Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4328307, 12 pages
https://doi.org/10.1155/2022/4328307

Research Article

Fault Detection Method for Wi-Fi-Based Smart Home Devices

Kefei Cheng,[1] Jiashun Xu,[2] Liang Zhang,[1] ChengXin Xu,[1] and Xiaotong Cui[1]

[1]*School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing, China*
[2]*College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China*

Correspondence should be addressed to Jiashun Xu; xjscqupt@163.com

Received 2 November 2021; Revised 12 September 2022; Accepted 8 October 2022; Published 2 November 2022

Academic Editor: Michele Girolami

At present, the dynamic nature and unstable network connections in the deployment environments of Wi-Fi-based smart home devices make them susceptible to component damage, crashes, network disconnections, etc. To solve these problems, researchers have used various fault detection methods, such as alarming when monitored fault parameters exceed the preset values, model-based mathematical methods, device signal processing-based methods, and artificial intelligence-based methods. However, these methods require large numbers of fault parameters, the model are complex, and their fault detection accuracy is relatively poor. To more quickly and accurately detect faults in smart home devices and ensure the continuity of people's daily work and lives, this paper analyzes both the Wi-Fi traffic characteristics of smart home devices and the complexity and difficulty of traditional fault detection methods and proposes a fault detection method based on TDD (Throughput and Delay Distribution). This method obtains throughput and data packet delay distribution by capturing Wi-Fi communication and sending test data. By dividing the throughput into heartbeat data and command information, we can calculate the real-time throughput and further calculate the similarity between the real-time throughput and the throughput in database. Also, the resulting delay distribution is compared with the probability distribution of delay in the database. When the throughput values are sufficiently similar and the delays are all in the normal range, the smart home secure devices are functioning properly. The experimental results show that the proposed TDD method can detect faults in household devices in real time and that it achieves high recall and good detection accuracy in Wi-Fi communication environment.

1. Introduction

A smart home system is an intelligent, comfortable, convenient, and energy-saving household environment that connects various automated household appliances through routing and network communication technologies [1]. Due to the low cost and connectivity convenience of Wi-Fi, it has become increasingly popular for use with smart home devices and people can manage their appliances remotely via their mobile phones by Wi-Fi. In this paper, we work on methods to improve the fault detection capabilities of smart home devices that use Wi-Fi technology as their communication approach.

Due to its multiplicity of functions, unstable network connections, dynamic deployment environments, and vulnerable Wi-Fi networks, smart home devices based on Wi-Fi are prone to failure. Such problems negatively affect users' work and lives and even can result in substantial damage. A reliable fault detection method should be able to discover the status of a smart home device or its components in real time and take corresponding measures when faults occur. Consequently, research on fault detection methods has become an important issue.

The existing methods of fault detection of such devices mainly include methods that monitor fault parameters, methods using mathematical models, methods using signal processing, and methods using artificial intelligence. Despite their ubiquity, the above methods have rarely been applied directly to smart home devices. Few researchers have worked on fault detection for smart home devices.

In this paper, we conducted an experiment aimed at determining which people impose the highest security requirements on different types of smart home devices. According to Wi-Fi traffic features, we developed a fault detection algorithm using TDD that can detect fault problems in real time. It requires only two fault parameters but

is able to accurately detect network faults on smart home devices. Moreover, this method applies to the overwhelming majority of smart home devices and does not require complicated mathematical models. The main contributions of this paper are summarized as follows.

(1) We proposed a fault detection method based on throughput. It captures the Wi-Fi traffic between smart home devices and smart home servers and divides the traffic into command information and heartbeat communication data. According to the packet size of command information and heartbeat communication data, the ideal throughput and real throughput are calculated. The difference between them exceeding a threshold is judged as faulty

(2) We proposed a fault detection method based on a Gaussian distribution which is a further fault judgment method based on the throughput-based fault detection method. When the difference between the ideal throughput and the real throughput is less than the threshold, the throughput-based fault judgment method is not sufficient to judge that the device is fault-free, and further judgments need to be made based on the delay. The proposed Gaussian distribution-based fault detection method extracts the delay of each data packet of the test communication flow and obtains the original delay probability distribution data. After that, if the throughput of the smart home device traffic is normal and the delay is distributed within the normal range, it can be judged as normal; otherwise, it can be judged as a fault

(3) In order to verify the effectiveness of the method in this paper, we compare our method with the detection method based on particle swarm algorithm and Gaussian distribution. We extract 1000 sets of data samples of existing smart home equipment kits in real time and carry out actual fault marking to obtain false-positive rate and false-negative rate of the two methods. At last, we verify the two methods for fault detection recall ratio and accuracy ratio and find that the proposed method in this paper has better recall ratio and accuracy ratio

2. Related Works

As living standard has improved, the use of smart devices has increased rapidly in daily life. However, these smart devices introduce additional potential security risks, which makes fault detection and diagnosis technology of smart home devices important.

Currently, there exists four methods of fault detection of smart home devices.

Methods that monitor fault parameters: when various parameters exceed a threshold value, the device should report a fault. Kim et al. [2] proposed an early fault detection method using Laplace trend statistics to monitor fault parameters that achieved some success. In 2016, Sahoo et al. [3] showed that it was possible to detect faults by reporting values that exceeded threshold parameters. However, an accurate threshold value is difficult to determine, and this approach also lacks data comparison and prediction.

Methods using mathematical models: in 2005, the generalized parity vector (GPV) extension model was proposed by Omana et al. [4], and a system fault detection and isolation method based on this GPV model was demonstrated in 2007 [5]. An airplane structural fault detection method used a numerical model to perform simulation and modeling; Fernando et al. [6] verified the effectiveness of this method. These types of methods can achieve real-time monitoring with high accuracy, but they are all difficult to implement.

Methods using signal processing: a device signal process technique based on time series analysis was proposed [7]; this technique enhanced the flags' ability to detect signal problems. In 2015, Qiao et al. [8] studied the signals and processing methods used in wind turbine status monitoring and fault detection. They comprehensively analyzed the performance of this type of method. Despite the good adaptability of this approach, it is difficult to accurately determine the relationships between signals.

Methods using artificial intelligence: there are two types: symbolic reasoning methods and numerical computation methods. In [9], the authors provided a comprehensive introduction to the fault detection field and its unsolved problems. In [10], the authors summarized various artificial intelligence methods and their applications in fault detection and location. In 2016, the authors of [11] used an artificial neural network method to detect faults which was effective and quick. However, the symbolic reasoning method lacks a valid expression, while the numerical computation method performs poorly at identifying various anomalous modes.

Although the above methods are representative, they are rarely applied in the detection of faults in smart home devices. The authors of [12] suggested that uncertainty exists in the monitoring and diagnosis of smart homes and fault detection for smart home devices is important. Son et al. [13] designed a fault diagnosis system that reverse-tracked the network fault—an approach that was both complicated and required large amounts of data. Ye et al. [14] proposed a fault detection method which achieved high accuracy but cannot detect missing events. In 2015, Hsieh et al. [15] designed a model that could be used to locate faults. However, the modeling process was complicated, and the accuracy was only passable.

In addition, fault detection and diagnosis technology is also an important academic topic that can be classified into 3 types: signal processing-based methods, mathematical model-based methods, and knowledge-based smart detection and diagnosis methods. The common point among these model types is that they all extract corresponding device information, analyze it, and then use a model or algorithm to achieve fault detection. Overall, we can divide the detection method into two steps: a parameter selection and analysis step and a model and algorithm analysis step.

To detect and diagnose faults, data and parameter selection is an essential step. In the available electronic device

fault detection technology literature, velocity, torque, noise level, and vibration frequency have all been used as parameters [16]. Sometimes, totally different technical parameters, such as thermal measurements and chemical analysis value, have been used to detect the degree and character of faults. The authors of [17] used sensor data as parameters, calculated their weighted averages, and performed data integration via the Dempster combination rule to implement available fault detection proving that fault detection and diagnosis can be carried out effectively. In [18], the authors constructed a multiple feature model to detect bearing faults that included the complex envelope spectrum, time frequency, and wavelet packet analysis as parameters. This study showed that multiple features could be combined to optimize a fault detection system. The authors of [19] proposed a multiple-feature-based layered dynamic fault detection method that extracted features via a differential-based feature extraction method and observed their dynamic trends. Then, based on these observations, a layered detection standard was proposed. Parameter selection can directly influence the final fault detection result (FDR). Thus, our goal is to create a detection method that is both simple and accurate.

Model and algorithm analysis methods include numerical analysis, mathematical model analysis, and various artificial intelligence algorithms, such as neural networks. The authors of [20] used a standard partial least squares approach to detect device faults that divided parameters into two parts, related and unrelated, and then designed a corresponding testing statistic to provide useful fault diagnosis information. This method achieved a high FDR; however, the parameter division requirement was difficult to implement and involved a complicated algorithm. Recently, many fault detection methods have chosen Gaussian distribution functions to analyze feature parameters [21–24]. A Gaussian distribution is also called the normal distribution function. Its distribution curve clearly shows a peak point, and the curve is bilaterally symmetrical based on the axis of the peak point [22]. The authors of [21] proposed a Gaussian process-model-based fault detection method that used a type of nonlinear regression algorithm. The classical regression algorithm always produces a forecasting probabilistic model by applying Gaussian modeling according to prior parameters and post conditions. The Gaussian modeling process in this method employed the maximum likelihood estimate, which complicated the process. In [24], the authors used distributed computing technology to form a Gaussian mixture model that reduced the number of required feature parameters and achieved better fault detection; however, it requires many iterative computations. For devices whose feature parameters do not have a Gaussian distribution, many researchers have used Gaussian fitting to perform data processing [25, 26]. In addition, some researchers in industrial fields have adopted machine learning methods to detect faults, among which neural networks are the most commonly used [27, 28]. In [27], the authors extracted the important feature parameters by using mathematical analysis, used them to train an artificial neural network, and then applied the artificial neural network to classify faults. They demonstrated the reliability of their method through the experiments in [27].

According to the above related work, fault detection technology needs to select the appropriate number of parameters or data and then establish a related model or perform calculations using an algorithm [17–19]. Inspired by the above work, our work focuses on establishing an easily formed method with limited parameters and a simple algorithm to perform fault detection. To achieve smart home fault detection, we propose a new fault detection method suitable for smart homes. This method dynamically captures data packets between smart home devices and home routers, uses throughput and delay as parameters and Renyi cross-entropy and Gaussian distribution for the model and algorithm, and finally achieves an effective and accurate smart home fault detection method.

3. Framework of the Developed Method

In this study, we developed a fault detection method for Wi-Fi-based smart home devices that uses Throughput and Delay Distribution. Figure 1 shows a flowchart of the proposed scheme. To implement this method, we first captured Wi-Fi traffic between the router and each smart home secure device. By classifying and identifying the traffic information, each smart home device can be recognized by a method that combines the port number and PPLD [29]. Based on the data packet size, the probability of each data flow, and the acknowledge packet to server for each data flow, heartbeat data can be distinguished from command information. In addition, a marked probe data packet was sent to the smart home devices to statistically estimate the data packet delay distribution. By comparing the resulting delay distribution with the probability distribution of delay in the database (which contained a probability distribution image of delay generated using large amounts of data flow record under both normal and abnormal network conditions), network problems in smart home devices could be detected. To calculate the throughput during a certain period under the premise of a strict heartbeat cycle, the sizes of the heartbeat and command information packets were determined; then, the throughput calculated in real time was compared with that in the database and the similarity between them was calculated using Renyi cross-entropy. When the throughput values are sufficiently similar and the delays are all in the normal range, the smart home secure devices are functioning properly. In contrast, when the heartbeat packets do not match the regular pattern (that is, a large difference exists between the calculated frequency and the historical frequency) or the frame delay failed to meet the requirements, then the smart home secure device is experiencing a fault, and further detection is needed.

The delay distribution for the probe data packets in this method is analyzed by a delay probability distribution graph formed by the improved Gaussian distribution algorithm. Therefore, the methods are divided in the following two ways: throughput-based fault detection and Gaussian distribution-based fault detection. These methods are used to construct the fault detection method based on TDD

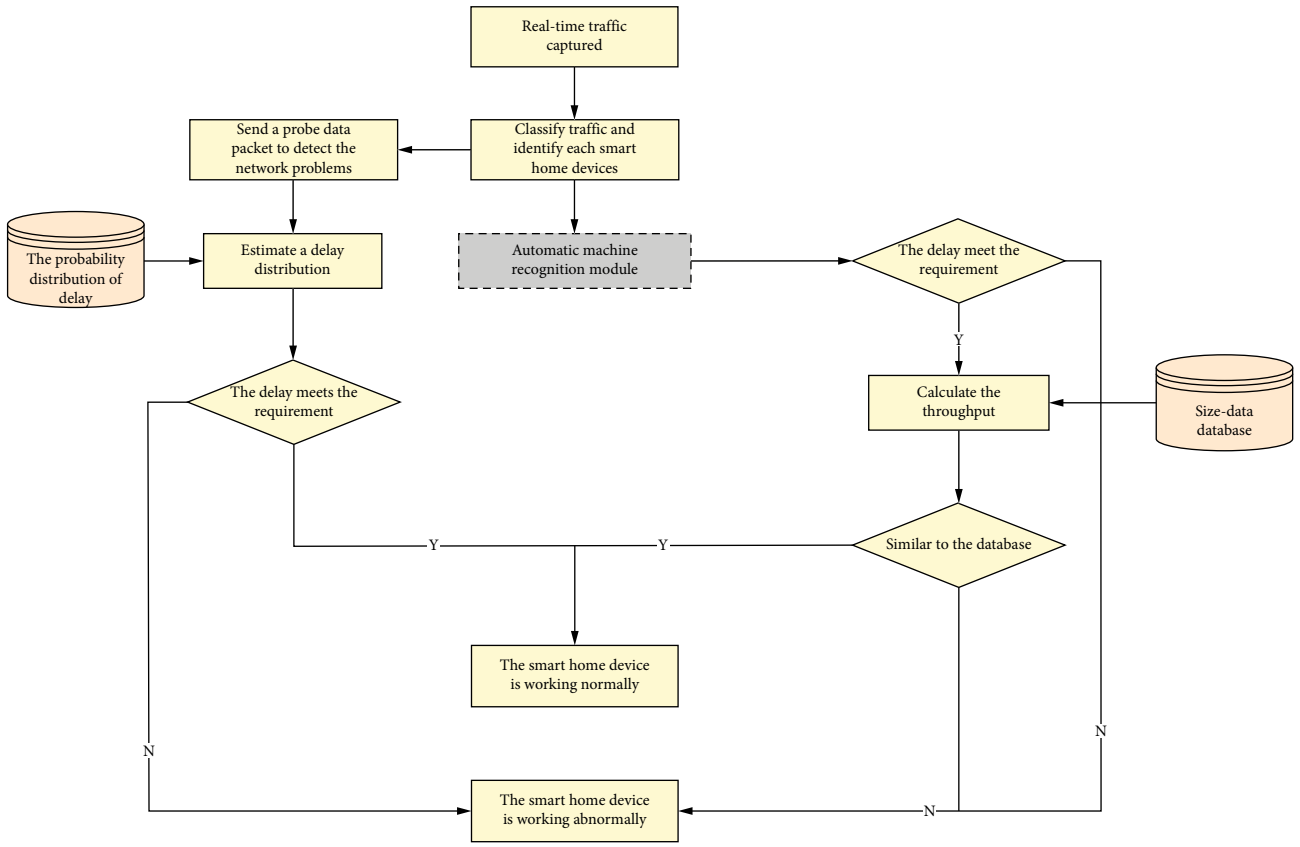

Figure 1: Flowchart of the proposed method.

(Throughput and Delay Distribution) proposed in this paper.

3.1. A Fault Detection Method Based on Throughput. A Wi-Fi communication-based smart home system is composed of various smart home devices. These devices communicate with their corresponding server via a Wi-Fi router. By observing the Wi-Fi traffic between devices and servers, we discovered that the traffic could mainly be divided into two types: command information and heartbeat communication data. The command information usually consists of the AWAY_FROM_HOME and AT_HOME commands; the former involves activating all connected detectors to provide security surveillance, and the latter involves deactivating detectors because the owner is home. The command information packet has an appropriate constant size. There are two different types of heartbeat communication data. One occurs during device startup procedures, while the other occurs during device communications. These data are periodic and have a small constant size. The startup heartbeat appears only when the device first connects with its server. The periodicity of the heartbeat data suggests that such messages are sent periodically; thus, their absence provides an approach for detecting faults. The heartbeat interval can easily be obtained from the captured network data labeled as heartbeat period in Table 1. Considering packet losses, a multiple of the heart period is used. For instance, if the captured heartbeat data time interval does not match a multiple of the heart period in Table 1, we can infer that the corresponding device is faulting. The throughput-based fault detection method depends on the condition of heartbeat data periodicity and requires abundant experimental data to form a comprehensive database. Table 1 shows the constructed database for a smart home device named KERUI used in our experiment.

To calculate the throughput T over a time interval, real-time traffic and data flow at a random unit time are required. After classifying the data flows in this time interval as either command information or heartbeat data, based on the size of the command information packets and heartbeat data packets in the database, an ideal throughput T' can be obtained for the selected time interval. Then, we use the α-(when $\alpha = 0.5$) order Renyi cross-entropy method from [30] to compare T with T', as follows:

$$R = I_{0.5}\left(TT'\right) = 2\log_2\sqrt{TT'}. \quad (1)$$

To compute the similarity, we first grab many data packets from/to the same smart home device. Each captured data stream can be acquired at any per unit time; $T_i(i = 0, 1, 2\cdots)$ and the corresponding $T_i(i = 0, 1, 2\cdots)$ are stored in the database. Then, the corresponding R_i is calculated by Formula (1), and the absolute value of R_i with the largest probability is chosen as the critical value η. Finally, η is selected as an appropriate threshold judgment criterion; that

Table 1: A database for a smart home device named KERUI.

Packet ID	Type	Protocol	Data size (byte)	Heart period (s)
1	Command request information	MQTT	231	—
2	Command response information	MQTT	305	—
3	Start of heartbeat request	MQTT	46	—
4	Start of heartbeat response	MQTT	67	—
5	Heartbeat request communication	TCP	2	49
6	Heartbeat response communication	TCP	2	49

is, when $|R| \geq \eta$, the smart home device is experiencing some fault. In addition, to further judge whether a device is faulty, we use a Gaussian distribution to compare the data packet delay.

3.2. A Fault Detection Method Based on a Gaussian Distribution. The improved fault detection method proposed in this paper is based on the Gaussian distribution function. By collecting abundant probe packets sent to Wi-Fi-based smart home devices, including normal network status and abnormal network status and then acquiring the delay for all the data packets and calculating their probabilities, we can obtain an initial delay probability distribution. Furthermore, by sampling all the delay experimental data using Gaussian fitting, we can obtain a Gaussian fitting function of the overall delay probability distribution and then store the probability distribution curve in the database. The Gaussian fitting we use is described as follows.

Suppose that d_i is the delay of a data packet, p_i is the probability distribution of d_i, $p_{\max}$ is the peak value of the Gaussian curve, $d_{\max}$ is the peak argument, and S is two times the variance. Then, we can describe a random set of data $(d_i, p_i)(i = 1, 2, 3 \cdots)$ using the Gaussian function as follows:

$$p_i = p_{\max} \times \exp\left[-\frac{(d_i - d_{\max})^2}{S}\right]. \tag{2}$$

Using the Gaussian function presented in (2), to solve the values of $p_{\max}$, $d_{\max}$, and S, we take the natural logarithm from both sides of Equation (2), and the result is as follows:

$$\ln p_i = \left(\ln p_{\max} - \frac{d_{\max}^2}{S}\right) + \frac{2d_i d_{\max}}{S} - \frac{d_i^2}{S}. \tag{3}$$

In Formula (3), by letting $\ln p_i = y_i$, $\ln p_{\max} - d_{\max}^2/S = x_0$, $2d_{\max}/S = x_1$, and $-1/S = x_2$, we can rewrite (3) into the following polynomial fitting function:

$$y_i = x_0 + x_1 b_i + x_2 d_i^2. \tag{4}$$

By applying all the experimental data (d_i, p_i) to (4), we obtain the following:

$$\begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & b_1 & b_1^2 \\ 1 & b_2 & b_2^2 \\ \cdots & \cdots & \cdots \\ 1 & b_n & b_n^2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}. \tag{5}$$

Formula (5) can be more simply notated as $Y = BX$. Based on the least squares principle, we can obtain a generalized least squares solution of a metric X consisting of the fitting constants x_0, x_1, and x_2 as follows:

$$X = \left(B^T B\right)^{-1} B^T Y. \tag{6}$$

By applying Formula (6) to Formula (3), we can obtain the values of $p_{\max}$, $d_{\max}$, and S and, consequently, the Gaussian function described in Formula (2). Then, we can draw the Gaussian distribution curve for this function and store it in the delay distribution database for use in judging whether a real-time probe data packet meets the delay requirement, which helps in diagnosing fault problems.

From a large number of original data packets, we can obtain the number of packets n that follow the normal delay of this device and the number m that have an abnormal delay. Thus, the fault rate p_l is $p_l = m/(m + n)$. Entering this p_l into the Gaussian function in Formula (2), we obtain the range of d_i, namely, the abnormal delay range for this device. Subsequently, if the delay of a data packet obtained in a time unit from probe traffic captured in real time is located in the abnormal delay range under the premise of normal packet loss and filtered retransmission, we can conclude that the smart home device is experiencing a fault. Otherwise, when $|R|$ in Formula (1) meets $|R| < \eta$ (namely, the throughput during a time unit is normal), we can conclude that the smart home device is functioning normally.

4. Experiments

4.1. Experience Design. Smart home security systems use sensor techniques, infrared techniques, fuzzy control techniques, and so on. Subdevices of smart home security devices usually include door sensors, window sensors, infrared probes, smoke detectors, and gas detectors. The deployment layout in a home environment is shown in Figure 2.

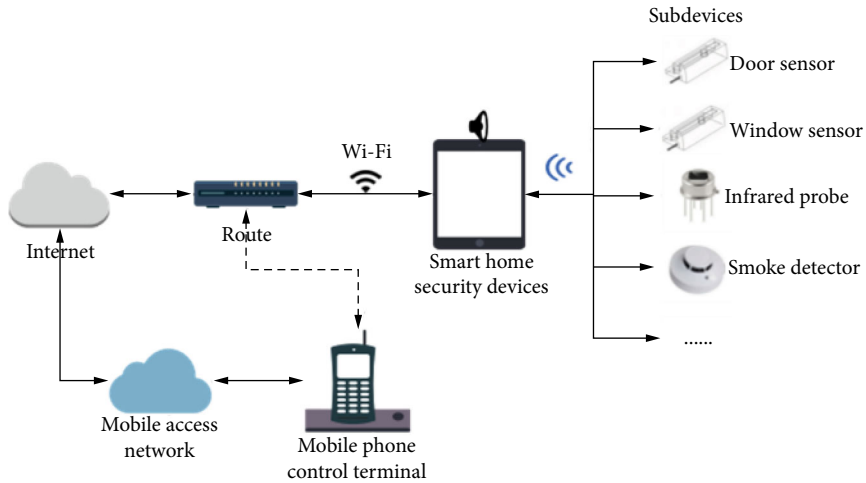

Figure 2: Home deployment layout.

BSSID	PWR	Beacons	#Data,	#/s	CH	MB	ENC	CIPHER	AUTH	ESSID
A8:9D:D2:14:78:BC	-77	1	0	0	1	54e.	WPA2	CCMP	PSK	CMCC_
8C:4C:DC:4F:91:FB	-45	28	0	0	2	54e	WPA2	CCMP	PSK	aubox
8C:4C:DC:4F:91:FA	-45	17	0	0	2	54e	WPA2	CCMP	PSK	aubox
E4:D3:32:E0:7B:82	-52	16	0	0	11	54e.	OPN			7B82
96:74:2A:B0:D4:6B	-55	36	0	0	3	54e.	WPA2	CCMP	PSK	CMCC-
42:E2:30:07:16:7F	-63	19	0	0	11	54e.	WPA2	CCMP	PSK	?.???
30:FC:68:DF:80:E9	-51	9	14	0	11	54e.	WPA2	CCMP	PSK	WILL
84:74:2A:B0:D4:6B	-59	30	52	0	3	54e.	WPA2	CCMP	PSK	CMCC-
5E:85:DE:BF:7E:11	-63	11	0	0	2	54e.	WPA2	CCMP	PSK	qiuqi
0A:D4:0C:E3:4E:64	-63	4	1	0	11	54e.	WPA2	CCMP	PSK	LieBa
88:5B:DD:4B:CD:15	-65	13	20	4	6	54e.	WPA2	CCMP	PSK	RDA-V
2E:85:56:9E:95:27	-65	7	0	0	11	54e.	WPA2	CCMP	PSK	RODMA
84:74:2A:B0:D5:E9	-63	11	220	51	13	54e.	WPA2	CCMP	PSK	CMCC-
76:74:2A:B0:D5:E9	-68	10	0	0	13	54e.	WPA	CCMP	PSK	<leng
96:74:2A:B0:D5:E9	-63	13	0	0	13	54e.	WPA2	CCMP	PSK	CMCC-
78:54:2E:58:E2:FE	-68	5	0	0	11	54e	WPA2	CCMP	PSK	Hemu
28:D9:8A:01:B4:58	-68	5	0	0	1	54e.	OPN			KK-PL
88:5B:DD:4B:CD:16	-62	18	0	0	6	54e.	OPN			<leng

Figure 3: Captured data packets in the monitoring mode.

BSSID	PWR	RXQ	Beacons	#Data,	#/s	CH	MB	ENC	CIPHER	AUTH	ESSID
E4:D3:32:E0:7B:82	-43	100	278	0	0	11	54e.	OPN			7B82
BSSID	STATION		PWR	Rate		Lost		Frames	Probe		
E4:D3:32:E0:7B:82	B4:43:0D:AA:50:61		-43	0 - 1e		0		1			
E4:D3:32:E0:7B:82	46:19:B6:0F:F9:1D		-43	0 - 1e		0		17			
E4:D3:32:E0:7B:82	B0:E2:35:26:EF:18		-76	0 - 1e		0		2			

Figure 4: Captured data packets in the monitoring mode.

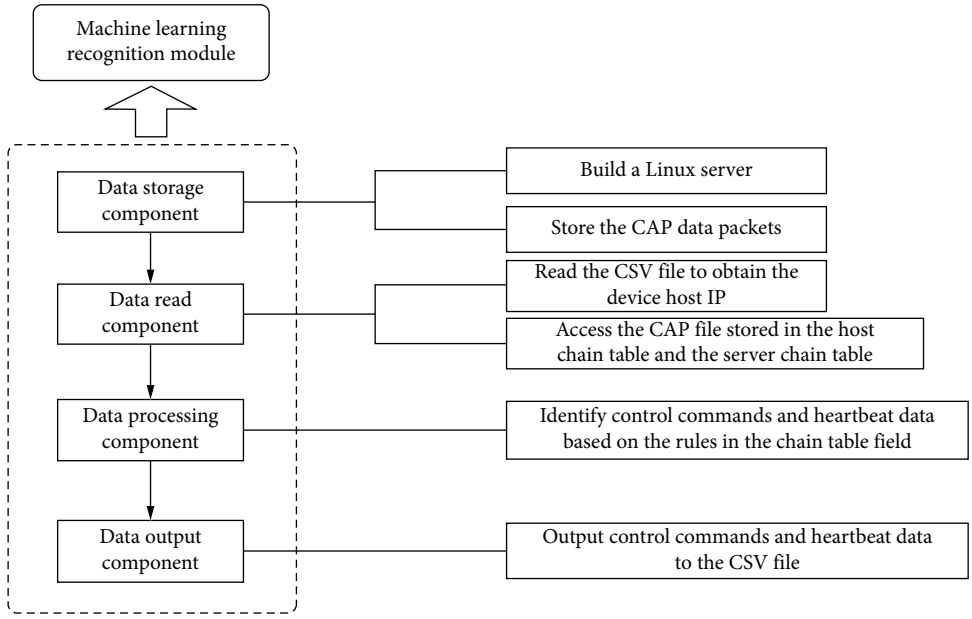

Figure 5: Automatic machine recognition module.

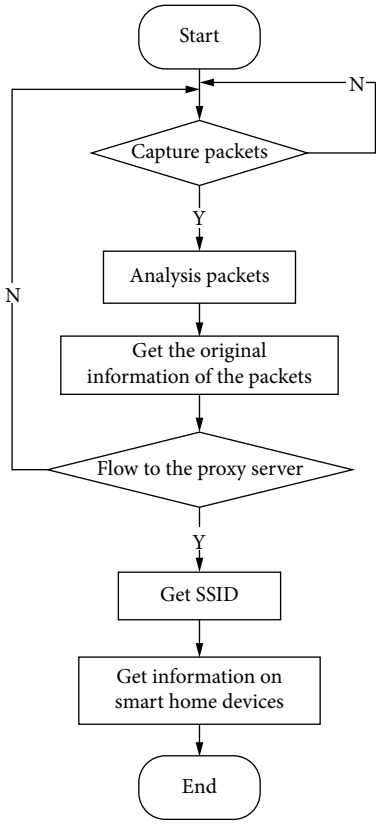

Figure 6: Flowchart of Wi-Fi traffic analysis.

Smart home security devices include both Wi-Fi receiving and sending modules. Controlling terminals such as smart cellphones can be used to control security devices via home Wi-Fi-connected or through cellular network devices and then further control the connected subdevices further. For our experiment, we chose the following home security devices: CHUANGO, ANJUBAO, and KERUI.

To test the fault detection method proposed in this paper, we conducted the following experiments. Step I involved capturing data packets between the security device and router, as shown in Figures 3 and 4. This step involved into 2 stages. In the first stage, we captured many data packets from security devices operating normally, while in the second stage, we captured data packets of devices with various failure statuses. Step II involved calculating statistics from the above data packets. In this step, we extracted the heartbeat period, the controlling commands, and other data based on a concrete machine recognition algorithm as shown in Figure 5. The extraction rules included the time interval of the heartbeat signal and the payload size. This step is important because it contributes to calculating T and T'. Step III involved sending a large number of probe data packets to the smart home security system. Similar to Step I, during this step, we captured two types of probe data packets that were in both normal state and various failure states. We marked the data packets in the failure states before mixing them with the normal data packets. Finally, the delay of data packets and a change curve were obtained. Step IV combined Steps I and II: we repeated Step I and Step II simultaneously and repeatedly. After obtaining the required data packets, we extracted data according to a time unit. Using this large amount of experimental data, we were able to validate the fault detection ability of our method.

4.2. Experience Results. The fault detection method for Wi-Fi-based smart home devices in this study was based on massive Wi-Fi traffic (including both normal and abnormal). We first analyzed a large amount of captured Wi-Fi traffic to obtain information about the connected smart home security devices. A flowchart of the Wi-Fi traffic analysis process is shown in Figure 6. Then, we further analyzed

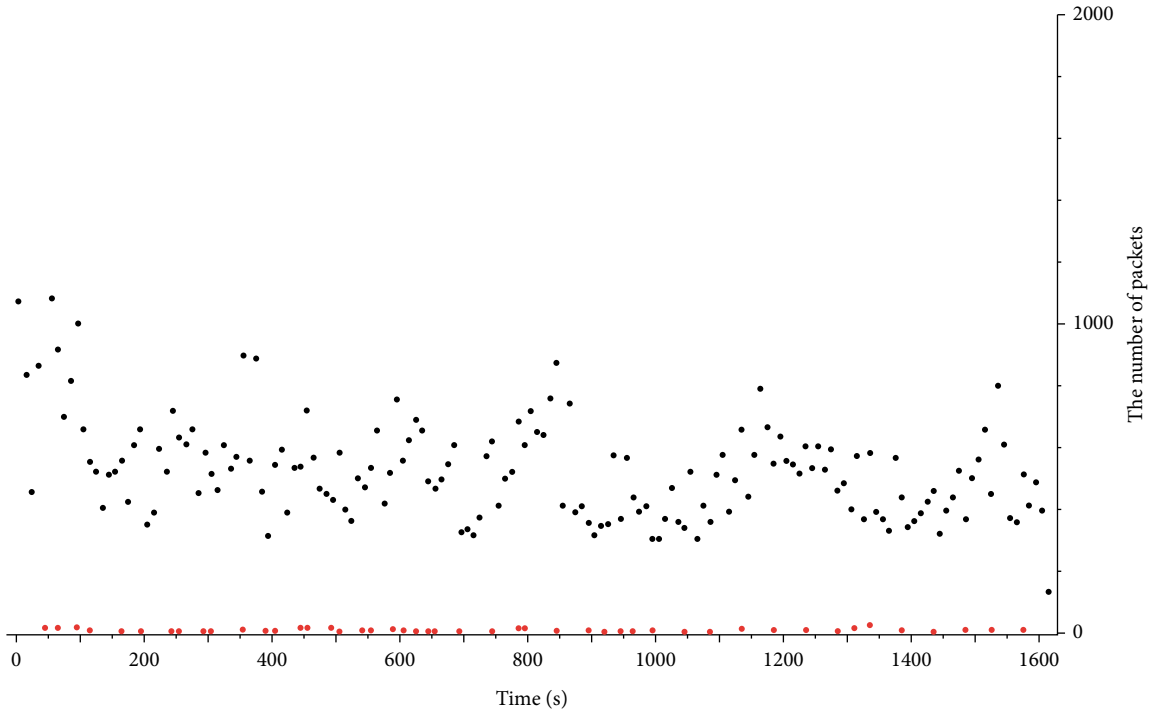

Figure 7: The distribution of the number of data packets within a given time period.

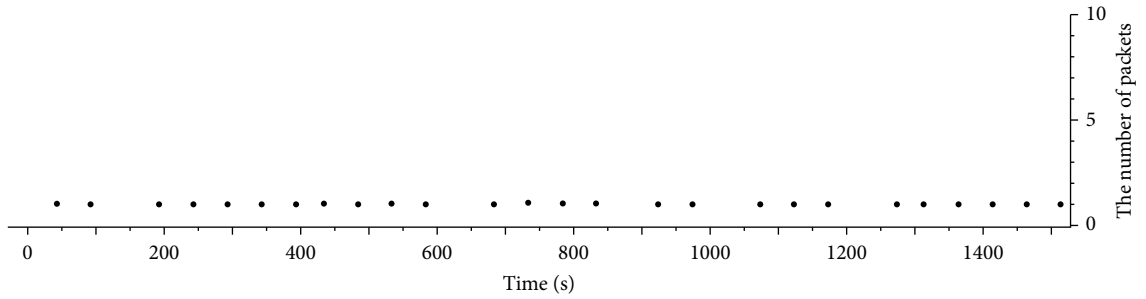

Figure 8: Heartbeat distribution during a selected time period.

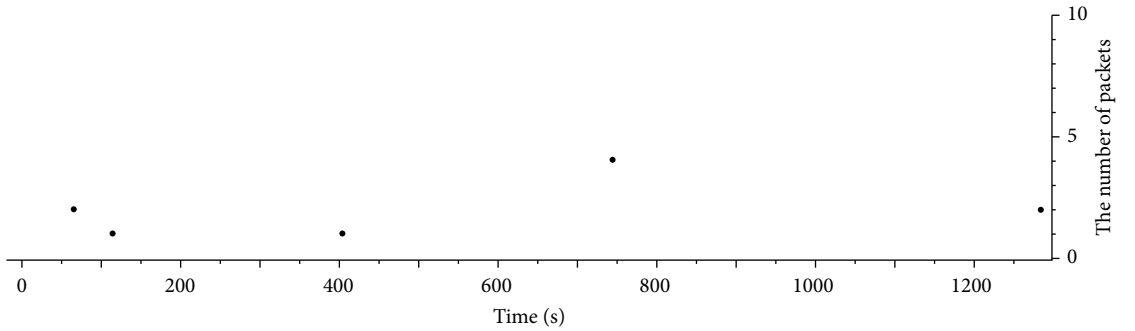

Figure 9: Control command distribution during a selected time period.

the captured data flows of the smart home security devices. Figure 7 shows the distribution of the numbers of KERUI data packets over a time period. The numbers in black represent all the packets over a specific time period, while the numbers in red represent only the packets from the KERUI smart home device.

After analyzing the traffic of this smart home device using the machine learning module, we filtered and marked

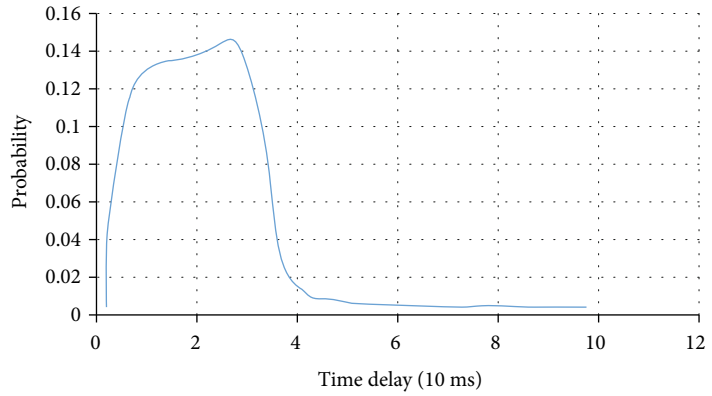

Figure 10: The probability distribution of delays calculated from the KERUI data.

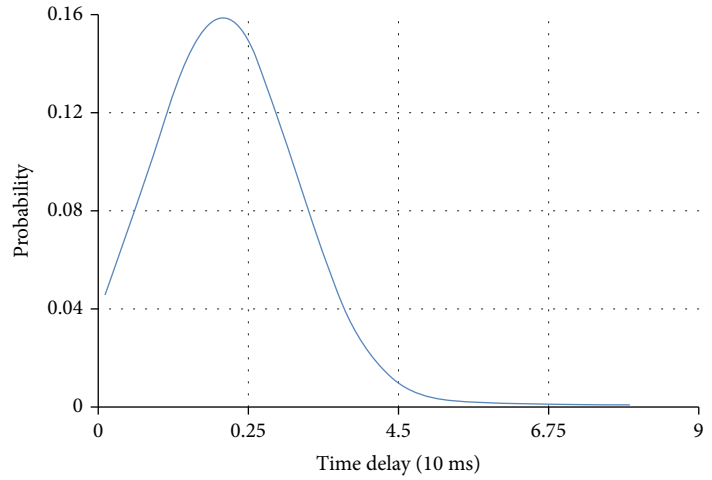

Figure 11: The Gaussian curve of the KERUI delays.

Table 2: Comparison of experimental results of the TDD method.

Smart home device	FNR	FPR	The TDD method				
			Actual fault ratio (%)	Detected fault ratio (%)	Recall ratio (%)	Accuracy ratio (%)	Detection time (s)
CHUANGO	0.0791	0.0609	92.75	93.44	97.84	93.58	1.40
ANJU BAO	0.0764	0.0597	93.00	93.86	98.07	93.46	1.49
KERUI	0.0865	0.0535	92.00	93.84	97.36	92.10	1.34

Table 3: Experimental results of TDD compared with another method.

Smart home device	The TDD method (%)				
	FNR	FPR	Recall	Accuracy (%)	Detection time (s)
CHUANGO	7.91	6.09	97.84	93.58	1.40
ANJUBO	7.64	5.97	98.07	93.46	1.49
KERUI	8.65	5.35	97.36	92.10	1.34
Smart home device	Detection method based on particle swarm optimization and Gaussian distribution (%)				
	FNR	FPR	Recall	Accuracy (%)	Detection time (s)
CHUANGO	1.42	43.39	96.98	87.91	1.40
ANJUBAO	2.31	49.53	98.37	87.63	1.48
KERUI	0.54	50.39	97.14	88.09	1.35

all the packets and then distinguished heartbeat packets from control command packets. Figure 8 shows the heartbeat distribution during the selected time period. From Figure 8, we can determine whether a heartbeat distribution matches the regular pattern. Figure 9 shows the distribution of control commands in this period of time. Control commands occur much less frequently and are irregular. Therefore, the throughput over a time unit is calculated and compared with stored values in a database to determine whether a fault exists.

Then, we extracted the delays in all the packets and calculated the frequencies of different delays. The results are shown in Figure 10.

According to the least squares method, we fitted the result into a Gaussian curve, as shown in Figure 11.

The probability distribution shown in Figure 11 is from the delay probability distribution database to determine the existence of faults from data packet delays. Using this database information, we can calculate the delay range when the fault occurs in a smart home device. Therefore, live captured data packets can be assessed using this method to determine faults in smart home devices.

Accuracy can be represented by the false-negative ratio (FNR) and false-positive ratio (FPR). True positives (TP) are the number of normal packets classified as positive; true negatives (TN) are the number of abnormal packets classified as negative; false positives (FP) are the number of abnormal packets classified as positive; and false negatives (FN) are the number of abnormal packets classified as negative.

$$\begin{aligned} \mathrm{FNR} &= \frac{\mathrm{FN}}{\mathrm{TP} + \mathrm{FN}}, \\ \mathrm{FPR} &= \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}. \end{aligned} \tag{7}$$

To verify the effectiveness of the method, we used existing smart home devices and captured 1000 groups of data sets, marked as normal or abnormal. Then, we calculated the FNR, FPR, recall ratio, and accuracy ratio as described above. The results are shown in Table 2. To verify the performance of the TDD method, we employ the fault diagnosis method based on particle swarm optimization and Gaussian distribution in [30] to compare with our method. The compared results of FPR, PNR, recall ratio, and accuracy ratio are shown in Table 3.

The results in Table 2 show that the fault detection method using TDD is able to detect faults that occur in smart home devices in real time. Furthermore, the fault probabilities obtained during the experiment are close to the actual fault rates in the original samples.

From the results in Table 3, the FPR of the method based on particle swarm optimization and Gaussian distribution is much higher than the FPR of our method, although the FNR of our method is slightly higher. To detect faults within smart home devices, we need to detect the fault data flow as accurately as possible, so we need the FPR value to be as low as possible. Meanwhile, the FN represents normal data flows detected as faults by mistake. Therefore, in order to detect faults comprehensively, FN cases must have little influence on the target of the fault detection methods. According to Table 3, the accuracy ratio of our method is approximately 6% higher than that of the method based on particle swarm optimization and Gaussian distribution. The results of the PSO and Gaussian method and those of our method are almost the same regarding detection time and recall ratio. Therefore, the performance of the PSO and Gaussian method is not as good as that of our method with regard to FPR and the accuracy ratio.

5. Conclusions and Future Work

In this paper, we developed a fault detection method for Wi-Fi-based smart home secure devices based on TDD. The TDD method captures real-time data traffic and then identifies smart home devices and their heartbeat and control command packets. If a device's heartbeat pattern matches a regular pattern, then we can detect faults from the throughput. In addition, probe data packets are marked and transmitted; then, faults can be detected by calculating the delay distribution of the data packets. Eventually, these results are combined to classify whether faults exist within the smart home devices. The proposed TDD method does not require a large number of characteristics, such as fault parameters or signals but can still detect real-time faults within smart home devices. The experimental results show that the method can detect smart home device faults in real time. In addition, the method has high recall and accuracy ratios.

In future work, we plan to add more relevant factors for detecting faults within smart home devices. In addition, the value of each factor should be verified from a multidimensional viewpoint to improve the accuracy ratio of the fault detection method and reduce the FNR and FPR values. In addition, as a method that detects faults using live network data flows, specific errors on the devices should be further analyzed according to the flow state. The corresponding warning information will also be addressed in future work because such messages can improve safety and comfort when using smart home devices.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

We gratefully acknowledge the invaluable contribution of Dr. Wu Yu and Dr. Yang Jie at School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, in this article. This paper is partly supported by the Science and Technology Research Program of Chongqing Municipal Education Commission (Grant No.

KJQN201900641 and Grant No. KJQN202000632) and the State Key Laboratory of Computer Architecture Research Fund (CARCH201902).

References

[1] H. Verma, M. Jain, K. Goel, A. Vikram, and G. Verma, "Smart home system based on Internet of Things," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 2073–2075, New Delhi, India, 2016.

[2] C. J. Kim, S.-J. Lee, and S.-H. Kang, "Evaluation of feeder monitoring parameters for incipient fault detection using Laplace trend statistic," *IEEE Transactions on Industry Applications*, vol. 40, no. 6, pp. 1718–1724, 2004.

[3] S. Sahoo, P. Rodriguez, and M. Sulowicz, "Evaluation of different monitoring parameters for synchronous machine fault diagnostics," *Electrical Engineering*, vol. 99, no. 2, pp. 551–560, 2017.

[4] M. Omana and J. H. Taylor, "Fault detection and isolation using the generalized parity vector technique in the absence of an a priori mathematical model," in *2007 IEEE International Conference on Control Applications*, Singapore, October 2007.

[5] F. Parra dos Anjos Lima, S. Silva Frutuoso de Souza, F. R. Chavarette, M. L. Martins Lopes, A. E. Turra, and V. Lopes Júnior, "Monitoring and fault identification in aeronautical structures using an ARTMAP-fuzzy-wavelet artificial neural network," *Advanced Materials Research*, vol. 1025-1026, pp. 1107–1112, 2014.

[6] W. Hu, L. Wen, L. Gao, and J. Ye, "Application of signal processing technology based on symbolic time series analysis to rotor broken fault detection," in *2010 International Symposium on Intelligent Signal Processing and Communication Systems*, pp. 1–4, Chengdu, December 2010.

[7] W. Qiao and D. Lu, "A survey on wind turbine condition monitoring and fault diagnosis—part II: signals and signal processing methods," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 10, pp. 6546–6557, 2015.

[8] P. Bilski and J. Wojciechowski, "Artificial intelligence methods in diagnostics of analog systems," *International Journal of Applied Mathematics & Computer Science*, vol. 24, no. 2, pp. 271–282, 2014.

[9] S. Nasiri, M. R. Khosravani, and K. Weinberg, "Fracture mechanics and mechanical fault detection by artificial intelligence methods: a review," *Engineering Failure Analysis*, vol. 81, pp. 270–293, 2017.

[10] Q. Yang, J. Li, S. Le Blond, and C. Wang, "Artificial neural network based fault detection and fault location in the DC microgrid," *Energy Procedia*, vol. 103, pp. 129–134, 2016.

[11] J. C. Augusto, J. Liu, P. McCullagh, H. Wang, and J. B. Yang, "Management of uncertainty and spatio-temporal aspects for monitoring and diagnosis in a smart home," *International Journal of Computational Intelligence Systems*, vol. 1, no. 4, pp. 361–378, 2008.

[12] J. Y. Son, J. H. Lee, J. Y. Kim, J. H. Park, and Y. H. Lee, "RAFD: resource-aware fault diagnosis system for home environment with smart devices," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 4, pp. 1185–1193, 2013.

[13] J. Ye, G. Stevenson, and S. Dobson, "Fault detection for binary sensors in smart home environments," in *2015 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 20–28, St. Louis, MO, USA, March 2015.

[14] C. H. Hsieh and J. Pei, "A fault diagnosis method for smart home services," in *2015 17th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pp. 452–455, Busan, South Korea, August 2015.

[15] S. Nandi, H. A. Toliyat, and X. Li, "Condition monitoring and fault diagnosis of electrical motors—a review," *IEEE Transactions on Energy Conversion*, vol. 20, no. 4, pp. 719–729, 2005.

[16] W. Jiang, B. Wei, C. Xie, and D. Zhou, "An evidential sensor fusion method in fault diagnosis," *Advances in Mechanical Engineering*, vol. 8, no. 3, 2016.

[17] T. W. Rauber, F. de Assis Boldt, and F. M. Varejão, "Heterogeneous feature models and feature selection applied to bearing fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 1, pp. 637–646, 2015.

[18] F. Zhou, J. H. Park, and Y. Liu, "Differential feature based hierarchical PCA fault detection method for dynamic fault," *Neurocomputing*, vol. 202, pp. 27–35, 2016.

[19] S. Yin, X. Zhu, and O. Kaynak, "Improved PLS focused on key-performance-indicator-related fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1651–1658, 2015.

[20] P. M. Van Every, M. Rodriguez, C. B. Jones, A. A. Mammoli, and M. Martínez-Ramón, "Advanced detection of HVAC faults using unsupervised SVM novelty detection and Gaussian process models," *Energy and Buildings*, vol. 149, pp. 216–224, 2017.

[21] X. Yang, P. Zhao, X. Zhang, J. Lin, and W. Yu, "A Gaussian-mixture model based detection scheme against data integrity attacks in the smart grid," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 147–161, 2016.

[22] C. Combastel, "An Extended Zonotopic and Gaussian Kalman Filter (EZGKF) merging set- membership and stochastic paradigms: toward non-linear filtering and fault detection," *Annual Reviews in Control*, vol. 42, pp. 232–243, 2016.

[23] T. A. Nakamura, R. M. Palhares, W. M. Caminhas et al., "Adaptive fault detection and diagnosis using parsimonious Gaussian mixture models trained with distributed computing techniques," *Journal of the Franklin Institute*, vol. 354, no. 6, pp. 2543–2572, 2016.

[24] Y. Bazi, L. Bruzzone, and F. Melgani, "Image thresholding based on the EM algorithm and the generalized Gaussian distribution," *Pattern Recognition*, vol. 40, no. 2, pp. 619–634, 2007.

[25] S. Deshmukh, S. Samouhos, L. Glicksman, and L. Norford, "Fault detection in commercial building VAV AHU: a case study of an academic building," *Energy and Buildings*, vol. 201, pp. 163–173, 2017.

[26] J. B. Ali, N. Fnaiech, L. Saidi, B. Chebel-Morello, and F. Fnaiech, "Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals," *Applied Acoustics*, vol. 89, no. 3, pp. 16–27, 2015.

[27] Q. Yi, L. Zhan-Ming, and L. Er-Chao, "Fault detection and diagnosis for non-Gaussian stochastic distribution systems with time delays via RBF neural networks," *ISA Transactions*, vol. 51, no. 6, pp. 786–791, 2012.

[28] K. Cheng, X. Liu, R. Zhang, and F. Lin, "A research on identification method for WiFi-based home automation device suites," in *2017 IEEE 2nd Advanced Information Technology,*

Electronic and Automation Control Conference (IAEAC), pp. 891–896, Chongqing, China, March 2017.

[29] T. Qin, L. Wang, Z. Liu, and X. Guan, "Robust application identification methods for P2P and VoIP traffic classification in backbone networks," *Knowledge-Based Systems*, vol. 82, pp. 152–162, 2015.

[30] C. Yu, R. Li, Q. He et al., "Fault diagnosis of nodes in WSN based on particle swarm optimization and Gaussian distribution," *Journal of Vibration Measurement & Diagnosis*, vol. 33, no. 1, pp. 149–152, 2013.