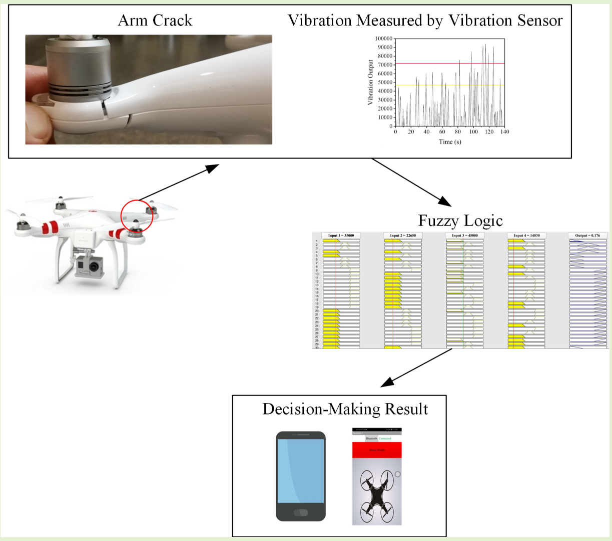
# Vibration-Based Fault Detection in Drone Using Artificial Intelligence

Mohamad Hazwan Mohd Ghazali and Wan Rahiman, *Member, IEEE*

***Abstract*—Recent years have seen a huge increase in the study of drones. There is a lot of published articles regarding drone, focusing on control optimization, fault detection, safety mechanisms, etc. In fault detection, most studies focused on the effects of faulty propellers and rotors, and there is very limited academic research on drone arms. In this paper, a fault detection based on the vibration of the multirotor arms using artificial intelligence (AI) is proposed. There are some cases in which, due to accident, the arm of the multirotor crack or loosen. This is normally unnoticeable without disassembly, and if not taken care of, it would have likely resulted in a sudden loss of flight stability, which will lead to a crash. Different types of AI methods are incorporated in this study, namely, fuzzy logic, neuro-fuzzy, and neural network (NN). Their results are compared to determine the best method in predicting the safety of the multirotor. Fuzzy logic and neuro-fuzzy methods provided acceptable decision-making, but the performance of the neuro-fuzzy approach depend on the dataset used because overfit model might give incorrect decision-making. This also applies to the NN technique. Because the vibration data are collected in the laboratory environment without consideration of wind effect, this framework is more suitable for early prediction before flying the multirotor in the outdoor environment.**



***Index Terms*—Artificial intelligence, fault detection, drones, vibration analysis, Internet of Things (IoT).**

## I. Introduction

THESE days the global market of multirotor is growing and attracting many researchers all over the globe to dive into this topic. Multirotor is a type of unmanned aerial vehicle (UAV) or drone that uses more than two motors. It has been widely applied in agriculture, military surveillance, photography, road mapping, long-range communication, and navigation [1]–[6]. Although multirotor is remotely controlled, the stability and navigation are typically aided by an onboard autopilot that relies on miniature sensors such as barometer, gyroscopes, accelerometers, and Global Positioning System (GPS) [7]. Multirotor is heavily surrounded by safety issues, particularly crash scenarios involving other human beings, animals, or other objects. This crash might cause a severe accident, as happened with the world champion skier Marcel Hirscher. He was nearly hit by a falling drone that was filming the World Cup slalom race [8]. Thus, many published articles concerning the safety of the multirotor are available, and some of them are discussed in the later section of this manuscript.

One of the most overlooked challenges in the use of multirotor is vibration. As stated by Rafiee *et al.* [9], vibration extensively exists in rotating machinery, and the vibration-based approaches are widely used in the condition monitoring and fault diagnosis systems of rotating machinery. This is one of the best ways to detect faults in the multirotor or can be applied for multirotor maintenance. If a multirotor collides or crashes, there might be some damages to the propellers or arms. The existence of such damages will produce unwanted vibrations that can significantly deteriorate the performance of the multirotor and eventually lead to a crash. These damages are sometimes not easily seen, especially on the multirotor arms, and therefore multirotor maintenance is important as it can detect these faults. The broken propeller is easily replaceable, but a faulty multirotor arm is difficult to be replaced.

Manuscript received March 1, 2022; accepted March 22, 2022. Date of publication March 30, 2022; date of current version April 29, 2022. This work was supported by the Collaborative Research in Engineering, Science, and Technology (CREST) under Grant 304/PELECT/6050424/C121. The associate editor coordinating the review of this article and approving it for publication was Prof. Boby George. *(Corresponding author: Wan Rahiman.)*

Mohamad Hazwan Mohd Ghazali is with the School of Electrical and Electronic Engineering, Universiti Sains Malaysia Engineering Campus, Nibong Tebal, Penang 14300, Malaysia.

Wan Rahiman is with the School of Electrical and Electronic Engineering and the Cluster of Smart Port and Logistic Technology (COSPALT), Universiti Sains Malaysia Engineering Campus, Nibong Tebal, Penang 14300, Malaysia (e-mail: wanrahiman@usm.my).

Digital Object Identifier 10.1109/JSEN.2022.3163401

Thus, a vibration-based fault detection system for multirotor is proposed in this article, focusing on the multirotor arms. To the best of the authors' knowledge, no published work analyzes the multirotor arm based on the vibration data. AI techniques of fuzzy logic, neuro-fuzzy, and NN are incorporated in this system to predict the safety of the multirotor, whether it is safe, partial safe, or not safe. This real-time decision-making is based on real-time data. From that, the possibility that multirotor will crash or safe to operate can be predicted.

The main contribution of this research is the study of the effects of the multirotor arms based on the vibration data, where healthy and faulty arms were simulated. The framework for the decision-making by the AI techniques and user interface via smartphones are other contributions of this study where the users can monitor the status of the multirotor based on the green, yellow, and red colour. This paper is organized as follows. In Section 2, the related works regarding the fault detection and diagnosis of the UAV are presented. Experimental works are discussed in Section 3, and the AI algorithms are described in Section 4. Section 5 presents the mobile application built explicitly as a user interface for this study. Finally, in Section 6, the conclusion and future work regarding the proposed method are explained.

## II. Related Works

Vibration analysis is among many different solutions concerning fault detection in multirotor. Verbeke and Debruyne [7] conducted experimental modal analysis (EMA) and numerical simulation to determine the dynamic characteristics of the multirotor frame in terms of mode shapes and natural frequencies. This method can determine the low-vibration regions where sensitive electronics should best be mounted. A similar analysis was performed by Li *et al.* [10], in order to propose an anti-vibration framework for the multirotor, including the best damper and isolator, based on their performance. They discovered that the structural vibration contributes a much higher vibration amplitude compared to motor vibration, and with the proposed damper and isolator, the vibration amplitude is lowered. The analysis of the random vibrations in multirotor was done by Abdulrahman Al-Mashhadani [11]. He introduced a mathematical analysis for random vibrations and proposed a control methodology to decrease the effect of unwanted vibrations. These vibrations might affect the accuracy of the data collected by the sensors in the multirotor. No application of AI methods is integrated into the works discussed above.

A technique for fault detection of physical impairment of UAV rotor blades is proposed by Bondyra *et al.* [12]. Based on the characteristics of the vibration signal, faulty rotor blades and its scale can be determined by support vector machine (SVM). Pourpanah *et al.* [13] used a hybrid method of Q-learning Fuzzy ARTMAP classifier (QFAM) and genetic algorithm (GA) to classify between healthy and broken propeller, based on vibration signals. The healthy propeller generates a smooth signal compared to the broken propeller. Ghalamchi and Mueller [14] presented a fault detection method for the multirotor propellers without using any external vibration sensors. The built-in accelerometer is used to measure the vibration signals. This method is effective in detecting the location of a damaged propeller, but it is limited by the flight trajectories. The work is then continued where specially designed trajectories are not required, and this can be seen in [15].

By utilizing the SVM method to classify the faulty and nominal flight conditions, and principal component analysis (PCA) to analyze the data, Baskaya *et al.* [16] conducted the fault detection and diagnosis on a fixed-wing UAV based on the actuator faults. From the simulated data, SVM provided excellent results in classifying the fault conditions. Iannace *et al.* [17] used the noise emitted by the UAV to develop a classification model based on neural network to detect faulty propeller blades. The faulty blades are simulated by applying two strips of paper tape to the upper surface of a blade. The proposed model achieved a 97% accuracy in detecting the faulty UAV propeller blades. Similar works were performed by Rangel-Magdaleno *et al.* [18] that implemented the discrete wavelet transform (DWT) and Fourier transform techniques, and Pechan and Sescu [19], who only performed the experimental analysis. A fault diagnosis on the UAV sensors was conducted by Guo *et al.* [20] using short-time Fourier transform (STFT) and convolutional neural network (CNN). The proposed methodologies can effectively detect the sensors' faults with a 99.6% accuracy.

Most of the studies performed by other researchers are offline methods, where the fault detection or decision-making is done not in real-time. Offline methods might take longer time to determine the condition of multirotor and real-time countermeasures cannot be taken if the measurement or decision-making is not done on the spot. Besides, the smartphone's application was not incorporated in the previous studies, where users can monitor the condition of the multirotor from the smartphone. Furthermore, very limited studies involve the hardware implementation of AI in the fault detection area. This automatic detection framework can detect the crack in the drone arms, classify its severity, and predict what will happen to the drone if it flies. To the best of the authors' knowledge, there is no published work that analyzes the multirotor's arm based on the vibration data.

## III. Experimental Work

Figure 1 shows the block diagram of this study. Firstly, vibration data are obtained from the vibration sensors and stored in the microcontroller. Then, the AI method of fuzzy logic will compute the vibration data collected and provide the decision-making, whether the multirotor is in a safe, partial safe, or not safe condition. The HC05 Bluetooth module will transmit the decision or output to the smartphone, and via the mobile application created, the user can monitor the condition of the multirotor in real-time.

All the vibration data are collected in an indoor environment and before the drone take-off. The SW420 vibration sensors (refer to Figure 2) are used to measure the vibration of the multirotor. The integrated LM393 comparator chip provides two output types: digital output (based on values 0 and 1) and analog output (value is in the form of received voltage). The analog output of the sensor will be utilized in this study, where the higher the vibration, the higher the analog output produced. The sensitivity of the SW420 vibration sensor can be adjusted to the desired value by turning the potentiometer. The sensors are connected to the Arduino UNO pin to store the collected data and are attached to the arms of the multirotor.

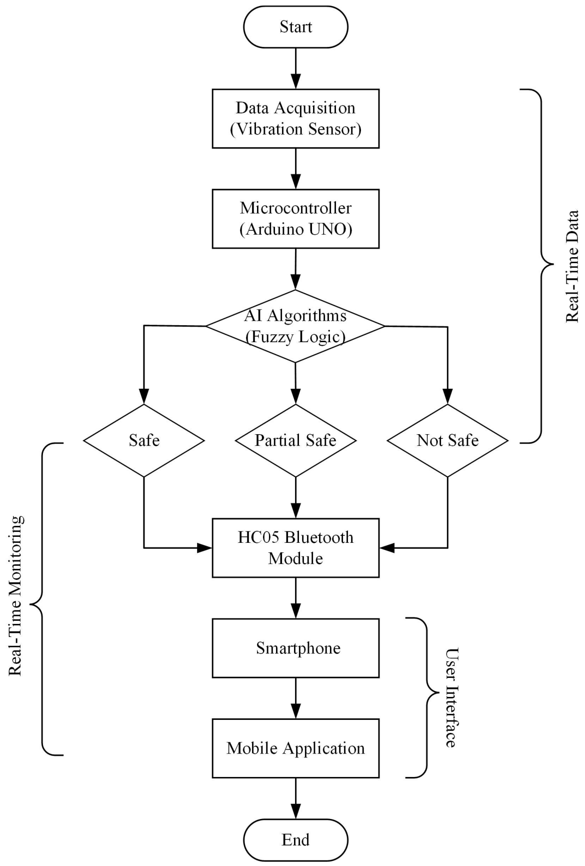


Fig. 1. The block diagram of the proposed method, starting with the data acquisition from the vibration sensors, and ends with the real-time monitoring of the multirotor condition in the smartphone.

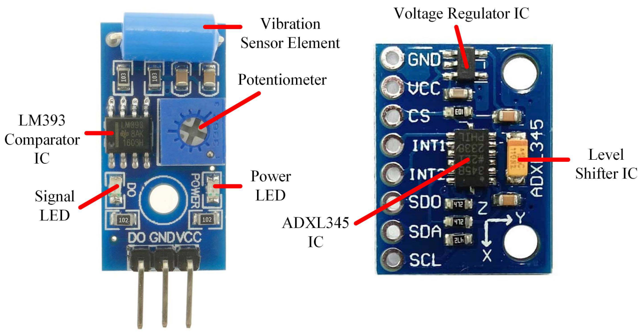


Fig. 2. The SW420 vibration sensor and ADXL345 accelerometer.

The voltage common collector (VCC), ground (GND), and digital output (DO) pins of the vibration sensor are connected to the 5V, GND, and digital pin of Arduino UNO, respectively. Compared to the popular ADXL345 accelerometer (refer to Figure 2), the SW420 vibration sensor is selected for fault detection in multirotor because it is lightweight and easier to install, as it does not require the I2C communication protocol to operate. As we only want to determine the level of vibration, the SW420 vibration sensor is sufficient. If we want to determine the faulty location in the multirotor, then the ADXL345 accelerometer is required as the frequency information is critical. A Fourier transform technique is required when using the ADXL345 accelerometer to classify the vibration level in terms of frequency, which is difficult to compute in real-time. However, the vibration level can still be determined using the time-domain data in terms of x, y, and z acceleration values. The technical specifications of the SW420 vibration sensor and ADXL345 accelerometer are listed in Appendix A.

TABLE I

THE SPECIFICATIONS OF MULTIROTOR USED IN THIS STUDY AND ADDITIONAL LOAD ADDED TO THE MULTIROTOR'S SYSTEM

| | |
|---|---|
| Type of multirotor | Quadcopter |
| Weight of Q7 2.4G quadcopter | 100 g with battery |
| Weight of DJI Phantom 1 | 840 g with battery |
| Payload of DJI Phantom 1 | 365 g |
| Max distance of Q7 2.4G quadcopter | 100 m |
| Max distance of DJI Phantom 1 | 300 m |
| Flight time of Q7 2.4G quadcopter | 5 to 7 minutes |
| Flight time of DJI Phantom 1 | 10 minutes |
| Weight of SW420 vibration sensors | 3g x 4 = 12 g (for quadcopter) |
| Weight of breadboard | 40 g |
| Weight of 5000 mAh powerbank | 150 g |
| Weight of Arduino UNO | 25 g |
| Weight of Arduino cable | 26 g |
| Weight of HC-05 bluetooth module | 3.5 g |
| Weight of the overall AI system | 256.5 g |

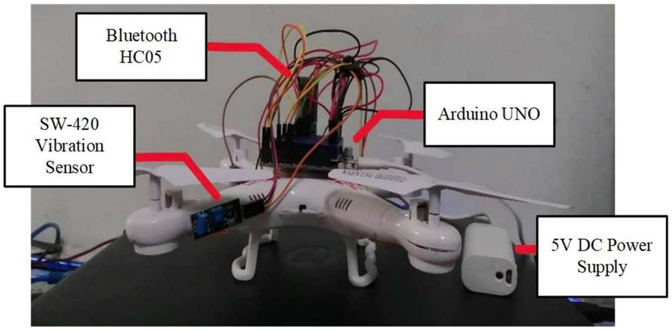
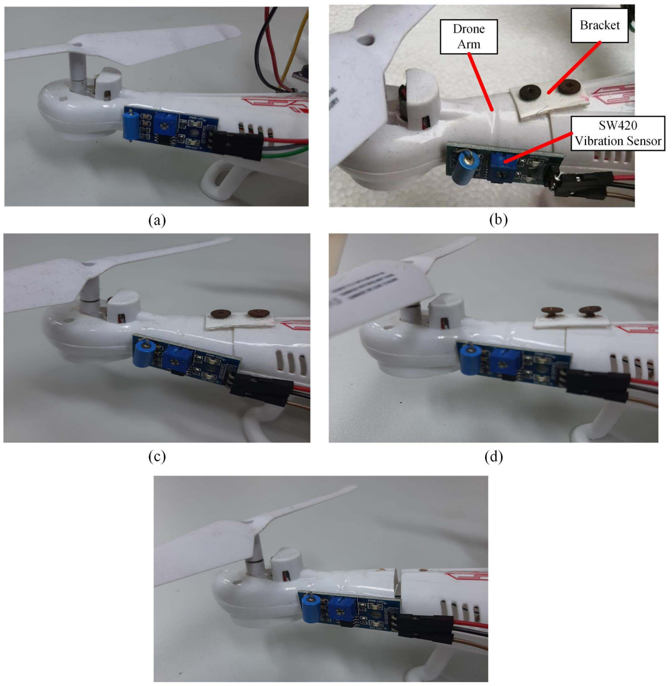


Fig. 3. The Q7 2.4G quadcopter applied for data collection in this study.

The multirotor used in this study are Q7 2.4G quadcopter and DJI Phantom 1, and their specification can be observed in Table I. Due to the cost factor, we used the Q7 2.4 G quadcopter for faulty simulation and data collection purposes, as depicted in Figure 3. Then, we implemented the proposed framework on DJI Phantom 1. The microcontroller used is Arduino UNO, where four SW420 vibration sensors are attached to the multirotor arms and connected to the microcontroller to record the vibration data. Bluetooth HC05 is incorporated to connect with the smartphones, and to power up the microcontroller, a 5000 mAh power bank is used. From Table I, this AI-based decision-making system is practical to be equipped to the DJI Phantom 1 as only a total mass of 256.5 g would be added to the multirotor, which is below the maximum payload (365 g). This was proved experimentally when the multirotor could successfully take off and hover with the AI system equipped to it in an indoor environment. For multirotor arms faulty simulation, it was modified by cutting and joining back using plastic brackets, which can be observed in Figure 4. This bracket can be adjusted to make it loosen to get the faulty data and tighten to get the normal data. Using the pulse reading from Arduino UNO to calibrate the degree of truth for the vibration sensor, the maximum pulse recorded is 100000, and thus, the range was set up from 0 to 100000. For neuro-fuzzy and NN, the pulse value is normalized from 0 to 1. All vibration sensors have the same degree of truth since the same motor and vibration sensors were applied.

TABLE II
EXPERIMENTAL RESULTS TO DETERMINE THRESHOLD VALUES

| Multirotor Arms Configuration | Drone Experimental Condition | Vibration Output Before Take Off | Threshold |
|---|---|---|---|
| Original arm condition | Can take off and hover safely | All vibration amplitudes are less than 47000 | Safe |
| 100% screwed | Can take off and hover safely | All vibration amplitudes are less than 47000 | Safe |
| 50% screwed | Can take off and hover for about 4 minutes | Some of the vibration amplitudes are above 47000, but all the amplitudes are below 72000 | Partial safe |
| 10% screwed | Can take off but crash immediately | Some of the vibration amplitudes are above 72000 | Not safe |
| Unscrewed | Can take off but crash immediately | Some of the vibration amplitudes are above 72000 | Not safe |



(a) (b) (c) (d) (e)

Fig. 4. Multirotor arms configuration: (a) Original arm condition, (b) 100% screwed (1.5 N.m), (c) 50% screwed (0.7 N.m), (d) 10% screwed (0.3 N.m), and (e) unscrewed multirotor arm.

There are five experimental conditions for vibration measurement in this study based on the multirotor arms configuration, as shown in Figure 4. The multirotor arms are tightened using the TSD-200 digital torque screwdriver. The 100%, 50%, and 10% screwed multirotor arms are equal to the torque value of 1.5 N.m, 0.7 N.m, and 0.3 N.m, respectively. The real-time data was collected about two minutes for each condition using the Parallax Data Acquisition tool (PLX-DAQ) Microsoft Excel add-in. After enabling the Macros, the vibration data can be recorded and plotted automatically in the Excel spreadsheet. These data are collected when the multirotor is on the ground. After collecting the vibration data, at each condition, the drone will take off and hover at a 1 m height using the experimental setup as shown in Figure 5.

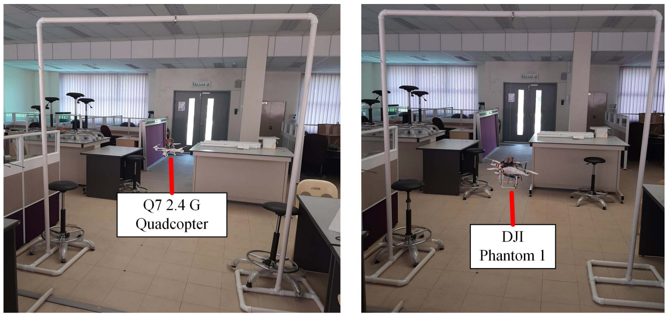


Fig. 5. The experimental setup to determine the threshold values and final implementation.

Figure 6 shows the vibration data collected under different multirotor conditions using the SW420 vibration sensor and ADXL345 accelerometer. Based on the experimental work, the vibration input can be divided into low vibration amplitude ($<$ 0.47 or 47000), medium vibration amplitude (0.47 or 47000 $\leq$ x $<$ 0.72 or 72000), and high vibration amplitude ($\geq$ 0.72 or 72000). The thresholds are determined using the experimental setup depicted in Figure 5, and the details are listed in Table II. Notice that the vibration measurement using the SW420 vibration sensor has no SI unit, as this type of sensor only produces analog (value is in the form of received voltage) and digital outputs (0 or 1). In this study, the analog output of the SW420 vibration sensor was utilized, which is directly proportional to the vibration amplitude. This is still acceptable because frequency values are not vital in this study, as long as the values for different multirotor conditions can be differentiated. For each multirotor arms configuration, the acceleration values (x-axis) are provided to relate the physical quantity with the "pulse" received by the acquisition board. The output, which is the safety of the multirotor, can be divided into safe state ($<$ 0.4), partial safe state (0.4 $\leq$ x $<$ 0.65), and not safe state ($\geq$ 0.65). Table III shows the decision-making results from the experiment, which will be the guidelines or parameters for AI systems.

## IV. ARTIFICIAL INTELLIGENCE ALGORITHMS

### A. Fuzzy Logic

Introduced by Lotfi Zadeh back in 1965, fuzzy logic has the ability to model the imprecise modes of reasoning to

TABLE III

GUIDELINES FOR THE FUZZY LOGIC AND NEURO-FUZZY SYSTEMS ACQUIRED FROM THE EXPERIMENTAL WORKS, E.G. IF THE VIBRATION MEASURED BY THE SENSOR A, B, C, AND D ARE LOW, THEN THE MULTIROTOR IS SAFE

| Sensor | Safe Multirotor Condition | Partial Safe Multirotor Condition | Not Safe Multirotor Condition |
|---|---|---|---|
| A | Low / Medium | A And B Medium / All Medium | High And Other Low / All High |
| B | Low / Medium | B And A Medium / All Medium | High And Other Low / All High |
| C | Low / Medium | C And D Medium / All Medium | High And Other Low / All High |
| D | Low / Medium | D AND C Medium / All Medium | High And Other Low / All High |

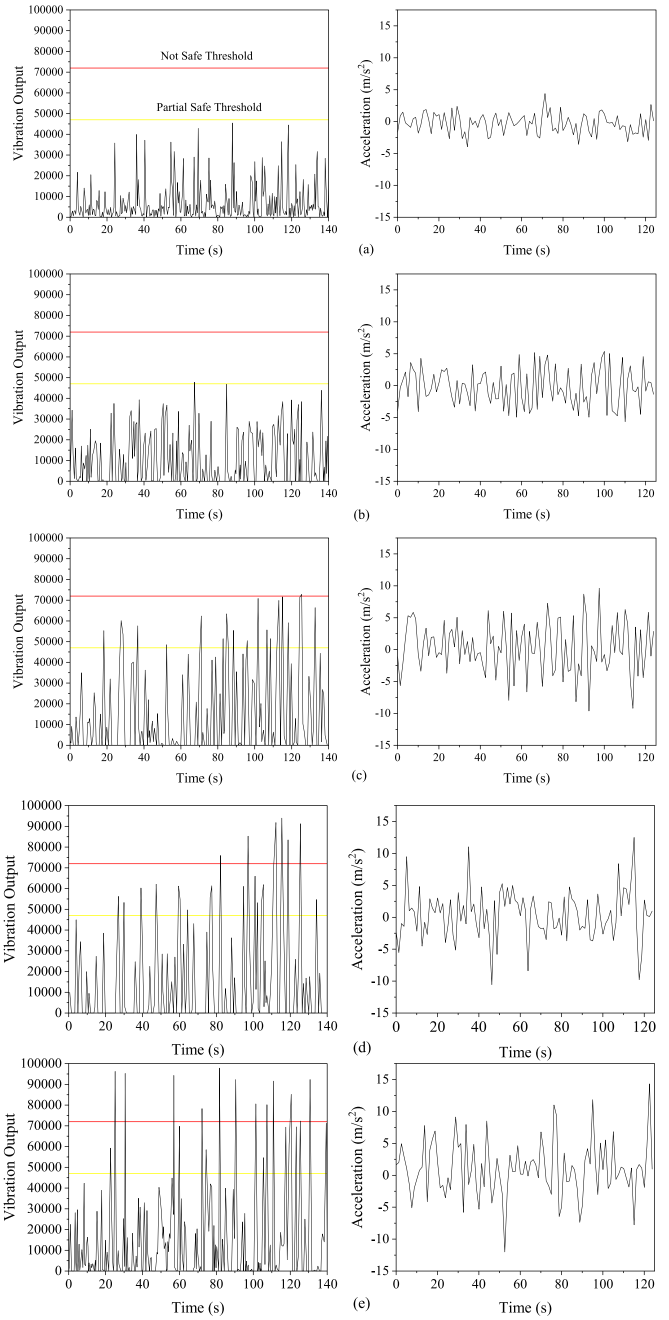


Fig. 6. Vibration output and acceleration values (x-axis) recorded for; (a) original arm condition, (b) 100% screwed (fully tightened), (c) 50% screwed (half tightened), (d) 10% screwed, and (e) unscrewed multirotor arm conditions.

TABLE IV

DETAILED SIMULATION PARAMETERS OF THE PROPOSED FUZZY LOGIC MODEL

| Step | Parameter | Detail |
|---|---|---|
| 1 | Number of inputs | 4 |
| | Number of membership function for each input | 3 |
| | Type of membership function for each input | Trapezoidal |
| 2 | Number of rules | 81 |
| 3 | Number of outputs | 1 |
| | Number of membership function for each output | 3 |
| | Type of membership function for each output | Triangular |

make rational decisions in an environment of uncertainty and imprecision [21]. Compared to classical logic, it permits the input to have a value between 0 (completely false) and 1 (completely true). Fuzzy logic has been utilized for fault detection in many areas such as robotics, machine vision, energy, industries, etc. In this study, Mamdani-type fuzzy logic with four inputs and one output is used. The inputs are the vibration measured by the four sensors attached to each of the multirotor's arms, which are recognized as sensor A, sensor B, sensor C, and sensor D. The output is the decision-making given by the fuzzy logic system.

Mamdani fuzzy logic toolbox in MATLAB was utilized to construct our proposed fuzzy logic model, which can be divided into three steps. The first step is fuzzification, where the input data are converted into fuzzy sets using fuzzy linguistic variables, fuzzy linguistic terms, and membership functions. At this step, a designer needs to specify the number of inputs, number of membership functions, and type of membership function. Next is creating the rules for the fuzzy logic system. These rules represent human experts' knowledge. In most cases, the rules can be determined based on human experiences or analysis conducted by the engineer. The fuzzy rules must be in the form of "If-Then" clauses. There are 81 rules defined manually in this fuzzy logic system. The last step is defuzzification. It is a process of producing quantifiable results in fuzzy logic for given fuzzy sets and corresponding membership functions. The number of outputs, the number of membership functions, and the type of membership function are specified in this stage. The detailed simulation parameter for the fuzzy logic model is listed in Table IV.

## B. Neuro-Fuzzy

The neuro-fuzzy, also known as a fuzzy neural network (FNN), hybridizes two AI approaches by combining the human-like reasoning style of fuzzy logic systems with the learning and connectionist structure of NN. A neuro-fuzzy system can be considered as a 3-layer feedforward NN, where the first layer represents input variables, the fuzzy rules as the second layer, and output variables as the third layer. It can be considered as an improvement to the neural network method in a way that prior knowledge about the training dataset can be encoded in the parameters of the neuro-fuzzy system [22]. In MATLAB, neuro-fuzzy or adaptive neuro-fuzzy inference system (ANFIS) implements the Takagi-Sugeno fuzzy type where the "If-Then" rules can be expressed as follow [23]:

$$\begin{aligned}\text{Rule1}: &\text{ IF } x \text{ is } A1 \text{ AND } y \text{ is } B1, \text{ THEN}\\ &f1 = p1x + q1y + r1\\ \text{Rule2}: &\text{ IF } x \text{ is } A2 \text{ AND } y \text{ is } B2, \text{ THEN}\\ &f2 = p2x + q2y + r2\end{aligned}$$

where $A1$, $A2$, and $B1$, $B2$ are the membership functions for inputs $x$ and $y$ respectively, whereas $p1$, $q1$, $r1$ and $p2$, $q2$, $r2$ represent the associated parameters of the output functions, $f1$ and $f2$.

In this study, the ANFIS network has four inputs (Sensors A, B, C, and D). The first layer represents the fuzzification process, the second layer represents the fuzzy rules with 81 nodes, and the third layer normalizes the membership functions. The fourth layer is the conclusive part of fuzzy rules, and finally, the fifth layer calculates the network output, which is the decision-making regarding the drone condition. ANFIS toolbox in MATLAB was used for the entire process of training, testing, and predicting the condition of the multirotor. Because ANFIS provides good learning and prediction capabilities, it can efficiently deal with uncertainties in any system [24]. However, the neuro-fuzzy technique heavily depends on the dataset.

For the sake of comparison and to determine the effect of different datasets on the neuro-fuzzy prediction model, 100 and 1000 datasets were used in this study. The ANFIS model with 100 and 1000 datasets are referred to as ANFIS 1 and 2 models, respectively. For the 100 and 1000 datasets, 400 and 4000 randomly generated inputs from MATLAB were used, respectively, where each dataset contains four inputs (inputs for Sensor A, B, C, and D) and one output. Both ANFIS models utilized the Gaussian-type membership function, and the fuzzy inference systems were generated via the grid partition technique. In both datasets, 80% are used to train the dataset, 10% are used for testing, and another 10% for checking. The inputs for the neuro-fuzzy were normalized to reduce the checking and testing error. After preparing the datasets, we trained both ANFIS models using the hybrid optimization method with epoch values of 10 (ANFIS 1) and 100 (ANFIS 2). The training performance of both ANFIS models can be observed in Figure 7.

The training performance of the ANFIS 1 model is better in terms of root mean square error (RMSE), mean square error (MSE), and mean absolute error (MAE) compared to the ANFIS 2 model. Figure 8 shows the testing performances of both ANFIS models, and based on the results, ANFIS 2 model has a superior testing performance compared to ANFIS 1 model in terms of RMSE, MSE, and MAE values.

### C. Artificial Neural Networks

Artificial neural network (ANN or NN) can be defined as an interconnected assembly of simple processing elements, units, or nodes, whose functionality is inspired by the way that the human brain processes information [25]. These large numbers of nodes are connected in layers, forming a network. NN can be divided into single-layer and multilayer NN. In the single-layer NN, also referred to as perceptron, there is only one neuron and computes only one output. For multilayer NN, there are three types of layers involved, which are input, hidden, and output layers. In the input layer, data processing takes place where data are manipulated before inputting it into the NN. Situated between the input and output layer, there is a hidden layer that conducts mathematical transformations by applying the numeric values called weights to the network inputs. The output layer is the last layer of the NN architecture and is the result of the data when passed through NN.

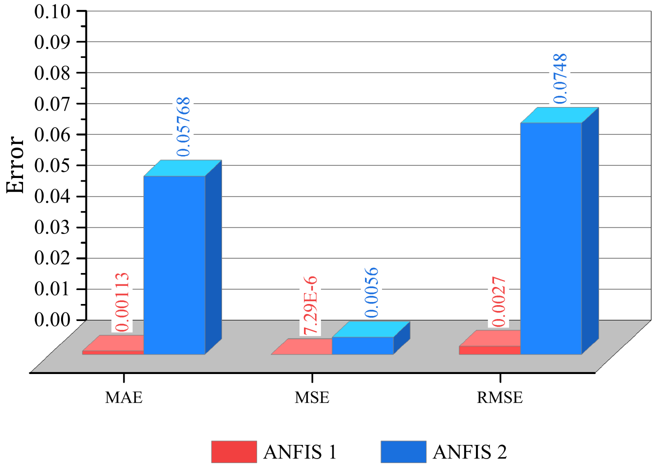


Fig. 7. The training performance of ANFIS models.

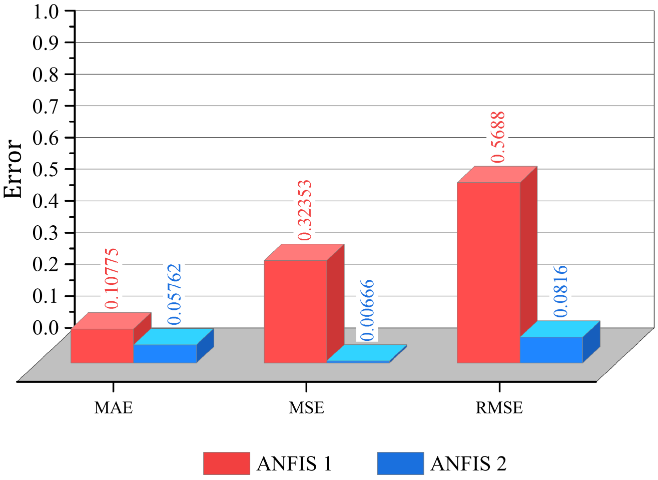


Fig. 8. The testing performance of ANFIS models.

In this study, the NN models for the fault detection in multirotor were designed using the MATLAB NN toolbox. Similar to the ANFIS technique, the performance of the NN model depends on the size and quality of the dataset. Two NN models are constructed: NN 1 (based on 100 datasets) and NN 2 (based on 1000 datasets). The datasets used are similar to those applied for the ANFIS models. For both models, 80% of the datasets are used for training, 10% are used for testing and finally, we utilized another 10% of the dataset for checking. After classifying the dataset and specifying the number of hidden neurons (10 and 100 hidden neurons for NN 1 and NN 2 models, respectively), the NN models are trained with the Levenberg-Marquardt backpropagation algorithm.

The training and testing performances of both NN models are depicted in Figures 9 and 10. NN 2 has a better training performance in terms of RMSE, MSE, and MAE compared to

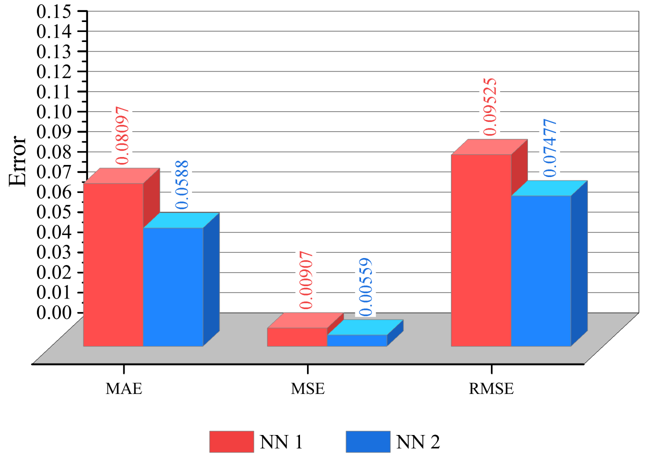


Fig. 9. The training performance of NN models.

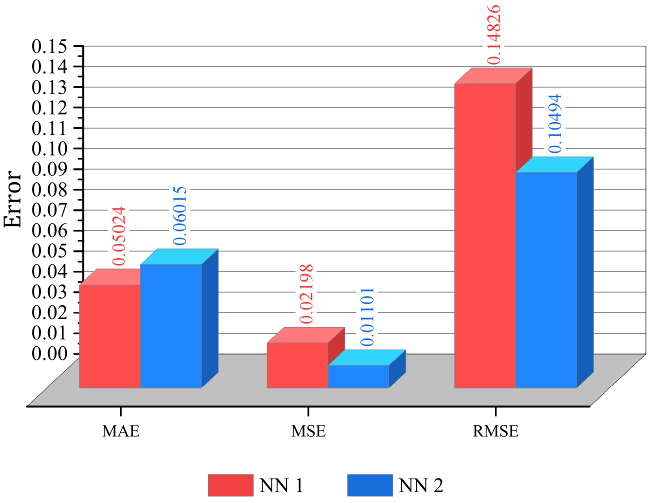


Fig. 10. The testing performance of NN models.

NN 1. However, the differences between the training errors for the NN models are smaller than the ANFIS models. For the testing part, NN 1 has a better MAE value, but poorer RMSE and MSE values compared to NN 2.

### D. Performance Comparison

Table V shows the decision-making results from different AI models when fed with artificial data from MATLAB. Input 1, 2, 3, and 4 represent the vibration recorded by sensors A, B, C, and D, respectively. When the value of input 1 is 35000 or 0.35 (low vibration amplitude), input 2 is 22650 or 0.227 (low vibration amplitude), input 3 is 45000 or 0.45 (low vibration amplitude), and input 4 is 14000 or 0.14 (low vibration amplitude), based on Table III, the decision-making should be safe. The output given by the fuzzy logic is 0.176, whereas, for the ANFIS 1 model, the output is 0.371. The ANFIS 2, NN 1, and NN 2 models gave 0.255, 0.342, and 0.268, respectively. Although all systems produced correct decision-making, the fuzzy logic, ANFIS 2, and NN 2 models performed better as the outputs were far from the threshold value of 0.4, which belongs to the partial safe state.

Ten new datasets with randomly generated inputs were fed into all the AI systems to validate their decision-making performances, which can be seen in Figure 11 and Table VI.

TABLE V
THE DECISION-MAKING FROM DIFFERENT AI MODELS

| Input 1 | Input 2 | Input 3 | Input 4 | AI Model | Output |
|---|---|---|---|---|---|
| 35000 | 22650 | 45000 | 14000 | Fuzzy Logic | 0.176 |
| 0.35 | 0.2265 | 0.45 | 0.14 | ANFIS 1 | 0.351 |
| 0.35 | 0.2265 | 0.45 | 0.14 | ANFIS 2 | 0.255 |
| 0.35 | 0.2265 | 0.45 | 0.14 | NN 1 | 0.342 |
| 0.35 | 0.2265 | 0.45 | 0.14 | NN 2 | 0.268 |

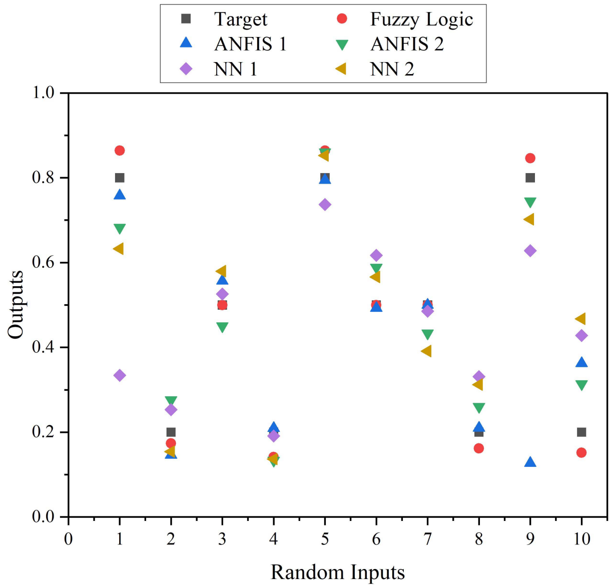


Fig. 11. The comparison between target outputs and outputs from different AI methods.

It can be seen that the fuzzy logic model performed better compared to the ANFIS and NN models, which in some cases, provided inaccurate decision-making results. For example, in dataset nine, when input 1 is low, input 2 is medium, input 3 is low, and input 4 is high, the decision-making should be not safe, as predicted by the fuzzy logic, ANFIS 2, and NN 2 models. However, the ANFIS 1 and NN 1 models underestimate the output, where the decision-making made indicates that the multirotor is in a safe and partial safe condition, respectively, as shown in Figure 11. This is dangerous because if the ANFIS 1 decision-making model is applied and the multirotor is predicted to be safe to operate, the multirotor will crash due to the incorrect prediction. This is due to the generalization error where during the training stage, the outputs produced matched the desired outcomes, but when fed with new data, it produced false decision-making. This model is referred to as an overfit model.

Overfitting occurs when ANFIS or NN models overtrain the dataset. This means that during training, the maximum number of epochs has been exceeded, which can lead to the predicted output being over its accuracy. Compared to other AI methods in this study, the ANFIS 2 is a good fit model because it produced acceptable results in training as well as when fed with new randomly generated data, which are different from the data used in the training stage. Thus, datasets are important when it comes to the generalization problem. Apart from that, fuzzy logic has less computational burden than neuro-fuzzy and NN.

TABLE VI
THE COMPARISON BETWEEN DIFFERENT AI TECHNIQUES USING TEN NEW RANDOM INPUTS

| Input 1 (Hz) | Input 2 (Hz) | Input 3 (Hz) | Input 4 (Hz) | Desired Output | Fuzzy Logic Output | ANFIS 1 Output | ANFIS 2 Output | NN 1 Output | NN 2 Output |
|---|---|---|---|---|---|---|---|---|---|
| 0.1211/12110 | 0.1544/15447 | 0.238/23897 | 0.7536/75360 | 0.8 (Not Safe) | 0.8644 (Not safe) | 0.7577 (Not Safe) | 0.6829 (Not Safe) | 0.3341 (Safe) | 0.6328 (Partial Safe) |
| 0.4135/41354 | 0.3967/39674 | 0.4433/44331 | 0.2154/21535 | 0.2 (Safe) | 0.1734 (Safe) | 0.1461 (Safe) | 0.276 (Safe) | 0.2534 (Safe) | 0.1541 (Safe) |
| 0.5563/55632 | 0.4931/49312 | 0.6122/61224 | 0.6875/68752 | 0.5 (Partial Safe) | 0.5 (Partial Safe) | 0.5567 (Partial Safe) | 0.4505 (Partial Safe) | 0.5259 (Partial Safe) | 0.5796 (Partial Safe) |
| 0.1535/15352 | 0.3563/35634 | 0.4169/41692 | 0.2136/21358 | 0.2 (Safe) | 0.1417 (Safe) | 0.2096 (Safe) | 0.1334 (Safe) | 0.1912 (Safe) | 0.1366 (Safe) |
| 0.8873/88731 | 0.9248/92483 | 0.7505/75051 | 0.6045/60451 | 0.8 (Not Safe) | 0.8644 (Not Safe) | 0.7944 (Not Safe) | 0.8607 (Not Safe) | 0.7372 (Not Safe) | 0.8525 (Not Safe) |
| 0.2542/25419 | 0.4568/45683 | 0.6489/64887 | 0.7155/71550 | 0.5 (Partial Safe) | 0.5 (Partial Safe) | 0.4928 (Partial Safe) | 0.5883 (Partial Safe) | 0.6171 (Partial Safe) | 0.5661 (Partial Safe) |
| 0.5004/50043 | 0.1217/12172 | 0.5529/55291 | 0.1998/19982 | 0.5 (Partial Safe) | 0.5 (Partial Safe) | 0.5001 (Partial Safe) | 0.4334 (Partial Safe) | 0.4853 (Partial Safe) | 0.3912 (Safe) |
| 0.6074/60744 | 0.5561/55609 | 0.4981/49807 | 0.1005/10053 | 0.2 (Safe) | 0.1615 (Safe) | 0.21 (Safe) | 0.2605 (Safe) | 0.3312 (Safe) | 0.3124 (Safe) |
| 0.4504/45039 | 0.6714/67142 | 0.145/14501 | 0.81/81004 | 0.8 (Not Safe) | 0.8462 (Not Safe) | 0.1268 (Safe)* | 0.7447 (Not Safe) | 0.6282 (Partial Safe)* | 0.7022 (Not Safe) |
| 0.6527/65271 | 0.5586/55863 | 0.5011/50107 | 0.2276/22764 | 0.2 (Safe) | 0.1514 (Safe) | 0.3626 (Safe) | 0.3139 (Safe) | 0.4282 (Partial Safe) | 0.4675 (Partial Safe) |

*Inaccurate decision-making result.

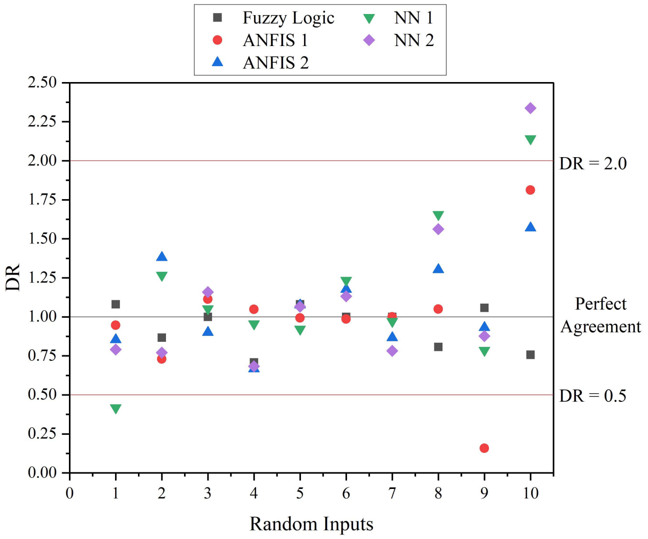


Fig. 12. The DR values of different AI techniques.

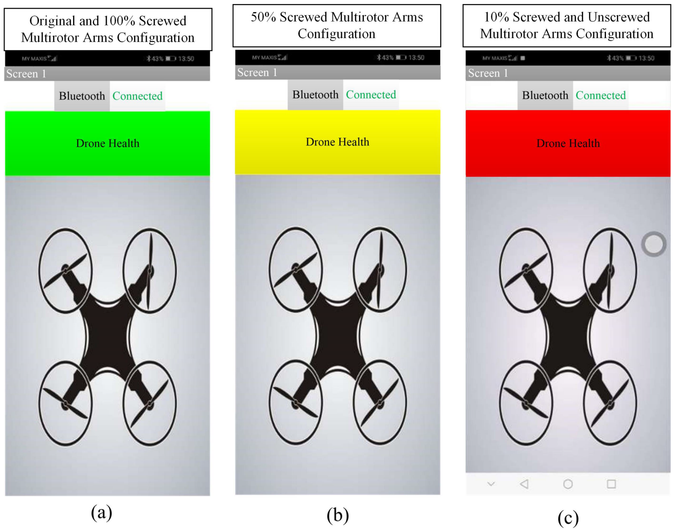

Original and 100% Screwed Multirotor Arms Configuration
50% Screwed Multirotor Arms Configuration
10% Screwed and Unscrewed Multirotor Arms Configuration
Screen 1
Bluetooth
Connected
Drone Health
(a)
(b)
(c)

Fig. 13. The real-time monitoring of the multirotor's condition from the smartphone as predicted by the proposed method; (a) green indicator means the multirotor is safe to operate, (b) yellow indicator means the multirotor is partially safe to operate, and (c) red indicator means the multirotor is not safe to operate.

We further evaluated all the AI methods using the discrepancy ratio (DR), which is the ratio of the predicted output to target out [26], calculated as

$$\mathrm{DR} = \frac{\text{Predicted Output}}{\text{Target Output}}. \tag{1}$$

The DR value of 1 represents the best model performance, whereas a value greater or lower than 1 indicates overestimation or underestimation, respectively. Based on Figure 12, there are DR values that exceed the DR = 2.0 threshold and are lower than the DR = 0.5 threshold. This will foreshadow poor ANFIS and NN models in providing the decision-making regarding the condition of multirotor. Thus, the fault detection system based on fuzzy logic will be incorporated into the multirotor for real-time application. It should also be taken into account that the vibration data collected when the multirotor is flying in the outdoor environment to be different compared to the data obtained in this study. Thus, this framework is more suitable for early prediction before flying the multirotor in an outdoor environment.

## V. User Interface

Since users are expectedly not familiar with the outputs in decimal form, the MIT app inventor is used to create the application in APK specifically for better and easier monitoring. MIT app inventor is an open-source web application that permits users to design software applications for the Android operating system. It adopts a graphical interface, allowing users to drag-and-drop visual objects to create the desired application using the block-based coding program. The created mobile application can be accessed using a smartphone. Firstly, users have to connect the smartphone with the HC05 Bluetooth module. Then, this application will show three possible decisions, which is green, referring to safe to operate, yellow, referring to partial safe, and red, referring to not safe to operate. This decision-making can be monitored in the smartphone through the application created and the HC05 Bluetooth module, which can be observed in Figure 13.

## VI. Conclusion

Real-time vibration-based fault detection using AI techniques was introduced in this study. This method used the vibration sensors attached to the multirotor's arms to obtain the vibration data, which was then fed to the AI decision-making systems. The AI techniques used were fuzzy logic, neuro-fuzzy, and NN, and these systems will make a decision whether the multirotor is safe, partially safe, or unsafe, which can be monitored using a smartphone. The performance between different AI systems were compared, and the fuzzy logic provided better results as it produced the results closest to the desired value. However, these findings are only based on indoor testing, and the neuro-fuzzy or NN method might perform better if the works are done in the outdoor environment. This study is also limited to only one parameter, which is the multirotor arms. This work can be extended by including other parameters such as propeller vibration, motor condition, and battery level. In terms of hardware, the built-in accelerometer of the multirotor system can also be used to measure the vibration, and an algorithm to differentiate between the propeller and multirotor arm vibrations can be developed. This approach avoids the need to install additional sensors to the multirotor and thus provides a potentially simpler and more power-efficient approach to real-time in-flight fault detection. A printed circuit board (PCB) can also be applied to minimize the wiring of the AI system and eliminate unwanted interference. This work only acts as a foundation for fault detection of multirotor. In the future, outdoor experiments or testing will be considered to represent the actual flight condition of the drone in terms of the effect of wind speed, different maneuverability, and aggressiveness. Performance comparison of fuzzy logic and neuro-fuzzy detection with other AI methods such as deep learning and SVM can also be made before proceeding to test the real-time in-flight energy usage of each AI system.

## Appendix A
## Technical Specifications of SW420 Vibration Sensor

| | |
|---|---|
| Weight | 2 g |
| Operating voltage | 3.3 V to 5 V DC |
| Operating current | 15 mA |
| Sensitivity | ± 1 |
| Dimension | 3.2 cm x 1.4 cm |
| Output | Analog and Digital (0 and 1) |

## Appendix B
## Technical Specifications of ADXL345 Accelerometer

| | |
|---|---|
| Weight | 1 g |
| Operating voltage | 3 V to 6 V DC |
| Operating current | 40 $\mu$A |
| Sensitivity | 3.9 mg/LSB for +/- 2g, 7.8 mg/LSB for +/- 4g, 15.6 mg/LSB for +/- 8g, and 31.2 mg/LSB for +/- 16g |
| Dimension | 2 cm x 1.5 cm |
| Measurement range | ± 16g |
| Frequency range | 10 Hz to 3200 Hz |
| Output | Digital |

## Acknowledgment

The authors would like to thank Synvue Sdn Bhd (1236749-D) for insights and feedback of the project development.

## References

[1] V. Puri, A. Nayyar, and L. Raja, "Agriculture drones: A modern breakthrough in precision agriculture," *J. Statist. Manage. Syst.*, vol. 20, no. 4, pp. 507–518, 2017.

[2] M. A. Ma'sum *et al.*, "Simulation of intelligent unmanned aerial vehicle (UAV) for military surveillance," in presented at the Int. Conf. Adv. Comput. Sci. Inf. Syst. (ICACSIS), Sanur Bali, Indonesia, Sep. 2013.

[3] M. A. Zulkipli and K. N. Tahar, "Multirotor UAV-based photogrammetric mapping for road design," *Int. J. Opt.*, vol. 2018, pp. 1–7, Oct. 2018.

[4] B. Jalil, G. R. Leone, M. Martinelli, D. Moroni, M. A. Pascali, and A. Berton, "Fault detection in power equipment via an unmanned aerial system using multi modal data," *Sensors*, vol. 19, no. 13, pp. 3014–3029, 2019.

[5] M. H. M. Ghazali, K. Teoh, and W. Rahiman, "A systematic review of real-time deployments of UAV-based Lora communication network," *IEEE Access*, vol. 9, pp. 124817–124830, 2021.

[6] A. Arenella, A. Greco, A. Saggese, and M. Vento, "Real time fault detection in photovoltaic cells by cameras on drones," presented at the Int. Conf. Image Anal. Recognit., Montreal, QC, Canada, Jul. 2017.

[7] J. Verbeke and S. Debruyne, "Vibration analysis of a UAV multirotor frame," presented at the Int. Conf. Noise Vib. Eng., Leuven, Belgium, Sep. 2016.

[8] (2015). *Drone Crashes Onto Piste, Misses Champion Skier by Inches*. G. Matias. Accessed: Dec. 7, 2020. [Online]. Available: https://edition.cnn.com/2015/12/23/sport/marcel-hirscher-drone-crash/index.html

[9] J. Rafiee, F. Arvani, A. Harifi, and M. H. Sadeghi, "Intelligent condition monitoring of a gearbox using artificial neural network," *Mech. Syst. Signal Process.*, vol. 21, no. 4, pp. 1746–1754, May 2007.

[10] Z. Li, M. Lao, S. K. Phang, M. R. A. Hamid, K. Z. Tang, and F. Lin, "Development and design methodology of an anti-vibration system on micro-UAVs," presented at the 9th Int. Micro Air Vehicle Conf. Flight Competition, Toulouse, France, Sep. 2017.

[11] M. Abdulrahman Al-Mashhadani, "Random vibrations in unmanned aerial vehicles, mathematical analysis and control methodology based on expectation and probability," *J. Low Freq. Noise, Vib. Act. Control*, vol. 38, no. 1, pp. 143–153, Mar. 2019.

[12] A. Bondyra, P. Gasior, S. Gardecki, and A. Kasinski, "Fault diagnosis and condition monitoring of UAV rotor using signal processing," presented at the Signal Process.: Algorithms, Architectures, Arrangements, Appl. (SPA), Poznan, Poland, Sep. 2017.

[13] F. Pourpanah, B. Zhang, R. Ma, and Q. Hao, "Anomaly detection and condition monitoring of UAV motors and propellers," presented at the IEEE Sensors, New Delhi, India, Oct. 2018.

[14] B. Ghalamchi and M. Mueller, "Vibration-based propeller fault diagnosis for multicopters," presented at the Int. Conf. Unmanned Aircr. Syst., Dallas, TX, USA, Jun. 2018.

[15] B. Ghalamchi, Z. Jia, and M. W. Mueller, "Real-time vibration-based propeller fault diagnosis for multicopters," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 1, pp. 395–405, Feb. 2020.

[16] E. Baskaya, M. Bronz, and D. Delahaye, "Fault detection and diagnosis for small UAVs via machine learning," presented at the IEEE/AIAA 36th Digit. Avionics Syst. Conf., St. Petersburg, FL, USA, Sep. 2017.

[17] G. Iannace, G. Ciaburro, and A. Trematerra, "Fault diagnosis for UAV blades using artificial neural network," *Robot.*, vol. 8, no. 3, pp. 59–76, 2019.

[18] J. d.-J. Rangel-Magdaleno, J. Urena-Urena, A. Hernandez, and C. Perez-Rubio, "Detection of unbalanced blade on UAV by means of audio signal," presented at the IEEE Int. Autumn Meeting Power Electron. Comput., Ixtapa, Mexico, Nov. 2018.

[19] T. Pechan and A. Sescu, "Experimental study of noise emitted by propeller's surface imperfections," *Appl. Acoust.*, vol. 92, pp. 12–17, 2015.

[20] D. Guo, M. Zhong, H. Ji, Y. Liu, and R. Yang, "A hybrid feature model and deep learning based fault diagnosis for unmanned aerial vehicle sensors," *Neurocomputing*, vol. 319, pp. 155–163, Nov. 2018.

[21] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, Apr. 1988.

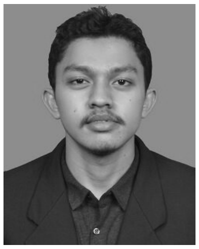
[22] J. M. Escano, M. A. Ridao-Olivar, C. Ierardi, A. J. Sanchez, and K. Rouzbehi, "Driver behavior soft-sensor based on neurofuzzy systems and weighted projection on principal components," *IEEE Sensors J.*, vol. 20, no. 19, pp. 11454–11462, Oct. 2020.

[23] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan. 1985.

[24] R. Zakaria, O. Yong Sheng, K. Wern, S. Shamshirband, D. Petkovic, and N. T. Pavlovic, "Adaptive neuro-fuzzy evaluation of the tapered plastic multimode fiber-based sensor performance with and without silver thin film for different concentrations of calcium hypochlorite," *IEEE Sensors J.*, vol. 14, no. 10, pp. 3579–3584, Oct. 2014.

[25] K. Gurney, *An Introduction to Neural Networks*. London, U.K.: UCL Press, 1997, pp. 13–16.

[26] H. M. Azamathulla, C. K. Chang, A. Ab. Ghani, J. Ariffin, N. A. Zakaria, and Z. Abu Hasan, "An ANFIS-based approach for predicting the bed load for moderately sized rivers," *J. Hydro-Environ. Res.*, vol. 3, no. 1, pp. 35–44, Jun. 2009.



**Mohamad Hazwan Mohd Ghazali** was born in Bukit Mertajam, Penang, Malaysia, in 1994. He received the bachelor's (Hons.) degree in manufacturing engineering with management and the master's degree in mechanical engineering from Universiti Sains Malaysia (USM) Engineering Campus, Nibong Tebal, Penang, in 2017 and 2019, respectively, where he is pursuing the Ph.D. degree in robotics and automation. His fields of study include artificial intelligence (AI), vibration analysis, and finite element analysis (FEA). His research interests are vibration analysis, intelligence systems, and unmanned aerial vehicles (UAVs).

**Wan Rahiman** (Member, IEEE) received the bachelor's degree from Cardiff University, U.K., specializing in manufacturing engineering, and the Ph.D. degree in fault detection and isolation for pipeline systems from the Control System Centre, University of Manchester, in 2009. After his bachelor's degree, he worked as an Engineer at Panasonic Manufacturing Ltd., Cardiff; and Lotus Cars Ltd., Norwich, for several years. Currently, he is a Senior Lecturer with the School of Electrical and Electronic Engineering, Universiti Sains Malaysia (USM) Engineering Campus, Penang, Malaysia, where he also serves as the Head for the Cluster of Smart Port and Logistic Technology (COSPALT). His research interests lie in the area of modeling nonlinear systems on a range of development research projects, particularly in drone and autonomous vehicle technologies.