IC-GVINS: A Robust, Real-time, INS-Centric GNSS-Visual-Inertial Navigation System

Xiaoji Niu, Hailiang Tang, Tisheng Zhang, Jing Fan, and Jingnan Liu

***Abstract*—Visual navigation systems are susceptible to complex environments, while inertial navigation systems (INS) are not affected by external factors. Hence, we present IC-GVINS, a robust, real-time, INS-centric global navigation satellite system (GNSS)-visual-inertial navigation system to fully utilize the INS advantages. The Earth rotation has been compensated in the INS to improve the accuracy of high-grade inertial measurement units (IMUs). To promote the system robustness in high-dynamic conditions, the precise INS information is employed to assist the feature tracking and landmark triangulation. With a GNSS-aided initialization, the IMU, visual, and GNSS measurements are tightly fused in a unified world frame within the factor graph optimization framework. Dedicated experiments were conducted in the public vehicle and private robot datasets to evaluate the proposed method. The results demonstrate that IC-GVINS exhibits superior robustness and accuracy in complex environments. The proposed method with the INS-centric architecture yields improved robustness and accuracy compared to the state-of-the-art methods. We open-source the proposed IC-GVINS and the multi-sensor datasets on GitHub (https://github.com/i2Nav-WHU/IC-GVINS).**

***Index Terms*—Multi-sensor fusion navigation, visual-inertial navigation system, factor graph optimization, state estimation.**

I. Introduction

Continuous, robust, and accurate positioning is essential for autonomous vehicles and robots in complex environments [1]. Visual-inertial navigation system (VINS) has become a practical solution for autonomous navigation due to its higher accuracy and lower cost [2]. It has been historically difficult to achieve a robust and reliable positioning for VINS in complex environments because the visual system is susceptible to illumination change and moving objects [3]. In contrast, the inertial measurement unit (IMU) is not affected by these external environment factors, and the inertial navigation system (INS) can achieve continuous high-frequency positioning independently [4]. The low-cost micro-electro-mechanical system (MEMS) INS cannot provide long-term (e.g. longer than 1 minute) high-accuracy positioning. Nevertheless, it can achieve decimeter-level positioning within several seconds [5]. However, most current VINSs are visual-centric or visual-driven, while the INS precision has not been well considered, such as in [6], [7]. Furthermore, the INS information contributes a little to the visual processes in these systems, which might degrade system robustness and accuracy in visual-degenerated environments. Hence, we propose an INS-centric VINS (IC-VINS) to utilize the INS advantages fully. We further incorporate the global navigation satellite system (GNSS) into the proposed IC-VINS to construct an INS-centric GNSS-visual-inertial navigation system (IC-GVINS) to perform continuous, robust, and accurate positioning in large-scale challenging environments.

Conventionally, the state estimation problem in VINS is addressed through filtering [8]–[11]. However, we have noticed some insufficient usage of the INS in recent filtering-based approaches. For example, OpenVINS [9] is a visual-driven system because the system will pause if no image is received. The independent INS should be adopted for real-time navigation without hesitation. Besides, the INS does not contribute to the feature tracking in [9]. Similarly, the direct image intensity patches were employed as landmark descriptors allowing for tracking non-corner features in an IEKF-based visual-inertial odometry (VIO) [10]. R-VIO [11] is a robocentric visual-inertial odometry within the multi-state constraint Kalman filters (MSCKF) framework. Though the filtering-based VINSs have exhibited considerable accuracy, they theoretically suffer from significant linearization errors, which may ruin the estimator and further degrade the robustness and accuracy [12].

By solving maximum a posterior (MAP) estimation, factor graph optimization (FGO) has been proven to be more efficient and accurate than the filtering-based approaches for VINS [2], [12]. Nevertheless, the INS information has not been fully used in most FGO-based VINSs. Besides, the IMU measurements have only been employed to construct a relative constraint factor, such as the IMU preintegration factor [5]–[7], [13], [14]. VINS-Mono [6] adopts a sliding-window optimizer to achieve pose estimation, but their estimator relies more on high-frequency visual observations. Besides, their visual processes [6] are relatively rough, which limits their accuracy in large-scale complex environments. In ORB_SLAM3 [7], the camera pose predicted by the INS is used to assist the ORB feature tracking instead of using the unreliable ad-hoc motion mode. ORB_SLAM3 is still driven by visual images, thus

Manuscript received: July 28, 2022; Revised: October 14, 2022; Accepted: November 17, 2022. This paper was recommended for publication by Editor Sven Behnke upon evaluation of the Associate Editor and Reviewers' comments. This research is funded by the National Key Research and Development Program of China (No. 2020YFB0505803) and the National Natural Science Foundation of China (No. 41974024). *(Corresponding author: Tisheng Zhang.)*

Xiaoji Niu, Hailiang Tang, Tisheng Zhang, Jing Fan, and Jingnan Liu are with the GNSS Research Center, Wuhan University, Wuhan 430079, China (e-mail: {xjniu, thl, zts, jingfan, jnliu}@whu.edu.cn).

Digital Object Identifier (DOI): see top of this page.

unsuitable for real-time navigation. Similarly, Kimera-VIO [13] is a keyframe-based visual-inertial estimator that can perform both full and fixed-lag smoothing using GTAM [15]. A novel approach is proposed in [14], which combines the strengths of the accurate VIO with the globally consistent keyframe-based bundle adjustment (BA). Their works [14] are built upon the reality that the INS accuracy might quickly degrade after several seconds of integration. However, as mentioned above, the INS can maintain decimeter-level positioning within several seconds [5], even for MEMS IMU.

The high-accuracy industrial-grade MEMS IMU has been widely used for autonomous navigation because the cost has been lower with improved accuracy [4]. However, the INS information has not been well considered, and the INS mechanization algorithm has been relatively rough in these optimization-based VINSs. Besides, most of these VINSs are driven by visual images and unsuitable for real-time applications that need stable and continuous positioning. Moreover, the visual system is delicate and can be easily affected by environments, especially in complex scenes. Hence, the independent INS can play a more critical role in both the state estimation and visual processes of VINS to improve the robustness and accuracy.

The GNSS can achieve absolute positioning in large-scale environments, and thus it has been widely used for outdoor navigation. By using the real-time kinematic (RTK) [4], the GNSS can perform centimeter-level positioning in open-sky environments. In VINS-Fusion [16], the GNSS is integrated into a global estimator, while the local estimator is a VINS. The GNSS can help estimate the IMU biases, but the GNSS is separated from the VINS estimator in [16]. The GNSS raw measurements are tightly incorporated into a VINS in GVINS [17], which can provide global estimation under indoor-outdoor environments. The approach in [17] is based on [6], but the visual processes have not been improved. Hence, GVINS [17] might also degrade robustness and accuracy in GNSS-denied environments. The GNSS can also help to initialize the VINS. In [18], the GNSS/INS integration and VINS are launched simultaneously to initialize a GNSS-visual-inertial navigation system for a land vehicle, but the approach is loosely coupled. G-VIDO [19] is a similar system, but they further incorporate the vehicle dynamic to improve the system accuracy. In [20], a tightly coupled optimization-based GNSS-Visual-Inertial odometry is proposed, but the GNSS does not contribute to the initialization of the visual system. Moreover, the GNSS works in a different world frame from the VINS system in all these systems [16]–[20], and the VINS has to be initialized separately. The GNSS can help initialize the INS first and further initialize the VINS. Hence, the GNSS and VINS can work in a unified world frame without extra transformation.

The visual system may be affected by various degenerated scenes in complex environments. The INS can independently provide precise and high-frequency poses in the short term and may not be affected by external environmental factors. Inspired by these advantages of the INS, we propose an INS-centric GNSS-visual-inertial navigation system to utilize the precise INS information fully. The GNSS is adopted to achieve an accurate initialization and perform absolute positioning in large-scale environments. The main contributions of our work are as follows:

- We propose a tightly-coupled INS-centric GNSS-visual-inertial navigation system (IC-GVINS) within the FGO framework to fully utilize the precise INS information. The INS-centric designs include the precise INS with the Earth rotation compensated, the GNSS-aided initialization, and the INS-aided visual processes.
- IC-VINS, the VINS subsystem of IC-GVINS, is a keyframe-based estimator with strict outlier-culling algorithms. The precise INS information is employed to assist the feature tracking and landmark triangulation and improve the robustness in high-dynamic conditions.
- The proposed method is evaluated in both the public vehicle and private robot datasets. Dedicated experiment results indicate that the proposed method yields improved robustness and accuracy compared to the SOTA methods in complex environments.
- We open-source the proposed IC-GVINS and the well-synchronized multi-sensor robot datasets on GitHub.

II. System Overview

The proposed IC-GVINS is driven by a precise INS mechanization, as depicted in Fig. 1. A GNSS/INS integration is conducted first to initialize the INS to obtain the rough IMU biases and absolute attitude estimation. The absolute attitude is aligned to the local navigation frame (gravity aligned) [4], [5], and thus the GNSS can be directly incorporated into the FGO without extra transformation. Once the INS is initialized, the prior pose from the INS is employed to assist the feature tracking and the landmark triangulation. Finally, the IMU, visual, and GNSS measurements are tightly fused within the FGO framework to achieve MAP estimation. The estimated states are fed back to the INS mechanization module to update the newest INS states for real-time navigation.

III. Methodology

In this section, the methodology of the proposed IC-GVINS is presented. The system core is a precise INS mechanization with the Earth rotation compensated. A GNSS/INS integration is conducted first to initialize the INS. The visual processes are assisted by the prior pose from the INS. Finally, all the measurements are tightly fused within the FGO framework.

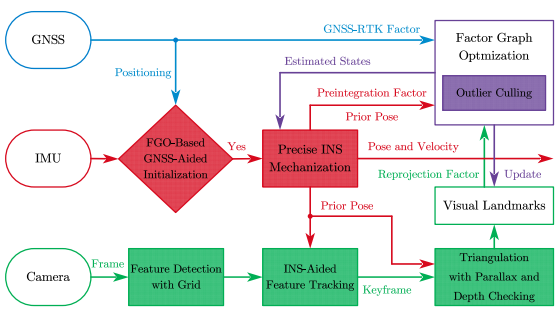

Fig. 1. System pipeline of the proposed IC-GVINS. The filled blocks denote the proposed works in this letter.

A. INS Mechanization

The Earth rotation compensation is not a negligible factor for industrial-grade or higher-grade MEMS IMUs. To fully utilize the INS precision, we follow our previous work in [5] to adopt the precise INS mechanization algorithm, compensating for the Earth rotation and the Coriolis acceleration [4]. The INS kinematic model is defined as follows:

$$\begin{aligned} \dot{\boldsymbol{p}}_{\mathrm{wb}}^{\mathrm{w}} &= \boldsymbol{v}_{\mathrm{wb}}^{\mathrm{w}}, \\ \dot{\boldsymbol{v}}_{\mathrm{wb}}^{\mathrm{w}} &= \mathbf{R}_{\mathrm{b}}^{\mathrm{w}} \boldsymbol{f}^{\mathrm{b}} + \boldsymbol{g}^{\mathrm{w}} - 2\left[\boldsymbol{w}_{\mathrm{ie}}^{\mathrm{w}} \times\right] \boldsymbol{v}_{\mathrm{wb}}^{\mathrm{w}}, \\ \dot{\mathbf{q}}_{\mathrm{b}}^{\mathrm{w}} &= \frac{1}{2} \mathbf{q}_{\mathrm{b}}^{\mathrm{w}} \otimes \begin{bmatrix} 0 \\ \boldsymbol{w}_{\mathrm{wb}}^{\mathrm{b}} \end{bmatrix}, \boldsymbol{w}_{\mathrm{wb}}^{\mathrm{b}} = \boldsymbol{w}_{\mathrm{ib}}^{\mathrm{b}} - \mathbf{R}_{\mathrm{w}}^{\mathrm{b}} \boldsymbol{w}_{\mathrm{ie}}^{\mathrm{w}}, \end{aligned} \tag{1}$$

where $\boldsymbol{p}_{\mathrm{wb}}^{\mathrm{w}}$ and $\boldsymbol{v}_{\mathrm{wb}}^{\mathrm{w}}$ are the position and velocity of the IMU frame (b-frame) in the world frame (w-frame), respectively; the quaternion $\mathbf{q}_{\mathrm{b}}^{\mathrm{w}}$ and the rotation matrix $\mathbf{R}_{\mathrm{b}}^{\mathrm{w}}$ denote the rotation of the b-frame with respect to the w-frame; the w-frame is defined at the initial position of the navigation frame (n-frame) or the local geodetic north-east-down (NED) frame; the IMU frame is defined as the body frame (b-frame); $\boldsymbol{g}^{\mathrm{w}}$ and $\boldsymbol{w}_{\mathrm{ie}}^{\mathrm{w}}$ are the gravity vector and the Earth rotation rate in the w-frame; $\boldsymbol{w}_{\mathrm{ib}}^{\mathrm{b}}$ is the compensated angular velocity from the gyroscope; $\otimes$ denotes the quaternion product. The precise INS mechanization can be formulated by adopting the kinematic model in (1) [5]. The INS pose is directly used for real-time navigation and provides aid for the visual processes, as depicted in Fig. 1.

B. GNSS-Aided Initialization

The initialization is an essential procedure for VINS, which determines the system robustness and accuracy [6], [7]. As an INS-centric system, the most critical task is to initialize the INS. An FGO-based GNSS/INS integration is adopted to initialize the INS, and the FGO framework is described in section III.C. A rough estimation of roll, pitch, and gyroscope biases can be obtained during stationary states by detecting zero-velocity conditions [21]. Dynamic conditions are needed to obtain the absolute attitude from the GNSS. Travelling along a straight line for land vehicles [21] or rarely moving sideways for unmanned aerial vehicles (UAVs) [22] is assumed during the initialization. The absolute attitude is essential for IC-GVINS as we can incorporate the GNSS directly without other coordinate transformations. Besides, the precise IMU preintegration needs the absolute attitude to compensate for the Earth rotation [5]. The GNSS is necessary to initialize the INS in the current implementation for IC-GVINS. Nevertheless, for non-GNSS applications, a stationary condition or a wheeled odometer can help to initialize the INS.

The initialized INS can provide prior pose for the visual processes; thus, the visual system is directly initialized with the INS aiding. Once the landmarks have been triangulated, the visual reprojection factors can be constructed using visual observations. A joint optimization is conducted to refine the state estimation further and improve the INS precision. According to our experiments, only 5 seconds of GNSS positioning (in dynamic conditions) is needed to perform an accurate initialization for the proposed method. In comparison, the GNSS-visual-inertial initialization time is 9 seconds in [18] and 4~9 seconds in [17]. Once the initialization is finished, the VINS subsystem IC-VINS can work independently without the GNSS.

C. INS-Aided Visual Processes

The VINS subsystem IC-VINS is a keyframe-based visual-inertial navigation system. The prior pose from the INS is utilized in the whole visual processes, including the feature tracking and the landmark triangulation. Strict outlier-culling algorithms are conducted to improve the robustness and accuracy further.

1) Feature Detection and Tracking

The Shi-Tomasi corner features are detected in our visual front end. The image is first divided into several grids with a set size, e.g. 200 pixels. The visual features are detected separately in each grid, and a minimum separation of two neighboring pixels is also set to maintain a uniform distribution of the features. Multi-thread technology is employed to improve detection efficiency.

The Lukas-Kanade optical flow algorithm is adopted to track the features. It is challenging for the optical flow algorithm with a limited pyramid level in high-dynamic scenes. Hence, we propose an INS-aided feature tracking algorithm to improve the system robustness. For those features without the initial depth, we predict the initial optical flow estimations by compensating the rotation, and the RANSAC is employed to reject outliers. For those features with depth, the initial optical flow estimations are calculated by projecting the depth into the image plane. We also track the features in the backward direction (from the current to the previous frame) and remove the failed matches. The continuity of the feature tracking can be significantly improved with the INS aiding, especially in high-dynamic conditions. Nevertheless, the prior pose only provides the initial estimations, and the optical flow algorithm determines the final estimations. The tracked features will be undistorted for further processes.

Once the features are tracked, the keyframe selection is conducted. We first calculate the average parallax between the current frame and the last keyframe. The prior pose from the INS is adopted to compensate for the rotation rather than the raw gyroscope measurements in [6]. If the average parallax is larger than a fixed threshold, e.g. 20 pixels, then the current frame is selected as a new keyframe. The selected keyframe will be used to triangulate landmarks and further construct the reprojection factors in the FGO. However, if the vehicle is in a stationary state or the average parallax is smaller than the threshold for a long time, no new optimization will be conducted in the FGO, which might degrade the accuracy. Hence, if no new keyframe is selected after a long time, e.g. 0.5 seconds, a new observation frame will be inserted into the keyframe queue. The observation frame will be used only one time and will be removed after the optimization.

2) Triangulation

With the prior pose from the INS, the triangulation has become a part of the visual front end. When a new keyframe is

selected, the triangulation will be conducted using the current and previous keyframes. The triangulation determines the initial depth of the landmarks, which will be further estimated in the FGO. Hence, a strict outlier-culling algorithm is conducted in the triangulation to prevent the outlier landmarks or poorly initialized landmarks from ruining the FGO estimator. Parallax is first calculated between the feature in the current keyframe and the corresponding feature in the first observed keyframe. If the parallax is too small, e.g. 10 pixels, the visual feature will be tracked until the parallax is enough, which can improve the precision of the triangulated depths. Then, the prior pose from the INS is used to triangulate the landmarks, and the depth of the landmark in its first observed keyframe can be obtained. We further check the depths to ensure the correctness of the triangulation. Only those depths within a range, e.g. 1~100 meters, will be added to the landmark queue or treated as outliers.

D. Factor Graph Optimization

A sliding-window optimizer is adopted to fuse all the measurements within the FGO framework tightly. When a new keyframe is selected or a new GNSS-RTK measurement is valid, a new time node will be inserted into the sliding window, and the FGO will be carried out to perform MAP estimation. The IMU preintegration factor is constructed between each consecutive time node. The FGO framework of the proposed IC-GVINS is depicted in Fig. 2.

1) Formulation

The state vector $\boldsymbol{X}$ in the sliding window of IC-GVINS can be defined as

$$\begin{aligned}\mathbf{X} &= \left[\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}_c^b, \delta_0, \delta_1, \ldots, \delta_l\right], \\ \boldsymbol{x}_k &= \left[\boldsymbol{p}_{\mathrm{wb}_k}^{\mathrm{w}}, \mathbf{q}_{\mathrm{b}_k}^{\mathrm{w}}, \boldsymbol{v}_{\mathrm{wb}_k}^{\mathrm{w}}, \boldsymbol{b}_{g_k}, \boldsymbol{b}_{a_k}\right], k \in [0, n], \\ \boldsymbol{x}_{\mathrm{c}}^{\mathrm{b}} &= \left[\boldsymbol{p}_{\mathrm{bc}}^{\mathrm{b}}, \mathbf{q}_{\mathrm{c}}^{\mathrm{b}}\right],\end{aligned} \tag{2}$$

where $\boldsymbol{x}_k$ is the IMU state at each time node, as shown in Fig. 2; the IMU state includes the position, attitude quaternion, and velocity in the w-frame, and the gyroscope biases $\boldsymbol{b}_g$ and accelerometer biases $\boldsymbol{b}_a$; n is the number of time nodes in the sliding window; $\boldsymbol{x}_{\mathrm{c}}^{\mathrm{b}}$ is the extrinsic parameters between the camera frame (c-frame) and the IMU b-frame; δ is the inverse depth parameter of the landmark in its first observed keyframe.

The MAP estimation in IC-GVINS can be formulated by minimizing the sum of the prior and the Mahalanobis norm of all measurements as

$$\min_{\boldsymbol{X}} \left\{ \begin{aligned} &\left\|\mathbf{r}_p - \mathbf{H}_p \boldsymbol{X}\right\|^2 + \sum_{k\in[1,n]} \left\|\mathbf{r}_{Pre}\left(\tilde{\boldsymbol{z}}_{k-1,k}^{Pre}, \boldsymbol{X}\right)\right\|_{\boldsymbol{\Sigma}_{k-1,k}^{\mathrm{Pre}}}^2 \\ &+\sum_{l\in\boldsymbol{L}} \left\|\mathbf{r}_V\left(\tilde{\boldsymbol{z}}_l^{V_{i,j}}, \boldsymbol{X}\right)\right\|_{\boldsymbol{\Sigma}_l^{V_{i,j}}}^2 + \sum_{h\in[0,m]} \left\|\mathbf{r}_{GNSS}\left(\tilde{\boldsymbol{z}}_h^{GNSS}, \boldsymbol{X}\right)\right\|_{\boldsymbol{\Sigma}_h^{GNSS}}^2 \end{aligned} \right\}, \tag{3}$$

where $\mathbf{r}_{Pre}$ are the residuals of the IMU preintegration measurements; $\mathbf{r}_V$ are the residuals of the visual measurements; $\mathbf{r}_{GNSS}$ are the residuals of the GNSS-RTK measurements; $\boldsymbol{\Sigma}$ is the covariance for each measurement; $\left\{\mathbf{r}_p, \mathbf{H}_p\right\}$ represents the prior from marginalization [6]; m is the number of the GNSS-RTK measurements in the sliding window; $\boldsymbol{L}$ is the landmark map in the sliding window, and l is the landmark in the map; i denotes the reference keyframe of the landmark l, and j is another keyframe. The Ceres solver [23] is adopted to solve this FGO problem.

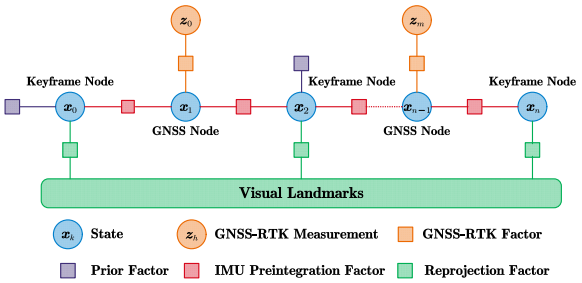

Fig. 2. FGO framework of the IC-GVINS. The visual landmarks are represented by a single block for better visualization.

2) IMU Preintegration Factor

The Earth rotation compensation has been proven to improve the accuracy of the industrial-grade MEMS-IMU preintegration, and thus we follow our refined IMU preintegration [5] in this letter. The residual of the IMU preintegration measurement can be written as

$$\begin{aligned}&\mathbf{r}_{Pre}\left(\tilde{\boldsymbol{z}}_{k-1,k}^{Pre}, \boldsymbol{X}\right) = \\ &\begin{bmatrix} \left(\mathbf{R}_{\mathrm{b}_{k-1}}^{\mathrm{w}}\right)^T \begin{pmatrix} \boldsymbol{p}_{\mathrm{wb}_k}^{\mathrm{w}} - \boldsymbol{p}_{\mathrm{wb}_{k-1}}^{\mathrm{w}} - \boldsymbol{v}_{\mathrm{wb}_{k-1}}^{\mathrm{w}} \Delta t_{k-1,k} \\ -0.5\boldsymbol{g}^{\mathrm{w}} \Delta t_{k-1,k}^2 + \Delta \boldsymbol{p}_{g/cor,k-1,k}^{\mathrm{w}} \end{pmatrix} - \Delta \hat{\boldsymbol{p}}_{k-1,k}^{Pre} \\ \left(\mathbf{R}_{\mathrm{b}_{k-1}}^{\mathrm{w}}\right)^T \begin{pmatrix} \boldsymbol{v}_{\mathrm{wb}_k}^{\mathrm{w}} - \boldsymbol{v}_{\mathrm{wb}_{k-1}}^{\mathrm{w}} - \boldsymbol{g}^{\mathrm{w}} \Delta t_{k-1,k} \\ +\Delta v_{g/cor,k-1,k}^{\mathrm{w}} \end{pmatrix} - \Delta \hat{\boldsymbol{v}}_{k-1,k}^{Pre} \\ 2\left[\left(\mathbf{q}_{\mathrm{b}_k}^{\mathrm{w}}\right)^{-1} \otimes \mathbf{q}_{\mathrm{w}_{\mathrm{i}(k-1)}}^{\mathrm{w}}\left(t_k\right) \otimes \mathbf{q}_{\mathrm{b}_{k-1}}^{\mathrm{w}} \otimes \hat{\mathbf{q}}_{k-1,k}^{Pre}\right]_{\mathrm{v}} \\ \boldsymbol{b}_{g_k} - \boldsymbol{b}_{g_{k-1}} \\ \boldsymbol{b}_{a_k} - \boldsymbol{b}_{a_{k-1}} \end{bmatrix},\end{aligned} \tag{4}$$

where $\Delta \boldsymbol{p}_{g/cor,k-1,k}^{\mathrm{w}}$ and $\Delta \boldsymbol{v}_{g/cor,k-1,k}^{\mathrm{w}}$ are the Coriolis correction term for position and velocity preintegration, respectively; $\Delta \hat{\boldsymbol{p}}_{k-1,k}^{Pre}$, $\Delta \hat{\boldsymbol{v}}_{k-1,k}^{Pre}$, and $\hat{\mathbf{q}}_{k-1,k}^{Pre}$ are the position, velocity and attitude preintegration measurements, respectively; quaternion $\mathbf{q}_{\mathrm{w}_{\mathrm{i}(k-1)}}^{\mathrm{w}}\left(t_k\right)$ is the rotation caused by the Earth rotation [5].

3) Visual Reprojection Factor

We follow [6], [17] to construct the visual reprojection factor in the unit camera frame. The observed feature in the pixel plane can be expressed as $\tilde{\boldsymbol{p}}_{\mathrm{p}}$. For a landmark l with its inverse depth δ_l in the first observed keyframe i, and another observed keyframe j, we can write the visual reprojection residual as

$$\begin{aligned}&\mathbf{r}_V\left(\tilde{\boldsymbol{z}}_l^{V_{i,j}}, \boldsymbol{X}\right) = \begin{bmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 \end{bmatrix}^T \cdot \left(\frac{\hat{\boldsymbol{p}}_{\mathrm{c}_j}}{\left\|\hat{\boldsymbol{p}}_{\mathrm{c}_j}\right\|} - \pi_{\mathrm{c}}^{-1}\left(\tilde{\boldsymbol{p}}_{\mathrm{p}_j}\right) \right), \\ &\hat{\boldsymbol{p}}_{\mathrm{c}_j} = \mathbf{R}_{\mathrm{b}}^{\mathrm{c}} \left(\mathbf{R}_{\mathrm{w}}^{\mathrm{b}_j} \left(\mathbf{R}_{\mathrm{b}_i}^{\mathrm{w}} \left(\mathbf{R}_{\mathrm{c}}^{\mathrm{b}} \frac{1}{\delta_l} \pi_{\mathrm{c}}^{-1}\left(\tilde{\boldsymbol{p}}_{\mathrm{p}_i}\right) + \boldsymbol{p}_{\mathrm{bc}}^{\mathrm{b}} \right) + \boldsymbol{p}_{\mathrm{wb}_i}^{\mathrm{w}} - \boldsymbol{p}_{\mathrm{wb}_j}^{\mathrm{w}} \right) \right) - \boldsymbol{p}_{\mathrm{bc}}^{\mathrm{b}} \right),\end{aligned} \tag{5}$$

where π_c^{-1} is the back camera projection function, which transforms a feature in the pixel plane $\boldsymbol{p}_p$ into the unit camera frame using the camera intrinsic parameters; $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ are two orthogonal bases that span the tangent plane of $\hat{\boldsymbol{p}}_{c_j}$.

4) GNSS-RTK Factor

The GNSS-RTK positioning in geodetic coordinates can be converted to the local w-frame as $\hat{\boldsymbol{p}}_{GNSS}^{w}$ [4]. By considering the GNSS lever-arms $\boldsymbol{l}_{GNSS}^{b}$ in the b-frame, the residual of the GNSS-RTK measurement can be written as

$$\mathbf{r}_{GNSS}\left(\tilde{\boldsymbol{z}}_h^{GNSS}, \boldsymbol{X}\right) = \boldsymbol{p}_{wb_h}^{w} + \mathbf{R}_{b_h}^{w}\boldsymbol{l}_{GNSS}^{b} - \hat{\boldsymbol{p}}_{GNSS,h}^{w}. \tag{6}$$

The GNSS RTK is directly incorporated into the FGO without extra coordinate transformation or yaw alignment as in [16]–[20], which benefits from the INS-centric architecture.

5) Outlier Culling

A two-step optimization is employed in the IC-GVINS. After the first optimization, the chi-square test is adopted to remove all unsatisfied visual reprojection factors from the optimizer rather than the landmark map. The second optimization is then carried out to achieve a better state estimation. Once these two optimizations are finished, the outlier-culling process is implemented. The position of the landmarks in the w-frame is first calculated. Each landmark depth and reprojection error are then evaluated in its observed keyframes. The unsatisfied feature observations, e.g. the depths are not within 1~100 meters or the reprojection errors exceed 4.5 pixels, will be marked as outliers and will not be used in the following optimization. Furthermore, the average reprojection error of each landmark is calculated, and the landmark will be removed from the landmark map if the error is larger than the threshold, e.g. 1.5 pixels. We both remove landmark outliers and feature observation outliers, which significantly improve the robustness and accuracy. We also employ the chi-square test to judge GNSS outliers after the first optimization. However, we do not remove the GNSS outliers but reweight them to mitigate their effects. This method can avoid removing the valid GNSS observations and thus improve the system robustness.

IV. Experiments and Results

A. Implementation and Evaluation Setup

The proposed IC-GVINS is implemented under the Robot Operating System (ROS) framework. The employed sensors include a monocular camera, a MEMS IMU, and a GNSS-RTK receiver. IC-VINS, the VINS subsystem of IC-GVINS, was adopted to evaluate the system robustness and accuracy during the GNSS outages. IC-VINS uses 5 seconds of GNSS for the system initialization. After initialization, IC-VINS uses only the monocular camera and the MEMS IMU. The noise parameter for the visual feature was set to 1.5 pixels without tuning, similar to VINS-Mono [6]. The noise parameters for the employed MEMS IMUs were tuned in the optimization-based GNSS/INS integration by batch processes [5].

We performed comparisons with the SOTA visual-inertial navigation systems VINS-Mono (without relocalization) [6] and OpenVINS [9] and the loosely-coupled GNSS/VINS integration VINS-Fusion (without relocalization) [16]. Here, VINS-Mono is employed because it is also a sliding-window VINS, similar to IC-VINS. Compared to VINS-Mono, our work has improved the front end in the feature detection, the feature tracking and the triangulation, and the back end with the improved IMU preintegration and the outlier-culling algorithm, as depicted in Fig. 1. The temporal and spatial parameters between the camera and IMU are all estimated and calibrated online. Evo [24] is adopted to quantitatively calculate the absolute rotation error (ARE) and absolute translation error (ATE). All the results in the following parts are running in real-time on a desktop PC (AMD R7-3700X). An onboard ARM computer (NVIDIA Xavier) was adopted to evaluate the real-time performance of IC-GVINS.

B. Public Dataset

We evaluated the proposed method in the KAIST Complex Urban Dataset [25]. This dataset was collected by a vehicle in complex urban environments, with a maximum speed of around 15 m/s. The employed sensors include the left camera (with a resolution of 1280x560), the industrial-grade MEMS IMU MTi-300 (with the gyroscope bias instability of 10 °/hr), and the VRS-RTK GPS. The sequences *urban38 and urban39* were adopted for the evaluation. The trajectory lengths are 11191 meters (2154 seconds) and 10678 meters (1856 seconds),

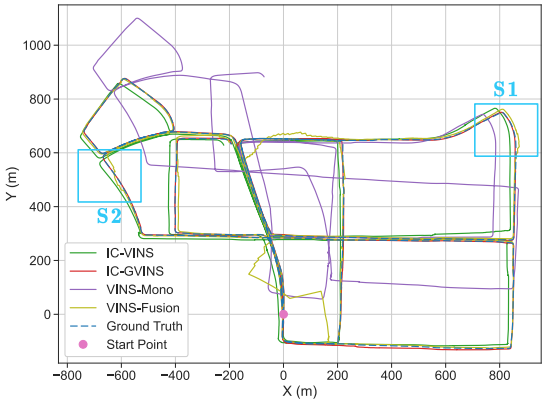

Fig. 3. The trajectories in the KAIST *urban38* dataset. VINS-Mono almost fails in this dataset, and it also occurs a large deviation for VINS-Fusion. The cyan rectangle denotes the GNSS-degenerated scenes in Fig. 5.

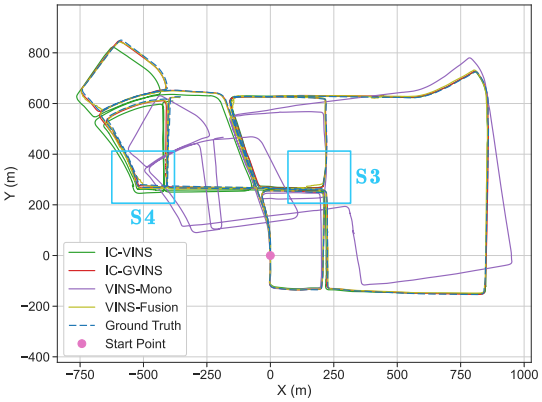

Fig. 4. The trajectories in the KAIST *urban39* dataset. The cyan rectangle denotes the GNSS-degenerated scenes in Fig. 5.

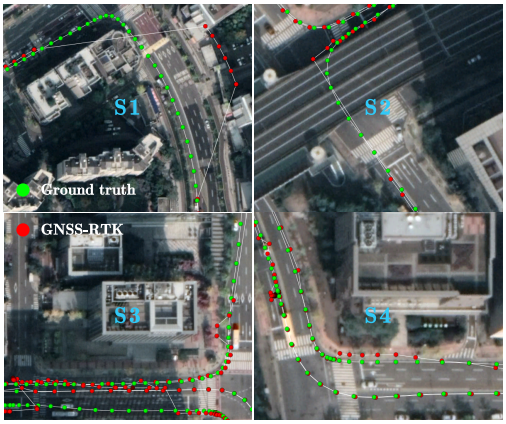

Fig. 5. The GNSS-degenerated scenes in the KAIST dataset. These scenes are marked in Fig. 3 and Fig. 4.

TABLE I
ABSOLUTE POSE ERROR IN THE KAIST DATASET

ARE / ATE (deg / m)	*urban38*	*urban39*
VINS-Mono	4.28 / 125.88	4.91 / 94.47
VINS-Fusion	8.64 / 32.05	6.33 / 10.01
IC-VINS	1.44 / 10.83	1.77 / 13.07
IC-GVINS	**1.31 / 4.27**	**1.32 / 3.84**

respectively. As the vehicle travels very fast, we used a max of 200 features for all the systems to improve the robustness. We failed to run OpenVINS in this dataset, and thus it is not included in this part.

The *urban38* and *urban39* are the two most difficult sequences in the KAIST dataset because of the high-speed motion and the large number of moving objects (mainly vehicles and pedestrians). Nevertheless, the proposed method exhibits superior accuracy in this dataset, as depicted in Fig. 3 and Fig. 4. IC-VINS has very few drifts in both two sequences, while VINS-Mono has large drifts, especially in the *urban38*. These complex scenes may result in the degeneration of the visual system but may not affect the INS. Hence, IC-VINS with the INS-centric architecture can survive and run well in these scenes. In contrast, VINS-Mono, relying much on the visual system, demonstrates unsatisfied robustness and accuracy and almost fails in the *urban38*. With the help of the GNSS, IC-GVINS is well aligned to the ground truth, though there are many GNSS-degenerated scenes, as depicted in Fig. 5. This benefits from the tightly-coupled structure of IC-GVINS, and thus the GNSS outlier can be judged and reweighted. As can be seen in Fig. 3 and Fig. 4, VINS-Fusion exhibits inferior accuracy in these GNSS-degenerated scenes because no outlier-culling method is adopted.

We calculated the absolute pose error in the *urban38* and *urban39*, as shown in Table I. IC-GVINS yields the best accuracy in this dataset, and the accuracy is significantly improved compared to IC-VINS. VINS-Fusion exhibits the worst rotation accuracy, mainly because of the effect of the GNSS outliers. IC-VINS also yields superior accuracy than VINS-Mono and even VINS-Fusion in the *urban38*. The results demonstrate that the proposed method with the INS-centric

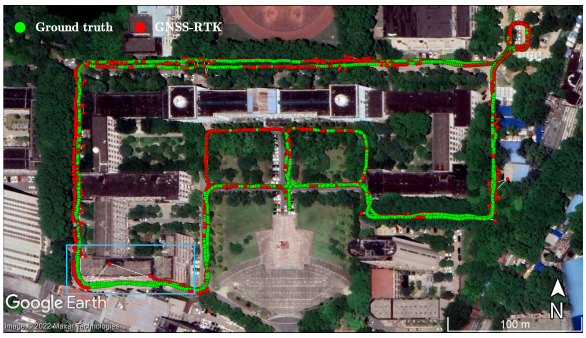

Fig. 6. The test scenes in the *building* dataset. The cyan rectangle denotes the GNSS-outage area in Fig. 7.

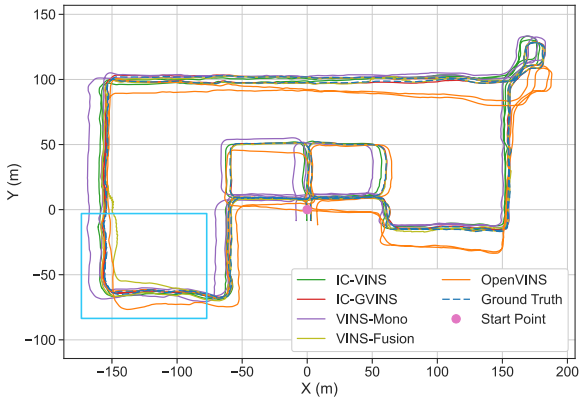

Fig. 7. The trajectories in the *building* dataset. The cyan rectangle corresponds to the GNSS-outage area in Fig. 6.

TABLE II
ABSOLUTE POSE ERROR IN THE ROBOT DATASET

ARE / ATE (deg / m)	VINS-Mono	VINS-Fusion	OpenVINS	IC-VINS	IC-GVINS
building	0.67 / 5.46	8.30 / 5.53	2.98 / 6.01	0.41 / 1.83	**0.40 / 0.86**

architecture is practical in these complex urban environments. Specifically, by fully using the INS information, the proposed method can mitigate the impact of the visual-challenging scenes and exhibit satisfied robustness and accuracy.

C. *Private Dataset*

The private dataset, *building*, was collected by a wheeled robot in complex campus scenes where there were many trees and buildings. Many fast-moving objects around the road also make this dataset highly challenging. The sensors include a monocular camera (Allied Vision Mako-G131 with a resolution of 1280x1024), an industrial-grade MEMS IMU (ADI ADIS16465 with the gyroscope bias instability of 2 °/hr), and a GNSS-RTK receiver (NovAtel OEM-718D). All the sensors have been synchronized through hardware trigger to the GNSS time. The intrinsic and extrinsic parameters of the camera have been calibrated using the Kalibr [26]. The employed ground-truth system is a high-accuracy Position and Orientation System (POS), using the GNSS RTK and a navigation-grade IMU. The ground truth (0.02 m for position and 0.01 deg for attitude) was generated by a post-processing GNSS/INS integration software. The average speed of the wheeled robot is about 1.5 m/s. The trajectory length of the

TABLE III
ABSOLUTE POSE ERROR IN DIFFERENT CONFIGURATIONS

ARE / ATE (deg / m)	*urban38*	*urban39*	*building*
IC-VINS	**1.44** / 10.83	1.77 / 13.07	**0.41** / **1.83**
IC-VINS-E	1.45 / **9.91**	2.08 / 15.64	0.62 / 2.09
IC-VINS-O	1.62 / 12.88	**1.65** / **12.12**	0.70 / 2.34
IC-VINS-I	1.54 / 11.37	2.23 / 15.83	0.50 / 1.90

The IC-VINS-E denotes the method without the Earth rotation compensation in IMU preintegration. The IC-VINS-O denotes the method without the strict outlier-culling strategy. The IC-VINS-I denotes the method without the INS aiding in feature tracking.

building dataset is 1337 meters (950 seconds). As there are rich visual textures in this dataset, we used a max of 120 features.

As shown in Fig. 6, many GNSS-degenerated scenes exist in the *building* dataset, and the GNSS is even interrupted in a narrow corridor. There are large drifts for VINS-Mono and OpenVINS, while there are only small drifts for IC-VINS, as depicted in Fig. 7. Besides, IC-GVINS is well aligned to the ground truth, even though there are GNSS outliers and outages. In contrast, VINS-Fusion has a notable drift because of the impact of the GNSS outliers, as depicted in Fig. 6.

We also calculated the absolute pose error, as exhibited in Table II. The results demonstrate that IC-VINS yields higher accuracy than VINS-Mono and OpenVINS. In addition, VINS-Fusion shows worse accuracy than VINS-Mono, because the GNSS outlier may ruin the estimator significantly. In contrast, IC-GVINS exhibits improved accuracy compared to IC-VINS and performs the best in the *building* dataset. As can be seen, the proposed INS-centric can fully utilize the INS information and thus can mitigate the impact of the visual-challenging scenes in complex environments. Moreover, the employed outlier-culling algorithm for visual and GNSS observations can significantly improve the system robustness.

D. Robustness Evaluation

To fully demonstrate the robustness of the proposed method, we further evaluated the effects of the Earth rotation compensation, the strict outlier-culling algorithm, and the INS aiding in feature tracking. Three extra configurations were employed for the evaluation, as shown in Table III.

1) Effect of the Earth Rotation Compensation

The gyroscope bias-instability parameters for MTi-300 (10 °/hr) in the KAIST dataset and ADIS16465 (2 °/hr) in the robot dataset are all smaller than the Earth rotation rate of 15 °/hr. Thus, it is necessary to compensate for the Earth rotation in the INS mechanization and IMU preintegration. We compared the results of IC-VINS and IC-VINS-E (without compensating for the Earth rotation), as depicted in Table III. The results indicate that the Earth rotation compensation can improve the system accuracy in the *urban39* and *building*, while the translation accuracy degrades a little in the *urban38*. MTi-300 is not precise enough to sense the Earth rotation; thus, the effect of the Earth rotation compensation for Mti-300 in the KAIST dataset should not be significant. Besides, the impact of the Earth rotation compensation cannot be effectively determined if the visual observations are sufficient, as mentioned in [5].

TABLE IV
ABSOLUTE POSE ERROR CONCERNING DIFFERENT VISUAL FEATURES IN THE ROBOT DATASET

ARE / ATE (deg / m)	120	60	30
IC-VINS	**0.41 / 1.83**	**0.55 / 1.82**	**0.65 / 2.21**
IC-VINS-E	0.62 / 2.09	0.90 / 2.45	0.69 / 2.40

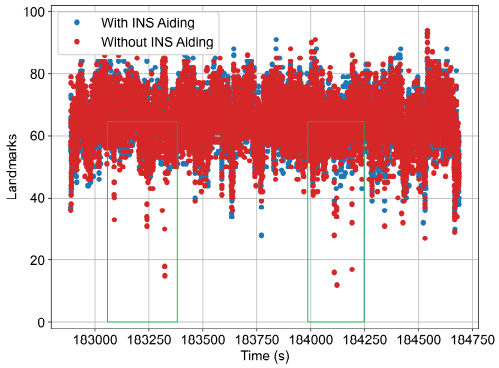

Fig. 8. Comparison of the number of the landmarks in the *building* dataset. The green rectangles in the figure denote the areas where it occurs speed bumps and potholes.

As ADIS16465 is more precise, we further evaluated the effect of the Earth rotation compensation by detecting different visual features in the robot dataset. As can be seen in Table IV, the effect of the Earth rotation compensation is more significant when the visual features are fewer. The results demonstrate that the Earth rotation compensation can improve the system accuracy, especially when the visual system is weak, i.e. the visual-challenging scenes. Hence, we suggest compensating for the Earth rotation if a high-grade IMU is employed, which can improve the system accuracy in complex environments.

2) Effect of the Strict Outlier-culling Algorithm

Previous results have demonstrated that the employed GNSS outlier-culling algorithm can significantly improve the system robustness and accuracy. As for the visual outlier-culling algorithm, we compared the results of IC-VINS and IC-VINS-O, as exhibited in Table III. IC-VINS-O uses only the outlier-culling algorithm in VINS-Mono [6] without using the strict outlier-culling algorithm described in section III.C.2 and section III.D.5. The results indicate that IC-VINS outperforms IC-VINS-O in the *urban38* and *building*, while the accuracy degrades a little in the *urban39*. The strict outlier-culling algorithm will result in fewer valid visual landmarks, but motions are needed to triangulate new landmarks. However, the vehicle has to stop at the traffic lights in the KAIST dataset frequently, and the passing vehicles may interrupt the feature tracking, resulting in fewer valid visual landmarks. The new landmarks cannot be created during stationary states with a monocular camera. Detecting more visual features in the KAIST dataset may solve this problem. Hence, we suggest employing the proposed outlier-culling algorithm to improve the robustness, especially in complex environments.

3) Effect of the INS Aiding in Feature Tracking

We also compared the results of IC-VINS and IC-VINS-I (without the INS aiding in the feature tracking) in Table III. The results illustrate that the INS aiding in feature tracking can

TABLE V
AVERAGE RUNNING TIME OF IC-GVINS

PC / Onboard (ms)	*urban38*	*urban39*	*building*
Front-end	11.5 / 35.9	11.8 / 39.8	14.4 / 32.4
FGO	18.4 / 73.2	18.3 / 76.5	17.4 / 101.5

Here, the FGO is only conducted when a new keyframe is selected.

improve the system accuracy, especially in the high-dynamic dataset, i.e. the KAIST dataset. For the low-speed wheeled robot, the effect of the INS aiding is limited. In the *building* dataset, there are several speed bumps and potholes which may cause aggressive motion, making feature tracking extremely challenging. Hence, we compared the landmarks in the *building* dataset to evaluate the effect of the INS aiding in feature tracking. As depicted in Fig. 8, without the INS aiding, the valid landmarks are far fewer than 20 in such cases and are even close to 0. With the help of INS aiding, the valid landmarks are more than 20 during the whole travel. The results demonstrate that the INS aiding can improve the robustness of the feature tracking significantly, especially in high-dynamic scenes.

E. Run time analysis

The average running times of IC-GVINS are shown in Table V. All the experiments are running within the ROS framework, demonstrating that IC-GVINS can perform real-time positioning on both the desktop PC (AMD R7-3700X) and the onboard ARM computer (NVIDIA Xavier).

V. CONCLUSIONS

A robust, real-time, INS-centric GNSS-visual-inertial navigation system is presented in this letter. As the visual system may be affected by degenerated scenes, the precise INS information is fully employed in the visual processes and state estimation to improve the system robustness and accuracy in complex environments. With the GNSS-aided initialization, the IMU, visual, and GNSS measurements can be tightly fused in a unified world frame within the FGO framework. We performed experiments in both the high-speed vehicle and the low-speed robot datasets. IC-GVINS exhibits superior robustness and accuracy in degenerated and challenging scenes. The results demonstrate that the proposed method with the INS-centric architecture can significantly improve the system robustness and accuracy compared to the SOTA methods in complex environments.

REFERENCES

[1] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*, 2nd ed. Cambridge, Mass: MIT Press, 2011.

[2] C. Cadena *et al.*, "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.

[3] J. Janai, F. Güney, A. Behl, and A. Geiger, "Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art," *ArXiv170405519 Cs*, Mar. 2021. [Online]. Available: http://arxiv.org/abs/1704.05519

[4] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Boston: Artech House, 2008.

[5] H. Tang, T. Zhang, X. Niu, J. Fan, and J. Liu, "Impact of the Earth Rotation Compensation on MEMS-IMU Preintegration of Factor Graph Optimization," *IEEE Sens. J.*, vol. 22, no. 17, pp. 17194–17204, Sep. 2022.

[6] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[7] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, 2021.

[8] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 3565–3572.

[9] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "OpenVINS: A Research Platform for Visual-Inertial Estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, May 2020, pp. 4666–4672.

[10] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback," *Int. J. Robot. Res.*, vol. 36, no. 10, pp. 1053–1072, Sep. 2017.

[11] Z. Huai and G. Huang, "Robocentric visual–inertial odometry," *Int. J. Robot. Res.*, p. 0278364919853361, Jul. 2019.

[12] G. Huang, "Visual-Inertial Navigation: A Concise Review," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 9572–9582.

[13] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an Open-Source Library for Real-Time Metric-Semantic Localization and Mapping," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, May 2020, pp. 1689–1696.

[14] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers, "Visual-Inertial Mapping With Non-Linear Factor Recovery," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 422–429, Apr. 2020.

[15] F. Dellaert, "Factor graphs and GTSAM: A hands-on introduction," Georgia Institute of Technology, 2012.

[16] T. Qin, S. Cao, J. Pan, and S. Shen, "A General Optimization-based Framework for Global Pose Estimation with Multiple Sensors," *ArXiv190103642 Cs*, Jan. 2019. [Online]. Available: http://arxiv.org/abs/1901.03642

[17] S. Cao, X. Lu, and S. Shen, "GVINS: Tightly Coupled GNSS–Visual–Inertial Fusion for Smooth and Consistent State Estimation," *IEEE Trans. Robot.*, pp. 1–18, 2022.

[18] R. Jin, J. Liu, H. Zhang, and X. Niu, "Fast and Accurate Initialization for Monocular Vision/INS/GNSS Integrated System on Land Vehicle," *IEEE Sens. J.*, vol. 21, no. 22, pp. 26074–26085, 2021.

[19] L. Xiong *et al.*, "G-VIDO: A Vehicle Dynamics and Intermittent GNSS-Aided Visual-Inertial State Estimator for Autonomous Driving," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–17, 2021.

[20] S. Han, F. Deng, T. Li, and H. Pei, "Tightly Coupled Optimization-based GPS-Visual-Inertial Odometry with Online Calibration and Initialization," *ArXiv220302677 Cs*, Mar. 2022. [Online]. Available: http://arxiv.org/abs/2203.02677

[21] Q. Zhang, S. Li, Z. Xu, and X. Niu, "Velocity-Based Optimization-Based Alignment (VBOBA) of Low-End MEMS IMU/GNSS for Low Dynamic Applications," *IEEE Sens. J.*, vol. 20, no. 10, pp. 5527–5539, May 2020.

[22] D. Wang, H. Lv, and J. Wu, "In-flight initial alignment for small UAV MEMS-based navigation via adaptive unscented Kalman filtering approach," *Aerosp. Sci. Technol.*, vol. 61, pp. 73–84, Feb. 2017.

[23] Agarwal, Sameer, Mierle, and Keir, "Ceres Solver — A Large Scale Non-linear Optimization Library," 2022. [Online]. Available: http://ceres-solver.org/

[24] M. Grupp, "evo." Jul. 21, 2022. [Online]. Available: https://github.com/MichaelGrupp/evo

[25] J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim, "Complex urban dataset with multi-level sensors from highly diverse urban environments," *Int. J. Robot. Res.*, vol. 38, no. 6, pp. 642–657, May 2019.

[26] J. Rehder, J. Nikolic, T. Schneider, T. Hinzmann, and R. Siegwart, "Extending kalibr: Calibrating the extrinsics of multiple IMUs and of individual axes," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 4304–4311.