# Motor Anomaly Detection for Unmanned Aerial Vehicles Using Reinforcement Learning

Huimin Lu, Yujie Li, Shenglin Mu, Dong Wang, Hyoungseop Kim, Seiichi Serikawa

***Abstract*— Unmanned aerial vehicles (UAVs) are used in many fields including weather observation, farming, infrastructure inspection, and monitoring of disaster areas. However, the currently available UAVs are prone to crashing. The goal of this paper is the development of an anomaly detection system to prevent the motor of the drone from operating at abnormal temperatures. In this anomaly detection system, the temperature of the motor is recorded using DS18B20 sensors. Then, using reinforcement learning, the motor is judged to be operating abnormally by a Raspberry Pi processing unit. A specially built user interface allows the activity of the Raspberry Pi to be tracked on a tablet for observation purposes. The proposed system provides the ability to land a drone when the motor temperature exceeds an automatically generated threshold. The experimental results confirm that the proposed system can safely control the drone using information obtained from temperature sensors attached to the motor.**

***Index Terms*— Unmanned aerial vehicle, Anomaly detection, Temperature sensor, Reinforcement learning**

## I. Introduction

In flight, an unmanned aerial vehicle (UAV) is unattended and controlled remotely. A drone is a type of UAV that is widely used for weather observation, spraying of agricultural chemicals, inspection of infrastructure, and monitoring of disaster areas. Rescue drones are used to capture live footage of a disaster area, provide first aid kits, and provide Wi-Fi links in a local area [18]. However, the use of drones in extreme environments raises a number of issues. In the USA alone, 418 accidents have been reported. Japan's Aeronautical Law requires drones to be operated safely when flying over a densely populated area [17]. To the best of our knowledge, however, no system has yet been devised to detect anomalies in the operation of the motor of a drone. In this paper, we develop a detection and protection system that lands the drone when an abnormal engine temperature is detected.

A common type of drone accident is the uncontrolled descent arising from issues such as insufficient battery capacity, motor malfunctioning or direct current (DC) problems, and loss of communication. To address these issues, an anomaly detection system is required. In this paper, we consider the case of motor abnormalities arising in flight. One key factor in the design life of a motor is the life of the bearings. This can be monitored in two ways. The first detects the deterioration of the grease due to heat. The second estimates the mechanical life by considering rolling fatigue. In the majority of cases, deterioration of the grease has a more significant role than mechanical failure. Temperature is the key factor influencing the service life of the bearing grease. We therefore monitor the temperature of the motor as a means of detecting abnormalities. This assumes that motor failure will usually be associated with temperature anomalies.

The proposed motor anomaly detection and crash prevention system uses temperature sensors and an ARM Cortex-A53 CPU (Raspberry Pi 2B). The temperature sensors are attached to the motor unit of the drone and abnormalities are detected using the Raspberry Pi located inside the drone. If an abnormal temperature is detected, the drone is landed automatically and flight is resumed if the temperature returns to normal. This system prevents the drone from crashing in extreme environments where failures can be catastrophic. The automatic motor anomaly detection system increases safety, allowing more active use of automatic piloting, and consequently decreasing crashes caused by communication loss and controller error. In this paper, instead of use a static threshold of temperature for abnormality detection, we use reinforcement learning for monitoring the temperature at each dynamic status. Different from supervised learning, reinforcement learning [15] is a kind of unsupervised learning methods. Therrien et al. [16] proposed a reinforcement learning and error-based processes for motor learning. To the best of our knowledge, the proposed system is the first work that solves the motor failure in real time without prior experience.

The remainder of this paper is organized as follows. In Section 2, we review related work on motor anomaly detection. Section 3 introduces the configuration of the proposed system. Experiments and results are reported in Section 4. Finally, Section 5 presents the conclusions.

This work was supported by Leading Initiative for Excellent Young Researcher (LEADER) of Ministry of Education, Culture, Sports, Science and Technology-Japan (16809746), Grant in Aid for Scientific Research of JSPS (17K14694), Research Fund of State Key Laboratory of Marine Geology in Tongji University (MGK1608), Research Fund of State Key Laboratory of Ocean Engineering in Shanghai Jiaotong University (1510), Research Fund of The Telecommunications Advancement Foundation, Fundamental Research Developing Association for Shipbuilding and Offshore and Strengthening Research Support Project of Kyushu Institute of Technology.

Huimin Lu, Hyoungseop Kim, and Seiichi Serikawa are with Kyushu Institute of Technology, 8048550, Japan (e-mail: luhuimin@ieee.org).

Yujie Li is with Yangzhou University, 225127 China (e-mail: liyujie@yzu.edu.cn).

Shenglin Mu is with the Ehime University, Japan (e-mail: mu.shenglin.du@ehime-u.ac.jp).

Dong Wang is with the Dalian University of Technology (e-mail: wdice@dlut.edu.cn).

## II. Related Work

Many methods are available for detecting motor anomalies. In this section, we summarize these methods and discuss the advantages and disadvantages of each.

### A. Abnormal Sound Detection

One type of system is based on abnormal sound detection [1] using case-based identification algorithms. As illustrated in Figure 1, in this method it is necessary to first calculate the sound data by feature vector and then apply an establishment distribution model, which is a form of supervised learning. Supervised learning requires a training dataset of sample elements and an appropriate classification for each of the samples. However, in the majority of cases it is difficult to train the datasets in an extreme environment. False detections therefore arise when using sound-based inspection systems against a noisy background.

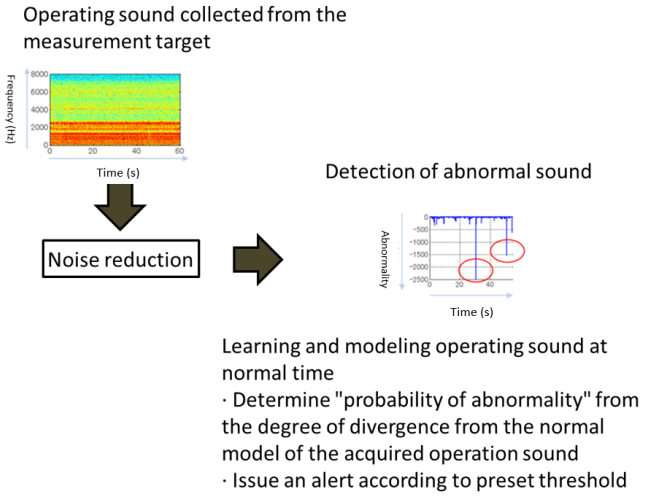



Fig. 1. Inspection system using abnormal sound detection.

### B. Infrared Imaging Detection

A second type of system collects infrared images of the motor and triggers an alarm when the temperature appears to equal or exceed a predetermined threshold [2, 3] (see Figure 2). A problem with this type of system is that it is difficult to mount a compact thermal imaging camera on a drone with sufficiently accurate infrared imaging to allow changes in temperature to be determined.

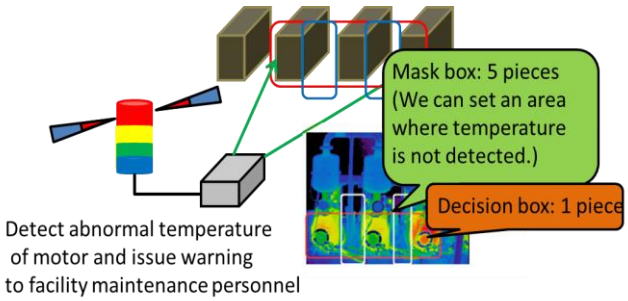



Fig. 2. Thermal imaging system based on abnormal temperature detection.

### C. Abnormal Rotation Detection

A third system uses a sensor to monitor the rotation speed of the fan and triggers an alarm when abnormal rotation is detected [4, 5] (see Figure 3). However, it is difficult to determine whether a reduction in rotation speed is due to an abnormality or a change in the air speed of the drone. The fan sensor also influences the flight characteristics of the drone.

As noted above, all conventional anomaly detection methods have drawbacks. This paper addresses the practicality of a drone-mounted anomaly detection system using temperature sensors (DS18B20) and a Raspberry Pi 2B.

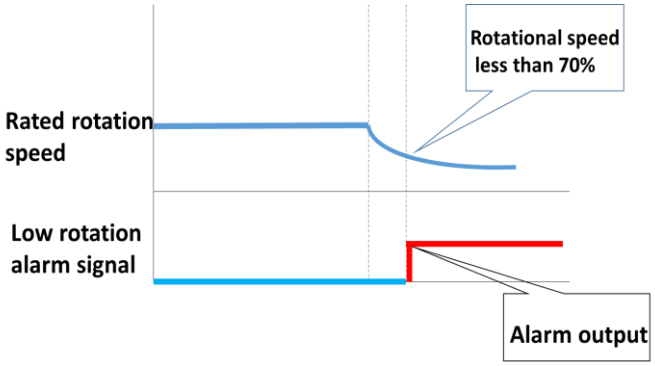



Fig. 3. Outline of the fan sensor system.

## III. System Configuration

The system developed in this paper is designed to land the drone if the motor temperature exceeds a threshold, defined as representing an abnormality. Previous studies have suggested that, in the case of a drone in sustained flight, this abnormal temperature is between 70 °C and 80 °C. As the flight time used in this work was as short as 10 min, 28 °C was adopted as the initial threshold.

### A. Overview

The experimental setup used in this research comprised two devices. The drone is equipped with temperature sensors and a Raspberry Pi 2B CPU; a Tablet is used to collect the observational data. The concept map of the system components is presented in Figure 4. We define the landing and flying model by the following equation.

$$\begin{cases} Landing: & t \geq \tilde{T} \\ Flying: & t < \tilde{T} \end{cases} \quad (1)$$

where $t$ is the real-time measured temperature and $\tilde{T}$ is the threshold temperature for determining deceleration. Notice that $\tilde{T}$ is not a fixed value, it changes with flight time. In Section III.C, we use reinforcement learning for determining the threshold temperature.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2737479, IEEE Internet of Things Journal



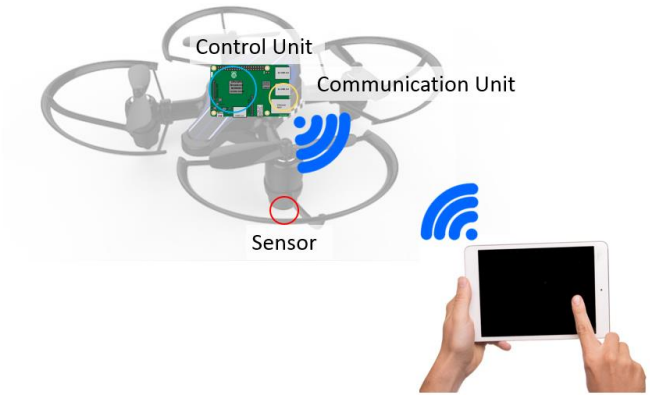



Fig. 4. System working diagram.

The processing flow of the proposed system is as follows:

- The Raspberry Pi is made operable from the Tablet;
- After takeoff, the drone assumes command from Raspberry Pi to drone;
- Drone advances after takeoff;
- The temperature of the motor is measured in flight;
- The drone is landed automatically when the measured temperature exceeds the threshold temperature.

## B. *Device Setting*

### *Drone Composition*

The experimental drone comprised four units: a detection unit (the temperature sensors), the drone itself, a control unit, and a communication unit. For the control unit, we propose a novel reinforcement-learning algorithm in the Raspberry Pi. This algorithm is described in detail in Section III.C. The communication unit implements Virtual Network Computing (VNC) to enable communication between the Raspberry Pi and the Tablet. The configuration is illustrated in Figure 5.

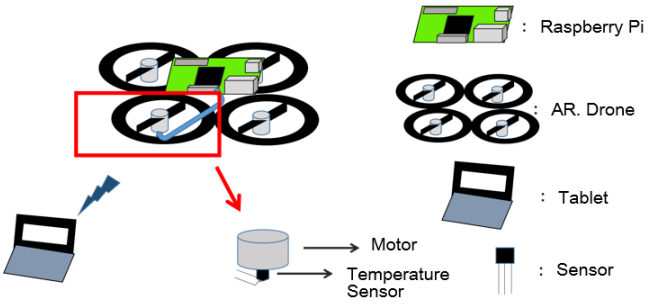



Fig. 5. Conceptual diagram of the system.

### *Detection Unit*

The detection circuit is displayed in Figure 6. The temperature sensor is attached to an exposed area of the motor at the lower side of the motor case, as indicated in Figure 7. As the 1-wire temperature sensor (DS18B20) has a built-in A/D converter, there is no signal deterioration due to the wiring length; the temperature data is converted to digital inside the sensor. In the Raspbian OS, this sensor is detected as a 1-wire device simply by loading the appropriate module.

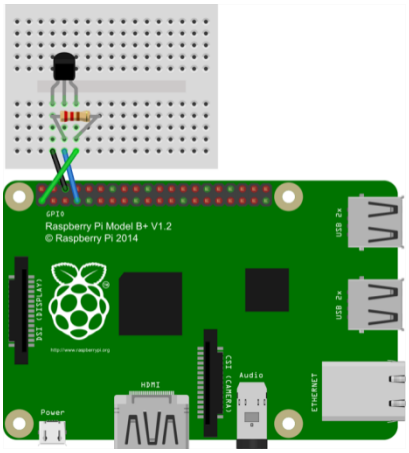



Fig. 6. Circuit diagram of the sensors.

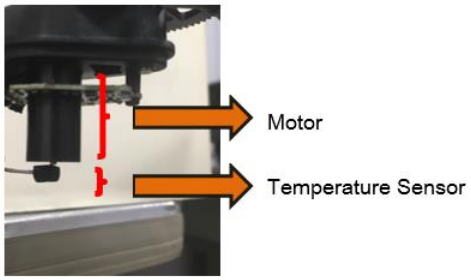



Fig. 7. The placement of the temperature sensor.

### *Control Unit*

A Raspberry Pi Model 2B is used in the control unit for processing. It was developed as a single-board computer for educational use and is less expensive than the standard Raspberry Pi. It is the lightest ARM series processor and is fully capable of meeting the requirements of light load and efficient real-time processing. A special feature is that it can use SD card storage rather than an internal hard drive. The control unit has two functions:

- To implement control of the drone by the Raspberry Pi;
- To decide whether to land the drone based on the data obtained from the temperature sensor.

### *Communication Unit*

A Wi-Fi connection is used to allow the Raspberry Pi to communicate with the drone as well as for the communication between the Tablet and drone using VNC [8]. The Service Set Identifier (SSID) of the drone used in this work was *ardrone2-132506* and the IP address was set as 192.168.1.1. The flowchart of the system is displayed in Figure 8. In this system, when the drone takes off, the Raspberry Pi automatically measures the temperature. If the real-time measured temperature $t$ is greater than the threshold temperature $\tilde{T}$, the drone is landed; otherwise, the drone continues to fly, monitoring the motor temperature.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2737479, IEEE Internet of Things Journal



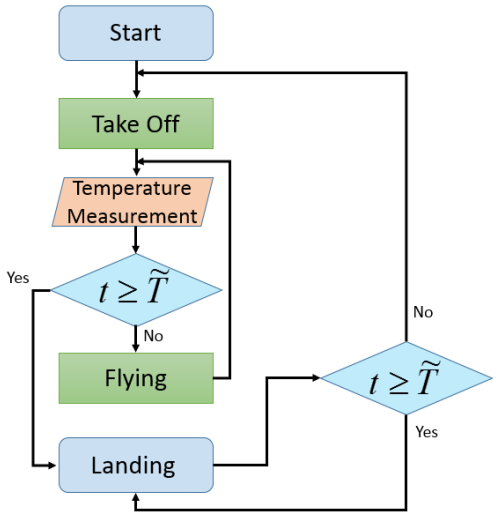



Fig. 8. Flowchart of abnormal temperature detection.

### C. Threshold Temperature Update by Reinforcement Learning

The algorithm adjusts the rate of the temperature increase based on changes of speed; this is the key for measuring the threshold temperature. In this paper, the optimal speed adjustment to suppress the rise in temperature is derived by comparison with the previously measured temperatures. Figure 9 is the flowchart of the system for detecting an increase in temperature.

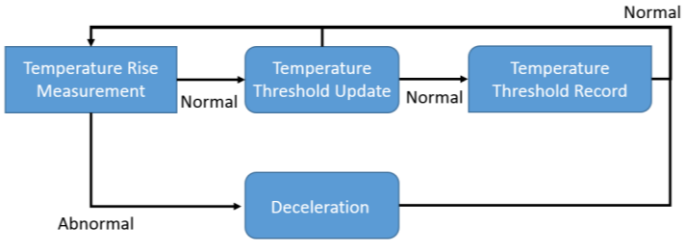



Fig. 9. Flowchart of the system for detecting abnormal increase in temperature and controlling drone speed.

We next introduce the details of the system for adjusting the speed of the drone, which is based on the detection of a rapid increase in temperature. First, let the temperature rise of the drone's motor be recorded every 2 s for five times be $\{T_1, T_2, \ldots, T_n\}$. The first five measurements are used to derive the average and standard deviation of the temperature rise. From the sixth measurement onward, the mean value $\bar{T}_{n+1}$ and the standard deviation $\sigma_{n+1}$ is used to define the threshold value.

$$\bar{T}_{n+1} = \frac{1}{n}\sum_{i=1}^{n} T_i \quad i = 1, \ldots, n \tag{2}$$

$$\sigma_{n+1} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(T_{n+1} - \bar{T}_{n+1})^2} \tag{3}$$

To update the threshold temperature and record the deceleration value, we have the following definitions.

***Definition* 1:** *If the measured value $t$ is greater than or equal to the sum of the mean values $\bar{T}_{n+1}$ and standard deviation $\sigma_{n+1}$, a rising abnormal condition exists. The drone performs deceleration.*

***Definition* 2:** *If the measured value $t$ is less than the sum of the mean values $\bar{T}_{n+1}$ and standard deviation $\sigma_{n+1}$, yet is greater than the subtraction of the mean values $\bar{T}_{n+1}$ and standard deviation $\sigma_{n+1}$, a normal condition exists. The drone continues to fly and updates the threshold temperature $\widetilde{T}$.*

***Definition* 3:** *If the measured value $t$ is less than the subtraction of the mean values $\bar{T}_{n+1}$ and standard deviation $\sigma_{n+1}$, an ideal condition exists. The drone continues to fly and records the deceleration value.*

According to the above definitions, based on the measured values and previous statistic values, the proposed system has three conditions. These are defined as the ideal condition, normal condition, and rising abnormal condition. We discuss the processing in the case of a rising abnormal condition and ideal condition together. There exist two types of condition for becoming an ideal condition: (1) the drone is in transition from a normal condition to an ideal condition or is in an ideal condition. It must only update the threshold temperature; (2) the drone is in transition from an abnormal condition to an ideal condition. The temperature increase rate is less than that already recorded; the temperature increase rate, threshold temperature value, and temperature increase rate are recorded. Finally, the deceleration value is recorded as the minimal value of the deceleration formed by the deceleration.

In the processing, from counts 1 to 25 (the first 50 s of measurement), the temperature value at a fixed time is rising abnormally from the initial translational speed 0.05 m/s in the normal condition as the drone speed increases. Figure 10 displays the relationship between time and motor temperature. Decelerate from the temperature increase rate rapidly as shown in Figure 11. By comparing the temperature rise under deceleration in the ascending abnormal condition with that in the ideal condition, the optimum deceleration value can be determined. For counts 1 to 25, we define the deceleration rate as $(0.001 + 0.0005 \times n)$ m/s when the *n-th* abnormality occurs. Let $x_0$ be the optimal value for deceleration at the time between counts 1 to 25. The optimal value of deceleration is 0.008 ($n$=14). Figure 11 displays the relationship between the time and speed changes.

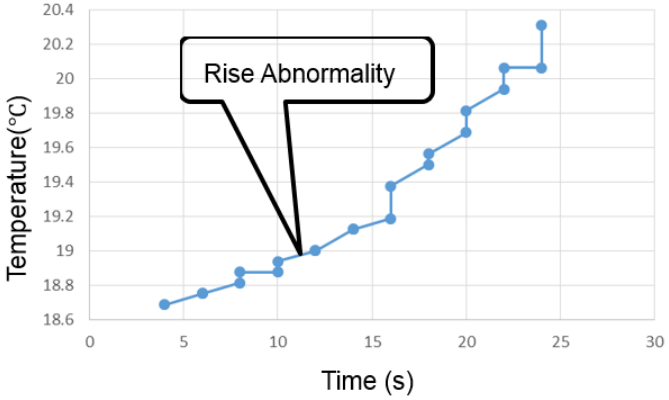



Fig. 10. Relationship between time and motor temperature (from 0 s to 50 s).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2737479, IEEE Internet of Things Journal



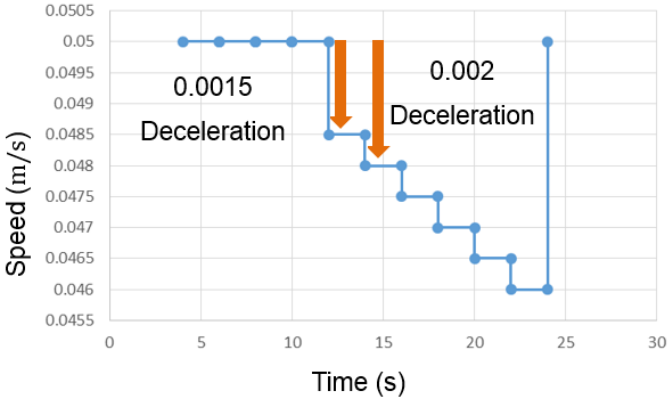



Fig. 11. Relationship between time and speed (from 0 s to 50 s).

From Figure 10 and Figure 11, we determined that the temperature rising rate exceeds the threshold of temperature rising rate between 10 s and 12 s and the deceleration process is initiated. The deceleration process continues for 22 s. From Figure 11, we determined that the deceleration rate increases continuously (from 0.0015 m/s to 0.002 m/s).

Then, for counts 26 to 35 (from 51 s to 70 s of the measurement), we define the deceleration rate $x_1$ as ($x_0$ + 0.0001×$n$) m/s when the *n-th* abnormality occurs. The change of motor temperature and speed are indicated in Figure 12 and Figure 13, respectively.

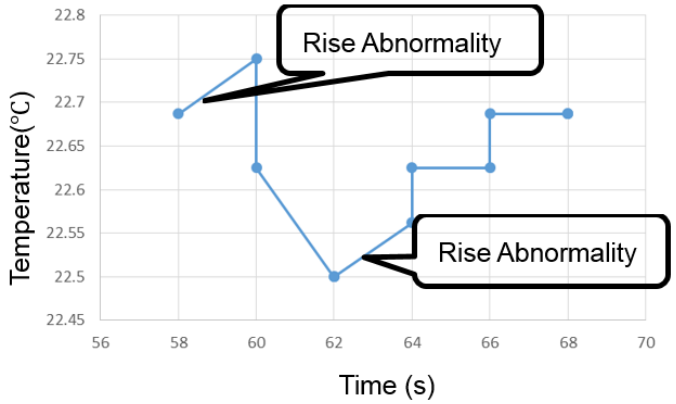



Fig. 12. Relationship between time and motor temperature (from 51 s to 70 s).

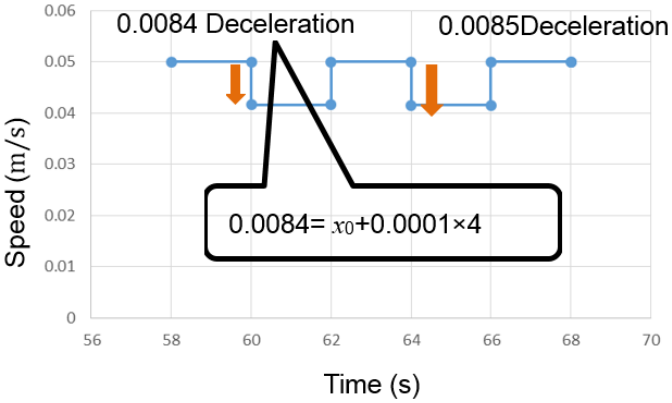



Fig. 13. Relationship between time and speed (from 51 s to 70 s).

For counts 36 to 45 (from 71 s to 90 s of the measurement), we consider $x_0$ as the optimal value and set the deceleration value as -0.0001. Therefore, the speed at the time of the *n-th* abnormality is a deceleration rate $x_2$ obtained by ($x_0$ - 0.0001×$n$) m/s from the initial speed 0.05 m/s. The change of motor temperature and speed are displayed in Figure 14 and Figure 15, respectively.

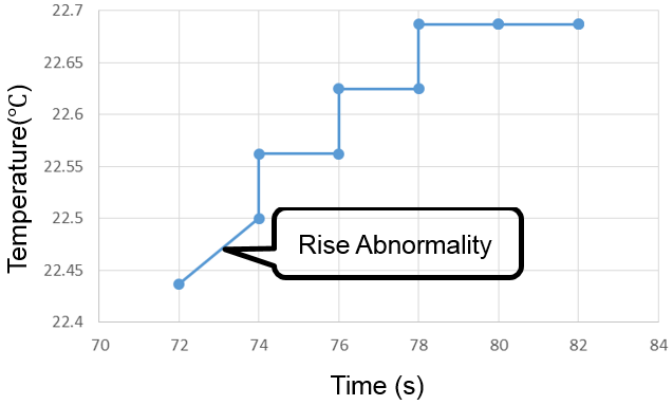



Fig. 14. Relationship between time and motor temperature (from 71 s to 90 s).

Finally, we have three values from 0 s to 90 s for detecting abnormal motor temperature. We choose the optimal threshold temperature value of the motor as:

$$\tilde{T} = \min\{T_{x0}, T_{x1}, T_{x2}\}. \quad (4)$$

where $T_x$ is the average temperature change of the motor at different deceleration rates of different operation conditions.

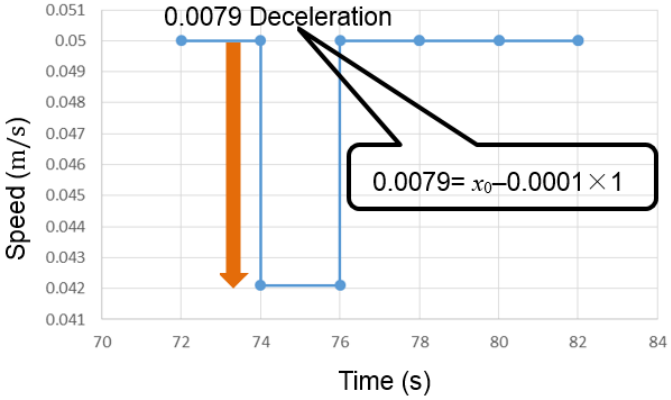



Fig. 15. Relationship between time and speed (from 71 s to 90 s).

From (4), we determined that the optimal threshold value of the motor temperature changes with time. This is superior to using a static threshold temperature for detecting.

## IV. Simulation Results

In this simulation experiment, a system was constructed to land the drone when the motor temperature exceeded a certain value. Simulations confirmed that this objective was achieved. When the system was operated with an initial threshold of 28°C, the Raspberry Pi could detect abnormalities and the drone was landed. We calculated the deceleration rates, for determining and testing the system performance.

### *A. Calculation of Deceleration Rates*

In the rising abnormality state, the process changes based on the number of times the temperature rise is measured. The real time speed and the changes of temperature of counts 1 to 25 are presented in Table I.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2737479, IEEE Internet of Things Journal



Table I. Speed and temperature changes from 0 s to 50 s.

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **1** | 0.05 | -0.125 |
| **2** | 0.05 | -0.125 |
| **3** | 0.05 | 0.063 |
| **4** | 0.05 | 0.062 |
| **5** | 0.05 | -0.125 |
| **6** | 0.05 | 0.063 |
| **7** | 0.0485 | 0.125 |
| **8** | 0.048 | 0.062 |
| **9** | 0.0475 | 0.125 |
| **10** | 0.047 | 0.125 |
| **11** | 0.0465 | 0.125 |
| **12** | 0.046 | 0 |
| **13** | 0.05 | 0.125 |
| **14** | 0.0455 | 0 |
| **15** | 0.05 | 0.188 |
| **16** | 0.045 | 0.125 |
| **17** | 0.0445 | 0.188 |
| **18** | 0.044 | 0.188 |
| **19** | 0.0435 | 0 |
| **20** | 0.05 | 0.125 |
| **21** | 0.043 | 0.125 |
| **22** | 0.0425 | 0 |
| **23** | 0.05 | 0.062 |
| **24** | ***0.042*** | ***-0.188*** |
| **25** | 0.05 | 0.25 |

From Table I, we identified that the minimum temperature change was -0.188°C, where the optimal deceleration rate was 0.008 m/s (= 0.05 m/s - 0.042 m/s). In the next experiment, we monitored the real time speed and the changes of temperature of counts 26 to 35; this is displayed in Table II.

Table II. Speed and temperature changes from 51 s to 70 s.

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **26** | 0.0415 | 0.063 |
| **27** | 0.0419 | 0.125 |
| **28** | 0.0418 | 0.125 |
| **29** | 0.0417 | 0 |
| **30** | 0.05 | 0.063 |
| **31** | ***0.0416*** | ***-0.125*** |
| **32** | 0.05 | 0.062 |
| **33** | 0.0415 | 0 |
| **34** | 0.05 | 0 |
| **35** | 0.05 | -0.125 |

From Table II, we determined that the minimum temperature change below 0.008 m/s at the first time measurement was -0.125 °C, where the optimal deceleration rate was 0.0084 m/s (= 0.05 m/s - 0.0416 m/s). In the next experiment, we monitored the real time speed and the changes of temperature of counts 36 to 45; this is displayed in Table III.

Table III. Speed and temperature changes from 71 s to 90 s.

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **36** | 0.05 | -0.125 |
| **37** | 0.05 | 0.063 |
| **38** | ***0.0421*** | ***0*** |
| **39** | 0.05 | 0 |
| **40** | 0.05 | 0 |
| **41** | 0.05 | 0 |
| **42** | 0.05 | -0.25 |
| **43** | 0.05 | 0 |
| **44** | 0.05 | -0.062 |
| **45** | 0.05 | -0.188 |

From Table III, we identified that the minimum temperature change above 0.008 m/s at the first time measurement was 0 °C, where the optimal deceleration rate was 0.0079 m/s (= 0.05 m/s - 0.0421 m/s).

### *B. Calculation of Optimal Deceleration Rate*

In the above stage, we obtained three deceleration rates (0.008 m/s, 0.0084 m/s, and 0.0079 m/s) in different operation conditions. To calculate the optimal deceleration rate, we set the three deceleration rates for testing. In the test experiment (from counts 46 to 55), we obtained the results in Table IV.

Table IV. Speed and temperature changes from counts 46 to 55 (Deceleration rate is 0.008 m/s).

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **46** | 0.05 | 0.125 |
| **47** | 0.042 | -0.062 |
| **48** | 0.05 | 0.063 |
| **49** | 0.042 | 0 |
| **50** | 0.05 | -0.063 |
| **51** | 0.05 | -0.188 |
| **52** | 0.05 | 0.063 |
| **53** | 0.042 | 0 |
| **54** | 0.05 | -0.125 |
| **55** | 0.05 | 0.125 |
| **Average Temp. Changes** | | ***-0.062*** |

From Table IV, we determined that the average temperature change was -0.062 °C at the deceleration rate of 0.008 m/s. In Table V and Table VI, we indicate that the average change was 0.125 °C at the deceleration rate of 0.0084 m/s and the average change was -0.061 °C at the deceleration rate of 0.0079 m/s, respectively.

Table V. Speed and temperature changes from counts 56 to 65 (Deceleration rate is 0.0084 m/s).

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **56** | 0.042 | 0.063 |
| **57** | 0.0416 | -0.188 |
| **58** | 0.05 | 0.062 |
| **59** | 0.0416 | 0 |
| **60** | 0.05 | 0.063 |
| **61** | 0.0416 | -0.125 |
| **62** | 0.05 | 0 |
| **63** | 0.05 | 0.187 |
| **64** | 0.0416 | -0.125 |
| **65** | 0.05 | 0.188 |
| **Average Temp. Changes** | | ***0.125*** |

Table VI. Speed and temperature changes from counts 66 to 75 (Deceleration rate is 0.0079 m/s).

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **66** | 0.0416 | 0.125 |
| **67** | 0.0421 | -0.125 |
| **68** | 0.05 | -0.187 |
| **69** | 0.05 | 0.063 |
| **70** | 0.0421 | -0.062 |
| **71** | 0.05 | 0.062 |
| **72** | 0.0421 | -0.125 |
| **73** | 0.05 | 0.063 |
| **74** | 0.0421 | 0.125 |
| **75** | 0.0324 | 0 |
| **Average Temp. Changes** | | ***-0.061*** |

From Table IV to Table VI, we can conclude that the deceleration rate 0.008 m/s is the optimal deceleration value,

having the lowest average temperature change, -0.062 ℃.

### C. Experimental Certification

To confirm the effectiveness of this result, we tested the optimal deceleration value 0.008 m/s in the conditions of climbing and falling of the drone. The results are presented in Table VII and Table VIII.

Table VII. Speed and temperature changes from counts 76 to 100 under climbing condition of the drone.

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **76** | 0.05 | 0.562 |
| **77** | 0.05 | 0.625 |
| **78** | 0.05 | 0.5 |
| **79** | 0.05 | 0.313 |
| **80** | 0.05 | 0.188 |
| **81** | 0.05 | 0.5 |
| **82** | 0.05 | 0.313 |
| **83** | 0.05 | 0.437 |
| **84** | 0.05 | 0.313 |
| **85** | 0.05 | 0.312 |
| **86** | 0.05 | 0.25 |
| **87** | 0.05 | 0.25 |
| **88** | 0.05 | 0 |
| **89** | 0.05 | 0.437 |
| **90** | 0.05 | 0.625 |
| **91** | 0.05 | 0.125 |
| **92** | 0.05 | 0.375 |
| **93** | 0.05 | 0.375 |
| **94** | 0.05 | 0.313 |
| **95** | 0.05 | 0.062 |
| **96** | 0.05 | -0.062 |
| **97** | 0.05 | 0.25 |
| **98** | 0.05 | 0.063 |
| **99** | 0.05 | -0.125 |
| **100** | 0.05 | 0.187 |

Table VIII. Speed and temperature changes from counts 101 to 125 under falling condition of the drone.

| Count | Speed (m/s) | Temperature changes (°C) |
|---|---|---|
| **101** | 0.05 | 0 |
| **102** | 0.05 | 0 |
| **103** | 0.05 | 0.063 |
| **104** | 0.05 | 0.313 |
| **105** | 0.05 | 0.375 |
| **106** | 0.05 | 0.438 |
| **107** | 0.042 | 0.25 |
| **108** | 0.05 | 0.375 |
| **109** | 0.042 | 0.375 |
| **110** | 0.034 | 0.375 |
| **111** | 0.026 | 0.187 |
| **112** | 0.05 | 0.25 |
| **113** | 0.05 | 0.375 |
| **114** | 0.042 | 0.375 |
| **115** | 0.034 | 0.313 |
| **116** | 0.05 | 0.375 |
| **117** | 0.042 | 0.25 |
| **118** | 0.05 | 0.125 |
| **119** | 0.05 | 0.375 |
| **120** | 0.042 | 0.312 |
| **121** | 0.05 | 0.125 |
| **122** | 0.05 | -0.063 |
| **123** | 0.05 | -0.063 |
| **124** | 0.05 | 0.438 |
| **125** | 0.042 | -0.063 |

From Table VII, we identified that the temperature increased from 19.375 ℃ to 27.687 ℃, which is less than the initial threshold of 28 ℃. From Table VIII, we determined that the temperature of the motor increased from 18.75 ℃ to 25.312 ℃, which is also below the initial threshold of 28 ℃. Hence, we confirmed that the optimal deceleration value of 0.008 m/s can suppress a temperature increase.

## V. Conclusion

In this paper, we proposed a reinforcement learning-based motor-temperature anomaly detection system for UAVs. Motor malfunction is a major cause of drone crashes. The proposed system employed temperature sensors and a Raspberry Pi CPU for processing. In simulations using a dynamic motor-temperature threshold, execution of successful automatic landing was demonstrated. This system facilitates the avoidance of motor failure, allowing future drones to fly for longer periods. In future, we will consider implementing deep learning methods [6, 7, 11–14] to select the threshold temperature. Further, we will attempt to use PID [9, 10] rather than Raspberry Pi for reducing the weight and improving the computing ability.

## References

[1] NTT DATA “Started providing IOT solution to detect abnormality from working sound,” NTT DATA,2016/10/3, https://latte.la/column/43321221

[2] Nippon Avionics Co., Ltd, “K-1. Motor abnormal temperature detection system -Infrared thermography- Nippon Avionics,” Nippon Avionics Co., Ltd, 1999, http://www.avio.co.jp/products/infrared/solution/k-heavy/k-01.html

[3] Hirozawa Seiki Seisakusho Co., Ltd, “Metal Feather Fan Sensor,” http://www.hirosawaseiki.co.jp/ikurafan/pages/Cat/IkuraFan1406/textunderscore 084.pdf

[4] Oriental Motor Co., Ltd, “Design life of motor -Technical data-,” Oriental Motor Co., Ltd, https://www.orientalmotor.co.jp/tech/reference/life01/

[5] Tamura Yuuto, “Detection and deterioration diagn0.313osis of electrical equipment based on current waveform pattern analysis,” http://vision.kyoto-u.ac.jp/japanese/happyou/pdf/Y-Tamura/textunderscore ASN/textunderscore 20131129.pdf

[6] Y Li, H Lu, J Li, X Li, Y Li, S Serikawa, “Underwater image de-scattering and classification by deep neural network,” Computers and Electrical Engineering, vol.54, pp.68-77, 2016.

[7] H Lu, B Li, J Zhu, Y Li, Y Li, X Xu, L He, X Li, J Li, S Serikawa, “Wound intensity correction and segmentation with convolutional neural networks,” Concurrency and Computation: Practice and Experience, vol.29, no.6, e3927, 2017.

[8] Raspberry Pi Setting, http://yamaryu0508.hatenablog.com/entry/2014/08/16/202441

[9] H. Lu, S. Serikawa, “Design of freely configurable safety light curtain using hemispherical mirrors,” IEEJ Transactions on Electrical and Electronic Engineering, vol.8, no.S1, pp.110-111, 2013.

[10] H. Lu, Y. Li, Y. Li, S. Serikawa, H. Kim, “Highly accurate energy-conserving flexible touch sensors,” Sensors and Materials, vol.29, no.6, pp.1-7, 2017.

[11] M. Chen, Y. Hao , K. Hwang, L. Wang, L. Wang, “Disease Prediction by Machine Learning over Big Healthcare Data,” IEEE Access, vol. 5, no. 1, pp. 8869-8879, 2017.

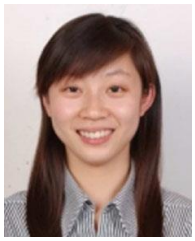
[12] M. Chen, J. Yang, Y. Hao, S. Mao, K. Hwang, “A 5G Cognitive System for Healthcare,” Big Data and Cognitive Computing, vol. 1, no. 1, DOI:10.3390/bdcc1010002, 2017.

[13] M. Chen, Y. Qian, S. Mao, W. Tang, X. Yang, “Software-defined mobile networks security,” Mobile Networks and Applications, vol.21, no.5, pp.729-743, 2016.

[14] M. Chen, Y. Ma, Y. Li, D. Wu, Y. Zhang, C. Youn, “Wearable 2.0: Enabling Human-Cloud Integration in Next Generation Healthcare Systems,” IEEE Communications Magazine, vol.55, no.1, pp.54-61, 2017.

[15] R. S. Sutton, A. G. Barto. Reinforcement Learning, 2016.
[16] A.S. Therrien, D.M. Wolpert, A.J. Bastian, "Effective reinforcement learning following cerebellar damage requires a balance between exploration and motor noise," Brain, vol.139, pp.101-114, 2016.
[17] K. Yang, Q. Han, H. Li, K. Zheng, Z. Su, X. Shen, "An efficient and fine-grained big data access control scheme with privacy-preserving policy," IEEE Internet of Things Journal, vol.4, no.2, pp.563-571, 2017.
[18] D. Chen, N. Zhang, Z. Qin, X. Mao, Z. Qin, X. Shen, X. Li, "S2M: a lightweight acoustic fingerprints-based wireless device authentication protocol," IEEE Internet of Things Journal, vol.4, no.1, pp.88-100, 2017.

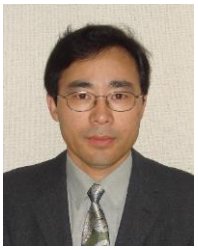
**Huimin Lu** received a B.S. degree in Electronics Information Science and Technology from Yangzhou University in 2008. He received M.S. degrees in Electrical Engineering from Kyushu Institute of Technology and Yangzhou University in 2011. He received a Ph.D. degree in Electrical Engineering from Kyushu Institute of Technology in 2014. From 2013 to 2016, he was a JSPS research fellow (DC2, PD, and FPD) at Kyushu Institute of Technology. Currently, he is an Assistant Professor in Kyushu Institute of Technology and an Excellent Young Researcher of MEXT-Japan. His research interests include computer vision, robotics, artificial intelligence, and ocean observing.



**Yujie Li** received the B.S. degree in Computer Science and Technology from Yangzhou University in 2009. She received M.S. degrees in Electrical Engineering from Kyushu Institute of Technology and Yangzhou University in 2012, respectively. She received a Ph.D. degree from Kyushu Institute of Technology in 2015. Currently, she is a Lecturer in Yangzhou University. Her research interests include computer vision, sensors, and image segmentation.

**Shenglin Mu** received his B.E. degree in Mechanical Engineering and Automation in 2007 from the Northeastern University, China, and his M.E. and Ph.D. degrees in Electronic and Information Systems Engineering in 2010 and 2013, respectively, from the Yamaguchi University, Japan. In 2013, he joined the Department of Electronic Control Engineering, National Institution of Technology, Hiroshima College. In 2017, he joined the Graduate School of Science and Engineering, Ehime University. His current research interests include control engineering, especially intelligent control, intelligent algorithms, mechatronics, robotics, and sensing technology.

**Dong Wang** received the B.E. degree in electronic information engineering and the Ph.D. degree in signal and information processing from Dalian University of Technology (DUT), Dalian, China, in 2008 and 2013, respectively. He is currently an Associate Professor with the School of Information and Communication Engineering, DUT. His current research interests include face recognition, person re-identification, and object tracking.



**Hyoungseop Kim** received a B.A. degree in Electrical Engineering from Kyushu Institute of Technology in 1994, Master and Ph.D. degrees from Kyushu Institute of Technology in 1996 and 2001, respectively. He is a Professor in the Department of Control Engineering at Kyushu Institute of Technology. His research interests are focused on medical application of image analysis.

**Seiichi Serikawa** received B.S. and M.S. degrees in Electronic Engineering from Kumamoto University in 1984 and 1986. He received a Ph.D. degree in Electronic Engineering from Kyushu Institute of Technology in 1994. Currently, he is a Vice President of Kyushu Institute of Technology and also serves as a Professor in Center for Socio-Robotic Synthesis and the Department of Electrical and Electronic Engineering. His current research interests include computer vision, sensors, and robotics.