{n_s} w_i ^w \mathbf{R}_{N,i}^c + \sum_{j=1}^{n_t} w_j ^w \mathbf{R}_{N,j}^c \\ ^w \mathbf{p}_{k+1} &= ^w \mathbf{R}_{N,k}^r \mathbf{p}_{k+1} + ^w \mathbf{P}_{N,k}. \end{aligned} \quad (7)$$

2) *Iterated Update by GNSS*: After converting the measurements to the world frame, we perform the state update of GNSS by IESKF. The GNSS-IMU residual is calculated by the position of predicted state and GNSS measurements in the world frame. The measurement model h_g is represented as

$$\begin{aligned} 0 &= h_g(\mathbf{x}_k, \mathbf{n}_g) \\ &\simeq h_g(\hat{\mathbf{x}}_k^\kappa, 0) + \mathbf{H}_g^\kappa \tilde{\mathbf{x}}_k^\kappa + \mathbf{v}_g \\ &= {}^i \mathbf{T}_g ({}^w \mathbf{T}_{N,k-1} {}^N \mathbf{T}_{E,k} {}^E \mathbf{p}_{g,k}) - {}^w \hat{\mathbf{p}}_{i,k}^\kappa + \mathbf{H}_g^\kappa \tilde{\mathbf{x}}_k^\kappa + \mathbf{v}_g \end{aligned} \quad (8)$$

where $\mathbf{H}_g^\kappa$ is the Jacobian matrix of the measurement model for the error state, $\mathbf{H}_g$ for short, $\mathbf{v}_g \in \mathcal{N}(0, \mathbf{R}_g)$ is the measurement noise from GNSS, ${}^i \mathbf{T}_g$ is the extrinsic matrix from GNSS to IMU, ${}^w \mathbf{T}_N$ and ${}^N \mathbf{T}_E$ are the transformation from the ENU frame to the world frame and from the ECEF frame to the ENU frame, and ${}^E \mathbf{p}_{g,k}$ and ${}^w \hat{\mathbf{p}}_{i,k}^\kappa$ are the measurement position from GNSS in the ECEF frame and the predicted position of the κ th iteration in the world frame, respectively.

Then, the system states are updated iteratively by the following formulas:

$$\begin{aligned} \mathbf{K}_g &= \hat{\mathbf{P}}_k \mathbf{H}_g^\top (\mathbf{H}_g \hat{\mathbf{P}}_k \mathbf{H}_g^\top + \mathbf{R}_g)^{-1} \\ \hat{\mathbf{x}}_k^{\kappa+1} &= \hat{\mathbf{x}}_k^\kappa \boxplus \tilde{\mathbf{x}}_k \\ &= \hat{\mathbf{x}}_k^\kappa \boxplus \mathbf{K}_g (-h_g(\hat{\mathbf{x}}_k^\kappa)) - (\mathbf{I} - \mathbf{K}_g \mathbf{H}_g) (\hat{\mathbf{x}}_k^\kappa \boxplus \hat{\mathbf{x}}_k^0). \end{aligned} \quad (9)$$

When the state converges or the maximum iterations are reached, the updated state and the posterior covariance at time t_k can be calculated

$$\bar{\mathbf{x}}_k = \hat{\mathbf{x}}_k^{\kappa+1}, \quad \bar{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_g \mathbf{H}_g) \hat{\mathbf{P}}_k. \quad (10)$$

D. LiDAR-Inertial Subsystem

1) *Distributed Backward Propagation*: Backward propagation and motion compensation can cope with the point distortion to improve the accuracy. In the fusion system, GNSS measurements trigger the predicted state to propagate forward from t_{k-1} to t_k , as shown in Fig. 3. When the LiDAR measurement at time t_{k+1} arrives, the predicted states are propagated from t_k to t_{k+1} . However, the scanned points are spread over the time t_{k-1} to t_{k+1} . The absence of the predicted states leads to the partial distortion points not being corrected, reducing the system accuracy. For dedistortion of all the points, we cache the previous predicted states and perform distributed backward propagation.

The process of distributed backward propagation is shown in Fig. 3. After GNSS updating, a temporary state buffer will be constructed to save all the predicted states of GNSS from t_{k-1} to t_k . When performing LiDAR update, the state with GNSS constraints will be taken out from the state buffer and used as

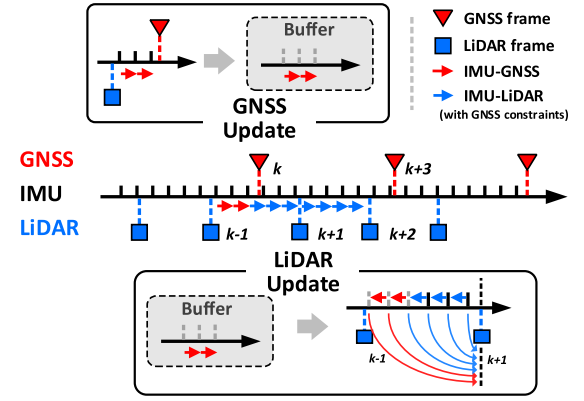


Fig. 3. Process of distributed backward propagation.

the initial state for the forward propagation of LiDAR. When this process is backward propagated, the predicted states used for GNSS from t_{k-1} to t_k are placed before the initial state. Then, all the predicted states from t_{k-1} to t_{k+1} are used for motion compensation. For the points $\mathbf{p}_{l,\gamma}$ at t_γ between t_{k-1} and t_{k+1} , the compensated points $\mathbf{p}_{l,k+1}$ can be represented as

$$\mathbf{p}_{l,k+1} = {}^i \mathbf{T}_l^{-1} ({}^w \mathbf{T}_i^{-1} {}^{k+1} \mathbf{T}_\gamma {}^w \mathbf{T}_i) {}^i \mathbf{T}_l \mathbf{p}_{l,\gamma} \quad (11)$$

where ${}^i \mathbf{T}_l$ is the extrinsic from the LiDAR frame to the IMU frame, ${}^w \mathbf{T}_i$ is the transformation from the IMU frame to the world frame, and ${}^{k+1} \mathbf{T}_\gamma$ is the relative transformation from t_γ to t_{k+1} , obtained by the IMU-GNSS predicted states when $\gamma \in (t_{k-1}, t_k)$ and by the IMU-LiDAR predicted states when $\gamma \in (t_k, t_{k+1})$.

2) *Iterated Update by LiDAR*: After distributed backward propagation, all the sampled points of the LiDAR are compensated. We follow [21] to perform direct scan-to-map registration without feature extraction and update states by LiDAR. The measurement model is

$$\begin{aligned} 0 &= h_j(\mathbf{x}_k, \mathbf{n}_l) \\ &\simeq h_j(\hat{\mathbf{x}}_k^\kappa, 0) + \mathbf{H}_j^\kappa \tilde{\mathbf{x}}_k^\kappa + \mathbf{v}_{l,j} \\ &= {}^w \mathbf{G}_j ({}^w \mathbf{T}_i {}^i \mathbf{T}_l \mathbf{p}_{j,k+1} - {}^w \mathbf{q}_j) + \mathbf{H}_j^\kappa \tilde{\mathbf{x}}_k^\kappa + \mathbf{v}_{l,j} \end{aligned} \quad (12)$$

where $\mathbf{p}_{j,k+1}$ is the j th point at time t_{k+1} , ${}^w \mathbf{q}_j$ is the centroid of the small patch, which is fit by the closest points, ${}^w \mathbf{G}_j$ is the normal vector of the corresponding patch, $\mathbf{H}_j^\kappa$ is the Jacobian matrix of the measurement model for the error state, and let $\mathbf{H}_l = [\mathbf{H}_1^\top, \dots, \mathbf{H}_m^\top]^\top$, ${}^w \mathbf{T}_i$ and ${}^i \mathbf{T}_l$ are the transformation from the IMU frame to the world frame and the precalibrated parameters from LiDAR frame to the IMU frame, and $\mathbf{v}_{l,j} \in \mathcal{N}(0, \mathbf{R}_l)$ is the measurement noise of $\mathbf{p}_{j,k+1}$.

Then, calculate the Kalman gain $\mathbf{K}_l$ and perform update [21]

$$\begin{aligned} \mathbf{K}_l &= (\mathbf{H}_l^\top \mathbf{R}_l^{-1} \mathbf{H}_l + \mathbf{P}_k^{-1})^{-1} \mathbf{H}_l^\top \mathbf{R}_l^{-1} \\ \bar{\mathbf{x}}_k^{\kappa+1} &= \hat{\mathbf{x}}_k^\kappa \boxplus \tilde{\mathbf{x}}_k \\ &= \hat{\mathbf{x}}_k^\kappa \boxplus \mathbf{K}_l (-h_l(\hat{\mathbf{x}}_k^\kappa)) - (\mathbf{I} - \mathbf{K}_l \mathbf{H}_l) (\hat{\mathbf{x}}_k^\kappa \boxplus \hat{\mathbf{x}}_k^0). \end{aligned} \quad (13)$$

The refined states with the covariance will be taken as the output of the fusion odometry after convergence or the maximum iterations reached

$$\bar{\mathbf{x}}_k = \hat{\mathbf{x}}_k^{\kappa+1}, \quad \bar{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_l \mathbf{H}_l) \hat{\mathbf{P}}_k. \quad (14)$$

V. SOLUTIONS FOR DEGRADATION AND FAILURE

A. Sensor Degradation Scenarios

GNSS measurements usually fluctuate significantly in occluded or indoor areas, resulting in the poor accuracy. To avoid unhealthy data, the measurement covariance and its difference from predicted states in the world frame are used to check GNSS degradation. We discard the data with low confidence before preprocessing. For the inaccuracy GNSS data that differs significantly from predicted states, they are not used for state update. The conditions for degradation checking are described as

$$\begin{aligned} \|\Sigma_{g,k}\|_2 &< c_{th}, \quad \sigma_{g_z,k}^2 < z_{th} \\ |{}^i\mathbf{T}_g^w \mathbf{p}_{g,k} - {}^w\hat{\mathbf{p}}_{i,k}| &> p_{th} \end{aligned} \quad (15)$$

where $\Sigma_{g,k}$ and $\sigma_{g_z,k}^2$ are the covariance matrix and the variance on the z -axis of GNSS measurements, respectively, c_{th} and z_{th} are the thresholds ($c_{th} = 45$ and $z_{th} = 50$ in this article), ${}^w\mathbf{p}_{g,k}$ and ${}^w\hat{\mathbf{p}}_{i,k}$ are the GNSS measurement and the position of the predicted state, respectively, and p_{th} is the corresponding threshold (2.5 m in this article).

In outdoor environments, one reason for LiDAR degradation is due to the openness of the scene, while it usually contributes to the health of GNSS measurements. Therefore, we leverage LiDAR degradation detection to skip degraded frames. Referring to [29] and [37], the smallest eigenvalue of $\mathbf{H}_l^T \mathbf{H}_l$ in the first iteration is set as the condition of LiDAR degradation. Due to the multimodal pipeline and healthy GNSS data in open scenarios, the system can maintain localization even in the presence of continuous LiDAR degradation.

B. GNSS Failure Scenarios

When GNSS measurements are lost, the system will cross GNSS processing (including frame transformation and GNSS update) and rely directly on the LiDAR-IMU subsystem. In this modality (LIO), the forward propagation is performed directly between the LiDAR scans and backward propagation is not performed in the distributed form. The system states and point cloud maps are output at LiDAR frequency but lack the global constraints of GNSS. In addition, the station center of ENU frame and the transformation to the world frame are temporarily retained. If the GNSS signal is restored, this information will be used again for the GNSS-IMU subsystem without reinitialization.

C. LiDAR Failure Scenarios

To check whether LiDAR is failing, abnormal data detection and the time t_{chk} between the current GNSS and last LiDAR are developed in our system. When anomalous points are sampled or t_{chk} is above a certain threshold (setting according to the frequency of LiDAR, 0.2 s in our system for 10-Hz LiDAR), that means LiDAR measurements are lost

$$t_{chk,k} = t_{g,k} - t_{l,end} < \alpha \quad (16)$$

where $t_{g,k}$ is the sampling time of the current GNSS, and $t_{l,end}$ is the time of the last point in the recent LiDAR measurement.

TABLE I
PUBLIC DATASET SPECIFICATIONS

Datasets	Environments	Sensors
M2DGR	Campus/ Streets	Velodyne VLP-32C LiDAR (200 m range, 360°×40° FoV, 10 Hz) u-blox M8T (GPS/Beidou, 4 Hz) Handsfree IMU (150 Hz) GT: GNSS/RTK (within 2 cm accuracy)
		Velodyne HDL-32E LiDAR (80 m range, 360°×40° FoV, 10 Hz) u-blox EVK-M8T (1 Hz) Xsens Mt10 IMU (100 Hz) GT: GNSS/RTK (within 5 cm accuracy)
UrbanNav	Urban	Velodyne VLP-16 LiDAR (100 m range, 360°×30° FoV, 10 Hz) MicroStrain 3DM-GX5-25 IMU Reach M GPS

When facing LiDAR failure, the states are only propagated forward to the GNSS measurement, and the distributed backward propagation will also be paused to perform the state update of GNSS separately. The system is operating as a GNSS-inertial odometry. In this modality, the estimated states will be output directly after GNSS coarse update instead of caching. In addition, the frame transformation between the ECEF and the world is fixed due to the instability of GNSS measurements. Notice that the map constructed by points will not be updated because of the failure of LiDAR.

VI. EXPERIMENTS

A. Datasets and Implementation

To verify the effectiveness of the proposed system, comprehensive experiments are conducted on the latest public datasets and tested in the real world. We perform qualitative and quantitative analysis on accuracy, efficiency, and robustness, respectively.

The latest public datasets are challenging and contain accurate ground truth, which contributes to evaluate with other navigation systems quantitatively. The public challenging datasets we selected are from the work of M2DGR [38], UrbanNav [39], and LIO-SAM [22], which contain multisensor data and different scenarios. The dataset specifications are shown in Table I. We also set up an unmanned ground vehicle (UGV) equipped with the sensor suite for the real-world experiments, as shown in Fig. 4. The sensors include a Velodyne VLP-16 LiDAR (360° × 30° field of view (FoV), 100-m detectable range, 10 Hz), a JY901 IMU with 200 Hz, and a u-blox M8T GNSS receiver (SPP Solution, Qingdao, China, GPS and Beidou model, 4 Hz) with a low-cost antenna. For spatial-temporal synchronization, the extrinsic is fixed by the mechanical module equipped with multiple sensors, and the temporal interpolation is conducted in our system without hardware-triggered alignment.

The implementation of our system is based on C++ and under the framework of robot operating system (ROS). The experiments both on the datasets and in the real world are carried out on a laptop (Intel¹ i7-7700HQ CPU and 8-GB

¹Registered trademark.

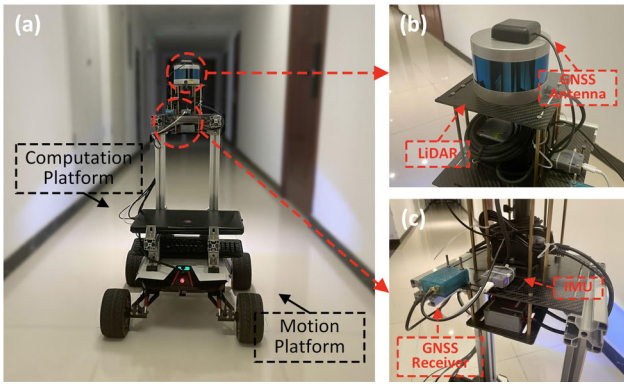


Fig. 4. Equipment configuration used in our real-world testing. (a) UGV for testing. (b) LiDAR and GNSS antenna. (c) GNSS receiver and IMU.

RAM) with Ubuntu 18.04. In addition, we compare our system with FAST-LIO2 [21], LIO-SAM (without closed-loop correction) [22], and LINS [10], which are the open-source SOTA fusion methods. For the sake of fairness, the experimental parameters follow the recommended configuration for each method. For our system, we set 1000-m local map, 0.5-m voxel filter, three-point skip operation, three-LiDAR max iteration, and ten-GNSS max iteration, which are also the same as the other tested system (except for GNSS iterations).

B. Accuracy Evaluation

A total of 11 sequences from the above public datasets are used for evaluating the system accuracy. We use the root-mean-square error (RMSE) and standard deviation (STD) of absolute pose error (APE), which contribute to evaluating the system overall accuracy and error fluctuations. All the estimated trajectories are output in the world frame while the ground truth is in the ECEF frame; thus, the trajectory alignment needs to be further performed before evaluating.

The experimental results of accuracy evaluation are shown in Table II. According to the results, it can be seen that on five of 11 sequences, the proposed method achieves the best accuracy in terms of the RMSE of APE. On the remaining of six sequences, FAST-LIO2 and GPS-enabled LIO-SAM each achieve better performance on three sequences, but our results are still competitive on these sequences. As for the STD of errors, our system achieves the same or lower error fluctuations with other methods in the six of 11 sequences, showing the satisfying stability. The results of trajectory comparison for partial sequences can be seen visually in Fig. 5. When all the sensors work well, multiple constraints are utilized in the fusion odometry to achieve competitive accuracy and stability.

Furthermore, we perform ablation experiments to test the effectiveness of distributed backward propagation. The methods applied by backward propagation and distributed backward propagation are compared, the results are shown in Fig. 6, and Fig. 7 presents more details on sequences gate01 and gate02. In most of the sequences, the experimental results reflect the positive effect of distributed backward propagation on accuracy. Due to the asynchronism between GNSS and LiDAR, the method without distributed backward propagation fails to fully utilize the predicted states to compensate for

TABLE II
APE (RMSE AND STD, m)

Seq ID	LINS	FAST-LIO2	LIO-SAM	LIO-SAM*	Proposed
gate01	0.15 (0.06)	0.18 (0.06)	0.61 (0.45)	0.15 (0.06)	0.14 (0.05)
gate02	0.33 (0.18)	0.33 (0.18)	0.33 (0.19)	0.33 (0.19)	0.31 (0.18)
street01	1.71 (0.93)	0.32 (0.12)	7.85 (5.93)	0.49 (0.20)	0.36 (0.14)
street02	5.43 (3.07)	2.81 (1.68)	3.96 (2.44)	1.22 (0.70)	2.67 (1.55)
street03	0.14 (0.05)	0.19 (0.06)	0.14 (0.06)	0.14 (0.06)	0.13 (0.05)
street04	2.65 (1.18)	0.46 (0.22)	0.88 (0.46)	2.36 (1.20)	0.51 (0.24)
street05	3.57 (1.78)	0.35 (0.13)	0.43 (0.16)	0.50 (0.21)	0.34 (0.12)
street06	1.72 (0.77)	0.39 (0.11)	0.40 (0.15)	0.76 (0.26)	0.42 (0.11)
street08	0.21 (0.09)	0.23 (0.09)	0.66 (0.55)	0.19 (0.09)	0.18 (0.06)
hk0314	1.88 (0.68)	1.62 (0.57)	1.77 (0.68)	1.37 (0.48)	1.60 (0.53)
hk0428	4.46 (1.30)	4.31 (1.33)	7.07 (2.75)	2.45 (1.04)	4.72 (1.18)

* LIO-SAM with the GPS factor enabled.

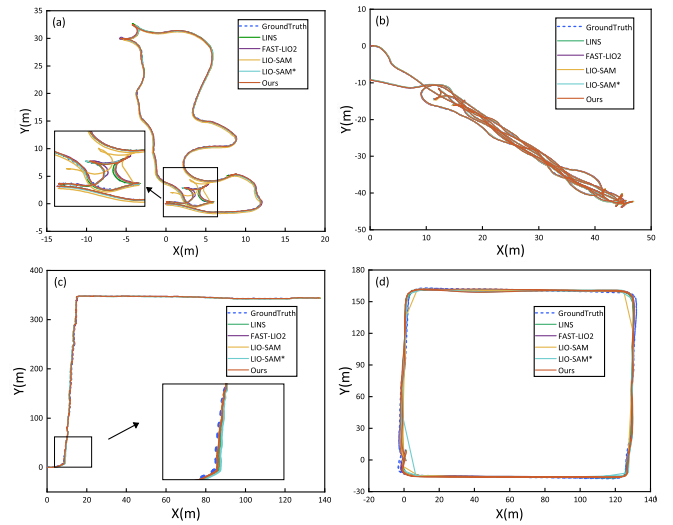


Fig. 5. Trajectories of tested methods on partial sequences: (a) gate01 sequence; (b) street03 sequence; (c) street06 sequence; and (d) ulhk0314 sequence.

motion distortion. This further makes only partial points can be compensated, rather than all of them. In street04 and street06, we think the reason for the loss of accuracy caused by distributed backward propagation is the quality of points. Unstructured features and dynamic objects in the environments increase the incorrect points, so in this case, reducing the points will instead improve the accuracy.

C. Efficiency Analysis

In this section, the same configurations are implemented to analyze the overall efficiency and the performance of fast and dynamic frame transformation. A total of 13 sequences are selected in this testing. For the SOTA methods involved in this

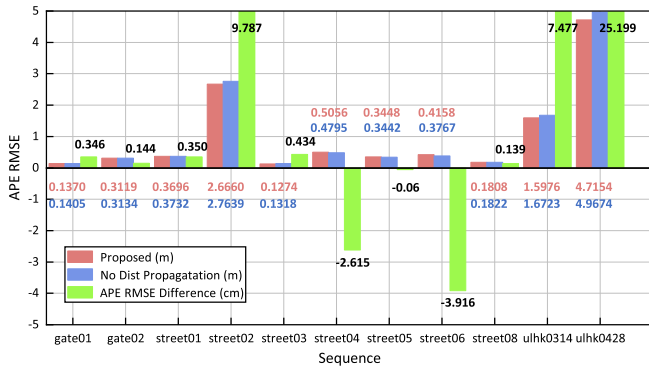


Fig. 6. Accuracy comparison of distributed backward propagation.

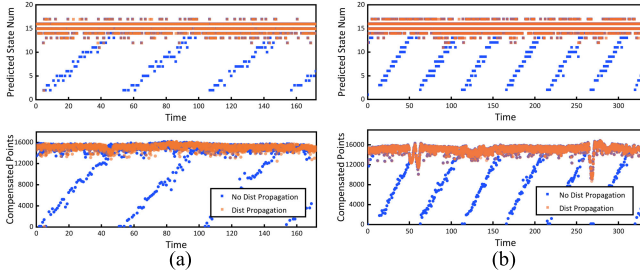


Fig. 7. Number of predicted states involved in backward propagation and the number of compensated points: (a) gate01 sequence and (b) gate02 sequence.

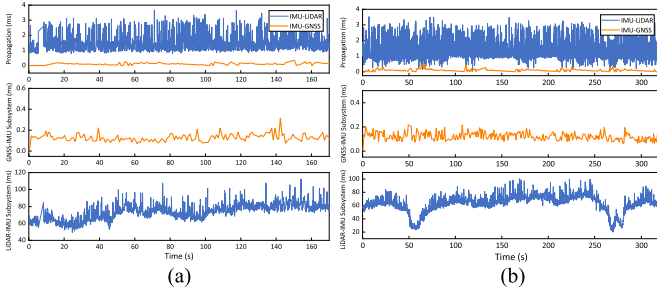


Fig. 8. Time consumption of different subsystems on sequences: (a) on the sequences of gate01 and (b) on the sequences of gate02.

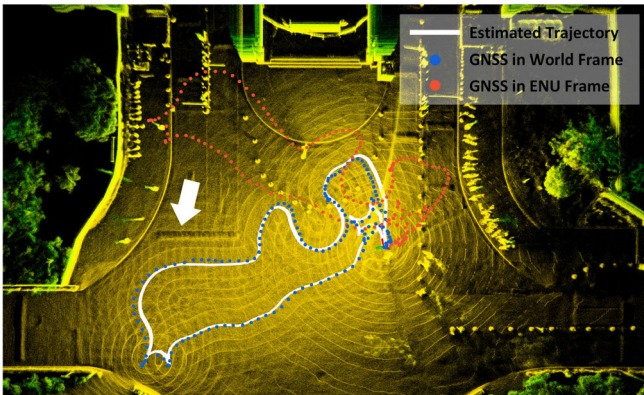


Fig. 9. Transformation of GNSS measurements between the ENU frame and the world frame.

testing, LIO-SAM (optimizer), LINS (filter-based frontend and optimizer-based backend), and FAST-LIO2 (filter-based) are based on different architectures. This allows the comparison with the proposed method in a visual way.

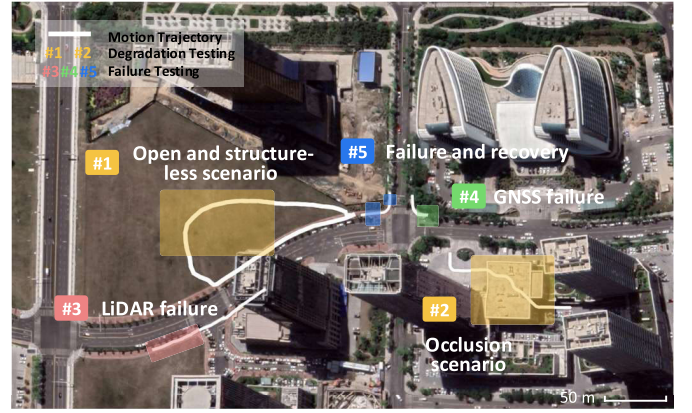


Fig. 10. Illustration of the real-world test sites. The areas covered by the yellow indicate the degraded scenes of LiDAR or GNSS. The red, green, and blue indicate the sensor failure situations.



Fig. 11. Environments of the real-world test site. (a) Center of the grass (open and structure-less). (b) Edge of the grass. (c) Test site of GNSS failure. (d) Test site of GNSS degradation.

The average time cost of the above methods is calculated for efficiency analysis, including data preprocessing, pose computation, and mapping. The results are shown in Table III, LIO-SAM (about 226 ms with GPS enabled and about 203 ms without GPS) has the largest processing time cost, and LINS is slightly more efficient (about 112 ms) because of the filter-based frontend. Although the best efficiency is achieved by FAST-LIO2 (about 46 ms), the proposed method is still competitive. We achieve an average time cost of 56 ms, which demonstrates a significant advantage over the optimizer-based methods. Further, we analyze the time consumption of different subsystems, as shown in Fig. 8. Affected by processing large amounts of points, the LiDAR-IMU subsystem consumes the majority of the computational resources, while the GNSS-IMU subsystem is faster due to the sparseness of data.

To verify the performance of frame transformation, the time usage for each step of the frame transformation on the sequence gate01 (duration 172 s) is calculated in Table IV. The maintenance of the sliding windows takes up most of the computational resources for frame transformation. However, the total time consumption is still small, with the total

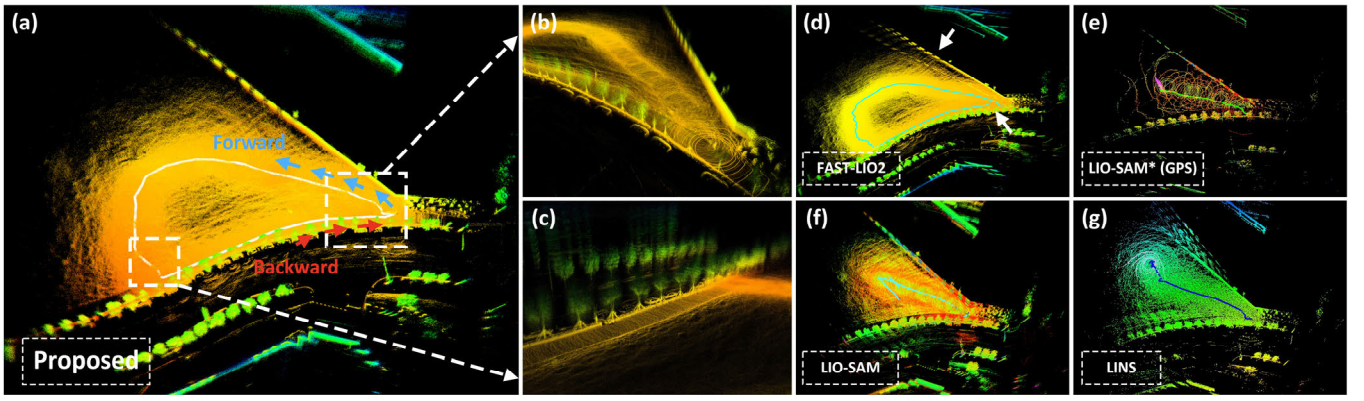


Fig. 12. Estimated trajectories and mapping results in LiDAR degraded scenario. (a) Results of the proposed method. (b) Mapping details at the start position. (c) Mapping details at the midway position. (d) Results of FAST-LIO2. (e) Results of LIO-SAM* (GPS factor enabled). (f) Results of LIO-SAM. (g) Results of LINS.

TABLE III
AVERAGE TIME COST PER PROCESSING (ms)

Seq ID	LINS	FAST-LIO2	LIO-SAM	LIO-SAM*	Proposed
gate01	103.13	53.08	192.48	246.49	62.49
gate02	95.89	48.51	231.17	240.92	57.42
street01	111.14	47.19	243.69	265.26	59.59
street02	92.04	44.34	144.90	196.39	50.66
street03	113.74	50.95	272.12	313.60	66.05
street04	88.49	45.46	210.96	198.56	53.76
street05	93.62	41.37	124.80	167.92	48.84
street06	94.00	46.11	111.02	180.52	51.80
street07	116.69	49.56	312.99	304.14	62.99
street08	116.13	51.35	326.41	336.85	66.31
hk0314	132.88	44.29	158.96	162.33	52.06
hk0428	138.10	49.18	182.34	186.31	59.13
park01	169.37	29.85	130.61	134.64	37.26

* LIO-SAM with the GPS factor enabled.

TABLE IV
TIME USAGE FOR EACH STEP OF FRAME TRANSFORMATION (ms)

Stage (s)	Setting Position	Sliding Windows	Weighting Computation	Number of Adjustment
0~50	0.0014	0.2023	0.0587	34
50~100	0	0.0970	0.0403	17
100~150	0	0.1973	0.0547	41
150~172	0	0.0424	0.0153	11
Total	0.0014	0.539	0.1691	103

maintenance cost of less than 1 ms on 172-s sequence. Fig. 9 provides the results of GNSS measurement transformation between ENU frame and the world frame, demonstrating that the accuracy frame transformation is achieved at a low time cost.

D. Robustness Test in Real World

Most public datasets cannot reflect the complexity of real environments due to the lack of sensor degradation and failure situations. Therefore, real-world experiments are performed to test the strong robustness of the proposed system. The experimental sites are chosen in the urban areas, which include degraded scenarios and failure situations of each sensor, as shown in Fig. 10. We perform the degradation testing in



Fig. 13. Experiment in GNSS degraded scene (the bottom right panel is the corresponding status of available satellites). (a) Estimated trajectory and mapping results. (b) Utilization of GNSS measurements in the world frame.

open and structure-less scenario and occlusion scenario for LiDAR and GNSS, respectively. In addition, the failure testing of LiDAR and GNSS is also performed in this section, where the failure of sensors is happened during the system operation. The specific environments of the tests can be shown in Fig. 11.

1) *Degraded Scenarios*: We test our system in degraded scenarios of LiDAR and GNSS, which contain open and structure-less grass, and the occlusion area by buildings. In the LiDAR degraded scenario, we drive the UGV to the center of the grass and return to the start position along the sidewalk, passing the typical degraded scenario. In addition, FAST-LIO2, LIO-SAM, and LINS, which are the SOTA methods, are also compared with the proposed method. The estimated trajectories and mapping results can be seen in Fig. 12. It shows that our system exhibits more accurate localization and mapping performance than other systems when LiDAR faces degradation. The estimated trajectory is closed-loop and

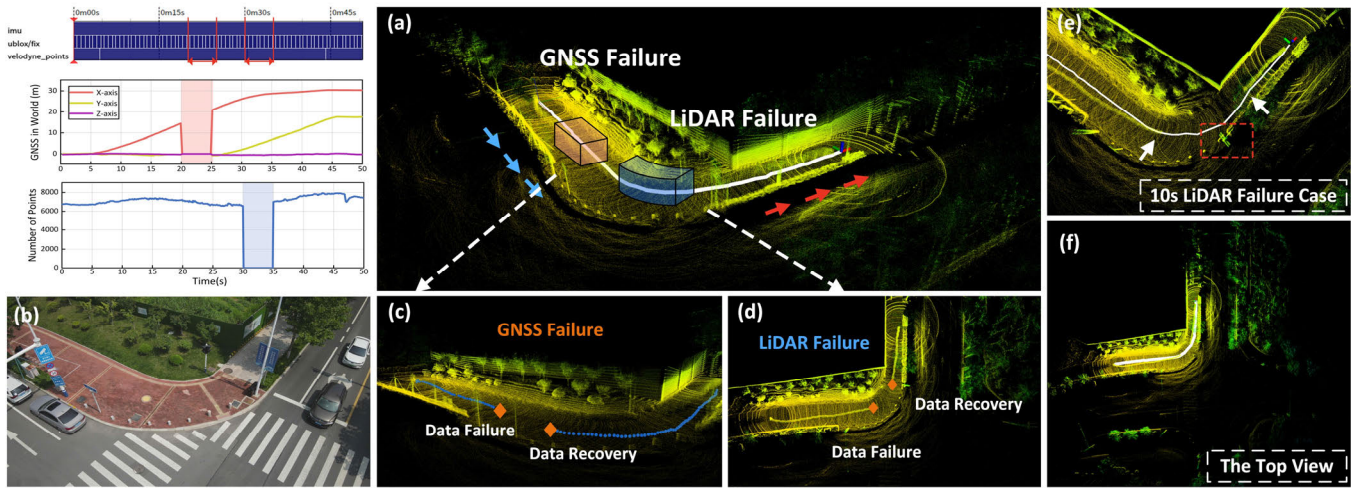


Fig. 14. Experiment in the situation of data anomaly and recovery. (a) Estimated trajectory and mapping results in the experiment. (b) Real-world testing site. (c) Details of GNSS data failure and recovery. (d) Details of LiDAR data failure and recovery. (e) 10-s LiDAR failure case. (f) Top view of mapping and trajectory.

corresponds to the real driving trajectory. When the UGV returns to the start position after passing through the grass, the right wall [see Fig. 12(b)] is rebuilt again, and the global map remains consistent. In contrast, FAST-LIO2 also completes this degradation testing, but the mapping deformation and unclosed-loop trajectory caused by accumulated drift can be inspected visually at the white arrows in Fig. 12(d). In addition, LIO-SAM (GPS enabled and without GPS) and LINS fail the degradation testing, which are the feature-based methods. The features of points are difficult to extract when the UGV is traveled to the center of grass. Enabling the GPS factor contributes to the long-term performance of LIO, but is not sufficient to cope with the severe degradation of LiDAR.

Then, we test our system in the semioccluded environment to verify the robustness to GNSS degradation. The UGV starts in an open square and crosses an occlusion area where the GNSS measurements are instability. The number of available satellites, the corresponding position dilution of precision (PDOP), and the average signal-to-noise ratio (SNR) are shown in the bottom right of Fig. 13. When being traveled into the occlusion area, PDOP fluctuates dramatically with the maximum value of 5.2, and SNR declines continuously from above 30 dB. It can be seen from Fig. 13(a) that the proposed system provides accuracy localization and mapping results when GNSS degraded. We not only check the health of raw GNSS measurements, also use the correlation between system states and the world-frame GNSS. The utilization of GNSS can be seen in Fig. 13(b), the stable measurements at the square are involved in state estimation, and those unstable data will be rejected. Therefore, the degradation of GNSS does not affect the overall performance of our system.

2) *Failure Situation:* Here, we show the strong robustness of the proposed system when facing sensor failure. For the comprehensiveness of experiments, the sensor failures are categorized into two situations: data recovery after anomaly and data cut off completely. We manually modified the data in our test to achieve data anomaly and recovery. The sensor data are set to the blank value for a fixed length of time (5 s for GNSS

and LiDAR), as shown in Fig. 14. The system first suffers from the abnormal GNSS data [Fig. 14(c)], the abnormal GNSS data are rejected, and the system operates as LiDAR-IMU subsystem. The mapping and state estimation will not cease, and the global constraints will be added again when GNSS recovers. Then, LiDAR data failure is implemented as the similar way. The invalid point clouds will also be checked and rejected by our system [Fig. 14(d)]. Therefore, GNSS-IMU subsystem operates separately, and the map will not be constructed during LiDAR failure. After data recovery, state estimation and mapping continue to be performed in the same scenario based on the fusion odometry. We further increase the challenges by extending LiDAR failure time, from 5 to 10 s. The results are shown in Fig. 14(e) where the white arrows indicate the failure length. Although minor errors (in the red square) are existed after data recovery, it is acceptable given the global consistent map and strong robustness.

To simulate the worst failure case, experiments are also performed by cutting off sensor data completely when the system is operating. All the sensors are valid in the initial phase, while the GNSS or LiDAR is then physically disconnected (at the hardware level) from the UGV. The corresponding data will not be measured by the system and cannot be recovered subsequently. In the testing of LiDAR failure, the failure segment begins at 1 min 46 s and lasts 45 s. The data sequence and testing results can be seen in Fig. 15, and our system operates as the fusion odometry (GNSS, LiDAR, and IMU) at the initial stage, which builds the map and estimates states at LiDAR frequency. When LiDAR fails, the system seamlessly switches to GNSS-inertial odometry, maintaining localization in the current world frame. The frequency of system output is based on GNSS measurements. The 3-D map constructed by points will not be updated due to the failure of LiDAR while the preexisting global maps are still available. From the results, our system helps the robot to complete the localization function of the whole travel despite the LiDAR failure. For GNSS failure testing, GNSS data will be cut off during the system operation instead of LiDAR. The failure segment is

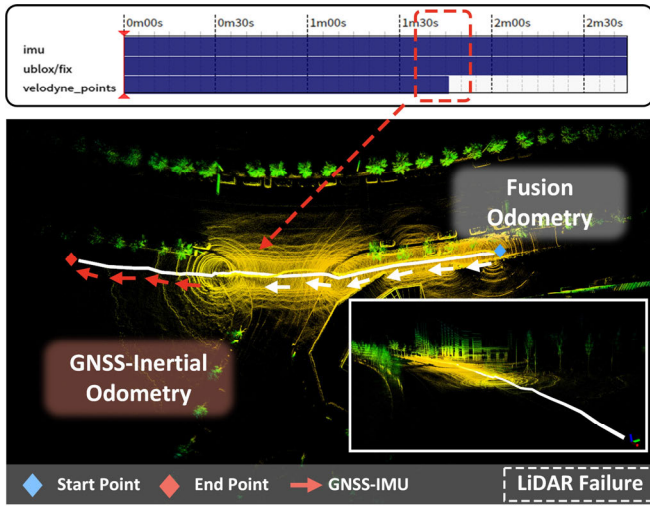


Fig. 15. Experimental results in the situation of LiDAR failure.

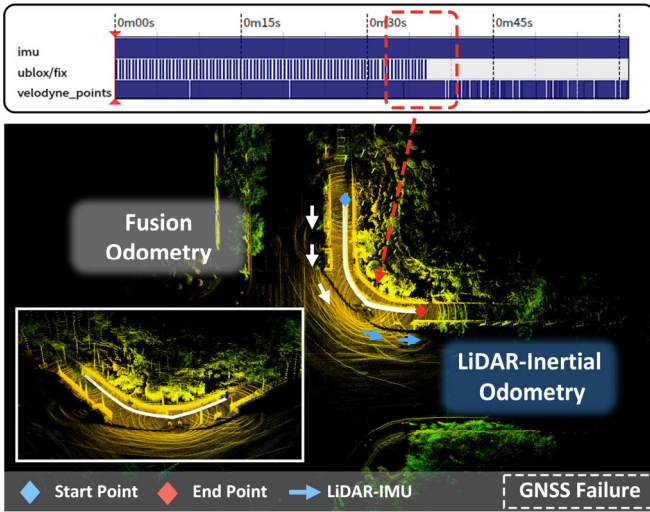


Fig. 16. Experimental results in the situation of GNSS failure.

from 40 s to the end (55 s), as shown in Fig. 16. The system will switch from fusion odometry to LIO when lacking the GNSS measurements. The system in this modality only loses the global constraints from GNSS, and the core functionality on localization is not affected. The result of mapping and localization is output at LiDAR frequency. Thanks to the design of multiple modalities, the proposed system is robust enough to tolerate the failure from GNSS or LiDAR.

VII. CONCLUSION

This article focuses on the performance of navigation systems in common extreme situations for practical implementation. A robust and fast SLAM framework with multiple modalities is proposed by integrating GNSS, IMU, and LiDAR. Benefiting from the multiple modalities based on subsystems and IESKF-based deployment, our system effectively achieves coarse-to-fine state estimation while handling sensor degradation and failures. Additionally, toward the problems caused by inconsistent frames and asynchronous data, we develop a fast and dynamic frame transformation method and distributed

backward propagation. Through extensive experiments, it can be demonstrated that our system achieves a high level of accuracy and efficiency, with strong robustness against partial sensor degradation and even failure.

There are also some limitations in the proposed system, which are expected to be addressed in the future work, such as coupling GNSS with LIO more tightly, and trajectory smoothing during multimodal switching.

REFERENCES

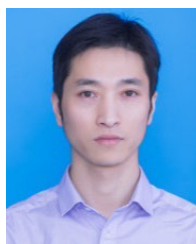
- [1] F. Yang, C. Cao, H. Zhu, J. Oh, and J. Zhang, "FAR planner: Fast, attemptable route planner using dynamic visibility update," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 9–16, doi: [10.1109/IROS47612.2022.9981574](https://doi.org/10.1109/IROS47612.2022.9981574).
- [2] C. Cao, H. Zhu, H. Choset, and J. Zhang, "TARE: A hierarchical framework for efficiently exploring complex 3D environments," in *Proc. Robot., Sci. Syst.*, Jul. 2021, pp. 1–9, doi: [10.15607/RSS.2021.XVII.018](https://doi.org/10.15607/RSS.2021.XVII.018).
- [3] L. Jarin-Lipschitz, X. Liu, Y. Tao, and V. Kumar, "Experiments in adaptive replanning for fast autonomous flight in forests," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 8185–8191, doi: [10.1109/ICRA46639.2022.9812235](https://doi.org/10.1109/ICRA46639.2022.9812235).
- [4] X. Liu et al., "Large-scale autonomous flight with real-time semantic SLAM under dense forest canopy," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 5512–5519, Apr. 2022, doi: [10.1109/LRA.2022.3154047](https://doi.org/10.1109/LRA.2022.3154047).
- [5] K. Mohta et al., "Fast, autonomous flight in GPS-denied and cluttered environments," *J. Field Robot.*, vol. 35, no. 1, pp. 101–120, Jan. 2018, doi: [10.1002/rob.21774](https://doi.org/10.1002/rob.21774).
- [6] T. Qin, T. Chen, Y. Chen, and Q. Su, "AVP-SLAM: Semantic visual mapping and localization for autonomous vehicles in the parking lot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 5939–5945, doi: [10.1109/IROS45743.2020.9340939](https://doi.org/10.1109/IROS45743.2020.9340939).
- [7] G. Wan et al., "Robust and precise vehicle localization based on multi-sensor fusion in diverse city scenes," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 4670–4677, doi: [10.1109/ICRA.2018.8461224](https://doi.org/10.1109/ICRA.2018.8461224).
- [8] H. Ye, Y. Chen, and M. Liu, "Tightly coupled 3D LiDAR inertial odometry and mapping," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 3144–3150, doi: [10.1109/ICRA.2019.8793511](https://doi.org/10.1109/ICRA.2019.8793511).
- [9] H. Li, B. Tian, H. Shen, and J. Lu, "An intensity-augmented LiDAR-inertial SLAM for solid-state LiDARs in degenerated environments," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–10, 2022, doi: [10.1109/TIM.2022.3190060](https://doi.org/10.1109/TIM.2022.3190060).
- [10] C. Qin, H. Ye, C. E. Pranata, J. Han, S. Zhang, and M. Liu, "LINS: A LiDAR-inertial state estimator for robust and efficient navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 8899–8906, doi: [10.1109/ICRA40945.2020.9197567](https://doi.org/10.1109/ICRA40945.2020.9197567).
- [11] W. Xu and F. Zhang, "FAST-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021, doi: [10.1109/LRA.2021.3064227](https://doi.org/10.1109/LRA.2021.3064227).
- [12] D. He, W. Xu, N. Chen, F. Kong, C. Yuan, and F. Zhang, "Point-LIO: Robust high-bandwidth light detection and ranging inertial odometry," *Adv. Intell. Syst.*, vol. 5, no. 7, Jul. 2023, Art. no. 2200459, doi: [10.1002/aisy.202200459](https://doi.org/10.1002/aisy.202200459).
- [13] S. P. Kleinschmidt and B. Wagner, "Visual multimodal odometry: Robust visual odometry in harsh environments," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Aug. 2018, pp. 1–8, doi: [10.1109/SSRR.2018.8468653](https://doi.org/10.1109/SSRR.2018.8468653).
- [14] M. Palieri et al., "LOCUS: A multi-sensor LiDAR-centric solution for high-precision odometry and 3D mapping in real-time," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 421–428, Apr. 2021, doi: [10.1109/LRA.2020.3044864](https://doi.org/10.1109/LRA.2020.3044864).
- [15] N. Khedekar, M. Kulkarni, and K. Alexis, "MIMOSA: A multi-modal SLAM framework for resilient autonomy against sensor degradation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 7153–7159, doi: [10.1109/IROS47612.2022.9981108](https://doi.org/10.1109/IROS47612.2022.9981108).
- [16] S. Zhao, H. Zhang, P. Wang, L. Nogueira, and S. Scherer, "Super odometry: IMU-centric LiDAR-visual-inertial estimator for challenging environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 8729–8736, doi: [10.1109/IROS51168.2021.9635862](https://doi.org/10.1109/IROS51168.2021.9635862).
- [17] J. Sola, "Quaternion kinematics for the error-state Kalman filter," 2017, *arXiv:1711.02508*.

- [18] V. Madyastha, V. Ravindra, S. Mallikarjunan, and A. Goyal, "Extended Kalman filter vs. error state Kalman filter for aircraft attitude estimation," in *Proc. AIAA Guid., Navigat., Control Conf.*, Aug. 2011, p. 6615, doi: [10.2514/6.2011-6615](https://doi.org/10.2514/6.2011-6615).
- [19] J. Havlík and O. Straka, "Performance evaluation of iterated extended Kalman filter with variable step-length," *J. Phys., Conf.*, vol. 659, Nov. 2015, Art. no. 012022, doi: [10.1088/1742-6596/659/1/012022](https://doi.org/10.1088/1742-6596/659/1/012022).
- [20] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot., Sci. Syst. Conf.*, 2014, pp. 1–9, doi: [10.15607/RSS.2014.X.007](https://doi.org/10.15607/RSS.2014.X.007).
- [21] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022, doi: [10.1109/TRO.2022.3141876](https://doi.org/10.1109/TRO.2022.3141876).
- [22] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled LiDAR inertial odometry via smoothing and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 5135–5142, doi: [10.1109/IROS45743.2020.9341176](https://doi.org/10.1109/IROS45743.2020.9341176).
- [23] Y. Wang, Y. Lou, W. Song, Z. Tu, Y. Wang, and S. Zhang, "Simultaneous localization of rail vehicles and mapping of surroundings with LiDAR-inertial-GNSS integration," *IEEE Sensors J.*, vol. 22, no. 14, pp. 14501–14512, Jul. 2022, doi: [10.1109/JSEN.2022.3181264](https://doi.org/10.1109/JSEN.2022.3181264).
- [24] S. Zhong, C. Wei, J. Hu, T. Zhang, J. Yu, and Y. Chen, "An outlier-robust GNSS-inertial-LiDAR localization system," in *Proc. Int. Conf. Sens., Meas. Data Analytics Era Artif. Intell. (ICSMD)*, Oct. 2021, pp. 1–6, doi: [10.1109/ICSMD53520.2021.9670764](https://doi.org/10.1109/ICSMD53520.2021.9670764).
- [25] K.-W. Chiang, G.-J. Tsai, H.-J. Chu, and N. El-Sheimy, "Performance enhancement of INS/GNSS/refreshed-SLAM integration for acceptable lane-level navigation accuracy," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 2463–2476, Mar. 2020, doi: [10.1109/TVT.2020.2966765](https://doi.org/10.1109/TVT.2020.2966765).
- [26] T. Li, L. Pei, Y. Xiang, X. Zuo, W. Yu, and T.-K. Truong, "P³-LINS: Tightly coupled PPP-GNSS/INS/LiDAR navigation system with effective initialization," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–13, 2023, doi: [10.1109/TIM.2023.3244811](https://doi.org/10.1109/TIM.2023.3244811).
- [27] C. Shu and Y. Luo, "Multi-modal feature constraint based tightly coupled monocular visual-LiDAR odometry and mapping," *IEEE Trans. Intell. Veh.*, vol. 8, no. 5, pp. 3384–3393, May 2023, doi: [10.1109/TIV.2022.3215141](https://doi.org/10.1109/TIV.2022.3215141).
- [28] S. Boche, X. Zuo, S. Schaefer, and S. Leutenegger, "Visual-inertial SLAM with tightly-coupled dropout-tolerant GPS fusion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 7020–7027, doi: [10.1109/IROS47612.2022.9981134](https://doi.org/10.1109/IROS47612.2022.9981134).
- [29] T. Shan, B. Englot, C. Ratti, and D. Rus, "LVI-SAM: Tightly-coupled LiDAR-visual-inertial odometry via smoothing and mapping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 5692–5698, doi: [10.1109/ICRA48506.2021.9561996](https://doi.org/10.1109/ICRA48506.2021.9561996).
- [30] J. Lin, C. Zheng, W. Xu, and F. Zhang, "R2LIVE: A robust, real-time, LiDAR-inertial-visual tightly-coupled state estimator and mapping," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7469–7476, Oct. 2021, doi: [10.1109/LRA.2021.3095515](https://doi.org/10.1109/LRA.2021.3095515).
- [31] W. Wu, X. Zhong, D. Wu, B. Chen, X. Zhong, and Q. Liu, "LIO-fusion: Reinforced LiDAR inertial odometry by effective fusion with GNSS/relocalization and wheel odometry," *IEEE Robot. Autom. Lett.*, vol. 8, no. 3, pp. 1571–1578, Mar. 2023, doi: [10.1109/LRA.2023.3240372](https://doi.org/10.1109/LRA.2023.3240372).
- [32] G. He, X. Yuan, Y. Zhuang, and H. Hu, "An integrated GNSS/LiDAR-SLAM pose estimation framework for large-scale map building in partially GNSS-denied environments," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, 2021, doi: [10.1109/TIM.2020.3024405](https://doi.org/10.1109/TIM.2020.3024405).
- [33] N. Li et al., "Indoor and outdoor low-cost seamless integrated navigation system based on the integration of INS/GNSS/LiDAR system," *Remote Sens.*, vol. 12, no. 19, p. 3271, Oct. 2020, doi: [10.3390/rs12193271](https://doi.org/10.3390/rs12193271).
- [34] T. Lupton and S. Sukkarieh, "Efficient integration of inertial observations into visual SLAM without initialization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 1547–1552, doi: [10.1109/IROS.2009.5354267](https://doi.org/10.1109/IROS.2009.5354267).
- [35] C. Hertzberg, R. Wagner, U. Frese, and L. Schröder, "Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds," *Inf. Fusion*, vol. 14, no. 1, pp. 57–77, Jan. 2013, doi: [10.1016/j.inffus.2011.08.003](https://doi.org/10.1016/j.inffus.2011.08.003).
- [36] S. Cao, X. Lu, and S. Shen, "GVINS: Tightly coupled GNSS-visual-inertial fusion for smooth and consistent state estimation," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2004–2021, Aug. 2022, doi: [10.1109/TRO.2021.3133730](https://doi.org/10.1109/TRO.2021.3133730).
- [37] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 809–816, doi: [10.1109/ICRA.2016.7487211](https://doi.org/10.1109/ICRA.2016.7487211).
- [38] J. Yin, A. Li, T. Li, W. Yu, and D. Zou, "M2DGR: A multi-sensor and multi-scenario SLAM dataset for ground robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2266–2273, Apr. 2022, doi: [10.1109/LRA.2021.3138527](https://doi.org/10.1109/LRA.2021.3138527).
- [39] D. Wang, W. Wen, L.-T. Hsu, and Y. Zhong. (2020). *An Open-Sourcing Localization Datasets Collected in Asian Urban Canyons, Including Tokyo and Hong Kong*. [Online]. Available: <https://github.com/IPNL-POLYU/UrbanNavDataset>



Jixin Gao was born in Harbin, China, in 1999. He is currently pursuing the M.S. degree in control science and engineering with the Qingdao Innovation and Development Base, Harbin Engineering University, Qingdao, China.

His research interests include simultaneous localization and mapping (SLAM), multisensor fusion, and robot navigation system.



Jianjun Sha received the Ph.D. degree in cartography and geographic information system from the Institute of Remote Sensing and Digital Earth (RADI), Chinese Academy of Sciences (CAS), Beijing, China, in 2012.

He is currently an Associate Professor of automatic control with the Qingdao Innovation and Development Base, Harbin Engineering University, Qingdao, China. He has presided more than three key scientific research projects. His research interests include remote sensing image processing, object detection, 3-D vision and robotics, and electro-optical countermeasure technology.

Dr. Sha has won Ministerial and Provincial-Level Science and Technology Awards several times.



Hongwen Li received the B.S. degree from the College of Electrical and Information Engineering, Changsha University of Science and Technology, Changsha, China, in 2021. He is currently pursuing the M.S. degree in information and communication engineering with the Qingdao Innovation and Development Base, Harbin Engineering University, Qingdao, China.

His current research interests include motion planning, machine learning, and nonlinear control for the quadrotor suspended payload system.



Yanheng Wang (Student Member, IEEE) received the M.S. degree in software engineering from the College of Information Technology, Shanghai Ocean University, Shanghai, China, in 2020. He is currently pursuing the Ph.D. degree in artificial intelligence with the College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin, China.

His research interests include remote sensing image change detection, object detection, and deep learning.