

# Fault Detection Algorithm for Gaussian Mixture Noises: An Application in Lidar/IMU Integrated Localization Systems

Penggao Yan | Zhengdao Li | Feng Huang | Weisong Wen | Li-Ta Hsu\*

Department of Aeronautical and Aviation Engineering, Faculty of Engineering, Hong Kong Polytechnic University, Hong Kong

## Correspondence

\*Li-Ta Hsu

Department of Aeronautical and Aviation Engineering, Faculty of Engineering, Hong Kong Polytechnic University, Hong Kong

Email: [lt.hsu@polyu.edu.hk](mailto:lt.hsu@polyu.edu.hk)

## Abstract

Fault detection is crucial to ensure the reliability of localization systems. However, conventional fault detection methods usually assume that noises in the system have a Gaussian distribution, limiting their effectiveness in real-world applications. This study proposes a fault detection algorithm for an extended Kalman filter (EKF)-based localization system by modeling non-Gaussian noises as a Gaussian mixture model (GMM). The relationship between GMM-distributed noises and the measurement residual is rigorously established through error propagation, which is utilized to construct the test statistic for a chi-squared test. The proposed method is applied to an EKF-based two-dimensional light detection and ranging/inertial measurement unit integrated localization system. Experimental results in a simulated urban environment show that the proposed method exhibits a 30% improvement in the detection rate and a 17%–23% reduction in the detection delay, compared with the conventional method with Gaussian noise modeling.

## Keywords

2D lidar/IMU-based localization, chi-squared test, EKF, fault detection, Gaussian mixture model, non-Gaussian noise

## 1 | INTRODUCTION

### *Gaussian noise modeling is not sufficient for developing fault detection methods:*

Fault detection is essential for localization and navigation systems in some safety-critical applications (Joerger & Pervan, 2016; Osechas et al., 2012; Pervan et al., 1998; Wang et al., 2016), for detecting the occurrence of faults in a system and determining the time at which faults occur (Gao et al., 2015). Fault detection methods can be primarily classified into model-based, knowledge-based, and signal-based methods (Gao et al., 2015). Among these methods, model-based methods, particularly statistical analysis of residuals, appear to be the most popular approach for detecting faults in localization and navigation systems (Angus, 2006; Gao et al., 2020; Puchalski & Giernacki, 2022; Yang et al., 2013), as system models are usually well established and known to designers in these applications (Gao et al., 2015). However, conventional model-based methods often assume that the noises in the system have a Gaussian distribution (Hsu et al., 2017; Joerger et al., 2014; Osechas et al., 2012; Pervan et al., 1998; Walter & Enge, 1995).

For example, Walter and Enge (1995) constructed a test statistic by using the sum of the squares of the range residual errors, which was utilized in a chi-squared test to detect potential faults in pseudorange measurements. The measurement noise was assumed to have a multivariate Gaussian distribution for normal conditions in which satellites have no malfunctions. Wang et al. (2016) designed a fault detection method for a navigation system based on the chi-squared test and the sequential probability ratio test. In their work, all system process noise and measurement noise were modeled as zero-mean Gaussian distributions. Unfortunately, noises in the real world usually have non-Gaussian properties. Examples can be found in global navigation satellite systems (GNSSs) (Rife & Pervan, 2012), micro-electromechanical system (MEMS) inertial sensors (Hou & El-Sheimy, 2003; Lethander & Taylor, 2023), and light detection and ranging (lidar) sensors (Xu et al., 2018). Thus, unrealistic Gaussian assumptions can result in increased false alarm rates (FARs) or degraded fault detection rates (FDRs) in real-world applications, limiting the reliability and effectiveness of methods aiming to prevent systems from faults.

*The Gaussian mixture model (GMM) is promising but remains under-explored in fault detection research:* The modeling of non-Gaussian noise has received increasing attention, and its application in the field of localization and navigation, including multi-sensor fusion, robust localization, and integrity monitoring, has been extensively explored (Davis & Blair, 2015; Langel et al., 2020; Rife, 2018; Wen et al., 2021). One of the most popular approaches is to model noises as a GMM, which represents a probability distribution as a weighted combination of multiple Gaussian distributions. Examples include works by Ali-Loytty and Sirola (2007), who proposed a Gaussian sum filter (GSF) method for hybrid positioning with non-Gaussian noises by approximating the prior density of the state as a Gaussian mixture, Pfeifer and Protzel (2019), who proposed a robust sensor fusion algorithm by adaptively tuning the GMM parameters of the noise distribution, and Blanch et al. (2008), Gao et al. (2022), and Yun et al. (2008), who developed over-bounding algorithms and derived protection levels for integrity monitoring by modeling measurement noises as a GMM. Nevertheless, few studies have considered non-Gaussian noises in fault detection. One example is the work by Yun et al. (2008), who developed a fault detection method using a GSF. In particular, the measurement noise was modeled as a GMM, and several parallel Kalman filters were developed to handle each Gaussian component. The fault detection process was realized by comparing the one-sided tail probability of the residual from the GSF with a predefined threshold. Wang et al. (2022) developed a similar algorithm, except that their approach involved summing the residual of each filter according to the mixture weight and subsequently taking the summation for a chi-squared test to identify potential faults. Numerical experiments have shown that these approaches exhibit improved detection performance compared with conventional Gaussian methods. However, the improvement can be attributed to at least two factors: 1) the difference between the GSF-based detection method and the conventional detection method based on the chi-squared test; 2) the difference between GMM-based noise modeling and Gaussian-based noise modeling. It is challenging to differentiate whether these improvements stem from the differences in the detection methods themselves or from the differences in noise modeling.

*Contributions of this paper:* Recently, Hashemi and Ruths (2019) proposed a fault detection method for a linear time-invariant control system with non-Gaussian noise. In their work, the residual of the observer in the linear time-invariant system is modeled as a GMM and used to construct the test statistics for a chi-squared test. This architecture is consistent with the conventional Gaussian method based on the chi-squared test, providing valuable insights for fairly comparing the effects

of non-Gaussian noise modeling on fault detection problems. Inspired by their work, this study aims to extend the idea to fault detection problems in localization systems with non-Gaussian noises. Specifically, we establish the relationship between the GMM-distributed noises and the residual in an extended Kalman filter (EKF)-based localization system. Then, we transform the residual to a variable whose distribution approaches a standard multivariate normal (MVN) distribution. The Mahalanobis distance from the transformed variable to a standard MVN distribution is taken as the test statistic for a chi-squared test to detect faults in the measurements. The proposed method is then applied to a localization system constructed by integrating lidar measurements and inertial measurement unit (IMU) measurements via the EKF. For a fair comparison of different detection algorithms, a simulated urban environment is constructed based on the three-dimensional (3D) simulator CARLA (Dosovitskiy et al., 2017), making it possible to simulate GMM-distributed noises and ensuring reproducibility of the experiments. Finally, we compare the proposed method with the conventional Gaussian method in the simulated environment, evaluating their performance for two types of measurement failures. The contributions of our study are two-fold:

1. A fault detection algorithm designed for GMM noises. This study presents a fault detection algorithm for localization systems with Gaussian mixture noises, including a comprehensive analysis of the relationship between noises and residuals, a transformation process for the residual based on the law of total covariance, and a chi-squared test. We prove that the measurement residual is a linear combination of the measurement and process noises and also exhibits a GMM distribution. In addition, the proposed method shares the same methodology as the conventional Gaussian method. Thus, the substantial difference in fault detection performance between these two methods is attributed to the difference in noise modeling, making it possible to fairly evaluate the effects of different noise modeling approaches on fault detection tasks in localization systems.
2. A simulated platform for a fair comparison of fault detection algorithms. This study establishes a simulated urban environment based on the 3D simulator CARLA, which provides a fault-free environment for lidar-based localization systems, guaranteeing a fair comparison for different fault detection algorithms. In addition, this platform enables fault injection at specified time periods, providing a valuable tool for evaluating the detection performance for different types of failure in different scenarios.

The remainder of this paper is organized as follows. Section 2 presents the fault detection method for a localization system with GMM noises. We first establish the relationship between residuals of the EKF and the multivariate GMM-distributed noises (Section 2.1), providing a theoretical basis for developing a transformation method based on GMM assumptions (Section 2.2). A fault detection algorithm based on the chi-squared test is then developed in Section 2.3. In Section 3, we apply the proposed fault detection method to an EKF-based lidar/IMU integrated localization system, which primarily includes the construction of the sensor platform, an IMU motion model (Section 3.1), and a two-dimensional (2D) lidar measurement model (Section 3.2). In Section 4, a simulated environment is constructed for fairly comparing the performance of fault detection algorithms, and the fault detection performance of the proposed method is examined in this simulated environment for two types of measurement failures. Finally, Section 5 presents a summary of this work.

## 2 | FAULT DETECTION WITH GMM NOISE MODELING

In this section, we present a fault detection algorithm for EKF-based localization systems with noises characterized by a GMM. We first establish the relationship between noises in the system and the measurement residual in the EKF through error propagation. We prove that the measurement residual is a linear combination of the measurement and process noises through error propagation and that its distribution is also a GMM. A transformation method is then constructed to transform the residual to a variable whose distribution approaches a standard MVN distribution. Finally, we calculate the Mahalanobis distance from the transformed variable to a standard MVN distribution, which is taken as the test statistic for a chi-squared test to detect faults in the measurements.

### 2.1 | Residual Analysis in the EKF

#### 2.1.1 | Relationship Between the Residual and Noises

A statistical analysis of residuals is vital in model-based fault detection methods for localization and navigation systems (Angus, 2006; Joerger & Pervan, 2016; Puchalski & Giernacki, 2022; Rife, 2013). A general measurement model in a localization system can be written as follows:

$$\mathbf{y}_k = h(\mathbf{x}_k) + \boldsymbol{\eta}_k \quad (1)$$

where  $\mathbf{x}_k$  is the system state at time  $k$ ,  $\mathbf{y}_k$  and  $\boldsymbol{\eta}_k$  are the measurement and noise vectors at time  $k$ , respectively, and  $h(\cdot)$  is the measurement function. A general state propagation model can be written as follows:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{v}_{k-1}) \quad (2)$$

where  $f(\cdot)$  is the state propagation function,  $\mathbf{u}_{k-1}$  is the external input at time  $k-1$ , and  $\mathbf{v}_{k-1}$  is the process noise at time  $k-1$ . In an EKF-based localization system, the propagation equations are given by the following:

$$\hat{\mathbf{x}}_k^- = f(\hat{\mathbf{x}}_{k-1}^+, \mathbf{u}_{k-1}, 0) \quad (3a)$$

$$\mathbf{P}_k^- = \mathbf{F}_{k-1} \mathbf{P}_{k-1} \mathbf{F}_{k-1}^T + \mathbf{G}_{k-1} \mathbf{Q}_{k-1} \mathbf{G}_{k-1}^T \quad (3b)$$

where  $\hat{\mathbf{x}}_{k-1}^+$  is the estimated state at time  $k-1$ ,  $\hat{\mathbf{x}}_k^-$  is the predicted state at time  $k$ ,  $\mathbf{P}_{k-1}$  is the covariance matrix of the estimated state given by the EKF at time  $k-1$ ,  $\mathbf{P}_k^-$  is the predicted covariance matrix of the estimated state at time  $k$ ,  $\mathbf{F}_{k-1}$  is the state transition matrix,  $\mathbf{G}_{k-1}$  is the noise Jacobian matrix with respect to  $\mathbf{v}_{k-1}$ , and  $\mathbf{Q}_{k-1}$  is the covariance matrix of  $\mathbf{v}_{k-1}$ . If the EKF receives measurements  $\mathbf{y}_k$  at time  $k$ , the Kalman gain  $\mathbf{K}_k$  is obtained by the following equation (Daum, 2005):

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (4)$$

where  $\mathbf{R}_k$  is the covariance matrix of  $\boldsymbol{\eta}_k$ . The estimated state at time  $k$  is obtained as follows:

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - h(\hat{\mathbf{x}}_k^-)) \quad (5)$$

where  $\mathbf{H}_k$  is the Jacobian matrix of  $h(\mathbf{x}_k)$ . The covariance of the estimated state is given by the following:

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \quad (6)$$

The residual corresponding to the measurements at time  $k$  is as follows:

$$\mathbf{r}_k = \mathbf{y}_k - h(\hat{\mathbf{x}}_k^-) \quad (7)$$

By substituting Equation (1) into Equation (7) and taking the first-order Taylor expansion at  $\hat{\mathbf{x}}_k^-$ , we can obtain the following:

$$\mathbf{r}_k = \mathbf{H}_k (\mathbf{x}_k - \hat{\mathbf{x}}_k^-) + \boldsymbol{\eta}_k \quad (8)$$

where  $\mathbf{H}_k$  is the Jacobian matrix of  $h(\cdot)$  defined with respect to  $\mathbf{x}_k$ . By substituting Equations (2) and (3) into Equation (8) and applying a first-order Taylor expansion at the point  $(\hat{\mathbf{x}}_{k-1}^+, \mathbf{u}_{k-1}, 0)$ , we obtain the following:

$$\mathbf{r}_k = \mathbf{H}_k (\mathbf{F}_{k-1} (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^+) + \mathbf{G}_{k-1} \mathbf{v}_{k-1}) + \boldsymbol{\eta}_k \quad (9)$$

where  $\mathbf{F}_{k-1}$  is the Jacobian matrix of  $f(\cdot)$  defined with respect to  $\mathbf{x}_{k-1}$  and  $\mathbf{G}_{k-1}$  is the Jacobian matrix of  $f(\cdot)$  defined with respect to  $\mathbf{v}_{k-1}$ . In Equation (9),  $\hat{\mathbf{x}}_{k-1}^+$  can be obtained by either state propagation or a measurement update, as discussed below.

(1)  $\hat{\mathbf{x}}_{k-1}^+$  is obtained by state propagation.

In this condition, the EKF does not receive external measurements (such as lidar measurements). The estimated state is given by the predicted state, i.e.,  $\hat{\mathbf{x}}_{k-1}^+ = \hat{\mathbf{x}}_{k-1}^-$ . Therefore, we have the following:

$$\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^+ = \mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^- \quad (10)$$

By substituting Equations (2) and (3) into Equation (10) and taking a first-order Taylor expansion at  $(\hat{\mathbf{x}}_{k-2}^+, \mathbf{u}_{k-2}, 0)$ , we obtain the following:

$$\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^+ = \mathbf{F}_{k-2} (\mathbf{x}_{k-2} - \hat{\mathbf{x}}_{k-2}^+) + \mathbf{G}_{k-2} \mathbf{v}_{k-2} \quad (11)$$

Repeating the operation in Equations (9)–(11) yields the following:

$$\mathbf{r}_k = \mathbf{H}_k \left( \prod_{i=1}^m \mathbf{F}_{k-i} (\mathbf{x}_{k-m} - \hat{\mathbf{x}}_{k-m}^+) + \sum_{i=2}^m \prod_{j=1}^{i-1} \mathbf{F}_{k-j} \mathbf{G}_{k-i} \mathbf{v}_{k-i} + \mathbf{G}_{k-1} \mathbf{v}_{k-1} \right) + \boldsymbol{\eta}_k \quad (12)$$

where  $m$  is the discrete time interval between the last measurement and the current measurement. To simplify the expression, we set the higher-order terms inside the parentheses (i.e., terms with more than two matrix multiplications) to zero and obtain the following:

$$\mathbf{r}_k = \mathbf{H}_k \mathbf{F}_{k-1} \mathbf{G}_{k-2} \mathbf{v}_{k-2} + \mathbf{H}_k \mathbf{G}_{k-1} \mathbf{v}_{k-1} + \boldsymbol{\eta}_k \quad (13)$$

(2)  $\hat{\mathbf{x}}_{k-1}^+$  is obtained by a measurement update.

In this case, we obtain the following expression by substituting Equation (5) into Equation (9):

$$\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^+ = \mathbf{x}_{k-1} - \left( \hat{\mathbf{x}}_{k-1}^- + \mathbf{K}_{k-1} \left( \mathbf{y}_{k-1} - h(\hat{\mathbf{x}}_{k-1}^-) \right) \right) \quad (14)$$

By substituting Equation (1) into Equation (14) and taking the first-order Taylor expansion at  $\hat{\mathbf{x}}_{k-1}^-$ , we have the following:

$$\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^+ = (\mathbf{I} - \mathbf{K}_{k-1} \mathbf{H}_{k-1}) (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^-) - \mathbf{K}_{k-1} \boldsymbol{\eta}_{k-1} \quad (15)$$

Therefore, we have the following relationship:

$$\begin{aligned} \mathbf{r}_k &= \mathbf{H}_k \mathbf{F}_{k-1} (\mathbf{I} - \mathbf{K}_{k-1} \mathbf{H}_{k-1}) (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^-) \\ &\quad - \mathbf{H}_k \mathbf{F}_{k-1} \mathbf{K}_{k-1} \boldsymbol{\eta}_{k-1} + \mathbf{H}_k \mathbf{G}_{k-1} \mathbf{v}_{k-1} + \boldsymbol{\eta}_k \end{aligned} \quad (16)$$

If we initialize the Kalman filter at  $t = k - 1$ , we have  $E[\hat{\mathbf{x}}_{k-1}^-] = \mathbf{x}_{k-1}$ .  $\hat{\mathbf{x}}_{k-1}^-$  is the unbiased estimator of  $\mathbf{x}_{k-1}$ , and its distribution can be represented by a Gaussian distribution  $\mathcal{N}(\mathbf{x}_{k-1}, \pi_0)$ , where  $\pi_0$  is the covariance matrix of  $\hat{\mathbf{x}}_{k-1}^-$ . We then apply the following definition:

$$\mathbf{e}_{k-1} = \mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}^- \quad (17)$$

with  $\mathbf{e}_{k-1} \sim \mathcal{N}(0, \pi_0)$ . Equation (16) can then be written as follows:

$$\begin{aligned} \mathbf{r}_k &= \mathbf{H}_k \mathbf{F}_{k-1} (\mathbf{I} - \mathbf{K}_{k-1} \mathbf{H}_{k-1}) \mathbf{e}_{k-1} \\ &\quad - \mathbf{H}_k \mathbf{F}_{k-1} \mathbf{K}_{k-1} \boldsymbol{\eta}_{k-1} + \mathbf{H}_k \mathbf{G}_{k-1} \mathbf{v}_{k-1} + \boldsymbol{\eta}_k \end{aligned} \quad (18)$$

As can be seen,  $\mathbf{r}_k$  is a linear combination of  $\boldsymbol{\eta}_k$ ,  $\boldsymbol{\eta}_{k-1}$ ,  $\mathbf{v}_{k-1}$ , and  $\mathbf{e}_{k-1}$ . Assuming that the covariance of the initial state, i.e.,  $\pi_0$ , is considerably small, we have  $\mathbf{e}_{k-1} \approx 0$ . Then,  $\mathbf{H}_k \mathbf{F}_{k-1} (\mathbf{I} - \mathbf{K}_{k-1} \mathbf{H}_{k-1}) \mathbf{e}_{k-1}$  has only a limited impact on Equation (18). If we know the ground truth of  $\mathbf{x}_{k-1}$  (which is possible in a simulation environment, as shown in Section 4), we can initialize the Kalman filter at  $t = k - 1$  with  $\hat{\mathbf{x}}_{k-1}^- = \mathbf{x}_{k-1}$ . Then,  $\mathbf{r}_k$  approaches a linear combination of  $\boldsymbol{\eta}_k$ ,  $\boldsymbol{\eta}_{k-1}$ , and  $\mathbf{v}_{k-1}$ :

$$\mathbf{r}_k = \boldsymbol{\eta}_k - \mathbf{H}_k \mathbf{F}_{k-1} \mathbf{K}_{k-1} \boldsymbol{\eta}_{k-1} + \mathbf{H}_k \mathbf{G}_{k-1} \mathbf{v}_{k-1} \quad (19)$$

This perfect initialization can be approximately regarded as the setting of  $E[\hat{\mathbf{x}}_{k-1}^-] = \mathbf{x}_{k-1}$  with a very small  $\pi_0$ .

If we do not initialize the Kalman filter at  $t = k - 1$ , we can iteratively apply Equations (7)–(19) until the first initialization of the EKF:

$$E[\hat{\mathbf{x}}_1^-] = \mathbf{x}_1 \quad (20)$$

We will find that  $\mathbf{r}_k$  approaches a linear combination of  $\mathbf{v}_{k-1}, \dots, \mathbf{v}_1$  and  $\boldsymbol{\eta}_k, \boldsymbol{\eta}_{k-1}, \dots, \boldsymbol{\eta}_1$  and that their coefficients are related to  $\mathbf{H}_k, \mathbf{H}_{k-1}, \dots, \mathbf{H}_1$ ,  $\mathbf{F}_{k-1}, \dots, \mathbf{F}_1$ ,  $\mathbf{G}_{k-1}, \dots, \mathbf{G}_1$ , and  $\mathbf{K}_{k-1}, \dots, \mathbf{K}_1$ .

### 2.1.2 | Distribution of the Residual

To focus on the development of the fault detection algorithm, we simply take the case that  $\hat{\mathbf{x}}_{k-1}^+$  is obtained by state propagation for further analysis and set the higher-order terms ( $> 2$ ) to zero. Then, Equation (13) is the final expression of the residual at time  $k$ . We use the following relationships:

$$\mathbf{V}_{k-1} = \mathbf{H}_k \mathbf{G}_{k-1} \quad (21)$$

and:

$$\mathbf{N}_{k-1} = \mathbf{H}_k \mathbf{F}_{k-1} \mathbf{G}_{k-2} \quad (22)$$

Then, Equation (13) can be written as follows:

$$\mathbf{r}_k = \mathbf{V}_{k-1} \mathbf{v}_{k-1} + \mathbf{N}_{k-1} \mathbf{v}_{k-2} + \boldsymbol{\eta}_k \quad (23)$$

In this study,  $\mathbf{v}_{k-1}$ ,  $\mathbf{v}_{k-2}$ , and  $\boldsymbol{\eta}_k$  are assumed to be non-Gaussian noises, and we use the multivariate GMM to model them. Assuming that  $\mathbf{v}_{k-1}$  and  $\mathbf{v}_{k-2}$  are independent and identically distributed (i.i.d.), we can obtain the probability density functions (PDFs) of  $\boldsymbol{\eta}$  and  $\mathbf{v}$  as follows:

$$f_{\boldsymbol{\eta}}(\mathbf{x}) = \sum_{j=1}^{m_1} p_j^{\boldsymbol{\eta}} \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_j^{\boldsymbol{\eta}}, \boldsymbol{\pi}_j^{\boldsymbol{\eta}}) \quad (24a)$$

$$f_{\mathbf{v}}(\mathbf{x}) = \sum_{j=1}^{m_2} p_j^{\mathbf{v}} \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_j^{\mathbf{v}}, \boldsymbol{\pi}_j^{\mathbf{v}}) \quad (24b)$$

where  $\sum_{j=1}^{m_1} p_j^{\boldsymbol{\eta}} = 1$ ,  $\sum_{j=1}^{m_2} p_j^{\mathbf{v}} = 1$ ;  $m_1$  and  $m_2$  are the number of Gaussian modes for  $\boldsymbol{\eta}$  and  $\mathbf{v}$ , respectively;  $p_j^{\boldsymbol{\eta}}$  and  $p_j^{\mathbf{v}}$  are the mixture weights (i.e., the prior probability of each Gaussian mode);  $\mathcal{N}(\cdot | \boldsymbol{\mu}_j^{\boldsymbol{\eta}}, \boldsymbol{\pi}_j^{\boldsymbol{\eta}})$  and  $\mathcal{N}(\cdot | \boldsymbol{\mu}_j^{\mathbf{v}}, \boldsymbol{\pi}_j^{\mathbf{v}})$  are the PDFs of each Gaussian mode;  $\boldsymbol{\mu}_j^{\boldsymbol{\eta}} \in \mathbb{R}^{n_1}$  and  $\boldsymbol{\mu}_j^{\mathbf{v}} \in \mathbb{R}^{n_2}$  are the means of each Gaussian mode;  $\boldsymbol{\pi}_j^{\boldsymbol{\eta}} \in \mathbb{R}^{n_1 \times n_1}$  and  $\boldsymbol{\pi}_j^{\mathbf{v}} \in \mathbb{R}^{n_2 \times n_2}$  are the covariance matrices of each Gaussian mode;  $n_1$  is the dimension of the measurement noise; and  $n_2$  is the dimension of the process noise. Inspired by the work of Hashemi and Ruths (2019), one can prove that  $\mathbf{r}_k$  has a multivariate GMM distribution by applying the convolution theorem to the characteristic function of each component of the residual (refer to Appendix B for details):

$$f_{\mathbf{r}_k}(\mathbf{x}) = \sum_{a=1}^{m_2} \sum_{b=1}^{m_2} \sum_{c=1}^{m_1} p_{abc} \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_{abc}, \boldsymbol{\pi}_{abc}) \quad (25)$$

where:

$$p_{abc} = p_a^{\mathbf{v}} p_b^{\mathbf{v}} p_c^{\boldsymbol{\eta}} \quad (26a)$$

$$\boldsymbol{\mu}_{abc} = \mathbf{V}_{k-1} \boldsymbol{\mu}_a^{\mathbf{v}} + \mathbf{N}_{k-1} \boldsymbol{\mu}_b^{\mathbf{v}} + \boldsymbol{\mu}_c^{\boldsymbol{\eta}} \quad (26b)$$

$$\boldsymbol{\pi}_{abc} = \mathbf{V}_{k-1} \boldsymbol{\pi}_a^{\mathbf{v}} \mathbf{V}_{k-1}^T + \mathbf{N}_{k-1} \boldsymbol{\pi}_b^{\mathbf{v}} \mathbf{N}_{k-1}^T + \boldsymbol{\pi}_c^{\boldsymbol{\eta}} \quad (26c)$$



## 2.2 | Transformation Methods

### 2.2.1 | Transformation Based on GMM Assumptions

Section 2.1 showed that the residual is a multivariate GMM-distributed random variable. To align with the architecture of the chi-squared detector, we transform the residual to a new random variable by subtracting its total mean and subsequently pre-multiplying by the principal square root matrix of the total covariance (Hashemi & Ruths, 2019):

$$\mathbf{T}_g = \Sigma^{-1/2} (\mathbf{r}_k - \boldsymbol{\mu}) \quad (27)$$

where  $\boldsymbol{\mu}$  is the total mean and  $\Sigma^{-1/2}$  is the principal square root matrix of the total covariance ( $\Sigma$ ) of the residual, respectively. According to the law of total covariance (Weiss et al., 2006),  $\boldsymbol{\mu}$  and  $\Sigma$  are given by the following:

$$\boldsymbol{\mu} = \sum_{a=1}^{m_2} \sum_{b=1}^{m_2} \sum_{c=1}^{m_1} p_{abc} \boldsymbol{\mu}_{abc} \quad (28a)$$

$$\Sigma = \sum_{a=1}^{m_2} \sum_{b=1}^{m_2} \sum_{c=1}^{m_1} p_{abc} \pi_{abc} + (\boldsymbol{\mu} - \boldsymbol{\mu}_{abc})(\boldsymbol{\mu} - \boldsymbol{\mu}_{abc})^T \quad (28b)$$

In Appendix C, we demonstrate that the transformed variable has a distribution that approaches the standard MVN distribution when each Gaussian component in the GMM-distributed residual has a small difference in the covariance. Therefore, we limit the proposed transformation method to specific applications in which the measurement error can be modeled by a GMM and the covariance of each component is similar. By setting the number of Gaussian components in the GMM to one, the above transformation formulation can be easily extended to situations in which the residual is formulated as a Gaussian distribution (Da, 1994; Liu et al., 2017).

### 2.2.2 | Transformation Based on Gaussian Assumptions

To evaluate the impact of GMM noise modeling on fault detection performance, we also establish a baseline transformation method that utilizes Gaussian assumptions for the noises. Specifically, the measurement noise  $\boldsymbol{\eta}_k^0$  and the process noise  $\mathbf{v}_k^0$  at time  $k$  are modeled as zero-mean Gaussian noises as follows:

$$\boldsymbol{\eta}_k^0 \sim \mathcal{N}(0, \pi_{\eta}^0) \quad (29a)$$

$$\mathbf{v}_k^0 \sim \mathcal{N}(0, \pi_v^0) \quad (29b)$$

where  $\pi_{\eta}^0 \in \mathbb{R}^{n_1 \times n_1}$  and  $\pi_v^0 \in \mathbb{R}^{n_2 \times n_2}$  are the covariance matrices and can be obtained by calculating the variance of samples generated from the noise distribution in Equation (24) through a Monte Carlo simulation. Following the derivation in Section 2.1, the measurement residual at time  $k$  is given by the following:

$$\mathbf{r}_k^0 = \mathbf{V}_{k-1} \mathbf{v}_{k-1}^0 + \mathbf{N}_{k-1} \mathbf{v}_{k-2}^0 + \boldsymbol{\eta}_k^0 \quad (30)$$



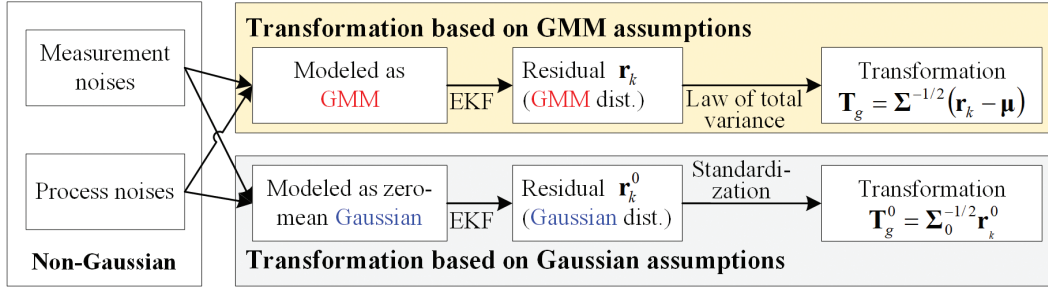


FIGURE 1 Transformation methods based on GMM assumptions and Gaussian assumptions

Let  $\Sigma_0 = \mathbf{V}_{k-1} \pi_v^0 \mathbf{V}_{k-1}^T + \mathbf{N}_{k-1} \pi_v^0 \mathbf{N}_{k-1}^T + \pi_\eta^0$ ; then, the distribution of the measurement residual is given as follows:

$$f_{\mathbf{r}_k^0}(\mathbf{x}) = \mathcal{N}(\mathbf{x} | 0, \Sigma_0^{-1/2}) \quad (31)$$

which is a zero-mean Gaussian distribution. Similar to Equation (27),  $\mathbf{r}_k^0$  can be transformed to a standard MVN-distributed random variable as follows:

$$\mathbf{T}_g^0 = \Sigma_0^{-1/2} \mathbf{r}_k^0 \quad (32)$$

Figure 1 plots the transformation process of the residual based on the two different assumptions. Obviously, the substantial difference between the two transformation methods lies in the method for modeling noise. This important feature enables us to develop a fault detection method with GMM noise modeling that shares the same methodology as the conventional Gaussian method, providing a way to fairly evaluate the effects of different noise modeling approaches on fault detection tasks in a localization system.

### 2.3 | Fault Detection Based on the Chi-Squared Test

The two transformations in Section 2.2, although different, yield a standardized measurement residual that is assumed to have a standard MVN distribution. The  $n_1 \times 1$  vector  $\mathbf{T}_g$  can be written as follows:

$$\mathbf{T}_g = [T_{g,1}, T_{g,2}, \dots, T_{g,n_1}] \quad (33)$$

where  $T_{g,1}, T_{g,2}, \dots, T_{g,n_1}$  are assumed to be mutually independent, standard normal random variables. We can construct the following test statistic  $t_g$ :

$$t_g = \mathbf{T}_g^T \mathbf{T}_g = T_{g,1}^2 + T_{g,2}^2 + \dots + T_{g,n_1}^2 \quad (34)$$

Because  $T_{g,1}, T_{g,2}, \dots, T_{g,n_1}$  are mutually independent,  $t_g$  follows the chi-squared distribution with  $n_1$  degrees of freedom (DOFs). From another perspective, Equation (34) represents the square of the Mahalanobis distance of  $\mathbf{T}_g$  from the standard MVN distribution, which is known to have the same chi-squared distribution. Similarly, we can construct the test statistic  $t_g^0$  in the context of the Gaussian assumption as follows:

$$t_g^0 = \mathbf{T}_g^{0T} \mathbf{T}_g^0 \quad (35)$$

Before performing the hypothesis test with the constructed test statistic, we give a brief introduction to the chi-squared distribution. Its cumulative distribution function is given by the following:

$$F(x; k) = \frac{\gamma\left(\frac{k}{2}, \frac{x}{2}\right)}{\Gamma\left(\frac{k}{2}\right)} \quad (36)$$

where  $\gamma(\cdot)$  is the lower incomplete gamma function,  $\Gamma(\cdot)$  is the gamma function, and  $k$  is the DOF. For a certain value of  $TD \in \mathbb{R}$ , the probability of the chi-squared statistic being less than  $TD$  is given by  $P(x < TD) = F(TD; k)$ , and the associated p-value is given by  $1 - F(TD; k)$ . The value of  $TD$  and the corresponding p-value are interrelated, and their association can be determined from tables of the chi-squared distribution (Fisher & Yates, 1953). The chi-squared test for fault detection associated with the test statistic  $t_g$  at a given significance level  $\alpha$  is as follows:

$$H_1 : t_g > TD_\alpha \quad (37a)$$

$$H_0 : t_g \leq TD_\alpha \quad (37b)$$

where  $TD_\alpha$  is determined by the following:

$$P(t_g > TD_\alpha | H_0) = \alpha \quad (38)$$

A similar hypothesis test can be conducted with the test statistic  $t_g^0$ , which is defined by the similar formulations in Equations (37)–(38) and is omitted for the sake of clarity. For convenience, we denote the fault detection method based on  $t_g$  as the **total Gaussian–GMM** method, and we denote the method based on  $t_g^0$  as the Gaussian method.

In Equation (38), the significance level  $\alpha$  can also be interpreted as the desired FAR in the context of fault detection when the assumption that  $\mathbf{T}_g$  has a standard MVN distribution is valid. However, this assumption is unrealistic in real-world applications. For example, the GMM cannot perfectly model the non-Gaussian noises in lidar measurements. Furthermore, the elements in the residual vector could be correlated, which violates the assumption and eventually results in degradation of the fault detection performance. Therefore, some criteria must be met to examine the real performance of the fault detection results. In this study, two criteria, the FAR and FDR in a period, are formulated as follows:

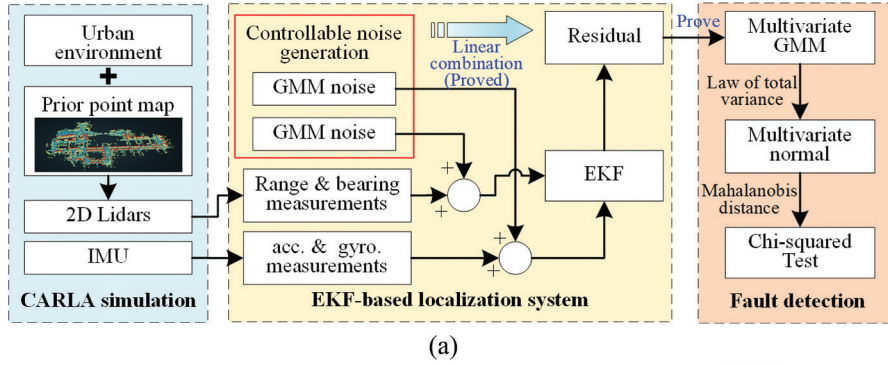
$$FAR = \frac{n_{FP}}{n_{FP} + n_{TN}} \quad (39a)$$

$$FDR = \frac{n_{TP}}{n_{TP} + n_{FN}} \quad (39b)$$

where  $n_{TP}$  is the number of faulty events that are classified as faulty (true positive),  $n_{TN}$  is the number of fault-free events that are classified as normal (true negative),  $n_{FP}$  is the number of fault-free events that are classified as faulty (false positive), and  $n_{FN}$  is the number of faulty events that are classified as normal (false negative).

### 3 | APPLICATION IN AN IMU/LIDAR INTEGRATED LOCALIZATION SYSTEM

This section presents the application of the proposed fault detection in an IMU/lidar integrated system. The architecture of the fault detection process in the localization system is illustrated in Figure 2. A lidar/IMU integrated localization system is constructed by utilizing EKF (Daum, 2005), where the state propagation equation is constructed from the kinematic model of the IMU motion and the measurement function is constructed by matching the extracted line features from seven 2D lidar points to the plane in the prior map. Notably, 2D lidars possess fewer moving parts and components, making them more reliable and easier to maintain in comparison to 3D lidars. This characteristic not only helps to prevent hardware faults but also simplifies the development and evaluation of fault detection methods. In addition, 2D lidars are low-cost but can only provide limited ranging measurements. As a result, it is of great importance to reliably detect faulty measurements from 2D lidars. In this EKF-based localization system, the process and measurement noises are modeled as a multivariate GMM. In Section 2, the nominal residual (defined as the residual of a fault-free system) was proven to be a multivariate GMM. Therefore, we leverage the law of total covariance to transform the residual into a variable whose distribution approaches a standard MVN distribution. Finally, we calculate the Mahalanobis distance from the transformed variable to a standard MVN distribution, which is taken as the test statistic for a chi-squared test to detect faults in the measurements.



ID	x	y	z	Ele	ID	x	y	z	Ele
①	-2	-0.9	1.6	0	⑤	0.7	0	1.8	10°
②	-1.4	0.9	1.4	0	⑥	1.4	-0.9	1.2	0
③	-0.7	0	2	-10°	⑦	2	0.9	1	0
④	0	0	2.4	0					

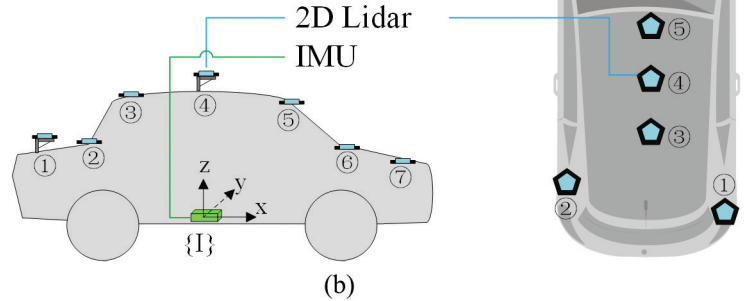


FIGURE 2 (a) Architecture of fault detection in the lidar/IMU integrated localization system; (b) sensor platform of the localization system; Ele: elevation angle

The sensor platform of the localization system is shown in Figure 2(b), where an IMU is placed at the chassis and seven 2D lidars are distributed around the vehicle. The center location and the elevation angle of each lidar in the IMU frame ( $\{I\}$ ) are given in the embedded table. Instead of defining seven lidar frames for each 2D lidar, for clarity, we only define a single lidar frame ( $\{L\}$ ) that is fixed at the center of the fourth lidar. Because the seven 2D lidars have similar measurement models, such a definition can more clearly illustrate the main idea without a loss of generality. Following this definition, we denote  ${}^L\mathbf{p}_I$  as the translation from  $\{I\}$  to  $\{L\}$  and  ${}^L\mathbf{q}_I$  as the rotation (the rotation matrix is  ${}^L\mathbf{R}$ ) from  $\{I\}$  to  $\{L\}$ .  ${}^I\mathbf{p}_L$  and  ${}^I\mathbf{R}$  are extrinsic calibration parameters calibrated in the setup stage of the system. Note that the extrinsic calibration parameters for each 2D lidar are different and are affected by the center location and the elevation angle of the 2D lidar plane.

The state vector  $\mathbf{x}$  of the system is defined as follows:

$$\mathbf{x} = \left[ {}^G\mathbf{p}_I^T, {}^G\mathbf{v}_I^T, {}^I\mathbf{q}_G^T, \mathbf{b}_a^T, \mathbf{b}_g^T \right]^T \quad (40)$$

where  ${}^G\mathbf{p}_I$  and  ${}^G\mathbf{v}_I$  are the position and velocity vectors of the IMU frame ( $\{I\}$ ) in the world frame ( $\{G\}$ ), respectively,  ${}^I\mathbf{q}_G = [q_w, q_x, q_y, q_z]^T$  denotes the rotation from  $\{G\}$  to  $\{I\}$  in terms of quaternions, and  $\mathbf{b}_a^T$  and  $\mathbf{b}_g^T$  are biases of the accelerometer and gyroscope measurements, respectively. In addition, the rotation matrix associated with  ${}^I\mathbf{q}_G$  is denoted by  ${}^I\mathbf{R}$ . The world frame is fixed at the center of the pre-built point cloud map in the east-north-up coordinate system. For clarity, all symbols used in this chapter are listed in Appendix A. The following sections briefly introduce the motion model and the measurement model of the IMU/lidar integrated system and, more importantly, determine the exact form of  $\mathbf{F}$ ,  $\mathbf{G}$ , and  $\mathbf{H}$ , which are used to construct the test statistic for the fault detection process.

### 3.1 | Motion Model

The kinematic model of the IMU motion in  $\{G\}$  is adopted to propagate the vehicle state (Lefferts et al., 1982). The discrete-time state propagation equation is given by the following:

$$\begin{aligned} \mathbf{x}_k &= f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{v}_{k-1}) \\ &= \begin{bmatrix} {}^G\mathbf{p}_{I,k-1} + {}^G\mathbf{v}_{I,k-1}\Delta t + \frac{1}{2}({}^G\hat{\mathbf{R}}_{k-1}(\mathbf{a}_{m,k-1} - \mathbf{b}_{a,k-1} - \mathbf{n}_a) + {}^G\mathbf{g})\Delta t^2 \\ {}^G\mathbf{v}_{I,k-1} + ({}^G\hat{\mathbf{R}}_{k-1}(\mathbf{a}_{m,k-1} - \mathbf{b}_{a,k-1} - \mathbf{n}_a) + {}^G\mathbf{g})\Delta t \\ \exp\left(\frac{\Delta t}{2}\boldsymbol{\Omega}[\mathbf{w}_{m,k-1} - \mathbf{b}_{g,k-1} - \mathbf{n}_g]\right)^I \mathbf{q}_{G,k-1} \\ \mathbf{b}_{a,k-1} + \mathbf{n}_{wa}\Delta t \\ \mathbf{b}_{g,k-1} + \mathbf{n}_{wg}\Delta t \end{bmatrix} \end{aligned} \quad (41)$$

where  $\Delta t$  is a short time period;  ${}^G\mathbf{p}_{I,k-1}$ ,  ${}^G\mathbf{v}_{I,k-1}$ ,  ${}^I\mathbf{q}_{G,k-1}$ ,  $\mathbf{b}_{a,k-1}$ , and  $\mathbf{b}_{g,k-1}$  are system states at the discrete time  $k-1$ ;  $\mathbf{a}_{m,k-1}$  and  $\mathbf{w}_{m,k-1}$  are the acceleration and gyroscope measurements, respectively;  $\mathbf{n}_a$  and  $\mathbf{n}_g$  are the noises of the acceleration and gyroscope measurements, respectively;  $\mathbf{n}_{wa}$  and  $\mathbf{n}_{wg}$  are zero-mean Gaussian white noise;  $\mathbf{v}_{k-1} = [\mathbf{n}_a^T, \mathbf{n}_g^T, \mathbf{n}_{wa}^T, \mathbf{n}_{wg}^T]^T$  is the process noise vector at

time  $k-1$ ;  ${}^G\hat{\mathbf{R}}_{k-1}$  represents the estimation of the vehicle orientation at time  $k-1$ ; and  $\Omega[\cdot]$  is a  $4 \times 4$  skew symmetric matrix:

$$\Omega[\mathbf{w}] = \begin{bmatrix} -[\mathbf{w}]_{\times} & \mathbf{w} \\ \mathbf{w}^T & 0 \end{bmatrix} \quad (42)$$

where  $[\cdot]_{\times}$  denotes the standard vector cross-product.

By linearizing Equation (41) with respect to  $\mathbf{x}_{k-1}$  and  $\mathbf{v}_{k-1}$ , we can obtain the state transition matrix  $\mathbf{F}_{k-1}$  and the noise Jacobian matrix  $\mathbf{G}_{k-1}$  as follows:

$$\mathbf{F}_{k-1} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{I}_3 \Delta t & 0 & -\frac{1}{2} {}^G\hat{\mathbf{R}}_{k-1} \Delta t^2 & 0 \\ 0 & \mathbf{I}_3 & 0 & -{}^G\hat{\mathbf{R}}_{k-1} \Delta t & 0 \\ 0 & 0 & \frac{\partial g({}^I\hat{\mathbf{q}}_{G,k-1}, \mathbf{w}_{m,k-1})}{\partial {}^I\mathbf{q}_{G,k-1}} & 0 & \frac{\partial g({}^I\hat{\mathbf{q}}_{G,k-1}, \mathbf{w}_{m,k-1})}{\partial \mathbf{w}_{GI,k-1}} \\ 0 & 0 & 0 & \mathbf{I}_3 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I}_3 \end{bmatrix} \quad (43a)$$

$$\mathbf{G}_{k-1} = \begin{bmatrix} -\frac{1}{2} {}^G\hat{\mathbf{R}}_{k-1} \Delta t^2 & 0 & 0 & 0 \\ -{}^G\hat{\mathbf{R}}_{k-1} \Delta t & 0 & 0 & 0 \\ 0 & \frac{\partial g({}^I\hat{\mathbf{q}}_{G,k-1}, \mathbf{w}_{m,k-1})}{\partial \mathbf{w}_{GI,k-1}} & 0 & 0 \\ 0 & 0 & \mathbf{I}_3 \Delta t & 0 \\ 0 & 0 & 0 & \mathbf{I}_3 \Delta t \end{bmatrix} \quad (43b)$$

where  $\frac{\partial g({}^I\hat{\mathbf{q}}_{G,k-1}, \mathbf{w}_{m,k-1})}{\partial {}^I\mathbf{q}_{G,k-1}}$  and  $\frac{\partial g({}^I\hat{\mathbf{q}}_{G,k-1}, \mathbf{w}_{m,k-1})}{\partial \mathbf{w}_{GI,k-1}}$  are derived in Appendix E.2.

### 3.2 | Measurement Model

The laser scan plane of a 2D lidar intersects with several planes. The intersection is called a line segment, which is fitted by a set of 2D lidar points, as shown in Figure 3(a). We adopt the method of Pfister et al. (2003) to extract line segments from raw 2D lidar measurements and find their associated planes. Details are provided in Appendix D.2. The 2D lidar measurement model is constructed by finding the shortest vector in the laser scan plane from the origin of  $\{G\}$  to the plane (Hesch et al., 2010; Zhao & Farrell, 2013), as illustrated in Figure 3(b). Assuming that the laser scan plane is intersected by plane  $\Pi_i$  at line  ${}^L\xi_i^\perp$  in  $\{L\}$ , we can find the shortest vector  ${}^L\mathbf{x}_i = \rho_i {}^L\xi_i$  in the lidar scan plane from the origin of  $\{L\}$  to  $\Pi_i$ , where  ${}^L\xi_i$  is the unit vector,  $\rho_i$  is the length of  ${}^L\mathbf{x}_i$ , and the point  $M$  is the intersection of  ${}^L\mathbf{x}_i$  and  ${}^L\xi_i^\perp$ . The laser beam  ${}^L\mathbf{x}_i$  in the scan laser plane can be represented by ranging and bearing parameters  $(\rho_i, \phi_i)$  in the polar coordinate system, as shown in Figure 3(c). Alternatively,  ${}^L\mathbf{x}_i$  can also be written as  ${}^L\mathbf{x}_i = [\rho_i \cos \phi_i \ \rho_i \sin \phi_i \ 0]^T$  in  $\{L\}$ . The shortest vector from the origin of  $\{G\}$  to plane  $\Pi_i$  is denoted by  $d_i {}^G\pi_i$  and is intersected by  $\Pi_i$  at point  $N$ , where  ${}^G\pi_i$  is the normal of the plane in  $\{G\}$  and  $d_i$  is the length of  $d_i {}^G\pi_i$ . The vector from  $M$  to  $N$  in  $\{G\}$  is denoted by  ${}^G\mathbf{t}_i$ . The measurement model can be written as follows:

$$\phi_i = h_1^i(\mathbf{x}) + \tilde{\phi}_i = \arctan \left( \text{sgn}({}^L d_i) \frac{a_2^i}{a_1^i} \right) + \tilde{\phi}_i \quad (44a)$$

$$\rho_i = h_2^i(\mathbf{x}) + \tilde{\rho}_i = \frac{|^L d_i|}{\sqrt{a_1^{i2} + a_2^{i2}}} + \tilde{\rho}_i \quad (44b)$$

$$^L d_i = d_i - {}^G \pi_i^T ({}^G \mathbf{p}_I + {}^G \mathbf{R}^I \mathbf{p}_L) \quad (44c)$$

$$^L \mathbf{a}_i = {}^L \mathbf{R}^G \pi_i = [a_1^i, a_2^i, a_3^i]^T \quad (44d)$$

where the  $i$ -th measurement  $\mathbf{y}_i = [\phi_i, \rho_i]^T$  gives the ranging and bearing parameters associated with the shortest vector in the laser plane intersected by  $\Pi_i$ ,  $\tilde{\phi}_i$  and  $\tilde{\rho}_i$  are measurement noises, and  $\text{sgn}(\cdot)$  is the sign function. Details are provided in Appendix D.1.

Let  $h^i(\mathbf{x}_k) = [h_1^i(\mathbf{x}_k) h_2^i(\mathbf{x}_k)]^T$  be the measurement function of the  $i$ -th measurement at time  $k$ ,  $\boldsymbol{\eta}_{i,k} = [\tilde{\phi}_{i,k} \tilde{\rho}_{i,k}]^T$  be the associated measurement noise,  $\hat{\mathbf{x}}_k^- = [{}^G \hat{\mathbf{p}}_I^T, {}^G \hat{\mathbf{v}}_I^T, {}^I \hat{\mathbf{q}}_G^T, \hat{\mathbf{b}}_a^T, \hat{\mathbf{b}}_g^T]^T$  be the predicted state, and  $\mathbf{H}_k^i$  be the Jacobian matrix of  $h^i(\mathbf{x}_k)$  defined with respect to  $\mathbf{x}_k$  and evaluated at  $\hat{\mathbf{x}}_k^-$ :

$$\mathbf{H}_k^i = \frac{\partial h^i(\hat{\mathbf{x}}_k^-)}{\partial \mathbf{x}_k} \quad (45)$$

As proven in Appendix E.3,  $\mathbf{H}_k^i$  is given by the following:

$$\mathbf{H}_k^i = \begin{bmatrix} \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} & \frac{1}{\hat{\mu}_{i,k}^-} \hat{\lambda}_{i,k}^{-T} {}^L \mathbf{R} \mathbf{J}_q^* ({}^I \hat{\mathbf{q}}_{G,k}, {}^G \pi_{i,k}) & \mathbf{0}_{1 \times 6} \\ -\text{sgn}({}^L \hat{d}_{i,k}^-) \frac{{}^G \pi_{i,k}^T}{\sqrt{\hat{\mu}_{i,k}^-}} & \mathbf{0}_{1 \times 3} & -\text{sgn}({}^L \hat{d}_{i,k}^-) \frac{{}^G \pi_{i,k}^T}{\sqrt{\hat{\mu}_{i,k}^-}} J_q ({}^I \hat{\mathbf{q}}_{G,k}, {}^I \mathbf{p}_L) + |{}^L \hat{d}_{i,k}^-| \hat{\kappa}_{i,k}^T {}^L \mathbf{R} \mathbf{J}_q^* ({}^I \hat{\mathbf{q}}_{G,k}, {}^G \pi_{i,k}) & \mathbf{0}_{1 \times 6} \end{bmatrix} \quad (46)$$

where:

$$\begin{aligned} \hat{\mu}_{i,k}^- &= a_1^{i-2} + a_2^{i-2}, \hat{\lambda}_{i,k}^{-T} = \text{sgn}({}^L \hat{d}_{i,k}^-) [-\hat{a}_2^{i-}, \hat{a}_1^{i-}, 0] \\ \hat{\kappa}_{i,k}^T &= -\hat{\mu}_{i,k}^{-\frac{3}{2}} [\hat{a}_1^{i-}, \hat{a}_2^{i-}, 0], {}^I \hat{\mathbf{a}}_{i,k} = {}^I \mathbf{R}_G^I \hat{\mathbf{R}}_k^{-G} \pi_{i,k} = [\hat{a}_1^{i-}, \hat{a}_2^{i-}, \hat{a}_3^{i-}]^T \end{aligned} \quad (47)$$

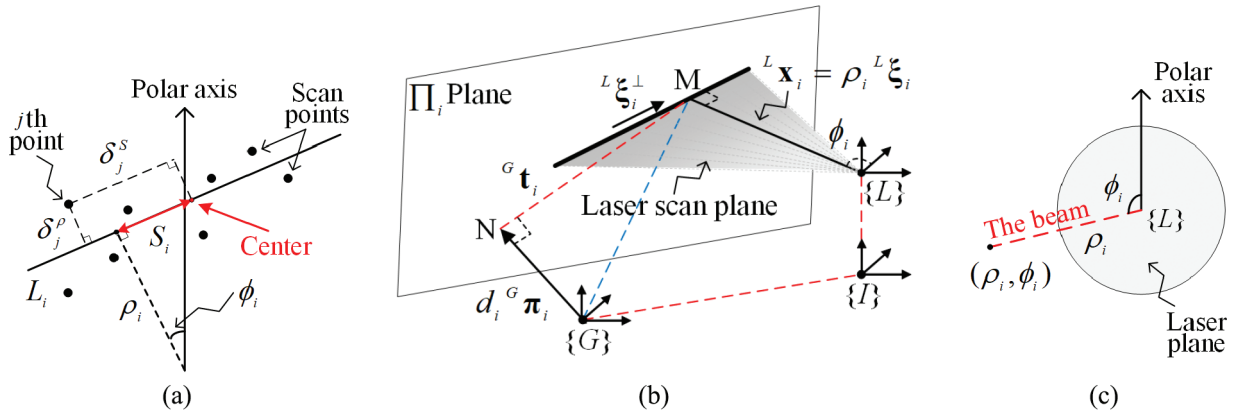


FIGURE 3 (a) A demonstration of extracting line segments from 2D lidar points; (b) a schematic for constructing the 2D lidar measurement model by finding the shortest vector in the laser scan plane from the origin of  $\{L\}$  to plane  $\Pi_i$ ; (c) laser plane with the polar coordinate system

For multiple measurements  $\mathbf{y}_k = [\mathbf{y}_{1,k}^T, \mathbf{y}_{2,k}^T, \dots, \mathbf{y}_{n,k}^T]^T$  at time  $k$ , the measurement function is written as follows:

$$\mathbf{y}_k = h(\mathbf{x}_k) + \boldsymbol{\eta}_k \quad (48)$$

where:

$$h(\mathbf{x}_k) = [h^1(\mathbf{x}_k)^T, h^2(\mathbf{x}_k)^T, \dots, h^n(\mathbf{x}_k)^T]^T \quad (49a)$$

$$\boldsymbol{\eta}_k = [\boldsymbol{\eta}_{1,k}^T, \boldsymbol{\eta}_{2,k}^T, \dots, \boldsymbol{\eta}_{n,k}^T]^T \quad (49b)$$

and  $n$  is the total number of measurements at time  $k$ . Then, the Jacobian matrix of  $h(\mathbf{x}_k)$  can be written as follows:

$$\mathbf{H}_k^T = [\mathbf{H}_k^{1^T}, \mathbf{H}_k^{2^T}, \dots, \mathbf{H}_k^{n^T}]^T \quad (50)$$

It is worth noting that the linearization process presented here differs from that of Hesch et al. (2010) and Zhao and Farrell (2013). In this work, we linearize the measurement function at the nominal state in terms of the position vector and quaternion, as shown in Equations (E27) and (E41) in Appendix E.3, which is more direct and intuitive than the approach in which linearization is performed in the error state (Hesch et al., 2010; Zhao & Farrell, 2013). In addition, linearization at the nominal state is beneficial for establishing clear relationships between the measurement noise and the state and between the measurement noise and the residual, both of which are shown in Section 2.1.

## 4 | NUMERICAL EXPERIMENTS

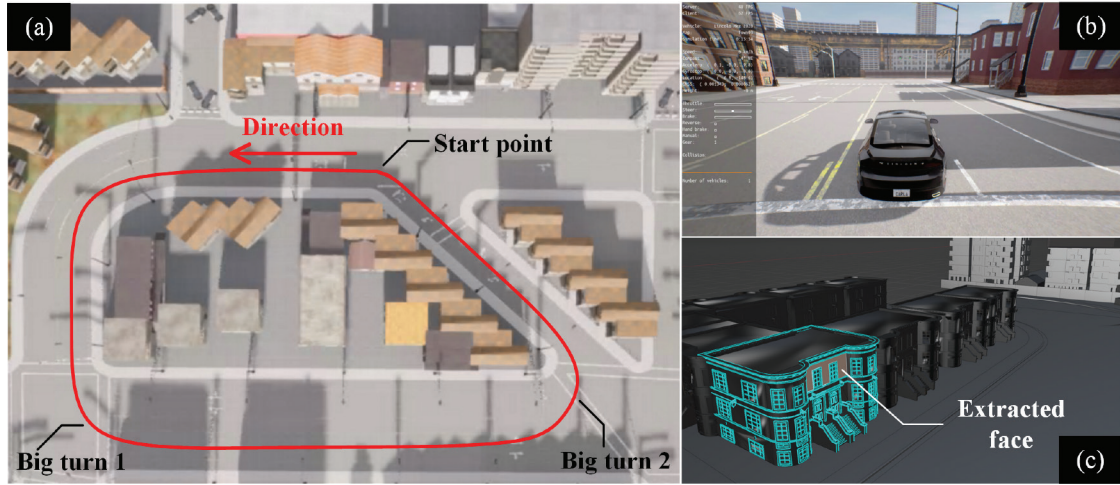
### 4.1 | Construction of the Simulation Platform

In this study, we build a simulated urban environment based on the 3D simulator CARLA (Dosovitskiy et al., 2017). Through automatic and manual checking, we can create a fault-free scenario in the simulated environment and inject specified faults at a specified time. The main reasons for constructing the simulation platform are listed below:

1. The measurement noise of lidars is uncontrollable in the real world. To simulate GMM-distributed noises and ensure reproducibility of the experiments, it is crucial to have a fully controllable environment to examine the performance of the proposed method.
2. Faults in real data sets are unpredictable. To the best of our knowledge, no method can always detect all faults. To control the fault types and the time at which faults occur, it is essential to use a simulation tool to intentionally generate faults of interest at a given time.

Snapshots of the constructed simulation platform are shown in Figure 4. The simulated vehicle is equipped with seven 2D lidars and a simulated IMU sensor, as shown in Figure 4(b). The measurements of 2D lidars are simulated by ray-cast technology, which can accurately reflect the real position of the point hit by each laser beam. Users can include additional noise in the 2D lidar measurements.





**FIGURE 4** Simulated platform

(a) The simulated urban environment is constructed based on CARLA, where the designed track of the ground vehicle is marked as the red line. (b) The simulated vehicle moves along the road by manual control or programmed control. (c) All faces of 3D objects in the simulated environment can be extracted and accurately represented by Blender with semi-automatic checking.

The simulated sensors can directly output readings of physical parameters (such as angular velocity and acceleration) of the vehicle through CARLA. Additional noises are incorporated into the readings with customized configurations, eventually simulating the IMU sensor. In the experiment, the noise from 2D lidar measurements and IMU measurements is configured to be multivariate GMM-distributed. The detailed configuration will be described in Section 4.2. Recall from Section 3.2 that the 2D lidar measurement model requires pre-stored plane information. In the simulated platform, it is much easier to extract this information without introducing additional errors. We accomplish this by exporting the 3D objects in the simulated environment to Blender (Blender Optimization Community, 2018), an open-source 3D modeling and rendering software, to extract the faces of all objects of interest, such as buildings, walls, and roads. As shown in Figure 4(c), the face (plane) information, including the normal, center, and vertices, can be accurately extracted by automatic operation and manual checking in Blender.

## 4.2 | Experimental Setup

Based on the constructed simulation platform, a ground vehicle is programmed to move along the track in Figure 4(a) at a constant speed of 30 km/h. The operation time is 57 s, and the length of the trajectory is 314.3 m. Figure 4(a) presents the simulated trajectory. In the experiment, the output frequency of the lidars is 10 Hz, and lidar measurement noises are configured to have a multivariate GMM distribution. To set the covariance of the 2D lidar measurement noises, we refer to the research conducted by the team from Nagoya University (Carballo et al., 2020), which evaluated the performance of mainstream lidar products. They found that the root-mean-square error of the ranging measurements varies roughly from 0.01 m to 0.08 m within a measured distance of 100 m. Because the bearing measurements can be directly obtained by referring to the rotation frequency of the lidar, we assume that the noise of the bearing measurements is relatively small. In this study, we use the GMM to set the distribution of the measurement noise

$\eta_i = [\tilde{\phi}_i, \tilde{\rho}_i]^T$ . The PDF of the noise  $\eta_i$  in each 2D lidar measurement is given by the following:

$$f_{\eta_i}(\cdot) = p_1 \mathcal{N}\left(\begin{bmatrix} 0 \\ \mu_1 \end{bmatrix}, \begin{bmatrix} 0.0003^2 & 0 \\ 0 & \delta_1^2 \end{bmatrix}\right) + p_2 \mathcal{N}\left(\begin{bmatrix} 0 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} 0.0003^2 & 0 \\ 0 & \delta_2^2 \end{bmatrix}\right) \quad (51)$$

where the bearing noise is set as zero-mean Gaussian with a standard deviation of 0.0003 rad and the range noise is determined by  $\mathbf{p} = [p_1, p_2]$ ,  $\boldsymbol{\mu} = [\mu_1, \mu_2]$ , and  $\boldsymbol{\delta} = [\delta_1, \delta_2]$ . In this experiment, we evaluate four settings for  $\mathbf{p}$ ,  $\boldsymbol{\mu}$ , and  $\boldsymbol{\delta}$ , as shown in Table 1. Taking the noise setting N1 as a baseline, we explore scenarios in which the difference between mixture weights is reduced (N2), the separation between components is decreased (N3), and the disparity between the variance of the components is reduced (N4). The PDFs of the range noises in the four settings are displayed in Figure 5. In addition, a degraded case for which the range noise is generated from a Gaussian distribution is also evaluated, as presented in Table 1.

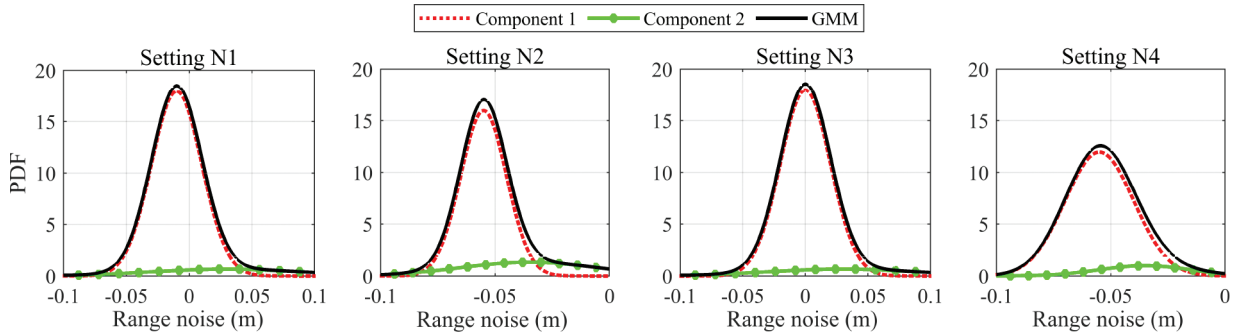
In this study, we assume that each lidar measurement is mutually independent. Therefore, for  $n$  lidar measurements, the PDF of the noise can be expressed as follows:

$$f_{\eta}(\cdot) = p_1 \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}_{2n \times 1}, \begin{bmatrix} \mathbf{M}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{M}_1 \end{bmatrix}\right) + p_2 \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}_{2n \times 1}, \begin{bmatrix} \mathbf{M}_2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{M}_2 \end{bmatrix}\right) \quad (52)$$

where  $\mathbf{M}_1$  and  $\mathbf{M}_2$  are the covariance matrix of each Gaussian component in the single-measurement case, as shown in Equation (51). In contrast, the PDF of the

**TABLE 1**  
Settings for Range Noises of Lidar Measurements

Noise Setting	Mixture $\mathbf{p}$	Mean $\boldsymbol{\mu}$ (m)	Std. $\boldsymbol{\delta}$ (m)
N1	[0.9, 0.1]	[-0.01, 0.03]	[0.02, 0.06]
N2	<b>[0.8, 0.2]</b>	[-0.01, 0.03]	[0.02, 0.06]
N3	[0.9, 0.1]	<b>[0.00, 0.03]</b>	[0.02, 0.06]
N4	[0.9, 0.1]	[-0.01, 0.03]	<b>[0.03, 0.04]</b>
Gaussian	-	0	0.03



**FIGURE 5** Distribution of range noises for 2D lidar measurements in four GMM settings

**TABLE 2**  
IMU Parameters

Sensor	Parameter	Value
Gyroscope	Bias	2 °/h
	Noise	0.15 °/ $\sqrt{h}$
Accelerometer	Bias	3.6 $\mu g$
	Noise	0.012 m/s/ $\sqrt{h}$

**TABLE 3**  
Lidar Range Measurement Failure Coefficients

Failure Type	Group	Coefficient	Value	Failure Time
Step failure	A1	Amplitude	0.1 m	4–20 s
	A2	Amplitude	0.25 m	
Slope failure	B1	Rate	0.05 m/s	34–44 s
	B2	Rate	0.1 m/s	

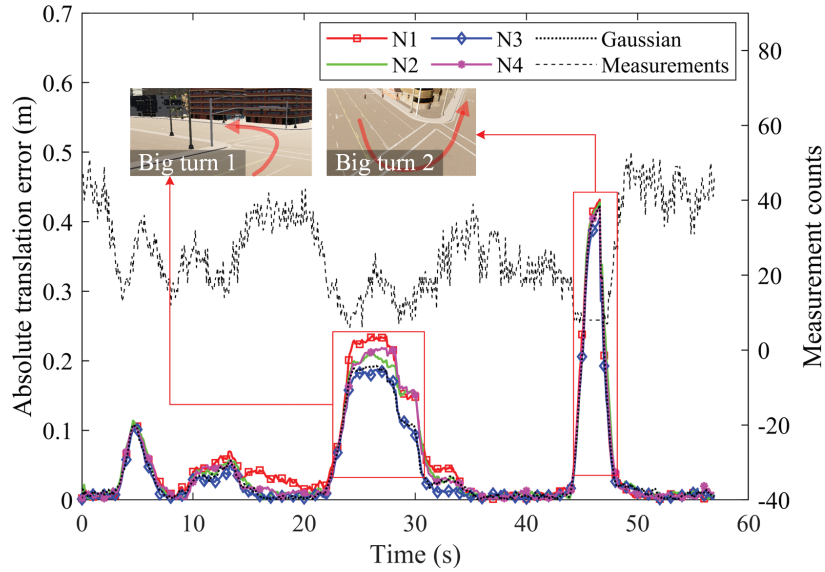
process noise  $\mathbf{v} = [\mathbf{n}_a^T, \mathbf{n}_g^T, \mathbf{n}_{wa}^T, \mathbf{n}_{wg}^T]^T \in \mathbb{R}^{12}$  is set as a zero-mean multivariate Gaussian distribution with the parameters shown in Table 2, which are simulated based on a MEMS-based IMU (Analog Devices, 2018). The output frequency of the simulated IMU is 100 Hz.

In the EKF setting, we use a perfect initial pose to initialize the EKF. In other words, the EKF is assumed to be converged at the initialization and will continue being converged in the following time steps if fault-free measurements are received. Although this is a strong assumption, it is applicable in the simulated environment. By making such an assumption, we can focus on the analysis of the residual instead of the convergence of the localization solution. In this study, the standard deviation of the estimated position is set to be 0.05 m, and the estimated orientation represented by quaternions is set as 0.02 at time  $k$ . To set  $\mathbf{R}$  and  $\mathbf{Q}$ , we calculate the total covariance of  $\boldsymbol{\eta}$  and  $\mathbf{v}$  and set their non-diagonal elements to zero.

In this study, we conducted simulations for two types of single-measurement failure, step failure and slope failure, to examine the fault detection performance across varied scenarios. The coefficients relating to these failures are listed in Table 3. In particular, the slope failure denotes a fault characterized by an escalating magnitude over time (Wang et al., 2016).

### 4.3 | Localization Performance

In the fault-free simulated environment, we examine the localization performance of the proposed lidar/IMU integrated localization system. Figure 6 presents the absolute translation error of the positioning results under five different settings of measurement noise. It is obvious that the localization performance remains relatively consistent across all noise settings. Throughout most epochs, the absolute translation error is below 0.1 m. In addition, the mean absolute translation error is approximately 0.06 m throughout the entire period. However, the absolute translation error shows an increase during the periods of 23–30 s and 45–48 s, coinciding with the vehicle's significant turns, which are depicted in the thumbnail of



**FIGURE 6** Absolute translation error of the positioning results in the fault-free simulated environment

Figure 6. This occurrence can be attributed to the lack of features at the turning locations, resulting in an insufficient number of measurements and, thus, poor observability (Joseph & Murthy, 2019; Sanandaji et al., 2014). Additional sensors, such as cameras and radars, can be employed to capture environmental information to improve the localization performance at these locations. However, such enhancements fall beyond the scope of this work.

#### 4.4 | Fault Detection Performance

In the experiment, we evaluate the fault detection performance of the proposed method with five settings of measurement noise: four settings are generated from a multivariate GMM distribution, and one setting represents a degraded case in which the measurement noise is generated from a Gaussian distribution.

##### 4.4.1 | Step Failure Detection Analysis

Table 4 shows the fault detection results for a step failure when the desired FAR  $\alpha$  is set as 0.05. In group A2, the FDR of the proposed total Gaussian-GMM method for each noise setting exceeds 85%, which is significantly larger than that of the Gaussian method. In particular, for the N2 noise setting, the FDR of the total Gaussian-GMM method surpasses that of the Gaussian method by 30%, showing the most substantial improvement among all noise configurations. As shown in Section 4.2, N2 represents the scenario in which the difference between mixture weights is reduced compared with the baseline setting N1, suggesting that the influence of the Gaussian component with a large variance on the noise distribution is amplified. Compared with the Gaussian method, the total Gaussian-GMM method can effectively capture such changes in the characteristics of the noise distribution, thereby exhibiting better detection performance. Nevertheless, the pre-eminence of the total Gaussian-GMM method in terms of the FDR is accompanied by the concession of an ascending FAR. In all noise settings, the FAR of the total

**TABLE 4**Statistical Results of FAR and FDR for a Step Failure ( $\alpha = 0.05$ )

Range Noise Setting	Failure Group	Total Gaussian-GMM Method		Gaussian Method	
		FAR (%)	FDR (%)	FAR (%)	FDR (%)
N1	A1	7.26	<b>21.88</b>	5.03	20.00
	A2		<b>94.38</b>		78.13
N2	A1	6.70	<b>23.13</b>	3.63	17.50
	A2		<b>85.00</b>		65.63
N3	A1	5.87	<b>21.25</b>	5.59	20.63
	A2		<b>96.25</b>		92.50
N4	A1	2.79	<b>20.00</b>	2.51	15.00
	A2		<b>85.00</b>		73.75
Gaussian $\mathcal{N}(0, 0.03^2)$	A1	3.07	18.75	3.07	18.75
	A2		93.13		93.13

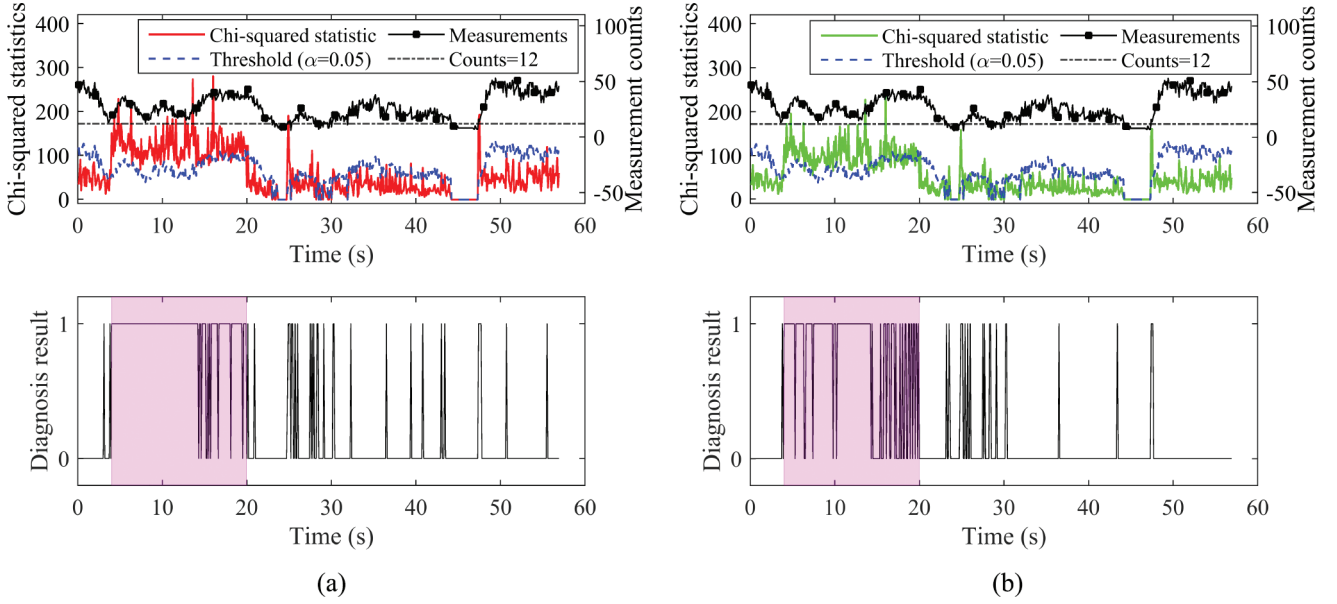
Gaussian-GMM method increases slightly but is still comparable to the desired FAR ( $\alpha = 0.05$ ). Because reducing the missed detection rate is typically prioritized over false positives for multi-sensor navigation systems, it may be acceptable to sacrifice the FAR to a certain extent.

In group A1, both methods yield a substantially lower FDR compared with that in group A2. This result occurs because the amplitude of the injected faults in group A1 is notably smaller than that in group A2, thereby hindering the ability of both methods to detect a fault. Despite this challenge, the total Gaussian-GMM method still exhibits a distinct advantage in detecting faults, showing a much higher FDR than the Gaussian method. This result implies that the total Gaussian-GMM method has greater sensitivity to small faults. For the degraded case in which the measurement noise is generated from a Gaussian distribution, the total Gaussian-GMM method exhibits the same performance as the Gaussian method, as anticipated. The detection results for  $\alpha = 0.01$  and  $\alpha = 0.001$  are listed in Appendix F; both sets of results concur with the findings outlined in Table 4.

Figure 7 displays statistic curves and diagnosis results for the two methods with the N1 noise setting for step failure (group A2). The number of valid line measurements from the lidars is also plotted in the figures, and the chi-squared test is only applied when the number of line measurements is no less than 12. As can be seen, the chi-squared statistics of the total Gaussian-GMM method substantially exceed those of the Gaussian method over the failure period of 4–20 s. In addition, the diagnosis results show that the total Gaussian-GMM method switches less frequently between fault and normal states within the failure period, suggesting that the total Gaussian-GMM method is more stable in fault detection tasks with step failures.

#### 4.4.2 | Slope Failure Detection Analysis

The detection results for a slope failure are listed in Table 5, where the delayed time refers to the interval between the commencement of the fault injection and the first stable detected epoch (i.e., the subsequent diagnosis consistently declares a fault throughout the remaining injection period). In group B1, the total Gaussian-GMM method demonstrates a greater sensitivity to slope failure than the Gaussian method. For the N1, N2, and N4 noise settings, the detection delay for



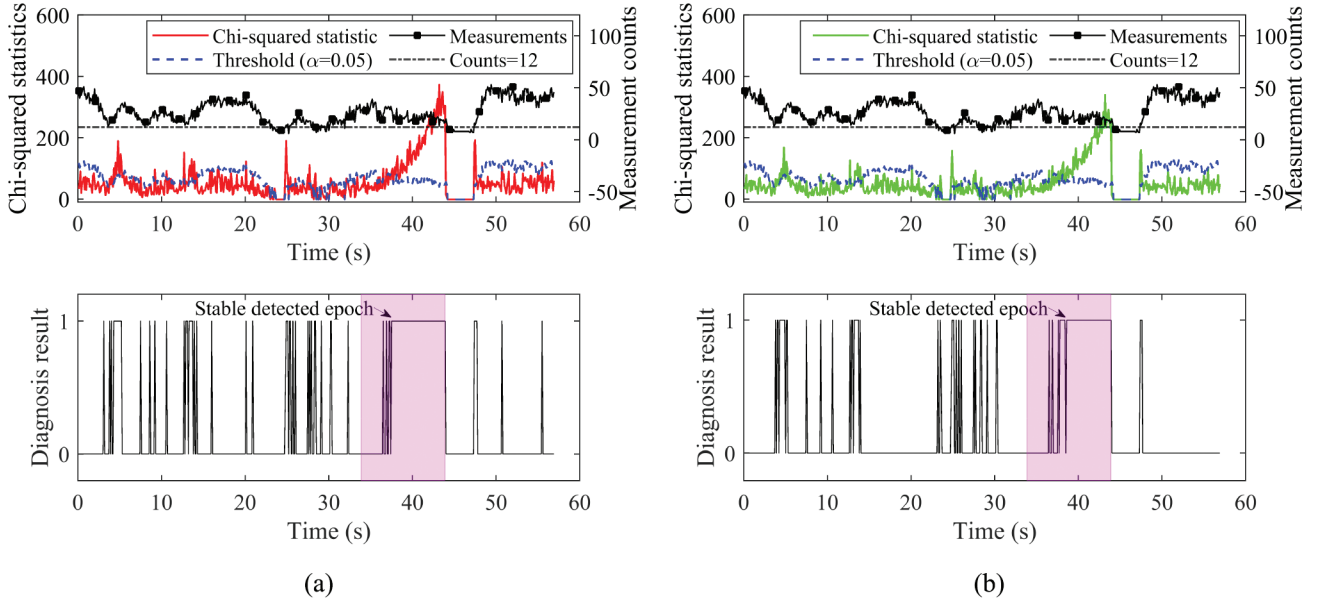
**FIGURE 7** Fault detection results for (a) the total Gaussian-GMM method and (b) the Gaussian method with the N1 noise setting for a step failure (group A2)  
In the diagnosis result plot, “1” indicates a fault, and “0” indicates normal. The failure period is 4–20 s, which is marked by the shaded area.

**TABLE 5**  
Delayed Time of Fault Detection for a Slope Failure ( $\alpha = 0.05$ )

Range Noise Setting	Failure Group	Total Gaussian-GMM Method	Gaussian Method
N1	B1	3.5 s	4.6 s
	B2	1.8 s	1.9 s
N2	B1	4.2 s	5.3 s
	B2	2.1 s	2.6 s
N3	B1	3.3 s	3.3 s
	B2	1.7 s	1.7 s
N4	B1	4.3 s	5.2 s
	B2	2.2 s	2.8 s
Gaussian $\mathcal{N}(0, 0.03^2)$	B1	5.3 s	5.3 s
	B2	1.8 s	1.8 s

the total Gaussian-GMM method is reduced by 17%–23% compared with that of the Gaussian method. Note that the delayed times for the two methods are the same for the N3 noise setting, in which there is a small separation between components. Recalling the transformation method based on the law of total covariance in Section 2.2.1, it is obvious that the second term of the total covariance tends to be zero if all components have a similar mean. In such cases, the total covariance of the GMM distribution approaches the variance of the fitted Gaussian distribution; consequently, the two fault detection methods will exhibit similar performance. This phenomenon is also verified in the step failure experiments under the same noise setting (N3), where the total Gaussian-GMM method exhibits only a 3% and 4% increase in FDR compared with the Gaussian method for groups A1 and A2, respectively. Similar observations are made for group B2, with





**FIGURE 8** Fault detection results for (a) the total Gaussian-GMM method and (b) the Gaussian method with the N1 noise setting for a slope failure (group B1). In the diagnosis result plot, “1” indicates a fault, and “0” indicates normal. The failure period is 34–44 s, which is marked by the shaded area.

the total Gaussian-GMM method generally exhibiting a shorter delayed time than the Gaussian method. For the Gaussian noise setting, the total Gaussian-GMM method degrades to the Gaussian method, yielding the same delayed time.

The statistic curve and diagnosis result for the N1 noise setting with slope failure (group B1) are depicted in Figure 8. It can be observed that the chi-squared statistics of both methods consistently rise during the failure period at 34–44 s because the magnitude of the fault increases with time. One distinct difference between the two methods can be observed in the diagnosis curve: the diagnosis state of the total Gaussian-GMM method switches more frequently during the unstable period than that of the Gaussian method, indicating that the total Gaussian-GMM method is more sensitive to small faults. However, such sensitivity will lead to an increase in false detection epochs during the normal period.

## 5 | CONCLUSION

This study proposes a fault detection method for an EKF-based localization system with GMM noises. The measurement and process noises are modeled as a multivariate GMM distribution, and the residual of the EKF is derived as a linear combination of these noises, which is shown to be multivariate GMM-distributed. Based on the law of total covariance, the residual is transformed into a variable whose distribution approximates a standard MVN distribution. The Mahalanobis distance from the transformed variable to a standard MVN distribution is calculated and taken as the test statistic for a chi-squared test to detect potential faults. The proposed fault detection method is then applied to a lidar/IMU integrated localization system based on the EKF, where the measurement function is constructed by fitting line measurements from seven 2D lidar points to the plane and state propagation is achieved by a kinematic model of IMU motion.



In a simulated urban environment constructed via the 3D simulator CARLA, we examined the detection performance of the proposed method for two types of measurement failures, assessing each under four types of GMM noise settings. In the step failure experiment, the FDR of the proposed total Gaussian-GMM method surpassed that of the conventional Gaussian method by up to 30%, demonstrating the effectiveness of the proposed method in detecting small faults. In addition, the detection delay for the total Gaussian-GMM method was reduced by 17%–23% compared with the Gaussian method for three of the four noise settings in the slope failure experiment, demonstrating the greater sensitivity of the proposed method to a gradually changing failure. Because the total Gaussian-GMM method shares the same fault detection methodology as the Gaussian method, the difference in detection performance between the two methods is attributed to the difference in noise modeling. Therefore, the experimental results suggest that GMM-based noise modeling is beneficial for fault detection tasks in localization systems with non-Gaussian noises, motivating further research in this direction.

This study has several limitations, which also point out future research directions. In deriving the relationship between the residual and noises, we ignored higher-order terms to reduce the computation load. However, this approximation underestimates the uncertainty of the residual, thereby increasing the probability of false detection events in the fault detection task. For integrity applications, some integrity and continuity budgets should be allocated to risks associated with ignoring higher-order terms. A possible solution is to determine the bounding relationship between higher-order terms and to then scale the lower-order terms to compensate for the effects of higher-order terms on the uncertainty of the residual. In addition, we assume that the transformation method based on the law of total covariance can transform the GMM-distributed residual to a variable whose distribution approaches a standard MVN distribution. However, this assumption holds only when each Gaussian component in the GMM-distributed residual has a small difference in the covariance. This constraint is satisfied in the lidar/IMU integrated system. However, when the measurement noise exhibits significantly heavy tails, the transformation method cannot guarantee similarity, thereby hindering the performance of the proposed fault detection algorithm. Further efforts should be made to develop an advanced transformation method to ensure similarity between the transformed distribution and the standard MVN distribution, widening the application of non-Gaussian fault detection algorithms in other localization systems, such as GNSS/IMU integrated systems. Moreover, this study focuses on the detection of single-measurement failure. Future work can extend the proposed fault detection algorithm to multiple-fault detection problems by adopting exhaustive search or greedy search algorithms as reported by Blanch et al. (2015). Last but not least, this study gives little concern to the effects of the geometrical configuration of 2D lidars on state estimation and fault detection performance. For practical purposes, future research could investigate the optimal geometrical configuration based on the simulated platform constructed in this study.

## REFERENCES

- Ali-Loytty, S., & Sirola, N. (2007). Gaussian mixture filters and hybrid positioning. *Proc. of the 20th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2007)*, Fort Worth, TX, 562–570. <https://www.ion.org/publications/abstract.cfm?articleID=7554>
- Analog Devices. (2018). *ADIS16465: Precision MEMS IMU module data sheet (Rev. C)*. <https://www.analog.com/media/en/technicaldocumentation/data-sheets/adis16465.pdf>
- Angus, J. E. (2006). RAIM with multiple faults. *NAVIGATION*, 53(4), 249–257. <https://doi.org/10.1002/j.2161-4296.2006.tb00387.x>

- Blanch, J., Walker, T., & Enge, P. (2015). Fast multiple fault exclusion with a large number of measurements. *Proc. of the 2015 International Technical Meeting of the Institute of Navigation*, Dana Point, CA, 696–701. <https://www.ion.org/publications/abstract.cfm?articleID=12662>
- Blanch, J., Walter, T., & Enge, P. (2008). Position error bound calculation for GNSS using measurement residuals. *IEEE Transactions on Aerospace and Electronic Systems*, 44(3), 977–984. <https://doi.org/10.1109/TAES.2008.4655357>
- Blender Optimization Community. (2018). *Blender - a 3D modelling and rendering package*. Blender Foundation. Stichting Blender Foundation, Amsterdam. <http://www.blender.org>
- Carballo, A., Lambert, J., Monrroy, A., Wong, D., Narksri, P., Kitsukawa, Y., Takeuchi, E., Kato, S., & Takeda, K. (2020). LIBRE: The multiple 3D lidar dataset. *Proc. of the 2020 IEEE Intelligent Vehicles Symposium (IV)*, Las Vegas, NV, 1094–1101. <https://doi.org/10.1109/IV47402.2020.9304681>
- Cline, R. E., & Plemmons, R. J. (1976). L2-solutions to underdetermined linear systems. *SIAM Review*, 18(1), 92–106. <https://doi.org/10.1137/1018004>
- Da, R. (1994). Failure detection of dynamical systems with the state chi-square test. *Journal of Guidance, Control, and Dynamics*, 17(2), 271–277. <https://doi.org/10.2514/3.21193>
- Daum, F. (2005). Nonlinear filters: Beyond the Kalman filter. *IEEE Aerospace and Electronic Systems Magazine*, 20(8), 57–69. <https://doi.org/10.1109/MAES.2005.1499276>
- Davis, B., & Blair, W. D. (2015). Gaussian mixture approach to long range radar tracking with high range resolution. *Proc. of the 2015 IEEE Aerospace Conference*, Big Sky, MT, 1–9. <https://doi.org/10.1109/AERO.2015.7119222>
- Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., & Koltun, V. (2017). CARLA: An open urban driving simulator. *Proc. of the 1st Conference on Robot Learning (CoRL)*, Mountain View, CA, 1–16. ArXiv. <https://doi.org/10.48550/arXiv.1711.03938>
- Fisher, R. A., & Yates, F. (1953). *Statistical tables for biological, agricultural and medical research*. Hafner Publishing Company. [https://books.google.com/books/about/Statistical\\_Tables\\_for\\_Biological\\_Agricu.html?id=qfw9AAAAAYAAJ](https://books.google.com/books/about/Statistical_Tables_for_Biological_Agricu.html?id=qfw9AAAAAYAAJ)
- Gao, G., Gao, S., Hong, G., Peng, X., & Yu, T. (2020). A robust INS/SRS/CNS integrated navigation system with the chi-square test-based robust Kalman filter. *Sensors*, 20(20), 5909. <https://doi.org/10.3390/s20205909>
- Gao, Z., Ding, S. X., & Cecati, C. (2015). Real-time fault diagnosis and fault-tolerant control. *IEEE Transactions on Industrial Electronics*, 62(6), 3752–3756. <https://doi.org/10.1109/TIE.2015.2417511>
- Gao, Z., Fang, K., Wang, Z., Guo, K., & Liu, Y. (2022). An error overbounding method based on a Gaussian mixture model with uncertainty estimation for a dual-frequency ground-based augmentation system. *Remote Sensing*, 14(5), 1111. <https://doi.org/10.3390/rs14051111>
- Hashemi, N., & Ruths, J. (2019). Generalized chi-squared detector for LTI systems with non-Gaussian noise. *Proc. of the 2019 American Control Conference (ACC)*, Philadelphia, PA, 404–410. <https://doi.org/10.23919/ACC.2019.8815139>
- Hesch, J. A., Mirzaei, F. M., Mariottini, G. L., & Roumeliotis, S. I. (2010). A laser-aided inertial navigation system (L-INS) for human localization in unknown indoor environments. *Proc. of the 2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK, 5376–5382. <https://doi.org/10.1109/ROBOT.2010.5509693>
- Hou, H., & El-Sheimy, N. (2003). Inertial sensors errors modeling using Allan variance. *Proc. of the 16th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS/GNSS 2003)*, Portland, OR, 2860–2867. <https://www.ion.org/publications/abstract.cfm?articleID=5475>
- Hsu, L. T., Tokura, H., Kubo, N., Gu, Y., & Kamijo, S. (2017). Multiple faulty GNSS measurement exclusion based on consistency check in urban canyons. *IEEE Sensors Journal*, 17(6), 1909–1917. <https://doi.org/10.1109/JSEN.2017.2654359>
- 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. (2016). Fault detection and exclusion using solution separation and chi-squared ARAIM. *IEEE Transactions on Aerospace and Electronic Systems*, 52(2), 726–742. <https://doi.org/10.1109/TAES.2015.140589>
- Joseph, G., & Murthy, C. R. (2019). Measurement bounds for observability of linear dynamical systems under sparsity constraints. *IEEE Transactions on Signal Processing*, 67(8), 1992–2006. <https://doi.org/10.1109/TSP.2019.2899812>
- Langel, S., Crespillo, O. G., & Joerger, M. (2020). Overbounding sequential estimation errors due to non-Gaussian correlated noise. *Proc. of the 33rd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2020)*, 1054–1067. <https://doi.org/10.33012/2020.17576>
- Lefferts, E., Markley, F., & Shuster, M. (1982). Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 5(5), 417–429. <https://doi.org/10.2514/3.56190>
- Lethander, K. A., & Taylor, C. N. (2023). Conservative estimation of inertial sensor errors using Allan variance data. *NAVIGATION*, 70(3). <https://doi.org/10.33012/navi.563>

- Li, M., & Schwartzman, A. (2018). Standardization of multivariate Gaussian mixture models and background adjustment of PET images in brain oncology. *The Annals of Applied Statistics*, 12(4), 2197–2227. <https://doi.org/10.1214/18-AOAS1149>
- Liu, C., Wang, H., Li, N., & Yu, Y. (2017). Sensor fault diagnosis of GPS/INS tightly coupled navigation system based on state chi-square test and improved simplified fuzzy ARTMAP neural network. *Proc. of the 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Macau, Macao, 2527–2532. <https://doi.org/10.1109/ROBIO.2017.8324800>
- MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge University Press. <https://www.cambridge.org/us/universitypress/subjects/computer-science/pattern-recognition-and-machine-learning/information-theory-inference-and-learning-algorithms?format=HB&isbn=9780521642989>
- Manning, C., & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT Press. <https://mitpress.mit.edu/9780262133609/foundations-of-statistical-natural-language-processing/>
- Osechas, O., Misra, P., & Rife, J. (2012). Carrier-phase acceleration RAIM for GNSS satellite clock fault detection. *NAVIGATION*, 59(3), 221–235. <https://doi.org/10.1002/navi.17>
- Pervan, B. S., Pullen, S. P., & Christie, J. R. (1998). A multiple hypothesis approach to satellite navigation integrity. *NAVIGATION*, 45(1), 61–71. <https://doi.org/10.1002/j.2161-4296.1998.tb02372.x>
- Pfeifer, T., & Protzel, P. (2019). Expectation-maximization for adaptive mixture models in graph optimization. *2019 International Conference on Robotics and Automation (ICRA)*, 3151–3157. <https://doi.org/10.1109/ICRA.2019.8793601>
- Pfister, S., Roumeliotis, S., & Burdick, J. (2003). Weighted line fitting algorithms for mobile robot map building and efficient data representation. *Proc. of the 2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, Taipei, Taiwan, 1304–1311. <https://doi.org/10.1109/ROBOT.2003.1241772>
- Puchalski, R., & Giernacki, W. (2022). UAV fault detection methods, state-of-the-art. *Drones*, 6(11), 330. <https://doi.org/10.3390/drones6110330>
- Rife, J., & Pervan, B. (2012). Overbounding revisited: Discrete error-distribution modeling for safety-critical GPS navigation. *IEEE Transactions on Aerospace and Electronic Systems*, 48(2), 1537–1551. <https://doi.org/10.1109/TAES.2012.6178077>
- Rife, J. H. (2013). The effect of uncertain covariance on a chi-square integrity monitor. *NAVIGATION*, 60(4), 291–303. <https://doi.org/10.1002/navi.45>
- Rife, J. H. (2018). Overbounding risk for quadratic monitors with arbitrary noise distributions. *IEEE Transactions on Aerospace and Electronic Systems*, 54(4), 1618–1627. <https://doi.org/10.1109/TAES.2018.2798258>
- Sanandaji, B. M., Wakin, M. B., & Vincent, T. L. (2014). Observability with random observations. *IEEE Transactions on Automatic Control*, 59(11), 3002–3007. <https://doi.org/10.1109/TAC.2014.2351693>
- Toschi, I., Rodríguez-González, P., Remondino, F., Minto, S., Orlandini, S., & Fuller, A. (2015). Accuracy evaluation of a mobile mapping system with advanced statistical methods. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40, 245–253. <https://doi.org/10.5194/isprsarchives-XL-5-W4-245-2015>
- Walter, T., & Enge, P. (1995). Weighted RAIM for precision approach. *Proc. of the 8th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS 1995)*, Palm Springs, CA, 1995–2004. <https://www.ion.org/publications/abstract.cfm?articleID=2524>
- Wang, J., Xu, C., Shi, M., & Lu, Z. (2022). Protection level for precise point positioning based on Gaussian mixture model. *Proc. of the China Satellite Navigation Conference (CSNC 2022)*, Beijing, China, 45–55. [https://doi.org/10.1007/978-981-19-2580-1\\_4](https://doi.org/10.1007/978-981-19-2580-1_4)
- Wang, R., Xiong, Z., Liu, J., Xu, J., & Shi, L. (2016). Chi-square and SPRT combined fault detection for multisensor navigation. *IEEE Transactions on Aerospace and Electronic Systems*, 52(3), 1352–1365. <https://doi.org/10.1109/TAES.2016.140860>
- Weiss, N., Holmes, P., & Hardy, M. (2006). *A course in probability*. Pearson Addison Wesley. <https://books.google.com.hk/books?id=Be9fJwAACAAJ>
- Wen, W., Pfeifer, T., Bai, X., & Hsu, L. T. (2021). Factor graph optimization for GNSS/INS integration: A comparison with the extended Kalman filter. *NAVIGATION*, 68(2), 315–331. <https://doi.org/10.1002/navi.421>
- Xu, F., Wang, J., Zhu, D., & Tu, Q. (2018). Speckle noise reduction technique for lidar echo signal based on self-adaptive pulse-matching independent component analysis. *Optics and Lasers in Engineering*, 103, 92–99. <https://doi.org/10.1016/j.optlaseng.2017.12.002>
- Yang, L., Knight, N. L., Li, Y., & Rizos, C. (2013). Optimal fault detection and exclusion applied in GNSS positioning. *The Journal of Navigation*, 66(5), 683–700. <https://doi.org/10.1017/S0373463313000155>
- Yun, Y., Yun, H., Kim, D., & Kee, C. (2008). A Gaussian sum filter approach for DGNSS integrity monitoring. *The Journal of Navigation*, 61(4), 687–703. <https://doi.org/10.1017/S0373463308004906>

Zhao, S., & Farrell, J. A. (2013). 2D LIDAR aided INS for vehicle positioning in urban environments. *Proc. of the 2013 IEEE International Conference on Control Applications (CCA)*, Hyderabad, India, 376–381. <https://doi.org/10.1109/CCA.2013.6662778>

**How to cite this article:** Yan, P., Li, Z., Huang, F., Wen, W., and Hsu, L. -T. (2025). Fault detection algorithm for Gaussian mixture noises: An application in lidar/IMU integrated localization systems. *NAVIGATION*, 72(1). <https://doi.org/10.33012/navi.684>

## APPENDIX A | NOTATION

TABLE A1  
Description of Symbols and Operators

Sym. <sup>1</sup>	Description	Sym.	Description
<b>Symbols in sensor platform and motion model</b>			
${}^G\mathbf{p}_I$	Translation vector from $\{I\}$ to $\{G\}$	${}^G\mathbf{R}$	Rotation matrix from $\{I\}$ to $\{G\}$
${}^G\mathbf{v}_I$	Velocity of $\{I\}$ frame in $\{G\}$ frame	${}^L\mathbf{p}_I$	Translation vector from $\{I\}$ to $\{L\}$
${}^I\mathbf{q}_G$	Rotation from $\{G\}$ to $\{I\}$ in quaternions	${}^L\mathbf{q}_I$	Rotation from $\{I\}$ to $\{L\}$ in quaternions
$\mathbf{b}_a$	Accelerometer measurement bias	${}^L\mathbf{R}$	Rotation matrix from $\{I\}$ to $\{L\}$
$\mathbf{b}_g$	Gyroscope measurement bias	$\mathbf{a}_m$	Accelerator measurement
$\mathbf{a}_{GI}$	Acceleration vector of $\{I\}$ w.r.t. <sup>2</sup> $\{G\}$	$\mathbf{w}_m$	Gyroscope measurement
$\mathbf{w}_{GI}$	Angular velocity of $\{G\}$ w.r.t. $\{I\}$	$\Sigma_a$	Covariance of $\mathbf{n}_a$
$\mathbf{n}_a$	Accelerometer measurement noise	$\Sigma_g$	Covariance of $\mathbf{n}_g$
$\mathbf{n}_g$	Gyroscope measurement noise	$\Sigma_{wa}$	Covariance of $\mathbf{n}_{wa}$
$\mathbf{n}_{wa}$	Noise related to $\mathbf{b}_a$	$\Sigma_{wg}$	Covariance of $\mathbf{n}_{wg}$
$\mathbf{n}_{wg}$	Noise related to $\mathbf{b}_g$		
<b>Symbols in lidar processing</b>			
$\Pi_i$	$i$ -th plane	${}^L\mathbf{x}_i$	Shortest vector in scan plane from $\{L\}$ to $\Pi_i$
${}^G\pi_i$	Normal of plane $\Pi_i$ in $\{G\}$	$d_i$	Shortest distance from $\{G\}$ to plane $\Pi_i$
${}^L\mathbf{z}_L$	Normal of the lidar scan plane	${}^L\xi_i$	Unit direction vector of ${}^L\mathbf{x}_i$
${}^Ld_i$	${}^Ld_i = d_i - {}^G\pi_i^T ({}^G\mathbf{p}_I + {}^G\mathbf{R}^T\mathbf{p}_L)$	$\rho_i$	Length of ${}^L\mathbf{x}_i$
${}^L\mathbf{a}_i$	${}^L\mathbf{a}_i = {}^L\mathbf{R}^T {}^G\pi_i = [a_1^i, a_2^i, a_3^i]^T$	$\phi_i$	Angle made by ${}^L\mathbf{x}_i$ and the polar axis
<b>Symbols in EKF-based localization</b>			
$\mathbf{x}_k$	State vector at discrete time $k$	$\mathbf{v}_k$	Process noise at time $k$
$\hat{\mathbf{x}}_k^-$	Predicted state at discrete time $k$	$\mathbf{u}_k$	Measurement from IMU at time $k$
$\hat{\mathbf{x}}_k^+$	Estimated state at discrete time $k$	$\mathbf{P}_k^-$	Predicted covariance of estimated state by EKF at time $k$
$n$	Number of measurements	$\mathbf{Q}_k$	Covariance of $\mathbf{v}_k$
$\mathbf{P}_k$	Covariance of estimated state by EKF at time $k$	$\mathbf{K}_k$	Kalman gain at time $k$
$\mathbf{F}_k$	State transition matrix at time $k$	$\mathbf{G}_k$	Noise Jacobian matrix at time $k$

(Continued)

TABLE A1 (Continued)

Sym. <sup>1</sup>	Description	Sym.	Description
$\mathbf{y}_{i,k}$	$i$ -th measurement at time $k$	$\eta_{i,k}$	Noise of $i$ -th measurement at time $k$
$\mathbf{y}_k$	Measurements at time $k$	$\boldsymbol{\eta}_k$	Measurement noise at time $k$
$h^i(\mathbf{x}_k)$	Measurement function associated with $\mathbf{x}_k$ and $\mathbf{y}_{i,k}$	$\mathbf{H}_k^i$	Jacobian matrix of $h^i(\mathbf{x}_k)$
$h(\mathbf{x}_k)$	Measurement function associated with $\mathbf{x}_k$	$\mathbf{H}_k$	Jacobian matrix of $h(\mathbf{x}_k)$
<b>Symbols in residual analysis and fault detection</b>			
$\mathbf{r}_k$	Residual w.r.t. measurements at $k$	$f_\eta(\cdot)$	PDF of the measurement noise $\boldsymbol{\eta}$
$m_1$	Number of Gaussian modes in $\boldsymbol{\eta}$	$f_{\mathbf{v}}(\cdot)$	PDF of the system noise $\mathbf{v}$
$m_2$	Number of Gaussian modes in $\mathbf{v}$	$f_{\mathbf{r}_k}(\cdot)$	PDF of the residual $\mathbf{r}_k$
$\mathbf{T}_g$	Transformation of $\mathbf{r}_k$ based on GMM assumptions	$t_g$	Test statistic of $\mathbf{r}_k$ based on GMM assumptions
$\mathbf{T}_g^0$	Transformation of $\mathbf{r}_k$ based on Gaussian assumptions	$t_g^0$	Test statistic of $\mathbf{r}_k$ based on Gaussian assumptions
<b>Operators</b>			
$[\cdot]_\times$	Standard vector cross-product	$\times$	Standard vector cross-product
$\cdot$	Vector dot product	$*$	Convolution operator

<sup>1</sup> Sym: Symbol<sup>2</sup> w.r.t.: with respect to

## APPENDIX B | DISTRIBUTION OF THE RESIDUAL

Equation (23) shows that  $\mathbf{r}_k$  is a linear combination of independent random variables  $\mathbf{V}_{k-1}\mathbf{v}_{k-1}$ ,  $\mathbf{N}_{k-1}\mathbf{v}_{k-2}$ , and  $\boldsymbol{\eta}_k$ . Therefore, the PDF of  $\mathbf{r}_k$  is the convolution of the PDF of these random variables, which can be written as follows:

$$f_{\mathbf{r}_k} = f_{\mathbf{V}_{k-1}\mathbf{v}} * f_{\mathbf{N}_{k-1}\mathbf{v}} * f_{\boldsymbol{\eta}} \quad (\text{B1})$$

where  $*$  is the convolution operator. Because  $\mathbf{v}_{k-1}$  and  $\mathbf{v}_{k-2}$  are assumed to be i.i.d., we drop the time index in the above equation. The characteristic functions of  $\mathbf{r}_k$ ,  $\boldsymbol{\eta}$ , and  $\mathbf{v}$  are given as follows:

$$\varphi_{\mathbf{r}_k}(\omega) = \int_{-\infty}^{\infty} f_{\mathbf{r}_k}(\mathbf{x}) e^{i\omega^T \mathbf{x}} d\mathbf{x} \quad (\text{B2a})$$

$$\varphi_{\boldsymbol{\eta}}(\omega) = \int_{-\infty}^{\infty} f_{\boldsymbol{\eta}}(\mathbf{x}) e^{i\omega^T \mathbf{x}} d\mathbf{x} = \sum_{j=1}^{m_1} p_j^\eta \int_{-\infty}^{\infty} \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_j^\eta, \boldsymbol{\pi}_j^\eta) e^{i\omega^T \mathbf{x}} d\mathbf{x} \quad (\text{B2b})$$

$$= \sum_{j=1}^{m_1} p_j^\eta \varphi_{\mathcal{N}}(\omega) = \sum_{j=1}^{m_1} p_j^\eta e^{i\omega^T \boldsymbol{\mu}_j^\eta - \frac{1}{2} \omega^T \boldsymbol{\pi}_j^\eta \omega}$$

$$\varphi_{\mathbf{v}}(\omega) = \int_{-\infty}^{\infty} f_{\mathbf{v}}(\mathbf{x}) e^{i\omega^T \mathbf{x}} d\mathbf{x} = \sum_{j=1}^{m_2} p_j^v e^{i\omega^T \boldsymbol{\mu}_j^v - \frac{1}{2} \omega^T \boldsymbol{\pi}_j^v \omega} \quad (\text{B2c})$$

where  $\varphi_{\mathcal{N}}(\omega)$  is the characteristic function of a multi-normal variable. Let  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}$  be the linear transformation of the random vector  $\mathbf{x}$ , where  $\mathbf{A}$  is a constant matrix and  $\mathbf{b}$  is a constant vector. The characteristic function of  $\mathbf{y}$  is given by the following:

$$\varphi_{\mathbf{y}}(\omega) = e^{i\omega^T \mathbf{b}} \varphi_{\mathbf{x}}(\mathbf{A}^T \omega) \quad (\text{B3})$$

Therefore, we have the following relationship:

$$\varphi_{\mathbf{N}_{k-1}\mathbf{v}}(\omega) = \varphi_{\mathbf{v}}\left(\mathbf{N}_{k-1}^T\omega\right) = \sum_{j=1}^{m_2} p_j^v e^{i\omega^T \mathbf{N}_{k-1} \mu_j^v - \frac{1}{2}\omega^T \mathbf{N}_{k-1} \pi_j^v \mathbf{N}_{k-1}^T \omega} \quad (\text{B4a})$$

$$\varphi_{\mathbf{V}_{k-1}\mathbf{v}}(\omega) = \varphi_{\mathbf{v}}\left(\mathbf{V}_{k-1}^T\omega\right) = \sum_{j=1}^{m_2} p_j^v e^{i\omega^T \mathbf{V}_{k-1} \mu_j^v - \frac{1}{2}\omega^T \mathbf{V}_{k-1} \pi_j^v \mathbf{V}_{k-1}^T \omega} \quad (\text{B4b})$$

According to the convolution theorem, we have the following:

$$\varphi_{\mathbf{r}_k}(\omega) = \varphi_{\mathbf{v}}\left(\mathbf{V}_{k-1}^T\omega\right) \cdot \varphi_{\mathbf{v}}\left(\mathbf{N}_{k-1}^T\omega\right) \cdot \varphi_{\eta}(\omega) \quad (\text{B5})$$

where  $\cdot$  indicates point-wise multiplication. Substituting Equations (B2b) and (B4) into Equation (B5), we can obtain the following:

$$\varphi_{\mathbf{r}_k}(\omega) = \sum_{a=1}^{m_2} \sum_{b=1}^{m_2} \sum_{c=1}^{m_1} p_{abc} e^{i\omega^T \mu_{abc} - \frac{1}{2}\omega^T \pi_{abc} \omega} \quad (\text{B6})$$

where:

$$p_{abc} = p_a^v p_b^v p_c^\eta \quad (\text{B7a})$$

$$\mu_{abc} = \mathbf{V}_{k-1} \mu_a^v + \mathbf{N}_{k-1} \mu_b^v + \mu_c^\eta \quad (\text{B7b})$$

$$\pi_{abc} = \mathbf{V}_{k-1} \pi_a^v \mathbf{V}_{k-1}^T + \mathbf{N}_{k-1} \pi_b^v \mathbf{N}_{k-1}^T + \pi_c^\eta \quad (\text{B7c})$$

Performing the inverse operation shown in Equation (B2b) yields the following:

$$\varphi_{\mathbf{r}_k}(\omega) = \sum_{a=1}^{m_2} \sum_{b=1}^{m_2} \sum_{c=1}^{m_1} p_{abc} \int_{-\infty}^{\infty} \mathcal{N}(\mathbf{x} | \mu_{abc}, \pi_{abc}) e^{i\omega^T \mathbf{x}} d\mathbf{x} \quad (\text{B8})$$

Therefore, we obtain the following:

$$f_{\mathbf{r}_k}(\mathbf{x}) = \sum_{a=1}^{m_2} \sum_{b=1}^{m_2} \sum_{c=1}^{m_1} p_{abc} \mathcal{N}(\mathbf{x} | \mu_{abc}, \pi_{abc}) d\mathbf{x} \quad (\text{B9})$$

which indicates that  $\mathbf{r}_k$  has a multivariate GMM distribution.

## APPENDIX C | SIMILARITY TO A STANDARD MVN DISTRIBUTION

To evaluate the similarity between the transformed distribution and the standard MVN distribution, we calculate the discretized Jensen—Shannon divergence between the two distributions (MacKay, 2003). Let  $P(x)$  and  $Q(x)$  be the probability mass functions of distributions  $P$  and  $Q$ , respectively. The discretized Jensen—Shannon divergence ( $D_{JS}$ ) between  $P$  and  $Q$  defined on an identical sample space is given as follows:

$$\begin{aligned} D_{JS}(P || Q) &= \frac{1}{2} D_{KL}(P || M) + \frac{1}{2} D_{KL}(Q || M) \\ D_{KL}(P || Q) &= \sum_{x \in S} P(x) \log \left( \frac{P(x)}{Q(x)} \right) \end{aligned} \quad (\text{C1})$$



where  $M = \frac{1}{2}(P+Q)$  and  $D_{KL}$  is the Kullback–Leibler divergence (relative entropy). The minimum value (0) of  $D_{JS}$  indicates that the two distributions are the same, whereas the maximum value (1) indicates that the two distributions are completely different (MacKay, 2003; Manning & Schutze, 1999). In our work,  $P$  refers to the transformed distribution from a bivariate GMM (BGMM) and  $Q$  refers to the standard MVN distribution. To comprehensively evaluate the similarity between the transformed distribution and the standard MVN distribution, we first consider the 2D case and then extend our findings to higher-dimensional cases.

### (1) Transformation of 2D BGMM

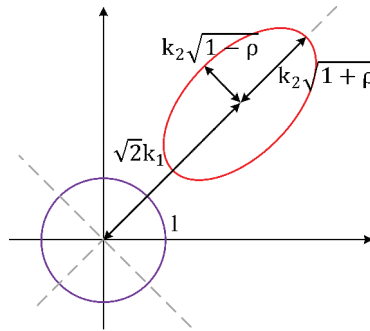
The PDF of a 2D BGMM can be formalized as follows:

$$f(\mathbf{x}) = p_1 f_N(\mathbf{x}; \mu_1, \Sigma_1) + (1 - p_1) f_N(\mathbf{x}; \mu_2, \Sigma_2) \quad (\text{C2})$$

where  $f_N(\mathbf{x}; \mu_1, \Sigma_1)$  and  $f_N(\mathbf{x}; \mu_2, \Sigma_2)$  are the PDFs of the first and second Gaussian components,  $\mu_1$  and  $\mu_2$  are the corresponding mean values,  $\Sigma_1$  and  $\Sigma_2$  are the corresponding standard deviations, and  $p_1$  and  $1 - p_1$  are the mixing weights of the two Gaussian components. Equation (C2) is determined by thirteen parameters, i.e.,  $p_1$ , two elements in  $\mu_1$ , two elements in  $\mu_2$ , four elements in  $\Sigma_1$ , and four elements in  $\Sigma_2$ , which span a thirteen-dimensional parameter space. It is difficult to determine the parameter combinations in this thirteen-dimensional parameter space such that the transformed distribution resembles the standard MVN distribution. Therefore, we use a reparameterization method to reduce the dimension of the parameter space (Li & Schwartzman, 2018). Specifically, we use four parameters,  $k_1$ ,  $k_2$ ,  $\rho$ , and  $p_1$ , to define a 2D BGMM, as shown in Table C1. Figure C1 shows 50% density contours of the reparameterized BGMM with  $\rho = 0.5$  and  $p_1 = 0.5$ . As can be seen,  $p_1$  determines the

**TABLE C1**  
Reparameterization of the 2D BGMM

	Mixture Weight	Mean	Variance
Component 1	$p_1$	$\mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
Component 2	$1 - p_1$	$\mu_2 = k_1 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$\Sigma_2 = k_2^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$



**FIGURE C1** 50% density contours of the reparameterized BGMM with  $\rho = 0.5$  and  $p_1 = 0.5$ . The purple ellipse represents the density contour of component 1, whereas the red ellipse represents the density contour of component 2.



weight of the two Gaussian components,  $k_1$  determines the distances between the centers of the two ellipses,  $k_2$  controls the scale of the ellipse of the second Gaussian component, and  $\rho$  determines the ellipse eccentricity of the second Gaussian component.

Based on the reparameterized BGMM, we conduct a Monte Carlo simulation to evaluate the similarity between the transformed distribution from a BGMM and the standard MVN distribution. Specifically, for each setting of  $(p_1, k_1, k_2, \rho)$ , we randomly generate  $N$  samples  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$  from the 2D BGMM, where  $N$  is set to  $10^5$  in our experiments. Then, we apply the proposed transformation method to each sample:

$$\mathbf{T}_{i,g} = \Sigma^{-1/2}(\mathbf{x}_i - \mu) \quad (\text{C3})$$

where  $\mu$  is the total mean and  $\Sigma^{-1/2}$  is the principal square root matrix of the total covariance ( $\Sigma$ ) of the BGMM. According to the law of total covariance (Weiss et al., 2006),  $\mu$  and  $\Sigma$  are given by the following:

$$\mu = p_1 \mu_1 + (1 - p_1) \mu_2 \quad (\text{C4a})$$

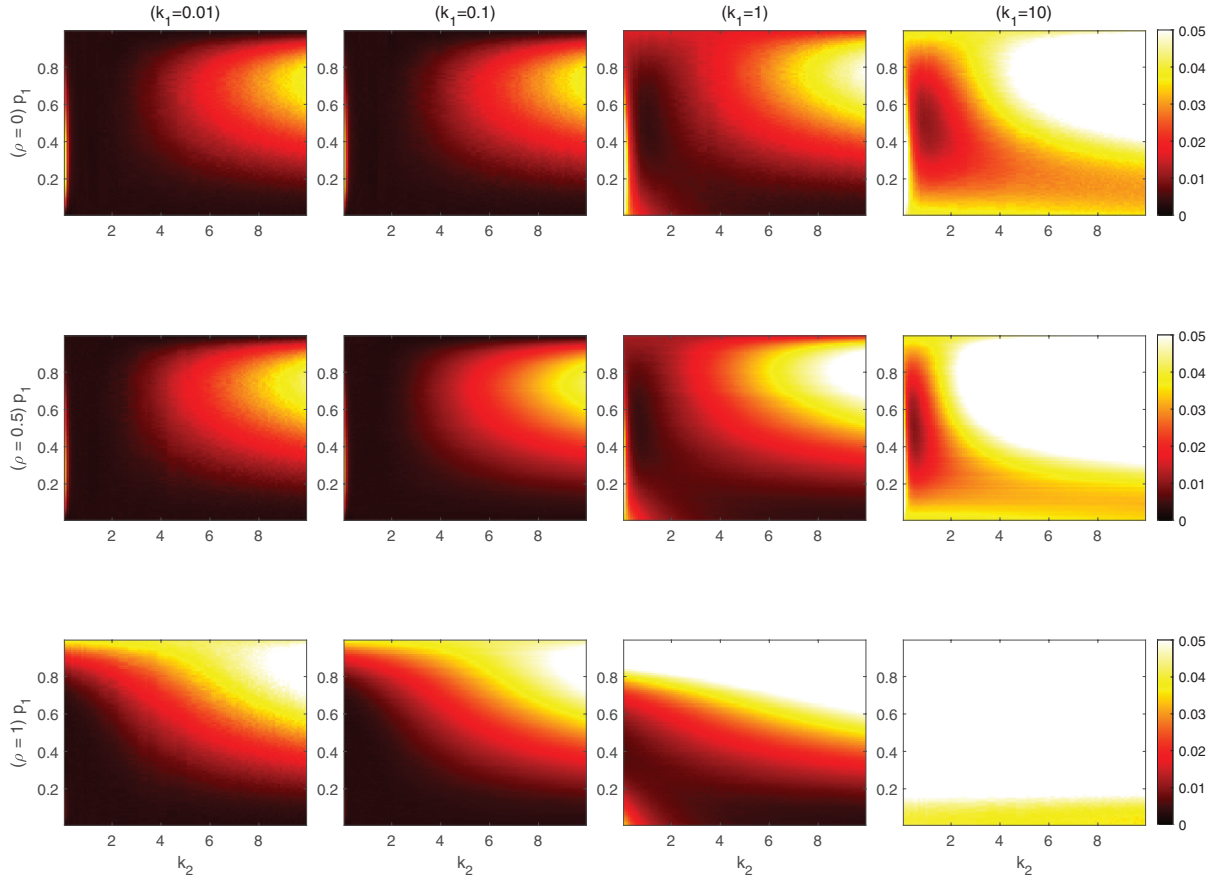
$$\Sigma = p_1 \Sigma_1 + (1 - p_1) \Sigma_2 + \sum_{j=1}^2 (\mu - \mu_j)(\mu - \mu_j)^T \quad (\text{C4b})$$

The transformed sample distribution is denoted by  $\{\mathbf{T}_{1,g}, \mathbf{T}_{2,g}, \dots, \mathbf{T}_{N,g}\}$ . Then, we randomly generate  $N$  samples  $\{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$  from a standard 2D MVN distribution. Let  $P = \{\mathbf{T}_{1,g}, \mathbf{T}_{2,g}, \dots, \mathbf{T}_{N,g}\}$  and  $Q = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ . We can employ Equation (C1) to calculate the overall similarity between  $P$  and  $Q$ . Because  $D_{JS}$  is defined for a one-dimensional distribution, we will show the similarity between  $P$  and  $Q$  in each dimension in the following paragraphs.

Figure C2 shows the  $D_{JS}$  between the first dimension of the transformed distribution and the standard 2D MVN distribution. The results are presented in the form of heatmaps with  $\rho \in \{0, 0.5, 1\}$  and  $k_1 \in \{0.01, 0.1, 1, 10\}$ , where  $p_1$  and  $k_2$  vary continuously. Interestingly, the lowest  $D_{JS}$  is centered in the region of  $k_2 \in [1, 3]$  (except for the case of  $\rho = 1, k_1 = 10$ ), where the two Gaussian components have a similar covariance. In this region, the total covariance in Equation (C4b) is similar to the covariance of each component, thereby accurately characterizing the uncertainty of the BGMM. Similar results are shown in Figure C3, which presents the  $D_{JS}$  between the second dimension of the transformed distribution and the standard 2D MVN distribution.

## (2) Transformation of Higher-Dimensional BGMM

We further examine the similarity between the transformed distribution from a  $K$ -dimensional ( $K > 2$ ) BGMM and the standard MVN distribution. Because it is impossible to conduct experiments with all values of  $K$  (which are infinite), we choose  $K$  to be 27, which is the average number of 2D lidar measurements in the simulated experiments in Section 4. We reparameterize the 27-dimensional (27D) BGMM using three parameters,  $p_1$ ,  $k_1$ , and  $k_2$ , as shown in Table C2. Similar to the 2D BGMM experiments, we conduct a Monte Carlo simulation to obtain the transformed sample distribution for different settings of  $p_1$ ,  $k_1$ , and  $k_2$ . To follow the analysis procedure utilized in the 2D BGMM experiments, we must calculate the  $D_{JS}$  between the transformed distribution and the standard MVN distribution for each dimension, yielding 27 heatmaps. As can be anticipated, this process



**FIGURE C2**  $D_{JS}$  between the first dimension of the transformed distribution and the standard MVN distribution

**TABLE C2**  
Reparameterization of the 27D BGMM

	Mixture Weight	Mean	Variance
Component 1	$p_1$	$\mu_1 = [0 \dots 0]_{1 \times 27}^T$	$\Sigma_1 = \mathbf{I}_{27}^{-1}$
Component 2	$1 - p_1$	$\mu_2 = k_1 [1 \dots 1]_{1 \times 27}^T$	$\Sigma_2 = k_2^2 \mathbf{I}_{27}$

<sup>1</sup>  $\mathbf{I}_{27}$  is a  $27 \times 27$  identity matrix.

would complicate the analysis. Therefore, we use an alternative approach to evaluate the similarity between the transformed distribution and the standard MVN distribution.

Let the transform sample distribution be  $P = \{\mathbf{T}_{1,g}, \mathbf{T}_{2,g}, \dots, \mathbf{T}_{N,g}\}$ . For each sample  $\mathbf{T}_{i,g}$  in  $P$ , we calculate its inner product with itself, i.e.,  $\mathbf{T}_{i,g}^T \mathbf{T}_{i,g}$ . Then, we obtain the distribution  $P_1 = \{\mathbf{T}_{1,g}^T \mathbf{T}_{1,g}, \mathbf{T}_{2,g}^T \mathbf{T}_{2,g}, \dots, \mathbf{T}_{N,g}^T \mathbf{T}_{N,g}\}$ . If the distribution  $P$  approaches the standard MVN distribution, the distribution  $P_1$  should approach a chi-squared distribution with 27 DOFs. Therefore, we can evaluate the similarity between the distribution  $P_1$  and the chi-squared distribution with 27 DOFs. Figure C4 shows the  $D_{JS}$  between the distribution  $P_1$  and the chi-squared distribution with 27 DOFs in the form of heatmaps with  $k_1 \in \{0.01, 0.1, 1, 10\}$ , where  $p_1$  and  $k_2$  vary continuously. Similar to the trends in the 2D BGMM experiments,

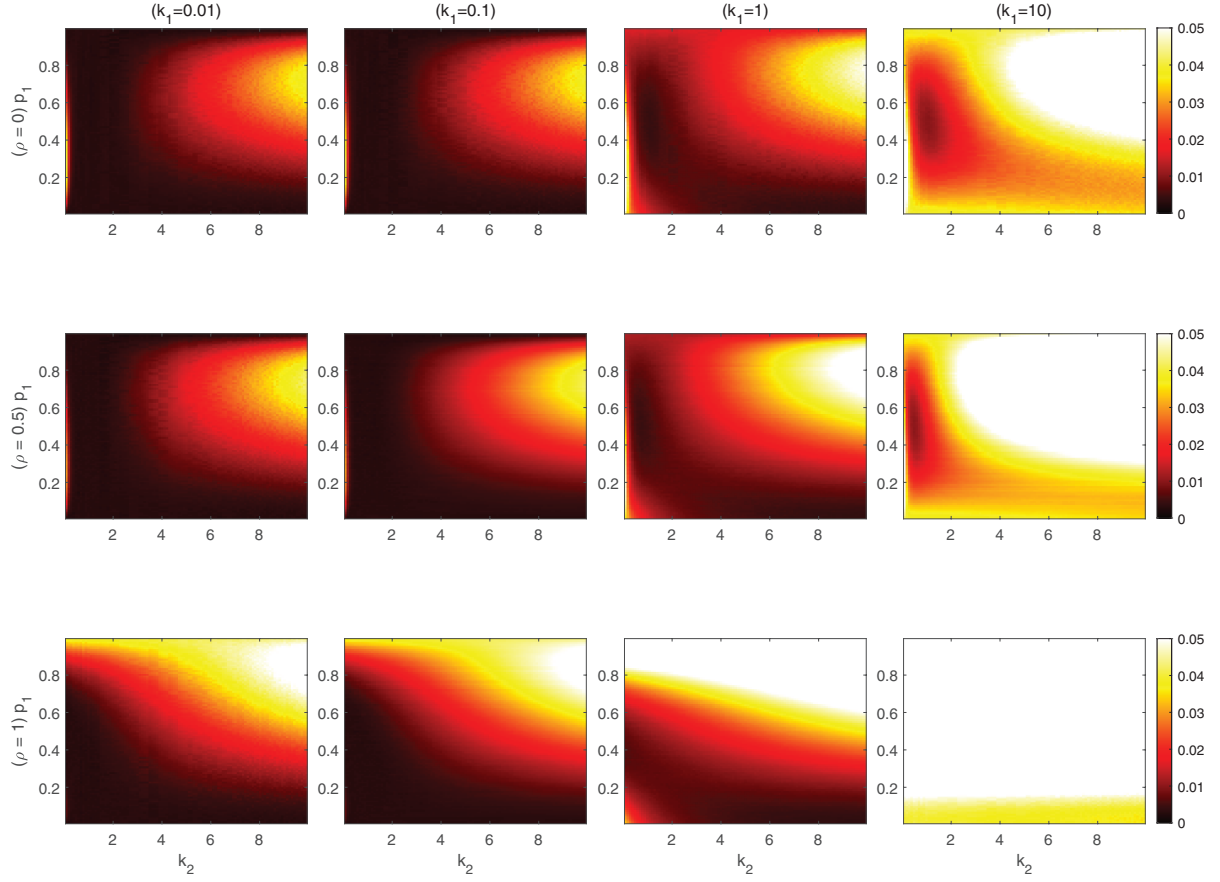


FIGURE C3  $D_{JS}$  between the second dimension of the transformed distribution and the standard MVN distribution

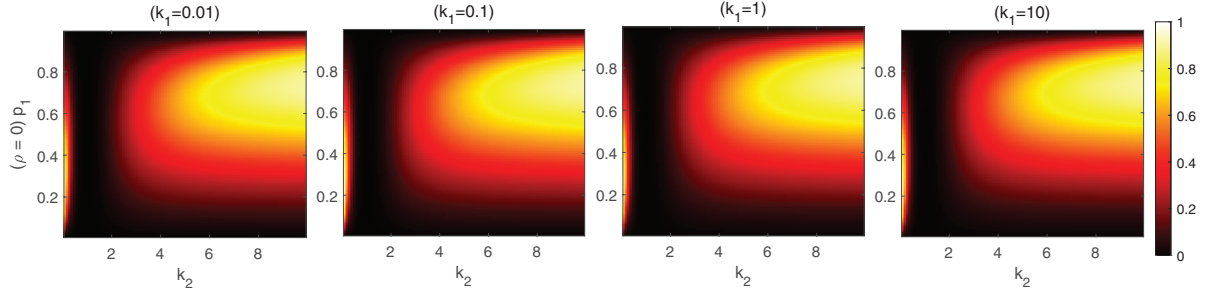


FIGURE C4  $D_{JS}$  between the distribution  $P_1$  and the chi-squared distribution with 27 DOFs

the lowest  $D_{JS}$  is centered in the region of  $k_2 \in [1, 3]$ . Therefore, the  $P_1$  distribution is similar to the chi-squared distribution with 27 DOFs in this region, suggesting that the transformed distribution from a 27D BGMM is similar to the 27D MVN distribution.

Through the analysis in the 2D and 27D BGMM experiments, we find that the similarity assumption holds when each Gaussian component in the GMM-distributed residual has a small difference in the covariance. This constraint is satisfied in the lidar/IMU integrated system because the lidar measurement error is slightly heavy-tailed (Toschi et al., 2015), indicating that the difference in the covariance between the two Gaussian components is insignificant.

## APPENDIX D | LIDAR PROCESSING

### Appendix D.1 | Least-Norm Solution for Lidar Measurement Models

Following the notation in Figure 3(b) and corresponding definitions in Section 3.2, we assume that the extrinsic calibration parameters  $({}^I\mathbf{p}_L, {}^I\mathbf{R})$  and the vehicle pose  $({}^G\mathbf{p}_I, {}^G\mathbf{R})$  are known, that the plane  $\Pi_i$  associated with the current lidar scan plan is found based on the vehicle pose  $({}^G\mathbf{p}_I, {}^G\mathbf{R})$  by a matching algorithm, and that  $d_i$  and  ${}^G\pi_i$  are known. We have the following constraints:

(1) Plane constraint (Zhao & Farrell, 2013)

${}^L\mathbf{x}_i$  is a laser beam; thus, it must be in the lidar scan plane (Zhao & Farrell, 2013). Let  ${}^L\mathbf{z}_L = [0\ 0\ 1]^T$  be the normal of the lidar scan plane; then, we have the following:

$${}^L\mathbf{z}_L^T {}^L\mathbf{x}_i = 0 \quad (\text{D1})$$

(2) Distance constraint (Hesch et al., 2010)

As can be seen in Figure 3(b), the vector from  $\{G\}$  to point  $M$  can be obtained by either of the following:

$${}^G\mathbf{p}_I + {}^G\mathbf{R}({}^I\mathbf{p}_L + {}^I\mathbf{R}{}^L\mathbf{x}_i) \quad (\text{D2})$$

or:

$$d_i {}^G\pi_i + {}^G\mathbf{t}_i \quad (\text{D3})$$

both of which are equivalent. Therefore, we have the following:

$${}^G\mathbf{p}_I + {}^G\mathbf{R}({}^I\mathbf{p}_L + {}^I\mathbf{R}{}^L\mathbf{x}_i) = d_i {}^G\pi_i + {}^G\mathbf{t}_i \quad (\text{D4})$$

By projecting Equation (D4) onto  ${}^G\pi_i^T$ , we obtain the following:

$${}^G\pi_i^T ({}^G\mathbf{p}_I + {}^G\mathbf{R}({}^I\mathbf{p}_L + {}^I\mathbf{R}{}^L\mathbf{x}_i)) = {}^G\pi_i^T d_i {}^G\pi_i + {}^G\pi_i^T {}^G\mathbf{t}_i \quad (\text{D5})$$

Because  ${}^G\pi_i \perp {}^G\mathbf{t}_i$  and  ${}^G\pi_i$  is a unit vector, we can write the following expression:

$${}^G\pi_i^T {}^G\mathbf{R}{}^L\mathbf{x}_i = d_i - {}^G\pi_i^T ({}^G\mathbf{p}_I + {}^G\mathbf{R}{}^I\mathbf{p}_L) \quad (\text{D6})$$

By combining Equations (D1) and (D6), we have the following:

$$\mathbf{A}^L \mathbf{x}_i = \mathbf{b} \quad (\text{D7})$$

where:

$$\mathbf{A} = \begin{bmatrix} {}^L\mathbf{z}_L^T \\ {}^G\pi_i^T {}^G\mathbf{R} \end{bmatrix}_{2 \times 3} \quad \mathbf{b} = \begin{bmatrix} 0 \\ {}^Ld_i \end{bmatrix}_{2 \times 1} \quad (\text{D8})$$

$${}^Ld_i = d_i - {}^G\pi_i^T ({}^G\mathbf{p}_I + {}^G\mathbf{R}{}^I\mathbf{p}_L) \quad (\text{D9})$$

Because  ${}^L\mathbf{x}_i = [\rho_i \cos \phi_i \ \rho_i \sin \phi_i \ 0]^T$  is the shortest vector in the lidar scan plane from the origin of  $\{L\}$  to  $\Pi_i$ , we can solve the following optimization problem to estimate the values of  $\rho_i$  and  $\phi_i$ :

$$\begin{aligned} {}^L\mathbf{x}_i^* &= \arg \min_{{}^L\mathbf{x}_i} \| {}^L\mathbf{x}_i \|^2 \\ \text{s.t. } \mathbf{A} {}^L\mathbf{x}_i &= \mathbf{b} \end{aligned} \quad (\text{D10})$$

where  ${}^L\mathbf{x}_i^*$  is the least-norm solution of the linear equation  $\mathbf{A} {}^L\mathbf{x}_i = \mathbf{b}$  (Cline & Plemmons, 1976). The solution is as follows:

$${}^L\mathbf{x}_i^* = \mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1} \mathbf{b} \quad (\text{D11})$$

We take the following:

$${}^L\mathbf{a}_i = {}^L\mathbf{R}^G \pi_i = [a_1^i, a_2^i, a_3^i]^T \quad (\text{D12})$$

which is a unit vector. We can rewrite Equation (D8) as follows:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 \\ a_1^i & a_2^i & a_3^i \end{bmatrix} \quad (\text{D13})$$

Therefore, we obtain the following:

$$\mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1} = \frac{\begin{bmatrix} -a_1^i a_3^i & a_1^i \\ -a_2^i a_3^i & a_2^i \\ 1 - a_3^{i2} & 0 \end{bmatrix}}{1 - a_3^{i2}} \quad (\text{D14})$$

Because  ${}^L\mathbf{a}_i = [a_1^i, a_2^i, a_3^i]^T$  is a unit vector,  $a_1^{i2} + a_2^{i2} + a_3^{i2} = 1$ . Thus, we have the following relationship:

$${}^L\mathbf{x}_i^* = \frac{1}{a_1^{i2} + a_2^{i2}} \begin{bmatrix} -a_1^i a_3^i & a_1^i \\ -a_2^i a_3^i & a_2^i \\ 1 - a_3^{i2} & 0 \end{bmatrix} \begin{bmatrix} 0 \\ {}^L d_i \end{bmatrix} = \begin{bmatrix} a_1^i \\ a_2^i \\ 0 \end{bmatrix} \frac{{}^L d_i}{a_1^{i2} + a_2^{i2}} \quad (\text{D15})$$

$$\rho_i^* \begin{bmatrix} \cos \phi_i^* \\ \sin \phi_i^* \\ 0 \end{bmatrix} = \begin{bmatrix} a_1^i \\ a_2^i \\ 0 \end{bmatrix} \frac{{}^L d_i}{a_1^{i2} + a_2^{i2}} = \begin{bmatrix} \frac{a_1}{\sqrt{a_1^{i2} + a_2^{i2}}} \\ \frac{a_2}{\sqrt{a_1^{i2} + a_2^{i2}}} \\ 0 \end{bmatrix} \frac{{}^L d_i}{\sqrt{a_1^{i2} + a_2^{i2}}} \quad (\text{D16})$$

Therefore, the least-norm solution can be obtained as follows:

$$\rho_i^* = \frac{|{}^L d_i|}{\sqrt{a_1^{i2} + a_2^{i2}}} \quad (\text{D17a})$$

$$\phi_i^* = \arctan \left( \text{sgn}({}^L d_i) \frac{a_2^i}{a_1^i} \right) \quad (\text{D17b})$$

## Appendix D.2 | Line Extraction and Feature Association

### (1) Extracting line segments from 2D lidar points

To extract line segments from raw 2D lidar measurements, we adopt the method of Pfister et al. (2003), which is summarized as follows. A line segment  $L_i$  is represented by three parameters,  $\rho_i$ ,  $\phi_i$ , and  $S_i$ , where  $(\rho_i, \phi_i)$  gives the ranging and bearing parameters associated with the vector perpendicular to  $L_i$  in the polar coordinate system and  $S_i$  is the distance from the line perpendicular to the line segment's center, as shown in Figure 3(a). For the  $j$ -th 2D lidar point, we denote the distance from the line segment as  $\delta_j^P$  and its distance from the center of the line segment projected on the line segment as  $\delta_j^S$ . The best-fitted line segment is found by minimizing  $\delta_j^P$  and  $\delta_j^S$  of all lidar points by the maximum likelihood estimation.

### (2) Associating line segments with planes

As discussed in Section 3.2, the laser scan plane intersects with several planes, and these intersections are the line segments. The parameters of these line segments can be predicted according to the current pose of the vehicle, as shown in Equation (44). Therefore, the association of line segments and planes can be achieved by examining the closeness between the predicted line segment and the extracted line segment. For the  $r$ -th plane  $\Pi_r$ , the line segment predicted by Equation (44) is parameterized by  $\hat{\mathbf{y}}_r = [\phi_r, \rho_r]^T$ . The extracted line segment can also be parameterized by  $\mathbf{y}_i = [\phi_i, \rho_i]^T$  in the polar coordinate system. The Mahalanobis distance from  $\hat{\mathbf{y}}_r$  to  $\mathbf{y}_i$  is given by  $\|\hat{\mathbf{y}}_r - \mathbf{y}_i\|_{\mathbf{P}_{y_i}}$ , where  $\mathbf{P}_{y_i}$  is the covariance matrix of the extracted line segment. The best-fit plane can be found by minimizing  $\|\hat{\mathbf{y}}_r - \mathbf{y}_i\|_{\mathbf{P}_{y_i}}$ . Assuming that the extracted line segment has zero-mean Gaussian noise, we find that the Mahalanobis distance follows a chi-squared distribution with two DOFs. If the  $r$ -th plane fails to satisfy  $\|\hat{\mathbf{y}}_r - \mathbf{y}_i\|_{\mathbf{P}_{y_i}} < \delta_a$ , the  $r$ -th plane is excluded from the association process. If none of the planes satisfy this condition, the extracted line segment  $\mathbf{y}_i$  is excluded when the final set of measurements is constructed. The threshold  $\delta_a$  is determined by a given significance level, which is set as 0.05 (Pfister et al., 2003).

## APPENDIX E | LINEARIZATION OF AN EKF-BASED LIDAR/IMU INTEGRATED SYSTEM

### Appendix E.1 | Derivative of the Rotation Matrix

Let  $\mathbf{g} = [g_x, g_y, g_z]^T$ ,  $\mathbf{q} = [q_w, \mathbf{q}_v^T]^T$ , and  $\mathbf{q}_v = [q_x, q_y, q_z]^T$ ; then, the rotation matrix corresponding to  $\mathbf{q}$  is as follows:

$$\mathbf{R}\{\mathbf{q}\} = \begin{bmatrix} 1 - 2(q_y^2 + q_z^2) & 2(q_x q_y - q_w q_z) & 2(q_x q_z + q_w q_y) \\ 2(q_x q_y + q_w q_z) & 1 - 2(q_x^2 + q_z^2) & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z - q_w q_y) & 2(q_y q_z + q_w q_x) & 1 - 2(q_x^2 + q_y^2) \end{bmatrix} \quad (\text{E1})$$

The derivative of  $\mathbf{R}^T\{\mathbf{q}\}\mathbf{g}$  with respect to  $\mathbf{q}$  can be solved by performing an element-wise derivation:

$$\frac{\partial \mathbf{R}^T\{\mathbf{q}\}\mathbf{g}}{\partial \mathbf{q}} = 2 \left[ \mathbf{u}_g \quad \left[ \mathbf{u}_g + q_w \mathbf{g} \right]_{\times} + (\mathbf{q}_v \cdot \mathbf{g}) \mathbf{I}_3 - \mathbf{g} \mathbf{q}_v^T \right] \quad (\text{E2})$$

where:

$$\mathbf{u}_g = \mathbf{g} \times \mathbf{q}_v = \left[ \mathbf{g} \right]_{\times} \mathbf{q}_v \quad (\text{E3})$$

Because  $\mathbf{R}\{\mathbf{q}\} = \mathbf{R}^T\{\mathbf{q}^*\}$ , where  $\mathbf{q}^*$  is the conjugate of  $\mathbf{q}$ , we can obtain the derivative of  $\mathbf{R}\{\mathbf{q}\}\mathbf{g}$  with respect to  $\mathbf{q}$ :

$$\frac{\partial \mathbf{R}\{\mathbf{q}\}\mathbf{g}}{\partial \mathbf{q}} = \frac{\partial \mathbf{R}^T\{\mathbf{q}^*\}\mathbf{g}}{\partial \mathbf{q}} = \frac{\partial \mathbf{R}^T\{\mathbf{q}^*\}\mathbf{g}}{\partial \mathbf{q}^*} \frac{\partial \mathbf{q}^*}{\partial \mathbf{q}} \quad (\text{E4})$$

where:

$$\frac{\partial \mathbf{q}^*}{\partial \mathbf{q}} = \begin{bmatrix} \frac{\partial q^*}{\partial q_w} & \frac{\partial q^*}{\partial q_x} & \frac{\partial q^*}{\partial q_y} & \frac{\partial q^*}{\partial q_z} \end{bmatrix} = \text{diag}[1, -1, -1, -1] \quad (\text{E5})$$

Therefore, we have the following:

$$\frac{\partial \mathbf{R}\{\mathbf{q}\}\mathbf{g}}{\partial \mathbf{q}} = \frac{\partial \mathbf{R}^T\{\mathbf{q}^*\}\mathbf{g}}{\partial \mathbf{q}^*} \text{diag}[1, -1, -1, -1] \quad (\text{E6})$$

For convenience, we introduce two symbols:

$$J_q(\mathbf{q}, \mathbf{g}) = \frac{\partial \mathbf{R}^T\{\mathbf{q}\}\mathbf{g}}{\partial \mathbf{q}} \quad (\text{E7a})$$

$$J_q^*(\mathbf{q}, \mathbf{g}) = \frac{\partial \mathbf{R}\{\mathbf{q}\}\mathbf{g}}{\partial \mathbf{q}} \quad (\text{E7b})$$

## Appendix E.2 | Linearization of the Rotation Part of the State Propagation Equation

The rotation part in the state propagation equation in Equation (41) is given by the following:

$${}^I \mathbf{q}_{G,k+1} = g\left({}^I \mathbf{q}_{G,k}, \mathbf{w}_{GI,k}\right) = \exp\left(\frac{\Delta t}{2} \Omega[\mathbf{w}_{GI,k}]\right) {}^I \mathbf{q}_{G,k} \quad (\text{E8a})$$

$$\mathbf{w}_{GI,k} = \mathbf{w}_{m,k} - \mathbf{b}_{g,k} - \mathbf{n}_g \quad (\text{E8b})$$

where  $\exp(\cdot)$  is a matrix function defined by the following:

$$\exp(\mathbf{M}) = \sum_{i=0}^{\infty} \frac{\mathbf{M}^i}{i!} \quad (\text{E9})$$

Therefore, Equation (E8a) can be expanded as follows:

$$g\left({}^I \mathbf{q}_{G,k}, \mathbf{w}_{GI,k}\right) = \left( \mathbf{I}_4 + \frac{1}{2} \Omega[\mathbf{w}_{GI,k}] \Delta t + \mathcal{O}\left(\left(\Omega[\mathbf{w}_{GI,k}]\right)^2\right) \right) {}^I \mathbf{q}_{G,k} \quad (\text{E10})$$



By keeping the first-order term and neglecting the higher-order terms in the above equations, we have the following:

$$g({}^I \mathbf{q}_{G,k}, \mathbf{w}_{GI,k}) = \left( \mathbf{I}_4 + \frac{1}{2} \Omega[\mathbf{w}_{GI,k}] \Delta t \right) {}^I \mathbf{q}_{G,k} = \begin{bmatrix} \mathbf{I}_3 - \frac{1}{2} \Delta t [\mathbf{w}_{GI,k}]_{\times} & \frac{1}{2} \Delta t \mathbf{w}_{GI,k} \\ -\frac{1}{2} \Delta t \mathbf{w}_{GI,k}^T & 1 \end{bmatrix} {}^I \mathbf{q}_{G,k} \quad (\text{E11})$$

Let  ${}^I \hat{\mathbf{q}}_{G,k} = [\hat{q}_{w,k}, \hat{q}_{x,k}, \hat{q}_{y,k}, \hat{q}_{z,k}]$  be the estimated vehicle orientation in terms of quaternions at time  $k$ ; then, the Jacobian matrix of  $g({}^I \mathbf{q}_{G,k}, \mathbf{w}_{GI,k})$  with respect to  ${}^I \mathbf{q}_{G,k}$  can be obtained by taking the first-order Taylor expansion at the point  $({}^I \hat{\mathbf{q}}_{G,k}, \mathbf{w}_{m,k})$ :

$$\frac{\partial g({}^I \hat{\mathbf{q}}_{G,k}, \mathbf{w}_{m,k})}{\partial {}^I \mathbf{q}_{G,k}} = \begin{bmatrix} \mathbf{I}_3 - \frac{1}{2} \Delta t [\mathbf{w}_{GI,k}]_{\times} & \frac{1}{2} \Delta t \mathbf{w}_{GI,k} \\ -\frac{1}{2} \Delta t \mathbf{w}_{GI,k}^T & 1 \end{bmatrix} \quad (\text{E12})$$

Similarly, the Jacobian matrix of  $g({}^I \mathbf{q}_{G,k}, \mathbf{w}_{GI,k})$  evaluated at  $({}^I \hat{\mathbf{q}}_{G,k}, \mathbf{w}_{m,k})$  with respect to  $\mathbf{w}_{GI,k}$  can be obtained as follows:

$$\begin{aligned} \frac{\partial g({}^I \hat{\mathbf{q}}_{G,k}, \mathbf{w}_{m,k})}{\partial \mathbf{w}_{GI,k}} &= \left[ \frac{\partial g({}^I \hat{\mathbf{q}}_{G,k}, \mathbf{w}_{m,k})}{\partial w_{x,k}}, \frac{\partial g({}^I \hat{\mathbf{q}}_{G,k}, \mathbf{w}_{m,k})}{\partial w_{y,k}}, \frac{\partial g({}^I \hat{\mathbf{q}}_{G,k}, \mathbf{w}_{m,k})}{\partial w_{z,k}} \right] \\ &= \frac{\Delta t}{2} \begin{bmatrix} \hat{q}_{z,k} & -\hat{q}_{y,k} & \hat{q}_{x,k} \\ \hat{q}_{y,k} & \hat{q}_{z,k} & -\hat{q}_{w,k} \\ -\hat{q}_{x,k} & \hat{q}_{w,k} & \hat{q}_{z,k} \\ -\hat{q}_{w,k} & -\hat{q}_{x,k} & -\hat{q}_{y,k} \end{bmatrix} \end{aligned} \quad (\text{E13})$$

### Appendix E.3 | Linearization of the Measurement Function

As defined in Section 3.2,  $h^i(\mathbf{x}_k)$  consists of two parts:  $h_1^i(\mathbf{x})$  and  $h_2^i(\mathbf{x})$ . We will derive  $\frac{\partial h_1^i(\hat{\mathbf{x}}_k^-)}{\partial \mathbf{x}_k}$  and  $\frac{\partial h_2^i(\hat{\mathbf{x}}_k^-)}{\partial \mathbf{x}_k}$  separately. To simplify our notation, we omit the indices  $i$  and  $k$  in  $\mathbf{x}_k$ ,  ${}^G \mathbf{p}_{I,k}$ ,  ${}^I \mathbf{q}_{G,k}$ ,  $\hat{\mathbf{x}}_k^-$ ,  ${}^I \hat{\mathbf{p}}_{G,k}$ ,  ${}^I \hat{\mathbf{q}}_{G,k}$ ,  $a_1^{i,k}$ ,  $a_2^{i,k}$ , and  $a_3^{i,k}$  during the following derivation without a loss of generality.

#### (1) First part of the Jacobian matrix

The first part of the Jacobian matrix  $\mathbf{H}_k^i$  defined in Equation (45) can be written as follows:

$$\frac{\partial h_1^i(\hat{\mathbf{x}}^-)}{\partial \mathbf{x}} = \left[ \frac{\partial h_1^i(\hat{\mathbf{x}}^-)}{\partial {}^G \mathbf{p}_I}, 0_{1 \times 3}, \frac{\partial h_1^i(\hat{\mathbf{x}}^-)}{\partial {}^I \mathbf{q}_G}, 0_{1 \times 6} \right] \quad (\text{E14})$$

where:

$$h_1^i(\mathbf{x}) = \arctan \left( \text{sgn}({}^L d_{i,k}) \frac{a_2}{a_1} \right) \quad (\text{E15})$$

$\mathbf{x}$ ,  ${}^G\mathbf{p}_I$ , and  ${}^L\mathbf{q}_G$  are defined in Equation (40),  ${}^L\mathbf{a}_{i,k} = {}^L\mathbf{R}^G\boldsymbol{\pi}_{i,k} = [a_1, a_2, a_3]^T$  is the discrete form of  ${}^L\mathbf{a}_i$  defined in Equation (D12), and  ${}^Ld_{i,k}$  is the discrete form of  ${}^Ld_i$  defined in Equation (D9). Applying the chain rule yields the following:

$$\frac{\partial h_1^i(\cdot)}{\partial {}^L\mathbf{q}_G} = \frac{\partial h_1^i(\cdot)}{\partial {}^L\mathbf{a}_{i,k}} \cdot \frac{\partial {}^L\mathbf{a}_{i,k}}{\partial {}^L\mathbf{q}_G} \quad (\text{E16})$$

By substituting Equation (E15) into Equation (E16), the first term of Equation (E16) can be written as follows:

$$\frac{\partial h_1^i(\cdot)}{\partial {}^L\mathbf{a}_{i,k}} = \frac{\partial \arctan\left(\text{sgn}\left({}^Ld_{i,k}\right)\frac{a_2}{a_1}\right)}{\partial {}^L\mathbf{a}_{i,k}} \quad (\text{E17})$$

where  $\text{sgn}(\cdot)$  is the sign function and  $\frac{d\text{sgn}(x)}{dx} = 2\delta(x)$ . Now, Equation (E17) can be reduced as follows:

$$\frac{\partial h_1^i(\cdot)}{\partial {}^L\mathbf{a}_{i,k}} = \left(1 + \frac{a_2^2}{a_1^2}\right)^{-1} \left(2\delta\left({}^Ld_{i,k}\right) \frac{\partial {}^Ld_{i,k}}{\partial {}^L\mathbf{a}_{i,k}} \cdot \frac{a_2}{a_1} + \text{sgn}\left({}^Ld_{i,k}\right) \frac{\partial \frac{a_2}{a_1}}{\partial {}^L\mathbf{a}_{i,k}}\right) \quad (\text{E18})$$

According to Equations (D6) and (D9), we have the following:

$${}^Ld_{i,k} = d_{i,k} - {}^G\boldsymbol{\pi}_{i,k}^T \left({}^G\mathbf{p}_I + {}^G\mathbf{R}^L\mathbf{p}_L\right) = {}^G\boldsymbol{\pi}_{i,k}^T {}^G\mathbf{R}^L\mathbf{x}_{i,k}^* \quad (\text{E19})$$

where  ${}^G\mathbf{R}^L\mathbf{x}_{i,k}^*$  represents the shortest vector from the origin of  $\{L\}$  to plane  $\Pi_i$  in  $\{G\}$  at time  $k$ , which cannot be zero. In addition,  ${}^G\boldsymbol{\pi}_{i,k}$  is the normal of the planar surface, which cannot be zero. Therefore,  ${}^Ld_{i,k} \neq 0$ , and thus,  $\delta\left({}^Ld_{i,k}\right) = 0$ . Then, Equation (E18) is simplified as follows:

$$\frac{\partial h_1^i(\cdot)}{\partial {}^L\mathbf{a}_{i,k}} = \text{sgn}\left({}^Ld_{i,k}\right) \frac{\partial \frac{a_2}{a_1}}{\partial {}^L\mathbf{a}_{i,k}} \left(1 + \frac{a_2^2}{a_1^2}\right)^{-1} \quad (\text{E20})$$

where:

$$\frac{\partial \frac{a_2}{a_1}}{\partial {}^L\mathbf{a}_{i,k}} = \left[\frac{\partial \frac{a_2}{a_1}}{\partial a_1}, \frac{\partial \frac{a_2}{a_1}}{\partial a_2}, \frac{\partial \frac{a_2}{a_1}}{\partial a_3}\right] = \left[-\frac{a_2}{a_1^2}, \frac{1}{a_1}, 0\right] \quad (\text{E21})$$

Therefore, Equation (E20) has the following expression:

$$\frac{\partial h_1^i(\cdot)}{\partial {}^L\mathbf{a}_{i,k}} = \frac{1}{\mu_{i,k}} \boldsymbol{\lambda}_{i,k}^T \quad (\text{E22})$$

where:

$$\mu_{i,k} = a_1^2 + a_2^2, \quad (\text{E23a})$$

$$\boldsymbol{\lambda}_{i,k}^T = \text{sgn}\left({}^Ld_{i,k}\right) [-a_2, a_1, 0] \quad (\text{E23b})$$

The second term in Equation (E16) is given as follows:

$$\frac{\partial^L \mathbf{a}_{i,k}}{\partial^I \mathbf{q}_G} = \frac{\partial^L \mathbf{R}^G \pi_{i,k}}{\partial^I \mathbf{q}_G} = {}^L_I \mathbf{R} \frac{\partial^I \mathbf{R}^{TG} \pi_{i,k}}{\partial^I \mathbf{q}_G} \quad (\text{E24})$$

The above equation contains the partial derivative of the rotation matrix with respect to the quaternion, which can be solved by taking the derivative of the rotation matrix in an element-wise manner, as shown in Appendix E.1. Therefore, Equation (E24) can be derived by using the conclusion in Appendix E.1:

$$\frac{\partial^L \mathbf{a}_{i,k}}{\partial^I \mathbf{q}_G} = {}^L_I \mathbf{R} J_q^* \left( {}^I \mathbf{q}_G, {}^G \pi_{i,k} \right) \quad (\text{E25})$$

where:

$$J_q^* \left( {}^I \mathbf{q}_G, {}^G \pi_{i,k} \right) = J_q \left( {}^I \mathbf{q}_G^*, {}^G \pi_{i,k} \right) \text{diag}[1, -1, -1, -1] \quad (\text{E26a})$$

$$J_q \left( {}^I \mathbf{q}_G, {}^G \pi_{i,k} \right) = 2 \left[ \mathbf{u}_\pi, \left[ \mathbf{u}_\pi + \mathbf{q}_w {}^G \pi_{i,k} \right]_\times + \left( \mathbf{q}_v \cdot {}^G \pi_{i,k} \right) \mathbf{I}_3 - {}^G \pi_{i,k} \mathbf{q}_v^T \right] \quad (\text{E26b})$$

$$\mathbf{u}_\pi = {}^G \pi_{i,k} \times \mathbf{q}_v = \left[ {}^G \pi_{i,k} \right]_\times \mathbf{q}_v \quad (\text{E26c})$$

$${}^I \mathbf{q}_G = \left[ q_w, \mathbf{q}_v^T \right]^T = \left[ q_w, q_x, q_y, q_z \right]^T \quad (\text{E26d})$$

Substituting Equations (E22) and (E25) into Equation (E16) yields the following:

$$\frac{\partial h_1^i(\cdot)}{\partial^I \mathbf{q}_G} = \frac{1}{\mu_{i,k}} \lambda_{i,kI}^T {}^L_I \mathbf{R} J_q^* \left( {}^I \mathbf{q}_G, {}^G \pi_{i,k} \right) \quad (\text{E27})$$

Similarly, according to the chain rule, we have the following:

$$\frac{\partial h_1^i(\cdot)}{\partial^G \mathbf{p}_I} = \frac{\partial h_1^i(\cdot)}{\partial^L \mathbf{a}_{i,k}} \cdot \frac{\partial^L \mathbf{a}_{i,k}}{\partial^G \mathbf{p}_I} \quad (\text{E28})$$

Because  ${}^L \mathbf{a}_{i,k} = {}^L_G \mathbf{R}^G \pi_{i,k}$ , which indicates that  ${}^L \mathbf{a}_{i,k}$  is only related to the rotation part of the state, Equation (E28) has the following expression:

$$\frac{\partial h_1^i(\cdot)}{\partial^G \mathbf{p}_I} = 0 \quad (\text{E29})$$

By substituting Equations (E27) and (E29) into Equation (E14), we obtain the final expression of the first part of the Jacobian matrix:

$$\frac{\partial h_1^i(\hat{\mathbf{x}}^-)}{\partial \mathbf{x}} = \left[ 0_{1 \times 6}, \frac{1}{\hat{\mu}_{i,k}^-} \hat{\lambda}_{i,k}^{-T} {}^L_I \mathbf{R} J_q^* \left( {}^I \hat{\mathbf{q}}_G^-, {}^G \pi_{i,k} \right), 0_{1 \times 6} \right] \quad (\text{E30})$$

where  $\hat{\mu}_{i,k}^-$  is defined in Equation (E23a) and  $\hat{\lambda}_{i,k}^-$  is defined in Equation (E23b), both of which are evaluated at  $\hat{\mathbf{x}}^-$ .

## (2) Second part of the Jacobian matrix

The second part of the Jacobian matrix  $\mathbf{H}_k^i$  can be written as follows:

$$\frac{\partial h_2^i(\hat{\mathbf{x}}^-)}{\partial \mathbf{x}} = \left[ \frac{\partial h_2^i(\hat{\mathbf{x}}^-)}{\partial^G \mathbf{p}_I}, 0_{1 \times 3}, \frac{\partial h_2^i(\hat{\mathbf{x}}^-)}{\partial^I \mathbf{q}_G}, 0_{1 \times 6} \right] \quad (\text{E31})$$

where:

$$h_2^i(\mathbf{x}) = \frac{|^L d_{i,k}|}{\sqrt{a_1^2 + a_2^2}} \quad (\text{E32})$$

We first consider the derivative with respect to the rotation part:

$$\frac{\partial h_2^i(\cdot)}{\partial^I \mathbf{q}_G} = \frac{1}{\sqrt{a_1^2 + a_2^2}} \frac{\partial |^L d_{i,k}|}{\partial^I \mathbf{q}_G} + |^L d_{i,k}| \frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial^I \mathbf{q}_G} \quad (\text{E33})$$

In the first part of Equation (E33), we have the following:

$$\frac{\partial |^L d_{i,k}|}{\partial^I \mathbf{q}_G} = \text{sgn}(|^L d_{i,k}|) \frac{\partial ^L d_{i,k}}{\partial^I \mathbf{q}_G} \quad (\text{E34})$$

Substituting Equation (D9) into Equation (E34) yields the following:

$$\frac{\partial |^L d_{i,k}|}{\partial^I \mathbf{q}_G} = \text{sgn}(|^L d_{i,k}|) \frac{\partial (d_{i,k} - {}^G \pi_{i,k}^T ({}^G \mathbf{p}_I + {}^G \mathbf{R}^I \mathbf{p}_L))}{\partial^I \mathbf{q}_G} \quad (\text{E35})$$

According to Appendix E.1, we have the following relationship:

$$\frac{\partial {}^G \mathbf{R}^I \mathbf{p}_L}{\partial^I \mathbf{q}_G} = \frac{\partial {}^I \mathbf{R}^{TI} \mathbf{p}_L}{\partial^I \mathbf{q}_G} = J_q({}^I \mathbf{q}_G, {}^I \mathbf{p}_L) \quad (\text{E36})$$

By substituting Equation (E36) into Equation (E35), we obtain the following:

$$\frac{\partial |^L d_{i,k}|}{\partial^I \mathbf{q}_G} = -\text{sgn}(|^L d_{i,k}|) {}^G \pi_{i,k}^T J_q({}^I \mathbf{q}_G, {}^I \mathbf{p}_L) \quad (\text{E37})$$

In the second part of Equation (E33), the chain rule is applied:

$$\frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial^I \mathbf{q}_G} = \frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial {}^L \mathbf{a}_{i,k}} \cdot \frac{\partial {}^L \mathbf{a}_{i,k}}{\partial^I \mathbf{q}_G} \quad (\text{E38})$$

By expanding  ${}^L \mathbf{a}_i$  into its element-wise form, we obtain the following:

$$\frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial {}^L \mathbf{a}_{i,k}} = \left[ \frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial a_1}, \frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial a_2}, \frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial a_3} \right] = -(a_1^2 + a_2^2)^{-\frac{3}{2}} [a_1, a_2, 0] \quad (\text{E39})$$

We then substitute Equations (E25) and (E39) into Equation (E38) as follows:

$$\frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial^I \mathbf{q}_G} = -(a_1^2 + a_2^2)^{-\frac{3}{2}} [a_1, a_2, 0] {}^L \mathbf{R} J_q^*({}^I \mathbf{q}_G, {}^G \pi_{i,k}) \quad (\text{E40})$$

By substituting Equations (E37) and (E40) into Equation (E33), the final expression of Equation (E33) is written as follows:

$$\frac{\partial h_2^i(\cdot)}{\partial {}^I \mathbf{q}_G} = -\text{sgn}({}^L d_{i,k}) \frac{{}^G \pi_{i,k}^T}{\sqrt{\mu_{i,k}}} J_q({}^I \mathbf{q}_G, {}^I \mathbf{p}_L) + |{}^L d_{i,k}| \kappa_{i,k}^T {}^L \mathbf{R} J_q^*({}^I \mathbf{q}_G, {}^G \pi_{i,k}) \quad (\text{E41})$$

where:

$$\kappa_{i,k}^T = -\mu_{i,k}^{-\frac{3}{2}} [a_1, a_2, 0] \quad (\text{E42})$$

Similarly, we have the following:

$$\frac{\partial h_2^i(\cdot)}{\partial {}^G \mathbf{p}_I} = \frac{1}{\sqrt{a_1^2 + a_2^2}} \frac{\partial |{}^L d_{i,k}|}{\partial {}^G \mathbf{p}_I} + |{}^L d_{i,k}| \frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial {}^G \mathbf{p}_I} \quad (\text{E43})$$

In the first part of Equation (E43), we have the following:

$$\frac{\partial |{}^L d_{i,k}|}{\partial {}^G \mathbf{p}_I} = \text{sgn}({}^L d_{i,k}) \frac{\partial {}^L d_{i,k}}{\partial {}^G \mathbf{p}_I} \quad (\text{E44})$$

We substitute Equation (D9) into Equation (E44) as follows:

$$\frac{\partial |{}^L d_{i,k}|}{\partial {}^G \mathbf{p}_I} = \text{sgn}({}^L d_{i,k}) \frac{\partial (d_{i,k} - {}^G \pi_{i,k}^T ({}^G \mathbf{p}_I + {}^G \mathbf{R}^I \mathbf{p}_L))}{\partial {}^G \mathbf{p}_I} = -\text{sgn}({}^L d_{i,k}) {}^G \pi_{i,k}^T \quad (\text{E45})$$

In the second part of Equation (E43), because  ${}^L \mathbf{a}_{i,k} = {}^L \mathbf{R}^G \pi_{i,k} = [a_1, a_2, a_3]^T$ ,  $a_1$  and  $a_2$  are only related to the rotation part of the state. Therefore, we have the following:

$$\frac{\partial (a_1^2 + a_2^2)^{-\frac{1}{2}}}{\partial {}^G \mathbf{p}_I} = 0 \quad (\text{E46})$$

Substituting Equations (E44) and (E46) into Equation (E43) yields the following:

$$\frac{\partial h_2^i(\cdot)}{\partial {}^G \mathbf{p}_I} = -\text{sgn}({}^L d_{i,k}) \frac{{}^G \pi_{i,k}^T}{\sqrt{\mu_{i,k}}} \quad (\text{E47})$$

By substituting Equations (E41) and (E47) into Equation (E31), we obtain the final expression of the second part of the Jacobian matrix:

$$\begin{aligned} & \frac{\partial h_2^i(\hat{\mathbf{x}}^-)}{\partial \mathbf{x}} \\ &= \begin{bmatrix} -\text{sgn}({}^L \hat{d}_{i,k}^-) \frac{{}^G \pi_{i,k}^T}{\sqrt{\hat{\mu}_{i,k}^-}}, 0_{1 \times 3}, -\text{sgn}({}^L \hat{d}_{i,k}^-) \frac{{}^G \pi_{i,k}^T}{\sqrt{\hat{\mu}_{i,k}^-}} J_q({}^I \hat{\mathbf{q}}_G^-, {}^I \mathbf{p}_L) + |{}^L \hat{d}_{i,k}^-| \hat{\kappa}_{i,k}^T {}^L \mathbf{R} J_q^*({}^I \hat{\mathbf{q}}_G^-, {}^G \pi_{i,k}), 0_{1 \times 6} \end{bmatrix} \end{aligned} \quad (\text{E48})$$

where  $\hat{\kappa}_{i,k}^T$  is defined in Equation (E42) and evaluated at  $\hat{\mathbf{x}}^-$ .

By re-arranging Equations (E30) and (E48), the Jacobian matrix of  $h^i(\mathbf{x}_k)$  defined with respect to  $\mathbf{x}_k$  and evaluated at  $\hat{\mathbf{x}}_k^-$  can be written as follows:

$$\mathbf{H}_k^i = \begin{bmatrix} \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} & \frac{1}{\hat{\mu}_{i,k}^-} \hat{\boldsymbol{\lambda}}_{i,k}^{-T} \mathbf{L} \mathbf{R} \mathbf{J}_q^* \left( {}^I \hat{\mathbf{q}}_G^-, {}^G \boldsymbol{\pi}_{i,k} \right) & \mathbf{0}_{1 \times 6} \\ -\text{sgn} \left( {}^L \hat{d}_{i,k}^- \right) \frac{{}^G \boldsymbol{\pi}_{i,k}^T}{\sqrt{\hat{\mu}_{i,k}^-}} & \mathbf{0}_{1 \times 3} & -\text{sgn} \left( {}^L \hat{d}_{i,k}^- \right) \frac{{}^G \boldsymbol{\pi}_{i,k}^T}{\sqrt{\hat{\mu}_{i,k}^-}} \mathbf{J}_q \left( {}^I \hat{\mathbf{q}}_G^-, {}^I \mathbf{p}_L \right) + \left| {}^L \hat{d}_{i,k}^- \right| \hat{\boldsymbol{\kappa}}_{i,k}^T \mathbf{L} \mathbf{R} \mathbf{J}_q^* \left( {}^I \hat{\mathbf{q}}_G^-, {}^G \boldsymbol{\pi}_{i,k} \right) & \mathbf{0}_{1 \times 6} \end{bmatrix} \quad (\text{E49})$$

## APPENDIX F | ADDITIONAL RESULTS FOR FAULT DETECTION

TABLE F1

Statistical Results of FAR and FDR for a Step Failure ( $\alpha = 0.01$ )

Range Noise Setting	Failure Group	Total Gaussian-GMM Method		Gaussian Method	
		FAR (%)	FDR (%)	FAR (%)	FDR (%)
N1	A1	4.47	<b>18.75</b>	2.79	14.38
	A2		<b>85.00</b>		71.88
N2	A1	3.07	<b>18.13</b>	1.40	12.50
	A2		<b>72.50</b>		52.50
N3	A1	4.47	14.38	3.35	15.00
	A2		<b>91.25</b>		88.75
N4	A1	1.68	<b>11.88</b>	1.40	9.38
	A2		<b>71.88</b>		60.63
Gaussian $\mathcal{N}(0, 0.03^2)$	A1	2.23	11.88	2.23	11.88
	A2		68.75		68.75

TABLE F2

Statistical Results of FAR and FDR for a Step Failure ( $\alpha = 0.001$ )

Range Noise Setting	Failure Group	Total Gaussian-GMM Method		Gaussian Method	
		FAR (%)	FDR (%)	FAR (%)	FDR (%)
N1	A1	2.23	<b>14.38</b>	1.96	8.75
	A2		<b>73.75</b>		62.50
N2	A1	1.11	<b>10.00</b>	0.84	6.88
	A2		<b>58.13</b>		37.50
N3	A1	1.68	<b>9.38</b>	1.68	8.13
	A2		<b>81.88</b>		76.20
N4	A1	1.40	<b>5.63</b>	0.84	3.13
	A2		<b>55.63</b>		47.50
Gaussian $\mathcal{N}(0, 0.03^2)$	A1	0.56	6.88	0.56	6.88
	A2		68.13		68.13