



# An Informational Approach for Fault Tolerant Data Fusion Applied to a UAV's Attitude, Altitude and Position Estimation

Majd Saied, Ali Rida Tabikh, Clovis Francis, Hussein Hamadi, Benjamin Lussier

## ► To cite this version:

Majd Saied, Ali Rida Tabikh, Clovis Francis, Hussein Hamadi, Benjamin Lussier. An Informational Approach for Fault Tolerant Data Fusion Applied to a UAV's Attitude, Altitude and Position Estimation. IEEE Sensors Journal, 2021, 21 (24), pp.27766-27778. 10.1109/JSEN.2021.3124731 . hal-03475134

**HAL Id: hal-03475134**

**<https://hal.science/hal-03475134>**

Submitted on 10 Dec 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# An Informational Approach for Fault Tolerant Data Fusion Applied to a UAV's Attitude, Altitude and Position Estimation

M. Saied, A. Tabikh, C. Francis, H. Hamadi and B. Lussier <sup>\*†‡§</sup>

October 19, 2021

## Abstract

This paper presents a fault tolerance architecture for data fusion mechanisms that tolerates sensor faults in a multirotor Unmanned Aerial Vehicle (UAV). The developed approach is based on the traditional duplication/comparison method and is carried out via error detection and system recovery to both detect and isolate the faulty sensors. It is applied on an informational framework using extended Informational Kalman Filters (IKF) for state estimation with prediction models based on available sensors measurements. Error detection is realized through residuals comparisons using the Bhattacharyya Distance (BD), an informational measure that estimates the similarity of two probability distributions. An optimal thresholding based on Bhattacharyya criterion is applied. In order to identify the faulty sensor, the Bhattacharyya distance between the a priori and a posteriori distributions of each IKF is also computed. The system recovery is done by substituting the erroneous state by an error-free state. The proposed architecture alleviates the assumption of a fault-free prediction model using the information surprise concept instead of hardware redundancy. The performance of the proposed framework is shown through offline validation using real measurements from navigation sensors of a multirotor UAV with fault injection.

Sensor Fusion, Fault Tolerance, Information Theory, Unmanned Aerial Vehicle

---

<sup>\*</sup>M. Saied is with the School of Engineering, Department of Electrical and Electronics Engineering, Lebanese International University (LIU), Bekaa, Lebanon (e-mail: majd.saied@liu.edu.lb).

<sup>†</sup>A. Tabikh and C. Francis are with the Scientific Research Center in Engineering, Lebanese University, Faculty of Engineering, Hadath, Lebanon (e-mail: cfrancis@ul.edu.lb).

<sup>‡</sup>H. Hamadi is with Altran, France

<sup>§</sup>B. Lussier is with the Heudiasyc Laboratory, University of Technology of Compi 'egne, France (e-mail: benjamin.lussier@hds.utc.fr).

# 1 Introduction

Multisensor data fusion finds wide application in many robotic fields such as localization, sensor networks, environment mapping and object recognition [1]. It consists of combining information from various sensors and knowledge sources to better describe the process or phenomenon of interest and achieve more precise reasoning than could be accomplished by the usage of only one sensor [2]. Data fusion is considered as a research field involving multiple fields including artificial intelligence, statistical estimation, information theory and signal processing [3]. Performing data fusion can improve reliability, confidence and consistency of the data.

Despite the evident potential benefit of these techniques, there are a number of issues and limitations that make data fusion approaches a challenging task. First, their implementation requires redundant, diverse and complementary sensors which increases the risks of hardware faults. Other issues are related to the data used in the fusion process, diversity and imperfection of the used sensors and the type of the application [2], [4], [5], [6].

In addition, the validation of data fusion approaches faces two major problems [7]:

- The difficulty of predicting and designing the behavior of fusion algorithms by formal techniques such as proof checking and formal model.
- The near-infinite execution context generated by the open environment of the robotic systems where conditions may vary at any moment and in many different ways. The validation of UAV systems for example needs multiple long flights to encounter all the possible situations, which makes it a costly, difficult and long process.

As an alternative to this validation, some works developed fault tolerance mechanisms for data fusion to remove or reduce the effects of faults on the system or process operation [8], [9]. Fault Tolerance is defined as one of the four means to attain dependability [10] and aims to allow the system to deliver a correct service even after the occurrence of errors [11, 12]. Fault tolerance methods are generally implemented via error detection and system recovery:

- Error detection aims to detect the erroneous state of the system before errors are propagated and cause system failures. It can be achieved by duplication-comparison, temporal watchdog and likelihood checks.
- System recovery allows the system to recover to an operational mode after fault occurrence. It is mainly achieved by error handling through recovery, pursuit or compensation.

Fault tolerant data fusion has been in use for over 50 years especially in aircraft navigation systems [13]. The existing approaches in the literature are mainly based on the duplication-comparison techniques that consist in comparing results from at least two redundant units that are independent of the

faults to tolerate and provide the same service. The duplication design methods incorporate two redundancy strategies: analytical redundancy and hardware redundancy.

- Duplication based on analytical models: In this first category, an analytical model is used as a diversification of physical sensors. Generally the data fusion mechanism applied in this type of approaches is the Kalman filter that uses the model of the system to estimate a measurement redundant to that delivered by a real sensor. Then the residue resulting from the combination of the measurements estimated by the model and delivered by the sensor is used as a fault indicator. Examples of duplication based on an analytical model of the fusion process are found in [14–18].
- Duplication based on hardware redundancy: These approaches combine multiple data sources. Unlike techniques based on the analytical model of the system, they use the analysis of some internal parameters to ensure fault tolerance. In [19], malfunction is detected based on the temporal analysis of conflict resulting from the fusion of data sources using the Smets’s Transferable Belief Model (TBM). The works in [20, 21] proposed the analysis of the static and dynamic reliability of data sources to detect the defective source. Other similar examples of fault tolerance in data fusion are mentioned in [22, 23].

With the growing applications of autonomous unmanned aerial vehicles, fault tolerant data fusion is becoming a critical requirement for safe and reliable operation. In fact, UAVs are capable of achieving different missions under unknown circumstances, where human involvement is unsafe or impossible [24]. Their operation depends on sensors that are subjected to some types of faults and failures [25], [26]. Therefore, it is fundamental for the UAVs to diagnose any sensors faults to ensure an accurate estimate of their states. In the literature, multi-sensor data fusion architectures applied on UAVs were investigated in different works for position, velocity and attitude estimation [27], [28], for estimation of unknown ship positions during landing operations using Extended Kalman Filters (EKF) [29], and diagnosis of the safety distance between drones and high-voltage overhead transmission lines [30]. A real analysis of sensor fusion tested on real UAV platforms was discussed in [31].

The fault tolerance (FT) in data fusion applied to UAVs was also considered in some studies. In [32], the authors proposed an architecture for integrated navigation systems of UAVs with application to height sensors measurements fusion. The FT scheme included a main Kalman filter with three sub-filters (radar altimeter/barometric altimeter/GPS). The Chi-Square test with test propagators used normally to test the consistency between a signal and its reference in a stochastic dynamic system was employed for fault isolation. A fault tolerant multi-sensor fusion for UAV attitude estimation was developed in [33] using the Unscented Information Filter (UIF). The proposed scheme, validated using real flight data, enables the on-line calibration of sensors which leads to an improved interpretation of the sensor health conditions. Similar approaches were

considered also in [34] with a comparison between different forms of Kalman filters applied for data fusion of Inertial Measurement Unit (IMU), Global Positioning System (GPS) receiver, and magnetometer sensors data. The work in [35] proposed a fault tolerance architecture based on duplication-comparison technique for perception targeting sensors and software faults on an outdoor quadrotor UAV. The architecture uses extended Kalman filters for data fusion and a weighted average voting system with an added analytical redundancy using the dynamic model of the quadrotor.

In most of the works considering these approaches, the fault indicators were obtained as a difference between deterministic values corresponding to the nominal estimations and the observations collected from the sensors. However, more robust indicators could be derived from the comparison of the probability distributions of the prediction and the observations using informational metrics such as  $\alpha$  divergences [36], Kullback-Leibler divergence [37] and Bhattacharyya distance [38]. These informational measures give more accurate information about sensor observations and estimations such as volume, orientation and uncertainties in distributions, which leads to more robust fault indicators [39].

Among the metrics cited before, the Bhattacharyya distance was used in fault detection and isolation techniques in different applications such as analog circuits [40], [41], multi-robot systems [42], rolling element bearings [43] and many other applications. In [42], the Bhattacharyya distance was applied with the Informational Kalman Filter (IF), that deals with the information matrix and vector, for diagnosis of sensor faults in wheeled mobile robots. The informational filter presents several advantages over the covariance form of the Kalman filter for multi-sensor systems. Its computational requirement is less than that of the covariance form since it requires the inversion of the information matrix which is of the dimension of the state vector instead of the inversion of the composite innovation covariance matrix which is of the dimension of the observation vector. In addition, it performs better than the Kalman Filter in the correction step since it requires only additions at this stage. It allows also for a distributed or decentralized data fusion architecture [44], and is therefore considered in this work.

This paper proposes a global informational approach for fault tolerant data fusion applied to UAV attitude, altitude and position estimation in absence of a dynamic model. The developed approach is based on the duplication/comparison traditional method. An informational framework is proposed and designed using extended Informational Kalman Filters for state estimation with prediction models based on available sensors measurements. For the diagnosis layer, fault indicators are generated and evaluated using the statistical Bhattacharyya metric. The system recovers from a fault by switching from the erroneous state into an error free state. Compared to the work presented in [35], the architecture in this paper does not make use of an estimation of the dynamic model of the UAV to isolate the faulty sensors since this estimation may introduce errors due to the model imperfections, the uncertainties in the system or the unusual environmental conditions. The evaluation of the information surprise provided by the sensors measurements is adopted instead for fault diagnosis. The concept

of surprise appeared remarkable for its simplicity and generality which resulted in its applicability to areas as diverse as learning, data mining [45], fault detection [44], etc. Based on the above review, the novel contributions of this work are summarized as below:

- The proposition of a stochastic fault tolerant multi-sensor data fusion architecture for a multirotor UAV that alleviates the assumption of a fault-free prediction model and reduces the number of redundant sensors needed for fault isolation. This is achieved by developing an algorithm based on the concept of the information surprise evaluated using the Bhattacharyya distance between the predicted and the corrected distributions of the extended Information filters;
- A thresholds optimization process using the Bhattacharyya criterion developed first in [42] to maximize the information surprise between the prior and the posterior distributions;

The rest of the paper is organized as follows: in Section 2, some preliminaries about the information filter and the Bhattacharyya distance are presented. The proposed fault tolerant data fusion architecture is detailed in Section 3. Then, offline real experimental data validation results are presented in Section 4. Finally a conclusion and a discussion are provided in Section 5.

## 2 Preliminaries

Consider the nonlinear system represented as:

$$\begin{aligned} x_k &= f(x_{k-1}, u_{k-1}) + \omega_{k-1} \\ z_k &= h(x_k) + v_k \end{aligned} \quad (1)$$

where, at a given instant  $k$ ,  $x_k \in \mathbb{R}^n$  is the state vector,  $z_k \in \mathbb{R}^m$  is the measurement vector,  $u_k \in \mathbb{R}^r$  is the input vector,  $\omega_k \in \mathbb{R}^n$  and  $v_k \in \mathbb{R}^m$  are white Gaussian noises associated respectively to the state model with a covariance matrix  $Q_k \in \mathbb{R}^{n \times n}$  and to the measurement model with a covariance matrix  $R_k \in \mathbb{R}^{m \times m}$ , and  $f : (x_{k-1}, u_{k-1}) \mapsto x_k$ , and  $h : x_k \mapsto z_k$  are two defined mapping.

### 2.1 Extended Information Filter

An Extended Information Filter (EIF) is applied to estimate the altitude, attitude and position of a multirotor UAV. It consists of two phases: (a) the prediction phase that uses the evolution model which is a representation of the theoretical operation of the UAV, (b) the update phase where sensors outputs are needed to correct the prediction. The information form of the Kalman filter is obtained by replacing the representation of the state estimate  $\hat{x}_k$  and covariance  $P_k$  in the EKF with the information state vector  $y_k$  and information matrix  $Y_k$ :

$$y_{k/k} = Y_{k/k} x_{k/k} \quad (2)$$

$$Y_{k/k} = P_k^{-1} \quad (3)$$

The EIF equations are formulated as in [44]:

- Prediction Step:

$$Y_{k/k-1} = [F_{k-1}Y_{k-1/k-1}F_{k-1}^T + B_{k-1}Q_kB_{k-1}^T + Q_{k-1}]^{-1} \quad (4)$$

$$y_{k/k-1} = Y_{k/k-1}f(x_{k-1/k-1}, u_{k-1}) \quad (5)$$

- Update Step:

$$Y_{k/k} = Y_{k/k-1} + \sum_{l=1}^w I_l(k) \quad (6)$$

$$y_{k/k} = y_{k/k-1} + \sum_{l=1}^w i_l(k) \quad (7)$$

$w$  is the number of sensor blocks.  $I_l(k)$  is the information matrix contribution of the sensor  $l$  written as:

$$I_l(k) = (H_k^l)^T (R_k^l)^{-1} H_k^l \quad (8)$$

and  $i_l(k)$  is the information state associated to the measurement from sensor  $l$  at time  $k$ ,  $z_k^l$ , expressed by:

$$i_l(k) = (H_k^l)^T (R_k^l)^{-1} [(z_k^l - h_{k/k-1}^l) + (H_k^l x_{k/k-1}^l)] \quad (9)$$

with  $h_{k/k-1}^l = h^l(x_{k/k-1})$ ,  $H_k^l \in R^{m \times n}$  and  $R_k^l$  being respectively the non linear measurement equation, its Jacobian and the covariance matrix related to the sensor  $l$ . The Jacobians  $F_k$ ,  $H_k$  and  $B_k$  are given as:

$$F_k = \frac{\partial f}{\partial x}|_{x=x_{k/k}}, \quad H_k = \frac{\partial h}{\partial x}|_{x=x_{k/k}}, \quad B_k = \frac{\partial f}{\partial u}|_{u=u_k} \quad (10)$$

## 2.2 Bhattacharyya Distance (BD)

The Bhattacharyya Distance (BD) is a measure of dissimilarity between two probability distributions  $p(x)$  and  $q(x)$ . It is based on the Bhattacharyya coefficient that estimates the overlap between two statistical populations [38]:

$$BD(p, q) = -\log[BC(p, q)] \quad (11)$$

with:

$$BC(p, q) = \sum_{x \in X} \sqrt{p(x) \cdot q(x)} \quad \text{or} \quad \int_{x \in X} \sqrt{p(x) \cdot q(x)} \quad (12)$$

being the Bhattacharyya coefficient for discrete and continuous probability distributions respectively. Considering two normally distributed classes  $p$  and  $q$ ,

with  $\mu_p$ ,  $\mu_q$ ,  $\sigma_p$  and  $\sigma_q$  being their means and variances respectively, the Bhattacharyya distance can be formulated as:

$$BD(p, q) = \frac{1}{4} \log\left(\frac{1}{4}\left(\frac{\sigma_p^2}{\sigma_q^2} + \frac{\sigma_q^2}{\sigma_p^2} + 2\right)\right) + \frac{1}{4}\left(\frac{(\mu_p - \mu_q)^2}{\sigma_q^2 + \sigma_p^2}\right) \quad (13)$$

For multivariate normal distributions,  $BD$  is expressed as:

$$BD = \frac{1}{8}(\mu_p - \mu_q)^T \sigma^{-1}(\mu_p - \mu_q) + \frac{1}{2} \log\left(\frac{|\det \sigma|}{|\det \sigma_p| \cdot |\det \sigma_q|}\right) \quad (14)$$

with  $\sigma = \frac{\sigma_p + \sigma_q}{2}$ .

From the equation above, one can notice that the Bhattacharyya distance can be divided into two terms:

- $\frac{1}{8}(\mu_p - \mu_q)^T \sigma^{-1}(\mu_p - \mu_q)$  representing the Mahalanobis distance (MD) between the two distributions;
- $\frac{1}{2} \log\left(\frac{|\det \sigma|}{|\det \sigma_p| \cdot |\det \sigma_q|}\right)$  representing the mutual information between  $p(x)$  and  $q(x)$  and the one with covariance  $\sigma$ ;

Bhattacharyya distance has been already used to measure class separability in classification problems and it was shown to be better than the Mahalanobis distance. When two classes have different variances and similar means, MD will be zero, while BD increases with the difference between the variances. It is also preferred over the Kullback-Leibler divergence because it does not require a reference distribution and is therefore symmetrical. However, the online computation of this distance between moving normal distributions may introduce some latency and detection delay that should be studied as we propose in our perspectives.

### 3 Comparison/Duplication Fault Tolerance Architecture

The fault tolerant data fusion architecture using comparison/duplication is presented in this section. It is based on the approach proposed in [7] and its general principle is illustrated in Fig. 1.

The duplication/comparison approach in data fusion consists of using at least two redundant units performing fusion of data collected from diversified or redundant sensors to estimate the system states and then comparing them to detect an abnormal behavior of the system. Once an error is detected, the outputs of the fusion blocks and the sensors measurements are analysed for error identification and for estimating the correct output. Under the assumption of a single fault, this approach can deal with one hardware sensor fault.

As shown in Fig. 1, the architecture consists of two parallel data fusion units DF1 and DF2 resulting from (S1,S2) and (S3,S4) sensor blocks respectively. S1



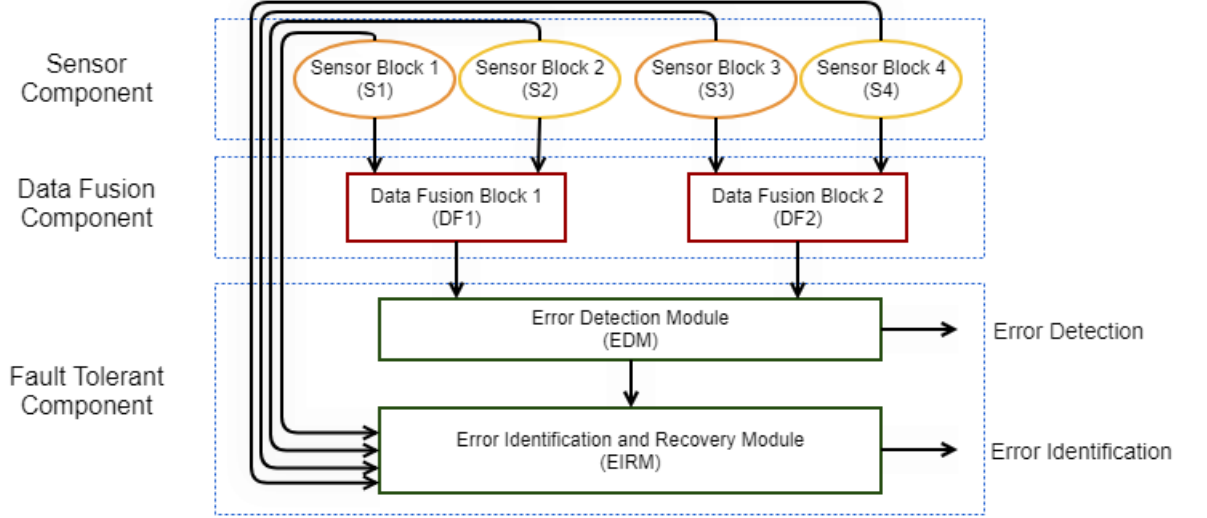


Figure 1: Duplication-comparison architecture for fault tolerant data fusion [7]

and S3 are supposed to be functionally similar (i.e. either redundant sensors or diversified sensors providing the same information), the same holds for S2 and S4. The outputs of the duplicated and diversified components are compared and are used as follows:

- The outputs of EIKF1 and EIKF2 are compared using the Bhattacharyya metric to detect the presence of an error.
- The outputs of the sensor blocks (S1,S3) and (S2,S4) are compared to identify the type of the fault.
  - If the outputs of the functionally similar sensors are different, then a sensor fault is diagnosed. The erroneous branch and the defective sensor can be identified by residual analysis and the fault is tolerated using the outputs of the error free branch.
  - If the sensors outputs are identical, then a software fault is diagnosed.

Each component of the architecture is detailed in Fig. 2, and in the following sections.

### 3.1 Sensor Component

In our work, the data fusion fault tolerant architecture is implemented on a multirotor UAV to provide real flight data for experimental validation using fault injection. This UAV is equipped with various redundant and diversified sensors grouped in sensor blocks.

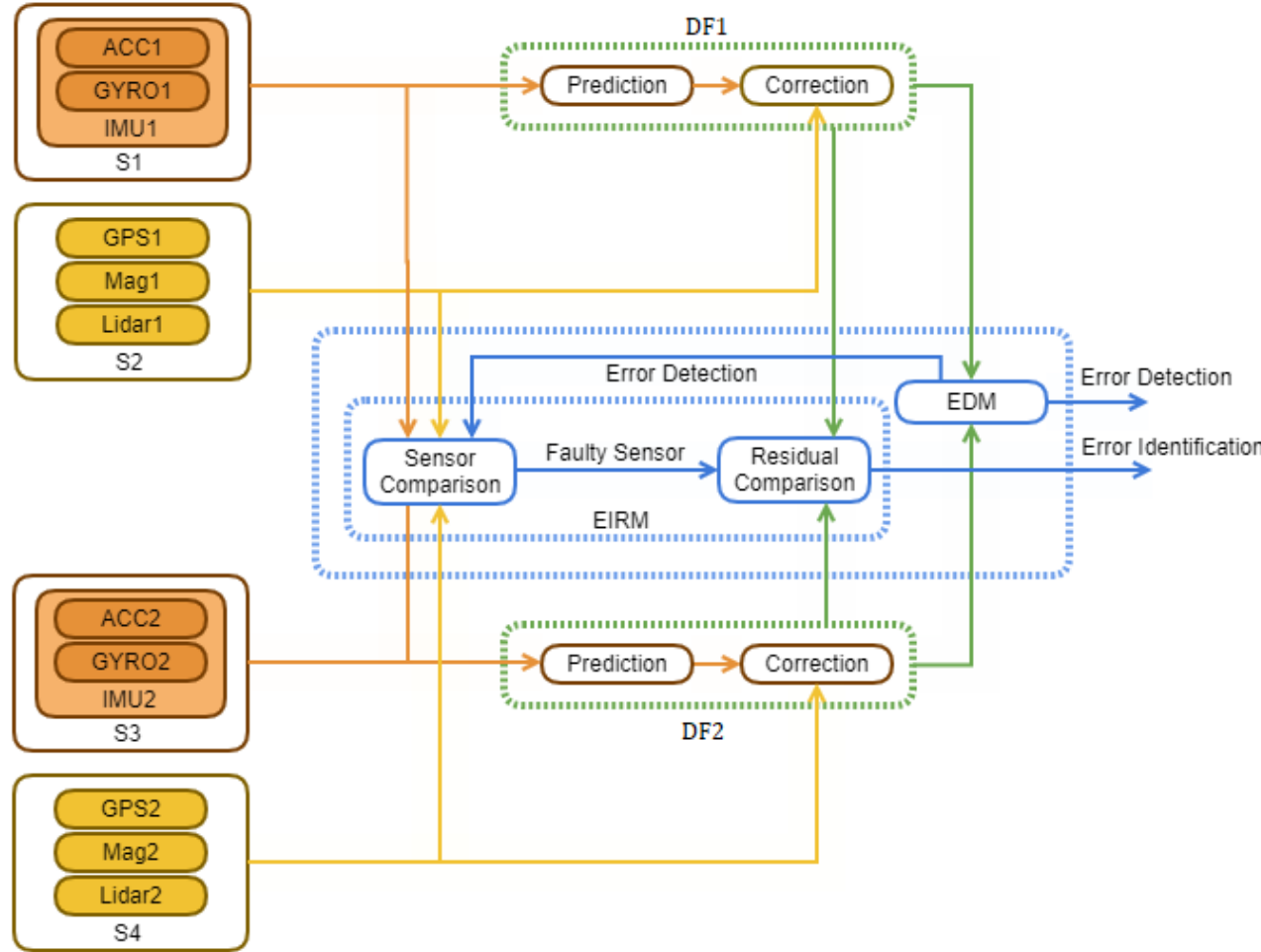


Figure 2: Fault Tolerance architecture implementation on the UAV system

- Sensor block S1: It consists of an *InvenSense ICM-20689* Inertial Measurement Unit (IMU1) that combines a 3-axis gyroscope (Gyro1), a 3-axis accelerometer (Acc1), and a Digital Motion Processor TM (DMP).
- Sensor block S2: It consists of one low cost Ublox NEO-M8N GPS module (GPS1) used to measure the absolute position of the UAV, one IST8310 magnetometer (Mag1) based on the anisotropic magneto resistance (AMR) technology to estimate the heading (yaw) of the UAV, and one LIDAR-Lite v3 Lidar (Lidar1) to measure the altitude of the UAV.
- Sensor block S3: It consists of a *Bosch BMI055* Inertial Measurement Unit (IMU2) that includes a digital, tri-axial 12-bit acceleration sensor (Acc2) and a triaxial 16-bit gyroscope (Gyro2).
- Sensor Block S4: It consists of one low cost Ublox NEO-M8N GPS module (GPS2), one IST8310 magnetometer (Mag2) in addition to one Lidar (Lidar2).

In addition to these sensors, a Real-time kinematic (RTK) GPS providing measurements with an accuracy of  $1cm$  is installed and will be used as ground truth. Note that in this architecture the sensors are not diversified, which means that we will not be able to tolerate common cause failures, such as GPS rebounds or absence of data due to being in a tunnel.

The specifications of the different used sensors are summarized in Table 1.

### 3.2 Data Fusion Component

The first Extended Information Kalman Filter (EIKF1) of the architecture uses data from the Inertial Measurement Unit system (IMU1) with the sensors GPS1, Mag1 and Lidar1. The sensor block S1 is used in the evolution model, and the sensor block S2 is used in the update step. The second fusion block has a similar construction.

We define the two coordinate frames: a body-fixed coordinate system ( $\mathcal{B}$ ), with its origin located at the accelerometer triad and aligned to the casing, and a local North-East-Down (NED) navigation cartesian frame ( $\mathcal{N}$ ). For most applications this frame is defined stationary with respect to the earth if the UAV is not expected to move over large distances. The rotation between the two frames is given by the three Euler angles: roll ( $\phi$ ), pitch ( $\theta$ ) and yaw ( $\psi$ ), and the rotating Direction Cosine Matrix is defined as:

$$\mathcal{R}_N^B = \begin{bmatrix} c\psi c\theta & -s\psi c\theta + c\psi s\theta s\phi & s\psi s\theta + c\psi \theta c\phi \\ s\psi c\theta & c\psi c\theta + s\psi s\theta s\phi & -c\psi s\theta + s\psi \theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \quad (15)$$

#### 3.2.1 Prediction Step

The prediction equations are expressed in terms of the accelerometer and gyroscope measurements. At each instant  $k$ , the quadrotor state and input vectors

Sensor	Specifications
<ul style="list-style-type: none"> <li>IMU modules: <ul style="list-style-type: none"> <li>Gyroscope <ul style="list-style-type: none"> <li>Nonlinearity: up to <math>\pm 0.05\%</math></li> <li>Cross-Axis Sensitivity: up to <math>\pm 1\%</math></li> <li>Full-scale range: <math>\pm 2000^\circ/s</math></li> </ul> </li> <li>Accelerometer <ul style="list-style-type: none"> <li>Nonlinearity: <math>\pm 0.5\%</math></li> <li>Cross-Axis Sensitivity: up to <math>\pm 1\%</math></li> <li>Full-scale range: <math>\pm 16g</math></li> </ul> </li> </ul> </li> <li>GPS module <ul style="list-style-type: none"> <li>Velocity accuracy: <math>0.05m/s</math></li> <li>Heading accuracy: <math>0.3^\circ</math></li> <li>Horizontal position accuracy: <math>2.5m</math></li> </ul> </li> <li>Magnetometer module <ul style="list-style-type: none"> <li>Sensitivity: <math>330LSB/G</math></li> <li>Dynamic range: <math>16(X \text{ and } Y)/25(Z) \text{ } G</math></li> <li>Accuracy: up to <math>1.5^\circ</math></li> </ul> </li> <li>Lidar Lite-V3 <ul style="list-style-type: none"> <li>Range: <math>40m</math></li> <li>Accuracy (<math>&lt; 5m</math>): <math>\pm 2.5cm</math></li> <li>Resolution: <math>\pm 1cm</math></li> </ul> </li> </ul>	

Table 1: Sensors Specifications

are defined as:

$$X_k = [\dot{x}_{b_k} \dot{y}_{b_k} \dot{z}_{b_k} \phi_k \theta_k \psi_k]^T \quad (16)$$

$$u_k = [a_{x_k} a_{y_k} a_{z_k} p_k q_k r_k]^T \quad (17)$$

where  $\dot{x}_b$ ,  $\dot{y}_b$ ,  $\dot{z}_b$  and  $p$ ,  $q$ ,  $r$  are the body-axis velocity components and angular rates respectively.

The accelerations  $a_x$ ,  $a_y$  and  $a_z$  are measured with respect to the fixed gravity vector such as:

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \frac{d}{dt} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} + \mathcal{R}_N^B \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \quad (18)$$

with  $\frac{d}{dt}[\dot{x} \ \dot{y} \ \dot{z}]^T$  being the time derivative of the velocity vector observed from the fixed frame and expressed as:

$$\frac{d}{dt} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \frac{d}{dt} \begin{bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{z}_b \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{z}_b \end{bmatrix} \quad (19)$$

Combining the equations (18) and (19) lead to the dynamics of the velocity states as below:

$$\frac{d}{dt} \begin{bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{z}_b \end{bmatrix} = \begin{bmatrix} r\dot{y}_b - q\dot{z}_b + a_x - g \sin \theta \\ p\dot{z}_b - r\dot{x}_b + a_y + g \sin \phi \cos \theta \\ q\dot{x}_b - p\dot{y}_b + a_z + g \cos \phi \cos \theta \end{bmatrix} \quad (20)$$

Similarly, using the rotation matrix derived above, the relationship between the angular rates and the time derivatives of the Euler angles are given by:

$$\frac{d}{dt} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ q \cos \phi - r \sin \phi \\ (q \sin \phi + r \cos \phi) \sec \theta \end{bmatrix} \quad (21)$$

Based on (20), (21) and assuming that the bias parameters follow a first-order Gauss-Markov noise model, the state dynamic equations can be written as:

$$X_{k+1/k} = f(X_{k/k}, u_k) + w_k \quad (22)$$

with  $X_{k+1/k}$  being the estimated state vector at time  $k+1$  given observations up to time  $k$ . The system is nonlinear and the jacobians should be calculated as in (10):

$$F_k = \begin{pmatrix} 0 & r_k & -q_k & 0 & -g c\theta_k & 0 \\ -r_k & -p_k & 0 & -g c\theta_k s\phi_k & -g c\phi_k s\theta_k & 0 \\ q_k & -p_k & 0 & -g c\theta_k s\phi_k & -g c\phi_k s\theta_k & 0 \\ 0 & 0 & 0 & q_k c\phi_k \tan \theta_k & \alpha_k & 0 \\ 0 & 0 & 0 & -r_k c\phi_k - q_k s\phi_k & 0 & 0 \\ 0 & 0 & 0 & \beta_k & \gamma_k & 0 \end{pmatrix} \quad (23)$$

with  $\alpha_k = (r_k c\phi_k + q_k s\phi_k)(\tan \theta_k^2 + 1)$ ,  $\beta_k = (q_k c\phi_k - r_k s\phi_k)/c\theta_k$ ,  $\gamma_k = s(\theta_k (r_k c\phi_k + q_k s\phi_k))/c\theta_k^2$ .

$$B_k = \begin{pmatrix} 0 & 0 & 1 & 0 & -\dot{z}_{b_k} & \dot{y}_{b_k} \\ 0 & 1 & 0 & \dot{z}_{b_k} & 0 & -\dot{x}_{b_k} \\ 0 & 0 & 1 & -\dot{y}_{b_k} & \dot{x}_{b_k} & 0 \\ 0 & 0 & 0 & 1 & s\phi_k \tan \theta_k & c\phi_k \tan \theta_k \\ 0 & 0 & 0 & 0 & c\phi_k & -s\phi_k \\ 0 & 0 & 0 & 0 & s\phi_k/c\theta_k & c\phi_k/c\theta_k \end{pmatrix} \quad (24)$$

The prediction is then achieved as detailed in (4) and (5).

### 3.2.2 Update Step

To deal with the accelerometer and gyroscope inaccuracies, correction measurements are obtained from the GPS, magnetometer and lidar sensors. The measurement vector is given by:

$$z_k = [v_{x_k} \ v_{y_k} \ v_{z_k} \ M_{x_k} \ M_{y_k} \ M_{z_k} \ \dot{z}_{L_k}]^T \quad (25)$$

where  $v_{x_k}$ ,  $v_{y_k}$ ,  $v_{z_k}$  are the GPS velocity measurements,  $M_{x_k}$ ,  $M_{y_k}$ ,  $M_{z_k}$  are the magnetometer measurements in the body frame and  $\dot{z}_{L_k}$  is the lidar altitude measurement.

The nonlinear observation equations of the magnetometer are given below:

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \mathcal{R}_N^B \cdot \begin{bmatrix} M_{e_x} \\ M_{e_y} \\ M_{e_z} \end{bmatrix} \quad (26)$$

where  $[M_{e_x} \ M_{e_y} \ M_{e_z}]^T$  is the local magnetic vector expressed in the navigation frame. The same transformation is needed for the GPS measurements. The observation model is thus nonlinear and given by:

$$h(x_k) = \begin{bmatrix} c\psi_k c\theta_k \dot{x}_{b_k} + \delta_{x\theta_k} \dot{y}_{b_k} + \delta_{x\psi_k} \dot{z}_{b_k} \\ s\psi_k c\theta_k \dot{x}_{b_k} + \delta_{y\theta_k} \dot{y}_{b_k} + \delta_{y\psi_k} \dot{z}_{b_k} \\ -s\theta_k \dot{x}_{b_k} + c\theta_k s\phi_k \dot{y}_{b_k} + c\theta_k c\phi_k \dot{z}_{b_k} \\ c\psi_k c\theta_k M_{e_{x_k}} + \delta_{x\theta_k} M_{e_{y_k}} + \delta_{x\psi_k} M_{e_{z_k}} \\ s\psi_k c\theta_k M_{e_{x_k}} + \delta_{y\theta_k} M_{e_{y_k}} + \delta_{y\psi_k} M_{e_{z_k}} \\ -s\theta_k M_{e_{x_k}} + c\theta_k s\phi_k M_{e_{y_k}} + c\theta_k c\phi_k M_{e_{z_k}} \\ \dot{z}_{b_k} \end{bmatrix} \quad (27)$$

with  $\delta_{x\theta_k} = -s\psi_k c\phi_k + c\psi_k s\theta_k s\phi_k$ ,  $\delta_{x\psi_k} = s\psi_k s\phi_k + c\psi_k s\theta_k c\phi_k$ ,  $\delta_{y\theta_k} = c\psi_k c\phi_k + s\psi_k s\theta_k s\phi_k$ ,  $\delta_{y\psi_k} = -c\psi_k s\phi_k + s\psi_k s\theta_k c\phi_k$ , and the update step is achieved according to (6)-(9).

For the implementation of the information filter on the UAV system, the parameter  $w$  in (6) and (7) will be equal to 3 and the vector  $z_k^l$  in (9) corresponds to  $[v_{x_k} \ v_{y_k} \ v_{z_k} \ 0 \ 0 \ 0]^T$  for the GPS sensor,  $[0 \ 0 \ 0 \ M_{x_k} \ M_{y_k} \ M_{z_k} \ 0]^T$  for the magnetometer and  $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ z_{L_k}]^T$  for the lidar.

### 3.3 Fault Tolerant Component

In this section, we present how sensors faults are detected and the system recovers using the fault tolerant component. We first detail the Error Detection Module, and then the Error Identification and Recovery Module.

#### 3.3.1 Error Detection Module

As described above and illustrated in Fig. 2, the first step in the fault tolerant component is the error detection module EDM. It compares the estimated states of the two Information Kalman filters (position, attitude and altitude). For this purpose we calculate the Bhattacharyya Distance  $BD_{(DF1, DF2)}$  between the outputs of the two data fusion blocks DF1 and DF2. At each time instant  $k$ , the models estimated from the data fusion blocks DF1 and DF2 are described by their means  $q_k^{[DF1]}$  and  $q_k^{[DF2]}$  and their covariance matrices  $\sigma_k^{[DF1]}$  and  $\sigma_k^{[DF2]}$  respectively, the residual is then expressed by:

$$BD_{(DF1, DF2)} = \frac{1}{8} (q_k^{[DF1]} - q_k^{[DF2]})^T \sigma^{-1} (q_k^{[DF1]} - q_k^{[DF2]}) + \frac{1}{2} \log \left( \frac{|\sigma|}{|\sigma_k^{[DF1]}| \cdot |\sigma_k^{[DF2]}|} \right) \quad (28)$$

Then this residual is compared to a certain threshold to detect the existence of faulty sensors. The decision is taken among two hypotheses:  $H_1$ , the hypothesis of existence of sensors faults in the UAV system and  $H_0$ , the hypothesis of absence of faults.

Let us define the following probabilities:

- $P_D = p(u_1/H_1) = \int_{Th_{Det}}^{\infty} p(x/H_1)dx$ : Detection probability, it is the probability of selecting  $H_1$  when  $H_1$  is true,
- $P_F = p(u_1/H_0) = \int_{Th_{Det}}^{\infty} p(x/H_0)dx$ : False alarm probability, it is the probability of selecting  $H_1$  when  $H_0$  is true,
- $P_0 = P(H_0)$  and  $P(H_1) = 1 - P_0$ : Prior probabilities,

with  $u_i$  being defined as the action of choosing the hypothesis  $H_i$ , and  $p(x/H_i)$  is the probability density function of the variable  $x$ , which corresponds to the Bhattacharyya distance  $BD_{(DF1,DF2)}$  in our case.

Different techniques were proposed and applied in the literature to fix and choose the thresholds. In Bayesian optimisation, the threshold selection followed the minimization of a mean risk function. Using the Neyman-Pearson criterion, it consists of optimizing the probability of detection while minimizing the false alarm probability [46]. Other techniques consider the optimisation problem in terms of quantity of information by minimizing the conditional entropy of the decision hypothesis or maximizing the mutual information [47], [48]. The technique adopted in this work is based on the Bhattacharyya criterion developed in [42]. The optimisation problem is formulated using the information surprise that evaluates the variation between the prior and the posterior distributions [45].

The Bhattacharyya criterion is defined as:

$$B_c = - \sum_j \left( \log \sum_i \sqrt{p(H_i/u_j)p(H_i)} \right) \quad (29)$$

Using Bayes theorem, it can be formulated as:

$$B_c = - \left[ \log \left[ p(H_0) \sqrt{\frac{p(u_0/H_0)}{p(u_0)}} + p(H_1) \sqrt{\frac{p(u_0/H_1)}{p(u_0)}} \right] \right. \\ \left. + \log \left[ p(H_0) \sqrt{\frac{p(u_1/H_0)}{p(u_1)}} + p(H_1) \sqrt{\frac{p(u_1/H_1)}{p(u_1)}} \right] \right] \quad (30)$$

Knowing that:

$$P(u_0) = P_0(1 - P_F) + (1 - P_0)(1 - P_D) \\ P(u_1) = P_0P_F + (1 - P_0)P_D \quad (31)$$

The Bhattacharyya criterion can be expressed as:

$$B_c = - \left[ \log[\alpha_0 + \beta_0] + \log[\alpha_1 + \beta_1] - \frac{1}{2} \log[\gamma_0] \right. \\ \left. - \frac{1}{2} \log[\gamma_1] \right] \quad (32)$$

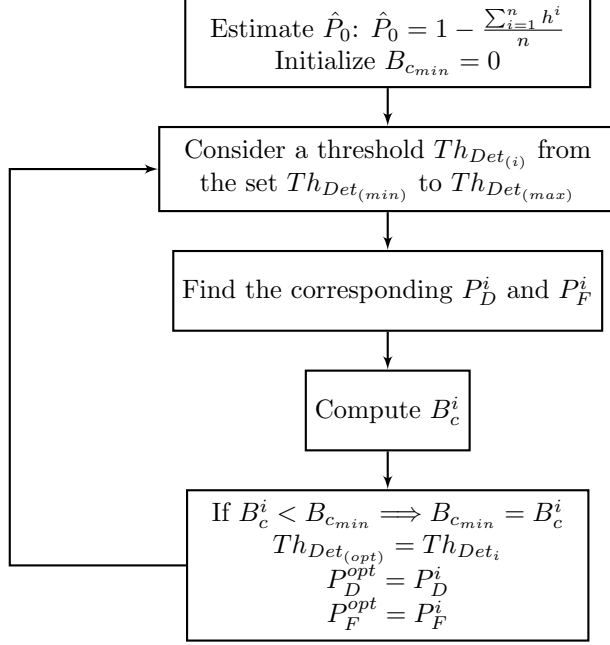


Figure 3: Residual threshold optimization

with:

$$\begin{aligned}
 \alpha_0 &= P_0 \sqrt{1 - P_F}, & \beta_0 &= (1 - P_0) \sqrt{1 - P_D} \\
 \alpha_1 &= P_0 \sqrt{P_F}, & \beta_1 &= (1 - P_0) \sqrt{P_D} \\
 \gamma_0 &= \frac{\alpha_0^2}{P_0} + \frac{\beta_0^2}{1 - P_0}, & \gamma_1 &= \frac{\alpha_1^2}{P_0} + \frac{\beta_1^2}{1 - P_0}
 \end{aligned} \tag{33}$$

According to (32), the Bhattacharyya criterion depends on the detection and false alarm probabilities  $P_D$  and  $P_F$ , and consequently depends on the choice of the detection threshold  $Th_{Det}$ .

The main objective is to determine the threshold that maximizes the information surprise which corresponds to the minimization of the Bhattacharyya criterion. The algorithm is based on the exhaustive search problem-solving technique. It starts by defining a threshold search interval, and then calculating the  $B_c$  value across this interval to find the threshold that leads to the minimum  $B_c$ . To estimate the *a priori* probability  $P_0$ , we assume having random samples of hypotheses  $h_i$ :  $h = (h^1, h^2, \dots, h^n)$ , where  $h^i = 0$  if the hypothesis  $H_0$  is selected, and  $h^i = 1$  if the hypothesis  $H_1$  is selected. The probability  $P_0$  is estimated as [44]:

$$\hat{P}_0 = 1 - \frac{\sum_{i=1}^n h^i}{n} \tag{34}$$

Fig. 3 summarizes the threshold optimization algorithm where  $Th_{Det(min)}$  and  $Th_{Det(max)}$  are fixed arbitrary based on the residuals sizes. Fixing these margins is simpler than fixing empirically the optimal threshold in traditional



techniques as a better choice of these two values decreases the convergence time to the optimal threshold but do not change this optimal value given by the algorithm when these values are big enough.

The choice of the probability  $P_0$  should be suitable to the application and to the frequency of occurrence of defects. It is normally estimated based on the historical system behavior as a priori knowledge of the probable fault cases is needed to guarantee their detection. Note that this is also the case when dealing with traditional techniques based on fixing thresholds empirically.

### 3.3.2 Error Identification and Recovery Module

Once an error is detected, the faulty component should be isolated and then the correct output of the system estimated. The procedure starts by comparing the probability distributions of the prediction and correction steps of each data fusion block to evaluate the information surprise [45] provided by the correction measurements. This is done by computing the two Bhattacharyya distances as described in the equations below:

$$BD_{DF1} = \frac{1}{8} (q_{k/k-1}^{[DF1]} - q_{k/k}^{[DF1]})^T \sigma^{-1} (q_{k/k-1}^{[DF1]} - q_{k/k}^{[DF1]}) + \frac{1}{2} \log \left( \frac{|\sigma|}{\sqrt{|\sigma_{k/k-1}^{[DF1]}| \cdot |\sigma_{k/k}^{[DF1]}|}} \right) \quad (35)$$

$$BD_{DF2} = \frac{1}{8} (q_{k/k-1}^{[DF2]} - q_{k/k}^{[DF2]})^T \sigma^{-1} (q_{k/k-1}^{[DF2]} - q_{k/k}^{[DF2]}) + \frac{1}{2} \log \left( \frac{|\sigma|}{\sqrt{|\sigma_{k/k-1}^{[DF2]}| \cdot |\sigma_{k/k}^{[DF2]}|}} \right) \quad (36)$$

If no fault exists, the *a priori* ( $k/k-1$ ) and *a posteriori* ( $k/k$ ) distributions should be convergent, and the information surprise must not exceed a predetermined threshold that depends on the number of observations, the size of the state vector, etc. On the other hand, if the residual  $BD_{DF1}$  exceeds a threshold  $Th_{DF1}$ , an error is detected in the first branch (in the block sensors  $S1$  or  $S2$ ). If the residual  $BD_{DF2}$  exceeds a threshold  $Th_{DF2}$ , an error is detected in the second branch (in the block sensors  $S3$  or  $S4$ ).

Then we proceed by comparing the outputs of similar sensors. We first compare the output of the sensor block  $S_1$  with that of the sensor block  $S_3$  by comparing the distance  $D_{(S1,S3)}$  to a threshold  $Th_{13}$ .  $D_{(S1,S3)}$  is defined as:

$$D_{S1,S3} = \sqrt{(\phi_1 - \phi_2)^2 + (\theta_1 - \theta_2)^2 + (\psi_1 - \psi_2)^2} \quad (37)$$

with  $\phi_i$ ,  $\theta_i$  and  $\psi_i$  being the Euler angles measured by the Inertial Measurement Unit  $IMU_i$ .

Second, we compare the output of sensor block  $S_2$  with that of the sensor block  $S_4$  by comparing the distances  $D_{(S2,S4)a}$ ,  $D_{(S2,S4)b}$  and  $D_{(S2,S4)c}$  to thresholds  $Th_{24a}$ ,  $Th_{24b}$  and  $Th_{24c}$ .  $D_{(S2,S4)a}$ ,  $D_{(S2,S4)b}$  and  $D_{(S2,S4)c}$  are defined as:

$$\begin{aligned} D_{(S2,S4)a} &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\ D_{(S2,S4)b} &= |z_1 - z_2| \\ D_{(S2,S4)c} &= |\psi_1 - \psi_2| \end{aligned} \quad (38)$$



Figure 4: Experimental quadrotor

with  $x_i$  and  $y_i$  being the position measured by the  $GPS_i$ ,  $z_i$  the altitude measured by the Lidar <sub>$i$</sub>  and  $\psi_i$  the yaw angle measured by the Magnetometer Mag <sub>$i$</sub> .

The diagnosis and fault-tolerant algorithm is detailed in Algorithm 1.

## 4 Experimental Validation

In this section we present the experimental validation of the proposed fault tolerant architecture using real data and fault injection. We start first by describing the experimental environment followed by the obtained results.

The data acquisition was conducted outdoor using a Tarot650 quadrotor built at the Heudiasyc Laboratory and shown in Fig. 4. The UAV is equipped with the different sensors detailed in Section 3. The experimental data included online fault injection, then was processed offline using a Matlab environment. We consider in this section the results of two fault injections: an additive fault on the first GPS sensor GPS1, and an additive fault on Lidar1. The covariance matrices of the state and measurement models are selected based on empirical tuning as:

$$Q_k = \begin{bmatrix} 10^{-4} \times I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 10^{-6} \times I_{3 \times 3} \end{bmatrix}, R_k = 0.5 \times I_{7 \times 7}$$

### 4.1 Additive fault on GPS1

First, an additive fault on GPS1 output is considered. This type of fault emulates a typical jump in the GPS position resulting from satellite signal rebounds. Note that using redundant GPS sensors of the same type would usually not tolerate this fault as it has a common cause. Between the instants  $t_1 = 7.26s$  and  $t_2 = 15.8s$ , a jump of  $1m$  is added in the  $x_1$  and  $y_1$  direction and the new

---

**Algorithm 1** Fault-Tolerant Algorithm

---

**Data:**  $Th_{Det}, Th_{DF1}, Th_{DF2}, Th_{13}, Th_{24a}, Th_{24b}, Th_{24c}$ **begin**

```
    Calculate  $BD_{(DF1,DF2)}, BD_{DF1}, BD_{DF2}$  Calculate  $D_{(S1,S3)}, D_{(S2,S4)a},$   
     $D_{(S2,S4)b}, D_{(S2,S4)c}$  if  $(BD_{(DF1,DF2)} > Th_{Det})$  then  
    | /* A fault is detected */ if  $(BD_{DF1} > Th_{DF1})$  then  
    | | /* The fault is in the first branch */ if  $(D_{(S1,S3)} > Th_{13})$  then  
    | | | S1 is faulty, Isolate  $EIKF1$   
    | else  
    | | if  $(D_{(S2,S4)a} > Th_{24a}) || (D_{(S2,S4)b} > Th_{24b}) || (D_{(S2,S4)c} > Th_{24c})$   
    | | then  
    | | | S2 is faulty, Isolate  $EIKF1$   
    | | else  
    | | | /* cannot identify the faulty sensor, keep running without  
    | | | modifications */  
    | | end  
    | end  
    | else  
    | | if  $(BD_{DF2} > Th_{DF2})$  then  
    | | | /* The fault is in the second branch */  
    | | else  
    | | | /* cannot identify the faulty branch, keep running without mod-  
    | | | ifications */ if  $(D_{(S1,S3)} > Th_{13})$  then  
    | | | | S3 is faulty, Isolate  $EIKF2$   
    | | else  
    | | | if  $(D_{(S2,S4)a} > Th_{24a}) || (D_{(S2,S4)b} > Th_{24b}) || (D_{(S2,S4)c} >$   
    | | |  $Th_{24c})$  then  
    | | | | S4 is faulty, Isolate  $EIKF2$   
    | | | else  
    | | | | /* cannot identify the faulty sensor, keep running without  
    | | | | modifications */  
    | | | end  
    | | end  
    | end  
    | end  
    | else  
    | | /* No fault is detected */  
    | end  
end
```

---

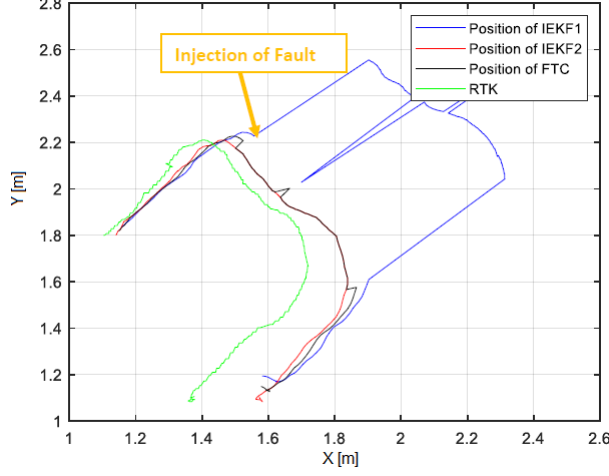


Figure 5: Real and estimated trajectories after an additive fault on GPS1

measurements  $x'_1$  and  $y'_1$  are then given below as:

$$\begin{aligned} x'_1(t_k) &= x_1(t_k) + 1 \\ y'_1(t_k) &= y_1(t_k) + 1 \end{aligned} \quad (39)$$

The UAV followed a given U-shape reference trajectory, presented in green by the RTK measurement (considered as ground truth) in Fig. 5. The trajectories estimated by the first and second Information filters in presence of faults on GPS1 data are presented respectively in blue and red in Fig. 5. It is clear how the position estimated from the first filter deviates significantly from all the other systems.

Fig. 6 gives the Bhattacharyya distance  $BD_{(DF1,DF2)}$  used for the fault detection. It presents a jump at the fault injection time indicating a drift between the outputs of the data fusion blocks. It should be compared to the threshold presented in black in Fig. 6 that corresponds to the minimum Bhattacharyya criterion obtained at the value 0.2 when  $P_0 = 0.6154$  (Fig. 7). It is shown that the system detects an error at time  $t_{Det} = 7.29s$ .

Once an error is detected, the faulty sensor should be isolated by applying Algorithm 1. Therefore, the set of residuals  $BD_{DF1}$ ,  $BD_{DF2}$ ,  $D_{(S1,S3)}$ ,  $D_{(S2,S4)a}$ ,  $D_{(S2,S4)b}$  and  $D_{(S2,S4)c}$  must be compared (Fig. 8 and 9). To fix the thresholds, an identical procedure to the one followed previously for the  $BD_{(DF1,DF2)}$  test is done for both  $BD_{DF1}$  and  $BD_{DF2}$ . Fig 8 shows how that a posteriori distributions of the second data fusion block are convergent and the residual  $BD_{DF1}$  exceeds its threshold value after the fault has been detected, indicating an error in the first branch. Similar reasoning on residual  $D_{(S1,S3)}$  shown in Fig. 9 enables to decide that the GPS1 is the faulty sensor. It should be noted that in Fig. 8, the fault is injected at time  $t = 7.26s$ . Before this instant, the

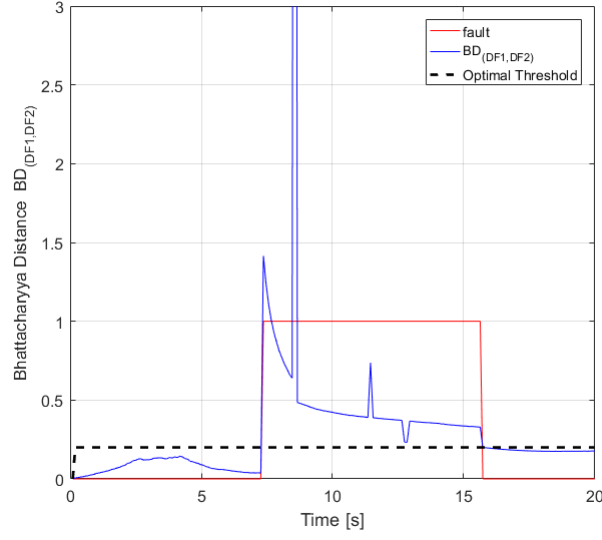


Figure 6: The  $BD_{(DF1, DF2)}$  for the fault detection after GPS1 fault

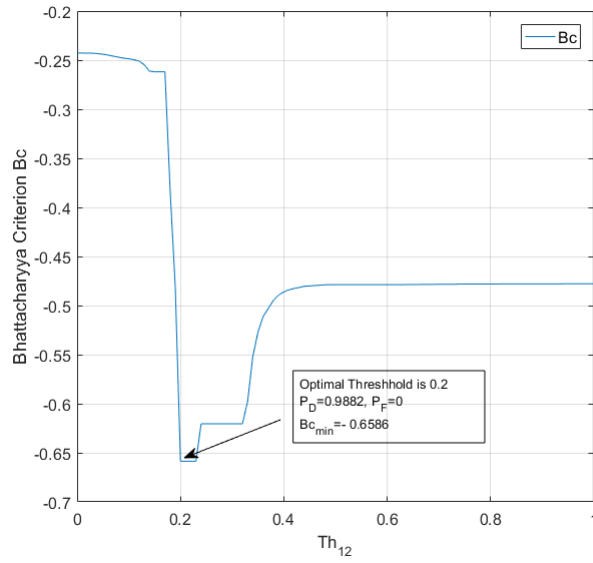


Figure 7: Optimal threshold value  $Th_{12}$  using the Bhattacharyya Criterion for  $P_0 = 0.6154$ .

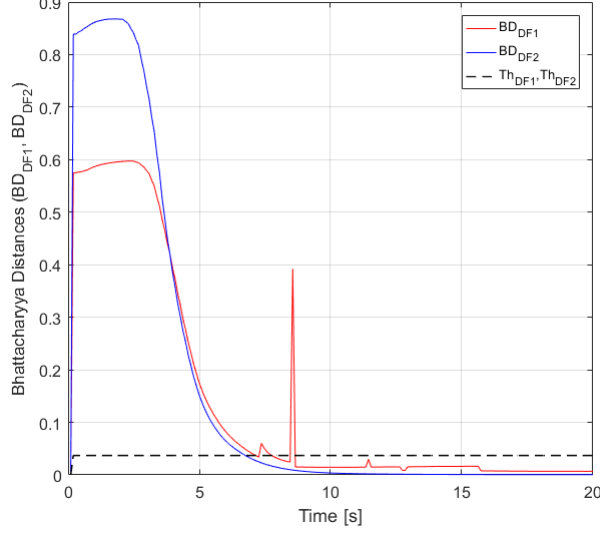


Figure 8: The residual test to isolate the failed branch after GPS1 fault

high values of the distances  $BD_{DF1}$  and  $BD_{DF2}$  are due to the time needed for the filters to converge. Practically, enough time should be provided to initialize the filter and ensure its convergence before launching the diagnosis algorithm. The black curve in Fig. 5 presents the output of the data fusion mechanism of the UAV. From an average of the outputs of IEKF1 and IEKF2, it goes to the output of IEKF2 after the exclusion of the first Information filter. To evaluate the accuracy of the fault tolerant controller, the position errors along the  $x$  and  $y$  axes are shown in Fig. 10. Note that the parts of the plots circled with red are outliers (false measurement acquired from real time operating GPSs).

## 4.2 Additive fault on Lidar2

During this experiment, the UAV is requested to perform a hovering flight at a  $3m$  altitude. At time  $t_1 = 67s$ , an additive fault of  $1m$  is added to the Lidar2 output as below:

$$z'_2(t_k) = z_2(t_k) + 1 \quad (40)$$

$$t_k \geq t_1$$

There is no need for an additional ground truth in this case since the precision of the Lidar installed on the UAV is below  $15cm$ , thus it is considered as a ground truth. The estimated altitudes by the two Information filters in the case of faults on the Lidar2 data are presented respectively in blue and red in Fig. 11. We can clearly see that the altitude estimated from the first filter deviates significantly from the one estimated from the second filter. The black curve presents the altitude of the UAV given as output of the data fusion mechanism.

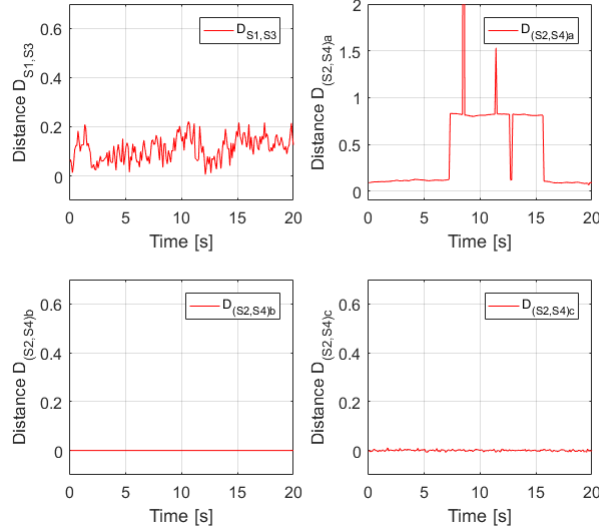


Figure 9: The different residual tests after GPS1 fault

It goes from an average of the altitudes of IEKF1 and IEKF2 to the value of IEKF1 once IEKF2 has been excluded as a system recovery.

This architecture could be easily adapted to detect and isolate software fault as already considered in [7]. The software faults that are generally introduced during the development stage are of human origin. Note however that software faults detection problem is out of scope of this paper.

When compared to similar architectures applied on unmanned aerial vehicles, the one presented in this paper presents the following advantages:

- Compared with [35], the proposed architecture overcomes the disadvantages of using a dynamic model to isolate the faulty sensor which requires to identify and tune the model parameters after each failure occurrence to avoid model drift.
- Compared with [32], less redundant sensors and Kalman filter blocks are needed in this architecture. The fault tolerant methods based on majority-voting methods or weighted-mean methods requiring at least three similar or diversified sensors measuring a certain state variable as in [32] are still used in many existing avionic systems. However, they are expensive and they can result in a significant increase in mass.
- Compared with [33,34], the fault-tolerant aspect of the sensor fusion algorithm is validated with sensor failures injected during real flight and not by superimposing simulated sensor failures on the actual flight data.

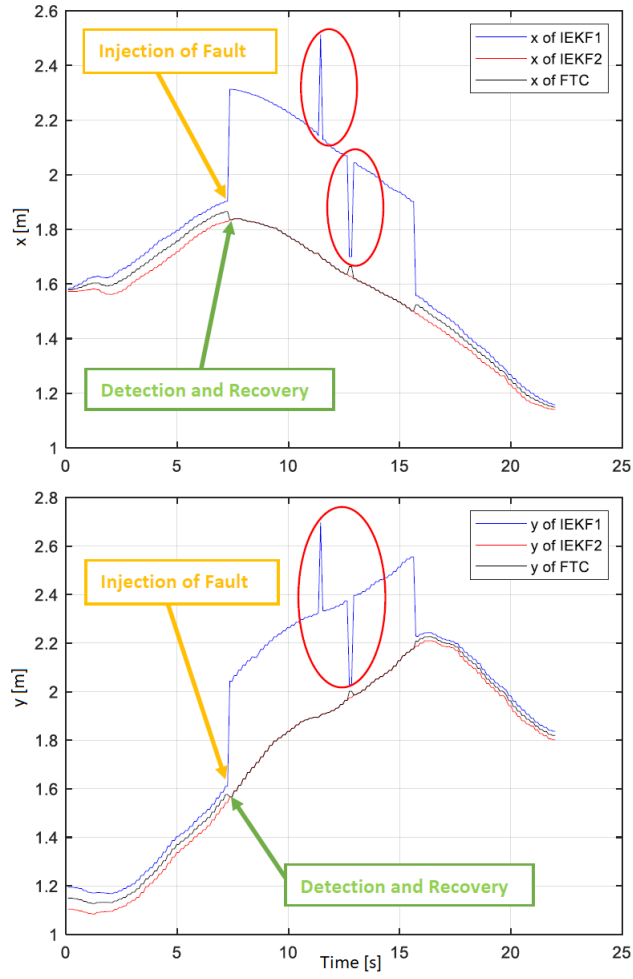


Figure 10: The position errors along the x and y axes



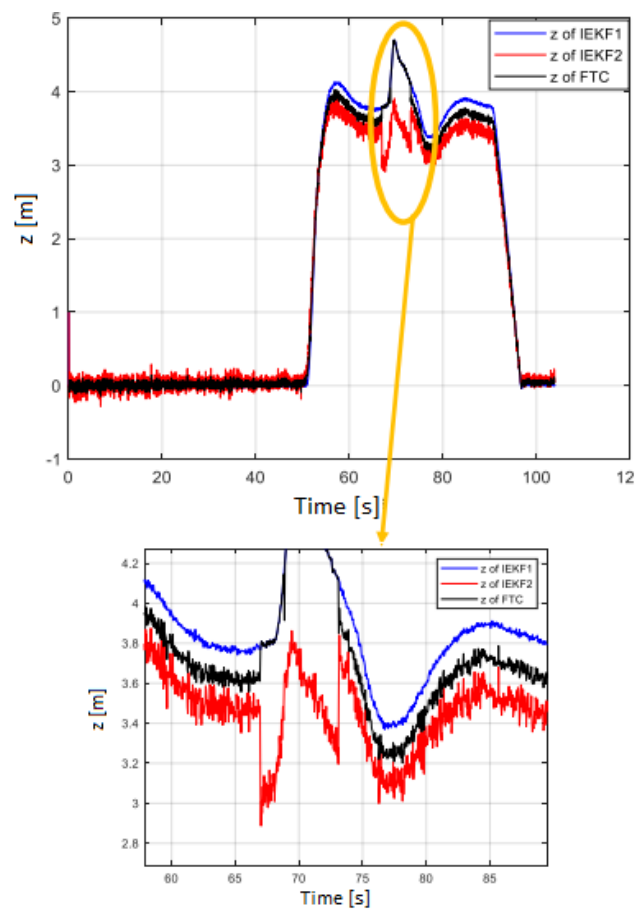


Figure 11: Estimated altitudes after an additive fault on Lidar2

## 5 Conclusion

In this work, a fault tolerant multi-sensor fusion architecture based on the informational form of the Kalman filter and the information theory concept is developed and applied to a UAV system. The fault detection and isolation steps were formulated using the Bhattacharyya distances between the outputs of the Information filters and between the predicted and the updated distributions of each IF respectively. The BD used in residual tests presents many advantages when compared to other metrics such as the Mahalanobis distance or the Kullback–Leibler metric since it increases with the difference between the variances of the distributions under study and is symmetrical. However, the latency introduced by the inertia of this distance when applied to measurement streams and its effect on the detection delay have not been considered and should be evaluated in future studies.

An optimal thresholding method using the Bhattacharyya criterion  $B_c$  is adopted instead of the traditional techniques based on heuristic methods to fix the false alarm probability. The computed threshold is affected by the historical system behavior and the prior probability of the hypothesis.

In future work, this architecture should be developed and extended to have the ability to tolerate more than one sensor fault, in addition to actuators faults. Its performance in case of small additive and drift faults should be further evaluated and could be improved by comparing the outputs of redundant sensors.

## Acknowledgements

The authors would like to thank the Heudiasyc Laboratory for providing real data used for validation.

## References

- [1] H. Durrant-Whyte and T. Henderson, *Multisensor Data Fusion*. Springer Handbook of Robotics, 2016, pp. 867–896.
- [2] B. Khaleghi, A. Khamis, F. O. Karray and S. N. Razavi, “Multisensor data fusion: A review of the state-of-the-art,” *Information Fusion*, vol. 14, no. 1, pp. 28–44, Jan, 2013.
- [3] D. Hall and J. Llinas, “An introduction to multisensor data fusion,” *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, Jan. 1997.
- [4] M. Kumar, D. Garg and R. Zachery, “A Generalized Approach for Inconsistency Detection in Data Fusion from Multiple Sensors,” *American Control Conference (ACC)*, Minneapolis, MN, USA, 2006, pp. 2078–2083.
- [5] P. Smets, “Analyzing the combination of conflicting belief functions,” *Information fusion*, vol. 8, no. 4, pp. 387–412, Oct. 2007.

- [6] Y. Zhu, E. Song, J. Zhou and Z. You, “Optimal Dimensionality Reduction of Sensor Data in Multisensor Estimation Fusion,” *IEEE Transactions on Signal Processing*, vol. 53, no. 5, pp. 1631–1639, May 2005.
- [7] K. Bader, B. Lussier and W. Schon, “A fault tolerant architecture for data fusion: A real application of Kalman filters for mobile robot localization,” *Robotics and Autonomous Systems*, vol. 88, pp. 11–23, Feb. 2017.
- [8] B. Du, Z. Shi, J. Song, H. Wang and L. Han, “A Fault-Tolerant Data Fusion Method of MEMS Redundant Gyro System Based on Weighted Distributed Kalman Filtering,” *Micromachines*, vol. 5, no. 10, Apr. 2019.
- [9] L. Shu-qing and Z. Sheng-xiu, “A Congeneric Multi-Sensor Data Fusion Algorithm and Its Fault-Tolerance,” *International Conference on Computer Application and System Modeling*, Taiyuan, China, 2010, pp. 339–342.
- [10] A. Avizienis, J.-C. Laprie, B. Randell and C. Landwehr, “Basic Concepts and Taxonomy of Dependable and Secure Computing,” *IEEE Transactions on Dependable And Secure Computing*, vol. 1, no. 1, pp. 11–33, March 2004.
- [11] H. Darvishi, D. Ciunzo, E.R. Eide and P. Salvo Rossi, “Sensor-Fault Detection, Isolation and Accommodation for Digital Twins via Modular Data-Driven Architecture,” *IEEE Sensors Journal*, 2021.
- [12] L. Liu, G. Han, Y. He, and J. Jiang, “Fault-Tolerant Event Region Detection on Trajectory Pattern Extraction for Industrial Wireless Sensor Networks,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2072–2080, March 2020.
- [13] D.J. Allerton and H. Jia, “A Review of Multisensor Fusion Methodologies for Aircraft Navigation Systems,” *Journal of Navigation*, vol. 58, no. 3, pp. 405–417, 2005.
- [14] J. Simanek, V. Kubelka and M. Reinstein, “Improving multi-modal data fusion by anomaly detection”, *Autonomous Robots*, vol. 39, pp. 139–154, 2015.
- [15] L. Hachemi, M. Guiatni and A. Nemra, “Fault Diagnosis and Reconfiguration for Mobile Robot Localization Based on Multi-Sensors Data Fusion,” *Unmanned Systems*, 2021.
- [16] B. Kellalib, N. Achour, V. Coelen and A. Nemra, “Towards simultaneous localization and mapping tolerant to sensors and software faults: Application to omnidirectional mobile robot,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 2021.
- [17] V. Kubelka, L. Oswald, F. Pomerleau, F. Colas, T. Svoboda and M. Reinstein, “Robust Data Fusion of Multimodal Sensory Information for Mobile Robots,” *Journal of Field Robotics*, vol. 32, no. 4, pp. 447–473, 2014.

- [18] N. Sadeghzadeh-Nokhodberiz and J. Poshtan, "Distributed Interacting Multiple Filters for Fault Diagnosis of Navigation Sensors in a Robotic System," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1383–1393, 2017.
- [19] V. Ricquebourg, M. Delafosse, L. Delahoche, B. Marhic, A. Jolly-Desodt and D. Menga, "Fault detection by combining redundant sensors: a conflict approach within the TBM framework," *Cognitive Systems with Interactive Sensors*, COGIS, 2007.
- [20] F. Delmotte and G. Gacquer, "Detection of defective sources with belief functions," *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Malaga, Spain, 2008, pp. 337–344.
- [21] F. Delmotte, "Detection of defective sources in the setting of possibility theory," *Fuzzy Sets Syst.*, vol. 158, no. 5, pp. 555–571, 2007.
- [22] L. Shu-qing and Z. Sheng-xiu, "A congeneric multi-sensor data fusion algorithm and its fault-tolerance," *International Conference on Computer Application and System Modeling*, ICCASM, 2010.
- [23] D.J. Allerton and H. Jia, "Distributed data fusion algorithms for inertial network systems," *Radar Sonar Navig. IET*, vol. 2, no. 1, pp. 51–62, 2008.
- [24] M. Silvagni, A. Tonoli, E. Zenerino and M. Chiaberge, "Multipurpose UAV for search and rescue operations in mountain avalanche events," *Geomatics, Natural Hazards and Risk*, vol. 8, no. 1, pp. 18–33, May 2017.
- [25] M Saied, T. Mahairy, C. Francis, H. Shraim, H. Mazeh and M. El Rafei, "Differential Flatness-Based Approach for Sensors and Actuators Fault Diagnosis of a Multirotor UAV," *IFAC-PapersOnLine*, vol. 52, no. 16, pp. 831–836, 2019.
- [26] R. Avram, X. Zhang and J. Muse, "Quadrotor Sensor Fault Diagnosis with Experimental Results," *Journal of Intelligent & Robotic Systems*, vol. 86, pp. 115–137, April 2017.
- [27] R. Sabatini, S. Ramasamy and A. Gradi, "Multisensor data fusion for unmanned aerial vehicles navigation and guidance," *International Conference and Exhibition Mechanical & Aerospace Engineering*, Philadelphia, USA, 2014.
- [28] R. Sabatini, S. Ramasamy, A. Gardi and L. R. Salazar, "Low-cost Sensors Data Fusion for Small Size Unmanned Aerial Vehicles Navigation and Guidance," *International Journal of Unmanned Systems Engineering*, vol. 1, no. 3, 2013.

- [29] X. Yang, L. Mejias and M. Garratt, “Multi-Sensor Data Fusion for UAV Navigation during Landing Operations,” *Australian Conference on Robotics and Automation*, Melbourne, Australia, pp. 1–10, 2011.
- [30] W. Zhang, Y. Ning and C. Suo, “A Method Based on Multi-Sensor Data Fusion for UAV Safety Distance Diagnosis,” *Electronics*, vol. 8, no. 12, Dec. 2019.
- [31] J. Garcia, J.M. Molina and J. Trincado, “Real evaluation for designing sensor fusion in UAV platforms,” *Information Fusion*, vol. 63, pp. 136–152, Dec. 2020.
- [32] K. Geng and N.A. Chulin, “Applications of multi-height sensors data fusion and fault-tolerant Kalman filter in integrated navigation system of UAV,” *Procedia Computer Science*, vol. 103, pp. 231–238, 2017.
- [33] Y. Gu, J. N. Gross, M. B. Rhudy and K. Lassak, “A Fault-Tolerant Multiple Sensor Fusion Approach Applied to UAV Attitude Estimation,” *International Journal of Aerospace Engineering*, 2016.
- [34] J. Gross, *Sensor Fusion Based Fault-Tolerant Attitude Estimation Solutions for Small Unmanned Aerial Vehicles*, PhD Thesis, West Virginia University, 2011.
- [35] H. Hamadi, *Fault-tolerant control of a multirotor unmanned aerial vehicle under hardware and software failures*, PhD Thesis, University of Technology of Compiegne, 2020.
- [36] S. Amari, “ $\alpha$ -Divergence Is Unique, Belonging to Both  $f$ - Divergence and Bregman Divergence Classes,” *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 4925–4931, Oct. 2009.
- [37] A. Zeroual, F. Harrou, Y. Sun and N. Messai, “Integrating Model Based Observer and Kullback–Leibler Metric for Estimating and Detecting Road Traffic Congestion,” *IEEE Sens. J.*, vol. 18, no. 20, pp. 8605–8616, Aug. 2018.
- [38] A. Bhattacharyya, “On a measure of divergence between two statistical populations defined by their probability distributions,” *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.
- [39] B. Abci, M. El Badaoui El Najjar, V. Cocquempot and G. Dherbomez, “An informational approach for sensor and actuator fault diagnosis for autonomous mobile robots,” *Journal of Intelligent & Robotic Systems*, vol. 99, pp. 387–406, Aug. 2020.
- [40] S. Srimani, K. Ghosh and H. Rahaman, “Parametric Fault Detection in Analog Circuits: A Statistical Approach,” *Asian Test Symposium (ATS)*, Hiroshima, Japan, pp. 275–280, 2016.

- [41] S. Srimani, M.K. Parai, K. Ghosh and H. Rahaman, “Parametric fault detection of analog circuits based on Bhattacharyya measure,” *Analog Integrated Circuits and Signal Processing*, vol. 93, pp. 477–488, Dec. 2017.
- [42] B. Abci, J. Nader, M. El Badaoui El Najjar and V. Cocquempot, “Fault-tolerant multi-sensor fusion and thresholding based on the Bhattacharyya distance with application to a multi-robot system,” *European Workshop on Advanced Control and Diagnosis*, Bologna, Italy, 2019.
- [43] J. Harmouche, C. Delpha and D. Diallo, “Improved Fault Diagnosis of Ball Bearings Based on the Global Spectrum of Vibration Signals,” *IEEE Transactions on Energy Conversion*, vol. 30, no. 1, pp. 376–383, March 2015.
- [44] J. Al Hage, M El Badaoui El Najjar and D. Pomorski, “Multi-sensor fusion approach with fault detection and exclusion based on the Kullback–Leibler Divergence: Application on collaborative multi-robot system,” *Information Fusion*, vol. 37, pp. 61–76, Sep. 2017.
- [45] P. Baldi and L. Itti, “Of bits and wows: A Bayesian theory of surprise with applications to attention,” *Neural Networks*, vol. 23, no. 5, pp. 649–666, Jun. 2010.
- [46] D.A. Pados, P. Papantoni-Kazakos, D. Kazakos and A. G. Koyiantis, “On-line threshold learning for Neyman-Pearson distributed detection,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 24, no. 10, pp. 1519–1531, Oct. 1994.
- [47] I.Y. Hoballah and P.K. Varshney, “An information theoretic approach to the distributed detection problem,” *IEEE Transactions on Information Theory*, vol. 35, no. 5, pp. 988–994, Sep. 1989.
- [48] D. Pomorski, “Entropy-based optimisation for binary detection networks,” *Proceedings of the Third International Conference on Information Fusion*, Paris, France, 2000.