Convolution Based Feature Extraction for Edge Computing Access Authentication

Feiyi Xie, Hong Wen, Jinsong Wu, Songlin Chen, Wenjing Hou, and Yixin Jiang

***Abstract*—In this article, a convolutional neural network (CNN) enhanced radio frequency fingerprinting (RFF) authentication scheme is presented for Internet of things (IoT). RFF is a non-cryptographic authentication technology, identifies devices through the waveforms of the RF transient signals by processing received RF signals on the edge server, which places no cost burden to low-end (low-cost) devices without implementing any encryption algorithm and meet the demands of the real-time access authentication in Internet of things. In the new scheme, the feasibility of extracting features based on one-dimensional (1D) signal convolution is discussed, referring to the method of extracting features from CNN, and combining with the characteristics of signal convolution. A convolution kernel for 1D signals is designed to extract the feature of signals in order to reduce training time and ensure classification accuracy. Therefore, it can improve the accuracy compared with these traditional algorithms, while saving the training time of updating parameters repeatedly as the neural network. The accuracy and training time of thealgorithm are verified in a real signal acquisition system. The results prove that the novel algorithm can effectively improve the classification accuracy in low signal-to-noise ratio (SNR), while keeps the training time in an acceptable range.**

***Index Terms*—Access authentication, convolution, edge computing, feature extraction, radio frequency fingerprinting (RFF).**

I. Introduction

With the increasing number of terminal devices connected to the Internet of Things (IoT), the data generated on the edge of the network grows accordingly, which puts forward higher requirements for real-time and security of data processing [1]–[3]. As cloud computing is not always efficient or safety for data processing especially for the data at the edge of the network, superior network systems are required to process such data in security without long propagation delay [4].

Manuscript received May 1, 2019; revised September 4, 2019; accepted September 8, 2019. Date of publication December 3, 2019; date of current version December 30, 2020. This work was supported in part by the National major R&D program under Grant 2018YFB0904900 and Grant 2018YFB0904905, in part by the Sichuan sci and tech service developement project under Grant 18KJFWSF0368, in part by the Sichuan sci and tech basic research condition platform project under Grant 2018TJPT0041, and in part by the Chile CONICYT under Grant Fondecyt Regular 181809. Recommended for acceptance by Dr. Q. Ye. *(Corresponding author: Hong Wen.)*

F. Xie, H. Wen, S. Chen, and W. Hou are with the University of Electronic Science and Technology of China, Chengdu 610051, China (e-mail: helloyuiki@std.uestc.edu.cn; sunlike@uestc.edu.cn; songlinch0061@163.com; uestc_hwj@126.com).

J. Wu is with the Department of Electrical Engineering, Universidad de Chile, Santiago 8320000, Chile (e-mail: wujs@ieee.org).

Y. Jiang is with the China Southern Grid, Guangzhou 100600, China (e-mail: jiangyx@csg.com).

Digital Object Identifier 10.1109/TNSE.2019.2957323

Edge computing, uses an open platform integrating network, computing, storage and application core capabilities to provide proximity services between the terminals and cloud, in most cases on the side near the terminal [5], [6]. Its applications are initiated on the edge to generate faster network service response and meet the basic needs in real-time business, application intelligence, security and privacy protection. Edge computing is between physical entities and industrial connections, or at the top of physical entities, while cloud computing can access the historical data of edge computing [7], [8].

Physical layer secure transmission, using artificial noise aided secrecy beamforming for secure transmission, enhances the security of IoT and edge computing with wireless broadcasting nature and the energy constraint of the physical objects [9] Another key strategy to enhance the security of edge computing is access authentication [10], [11]. Because most terminal devices have limited resource in computing, communication and storage to support such complex cryptography-based access authentication algorithms [12]. Non-cryptographic access authentication algorithm is more suitable for edge computing [13].

Radio frequency fingerprinting (RFF), as a non-cryptographic classification technology, classifies devices through the waveforms of their transient signals [14], [15]. RFF classifies devices by processing received signals on the edge server, there is no requirement for these low-end (low-cost) devices to implement any encryption algorithm but to transmit signals [16]. Therefore, RFF is suitable to apply for the access authentication in edge computing of IoT. [17]–[20]

The common decision methods in RFF include conventional threshold decision method and classification algorithm based on machine learning or deep learning. The threshold decision method generally uses similarity measure parameters such as correlation coefficient and Euclidean distance, which cannot reach a high enough identification accuracy, resulting in a high error rate (ER) proved by our previous work, which cannot ensure the security of edge computing [21]. Classification algorithms based on deep learning, especially convolutional neural network (CNN), have higher accuracy and are often used in large-scale image processing. Although the most current research still focuses on two-dimensional (2D) or three-dimensional (3D) image processing [22], [23]. Few studies applied CNN to one-dimensional (1D) signal processing [24]. This paper explores to apply CNN to 1D RFF identification. Thus, it cannot meet the real-time requirement in edge computing. Some studies in signal processing focus on signal convolution, but most of them are traditional

convolution algorithms, which are not related to classification situation [25]–[28].

However, the neural network need to update parameters repeatedly, which results in a long training time and is not suitable for real-time processing requirement of edge computing [29].

Inspired by the convolution based feature extraction steps in CNN, a novel algorithm of signal feature extraction by 1D convolution is proposed in this paper. According to the periodic characteristics of the baseband sinusoidal wave, several convolution kernel functions are designed. Characteristics in amplitude, frequency variation or pulse perturbation can be extracted through convoluted with the original signal. DWT can be used as a feature reduction method for signal afterwards, by which the sampling points of training set are greatly reduced while most feature information is retained. The output sample vectors can be directly used in the training of machine learning, such as support vector machine (SVM), k-nearest neighbors (kNN) and decision tree, instead of updating parameters repeatedly in neural networks. The key contributions of this work are summarized as follows:

1) A group of targeted 1D convolution kernels are first proposed in this paper, which can be used to extract the signal feature. It improves classification accuracy without increasing the training time of machine learning.
2) Fixed convolution kernels, trained and classified in simple machine learning such as SVM, instead of updated repeatedly in neural networks, can efficiently reduce training time of deep learning, such as CNN.
3) A complete experiment system is set up in this work, including signal acquisition, signal processing and classification. Experiments show that the method proposed in this paper can achieve a similar classification accuracy as CNN in both low and high signal-to-noise ratio (SNR).

The remainder of this paper is organized as follows: Related methods, CNN and DWT, are introduced in Section II. Section III presented the main algorithm, convolution based signal feature extraction, and detailed derivation of this algorithm is performed, including convolutional kernels and DWT based pooling. Experimental process and the related test system are reported in Section IV. The result of the simulation experiment is presented and discussed in Section V, and Section VI concludes the paper.

II. Related Methods

A. Convolution

Convolution is a mathematical method of integral transformation, which has been widely used in fields like mathematics, statistics, physics and signal processing.

The 1D convolution of continuous functions is defined as an integral process. The definition is

$$f(x) * g(x) = \int_{-\infty}^{\infty} f(\tau) g(x - \tau) d\tau \tag{1}$$

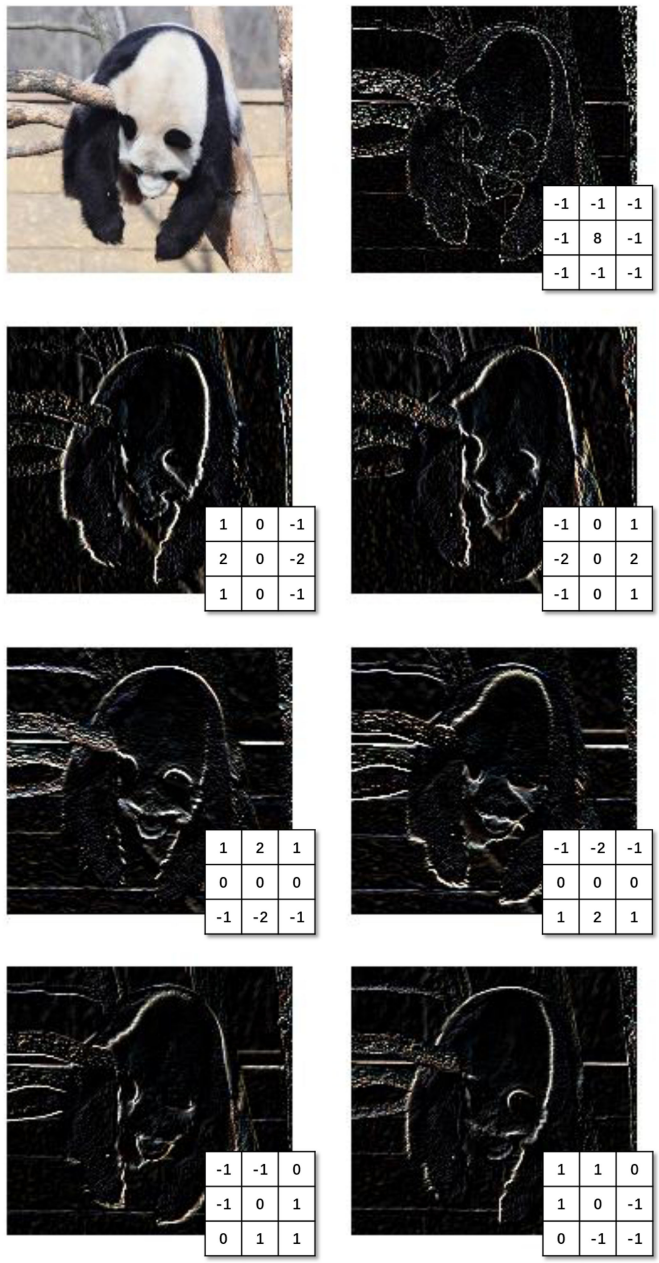


Fig. 1. Image features extracted by different convolution kernels

where * represents convolution. For discrete sequences, convolution is the sum of the multiplication of the two sequences in a certain range

$$f[n] * g[n] = \sum_{\tau=-\infty}^{\infty} f[\tau] g[n - \tau] \tag{2}$$

2D or 3D convolution for image processing is similar. A 2D or 3D matrix, which is called convolution kernel, is moved in a fixed step on the image to calculate the sum of the multiplication of the convolution kernel and the corresponding part of the image.

A series of presupposed convolution kernels can be used to extract features of the edge or in a certain direction. Fig. 1 shows

several commonly used presupposed convolution kernels and the image features they extract.

B. Convolutional Neural Network

Computer vision has many kinds of research areas including image classification, object detection and image style migration. Deep neural network can solve some small-scale image problems simply. However, when applied to large-scale images, the input scale becomes very large, making the parameters cannot be solved through the neural network.

CNN, which was first proposed for large-scale image processing, has a forward propagation and a backward propagation, while the forward propagation is divided into input layer, convolutional ($conv$) layer, activation (mostly ReLU) layer, pooling layer and full connection (FC) layer. The input layer and ReLU layer are similar to the traditional neural network. [22]

Convolution layer is the core component of convolution neural network with the purpose of using the convolution kernel to convolute with the target image to perform edge detection. The convolution kernel window slides on the image with a specified step, multiplied with the elements of the corresponding region on the image. The sum is a pixel value of the new image. Generally, different convolution kernels are set up to detect the edges of the image.

The pooling layer is used to compress the data and parameters, reducing over-fitting. For image processing, the most important role of the pooling layer is to compress the image, reducing the dimension of the feature on the premise, while keeping the image features invariant. The methods of pooling are divided into maximum pooling (represented by the maximum value of the pooling area) and average pooling (represented by the average value of the elements in the pooling area), where maximum pooling is more commonly used.

The FC layer connects the outputs of each neuron to a 1D vector for judgment at the end of CNN.

The convolution kernel is a kind of filter to extract the features of an image in a direction and get an activation map. Each pixel in the activation map is the weighted average of a small area of the input image. The weights are defined by a function called convolution kernel, which is convoluted with the input image at different positions. The convolution kernel is usually a square of $n * n$, or $n * n * l$ when there is more than one layer. The weights, or the parameters of the kernel, are fixed when convoluted with different areas of the input image.

In CNN, different from presupposed convolution kernel, the parameters of convolution kernels only need to be initialized, and will be updated when CNN propagates backward.

CNN for signals is similar to that for images, for the signal is a 1D sequence, the corresponding convolution kernel should be 1D. Most signals are periodic, such as sinusoidal function signal and square wave signal, which are not invariable even in the steady state. Similar 1D convolution kernels are not applicable in signal processing, which will be further discussed in Section III.

C. Discrete Wavelet Transform

In most cases of signal reception processing, the received signals are discrete. The discrete wavelet transform (DWT) provides sufficient information for both analysis and synthesis of the original signal, with a significant reduction in the computation time, compared with continuous wavelet transform (CWT). In discrete cases, filters with different cutoff frequencies are used to analyze signals at different scales of features. A series of high pass filters are used to analyze the high frequency part of the signals, and a series of low pass filters are used to analyze the low frequency part.

$$a_j = a_{j+1} \oplus d_{j+1} \tag{3}$$

where, a_j represent the undecomposed original signal, a_{j+1} and d_{j+1} respectively represent high frequency and low frequency signals.

Mallat algorithm is a fast decomposition algorithm of the DWT. Its formulas are as follows:

$$a_{j+1}[n] = h[n] * a_j[n] = \sum_k h[k] a_j[2n-k] \tag{4}$$

$$d_{j+1}[n] = g[n] * a_j[n] = \sum_k g[k] a_j[2n-k] \tag{5}$$

where, $h[n]$ and $g[n]$ represent the tap coefficient sequence of the low pass and high pass part of the half band filter [30], [31].

The low frequency component contains the characteristics of the signal, while high frequency component contains the details of the signal. The sample size of the processed discrete signal reduces by half whenever being done a DWT, i.e. $a_{j+1}[n]$ and $d_{j+1}[n]$ have half the sample points of $a_j[n]$.

III. Algorithms and Derivation

In image processing, convolution kernel functions can clearly display the color saltation points. The convolution result becomes zero where the color remains unchanged, while in the position where the color has a saltation, the convolution result is a non-zero value. In signal processing, unlike images, most signals are jittery, such as triangular wave or square wave. Steady state signals do not mean that the signal is completely unchanged, but that its amplitude, frequency and phase do not change. Thus, the simple convolution kernel like [-1, 0, 1] cannot be used to extract features from sine or cosine signal waveforms. A novel convolution kernel, applicable for feature extraction of triangular function waveform, is proposed, by which the unstable part of the signal is highlighted, while the steady-state part is maintained. Two kinds of convolution kernels are presented as follows, and their convolution result are analyzed.

A. Double Frequency Convolution Kernel

Considering the convolution characteristics of periodic signals, for signals as $f(t) = \sin \omega t$

$$f(t) * \sin 2\omega t = 0 \tag{6}$$

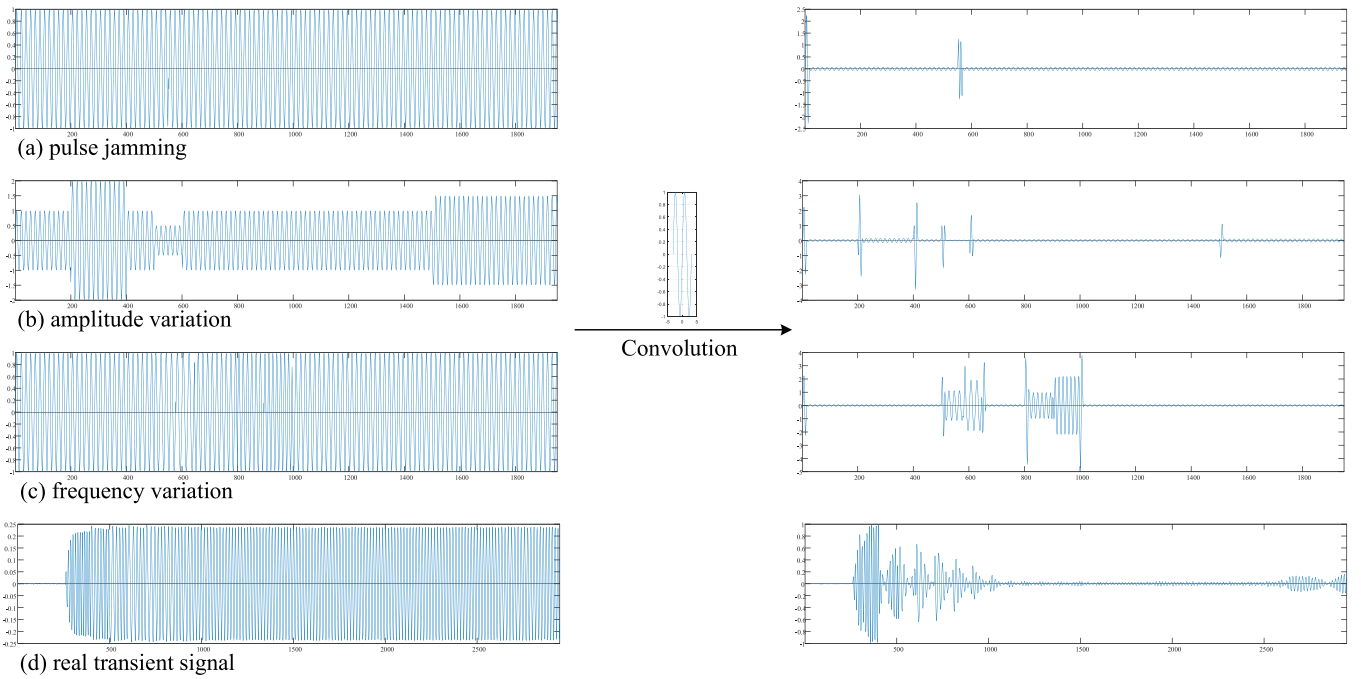


Fig. 2. Convolution results of typical waveforms

To facilitate calculation and storage, the definition domain of $\sin 2\omega t$ is limited to $(-\pi, \pi)$

$$s_0(t) = \sin 2\omega t[u(t+\pi) - u(t-\pi)] \tag{7}$$

of which the waveform is shown in the middle of Fig. 2. It has the same characteristics as $\sin 2\omega t$.

$$f(t) * s_0(t) = 0 \tag{8}$$

Typical signal changes can be summarized as (a) pulse jamming, (b) amplitude variation and (c) frequency variation, which are illustrated on the left of Fig. 2.

a) For pulse jamming, suppose it is located at $t = t_0$, the waveform is

$$f_p(t) = \sin \omega t + \delta(t - t_0) \tag{9}$$

When convoluted with the kernel $s_0(t)$

$$\begin{aligned} f_p(t) * s_0(t) &= [\sin \omega t + \delta(t - t_0)] * s_0(t) \\ &= s_0(t) * \sin \omega t + s_0(t) * \delta(t - t_0) \\ &= s_0(t - t_0) \end{aligned} \tag{10}$$

Thus, there would be a pulse of the same shape as the convolution kernel.

b) For amplitude variation, when the waveform amplitude becomes to k times of the original

$$f_a(t) = k \sin \omega t \tag{11}$$

When convoluted with the kernel $s_0(t)$

$$s_0(t) * k \sin \omega t = 0 \tag{12}$$

At the edge of amplitude variation, suppose the amplitude changes at $c + t_0$

$$f_a{}'(t) = \begin{cases} \sin \omega t & t \in [-\pi + t_0, c + t_0) \\ k \sin \omega t & t \in [c + t_0, \pi + t_0) \end{cases} \tag{13}$$

When convoluted with the kernel $s_0(t)$

$$\int_{-\pi}^{\pi} s_0(t) f_a{}'(t) dt = \int_{-\pi}^{c} s_0(t) \sin \omega t dt + \int_{c}^{\pi} s_0(t) k \sin \omega t dt \tag{14}$$

When $c \in (-\pi\pi)$ and $c \neq 0$, then the convolution result

$$\int_{-\pi}^{\pi} s_0(t) f_a{}'(t) dt \neq 0 \tag{15}$$

which will show a pulse in convolution result.

c) For frequency variation, when the frequency of the waveform changes

$$f_r(t) = \sin n\omega t \tag{16}$$

When convoluted with the kernel $s_0(t)$

$$s_0(t) * \sin(n\omega t) = \frac{4 \sin(n\omega\pi)}{(n\omega)^2 - 4} \sin(n\omega(t - \tau)) \tag{17}$$

where τ is a phase shift by convolution. Set $n\omega{=}m$, the amplitude function of m

$$g(m) = \frac{4 \sin(m\pi)}{m^2 - 4} \tag{18}$$

is illustrated in Fig. 3.

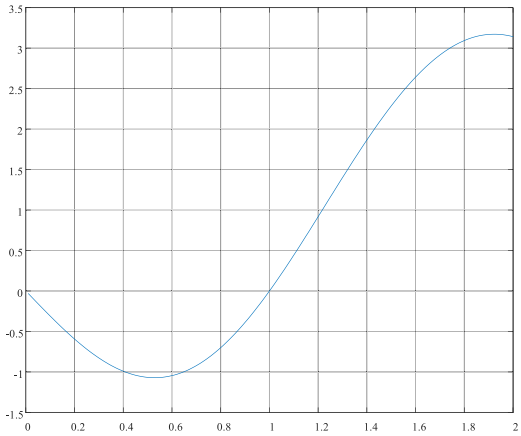


Fig. 3. Amplitude function of frequency variation via double frequency convolution kernel

When $m = 1$, the amplitude function $g(m) = 0$ as mentioned before. When $m \in (1, 1+e)(e$ is a small value), $g(m)$ is monotonically increasing, which means when the value of m increases from 1, the amplitude of the convolution result increases correspondingly. On the contrary, when $m \in (1-e, 1)$, $g(m)$ is also monotonic, which means when m decreases from 1, the absolute value of amplitude will still increase.

The above mentioned three typical waveforms and one real transient signal are shown at the left of Fig. 2, their convolution results are illustrated at the right.

a) The convolution result shows an obvious pulse when there is a pulse jamming in waveform, while the rest is basically 0. The shape of the pulse in the convolution result is consistent with that of the convolution kernel. The rest part cannot be 0 completely, because the waveforms in simulation are discrete.
b) The convolution result will show a pulse whenever there is an amplitude jump, whether the amplitude increases or decreases. The shape of the pulse is irregular, because the phases of the jump position are different and uncertain.
c) The amplitude of the convolution result will increase when the frequency of the waveform changes, whether increases or decreases. There will be a higher amplitude when the frequency variation is greater.
d) In the convolution result, the steady-state part of the signal is basically 0. While the convolution result of the transient part shows obvious fluctuation, especially for these parts with greater frequency variation.

The convolution kernel can effectively detect the amplitude and frequency variation of waveforms. However, because its convolution result with steady-state waveforms of any amplitude is 0, some other waveform information cannot be extracted. Moreover, the convolution kernel can only detect the variation in amplitude and frequency, but cannot accurately detect whether they increase or decrease.

Therefore, this convolution kernel can only be used to detect the changes in waveforms, while not able to extract the features of these changes effectively. Experiment proves that the convolution kernel with double frequency has a poor performance in signal classification or access authentication. Nevertheless, it can be used to accurately detect the transient part of the signal, where it has obvious variation. As shown in Fig. 2(d), the convoluted waveform has a high amplitude from sample point 201 to 1200, which shows a great changes in amplitude and frequency of the original waveform. While the rest part is basically 0, which means this part is practically stable.

B. Same Frequency Convolution Kernel

To ensure that there is enough information after convolution, two convolution kernels are proposed, which preserves the shape of the steady part of the waveform, while extracting the features of the variation part. The frequency of these two kernels is equal to that of the signal, expressed as

$$s_1(t) = \sin \omega t[u(t+\pi) - u(t-\pi)] \tag{19}$$

$$s_2(t) = \cos \omega t[u(t+\pi) - u(t-\pi)] \tag{20}$$

Their convolution results have these following characteristics (The convolution results of $s_1(t)$ and $s_2(t)$ in pulse jamming and amplitude variation are similar, the following will only show the results of $s_1(t)$)

Steady-state waveform

$$s_1(t) * \sin \omega t = \pi \sin \omega t \tag{21}$$

Pulse jamming

$$s_1(t) * [\sin \omega t + \delta(t - t_0)] = \pi \sin \omega t + s_1(t - t_0) \tag{22}$$

Amplitude variation

$$s_1(t) * k \sin \omega t = k\pi \sin \omega t \tag{23}$$

Frequency variation (The convolution results show difference between $s_1(t)$ and $s_2(t)$ in amplitude, the corresponding phase variation is neglected. Hence, $\sin(n\omega t)$ and $\cos(n\omega t)$ represent the same signal.)

$$f_s^*(t) = s_1(t) * \sin(n\omega t) = -\frac{2\sin(n\omega\pi)}{(n\omega)^2 - 1} \sin(n\omega t) \tag{24}$$

$$f_c^*(t) = s_2(t) * \cos(n\omega t) = -\frac{2n\omega \sin(n\omega\pi)}{(n\omega)^2 - 1} \cos(n\omega t) \tag{25}$$

Set $n\omega{=}m$, the amplitude functions of $f_s^*(t)$ and $f_c^*(t)$

$$g_1(m) = -\frac{2\sin(m\pi)}{m^2 - 1} \tag{26}$$

$$g_2(m) = -\frac{2m\sin(m\pi)}{m^2 - 1} \tag{27}$$

and their derivatives

$$g_1{}'(m) = \frac{4m\sin(m\pi)}{(m^2-1)^2} - \frac{2\pi\cos(m\pi)}{m^2 - 1} \tag{28}$$

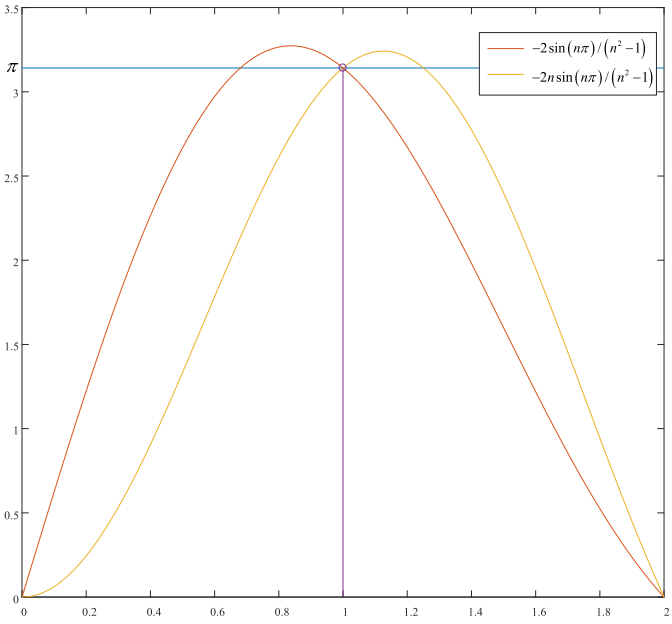


Fig. 4. Amplitude functions of frequency variation via same frequency convolution kernels

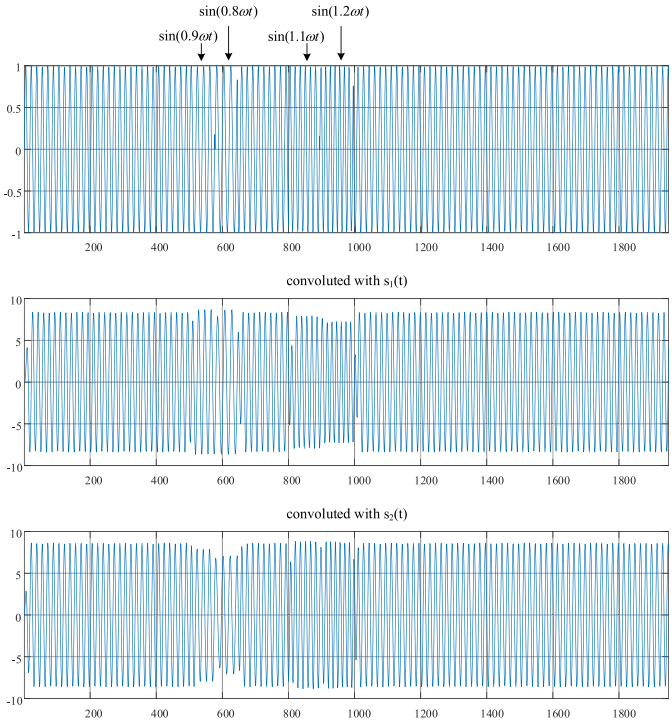


Fig. 5. Convolution result transform caused by frequency variation

$$g_2'(m) = \frac{4m^2\sin(m\pi)}{(m^2-1)^2} - \frac{2\sin(m\pi)+2m\pi\cos(m\pi)}{m^2-1} \quad (29)$$

are illustrated in Fig. 4.

$g_1(m)$ is monotonically decreasing in (1, 2), while $g_2(m)$ is monotonically increasing in (0, 1). Thus, after convoluted with $s_1(t)$, the amplitude of the function will show a significant decline when the frequency of the original signal rises.

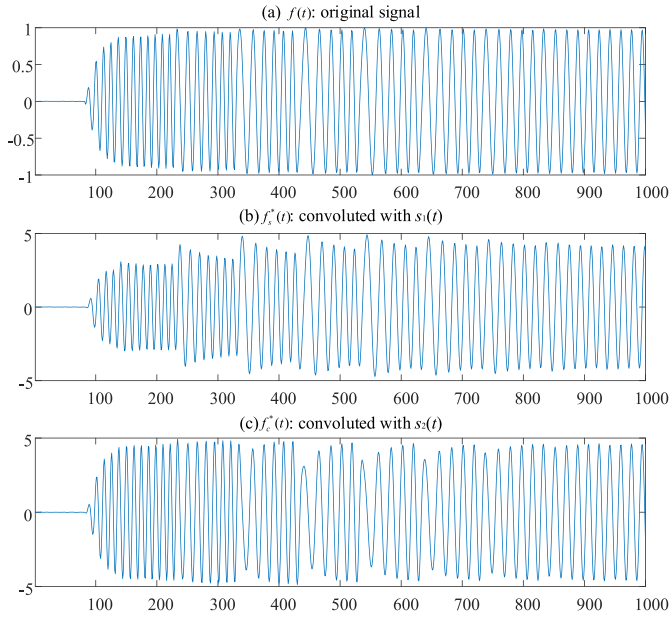


Fig. 6. Convolution result of practical signal

Meanwhile, the amplitude of the function convoluted with $s_2(t)$ will decrease obviously when the frequency drops.

The convolution results of $g_1(m)$ and $g_2(m)$ can be used together to extract the features of the frequency variation of the original signal, as shown in Fig. 5.

Similar results are obtained when convoluted with practical signal. Fig. 6(a) shows the practical signal collected from RF module, while its convolution results are illustrated as Fig. 6 (b) and (c), making the frequency change to amplitude change and facilitating the training of the neural network.

C. DWT Based Pooling

In CNN for image processing, pooling is to compress the input feature image. It makes the feature image smaller, simplifying the computational complexity of the neural network, and compresses the feature and extracts the main part. Max pooling and avy pooling are the two most commonly used pooling methods, while they are not suitable for signal processing, because the large amount of feature loss in signal is intolerable.

In digital signal processing, DWT can effectively extract the feature of the signal. Original signal is decomposed into two frequency parts by a half band low pass filter. The low frequency part contains most of the information of the original signal, of which the sampling points are halved compared with the original one.

Accordingly, DWT can be used to reduce the dimensionality of sample signals, just as the role that pooling plays in CNN.

Fig. 7 illustrates the two decomposed waveforms of the convoluted signal in Fig. 6(c). The low frequency part waveform $a_1(t)$ is almost the same as the original signal, of which most features are retained. While the dimensions, i.e. the number of sampling points, have been reduced by half. Similarity, the

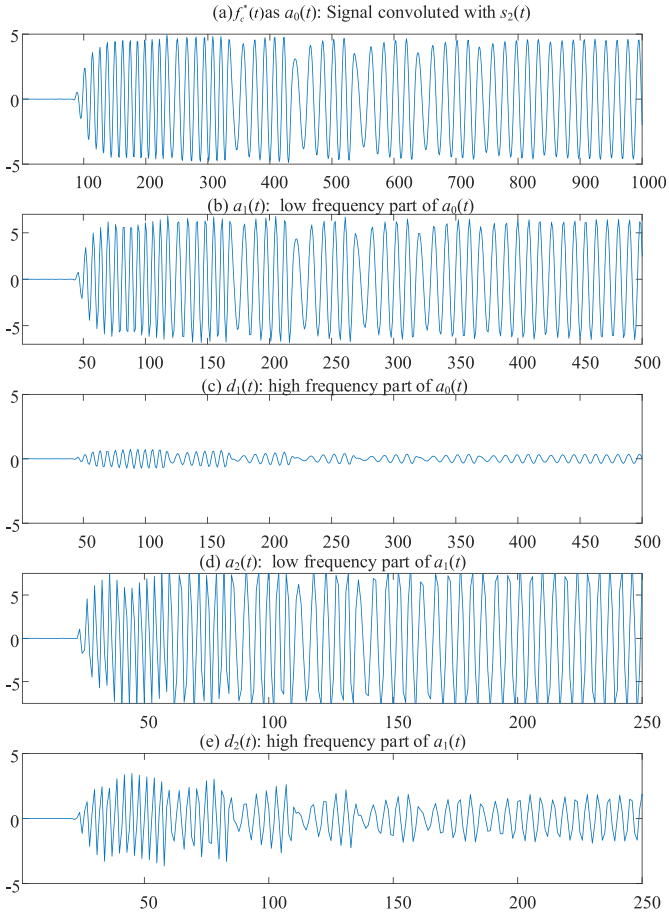


Fig. 7. Decomposed waveforms via two level DWT

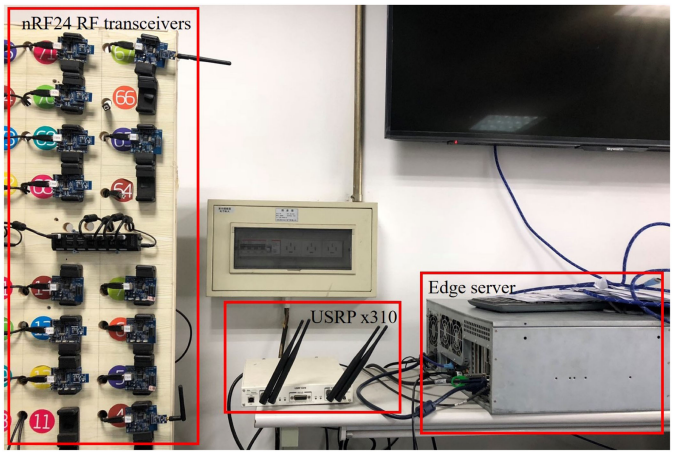


Fig. 8. Practical signal acquisition system

sampling points of $a_2(t)$ are halved compared with $a_1(t)$. However, due to the insufficient sampling points, the waveform has shown a distortion.

In this way, the output sample size can be effectively reduced, which can correspondingly reduce the computational complexity of machine learning and avoid over-fitting. The layer of DWT should be decided depend on the specific experimental result.

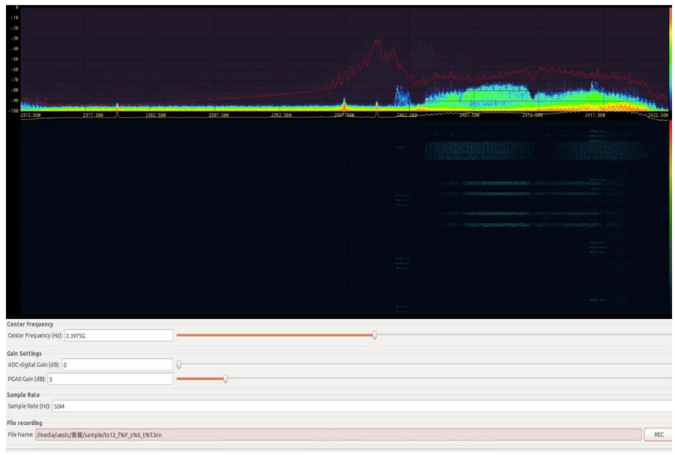


Fig. 9. Signal sampling settings in GUN radio

IV. Experiment System Model

A complete system is built for this access authentication experiment, including actual signal acquisition, signal processing and machine learning based classification.

The experimental edge computing system contains a edge server with signal transmission equipment (USRP X310) and several terminals (nRF24 RF transceiver) to realize the signal acquisition part, as shown in Fig. 8. Specifically, 10 terminals transmit pulse signals without information at 2.4 GHz. The edge server uses GNU radio software connected with USRP to receive signals with a central frequency of 2.3975 GHz, a sampling rate of 50 MHz and a PGA gain of 3 dB, as shown in Fig. 9. Tests show that it has a better reception effect when the center frequency is slightly lower than that of the transmitter.

Because the signals received by USRP are discrete, discrete signal processing will be used in the rest part of this section, instead of continuous signal processing in Section III.

Before extracting features of signals through convolution, the collected signals need to be preprocessed, including signal cutting, noise addition and normalization.

First, the collected signals are cut into individual parts for the next processing. Transient signal, signal start from scratch, has some changes in amplitude and frequency. As shown in Fig. 2 (d), transient signal can be extracted by convoluted with $s_0[n]$, which is a discrete function of $s_0(t)$ represented in formula (7), named as pre-convolution function in this experiment. Then, the Gauss white noise is generated at a certain SNR, and mixed with the signal to produce noisy signals for simulation test. Normalization limits the amplitude of the signal from -1 to 1, which can be measured uniformly.

Signal processing part is similar to image processing, which contains convolutional layer, DWT based pooling layer and full connecting (FC) layer. The sequence of steps is shown in Fig. 10.

1) Convolutional layer: The transient signal $f[n]$ is convoluted with convolution kernel functions $s_1[n]$ and $s_2[n]$ in formula (19) and (20) respectively, as $f_s^*[n]$ and $f_c^*[n]$.

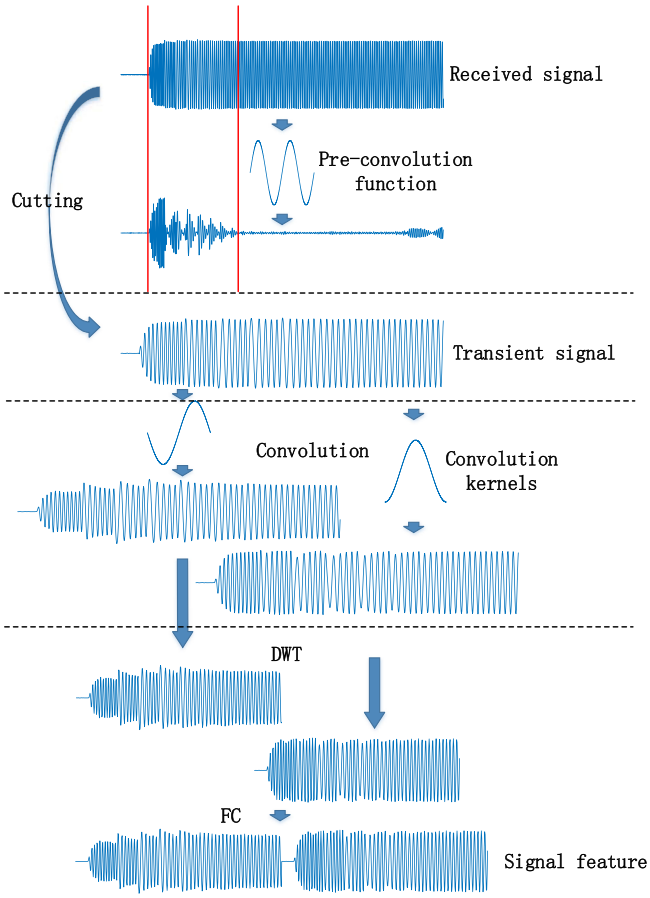


Fig. 10. Signal processing flow

$$f_s^*[n] = s_1[n] * f[n] \tag{30}$$

$$f_c^*[n] = s_2[n] * f[n] \tag{31}$$

2) Pooling layer: Perform DWT on convoluted functions $f_s^*[n]$ and $f_c^*[n]$, preserving their low-frequency parts, as $a_{s1}[n]$ and $a_{c1}[n]$.

$$(a_{s1}[n], d_{s1}[n]) = dwt(f_s^*[n]) \tag{32}$$

$$(a_{c1}[n], d_{c1}[n]) = dwt(f_c^*[n]) \tag{33}$$

where $dwt(*)$ means to do DWT as shown in formula (4) and (5). The high-frequency parts $d_{s1}[n]$ and $d_{c1}[n]$ won't be used in the next experiment, which are lacking in information.

Convolutional layer and pooling layer executed repeatedly before FC layer, if necessary, as shown in Fig. 11.

3) FC layer: Since waveform $a_{s1}[n]$ and $a_{c1}[n]$ are already 1D, only need to be linked together.

The connected waveform can be regarded as a feature of the original signal, which can be used as a training sample in the next classification.

V. Simulation Result and Discussion

All experimental simulation results presented in this section were generated using the signal processing system represented in Section IV and SVM as classifier. The original signals and several groups of convoluted waveforms were used in the experiment as training and test set. SNR was set from 0 to 25 dB, which corresponds to the SNR of the real signal. Two experiments were carried out to test the classification accuracy and the time cost of the above-mentioned methods. Each experiment was proceeded 100 times and calculate the average classification accuracy to suppress the effect of randomness on classification result.

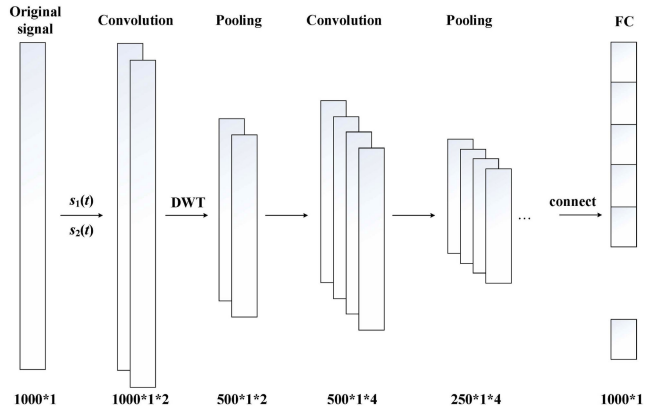


Fig. 11. Signal processing in multi-layer CNN mode

A. Classification Accuracy

The original transient signals, two groups of waveforms convoluted with kernel functions $s_1[n]$ and $s_2[n]$, and the full connected waveforms processed through DWT were used to test the classification accuracy, each of them has 1000 sampling points, as illustrated in Fig. 10. Besides, a group of full connected waveforms without DWT were used to test the effect of DWT, which has 2000 sampling points. SVM was adopted as machine learning algorithm in the experiment, these groups of waveforms were used as training set respectively. Each training set has 100 samples. Meanwhile, another 100 different samples were used as test set. The classification result is shown in Fig. 12.

Whether convoluted with $s_1[n]$ or $s_2[n]$, the classification results have a higher accuracy than that of the original signal, especially when SNR is in 0 to 5 dB. The full connected waveforms processed through convolution and DWT, which are recommended as the features in this paper, have a superior classification result. Because it contains the feature of the above two waveforms.

However, the full connected waveforms processed through convolution but without DWT have a much lower classification accuracy than that of the original signal. This result is due to its excessive sample points, which leads to a serious over-fitting.

B. Complexity Evaluation

The training time of sample sets with higher classification accuracy will be discussed in this part to find optimal algorithm in different situations. The original transient signal and the full connected waveform, which was proved to have the highest accuracy were used as training sets. This full connected waveform, which is processed through one-layer convolution and DWT, is

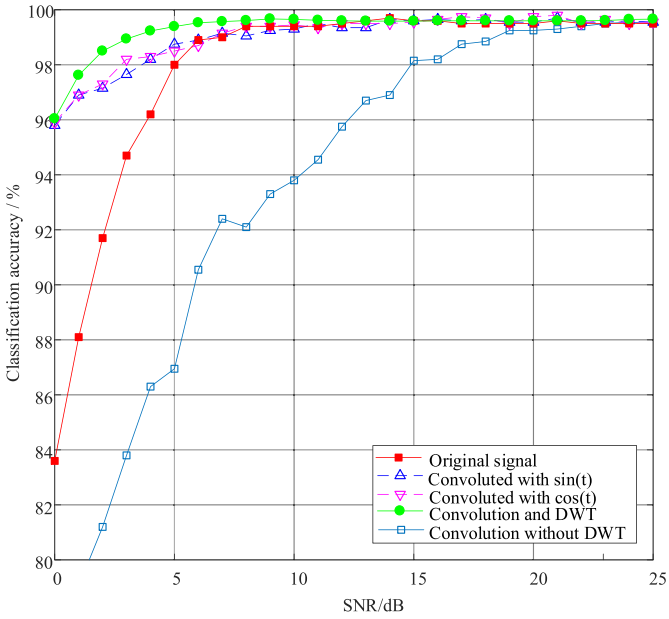


Fig. 12. Classification result of different waveforms

named as single-layer processed waveform. A group of double-layer processed waveforms were also tested in the experiment. The classification accuracy and time cost are illustrated in Fig. 13.

Fig. 13 shows that the double-layer processed waveform has a similar classification accuracy to the single-layer ones. While, the single-layer processed waveform has a shorter training time than the double-layer ones. Compared with the classification using the original signal as training set, the training time of single-layer process is only increased by about 10%. While the classification accuracy has been greatly improved in the range of 0 to 10 dB. Considering both classification accuracy and time cost, the single-layer ones has more advantages.

When SNR rises to 10 dB or more, the classification accuracy of the original transient signal is no longer inferior, reaches 99.6% or more. Thus, the best choice in high SNR is using the original transient signals as training set directly. Convolution and DWT based signal feature extraction is more suitable for low SNR scenarios, from 0 to 10 dB.

In addition, as shown in Fig. 13(b), the training time tends to decrease with the increase of SNR. That is because with the increase of SNR, the difficulty for SVM to find the optimal hyperplane decreases, hence, the training time, which is the time needed to find the optimal hyperplane, decreases.

The training results and time cost of CNN are tested in the experiment. When SNR is 5 dB, the single-layer CNN needs 1800 loops to achieve a similar result, accuracy higher than 99%, to that of SVM. The training time is listed in Table I.

The training time of CNN cannot meet the real-time requirement of edge computing. SVM-based classification algorithm has much shorter training time than that of CNN on the premise of achieving the same classification accuracy.

Considering classification accuracy and complexity, SVM-based classifier used in the simulation experiment is superior to CNN.

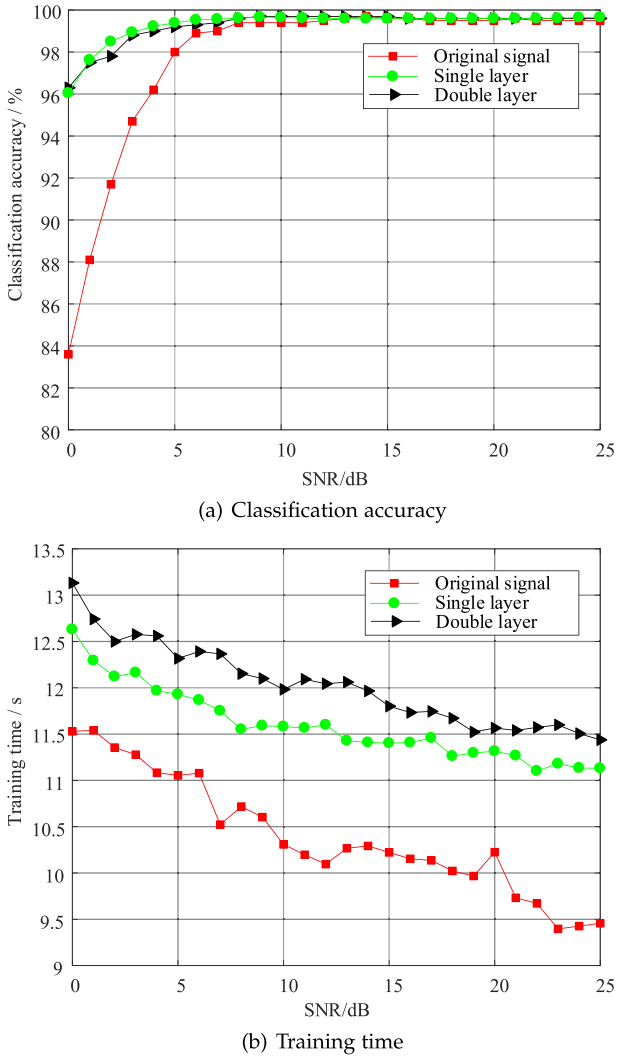


Fig. 13. Classification result of accuracy and training time of SVM

TABLE I
TRAINING TIME OF SVM AND CNN

Algorithm	SVM			CNN
	Original signal	Single-layer	Double-layer	
Training time (s)	10.717	11.552	12.157	705.557

VI. CONCLUSION

In this paper, an algorithm of signal feature extraction based on 1D convolution is proposed, which consider both the lightweight and accuracy of authentication for the terminals in the edge computing system. Experimental simulations show that the algorithm improves the classification accuracy in low SNR, while keep the training time of SVM in an acceptable range, by which the efficiency and security of edge computing access authentication is enhanced. In addition, the complete system of signal acquisition, processing and classification adopted in this paper can provide reference for access authentication of edge computing based on RFF.

ACKNOWLEDGMENTS

This work was supported in part by the National major R&D program underGrant2018YFB0904900 and Grant2018YFB0904905, inpart bythe Sichuan sci and tech service developement project under Grant 18KJFWSF0368, in part by the Sichuan sci and tech basic research condition platform project under Grant 2018TJPT0041, and in part by the Chile CONICYT under Grant Fondecyt Regular 181809.

REFERENCES

[1] H. Khaleel *et al.*, "Heterogeneous applications, tools, and methodologies in the car manufacturing industry through an IoT approach," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1412–1423, Sep. 2017.

[2] G. Fortino, W. Russo, C. Savaglio, W. Shen, and M. Zhou, "Agent-oriented cooperative smart objects: From IoT system design to implementation," *IEEE Trans. Syst. Man Cybernetics, Syst.*, vol. 48, no. 11, pp. 1939–1956, Nov. 2018.

[3] X. Peng, J. Ren, L. She, D. Zhang, J. Li, and Y. Zhang, "BOAT: A block-streaming APP execution scheme for lightweight IoT devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1816–1829, Jun. 2018.

[4] X. Li, Y. Huo, R. Zhang, and L. Hanzo, "User-centric visible light communications for energy-efficient scalable video streaming," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 1, pp. 59–73, Mar. 2017.

[5] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, Feb. 2014.

[6] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[7] J. Huang, P. Tsai, and I. Liao, "Implementing publish/subscribe pattern for CoAP in fog computing environment," in *Proc. IEEE AnnualIEMCON*, Oct. 2017, pp. 198–203.

[8] J. Zhao, C. Qiao, S. Yoon, and R. S. Sudhaakar, "Enabling multi-hop communications through cross-layer design for hybrid WSNs with transmit-only nodes," in *Proc. IEEE GLOBECOM*, Dec. 2011, pp. 1–5.

[9] L. Hu *et al.*, "Cooperative jamming for physical layer security enhancement in Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 219–228, Feb. 2018.

[10] H. Chien, "SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity," *IEEE Trans. Dependable Secure Comput.*, vol. 4, no. 4, pp. 337–340, Oct.–Dec. 2007.

[11] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1086–1090, Mar. 2009.

[12] Y. Chen *et al.*, "Lightweight one-time password authentication scheme based on radio frequency fingerprinting," *IET Commun.*, vol. 12, no. 12, pp. 1477–1484, Mar. 2018.

[13] F. Pan, Z. Pang, M. Xiao, H. Wen, and R. Liao, "Clone detection based on physical layer reputation for proximity service," *IEEE Access*, vol. 7, pp. 3948–3957, Jan. 2019.

[14] I. Kennedy, P. Scanlon, F. Mullany, M. Buddhikot, K. Nolan, and T. Rondeau, "Radio transmitter fingerprinting: a steady state frequency domain approach," in *Proc. IEEE Veh. Tech. Conf.*, 2008, pp. 1–5.

[15] R. Klein, M. Temple, and M. Mendenhall, "Application of wavelet-based RF fingerprinting to enhance wireless network security," *IEEE/KICS J. Commun. Netw.*, vol. 11, no. 6, pp. 544–555, Dec. 2009.

[16] F. Xie *et al.*, "Optimized coherent integration based radio frequency fingerprinting in Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3967–3977, Oct. 2018.

[17] M. Williams, M. Temple, and D. Reising, "Augmenting bit-level network security using physical layer RF-DNA fingerprinting," in *Proc. GLOBECOM*, 2010, pp. 1–6.

[18] C. Bertoncini, K. Rudd, B. Nousain, and M. Hinders, "Wavelet fingerprinting of radio-frequency identification (RFID) tags," *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4843–4850, Dec. 2012.

[19] G. Revadigar, C. Javali, W. Hu, and S. Jha, "DLINK: Dual link based radio frequency fingerprinting for wearable devices," in *Proc. IEEE Conf. Local Computer Netw.*, 2015, pp. 329–337.

[20] D. Chen *et al.*, "S2M: A lightweight acoustic fingerprints-based wireless device authentication protocol," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 88–100, Feb. 2017.

[21] F. Xie, S. Chen, Y. Chen, R. Liao, and H. Wen, "Algorithm and implementation of radio frequency fingerprinting based on multi-resolution analysis," in *Proc. IEEE Int. Symp. EMC*, 2017, pp. 1–6.

[22] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[23] H. Pham, L. Khoudour, A. Crouzil, P. Zegers, and S. A. Velastin, "Learning to recognise 3D human action from a new skeleton-based representation using deep convolutional neural networks," *IET Comput. Vision*, vol. 13, no. 3, pp. 319–328, 2019.

[24] Y. Chen and W. Chen, "Finger ECG based two-phase authentication using 1D convolutional neural networks," in *Proc. Annu. Int. Conf. EMBC*, Jul. 2018, pp. 336–339.

[25] M. Unser and I. Daubechies, "On the approximation power of convolution-based least squares versus interpolation," *IEEE Trans. Signal Process.*, vol. 45, no. 7, pp. 1697–1711, Jul. 1997.

[26] G. Harikumar and Y. Bresler, "FIR perfect signal reconstruction from multiple convolutions: Minimum deconvolver orders," *IEEE Trans. Signal Process.*, vol. 46, no. 1, pp. 215–218, Jan. 1998.

[27] M. J. Narasimha, "Modified overlap-add and overlap-save convolution algorithms for real signals," *IEEE Signal Process. Lett.*, vol. 13, no. 11, pp. 669–671, Nov. 2006.

[28] J. Martinez, R. Heusdens, and R. C. Hendriks, "A generalized poisson summation formula and its application to fast linear convolution," *IEEE Signal Process. Lett.*, vol. 18, no. 9, pp. 501–504, Sep. 2011.

[29] H. Song, H. Wen, H. Lin, and R. Liao, "Secure cooperative transmission with imperfect channel state information based on BPNN," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10 482–10 491, Nov. 2018.

[30] C. S. Félix, F. G. Walter, G. C. Raúl, and C. F. Dora, "Multiresolution analysis based on mallat pyramidal algorithm applied to GPR data," in *Proc. IEEE Ground Penetrating Radar*, Jun. 2014, pp. 647–650.

[31] M. J. Shensa, "The discrete wavelet transform: Wedding the a trous and Mallat algorithms," *IEEE Trans. Signal Process.*, vol. 40, no. 10, pp. 2464–2482, Oct. 1992.

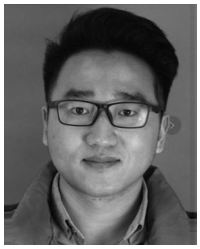
Feiyi Xie received the B.Eng. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2012, and he is currently working toward the Ph.D. degree with the University of Electronic Science and Technology of China, Chengdu, China.

He is currently engaged in the study of physical layer security and edge computing. His main research interests include wireless and mobile communications.

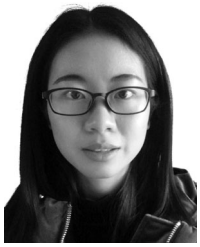
Hong Wen was born in Chengdu, China. She received the M.Sc. degree in electrical engineering from the Sichuan Union University of Sichuan, Chengdu, China, in 1997, and the Ph.D. degree from the Department of Communication and Computer Engineering, Southwest Jiaotong University, Chengdu, China. From 2008 to 2009, she was a Visiting Scholar and a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada.

She is currently a Professor with the University of Electronic Science and Technology of China. Her current research interest includes wireless communication systems and security.

Jinsong Wu is currently the Founder and Founding Chair of Technical Committee on Green Communications and Computing, IEEE Communications Society. He is also the Founder and Editor of Series on Green Communication and Computing networks of the IEEE Communications Magazine. He was a Guest Editor for the IEEE SYSTEMS JOURNAL, the IEEE ACCESS JOURNAL, the IEEE JOURNAL ON SELECTED AREAS ON COMMUNICATIONS (JSAC), the IEEE COMMUNICATIONS MAGAZINE, the IEEE WIRELESS COMMUNICATION, and *Elsevier Computer Networks*. He is currently an Area Editor for the JSAC Series on Green Communications and Networking.



Songlin Chen was born in Chengdu, P.R.China. He is currently working toward the Ph.D. degree in communication and information system with the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China. His current main research interests include physical security in wireless communication systems, edge computing, and smart grid.



Wenjing Hou is currently working toward the master's degree in communication and information system from the National Key Laboratory of Science and Technology on Communication, University of Electronic Science and Technology of China, Chengdu, China.

Her current research interests include wireless communication systems security and edge computing based terminal security research.

Yixin Jiang received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2006.

From 2011, he was a Senior Researcher with EPRI, China Southern Grid. He has authored and coauthored more than 80 papers in research journals and IEEE conference proceedings in these areas. His research interests include smart grid, security and performance evaluation in wireless communication and mobile computing.