

# Supervised Contrastive Learning for RFF Identification With Limited Samples

Yang Peng<sup>1</sup>, Changbo Hou<sup>2</sup>, Yibin Zhang<sup>3</sup>, Yun Lin<sup>4</sup>, *Member, IEEE*, Guan Gui<sup>5</sup>, *Senior Member, IEEE*, Haris Gacanin<sup>6</sup>, *Fellow, IEEE*, Shiwen Mao<sup>7</sup>, *Fellow, IEEE*, and Fumiyuki Adachi<sup>8</sup>, *Life Fellow, IEEE*

**Abstract**—Radio-frequency fingerprint (RFF), which comes from the imperfect hardware, is a potential feature to ensure the security of communication. With the development of deep learning (DL), DL-based RFF identification methods have made excellent and promising achievements. However, on one hand, existing DL-based methods require a large amount of samples for model training. On the other hand, the RFF identification method is generally less effective with limited amount of samples, while the auxiliary data set and the target data set often needs to have similar data distribution. To address the data-hungry problems in the absence of auxiliary data sets, in this article, we propose a supervised contrastive learning (SCL)-based RFF identification method using data augmentation and virtual adversarial training (VAT), which is called “SCACNN.” First, we analyze the causes of RFF, and model the RFF identification problem with augmented data set. A nonauxiliary data augmentation method is proposed to acquire an extended data set, which consists of rotation, flipping, adding Gaussian noise, and shifting. Second, a novel similarity radio-frequency fingerprinting encoder (SimRFE) is used to map the RFF signal to the feature coding space, which is based on the convolution, long short-term-memory, and a fully connected deep neural network (CLDNN). Finally, several secondary classifiers are employed to identify the RFF feature coding. The simulation results show that the proposed SCACNN has a greater identification ratio than the other classical RFF identification methods. Moreover, the identification ratio of the proposed SCACNN achieves an accuracy of 92.68% with only 5% samples.

**Index Terms**—Data augmentation, deep learning (DL), physical layer security, radio-frequency fingerprint (RFF), supervised contrastive loss, virtual adversarial training (VAT).

Manuscript received 4 August 2022; revised 4 January 2023 and 6 March 2023; accepted 28 April 2023. Date of publication 3 May 2023; date of current version 25 September 2023. This work was supported by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYCX21-0736. (*Corresponding author: Guan Gui.*)

Yang Peng, Yibin Zhang, and Guan Gui are with the College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail: 2020010210@njupt.edu.cn; 2021010208@njupt.edu.cn; guiguan@njupt.edu.cn).

Changbo Hou and Yun Lin are with the College of Information and Communication Engineering, Harbin Engineering University, Harbin 150000, China (e-mail: houchangbo@hrbeu.edu.cn; linyun@hrbeu.edu.cn).

Haris Gacanin is with the Institute for Communication Technologies and Embedded Systems, RWTH Aachen University, 52062 Aachen, Germany (e-mail: harisg@dsp.rwth-aachen.de).

Shiwen Mao is with the Department of Electrical and Computer Engineering, Auburn University, Auburn AL 36849 USA (e-mail: smao@ieee.org).

Fumiyuki Adachi is with the International Research Institute of Disaster Science, Tohoku University, Sendai 9800845, Japan (e-mail: adachi@ecei.tohoku.ac.jp).

Digital Object Identifier 10.1109/JIOT.2023.3272628

## I. INTRODUCTION

WITH the development of wireless communication technology, the Internet of Things (IoT) has been widely used in numerous fields, such as smart home, intelligent driving, and intelligent transportation [1], [2]. In addition, the Internet of Everything (IoE) represents an inevitable trend of current industries. Moreover, the wide deployment of the fifth-generation (5G) wireless communications has brought the benefits of low latency and high speed, which will further promote the development of these industries. However, the rapid increase of communication and IoT devices has also brought about many security issue [3], [4], [5], [6]. Traditional authentication and identification techniques are usually based on the cryptography, which may be at risk in facing malicious users with huge computing resources [7]. Therefore, an effective authentication and identification method for IoT devices is needed.

In recent years, great achievements have been made in device authentication and identification based on the physical layer features. The specific emitter identification (SEI) is a technology that relies on the characters of radio-frequency (RF) signals [8], [9]. The RF fingerprinting (RFF) identification is an effective method in SEI. Specifically, the difference in RF signals can be considered as fingerprints of the RF signals with the ability to uniquely and stably represent the emitters [10], [11]. In addition, RFF comes from the imperfection of hardware, which is inevitable in the manufacturing process [13]. RFF identification consists of data preprocessing, feature extraction, and identification. The data preprocessing usually contains power normalization, synchronization, and data cleaning. It aims at normalizing the RF signal and mitigating the impact of nonstandard RF signals on feature extraction.

As mentioned above, the difference of RF signals are caused by the hardware manufacturing process, which are usually very small to distinguish. Therefore, the feature extraction is the principal step of RFF identification, which is employed to reduce the dimension of RF signal and “amplify” the difference. Traditional feature extraction of RFF identification usually employ statistical processing technologies. Then, classic machine learning methods, such as support vector machine (SVM), multilayer perception (MLP), and linear Bayesian classifier, are employed for the identification [14], [15], [16]. Most of the traditional RFF identification methods can achieve a near perfect performance. However, the signal

processing-based feature extraction is usually based on the prior knowledge of the devices., which is device specific and cannot be generalized to another signal. Second, traditional feature extraction usually requires manual feature extraction with expert knowledge, which is impractical in dynamic communication scenarios.

Deep learning (DL) has achieved a great performance in computer vision (CV), natural language processing (NLP), and intelligent communication. Furthermore, DL shows excellent abilities in data mining and feature extraction, which has been employed in channel state information (CSI) feedback [17], [18], resource allocation [20], beamforming [21], [22], and automatic modulation classification (AMC) [23], [24], [25], [26]. Meanwhile, the DL-based RFF identification method performs feature extraction and identification at the same time. It can also obtain more robust and effective RFF features of RF signals using a neural network, when compared with traditional methods [27], [28], [29], [30], [31], [32], [33], [34]. However, the neural network is a data-driven model, which depends on the availability of historical data sets. The data sets used in most DL-based RFF identification methods are capable of supporting neural network fitting. Nevertheless, in the real world scenarios, a large amount of labeled data may not be available, which leads to a sharp degradation of the performance of DL-based RFF identification methods. Although this problem can be solved with few-shot learning, it still requires a massive auxiliary data set with a similar data distribution to get optimum weights for the neural network. In this article, we focus on RFF identification with nonauxiliary, limited samples, which is a limited data set and unable to support the neural network fitting.

Compared with the softmax-based RFF identification methods, the metric-learning-based methods have better performance with limited samples. Meanwhile, the supervised contrastive learning is the improved form of metric learning. In this article, a supervised contrastive-learning-based RFF identification method is proposed, which contains data augmentation, a similarity RFF encoder (SimRFE), and a secondary classifier. Specifically, data augmentation uses prior knowledge to extend the raw data set with simple mathematical operations, which does not require extra auxiliary data sets. Then, unlike the traditional softmax-based RFF identification method, we use a SimRFE to map the original RF signal to a high-dimensional feature space and constrain the feature coding with a supervised contrastive loss (SCL). Finally, several secondary classifiers based on neural network, traditional machine learning, and distance similarity measurement are used to identify the RFF feature coding. The main contribution of this article is summarized as follows.

- 1) Different feature embedding methods are employed to improve the performance of RFF identification with limited samples, including a data augmentation method and a model-based virtual adversarial training (VAT) method.
- 2) We propose an SCL-based RFF identification method, which has a better feature clustering performance. The optimal hyperparameters of the proposed method are also provided to balance the training cost with the identification performance.

- 3) Several typical machine-learning-based secondary classifiers are employed for identification of RFF feature codings. A simple distance similarity measurement-based classifier is also proposed for the identification with a simple mathematical operation.

The remainder of this article is organized as follows. In Section II, we introduce related RFF identification methods and DL with limited samples. The RFF signal is modeled and the RFF identification problem is described in Section III. In Section IV, we present the proposed SCL-based RFF identification methods. Various simulation results are provided to show the performance of our proposed system in Section V. Finally, we conclude our work in Section VI.

## II. RELATED WORKS

In this section, we introduce the traditional RFF identification methods with artificial RFF feature coding, the DL-based RFF identification methods, and three feasible schemes for DL-based RFF identification with limited samples.

### A. RFF Identification Methods

1) *Traditional RFF Identification Methods:* Traditional RFF identification methods usually consist of feature extraction and identification. The feature extraction aims at acquiring artificial RFF features via prior knowledge, which also reduces the dimension of the high-dimensional raw RF samples. Then, the artificial RFF features are identified with a traditional machine learning method. Cobb et al. [14] sliced the RF signal into slices, and obtained the instantaneous amplitude, phase, and frequency of RF signals slices as characteristics according to the definition of I/Q signals. Then, the statistical characteristics, including standard deviation, variance, kurtosis, and skewness of those three instantaneous characteristics, were used to represent the RFF features. Multiple discriminant analysis (MDA) and a linear Bayesian classifier were employed to reduce the dimension of RFF features and identify them, respectively. Tu et al. [16] employed conventional signal processing, such as dual tree complex wavelet transform (DT-CWT), short-time fourier transform (STFT), and Wigner-Ville distribution (WVD), to extract the RFF features. Robust principle component analysis (RPCA) and SVM were employed. Furthermore, constellation error [35], [36], IQ imbalance [37], and modulation shape [38] have all been utilized for RFF identification.

2) *DL-Based RFF Identification Methods:* Unlike the traditional RFF identification methods, the DL-based RFF identification benefit from the powerful feature extraction and model fitting capabilities, and avoid the complex feature extraction operation. Usually, the RF signal is in the form of I/Q signals, while there are several methods which convert I/Q signals into the form of image. Existing DL-based RFF identification methods can be divided into time-series signal-based and image-form signal-based methods.

The time-series data set mainly consists of I/Q signals, spectrum, or signal components in the form of time series. Merchant et al. [27] employed the multilayer neural network to identify RFF signals. Specifically, the authors estimate the

ideal signal, and then subtract the ideal signal from the measured signal to get the error signal, which can be considered as the RFF feature. Then, the CNN is used to identify the error signal. Yu et al. [28] proposed a denoising autoencoder to reduce the influence of noise, which achieved considerable improvements at all SNR levels. Yu et al. [29] also proposed a multisampling convolutional neural network (MSCNN) to identify RF devices. The MSCNN extracts multiscale RFF feature from the selected region of interesting (ROI). The MSCNN achieves a 97% identification ratio for 54 CC2530 ZigBee devices under the line-of-sight (LOS) scenarios with SNR = 30 dB. Ding et al. [31] employed a supervised dimensionality reduction method to compress the dimensions of the bispectrum, and then, adopted a convolutional neural network to identify specific emitters.

In addition, recurrent neural networks (RNNs) have been used to handle time series. Roy et al. [32] studied the performance of long short-term memory (LSTM), gated recurrent unit (GRU), and convolutional LSTM (ConvLSTM) for RFF identification. RF signals collected from 8 USRP B210 were used, while achieved an accuracy of over 92%.

The common image-form signals include spectrogram and constellation diagram. Shen et al. [33], [39] used STFT to convert the I/Q signal into spectrogram. Then, a metric-learning-based neural network was used to extract RFF features. Note that Peng et al. [40] used the difference constellation trace figure (DCTF) to replace I/Q signals, which was focused on the carrier frequency offset between the transmitter and receiver. Tu et al. used a window function to slide on the constellation diagrams. Then, the number of sampling points in the window were countered to generate the contour stella image [41]. Peng et al. [42] converted the I/Q signal into heat constellation trace figure (HCTF), which used the area of the Voronoi diagram to calculate the heat of the sampling points trace. A slice integration cooperation was also employed to improve the performance. The simulation results showed that all of the RFF identification methods based on the constellation diagram were effective.

### B. DL With Limited Samples

In order to achieve a good performance with limited samples, we start with data set preprocessing and feature extraction. For example, data augmentation extends the raw data set, while a better feature extraction model has the ability of dealing with limited data more effectively.

1) *Data Augmentation*: Huang et al. [43] proposed a data augmentation method for AMC, which contained Rotation, Flip, and Gaussian. Cai et al. studied the performance of data augmentation for small sample communication device recognition. The RFF samples were augmented with different operations, which consists of noise disturbance, amplitude and time-delay transformation, frequency offset, and phase shift transformation [44]. The VAT was also used to augment the RFF samples during the training process. The simulation results demonstrated the feasibility of these methods.

2) *Metric Model and Contrastive Model*: As can be seen in Fig. 1, the metric model and contrastive model are similar in

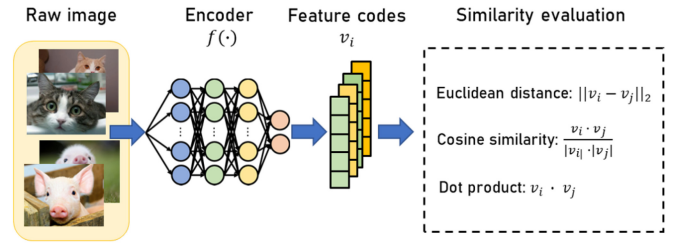


Fig. 1. Similar structure of the metric model and the contrastive model.

structure. Both of them extract features by measuring the similarity of feature vectors, which can be considered as “learning to compare.”

The Siamese neural network is a widely used metric learning method, which extracts features with a weight-shared “encoder” and measures similarity with the Euclidean distance [45]. Unlike metric learning, contrastive learning is usually used in semisupervised learning. Chen et al. [46] proposed a simple framework for contrastive learning of visual representations (SimCLR), which is a classical contrastive model. In addition, van den Oord et al. [47] modified the loss function in SimCLR for supervised representation learning. The simulation results showed that the supervised contrastive loss has a strong ability of feature representation. Wu et al. [48] and Sun [49] introduced the Siamese network for SEI, which achieved a great performance in both open set identification and few-shot learning.

3) *Meta Model*: Meta learning is also a very effective method in the face of limited samples, which uses auxiliary data sets to build multiple subtasks for the most generalized model. Yang et al. [50] first introduced the model-agnostic metalearning for SEI. The RF signals from 20 ZigBee devices were used as the auxiliary data set, while the generalizability of the model has also been demonstrated with the performance on a limited UAV data set.

Considering that the meta model has complex pretraining and unstable gradients [51], the metric model and contrastive model are more suitable for RFF identification. Those models are easy to implement, since there are similar to the conventional DL-based model. The supervised contrastive loss (SCL) in [47] can be regarded as an improved form of triplet loss, which is also a typical metric loss function. Therefore, we employ the supervised contrastive model in this article.

## III. DATA MODELING AND PROBLEM FORMULATION

### A. RFF Signal Modeling

As in [27], the collected I/Q modulation RF signal is described as

$$r_{\text{meas}}(t) = r_{\text{ideal}}(t) + r_{\text{error}}(t) \quad (1)$$

where  $r_{\text{meas}}(t)$  means the collected signal at the receiver, and  $r_{\text{error}}(t)$  means the error signal, which is considered as the RFF.

However, as can be seen in Fig. 2, the RF signal is generated by passing the baseband signal to digital to analog converter (DAC), bandpass filter, oscillator, and power amplifier (PA). Each of the modules will introduce some RF signal

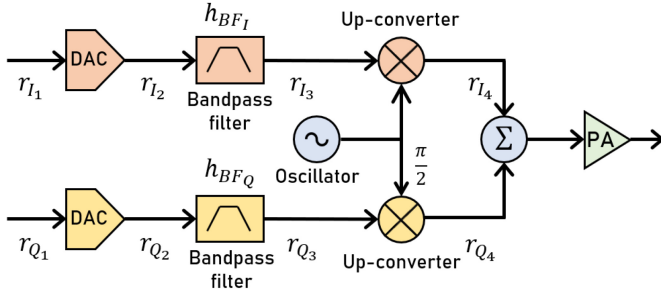


Fig. 2. Model of the RFF causes in the transmitter.

error. Therefore, (1) is not sufficient to describe the complex RFF signal, and the specific details of  $r_{\text{error}}(t)$  are also worth studying. The baseband signal can be described as

$$r_{\text{base}}(t) = \begin{bmatrix} r_{I_1}(t) \\ r_{Q_1}(t) \end{bmatrix} \quad (2)$$

where  $r_{I_1}(t)$  and  $r_{Q_1}(t)$  are the baseband signals of the  $I$  and  $Q$  channels, respectively. Then,  $r_{\text{base}}$  is transmitted into the DAC, and the signal after DAC is described as

$$r_{\text{DC}}(t) = \begin{bmatrix} r_{I_2}(t) \\ r_{Q_2}(t) \end{bmatrix} = \begin{bmatrix} r_{I_1}(t) + \mu_I \\ r_{Q_1}(t) + \mu_Q \end{bmatrix} \quad (3)$$

where  $\mu_I$  and  $\mu_Q$  are the direct current (dc) offsets of the  $I$  and  $Q$  channels, respectively. Then, the bandpass filter is used to filter  $r_{\text{DC}}$ , and the filter error is described as

$$r_{\text{BF}}(t) = \begin{bmatrix} r_{I_3}(t) \\ r_{Q_3}(t) \end{bmatrix} = \begin{bmatrix} h_{\text{BF}_I}(t) \otimes r_{I_2}(t) \\ h_{\text{BF}_Q}(t) \otimes r_{Q_2}(t) \end{bmatrix} \quad (4)$$

where  $h_{\text{BF}_I}(t)$  is the filter transfer function of the  $I$  channel, and  $h_{\text{BF}_Q}(t)$  is the transfer function of the  $Q$  channel.  $\otimes$  symbol means the convolution operation. When  $r_{\text{BF}}(t)$  is passed into the upconverter, the RF signal will be corrupted by the gain imbalance. In addition, the oscillator will also introduce a phase offset and the frequency offset in the carrier. The phase difference of the  $I$  channel and  $Q$  channel is not exactly equal to  $\pi/2$ , which will also introduce errors in  $r_{\text{meas}}(t)$ . The signal  $r_{\text{up}}(t)$  is given as

$$r_{\text{up}}(t) = \begin{bmatrix} r_{I_4}(t) \\ r_{Q_4}(t) \end{bmatrix} = \begin{bmatrix} \lambda_I \cdot r_{I_3}(t) \cdot \cos[(\omega_c + \Delta\omega)t + \Delta\varphi + \varphi_I] \\ \lambda_Q \cdot r_{Q_3}(t) \cdot \sin[(\omega_c + \Delta\omega)t + \Delta\varphi + \varphi_Q] \end{bmatrix} \quad (5)$$

where  $\lambda$  is the gain imbalance of the upconverters,  $\varphi$  is the quadrature offset error, and  $\Delta\omega$  and  $\Delta\varphi$  are the offset of phase and frequency, respectively. Then,  $r_{\text{in}}$  is written as

$$\begin{aligned} r_{\text{in}}(t) &= r_{I_4}(t) + r_{Q_4}(t) \\ &= \mathcal{R} \left[ (X_I + j \cdot X_Q) e^{-j[(\omega_c + \Delta\omega)t + \Delta\varphi]} \right] \\ &= \sqrt{k_I^2 + k_Q^2 + 2 \sin(-\varphi_I + \varphi_Q) \cdot \cos[\omega_{\text{in}}t + \varphi_{\text{in}}]} \quad (6) \end{aligned}$$

where  $k_I = \lambda_I \cdot r_{I_3}(t)$ ,  $k_Q = \lambda_Q \cdot r_{Q_3}(t)$ ,  $\omega_{\text{in}} = \omega_c + \Delta\omega$ , and  $\varphi_{\text{in}} = \Delta\varphi + \arctan([k_I \cos \varphi_I + k_Q \sin \varphi_Q] / [-k_I \sin \varphi_I + k_Q \cos \varphi_Q])$ .  $X_I$  and  $X_Q$  are expressed as

$$\begin{aligned} X_I &= \lambda_I \cdot r_{I_3}(t) \cdot \cos \varphi_I - j \cdot \lambda_I \cdot r_{I_3}(t) \cdot \sin \varphi_I \\ X_Q &= -\lambda_Q \cdot r_{Q_3}(t) \cdot \sin \varphi_Q - j \cdot \lambda_Q \cdot r_{Q_3}(t) \cdot \cos \varphi_Q. \quad (7) \end{aligned}$$

The nonlinearity of the PA plays a significant role in  $r_{\text{error}}$ , which can be described with the baseband nonlinear model. In this article, the memory effects of PA is ignored. The AM-PM conversion of PA can also be ignored, when the carrier frequency is much higher than the baseband bandwidth. The Saleh model [52] and the complex coefficient polynomial model [53] are employed to model the nonlinearity of the PA.

1) *Saleh Model*: The AM-AM conversion of PA in the Saleh model is written as

$$A_{\text{PA}}(t) = \frac{\alpha_a A(t)}{1 + \beta_a [A(t)]^2} \quad (8)$$

$$r_{\text{out}}(t) = A_{\text{PA}}(t) \cos(\omega_{\text{in}}t + \varphi_{\text{in}}) \quad (9)$$

where  $\alpha_a$  and  $\beta_a$  are the fitting parameters of PA.  $A(t)$  is the amplitude of the RF signal and  $A(t) = |r_{\text{in}}(t)|$ , and  $\omega_{\text{in}}$  and  $\varphi_{\text{in}}$  are the frequency and the phase of  $r_{\text{in}}$ , respectively.

2) *Complex Coefficient Polynomial*: The AM-AM conversion of PA in the complex coefficient polynomial model is as follows:

$$r_{\text{out}}(t) = \sum_{k=1}^N a_k \cdot [r_{\text{in}}(t) + \mathcal{H}(r_{\text{in}}(t))] r_{\text{in}}(t) + \mathcal{H}(r_{\text{in}}(t)) \Big|^{k-1} \quad (10)$$

where  $\mathcal{H}(\cdot)$  means the Hilbert transform. Considering the model of the RFF signal,  $r_{\text{error}}(t)$  in (1) is a function of  $r_{\text{ideal}}$ . Therefore, there will be a residual component of  $r_{\text{ideal}}(t)$ , which will influence the performance of RFF identification. We rewrite (1) as follows:

$$r_{\text{meas}}(t) = G_r(r_{\text{ideal}}(t), \theta_i) \quad (11)$$

where  $G_r$  denotes the model of the RFF signal, and  $\theta_i = \{\mu_i, h_{\text{BF}_i}, \Delta\omega_i, \Delta\varphi_i, \varphi_i, \mathbf{w}_{\text{PA}}^i\}$  represents the parameters which belong to  $y_i$ , and the  $y_i$  is the real label of RF device.

## B. Problem Description

The RFF identification is based on the imperfection of the transmitter, which is represented by  $\theta_i$  in (11). In addition,  $\theta_i$  can be fitted with  $r_{\text{meas}}(t)$  and  $r_{\text{ideal}}(t)$ . However,  $r_{\text{ideal}}(t)$  may be hard to obtain, and it is impracticable to achieve  $\theta_i$  when considering the error of the modeling. Thus, most of the RFF identification methods extract the feature vector of  $\theta_i$  rather than finding the specific value with  $r_{\text{meas}}(t)$ .

1) *RFF Identification Problem*: We use  $\mathbf{r}_n$  to represent  $r_{\text{meas}}(t)$ , which is the  $n$ th RFF sample collected from an RF device. Let  $y_n$  represent the real category of  $n$ th RFF sample. The data set of RFF signal is denoted as  $\mathbf{D}_{\text{raw}} = \{\mathbf{r}_n, y_n\}$ . RFF identification can be described as a feature extractor problem, which is given as follows:

$$\begin{aligned} \mathbf{D}_h &= \{\mathbf{h}_n, y_n\} \\ \mathbf{h}_n &= f_{\text{RFF}}(\mathbf{r}_n) = f_{\text{RFF}}(G_r(r_{\text{ideal}}^n(t), \theta_i^n)) \end{aligned} \quad (12)$$

where  $\mathbf{h}_n$  is the feature of  $\mathbf{r}_n$  and  $f_{\text{RFF}}(\cdot)$  is the feature extractor.  $\mathbf{D}_h$  is the data set of features.  $r_{\text{ideal}}^n(t)$  is the disturbed component of  $\mathbf{r}_n$ .  $\mathbf{h}_n$  should reduce the influence of  $r_{\text{ideal}}^n(t)$ , which is described as

$$P(\mathbf{h}_n | r_{\text{ideal}}^n(t)) = \frac{P(\mathbf{h}_n, r_{\text{ideal}}^n(t))}{P(r_{\text{ideal}}^n(t))} \quad (13)$$



where  $P(r_{\text{ideal}}^n(t))$  is the marginal probability distribution, and  $P(\mathbf{h}_n, r_{\text{ideal}}^n(t))$  is the joint probability distribution.

Then,  $\mathbf{h}_n$  is considered as the RFF feature. The softmax-based identification methods of the RFF feature  $\mathbf{h}_n$  are written as

$$\min \mathbb{E}_{(\mathbf{h}_n, y_n) \sim \mathbf{D}_h} \mathcal{L}(\hat{y}_n, y_n) \quad (14)$$

where  $\hat{y}_n$  is the predicted category of  $\mathbf{h}_n$ , and  $\mathcal{L}(\cdot)$  is used to evaluate the difference between  $\hat{y}_n$  and  $y_n$ . The frequently used loss function  $\mathcal{L}(\cdot)$  here is cross-entropy loss function, which is written as

$$\mathcal{L}_{\text{CE}} = \frac{1}{N} \sum_{n=1}^N \sum_{d=1}^{n_D} \text{logit}_n^d \cdot \log P_n^d \quad (15)$$

where  $\text{logit}_n^d$  represents the logical value. When  $\hat{y}_n \neq y_d$ ,  $\text{logit}_n^d = 0$ , on the contrary,  $\text{logit}_n^d = 1$ .  $y_d$  is the  $d$ th category.  $P_n^d$  is the probability of  $\mathbf{r}_n$  belong to  $y_d$ . In addition, the metric and contrastive-based identification methods are written as

$$\min \mathbb{E}_{(\mathbf{h}_n, y_n; h_c, y_c) \sim h_N} \mathcal{L}(y_n, y_c) \quad (16)$$

where  $h_N$  is the set of all RFF features.  $h_c$  is metric or contrastive RFF feature, which usually consists of positive samples and negative samples, and  $y_c$  is the category. Contrastive loss function in the Siamese network is a typical  $\mathcal{L}(\cdot)$ , which is written as

$$\mathcal{L}_{\text{Siam}} = \frac{1}{2} \left[ (1 - \text{logit}_n^c) \cdot d(\mathbf{h}_n, h_c) + \text{logit}_n^c \cdot \max(0, \text{margin} - d(\mathbf{h}_n, h_c)) \right] \quad (17)$$

where  $d(\mathbf{h}_n, h_c)$  is the Euclidean distance of  $\mathbf{h}_n$  and  $h_c$ .  $\max(\cdot)$  means the maximum value of two elements.  $\text{margin}$  is the ideal extra-class distance. When  $y_n \neq y_c$ ,  $\text{logit}_n^c = 0$ ; otherwise, it is equal to 1.

2) *Limited Samples RFF Identification Problem*: It is obvious that the more samples in  $\mathbf{D}_h$ , the better performance of the neural network. Thus, the data augmentation is employed to acquire an extended data set without using an auxiliary data set. The extended data set is described as:  $\mathbf{D}_{\text{raw}} = \mathbf{D}_{\text{raw}} \cup \mathbf{D}_{\text{aug}} = \{\hat{r}_n, \hat{y}_n\}$  and  $\mathbf{D}_{\text{aug}} = f_{\text{ag}}(\mathbf{D}_{\text{raw}}) = \{r'_n, y'_n\}$ , where  $f_{\text{ag}}(\cdot)$  is the function of data augmentation.  $f_{\text{ag}}$  usually contains simple mathematical operations, such as adding Gaussian noise, introducing channels, etc. Then,  $\hat{\mathbf{h}}_n$  is considered as the augmented RFF feature. The identification of the augmented RFF feature  $\hat{\mathbf{h}}_n$  is given by

$$\min \mathbb{E}_{(\hat{\mathbf{h}}_n, \hat{y}_n) \sim \mathbf{D}_h \cup \mathbf{D}'_h} \mathcal{L}(\hat{y}_n, \hat{y}'_n) \quad (18)$$

$$\min \mathbb{E}_{(\hat{\mathbf{h}}_n, \hat{y}_n; \hat{h}_c, \hat{y}_c) \sim h_N \cup h'_N} \mathcal{L}(\hat{y}_n, \hat{y}'_c) \quad (19)$$

where  $\hat{y}_n$  is the predict category of  $\hat{\mathbf{h}}_n$ .

It is also a feasible method to solve this problem by setting regularization constraints on the identification model. The regularization training method based on adversarial samples are popular which constructs virtual samples by adding perturbations. which is given by

$$\mathbf{h}_n^{\text{adv}} = f_{\text{RFF}}(\mathbf{r}_n + \mathbf{r}_{\text{adv}}) \quad (20)$$

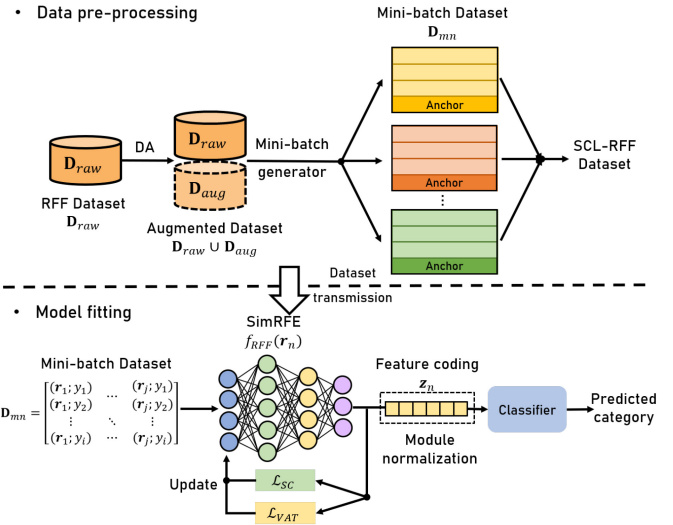


Fig. 3. Framework of the proposed DL-based RFF identification method using SCL.

where  $\mathbf{h}_n^{\text{adv}}$  is the feature of RFF signal with perturbations, and  $\mathbf{r}_{\text{adv}}$  is the perturbations which is depended on the RFF identification model. The detailed identification of the regularization training method is given in Section IV.

## IV. PROPOSED RFF IDENTIFICATION METHOD

### A. Framework of Proposed RFF Identification Method

As shown in Fig. 3, the proposed system is divided into two parts: 1) data preprocessing and 2) model fitting. RF signals from different signal generators are collected. However, the collected signals are not sufficient to achieve a high identification ratio and there is no auxiliary data set can be used. Therefore, the raw data set is augmented to obtain an extended data set. In order to adapt to SCL, the augmented data set is converted to minibatch data set. In the part of model fitting, the minibatch data set is inputted into the SimRFE. Then, the SCL and VAT is used to get a robust and specific features, which should be normalized by the module normalization (MN). Finally, the classifier is employed to get the predicted category.

### B. Data Augmentation

The measured signal above is described as

$$\begin{aligned} r_s(t) &= \sqrt{r_I^2(t) + r_Q^2(t)} \\ \phi_s(t) &= \arctan \frac{r_I(t)}{r_Q(t)} \end{aligned} \quad (21)$$

where  $r_I^2(t)$  and  $r_Q^2(t)$  denote the measured signal of the  $I$  channel and  $Q$  channel in the receiver, respectively. Here,  $r_R$  is the complete set of RF signals,  $P(r_R, y_i)$  is the marginal probability distribution of the complete set, which belongs to  $y_i$ . Data augmentation can be considered as to acquire the extended data set with prior knowledge. Then, the extended data set should be label reserved, which is described as  $y'_n = y_n$ . In addition, data augmentation aims to generate a different sample space distribution. Therefore, the main idea of

augmentation is to acquire labels preserved in the extended samples, which are least similar to the raw sample. However, it is unrealistic to measure the label reservation between  $\mathbf{D}_{\text{raw}}$  and  $\mathbf{D}_{\text{aug}}$ . In this article, all the extended data set are based on the transformation of raw data. Four different simple data transformation are employed: 1) *Rotation*; 2) *Flipping*; 3) *Gaussian* [43]; and 4) *Shifting*.

1) *Rotation*: Considering the modulation of the RF signal used, the RF signal in the  $I$  channel and  $Q$  channel share the same generation method, except the different symbols (the difference of carrier frequency can be removed by phase shifting). Therefore, it is reasonable to exchange the  $I$  channel and  $Q$  channel to acquire more samples, described as

$$\mathbf{r}'_n = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{bmatrix} \begin{bmatrix} r_I \\ r_Q \end{bmatrix} \quad (22)$$

where  $r_s$  denotes an extended RF sample in the receiver and  $\varphi$  is the rotation angle. Unlike the rotation angle in CV,  $\varphi$  is aimed at the value of in-phase and quadrature components. Thus,  $\varphi$  will affect the quality of the extended data set. In order to avoid superposition of in-phase and quadrature components, the ideal  $\varphi$  values are  $\pi/2$ ,  $\pi$ , and  $3\pi/2$ . When  $\varphi$  moves away from the ideal angle, the result of augmentation will be come unpredictable. However, *Rotation* will also introduce troubles in the RFF identification. As can be seen in Fig. 2, it is obvious that when the ideal angel is  $\pi/2$  or  $2\pi/3$ , the difference between the  $I/Q$  channels will be interference. As can be seen in (11),  $\mathbf{r}_n$  and  $\mathbf{r}'_n$  have different  $\mu_i$ ,  $h_{\text{BF}_i}$ , and  $\varphi_i$ . Therefore, *Rotation* acquires the extended data set at the cost of losing RFF features.

2) *Flipping*: As can be seen above, the  $I$  channel and  $Q$  channel can be exchanged to acquire more data samples. Thus,  $r_I$  and  $r_Q$  can be flipped, which is described as

$$\mathbf{r}'_n = \begin{bmatrix} \pm r_I \\ \pm r_Q \end{bmatrix}. \quad (23)$$

*Flipping* can be considered as setting the initial symbol to its opposite one. *Flipping* augments the data set with a simple operation, and  $\mathbf{r}_n$  and  $\mathbf{r}'_n$  will have different  $\mu_{is}$ .

3) *Adding Gaussian Noise*: The RF signal will be affected by the noise in transmission, and the SNR of the RF signal is not constant. Thus, Gaussian noise can be added to the RF signal to acquire the extended data set, which is described as

$$\mathbf{r}'_n = \begin{bmatrix} r_I \\ r_Q \end{bmatrix} + N_s \quad (24)$$

where  $N_s$  is a stochastic time series and it obeys the Gaussian distribution  $\mathcal{N}(\mu, \sigma)$ . However, it is difficult to find the optimal value of  $\mu$  and  $\sigma$ . Let the power of  $N_s$  be within  $[a, b]$  dB. If the SNR of  $r_s(t)$  is out of the range, the extended data set will be corrupted, which will be counterproductive to RFF identification. In addition, if  $N_s$  is too small, the result of data augmentation may be poor. Here,  $\mu$  is set as: 0, while  $\sigma$  is set as: 0.0005, 0.001, and 0.002, respectively,

4) *Shifting*: Inspired by the *Shifting* in CV,  $r_s$  can be considered as an  $L \times H$  image, where  $L$  is the length of the RF signal, and  $H = 2$  is the height. Then, the *Shifting* in  $r_s$  is

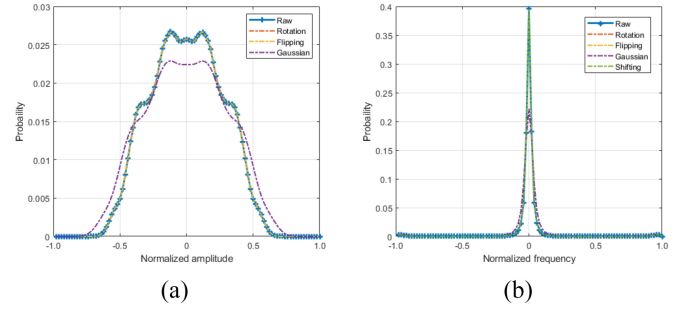


Fig. 4. Raw data set versus Augmented data set. (a) Amplitude. (b) Frequency.

described as

$$\mathbf{r}'_n = \begin{bmatrix} i_k, i_{k+1}, i_L, i_1, i_2, \dots, i_{k-2}, i_{k-1} \\ q_k, q_{k+1}, q_L, q_1, q_2, \dots, q_{k-2}, q_{k-1} \end{bmatrix} \quad (25)$$

where  $i_k$  and  $q_k$  denote the  $k$ th sampling point of the  $I$  channel and  $Q$  channel, respectively. Unlike CV, RFF identification does not rely on the information of the signal itself, which makes it free from the risk of cutting off the interest area. However, when the RF signal is composed of a constant frame structure, *Shifting* may decrease the accuracy of RFF identification. In addition, as can be seen in (4),  $r_{\text{BF}_i}(t)$  is obtained by the convolution operation. Therefore, the shifted data will have a different  $h_{\text{BF}_i}$ . In addition,  $r_{\text{up}}$  and  $r_{\text{out}}$  are generated from  $r_{\text{BF}_i}(t)$ . Therefore,  $\mathbf{r}'_n$  after *Shifting* only share the same  $\mu_i$  with  $\mathbf{r}_n$ .

Here, the self-collected data set Device 1 is taken as an example. The data distributions of raw data set and augmented data set are shown in Fig. 4. Only the adding Gaussian noise has significant change. The amplitude and frequency are also normalized to avoiding interference. Considering the Shift does not change the amplitude of raw data set, it only given in the data distributions of frequency. The data distributions of the phase are also analyzed, while the distributions of those methods are quiet similar.

### C. Virtual Adversarial Training

The VAT is the most classical regularization training method, which can improve the generalization of the model by virtual adversarial samples. Unlike the traditional adversarial training methods, the virtual perturbations are obtained through gradient, which is described as

$$\mathbf{g}_n \leftarrow \nabla_{r_{\text{adv}}} \text{KL}(\mathbf{r}_{\text{adv}}) |_{r_{\text{adv}} = \xi \cdot \mathbf{d}_n} \quad (26)$$

$$\mathbf{r}_{\text{adv}} = \epsilon \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} \quad (27)$$

where  $\mathbf{g}_n$  is the gradient of the distance between raw RFF signal and RFF signal with perturbations,  $\mathbf{d}_n$  is a random unit vector obeying Gaussian distribution,  $\xi$  and  $\epsilon$  are tiny numbers used to control the intensity of perturbations, and  $\text{KL}(\cdot)$  is the Kullback–Leibler (KL) divergence, which is described as

$$\text{KL}(\mathbf{r}_{\text{adv}}) = \sum f_{\text{RFF}}(\mathbf{r}_n) \log \frac{f_{\text{RFF}}(\mathbf{r}_n)}{f_{\text{RFF}}(\mathbf{r}_n + \mathbf{r}_{\text{adv}})} \quad (28)$$

where  $f_{\text{adv}}(\cdot)$  is the RFF identification model. Then, the KL divergence is added as the loss function. Considering the

---

**Algorithm 1: Detailed Procedures of SCL Algorithm**


---

**Input:**  $\mathbf{z}_i \leftarrow$  RFF feature coding;  $n_D \leftarrow$  number of categories;  $N \leftarrow$  number of samples;  
**Output:**  $\mathcal{L} \leftarrow$  supervised contrastive loss;  
**1 Set hyperparameters:**  
**2**  $n_{\text{view}} \leftarrow$  View;  $\tau \leftarrow$  number of temperature ; **Logit** = 0;  
**3 for**  $i=1$  to  $n_{\text{view}} \cdot n_D$  **do**  
**4**     **Logit**<sub>dot</sub>( $i, j$ ) =  $\frac{\mathbf{z}_i \cdot \mathbf{z}_j}{\tau}$ ;  
**5**     **Logit**<sub>e</sub>( $i, j$ ) =  $\exp(\mathbf{Logit}_{\text{dot}}(i, j))$ ;  
**6**     **Logit**<sub>e</sub>( $i, j$ ) = {**Logit**<sub>e</sub><sup>+</sup>( $i, j$ ); **Logit**<sub>e</sub><sup>-</sup>( $i, j$ )};  
**7**     **Logit**( $i$ ) =  $\frac{1}{N^+} \log \frac{\mathbf{Logit}_{\text{e}}^+(i, j)}{\sum_{i \neq j} \mathbf{Logit}_{\text{e}}(i, j)}$ ;  
**8**     **Logit** = **Logit** + **Logit**( $i$ );  
**9 end**  
**10**  $\mathcal{L} = \mathbf{Logit}$ ;  
**11 return**  $\mathcal{L}$ ;  


---

generation of perturbations, the VAT can be considered as a model-based data augmentation method.

#### D. SCL-Based RFF Identification Method

As mentioned above, SCL is evolved from the normalized temperature-scaled cross-entropy (NT-Xent) loss in semisupervised contrastive learning (SimCLR). The NT-Xent loss is mainly used in SimCLR. The main steps of SimCLR are as follows: first, a huge number of unlabeled data are augmented, and the augmented sample and the raw sample are considered to have the same label. Second, all the samples are mapped to the feature space by an encoder. Finally, feature coding is constrained by the NT-Xent loss to increase the similarity between the raw sample and the augmented sample.

When the NT-Xent loss is introduced to SCL, NT-Xent loss is incapable of using the information of label. Thus, the SCL is proposed, which is described as

$$\mathcal{L}_{\text{SC}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)} \quad (29)$$

where  $A(i)$  is the set of all the samples except the  $i$ th sample,  $P(i) \equiv \{p \in A(i) : y_p = y_i\}$  is the set of positive samples in  $A(i)$ , and  $\tau$  is the temperature. In order to distinguish from the hidden layer features of other metric and contrastive models,  $\mathbf{z}$  is used here to represent the feature coding.

The detailed procedure of SCL is introduced in Algorithm 1. It can be seen that when there is only one positive sample, SCL is the same as the NT-Xent loss. Meanwhile, if there is only one class of negative examples, SCL can be regarded as a deformation of triplet loss. In this article, we propose two modules for RFF identification with SCL.

1) *Module Normalization Layer:* The MN layer in this article is described as

$$L_{\text{Norm}}(\mathbf{z}_i) = \frac{\mathbf{z}_i}{|\mathbf{z}_i|}. \quad (30)$$

As can be seen in (29), the similarity of feature coding is measured by the value of the dot product. It is obvious that the module of feature coding will affect the performance of

SCL. Thus, an analysis of gradients is implemented, which is focused on the module of feature coding

$$\mathcal{L}_i = \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)}. \quad (31)$$

Then, the partial derivative of  $\mathcal{L}_i$  with respect to  $\mathbf{z}_j$  is described in three scenarios

$$j = \begin{cases} j = i & (a) \\ j \in P(i) & (b) \\ j \in N(i) & (c) \end{cases} \quad (32)$$

where  $N(i) \equiv \{n \in A(i) : y_n \neq y_i\}$  is the set of negative samples in  $A(i)$ . The partial derivative of each scenario is as follows.

When  $i = j$ ,

$$\frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_j} = \frac{-1}{|P(i)| \cdot \tau} \sum_{p \in P(i)} (\mathbf{z}_p - |P(i)| \cdot P_{ia}), \quad (33)$$

when  $\mathbf{z}_j$  is belong to positive samples,

$$\frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_j} = \frac{-1}{|P(i)| \cdot \tau} (\mathbf{z}_i - |P(i)| \cdot P_{ij}), \quad (34)$$

when  $\mathbf{z}_j$  is belong to negative samples,

$$\frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_j} = \frac{-1}{|P(i)| \cdot \tau} (-|P(i)| \cdot P_{ij}), \quad (35)$$

where  $P_{ix} = ([\sum_{x \in X(i)} \mathbf{z}_x \cdot \exp(\mathbf{z}_i \cdot \mathbf{z}_x / \tau)] / [\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)])$ . Then,  $(\partial \mathcal{L} / \partial \mathbf{z}_j)$  is given by

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{z}_j} &= \sum_{i \in I} \frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_j} \\ &= \left[ \frac{\partial \mathcal{L}_1}{\partial \mathbf{z}_j}, \frac{\partial \mathcal{L}_2}{\partial \mathbf{z}_j}, \dots, \frac{\partial \mathcal{L}_{i-1}}{\partial \mathbf{z}_j}, \frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_j} \right] \cdot \begin{bmatrix} \frac{\partial \mathcal{L}}{\partial \mathcal{L}_1} \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_2} \\ \dots \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_{i-1}} \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_i} \end{bmatrix} \end{aligned} \quad (36)$$

and  $(\partial \mathcal{L} / \partial \mathbf{z})$  is described as

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{z}} &= \begin{bmatrix} \frac{\partial \mathcal{L}_1}{\partial \mathbf{z}_1} & \frac{\partial \mathcal{L}_2}{\partial \mathbf{z}_1} & \dots & \frac{\partial \mathcal{L}_{i-1}}{\partial \mathbf{z}_1} & \frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_1} \\ \frac{\partial \mathcal{L}_1}{\partial \mathbf{z}_2} & \frac{\partial \mathcal{L}_2}{\partial \mathbf{z}_2} & \dots & \frac{\partial \mathcal{L}_{i-1}}{\partial \mathbf{z}_2} & \frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{\partial \mathcal{L}_1}{\partial \mathbf{z}_{j-1}} & \frac{\partial \mathcal{L}_2}{\partial \mathbf{z}_{j-1}} & \dots & \frac{\partial \mathcal{L}_{i-1}}{\partial \mathbf{z}_{j-1}} & \frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_{j-1}} \\ \frac{\partial \mathcal{L}_1}{\partial \mathbf{z}_j} & \frac{\partial \mathcal{L}_2}{\partial \mathbf{z}_j} & \dots & \frac{\partial \mathcal{L}_{i-1}}{\partial \mathbf{z}_j} & \frac{\partial \mathcal{L}_i}{\partial \mathbf{z}_j} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \mathcal{L}}{\partial \mathcal{L}_1} \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_2} \\ \dots \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_{i-1}} \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_i} \end{bmatrix} \\ &= \mathcal{J}_{\mathbf{z}}^T \cdot \begin{bmatrix} \frac{\partial \mathcal{L}}{\partial \mathcal{L}_1} \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_2} \\ \dots \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_{i-1}} \\ \frac{\partial \mathcal{L}}{\partial \mathcal{L}_i} \end{bmatrix} \end{aligned} \quad (37)$$

where  $\mathcal{J}_{\mathbf{z}}$  is the Jacobian, all the elements in the matrix should meet three scenarios above. It is obvious that when the module of  $\mathbf{z}$  is too large,  $\mathcal{J}_{\mathbf{z}}$  will be more sparse. Thus, the back propagation will be slowed down or even ‘‘killed.’’ In addition, in reality, the loss is limited by the numeric type. If the module is unconstrained, the  $\exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)$  term will be out of control.

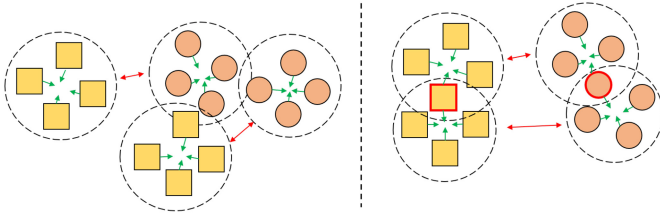


Fig. 5. Sketch of maps of SCL with minibatch. The red “ $\leftrightarrow$ ” means expand, the green “ $\leftarrow$ ” means close and the samples with red line is the “anchor.”

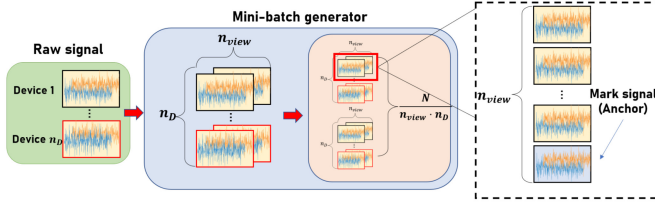


Fig. 6. Framework of the minibatch generator.

### Algorithm 2: Detailed Procedures of Minibatch Generator

---

**Input:**  $\mathbf{D}_{raw} \leftarrow$  RFF dataset;  $n_D \leftarrow$  number of categories;  $N \leftarrow$  number of samples;

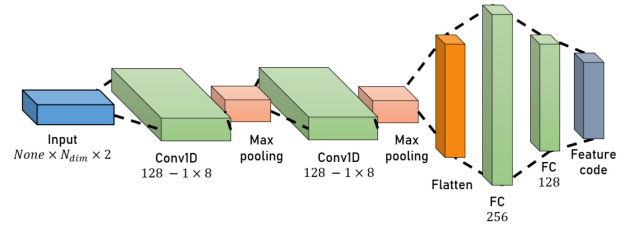
**Output:**  $\mathbf{D}_{mb} \leftarrow$  mini-batch dataset;

- 1 **Set hyperparameters:**
- 2  $n_{view} \leftarrow$  view;  $\lambda \leftarrow$  mark interval;
- 3 **count** = 0;
- 4  $\mathbf{x}_{anchor}^d = \mathbf{x}_1^d$
- 5 **for**  $n = 1$  to  $\text{int}(\frac{1}{n_{view} \cdot n_D})$  **do**
- 6      $\mathbf{D}_{view}^d = \{\mathbf{x}_1^d, \mathbf{x}_2^d, \dots, \mathbf{x}_{n_{view}-1}^d, \mathbf{x}_{anchor}^d\}$ ;
- 7      $\mathbf{D}_{mb}^n = \{\mathbf{D}_{view}^1, \mathbf{D}_{view}^2, \dots, \mathbf{D}_{view}^{n_D-1}, \mathbf{D}_{view}^{n_D}\}$ ;
- 8     **count** = **count** + 1;
- 9     **if** **count**  $\geq \lambda$  **then**
- 10          $\mathbf{x}_{anchor}^i = \mathbf{x}_1^i$ ;
- 11         **count** = 0;
- 12     **end**
- 13      $\mathbf{D}_{mb} = \{\mathbf{D}_{mb}^1, \mathbf{D}_{mb}^2, \dots, \mathbf{D}_{mb}^{n-1}, \mathbf{D}_{mb}^n\}$ ;
- 14 **end**
- 15 **return**  $\mathbf{D}_{mb}$ ;

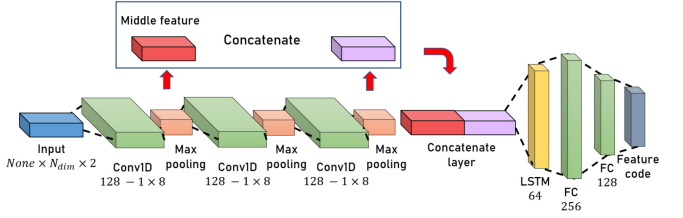
---

2) *Minibatch Generator*: The minibatch in SCL [47] refers to the semisupervised learning in CV. Therefore, the minibatch in RFF identification needs improvement. As shown in Algorithm 1 and (37), the samples in different minibatch are independent of each other. It will lead to the  $\mathbf{x}_i$  with the same  $\mathbf{y}$  be mapped into  $\mathbf{z}_i$  with various distributions, which can be seen in the left plot of Fig. 5. Thus, this article employs a minibatch generator to constrain  $\mathbf{z}_i \in \mathbf{y}$  in different minibatches, which is shown in the right plot of Fig. 5. With the “anchor,” the samples of different minibatch are linked together. The details of the minibatch generator are given in Algorithm 2.

As shown in Algorithm 2 and Fig. 6, the last sample in each minibatch is chosen as the mark signal or in other words, the anchor. The mark interval  $\lambda$  is employed to adjust the number of anchors between the minibatch.



(a)



(b)

Fig. 7. Structure of the neural network. (a) CNN [27]. (b) CLDNN.

### E. Proposed SimRFE

Contrary to the traditional softmax-based supervised RFF identification method, the SCL-based method consists of the SimRFE and the secondary classifier. The SimRFE shares the same feature extraction layers with the traditional method, but the last fully connection layers are different. In this article, two neural networks are employed to analyze the performance of the SCL-based method. One of them is considered as a weak neural network, and the other is considered as a competent neural network to study the stability of the proposed method. The structure of the neural networks can be seen in Fig. 7.

The neural network in Fig. 7(a) is a typical CNN, which is modified from [27]. The neural network in Fig. 7(b) is a convolution, long-short-term-memory, and fully connected deep neural network (CLDNN) [54]. The specific parameters are similar to the CNN. The CLDNN can be considered as a combination of CNN and LSTM. The CNN can reduce the frequency variance of RF signals, and the LSTM is employed to handle the middle features with a large number of time features. In order to acquire the middle features at different time scales, the output of the first convolution layer and the second convolution layer are concatenated. Then, the features are passed into the fully connected layers with the same structure. The last layer of both neural networks is dependent on the loss function. The Adam is employed as the optimizer, and the initial learning rate is 0.001.

### F. Secondary Classifier

Contrary to the softmax-based RFF identification method, the SCL and triplet loss-based method needs a secondary classifier for RFF identification with the feature codings. Thus, this article employs several classification models, including DL, machine learning, and simple mathematics.

1) *DL*: The softmax-based CNN and fully connection neural network are used here, which are denoted as *DL1* and *DL2*, respectively. Both neural networks share partial of their structure, which is shown in Table I.



TABLE I  
 STRUCTURE OF THE DL SECONDARY CLASSIFIER

Layers	Size	Activation function
Conv1D	$128 \times 1 \times 4$	PRReLU
Dropout	–	–
Dense	128	PRReLU
Dropout	–	–
Dense	$n_D$	Softmax

---

**Algorithm 3:** Detailed Procedures of Distance Similarity Measurement
 

---

**Input:**  $\mathbf{z} \leftarrow$  RFF feature codes;  $n_D \leftarrow$  number of categories;  $N_{test} \leftarrow$  number of samples;  
**Output:**  $\mathbf{R}_{dp}^{N_{test} \times n_D} \leftarrow$  Dot product matrix  
 1  $\hat{\mathbf{z}}^j = \frac{1}{|I(i,j)|} \sum_{i \in I(i,j)} \mathbf{z}_i$ ;  
 2 **for**  $i = 1$  to  $N_{test}$  **do**  
 3     **for**  $j = 1$  to  $n_D$  **do**  
 4          $\mathbf{R}_{dp}(i, j) = \mathbf{z}_i \times \hat{\mathbf{z}}^j$   
 5     **end**  
 6 **end**  
 7 **return**  $\mathbf{R}_{dp}^{N_{test} \times n_D}$ ;

---

The CNN is composed of a 1-D convolution layer with a  $1 \times 4$  kernel, and two dense layers. While the fully connection neural network consists of the last two dense layers, which are the same as the layers in CNN.

2) *Machine Learning*: Considering the cost of neural networks, this article also employs the traditional machine learning method as the secondary classifier, i.e., SVM, which is a classical supervised learning method.

3) *Distance Similarity Measurement*: Both the secondary classifiers above need model fitting and complex parameter passing. Considering the core of the SCL, this article also implements a classification index with the dot product. The detail is given in Algorithm 3, where  $I(i, j) \equiv \{i \in I : y_i = y_j\}$  is the set of all the samples belonging to the  $j$ th category, and  $\hat{\mathbf{z}}^j$  is the “standard” feature coding of the  $j$ th sample. It is obvious that the dot classifier is based on simple mathematical operations without model fitting and complex parameter passing. Then, the max value of column in the  $i$ th row indicates the category  $i$ , i.e.,

$$\hat{y}_i = \arg \max(\mathbf{R}_{dp}). \quad (38)$$

### G. Comparison Methods

In this article, an SCL-based RFF identification with data augmentation and VAT is employed to solve the limited samples RFF identification problem, which is named as “SCACNN.” The other RFF identification methods for comparison are as follows.

1) *Direct and DA-Aid Methods*: The direct methods are the classical RFF identification. Here, two different RFF identification architectures are used: softmax-based [27], [31], triplet-based [34], based on training on  $\mathbf{D}_{raw}$ . Besides, the

 TABLE II  
 PARAMETERS SETTING OF THE RFF DATA SET

Experiment parameters	Our dataset	Public dataset
Sampling frequency	5 MHz	
Modulation mode	16QAM	
Number of devices	7	16
Length of samples	5,120	128
Number of samples	63,000	2,500,800
Carrier frequency	443 MHz	2.45 GHz
Signal generator	VSG25A	USRP X310
Spectrum analyzer	BB60C	USRP B210

DA-aid methods are based on training on  $\mathbf{D}_{raw} \cup \mathbf{D}_{aug}$  [43]. It is worth noting that all of the network structures are replaced with suitable structure for fair comparison.

2) *SiameseNet Method*: The SiameseNet method is an effective contrastive learning architecture, and is also widely used in RFF identification. Here, the SiameseNet method [48], [49] uses  $\mathbf{D}_{raw}$  for training, and its specific implementation steps have been introduced in Section II. In addition, the CNN of the SiameseNet is also replaced with a suitable structure to adapt to the limited samples RFF identification in this article.

3) *Instantaneous Feature*: The instantaneous feature is a traditional RFF identification, which extracts the statistical feature of I/Q samples. First, the instantaneous amplitude, phase, and frequency are extracted, and the raw I/Q samples are divided into slices. Then, the variance, skewness, and kurtosis of the instantaneous components of each slice are used to represent the I/Q signal [16].

## V. EXPERIMENTAL RESULTS

### A. Data Set and Experiment Setup

In this article, to verify the universality of our proposed SCACNN, two data sets are used: 1) an RFF data set collected by ourselves and 2) the public RFF data set [55]. The first data set is collected by the spectrum analyzer BB60C. The signal generator VSG25A with different PAs are used to generate RF signals. Each device is collected with LOS transmission. The environment in which the sampled signals are collected is approximately a noise-free environment. In addition, Our data set does not contain a special signal structure, it is only a simple 16QAM modulated signal transmitted by single antenna. The public data set collects the RF signal from different Wi-Fi devices, each emulated by a USRP X310. A fixed USRP B210 is used to collect the RF signals. Both of the parameters in the data set are set to the same to avoid the influence of the other feature in the signal. The detailed settings of data sets are shown in Table II.

The empirical cumulative distribution function (ECDF) and probability distribution are distribution measurement methods. As can be seen in Fig. 8, all of the devices has the similar distributions. More specifically, the Device 3 and Device 6 may be easily classified. However, the other devices need appropriate RFF signal identification methods.

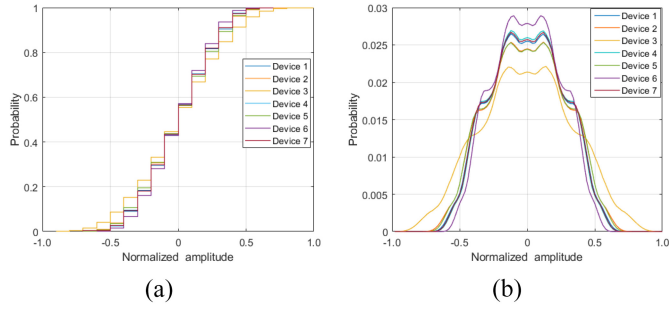


Fig. 8. Data distribution of our data set. (a) ECDF. (b) Probability distribution.

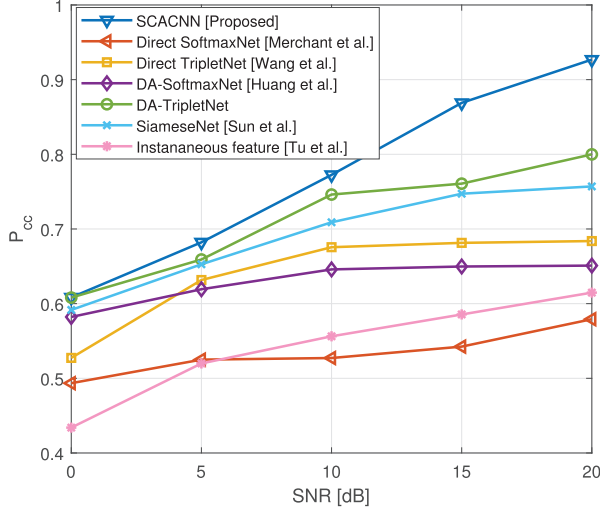


Fig. 9. Performance of different RFF identification methods.

Both the above data sets are divided into three sets: 1) a training data set; 2) a validation data set; and 3) a test data set. The training data set consists of 60% samples, while the validation data set takes 20% of samples and the test data set takes the rest. RFF identification achieves a great result with the complete training data set and validation data set. Then, a few samples of the complete training data set are randomly selected to simulate the case of RFF identification with a limited data set. Significantly, considering the difference of the data sets, the data ratio mentioned below is a relative value, i.e., 5% of samples in data set A is not equivalent to the same data ratio of samples in data set B.

Data preprocessing is carried out with MATLAB. The loss functions and neural networks are built with Python and Tensorflow. A PC with GeForce RTX 2070 is used to train the neural networks in this article.

### B. Performance of Different RFF Identification Method

The performance of different RFF identification methods with 5% RFF samples is shown in Fig. 9. The proposed SCACNN is compared with several classical RFF identification methods. It is obvious that the proposed SCACNN has the best performance in all of the SNR environments.

From the perspective of the RFF identification model architecture, the contrastive loss architecture shows better

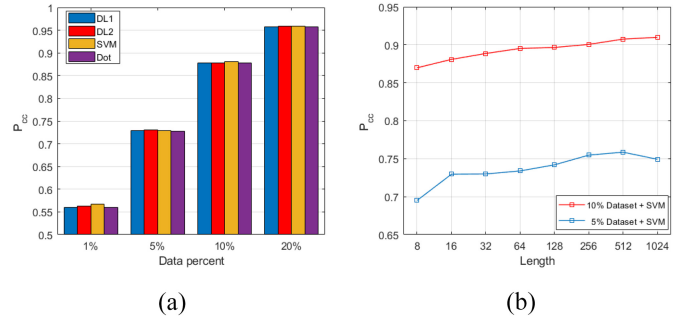


Fig. 10. Performance of different hyperparameters. (a) Different secondary classifier. (b) Different length of  $\mathbf{z}_i$ .

TABLE III  
OPERATION TIME OF DIFFERENT SECONDARY CLASSIFIERS

Data ratio	DL1	DL2	SVM	Dot
0.05	13.2825s	11.2577s	0.0453s	0.0023s
0.1	17.4738s	13.8143s	0.0629s	0.0025s

performance than the Softmax loss architecture except the DA-SoftmaxNet. It indicates that the contrastive loss architecture does well in dealing with the limited samples problem, or in other words, the contrastive loss architecture has stronger feature extraction capability than the Softmax one. In addition, the DA-aid also shows wonderful performance improvement. However, considering the basic performance of SoftmaxNet, the DA-SoftmaxNet in [43] is incapable to solve the limited samples problem. Besides, the instantaneous feature also shows terrible performance, which indicates that traditional RFF feature extraction will also facing the limited samples problem.

### C. Performance of Different Hyperparameters

There are lots of hyperparameters that can influence the performance of the proposed SCACNN. Here, the different secondary classifiers, the length of  $\mathbf{z}_i$ , and the use of the MN layer are considered. As can be seen in Fig. 10(a), the secondary classifiers of the SCACNN shows negligible difference, and the same is true for the classifiers of the triplet loss-based method. The Dot scheme in Fig. 10(a) means the distance similarity measurement.

It is obvious that the Dot shows the worst performance in all case, while the performance of *DL1* and *DL2* is similar which is slightly less effective than the SVM. However, even the difference between the Dot and the SVM is within 1%. From the perspective of computational cost, as can be seen in Table III, the operation time of secondary classifiers are shown. The Dot scheme has the east computational cost, while the *DL1* and *DL2* have thousand-fold operation time. Even the SVM also has twentyfold operation time. Finally, considering the performance and operation time, the SVM and the Dot scheme are both optional secondary classifiers. Here, in order to facilitate comparison with other contrastive architecture models, the SVM is selected as the representative of secondary classifier in this article.

TABLE IV  
ACCURACY OF RFF IDENTIFICATION WITH DIFFERENT  
DATA AUGMENTATION METHODS

Method	Data ratio	Benchmark	Rotation	Flipping	Gaussian	Shifting
SoftmaxNet	5%	0.5792	0.6469	0.6457	0.5772	0.6022
	10%	0.7824	0.8553	0.8194	0.7886	0.8554
TripletNet	5%	0.6838	0.7435	0.7149	0.6681	0.7392
	10%	0.7629	0.8545	0.7964	0.7760	0.8287
SCL	5%	0.7296	<b>0.8888</b>	0.8627	0.7499	0.8458
	10%	0.8808	<b>0.9675</b>	0.9384	0.8900	0.9274

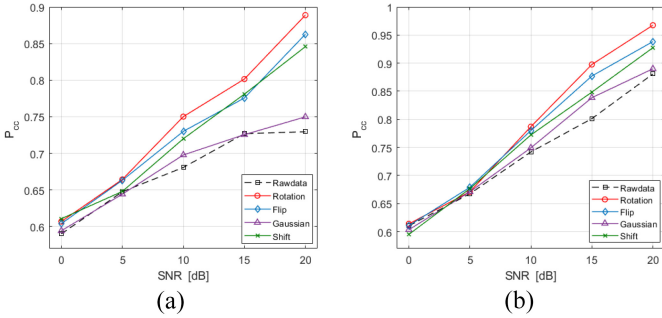


Fig. 11. Performance of the SCACNN identification method with data augmentation methods. (a) 5% samples. (b) 10% samples.

Considering the performance of the proposed method with different sizes, only the data set with 10% and 5% samples are shown. In Fig. 10(b), it can be seen that the accuracy of the proposed method increases with the increase of the length of  $\mathbf{z}_i$ . More specifically, the performance with 10% samples shows a relatively constant rate of increase, while the performance with 5% has a sharp increase between length 8 and 16. It is obvious that the longer the feature coding, the richer the feature distribution space. However, a longer feature coding will also increase the cost of computation. Significantly, the length of  $\mathbf{z}_i$  grows as a power of 2, so the actual growth rate is even slower. The feature coding with length of 16 has the optimal performance in both the 10% and 5% data sets.

#### D. Performance of Different Data Augmentation

The performance of the SCL-based RFF identification method with different data augmentation methods are also studied. The accuracy of RFF identification with different data augmentation methods in 20-dB environment is shown in Table IV. As can be seen, all the proposed data augmentation methods improve the accuracy of RFF identification. *fRotation* achieves the greatest improvement in all the cases, while *Flipping* and *Shifting* are slightly inferior to *Rotation*. In view of the loss functions, the SCL-based RFF identification method achieves the best performance with all the augmentation. What is more, the accuracy improvement of the SCL-based method also far exceeds the others. The SCL-based method with *Rotation* augmentation and 5% samples achieves an accuracy of up to 88.88%, and 96.75% with *Rotation* augmentation and 10% samples.

The performance of the SCL-based RFF identification method in different SNR environments is shown in Fig. 11. When the SNR is lower than 10 dB, the improvement of data

TABLE V  
ABLATION EXPERIMENT OF DATA AUGMENTATION

Method	0 dB	5 dB	10 dB	15 dB	20 dB
SCL-based	0.5901	0.6477	0.6808	0.7269	0.7296
SCL + VAT	0.6006	0.6756	0.7122	0.7656	0.7687
SCL + DA	0.6070	0.6644	0.7502	0.8015	0.8888
SCACNN	0.6084	0.6821	0.7725	0.8689	<b>0.9268</b>

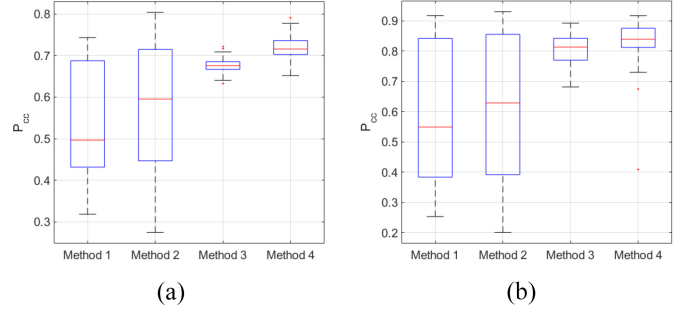


Fig. 12. Ablation experiment of SCL components. (a) 5% samples. (b) 10% samples.

augmentation is small. *Rotation* shows better performance, when the SNR is up to 10 dB.

#### E. Ablation Experiment

In order to demonstrate the effectiveness of each components in SCACNN, ablation experiments were carried out in this article. As shown in Table V, the SCACNN has the best performance in all SNR environment. It is obvious that both the DA and VAT have significant performance improvement. The SCL + VAT can improve the accuracy by about 3% when the SNR is up to 5 dB. However, the SCL + VAT also reach the upper bound when the SNR is 15 dB. The SCL + DA has a adequate improvement when the SNR is up to 10 dB. And different from SCL + VAT, the SCL + DA has a huge improvement when the SNR is 20 dB, which indicate the DA can provide additional RFF sample information. Finally, the SCACNN combines the advantages of DA and VAT, and its performance improves with the increase of SNR.

Some specific experiment result about the minibatch generator and the MN layer are also given in Fig. 12. Method 1 is the initial SCL-based method, Method 2 is the initial SCL + minibatch generator, Method 3 is the initial SCL + MN layer, and the methods is the proposed improved SCL-based method. It is obvious that the module layer has huge impact on the result, the Method 3 and Method 4 show great stability. Meanwhile, the minibatch generator can improve performance of the SCL-based method, the Method 2 and Method 4 is better than the Method 1 and Method 4, respectively. It is noting that the minibatch generator will also slightly reduce the stability, which can be ignored when compared with the improvement of identification ratio.

#### F. Performance of Different Scenarios

The proposed SCL-based RFF identification method is not only for the data set collected by ourselves and CLDNN, but

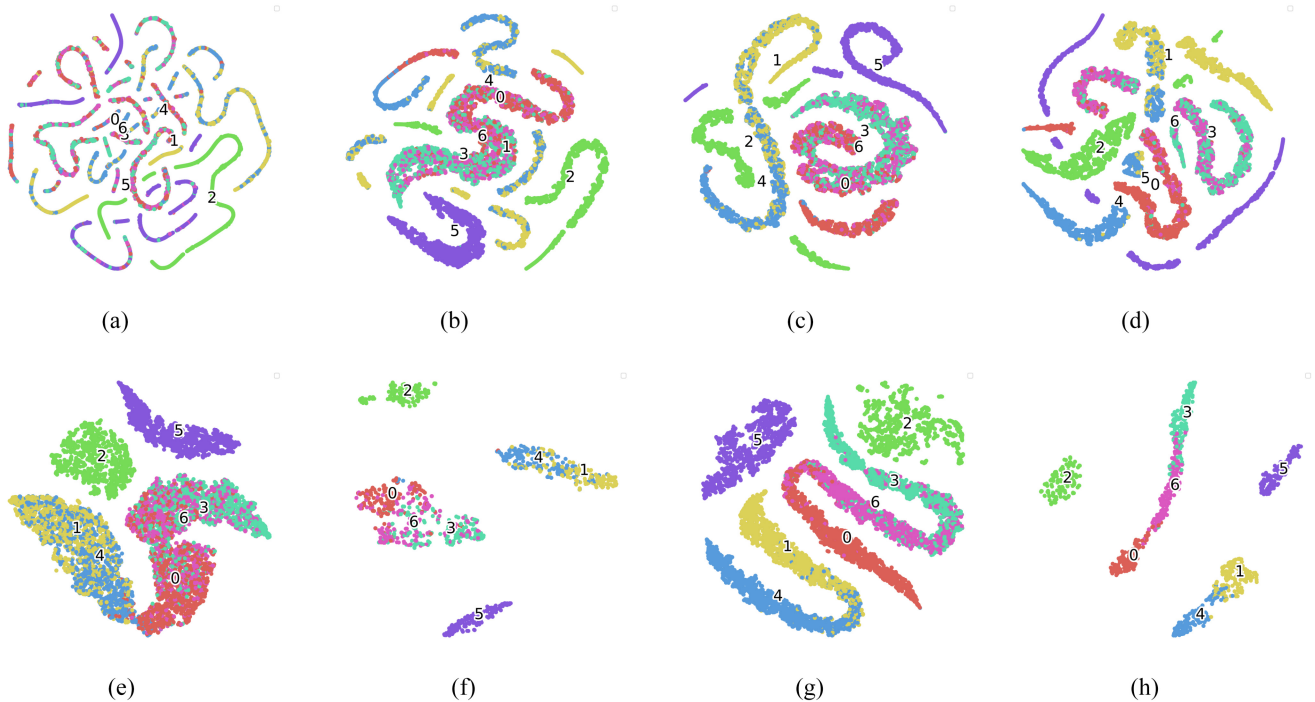


Fig. 13. Feature visualization of SCACNN and other RFF identification method. (a) TripletNet. (b) DA + TripletNet. (c) SiameseNet. (d) DA + SiameseNet. (e) SCL-based. (f) SCL + VAT. (g) SCL + DA. (h) SCACNN.

TABLE VI  
ACCURACY OF DIFFERENT LOSS FUNCTIONS WITH  
THE WEAK NEURAL NETWORK

Data ratio	Softmax	SCL			
		DL1	DL2	SVM	Dot
10%	0.5628	0.6828	0.6789	0.6818	0.6790
20%	0.6049	0.7300	0.7286	0.7282	0.7179
30%	0.6570	0.7893	0.7886	0.7901	0.7857
40%	0.6723	0.8543	0.8518	<b>0.8593</b>	0.8415

TABLE VII  
ACCURACY OF DIFFERENT LOSS FUNCTIONS WITH PUBLIC DATA SET

Time	Data ratio	Benchmark		Rotation	
		Softmax	SCL+SVM	Softmax	SCL+SVM
1	0.5%	0.5297	0.6295	0.5869	0.9344
	1%	0.6401	0.8630	0.8322	0.9656
	5%	0.9492	0.9749	0.9481	0.9919
2	0.5%	0.5115	0.6270	0.6723	0.9325
	1%	0.5944	0.8868	0.7138	0.9818
	5%	0.9121	<b>0.9881</b>	0.9240	<b>0.9969</b>

also has strong generalizability. Here is the performance of our proposed method with the weak neural network and the public data set. As can be seen in Table VI, the SCL-based RFF identification method also outperforms the softmax-based method in the 20-dB SNR environment. From the view of the neural network, compared with the competent neural network, the weak neural network takes more samples to achieve the same accuracy. In addition, the performance of the proposed SCL-based method decrease less than the softmax-based method. Thus, the proposed SCL-based RFF method is more robust to the fitting ability of neural networks. The secondary classifiers shows similar performance as before, while the SVM also achieves the best performance.

The public data set consists of the Wi-Fi signals generated by USRP X310, which is collect with USRP B210 [55]. The data set contains Wi-Fi signals which are collected twice at different distances. In this article the 12 of the Wi-Fi signals collected at a distance of 2 ft are used. The performance of RFF identification with the public data set is show in Table VII. Here, the performance of the benchmark and augmented with *Rotation* are shown. It is obvious

that the SCL-based method shows an excellent improvement, when compared with the softmax-based method. *Rotation* also improves the accuracy of the SCL-based method in all the cases. However, the accuracy of the softmax-based method has a negligible increase after *Rotation*, when the data ratio is 5%. Unlike the performance with our data set, *Shifting* achieves a poor performance, which may because the public data set is composed of specific data structure. This will also appear with the ADS-B data set [56].

### G. Feature Visualization

The feature clustering results of different RFF identification methods with 5% samples are shown in Fig. 13. It is obvious that the proposed SCACNN has the best clustering result. The SCACNN has a clear clustering boundary, while the results of TripletNet and SiameseNet seem a little messy. In addition, the data augmentation has greater improvement with SCL-based method than the TripletNet and SiameseNet. It is worth noting that the VAT shrinks the intraclass distances and expands the interclass distances between feature coding.



## VI. CONCLUSION

In this article, we proposed a novel supervised contrastive loss base RFF identification method, which showed excellent performance in the case of using limited samples. A data augmentation method and a model-based regularization constraints were employed to improve the training model performance. Specifically, the proposed method consists of Data preprocessing and Model fitting. The raw RFF data set is converted into augmented minibatch data set. Then, SimRFE maps the RFF samples to the feature space with SCL and VAT, while different secondary classifiers are employed for RFF identification. Although the distance similarity measurement showed a slightly inferior performance, it is still valuable in future work for its simplicity. The simulation results indicated the proposed SCACNN are effective. Feature visualization also demonstrated that the ability of SCACNN was better than TripletNet and SiameseNet. In addition, the feature visualization indicates that it is necessary to further limit the feature intraclass distances. It is worth noting that the current study still has some open problems. The augmented data set exists redundant signal, which increases the cost of training. Thus, a augmented data set evaluation algorithm is needed to choose the representative samples. In addition, the “anchor” in minibatch is selected randomly, when outlier samples are selected, the performance of RFF identification system will be affected. Hence, these questions are worth to address in our future work.

## REFERENCES

- [1] L. Chettri and R. Bera, “A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems,” *IEEE Internet Things J.*, vol. 7, no. 1, pp. 16–32, Jan. 2020.
- [2] T. B. da Silva, R. P. dos Santos Chaib, C. S. Arismar, R. da Rosa Righi, and A. M. Alberti, “Toward future Internet of Things experimentation and evaluation,” *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8469–8484, Jun. 2022.
- [3] N. Xie, Z. Li, and H. Tan, “A survey of physical-layer authentication in wireless communications,” *IEEE Commun. Surveys Tuts.*, vol. 23, no. 1, pp. 282–310, 1st Quart., 2021.
- [4] J. Cao et al., “A survey on security aspects for 3GPP 5G networks,” *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 170–195, 1st Quart., 2020.
- [5] Y. Liu, M. Dong, K. Ota, and A. Liu, “ActiveTrust: Secure and trustable routing in wireless sensor networks,” *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 2013–2027, 2016.
- [6] P. Angueira et al., “A survey of physical layer techniques for secure wireless communications in industry,” *IEEE Commun. Surveys Tuts.*, vol. 24, no. 2, pp. 810–838, 2nd Quart., 2022.
- [7] J.-R. Jiang, “Short survey on physical layer authentication by machine-learning for 5G-based Internet of Things,” in *Proc. 3rd IEEE Int. Conf. Knowl. Innov. Invent. (ICKII)*, Kaohsiung, Taiwan, Dec. 2020, pp. 41–44.
- [8] Y. Qian, J. Qi, X. Kuai, G. Han, H. Sun, and S. Hong, “Specific emitter identification based on multi-level sparse representation in automatic identification system,” *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 2872–2884, 2021.
- [9] K. Tan, W. Yan, L. Zhang, M. Tang, and Y. Zhang, “Specific emitter identification based on software-defined radio and decision fusion,” *IEEE Access*, vol. 9, pp. 86217–86229, 2021.
- [10] L. Sun, X. Wang, Z. Huang, and B. Li, “Radio-frequency fingerprint extraction based on feature inhomogeneity,” *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17292–17308, Sep. 2022.
- [11] P. Shao and Z. Chen, “Radio frequency fingerprint feature extraction based on IQ data distribution features,” in *Proc. Photon. Electromagn. Res. Symp. (PIERS)*, Hangzhou, China, Apr. 2022, pp. 161–165.
- [12] Y. Xing, A. Hu, J. Zhang, J. Yu, G. Li, and T. Wang, “Design of a robust radio-frequency fingerprint identification scheme for multimode LFM radar,” *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10581–10593, Oct. 2020.
- [13] 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, pp. 1180–1192, 2015.
- [14] W. E. Cobb, E. D. Laspe, R. O. Baldwin, M. A. Temple, and Y. C. Kim, “Intrinsic physical-layer authentication of integrated circuits,” *IEEE Trans. Inf. Forensics Security*, vol. 7, pp. 14–24, 2012.
- [15] J. Zhang, F. Wang, O. A. Dobre, and Z. Zhong, “Specific emitter identification via Hilbert–Huang transform in single-hop and relaying scenarios,” *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 1192–1205, 2016.
- [16] Y. Tu, Z. Zhang, Y. Li, C. Wang, and Y. Xiao, “Research on the Internet of Things device recognition based on RF-fingerprinting,” *IEEE Access*, vol. 7, pp. 37426–37431, 2019.
- [17] Y. Zhang, J. Sun, G. Gui, H. Gacanin, and H. Sari, “A generalized channel dataset generator for 5G new radio systems based on ray-tracing,” *IEEE Wireless Commun. Lett.*, vol. 10, no. 11, pp. 2402–2406, Nov. 2021.
- [18] G. Gui, J. Wang, J. Yang, M. Liu, and J.-L. Sun, “Frequency division duplex massive multiple-input multiple-output downlink channel state information acquisition techniques based on deep learning,” *J. Data Acquisit. Process.*, vol. 37, no. 3, pp. 502–511, May 2022.
- [19] X. Wang, X. Wang, and S. Mao, “Indoor fingerprinting with bimodal CSI tensors: A deep residual sharing learning approach,” *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4498–4513, Mar. 2021.
- [20] Y. Yin, M. Liu, G. Gui, H. Gacanin, H. Sari, and F. Adachi, “Cross-layer resource allocation for UAV-assisted wireless caching networks with NOMA,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3428–3438, Apr. 2021.
- [21] H. Huang, Y. Peng, J. Yang, W. Xia, and G. Gui, “Fast beamforming design via deep learning,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 1065–1069, Jan. 2020.
- [22] H. Huang, G. Gui, H. Gacanin, C. Yuen, H. Sari, and F. Adachi, “Deep regularized waveform learning for beam prediction with limited samples in non-cooperative mmWave systems,” *IEEE Trans. Veh. Technol.*, early access, Feb. 23, 2023, doi: [10.1109/TVT.2023.3248224](https://doi.org/10.1109/TVT.2023.3248224).
- [23] Y. Wang, G. Gui, T. Ohtsuki, and F. Adachi, “Multi-task learning for generalized automatic modulation classification under non-Gaussian noise with varying SNR conditions,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 3587–3596, Jun. 2021.
- [24] S. Huang, C. Lin, W. Xu, Y. Gao, Z. Feng, and F. Zhu, “Identification of active attacks in Internet of Things: Joint model- and data-driven automatic modulation classification approach,” *IEEE Internet Things J.*, vol. 8, no. 3, pp. 2051–2065, Feb. 2021.
- [25] X. Zhang, Y. Wang, Y. Lin, and G. Gui, “A comprehensive survey of deep learning-based automatic modulation recognition methods,” *Radio Commun. Technol.*, vol. 48, no. 4, pp. 697–710, Apr. 2022.
- [26] X. Zhang et al., “NAS-AMR: Neural architecture search-based automatic modulation recognition for integrated sensing and communication systems,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 3, pp. 1374–1386, Sep. 2022.
- [27] K. Merchant, S. Revay, G. Stantchev, and B. Noursain, “Deep learning for RF device fingerprinting in cognitive communication networks,” *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 160–167, Feb. 2018.
- [28] J. Yu et al., “Radio frequency fingerprint identification based on denoising autoencoders,” in *Proc. Int. Conf. Wireless Mobile Comput. Netw. Commun. (WiMob)*, Barcelona, Spain, Oct. 2019, pp. 1–6.
- [29] J. Yu, A. Hu, G. Li, and L. Peng, “A robust RF fingerprinting approach using multisampling convolutional neural network,” *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6786–6799, Aug. 2019.
- [30] X. Fu et al., “Semi-supervised specific emitter identification method using metric-adversarial training,” *IEEE Internet Things J.*, early access, Jan. 27, 2023, doi: [10.1109/JIOT.2023.3240242](https://doi.org/10.1109/JIOT.2023.3240242).
- [31] L. Ding, S. Wang, F. Wang, and W. Zhang, “Specific emitter identification via convolutional neural networks,” *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2591–2594, Dec. 2018.
- [32] D. Roy, T. Mukherjee, M. Chatterjee, and E. Pasilio, “RF transmitter fingerprinting exploiting spatio-temporal properties in raw signal data,” in *Proc. 18th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Boca Raton, FL, USA, Dec. 2019, pp. 89–96.
- [33] 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, Aug. 2021.

- [34] Y. Wang, G. Gui, Y. Lin, H.-C. Wu, C. Yuen, and F. Adachi, "Few-shot specific emitter identification via deep metric ensemble learning," *IEEE Internet Things J.*, vol. 9, no. 24, pp. 24980–24994, Dec. 2022.
- [35] Y. Huang and H. Zheng, "Radio frequency fingerprinting based on the constellation errors," in *Proc. 18th Asia-Pacific Conf. Commun. (APCC)*, Jeju Island, South Korea, Oct. 2012, pp. 900–905.
- [36] 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.
- [37] F. Zhuo, Y. Huang, and J. Chen, "Radio frequency fingerprint extraction of radio emitter based on I/Q imbalance," *Procedia Comput. Sci.*, vol. 107, pp. 472–477, Apr. 2017.
- [38] B. Danev, T. S. Heydt-Benjamin, and S. Capkun, "Physical-layer identification of RFID devices," in *Proc. 18th USENIX Security Symp.*, Montreal, QC, Canada, Aug. 2009, pp. 199–214.
- [39] 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.
- [40] 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, Jan. 2020.
- [41] Y. Lin, Y. Tu, Z. Dou, L. Chen, and S. Mao, "Contour Stella image and deep learning for signal recognition in the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 34–46, Mar. 2021.
- [42] Y. Peng, P. Liu, Y. Wang, G. Gui, B. Adebisi, and H. Gacanin, "Radio frequency fingerprint identification based on slice integration cooperation and heat constellation trace figure," *IEEE Wireless Commun. Lett.*, vol. 11, no. 3, pp. 543–547, Mar. 2022.
- [43] L. Huang, W. Pan, Y. Zhang, L. Qian, N. Gao, and Y. Wu, "Data augmentation for deep learning-based radio modulation classification," *IEEE Access*, vol. 8, pp. 1498–1506, 2020.
- [44] Z. Cai, W. Ma, X. Wang, H. Wang, and Z. Feng, "The performance analysis of time series data augmentation technology for small sample communication device recognition," *IEEE Trans. Rel.*, early access, Jun. 9, 2022, doi: [10.1109/TR.2022.3178707](https://doi.org/10.1109/TR.2022.3178707).
- [45] L. Bertinetto, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, Jul. 2016, pp. 850–865.
- [46] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Vienna, Austria, Jul. 2020, pp. 1597–1607.
- [47] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.
- [48] Y. Wu, Z. Sun, and G. Yue, "Siamese network-based open set identification of communications emitters with comprehensive features," in *Proc. 6th Int. Conf. Commun. Image Signal Process. (CCISP)*, Chengdu, China, Nov. 2021, pp. 408–412.
- [49] G. Sun, "RF transmitter identification using combined siamese networks," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–13, 2021.
- [50] N. Yang et al., "Specific emitter identification with limited samples: A model-agnostic meta-learning approach," *IEEE Commun. Lett.*, vol. 26, no. 2, pp. 345–349, Feb. 2022.
- [51] A. Antoniou, H. Edwards, and A. Storkey, "How to train your MAML," 2018, *arXiv:1810.09502*.
- [52] A. A. M. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Trans. Commun.*, vol. COM-29, no. 11, pp. 1715–1720, Nov. 1981.
- [53] M.-W. Liu and J. F. Doherty, "Specific emitter identification using nonlinear device estimation," in *Proc. IEEE Sarnoff Symp.*, Princeton, NJ, USA, Apr. 2008, pp. 1–5.
- [54] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, South Brisbane, QLD, Australia, Apr. 2015, pp. 4580–4584.
- [55] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "ORACLE: Optimized radio classification through convolutional neural networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Paris, France, Apr./May 2019, pp. 370–378.
- [56] Y. Tu et al., "Large-scale real-world radio signal recognition with deep learning," *Chin. J. Aeronaut.*, vol. 35, no. 9, pp. 35–48, Sep. 2022.