Semi-Supervised Specific Emitter Identification Method Using Metric-Adversarial Training

Xue Fu, Graduate Student Member, IEEE, Yang Peng, Graduate Student Member, IEEE, Yuchao Liu, Yun Lin, Member, IEEE, Guan Gui, Senior Member, IEEE, Haris Gacanin, Fellow, IEEE, and Fumiyuki Adachi, Life Fellow, IEEE

Abstract—Specific emitter identification (SEI) plays an increasingly crucial and potential role in both military and civilian scenarios. It refers to a process to discriminate individual emitters from each other by analyzing extracted characteristics from given radio signals. Deep learning (DL) and deep neural networks (DNNs) can learn the hidden features of data and build the classifier automatically for decision making, which have been widely used in the SEI research. Considering the insufficiently labeled training samples and large-unlabeled training samples, the semi-supervised learning-based SEI (SS-SEI) methods have been proposed. However, there are few SS-SEI methods focusing on extracting the discriminative and generalized semantic features of radio signals. In this article, we propose an SS-SEI method using metric-adversarial training (MAT). Specifically, pseudo labels are innovatively introduced into metric learning to enable semi-supervised metric learning (SSML), and an objective function alternatively regularized by SSML and virtual adversarial training (VAT) is designed to extract discriminative and generalized semantic features of radio signals. The proposed MAT-based SS-SEI method is evaluated on an open-source large-scale real-world automatic-dependent surveillance–broadcast (ADS-B) data set and Wi-Fi data set and is compared with the state-of-the-art methods. The simulation results show that the proposed method achieves better identification performance than existing state-of-the-art methods. Specifically, when the ratio of the number of labeled training samples to the number of all training samples is 10%, the identification accuracy is 84.80% under the ADS-B data set and 80.70% under the Wi-Fi data set. Our code can be downloaded from https://github.com/lovelymimola/MAT-based-SS-SEI.

Index Terms—Alternating optimization, deep metric learning, semi-supervised learning (SSL), specific emitter identification (SEI), virtual adversarial training.

I. INTRODUCTION

Advances in the Industrial Internet of Things (IIoT) have contributed to the development of Industry 4.0 and IIoT establishes internal communication using multiple types of wireless terminals [1], [2], which raises critical security issues in IIoT. Specific emitter identification (SEI) has been considered as one of the efficient techniques for IIoT security by authenticating the identity of a transmitting device [3], [4], [5], [6], [7]. SEI also plays an increasingly important role in cognitive radio (CR) networks [8] and it can be regarded as a countermeasure against attackers that disguise themselves as primary users and occupy a licensed part of the spectrum and cause a denial-of-service attack for secondary users in CR [9].

SEI refers to a process to discriminate individual emitters from each other by analyzing extracted characteristics of the received radio signals [10]. Extracted characteristics also known as the radio frequency fingerprints (RFFs) that are originated from the imperfections of the analog components of emitters. RFFs are unique to each other and hard to be reproduced [9]. The SEI methods can be divided into two types that are the transient signals-based SEI methods and the steady-state signals-based SEI methods. Initially, the RFF-based SEI methods were focusing on identification of transient signals [12], [13]. The transient signals-based SEI methods use the transition from the turn-off to the turn-on of an emitter that is occurred before the transmission of the actual data of a signal. Therefore, a higher sampling rate is required to extract the transient signal due to its short period. Also, the reliability of the phase and amplitude information is a serious challenge in this area [14]. In addition, channel noise significantly affects transient signals more than steady-state signals when using radio signals for SEI [15]. On the other hand, steady-state signals refer to the modulated parts of the signals transmitted by an emitter at a steady power, and nowadays most of the RFFs-based SEI methods are based on steady-state signals.

A typical RFFs-based SEI approach operates in two stages in which the first stage is capturing the signals and the
The second stage is extracting proper characteristics from captured signals and identifying them [16]. Most of the existing SEI methods based on RFF only focus on the second stage on the prerequisite that signals have been captured. Feature extraction and identification of signals is considered as an important stage in the RFF-based SEI methods. Deep learning (DL) and deep neural networks (DNNs) can learn the hidden features of data and build the classifier automatically for decision making in many successful applications [17], [18], [19], [20], [21], [22], [23], [24]. Because of this feature, many SEI methods combined time-domain complex baseband signals with DNN [9], [25], [26]. To further improve the identification performance, these methods employed signals in transform domain and DNN, such as bispectrum [27], the Hilbert–Huang transform [28], constellation [29], and so on. Specifically, Chen et al. [25] used an inception-residual neural network to classify large-scale real-world radio aircraft communications addressing and reporting system (ACARS) and automatic-dependent surveillance–broadcast (ADS-B) signal data with categories of 3143 and 5757, respectively. Wang et al. [26] used a complex-valued neural network (CVNN) with compression to classify seven power amplifiers (PAs). Merchant et al. [9] presented a convolutional neural network (CNN) using time-domain complex baseband error signal for seven ZigBee devices identification. Ding et al. [27] adopted a CNN to identify five universal software radio peripherals (USRPs) using the compressed bispectrum of the received signal. Pan et al. [28] presented a deep residual network (ResNet) to identify five PAs using the Hilbert–Huang spectrum of received signal. Peng et al. [29] identified seven PAs using heat constellation trace figure (HCTF) and DNNs.

The success of DL and DNNs often hinges on the availability of a sufficient number of labeled training samples, as shown in Table I(a), where thousands of samples were labeled to train the DNNs [30]. However, in practical SEI tasks, annotation of radio signals is quite expensive, resulting in impossibility of training the DNNs adequately. An attractive approach toward mitigating insufficiently labeled radio signals is semi-supervised learning (SSL) which makes full use of the information embedded in both labeled radio signals and unlabeled radio signals to approach similar performance to that of the well-trained counterpart.

In this article, we propose an SSL-based specific emitter identification (SS-SEI) method using metric-adversarial training (MAT). Specifically, a well-designed object functions that is cross-entropy (CE) loss alternatively regularized by semi-supervised metric learning (SSML) or virtual adversarial training (VAT), where the novel SSML is used to extract the discriminative semantic features of radio signals using the Euclidean distance or cosine similarity, and VAT is used to extract the generalized semantic features of radio signals. The main contributions of this article are summarized as follows.

1) We present the MAT-based SS-SEI method, where VAT is used to extract the generalized semantic features of radio signals, and SSML is used to extract the discriminative semantic features of radio signals. VAT and SSML are alternatively used as the regularization term of the objective function, which has a better identification performance and faster convergence rate than simultaneous way.

2) We innovatively introduce the pseudo labels into metric learning (ML), which enables the ML to work for both labeled and unlabeled radio signals on semantic feature space. In addition, this trick is metric-agnostic and we verified the effectiveness on center loss (CL) and proxy anchor loss.

3) The proposed SS-SEI method is evaluated on an open-source large-scale real-world ADS-B data set and an open-source Wi-Fi data set and is compared with the four latest SS-SEI methods. The simulation results show that the proposed SS-SEI method achieves state-of-the-art identification performance.

II. RELATED WORK

In this review, we focus on methods closely related to the MAT-based SS-SEI method. MAT is an SSL framework containing a variety of semi-supervised principles, such as consistency regularization and pseudo-labels, which is suitable for SEI. In addition, ML is another important factor of MAT’s success. Therefore, related works about SSL-based signal identification and ML-based signal identification are reviewed in this article.

A. SSL-Based Signal Identification Methods

In this article, the framework of SSL is divided into consistency regularization-based framework, entropy minimization-based framework, pseudo-label-based framework, unsupervised component-based framework, such as autoencoder and generative adversarial network (GAN), and the hybrid framework, such as FixMatch [31], that is a combination of consistency regularization and pseudo-label.

There are many researchers who made effort on signal identification methods based on the above SSL frameworks. For example, Xie et al. [32] proposed an SS-SEI method-based bispectrum analysis and CNN with VAT [54] to identify six USRPs. Gong et al. [33] presented a quadruple-structured framework-based SS-SEI method to identify multiple emitters, including PAs, shortwave stations, ultrashortwave stations, and Wi-Fi devices, where the framework consisted of an autoencoder and a Triple-GAN [34]. Wang et al. [37] proposed a convolutional autoencoder for SSL-based modulated signal classification. Tan et al. [38] introduced a GAN using bispectrum of the signal as inputs for SS-SEI and analyzed the identification performance on 12 emitters constructed by six USRPs with six modulated type. Ren et al. [39] proposed an SS-SEI method based on ResNet18 and meta pseudo-labels [40], where by using the time-frequency grayscale image with short-time Fourier transform (STFT) as the input of ResNet18 and the SS-SEI method was evaluated on data set of 15 mobile phones. Medaiyese et al. [15] presented a hierarchical learning framework for unmanned aerial vehicles (UAVs)
TABLE I
RELATED WORKS. (A) CONVENTIONAL DL AND DNNs-BASED SIGNAL IDENTIFICATION METHODS. (B) RELATED SSL-BASED SIGNAL IDENTIFICATION METHODS. (C) RELATED ML-BASED SIGNAL IDENTIFICATION METHODS

(a)

| Conventional Methods | DNN | Emitter Type | Sample Format | Number of Samples | Performance |
|----------------------|-----|--------------|---------------|-------------------|-------------|
| Chen et al. [25]     | Inception-residual neural network | 3,143 ACARS and 5,157 ADS-B | IQ | 990,000 and 13,000,000 | SNR > 9 dB, $P_{cc} > 92\%$ |
| Wang et al. [26]     | CVNN | 7 PAs | IQ | more than 32,000 per device | SNR > 10 dB, $P_{cc} > 75\%$ |
| Merchant et al. [9]  | CNN | 7 ZigBee | IQ with error | 1,000 per device | SNR > 10 dB, $P_{cc} > 92.70\%$ |
| Ding et al. [27]     | CNN | 5 USRPs | Compressed bispectrum | 0 to 30 dB, 300 from one device at each SNR | SNR > 10 dB, $P_{cc} > 91.8\%$ |
| Pan et al. [28]      | Deep ResNet | 5 PAs | Hilbert-Huang spectrum | $\{10, 12, 14, 16, 18, 20, 22, 24\}$ dB, 5,000 per SNR | SNR > 10 dB, $P_{cc} > 62\%$ |
| Peng et al. [29]     | InceptionV3 ResNet50, Xception | 7 PAs | Heat constellation trace figure | 160 to 260 per device | SNR > 0 dB, $P_{cc} > 91.07\%$ |

(b)

| Related Works | SSL Framework | Emitter Type | Sample Format | Number of Samples | Performance |
|---------------|---------------|--------------|---------------|-------------------|-------------|
| Xie et al. [32] | CNN and VAT | 6 USRPs | Bispectrum distribution | 20,000 per emitter at a specific SNR | $R_1$ is 10%, $P_{cc} > 90.80\%$ |
| Gong et al. [33] | TripleGAN combined with autoencoder | 5 shortwave stations, 5 PAs | IQ | 10,000 per emitter | $R_1$ is 10%, $P_{cc} > 90.50\%$, $P_{cc} > 94.90\%$, $P_{cc} > 96.20\%$ and $P_{cc} > 96.50\%$, respectively |
| Wang et al. [37] | Convolutional autoencoder | 4 modulations | IQ | 20,000 per modulation at a specific SNR | $R_1$ is 5%, $P_{cc} > 90.65\%$ |
| Tan et al. [38] | CGAN | 12 emitters constructed by 6 USRPs | Bispectrum | 1,000 per emitter per modulation | $R_2$ is 70%, $P_{cc} > 90\%$ |
| Ren et al. [39] | ResNet18 and Meta Pseudo Labels | 15 mobile phones | Time-frequency grayscale image | 900 slices per model phone | $R_3$ is 1%, $P_{cc} = 91.20\%$ |
| Medaiye et al. [15] | Denoising autoencoder and local autoencoder | UAV and non-UAV (bluetooth and Wi-Fi) | Hilbert-Huang and wavelet packet transform | 234,500 slices | $R_2$ is 10%, $P_{cc} > 84.10\%$ |

$R_1$ denotes the number of labeled training samples to the number of unlabeled training samples ratio. $R_2$ denotes the number of labeled training samples to the number of all training samples ratio.

(c)

| Related Works | DNN and Metric Loss | Emitter Type | Sample Format | Number of Samples | Performance |
|---------------|---------------------|--------------|---------------|-------------------|-------------|
| Dong et al. [42] | CNN and Center Loss | 11 modulations | IQ | ranging from 207 to 1,248 per modulation | SNR > 8 dB, $P_{cc} > 86.79\%$ |
| Shen et al. [43] | ResNet and Triplet Loss | 10 LoRaS | Channel independent spectrogram of preamble | 500 per device (pretraining) and 100 per device (retraining) | $P_{cc} > 98\%$ |
| Gong et al. [45] | CNN and Circle Loss | 10 ISM devices | IQ | 10 million per device | SNR > 10 dB, $P_{cc} > 94\%$ |
| He et al. [47] | DNN and Triplet Loss | 11 and 6 ship-radiated noise | 61 acoustic features | Not mentioned | SNR > 10 dB, $P_{cc} > 90\%$, respectively |
| Wang et al. [48] | CVNN, Triplet Loss and Center Loss | 90 ADS-B in pretraining, 30 ADS-B in fine-tuning | IQ | 200-500 samples per aircraft in pretraining, 1-20 samples per aircraft in fine-tuning | $P_{cc} > 90\%$ in one-shot |

Detection and identification, where an SSL-based UAVs detection method based on denoising autoencoder and local outlier factor [41] was introduced. The details of the above literatures are shown in Table I(b).

In different SSL-based signal identification methods, unlabeled training data set participates in training process of DNNs in different ways, which brings different performance benefits. Specifically, method [32] had strong anti-noise performance due to training with VAT, and methods [15], [33], [37] had strong capability to extract key features because of the utilization of autoencoder. However, the discrimination of features merely brought by softmax with CE loss and reconstruction loss is limited. DML is one of the solutions to improve discrimination of features.

B. ML-Based Signal Identification Methods

ML aims to train a DNN that makes tighter and clearer decision boundaries. It can be categorized into pair-based ML and proxy-based ML. The pair-based ML is built upon pairwise distances between samples in the semantic space, such as triple loss [44], circle loss [46], and so on. In the proxy-based ML,
each sample is encouraged to be close to proxies of the same category and far apart from those of different categories, such as center loss [49] and proxy-anchor (PA) loss [50], where the proxies are representative of a subset of training data set and learned as a part of the network parameters.

There are many researchers who made effort on signal identification methods based on ML or the combination of ML and SSL framework. For example, SSRCNN [42] was an SSL-based modulated signal classification method which consisted of a neural network and a sophisticated design of loss functions, where the loss function consisted of center loss, CE loss, and Kullback–Leibler divergence loss. Shen et al. [43] exploited channel-independent spectrograms of preambles as inputs and a lightweight ResNet as an RFF extractor to detect the rogue LoRa devices and classify the legitimate LoRa devices, where the RFF extractor was optimized by triplet loss [44]. Gong et al. [45] used circle loss [46] to optimize a CNN feature extractor and further identify the signals emitted from ten different ISM devices. He et al. [47] proposed triplet loss which was different from the triplet loss of [44] and DNN to classify ship-radiated noise. Wang et al. [48] presented a well-designed objective function composed of triplet loss and CL for a discriminative feature embedding and further identified aircrafts in few-shot scenarios. The details of the above literatures are shown in Table I(c).

Most of the ML-based signal identification methods utilize a similarity measure in a fully supervised way or directly combine ML with SSL framework where the ML works for labeled samples and the SSL framework works for unlabeled samples or all training samples such as SSRCNN [42]. Intuitively, there are amount of information embedded in unlabeled samples worth being learned by ML. However, the lack of labels makes these information inaccessible for ML.

The shortcomings of the above-related works can be summarized as limited feature discrimination and information inaccessible of unlabeled samples in ML. To solve these problems, the MAT-based SS-SEI method is proposed in this article.

III. SIGNAL MODEL AND PROBLEM FORMULATION

A. Signal Model

Only one receiver is employed for an SEI application to capture possible radio signal from a certain interested space. K emitters are considered to be activated at a time and it is assumed that the radio signals from each emitters can be captured individually. The received radio signal for the kth emitter can be formulated as

\[ r_k(t) = s_k(t) * h_k(t) + n_k(t), \quad k = 1, 2, \ldots, K \]  

where \( r_k(t) \) is the received radio signal, \( s_k(t) \) is the transmitted radio signal, \( h_k(t) \) stands for the channel impulse response between transmitter and receiver, \( n_k(t) \) denotes an additive white Gaussian noise, and * means the convolution operation.

B. Problem Formulation

Let \( \mathcal{R} \) and \( \mathcal{Y} \) be the sample space and category space, respectively. \( \mathbf{r}_k \in \mathcal{R} \) represents the input sample, which is a signal sample from one emitter or with IQ format; \( y \in \mathcal{Y} \) denotes the real category of the corresponding emitter.

1) SEI Problem: Considering a general machine learning-based SEI problem and a training data set \( D = \{(r_k, y_k)\}_{k=1}^K \), the goal of the problem is to produce a mapping function \( f \in \mathcal{F} : \mathcal{R} \rightarrow \mathcal{Y} \) and its expected error is minimized, i.e.,

\[ \min_{f \in \mathcal{F}} \mathbb{E}_{(r, y) \sim D} \mathcal{L}(f(r), y) \]  

where \( \mathcal{L}(f(r), y) \) stands for the loss that compares the prediction \( f(r) \) to its ground-truth category. The expected error, however, is approximated by

\[ \min_{f \in \mathcal{F}} \mathbb{E}_{(r, y) \sim D} \mathcal{L}(f(r), y) \]  

because the joint distribution \( P_{R \times Y} \) is unknown. Therefore, the generalization error \( \varepsilon = |\mathbb{E}_{em} - \mathbb{E}_{ex}| \) must be considered to prevent overfitting. Equation (3) can be rewritten as

\[ \min_{f \in \mathcal{F}, s.t. f(r_i) = y_i \forall (r_i, y_i) \in D_l}, \]  

More supervised samples contained in \( D_l \) will bring more constraints on \( f \) and then it will bring a good generalization.

2) Semi-Supervised SEI Problem: In the semi-supervised SEI problem, the training data set is \( D_l = \{(r_1, y_1), \ldots, (r_L, y_L), (r_{L+1}, \ldots, r_Y)\} \), where there are \( L \) labeled training samples and \( u = N - L \) unlabeled training samples. For the convenience of discussion, we use \( D_l = \{(r'_i, y'_i) | i = 1, \ldots, L\} \) to denote a labeled training data set, and \( D_{ul} = \{r'_i | i = 1, \ldots, N - L\} \) to denote an unlabeled data set, The relationship between \( D_l, D_i, D_{ul} \) is \( D_l = D_i \cup D_{ul} \). Usually, \( N - L \gg L \).

Considering a general machine learning-based SS-SEI problem, the goal of the SS-SEI problem is also to produce a mapping function \( f \in \mathcal{F} : \mathcal{R} \rightarrow \mathcal{Y} \) and its expected error (2) is minimized. Due to the limitation of labeled training data set and additional information on the data distribution from unlabeled training data set, the expected error, however, is approximated by

\[ \min_{f \in \mathcal{F}} \mathbb{E}_{(r, y) \sim D} \mathcal{L}(f(r), y) + \mathbb{E}_{(r) \sim D_{ul}} \mathcal{L}_{ul}(\ast) \]  

where \( \mathcal{L}_{ul}(\ast) \) stands for the loss that takes into account the unlabeled training data set to have a more accurate prediction. Reconstruct loss and Kullback–Leibler divergence can be used as \( \mathcal{L}_{ul}(\ast) \). There is an important prerequisite that the distribution of samples, which the unlabeled training data set will help elucidate, are relevant for the SEI problem.

IV. PROPOSED MAT-BASED SS-SEI METHOD

In this section, we present the framework of MAT-based SS-SEI in Section IV-A and its training procedure in Section IV-B. Section IV-A shows the overview of the framework and describes in detail the functions and benefits of each component of the proposed MAT for SS-SEI. Section IV-B shows the training procedure of the proposed MAT-based SS-SEI method.
The standard CE loss can be formulated as:

\[ L_{CE} = - \frac{1}{L} \sum_{i=1}^{L} \log q_{y_i}^{\hat{r}_i} \]  

where \( q_{y_i}^{\hat{r}_i} \) is predicted class distribution by the CVNN for \( r_i \) and \( q_{y_i}^{\check{r}_i} \) is the \( y_i \)th value of \( q_{y_i}^{\hat{r}_i} \).

In this article, we introduce semi-supervised cross-entropy (SS-CE) loss to learn the information embedded in unlabeled samples, which can be formulated as:

\[
L_{CE}^s = - \frac{1}{L} \sum_{i=1}^{L} \log q_{y_i}^{\hat{r}_i} \\
- \frac{1}{N-L} \sum_{j=1}^{N-L} \mathbb{1} \left( \max \left( \{ q_{y_i}^{\check{r}_i} \} > \tau \right) \log q_{y_i}^{\check{r}_i} \right)
\]

where \( y_i \) is the pseudo label of \( r_i \), and \( q_{y_i}^{\check{r}_i} \) is predicted class distribution by the CVNN for \( r_i \), and \( \tau \) is the confidence threshold. \( q_{y_i}^{\check{r}_i} \) and \( q_{y_i}^{\hat{r}_i} \) can be further formulated as:

\[
q_{y_i}^{\check{r}_i} = \text{softmax} \left( f \left( g \left( r_i \right) \right) \right) \\
q_{y_i}^{\hat{r}_i} = \arg \max_k \left( q_{y_i}^{\check{r}_i} \right)
\]

where \( g(\cdot) \) is the feature extractor of CVNN and \( f(\cdot) \) is the classifier of CVNN.

3) Discriminative Semantic Features Extraction: Building on the foundation of classification backbone, the CVNN in MAT incorporates ML such that the CVNN is trained to project input samples onto embedding space in which semantically similar features (i.e., radio signals of the same category) are closely grouped together. Therefore, more discriminative semantic features are extracted compared to CVNN trained only with CE loss. It is worthwhile to point that we innovatively introduce the pseudo labels into ML so that the ML can work for both labeled and unlabeled radio signals on semantic feature space and the CVNN in MAT incorporates the SSML for discriminative semantic feature extraction. The proposed SSML is metric-agnostic and the principle of SSML is explained by CL [49] and PA loss [50].

In this article, the semantic features are obtained from the features extractor of CVNN, namely, the output of LazyLinear (1024) or LazyLinear (128) in Table II, by which the standard CL can be formulated as:

\[
L_{Center} = \frac{1}{2L} \sum_{i=1}^{L} \left\| g(r_i) - c_{y_i}^l \right\|^2
\]

where \( g(\cdot) \) represents the feature extractor, \( g(r_i) \) denotes the semantic features of radio signal sample \( r_i \), and \( c_{y_i}^l \) represents the trainable semantic center features of category \( y_i \). For each category of radio signals, the CL simultaneously learns a center of semantic features and penalizes the Euclidean distances between its semantic features and the corresponding center. In addition, the standard PA loss can be formulated as:

\[
L_{PA} = \frac{1}{|P^+|} \sum_{p \in P^+} \log \left( 1 + \sum_{r_i \in R_i^+} e^{-\alpha(s(g(r_i), p)-\delta)} \right) \\
+ \frac{1}{|P^-|} \sum_{p \in P^-} \log \left( 1 + \sum_{r_i \in R_i^-} e^{\alpha(s(g(r_i), p)+\delta)} \right)
\]
where $\delta > 0$ represents a margin, and $\alpha > 0$ denotes a scaling factor, and $P$ is the set of all proxies, and $P^+$ stands for the set of positive proxies of semantic features, and $P^-$ stands for the set of negative proxies of semantic features, and $s(\cdot, \cdot)$ denotes the cosine similarity between two features, and $R_i^p$ denotes the set of positive labeled samples of $p$, and $R_i^n = R_i - R_i^p$ denotes the set of negative labeled samples of $p$.

Building on the foundation of standard ML, we introduce the pseudo labels into standard ML. For the CL, the semi-supervised center loss (SS-CL) can be formulated as

$$L_{\text{Center}}^C = \frac{1}{2L} \sum_{i=1}^L \left[ g(r_i) - c_{y_i}^2 \right]$$

$$+ \frac{1}{2(N-L)} \sum_{j=1}^{N-L} \left[ \max\left(q_{ulj}\right) > \tau \right] \left[ g(r_i^j) - c_{y_i}^2 \right].$$

(12)

In the same way, the semi-supervised proxy anchor (SS-PA) loss can be formulated as

$$L_{\text{PA}}^s = \frac{1}{|P^+|} \sum_{p \in P^+} \log \left( 1 + \sum_{r_i \in R_i^+} e^{-\alpha \delta r_i(p_i - \delta)} \right)$$

$$+ \frac{1}{|P|} \sum_{p \in P^-} \log \left( 1 + \sum_{r_i \in R_i^-} e^{\alpha \delta r_i(p_i + \delta)} \right)$$

$$+ \frac{1}{|P|^+} \sum_{p \in P^+} \log \left( 1 + \sum_{r_i \in R_i^+} \max\left(q_{ul} \right) > \tau \right) e^{-\alpha \delta r_i(p_i - \delta)}$$

$$+ \frac{1}{|P|} \sum_{p \in P^-} \log \left( 1 + \sum_{r_i \in R_i^-} \max\left(q_{ul} \right) > \tau \right) e^{\alpha \delta r_i(p_i + \delta)}.$$  

(13)

Intuitively, the SS-CE loss forces the semantic features of different categories staying apart roughly. The SS-CL efficiently pulls the semantic features of the same categories to their center using the Euclidean distance. The SS-PA loss enlarges the intercategory difference but also reduces intracategory variations using cosine similarity. With the joint loss function of CE loss and SS-CL or SS-PA loss, the CVNN is trained to obtain the semantic features with intercategory dispersion and intracategory compactness as much as possible. The objective function regularized by SSML for discriminative semantic features extraction can be formulated as

$$L_2 = \omega_1 L_{\text{CE}} + \omega_2 L_{\text{SSML}}$$

(14)

where $L_{\text{SSML}}$ is $L_{\text{Center}}^C$ or $L_{\text{PA}}^s$, and scalars $\omega_1$ and $\omega_2$ are used for balancing the two loss terms. Different scalars lead to different semantic features distributions. With proper scalars, the discrimination of semantic features can be significantly improved. In this article, we use the automatic weight [51] to get rid of the manual tuning of scalars.

4) Generalized Semantic Features Extraction: In practice, the evaluation of the objective function will always be an empirical approximation over the sample space as illustrated in (3), and however, the number of the samples that can be used to tune the parameters of model is finite, especially in semi-supervised scenarios. Therefore, even with successful optimization and low-training error, the testing error can be large in the SS-SEI, that is, the generalization performance of model are not sufficient. It is known that the generalization performance of DNNs can be improved by applying random perturbations to samples to generate artificial samples and encouraging the DNNs to assign a similar output to the set of artificial samples derived from the same samples [52]. Adversarial training [53] is one of the successful attempts that improve generation performance by applying random perturbations. VAT [54] is an improved adversarial training which can be applied to the SSL, and we use VAT to achieve generalized semantic features extraction in our SS-SEI method.

The VAT defines the local distributional smoothness (LDS) to be the divergence-based distributional robustness of the model against the virtual adversarial direction, and LDS can be formulated as

$$\text{LDS}(r_s, \theta) = D[f(y|\hat{r}_s, \hat{\theta}), f(y|r_s + p_{\text{adv}}, \theta)]$$

(15)

$$p_{\text{adv}} = \arg \max_{p :||p||_2 < \epsilon} D[f(y|\hat{r}_s, \hat{\theta}), f(y|\hat{r}_s + p, \theta)].$$

(16)

where $r_s$ represents either $r_i$ or $u_{il}$. $D[f, f']$ is Kullback–Leibler divergence in our SS-SEI method, $\theta$ stands for the current model weight, $f(y|\hat{r}_s, \hat{\theta})$ is the current estimation of true distribution of the output label of $r_s$, $f(y|r_s + p, \theta)$ is the current estimation of distribution of the output label of $r_s$ with virtual adversarial perturbation, and $p_{\text{adv}}$ is virtual adversarial perturbation which can be approximated by

$$p_{\text{adv}} \approx \epsilon \frac{g}{||g||_2}$$

(17)

where $g = \nabla_{r_s} D[f(y|\hat{r}_s, \hat{\theta}), f(y|\hat{r}_s + p, \theta)]$ and $\epsilon$ represents perturbation intensity. The gradient $\nabla_{r_s} D[f(y|\hat{r}_s, \hat{\theta}), f(y|\hat{r}_s + p, \theta)]$ can be efficiently computed by backpropagation of the CVNN.

The SS-CE loss is used as the classification backbone and the LDS is used as a way for enhancing the generalization performance of model. The objective function regularized by VAT for generalized semantic features extraction can be formulated as

$$L_1 = \omega_1 L_{\text{CE}} + \omega_2 L_{\text{VAT}}$$

(18)

$$L_{\text{VAT}} = \frac{1}{N_l + N_{ul}} \sum_{r_s \in D_l, D_{ul}} \text{LDS}(r_s, \theta).$$

(19)

where scalars $\omega_1$ and $\omega_2$ are used for balancing the two loss terms. In this article, we use the automatic weight [51] to get rid of the manual tuning of scalars.

B. Training Procedure

The full training procedure with object function is described in Algorithm 1. Alternative optimization is used during training. Specifically, the objective function regularized by VAT for generalized semantic features extraction, that is (18), is
Algorithm 1: Training Procedure of MAT-Based SS-SEI

Require:
- \( D_l, D_u \): Labeled and unlabeled training dataset, respectively;
- \( T \): Number of training iterations;
- \( B \): Number of batches in a training iteration;
- \( \theta_m, \theta_a \): Parameters of CVNN, trainable semantic center features of center loss or trainable proxies of proxy-anchor loss, respectively;
- \( l_r m, l_r a \): Learning rate of CVNN, center loss or proxy-anchor loss, respectively;
- \( z_l, z_u \): Semantic features of labeled training samples and unlabeled training samples, respectively;
- \( q_l, q_u \): Predicted class distribution of labeled training samples and unlabeled training samples, respectively;
- \( \omega_1, \omega_2, \omega_3, \omega_4 \): Scalars for balancing the loss terms;

Dataset preprocessing:
\[
D_l \leftarrow \frac{D_l - \min(D_l)}{\max(D_l) - \min(D_l) + D_u};
\]
for \( t = 1 \) to \( T \) do
  for \( b = 1 \) to \( B \) do
    Sample a batch of labeled training samples \((r_l, y_l)\).
    Sample a batch unlabeled training samples \((r_u)\).
    if \( b \equiv 2 \mod 2 \) then
      Forward propagation:
      Extracting the semantic features:
      \[
z_l, z_u = g(\theta_l^{b, i}, \theta_a^{b, i}, r_l, r_u);
      \]
      Predicting the class distribution:
      \[
      \hat{q}_{l}, \hat{q}_{u} = f(\hat{q}_l^{b, i}; z_l, z_u);
      \]
      Calculating the loss:
      \[
      L_2 = \omega_2 L_{SSML}((z_l, z_u), \theta_a) + \omega_1 L_{SSML}((z_l, z_u), \theta_a);
      \]
      Backward propagation:
      Updating the parameters of trainable proxies:
      \[
      \theta_a^{b, i+1} \leftarrow \text{Adam}(\nabla_{\theta_a} L_2, l_r a, \theta_a);
      \]
    else
      Forward propagation:
      Extracting the semantic features:
      \[
z_l, z_u = g(\theta_l^{b, i}, \theta_a^{b, i}, r_l, r_u);
      \]
      Predicting the class distribution:
      \[
      \hat{q}_{l}, \hat{q}_{u} = f(\hat{q}_l^{b, i}; z_l, z_u);
      \]
      Calculating the loss:
      \[
      L_1 = \omega_1 L_{CE}((\hat{q}_l, \hat{q}_u), (y_l, y_u)) + \omega_2 L_{SSML}((\hat{r}_l, \hat{r}_u));
      \]
      Backward propagation:
      Updating the parameters of CVNN:
      \[
      \theta_m^{b, i+1} \leftarrow \text{Adam}(\nabla_{\theta_m} L_1, l_r m, \theta_m);
      \]
  end
end

operated when \( t \in \{1, 3, 5, \ldots, T - 1\} \), and the objective function regularized by SSML for discriminative semantic features extraction, that is (20), is operated when \( t \in \{2, 4, 6, \ldots, T\} \).
performance of the SS-SEI method but also evaluates whether the SS-SEI methods have the ability to handle real environment. The identification performance of our MAT-based SS-SEI method and comparative methods under ADS-B and Wi-Fi data set are shown in Table III. We observe the clear superiority of our SS-SEI method over comparative methods under the ADS-B data set and Wi-Fi data set. For the ADS-B data set, when the number of labeled training samples to the number of all training samples ratio is 10%, the identification accuracy of the comparative methods is more than 60% but less than 80%, while our SS-SEI method can reach more than 80%. For the Wi-Fi data set, when the number of labeled training samples to the number of all training samples ratio is 10%, the identification accuracy of comparative methods is more than 20% but less than 50%, while our SS-SEI method can reach more than 80%.

**E. Visualization of Semantic Features MAT Versus Comparative Methods**

A well-designed objective function is presented to extract the discriminative and generalized semantic features in this article, where the objective function is alternatively regularized by SSML and VAT during training. The dimensionality of the extracted semantic features is reduced to two dimensions by $t$-distributed stochastic neighbor embedding ($t$-SNE) [59] for visualization, as shown in Fig. 2. We only show the visualization under the Wi-Fi data set and the ratio is 10% because of the limited space, and the visualization of other scenarios can be seen in our GitHub (https://github.com/lovelymimola/MAT-based-SS-SEI).

It can be observed that the semantic features of different categories extracted by CVNN stay apart roughly because CVNN is merely optimized by CE loss in a fully supervised way and the data distribution information included in the limited labeled training samples is insufficient. DRCN, SSRCNN, Triple-GAN, and SimMIM not only use the data distribution information included in the limited labeled training samples but also use the data distribution information included in the large-unlabeled training samples, and therefore the semantic features are more discriminative than that of the CVNN. We also observe the clear superiority of our MAT-based SS-SEI method over comparative methods in visualization of semantic features.
features. Specifically, the semantic features with the intercategory dispersion and intracategory compactness are obtained by our proposed MAT-based SS-SEI method.

F. Training Time MAT Versus Comparative Methods

The proposed MAT is essentially a well-designed loss function and novel training strategy, which can be used in a variety of DNNs to identify different radio signals or emitters, and the CVNN is used to verify the effectiveness of MAT in this article. Therefore, we do not analyze the model complexity of the MAT-based SS-SEI method, and we analyze the average time per-iteration of training process as shown in Table IV. It can be observed that the average time per-iteration of DRCN and TripleGAN is more than that of CVNN, SimMIM, MAT-CL, and MAT-PA because the structure of DRCN contains not only encoder and classifier but also decoder, and TripleGAN contains not only encoder and classifier but also decoder, generator, and discriminator. Therefore, the DRCN-based SEI method and TripleGAN-based SEI method take extra time to train the decoder, generator, and discriminator. The structure of neural network of SimMIM contains encoder, decoder, and classifier, but the decoder is lightweight and the computational complexity does not increase sharply. Although the structure of neural network of MAT-CL and MAT-PA is same as the CVNN, the objective function of MAT-CL and MAT-PA is more complex than CVNN and thus the average time per-iteration of MAT-CL and MAT-PA is slightly more than that of CVNN.

G. Ablation Analysis of Proposed MAT

We include an extensive ablation study to tease apart the importance of the different components of MAT. In addition, the ablation analysis which is often ignored in the SS-SEI methods, MAT with fully supervised algorithm that only uses labeled training samples during training process, is considered in this article. Ablation details are shown in Table V. The identification performance with different ablation analysis are shown as Table VI. It can be observed that all of factors are crucial to the MAT-based SEI method’s success, and the identification performance of MAT will decrease if any factor is ablated. It also can be observed that the identification performance of MAT-* w/o UTS is better than that of MAT-* under some scenarios because the MAT is sensitive to the amount of labeled and unlabeled samples which is the shortcoming of most SSL methods [60].

H. Alternating Optimization Versus Simultaneous Optimization

In this article, the objective function is alternatively regularized by SSM and VAT that is extremely different from standard optimization termed simultaneous optimization, i.e.,

$$L = \omega_1 L_{CE} + \omega_2 L_{VAT} + \omega_3 L_{SSML}. \quad (20)$$

The identification performance of two optimization approaches on the ADS-B data set is shown in Fig. 3. It can be observed the clear superiority of identification performance of alternating optimization over simultaneous optimization, and the gaps of identification performance between alternating optimization and simultaneous optimization are $[-0.30\%, 2.70\%]$ and $[-0.40\%, 4.90\%]$ in Fig. 3(a) and (b), respectively. The training loss of two optimization approaches under ADS-B data set and the number of labeled training samples to the number of all training samples ratio is 10% is shown in Fig. 4. It can be observed that the convergence rate of alternating optimization is faster than that of simultaneous optimization. The advantages of alternating optimization are the higher identification performance and faster convergence rate.

---

**TABLE IV**

| Ratio  | CVNN  | DRCN  | SSRCNN | TripleGAN | SimMIM | MAT-CL (Proposed) | MAT-PA (Proposed) |
|--------|-------|-------|--------|-----------|--------|------------------|------------------|
| 5%     | 0.26s | 3.80s | 2.96s  | 3.14s     | 1.19s  | 1.93s            | 1.96s            |
| 10%    | 0.39s | 6.61s | 4.59s  | 5.74s     | 1.29s  | 3.35s            | 3.36s            |
| 20%    | 0.71s | 7.90s | 4.43s  | 10.66s    | 1.54s  | 4.30s            | 4.27s            |
| 50%    | 1.68s | 8.92s | 3.91s  | 16.23s    | 2.15s  | 5.58s            | 5.39s            |
| 100%   | 3.22s | 12.37s| 3.59s  | 32.43s    | 3.40s  | 6.88s            | 6.75s            |

---

**TABLE V**

| Components | SS-CE | SSML | VAT | UTS |
|------------|-------|------|-----|-----|
| MAT-* w/o SSML | ✓     | ✓    | ✓   | ✓   |
| MAT-* w/o VAT   | ✓     | ✓    | ✓   | ✓   |
| MAT-* w/o UTS   | ✓     | ✓    | ✓   | ✓   |
| MAT-* (Proposed)| ✓     | ✓    | ✓   | ✓   |

UTS denotes the unlabeled training samples and * means the SSML is SS-CL or SS-PA and w/o is an abbreviation for without.

---

**TABLE VI**

| Methods | ADS-B | Wi-Fi |
|---------|-------|-------|
| MAT-PA w/o SSML | 69.60% | 97.40% |
| MAT-PA w/o VAT    | 61.00% | 96.80% |
| MAT-PA w/o UTS    | 71.40% | 98.70% |
| MAT-PA (Proposed) | 74.00% | 97.30% |
| MAT-CL w/o SSML   | 69.60% | 97.40% |
| MAT-CL w/o VAT    | 57.60% | 97.30% |
| MAT-CL w/o UTS    | 65.80% | 97.90% |
| MAT-CL (Proposed) | 70.06% | 99.10% |
TABLE VII
IDENTIFICATION PERFORMANCE OF MAT-ML AND MAT-SSML

| Ratio | MAT-CL (ADS-B) | MAT-PA (ADS-B) | MAT-CL (Wi-Fi) | MAT-PA (Wi-Fi) |
|-------|----------------|----------------|----------------|----------------|
|       | ML  | SSML | ML  | SSML | ML  | SSML | ML  | SSML |
| 5%    | 70.50% | 70.50% (↑) | 72.00% | 74.00% (↑) | 29.33% | 27.20% (↑) | 28.82% | 28.82% (↑) |
| 10%   | 86.50% | 83.80% (↓) | 84.80% | 84.80% (↑) | 57.43% | 80.70% (↑) | 54.96% | 54.96% (↑) |
| 20%   | 95.00% | 95.00% (↑) | 93.60% | 93.90% (↑) | 99.76% | 99.76% (↑) | 99.70% | 98.18% (↑) |
| 50%   | 98.60% | 99.10% (↑) | 97.80% | 97.30% (↑) | 99.79% | 99.79% (↑) | 99.76% | 99.77% (↑) |
| 100%  | 99.40% | 99.40% (↑) | 99.30% | 99.30% (↑) | 99.79% | 99.79% (↑) | 99.77% | 99.77% (↑) |

Fig. 3. Identification performance of two optimization approaches on the ADS-B data set, where the ratio is the number of labeled training samples to the number of all training samples ratio. Alternating optimization and simultaneous optimization for (a) MAT-PA and (b) MAT-CL.

I. SSML Versus ML

The identification performance of ML and SSML is shown as Table VII, where (↑) means the identification performance of SSML is better than that of ML, and (↓) means the identification performance of SSML is worse than that of ML, and (−) means the identification performance of SSML is same as that of ML, and the ratio is the number of labeled training samples to the number of all training samples. The visualization of semantic features extracted by MAT with CL and MAT with SS-CL when ratio is 10% are shown in Fig. 5(a) and (b), respectively. It can be observed that the proposed trick can improve the identification performance and increase the intercategory dispersion and intracategory compactness of the extracted semantic features. In addition, the tolerable loss (i.e., 0.50%–2.70%) of identification performance is brought by SSML due to the sensitivity of SSL to the amount of labeled and unlabeled data.

Fig. 4. Training loss of two optimization approaches under the ADS-B data set and the number of labeled training samples to the number of all training samples is 10%. (a) Training loss of MAT-PA. (b) Training loss of MAT-CL.

VI. CONCLUSION

In this article, we proposed an SS-SEI method using MAT. Specifically, pseudo labels are innovatively introduced into ML and the SSML was proposed and used to extract the discriminative semantic features. VAT was used to extract the generalized semantic features. More specifically, an object function (i.e., the CE loss regularized by SSML or VAT) and an alternating optimization way were designed to achieve an SS-SEI method. The proposed MAT-based SS-SEI method was evaluated on an open source large-scale real-word ADS-B...
Future work.

Achieving open-set SEI identification based on MAT is our MAT-based SS-SEI method on multiple SEI data set and set. Evaluating the identification performance of the proposed under the ADS-B data set and 80%

10%, the identification accuracy of MAT-CL was 83%ing samples to the number of all training samples ratio was (a) and (b), respectively. The silhouette coefficient of MAT with CL and MAT with SS-CL when the data set is Wi-Fi and the number of labeled training samples to the number of all training samples ratio is 10% is (a) and (b), respectively.

Fig. 5. Visualization of semantic features of (a) MAT with CL and (b) MAT with SS-CL when the data set is Wi-Fi and the number of labeled training samples to the number of all training samples ratio is 10% is (a) and (b), respectively. The silhouette coefficient of MAT with CL and MAT with SS-CL is 0.13 and 0.54, respectively.

data set and Wi-Fi data set and was compared with four latest SS-SEI methods. The simulation results showed that the proposed MAT-based SS-SEI method achieved state-of-the-art identification performance. When the number of labeled training samples to the number of all training samples ratio was 10%, the identification accuracy of MAT-CL was 83.80% under the ADS-B data set and 80.70% under the Wi-Fi data set. Evaluating the identification performance of the proposed MAT-based SS-SEI method on multiple SEI data set and achieving open-set SEI identification based on MAT is our future work.

REFERENCES

[1] H. Yao, T. Mai, J. Wang, Z. Ji, C. Jiang, and Y. Qian, “Resource trading in blockchain-based industrial Internet of Things,” IEEE Trans. Ind. Informat., vol. 15, no. 6, pp. 3602–3609, Jun. 2019.

[2] X. Wang, L. T. Yang, Y. Wang, L. Ren, and M. J. Deen, “ADTT: A highly efficient distributed tensor-train decomposition method for IoIT big data,” IEEE Trans. Ind. Informat., vol. 17, no. 3, pp. 1573–1582, Mar. 2021.

[3] R. Zhao, Y. Wang, Z. Xue, T. Ohtsuki, B. Adesibi, and G. Gui, “Semi-supervised federated learning based intrusion detection method for Internet of Things,” IEEE Internet Things J., early access, May 18, 2022, doi: 10.1109/JIOT.2022.3175918.

[4] R. Xie et al., “A generalizable model-and-data driven approach for open-set RFF authentication,” IEEE Trans. Inf. Forensics Security, vol. 16, pp. 4435–4450, 2021.

[5] S. I. Popoola, B. Adesibi, M. Hammoudeh, G. Gui, and H. Gacanin, “Hybrid deep learning for botnet attack detection in the Internet-of-Things networks,” IEEE Internet Things J., vol. 8, no. 6, pp. 4944–4956, Mar. 2021.

[6] K. Zhang, X. Liang, R. Lu, and X. Shen, “Sybil attacks and their defenses in the Internet of Things,” IEEE Internet Things J., vol. 1, no. 5, pp. 372–383, Oct. 2014.

[7] X. Liang, R. Lu, X. Lin, and X. Shen, Security and Privacy in Mobile Social Networks (SpringerBriefs in Computer Science). New York, NY, USA: Springer, 2013.

[8] O. A. Dobre, “Signal identification for emerging intelligent radios: Classical problems and new challenges,” IEEE Instrum. Meas. Mag., vol. 18, no. 2, pp. 11–18, Apr. 2015.

[9] K. Merchant, S. Revay, G. Stantchev, and B. Nouisain, “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.

[10] A. C. Polak and D. L. Goeckel, “Identification of wireless devices of users who actively fake their RF fingerprints with artificial data distortion,” IEEE Trans. Wireless Commun., vol. 14, no. 11, pp. 5889–5899, Nov. 2015.

[11] J. Zhang, S. Rajendran, Z. Sun, R. Woods, and L. Hanzo, “Physical layer security for the Internet of things: Authentication and key generation,” IEEE Wireless Commun., vol. 26, no. 5, pp. 92–98, Oct. 2019.

[12] J. Toonstra and W. Kinsner, “Transient analysis and genetic algorithms for classification,” in Proc. IEEE Conf. Comput. Power Comput., May 1995, pp. 432–437.

[13] J. Toonstra and W. Kinsner, “A radio transmitter fingerprinting system ODO-1,” in Proc. Can. Conf. Electr. Comput. Eng., May 1996, pp. 66–63.

[14] I. O. Kennedy, P. Scanlon, F. J. Mullany, M. M. Buddhikot, K. E. Nolan, and T. W. Rondeau, “Radio transmitter fingerprinting: A steady state frequency domain approach,” in Proc. IEEE 68th Veh. Technol. Conf. (VTC-Fall), Calgary, AB, Canada, Sep. 2008, pp. 1362–1366.

[15] O. O. Medaiyese, M. Ezyma, A. F. Lauf, and A. A. Adeniran, “Hierarchical learning framework for UAV detection and identification,” IEEE J. Radio Freq. Identifi., vol. 6, pp. 176–188, 2022.

[16] N. Soltanieh, Y. Norouzi, Y. Yang, and N. C. Karmakar, “A review of radio frequency fingerprinting techniques,” IEEE J. Radio Freq. Identifi., vol. 4, pp. 222–233, 2020.

[17] J. Zhou et al., “Multiscale network traffic prediction method based on deep echo-state network for Internet of Things,” IEEE Internet Things J., vol. 9, no. 21, pp. 21862–21874, Nov. 2022.

[18] Y. Lin, Y. Tu, Z. Dou, L. Chen, and S. Mao, “Contour Stellia 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.

[19] F. Tang, Y. Kawamoto, N. Kato, and J. Liu, “Future intelligent and secure vehicular networks toward 6G: Machine-learning approaches,” Proc. IEEE, vol. 108, no. 2, pp. 292–307, Feb. 2020.

[20] N. Kato, B. Mao, F. Tang, Y. Kawamoto, and J. Liu, “Ten challenges in advancing machine learning technologies toward 6G,” IEEE Wireless Commun., vol. 27, no. 3, pp. 96–103, Jun. 2020.

[21] S. Chang, S. Huang, R. Zhang, Z. Feng, and L. Liu, “Multitask-learning-based deep neural network for automatic modulation classification,” IEEE Internet Things J., vol. 9, no. 3, pp. 2192–2206, Feb. 2022.

[22] 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.

[23] Z. Wang, W. Zhou, L. Chen, F. Zhou, F. Zhu, and L. Fan, “An adaptive deep learning-based UAV receiver design for coded MIMO with correlated noise,” Phys. Commun., vol. 47, pp. 10–18, Aug. 2021.

[24] 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. 5, pp. 6786–6799, Aug. 2019.

[25] L. Chen, S. Zheng, L. Yang, and X. Yang, “Deep learning for large-scale real-world ACARS and ADS-B radio signal classification,” IEEE Access, vol. 7, pp. 89256–89264, 2019.

[26] Y. Wang, G. Gui, H. Gacanin, T. Ohtsuki, O. A. Dobre, and H. V. Poor, “An efficient specific emitter identification method based on complex-valued neural networks and network compression,” IEEE J. Sel. Areas Commun., vol. 39, no. 8, pp. 2305–2317, Aug. 2021.

[27] 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.
Y. Pan, S. Yang, H. Li, and W. Wang, “Specific emitter identification based on deep residual networks,” *IEEE Access*, vol. 7, pp. 54425–54434, 2019.

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.

G.-J. Qi and J. Luo, “Small data challenges in big data era: A survey of recent progress on unsupervised and semi-supervised methods,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 4, pp. 2168–2187, Apr. 2022.

K. Sohn et al., “FixMatch: Simplifying semi-supervised learning with consistency and confidence,” in *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2020, pp. 596–608.

C. Xie, L. Zhang, and Z. Zhong, “Virtual adversarial training-based semisupervised specific emitter identification,” *Wireless Commun. Mobile Comput.*, vol. 2022, no. 1, pp. 1–14, Jan. 2022.

J. Gong, X. Xu, Y. Qin, and W. Dong, “A generative adversarial network based framework for specific emitter characterization and identification,” in *Proc. 11th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Xi’an, China, Oct. 2019, pp. 1–6.

C. Li, K. Xu, J. Zhu, J. Liu, and B. Zhang, “Triple generative adversarial networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 9629–9640, Dec. 2022.

K. Huang, J. Yang, H. Liu, and P. Hu, “Deep learning of radio frequency fingerprints from limited samples by masked autoencoding,” *IEEE Wireless Commun. Lett.*, early access, Jun. 21, 2022, doi: 10.1109/LWC.2022.3184674.

Y. Wang, G. Gui, H. Gacanin, T. Ohtsuki, H. Sari, and F. Adachi, “Few-shot specific emitter identification via deep metric ensemble learning,” *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2369–2374, Apr. 2022.

Y. Sun et al., “Circle loss: A unified perspective of pair similarity optimization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 6398–6407.

K. Tan, W. Yan, L. Zhang, Q. Ling, and C. Xu, “Semi-supervised learning with meta pseudo time-frequency labels,” in *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2020, pp. 596–608.

S. Kim, D. Kim, M. Cho, and S. Kwak, “Proxy anchor loss for deep metric learning,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3235–3244.

J. Tu et al., “Large-scale real-world radio signal recognition with deep metric learning,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–11.

M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, “LOF: Identifying density-based local outliers,” in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, 2000, pp. 93–104.

Y. Dong, X. Jiang, L. Cheng, and Q. Shi, “SSRCNN: A semi-supervised learning framework for signal recognition,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 3, pp. 780–789, Sep. 2021.

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.

F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.

Y. Peng, P. Liu, Y. Wang, G. Gui, B. Adebisi, and H. Gacanin, “FaceNet: A unified embedding for face recognition and clustering,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.

J. Gong, X. Qin, and X. Xu, “Multi-task based deep learning approach for open-set wireless signal identification in ISM band,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 1, pp. 121–135, Mar. 2022.

Y. Sun et al., “Circle loss: A unified perspective of pair similarity optimization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 6398–6407.

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. 4, pp. 2369–2374, Apr. 2022.

Y. Ren, P. Ren, and T. Zhang, “Deep RF device fingerprinting by semi-supervised learning with meta pseudo time-frequency labels,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11557–11568.

Y. Tu et al., “Large-scale real-world radio signal recognition with deep metric learning,” *Chin. J. Aeronaut.*, vol. 35, no. 9, pp. 35–48, Sep. 2022.

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.

A. Paszke et al., “Automatic differentiation in PyTorch,” in *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, USA, Dec. 2017, pp. 1–4.

D. Kinga and J. B. Adam, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2014, pp. 1–15.

L. van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

A. Oliver, A. Odena, C. Raffel, E. D. Cubuk, and I. J. Goodfellow, “Realistic evaluation of semi-supervised learning algorithms,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, Apr./May 2018, pp. 1–15.