Hindawi
Security and Communication Networks
Volume 2021, Article ID 3314595, 12 pages
https://doi.org/10.1155/2021/3314595

Research Article

Spoofing Attack Detection Using Machine Learning in Cross-Technology Communication

Quan Sun,[1] **Xinyu Miao**,[1] **Zhihao Guan**,[1] **Jin Wang**,[1] **and Demin Gao**[1,2]

[1]*College of Information Science and Technology, Nanjing Forestry University, Nanjing, China*
[2]*Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55414, USA*

Correspondence should be addressed to Demin Gao; dmgao@njfu.edu.cn

Received 15 April 2021; Revised 13 July 2021; Accepted 4 August 2021; Published 28 August 2021

Academic Editor: Shehzad Ashraf Chaudhry

Cross-technology communication (CTC) technique can realize direct communication among heterogeneous wireless devices (e.g., WiFi, ZigBee, and Bluetooth in the 2.4 G ISM band) without gateway equipment for forwarding, which makes heterogeneous wireless communication more convenient and greatly reduces communication costs. However, compared with the traditional homogeneous network model, CTC technique also makes it easier to implement spoofing attacks in heterogeneous networks. WiFi devices with long communication distances and sufficient energy supply can directly launch spoofing attacks against ZigBee devices, which brings severe security concerns for heterogeneous wireless communications. In this paper, we focus on the CTC spoofing attack, especially spoofing attacks from WiFi to ZigBee and propose a machine learning-based method to detect spoofing attacks for heterogeneous wireless networks by using physical-layer information. First, we model the received signal strength (RSS) data of legitimate ZigBee devices to construct a one-class support vector machine (OSVM) classifier for detecting CTC spoofing attacks depending on the obtained training samples. Then, we simulated CTC spoofing attacks in a live testbed and evaluated the performance of our detection method. Results show that our approach is highly effective in spoofing detection. Even if the distance between the legitimate ZigBee device and WiFi attacker is near each other (i.e., less than 2 m) and does not require a large number of samples, the detection rate and precision of our method are both over 90%. Finally, we employ the OSVM classifier to obtain samples of spoofing attacks and then explore using SVM to further improve the performance of the classifier.

1. Introduction

In recent decades, more and more demand for wireless communications has caused issues correlated to communication security. With the rapid development of the Internet-of-Things technology, the unprecedented proliferation of wireless devices has brought great convenience to our lives. According to a recent report [1], the number of Internet-of-Things (IoT) devices is expected to reach 55 billion by 2025, which will cause the intense coexistence of wireless technologies [2]. Many of today's wireless technologies, such as WiFi, ZigBee, and Bluetooth, coexist and share the unlicensed spectrum (e.g., 2.4 G ISM band), which inevitably renders wireless devices to compete for channel and interfere with each other [3]. Besides, due to the openness of the wireless transmission medium, as a new type of attack, the CTC spoofing attack is incredibly easy to implement and can impair network performance significantly.

As an emerging research work, the CTC technique provides a promising direction for direct communication between heterogeneous wireless devices [4]. According to CTC, using different layer modulations, the existing CTC works can be divided into two categories: packet-level modulation [5] and physical-level modulation [6]. Specifically, compared with coarse-grained packet-level modulation, physical-level modulation can achieve high-speed throughput by directly simulating the heterogeneous signals in the physical layer [7]. Unfortunately, the security of CTC has not always been considered in the design, and thus, the application of CTC could potentially suffer from severe security concerns [8]. In traditional wireless networks,

spoofing attacks usually occur in homogeneous networks, that is, using ZigBee devices to attack another ZigBee device or WiFi to attack another WiFi device. However, CTC technology allows spoofing attacks to occur in heterogeneous networks, where WiFi devices can be used to directly attack ZigBee devices. For example, suppose that a malicious WiFi transmitter exists or has been compromised by an attacker. It could send the spoofed packets in the same frequency band to control the Bluetooth or ZigBee receiver via CTC.

The large-scale deployment of wireless devices has attracted a large number of malicious attacks, and in particular, the security issues of identity-based spoofing attacks have become extremely challenging. For example, in an IEEE 802.11 wireless LANs (WLANs), it is effortless for an adversary to change MAC addresses and then masquerade as an authorized wireless access point (AP) by simply issuing the ifconfig command [9]. Besides, spoofing attack is considered as the first step for several other types of attacks, such as traffic injection attacks, session hijacking, man-in-the-middle attacks, and various types of denial-of-service (DoS) attacks [9–11]. A variety of cryptographic authentication methods are commonly employed to prevent spoofing attacks for the homogeneous network. However, cryptographic authentication requires extrainfrastructural overhead, key distribution, management, and maintenance mechanisms [9, 11, 12]. Because of the limited power and resources of wireless sensors, these cryptographic schemes are not always desirable to be adopted [13]. In light of these circumstances, some advances [14] in noncryptographic mechanisms provide promising opportunities for securing CTCs in heterogeneous wireless networks.

In this paper, we focus on the spoofing attack based on CTC especially from WiFi to ZigBee and propose to utilize RSS, a physical property correlated to both the environmental conditions and distance between the sender and the receiver (not dependent on cryptography), as the basis for detecting spoofing. Specifically, WiFi devices with comprehensive deployment and more extended transmission range can easily launch CTC spoofing attacks when short-range ZigBee devices communicate with each other. The WiFi device masquerades as a ZigBee device to send spoofed data packets to other ZigBee devices in the same frequency band, but ZigBee devices cannot distinguish whether the data comes from the WiFi device or other ZigBee devices.

Spoofing attacks in cross-technology communication are more difficult to monitor than traditional homogeneous networks. Compared with traditional homogeneous networks, spoofing attacks in cross-technology communications are more difficult to monitor mainly because (1) WiFi devices have a longer transmission distance, which allows a WiFi signal to cover a wider range and can spoof more ZigBee devices, and (2) WiFi devices are usually powered by AC power, which makes the energy supply more sufficient and can continuously broadcast spoofing signals to other ZigBee devices. To counteract the aforementioned spoofing attacks over CTC links, we propose to detect spoofing attacks by utilizing machine learning algorithms based on RSS spatial correlation. Furthermore, our method does not require additional overhead, and wireless devices and sensors do not need to be modified.

The contributions are summarized as follows:

(1) We study CTC spoofing attack from WiFi device to ZigBee device using machine learning methods grounded on RSS physical property.

(2) We proposed two classifiers based on OSVM and SVM models. In the first one, we model the RSS data of legitimate ZigBee devices to construct an OSVM classifier for detecting CTC spoofing attacks depending on the obtained training samples. In the second one, we used SVM to further improve the performance of classifier based on the classification results of OSVM when large-scale spoofing attacks break out in the network.

(3) We simulated CTC spoofing attacks in a live testbed and evaluated the performance of our detection method. Results show that our approach is highly effective in spoofing detection. Even if the distance between the legitimate ZigBee device and WiFi attacker is near each other (i.e., less than 2 m) and does not require a large number of samples, the detection rate and precision of our method are both over 90%.

The remainder of this paper is organized as follows. Section 2 reviews the related work, and Section 3 introduces the background knowledge of preliminary work. Section 4 presents system design in detail. Sections 5 contains experimental results, and Section 6 concludes the paper.

2. Related Work

The traditional security approach to prevent spoofing attacks is to use cryptographic-based authentication [15–18]. Wu et al. [15] proposed a framework based on secure and efficient key management (SEKM). The work in [16] introduced a key management mechanism based on periodic key refresh and host revocation to avoid the leak of authentication keys. Bohge and Trappe [17] proposed an authentication framework for hierarchical, ad hoc sensor networks. In addition, in [18], the authors implemented the binding approaches of cryptographically generated addresses (CGA) to defend against spoofing attacks. However, because of the limited power and resources of wireless devices and sensor nodes, it is not always desirable to deploy these cryptographic schemes.

Some advances based on physical properties associated with wireless transmission provide promising opportunities for detecting spoofing attacks. Faria and Cheriton [19] proposed to detect identity-based attacks in wireless networks using signalprints. Signalprint was defined as the vector of median RSS for a MAC address in multiple air monitors. The work of [10] observed that, as a result of antenna diversity, the RSS readings tend to follow a mixture of multiple Gaussian distributions. They further proposed to build legitimate RSS profiles based on the Gaussian mixture model (GMM) clustering algorithm. The research in [13] proposed a method based on the K-means clustering

algorithm to detect and localize MAC address spoofing in both 802.11 WLANs and 802.15.4 ZigBee networks. The strategy proposed in [9] utilized the K-medoids algorithm to detect spoofing attacks and then determined the number of attackers and localized multiple adversaries. This algorithm is superior to K-means clustering algorithm because it is robust against any noise and outliers that the data might contain.

In its early days, how to avoid, mitigate, or tolerate cross-technology interference has drawn many researchers attention [20–24]. Recent advances in CTC have been expected to settle the issue of CTI and establish direct communication across technologies. According to CTC, using different layer modulations, the existing CTC works can be divided into two categories: packet-level modulation and physical-level modulation. In Esense [25], GSense [22], and FreeBee [26], RSS is used to measure WiFi signals to enable communication between WiFi and ZigBee devices. In comparison, with existing CTCs deploying packet-level modulation using the packet length [25], timing [26], energy level [27], and sequence patterns [28, 29], WEBee is the first physical-layer CTC design, which carefully fills the payload of a high-speed WiFi frame to directly emulate a low-speed ZigBee frame. In addition, similar to WEBee, TwinBee [30] and LongBee [31] enable CTC via physical signal emulation, where WiFi radio generates the desire of a ZigBee radio by manipulating the WiFi payload.

The strategy proposed in [32] adopted a collaborative mechanism to enable the spoofing attack detection for CTC in heterogeneous wireless networks by measuring the corresponding RSS on WiFi devices. The work in [33] implements a reactive jamming system, JamCloak, that can attack most existing CTC protocols. In addition, they proposed a practical detection and mitigation approach against reactive jamming attack over CTC links such as JamCloak. In [8], the authors observed a new attack named as CTC waveform emulation attack, where the WiFi attacker can capture the preintercepted ZigBee control message and hide it into the signal so as to manipulate the ZigBee device via transmitting the WiFi emulation signal. For detecting this attack, they utilized higher-order statistics at the ZigBee receiver to analyze the constellation.

As a relatively new technology, the security of CTC has not always been considered in design and currently relevant research work is scarce. Therefore, similar to traditional homogeneous wireless spoofing attacks, CTC spoofing attacks can also be easily implemented and cause significant damage to network performance. In this paper, we focus on the problem of CTC spoofing attack detection. We propose to use the spatial correlation of RSS inherited from wireless nodes and combine two machine learning methods for detecting CTC spoofing attacks.

3. Preliminary

3.1. Cross-Technology Communication. Compared with packet-level modulation, physical-level modulation is more fine-grained and thus achieve high-speed throughput by directly simulating the heterogeneous signals in the physical layer. As a pioneering research, WEBee [6] employs a WiFi signal to emulate another ZigBee signal without changing hardware or firmware. As shown in Figure 1, WEBee meticulously fills the payload of a transmitted WiFi frame in order that the RF waveform of the payload resembles that of ZigBee signals. When the ZigBee devices receive such a WiFi frame, it will ignore the WiFi header, the preamble, and trailer as noise, while the payload will successfully pass the ZigBee preamble detection, and then, the ZigBee receiver will demodulate the emulated ZigBee frame.

3.2. Theoretical Analysis of the Spatial Correlation of RSS. The received signal strength (RSS) involved in our research is closely related to its physical space position and is easily available in existing wireless networks. Although affected by random noise, environmental deviation, and multipath effects, the RSS values measured at the same physical location are similar, and the RSS values measured at different physical spatial locations are distinct. Therefore, the RSS values present strong physical spatial correlation characteristics.

We define the RSS value vector as $s = \{s_1, s_2, , s_n\}$, where n represents the number of reference points, and the reference points determine their positions by acquiring the RSS values of the wireless nodes. Generally, the RSS value of the wireless node obtained at the ith reference point satisfies the following logarithmic distribution [34]:

$$s_i(d_j)[\mathrm{d}B_m] = P(d_0)[\mathrm{d}B_m] - 10\gamma \log\left(\frac{d_j}{d_0}\right) + X_i, \quad (1)$$

where $P(d_0)$ indicates the maximum power of the sensor to the reference range d_0, d_j means the distance from sensor j to the ith landmark, γ represents the path loss exponent, and X_i indicates the shadow fading which follows zero mean Gaussian distribution with δ standard deviation [34, 35]. And, for the sake of simple, we suppose the wireless devices have the same transmission power. The RSS distance from one device to another in the signal space of the ith landmark can be formulated as

$$\Delta s_i = 10\gamma \log\left(\frac{d_2}{d_1}\right) + \Delta X, \quad (2)$$

where ΔX follows zero mean Gaussian distribution with $\sqrt{2\delta}$ standard deviation. The squared value of RSS distance in N-dimensional signal space (i.e., n reference points) can be given as

$$\Delta D^2 = \sum_{i=1}^{n} \Delta s_i^2, \quad (3)$$

where Δs_i with $i = 1, 2 \ldots, n$ denotes the RSS distance at ith landmark, represented by equation (2).

4. System Design

4.1. Network Architecture. The network architecture is provided in Figure 2, consisting of wireless devices (i.e., ZigBee devices and WiFi devices), server and console. Suppose there is such a situation that regular

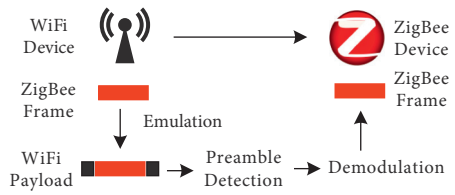

Figure 1: Physical-level CTC with signal emulation.

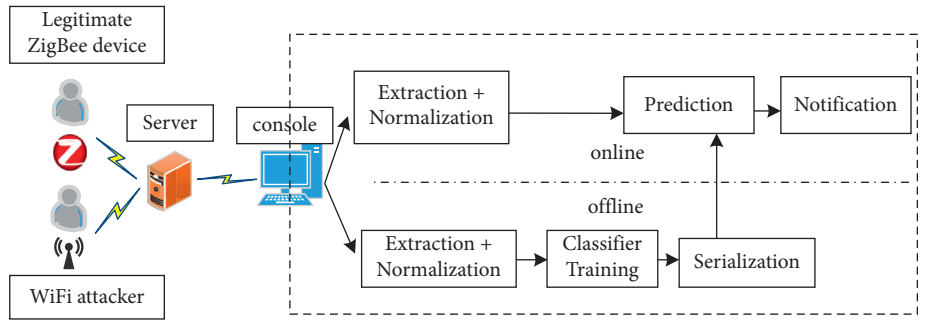

Figure 2: Network architecture and profiling.

communication between ZigBee devices is in progress. Since the WiFi device can directly communicate with the ZigBee device, the adversary can leverage the WiFi device to masquerade as a ZigBee device to launch a spoofing attack. At this time, monitoring sensors located in fixed locations can receive the WiFi attacker frames in real-time for spoofing detection. The server receives packets monitored by the monitoring sensors for global detection. The console receives data packets, utilizes timestamps or sequence numbers to normalize RSS samples, combines the data packets, and constructs the samples.

Figure 3 shows the method of training the OSVM model. Before using the OSVM model to classify the RSS samples in the real environment (i.e., there are spoofing attacks), a set of legitimate RSS samples need to be used to train the OSVM model. First, all incoming RSS data samples are preprocessed. The preprocessing of the sample includes the following. (1) Eliminate outliers: for filtering the outliers of the collected sample set, we use the 3σ criterion to eliminate noise data and leave them blank. (2) Filling the missing values: since there may be missing values in the collected sample set, we can use the KNN data filling algorithm to effectively fill in continuous or intermittent missing values and vacant noise data. The preprocessed data will then be divided into two parts: the training set and the test set. They are used to ensure a fully applicable OSVM model.

Figure 4 describes the process of spoofing attack detection. When an ideal model is obtained, the spoofing attack detection process will be used, as shown in Figure 4. Finally, the OSVM classifier is used to classify the incoming RSS data into spoofing data and legitimate data.

4.2. Attack Detection Using OSVM Analysis. As shown in Figure 5, we take an example to illustrate the spoofing attack process of the WiFi device against ZigBee devices. Because of

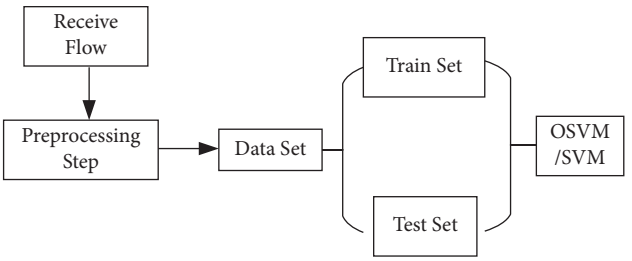

Figure 3: Train OSVM/SVM model.

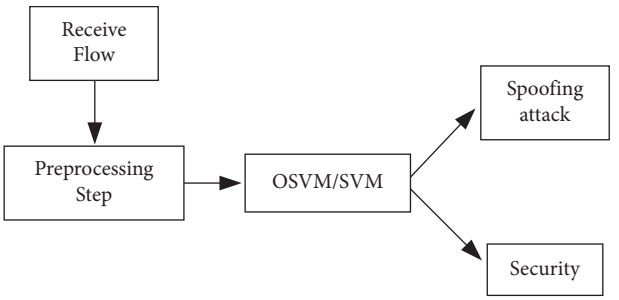

Figure 4: Flow of spoofing attack detection.

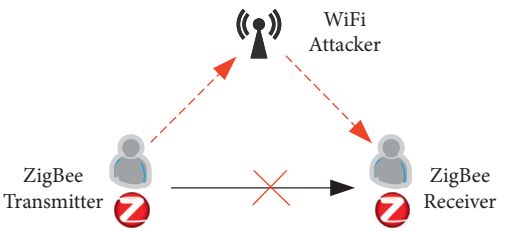

Figure 5: Flow of spoofing attack detection.

the broadcasting nature of wireless transmission medium, WiFi devices located in nearby locations can easily receive data frames from ZigBee transmitters. Therefore, the WiFi attacker launches a spoofing attack including two steps. First, the attacker is in the channel listening state, obtains the sending frame of the ZigBee device, then forges the identity information to masquerade as a legitimate ZigBee device, and finally successfully launches a malicious attack. We detect the CTC spoofing attack by measuring the WiFi data packet's RSS value on the monitoring sensors. We analyze the RSS values of WiFi devices collected by monitoring sensors scattered in different locations to monitor attackers. The RSS value of the data packet sent by wireless device i is expressed as the following vector:

$$x_i = \left[\mathrm{RSS}_i^1, \mathrm{RSS}_i^2, \mathrm{RSS}_i^k, \cdots\cdots\right]^T, \tag{4}$$

where RSS_i^k is the RSS value of node i from monitoring sensor k. Then, we use the signal strength vector of all deployed monitoring sensors as signal fingerprints of legitimate ZigBee devices and use these signal fingerprint vector $\mathbf{x}_i$ to construct the training set in real time. Finally, we train the OSVM classifier to detect whether the vector sample belongs to a legitimate ZigBee device.

(1) *Training Set*. We utilize RSS vector samples from legitimate ZigBee devices to fill the training set. Each sample includes the receiving time from the transmitter to the monitoring sensor, the transmitter's MAC address, and the RSS value. The training dataset $\mathbf{X}$ is given by

$$\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}, \tag{5}$$

where n is the number of samples of the signal fingerprint vector $\mathbf{x}$ in the training dataset $\mathbf{X}$.

(2) *One-Class SVM Classification*. The main goal of OSVM is to generate decision functions based on feature vectors in the training dataset. In this system, OSVM detects malicious devices by finding a suitable hyperplane in a nonlinear space. Therefore, the target is expressed as the following quadratic optimization problem:

$$\begin{aligned} &\min_{w,\xi_i,b} \quad \frac{1}{2}\|w\|^2 + \frac{1}{vn}\sum_{i=1}^{n}\xi_i - b, \\ &\text{s.t. } \mathbf{w}^T\varphi(\mathbf{x}_i) \geq b - \xi_i, \xi_i \geq 0, \quad \mathrm{i} = 1, 2, \ldots, n, \end{aligned} \tag{6}$$

where n is the number of training samples, $\varphi(\cdot)$ is the nonlinear mapping function for feature vectors in the training set, w is the weight vector for the model, ξ_i is the nonzero slack variable so that the model has a certain tolerance, and the regularization parameter $v \in (0, 1)$ is set to 0.01 to control the tolerance. In all training data, the vector sample $\mathbf{x}_i^*$ to be subjected to $(w^T\varphi(\mathbf{x}_i) - b = \xi_i)$ is the support vector, which is located at the edge of the decision function. Therefore, the classifier can be written as

$$f(\mathbf{x}) = \mathrm{sgn}(w\varphi(\mathbf{x}) - b). \tag{7}$$

The case of $f(\mathbf{x}) = 1$ indicates that the signal strength fingerprint vector $\mathbf{x}$ comes from a legitimate ZigBee device; while $f(\mathbf{x}) = -1$ indicates that it comes from a WiFi attacker.

To determine whether the test sample vector $\mathbf{x}_j$ falls within the hyperplane, the nonlinear kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ is used in the decision function, which is given by [36]

$$f_{\mathrm{OSVM}}(\mathbf{x}_j) = \mathrm{sgn}\left(\sum_{i=1}^{n}\mu_i \mathrm{K}(\mathbf{x}_i, \mathbf{x}_j) - b\right), \quad 0 < \mu_i < \frac{1}{vn}, \tag{8}$$

where μ_i is the Lagrange multiplier obtained by using the $\varphi(\cdot)$ function to maximize the margin. In most circumstances; we use three different kernel functions $K(\mathbf{x}_i, \mathbf{x}_j)$, namely, linear function, polynomial function, and radial basis Function (RBF) given by [37]

$$\begin{aligned} K(\mathbf{x}_i, \mathbf{x}_j) &= \mathbf{x}_i^T\mathbf{x}_j, \\ K(\mathbf{x}_i, \mathbf{x}_j) &= (\mathbf{x}_i^T\mathbf{x}_j + 1)^r, \\ K(\mathbf{x}_i, \mathbf{x}_j) &= e^{\left(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2\right)}. \end{aligned} \tag{9}$$

The linear kernel function is mainly used in the case of linearly separable. Compared with polynomial and RBF kernel function, it has fewer parameters, so the calculation speed is faster. For linearly separable data, its classification effect is very ideal. The polynomial kernel function can map a low-dimensional input space to a high-dimensional feature space, but it has many parameters. When the order of the polynomial is relatively high, the computational complexity will be too large to be calculated. The RBF function can map a sample to a higher-dimensional space. It has better performance regardless of whether in large-sample or small-sample training, and its parameters are less than the polynomial kernel function. By comparing the calculation results of these three kernel functions, in this study, we use the RBF function with the highest detection rate.

4.3. Attack Detection Using SVM Analysis. In this section, we explore the use of support vector machine algorithm to further improve the performance of the classifier based on the classification results of OSVM when training data are available in the offline phase. In particular, SVM is a set of kernel-based learning methods for data classification, including the training phase and the testing phase [38]. Every sample instance in the training set contains a class label and attribute (i.e., feature). For instance, for CTC spoofing attacks from WiFi to ZigBee, if there are no spoofing attacks, we can use the label value "+1" to mark the result; if there are spoofing attacks, we can use the label value "−1" to mark the result.

Training samples are able to be obtained by means of monitoring network activities regularly. The labeled training

set D with I feature vectors is given by equation (10), where y_i is the label of $\mathbf{x}_i$:

$$D = \{\mathbf{x}_i | \mathbf{x}_i \in R^n, y_i \in \{-1, +1\}\}. \tag{10}$$

Each feature vector $\mathbf{x}_i$ is an n-dimensional real vector of

$$x_i = \left[\mathrm{RSS}_i^1, \mathrm{RSS}_i^2, \mathrm{RSS}_i^k, \cdots\cdots\right]^T. \tag{11}$$

We aim to maximize margin hyperplane that divides feature vectors with $y_i = +1$ and $y_i = -1$. The hyperplane can be formulated as follows:

$$\mathbf{w}^T\mathbf{x} + b = 0, \tag{12}$$

where w represents the normal vector of the hyperplane, b indicates the bias variable, and $\mathbf{x}$ means the feature vector of the sample that lies on the hyperplane, as seen in Figure 6. We select the hyperplane that maximizes the margin between positive and negative samples. The following constraint needs to be satisfied:

$$y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geq 1, \ \forall i = 1, 2, \ldots, n. \tag{13}$$

As seen in Figure 7, if the training samples in the transformed space are linearly nonseparable, the optimization problem can be modified by introducing slack variables $\xi_i \geq 0$:

$$y_i\left(\mathbf{w}^T\mathbf{x_i} + b\right) \geq 1 - \xi_i, \ \forall i = 1, 2, \ldots, n. \tag{14}$$

The hyperplane w^T is computed by solving the following optimization problem in the primal form [39]:

$$\begin{aligned} &\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}\xi_i, \\ &\text{s.t. } y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geq 1 - \xi_i, \quad \forall \mathbf{x}_i \in D, \xi_i \geq 0. \end{aligned} \tag{15}$$

Its dual is

$$\begin{aligned} &\min \quad \frac{1}{2}\sum_{i=1,j=1}\alpha_i\alpha_j y_i y_j x_i^T x_j - \sum_{i=1}^{\infty}\alpha_i, \\ &\text{s.t. } \sum_{i=1}\alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \end{aligned} \tag{16}$$

where $C > 0$ means a compromise parameter between error and margin. We can use "kernel tricks" to solve nonlinear SVM problems. Suppose that the kernel function is represented by $K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j)$. Therefore, equation (16) can be formulated as

$$\begin{aligned} &\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \mathrm{K}\left(\mathbf{x}_i, \mathbf{x}_j\right) - \sum_{i=1}^{n}\alpha_i, \\ &\text{s.t. } \mathrm{C} \geq \alpha_i \geq 0, \quad \sum_{i=1}^{n}\alpha_i y_i = 0. \end{aligned} \tag{17}$$

We use the following kernel function for testing:

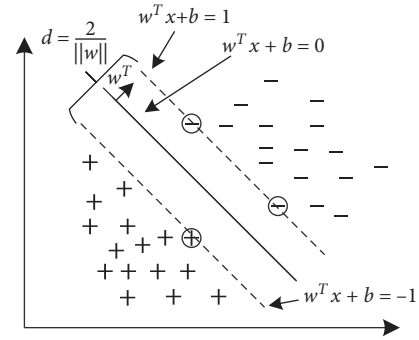

Figure 6: Linear separable dataset.

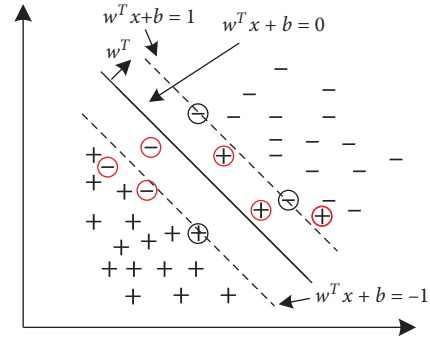

Figure 7: Linear nonseparable dataset.

$$\begin{aligned} K\left(\mathbf{x}_i, \mathbf{x}_j\right) &= \mathbf{x}_i^T\mathbf{x}_j, \\ K\left(\mathbf{x}_i, \mathbf{x}_j\right) &= \left(\mathbf{x}_i^T\mathbf{x}_j + 1\right)^r, \\ K\left(\mathbf{x}_i, \mathbf{x}_j\right) &= \tan h\left(\alpha\mathbf{x}_i^T\mathbf{x}_j + c\right), \\ K\left(\mathbf{x}_i, \mathbf{x}_j\right) &= e^{\left(-\gamma\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right)}. \end{aligned} \tag{18}$$

In addition, given a signal strength vector sample $\mathbf{x}$, the decision function is expressed as follows:

$$f_{\mathrm{SVM}}(\mathbf{x}) = \mathrm{sgn}\left(\sum_{i=1}^{n}\alpha_i y_i \mathrm{K}(\mathbf{x}_i, \mathbf{x}) + b\right), \tag{19}$$

where $f_{\mathrm{SVM}}(\mathbf{x})$ is used to detect the test sample $\mathbf{z}$. If $f_{\mathrm{SVM}}(\mathbf{x}) = 1$, it indicates that the test sample $\mathbf{z}$ belongs to a legitimate ZigBee device, that is, there is no spoofing attack. Otherwise, $f_{\mathrm{SVM}}(\mathbf{x}) = -1$, and the test sample $\mathbf{z}$ comes from an WiFi attacker, and there is a spoofing attack on the network.

In daily wireless communications, spoofing attacks are relatively rare compared with legitimate communications. This also means that we can easily obtain a large number of legitimate communication samples, while spoofing attack samples are more difficult to obtain and the number is scarce [40, 41]. However, using traditional classification algorithms to study wireless communication security requires roughly the same number of two-class samples [42, 43]. In this article, due to the diverse means of implementing spoofing attacks and the lack of training samples for spoofing attacks,

we use the OSVM algorithm to train legitimate data and construct a classifier to finally detect spoofing attacks and obtain spoofing attack samples. As the number of spoofing attack samples detected by the OSVM classifier increases, we explore using SVM algorithm to further improve the performance of the classifier.

5. Experimental Section

5.1. Experimental Setting. In this section, USRP N210 and ZigBee devices (i.e., MICAz nodes) are used to test the spoofing detection performance of the proposed model under heterogeneous networks. The scene setting is shown in Figure 8. Two MICAz nodes conduct normal data communication. The USRP N210 device is simulated as a WiFi signal transmitter and can directly transmit data to the ZigBee device [44, 45]. In this process, the USRP N210 is disguised as a legitimate ZigBee device to perform spoofing attacks on other ZigBee devices.

The proposed spoofing detection experiments were performed in an indoor environment. Figure 8 shows that the WiFi device located near (about 3 m) legitimate ZigBee device is launching a spoofing attack. The ZigBee transmitter and receiver are communicating. However, there is a malicious WiFi device near them. Since the WiFi device can receive ZigBee packets, the malicious WiFi device can masquerade as ZigBee transmitter and launch CTC spoofing attack on ZigBee receiver. This is exactly the CTC spoofing attack we want to detect. As shown in Figure 9, we used 20 testing locations marked with dots to cover an area of $8 \times 10\,m^2$. To evaluate our proposed method, we assumed two scenarios of spoofing attacks. The first scenario is when the WiFi attackers are in our room, we chose ten locations to be the location of the legitimate ZigBee device (e.g., location 1–10) and used the remaining locations (e.g., locations 11–20) as the location of the attacker. The specific operation is as follows: ten locations (marked with the red dot in Figure 9) are selected as the locations of the ZigBee device and moved between the ten locations, the USRP N210 is located at locations 11–20 (marked with the purple dot in Figure 9) and moves between the ten locations, and the USRP N210 conducts spoofing attacks on ZigBee devices at these different locations. To detect the attack, four sensors represented by triangles were placed to measure the RSS of the audible frames, and we collected 200 packet-level RSS samples at each location. For detecting spoofing attacks from locations 11–20, we used standard RSS samples at locations 1–10 to train the OSVM classifier. The other scenario is when WiFi devices with comprehensive deployment and more extended transmission range are outside the room, we chose to use data from all test locations (i.e., locations 1–20) to train the OSVM classifier.

5.2. Signal Strength Analysis. Figure 10 shows the data distribution of 10 locations from four monitoring sensors, respectively. We collected 200 RSS samples from each location, a total of 2000 samples. We found RSS oscillation for a stationary device, and the RSS values at the same position

Figure 8: Experimental setting.

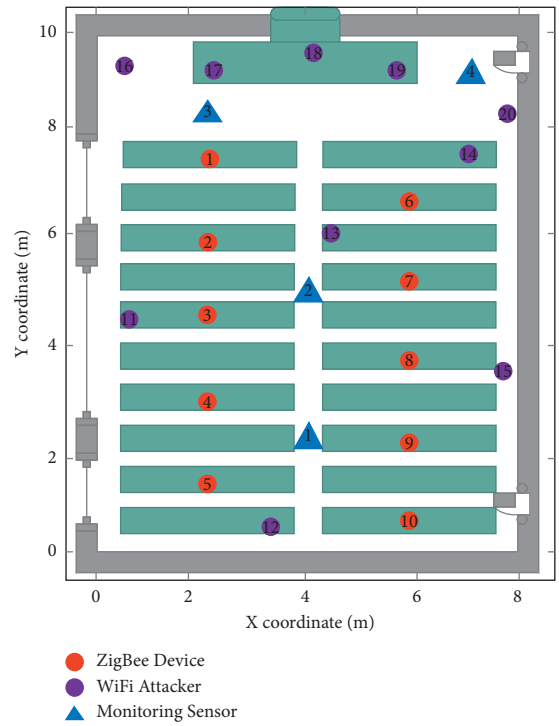

Figure 9: Network topology of the experiments.

are close to each other. There may be several factors for RSS oscillation, for example, multipath effect and obstacles that may cause signal oscillation, particularly, when the distance from the sender to the receiving device is large. However, so as to alleviate this influence, we can collect more samples at each location and then apply data cleaning techniques.

The corresponding probability histogram of RSS from 10 locations is given in Figure 11. Some researchers point out that the RSS samples of a given transmitter/sensor pair fit a Gaussian distribution [12, 13], while other researchers report that it is not rare to see non-Gaussian distributions of RSS samples, suggesting that those distributions are a mixture of multiple Gaussian distributions [10]. As shown in Figure 11, we also discovered this phenomenon, that is, a mixture of multiple Gaussian distributions. For example, in Figure 11(a), it can be seen that there are four Gaussian distributions. In Figure 11(b), two Gaussian distributions can be seen and so on. The four subplots show that the signal strength range of the four sampled locations is [−90, −30]

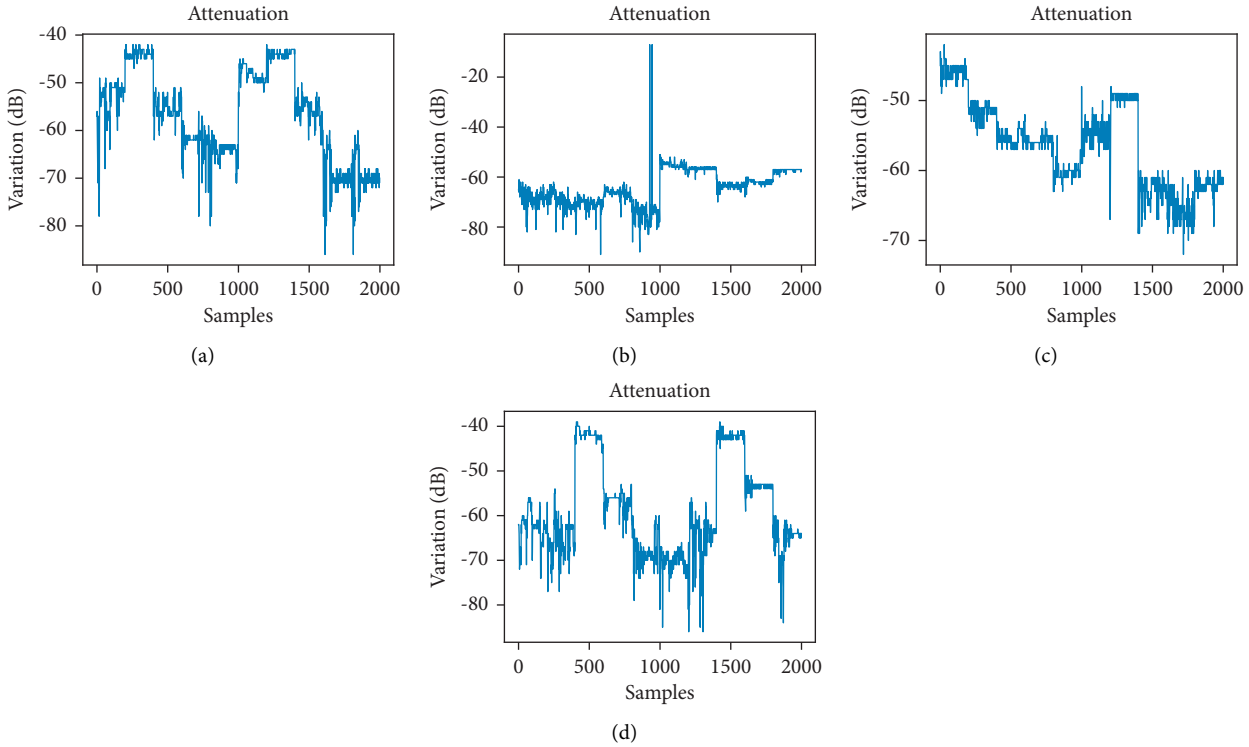

Figure 10: The RSS data distribution at 10 locations from four monitoring sensors. (a) 1st sensor. (b) 2nd sensor. (c) 3rd sensor. (d) 4th sensor.

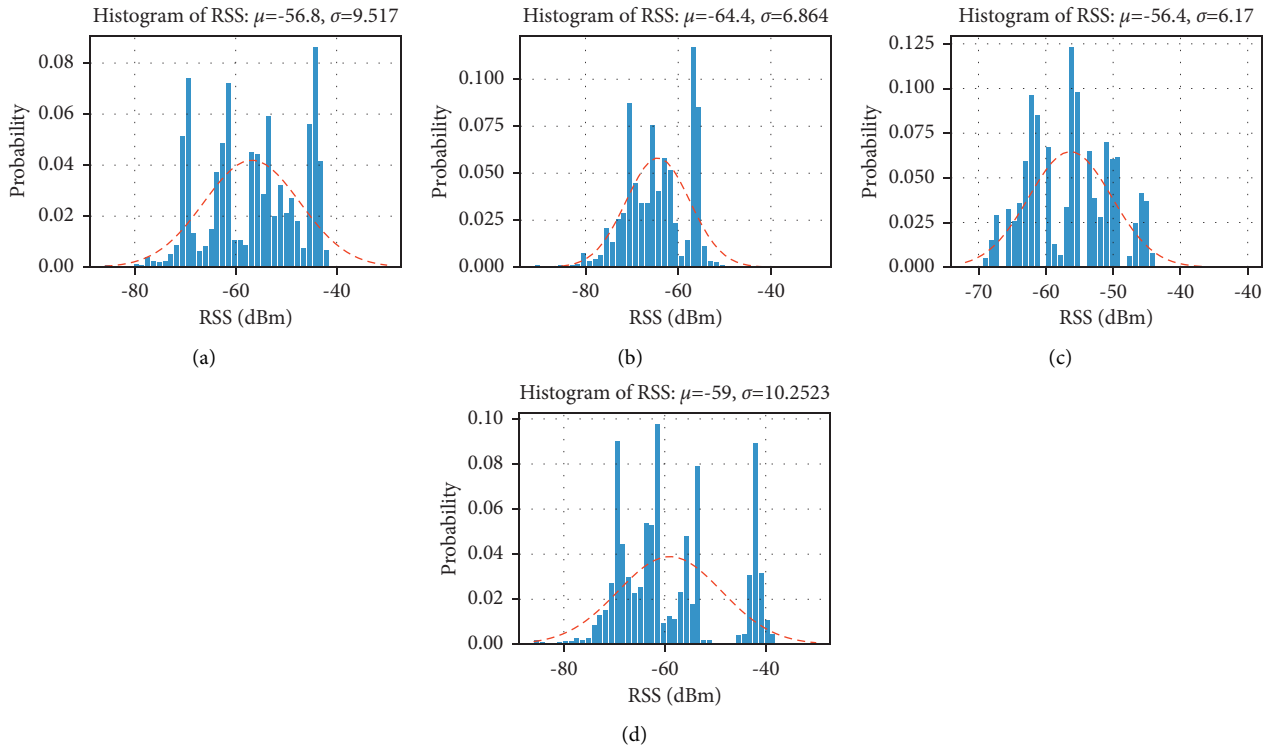

Figure 11: The probability histogram of RSS at 10 locations from four monitoring sensors. (a) 1st sensor. (b) 2nd sensor. (c) 3rd sensor. (d) 4th sensor.

dBm, and the mean μ and standard deviation σ of the signal at each location are calculated.

5.3. Performance Comparison. In order to evaluate the performance of the model proposed in this article in actual scenarios, this article chooses K-means, KNN, logistic regression (LR), and random forest (RF) to compare experiments with the method used in this article.

To evaluate the proposed schemes performance in the real scenarios, we simulated ten attack locations (i.e., locations 11–20) in Figure 9. Furthermore, we implemented the spoofing monitoring program, in which the distance between the legitimate ZigBee device and the WiFi attacker was less than 2 m, 2-3 m, and 3–5 m, respectively. Besides, we found that the performance of our method improves with increasing attack distance. When the attack distance is more excellent than 5 m, our spoofing detection method has extremely high detection performance; therefore, CTC spoofing attacks will be easily detected.

In this section, we introduce the comparison results of the accuracy for five spoofing attack detection methods. As shown in Table 1, five algorithms are used to compare the minimum, average, and maximum accuracy rates and standard deviations of accuracy rates under three different attack distances. When the distance between the legitimate ZigBee device and the WiFi attacker is less than 2 m, the average, maximum, and minimum accuracy rates of the K-Means method and the standard deviation of the accuracy rates are 78.95%, 80.72%, 63.66%, and 9.441%, respectively. The corresponding values of the KNN method are 81.34%, 90.26%, 63.53%, and 12.235%. Same as the above values, LR: 68.13%, 88.67%, 65.21%, and 11.127%, RF: 80.40%, 92.23%, 77.53%, and 7.215%, and OSVM: 92.17%, 94.45%, 85.51%, and 4.835%. When the distance is 2-3 m, K-Means: 83.38%, 89.05%, 62.37%, and 10.518%, KNN: 82.27%, 91.57%, 62.45%, and 13.56%, LR: 71.08%, 91.23%, 58.34%, and 15.233%, RF: 85.43%, 94.66%, 82.75%, and 5.755%, and OSVM: 95.38%, 97.89%, 91.60%, and 3.236%. When the distance is 3–5 m, K-Means: 88.75%, 90.31%, 68.50%, and 8.293%, KNN: 93.76%, 95.35%, 80.53%, and 7.411%, LR: 93.24%, 95.74%, 86.13%, and 4.712%, RF: 96.58%, 97.65%, 92.82%, and 2.154%, and OSVM: 97.76%, 98.77%, 96.31%, and 1.624%. It can be seen that, as the distance increases, the accuracy of these methods is improving.

When the distance is less than 2 m, the effect of the logistic regression algorithm is not ideal, the average accuracy rate is only 68.13%, the accuracy difference between K-Means, KNN, and random forest algorithm is small, and the method used in this article reaches 92.17%. Compared with the other four algorithms, it has a higher accuracy rate. When the distance is 2-3 m, the accuracy of the logistic regression algorithm is still the lowest. The average accuracy of KNN, K-Means, and random forest algorithms are 82.27%, 83.38%, and 85.43%, respectively. The OSVM algorithm used in this paper reaches 95.38%. As the distance increases to 3–5 m, the accuracy of the five algorithms improves. The minimum algorithm accuracy rate is 88.75%. Random forest and the method used in this paper exceed 95%, which has a higher accuracy rate for detecting spoofing attacks.

Table 1: Comparison of the accuracy of five algorithms in different test distances.

Test distance	Method	Mean	Std	Max	Min
Locations < 2 m apart	K-means	78.95	9.441	80.72	63.66
	KNN	81.34	12.235	90.26	63.53
	LR	68.13	11.127	88.67	65.21
	RF	80.40	7.215	92.23	77.53
	OSVM	92.17	4.835	94.45	85.51
Locations > 2 m and < 3 m apart	K-means	83.38	10.518	89.05	62.37
	KNN	82.27	13.56	91.57	62.45
	LR	71.08	15.233	91.23	58.34
	RF	85.43	5.755	94.66	82.75
	OSVM	95.38	3.236	97.89	91.60
Locations > 3 m and < 5 m apart	K-means	88.75	8.293	90.31	68.50
	KNN	93.76	7.411	95.35	80.53
	LR	93.24	4.712	95.74	86.13
	RF	96.58	2.154	97.65	92.82
	OSVM	97.76	1.624	98.77	96.31

In the three attack scenarios of the experiment, when the distance between the WiFi attacker and the ZigBee device is small (that is, less than 2 m), the accuracy of K-means, KNN, logistic regression, and random forest algorithms are all below 90%, of which logistic regression and the accuracy of the OSVM algorithm differs by 24%, and the accuracy of the other three algorithms differs from that of the OSVM algorithm by more than 10%. When the distance is 2-3 m and 3–5 m, the OSVM algorithm also performs higher than the other four algorithms. When the spoofing attack distance is small (that is, less than 2 m), the accuracy of the other four algorithms is significantly lower than that of the OSVM algorithm, which shows that, in the detection of small-distance spoofing attacks, the use of the OSVM algorithm in this paper has a greater advantage. When the distance is 2-3 m and 3–5 m, the accuracy of the OSVM algorithm is also the highest. Comparing the standard deviation of the accuracy of the five algorithms, it is found that the standard deviation of the accuracy of the OSVM algorithm is always smaller than the other four algorithms, which indicates that the detection performance of the OSVM is more stable than the other four algorithms. In summary, compared to K-means, KNN, logistic regression, and random forest algorithms, the OSVM algorithm has the best detection performance and the most stable model, so it is suitable for different test distance scenarios.

In order to evaluate the computational cost of these two methods, we conducted tests on the laptop equipped with 2.3 GHz CPU and 4 GB memory. Table 2 shows the comparison of the average test time, standard deviation, and minimum and maximum values of 3000 spoofing attack test samples using these five methods. The average, maximum, and minimum test time of the K-Means method and standard deviation of the test time are 0.12666 s, 0.38098 s, 0.025972 s, and 0.03786 s, respectively. Same as the above

Table 2: Comparison of the test time for five methods.

	K-means	KNN	LR	RF	OSVM
Mean	0.12666	0.22578	0.31306	2.01943	0.021941
Std	0.03786	0.00213	0.01135	0.04533	0.003695
Max	0.38098	0.23999	0.32029	2.4402	0.04687
Min	0.025972	0.21863	0.30893	1.97665	0.018965

Table 3: Comparison of metrics based on the OSVM method for different attack locations.

Metrics	Locations < 2 m apart Values	Locations > 2 m and < 3 m apart Values	Locations > 3 m and < 5 m apart Values	SVM Values
Detection rate	92.17	95.38	97.76	98.67
Precision	93.42	99.29	1.0	98.29
F-measure	90.56	94.17	96.49	98.62
AUC	91.3	94.2	96.4	99.76

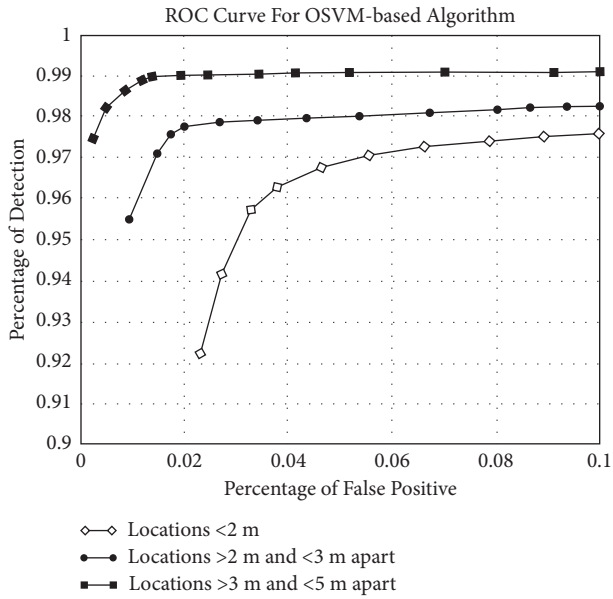

Figure 12: The ROC curve of different attack locations.

values, KNN: 0.22578 s, 0.23999 s, 0.21863 s, and 0.00213 s, LR: 0.31306 s, 0.32029 s, 0.30893 s, and 0.01135 s, RF: 2.01943 s, 2.4402 s, 1.97665 s, and 0.04533 s, and OSVM: 0.021941 s, 0.04687 s, 0.018965 s, and 0.003695 s.

The average test time of these five methods is sorted from small to large: OSVM, K-Means, KNN, LR, and RF. Among them, the OSVM method is the fastest, followed by K-means, with an average time of 0.021941 seconds and 0.12666 seconds, respectively. The random forest algorithm has the longest test time, with an average test time of 2.01943 seconds. We observe that the OSVM-based solution is about 100 ms faster than the second-ranked K-means detection method. This shows that the method in this paper is superior to the other four algorithms in terms of computational speed, which also means that, in the experiment, using the OSVM-based method, we can immediately detect the ongoing spoofing attack with very low latency.

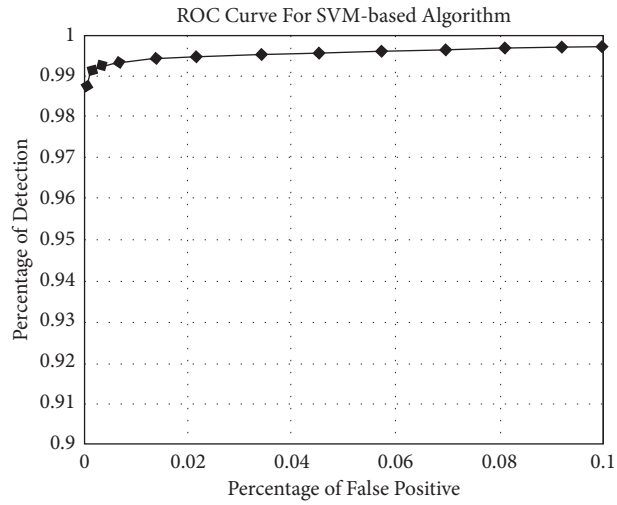

Figure 13: ROC curve for SVM-based algorithm.

5.4. Experimental Evaluation and Result. In this section, we introduce the detailed evaluation results of the OSVM algorithm proposed in this paper. Table 3 lists the WiFi attacker's detection rate, precision, F-measure, and AUC value at different distances from the legitimate ZigBee device. The corresponding receiver operating characteristic (ROC) curve is plotted in Figure 12. The results are encouraging. When the distance from the WiFi attacker to the legitimate ZigBee device is less than 2 meters, the detection rate for the false alarm rate of less than 3% is higher than 90%; when the distance between the WiFi attacker and the legitimate ZigBee device is 2-3 meters, although the false alarm rate reaches zero, the detection rate reaches 95.38%; when the distance between the WiFi attacker and the legitimate ZigBee device is 3–5 meters, the detection rate still exceeds 97%.

We utilize SVM to further improve the performance of classifier based on the classification results of OSVM when large-scale spoofing attacks break out in the network. Therefore, we utilize the OSVM classifier to detect the RSS

data of abnormal network traffic. As the number of spoofing attack samples increases, we use roughly the same number of two-class samples to train the SVM classifier. In order to reasonably evaluate the generalization error of the model, we use the grid search method. After 10-fold cross-validation, we find the best penalty coefficient C, the best kernel function is "rbf," and, finally, we get a 98.67% detection rate. Other metrics are shown in Table 3. The corresponding ROC curve is shown in Figure 13.

6. Conclusions

In this article, we proposed a machine learning-based method to detect spoofing attacks for heterogeneous wireless networks by using physical-layer information. To be more specific, WiFi devices with wide deployment and longer transmission range can easily launch CTC spoofing attacks when short-range ZigBee devices communicate with each other. Due to the lack of CTC spoofing attack samples, we propose to model OSVM classifier based on the RSS data of legitimate ZigBee devices. We simulated CTC spoofing attacks in a live testbed and evaluated the performance of our detection method. Results show that our approach is highly effective in spoofing detection. Even if the distance between the legitimate ZigBee device and WiFi attacker is near each other (i.e., less than 2 m) and does not require a large number of samples, the detection rate and precision of our method are both over 90%. We employ the OSVM classifier to obtain samples of spoofing attacks and, finally, explore using SVM to further improve the performance of the classifier.

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 there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the project fund of Future Network of Jiangsu (China), the project funded by China Postdoctoral Science Foundation (Grant nos. 2018T110505 and 2017M611828), and the Priority Academic Program Development (PAPD) of Jiangsu Higher Education Institutions.

References

[1] "Iot report how internet of things technology is now reaching mainstream companies and consumers," [Online]. Available: https://www.businessinsider.com/%20internet-of-things-report. Accessed on: Nov 2, 2019, 2019.

[2] X. Zheng, Z. Cao, J. Wang, Y. He, and Y. Liu, "Interference resilient duty cycling for sensor networks under co-existing environments," *IEEE Transactions on Communications*, vol. 65, no. 7, pp. 2971–2984, 2017.

[3] Y. Chen, M. Li, P. Chen, and S. Xia, "Survey of cross-technology communication for IoT heterogeneous devices," *IET Communications*, vol. 13, no. 12, pp. 1709–1720, 2019.

[4] X. Zheng, Y. He, and X. Guo, "Stripcomm: interference-resilient cross-technology communication in coexisting environments," in *Proceedings of the IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*, pp. 171–179, Honolulu, HI, USA, April 2018.

[5] D. Xia, X. Zheng, L. Liu, C. Wang, and H. Ma, "C-chirp: towards symmetric cross-technology communication over asymmetric channels," in *Proceedings of the 2020 17th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 1–9, Como, Italy, June 2020.

[6] Z. Li and T. He, "Webee: physical-layer cross-technology communication via emulation," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, pp. 2–14, Snowbird, UT, USA, October 2017.

[7] Y. He, J. Guo, and X. Zheng, "From surveillance to digital twin: challenges and recent advances of signal processing for industrial internet of things," *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 120–129, 2018.

[8] X. Zhang, P. Huang, L. Guo, and Y. Fang, "Hide and seek: waveform emulation attack and defense in cross-technology communication," in *Proceedings of the IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1117–1126, Dallas, Texas, USA, July 2019.

[9] J. Yang, Y. Chen, W. Trappe, and J. Cheng, "Detection and localization of multiple spoofing attackers in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, pp. 44–58, 2012.

[10] Y. Sheng, K. Tan, G. Chen, D. Kotz, and A. Campbell, "Detecting 802.11 MAC layer spoofing using received signal strength," in *Proceedings of the IEEE INFOCOM 2008-The 27th Conference on Computer Communications*, pp. 1768–1776, Phoenix, Arizona, April 2008.

[11] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Spoofing detection in IEEE 802.15.4 networks based on received signal strength," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2648–2660, 2013.

[12] Y. Chen, J. Yang, W. Trappe, and R. P. Martin, "Detecting and localizing identity-based attacks in wireless and sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2418–2434, 2010.

[13] Y. Chen, W. Trappe, and R. P. Martin, "Detecting and localizing wireless spoofing attacks," in *Proceedings of the 2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and, Networks*, pp. 193–202, San Diego, CA, USA, June 2007.

[14] K. Zeng, K. Govindan, and P. Mohapatra, "Non-cryptographic authentication and identification in wireless networks [security and privacy in emerging wireless networks," *IEEE Wireless Communications*, vol. 17, no. 5, pp. 56–62, 2010.

[15] B. Wu, J. Wu, E. B. Fernandez, and S. Magliveras, "Secure and efficient key management in mobile ad hoc networks," in *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium*, pp. 8–19, NW Washington, DC, USA, June 2005.

[16] A. Wool, "Lightweight key management for ieee 802.11 wireless lans with key refresh and host revocation," *Wireless Networks*, vol. 11, no. 6, pp. 677–686, 2005.

[17] M. Bohge and W. Trappe, "An authentication framework for hierarchical ad hoc sensor networks," in *Proceedings of the 2nd ACM workshop on Wireless security*, pp. 79–87, San Diego, CA, USA, September 2003.

[18] T. Aura, "Cryptographically generated addresses (CGA)," in *Proceedings of the International Conference on Information Security*, pp. 29–43, Springer, Berlin, Heidelberg, October 2003.

[19] D. B. Faria and D. R. Cheriton, "Detecting identity-based attacks in wireless networks using signalprints," in *Proceedings of the 5th ACM workshop on Wireless security*, pp. 43–52, New York, NY, USA, September 2006.

[20] F. Adib, S. Kumar, O. Aryan, S. Gollakota, and D. Katabi, "Interference alignment by motion," in *Proceedings of the 19th annual international Conference on Mobile computing & networking*, pp. 279–290, Miami, FL, USA, September 2013.

[21] L. Chen, R. Fan, K. Bian, M. Gerla, T. Wang, and X. Li, "On heterogeneous neighbor discovery in wireless sensor networks," in *Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM)*, pp. 693–701, Hong Kong, China, May 2015.

[22] X. Zhang and K. G. Shin, "Gap sense: lightweight coordination of heterogeneous wireless devices," in *Proceedings of the 2013 proceedings IEEE INFOCOM*, pp. 3094–3101, Turin, Italy, April 2013.

[23] X. Zhang and K. G. Shin, "Enabling coexistence of heterogeneous wireless systems: case for ZigBee and WiFi," in *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 1–11, Miami, FL, USA, September 2011.

[24] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi, "Successive interference cancellation: carving out MAC layer opportunities," *IEEE Transactions on Mobile Computing*, vol. 12, no. 2, pp. 346–357, 2012.

[25] K. Chebrolu and A. Dhekne, "Esense: communication through energy sensing," in *Proceedings of the 15th annual international Conference on Mobile computing and networking*, pp. 85–96, Turin, Italy, August 2009.

[26] S. M. Kim and T. He, "Freebee: cross-technology communication via free side-channel," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pp. 317–330, New York, NY, USA, September 2015.

[27] X. Guo, Y. He, and X. Zheng, "Wizig: cross-technology energy communication over a noisy channel," *IEEE/ACM Transactions on Networking*, vol. 8, no. 1, pp. 156–169, 2020.

[28] W. Jiang, Z. Yin, S. M. Kim, and T. He, "Transparent cross-technology communication over data traffic," in *Proceedings of the IEEE INFOCOM 2017-IEEE Conference on Computer Communications*, pp. 1–9, Atlanta, GA, USA, May 2017.

[29] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-morse: cross-technology communication with transparent morse coding," in *Proceedings of the IEEE INFOCOM 2017-IEEE Conference on Computer Communications*, pp. 1–9, Atlanta, GA, USA, May 2017.

[30] Y. Chen, Z. Li, and T. He, "TwinBee: reliable physical-layer cross-technology communication with symbol-level coding," in *Proceedings of the IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pp. 153–161, Honolulu, HI, USA, May 2018.

[31] Z. Li and T. He, "LongBee: enabling long-range cross-technology communication," in *Proceedings of the IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pp. 162–170, Honolulu, HI, USA, May. 2018.

[32] B. Lu, Z. Qin, M. Yang, X. Xia, R. Zhang, and L. Wang, "Spoofing attack detection using physical layer information in cross-technology communication," in *Proceedings of the 2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 1-2, Como, Italy, June 2018.

[33] G. Chen and W. Dong, "Jamcloak: reactive jamming attack over cross-technology communication links," in *Proceedings of the 2018 IEEE 26th International Conference on Network Protocols (ICNP)*, pp. 34–43, Cambridge, UK, USA, September 2018.

[34] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall PTR, New Jersey, Hoboken, 1996.

[35] T. K. Sarkar, Z. Zhong Ji, K. Kyungjung Kim, A. Medouri, and M. Salazar-Palma, "A survey of various propagation models for mobile communication," *IEEE Antennas and Propagation Magazine*, vol. 45, no. 3, pp. 51–82, 2003.

[36] G. I. Schuëller and H. A. Jensen, "Computational methods in optimization considering uncertainties - an overview," *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 1, pp. 2–13, 2008.

[37] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 1, pp. 1–39, 2012.

[38] T. Zhang, "An introduction to support vector machines and other kernel-based learning methods," *AI Magazine*, vol. 22, no. 2, pp. 103–120, 2001.

[39] O. Chapelle, "Training a support vector machine in the primal," *Neural Computation*, vol. 19, no. 5, pp. 1155–1178, 2007.

[40] B. Dga, W. C. Shuai, D. Yunhuai, J. Wenchao, L. Zhijun, and H. Tian, "Spoofing-jamming attack based on cross-technology communication for wireless networks," *Computer Communications*, vol. 177, pp. 86–95, 2021.

[41] D. Gao, Z. Li, Y. Liu, and H. Tian, "Neighbor discovery based on cross-technology communication for mobile applications," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 11179–11191, 2020.

[42] Y. H. Xu, X. Liu, W. Zhou, and G. Yu, "Generative adversarial LSTM networks learning for resource allocation in UAV-served M2M communications," *IEEE Wireless Communications Letters*, vol. 10, no. 7, pp. 1601–1605, 2021.

[43] Y. H. Xu, G. Yu, and Y. T. Yong, "Deep reinforcement learning-based resource scheduling strategy for reliability-oriented wireless body area networks," *IEEE Sensors Letters*, vol. 5, no. 1, pp. 1–4, 2020.

[44] S. Wang, Z. Yin, S. M. Kim, and T. He, "Achieving spectrum efficient communication under cross-technology interference," in *Proceedings of the 2017 26th International Conference on Computer Communications and Networks (ICCCN)*, pp. 1–8, Vancouver, Canada, August 2017.

[45] D. Gao, S. Zhang, F. Zhang, T. He, and J. Zhang, "RowBee: a routing protocol based on cross-technology communication for energy-harvesting wireless sensor networks," *IEEE Access*, vol. 7, no. 1, pp. 40663–40673, 2019.