Deep Learning Based RF Fingerprint Identification Using Differential Constellation Trace Figure

Linning Peng, *Member, IEEE*, Junqing Zhang, Ming Liu and Aiqun Hu

***Abstract*—This paper proposes a novel deep learning-based radio frequency fingerprint (RFF) identification method for internet of things (IoT) terminal authentications. Differential constellation trace figure (DCTF), a two-dimensional (2D) representation of differential relationship of signal time series, is utilized to extract RFF features without requiring any synchronization. A convolutional neural network (CNN) is then designed to identify different devices using DCTF features. Compared to the existing CNN-based RFF identification methods, the proposed DCTF-CNN possesses the merits of high identification accuracy, zero prior information and low complexity. Experimental results have demonstrated that the proposed DCTF-CNN can achieve an identification accuracy as high as 99.1% and 93.8% under SNR levels of 30 dB and 15 dB, respectively, when classifying 54 target ZigBee devices, which significantly outperforms the existing RFF identification methods.**

***Index Terms*—Physical layer security, radio frequency fingerprint, differential constellation trace figure, convolutional neural network**

I. Introduction

RADIO frequency fingerprint (RFF)-based device identification is an emerging physical layer (PHY) security technique to classify the identity of wireless devices [1], [2], which could be used in many internet of things (IoT) applications such as vehicular communications, electronic vehicle license plate authentication, etc. There are subtle differences among wireless components because of the manufacturing imperfections. These differences are termed as RFF, which are unique and persistent, and can be treated as the "DNA" of a device [3]. RFF will be presented in signal waveforms and RFF identification designs advanced signal processing protocols to extract these unique and inherent physical features of each device to authenticate the user identity.

RFF identification usually involves a two-stage protocol, including training and classification, which is a classical machine learning classification problem. At the training stage, the devices will enrol their RFF by transmitting a signal to the authenticator who will extract a particular transmitter RFF feature, e.g., carrier frequency offset (CFO), In-phase (I) and Quadrature (Q) offset, non-linear variation. The CFO is caused by the oscillator variations. The I/Q offset is generated from I/Q channel gain and DC offset. The transmitter amplifier non-linearity creates unique non-linear behaviour of waveforms near maximal power. The authenticator will maintain a table of the device index and its RFF features in the database. During the classification stage, the authenticator will classify the index/label of the device according to the previously stored database when it receives a new transmission from a device to be classified. Many machine learning algorithms have been used, including k-nearest neighbor (KNN), support vector machine (SVM), random forest (RndF), multiple discriminant analysis (MDA), etc [4], [5]. However, the identification accuracies of these algorithms are limited especially when there are a large number of targets.

There are recent research efforts employing deep learning to improve the identification accuracy. Deep learning has been used widely for wireless channel estimations [6], [7], human motion behavior detections [8], speech language identifications [9] and RFF device identifications [10], [11]. Time domain I/Q complex baseband samples are directly used as the CNN input in most of convolutional neural network (CNN)-based RFF device identification. As the data samples are sequential input, we call this sample-based CNN. Merchant *et al.* achieved an accuracy of 91% over seven target devices in experiments [11]. However, in order to obtain stable I/Q samples for CNN training and verification, prior knowledge is required for carrier frequency and time synchronization [11]. In addition, due to the complexity consideration, the length of the I/Q sequence in existing sample-based CNN schemes are usually quite short [11], which results in a limited identification accuracy. Ding *et al.* designed a sophisticated bispectrum-based CNN method for RFF identification and achieved an accuracy of 87% over five target devices [10]. However, it is worth noting that the spectrum-based RFF features are sensitive to the noise and environment [12]. The bispectrum estimation also requires high computational complexity. Hilbert-Huang transform (HHT) is introduced for RFF feature extractions in [13]. Similar to the bispectrum, different Hilbert spectrums are employed for identification. An algorithm employing the Fisher's discriminant ratio (FDR) is used to select elements of the Hilbert spectrum for classifications.

A differential constellation trace figure (DCTF)-based feature extraction method is introduced in [14], which converts the time domain I/Q samples to a 2D image containing RFF features. DCTF essentially provides a visualization of the sta-

This work was supported in part by the National Natural Science Foundation of China under Grant 61501022, 61571110, 61601114, 61602113 and Purple Mountain Laboratories for Network and Communication Security. The work of J. Zhang was supported by Royal Society Research Grants under grant ID RGS/R1/191241.

L. Peng and A. Hu are with the School of Cyber Science and Engineering, Southeast University, No. 2 Sipailou, Nanjing, China and Purple Mountain Laboratories, Nanjing, China (email: {pengln, aqhu}@seu.edu.cn).

J. Zhang is with the Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, L69 3GJ, United Kingdom. (email: junqing.zhang@liverpool.ac.uk).

M. Liu is with the Beijing Key Lab of Transportation Data Analysis and Mining, Beijing Jiaotong University, No. 3 Shangyuancun, Beijing, China. (email: mingliu@bjtu.edu.cn).

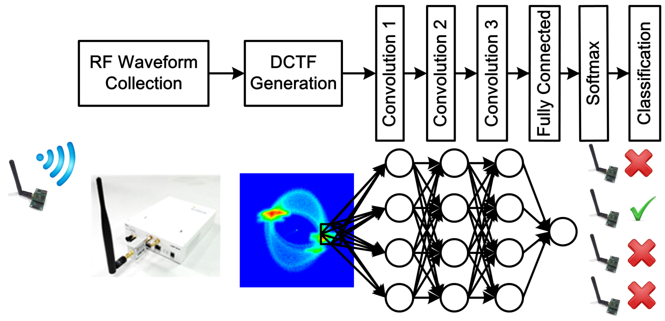

Fig. 1. DCTF-CNN-based RFF identification

tistical distribution of differential operation. The concentration of the differential results reflects the statistical average, and their spread suggests the variation due to the noise and channel randomness. In general, the high intensity in DCTF represents the influence of RFF features, such as CFO, I/Q offset, non-linear variation and so on.

Motivated by the success of CNN at complex classification tasks in image recognition, this paper applies CNN to classify DCTFs obtained from different devices. Being different from most of RFF identification schemes using sample-based CNN, the proposed scheme directly classifies different targets using the "fingerprint" liked figures, which is termed as figure-based CNN. The main contributions of this paper are as follows:

- We propose and design a novel DCTF and CNN-based scheme for RFF identification. The generated DCTF is a "fingerprint" liked figure and thus image recognition CNNs can be used for identification.
- We analyze the source of the RFF behaviours in DCTF generation. Compared with existing sample-based CNN for RFF identification, the proposed DCTF-based feature extraction does not require any prior information for synchronization. In addition, the CFO feature, which is usually eliminated in sample-based CNN schemes, is naturally retained in DCTF-CNN.
- The DCTF-CNN parameters are optimized for RFF identification, which enables that the DCTF-CNN scheme outperforms existing CNN-based RFF identification methods in our 54 ZigBee devices classification.
- The implementation complexity of DCTF-CNN is investigated. The DCTF generation is a statistical process, which is with low complexity. The final DCTF-based CNN complexity is determined by the DCTF size. Therefore, DCTF-CNN can utilize any segment of the received signal, regardless of the original signal length.

II. DCTF-CNN-BASED RFF IDENTIFICATION

The system block diagram is shown in Fig. 1. We first sample the waveforms from RF devices. A DCTF generation method is then employed to generate the "fingerprint" figure for each device. A deep learning CNN is designed to train and classify different RF modules based on their unique DCTFs.

DCTF is beneficial because it would include most of the RFF. The transmitted RF signal can be written as:

$$X(t) = \big(\beta_I(x_I(t)+\alpha_I)+j\beta_Q(x_Q(t)+\alpha_Q)\big)\cdot e^{-j2\pi f_c^t t}, \quad (1)$$

where $x_I(t)$ and $x_Q(t)$ are the signals at the I and Q branches, respectively, f_c^t is the carrier frequency at transmitter, α_I and α_Q are the DC offset at I/Q channels, β_I and β_Q are the I/Q gain imbalances. We assume that the channel and receiver is ideal, the received signal could be downconverted as:

$$Y(t) = y_I(t) + jy_Q(t) = X(t)\cdot e^{j2\pi f_c^r t}, \quad (2)$$

where $y_I(t)$ and $y_Q(t)$ are the received baseband I and Q signal components, respectively; f_c^r is the carrier frequency at the receiver. Because of the manufacturing imperfections, f_c^t and f_c^r will deviate with a slight CFO, $\psi = f_c^r - f_c^t$.

The differential process is carried out with no frequency and time synchronization required, which can be given as

$$\begin{aligned} D(t) &= d_I(t) + jd_Q(t) \\ &= \big(y_I(t) + jy_Q(t+\varepsilon)\big)\cdot\big(y_I(t+\lambda)+jy_Q(t+\lambda+\varepsilon)\big)^*, \end{aligned} \quad (3)$$

where $d_I(t)$ and $d_Q(t)$ are the I/Q channel signals, λ is the differential time interval, ε is the introduced I/Q phase mismatch to enlarge the fingerprint feature, and $(\cdot)^*$ denotes the conjugation operation. Upon substitution of (1) and (2) into (3), the differentiated signal can be further written as

$$\begin{aligned} D(t) =& \big(\beta_I(x_I(t)+\alpha_I)+j\beta_Q(x_Q(t+\varepsilon)+\alpha_Q)\big)\cdot e^{j2\pi\psi t} \\ &\cdot\big(\beta_I(x_I(t+\lambda)+\alpha_I)-j\beta_Q(x_Q(t+\lambda+\varepsilon)+\alpha_Q)\big) \\ &\cdot e^{-j2\pi\psi(t+\lambda)} \\ =& \big(\beta_I^2(x_I(t)+\alpha_I)\cdot(x_I(t+\lambda)+\alpha_I) \\ &+\beta_Q^2(x_Q(t+\varepsilon)+\alpha_Q)\cdot(x_Q(t+\lambda+\varepsilon)+\alpha_Q) \\ &+j(\beta_I\beta_Q(x_Q(t+\varepsilon)+\alpha_Q)\cdot(x_I(t+\lambda)+\alpha_I) \\ &-\beta_I\beta_Q(x_I(t)+\alpha_I)\cdot(x_Q(t+\lambda+\varepsilon)+\alpha_Q))) \\ &\cdot e^{-j2\pi\lambda\psi}. \end{aligned} \quad (4)$$

The CFO ψ turns into a fixed parameter $e^{-j2\pi\lambda\psi}$ in (4), which causes a fixed I/Q sample rotation. For the simplicity of analysis, we assume that ε is 0. We analyse the distribution of samples in $D(t)$ with the following two simple scenarios:

$$\begin{aligned} &\text{(i)} \quad x_I(t) = x_I(t+\lambda), x_Q(t) = x_Q(t+\lambda); \\ &\text{(ii)} \quad x_I(t) = -x_I(t+\lambda), x_Q(t) = -x_Q(t+\lambda). \end{aligned} \quad (5)$$

In the case (i), $D(t)$ can be simplified as

$$\begin{aligned} \text{(i)} \quad D(t) = &\big((\beta_I^2 x_I^2(t) + \beta_Q^2 x_Q^2(t)) + \beta_I^2\alpha_I(\alpha_I + 2x_I(t)) \\ &+ \beta_Q^2\alpha_Q(\alpha_Q + 2x_Q(t))\big)\cdot e^{-j2\pi\lambda\psi}, \end{aligned} \quad (6)$$

In the case (ii), $D(t)$ can be simplified as

$$\begin{aligned} \text{(ii)} \quad D(t) = &\big(-(\beta_I^2 x_I^2(t) + \beta_Q^2 x_Q^2(t)) + \beta_I^2\alpha_I^2 + \beta_Q^2\alpha_Q^2 \\ &+ j2\beta_I\beta_Q(x_Q(t)\alpha_I - x_I(t)\alpha_Q)\big)\cdot e^{-j2\pi\lambda\psi}. \end{aligned} \quad (7)$$

In (6) and (7), I/Q DC offset α_I and α_Q are much smaller than the signal components $x_I(t)$ and $x_Q(t)$. Therefore, the I channel signal $(\beta_I^2 x_I^2(t) + \beta_Q^2 x_Q^2(t))$ could present the transmitted signal power with the I/Q gain imbalance impact, which could be mostly affected by amplifier non-linearity. In addition, we can also find that in (7), $D(t)$ exists residual Q channel signal caused from I/Q DC offset α_I and α_Q. The differentiated signal $D(t)$ owns different gathering centers

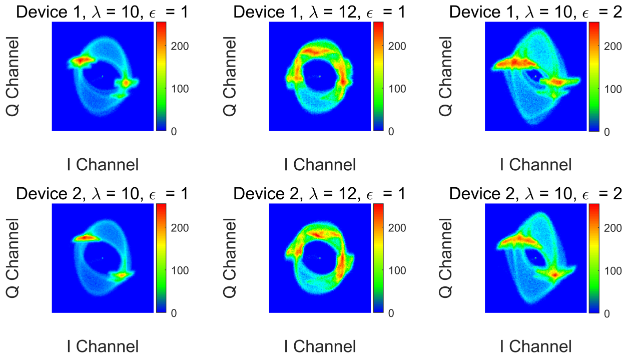

Fig. 2. Generated DCTFs from real devices with different parameters

around the maximal values at I channel with a fixed $e^{-j2\pi\lambda\psi}$ rotation.

It is intuitive that a method should be designed to present these features. Constellation map is straightforward to evaluate the signal qualities but synchronization will be required. An alternate constellation trace figure is employed to present the differential result $D(t)$. Furthermore, it is sensible to investigate the trace distribution density in the figure, which is the essential characteristic due to the RFF. A measurement matrix Φ is built to count the distribution density.

The measurement matrix Φ could be generated with a size of $M \times N$ pixel grids ranging from -A to +A. The values of elements $\Phi_{m,n}$ are initialized as zero. For each $D(t)$, we get the index (m, n) of Φ by following process:

$$m = [\frac{d_I(t) + A}{2A}M], n = [\frac{d_Q(t) + A}{2A}N], \qquad (8)$$

where $[\cdot]$ is the round operation. Then we add the element of $\Phi_{m,n}$ by 1 when $D(t)$ falls to the index (m, n). Finally, when all of the samples are counted and added in the specific index of measurement matrix, Φ could represent the distribution density of the entire waveform. The elements of Φ are normalized by re-scaling the pixel values from 0 to 255, which can be seen as a $M \times N$ image.

In practical systems, wireless signal is affected by channel fading, which causes different received signal powers. Therefore, the received signal is initially power-normalized. After differential process in (3), the ranging value A in (8) should be larger than 3 in order to avoid overflow in low SNR scenarios. Finally, we can obtain DCTF from measurement matrix Φ generated in aforementioned process. Some DCTF examples are shown in Fig. 2.

As DCTF is a 2D image, it inspires us to employ the popular deep learning CNN. Being different from the existing CNN designed to identify the I/Q samples, the neural network in our method aims to recognize the underlying RFF patterns in a 2D image. The neural network is trained with DCTFs from different devices in different conditions, e.g., SNR, λ, ε, image sizes ($M \times N$). In the classification stage, the scheme will classify the target device by finding out its label based on the network parameters provided by the CNN training stage.

III. Experimental Design

A. Experiment Setup

Extensive experiments were carried out to evaluate the performance of DCTF-CNN. $L = 54$ TI CC2530 system-on-chip ZigBee modules are employed as the target devices to be classified. A universal software radio peripheral (USRP) software defined radio (SDR) hardware platform is built to collect ZigBee RF waveforms. A PC installed with MATLAB R2018b and NVIDIA GTX1060 graphic card is setup for signal processing and carrying out CNN training and classification.

B. ZigBee Burst Collection

Similar to the most of the deep learning-based RF fingerprint identification work, we performed ten measurements for each ZigBee device at different locations with line-of-sight (LOS) transmission. The carrier frequency of the ZigBee device was set as 2505 MHz with offset quadrature phase shift keying (OQPSK) modulation. It took five minutes for the devices to warm up and reach a steady working condition.

A USRP receiver was employed to capture RF waveforms from different ZigBee devices at 2505 MHz. The receiver sampling rate was 10 Msample/s, which owns ten times oversampling compared to ZigBee 1M chip rate. Each measurement was composed of 9 frame segments and each segment can generate a DCTF. For each frame segment, the USRP receiver captured approximately 40K samples.

We use a spectrum estimation method to evaluate the SNR of the received signal. Due to the different locations and transmitter variations, the estimated SNR of the captured signals was 20∼25 dB. In order to save experiment time, we added different levels of additive white Gaussian noise (AWGN), varying from 0∼30 dB, to the received waveforms in order to emulate various signal qualities.

In order to optimize the DCTF-CNN parameters and evaluate the performance, we divided the overall ten measurement data into two groups, each with five measurements. The first group was used to find the optimal parameters of DCTF and CNN, including λ, ε, image size, batch size and initial learning rate; the second group was used for testing performance with different SNRs. For each group, we added specified AWGN via ten simulations and used the waveforms from two measurements for training. The training and validation sets were randomly split with 85% and 15% respectively. The rest three measurements served as separate test sets. In each measurement, we can obtain nine frames for DCTF generation. Therefore, for each group, we have $54{*}9{*}10{*}2{*}0.85 = 8262$ training DCTFs, $54{*}9{*}10{*}2{*}0.15 = 1458$ validation DCTFs and $54 * 9 * 10 * 3 = 14580$ testing DCTFs.

C. CNN Design

Inspired by the famous LeNet-5 network which has been successfully used to solve the canonical MNIST digits recognition [15], we chose a neural network with three 2D convolutional layers followed by a fully connected layer. The kernel size was set to [3×3] to capture the local details of RFF in the DCTF. Channel numbers were chosen to 16, 32 and 64 to accommodate more higher level features. The rectified linear unit (ReLU) was employed as the activation function to reduce the likelihood of vanishing gradient and accelerate the training. Two [2×2] max pooling layers were applied after the first two convolutional layers to down-sample the output features and limit the size of the neural network. A fully connected layer of L outputs together with the Softmax activation was used in the last stage to perform the classification among L target devices. Cross entropy for L mutually exclusive classes was adopted as the loss function of classification. Gradient descent algorithm was used in the network training process. For an input image size of [65×65], the parameters of each layer are as follows.

- **The first convolutional layer:** There are $(3^2) * 1 * 16$ weights and 16 biases. The number of trainable parameters is 160. After a [2×2] max pooling, the image size is [32×32].
- **The second convolutional layer:** There are $(3^2) * 16 * 32$ weights and 32 biases. The number of trainable parameters is 4640. After a [2×2] max pooling, the image size is [16×16].
- **The third convolutional layer:** There are $(3^2) * 32 * 64$ weights and 64 biases. The number of total trainable parameters is 18464.
- **The last fully connected layer:** As the final image size is [16×16] and channel number is 64, there are $(16^2) * 64$ weights and 1 bias for each tensor. The number of trainable parameters is L×16,385 when there are L target devices.

Therefore, the total trainable parameters of this network is 160+4640+18464+L×16385 = 23,264 + L×16,385, with weights and biases.

IV. Experimental Results and Analysis

A. *Performance with Different DCTF Generation Parameters*

When receivers have zero prior information of the target signal, the differential interval λ and introduced I/Q phase mismatch distortion ε are arbitrarily chosen because the signal symbol rate and modulation type are not known at the receiver. This section investigated effects of the differential interval and I/Q phase mismatch distortion. The SNR was 30 dB and the DCTF image size was 65 × 65.

Some examples of the obtained DCTFs with different generation parameters were shown in Fig. 2. The DCTFs varied greatly with λ and ε. The diversity among different devices was also distinguishable under specific parameter setup.

Fig. 3 depicts the accuracy of DCTF-CNN with different DCTF generation parameters. As shown in Fig. 3, although the accuracy dramatically decreases when extremely short differential interval λ is selected, the overall accuracy is higher than 97% when λ is larger than 3. In addition, Fig. 3 shows that the overall accuracy is always higher than 97% when ε is selected as 0, 1, 2 and 3. In particular, the introduced I/Q phase mismatch distortion ε of 1 had the best performance.

B. *Performance with Different DCTF Image Qualities*

DCTF image quality is affected by the DCTF size, $M \times N$, and the number of used samples for generating DCTF. It is obvious that the larger size of the DCTF image, the lower chance of the different samples falling into the same pixel, which consequently requires more data samples to maintain the details of the figure. On the other hand, lower size of DCTF will blur features among devices. In addition, the larger number of used samples for generating DCTF, the better quality of the image, at a cost of higher complexity. This section evaluated the effect of the image quality on the identification accuracy, when SNR = 30 dB, $\lambda = 10$ and $\varepsilon = 1$.

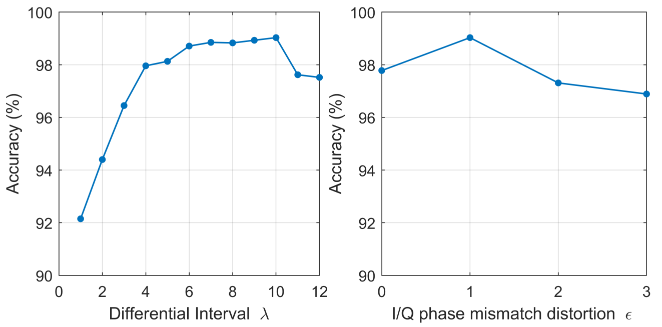

Fig. 3. DCTF-CNN accuracy with different intervals λ and I/Q phase mismatch distortions ε

TABLE I
CNN TRAINING COST WITH DIFFERENT DCTF SIZES

Size	Epochs	Time (s)	Size	Epochs	Time (s)
257 × 257	23	498	**33 × 33**	33	28
129 × 129	42	202	**17 × 17**	23	19
65 × 65	31	52	**9 × 9**	30	23

The accuracy with different DCTF image qualities is depicted in Fig. 4. An image size of 65 × 65 pixels offers the best performance for our designed CNN. When DCTF image size was smaller than 17 × 17, the RFF feature got blurred. It is also evident that the smaller DCTF image size, the less network complexity. Table I demonstrates the CNN training time cost with different DCTF sizes, which shows that the overall complexity of DCTF-CNN is very low. The maximal training time was only 498 seconds even with 257 × 257 DCTF image size. It is worth noting that the smaller size of DCTF image, the lower complexity of network computation. Therefore, the DCTF image size with 65 × 65 and 33 × 33 are preferable setup choices that achieve a satisfactory tradeoff between performance and complexity.

When DCTF image size is fixed, Fig. 4 shows that the more samples used for generating DCTF, the better identification accuracy, thanks to the better DCTF image quality. However, we find that with the DCTF size of 65 × 65, the accuracy difference between 20K and 40K samples was negligible. The DCTF-CNN accuracy can be higher than 97% when more than 10K samples were used.

C. *Performance with Different CNN Training Parameters*

CNN efficiency and accuracy are affected by batch sizes, maximal epoches and initial learning rates. In particular, batch size is one of the most important hyperparameters to tune

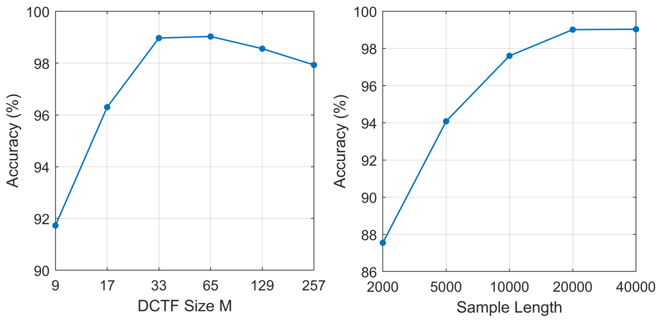

Fig. 4. DCTF-CNN accuracy with different DCTF image sizes (40000 samples) and number of used samples (65 × 65 image size)

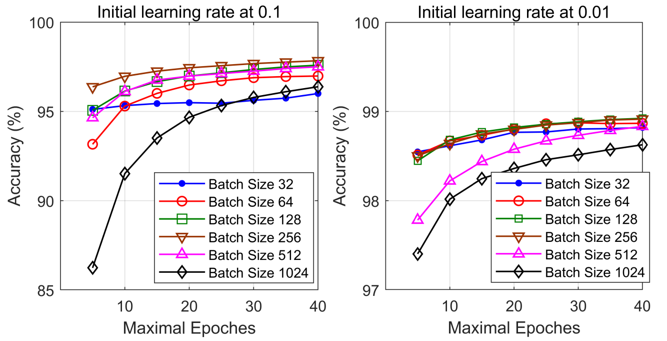

Fig. 5. DCTF-CNN accuracies with different batch sizes

in deep learning training [16]. In our 54 ZigBee devices identification problem, 8262 training DCTFs with the size of 65 × 65 were used. We evaluated the performance with different batch sizes from 32 to 1024 and different maximal epoches from 5 to 40. We also evaluated the performance with different initial learning rates at 0.01 and 0.1. Extensive simulations were taken with 20 loops averaging. The obtained results are shown in Fig. 5.

Fig. 5 demonstrates that the network training can be faster and more accurate when initial learning rate is reduced from 0.1 to 0.01. In addition, the system performance is better when the batch size is between 128 and 256. Larger batch size requires more hardware resources to load DCTFs and more epoches to get the converged result. However, it is observed that DCTF-CNN accuracies dramatically deteriorated when batch size was larger than 256 for both initial learning rates at 0.01 and 0.1. On the other hand, small batch sizes also cause performance degradations when initial learning rate is 0.1.

D. Performance with Different SNR

Configured with the optimized parameters, we evaluated the performance of the DCTF-CNN under different SNR levels in the range of {0, 5, 10, 15, 20, 25, 30} dB. The λ was set to 10 and ε was set to 1. The DCTF size was 65 × 65, 40K samples were used to generate a DCTF.

Fig. 6 compares the identification accuracy among the DCTF-based k-means classification method [14], time-domain I/Q signal-based CNN method [11], bispectrum-based CNN [10] and Hilbert-Huang Transform (HHT)-based method [13]. The same data set was used for all of the evaluations. The FFT length for bispectrum generation is 128. The HHT size is 512, the maximum number of intrinsic mode functions (IMF) extracted is 5. The DCTF-CNN confusion matrixes at 10 dB and 15 dB SNRs for the 54 ZigBee devices identification problem are depicted in Fig. 7. When SNR is higher than 20 dB, the accuracy of the proposed DCTF-CNN is higher than 98% and can even reach 99.1% at 30 dB.

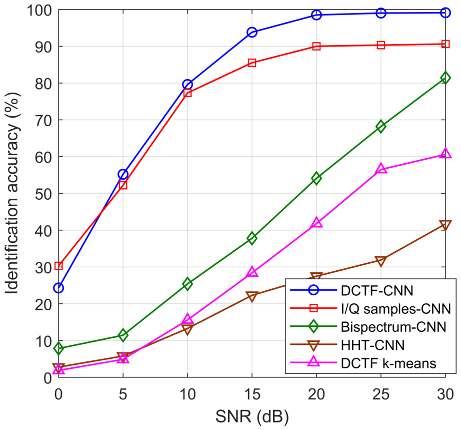

Fig. 6. Identification accuracy under different SNRs

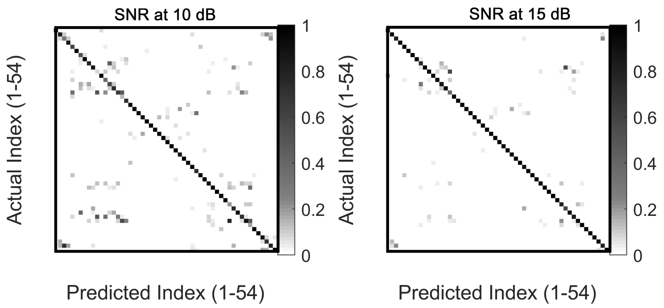

Fig. 7. DCTF-CNN confusion matrixes at 10 dB and 15 dB for the 54 ZigBee devices identification problem

The DCTF with a k-means clustering algorithm can achieve an accuracy of 60.1%. The best performance was around 96% even when the system was enhanced by multiple features [14]. Thanks to the feature representation and classification capability of CNN, the proposed DCTF-CNN acquired a significant performance enhancement, especially in high SNR scenarios. Our DCTF-CNN scheme can achieve an identification accuracy of 99.1% among 54 target devices in contrast to the 91.4% accuracy with I/Q samples-based CNN, 81.4% accuracy with bispectrum-based CNN and 41.7% accuracy with HHT-based CNN. In low SNR scenarios, the DCTF-CNN had similar performance compared to I/Q samples-based CNN, but also with higher accuracy than spectrum-based CNN methods.

V. Conclusion

In this correspondence, we proposed a novel RFF identification method that combined the DCTF-based feature extraction and the CNN-based classification. Different from existing sample-based CNN RFF identification methods that exploit complex (I/Q channel) signal samples, we directly classified target devices without requirement of any synchronization and compensation. In addition, the complexity of DCTF-CNN is very low due to the small input figure size. We carried out extensive experiments with a testbed including 54 ZigBee devices. We initially investigated parameter optimizations in our DCTF-CNN system. With the help of optimal DCTF generation parameters and CNN training parameters, the identification accuracy is as high as 99.1% and 93.8% at the SNR levels of 30 dB and 15 dB, respectively, which significantly outperforms the existing deep learning-based RFF identification methods. The future work includes study of target movements in DCTF-CNN identification.

References

[1] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 94–104, 2016.

[2] B. Danev, D. Zanetti, and S. Capkun, "On physical-layer identification of wireless devices," *ACM Computing Surveys*, vol. 45, no. 1, pp. 1–29, 2012.

[3] D. R. Reising, M. A. Temple, and J. A. Jackson, "Authorized and rogue device discrimination using dimensionally reduced RF-DNA fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 6, pp. 1180–1192, Jun. 2015.

[4] H. J. Patel, M. A. Temple, and R. O. Baldwin, "Improving ZigBee device network authentication using ensemble decision tree classifiers with radio frequency distinct native attribute fingerprinting," *IEEE Trans. Rel.*, vol. 64, no. 1, pp. 221–233, Mar. 2015.

[5] P. Robyns, E. Marin, W. Lamotte, P. Quax, D. Singelée, and B. Preneel, "Physical-layer fingerprinting of LoRa devices using supervised and zero-shot learning," in *Proc. ACM Conf. Security Privacy in Wireless and Mobile Networks (WiSec)*, Boston, USA, Jul. 2017, pp. 58 – 63.

[6] H. Huang, J. Yang, Y. Song, H. Huang, and G. Gui, "Deep learning for super-resolution channel estimation and doa estimation based massive mimo system," *IEEE Trans. Veh. Tech.*, vol. 67, no. 9, pp. 8549 – 8560, Sep. 2018.

[7] J. S. Choi, W. H. Lee, J. H. Lee, J. H. Lee, and S. C. Kim, "Deep learning based nlos identification with commodity wlan devices," *IEEE Trans. Veh. Tech.*, vol. 67, no. 4, pp. 3295–3303, Apr. 2018.

[8] R. Zhang and S. Cao, "Real-time human motion behavior detection via cnn using mmwave radar," *IEEE Sensors Letters*, vol. 3, no. 2, pp. 1–4, Feb. 2019.

[9] Z. Ma, H. Yu, W. Chen, and J. Guo, "Short utterance based speech language identification in intelligent vehicles with time-scale modifications and deep bottleneck features," *IEEE Trans. Veh. Tech.*, vol. 68, no. 1, pp. 121–128, Jan. 2019.

[10] L. Ding, S. Wang, F. Wang, and Z. Wei, "Specific emitter identification via convolutional neural networks," *IEEE Commun. Letters*, vol. 22, no. 12, pp. 2591–2594, Sep. 2018.

[11] K. Merchant, S. Revay, G. Stantchev, and B. Nousain, "Deep learning for RF device fingerprinting in cognitive communication networks," *IEEE J. Sel. Topics in Signal Processing*, vol. 12, no. 1, pp. 160–167, feb 2018.

[12] W. Wang, Z. Sun, S. Piao, B. Zhu, and K. Ren, "Wireless physical-layer identification: Modeling and validation," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 9, pp. 2091–2106, Sep. 2016.

[13] J. Zhang, F. Wang, O. Dobre, and Z. Zhong, "Specific emitter identification via hilbert-huang transform in single-hop and relaying scenarios," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 6, pp. 1192–1205, Jun. 2016.

[14] L. Peng, A. Hu, J. Zhang, Y. Jiang, J. Yu, and Y. Yan, "Design of a hybrid RF fingerprint extraction and device classification scheme," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 349 – 360, Feb. 2019.

[15] L. Yann, C. Corinna, and B. Christopher, J.C., "The MNIST database of handwritten digits," accessed on 5 Sep., 2019. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[16] S. L. Smith, P.-J. Kindermans, C. Ying, and Q. V. Le, "Don't decay the learning rate, increase the batch size," in *Proc. 6th International Conference on Learning Representations (ICLR)*, Vancouver, Canada, Apr. 2018.