### **ORIGINAL ARTICLE**

## WILEY 🛞 ION

### Integrity monitoring of Graph-SLAM using GPS and fish-eye camera

### Sriramya Bhamidipati<sup>1</sup> Grace Xingxin Gao<sup>2</sup>

<sup>1</sup> University of Illinois at Urbana-Champaign <sup>2</sup> Stanford University

Correspondence Grace Xingxin Gao, Stanford University. Email: gracegao@stanford.edu

**Funding information** National Science Foundation, Grant/Award Number: 2006162

### Abstract

We propose a Simultaneous Localization and Mapping (SLAM)-based Integrity Monitoring (IM) algorithm using GPS and fish-eye camera to compute the protection levels while accounting for multiple faults in GPS and vision. We perform graph optimization using GPS pseudoranges, pixel intensities, vehicle dynamics, and satellite ephemeris to simultaneously localize the vehicle, GPS satellites, and key image pixels in the world frame. We estimate the fault mode vector by analyzing the temporal correlation across pseudorange residuals and spatial correlation across pixel intensity residuals. To isolate the vision faults, we develop a superpixel-based piecewise random sample consensus. For the estimated fault mode, we compute the protection levels by performing worst-case failure slope analysis on the batch realization of linearized Graph-SLAM formulation. We perform real-world experiments in an alleyway in Stanford, California and a semiurban area in Champaign, Illinois. We demonstrate higher localization accuracy and tighter protection levels as compared to GPS-only SLAM-based IM.

#### INTRODUCTION 1

Integrity serves as a key performance measure in assessing the correctness of the estimated position (Ochieng et al., 2003). In urban environments, GPS systems receive fewer measurements due to degraded satellite visibility. They also suffer from received signal faults caused by multipath and satellite faults caused by anomalies in the broadcast navigation message. To address the above-mentioned challenges, one possible solution is to incorporate additional redundancy through the sensor fusion of GPS and vision. Vision sensor performs well in urban areas due to the feature-rich surroundings (Hol, 2011). Sensor fusion (Krishnaswamy, 2008) integrates measurements from multiple sensors to improve the accuracy of the vehicle and provide robust performance. Individual sensors, such as GPS and camera, have inherent limitations in operability that are reliably corrected by combining these complementary sensors in a sensor fusion framework. In particular, occlusion and illumination variations in multiple pixel intensities induce data association errors across images, thereby termed as vision faults (Miro, Zhou, & Dissanayake, 2006). Therefore, there is a need for the development of multi-sensor Integrity Monitoring (IM) techniques that account for multiple faults in both GPS and vision.

#### | Related work 1.1

The concept of Receiver Autonomous Integrity Monitoring (RAIM) (Hewitson & Wang, 2006) has been initially developed in the aviation sector for executing safety-critical applications. However, extending the concept of integrity from aviation to urban transportation is not entirely straightforward (Zhu, Marais, Bétaille, & Berbineau, 2018). This is because, as mentioned earlier, the sensors operating in urban areas face additional challenges (Joerger & Spenko, 2017) due to static infrastructure, such as buildings and thick foliage, dynamic obstacles, such as traffic -WILEY 🛞 ION

and pedestrians, and environmental conditions, such as shadows, sunlight, and weather. Research (Joerger & Spenko, 2017; Pullen, Walter, & Enge, 2011a) describes the initial parallels between the established integrity requirements in aviation and the early safety standards followed in the transportation sector. Based on this, the standardized integrity requirements in aviation are found to be too conservative for urban applications.

A collection of earlier works (Binjammaz, Al-Bayatti, & Al-Hargan, 2013) adapted the traditional snapshot RAIM techniques, which were originally developed for aviation industry, to urban context. Salós, Martineau, Macabiau, Bonhoure, and Kubrak (2013) studied the RAIM availability in the context of electronic toll collection by designing a modified weighted Least-Squares (LS) residual-based RAIM that considers a constant mis-detection probability. Another work (Zhang, Li, Cui, & Liu, 2017a) proposed an adaptive RAIM by characterizing the position errors and measurement residuals as a chi-squared distribution. However, the limitation of this approach is that all pseudorange measurements are considered to have the same error variance, which is not a realistic assumption in an urban scenario.

Prior literature on GPS-based IM for urban navigation addresses the insufficient availability of GPS measurements by incorporating multiple GNSS constellations (Borio & Gioia, 2016) and characterizing the measurement errors associated with the Non-Line-Of-Sight (NLOS) satellite signals (Groves & Jiang, 2013). However, due to the dense building infrastructure, the geometry of satellites is skewed, which still degrades the confidence in the estimated integrity bounds. In this context, Zhang, Li, Cui, and Liu (2017b) proposed an adapted-RAIM approach that accounts for the satellite geometry in deciding whether to include or exclude a potential multipath-affected measurement. Santa, Úbeda, Toledo, and Skarmeta (2006) and Pullen, Walter, and Enge (2011b) analyzed the existing infrastructure of Ground-Based Augmentation System (GBAS) and Satellite-Based Augmentation System (SBAS) for use by non-aviation users. However, these augmentation systems have selected coverage and comply with the aviation integrity standards, which limits their applicability in the urban transportation sector.

In addition to the above-mentioned snapshot algorithms, a rich literature exists on utilizing sequential approaches for RAIM analysis in urban areas. In Le Marchand, Bonnifait, Ibañez-Guzmán, Peyret, and Bétaille (2008), the authors developed a Kalman Filter (KF)-based Fault Detection and Exclusion (FDE) algorithm where the measurement residuals were independently compared against a threshold to be flagged as either faulty or reliable. Tominaga and Kubo (2019) proposed a KF approach where the measurement noise covariance matrix is adaptively estimated, and thereafter, the position error bounds are estimated as being proportional to the state uncertainty. Tran and Lo Presti (2019) designed a KF-based Solution Separation (SS) RAIM methodology for urban navigation that combines the smoothness and higher accuracy of KF with the flexible framework of SS RAIM. Zhu et al. (2018) and Margaria and Falletti (2016) summarize the existing state-of-the-art techniques for IM in urban contexts and also point out several open research issues in this field.

Some existing works rely on external information sources to perform urban GPS-based IM. In Velaga, Ouddus, Bristow, and Zheng (2012), the authors developed a sequential map-aided IM technique that checks for outliers in position and Geographic Information System (GIS) using traditional RAIM (Walter, Blanch, Enge, Pervan, & Gratton, 2008) and weight-based topological map-matching process, respectively. Another paper (Binjammaz et al., 2013) developed three phases of integrity checks that include assessing the position quality via traditional RAIM, speed integrity via GPS Doppler, and mapmatching accuracy via fuzzy inference system. However, these approaches have practical limitations because the offline map database is not always available, and its accuracy cannot be guaranteed due to the dynamic changes in the urban surroundings. Another line of prior work (El-Mowafy & Kubo, 2017; Li, Bonnifait, Ibanez-Guzman, & Zinoune, 2017; Toledo-Moreo, Bétaille, & Peyret, 2009) utilizes the odometry information obtained from other sensors, such as inertial measurement units, speedometer, and camera, to perform GPS integrity analysis. But the drawbacks of these approaches are that they do not account for the faults in the external sensors and the simultaneous occurrence of multiple faults.

Nowadays, research on multi-sensor navigation in urban areas using GPS and vision has received great attention (Bonin-Font, Ortiz, & Oliver, 2008; Won et al., 2014). Dusha and Mejias (2012) and Chen, Hu, Zhang, Shi, and Li (2018) proposed a visual-GPS methodology, wherein the GPS measurements are utilized to estimate the scale ambiguity in visual odometry while the camera images aid in attitude estimation. Won, Yun, Lee, and Sung (2012) propose an integration of GPS with a monocular camera to improve the navigation performance. In Shepard and Humphreys (2014), the authors proposed a Simultaneous Localization and Mapping (SLAM) framework to estimate highprecision position and attitude using carrier-phase GPS measurements and feature-based visual odometry (Klein & Murray, 2007). While the existing literature addresses the notion of position accuracy, none of the state-of-theart techniques provide any safety guarantees for the multisensor estimated navigation solution.

We leverage the generalized and flexible platform developed in our prior work (Bhamidipati & Gao, 2018), which



(c) Multipath effects in urban areas

**FIGURE 1** Characteristics of Graph-SLAM using GPS-only measurements; (a) Graph-SLAM framework utilizes GPS pseudoranges, vehicle motion inputs, and satellite ephemeris to localize both vehicle and GPS satellites; advantage of localizing GPS satellites during faults, namely, (b) broadcast anomalies and (c) multipath effects [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

is Simultaneous Localization and Mapping (SLAM)-based Fault Detection and Isolation (FDI), as the basis for assessing the multi-sensor integrity. This paper is based on our recent ION GNSS+ 2019 conference paper (Bhamidipati & Gao, 2019b). Another extension of the SLAM-based FDI platform, described in our prior work (Bhamidipati & Gao, 2019a), assesses the integrity of cooperative localization using a network of receivers. SLAM (Cadena et al., 2016), a well-known technique in robotics, utilizes sensor measurements to estimate the landmarks in a threedimensional (3D) map while simultaneously localizing the robot within it. Analogous to this, our prior work (Bhamidipati & Gao, 2018) on SLAM-based FDI combines the sequential data of GPS pseudoranges, vehicle motion model, and satellite broadcast ephemeris in a graph framework to simultaneously localize the robot, which is the GPS receiver, landmarks in the map, which are the GPS satellites. A key feature of this platform is that it utilizes graph optimization techniques (Latif, Cadena, & Neira, 2014) and, therefore, does not require any prior assumption regarding the distribution of states. As seen in Figure 1(a), given that we localize the landmarks as well, our SLAMbased FDI does not require any prior information regarding the surrounding 3D maps.

Utilizing GPS satellites as landmarks is especially fruitful during faults, namely broadcast anomalies and multipath effects. As seen in Figure 1(b), the location of GPS satellites estimated via satellite ephemeris is incorrect during broadcast anomalies and during multipath; as seen in Figure 1(c), the received GPS pseudoranges have additional bias. In both scenarios, the conventional residual-based FDI module (Joerger, Chan, & Pervan, 2014) flags the measurements as faulty and excludes them from navigation solution estimation. However, excluding the faulty ones lower the total number of measurements available for navigation. In contrast, our SLAM-based FDI utilizes the sequential data of vehicle motion inputs and GPS pseudoranges to estimate the position of GPS satellites', thereby mitigating the effect of broadcast anomalies. Also, during multipath, instead of estimating the GPS satellite location that agrees with the broadcast ephemeris, our framework intelligently localizes the landmark to a virtual location that complies with the received GPS pseudorange. The concept of estimating virtual satellite location, which corresponds to the reflection of an actual GPS satellite across a building surface, has been demonstrated in existing literature (Ng & Gao, 2016). Therefore, unlike traditional GPS algorithms, the flexibility provided -Wiley 🛞 ION -

by our SLAM-based FDI in localizing the GPS satellites improves the number of reliable measurements available for navigation. A real-world experimental validation that demonstrates an improved vehicle positioning accuracy by utilizing our SLAM-based FDI framework is shown later in Figures 11(a) and 11(b).

### **1.2** | Key contributions

We propose the SLAM-based IM algorithm using GPS and fish-eye camera to compute the error bounds, termed as *protection levels*, of the Graph-SLAM estimated navigation solution. In the graph framework, we consider *global landmarks* as the GPS satellites and additional *local landmarks* as the key image pixels in the world frame. We constrain the graph via GPS pseudoranges, vision pixel intensities, vehicle dynamics, and satellite ephemeris.

To obtain vision pixel intensities, we opt for a fish-eye camera mounted on a vehicle and point it upwards for the following reasons: firstly, given its wide ( $\geq 180^\circ$ ) Field-Of-View (FOV), the image pixels are spatially spread out in different directions with respect to the vehicle, thereby, compensating for the restricted spatial geometry of the limited global landmarks, i.e., GPS satellites. Secondly, given that the camera is pointing upwards, the unstructured skyline of the buildings aids in resolving the attitude of the vehicle. Thirdly, the fish-eye image captures the open-sky section with respect to the vehicle that is used to distinguish the Line-Of-Sight (LOS) GPS satellites from that of the NLOS ones (Shytermeja, Garcia-Pena, & Julien, 2014).

The main contributions of this paper are listed as follows:

- 1. We design a Graph-SLAM framework using GPS and fish-eye camera that simultaneously estimates the vehicle, GPS satellites, and key image pixels and also accounts for multiple faults in both GPS and vision.
- 2. To detect and isolate GPS faults, we evaluate the deviation in each pseudorange residual against its corresponding empirical Cumulative Distribution Function (CDF), which represents the measurement errors in non-faulty conditions.
- We develop a superpixel based piecewise Random Sample Consensus (RANSAC) (Conte & Doherty, 2009; Li & Chen, 2015) technique that performs spatial voting across pixel intensity residuals to detect and isolate vision faults.
- 4. Utilizing the fault mode vector estimated from multiple FDI, we derive the protection levels by applying worstcase failure mode slope analysis (Joerger et al., 2014; Salós et al., 2013) to the batch realization of the linearized Graph-SLAM formulation.

5. We have conducted two real-world experiments using a ground vehicle, one in an alleyway in Stanford, California and the other in a semi-urban area in Champaign, Illinois. We have demonstrated the successful detection of multiple measurement faults in GPS and fish-eye camera. We have also validated tighter protection levels and lower localization errors achieved via the proposed algorithm as compared to SLAM-based IM that utilizes only GPS measurements.

The rest of the paper is organized as follows: Section II describes our SLAM-based IM that utilizes GPS and fish-eye camera; Section III experimentally validates the detection of multiple faults in both GPS and vision, lower localization accuracy, and tighter protection levels via the proposed algorithm as compared to GPS-only SLAM-based IM; Section IV concludes the paper.

# 2 | SLAM-BASED IM USING GPS AND FISH-EYE CAMERA

In this work, we focus on the measurement faults related to GPS and vision that frequently occur in urban areas. While the formulation of the proposed algorithm is capable of addressing other faults, for simplicity, we consider no measurement faults associated with the receiver motion model and satellite broadcast ephemeris. For reference, our prior work (Bhamidipati & Gao, 2018) describes details on utilizing SLAM-based FDI to account for broadcast anomalies, described earlier in Figure 1(b).

In Figure 2, we outline the architecture of the proposed multi-sensor SLAM-based IM using GPS and fish-eye camera, which is summarized as follows:

- 1. During initialization, we create a 3D world map via the estimated Position, Velocity, and Time (PVT) of the receiver and satellites computed from an existing GPS algorithm (Lashley, Bevly, & Hung, 2010). We also compute a series of camera images to compute the intrinsic and extrinsic camera parameters. Further details regarding the initialization are given in Section 3.2.
- 2. We pre-process the raw image obtained from the fisheye camera using our hybrid sky detection algorithm to distinguish the sky pixels from the non-sky pixels. The detected sky pixels are used to distinguish the LOS and NLOS satellites. This information is thereafter used in formulating the GPS measurement covariance.
- 3. We consider the non-sky pixels along with GPS pseudoranges and carrier-to-noise density  $(C/N_0)$  values, receiver motion model, and satellite orbital model as inputs to our algorithm. We combine the measurements in an extended graph optimization module to estimate

FIGURE 2 Architecture of our SLAM-based IM algorithm using GPS and fish-eye camera [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]



the overall state vector, which consists of the state vector of the vehicle, GPS satellites, and key image pixels using an M-estimator based Levenberg Marquardt algorithm (Lourakis et al., 2005; Shevlyakov, Morgenthaler, & Shurygin, 2008).

- 4. We independently analyze the measurement residuals against an empirical Normal distribution to detect and isolate GPS faults. We develop a superpixel-based piecewise RANSAC to perform spatial voting for the detection and isolation of vision faults. Based on the estimated fault status of the measurements, we formulate the measurement fault mode, which has binary entries, such that 0 indicates non-faulty and 1 represents faulty.
- 5. Utilizing the estimated fault mode and overall state vector, we formulate the batch realization of linearized Graph-SLAM formulation and, subsequently, compute the protection levels using worst-case failure mode slope analysis (Joerger et al., 2014; Salós et al., 2013).

The proposed SLAM-based IM algorithm using GPS and fish-eye camera consists of three main modules, namely measurement pre-processing, extended graph optimization, and IM for Graph-SLAM. We describe the details as follows.

#### Pre-processing the measurements 2.1

We consider the following measurements as inputs to our SLAM-based IM algorithm: GPS pseudoranges and  $C/N_0$ values from the GPS receiver, pixel intensities from the fish-eye image, motion input obtained from the vehicle motion model, and satellite ephemeris decoded from the navigation message.

### 2.1.1 | Vision module

We pre-process the raw image obtained from the fisheye camera using our hybrid sky detection algorithm to distinguish the sky-pixels from the non-sky pixels. The pipeline of our hybrid sky detection is seen in Figure 3(a). Our hybrid sky detection takes into account not only the pixel intensities but also prior knowledge regarding the spatial location of the sky pixels.

We convert the raw image to gray scale and then perform the median blur operation. The median blur (Wang, Li, Yang, & Gong, 2010) is a nonlinear filter that reduces the noise in the image while keeping the edges relatively sharp. Next, we compute the gradient by combining the magnitude obtained via two Sobel operations (Gao, Zhang, Yang, & Liu, 2010), one executed in horizontal and the other in vertical direction. An example of the image obtained from the fish-eye camera operating in urban areas and pointing upwards is seen in Figure 3(b).

We observe that the probability of sky is highest close to the center and exponentially decreases outwards (Haque, Murshed, & Paul, 2008). Therefore, the corresponding location-based sky probability, denoted by  $p_{loc}$  is given by

$$p_{loc}(\boldsymbol{u}) = \exp\left(-\frac{2\|\boldsymbol{u} - \boldsymbol{c}_{loc}\|}{|\Pi|}\right), \quad (1)$$

where  $\boldsymbol{u}$  is the two-dimensional (2D) image coordinates, such that  $u \in \Pi$ ,  $\Pi$  represents the pre-defined domain of the image coordinates.  $|\cdot|$  denotes the cardinality of the image domain, and  $\|\cdot\|$  denotes the 2-norm residual.  $c_{loc} \subset$  $u \in \Pi$  denotes the pre-defined center coordinates in the 2D image frame.

Combining the location probability with Otsu's method of intensity thresholding (Moghaddam, Cheriet, & Adotsu, 2012), we compute the hybrid optimal border, seen in Figure 3(c), that separates the sky region, represented by subscript sky, from that of the infrastructure, denoted by subscript *inf*. We minimize the variance of sky and



**FIGURE 3** An example showing the pipeline of pre-processing the fish-eye image in the vision module [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

infrastructure to estimate the Otsu's intensity threshold  $I_{otsu}$  as

$$I_{otsu} = \underset{k \in I}{\operatorname{argmin}} \left( \omega_{sky}(k) \sigma_{sky}^{2}(k) + \omega_{inf}(k) \sigma_{inf}^{2}(k) \right)$$

from (Moghaddam, Cheriet, & Adotsu 2012), (2)

where

588

- *I* denotes the intensity vector that stacks all the pixel intensities in the fish-eye image, such that  $I = \{I(\boldsymbol{u}) \mid \boldsymbol{u} \in \Pi\}$ , where  $I(\boldsymbol{u}) : \Pi \to \mathbb{R}$  denotes the intensity of any 2D pixel coordinates  $\boldsymbol{u}$ ;
- $\omega_{sky}(k)$  and  $\omega_{inf}(k)$  denote the weights associated with the sky and building infrastructure, respectively, such that  $\omega_{sky}(k) = \frac{1}{|\Pi|} \sum_{\boldsymbol{u} \in |\Pi|} \mathbb{I}\{I(\boldsymbol{u}) < k\}$  and  $\omega_{inf}(k) = \frac{1}{|\Pi|} \sum_{\boldsymbol{u} \in |\Pi|} \mathbb{I}\{I(\boldsymbol{u}) > k\};$
- $\sigma_{sky}^2(k)$  and  $\sigma_{inf}^2(k)$  denote the variance of the pixel intensities associated with the sky and building infrastructure.

Utilizing Equations (1) and (2), we compute the hybrid sky probability, denoted by  $p_{sky}$  at any 2D image coordinate  $\boldsymbol{u}, \boldsymbol{u} \in \Pi$  as

$$p_{sky}(\boldsymbol{u}) = \exp\left(\frac{-\left|I(\boldsymbol{u}) - I_{otsu}\right|}{\left|I_{max} - I_{min}\right|}\right) p_{loc}(\boldsymbol{u}), \qquad (3)$$

where  $I_{max}$  and  $I_{min}$  are the maximum and minimum intensity values in the fish-eye image, such that  $I_{max} = \max_{\boldsymbol{u} \in \Pi} I(\boldsymbol{u})$  and  $I_{min} = \min_{\boldsymbol{u} \in \Pi} I(\boldsymbol{u})$ , respectively. Considering  $\eta$  as the pre-defined sky threshold, if  $p_{sky}(\boldsymbol{u}) > \eta$ , it is categorized as sky pixel and non-sky pixel otherwise. The sky-enclosed area in the fish-eye image is seen in Figure 3(d). The value of  $\eta$  is set during initialization and explained later in Section 3.2.

Using the non-sky detected pixels, we describe the vision measurement model of an omni-directional fish-eye camera in Eq (4), which is formulated via direct image alignment (Caruso, Engel, & Cremers, 2015) and semi-dense inverse depth map estimation (Engel, Schöps, & Cremers, 2014). Instead of utilizing standard algorithms (Klein & Murray, 2007; Mur-Artal, Montiel, & Tardos, 2015) that rely on feature detection and extraction, we perform direct image alignment that directly processes the pixel intensities of keyframe and current image frame to estimate the relative change in position and orientation of the vehicle. The selected pixels of the key frame, termed as key pixels, refer to the ones that exhibit a high intensity gradient. Details regarding the keyframe selection are explained in the existing literature (Engel et al., 2014).

$$I_{kf}(\boldsymbol{u}) = I_t(\pi(w(\Delta \boldsymbol{\mu}_t, \boldsymbol{u}))) + \eta_{vis}(\boldsymbol{u})$$
  
from (Engel et al., 2014), (4)

with  $\eta_{vis}(\boldsymbol{u})$  is pixel noise and from Caruso et al. (2015),

$$w(\Delta \boldsymbol{\mu}, \boldsymbol{u}) = \mathbf{R}(\Delta \boldsymbol{\mu}) \pi^{-1} (\boldsymbol{u}, d_{kf}(\boldsymbol{u})) + \mathbf{t}(\Delta \boldsymbol{\mu}),$$
$$\pi(\mathbf{p}) = \begin{bmatrix} f_x \frac{p_x}{p_z + \|\mathbf{p}\|\xi} \\ f_y \frac{p_y}{p_z + \|\mathbf{p}\|\xi} \end{bmatrix} + \begin{bmatrix} c_x \\ c_y \end{bmatrix},$$

$$\pi^{-1}(\boldsymbol{u}, d) = \frac{1}{d} \left( \frac{\xi + \sqrt{1 + (1 - \xi)^2 (\hat{u}^2 + \hat{v}^2)}}{\hat{u}^2 + \hat{v}^2 + 1} \begin{bmatrix} \hat{u}^2 \\ \hat{v}^2 \\ 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \xi \end{bmatrix} \right),$$

where the subscript kf refers to keyframe,

- $I_{kf}(\boldsymbol{u})$  :  $\Pi_{kf} \to \mathbb{R}$  denotes the intensity of any 2D pixel coordinates  $\boldsymbol{u}$  in the reference keyframe, and  $\Pi_{kf} \subset \mathbb{R}^2$  denotes the image domain of keyframe;
- $I_t(\boldsymbol{u})$ :  $\Pi_{ns} \to \mathbb{R}$  denotes the intensity of any 2D pixel coordinates  $\boldsymbol{u}$  in the current frame, and  $\Pi_{ns} \subseteq \Pi$  denotes the image domain consisting of non-sky pixels;
- $\pi$  :  $\mathbb{R}^3 \to \Pi_{ns}$  denotes the mapping function of any 3D world position coordinates  $\mathbf{p} = [p_x, p_y, p_z]$  to a 2D pixel in the image frame;
- $w(\Delta \mu, \boldsymbol{u})$  denotes the 3D warp function that unprojects the pixel coordinates  $\boldsymbol{u}$  and transforms it by a relative state vector  $\Delta \mu$ . The relative state vector  $\Delta \mu$  indicates the difference between the current vehicle pose, denoted by  $\mu_t = [\mathbf{x}, \boldsymbol{\psi}]_t$  with respect to that of the keyframe, denoted by  $\mu_{kf}$ , such that  $\Delta \mu = \mu_t - \mu_{kf}$ ; here,  $\mathbf{x} = [x, y, z]$  denotes the 3D vehicle position and  $\boldsymbol{\psi}$  denotes the 3D orientation with respect to the world frame;  $\mathbf{R} \in$ SO(3) and  $\mathbf{t} \in \mathbb{R}^3$  denotes the rotation matrix and translation vector of  $\Delta \mu$ , respectively;
- $\pi^{-1}$ :  $\Pi_{ns} \times \mathbb{R}^+ \to \mathbb{R}^3$  denotes the inverse mapping of 2D pixel coordinates to 3D world coordinates via an inverse distance represented by *d*. Here,  $\hat{u} = (u - c_x)/f_x$  and  $\hat{v} = (v - c_y)/f_y$  denotes the transformed 2D pixel coordinates. The intrinsic camera parameters are estimated during initialization and are denoted by  $f_x$ ,  $f_y$ ,  $c_x$ ,  $c_y$  and  $\xi$ ;
- $d_{kf}(\boldsymbol{u})$  denotes the inverse distance of the pixel coordinates in the keyframe.

Direct image alignment accounts for the similarity in urban building infrastructure by preserving the spatial context of the image. We compute a semi-dense map that estimates the inverse-depth of key pixels in the keyframe by executing a short-temporal baseline matching across images. This vision measurement model is utilized later in our extended graph optimization to formulate the corresponding vision odometry-based component of the cost function.

### 2.1.2 | GPS module

In the GPS module, considering N visible satellites, we describe the GPS measurement model as

$$\rho^{k} = \|\mathbf{y}^{k} - \mathbf{x}\| + (c\delta t - c\delta t^{k}) + \eta^{k}, \qquad (5)$$

where **x** and **y**<sup>*k*</sup> denotes the 3D position of the vehicle and  $k^{th}$  satellite, respectively;  $c\delta t$  and  $c\delta t^k$  represents the receiver clock bias and  $k^{th}$  satellite clock bias corrections, respectively;  $\eta^k$  represents the measurement noise related to  $k^{th}$  satellite.

We also formulate the measurement covariance of  $k^{th}$  satellite via the measured  $C/N_0$  values and the sky area detected via Equation. (3) in the vision pre-processing module. Note that the classification of the satellite as either LOS or NLOS depends on the unknown state vector of the vehicle and  $k^{th}$  satellite. Therefore, the measurement covariance of  $k^{th}$  satellite is given by

$$\left(\sigma^{k}\left(\boldsymbol{x}_{t},\boldsymbol{y}_{t}^{k}\right)\right)^{2} = \sqrt{b^{k} + a^{k} \frac{1}{(C/N_{0})^{k}}}$$
  
from (Shytermeja et al., 2014), (6)

where

- $\mathbf{x}_t$  denotes the vehicle state vector at  $t^{th}$  time instant comprising of 3D position, 3D velocity, clock bias, clock drift, and 3D attitude, respectively, such that  $\mathbf{x}_t = [\mathbf{x}, c\delta t, \dot{\mathbf{x}}, c\dot{\delta}t, \boldsymbol{\psi}]_t$ ;
- $\mathbf{y}_t^i$  denotes the state vector of  $i^{th}$  satellite comprising of its 3D position, 3D velocity, clock bias, and clock drift corrections, such that  $\mathbf{y}_t^k = [\mathbf{y}^k, c\delta t^i, \dot{\mathbf{y}}^k, c\delta i^k]_t$ ,  $i \in \{1, ..., N\}$ ;
- $b^k$  and  $a^k$  are the vision coefficients, such that  $b^k = \frac{b_{LOS}}{p_{sky}(\pi(\mathbf{y}^k))}$  and  $a^k = \frac{a_{LOS}}{p_{sky}(\pi(\mathbf{y}^k))}$  when  $p_{sky}(\pi(\mathbf{y}^k)) > \eta$  and  $b^k = \frac{b_{NLOS}}{p_{sky}(\pi(\mathbf{y}^k))}$  and  $a^k = \frac{a_{NLOS}}{p_{sky}(\pi(\mathbf{y}^k))}$  otherwise;  $\eta$  is the pre-defined threshold explained in Equation (3);  $b_{LOS}$ ,  $b_{NLOS}$ ,  $a_{LOS}$  and  $a_{NLOS}$  are constant predetermined coefficients, which are set during initialization and explained later in Section 3.2, and  $\pi(\mathbf{y}^k)$  denotes the projection of the state vector of  $k^{th}$  satellite in the image frame.

### 2.2 | Extended graph optimization

In our extended graph optimization module, our cost function consists of four error terms, namely GPS pseudoranges, non-sky pixel intensities, receiver motion model, and satellite orbital model, as follows:

$$e_{t} \left(\boldsymbol{\theta}_{t}\right) = \sum_{k=1}^{N} \Lambda\left(\left(\left(\bar{r}_{t}^{k}+1\right)\sigma_{t}^{k}\right)^{-1}\left|\boldsymbol{\rho}_{t}^{k}-h\left(\mathbf{x}_{t},\mathbf{y}_{t}^{k}\right)\right|\right) + \sum_{k=1}^{N} \Lambda\left(\left(\left(\overline{\boldsymbol{\Omega}}_{t}^{k}\right)^{-1}\left\|\boldsymbol{y}_{t}^{i}-f\left(\mathbf{u}_{\mathbf{y},t}^{k},\bar{\mathbf{y}}_{t-1}^{k}\right)\right\|\right)$$

$$WILEY \bigoplus ION + \Lambda \left( \left( \bar{\boldsymbol{\chi}}_{t} + \overline{\boldsymbol{\Sigma}}_{t} \right)^{-1} \| \boldsymbol{x}_{t} - g \left( \boldsymbol{u}_{\boldsymbol{x},t}, \bar{\boldsymbol{x}}_{t-1} \right) \| \right) + \sum_{\boldsymbol{u} \in \Pi_{ns}} \Lambda \left( \left( \left( \bar{\boldsymbol{s}}_{t-1} \left( \boldsymbol{u} \right) + 1 \right) \omega_{t} \left( \boldsymbol{u} \right) \right)^{-1} | \boldsymbol{I}_{kf} \left( \boldsymbol{u} \right) - \boldsymbol{I}_{t} \left( \pi \left( w \left( \Delta \boldsymbol{\mu}_{t}, \boldsymbol{u} \right) \right) \right) \right),$$
(7)

where

- $\theta_t$  denotes the overall state vector comprising of the state vector of the vehicle, GPS satellites, and key pixels in the world frame, given by  $\theta_t = [\mathbf{x}_t, \mathbf{y}_t^1, \dots, \mathbf{y}_t^N, \mathbf{p}_t^j \;\forall j \in \{1, \dots, |\Pi|_{kf}\}$  and is unknown to be estimated via graph optimization;
- A denotes the M-estimator function of weighted residuals, where M-estimator (Shepard & Humphreys, 2014) represents a robust regression method that accounts for data containing outliers, extreme observations, or that does not follow a normal distribution. Here,  $\Lambda(\cdot)$ is a symmetric, positive-definite function with a unique minimum at zero. For instance, the conventional leastsquares is a special case of M-estimator, where  $\Lambda(\Delta z) =$  $(\Delta z)^2$ . The choice of M-estimator for this work is given in Section 3.2;
- $\bar{r}_{t-1}^k$  denotes the estimated fault status of the received GPS pseudorange from  $k^{th}$  satellite at the past time instant; similarly,  $\bar{s}_{t-1}(\boldsymbol{u})$  denotes the estimated vision fault status of any 2D pixel  $\boldsymbol{u} \in \Pi_{kf}$  at the previous time instant; the GPS and vision fault status are estimated later in Section 2.3.1;
- *h* denotes the GPS measurement model obtained from Equation (5), such that  $h(\mathbf{x}_t, \mathbf{y}_t^k) = \|\mathbf{y}^k - \mathbf{x}\| + (c\delta t - c\delta t^k)$ ; *g* denotes the motion model of the receiver and *f* denotes the satellite orbital model;  $\bar{\mathbf{x}}_{t-1}$  and  $\bar{\mathbf{y}}_{t-1}^k$  denote the estimated state vector of the vehicle and  $k^{th}$  satellite, respectively, at the previous time instant;  $\mathbf{u}_{\mathbf{x},t}$  and  $\mathbf{u}_{\mathbf{y},t}^k$  denote the motion control inputs of the vehicle and  $k^{th}$  satellite, respectively; later in Section 3.2, we describe the choice of vehicle motion model and satellite orbital model used to validate our algorithm;
- $\bar{\Sigma}_t$  and  $\bar{\Omega}_t^k$  denote the estimated covariance matrices of the vehicle state vector and  $k^{th}$  satellite state vector. For simplicity, we estimate the state covariances via a statistical process known as propagation of uncertainty (Engel et al., 2014);
- $\bar{\boldsymbol{\chi}}_t$  denotes the integrity vector comprising of protection levels associated with each term in the vehicle state vector  $\boldsymbol{x}_t$ . Details regarding estimating the covariances and protection levels are explained later in Section 2.3;
- $\sigma_t^k$  denotes the measurement covariance of the  $k^{th}$  satellite and is estimated from Equation (6); similarly,  $\omega_t(\boldsymbol{u})$  denotes the covariance associated with the intensity of

non-sky pixel **u** and is estimated based on existing literature (Eckenhoff, Geneva, & Huang, 2019).

The first three terms in the cost function  $\mathbf{e}_t$  given in Equation (7) correspond to the residuals associated with the GPS pseudoranges, satellite ephemeris, and vehicle state vector. These terms are equivalent to the terms considered in our prior work on GPS-only SLAM-based FDI (Bhamidipati & Gao, 2018). The last term represents the summation of intensity residuals across non-sky pixels based on the vision measurement model explained in Equation (4). In particular, we perform sub-graph optimization at each instant, as seen in Equation (8), where the cost function is formulated using the past history of measurements.

$$\bar{\theta}_T = \operatorname*{argmin}_{\theta_{t-T},\dots,\theta_t} \left( \sum_{s=t-T}^t e_s(\theta_s) \right), \tag{8}$$

where *T* denotes the number of time instants utilized in the sub-graph optimization thread and  $\bar{\theta}_T = [\bar{\theta}_{t-T}, ..., \bar{\theta}_t]^{\top}$  denotes our SLAM-based IM estimate of the overall state vector computed during the sub-graph optimization. We estimate the key pixels in the world frame, represented by  $p_t^j$ , via inverse-mapping defined in Equation (4).

In addition to sub-graph optimization, which is executed at each instant, we also periodically perform full-graph optimization, which optimizes the entire graph and binds the overall drift errors in our 3D world map. In this work, when a new reference keyframe is chosen, we perform fullgraph optimization to compute its corresponding semidense inverse depth map. This occurs at a slower update rate as compared to sub-graph optimization so as to reduce the overall computational complexity. We referred to Engel et al. (2014) for executing the full-graph optimization.

## 2.3 | IM for multi-sensor Graph-SLAM framework

By expressing our extended graph optimization, explained earlier in Section 2.2, as a nonlinear weighted LS problem, we compute the protection levels of the estimated vehicle position using worst-case failure mode slope analysis (Joerger et al., 2014). For this, we refer to the existing rich literature (Joerger et al., 2014; Salós et al., 2013) on worstcase failure mode slope analysis for a linear LS estimator. The formulation of our extended graph optimization as a nonlinear WLS problem is shown later in Section 2.3.2. However, there are challenges involved in applying worstcase failure slope analysis for the protection-level computation of Graph-SLAM framework. The challenges and our design solutions are listed as follows:

- 1. The worst-case failure mode slope is derived for a linear measurement model, but the cost function associated with the Graph-SLAM is nonlinear. So, we linearize the formulation of the cost function at the estimated overall state vector.
- 2. The Graph-SLAM is a sequential method, whereas the worst-case failure mode slope falls under the category of snapshot IM technique. Therefore, we formulate a batch realization of the linearized Graph-SLAM that incorporates not only the measurements received at the current time instant, but also the past time history. This incorporates the temporal aspect of Graph-SLAM in computing the protection level.
- 3. Our Graph-SLAM framework consists of a large number of states and measurements. However, evaluating all possible measurement fault modes is computationally cumbersome. Therefore, using multiple FDI modules, we directly estimate one fault mode, which is a binary vector, such that an element of the fault mode is one if the corresponding measurement is flagged as faulty and zero if otherwise.

### 2.3.1 | Multiple FDI

Based on the estimated overall state vector from the extended graph optimization explained in Section 2.2, we independently compute the measurement residuals associated with GPS pseudoranges and non-sky pixel intensities. In our multiple FDI module, we evaluate the GPS residuals by analyzing them against the temporal correlation of their non-faulty error distribution and vision residuals using spatial correlation across image pixels.

**GPS faults**: To detect and isolate GPS faults in pseudoranges, we evaluate each residual against an empirical Normal distribution, which represents the measurement error distribution during non-faulty conditions. This is justified because the GPS measurements follow a Gaussian distribution during non-faulty conditions (Bhamidipati & Gao, 2018). We empirically compute the non-faulty distribution of GPS measurements during initialization, which is described later in Section 3.2. Thereafter, deviation of the measurement residual, denoted by  $\Delta \rho^k$ , from the CDF of its empirical distribution, denoted by  $\Phi^k_{\Delta\rho}$ , is categorized as a fault and the corresponding fault status  $\bar{r}_t^k$  is computed in Equation (9). The formulation of fault status is explained in our prior work (Bhamidipati & Gao, 2019a).

$$\bar{r}_t^k = 4 \Big( \Phi_{\Delta\rho}^k (\Delta\rho^k) - 0.5 \Big)^2 \quad \forall \ k \in \{1, \dots, N\}.$$
(9)

**Vision faults**: Unlike GPS faults, vision faults caused by data association errors exhibit a high spatial correlation across image pixels and low temporal correlation. This is justified because the vision faults are localized to a group of neighboring pixels and are not isolated to a stand-alone pixel. We developed a superpixel-based piecewise RANSAC that performs spatial voting across pixel intensity residuals to detect and isolate vision faults. RANSAC (Conte & Doherty, 2009), a popular outlier detection method in image processing, estimates the optimal fitting parameters of a model via random sampling of data containing both inliers and outliers.

WILEY 🛞 ION

The steps involved in the superpixel-based piecewise RANSAC are described as follows: first, we segment the image into clusters, known as superpixels, based on the color similarity and space proximity between image pixels using superpixel segmentation (Li & Chen, 2015). During initialization, which is described later in Section 3.2, we pre-define the number of superpixels into which the non-sky pixels are segmented to be  $\Gamma$ . For each non-sky superpixel, we denote the pixel intensity vector as  $I^j \forall j \in \{1, ..., \Gamma\}$ , which stacks the pixel intensities of each superpixel. We represent as a 2D plot the expected and received intensities, which are associated with the pixels of the current image and keyframe, respectively. The mapping between the key pixels and pixels of the current image was estimated earlier during extended graph optimization. Next, we estimate the fitted line using RANSAC that passes through the optimal set of inliers and, thereafter, compute the fraction of outliers in the superpixel, which is defined by  $v_i^j$ . Then, utilizing the estimated model parameters of the fitted line, we evaluate the corresponding fraction of outliers at all the other non-sky superpixels, denoted by  $\nu_k^j \forall k \in \{1, \dots, \Gamma\} - j$ . Finally, the fault status at each superpixel is computed as the product of all the estimated outlier fractions, as seen in Equation (10), and the same fault status is assigned to all the pixels within that superpixel. This procedure is repeated for all the non-sky superpixels to compute the fault status of all the non-sky pixels in the keyframe. Our algorithm considers an underlying assumption that there is a sufficient number of superpixels to reach a common consensus. If the number of superpixels associated with non-sky pixels is less, such as in a open-sky setting, a pre-defined penalty is assigned to the vision fault status.

$$\bar{s}_t(\boldsymbol{u}) = \boldsymbol{\nu}_1^j \cdots \boldsymbol{\nu}_{\Gamma}^j \quad \forall \boldsymbol{u} \in \boldsymbol{I}^j.$$
(10)

### 2.3.2 | Protection-level computation

In accordance with the design solutions listed earlier, we linearize the sub-graph optimization framework, seen in Equations (7) and (8), using the Taylor-series expansion (Shepard & Humphreys, 2014). Using measurement equations described in Equations (4) and (5), we derive the batch realization of linearized Graph-SLAM in Equation (11), which is based on the formulation of batch realization derived for a linear LS estimator in Joerger and Pervan (2013). Batch realization refers to stacking the measurement equations obtained across the desired time history.

-WILEY 🛞 ION

$$\Delta \boldsymbol{z}_T = \boldsymbol{C}_T \Delta \boldsymbol{\theta}_T + \boldsymbol{\eta}_T + \boldsymbol{f}_T, \qquad (11)$$

where

- $\Delta \mathbf{z}_T$  denotes the overall measurement vector of subgraph optimization, such that  $\Delta \mathbf{z}_T = [\Delta z_{t-T}, ..., \Delta z_t]^T$  where  $\Delta z_t$  denotes the measurement vector at  $t^{th}$  time and concatenates the GPS pseudoranges, vehicle motion input, satellite ephemeris, and keyframe pixel intensities; similarly,  $\boldsymbol{\eta}_T$  denotes the overall measurement noise;
- $C_T$  denotes the linearized overall measurement model that represents the sub-graph optimization, such that  $C_T = diag\{C_{t-T}, ..., C_t\}$ . Here,  $C_t$  denotes the measurement model at  $t^{th}$  time that vertically stacks the Jacobian associated with the GPS pseudoranges, denoted by H, vehicle motion model and satellite orbital model, denoted by A and non-sky pixel intensities, denoted by B, such that  $C_t = [H, A, J]_t^T$ ;
- $f_T$  denotes the overall fault vector associated with the overall measurement vector and, thereby, stacks measurement faults obtained from individual sensor sources and across time instants from t T to t.

Re-writing Equations (7) and (8), we express our Mestimator-based Graph-SLAM formulation that is solved via Levenberg-Marquardt (Ranganathan, 2004) algorithm, as a weighted nonlinear LS problem in Equation (12).

$$\Delta \bar{\theta}_T = K_T \Delta z_T, \qquad (12)$$

where

- $\mathbf{K}_T$  denotes the overall estimation matrix, such that  $\mathbf{K}_T = diag\{K_{t-T}, ..., K_t\}$ . Here,  $K_t$  denotes the estimation matrix at  $t^{th}$  time, such that  $K_t = [V_t H_t^\top S_t^{-1}, V_t A_t^\top R_t^{-1}, V_t J_t^\top P_t^{-1}]$  and Vdenotes the pseudo-inverse matrix, such that  $V_t =$  $(H_t^\top S_t^{-1} H_t + A_t^\top R_t^{-1} A_t + J_t^\top P_t^{-1} J_t + \beta diag(H_t^\top H_t + A_t^\top A_t + J_t^\top J_t))^{-1}$ ;
- $\beta$  denotes the damping factor of the LM algorithm that is set during initialization, which is described later in Section 3.2.
- S<sub>t</sub>, R<sub>t</sub>, P<sub>t</sub> denotes the M-estimator-based weight functions for the GPS pseudoranges, motion models of vehi-

cle and satellites, and non-sky pixel intensities, respectively, and evaluated at  $\theta = \overline{\theta}_t$ . Details regarding the weight function of an M-estimator are explained later in our prior work (Bhamidipati & Gao, 2018);

Next, we define the overall test statistic, denoted by  $\zeta$ , as the summation of the weighted squared residuals across all the measurements. We consider an assumption that the overall test statistic is the chi-square distributed, denoted by  $\chi_k^2$  under non-faulty conditions, and non-central chi-squared, denoted by  $\chi_{k,\lambda}^2$ , under the presence of GPS faults or vision faults or both.

$$\zeta = \left(\Delta \boldsymbol{z}_T - \boldsymbol{C}_T \Delta \bar{\boldsymbol{\theta}}_T\right)^{\mathsf{T}} \left(\Delta \boldsymbol{z}_T - \boldsymbol{C}_T \Delta \bar{\boldsymbol{\theta}}_T\right), \quad (13)$$

with

$$\zeta = \begin{cases} \chi_k^2 & \mathbf{f}_T = \mathbf{0} \text{ ornon} - \text{faulty,} \\ \\ \chi_{k,\lambda}^2 & \text{otherwise,} \end{cases}$$
(14)

where *k* denotes the number of redundant measurements, i.e., the difference between the number of overall measurements, denoted by *n* and overall states, denoted by *l*, such that k = n - l.  $\lambda$  indicates the non-centrality parameter associated with the overall test statistic during faulty conditions.

According to the worst-case failure mode slope analysis (Salós et al., 2013), as seen in Figure 5, the protection level is calculated as the projection in the position domain of the measurement faults that would generate a non-centrality parameter  $\lambda = \lambda_{th}$  in the overall test statistic  $\zeta$  with the maximum slope. In particular, the noncentrality parameter  $\lambda_{th}$  is estimated from the false-alarm, denoted by  $p_{FA}$  and mis-detection rates, denoted by  $p_{MD}$ , which are set according to the pre-defined integrity requirements.

We formulate the overall measurement fault mode, denoted by  $\mathbf{b}_T$ , where  $\mathbf{b}_T = [\mathbf{b}_{t-T}, \dots, \mathbf{b}_t]^{\mathsf{T}}$ . Here,  $\mathbf{b}_t$  is the measurement fault mode, which is estimated in Equation (15) using GPS and vision fault status obtained via Equations (9) and (10). For this, we consider a pre-defined fault threshold, denoted by  $\kappa$ , such that if the fault status is above  $\kappa$ , the measurement is flagged as faulty in the computation of protection levels. Given that we consider measurement faults in only GPS and vision, the fault status of receiver and satellite motion models is set to zero. However, the corresponding fault vector, which comprises of the exact measurement fault magnitudes, is still unknown. According to Salós et al. (2013), for a given fault mode, the worst-case fault direction that maximizes the integrity risk is the one that maximizes the failure mode slope, which is **FIGURE 4** Pipeline for superpixel-based piecewise RANSAC that is used to estimate the fault status of vision-based superpixels [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]





**FIGURE 5** Protection levels computed as the intersection of worst-case failure mode slope and non-centrality parameter (Salós et al., 2013) [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

seen in Figure 5 and is denoted by  $g_b$ .

$$\mathbf{b}_{t} = \left[\mathbbm{1}_{\left\{\bar{r}_{t}^{1} > \kappa\right\}}, \dots, \mathbbm{1}_{\left\{\bar{r}_{t}^{1} > \kappa\right\}}, \mathbf{0}, \mathbbm{1}_{\left\{\bar{s}(\mathbf{u})_{t} > \kappa\right\}} \\ \forall \mathbf{u} \in \Pi_{kf}\right]_{t}.$$
(15)

In Equation (16), we define the square of failure mode slope, denoted by  $g_b^2$ , as the ratio of squared state estimation error in the vehicle position, which is denoted by  $\epsilon$ , over the squared overall test statistic, which is denoted by  $\zeta$ . Using the linearized equations seen in Equations (11)-(13), we derive the failure slope for the sub-graph optimization framework in terms of the unknown fault vector. For this, we consider  $C_T K_T C_T$  is negligible, which is a valid approximation after the iterative convergence of the graph optimization at any time instant since  $\beta << 1$ . Considering  $n_b$  to be the number of non-zero entries in the fault mode  $\mathbf{b}_T$  estimated via multiple FDI module, we define the

fault matrix, denoted by  $B_b$ , as  $B_b = [I_{n_b \times n_b}, \mathbf{0}_{(n-n_b) \times n_b}]^T$ , where  $I_{n_b \times n_b}$  is an identity matrix of size  $n_b \times n_b$  and  $\mathbf{0}_{(n-n_b) \times n_b}$  is a zero matrix of size  $(n - n_b) \times n_b$ . Note that I is different from the intensity vector I defined earlier in Equation (2). We re-arrange the rows of the  $m_{\epsilon}$ and  $M_{\zeta}$  matrices, such that the faulty measurements are stacked first. Thereafter, we define a transformed fault vector, denoted by  $f_{\zeta}$ , such that  $f_T = B_b M_{\zeta} f_{\zeta}$ . Based on the above-mentioned steps, we describe the failure slope formulation of Graph-SLAM framework in Equation (16).

593

WILEY 🛞 ION

$$g_{\mathbf{b}}^{2} = \frac{\boldsymbol{\varepsilon}^{T}\boldsymbol{\varepsilon}}{\zeta} = \frac{\left(\Delta\boldsymbol{\theta}_{T} - \Delta\bar{\boldsymbol{\theta}}_{T}\right)^{\top}\left(\Delta\boldsymbol{\theta}_{T} - \Delta\bar{\boldsymbol{\theta}}_{T}\right)}{\left(\Delta\mathbf{z}_{T} - \mathbf{C}_{T}\Delta\bar{\boldsymbol{\theta}}_{T}\right)^{\top}\left(\Delta\mathbf{z}_{T} - \mathbf{C}_{T}\Delta\bar{\boldsymbol{\theta}}_{T}\right)},$$
$$= \frac{\mathbf{f}_{T}^{\top}\left[\left(\boldsymbol{\alpha}^{\top}\mathbf{K}_{T}\right)^{\top}\left(\boldsymbol{\alpha}^{\top}\mathbf{K}_{T}\right)\right]\mathbf{f}_{T}}{\mathbf{f}_{T}^{\top}\left[\left(\mathbf{I} - \mathbf{C}_{T}\mathbf{K}_{T}\right)^{T}\left(\mathbf{I} - \mathbf{C}_{T}\mathbf{K}_{T}\right)\right]\mathbf{f}_{T}},$$
$$= \frac{\mathbf{f}_{\zeta}^{\top}M_{\zeta}^{T}m_{\varepsilon}m_{\varepsilon}^{\top}M_{\zeta}\mathbf{f}_{\zeta}}{\mathbf{f}_{\zeta}^{\top}\mathbf{f}_{\zeta}},$$
(16)

where  $\alpha$  extracts a state of interest from the overall state vector  $\boldsymbol{\theta}$ , such that  $\alpha^T = [\mathbf{0}_{1 \times (l-1-l_a)}, 1, \mathbf{0}_{1 \times l_a}]$  where  $l_a$ indicates the number of states before the state of interest.  $M_{\zeta}$  denotes the residual matrix, such that  $M_{\zeta} = (B_{\mathbf{b}}^T [(\mathbf{I} - \mathbf{C}_T \mathbf{K}_T)]B_{\mathbf{b}})^{-1/2}$  and  $m_{\varepsilon}$  represents the state gain matrix, such that  $m_{\varepsilon} = B_{\mathbf{b}}^T (\alpha^T \mathbf{K}_T)^T$ .

Referring to Joerger et al. (2014), for a given fault mode but unknown fault vector, the worst-case failure slope equals the maximum eigenvalue of the corresponding failure slope formulation. Therefore, we express the worstcase failure slope of the Graph-SLAM framework as

$$\bar{g}_{\boldsymbol{b}}^2 = m_{\boldsymbol{\varepsilon}}^T M_{\boldsymbol{\zeta}} M_{\boldsymbol{\zeta}}^T m_{\boldsymbol{\varepsilon}}.$$
(17)

Next, we compute protection level  $\bar{\chi}_t$  for the state of interest considered in  $\alpha^T$ , seen in Equation (18), as the y-coordinate that corresponds to the integrity metric  $\lambda_{th}$  along the line passing through the origin and with slope given by  $\bar{g}_b^2$ .

$$\bar{\chi}_t = \sqrt{\lambda_{th} \bar{g}_b^2}.$$
(18)

### 3 | EXPERIMENT RESULTS

We analyze the performance of the proposed multisensor SLAM-based IM algorithm that utilizes a GPS receiver and a fish-eye camera. In particular, we validate the improved localization accuracy and tighter protection levels obtained via the proposed multi-sensor SLAM-based IM as compared to GPS-only SLAM-based IM.

### 3.1 | Experimental Setup

We utilize a ground vehicle to perform two real-world experiments: one in the alleyway of Stanford, California and the other in the semi-urban area of Champaign, Illinois. Our experimental setup comprises of a UBLOX NEO-M8U GPS receiver that is connected to a roof-mounted patch antenna. We also have an upward-facing ELP USB camera with 8 mega pixel resolution and fitted with a 180° FOV fish-eye lens. In addition, we utilize the navigation solution estimated by a high-fidelity RTK-GNSS Trimble BD940-INS receiver equipped with a Zephyr antenna as the ground truth reference. We utilize the Robot Operating System (ROS) to publish the measurements obtained from the GPS receiver and fish-eye camera. We record the data in rosbag files, which are later post-processed to validate the proposed algorithm. We chose ROS as our data collection platform because it provides an easy-to-implement interface to obtain measurements from varied sensors and synchronize them to a common UNIX epoch time. The frame transformations between the body frame of GPS receiver, fish-eye camera, and ground truth is measured before the start of the experiment and is given as a userdefined input during the initialization of our SLAM-based IM algorithm.

### 3.2 | Implementation details

Before we post-process the experimental data, we refer to the existing work (Scaramuzza, Martinelli, & Siegwart, 2006) on omni-directional cameras for executing the calibration procedure. In particular, we capture a multiple images of the checkerboard taken from different viewpoints, to estimate the intrinsic parameters of the fish-eye camera, namely  $f_x$ ,  $f_y$ ,  $c_x$ ,  $c_y$ ,  $\xi$ , which are defined earlier in Equation (4). These intrinsic parameters map the image plane to a 3D camera frame, where the origin of the camera frame is at its optical center, x and y axes are aligned with the image plane. This process also undoes the distortion caused in the images because of its wide FOV.

As explained earlier in Section 2, we perform initialization in two stages: one in open-sky and the other in an urban setting. In the first stage of open-sky initialization, we create a 3D world map using the PVT of receiver and GPS satellites, which are estimated via weighted LS (Salós et al., 2013). Here, world frame represents the Earth-Centered Earth-Fixed (ECEF) frame in which the GPS operates. We also empirically analyze the pseudorange residuals to independently compute the non-faulty Gaussian distribution of all visible GPS satellite signals. We estimate the vehicle motion inputs by added simulated measurement noise to the ground truth velocity measurements. This is justified because, as explained earlier, in this work, we consider no faults/anomalies in the vehicle motion model and satellite ephemeris. However, for readers interested in accounting for faults in the vehicle motion model, the motion inputs can be extracted from an IMU sensor and the fault mode vector  $\boldsymbol{b}_t$ , described in Equation (15), can be modified accordingly. The second stage of initialization is executed after the vehicle enters the semi-urban/urban surroundings. We process multiple instances of short-temporal baseline images to select a reference keyframe. Based on the keyframe, we compute the extrinsic parameters, i.e., frame transformation from the camera to world frame.

We utilize a combination of Huber M-estimator (Huber, 2004) and Tukey bisquare M-estimator (Hoaglin, 2003) to formulate the cost function. The justification regarding our choice of M-estimator is explained in our prior work (Bhamidipati & Gao, 2018). During initialization and when the total number of landmarks are  $\leq 15$ , we utilize Huber M-estimator to ensure convergence and in all the other conditions, we opt for a bisquare M-estimator to achieve better accuracy. Detailed explanation regarding the M-estimator equations and the corresponding weight functions are discussed in our prior work (Bhamidipati & Gao, 2018). Based on heuristic analysis, we set the parameters of different modules in the proposed multi-sensor SLAM-based IM as follows: vision coefficients as  $b_{LOS} = 2$ ,  $b_{NLOS} = 3$ ,  $a_{LOS} = 5$  and  $a_{NLOS} = 50$ , number of superpixels as  $\Gamma = 50$ , damping factor of LM algorithm as  $\beta =$ 0.02. To minimize mis-detection rates, we set a conservative value of  $\eta = 0.7$  for the pre-defined sky threshold described in Equation (3). Also, the fault status threshold of our multi-sensor SLAM-based IM is set as  $\kappa = 0.65$ .



**FIGURE 6** Route taken by a ground vehicle during the first experiment conducted for 45 s. For t < 8 s, which is highlighted in magenta, the vehicle operates in open-sky conditions. Between t = 8 - 25 s, which is indicated by the red highlighted region, the vehicle passes from under an overhead bridge and suffers from severely degraded satellite visibility. For t > 25 s, which is highlighted in blue, the vehicle operates in the semi-urban region with tall buildings that block a section of the sky [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

## 3.3 | Experiment-1: Alleyway in Stanford, California

The first set of experiments are conducted in an alleyway in Stanford, California where the ground vehicle drove along the street for a duration of 45 s. As seen in Figure 6, the entire duration of the experiment can be divided into three urban contexts: first is the magenta highlighted region during t < 8 s that represents relatively open-sky conditions, second is the red highlighted region between t = 8 - 25 s that represents the region where the ground vehicle passes from under an overhead bridge, and third is the blue highlighted region for t > 25 s that indicates a semi-urban setting with feature-rich visual surroundings. The aim of these experiments is to analyze the convergence trend of the proposed multi-sensor SLAMbased IM algorithm and its performance in the presence of different urban challenges.

In Figure 7, we demonstrate an improved performance of the proposed algorithm that utilizes the GPS and fisheye camera. Our first and second stage of initialization, explained earlier in Section 3.2, is carried out till t = 2 s. We observe that the weighted LS approach using GPS-only shows large localization errors both when the ground vehicle passes from under the overhead bridge and also when surrounded by tall buildings. Unlike the LS approach that shows a mean of 19.51 m and standard deviation of 9.12 m, the proposed multi-sensor SLAM-based IM algo-

rithm demonstrates a low mean of 6.14 m and standard deviation of 0.97 m. We also demonstrate robust protection levels that accurately bind the localization errors for the entire duration. The convergence of our protection levels is achieved at around  $t \approx 7$  s. The size of the protection level shows a low mean of 6.9 m and standard deviation of 2.44 m. Here, the size of the protection level is indicated in Figure 7 by the gap between the black line, which denotes the protection level and blue line, which denotes the localization error of SLAM-based IM.

## 3.4 | Experiment-2: Semi-urban area of Champaign, Illinois

The second set of experiments are conducted in the semiurban area of Champaign, Illinois for a duration of 100 s. During t = 0 - 70 s, the ground vehicle operates in opensky conditions, thereby experiencing no GPS faults but less visual features. In Figure 8, the blue highlighted region suffers from vision challenges, namely illumination variations due to sunlight and shadows, that cause data association errors across images. Similarly, the red highlighted region is enclosed with tall buildings that lead to multipath effects in the GPS measurements. For instance, at t = 78 s, we showcase the true overlap of the GPS satellite positions over the fish-eye image, where three out of the seven visible satellites are likely to be affected by multipath.



**FIGURE 7** Comparison of the localization errors obtained via multi-sensor SLAM-based IM that utilizes GPS and fish-eye camera, which is plotted in blue, with that of the traditional least squares, which is plotted in red. Tight protection levels are also achieved via multi-sensor SLAM-based IM that accurately binds the localization errors for the entire duration [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

**FIGURE 8** Route taken by a ground vehicle during the experiment conducted for 100 s. Between t = 70 - 100 s, the vehicle experiences GPS faults due to multipath and vision faults due to illumination variations. At t = 78 s, the overlap of the skyplot of GPS satellites with the fish-eye image shows the multipath-affected GPS measurements [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

596



Figure 9 shows the average fault status of GPS pseudoranges and vision superpixels, as indicated in red and blue, respectively. Given that the ground vehicle navigates in open-sky conditions for t < 70 s, the average GPS fault status estimated via our multiple FDI module is low, whereas the average vision fault status is high due to the feature-less surroundings. As the vehicle passes through the red highlighted region shown in Figure 8 that represents the semi-urban area, the average fault status of vision is low but that of the GPS increases due to multiple FDI modules in the challenging semi-urban area, i.e., for t > 70 s during which the ground vehicle experiences GPS

and vision faults. Figure 10(a) plots that the individual GPS fault status of three out of the seven visible satellites with PRNs 6, 12, and 2. In accordance with the skyplot shown in Figure 8, our proposed SLAM-based IM algorithm successfully flags the satellites with PRN 6 and 12 as faulty while accurately estimating the high-elevation satellite with PRN 2 as non-faulty. A key point to note is that we consider fault status to simply indicate the confidence in the reliability of the landmark-measurement pair. Based on our earlier discussion related to Figure 1, the fault status doesn't flag the measurement based on whether its a reflected signal or not, but on the relevance of information provided by the measurement. During the same duration,



**FIGURE 9** Performance of our multiple FDI module via average fault status of GPS pseudoranges, indicated in red, and vision superpixels, indicated in blue. When the ground vehicle navigates through the semi-urban region, i.e., for t > 70 s, the average fault status associated with GPS is high due to multipath, whereas vision is low due to rich features [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]



**FIGURE 10** Estimated fault status of (a) GPS measurements and (b) vision superpixels during t = 70 - 100 s, i.e., when the ground vehicle navigates through the semi-urban area. In (a), our multiple FDI module successfully detects satellites with PRN 6, 12 as faulty while accurately estimating the PRN 2 as non-faulty. In (b), where each marker indicates a superpixel, the trend of fault status associated with superpixels is low given the rich features but, later, increases due to illumination changes [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

we also analyze the vision fault status associated with the superpixels. In Figure 10(b), at each time instant, we plot the top four fault statuses of the superpixels, such that each marker represents a superpixel. We observe that in the urban region, the value of the associated vision fault status decreases due to feature-rich tall buildings in urban areas. However, when the vehicle enters the blue highlighted region seen in Figure 10(b), the illumination variations induced by the bright sunlight causes the fault status associated with certain superpixels to show an increasing trend.

We demonstrate the improved performance of the SLAM-based IM algorithm that utilizes GPS and fisheye camera seen in Figure 11(a), as compared to the SLAM-based IM algorithm that utilizes GPS-only seen in Figure 11(b). By utilizing GPS and fish-eye camera, we demonstrate higher localization accuracy, with a mean of 8.8 m and standard deviation of 1.73 m, as compared to employing GPS-only that shows a mean of 16.2 m and standard deviation of 2.86 m. We also validate that a lower mean size of protection levels are estimated using GPS and fish-eye camera, i.e., 6.5 m than using GPS-only, i.e.,



**FIGURE 11** Comparison of SLAM-based IM in Champaign, Illinois: (a) using GPS and fish-eye camera; (b) using GPS-only. Lower localization errors and tighter protection levels are achieved via GPS and fish-eye camera as compared to GPS-only [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

10.5 m thereby, achieving tighter protection levels. Here, the size of the protection level denotes the gap between the black line, which indicates the protection level, and blue line, which indicates the localization error associated with SLAM-based IM.

In Figure 11(a), we also compare the localization accuracy of the proposed multi-sensor SLAM-based IM, indicated by the blue line, with that of multi-sensor SLAM-based IM that excludes GPS satellites as landmarks, indicated by the green line. To perform this analysis, instead of considering the GPS satellite state vector  $\mathbf{y}^k$  as unknown in the overall state vector  $\boldsymbol{\theta}$ , we consider this to be known information that is computed via satellite ephemeris. We observe that by excluding GPS satellites as landmarks, we achieve localization errors that are better than GPS-only Graph-SLAM but equal or worse than multi-sensor Graph-SLAM. This is justified because, as explained earlier in Section 1.1, not considering GPS satellites as landmarks potentially lowers the available measurement redundancy in the presence of multipath effects.

### 4 | CONCLUSIONS

598

We proposed a Simultaneous Localization and Mapping (SLAM)-based Integrity Monitoring (IM) algorithm using a GPS and fish-eye camera that estimates the protection levels of the Graph-SLAM framework while accounting for multiple faults in GPS and vision. We developed a hybrid sky detection algorithm to distinguish the non-sky and sky pixels, which are later used in graph optimization and GPS measurement covariance, respectively. By utilizing the GPS pseudoranges, non-sky pixel intensities and receiver and satellite motion models, we performed graph optimization via the M-estimator-based Levenberg Marquardt algorithm. We simultaneously estimated the state vector of the vehicle, GPS satellites, and key image pixels in the world frame. We estimated the fault mode vector by independently evaluating the measurement residuals against an empirical Normal distribution for GPS faults and using our developed superpixel-based piecewise RANSAC for vision faults. We computed the protection levels via the worst-case failure slope analysis that estimates the maximum eigenvalue associated with the failure slope formulation for the batch realization of the linearized Graph-SLAM framework.

We conducted two real-world experiments using a ground vehicle: one in an alleyway in Stanford, California and the other in a semi-urban area in Champaign, Illinois. In the first set of experiments, we demonstrated the robust performance of the proposed SLAM-based IM algorithm under different urban conditions, namely open-sky, overhead bridge, and dense building infrastructure. In the second set of experiments, we successfully detected and isolated multiple measurement faults in GPS and vision. We demonstrated higher localization accuracy using our proposed algorithm with a mean of 8.8 m and standard deviation of 1.73 m, as compared to GPS-only that shows a mean of 16.2 m and standard deviation of 2.86 m. We also validated that the mean size of protection levels are estimated using GPS and fish-eye camera, i.e., 6.5 m is lower than using GPS-only, i.e., 10.5 m.

### ORCID

### Sriramya Bhamidipati D https://orcid.org/0000-0002-2158-5365

*Grace Xingxin Gao* https://orcid.org/0000-0002-1807-8637

### REFERENCES

Bhamidipati, S., & Gao, G. X. (2018). Multiple GPS fault detection and isolation using a Graph-SLAM framework. In Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018), pp. 2672–2681. https:// doi.org/10.33012/2018.16030

- Bhamidipati, S., & Gao, G. X. (2019a). Distributed cooperative SLAMbased integrity monitoring via a network of receivers. In Proceedings of the 32st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019), pp. 2023– 2034. https://doi.org/10.33012/2019.16882
- Bhamidipati, S., & Gao, G. X. (2019b). SLAM-based integrity monitoring using GPS and fish-eye camera. In Proceedings of the 32st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019), pp. 4116–4129. https://doi.org/ 10.33012/2019.17117
- Binjammaz, T., Al-Bayatti, A., & Al-Hargan, A. (2013). GPS integrity monitoring for an intelligent transport system. In 2013 10th Workshop on Positioning, Navigation and Communication (WPNC), pp. 1–6. https://doi.org/10.1109/WPNC.2013.6533268
- Bonin-Font, F., Ortiz, A., & Oliver, G. (2008). Visual navigation for mobile robots: A survey. *Journal of Intelligent and Robotic Systems*, 53(3), 263. https://doi.org/10.1007/s10846-008-9235-4
- Borio, D., & Gioia, C. (2016). Galileo: The added value for integrity in harsh environments. *Sensors (Switzerland)*, 16(1), 1–23. https:// doi.org/10.3390/s16010111
- Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., ... Leonard, J. J. (2016). Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6), 1309–1332. https://doi.org/ 10.1109/TRO.2016.2624754
- Caruso, D., Engel, J., & Cremers, D. (2015). Large-scale direct SLAM for omnidirectional cameras. In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 141–148. https: //doi.org/10.1109/IROS.2015.7353366
- Chen, X., Hu, W., Zhang, L., Shi, Z., & Li, M. (2018). Integration of low-cost GNSS and monocular cameras for simultaneous localization and mapping. *Sensors*, 18(7), 2193. https://doi.org/10.3390/ s18072193
- Conte, G., & Doherty, P. (2009). Vision-based unmanned aerial vehicle navigation using geo-referenced information. *EURASIP Journal on Advances in Signal Processing*, 2009, 387308. https://doi.org/ 10.1155/2009/387308
- Dusha, D., & Mejias, L. (2012). Error analysis and attitude observability of a monocular GPS/visual odometry integrated navigation filter. *The International Journal of Robotics Research*, 31(6), 714– 737. https://doi.org/10.1177/0278364911433777
- Eckenhoff, K., Geneva, P., & Huang, G. (2019). Closed-form preintegration methods for graph-based visual-inertial navigation. *The International Journal of Robotics Research*, *38*(5), 563–586. https: //doi.org/10.1177/0278364919835021
- El-Mowafy, A., & Kubo, N. (2017). Integrity monitoring of vehicle positioning in urban environment using RTK-GNSS, IMU and speedometer. *Measurement Science and Technology*, 28(5). http: //doi.org/10.1088/1361-6501/aa5c66
- Engel, J., Schöps, T., & Cremers, D. (2014). LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision*, pp. 834–849.
- Gao, W., Zhang, X., Yang, L., & Liu, H. (2010). An improved Sobel edge detection. In 2010 3rd International Conference on Computer Science and Information Technology, pp. 67–71. https://doi.org/10. 1109/ICCSIT.2010.5563693

Groves, P. D., & Jiang, Z. (2013). Height aiding, c/n 0 weighting and consistency checking for GNSS NLOS and multipath mitigation in urban areas. *The Journal of Navigation*, 66(5), 653–669. https: //doi.org/10.1017/S0373463313000350

WILEY 🛞 ION

- Haque, M., Murshed, M., & Paul, M. (2008). A hybrid object detection technique from dynamic background using Gaussian mixture models. In 2008 IEEE 10th Workshop on Multimedia Signal Processing, pp. 915–920. https://doi.org/10.1109/MMSP.2008. 4665205
- Hewitson, S., & Wang, J. (2006). GNSS receiver autonomous integrity monitoring (RAIM) performance analysis. *GPS Solutions*, 10(3), 155–170. https://doi.org/10.1007/s10291-005-0016-2
- Hoaglin, D. C. (2003). John W. Tukey and data analysis. *Statistical Science*, 18(3), 311–318. Retrieved from https://www.jstor.org/stable/ 3182748
- Hol, J. D. (2011). Sensor fusion and calibration of inertial sensors, vision, ultra-wideband and GPS (Ph.D. dissertation). Linköping University Electronic Press, Linköping, Sweden.
- Huber, P. J. (2004). *Robust statistics* (Vol. 523). Hoboken, NJ: John Wiley & Sons.
- Joerger, M., Chan, F.-C., & Pervan, B. (2014). Solution separation versus residual-based RAIM. NAVIGATION, 61(4), 273–291. https: //doi.org/10.1002/navi.71
- Joerger, M., & Pervan, B. (2013). Kalman filter-based integrity monitoring against sensor faults. *Journal of Guidance, Control, and Dynamics*, 36(2), 349–361. https://doi.org/10.2514/1.59480
- Joerger, M., & Spenko, M. (2017). Towards navigation safety for autonomous cars. *Inside GNSS*. Retrieved from https://insidegnss. com/towards-navigation-safety-for-autonomous-cars/
- Klein, G., & Murray, D. (2007). Parallel tracking and mapping for small AR workspaces. In 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 225–234. https: //doi.org/10.1109/ISMAR.2007.4538852
- Krishnaswamy, K. (2008). *Sensor fusion for navigation* (US Patent App. 11/673,906). Washington, DC: US Patent and Trademark Office.
- Lashley, M., Bevly, D. M., & Hung, J. Y. (2010). A valid comparison of vector and scalar tracking loops. In *IEEE/ION Position, Location* and Navigation Symposium, pp. 464–474. https://doi.org/10.1109/ PLANS.2010.5507215
- Latif, Y., Cadena, C., & Neira, J. (2014). Robust graph SLAM backends: A comparative analysis. In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2683–2690. https: //doi.org/10.1109/IROS.2014.6942929
- Le Marchand, O., Bonnifait, P., Ibañez-Guzmán, J., Peyret, F., & Bétaille, D. (2008). Performance evaluation of fault detection algorithms as applied to automotive localisation. In ENC-GNSS 2008 — European Navigation Conference. Retrieved from https://hal. archives-ouvertes.fr/hal-00445170/file/Performance\_Evaluation\_ of\_Fault\_Detection\_Algorithms\_as\_Applied\_to\_Automotive\_ Localisation.pdf
- Li, F., Bonnifait, P., Ibanez-Guzman, J., & Zinoune, C. (2017). Lanelevel map-matching with integrity on high-definition maps. In 2017 IEEE Intelligent Vehicles Symposium (IV), pp. 1176–1181. https: //doi.org/10.1109/IVS.2017.7995872
- Li, Z., & Chen, J. (2015). Superpixel segmentation using linear spectral clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1356–1363. https://doi.org/10. 1109/CVPR.2015.7298741

### -WILEY 🛞 ION

- Lourakis, M. I. A. (2005). A brief description of the Levenberg-Marquardt algorithm implemented by levmar. *Foundation of Research and Technology*, 4(1), 1–6. Retrieved from https://users. ics.forth.gr/~lourakis/levmar/levmar.pdf
- Margaria, D., & Falletti, E. (2016). The local integrity approach for urban contexts: Definition and vehicular experimental assessment. Sensors, 16(2), 154. https://doi.org/10.3390/s16020154
- Miro, J. V., Zhou, W., & Dissanayake, G. (2006). Towards vision based navigation in large indoor environments. In 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2096–2102. https://doi.org/10.1109/IROS.2006.282487
- Moghaddam, R. F., & Cheriet, M. (2012). AdOtsu: An adaptive and parameterless generalization of Otsu's method for document image binarization. *Pattern Recognition*, 45(6), 2419–2431. https: //doi.org/10.1016/j.patcog.2011.12.013
- Mur-Artal, R., Montiel, J. M. M., & Tardos, J. D. (2015). ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, *31*(5), 1147–1163. https://doi.org/10.1109/TRO. 2015.2463671
- Ng, Y., & Gao, G. X. (2016). Direct position estimation utilizing nonline-of-sight (NLOS) GPS signals. In Proceedings of the 29th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018), pp. 1279–1284. https://doi.org/ 10.33012/2016.14855
- Ochieng, W. Y., Sauer, K., Walsh, D., Brodin, G., Griffin, S., & Denney, M. (2003). GPS integrity and potential impact on aviation safety. *The Journal of Navigation*, 56(1), 51–65. https://doi.org/10. 1017/S0373463302002096
- Pullen, S., Walter, T., & Enge, P. (2011a). Integrity for nonaviation users. GPS World, 22(7), 28–36. Retrieved from https:// web.stanford.edu/group/scpnt/gpslab/pubs/papers/Pullen\_ GPSWorld\_Jul2011\_Integrity\_for\_Non-Aviation\_Users.pdf
- Pullen, S., Walter, T., & Enge, P. (2011b). SBAS and GBAS integrity for non-aviation users: Moving away from specific risk. In *Proceedings of the ION ITM*, pp. 533–545. Retrieved from https://pdfs.sema nticscholar.org/e7ec/6867715bf2e2600f601271101ef5fa2b4da9.pdf
- Ranganathan, A. (2004). The Levenberg-Marquardt algorithm. *Tutoral on LM Algorithm*, 11(1), 101–110.
- Salós, D., Martineau, A., Macabiau, C., Bonhoure, B., & Kubrak, D. (2013). Receiver autonomous integrity monitoring of GNSS signals for electronic toll collection. *IEEE Transactions on Intelligent Transportation Systems*, 15(1), 94–103. https://doi.org/10.1109/ TITS.2013.2273829
- Santa, J., Úbeda, B., Toledo, R., & Skarmeta, A. F. (2006). Monitoring the position integrity in road transport localization based services. In *IEEE Vehicular Technology Conference*, pp. 1–5. https://doi.org/ 10.1109/VTCF.2006.575
- Scaramuzza, D., Martinelli, A., & Siegwart, R. (2006). A toolbox for easily calibrating omnidirectional cameras. In 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5695–5701. https://doi.org/10.1109/IROS.2006.282372
- Shepard, D. P., & Humphreys, T. E. (2014). High-precision globallyreferenced position and attitude via a fusion of visual slam, carrierphase-based GPS, and inertial measurements. In 2014 IEEE/ION Position, Location and Navigation Symposium-PLANS 2014, pp. 1309–1328. https://doi.org/10.1109/PLANS.2014.6851506
- Shevlyakov, G., Morgenthaler, S., & Shurygin, A. (2008). Redescending M-estimators. Journal of Statistical Planning and Inference, 138(10), 2906–2917. https://doi.org/10.1016/j.jspi.2007.11.008

- Shytermeja, E., Garcia-Pena, A., & Julien, O. (2014). Proposed architecture for integrity monitoring of a GNSS/MEMS system with a Fisheye camera in urban environment. In *International Conference on Localization and GNSS 2014 (ICL-GNSS 2014)*, pp. 1–6. https://doi.org/10.1109/ICL-GNSS.2014.6934179
- Toledo-Moreo, R., Bétaille, D., & Peyret, F. (2009). Lane-level integrity provision for navigation and map matching with GNSS, dead reckoning, and enhanced maps. *IEEE Transactions on Intelligent Transportation Systems*, 11(1), 100–112. https://doi.org/10. 1109/TITS.2009.2031625
- Tominaga, T., & Kubo, N. (2019). Performance assessment of adaptive Kalman filter-based GNSS PVT and integrity in dense urban environment. *Transactions of Navigation*, 4(2), 49–58. Retrieved from https://www.jstage.jst.go.jp/article/jintransnavi/4/2/4\_49/\_ pdf/-char/ja
- Tran, H. T., & Lo Presti, L. (2019). Kalman filter-based ARAIM algorithm for integrity monitoring in urban environment. *ICT Express*, 5(1), 65–71. https://doi.org/10.1016/j.icte.2018.05.002
- Velaga, N. R., Quddus, M. A., Bristow, A. L., & Zheng, Y. (2012). Mapaided integrity monitoring of a land vehicle navigation system. *IEEE Transactions on Intelligent Transportation Systems*, 13(2), 848–858. https://doi.org/10.1109/TITS.2012.2187196
- Walter, T., Blanch, J., Enge, P., Pervan, B., & Gratton, L. (2008). Shaping aviation integrity: Two RAIMs for safety. *GPS World*, 19(4), 42– 49.
- Wang, C.-Y., Li, L.-L., Yang, F.-P., & Gong, H. (2010). A new kind of adaptive weighted median filter algorithm. In 2010 International Conference on Computer Application and System Modeling (ICCASM 2010), Vol. 11, pp. V11–667. https://doi.org/10.1109/ ICCASM.2010.5623118
- Won, D. H., Lee, E., Heo, M., Lee, S.-W., Lee, J., Kim, J., ... Lee, Y. J. (2014). Selective integration of GNSS, vision sensor, and INS using weighted DOP under GNSS-challenged environments. *IEEE Transactions on Instrumentation and Measurement*, 63(9), 2288– 2298. https://doi.org/10.1109/TIM.2014.2304365
- Won, D. H., Yun, S., Lee, Y. J., & Sung, S. (2012). Integration of limited GNSS signals with monocular vision based navigation. In 2012 12th International Conference on Control, Automation and Systems, pp. 1364–1367. Retrieved from https://ieeexplore.ieee.org/document/ 6393048
- Zhang, L., Li, J., Cui, T., & Liu, S. (2017a). An adapted RAIM algorithm for urban canyon environment. In 2017 Forum on Cooperative Positioning and Service (CPGPS), pp. 116–121. https://doi.org/ 10.1109/CPGPS.2017.8075108
- Zhang, L., Li, J., Cui, T., & Liu, S. (2017b). An adapted RAIM algorithm for urban canyon environment. In 2017 Forum on Cooperative Positioning and Service (CPGPS), pp. 116–121. https://doi.org/ 10.1109/CPGPS.2017.8075108
- Zhu, N., Marais, J., Bétaille, D., & Berbineau, M. (2018). GNSS position integrity in urban environments: A review of literature. *IEEE Transactions on Intelligent Transportation Systems*, 19(9), 2762– 2778. https://doi.org/10.1109/TITS.2017.2766768

**How to cite this article:** Bhamidipati S, Gao GX. Integrity monitoring of Graph-SLAM using GPS and fish-eye camera. *NAVIGATION*. 2020;67: 583–600. https://doi.org/10.1002/navi.381