Design of Noise Robust Open-Set Radio Frequency Fingerprint Identification Method

Min Wang*, Linning Peng*†, Lingnan Xie*, Junqing Zhang‡, Ming Liu§, Hua Fu*†
*School of Cyber Science and Engineering, Southeast University, Nanjing, China
†Purple Mountain Laboratories for Network and Communication Security, Nanjing, China
‡Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, United Kingdom
§Beijing Key Lab of Transportation Data Analysis and Mining, Beijing Jiaotong University, Beijing, China

***Abstract*—Radio frequency fingerprint (RFF) identification (RFFI) is a promising technique for device authentication at the physical layer of the communication stacks. However, practical challenges, particularly in low signal-to-noise ratio (SNR) scenarios, and the lack of comprehensive studies on open-set recognition hinder the widespread application of RFFI. This paper presents an unsupervised open-set RFF identification algorithm designed to address the robustness challenges associated with low SNR. Our approach integrates the Noise2Noise method for denoising, drawing inspiration from its successful applications in image and speech processing. The proposed framework utilizes an image-based autoencoder (AE) to extract features from the differential constellation trace figure (DCTF) of the signals after Noise2Noise denoising. The open-set recognition task is performed by cosine distance measurement. We carried out extensive experimental evaluation involving 18 ZigBee devices and a USRP software-defined radio platform. Our proposed method can achieve a gain up to 25% under low SNRs.**

***Index Terms*—Radio frequency fingerprint, Noise2Noise, autoencoder, open-set recognition**

I. Introduction

Radio frequency fingerprint (RFF) originates from unique hardware imperfections that occur during the manufacturing process, making it difficult to clone or tamper with [1]. It can serve as a mechanism for device access authentication, which can enhance network security at the physical layer without impacting upper-layer protocols. Numerous RFF identification (RFFI) prototypes have been developed for various wireless communication technologies, including Wi-Fi [2], ZigBee [3], LoRa [4], Bluetooth [5], and so on.

RFF is a multiple-class classification problem, hence deep learning is very suitable thanks to its excellent feature extraction capability [6]. Most of the existing deep learning-based RFFI work focuses on closed-set recognition [6], which means the devices remain the same during the training and test stage. However, it is not practical for a security protocol because there will always be rogue devices. Therefore, open-set recognition (OSR) [7] is more applicable for RFFI, which considers both known devices and unknown devices.

There have been research efforts exploring OSR in RFFI. Hanna [8] *et al.* carried out a comprehensive study on deep learning-based OSR in RFFI, including binary classifier, discriminating classifier, autoencoder (AE), and OpenMax. The work in [9] designed an RFF database to save the RFF features of the legitimate (known) devices and then use k nearest neighbour (kNN) to detect rogue (unknown) devices by comparing their features with the RFF database. However, the approach uses supervised learning to train the RFF extractor, which needs labeled data. However, in practical applications, acquiring a large volume of labeled data is often challenging. A novel classification network training methodology was introduced in [10], leveraging a combination of adversarial samples and outer samples. Although such method can fulfill the OSR task, the effectiveness of this method relies heavily on the generation of adversarial samples.

It is common in wireless communications that noise affects signal qualities. Because RFFs between different devices are very similar, noise has an even more severe impact on RFF, which requires designing noise robust RFFI approaches [11]. Shen *et al.* used integrated signals from multiple receivers and multiple packets, which achieved a 40% improvement at SNR of 10 dB [11]. Yu *et al.* designed a deep learning-based Denoising Autoencoders to extract RFF features of signals, obtaining a 14% improvement at SNR of -10 dB [12]. Wu *et al.* proposed a convolutional neural network (CNN) with a dynamic shrinkage threshold, which achieved a 20% enhancement at SNR of 0 dB [13]. Wang *et al.* proposed a signal data augmentation solution consisting of construction of sampling points and smooth filtering, which achieved a 10% enhancement at SNR of 3 dB [14]. Zhao *et al.* used the SG-filtering method in the signal processing stage to improve the SNR, achieving a classification accuracy of 93.85% at SNR of -5 dB [15].

In order to address the aforementioned challenges, this paper proposes an unsupervised and robust open-set RFFI method. The approach incorporates a denoising step using the Noise2Noise method, inspired by [16]. OSR is achieved by comparing the features of test samples with pre-saved templates, which are obtained by an unsupervised autoencoder (AE). We carried out extensive experimental evaluation using ZigBee devices and USRP software-defined radio (SDR) platform. The main contributions of this work are summarized as follows:

- We employ the Noise2Noise method for denoising to enhance the detection accuracy. This method does not rely on precise noise models; instead, it learns how to remove noise from pairs of noisy signals.
- We propose an image-based AE model as the feature

extractor to obtain RFF from the differential constellation trace figure (DCTF) of I/Q signals after denoising with Noise2Noise method. The open-set recognition task can be performed by comparing feature similarity scores with a pre-defined threshold.
- We conducted an extensive experimental evaluation using 7308 packets collected from 18 ZigBee devices in a line-of-sight (LOS) scenario. Experimental results demonstrate that the proposed approach can successfully detect unknown devices with high accuracy. The accuracy of unknown device detection reaches 0.7519 even at -5 dB SNR. Our proposed method outperforms the AE without denoising by up to 25% (achieved at 0 dB) under low SNRs (-5$\sim$15 dB).

The remainder of this paper is organized as follows: In Section II, we briefly introduce the key technologies of the proposed framework, including model formulation and model architecture. Section III elaborate on the implementation details and show the results of the proposed method at different SNRs. Finally, we conclude the paper in Section IV.

II. System Design

The proposed robust open-set RFF identification scheme is illustrated in Fig. 1. The protocol includes the training and identification stage. The key algorithms include signal processing, Noise2Noise, DCTF generation, RFF feature extractor training, and identification. These algorithms will be explained in detail.

This paper uses ZigBee/IEEE 802.15.4 as a case study. The overall architecture of the approach is applicable to other wireless technologies.

A. Signal Processing

The denoising process in Noise2Noise requires the signals compensated with frequency and phase offsets. In this paper, we employed the algorithms described in [17] to estimate and compensate for carrier frequency offset (CFO) and phase offset (PHO). Interested readers please refer to [17] for the technical details.

B. Noise2Noise for Denoising

Noise2Noise has been successful in image and speech denoising applications [18], [19]. Considering a deep neural network (DNN) with parameters, loss function $\mathcal{L}$, input x, output f_θ, and target y, the DNN learns denoising of input I/Q signals by solving the optimization problem:

$$\arg\min_{\theta} \mathbb{E}\{\mathcal{L}(f_\theta(x), y)\}, \tag{1}$$

where $\mathbb{E}(\cdot)$ denotes the expectation operator.

The Noise2Noise method is based on the following two key assumptions:

1) The noise added to the input and target is sampled from a zero-mean distribution and is uncorrelated with the input.
2) The correlation between the input noise and target noise is close to zero.

In the RFFI context, noisy I/Q signals are clean I/Q signals affected by noise. Considering a clean I/Q signal y, two noisy I/Q signals x_1 and x_2 are generated by independently sampling noise distributions n_1 and n_2. We can use $x_1 = y + n_1$ and $x_2 = y + n_2$ as the input and output of the DNN, respectively. The optimization objective can be redefined as

$$\arg\min_{\theta} \mathbb{E}\{\mathcal{L}(f_\theta(x_1), x_2)\}. \tag{2}$$

Although the optimization objective has changed, our mapping objective remains $y = f_\theta(x)$, because the noise between I/Q signals is independent. $f_\theta(x_1)$ can be represented with $\hat{y}$. For the specific form of the loss function, taking L2 loss as an example, we can represent the loss function as follows:

$$\begin{aligned} L_2(\hat{y}, y + n_2) &= \frac{1}{N}\sum_{n=1}^{N}\left[y(n) + n_2(n) - \hat{y}(n)\right]^2 \\ &= \frac{1}{N}\sum_{n=1}^{N}\{[y(n) - \hat{y}(n)]^2 + 2n_2(n)[y(n) - \hat{y}(n)] + n_2(n)^2\} \\ &= L_2(\hat{y}, y) + \sigma(n_2)^2 + \frac{2}{N}\sum_{n=1}^{N} n_2(n)[y(n) - \hat{y}(n)]. \end{aligned} \tag{3}$$

For Gaussian random noise, the second term, $\sigma(n_2)^2$, tends towards 0 because the noise is independent of the clean I/Q signals, and its mean is 0. The second term represents the noise variance, which is independent of model parameters, and thus, it does not affect the training. In summary, the uniqueness of the Noise2Noise method lies in its use of noisy inputs and noisy targets during the training phase, as opposed to the traditional use of clean training data. This approach enables the Noise2Noise method to better handle noise and imperfect data present in practical application scenarios, providing a powerful tool for the field of signal processing.

In this paper, we employed the U-Net neural network architecture for I/Q signal denoising. The U-Net architecture consists of symmetric blocks of encoders and decoders. Typically, U-Net is applied to 2D images, but in this study, we modified the structure for 1D inputs by using 1D convolutional layers. The network is portrayed in Fig. 2 and comprises 5 encoder and 5 decoder blocks. In each encoder block, two convolutional layers with a rectified linear unit (ReLu) activation function are employed. In each decoder block, three convolutional layers with a ReLu activation function are used, and except for the last layer, they are followed by a leaky ReLU activation function. In each decoder stage, the size of the feature maps is increased using ConvTranspose1d layers, while skip connections are utilized to combine features from the encoder with those from the decoder (see the grey blocks in Fig. 2), preserving important feature information. As the input data in this model is I/Q signals, the number of input and output channels is 2, representing the real and imaginary parts of I/Q signals. The number of channels used in this work are 32, 48, 64 and 96 in the bottleneck.

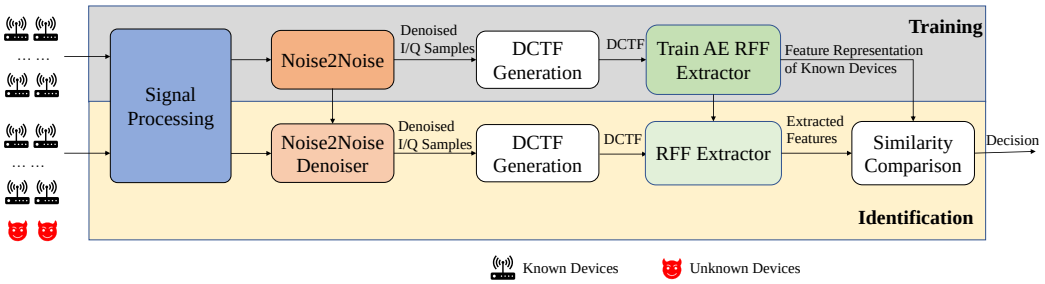


Fig. 1. The proposed open-set RFF identification scheme.

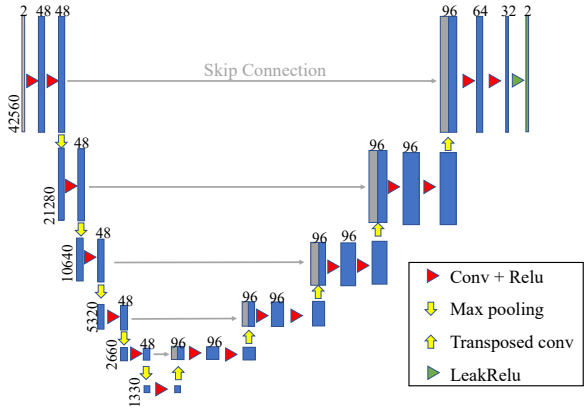


Fig. 2. U-Net architecture used in our experiments.

C. DCTF Generation

The DCTF is a specially designed feature in previous studies on RFFI [17]. Reflecting the features of RFF through statistical information, such as the contour and point distribution of DCTF, enables a more intuitive identification of ZigBee devices. Obtaining a stable DCTF does not require the estimation and compensation of carrier frequency offset, implying that preserving frequency offset features is beneficial for improving RFF identification accuracy. Therefore, we first reintroduce the frequency offset estimated by signal processing to the I/Q signals $Y[n]$ after Noise2Noise denoising, given as

$$Y'[n] = Y[n] \cdot e^{j2\pi\widehat{\Delta f} n T_s}, \tag{4}$$

where $\widehat{\Delta f}$ is the estimated CFO in Section II-A. Then, we perform a differential operation on the $Y'[n]$, given as

$$D[n] = Y'[n] \cdot Y'[n+\lambda], \tag{5}$$

where λ is the differential interval and $D[n]$ is the result which is a complex vector. $D[n]$ can then be mapped to the I/Q complex plane to generate a constellation diagram, namely DCTF. More specifically, a two-dimensional matrix Γ with a size of $[n \times n]$ ranging from $-T$ to T is defined. The coordinate (i, j) of $D[n]$ in Γ is given by

$$\begin{aligned} i &= \frac{\lceil D_I[n] + T \rceil}{2T} \cdot n, \\ j &= \frac{\lceil D_Q[n] + T \rceil}{2T} \cdot n, \end{aligned} \tag{6}$$

where $D_I[n]$ and $D_Q[n]$ are real and imaginary part of $D[n]$, respectively. Then the DCTF can be generated by the density of the sampling points in each pixel Γ.

D. AE-based RFF Extractor Training

In the training stage, we can create a training dataset with DCTF diagrams extracted from legitimate known devices. In this paper, we used an AE model, whose architecture is illustrated in Fig. 3. The encoder part includes 4 Conv2d layers and 4 MaxPool2d layers to progressively extract and reduce the dimensions of the input DCTF features. The decoder part includes 4 Conv2d layers and 4 ConvTranspose2d layers to reconstruct the DCTF.

The central fully connected layer in the AE is commonly referred to as the embedding layer, and the latent features are denoted as $z = \{z^1, ..., z^m, ..., z^M\}$ in the embedding layer. In this paper, after the training is completed, we can get one latent feature for each training sample of the known devices, denoted as $\{z_j^{kd}\}$. In addition, we also used the encoder part as the RFF extractor.

E. Open-Set RFF Identification

In the identification phase, the collected IQ samples are first processed by the same signal processing algorithms from the training stage. Then, the samples are denoised by the Noise2Noise denoiser trained in the training stage. The denoised IQ samples are converted to the DCTF diagram, which is then input to the RFF extractor. Finally, the extracted feature, z, is compared with the feature representation of known devices, $\{z_j^{kd}\}$. The cosine distance is adopted to calculate the similarity score S, defined as

$$S = \max_{1 \le j \le N} \frac{\|z\| \cdot \|z_j^{kd}\|}{z \cdot z_j^{kd}}, \tag{7}$$

where N is the number of training set samples.

The decision can be made by comparing S with a predefined threshold γ, given as

$$\hat{C} = \begin{cases} \text{Unknown device}, & \text{when } S < \gamma \\ \text{Known device}, & \text{when } S \ge \gamma \end{cases} \tag{8}$$

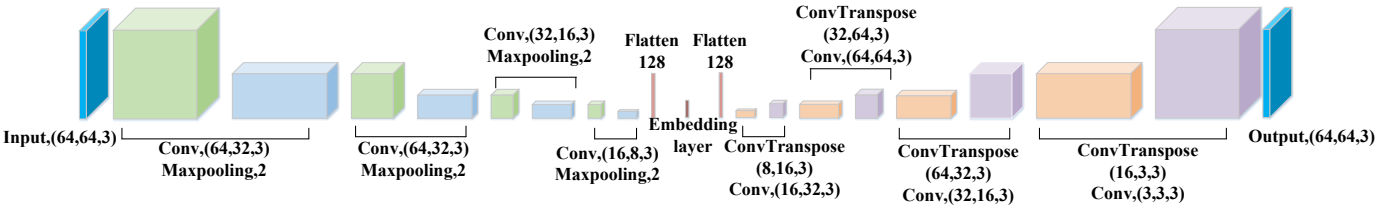


Fig. 3. AutoEncoder architecture used in our experiments.

(a) Photo of the 18 ZigBee devices.

(b) Photo of the USRP receiver.

Fig. 4. Experimental platform.

III. Experimental Evaluation

A. Experimental Setup

1) Experimental Hardware Settings: The photos of the experimental devices are shown in Fig. 4, comprising a USRP receiver and 18 ZigBee devices. We utilized an Ettus Research X310 USRP software-defined radio (SDR) platform as the receiver. The USRP X310 was configured with a center frequency of 2507 MHz and a sampling rate of 10 MSamples/s. We also used a total of 18 TI CC2530 ZigBee devices, with devices 1-12 designated as known-class devices and devices 13-18 designated as unknown-class devices. The ZigBee devices use IEEE 802.15.4 physical layer standard.

2) Neural Network Training Settings: The training process was conducted on a PC with the Ubuntu 18.04 operating system and equipped with a GeForce RTX 2080 Ti GPU. We utilized two models: a Noise2Noise network model and an AE network model. These models operate in a sequential manner, with the output of the Noise2Noise model serving as input for the AE model for further processing. For both the Noise2Noise model and the AE model, we employed the Adam optimizer with a learning rate of 0.001 and a batch size of 64. The training process involved minimizing the mean squared error (MSE) loss. We repeated the training process until the validation accuracy did not improve over ten epochs, and stored the best validation parameters for performance evaluation.

B. Datasets

The dataset used in this work comprises signals collected from ZigBee devices 1-18. Each transmission includes the same bit sequence. A total of 7308 I/Q signal frames transmitted by ZigBee devices were collected. The dataset was collected with line-of-sight between ZigBee devices and USRP X310.

To verify the performance of the proposed scheme at different SNR levels, additive white Gaussian noise (AWGN) with varying power levels, {-5, 0, 5, 10, 15, 20, 25, 30} dB, was applied to each captured signal using MATLAB. In the Noise2Noise model, the ratio of the training, validation, and test sets was 7:1:2. The training and validation sets were augmented with AWGN featuring a random SNR ranging from -5 dB to 30 dB, enabling the model to learn features across the entire SNR spectrum for denoising purposes. The validation and test sets comprise seven groups with fixed SNRs of -5 dB to 30 dB to evaluate the denoising performance of Noise2Noise. In the AE model, the data is derived from the DCTF. A five-fold cross-validation was conducted during the performance evaluation.

C. Evaluation of Denoising

The concept of Dynamic Time Warping (DTW) was introduced in the field of speech recognition to address the drawbacks arising from inflexibility along the time axis [20]. The DTW distance between two sequences $A = [a_1, a_2, \ldots, a_i, \ldots, a_m]$ and $B = [b_1, b_2, \ldots, b_j, \ldots, b_n]$ is $D(m, n)$, which is calculated employing dynamic programming to assess the following recurrence:

$$D(i,j) = \min \left\{ \begin{array}{l} D(i, j-1) \\ D(i-1, j) \\ D(i-1, j-1) \end{array} \right\} + d(a_i, b_j), \quad (9)$$

where $d(a_i, b_j)$ is the distance between the points a_i and b_j. A smaller DTW distance indicates higher similarity and better denoising performance. We used the Matlab 'dtw' function to calculate the DTW distance[1].

Table I provides the DTW distance between the original I/Q signals and the reconstructed I/Q signals using the Noise2Noise method, the smoothing-based denoising method in [14], and the SG filter-based denoising method in [15]. Regardless of the SNR, the Noise2Noise method consistently exhibits lower DTW distance values compared to the other two methods, indicating a better denoising performance. Fig. 5 and Fig. 6 exemplifies the denoising performance of Noise2Noise showcasing IQ samples and DCTF, respectively. The IQ samples denoised by the Noise2Noise method are very close to the original IQ samples. The denoised DCTF also resembles the original DCTF quite well.

[1]https://www.mathworks.com/help/signal/ref/dtw.html

TABLE I
COMPARISON OF DTW DISTANCE FOR DIFFERENT DENOISING METHODS AT VARIOUS SNR

SNR(dB)	SG-filter	Smooth	Noise2Noise
-5	1116.05	1134.48	843.83
0	847.83	883.59	544.69
5	623.99	661.35	419.52
10	454.85	487.13	258.37
15	422.68	450.77	235.73
20	415.14	440.13	228.17
25	414.15	436.96	225.38
30	414.83	435.20	225.15

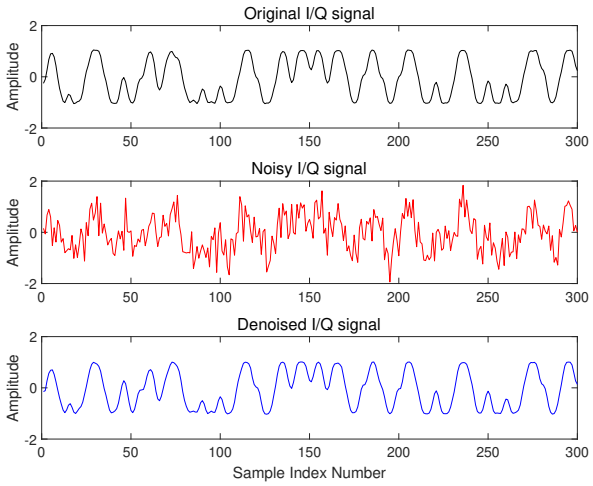
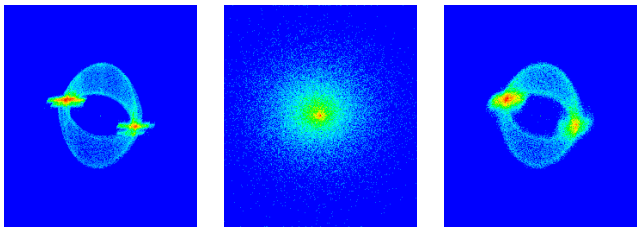


Fig. 5. Original IQ samples, 0 dB noisy IQ samples, and denoised IQ samples by Noise2Noise.



(a) Original DCTF (b) Noisy DCTF (c) Denoised DCTF

Fig. 6. DCTFs of the original signal, 0dB noisy signal, and denoised signal by Noise2Noise.

D. Evaluation of Unknown Device Detection

To gain an intuitive understanding of the detection performance for unknown devices, we used t-SNE to visualize the feature space embeddings of known and unknown devices. Fig. 7 illustrates the t-SNE visualization feature map at 0 dB SNR. Even under such low SNR, it is evident that the feature clusters of unknown devices in the test set are distinct from both the feature clusters of unknown devices in the training set and the known devices in the test set. On the other hand, the feature clusters of known devices in the training set and test set exhibit significant overlap.

The receiver operating characteristic (ROC) curve is often used to evaluate unknown device detection. The AUC quantifies the discriminative power of a classification model by measuring its ability to distinguish between positive and negative instances across various decision thresholds. Fig. 8 depicts the ROC curve at different SNRs. It can be seen that even at -5dB SNR, the AUC remained high, reaching 0.9635, which indicates the robustness of our approach.

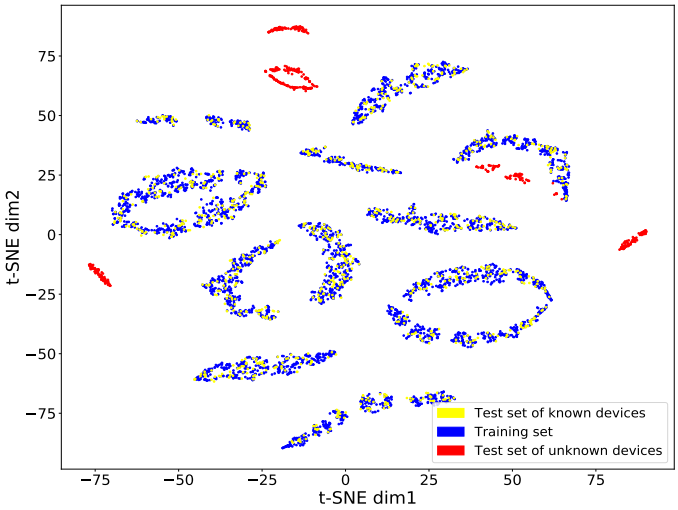


Fig. 7. t-SNE visualization of features at 0 dB SNR.

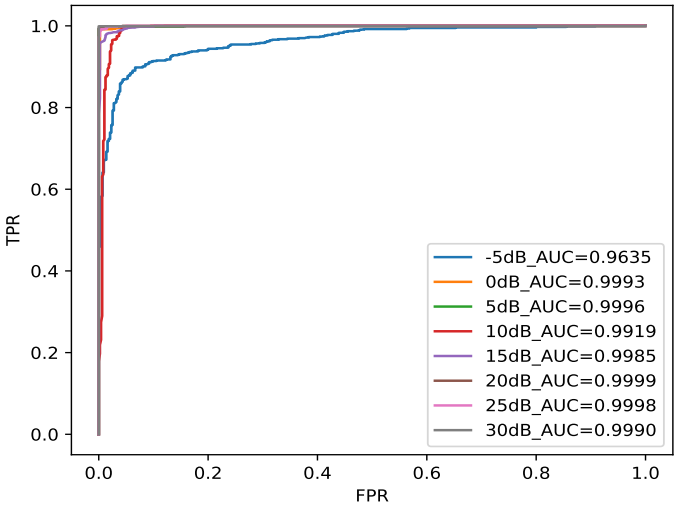


Fig. 8. ROC curves of unknown device detection at different SNR levels.

While ROC and AUC describe the overall detection performance with all the possible thresholds, the detection accuracy is also a key metric given a particular threshold. After selecting a threshold of 0.9, we can calculate True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). The detection accuracy can then be defined as

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{10}$$

The comparison of unknown device detection performance under different SNRs, including the DCTF trained with the AE without denoising (Benchmark) and the AE with Noise2Noise,

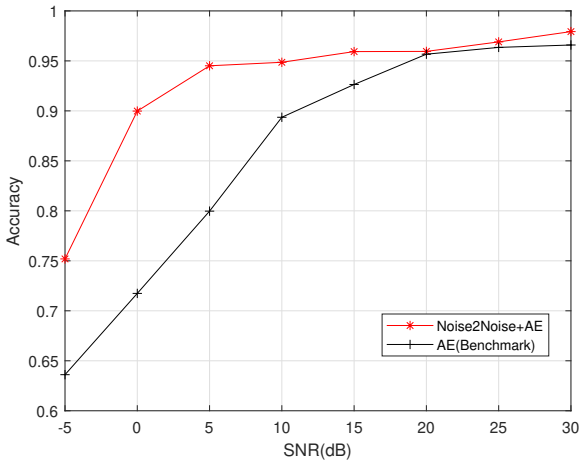


Fig. 9. Unknown device detection performance comparison at different SNR levels. The threshold is 0.9.

is presented in Fig. 9. The model with the highest accuracy was selected as the best model in the five-fold cross-validation for each method. It can be obtained from Fig. 9 that even at -5 dB SNR, the accuracy remained robust, reaching 0.7519, which indicates the robustness of our detection methodology. With the increase in SNR, the accuracy also rises. In the range of 0 to 30 dB, the accuracy fluctuates between 0.89 and 0.97. In addition, our proposed approach outperforms the benchmark up to 25% (achieved at 0 dB), i.e., $\frac{90\%-72\%}{72\%}$, at low SNRs under AWGN channels.

IV. Conclusion

In this paper, we proposed a robust open-set RFF identification scheme that incorporates denoising using the Noise2Noise method and unsupervised feature extraction based on the AE. Firstly, the Noise2Noise approach denoised the IQ samples, which can significantly improve the signal quality. The IQ samples were then converted to DCTF diagrams. The RFF features were extracted from DCTF by unsupervised AE models. In the inference stage, RFF features of the known devices and test devices were compared with their similarity and a decision can be made based on their cosine similarity. Our experiments demonstrate that the proposed scheme exhibits excellent recognition performance. Specifically, the proposed method achieves robust accuracy in unknown device detection, reaching 0.7519 even at -5 dB SNR. Within the SNR range of 0 to 30 dB, the accuracy consistently hovered around 0.89 to 0.97 and outperformed the benchmark by up to 25% at low SNRs (-5∼15 dB).

Acknowledgment

This work was supported in part by the National Key Research and Development Program of China (2022YFB4300300), National Natural Science Foundation of China under Grant 62171120, 62001106, 61971029, and 62221001, Jiangsu Provincial Key Laboratory of Network and Information Security No. BM2003201, Guangdong Key Research and Development Program under Grant 2020B0303010001, Chongqing Natural Science Joint Fund Project under Grant No. CSTB2023NSCQ-LZX0121, and Purple Mountain Laboratories for Network and Communication Security. The work of J. Zhang was in part supported by the UK EPSRC New Investigator Award under grant ID EP/V027697/1.

References

[1] L. Xie, L. Peng, J. Zhang, and A. Hu, "Radio frequency fingerprint identification for Internet of things: A survey," *Security and Safety*, vol. 3, p. 2023022, 2024.

[2] W. Nie, Z.-C. Han, M. Zhou, L.-B. Xie, and Q. Jiang, "UAV detection and identification based on WiFi signal and RF fingerprint," *IEEE Sensors J.*, vol. 21, no. 12, pp. 13 540–13 550, 2021.

[3] L. Peng, J. Zhang, M. Liu, and A. Hu, "Deep learning based RF fingerprint identification using differential constellation trace figure," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 1091–1095, 2020.

[4] G. Shen, J. Zhang, A. Marshall, L. Peng, and X. Wang, "Radio frequency fingerprint identification for LoRa using deep learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2604–2616, 2021.

[5] A. Jagannath, Z. Kane, and J. Jagannath, "RF fingerprinting needs attention: Multi-task approach for real-world WiFi and Bluetooth," in *Proc. IEEE Global Communications Conference*, 2022, pp. 4607–4612.

[6] G. Shen, J. Zhang, and A. Marshall, "Deep learning-powered radio frequency fingerprint identification: Methodology and case study," *IEEE Commun. Mag.*, vol. 61, no. 9, pp. 170 – 176, 2023.

[7] C. Geng, S.-j. Huang, and S. Chen, "Recent advances in open set recognition: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3614–3631, 2020.

[8] S. Hanna, S. Karunaratne, and D. Cabric, "Open set wireless transmitter authorization: Deep learning approaches and dataset considerations," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 59–72, 2021.

[9] G. Shen, J. Zhang, A. Marshall, and J. R. Cavallaro, "Towards scalable and channel-robust radio frequency fingerprint identification for LoRa," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 774–787, 2022.

[10] Y. Xu, X. Qin, X. Xu, and J. Chen, "Open-set interference signal recognition using boundary samples: A hybrid approach," in *2020 International Conference on Wireless Communications and Signal Processing (WCSP)*, 2020, pp. 269–274.

[11] G. Shen, J. Zhang, A. Marshall, R. Woods, J. Cavallaro, and L. Chen, "Towards receiver-agnostic and collaborative radio frequency fingerprint identification," *IEEE Trans. Mobile Comput.*, pp. 1–17, 2023.

[12] J. Yu, A. Hu, F. Zhou, Y. Xing, Y. Yu, G. Li, and L. Peng, "Radio frequency fingerprint identification based on denoising autoencoders," in *Proc. Int. Conf. on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2019, pp. 1–6.

[13] W. Wu, S. Hu, D. Lin, and Z. Liu, "DSLN: Securing internet of things through RF fingerprint recognition in low-snr settings," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3838–3849, 2022.

[14] W. Wang and L. Gan, "Radio frequency fingerprinting improved by statistical noise reduction," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 8, no. 3, pp. 1444–1452, 2022.

[15] X. Zhao, L. Wang, Q. Wang, and J. Wang, "A hierarchical framework for drone identification based on radio frequency machine learning," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, 2022, pp. 391–396.

[16] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning image restoration without clean data," *arXiv preprint arXiv:1803.04189*, 2018.

[17] 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, 2019.

[18] M. M. Kashyap, A. Tambwekar, K. Manohara, and S. Natarajan, "Speech denoising without clean training data: A noise2noise approach," *arXiv preprint arXiv:2104.03838*, 2021.

[19] Y. Mansour and R. Heckel, "Zero-shot noise2noise: Efficient image denoising without any data," in *Proc. IEEE/CVF Conf Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 14 018–14 027.

[20] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Proc. 3rd Int. Conf. Knowledge Discovery and Data Mining*, 1994, pp. 359–370.