A RF Fingerprint Clustering Method Based on Automatic Feature Extractor

Pinhong Xiao, Di Lin(✉), and Mengjuan Wang

University of Electronic Science and Technology of China, Chengdu, Sichuan, China
202022090634@std.uestc.edu.cn, lindi@uestc.edu.cn

Abstract. RF fingerprint technology has received extensive attention and research in recent years due to its immutable nature. RF fingerprinting technology can be used as a wireless network security mechanism alone or combined with existing security mechanisms to enhance wireless network security. The early RF fingerprint research widely used the method of artificial feature extraction, but this method relies too much on expert experience. This paper proposes a semi-supervised learning approach for RF fingerprint recognition. This work directly uses the original I/Q sequence data, designs a fingerprint extractor based on the convolutional neural network (CNN), and uses K-means and DBSCAN algorithms to cluster the fingerprints. The experimental results demonstrate that after training with a small amount of labeled data, the fingerprint extractor can effectively extract features of unknown signals, and these features can well allow unknown similar devices to be clustered together by the clustering algorithm.

Keywords: RF fingerprint identification · Semi-supervised Learning · Feature extraction

1 Introduction

RF fingerprints are widely present in wireless communication equipment. Just like everyone has different fingerprints, each wireless device also has different RF fingerprints. RF fingerprints arise from hardware differences within the device, including manufacturing tolerances generated during the production of electronic components and drift tolerances generated during use [1], and are both unique and short-time invariant. By analyzing the received RF signal, various features of the device can be extracted, and the multi-dimensional features together constitute the RF fingerprint library of the device. This method of extracting device hardware features based on communication signals is called RF fingerprint extraction, and the method of using RF fingerprints to identify different wireless devices is called RF fingerprint identification. The traditional wireless network protection scheme is usually a security protocol based on a password mechanism, but the security protocol may have flaws, and attackers can crack the authentication mechanism by cracking passwords, statistical analysis, and other methods. RF fingerprint extraction and identification both work at the physical layer and cannot be tampered with. It can

X. Jiang (Ed.): MLICOM 2022, LNICST 481, pp. 108–116, 2023.
https://doi.org/10.1007/978-3-031-30237-4_10

be used as an authentication mechanism alone or combined with traditional security mechanisms to bring higher wireless network security performance.

Traditional RF fingerprinting methods rely heavily on expert experience, which requires expert experience and expertise to analyze and feature the signal, and then input the features into the classifier for classification. With the rise of deep learning, deep learning-based RF fingerprinting methods have been widely studied. Literature [2] generates device fingerprint data through MATLAB simulation and proposes a RF fingerprint recognition method based on CNN model. It uses the original IQ signal to input the CNN model without analyzing or extracting signal features, and has achieved good results, greatly reducing the dependence on expert experience.

In the field of RF fingerprinting, in addition to supervised machine learning and deep learning research, a small amount of unsupervised learning research has also appeared. In contrast to supervised learning, unsupervised RF fingerprinting techniques do not require training data registered in advance. This approach must find the implied structure in the signal and cluster similar signals together then map them to a device. In [3], an infinite Gaussian mixture model are used to detect the number of devices and classify them based on two features: frequency bias and phase shift deviation. At present, the unsupervised RF fingerprint identification method still needs further research.

In this paper, we propose a semi-supervised RF fingerprint recognition method based on CNN and DBSCAN algorithm. This method requires only a small amount of labeled data to train the CNN for signal feature extraction, and then unsupervised clustering can be performed on completely unknown data. Experiments show that the method can effectively extract signal features and achieve good clustering results in unknown signals.

2 Related Work

2.1 Supervised Learning

Methods Based on Manual Feature Selection: The manually selected features mainly include many parameters with actual physical meaning and statistical features after Fourier transform, Hilbert transform, and other transformations on the target signal segment. According to the different target signal types, the selected features are also different. Generally, the RF fingerprint features are divided into two types: RF fingerprint features based on transient signals and RF fingerprint features based on steady-state signals. The features extracted based on transient signals mainly include transient signal duration, spectral features, wavelet domain features, fractal dimension, envelope features, etc. Early studies almost all revolve around transient signal RF fingerprinting techniques. With the development of technology, almost all digital communication systems include a leading sequence before the data segment to simplify the design of the receiver [4]. Stable leads provide a stable identifiable steady-state signal, so the focus of research in this area is beginning to shift to steady-state signals. After feature selection and feature extraction, the signal features are input into the machine learning classification model or deep learning classification model for supervised training, and then the trained classification model is obtained. Reference [5] extracts the parametric features such as information dimension, constellation features, and phase noise spectrum in RF

signals and classified them by machine learning models such as BaggedTree and Fine-GaussianSVM. Reference [6] uses spectral correlation function to extract signal features and proposed a deep confidence network to classify signals. In the method of manual feature selection, both the selection of features and the construction of classifiers have a great impact on the final experimental results. However, feature selection often relies on expert experience and has defects such as poor feature generalization ability. It often performs poorly in the face of a large number of devices or in low signal-to-noise ratio environments.

Methods Based on Raw I/Q Data: This method uses the target I/Q signal directly as model input. In earlier studies, the original signal was used directly as a device fingerprint for identification and authentication, using different methods of similarity measures, including Euclidean distance, Marxian distance, cosine similarity, Pearson correlation coefficient, etc. [7]. However, such fingerprints contain too much redundant information and dimension, which leads to insufficient recognition efficiency and dimension disaster. Moreover, such methods are easily affected by the environment, and their performance fluctuates greatly with noise. In recent years, with the development of deep learning technology, compared with traditional machine learning technology, the characteristics of deep learning automatic feature extraction have natural advantages in the face of complex original signals. Kevin Merchant [8] et al. built a CNN model for automatic feature extraction, using the time-domain complex baseband error signal to train the CNN. Tong Jian [9] et al. also used the CNN model to perform RF fingerprinting on the processed I/Q signals, and also considered some feature engineering to reduce the influence of the channel in the case of low signal-to-noise ratio and low computing power.

2.2 Semi-supervised Learning and Unsupervised Learning

Supervised RF fingerprint technology has proven its effectiveness in a large number of literature. However, in the actual environment, it is often infeasible to register the fingerprint database in advance. At present, there is still relatively little research on semi-supervised and unsupervised RF fingerprint recognition technology. Reference [3] proposes a non-parametric Bayesian clustering method to detect the number of devices, selected frequency difference and phase shift difference as features, and performed unsupervised passive classification of multiple devices. When the environment contains 4 devices, this method achieves high accuracy. Reference [10] applies unsupervised and semi-supervised methods to the field of radio modulation classification. They first constructed a convolutional self-encoder for the unsupervised extraction of modulation features of radio signals, while a CNN was used for the supervised feature extraction. Finally, the extracted features are clustered using a clustering algorithm.

3 Approach

3.1 Dataset

In this work, multiple transmitters and one receiver are simulated using MATLAB, and the RF fingerprint dataset is generated by simulation. The AWGN channel is simulated

between the transmitter and the receiver. The signal is modulated by QPSK, the signal-to-noise ratio is 20 db, the sampling rate is 20 Mbps, the sampling time interval is 0.05 us, and the number of sub-carriers is 52, including 4 pilot signals and 48 data signals. We generate the RF fingerprint of the device by simulating conditions such as DC offset, phase offset, I/Q imbalance, etc. In recent years, there are also studies on the application of CNN in the field of I/Q signal processing. In [2], the I/Q signal is directly input into the CNN for device classification. Reference [10] uses CNN to extract the features of raw I/Q data, then carry out the modulation classification of the signal.

3.2 CNN Model

Compared with ordinary feedforward neural networks, CNNs are locally connected and have shared parameters. CNNs perform mathematical convolution operations locally on the input data through convolution kernels, enabling them to discover meaningful features of the input data and making the amount of computation in the network reduced. CNNs have been widely used in the field of image processing and image recognition with great success. CNNs have also been applied to sequence data such as natural language processing and audio processing, and achieved remarkable results [11]. In recent years, there are also studies on the application of CNN in the field of I/Q signal processing. In [2], raw I/Q data is directly input into the CNN model for device classification. Reference [10] uses CNN to extract the features of the I/Q signal, then carry out the modulation classification of the signal. The architecture c the CNN feature extractor designed in this paper is shown in Fig. 1. Our network has a total of five layers, including two convolutional layers, two fully connected layers, and an output layer. Our simulation produces the raw I/Q sequence, containing both real and imaginary parts, and we treat it as $2 \times N$ dimensional vector. As shown in Fig. 2, inspired by [2], we divide the I/Q sequence in the form of a sliding window in order to enable the model to detect the damage at any position on the I/Q sequence. The size of the window is 128, the step size of each slide is 1, and each input sample of the model is a 2×128 dimensional vector. The input sample is first processed in two convolutional layers. In order to fully extract the sample features, we use a 2×5 convolutional kernel and a 1×5 convolutional kernel for convolution in the first and second convolutional layers. Between the second convolutional layer and the fully connected layer, we dropout at a ratio of 0.5 to control overfitting. The first fully connected layer has 128 neurons and the second fully connected layer has 28 neurons. The output of the second fully connected layer will be used to input the Softmax layer to train the CNN model. When the CNN training is completed, we will use the output of the second fully connected layer as the feature of the I/Q sample, which is the RF fingerprint. This feature has dimension 28 and it will be fed into the clustering model for unsupervised clustering.

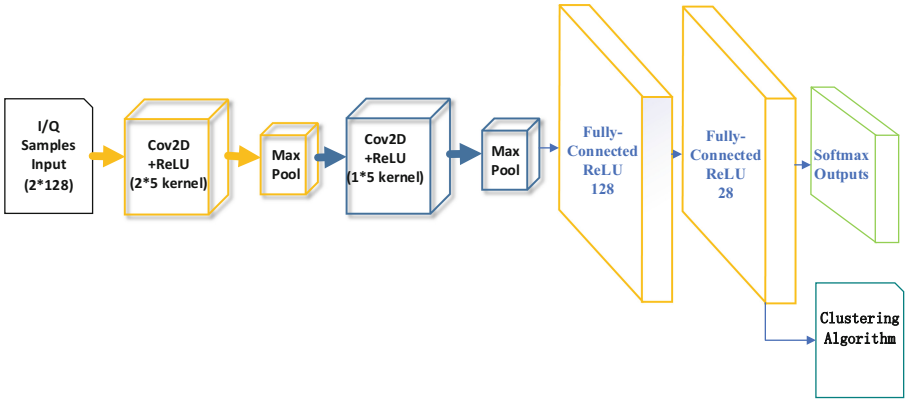

Fig. 1. The CNN model designed to extract features of raw I/Q in this work.

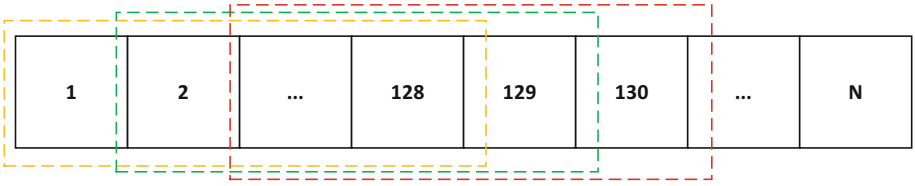

Fig. 2. An example of sliding operation using a window of length 128 over I/Q sequences.

3.3 Clustering Algorithm

After extracting the RF fingerprints of the signals, we use unsupervised clustering algorithms to cluster similar fingerprints to the same transmitting device. The dimensionality of the features extracted by CNN is 28. To improve the clustering effect, we firstly use PCA to reduce the dimensionality of the features. After experiments, we find that when the dimension is reduced to 7, more than 90% of the variance of the original fingerprint can be retained.

Once the CNN model is trained and deployed, we no longer need any prior information and can perform RF fingerprinting in a completely new unknown environment. We consider both an environment with a known number of devices and a totally blind environment with an unknown number of devices. In an environment where the number of devices is known, we use the K-means algorithm for clustering. In a completely unknown environment, we use the DBSCAN algorithm for clustering, because the DBSCAN algorithm does not require information about the number of clusters, and it can achieve better results in irregular shape data. When the number of devices is small, the DBSCAN algorithm shows good performance. However, as the number of devices increases, the clustering performance of the DBSCAN algorithm shows a more significant decrease.

4 Experimental and Results Analysis

In this work, we use MATLAB to simulate and generate RF fingerprints of 30 devices, with about 300,000 pieces of data. We use 2, 5, 8, and 10 devices to train the CNN feature extraction model, taking 15% of the data as the validation set and 15% of the data as the test set. The test accuracy is shown in Table 1. When the number of devices is less than 10, the test accuracy of our model has reached more than 98%, which shows that the feature extraction model can effectively extract the fingerprint features of the device.

Table 1. The Testing Accuracies of CNN feature extractors

Number of devices	Testing Accuracy
2 devices	1.00
5 devices	0.990
8 devices	0.989
10 devices	0.988

In addition to the above 10 devices, we used different sampling rates and random RF fingerprint parameters to generate 20 brand new devices. It is worth noting that these devices are completely unknown to our feature extraction model, which can reflect the feature extraction ability of the model for unknown signals. We use K-means and DBSCAN algorithms to cluster the signals respectively, and use normalized mutual information(NMI) to measure the similarity of the clustering results. The experimental results are shown in Fig. 3 and Table 2. We can see that the K-means algorithm has relatively stable performance when the number of devices is known. Within 10 devices, the NMI index of K-means can reach more than 0.95. In the environment of 10 to 20 devices, with the increase of the number of devices, the performance of K-means does not decrease significantly, and the NMI index remains at 0.9. This proves that the CNN feature extraction model can effectively extract signal features. When the number of devices is small, the DBSCAN algorithm shows good performance. However, as the number of devices increases, the clustering performance of the DBSCAN algorithm decreases significantly. Considering the performance of K-means, this performance drop may be caused by some limitations of the DBSCAN algorithm. The DBSCAN algorithm is highly sensitive to the domain threshold (Eps) and the point threshold (MinPts), which may need to be dynamically adjusted as the number of devices changes [12].

As described in Sect. 3, the feature dimension extracted by CNN is 28, and we use PCA to reduce the feature dimension to 3. For visualization in two-dimensional space, we use the t-SNE algorithm to map the features to the two-dimensional space. When the number of devices is 10, the clustering results using K-means algorithm and DBSCAN algorithm are shown in Fig. 4 and Fig. 5. We can see that the DBSCAN algorithm does not discover all device classes.

Table 2. The normalized mutual information score using K-means and DBSCAN algorithm

Number of devices	NMI using K-means	NMI using DBSCAN
2 devices	1.00	1.00
3 devices	1.00	1.00
5 devices	0.956	0.906
10 devices	0.968	0.909
15 devices	0.891	0.651
20 devices	0.899	0.526

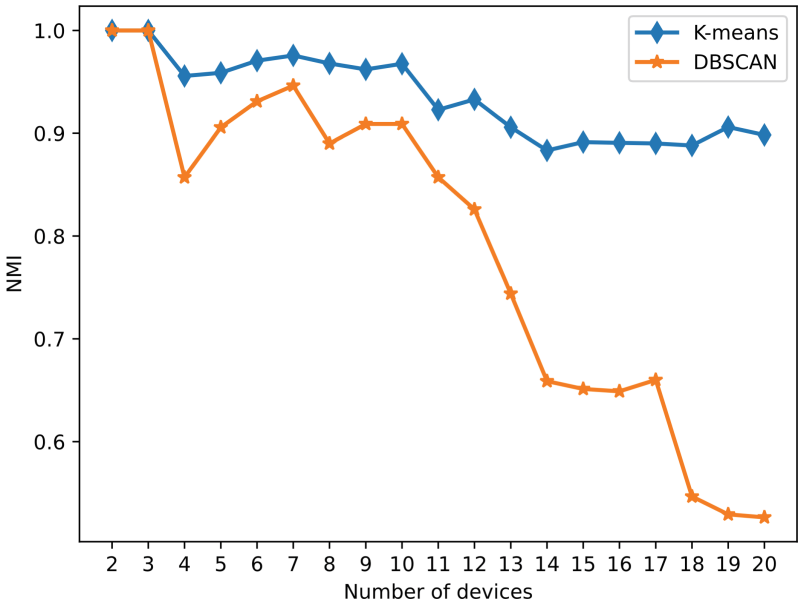

Fig. 3. The normalized mutual information score using K-means and DBSCAN algorithm.

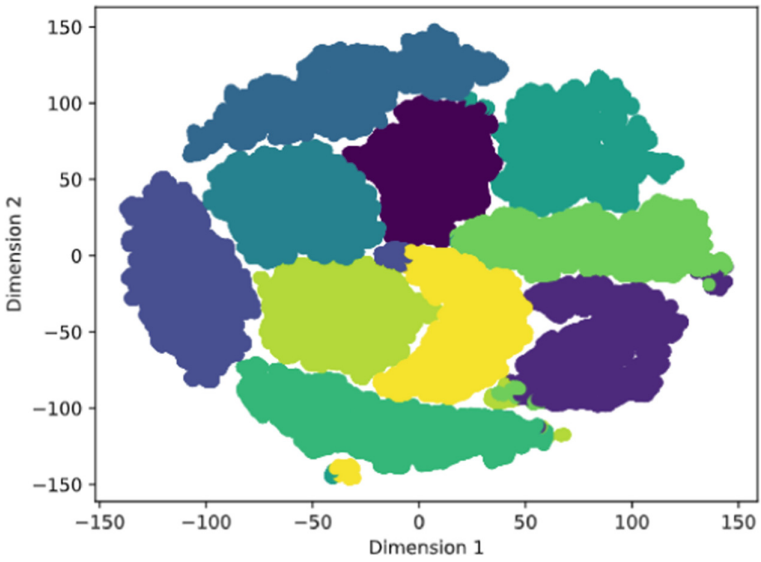

Fig. 4. Visualization of K-means clustering results using t-SNE

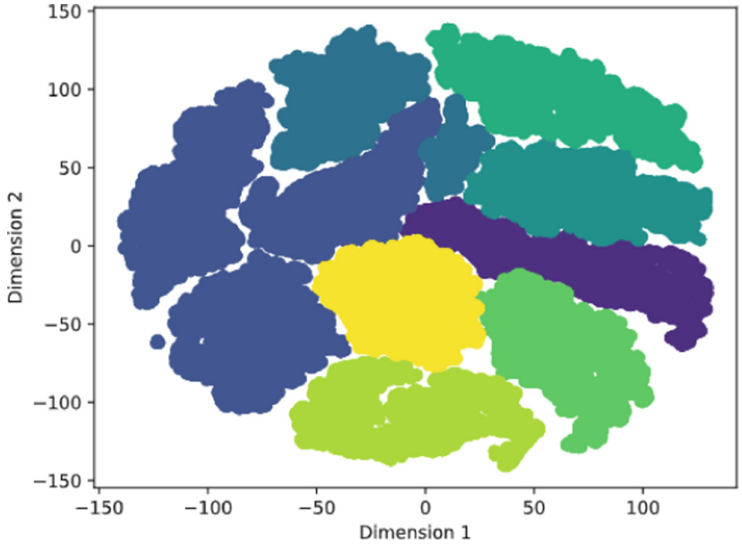

Fig. 5. Visualization of DBSCAN clustering results using t-SNE

5 Conclusion

This paper applies a semi-supervised method to RF fingerprinting, constructs a CNN feature extraction model, and uses only a small amount of data to train the model, avoiding the dependence on expert experience. After the deployment of CNN model, automatic

feature extraction can be performed. After PCA dimensionality reduction of the features, we use K-means algorithm and DBSCAN algorithm for clustering respectively. The K-means algorithm has achieved stable and good results when the number of devices is known, while the DBSCAN algorithm shows some degradation in performance when facing more devices. The possible reason is that the Eps and MinPts we chose are solidified and not dynamically adjusted to accommodate multiple devices.

In future work, we will make some improvements to the method of this paper. Firstly, we will conduct experiments on low SNR environments and Rayleigh fading channels to improve the noise immunity of our model. Besides, we will improve the clustering effect by optimizing the DBSCAN algorithm, or choose other more suitable clustering methods.

References

1. Ureten, O., Serinken, N.: Detection of radio transmitter turn-on transients. Electronics Letters 35(23), 23 (1999)
2. Riyaz, S., Sankhe, K., Ioannidis, S., Chowdhury, K.: Deep learning convolutional neural networks for radio identification. IEEE Commun. Mag. **56**(9), 146–152 (2018)
3. Nguyen, N.T., Zheng, G., Han, Z., Zheng, R.: Device fingerprinting to enhance wireless security using nonparametric Bayesian method. Proc. IEEE INFOCOM **2011**, 1404–1412 (2011)
4. Scanlon, P., Kennedy, I.O., Liu, Y.: Feature extraction approaches to RF fingerprinting for device identification in femtocells. Bell Labs Tech. J. **15**(3), 141–151 (2010)
5. Hu, S., Wang, P., Peng, Y., et al.: Machine learning for RF fingerprinting extraction and identification of soft-defined radio devices. In: The International Conference on Artificial Intelligence in China (2019)
6. Mendis, G.J., Wei-Kocsis, J., Madanayake, A.: Deep learning based radio-signal identification with hardware design. IEEE Trans. Aeros. Electron. Syst. **55**(5), 2516–2531 (2019)
7. Langley, L.E.: Specific emitter identification (SEI) and classical parameter fusion technology. In: Proceedings of WESCON 1993, pp. 377–381 (1993)
8. Merchant, K., Revay, S., Stantchev, G., Nousain, B.: Deep learning for RF device fingerprinting in cognitive communication networks. IEEE J. Sel. Topics Signal Process. **12**(1), 160–167 (2018)
9. Jian, T., et al.: Deep learning for RF fingerprinting: a massive experimental study. IEEE Internet Things Mag. **3**(1), 50–57 (2020)
10. O'Shea, T.J., West, N., Vondal, M., Clancy, T.C.: Semi-supervised radio signal identification. In: 2017 19th International Conference on Advanced Communications and Technology (ICACT), pp. 33–38 (2017)
11. Cui, R., Liu, H., Zhang, C.: Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1610–1618 (2017)
12. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: AAAI Press, pp. 226–231 (1996)