Safety-Quantifiable Line Feature-Based Monocular Visual Localization With 3D Prior Map

Xi Zheng, Weisong Wen, *Member, IEEE*, and Li-Ta Hsu

***Abstract*— Accurate and safety-quantifiable localization is of great significance for safety-critical autonomous systems, such as Autonomous ground vehicles (AGVs) and autonomous aerial vehicles (AAVs). The visual odometry-based method can provide accurate positioning in a short period but is subject to drift over time. Moreover, the quantification of the safety of the localization solution (the error is bounded by a certain value) is still a challenge. To fill the gaps, this paper proposes a safety-quantifiable line feature-based visual localization method with a prior map. The visual-inertial odometry provides a high-frequency local pose estimation, which serves as the initial guess for the visual localization. By obtaining a visual line feature pair association, a foot point-based constraint is proposed to construct the cost function between the 2D lines extracted from the real-time image and the 3D lines extracted from the high-precision prior 3D point cloud map. Moreover, a global navigation satellite system (GNSS) receiver autonomous integrity monitoring (RAIM) inspired method is employed to quantify the safety of the derived localization solution. Among that, an outlier rejection (also well-known as fault detection and exclusion) strategy is employed via the weighted sum of squares residual with a Chi-squared probability distribution. A protection level (PL) scheme considering multiple outliers is derived and utilized to quantify the potential error bound of the localization solution in both position and rotation domains. The effectiveness of the proposed safety-quantifiable localization system is verified using the datasets collected by AAV and AGV in indoor and outdoor environments, respectively. The open-source code is available at https://github.com/ZHENGXi-git/SafetyQuantifiable-PLVINS**

***Index Terms*— Safety quantification, state estimation, visual localization, prior map, outlier rejection, protection level.**

I. Introduction

ACCURATE, cost-effective, and reliable localization is of great importance for the realization of safety-critical autonomous navigation systems, such as autonomous ground vehicles (AGV) [1], [2] and autonomous aerial vehicles (AAV) [3]. The global navigation satellite system (GNSS) [4] is widely used for providing globally-referenced positioning for autonomous systems with navigation requirements. However, positioning accuracy is severely degraded in highly urbanized cities such as Hong Kong, due to signal reflection caused by static buildings leading to the notorious non-line-of-sight (NLOS) receptions [5] and multipath effects, so-called GNSS outlier measurements.

Instead of relying on the GNSS, the prior map-based light detection and ranging (LiDAR) localization methods [6], [7] attracted lots of attention due to their robustness and accuracy, where the matching between the real-time point clouds and the prior map was employed to estimate the location of the system within the map. However, the consumption of computational resources in point clouds registration [8] and the high-cost LiDAR sensor limit the massive deployment of these localization solutions [9]. Recently, the 2D visual-based localization in a 3D prior map attracted lots of attention, since this data fusion technique offers considerable potential for balancing the localization performance of AGVs and AAVs across factors such as cost-effectiveness, real-time, accuracy, and robustness [10], [11], [12], [13]. But the visual localization within the prior map is still challenging, due to the pattern difference between the 2D pixel texture measurements from the visual image and 3D point cloud information from the prior map [14]. Moreover, reliably quantifying the safety of the derived localization results is significant for safety-critical autonomous systems. the navigation solution is certified as safe only when its potential error is quantifiable. However, the safety quantification of the map-based localization method is still an open question.

A. Visual Localization With Prior Point Cloud Map

To fill the gap between the 2D image and 3D point cloud, some researchers transform data into the same dimensional space for the association. For example, the photometry matching-based methods proposed to project 3D points into synthetic images and match them within an intensity image by maximizing the normalized mutual information [15] or minimizing the normalized information distance [16]. However, the nonlinear projection transformation from points to images would be challenged by the discreteness of the point cloud and limit the localization accuracy. Besides, the geometry matching-based methods employed bundle adjustment (BA) [17] to reconstruct a point cloud, and then aligned the two point clouds with registration methods, such as the iterative closest point (ICP) [8], and normal distribution transform (NDT) [18], [19]. The hybrid of photometric and geometric

Received 28 November 2022; revised 16 October 2023, 7 May 2024, and 13 March 2025; accepted 4 May 2025. This work was supported in part by the Innovation and Technology Fund under the Project Safety-Certified Multi-Source Fusion Positioning for Autonomous Vehicles in Complex Scenarios (ZPE8), in part by the Germany/Hong Kong Joint Research Scheme under the project Maximum Consensus Integration of GNSS and LiDAR (RADM), in part by the Research Center of Deep Space Exploration (RC-DSE) under the Project Multi-Robot Collaborative Operations (BBDW), and in part by the PolyU Research Institute for Advanced Manufacturing (RIAM) under the Project Autonomous Aerial Vehicle Aided High Accuracy Addictive Manufacturing for Carbon Fiber Reinforced Thermoplastic Composites Material (CD8S). The Associate Editor for this article was F. Wang. *(Corresponding author: Weisong Wen.)*

The authors are with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: zheng-xi.zheng@connect.polyu.hk; welson.wen@polyu.edu.hk; lt.hsu@polyu.edu.hk).

Digital Object Identifier 10.1109/TITS.2025.3572620

methods calculated two types of depth images from point clouds and a stereo camera, respectively, and then compared them for camera localization [11], [20]. However, large-scale BA operation is time-consuming, and the point cloud reconstructed based on image features is usually sparse. Meanwhile, the range of the depth recovery from the stereo camera is limited because of the requirement for dense texture information.

The other research stream proposed to match the 2D representative features with the 3D point cloud by estimating the scale and the pose of the system simultaneously. For example, Caselitz et al. proposed a 7-DoF ICP scheme to estimate the scale and pose between 3D LiDAR map and sparse points calculated from BA method [21]. Liu et al. [10] exploited global contextual information between 2D images and 3D point map to handle the matching ambiguity and solve it on a Markov graph structure by a random walk algorithm. These methods only exploited the potential of corner features but did not fully explore the geometry information (such as lines) from environments. It cannot be ignored that high-definition (HD) map-based localization is favored by academia and industry in autonomous driving [22]. Specifically, Schreiber et al. [23] matched road lane markings and curbs detected on stereo camera with an HD-map for localization by Kalman Filter, where the maps are created by a sensor setup containing a high precision GNSS unit, Velodyne laser scanner, and cameras.

In addition, some researchers applied semantic information to get cross-modal correspondences between 2D image matching with 3D prior maps to solve localization problem. Specifically, Liang et al. constructed a hybrid constraint with Dirichlet distribution between semantic and structure information from prior maps, and then applied the Expectation-Maximization algorithm to jointly optimize data associations and poses [24]. Zhang et al. utilized semantic information to build a point-to-plane ICP algorithm for cross-model registration and executed localization task by a decoupling optimization strategy [25]. Qin et al. [9] proposed a lightweight HD-map composed of semantic elements such as lane lines, crosswalks, and ground signs on the road. Vehicles are localized in the semantic map by online image segmentation and the ICP method. Zhou et al. [26] developed a unified pose graph-based estimator including crowd-sourced mapping and a low-cost map-aided localization system based on the semantic road marking. On the other hand, some learning-based networks conducted image-to-point cloud registration by training various datasets, but these methods ultimately suffer from the dependence on labeled datasets and limited generality [27], [28].

Interestingly, the line feature was utilized by visual-simultaneously localization and mapping (VSLAM) communities [29], [30], [31] where the lines are employed as an additional feature correspondence to estimate the motion of the system. In some low-textured areas (the conventional corner features are limited), line segments containing geometrical structure information are particularly effective [32]. Furthermore, lines usually have more stable constraints than points in both 3D space and 2D images, as a line segment consists of multiple points [33]. The results [29], [30] revealed that the line features could effectively exploit the potential of the geometry information within the environment. Considering point clouds with salient structural information and the potential of the prior map, the work in [34] proposed a line feature-based visual localization method where the 2D line feature was associated with the 3D lines extracted from the prior point cloud map aided by the high-frequency initial guess of the pose from the visual-inertial navigation system (VINS) [35]. However, the work in [34] constructed the constraints of the line correspondence based on the line-to-line distance, which fails to fully explore the geometric correlation of the line features. *As an extension, this paper proposes an improved line feature-based point-to-point constraint model aided by a carefully designed line correspondence selection strategy to achieve reliable visual localization within the prior map.*

B. Safety Quantification of Localization Solution

When the localization errors can be estimated and monitored simultaneously, it is possible to determine whether a navigation system is safe or not [36]. Here, we introduce the concept of **safety quantification**, a mechanism that can predict and monitor localization errors within a navigation system. By quantifying these errors, it becomes possible to assess the system's safety in real-time. This approach enables a systematic evaluation of whether the navigation system operates within safe parameters, ensuring both reliability and accuracy. Safety quantification provides a framework for understanding and interpreting the prediction of localization errors, thereby enhancing the overall safety of navigation systems. Reliably quantifying the safety of the derived localization solution is significant for safety-critical autonomous systems before their massive production and deployment. For autonomous driving vehicles, a large localization error can affect the subsequent planning and decision-making steps of the vehicle, which may lead to serious safety accidents. Consequently, localization error prediction is important and necessary to reduce the probability of safety accidents and guarantee the correctness of decision-making. Unfortunately, the safety quantification of the map-based localization solution has not been effectively studied, which still has a big gap in front.

For state estimation problem, the positioning accuracy of a navigation system is inevitably affected by observation noise. In addition, the outliers in the observations as erroneous constraints are also an important factor influencing the accuracy (outliers i.e. the fault measurements). Therefore, the positioning error of the navigation solution is mainly caused by two parts: *(1) The measurements noise:* usually constructed as a random Gaussian distribution with zero mean [37]. (2) The outliers in observations: researchers studied how to remove the outliers to get better results [38]. In short, the key to quantifying the safety of the localization solution is to reliably account for the errors arising from those two components.

In visual localization, several outlier rejection algorithms have existed. The popular methods are random sample consensus (RANSAC) and its variants [39], [40]. RANSAC algorithm relies on the selection of the kernel parameter, which can be slow and fail in the presence of high outlier rates [41]. On the other hand, the global optimization methods based on branch-and-bound and mixed integer programming strategies try to make the global outlier searching problem tractable and faster [14], [42]. These methods could lead to a high computational load and a satisfactory initial guess of the state is required. Another method for outlier detection is the

parity space approach, which projects the measurements into a designed parity space, where the outliers can be highlighted and detected [43]. However, even after the outlier exclusion operation, system cannot 100% guarantee that there are no outliers in the existing observations. So, outliers remaining in the observations still have a negative impact on localization accuracy, which is commonly ignored.

Interestingly, a widely used receiver autonomous integrity monitoring (RAIM) theory in the GNSS field can be used to estimate the positioning error. RAIM argues that positioning errors are not only caused by the noise of measurements but also related to the received satellite. Even after passing the fault detection and exclusion (FDE) operation, received satellite signals can still be fault observations [37], [38]. On this basis, RAIM calculates a statistical value called the protection level (PL) to predict the localization error through the principle of error propagation in state estimation. In other words, PL aims to account for the maximal positioning error that is caused by the measurement noise and the remaining outliers that survived after FDE [37], [38]. Initially, the calculation of PL was based on the assumption that only one outlier is left in the observations [37]. The work proposed in paper [38] extended the basic assumption from the single outlier to multiple outliers.

Inspired by the RAIM theory in the GNSS field, Legrand et al. extended the RAIM attribute from pure GNSS to GNSS/INS localization system to evaluate the safety of train movement [44]. Arana et al. leverage integrity monitoring to quantify robot localization safety in GNSS-denied environments based on the extended Kalman filter and LiDAR/IMU platform [45]. Mur-Artal and Tardós drew on RAIM to monitor the safety of visual localization based on ORB-SLAM2 [46] and determined an approximate upper bound of the positioning error by calculating the protection level [46], [47]. Dr. Zhu proposed a preliminary visual positioning pipeline similar to RAIM and focused on the error model of visual pixel features for fault detection and elimination, but this pipeline has not been systematically verified [48], [49]. Meanwhile, these methods mainly rely on the single-outlier assumption in PL calculation, which limits their generality for cases with multiple faults. Moreover, the RAIM theory in GNSS only considered safety quantification in the positioning domain, but the safety quantification in the orientation is also equally important for autonomous systems (e.g., AAVs), which is not fully investigated. *In this paper, a RAIM-inspired safety quantification method is proposed to estimate the maximum localization error* under the multiple-outlier assumption, and in both positioning and orientation domains.

C. Key Contributions of This Paper

This paper proposes a safety-quantifiable line feature-based visual localization method with a LiDAR point cloud prior map. The main contributions of this paper are summarized as follows:

1) *Pixel-level point-to-point line constraint model for visual localization:* By transforming the 2D-3D line association-based optimization problem from minimizing the line-to-line distance model into a point-to-point reprojection model, a new reliable pixel-level constraint model based on line features is proposed for reliable localization within the prior map.
2) *Safety quantification considering multiple faults:* A RAIM-inspired safety quantification pipeline is proposed, which accounts for the localization safety both for the positioning and orientation domains. This paper systematically derives the safety quantification process by FDE and PL estimation under the multiple faults situation.
3) *A comprehensive framework for safety quantifiable visual localization:* This work proposes an integral framework including visual localization and safety quantification for autonomous systems. The effectiveness of the proposed pipeline is verified with challenging datasets collected by both AAV and AGV systems. Moreover, a detailed analysis of the localization safety towards the line features degeneracy is presented.

The rest of this paper is organized as follows. The overview of the proposed system pipeline is given in Section II. The methodology of the proposed visual localization with a prior map based on line feature correspondences is presented in Section III. In Section IV, the derivation of the safety quantification is presented including the FDE, and the protection level estimation considering multiple faults is presented. Experimental results and discussions about the properties of PL and feature degeneration are provided in Section V. Section VI concludes this paper and outlines the future research directions. The *Appendix* supplements the details of the Jacobian matrix used in Section III.

II. Overview of the Proposed Method

The proposed system pipeline is shown in Fig. 1. The system mainly includes two parts: the visual localization module and the safety quantification module. Firstly, a high-precision 3D point cloud as a prior map is generated in advance, in which all 3D lines are extracted and stored offline [50]. Then the 3D lines are projected into image frames based on the initial field of view (FoV) of the camera that is provided by the pose estimation from the VINS [35] and a given starting position. On the other hand, the images captured by an onboard camera are used to detect 2D lines in real-time (Section III-A). The two types of lines are matched into line feature correspondences, which serve as the predictions and measurements of the state optimization model for absolute visual localization (Section III-B) in this prior map. Secondly, the safety quantification is implemented based on a reliable FDE and protection level calculation. Specifically, a Chi-squared test is applied to detect outliers based on the weighted residual iteratively (Section IV-B). After removing the detected outliers, the surviving observations are re-input into the optimization model for state estimation until the Chi-squared test passes. Then, the protection level is estimated based on the multiple fault assumption by quantitatively evaluating the safety of the visual localization solution (Section IV-C).

The notations are defined as follows. $(\cdot)^w$ is the world frame, and the origin coincides with the prior map starting point. $(\cdot)^b$ is the body frame, and the origin is defined on the inertial measurement unit (IMU). $(\cdot)^c$ is the camera frame. The translation vector and rotation matrix are represented by $\mathbf{R}$ and $\mathbf{t}$, respectively, and also by their corresponding Lie

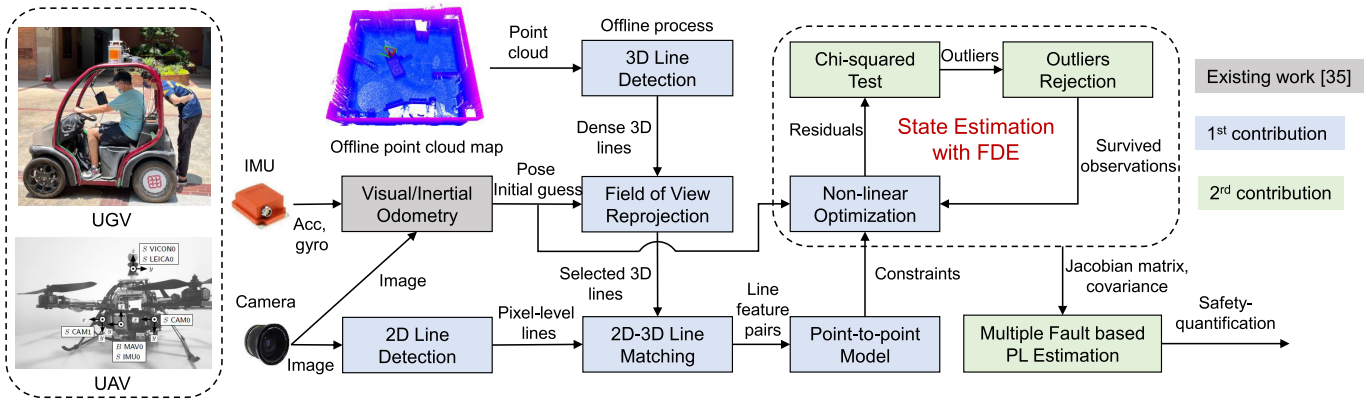


Fig. 1. The framework of the proposed system. The 3D lines are extracted from prior 3D point cloud map in advance. The 2D lines are detected in the camera frame online(see Section III-A) After giving the state initial guess by VINS [35], the two types of lines are matched as correspondences (see Section III-A), which are observations for the optimization model (see Section III-B). Based on the residuals of optimization, a Chi-squared test is utilized to reject outliers iteratively (see Section IV-B, fault detection and exclusion (FDE), i.e., outlier rejection). When the test is passed, a protection level (see Section IV-C) is calculated to quantify the safety of state estimation.

algebra $\boldsymbol{\xi}$ [51]. $\mathbf{R}_w^b, \mathbf{t}_w^b$ means the transformation from the world frame to the body frame, $\mathbf{R}_b^c, \mathbf{t}_b^c$ is from the body frame to the camera frame. b_k is the subscript of the body frame in k^{th} image. The state variables in this paper are the pose of UGV or AAV at world frame while taking the k^{th} image, expressed by $\mathbf{R}_{b_k}^w$ and $\mathbf{t}_{b_k}^w$. $\mathbf{P}(X, Y, Z)$ is a 3D point and $\mathbf{p}(\mu, \nu)$ means a 2D image pixel point. The notation of the Chi-Squared distribution is χ^2.

III. State Estimation: Visual Localization With Prior Map

A. 2D-3D Line Matching

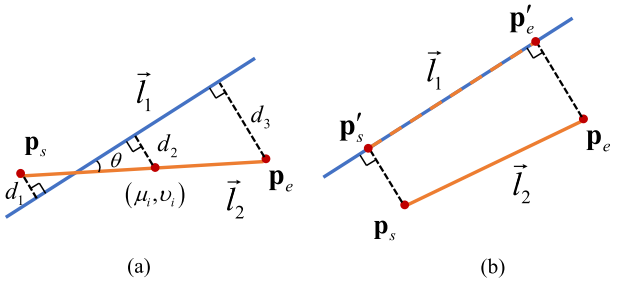
In this paper, the 2D lines are detected by a fast line detection (FLD) method [52], and are indicated by two endpoints $(\mathbf{p}_s, \mathbf{p}_e)$ in 2D image. The line features existing in 3D point clouds are extracted by a segment-based 3D line detection method [50]. In the method, the 3D points are segmented by planes and projected into a virtual image, then the 2D lines detected on this image are re-projected into 3D space to get line segments. Once the 3D lines are obtained, the dense point cloud can be replaced by line segments that are represented by two 3D endpoints $(\mathbf{P}_s, \mathbf{P}_e)$, and only these lines need to be loaded during visual localization.

To get line correspondences, we first project the 3D lines into the camera imaging plane according to the initial pose of the camera from VINS. Then, these lines are matched with the 2D lines captured from the actual world in this image depending on the defined similarity criteria. The criteria in this paper are mainly determined by the distance, angle, and overlap between two line segments [14], [34], which are shown in Fig. 2. Specifically, the general function of a 2D line l is

$$Ax + By + C = 0, \tag{1}$$

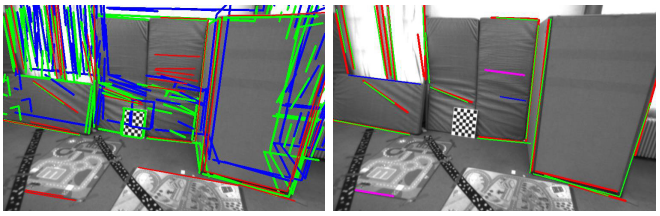
the line distance D is defined as the sum of distances from n equally interval sampling points $(\mu_i, \nu_i), i = 1 \cdots n$ (containing endpoints) on one line segment to the other (shown in Fig. 2 (a)), the equation is,

$$D = \sum_{i=1}^{n} d_i = \sum_{i=1}^{n} \frac{|A\mu_i + B\nu_i + C|}{\sqrt{A^2 + B^2}}. \tag{2}$$



(a) (b)

Fig. 2. An illustration of line similarity criteria. (a) The line distance is the sum lengths of the dashed lines, denoted by $\sum d_i$. The angle of two lines $\theta = \angle(\vec{l}_1, \vec{l}_2)$ (b) the line overlap rate refers to the percentage of the orange dashed line on the blue line.



(a) (b)

Fig. 3. (a) The red lines in the image are the detected 2D lines by FLD method; the blue lines are the projected lines from 3D map by VIO FoV, and the FoV of green lines is based on our state estimation result (More details can be seen in Section III-B). (b) The red and green line pairs are the valid matching associations. The blue and purple lines are the rejected outliers (see Section IV-B).

By assigning appropriate thresholds, which are experimentally determined, the algorithm can obtain line correspondences that meet requirements. Fig. 3 (a) shows the line detection result at one camera frame, and Fig. 3 (b) is the line matching result on this frame.

B. Point-to-Point Line-Based Optimization Model

Generally, the cost function among line features can be built based on the sum of distances between line correspondences. In this paper, the optimization model is transformed from minimizing the distance of lines to point reprojection error [51].

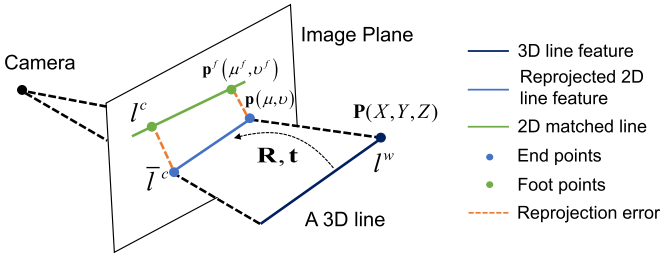


Fig. 4. An illustration of line reprojection error from the 3D prior map to the 2D image. A 3D point $\mathbf{P}(X, Y, Z)$ on l^w is projected into a camera image with $\mathbf{p}(\mu, \nu)$ based on the transformation $\mathbf{R}, \mathbf{t}$. The foot point $\mathbf{p}^f(\mu^f, \nu^f)$ of $\mathbf{p}(\mu, \nu)$ on l^c is found to build a pixel cost function that is minimized.

Assuming there is a line pair l_1, l_2, the general function of l_1 is shown in (1), and the distance between them can be calculated by the sum distance of the points on l_2 from l_1, which is consistent with Equation (2). Take one point $\mathbf{p}(\mu, \nu)$ on l_2 as an example, the distance from $\mathbf{p}$ to l_1 is equivalent to the distance from the point $\mathbf{p}$ to its vertical foot point $\mathbf{p}^f(\mu^f, \nu^f)$ on l_1. This point-to-point distance is essentially the same as the reprojection error in visual pose estimation, both of them are used to represent the difference between the coordinates of two-pixel points. Based on this, the objective function can be built by minimizing the square sum of the point pair error,

$$\mathbf{F} = \min \sum \left\| \mathbf{p}^f - \mathbf{p} \right\|_{\mathbf{Q}}^2, \tag{3}$$

where $\mathbf{p}^f$ and $\mathbf{p}$ are the measurement and prediction of this function, respectively. The $\mathbf{Q}$ is the covariance of observations and is experimentally determined. The coordinate of the foot point $\mathbf{p}^f(\mu^f, \nu^f)$ can be computed as

$$\mu^f = \frac{B^2\mu - AB\nu - AC}{A^2 + B^2}, \nu^f = \frac{A^2\nu - AB\mu - BC}{A^2 + B^2}. \tag{4}$$

The specific process of constructing an optimization model is illustrated in Fig. 4. Suppose there is a 3D line segment l^w in space at the time while the k^{th} image is taken by the camera, and one of its endpoints is $\mathbf{P}(X, Y, Z)$. The transformation of the camera during this time from the world frame is $\mathbf{T}_w^c = [\mathbf{R}|\mathbf{t}]$ (corresponding to Lie algebra $\boldsymbol{\xi}$ [53]), the 3D line l^w is projected into the image plane of the camera shown as $\overline{l^c}$, which is matched with the line segment l^c in this image. Based on the pre-calibrated extrinsic matrix $\mathbf{T}_w^c$, this projection model can be expressed as

$$s \begin{bmatrix} \mu \\ \nu \\ 1 \end{bmatrix} = \mathbf{KT} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{K} \exp\left(\boldsymbol{\xi}^\wedge\right) \mathbf{P}, \tag{5}$$

where $\mathbf{p}(\mu, \nu)$ is the pixel location of the endpoint $\mathbf{P}(X, Y, Z)$ in an image plane. s is the scale and $\mathbf{K}$ is the intrinsic matrix of the camera [54]. The symbol $\wedge$ is a vector to skew-symmetric conversion. Combined with the state variables we defined in Section II, the transformation $\mathbf{T}_w^c$ can be decomposed as

$$\begin{aligned} \mathbf{R}_w^c &= \mathbf{R}_{b_k}^c \mathbf{R}_w^{b_k} \\ \mathbf{t}_w^c &= \mathbf{R}_{b_k}^c \mathbf{t}_w^{b_k} + \mathbf{t}_{b_k}^c. \end{aligned} \tag{6}$$

where $\mathbf{R}_{b_k}^c$ and $\mathbf{t}_{b_k}^c$ refer to the transformation between the camera and IMU at the k^{th} image, which are invariant parameters because the camera and IMU are fixedly mounted on autonomous robots, so we have $\mathbf{T}_{b_k}^c = \mathbf{T}_{b_0}^c = \mathbf{T}_b^c$, and this transformation can be obtained by the joint camera-IMU calibration [55]. In our system, the initial guess of states $\mathbf{T}_w^{b_k}$ is provided by VINS [35].

In the image plane, After obtaining the prediction $\mathbf{p}(\mu, \nu)$ based on (5), the measurement $\mathbf{p}^f(\mu^f, \nu^f)$ on the matched line pair l^c can be computed by (4). So the error function depended on the point-to-point reprojection error can be shown as

$$\mathbf{e} = \mathbf{p}^f - \mathbf{p} = (\mu^f - \mu)^2 + (\nu^f - \nu)^2. \tag{7}$$

Therefore, the cost function (3) can be rewritten by

$$\boldsymbol{\xi}^* = \min_{\boldsymbol{\xi}} \sum_{k=1}^{N} \left\| \mathbf{p}_k^f(\mu^f(\boldsymbol{\xi}), \nu^f(\boldsymbol{\xi})) - \frac{1}{s_k}(\mathbf{K} \exp\left(\boldsymbol{\xi}^\wedge\right) \mathbf{P}_k) \right\|_{\mathbf{Q}}^2, \tag{8}$$

where N is the number of line pairs on the current image. The proposed method selects two endpoints from each set of pairs as observations for the optimization model. Moreover, the measurements $\mathbf{p}^f$ is a variable related to the state $\boldsymbol{\xi}$.

C. Optimization Solver

For the non-linear optimization problem in this paper, The Gauss-Newton (GN) method is employed and is implemented by the Ceres solver [56] for solving. First, the error function is linearized by Taylor expansion about $\mathbf{x}$ shown as

$$\mathbf{e}(\mathbf{x} + \Delta\mathbf{x}) \approx \mathbf{e}(\mathbf{x}) + \mathbf{J}^\top \Delta\mathbf{x}. \tag{9}$$

where $\mathbf{x}$ is the state variable, and $\mathbf{J}$ is the Jacobian matrix. If the Jacobian matrix is known, given the initial guess of states $\mathbf{x}$, the optimization increment $\Delta\mathbf{x}$ can be obtained by

$$\mathbf{H}\Delta\mathbf{x} = \mathbf{g} \tag{10}$$

where $\mathbf{H}$ is $\mathbf{J}^\top\mathbf{J}$, $\mathbf{g}$ is $-\mathbf{J}^\top\mathbf{e}$ [57]. Then the states are updated by $\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta\mathbf{x}$, until the algorithm converges, where i is the iterative index.

In terms of the Jacobian matrix in this paper, the proposed method utilizes the perturbation model in Lie algebra to derive it [51]. Based on the chain rule and (7), the derivation of each error term with respect to the state variable can be calculated by

$$\mathbf{J} = \frac{\partial \mathbf{e}}{\partial \delta\boldsymbol{\xi}} = \lim_{\delta\boldsymbol{\xi}\to 0} \frac{\mathbf{e}(\delta\boldsymbol{\xi} \oplus \boldsymbol{\xi}) - \mathbf{e}(\boldsymbol{\xi})}{\delta\boldsymbol{\xi}} = \frac{\partial \mathbf{e}}{\partial \mathbf{p}} \frac{\partial \mathbf{p}}{\partial \mathbf{P}'} \frac{\partial \mathbf{P}'}{\partial \delta\boldsymbol{\xi}}. \tag{11}$$

Here $\oplus$ refers to the disturbance left multiplication in Lie algebra, $\delta\boldsymbol{\xi}$ is the perturbation value. The $\mathbf{P}'$ is a point in the camera frame, which is transformed by the 3D space point $\mathbf{P}$ in the world frame, referring to Equation (5), we have

$$\mathbf{P}' = (\mathbf{TP})_{1:3} = \exp\left(\boldsymbol{\xi}^\wedge\right) \mathbf{P}_{1:3} = [X', Y', Z']^\top. \tag{12}$$

The specific calculation procedure for the three components of the Jacobian matrix in (11) can be found in Appendix A. Therefore, the localization problem could be solved iteratively based on the devised Jacobian matrices.

IV. Safety Quantification of Visual Localization With Prior Map

A. Observation Model

Considering the state estimation problem in the previous section, a general observation equation could be constructed as follows,

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \epsilon, \tag{13}$$

where $\mathbf{x}$ is the state variable and ϵ is the noise item. $\mathbf{z}$ is the measurements, corresponding to the foot point $\mathbf{p}^f$. $\mathbf{h}(\cdot)$ is a nonlinear measurement model and corresponds to (5) in Section III, which can be linearized by a first-order Taylor expansion as shown in Section III-C to get an approximated linear model,

$$\hat{\mathbf{z}} = \mathbf{J}\Delta\mathbf{x} + \epsilon \tag{14}$$

where $\mathbf{J}$ is the Jacobian matrix. $\hat{\mathbf{z}} = \mathbf{z} - \mathbf{h}(\mathbf{x}_0)$ is the shifted measurement according to the operating point $\mathbf{x}_0$ (initial guess of state variables). Here we assume that the error introduced by the linearization is negligible. As in the case of visual localization, the frame frequency is high and the movement between successive frames is small, making the operating point close to the state convergence value [58].

Generally, the noise item ϵ is assumed to be a Gaussian distribution with zero mean and a known covariance matrix $\mathbf{Q}$,

$$\epsilon \sim \mathcal{N}(0, \mathbf{Q}). \tag{15}$$

However, in this section a fixed bias $\boldsymbol{b}$ is added into the error model as

$$\epsilon \sim \mathcal{N}(\boldsymbol{b}, \mathbf{Q}). \tag{16}$$

The bias $\boldsymbol{b}$ represents another source of error, here called outliers, the details will be discussed in the following section.

For the linear problem in (14), we can solve it by the weighted least squares method, and the solution is

$$\Delta\hat{\mathbf{x}} = (\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}\mathbf{J}^\top\mathbf{W}\hat{\mathbf{z}}, \tag{17}$$

where $\mathbf{W}$ is the weighted matrix or information matrix and is the inverse of noise covariance matrix $\mathbf{Q}$, Note that the Jacobian matrix $\mathbf{J}$ could be updated in each iteration.

B. Outlier Rejection

Referring to Zhu's work [48], we could briefly classify the errors associated with the raw measurements (line features in this paper) in the proposed framework as follows. The first category is the photometric noise caused by camera calibration error, camera over-exposure, and motion blur. These errors will affect the detection accuracy of line features and are expressed as the covariance matrix $\mathbf{Q}$ of the error model in (16). The second category is the fixed estimation deviations introduced by outliers, and we use the bias $\boldsymbol{b}$ in (16) to represent the effect of the outliers on the state estimation. However, it is undeniable that the presence of outliers is associated with the first category, as the category can cause feature misdetection and mismatching.

Due to the existence of noise and bias in measurements, the results of state estimation could contain residual ε. Combining (14) and (17), the residual ε can be written as

$$\begin{aligned} \varepsilon &= \hat{\mathbf{z}} - \mathbf{J}\Delta\hat{\mathbf{x}} = (\mathbf{I} - \mathbf{P})\hat{\mathbf{z}} \\ \mathbf{P} &= \mathbf{J}(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}\mathbf{J}^\top\mathbf{W}. \end{aligned} \tag{18}$$

In the actual calculation, the residual is obtained from the optimization solver in the previous section.

To evaluate the quality of measurements, a weighted sum of the squared errors (WSSE) [37] based on the residual is defined as

$$\varepsilon^\top\mathbf{W}\varepsilon = [(\mathbf{I} - \mathbf{P})\hat{\mathbf{z}}]^\top\mathbf{W}[(\mathbf{I} - \mathbf{P})\hat{\mathbf{z}}], \tag{19}$$

which can be further simplified as

$$\begin{aligned} \varepsilon^\top\mathbf{W}\varepsilon \Rightarrow \hat{\mathbf{z}}^\top\mathbf{W}(\mathbf{I} - \mathbf{P})\hat{\mathbf{z}} &= \hat{\mathbf{z}}^\top\mathbf{S}\hat{\mathbf{z}} \\ \mathbf{S} &= \mathbf{W}(\mathbf{I} - \mathbf{J}(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}\mathbf{J}^\top\mathbf{W}). \end{aligned} \tag{20}$$

Next, a Chi-squared test for WSSE is used to determine whether there are outliers in the measurements:

$$\hat{\mathbf{z}}^\top\mathbf{S}\hat{\mathbf{z}} > \chi^2_{1-\alpha}(n - m) \equiv \mathbf{TD}, \tag{21}$$

where, $\chi^2_{1-\alpha}(n - m)$ is the $1 - \alpha$ quantile of the central Chi-squared distribution with $n - m$ degrees of freedom (DoF) [38]. n is the number of measurements, m is the number of state variables, α is the *probability of false alarm* (P_{fa}) and is set to 0.05 in this paper. $\mathbf{TD}$ is a threshold that can be obtained by checking the χ^2 distribution table according to the DoF and quantile $1 - \alpha$. If

$$\hat{\mathbf{z}}^\top\mathbf{S}\hat{\mathbf{z}} > \mathbf{TD}, \tag{22}$$

the probability of outliers existing in the measurements is $1 - \alpha$ (95%).

Because of the properties of the χ^2 distribution and the error model (16), outliers in the measurements can be rejected by (21). If the bias $\boldsymbol{b}$ in (16) is not equal to 0, the WSSE will not fit a central χ^2 distribution but a non-central distribution. The non-centrality parameter is $\lambda \equiv \boldsymbol{b}^\top\mathbf{S}\boldsymbol{b}$ [38], which is shown as

$$E(\hat{\mathbf{z}}^\top\mathbf{S}\hat{\mathbf{z}}) - (n - m) = \boldsymbol{b}^\top\mathbf{S}\boldsymbol{b}. \tag{23}$$

Therefore, the central Chi-squared distribution in (21) cannot be satisfied only if all outliers are theoretically removed from the observations (i.e. the bias in the error model is equal to 0). In this way, a greedy algorithm [47] can be applied to iteratively exclude the observations corresponding to the largest residuals (considered to be outliers) until the (21) is satisfied.

C. Protection Level Estimation

Even if the Chi-squared test is passed, one or more outliers may still remain in the measurements and the existence of the potential outliers is denoted by the P_{fa}. Therefore, the error of the localization results based on the surviving measurements is mainly caused by two factors: (1) the potential missed outliers described by the probability P_{fa}. (2) the Gaussian noise associated with the surviving measurements.

The general expression of PL can be expressed as

$$\mathbf{PL} = \mathbf{P}_b + \mathbf{P}_n. \tag{24}$$

In particular, the first term is used to protect the bias-induced error in localization, and the second term is used to bound the noise-induced error. Meanwhile, this paper aims to consider the case where multiple outliers are contained in observations even after the Chi-squared test. Referring to RAIM and [47], we divide the translation into the x, y, z, and rotation into the $roll$, $pitch$, and yaw when calculating the PL in the proposed method, and then use an index vector formed by 0 and 1 to represent a state variable, for example

$$\begin{aligned} &x : \mathbf{H}_1 = [1,0,0,0,0,0]; \mathit{roll} : \mathbf{H}_4 = [0,1,0,0,0,0] \\ &y : \mathbf{H}_2 = [0,0,1,0,0,0]; \mathit{pitch} : \mathbf{H}_5 = [0,0,0,1,0,0] \\ &z : \mathbf{H}_3 = [0,0,0,0,1,0]; \mathit{yaw} : \mathbf{H}_6 = [0,0,0,0,0,1]. \end{aligned} \tag{25}$$

In terms of the first term in PL in (24), the bias vector for the i^{th} state can be given based on error propagation and (17) [38]:

$$\boldsymbol{b}_i^* = \mathbf{H}_i(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}\mathbf{J}^\top\mathbf{W}\boldsymbol{b}, \tag{26}$$

so the norm of the bias vector is

$$\begin{aligned} \|\boldsymbol{b}_i^*\| &= \sqrt{\boldsymbol{b}^\top\mathbf{D}\boldsymbol{b}} \\ \mathbf{D}_i &= \mathbf{W}\mathbf{J}(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}\mathbf{H}_i^\top\mathbf{H}_i(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}\mathbf{J}^\top\mathbf{W}. \end{aligned} \tag{27}$$

For the multiple-outliers case, assuming there are n measurements in total, r of them have biases, and $1 \le r \le (n-m)$, for example $n = 7$, $r = 2$. Since we do not know which two observations are true outliers, An index matrix cluster $\mathbf{A}_j$ composed of 0 and 1 with size $n \times r$ is defined to choose the two outliers from the seven observations, where the observation referred to by number 1 is an outlier. For instance,

$$\mathbf{A}_1 = \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0 \end{bmatrix}^\top, \tag{28}$$

which means the first and second measurements are outliers. In this case, matrix cluster $\mathbf{A}_j$ need to iterate through all combinations of "two of seven", which is denoted by $C_7^2 = \frac{7\times6}{2!} = 21$, so $j = 21$ this cluster has 21 different matrices. Then, the biases in observations can be selected iteratively by

$$\boldsymbol{b}_{ij} = \mathbf{A}_j\boldsymbol{b}_i, \tag{29}$$

To make sure the PL is to bound the error caused by biases, the maximum bias-induced error results from the example $n = 7$, $r = 2$ needs to be calculated. However, since the Chi-squared test has been passed, the condition $\boldsymbol{b}^\top\mathbf{S}\boldsymbol{b} \le \mathbf{TD}$ also needs to be satisfied [38]. Based on (27) and (29), a constrained optimization model can be constructed as

$$\begin{aligned} \mathbf{P}_{b_i} &= \max \boldsymbol{b}_{ij}^* = \sqrt{\boldsymbol{b}_{ij}^\top\mathbf{D}\boldsymbol{b}_{ij}} \\ &= \max_{\mathbf{A}_j\in\mathbf{A}_r, \boldsymbol{b}_i} \sqrt{\boldsymbol{b}_i^\top\mathbf{A}_j^\top\mathbf{D}_i\mathbf{A}_j\boldsymbol{b}_i} \\ s.t.\ &\boldsymbol{b}_i^\top\mathbf{A}_j^\top\mathbf{S}\mathbf{A}_j\boldsymbol{b}_i = \mathbf{TD}, \end{aligned} \tag{30}$$

where, $\mathbf{S}$ and $\mathbf{D}_i$ can be found in (20), (27). According to [38] this optimization question can be equally simplified by linear algebra theory as follows,

$$\max \boldsymbol{b}_{ij}^* = \max_{\mathbf{A}_j\in\mathbf{A}_r} \sqrt{\Lambda_{\max}(\mathbf{A}_j, \mathbf{D}_i, \mathbf{S})\Gamma}, \tag{31}$$

where, $\Gamma = \mathbf{TD}$, $\Lambda_{\max}(\mathbf{A}_j, \mathbf{D}_i, \mathbf{S})$ is the largest eigenvalue of

$$\mathbf{A}_j^\top\mathbf{D}_i\mathbf{A}_j(\mathbf{A}_j^\top\mathbf{S}\mathbf{A}_j)^{-1}. \tag{32}$$

By iterating through all $\mathbf{A}_j$ and comparing the corresponding largest eigenvalue of (32), we can obtain the maximum bias-induced error in this direction.

On the other hand, the noise-induced error can be computed by

$$\mathbf{P}_{n_i} = k_i\sqrt{[(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}]_{i,i}}, \tag{33}$$

where k is the number of the standard deviation. Empirically, the k is set to 3, so-called three-sigma rule of thumb (or 3σ rule), the corresponding probability of the values that lie within the 3σ interval is 99.73% [59]. Substituting (31),(33) into (24), we got,

$$\mathbf{PL}_i = \max_{\mathbf{A}_j\in\mathbf{A}_r} \sqrt{\Lambda_{\max}(\mathbf{A}_j, \mathbf{D}_i, \mathbf{S})\Gamma} + k_i\sqrt{[(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}]_{i,i}}. \tag{34}$$

The integral calculation of PL can be obtained from the above equation.

V. Experiment Results and Discussions

The proposed framework is evaluated in real-world environments using both the public indoor dataset for the AAV and the outdoor dataset collected by UGV. We compare the performance of the proposed method with other state-of-the-art methods to evaluate the proposed optimization model and the safety-quantification method. The root mean squared error (RMSE) of absolute trajectory error (ATE) [60] is the evaluation criterion of algorithm accuracy. All experiments are implemented on a desktop with Intel Core i9-12900K and Ubuntu 20.04. The proposed framework is developed by the robot operating system (ROS) [61] and the C++ language. In terms of the evaluation of the localization performance, the algorithms compared include:

1) ***VINS***: the VINS framework developed in [35].
2) ***Benchmark***: the line-to-line visual localization method developed in [34].
3) ***PPL***: the proposed visual localization method without outlier rejection.
4) ***PPL-OR***: the proposed visual localization method with outlier rejection.

To evaluate the proposed safety-quantification function, the estimated PL is compared with the exact localization error. It is certified as safe if the estimated PL could effectively account for the potential error of the localization solution. The comparison methods include:

1) ***3σ***: a typical method used to represent the uncertainty of the state [47], the computational model can be seen in (33).
2) ***PL***: the proposed safety-quantification model.

A. AAV Indoor Experiment

1) Dataset and Comparison Methods: The EuRoc MAV (Micro Aerial Vehicle) dataset [62] is selected for evaluation, which is collected by global shutter stereo cameras

TABLE I
RMSE [60] OF ATE (M)

Sequence	VINS [35]	Benchmark [34]	PPL [ours]	PPL-OR [ours]
V1_01_esay	**0.098**	0.164	0.153	0.152
V1_02_medium	0.110	0.152	**0.092**	0.099
V1_03_difficult	0.189	0.217	0.186	**0.161**
V2_01_esay	**0.096**	0.192	0.185	0.166
V2_02_medium	0.167	0.281	0.155	**0.136**
V2_03_difficult	0.327	0.441	0.319	**0.312**

RMSE of ATE: The root mean squared error of absolute trajectory error.

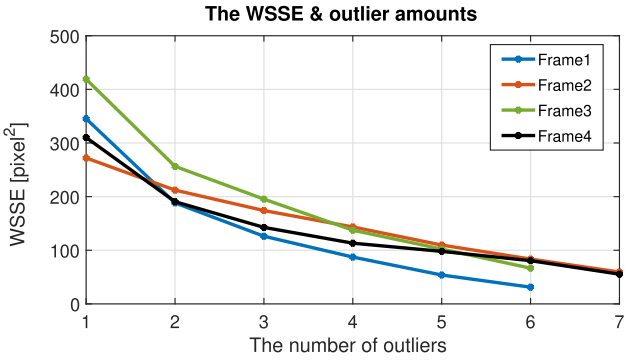


Fig. 5. The plot of the relevance WSSE and the number of outliers. The 4 frames are selected from the V1_02_medium dataset. As the outliers are detected and excluded in turn, the residual gradually decreases.

(Aptina MT9V034, WVGA burri2016euroc, 20 FPS), synchronized IMU unit (ADIS16448, 200 Hz). The 3D point cloud prior maps are constructed by a Leica MS50 with an accuracy of about 1 mm, and the ground-truth states are obtained by a Vicon (100 Hz) and Leica MS50. The dataset includes six trajectories in two different rooms (V1 and V2), and there are three separate tracks of varying difficulty in each room. The MAV can be seen in Fig. 1. In particular, our current algorithm applies the direct output of VINS without loop-closure for state initialization. However, in the proposed framework, the 3D point cloud prior map is introduced to compensate for VINS drift while obtaining absolute localization results.

2) Verification of Optimization Model and Outlier Rejection: The comparison results of the pose estimation accuracy of VINS [35], benchmark [34], and ours are presented through the RMSE of ATE in Table I. It can be seen that the proposed methods outperform benchmark in all scenarios. Specifically, the average accuracy improvements of ours over benchmark are 27.1% and 22.8% with and without outlier rejection (OR). Among that, the difference between PPL (our method without OR module) and benchmark is the optimization model based on the line features. This experimental result validates the proposed optimization model has better convergence performance than benchmark. The medium and difficult datasets in Table I refer to the fast flying speed and trajectory maneuverability of AAVs, which is challenging for visual-based localization methods. As shown in Table I, in medium and difficult scenarios, the accuracy of our method with OR (PPL-OR) is 12.0% and 35.4% better than VINS and benchmark, respectively. This means that our method based on structured lines is more resistant to disturbances in complex motion scenarios.

In addition, the proposed method with outlier rejection (PPL-OR) generally has a better performance than PPL, which reflects the effectiveness of the Chi-squared test-based outlier rejection algorithm. Details about the OR efficiency are shown in Fig. 5, which displays the variation of WSSE of four keyframes as the outliers are detected and removed by a greedy algorithm sequentially. The x-axis is the number of outliers removed and the y-axis represents the residual of the observation model. By sequentially rejecting the visual outliers, the WSSE decreases.

3) Verification of PL: This subsection presents a protection level-based safety quantitative analysis by taking the classic V1_02_medium sequence as an example. In this experiment,

TABLE II
THE b_r (%) OF PL / 3σ WITH ERROR USING DIFFERENT NUMBER OF OUTLIERS IN TRANSLATION AND ROTATION. THE COVARIANCE IS 7-PIXEL^2

Num.	1	2	3
x	**71.6%**/ 19.6%	**74.2%**/ 19.6%	**74.2%**/ 19.6%
y	**75.7%**/ 24.8%	**76.8%**/ 24.8%	**76.8%**/ 24.8%
z	**71.3%**/ 22.9%	**73.5%**/ 22.9%	**73.5%**/ 22.9%
$roll$	**43.9%**/ 20.8%	**65.2%**/ 20.8%	**65.2%**/ 20.8%
$pitch$	**66.6%**/ 31.0%	**85.3%**/ 31.0%	**85.3%**/ 31.0%
yaw	**8.23%**/ 1.99%	**18.8%**/ 1.99%	**18.8%**/ 1.99%

TABLE III
THE b_r (%) OF PL / 3σ WITH ERROR USING DIFFERENT COVARIANCE ASSUMPTION IN TRANSLATION AND ROTATION. THE NUMBER OF OUTLIERS IS TWO

Cov.	3-pixel^2	5-pixel^2	7-pixel^2
x	**42.5%**/ 7.51%	**59.8%**/ 13.5%	**74.2%**/ 19.6%
y	**47.1%**/ 10.0%	**64.2%**/ 18.3%	**76.8%**/ 24.8%
z	**48.0%**/ 11.8%	**61.3%**/ 16.0%	**73.5%**/ 22.9%
$roll$	**36.5%**/ 11.0%	**51.8%**/ 15.3%	**65.2%**/ 20.8%
$pitch$	**60.3%**/ 14.8%	**74.8%**/ 22.5%	**85.3%**/ 31.0%
yaw	**6.03%**/ 0.81%	**10.9%**/ 1.53%	**18.8%**/ 1.99%

we compare the bound rate of PL and 3σ (33) [47] on state estimation errors, the bound rate b_r can be computed as

$$b_r = \frac{n}{m} \tag{35}$$

where, n is the number of $PL/3\sigma \geq error$, m is the number of all frames. If the estimated PL could bound the actual error of localization solution, it indicates that the proposed safety quantification is effective.

The results of translation and rotation are shown in Fig. 6 and Fig. 7, respectively, expressed by XPL, YPL, ZPL, RollPL, PitchPL, and YawPL. In Fig. 6, it is clear PL performs much better than 3σ for error bounding, with PL having more than 70% bound rate in all three directions, while 3σ is only about 20% (see Table III). And it can be seen that PL has a similar trend to the error in most cases. In Fig. 7, PL quantifies

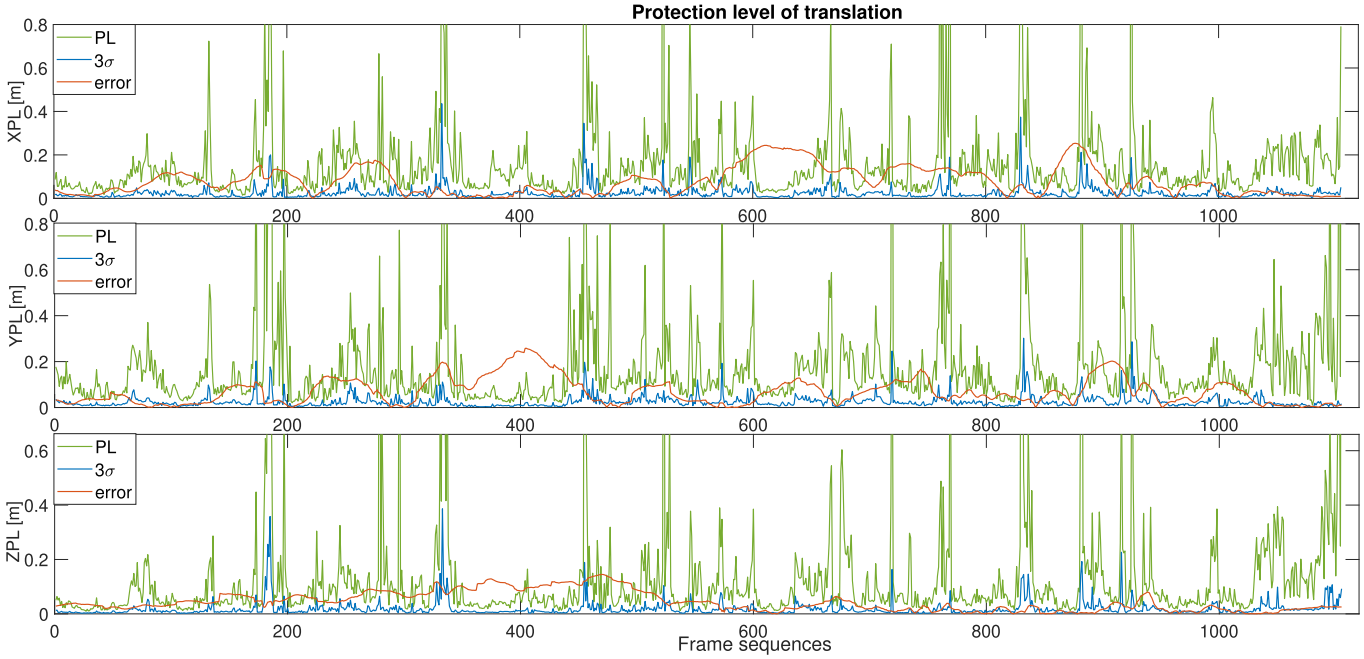


Fig. 6. The comparison results of protection level, 3σ, and estimation error in translation under V1_02_medium sequence. The measurement covariance is 7-pixel^2 and the number of outliers is set to two.

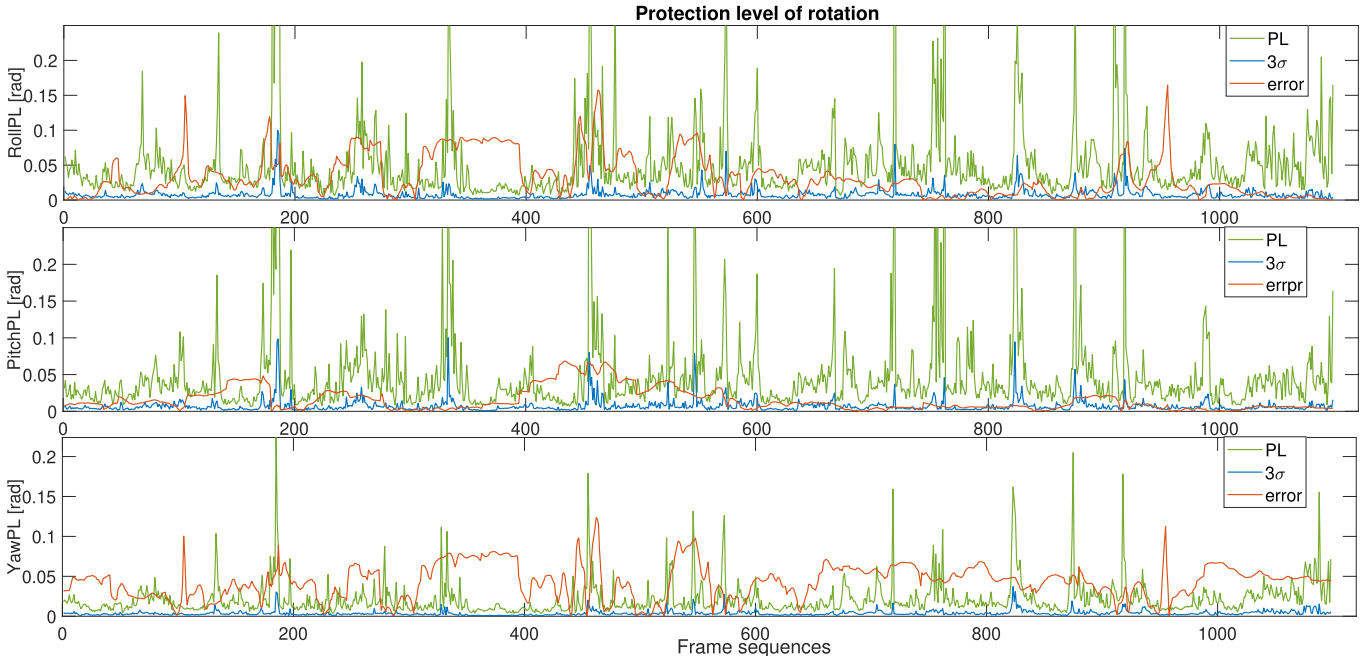


Fig. 7. The comparison results of protection level, 3σ, and estimation error in rotation under V1_02_medium sequence. The measurement covariance is 7-pixel^2 and the number of outliers is set to two.

the error significantly better than 3σ, but its performance is uneven in three directions, especially in yaw rotation. The main reasons for this phenomenon are the limitations of the line features and the IMU sensor. In terms of line features, since there are many horizontal lines in space, when the object is rotated around z-axis to produce yaw rotation, the motion will be parallel to the lines making the motion useless for feature matching and state optimization, which is one of the most common degradation symptoms of line features [31].

4) The Discussion on PL: Ideally, the PL should be able to cover the error, but the facts contradict this. Based on the derivation of PL in Section IV-C, the calculation of PL is affected by bias-induced error resulting from outliers and noise-induced error because of observation noises. Firstly, the number of outliers existing in the current observations after passing the Chi-square test is unknown. Secondly, the noise covariance matrix of the visual measurements cannot be obtained exactly, which is given empirically in this paper. All

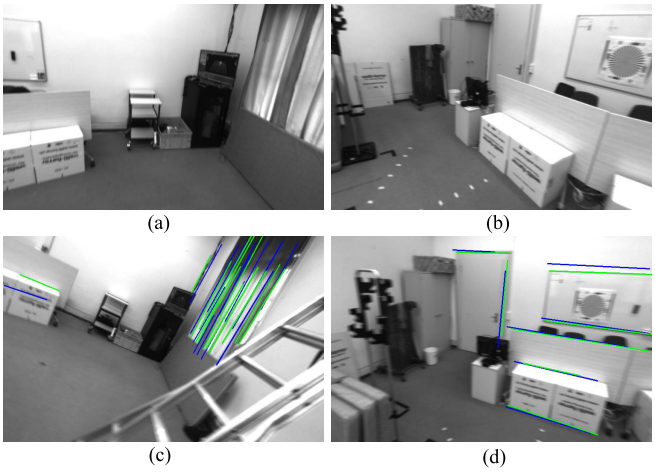


Fig. 8. (a), (b) two examples of the motion blur. (c), (d) two examples of the feature degeneration. The blue and green lines are the feature matching pairs.

these uncertainties are factors that can affect the performance of PL estimation.

To gain more insight into PL, we evaluate the bound rate of PL and 3σ under different assumptions with multiple outliers and noise covariances in Table II and Table III, respectively. From Table II, it can be seen that increasing the number of outliers can improve the bound rate of PL, but when the number of outliers is set to 3 or more, it will not change, indicating that the largest number of outliers in all sequences is 2. In Table III, the measurement noise of the line feature is set to 3, 5, and 7 pixels, respectively. In particular, the measurement noise of a single pixel point feature is usually set between 1-3 pixels [63]. Since this paper uses line features, the detection of lines will introduce more errors than point features. Also, the 3D line features extracted from the point cloud map could impose observation error, so the covariance assumption is amplified here. It can be seen from Table III that as the covariance increases, the rate rises, which means PL is getting larger. In summary, more outliers and larger covariance increase PL, which is consistent with the definition of PL for quantifying safety, where larger error noise could lead to larger PL.

5) The PL and Feature Degeneration: It is well known that when discussing localization safety, it is inevitable to consider the impact of the feature quality on it [64]. Here, the quality of features can be considered in terms of the feature detection error and the feature distribution. In this paper, PL is used as a metric for safety quantification, so we will discuss the impact of features on the reliability of safety quantification around anomalous PL values. The following analysis is done for two special cases. First, near frame 400 in Fig. 6, the PL is too conservative to bound the localization error. Two images in the area around frame 400 are shown in Fig. 8 (a) and (b). It is clear to see that images suffer from severe motion blur, which could potentially trigger additional detection errors in feature measurements [48]. Since the feature covariance that we assume during PL computation is not dynamically changing, PL is unable to accurately estimate localization errors in scenarios where feature detection errors increase, which means that the safety quantification is inaccurate when the feature detection error is suddenly large.

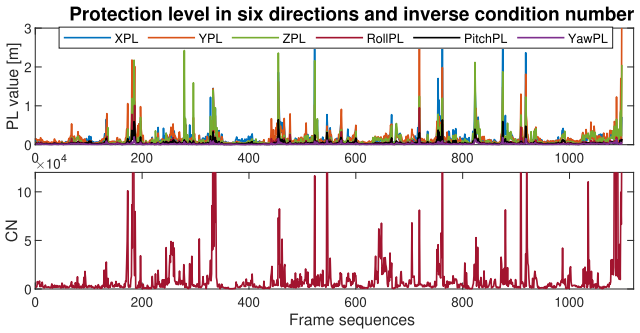


Fig. 9. The comparison of condition number (CN) and protection level in six transformation directions. The assumption of the measurement covariance is 7-pixel and the number of outliers is set to two.

In the second case, the anomalous PL values were significantly larger than the localization error and were not informative, such as frame 200. In addition to the feature detection errors described above, the spatial distribution of the line features utilized for the localization is the other source affecting the PL estimation. It is expected that the features are evenly distributed within the field of view, which could introduce ideal constraints. However, the features can flock together where the degeneration of the state estimation will occur [65]. To assess the impacts of the feature distribution on the localization performance, the condition number (CN) is introduced in [66]. In particular, CN is determined by the ratio between the maximum and minimum eigenvalues of $\mathbf{J}^\top\mathbf{J}$, which can be shown as

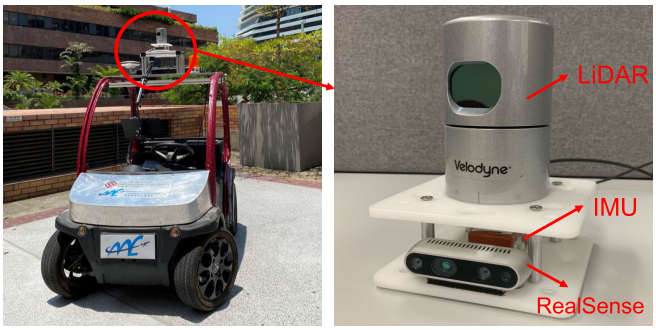
$$\kappa = \frac{\sigma_{max}}{\sigma_{min}} \tag{36}$$

where the $\mathbf{J}$ denotes the Jacobian matrix of the visual line constraint, which is rigorously derived in the appendix of this paper. A low condition number means being well-conditioned, while a high condition number indicates ill-conditioning.

Fig. 9 displays the CN and PL in six directions. As seen from the figure, PL has a highly consistent trend with CN at the anomalous mutation, which indicates these frames have ill-conditional constraints and feature degeneration. Based on this discovery, we can conclude that feature degeneration largely causes exceptional maxima peaks in PL. On the other hand, we could utilize PL to remind the feature degeneration phenomenon during state estimation to avoid serious optimization failures. Fig. 8 (c) and (d) show two typical feature degeneration scenarios. It is clear that most of the features in images are parallel, which makes the constraint with its vertical direction very weak, resulting in an ill-conditioned case.

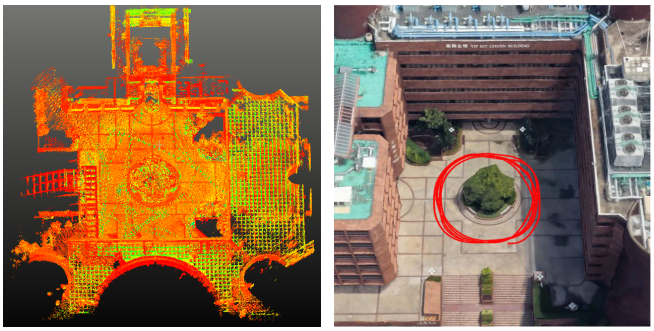
B. UGV Outdoor Experiment

1) Experiment Setup: The UGV experiment is conducted on the CARLA simulator [67] and real-world environment in the Hong Kong Polytechnic University (PolyU) campus. The real-world UGV platform used in this experiment is shown in Fig. 10 (a). The detailed sensor module is displayed in Fig. 10 (b), which contains a LiDAR (HDL 32E Velodyne, 10 Hz), an IMU (Xsens Mti 10, 200 Hz), and a RealSense camera (D435i, 640 × 480, 30 Hz). The dataset is collected by driving around a square on the Hong Kong Polytechnic



(a) (b)

Fig. 10. (a) The UGV platform loaded with customized sensor unit. (b) The detailed Sensor unit including LiDAR, IMU, and RealSense camera.



(a) (b)

Fig. 11. (a) The regional point cloud of the Hong Kong Polytechnic University. (b) The corresponding Google 3D map of the same area. The red line indicates the trajectory of the UGV during the testing.

University (PolyU) campus (Fig. 11). The total path length is 295.5 m. The IMU data and images captured by RealSense are the input of VINS. The data from LiDAR and IMU are processed by LIO_SAM implementation [68], and used to build a prior point cloud map shown in Fig. 11 (a), the origin of this 3D map is defined at the starting position of the IMU. The trajectory estimated by LIO_SAM is regarded as the ground truth of localization. With the help of the LiDAR loop closure, centimeter-level accuracy can be obtained from the LIO_SAM in the evaluated scene.

In CARLA simulator, a vehicle equipped with a 64-channels LiDAR (rotation frequency 100, range 200m), IMU (100Hz, measurement noise 10^{-2}), bias random walk 10^{-5}), and a monocular camera (resolution 900 $\times$ 600, FoV 90 degree) derived around a town to construct prior point cloud map and dataset. The view of the town and the trajectory of the vehicle are shown in Fig. 12. In this town, we run three different routes with lengths of 1km (red trajectory labeled with CARLA_01), 2km (green trajectory labeled with CARLA_02), and 3km (blue trajectory labeled with CARLA_03). The speed of the car is 30km/h. These three paths pass through a variety of scenarios, such as urban canyons, tunnels, and shrubs, which are used to verify the effectiveness of our algorithm compared to the VINS and benchmark. Accordingly, similar evaluation criteria and comparisons with AAV experiment are adopted in this section.

2) System Evaluation: Table IV lists the RMSE of ATE comparison results of the proposed method with VINS and benchmark. It can be seen that our method outperforms others

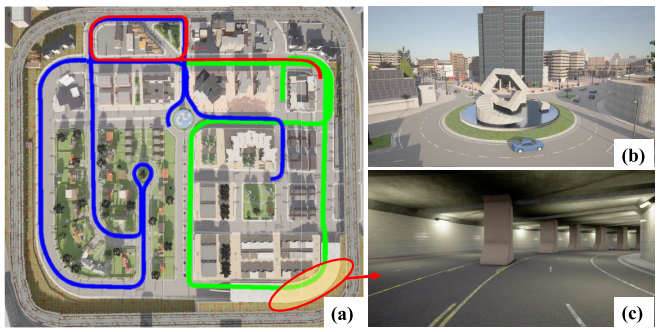


Fig. 12. (a) The Carla simulation town and three driving trajectories with different lengths. (b) A scene in the town. (c) The tunnel scene marked by the red circle in the figure (a).

TABLE IV
RMSE [60] OF ATE (M)

Dataset	VINS [35]	Benchmark [34]	PPL [ours]	PPL-OR [ours]
PolyU	1.688	2.191	1.607	**1.602**
CARLA_01	2.380	3.626	2.137	**1.931**
CARLA_02	4.613	4.312	3.677	**3.321**
CARLA_03	7.894	7.986	7.733	**7.311**

RMSE of ATE: The root mean squared error of absolute trajectory error.

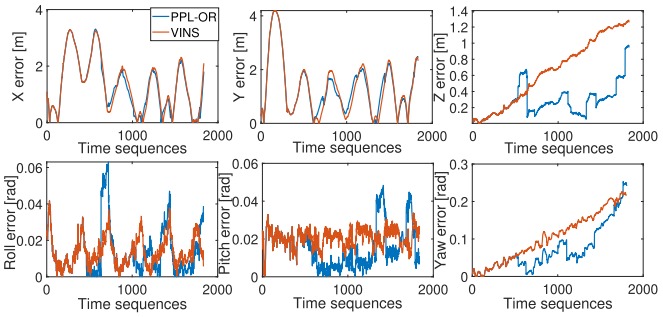


Fig. 13. The position and orientation error of PPL-OR compared with VINS.

in CARLA simulated dataset and real-world environment. Specifically, As the running trajectory grows gradually in CARLA dataset, the proposed methods have higher accuracy than VINS by 18.91%, 27.98%, 5.10%, and than benchmark by 46.83%, 22.97%, 8.46% in 1-3km. Meanwhile, our method (PPL-OR) improves the localization accuracy by 5% and 26.9% over VINS and benchmark in the real-world environment. Fig. 13 presents the estimation errors of translation and rotation, it is clear our algorithm is able to reduce the drift problem of VINS to some extent, which is evident in the z and yaw motions, while the errors in other directions can be offset periodically because the UGV trajectory is a constantly repeating circular motion. However, as the path keeps getting longer and the VINS drift increases, the effectiveness of our algorithm for removing the cumulative error gradually diminishes (see Fig. 13) because the proposed method uses the output of VINS as the initial guess.

Additionally, the time statistics of VINS, benchmark, and our method on three datasets are displayed in Table VI,

TABLE V

THE b_r (%) OF PL WITH ERROR USING DIFFERENT NUMBERS OF OUTLIERS IN TRANSLATION AND ROTATION WITH COVARIANCE IS 7-PIXEL

Num.	x	y	z	$roll$	$pitch$	yaw
1	83.1%	66.2%	50.8%	64.6%	53.9%	5.13%
2	85.1%	69.2%	54.9%	66.7%	56.4%	5.64%
3	85.1%	69.2%	54.9%	66.7%	56.4%	5.64%

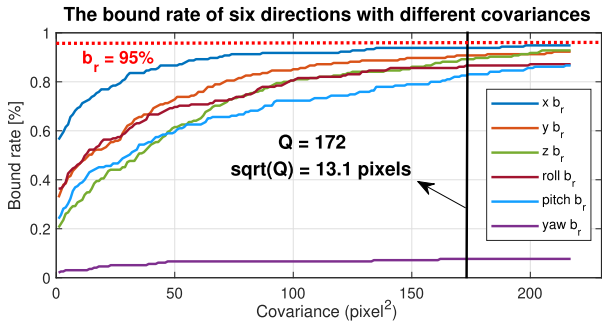


Fig. 14. The bound rate (b_r) of six directions with different measurement noises from 3-pixel to 15-pixel. The red dash line means the $b_r = 95\%$. The covariance corresponding to the black line is 13.1-pixel^2.

TABLE VI

TIME STATISTICS OF FEATURE PROCESSING / OPTIMIZATION TREADS (MS)

Datasets	VINS [35]	Benchmark [34]	PPL-OR [ours]
EuRoC	5.8 / 15.4	11.3 / 7.5	9.9 / 1.5
PolyU	2.9 / 23.3	5.6 / 4.8	4.8 / 1.3
CARLA_01	8.2 / 20.1	48.9 / 52.1	46.5 / 1.8

where Feature Processing includes the feature detection and matching modules. Optimization consists of pose estimation and marginalization modules. The optimization computation of our method is fast due to the fact that we use fewer line constraints than point constraints during pose estimation, as well as the proposed outlier rejection module refines the constraints to accelerate the convergence of the estimator. However, on the CARLA dataset, the feature processing module of the prior-map based methods has significantly increased in time consumption. This is because the prior line maps are large in size, which requires the algorithm to spend more time completing cross-modal matching. Overall, our algorithm satisfies the real-time requirement while loading a reasonable prior line map. Therefore, in practice, maintaining the size of the loaded prior map within a rational range is vital to the real-time performance of the method.

3) PL Result: Based on the discussion of PL in the previous subsection, the safety quantification of UGV in the real-world environment is verified by adjusting the number of outliers and the covariance of measurements. By fixing the covariance to 7 pixel^2, the relationship between the bound rate of PL and the number of outliers is shown in Table V. It is clear that the number of residual outliers in the UGV dataset is two, which is the same as the AAV experiment. Next, by fixing the number of outliers to 2 and changing the covariance from

TABLE VII

THE b_r (%) OF PL / 3σ UNDER SIX TRANSFORMATION DIRECTIONS WITH COVARIANCE IS 13.1-PIXEL^2 AND OUTLIERS ARE 2

	x	y	z	$roll$	$pitch$	yaw
PL	**93.9%**	**89.7%**	**88.2%**	**85.6%**	**80.5%**	**7.69%**
3σ	40.0%	16.4%	6.67%	20.5%	12.8%	0.51%

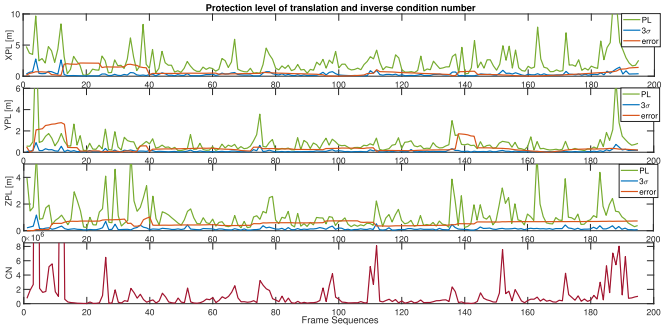


Fig. 15. The CN and comparison of protection level, 3σ, and error in three translation directions. The assumption of the measurement covariance is 13.1-pixel^2 and the number of outliers is set to two.

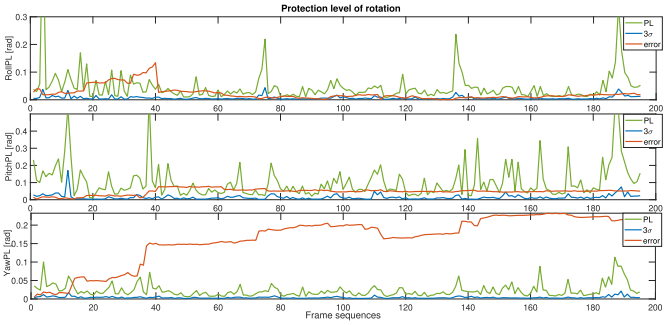


Fig. 16. The comparison results of protection level, 3σ, and error in three rotation directions. The assumption of the measurement covariance is 13.1-pixel^2 and the number of outliers is set to two.

3-pixel^2 to 15-pixel^2, the trends of the bound rate of the PL in six directions are shown in Fig. 14. As seen from the figure, the bound rates of the PL in six directions gradually level off when the covariance is equal to 13.1-pixel^2 (the black line). In particular, the bound rate in x-direction is infinitely close to 95% (the red dashed line), which is consistent with the probability $(1-\alpha)$ set in the outlier rejection section. Based on this figure, the final covariance in the UGV experiment is set to 13.1-pixel^2, and the corresponding bound rates of PL and 3σ are shown in Table VII. Fig. 15 and Fig. 16 are the trends of PL, 3σ, error, and the ICN in translation and rotation, respectively.

4) Discussion: In terms of accuracy improvement, the proposed method can eliminate the cumulative drift of VINS to a certain extent by using prior map-based localization. However, because of the loosely coupled form with VINS, the proposed method is greatly influenced by the initial guess of the state from VINS, and when the drift is severe, our method will have difficulty converging (Fig. 13). Furthermore, the measurement noise of the line feature in UGV dataset is about 13 pixels, which is even greater than the noise in AAV experiment. Excluding the noise of 2D line feature detection in images,

the 3D line detection in point cloud map could induce more errors than AAV experiment because of the low-precision point cloud. It follows that the denseness and precision of the point cloud map are also important factors affecting the performance of the algorithm and the PL calculation, as they impact the line detection and the error model.

Meanwhile, the same to the AAV experiment, the performance of PL is particularly poor at the yaw rotation. Comparing the localization error and PL of the yaw axis with the $roll$ and $pitch$ axes in Fig. 7 and Fig. 16, it is obvious that the overall error of the yaw axis is larger than the others, and the PL is indeed smaller. Apart from the degeneration caused by the poor geometry of the visual line features, the error involved in the IMU is another source leading to the conservative estimation of the PL. On the one hand, the yaw angle of the IMU is not directly observable by the system [69]. On the other hand, the circular trajectory of the UGV experiment leads to additional challenges for the yaw estimation, which will excite more errors on the yaw. As a result, the accuracy of the initial guess of the state from the VINS is not guaranteed, which can lead to poor estimation of the yaw. This phenomenon is shown in the yaw axis of Fig. 13, which presents a severe divergence trend. In summary, the above-mentioned reasons are leading to the poor performance of PL on the yaw axis.

VI. Conclusion and Future Work

This paper presents a safety-quantifiable line feature-based monocular visual localization framework with point cloud maps. A new point-to-point line constrained optimization model is built to achieve accurate and robust localization in point cloud maps to eliminate cumulative drift occurring in relative positioning. Furthermore, a safety quantification strategy motivated by RAIM system is introduced into the current autonomous systems to evaluate the maximum value of the state estimation errors, including three translation and three rotation directions with multiple faults assumption. The experimental results on AAV and AGV platforms show that the proposed method outperforms the comparison algorithms in accuracy and robustness, and the error quantification of PL for state estimation is efficient in the multiple outlier assumption as well as in the translational and rotational motions. We also provide an open-source implementation to benefit the community.

There is still potential for improving our algorithm. For instance, the current system's loosely coupled model with VINS has limitations in enhancing localization accuracy. Future work could explore a tightly coupled approach to improve precision and make fuller use of prior 3D map information, such as surface structures, to further refine the algorithm's performance. Moreover, the impact of prior map size on computational efficiency also needs to be considered. When the prior map has a large coverage area, the calculation can be accelerated by adjusting the map storage data structure. Meanwhile, the updating of the prior map also needs to be considered. The performance of PL predicting localization errors can be improved by incorporating dynamic error models and assessing feature quality, while also addressing practical challenges like sensor noise, poor lighting, and environments with sparse textures. In this study, the computed PL is used solely to predict state estimation errors. A promising direction for future research is to explore its broader applicability within robotic systems. Inspired by the GNSS domain [70], we could predefine a safety-related localization accuracy threshold and utilize PL-predicted errors to evaluate whether the current localization results meet operational requirements, enabling safer real-world robotic deployments. Additionally, PL's potential applications extend beyond localization; its error predictions could also be leveraged in mapping and planning processes, opening new avenues for exploration.

Appendix A
The Derivation of Jacobian Matrix

The analytical expression of the Jacobian matrix of the error function $\mathbf{e}$ with respect to state variables is derived in this section. We know the Jacobian can be computed by the chain rule, like this

$$\frac{\partial \mathbf{e}}{\partial \delta\boldsymbol{\xi}} = \lim_{\delta\boldsymbol{\xi}\to 0} \frac{\mathbf{e}\left(\delta\boldsymbol{\xi} \oplus \boldsymbol{\xi}\right) - \mathbf{e}(\boldsymbol{\xi})}{\delta\boldsymbol{\xi}} = \frac{\partial \mathbf{e}}{\partial \mathbf{p}} \frac{\partial \mathbf{p}}{\partial \mathbf{P}'} \frac{\partial \mathbf{P}'}{\partial \delta\boldsymbol{\xi}}. \tag{37}$$

And we need to solve each partial derivative of this equation, the first item based on (4) and (7) can be computed by

$$\begin{aligned}\frac{\partial \mathbf{e}}{\partial \mathbf{p}} &= \begin{bmatrix} \frac{\partial \mathbf{e}}{\partial \mu} & \frac{\partial \mathbf{e}}{\partial \nu} \end{bmatrix} \\ &= \frac{2}{A^2+B^2} \begin{bmatrix} A^2(\mu^f - \mu) + AB(\nu^f - \nu) \\ AB(\mu^f - \mu) + B^2(\nu^f - \nu) \end{bmatrix}^\top . \end{aligned} \tag{38}$$

For the second item, according to (5) and (12), $\mathbf{P}'$ is a projected point in the camera frame and its pixel location $\mathbf{p}(\mu, \nu)$ on the image plane is

$$s\mathbf{p} = \mathbf{K}\mathbf{P}', \tag{39}$$

$$\mu = f_x \frac{X'}{Z'} + c_x, \quad \nu = f_y \frac{Y'}{Z'} + c_y, \tag{40}$$

f_x, f_y, c_x and c_y is the intrinsic parameters of the camera,

$$\frac{\partial \mathbf{p}}{\partial \mathbf{P}'} = -\begin{bmatrix} \frac{\partial \mu}{\partial X'} & \frac{\partial \mu}{\partial Y'} & \frac{\partial \mu}{\partial Z'} \\ \frac{\partial \nu}{\partial X'} & \frac{\partial \nu}{\partial Y'} & \frac{\partial \nu}{\partial Z'} \end{bmatrix} = -\begin{bmatrix} \frac{f_x}{Z'} & 0 & -\frac{f_x X'}{Z'^2} \\ 0 & \frac{f_y}{Z'} & -\frac{f_y Y'}{Z'^2} \end{bmatrix}. \tag{41}$$

For the third part, $\mathbf{P}' = \mathbf{T}\mathbf{P}$ and give $\mathbf{T}$ a left perturbation $\Delta\mathbf{T} = \exp\left(\delta\boldsymbol{\xi}^\wedge\right)$, whose Lie algebra is $\delta\boldsymbol{\xi} = [\delta\boldsymbol{\rho}, \delta\boldsymbol{\phi}]^T$ [51], then,

$$\begin{aligned}\frac{\partial (\mathbf{T}\mathbf{P})}{\partial \delta\boldsymbol{\xi}} &= \lim_{\delta\boldsymbol{\xi}\to\mathbf{0}} \frac{\exp\left(\delta\boldsymbol{\xi}^\wedge\right)\exp\left(\boldsymbol{\xi}^\wedge\right)\mathbf{P} - \exp\left(\boldsymbol{\xi}^\wedge\right)\mathbf{P}}{\delta\boldsymbol{\xi}} \\ &= \lim_{\delta\boldsymbol{\xi}\to\mathbf{0}} \frac{\left(\mathbf{I} + \delta\boldsymbol{\xi}^\wedge\right)\exp\left(\boldsymbol{\xi}^\wedge\right)\mathbf{P} - \exp\left(\boldsymbol{\xi}^\wedge\right)\mathbf{P}}{\delta\boldsymbol{\xi}} \\ &= \lim_{\delta\boldsymbol{\xi}\to\mathbf{0}} \frac{\delta\boldsymbol{\xi}^\wedge \exp\left(\boldsymbol{\xi}^\wedge\right)\mathbf{P}}{\delta\boldsymbol{\xi}} \\ &= \lim_{\delta\boldsymbol{\xi}\to\mathbf{0}} \frac{\begin{bmatrix} \delta\boldsymbol{\phi}^\wedge & \delta\boldsymbol{\rho} \\ \mathbf{0}^T & 0 \end{bmatrix}\begin{bmatrix} \mathbf{R}\mathbf{P} + \mathbf{t} \\ 1 \end{bmatrix}}{\delta\boldsymbol{\xi}} \\ &= \lim_{\delta\boldsymbol{\xi}\to\mathbf{0}} \frac{\begin{bmatrix} \delta\boldsymbol{\phi}^\wedge(\mathbf{R}\mathbf{P} + \mathbf{t}) + \delta\boldsymbol{\rho} \\ \mathbf{0}^T \end{bmatrix}}{[\delta\boldsymbol{\rho}, \delta\boldsymbol{\phi}]^T} \\ &= \begin{bmatrix} \mathbf{I} & -(\mathbf{R}\mathbf{P} + \mathbf{t})^\wedge \\ \mathbf{0}^T & \mathbf{0}^T \end{bmatrix}, \end{aligned} \tag{42}$$

taking out the first 3 dimensions, we have

$$\frac{\partial \mathbf{P}'}{\partial \delta \boldsymbol{\xi}} = \left[\mathbf{I} \; -\mathbf{P}'^{\wedge} \right], \tag{43}$$

where,

$$-\mathbf{P}'^{\wedge} = \begin{bmatrix} 0 & Z' & -Y' \\ -Z' & 0 & X' \\ Y' & -X' & 0 \end{bmatrix}. \tag{44}$$

Merging formula (38) (41) and (42), the final Jacobian matrix can be solved as

$$\mathbf{J} = \frac{\partial \mathbf{e}}{\partial \delta \boldsymbol{\xi}} = -\frac{2}{A^2 + B^2}[J_1, J_2, J_3, J_4, J_5, J_6], \tag{45}$$

among that,

$$\begin{aligned} J_1 &= m\frac{f_x}{Z'} \\ J_2 &= n\frac{f_y}{Z'} \\ J_3 &= -\frac{m f_x X' + n f_y Y'}{Z'^2} \\ J_4 &= -n f_y - \frac{m f_x X' + n f_y Y'}{Z'^2} Y' \\ J_5 &= m f_x + \frac{m f_x X' + n f_y Y'}{Z'^2} X' \\ J_6 &= -m\frac{f_x Y'}{Z'^2} + n\frac{f_y X'}{Z'^2}, \end{aligned} \tag{46}$$

with,

$$\begin{aligned} m &= A^2(\mu^f - \mu) + AB(\nu^f - \nu) \\ n &= AB(\mu^f - \mu) + B^2(\nu^f - \nu) \end{aligned} \tag{47}$$

ACKNOWLEDGMENT

The authors would like to thank Yihan Zhong for his help and support with the dataset collection of UGV.

REFERENCES

[1] A. Broggi et al., "Extensive tests of autonomous driving technologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1403–1415, Sep. 2013.

[2] W. Wen et al., "UrbanLoco: A full sensor suite dataset for mapping and localization in urban scenes," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 2310–2316.

[3] X. Zhou et al., "Swarm of micro flying robots in the wild," *Sci. Robot.*, vol. 7, no. 66, p. 5954, May 2022.

[4] P. K. Enge, "The global positioning system: Signals, measurements, and performance," *Int. J. Wireless Inf. Netw.*, vol. 1, no. 2, pp. 83–105, Apr. 1994.

[5] J. Breßler, P. Reisdorf, M. Obst, and G. Wanielik, "GNSS positioning in non-line-of-sight context—A survey," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 1147–1154.

[6] W. Ding, S. Hou, H. Gao, G. Wan, and S. Song, "LiDAR inertial odometry aided robust LiDAR localization system in changing city scenes," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 4322–4328.

[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.

[8] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," *Proc. SPIE*, vol. 1611, pp. 586–606, Apr. 1992.

[9] T. Qin, Y. Zheng, T. Chen, Y. Chen, and Q. Su, "A light-weight semantic map for visual localization towards autonomous driving," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 11248–11254.

[10] L. Liu, H. Li, and Y. Dai, "Efficient global 2D-3D matching for camera localization in a large-scale 3D map," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2372–2381.

[11] Y. Xu, V. John, S. Mita, H. Tehrani, K. Ishimaru, and S. Nishino, "3D point cloud map based vehicle localization using stereo camera," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 487–492.

[12] K. Yabuuchi, D. R. Wong, T. Ishita, Y. Kitsukawa, and S. Kato, "Visual localization for autonomous driving using pre-built point cloud maps," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2021, pp. 913–919.

[13] Y. Lu, H. Ma, E. Smart, and H. Yu, "Real-time performance-focused localization techniques for autonomous vehicle: A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 6082–6100, Jul. 2022.

[14] M. Brown, D. Windridge, and J.-Y. Guillemaut, "A family of globally optimal branch-and-bound algorithms for 2D–3D correspondence-free registration," *Pattern Recognit.*, vol. 93, pp. 36–54, Sep. 2019.

[15] R. W. Wolcott and R. M. Eustice, "Visual localization within LiDAR maps for automated urban driving," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 176–183.

[16] A. D. Stewart and P. Newman, "LAPS–localisation using appearance of prior structure: 6-DoF monocular camera localisation using prior pointclouds," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 2625–2632.

[17] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—A modern synthesis," in *Proc. Int. Workshop Vis. Algorithms*. Cham, Switzerland: Springer, 1999, pp. 298–372.

[18] P. Biber and W. Strasser, "The normal distributions transform: A new approach to laser scan matching," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Jun. 2003, pp. 2743–2748.

[19] X. Zuo, P. Geneva, Y. Yang, W. Ye, Y. Liu, and G. Huang, "Visual-inertial localization with prior LiDAR map constraints," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3394–3401, Oct. 2019.

[20] Y. Kim, J. Jeong, and A. Kim, "Stereo camera localization in 3D LiDAR maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–9.

[21] T. Caselitz, B. Steder, M. Ruhnke, and W. Burgard, "Monocular camera localization in 3D LiDAR maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 1926–1931.

[22] L. Meng, "The 'here and now' of HD mapping for connected autonomous driving," in *New Thinking in GIScience*. Cham, Switzerland: Springer, 2022, pp. 329–340.

[23] M. Schreiber, C. Knöppel, and U. Franke, "LaneLoc: Lane marking based localization using highly accurate maps," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2013, pp. 449–454.

[24] S. Liang, Y. Zhang, R. Tian, D. Zhu, L. Yang, and Z. Cao, "SemLoc: Accurate and robust visual localization with semantic and structural constraints from prior maps," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 4135–4141.

[25] C. Zhang, H. Zhao, C. Wang, X. Tang, and M. Yang, "Cross-modal monocular localization in prior LiDAR maps utilizing semantic consistency," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 4004–4010.

[26] Y. Zhou, X. Li, S. Li, and X. Wang, "Visual mapping and localization system based on compact instance-level road markings with spatial uncertainty," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10802–10809, Oct. 2022.

[27] D. Cattaneo, M. Vaghi, A. L. Ballardini, S. Fontana, D. G. Sorrenti, and W. Burgard, "CMRNet: Camera to LiDAR-map registration," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 1283–1289.

[28] Y. Jeon and S.-W. Seo, "EFGHNet: A versatile image-to-point cloud registration network for extreme outdoor environment," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 7511–7517, Jul. 2022.

[29] A. Pumarola, A. Vakhitov, A. Agudo, A. Sanfeliu, and F. Moreno-Noguer, "PL-SLAM: Real-time monocular visual SLAM with points and lines," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Singapore, May 2017, pp. 4503–4508.

[30] J. Lu, Z. Fang, Y. Gao, and J. Chen, "Line-based visual odometry using local gradient fitting," *J. Vis. Commun. Image Represent.*, vol. 77, May 2021, Art. no. 103071.

[31] H. Lim, J. Jeon, and H. Myung, "UV-SLAM: Unconstrained line-based SLAM using vanishing points for structural mapping," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1518–1525, Apr. 2022.

[32] Y. He, J. Zhao, Y. Guo, W. He, and K. Yuan, "PL-VIO: Tightly-coupled monocular visual–inertial odometry using point and line features," *Sensors*, vol. 18, no. 4, p. 1159, Apr. 2018.

[33] D. G. Kottas and S. I. Roumeliotis, "Efficient and consistent vision-aided inertial navigation using line observations," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 1540–1547.

[34] H. Yu, W. Zhen, W. Yang, J. Zhang, and S. Scherer, "Monocular camera localization in prior LiDAR maps with 2D-3D line correspondences," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 4588–4594.

[35] 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.

[36] P. Antonante, D. I. Spivak, and L. Carlone, "Monitoring and diagnosability of perception systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 168–175.

[37] T. Walter and P. Enge, "Weighted RAIM for precision approach," in *Proc. I_{ON} GPS*, Sep. 1995, pp. 1995–2004.

[38] J. E. Angus, "RAIM with multiple faults," *Navigation*, vol. 53, no. 4, pp. 249–257, Dec. 2006.

[39] P. H. S. Torr and A. Zisserman, "MLESAC: A new robust estimator with application to estimating image geometry," *Comput. Vis. Image Understand.*, vol. 78, no. 1, pp. 138–156, Apr. 2000.

[40] M. A. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[41] P. Speciale, D. P. Paudel, M. R. Oswald, T. Kroeger, L. V. Gool, and M. Pollefeys, "Consensus maximization with linear matrix inequality constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5048–5056.

[42] T.-J. Chin, Y. H. Kee, A. Eriksson, and F. Neumann, "Guaranteed outlier removal with mixed integer linear programs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5858–5866.

[43] A. Das and S. L. Waslander, "Outlier rejection for visual odometry using parity space methods," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 3613–3618.

[44] C. Legrand, J. Beugin, J. Marais, B. Conrard, E.-M. El-Koursi, and M. Berbineau, "From extended integrity monitoring to the safety evaluation of satellite-based localisation system," *Rel. Eng. Syst. Saf.*, vol. 155, pp. 105–114, Nov. 2016.

[45] G. Duenas Arana, O. Abdul Hafez, M. Joerger, and M. Spenko, "Integrity monitoring for Kalman filter-based localization," *Int. J. Robot. Res.*, vol. 39, no. 13, pp. 1503–1524, Nov. 2020.

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

[47] C. Li and S. L. Waslander, "Visual measurement integrity monitoring for UAV localization," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Sep. 2019, pp. 22–29.

[48] C. Zhu, M. Meurer, and C. Günther, "Integrity of visual navigation—Developments, challenges, and prospects," *NAVIGATION, J. Inst. Navigat.*, vol. 69, no. 2, pp. 1–27, 2022.

[49] C. Zhu, C. Steinmetz, B. Belabbas, and M. Meurer, "Feature error model for integrity of pattern-based visual positioning," in *Proc. ION GNSS+, Int. Tech. Meeting Satell. Division Inst. Navigat.*, Oct. 2019, pp. 2254–2268.

[50] X. Lu, Y. Liu, and K. Li, "Fast 3D line segment detection from unorganized point cloud," 2019, *arXiv:1901.02532*.

[51] X. Gao, T. Zhang, Y. Liu, and Q. Yan, *14 Lectures on Visual SLAM: From Theory to Practice*. Beijing, China: House of Electronics Industry, 2017.

[52] J. H. Lee, S. Lee, G. Zhang, J. Lim, W. K. Chung, and I. H. Suh, "Outdoor place recognition in urban environments using straight lines," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 5550–5557.

[53] V. S. Varadarajan, *Lie Groups, Lie Algebras, and Their Representations*, vol. 102. Cham, Switzerland: Springer, 2013.

[54] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[55] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 1280–1286.

[56] S. Agarwal and K. Mierle, "Ceres solver: Tutorial & reference," *Google Inc*, vol. 2, no. 72, p. 8, 2012.

[57] J. Nocedal and S. J. Wright, *Numerical Optimization*. Cham, Switzerland: Springer, 1999.

[58] F. Gustafsson, "Statistical signal processing approaches to fault detection," *Annu. Rev. Control*, vol. 31, no. 1, pp. 41–54, 2007.

[59] F. Pukelsheim, "The three sigma rule," *Amer. Statistician*, vol. 48, no. 2, pp. 88–91, May 1994.

[60] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vilamoura-Algarve, Portugal, Oct. 2012, pp. 573–580.

[61] M. Quigley et al., "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, 2009, vol. 3, no. 3, p. 5.

[62] M. Burri et al., "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, Sep. 2016.

[63] K. Wu, A. Ahmed, G. Georgiou, and S. Roumeliotis, "A square root inverse filter for efficient vision-aided inertial navigation on mobile devices," in *Proc. Robot., Sci. Syst.*, vol. 2, pp. 2–10, Jul. 2015.

[64] S. Yi, S. Worrall, and E. Nebot, "Metrics for the evaluation of localisation robustness," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 1247–1253.

[65] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Stockholm, Sweden, May 2016, pp. 809–816.

[66] E. W. Cheney and D. R. Kincaid, *Numerical Mathematics and Computing*. Boston, MA, USA: Cengage Learning, 2012.

[67] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn.*, vol. 78, 2017, pp. 1–16.

[68] 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.

[69] L. Zhang and T. Zhang, "A fast calibration method for dynamic lever-arm parameters for IMUs based on the backtracking scheme," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 9, pp. 6946–6956, Sep. 2020.

[70] N. Zhu, J. Marais, D. Bétaille, and M. Berbineau, "GNSS position integrity in urban environments: A review of literature," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 9, pp. 2762–2778, Sep. 2018.

Xi Zheng received the B.Eng. and M.Eng. degrees in astronautics from Northwestern Polytechnical University, Xi'an, China, in 2015 and 2018, respectively. She is currently pursuing the Ph.D. degree with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University. Her research interests include visual SLAM and safety quantifiable localization.

Weisong Wen (Member, IEEE) was born in Ganzhou, Jiangxi, China. He received the Ph.D. degree in mechanical engineering, from The Hong Kong Polytechnic University. He was a Visiting Student Researcher with the University of California at Berkeley (UCB) in 2018. He is currently a Research Assistant Professor with the Department of Aeronautical and Aviation Engineering. His research interests include multi-sensor integrated localization for autonomous vehicles, SLAM, and GNSS positioning in urban canyons.

Li-Ta Hsu received the B.S. and Ph.D. degrees in aeronautics and astronautics from National Cheng Kung University, Taiwan, in 2007 and 2013, respectively. He is currently an Associate Professor with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, before he was a Post-Doctoral Researcher with the Institute of Industrial Science, The University of Tokyo, Japan. In 2012, he was a Visiting Scholar with University College London, U.K. His research interests include GNSS positioning in challenging environments and localization for pedestrian, autonomous driving vehicle, and autonomous aerial vehicle.