

# Distributed Few-Shot Learning for Intelligent Recognition of Communication Jamming

Mingqian Liu, *Member, IEEE*, Zilong Liu, *Senior Member, IEEE*, Weidang Lu, *Senior Member, IEEE*, Yunfei Chen, *Senior Member, IEEE*, Xiaoteng Gao and Nan Zhao, *Senior Member, IEEE*

**Abstract**—Effective recognition of communication jamming is of vital importance in improving wireless communication system's anti-jamming capability. Motivated by the major challenges that the jamming data sets in wireless communication system are often small and the recognition performance may be poor, we introduce a novel jamming recognition method based on distributed few-shot learning in this paper. Our proposed method employs a distributed recognition architecture to achieve the global optimization of multiple sub-networks by federated learning. It also introduces a dense block structure in the sub-network structure to improve network information flow by the feature multiplexing and configuration bypass to improve resistance to over-fitting. Our key idea is to first obtain the time-frequency diagram, fractional Fourier transform and constellation diagram of the communication jamming signal as the model-agnostic meta-learning network input, and then train the distributed network through federated learning for jamming recognition. Simulation results show that our proposed method leads to excellent recognition performance with a small data set.

**Index Terms**—Jamming recognition, federated learning, few-shot learning, model-agnostic meta-learning.

## I. INTRODUCTION

WIRELESS communication systems need to deal with increasingly complex jamming environments [1]-[4]. Reliable transmission may be difficult to attain when a variety of jamming signals are present. Anti-jamming is possible by first carrying out jamming recognition which aims to understand and distinguish different types of communication jamming. The question at the centre is how to identify the jamming type accurately and rapidly [5]. Towards this end, machine learning has been widely adopted in recent years for efficient jamming recognition [6]. In particular, distributed machine learning based jamming recognition has attracted

growing research interest owing to its salient advantages, such as accommodation of massive nodes, wide-range distribution jamming, reliability and robustness of recognition performance [7]-[9].

Jamming recognition in wireless communication mainly focuses on the selection of feature parameters and the design of classifiers. From the perspective of efficient machine learning, it is essential to reduce the computational complexity in feature calculation. Various characteristics of communication jamming in both time- and frequency- domains were analyzed in [10], upon which two classifiers of neural network and decision tree were designed to realize jamming recognition. In [11], a broadband communication jamming recognition method based on graphs and neural networks was proposed. Short-time Fourier transform modulus were used as the network input, whereas a double hidden layer network was used to achieve jamming recognition. Convolutional neural networks (CNN) for the classification of wireless modulation signals was proposed in [12] to automatically extract signal features, leading to improved recognition performance than traditional feature extraction methods. Afterwards, a variety of deep learning networks for modulation recognition have been proposed in [13]. With the aid of real CNN and complex CNN, real residual network and complex residual network, four novel jamming recognition methods based on deep learning network have been developed in [14].

Despite a rich body of literature in machine learning based jamming recognition, it is often difficult to obtain a large amount of data for training in industrial signal processing. A promising direction in recent years is to use few-shot learning (FSL), which contains only a limited number of samples with supervised information. To list a few, a single-sample learning method based on the generative adversarial net (GAN) has been developed in [15] to achieve efficient distribution of the sampled data. For regular expression, the residual paired network was used to measure the similarity between sample pairs. Furthermore, a single-sample learning method was proposed in [16] by using siamese network [17]- [18]. Two CNNs were used to extract the input image features, map them to a one-dimensional vector, judge the distance between the sample pairs through the distance function, and then select the sample label with the smallest distance as the classification results. In [19], a matching network has been adopted in the long and short-term memory (LSTM) network to map samples to a low-dimensional feature space, after which the similarity of the labeled samples is calculated.

That said, to the best of our knowledge, very few works are

This work was supported by the National Natural Science Foundation of China under Grant 62071364, in part by the Aeronautical Science Foundation of China under Grant 2020Z073081001, in part by the Fundamental Research Funds for the Central Universities under Grant JB210104, and in part by the 111 Project under Grant B08038. (Corresponding author: Weidang Lu.)

M. Liu and X. Gao are with the State Key Laboratory of Integrated Service Networks, Xidian University, Shaanxi, Xi'an 710071, China (e-mail: mqliu@mail.xidian.edu.cn, xdugxt@163.com).

Z. Liu is with the School of Computer Science and Electronic Engineering (CSEE), University of Essex, Colchester CO4 3SQ, U.K. (e-mail: zilong.liu@essex.ac.uk).

W. Lu is with the College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: luweid@zjut.edu.cn).

Y. Chen is with the School of Engineering, University of Warwick, Coventry, West Midlands United Kingdom of Great Britain and Northern Ireland CV4 7AL (e-mail: yunfei.chen@warwick.ac.uk).

N. Zhao is with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, P. R. China. (email: zhaonan@dlut.edu.cn).

known on the applications of FSL in jamming recognition. It is noted that the goal of FSL is to learn a model that can quickly adapt to new tasks with only a small amount of data and the number of training iterations. To this end, model-agnostic meta-learning (MAML) has attracted significant research attention owing to its strength on small-sample learning problems, in which the training and testing of meta-learning use small-sample tasks as the basic unit, and only small-sample data is used in the training and testing phases. By exploiting small samples of unlabeled data and high-dimensional characteristics of jamming signals in wireless communication, we propose a novel jamming recognition framework based on distributed FSL networks. The main contributions of this work are summarized as follows:

- To eliminate the redundancy between related information in unsupervised methods, we introduce an intelligent representation of the smooth pseudo Wegener-willie distribution (SPWVD), fractional Fourier transform (FRFT) and constellation diagram for efficient processing of communication jamming signals. Such a representation not only has the least cross-term interference, high resolution and high classification accuracy for signals at different times or frequencies, but can also substantially suppress the effect of signal noise.
- Through our proposed sub-network model based on MAML, we show that the features of jamming signals can be extracted automatically to overcome the disadvantages of traditional manual feature extraction. Our proposed multiple agents combine a small amount of new information with their prior knowledge to attain enhanced generalization performance and improved recognition accuracy, and at the same time, avoid overfitting to new data.
- We employ federated learning for learning and recognition in distributed networks, aiming to improve the reliability of communication interference identification. On the premise of meeting data privacy and security requirements, we show that sub-networks can use their own data more efficiently and accurately.
- Compared with existing methods, our numerical experiments indicate that the proposed structure can automatically achieve FSL. Moreover, the communication jamming signals recognition performance can be improved without affecting the wireless transmission performance.

The remainder of this paper is organized as follows. First, we introduce the system model in Section II. In Section III, we present an intelligent representation of communication jamming signals. We propose intelligent recognition of communication jamming based on sub-network learning and federated learning in Section IV and Section V, respectively. Simulation studies are given in Section VI. Finally, Section VII concludes the whole paper.

## II. SYSTEM MODEL

In this paper, we consider a system model containing a transmission network with multiple sub-nodes. Each node has an independent sub-network model and jamming sample

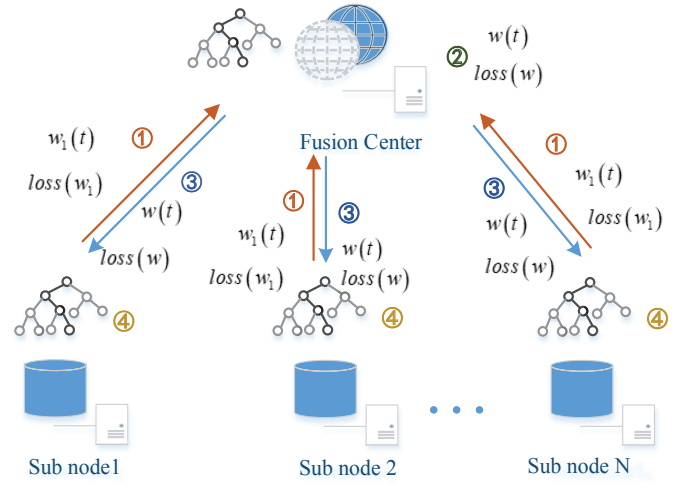


Fig. 1. System model of communication jamming recognition in distributed network.

database, as shown in Fig. 1. In the training process, a central node is selected from all edge nodes as the fusion center, who is responsible for parameter fusion and output coordination to complete federated learning. Each sub-node is equipped with the same sub-network model based on MAML. The parameters of all sub-networks are  $\{w_1, w_2, \dots, w_N\}$ , the loss of sub-networks is  $\{loss(w_1), loss(w_2), \dots, loss(w_N)\}$ , the global parameter at the center node is  $w$ , and the global loss function is  $loss(w)$ . In this system model, a distributed network based on federated learning is used to achieve global optimization and obtain the output model. In the each sub-node, the received signal  $y(t)$  can be expressed as

$$y(t) = x(t) + J(t) \quad (1)$$

where  $x(t)$  denotes the useful signal and  $J(t)$  represents communication jamming. Communication jamming is a class of electronic jamming that hinders the information transmission in wireless communication [20]. Based on the jamming effect, communication jamming may be categorized as suppressive jamming and deceptive jamming. Suppressive jamming is to send a noise-like energy signal to occupy the entire frequency spectrum of the target signal at all times. Deceptive jamming means that one party successfully pretends to be the other party and obtains illegal benefits by forging data or signals.

Typical suppressive jamming includes single-tone jamming, multi-tone jamming, noise frequency modulation (FM) jamming, noise band jamming, linear FM (LFM) jamming, etc. Single-tone jamming is the simplest form of jamming, which can be expressed as

$$J(t) = A \exp(j(2\pi f_c t + \varphi)), \quad (2)$$

where  $A$  represents the jamming signal amplitude,  $f_c$  stands for the carrier frequency, and  $\varphi$  denotes the initial phase. Multi-tone jamming is composed of multiple single-tone jamming, i.e.,

$$J(t) = \sum_{m=1}^M A_m \exp(j(2\pi f_m t + \varphi_m)), \quad (3)$$

where  $A_m, f_m$  and  $\varphi_m$  denote the amplitude, carrier frequency and initial phase of the first single-tone jamming, respectively. Noise band jamming causes the noise energy to be distributed within a certain bandwidth, which is expressed as

$$J(t) = U_n(t) \exp(j(2\pi f_c t + \varphi)), \quad (4)$$

where  $U_n(t)$  is the white noise with zero mean and variance  $\sigma_n^2$ . The amplitude of noise FM jamming remains unchanged, and its frequency changes due to modulation noise. Therefore, FM jamming can be expressed as

$$J(t) = A \exp\left(j\left(2\pi f_c t + k_{fm} \int_0^t \xi(t') dt'\right)\right), \quad (5)$$

where  $k_{fm}$  is the frequency modulation coefficient,  $\xi(t)$  denotes the modulated Gaussian noise with zero mean and the variance  $\sigma_n^2$  of 1. The linear frequency modulation (LFM) jamming refers to jamming with continuous linear changes in frequency over time, which can be expressed as

$$J(t) = A \exp[j(2\pi f_c t + \pi k t^2 + \varphi)], \quad (6)$$

where  $k$  the modulation slope.

For a direct-spread communication system (DS-SS), when the jamming signal consists of the useful signal's spreading code and carrier frequency, the jamming signal may seriously affect the operation of the communication system. However, it is very difficult to obtain the spreading code to accurately synchronize the signal. DS-SS signal carrier modulation usually adopts phase shift keying modulation. Because this modulation can suppress the carrier of the transmitted signal, it is difficult for the jamming to achieve carrier frequency targeted jamming, while the sender can use more power to transmit information, which can achieve the highest transmission efficiency within a certain bandwidth. Random binary code modulation jamming uses a pseudo-random code that has certain correlation with the adopted spreading code, which can also achieve good jamming effects when the synchronization is good. In this work, we are interested in the tackling of BPSK jamming which can be expressed as

$$J(t) = s(t) \cos(2\pi f_c t + \varphi), \quad (7)$$

where  $s(t) = A \sum_n a_n g(t - nT_s)$ .  $A$  stands for the jamming amplitude,  $a_n \in \{-1, 1\}$ ,  $g(t)$  the raised cosine roll-off pulse whose pulse width is  $T_s$ . When the symbol rate is less than the true symbol rate, BPSK jamming reduces to BPSK narrowband jamming (BPSK-NBJ) or BPSK broadband jamming (BPSK-WBJ).

### III. INTELLIGENT REPRESENTATION OF COMMUNICATION JAMMING SIGNALS

When one uses the jamming signal sequence directly as inputs, a large number of samples may be required for training. To avoid overfitting, feature extraction of the jamming signal is often carried out in order to highlight the difference between jamming signals, accelerate network convergence and improve recognition performance. For intelligent representation, the time-frequency diagram [21], FRFT [22] and constellation

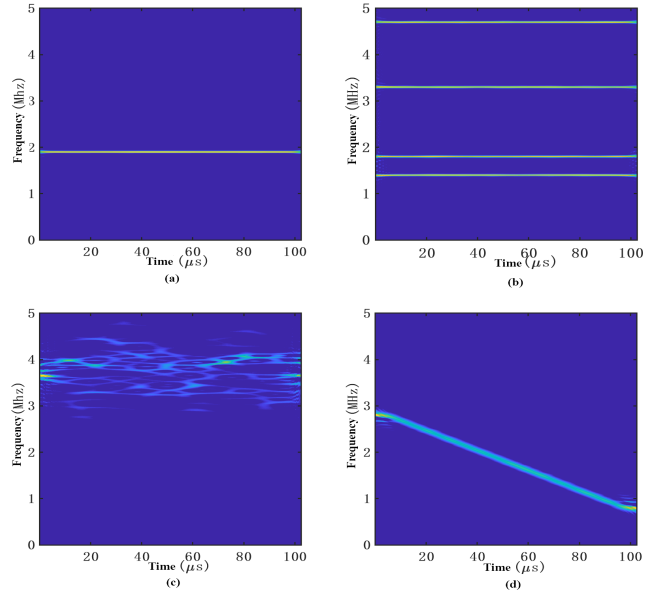


Fig. 2. Time-frequency diagram of communication jamming signals.

diagram of the jamming signal are extracted and then superimposed into a three-channel data set as network input. The constellation diagram is mainly used to identify deceptive jamming and suppressive jamming, the fractional Fourier transform is mainly used to identify LFM jamming, whilst the time-frequency diagram is mainly used to identify suppressive jamming.

Wegener-willie distribution (WVD) is a Cohen-like bilinear transformation that can describe the energy distribution of a signal on the time-frequency plane. Since WVD is a quadratic nonlinear transformation, serious cross-term jamming may occur. The cross-term jamming can be suppressed by the windowing method to smooth the frequency domain, so that the smooth pseudo WVD (SPWVD) can be obtained. Mathematically, this can be expressed as

$$S_{PW_J}(t, f) = \int_{-\infty}^{+\infty} h(\tau) * \left[ \int_{-\infty}^{+\infty} g(u - \tau) J\left(t + \frac{\tau}{2}\right) J^*\left(t - \frac{\tau}{2}\right) e^{-j2\pi f \tau} du \right] d\tau, \quad (8)$$

where  $S_{PW_J}(t, f)$  denotes the SPWVD of the jamming signal,  $h(\tau)$  the time window function,  $g(u - \tau)$  stands for the frequency window function, and  $J(t)$  the jamming signal. The SPWVD of single-tone jamming is shown in Fig. 2 (a), which consists of a horizontal line. Multi-tone jamming is composed of multiple horizontal lines as shown in Fig. 2 (b). Noise FM jamming is shown in Fig. 2 (c), which was a 1 MHz bandwidth, and the energy distribution in the bandwidth is rather messy. The LFM jamming is shown in Fig. 2 (d), where the frequency changes linearly with time, as a diagonal line in the time-frequency diagram.

FRFT may be regarded as a representation method on the fractional Fourier domain formed by the signal in the time-frequency plane and the coordinate axis is rotated counter-clockwise at any angle around the origin. The representation of the signal in the fractional Fourier domain combines the

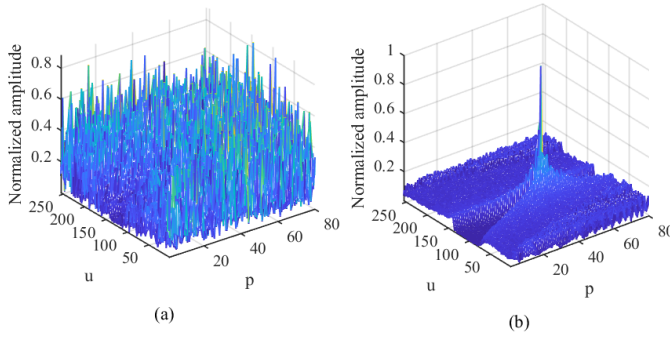


Fig. 3. FRFT diagram of communication jamming signals.

information of the signal in the time domain and the frequency domain at the same time, so it is considered to be a time-frequency analysis method and has a good focus on linear frequency modulation signal. FRFT maps the energy distribution of a signal on the ordinary time-frequency plane to the distribution on the FRFT plane. The FRFT plane coordinate axis is obtained by rotating the time-frequency plane counterclockwise by a certain angle. The FRFT calculation method of the jamming signal  $J(t)$  is

$$X_p(u) = F^p[J](u) = \int_{-\infty}^{+\infty} J(t) K_p(t, u) dt, \quad (9)$$

where  $F^p$  is the FRFT operator and the kernel function  $K_p(t, u)$  is

$$K_p(t, u) = \{A_\alpha e^{j(t^2 \cot \alpha/2 - ut \csc \alpha + u^2 \cot \alpha/2)} \mid \alpha \neq n\pi/\delta(t-u), \alpha = 2n\pi/\delta(t+u), \alpha = (2n+1)\pi\} \quad (10)$$

Here,  $A_\alpha = \sqrt{(1-j \cot \alpha)/2\pi}$ ,  $\alpha = p\pi/2$  represents the rotation angle of the time-frequency plane. FRFT is the expansion of a signal on a set of orthogonal chirp bases. With the help of FRFT to focus on the linear frequency sweep signal, the linear frequency sweep jamming signal can be well identified from other jamming signals. When extracting this feature, we need to continuously adjust the value  $p$  to obtain the fractional transformation matrix of the jamming signal as

$$X_J = [X_{p_1}(u), \dots, X_{p_i}(u) \dots X_{p_N}(u)], \quad (11)$$

where  $X_{p_i}(u)$  represents the fractional Fourier transform of order  $p_i$ . The FRFT of LFM jamming is shown in Fig. 3. From Fig. 3 (b), it can be seen that the energy of the LFM jamming gathers into a very high peak at  $p=0.31$ , while other jamming signals FRFT is shown in Fig. 3 (a), the energy distribution is relatively stray. Therefore, FRFT can identify LFM jamming from other jamming signals.

The constellation diagram contains rich modulation information, which is mainly used to identify suppressed jamming signals from deceptive jamming signals. The random binary

code jamming constellation diagram is shown in Fig. 4 (a), whereas the constellation diagram of single-tone jamming is shown in Fig. 4 (b). The signal points are symmetrically scattered around the unit circle, and the constellation diagrams of multi-tone jamming and part of the frequency band noise jamming are more spurious, which are shown in Fig. 4 (c) and Fig. 4 (d). From Fig. 4, it can be seen that the signal points are scattered around the center of the circle. Therefore, deceptive jamming can be identified from suppressive jamming by using the constellation diagram.

According to the above intelligent representation, the jamming signal features can be expressed as

$$\gamma = [S_{PW_J}(t, f), X_J(u), S_J] \quad (12)$$

where  $S_{PW_J}(t, f)$  represents the SPWVD transform,  $X_J(u)$  stands for the FRFT, and  $S_J$  is the constellation diagram of the jamming signal.

#### IV. SUB-NETWORK BASED ON MODEL-AGNOSTIC META-LEARNING

##### A. DenseNet Network Structure

Densely Connected Convolutional Networks (DenseNet) has a major advantage of overcoming stereotypical thinking of widening and deepening the network. The use of feature multiplexing and configuration bypass greatly reduces the scale of network parameters, helping reduce the chance of gradient disappearance for faster network convergence, and has a good regularization effect and anti-overfitting ability.

DenseNet is mainly composed of a dense block and a transition layer which is shown in Fig. 5. Assume that  $x_i$  is the output of the  $i$ -th layer in the dense block, and  $H_i(\cdot)$  stands for the nonlinear transformation function of the  $i$ -th layer, which consists of batch normalization (BN), activation function ReLU, and convolutional layer Conv. The different network layers inside the dense block adopt the form of dense connection, i.e., the input of the  $i$ -th layer is a stack of the output of the  $i-1$ -th layer and the outputs of all layers in between. Then,  $x_i$  can be expressed as

$$x_i = H_i([x_0, x_1, \dots, x_{i-1}]), \quad (13)$$

where  $[\cdot]$  represents the splicing of feature maps. The number of channels output of  $H_i(\cdot)$  is a fixed value  $k$ , and then the  $i$ -th layer network will have  $k_0 + k \times (i-1)$  feature maps,  $k_0$  is the number of channels in the input layer, and  $k$  denotes growth rate, which is generally smaller.

The transition layer is used to connect two dense blocks, whose function is to adjust the size of the feature map. The transition layer is generally composed of a  $1 \times 1$  convolutional layer and a pooling layer with a step size of 2. The number of intelligent representations output by the transition layer is  $\theta m$ , where  $m$  is the number of intelligent representations output by the previous dense block of the transition layer, and  $0 < \theta \leq 1$  is the compression factor. When  $\theta$  is less than 1, the transition layer can reduce the combination of network parameters and features.

Drawing on the idea of dense blocks in DenseNet, this paper introduces a network model including multiple dense blocks

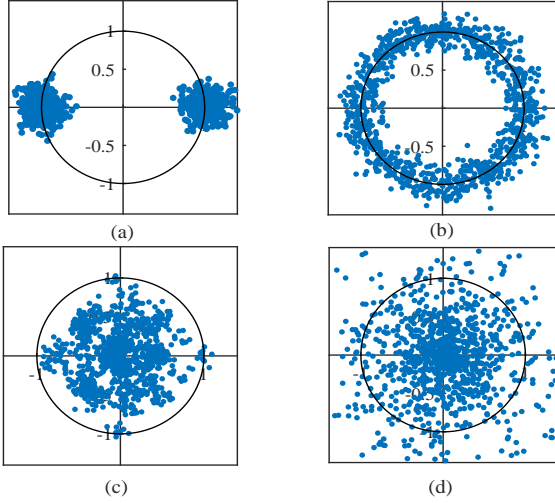


Fig. 4. Constellation diagram of communication jamming signals.

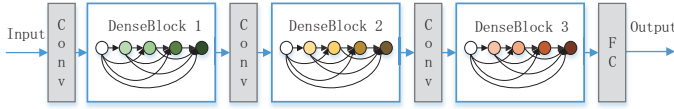


Fig. 5. Network structure based on DenseNet.

and transition layers for communication jamming recognition. The proposed structure is shown in Fig. 6, for a network of  $128 \times 128 \times 3$  jamming signal feature map. The feature map first passes through a  $7 \times 7$  convolutional layer with a step size of 2, and a  $4 \times 4$  maximum pooling layer with a step size of 2. Then the output enters the first dense block. The  $3 \times 3$  convolutional layers with a step size of 1 are used in the dense block to keep the feature map size unchanged, and the growth rate of the dense block is 8. The number of convolution kernels used by the convolution layer is 8. Then, it passes through three transition layers and two dense blocks to reach the fully connected layer, and finally uses the normalized exponential function softmax to obtain the recognition results.

### B. Model-agnostic Meta-learning

When the number of training samples is small, traditional network training methods are no longer adaptable, which may lead to serious overfitting. MAML is an excellent model-independent meta-learning method that can solve the problem of overfitting [23]–[25]. The purpose of using MAML is to get a better initialization parameter of the communication jamming recognition model, and to complete the training of the next task using this parameter. Meanwhile, the samples in each task are divided into support set and query set. The training process is shown in Fig. 7.

MAML firstly initializes the main network parameters  $\phi^0$ , then selects some samples from the collected communication jamming samples to form the training task  $m$ , and copies the main network parameters to get the unique network  $\hat{\theta}^m$  of the  $m$  task. MAML uses the support set of task  $m$  to optimize

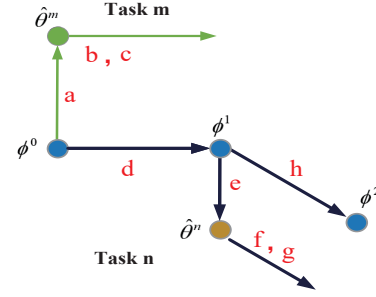


Fig. 6. MAML training flowchart.

the unique network of the task, get query set's loss  $l^m(\hat{\theta}^m)$  based on  $\hat{\theta}^m$ , and calculate the gradient of  $l^m(\hat{\theta}^m)$  to  $\hat{\theta}^m$ . Using the gradient and the learning rate of the main network  $\alpha_{meta}$ , it updates the main network parameters to obtain  $\phi^1$  as

$$\phi^1 = \phi^0 - \alpha_{meta} \nabla l^m(\hat{\theta}^m). \quad (14)$$

Finally, it selects the next training task to perform the same update operation on the main network.

From the above, the training steps of MAML network are summarized in **Algorithm 1**.

---

#### Algorithm 1 Sub-network training based on MAML

---

- 1: Select  $N$  training tasks and several test tasks from the communication jamming samples;
  - 2: Build the main network for communication jamming recognition, and initialize the parameter  $\phi^0$ ;
  - 3: Iterative training of communication jamming recognition network;
  - 4: Select training task  $m$ , copy the main network and its parameters  $\hat{\theta}^m = \phi^0$ ;
  - 5: Use the support set of task  $m$  to optimize and update  $\hat{\theta}^m$  once based on the learning rate  $\alpha_m$  of task  $m$ ;
  - 6: Use the query set of the  $m$  task to calculate the loss  $l_m(\hat{\theta}^m)$ , and calculate the gradient of  $l_m(\hat{\theta}^m)$  to  $\hat{\theta}^m$  for  $\hat{\theta}^m$  optimization;
  - 7: Use the gradient obtained in Step 4 to multiply the learning rate  $\alpha_{meta}$  of the main network to update  $\phi^0$  to obtain  $\phi^1$ ;
  - 8: Select a new training task and repeat Step 4 to Step 7;
  - 9: Use the support set of the test task to tune the recognition network, and use the query set of the test task to evaluate the performance of the recognition network.
- 

The network uses the cross entropy function to calculate the loss, which can be expressed as

$$loss_i(w) = \sum_{i=1}^N y_i \ln(\hat{y}_i), \quad (15)$$

where  $N$  represents the length of the network output vector,  $y_i$  stands for the actual value, and  $\hat{y}_i$  denotes the predicted value.

## V. DISTRIBUTED LEARNING BASED ON FEDERATED LEARNING

Due to the sensitivity of communication jamming data, jamming samples may be collected and stored in multiple locations. Jamming samples cannot be directly uploaded through the network for network training. Due to this, a distributed network is constructed to train local samples by federated learning to achieve global optimization without exchanging samples [26]- [28].

### A. Global Loss Function

Federated learning needs to train the local sub-network, and all sub-networks use the local data set to perform one or more rounds of updated network parameters  $w_i$ , which are sent to the central node through the communication network. The central node aggregates the received sub-network parameters through certain aggregation rule to obtain the global parameter  $w$ , then the global parameters are sent to each sub-network to update training. Federated learning can obtain the global loss function at the central node by

$$loss(w) = \frac{\sum_{i=1}^N loss_i(w) \times D_i}{\sum_{i=1}^N D_i}, \quad (16)$$

where  $loss_i(w)$  denotes the loss of the  $i$ -th sub-network on the global parameters using the local sample set,  $D_i$  the size of the local sample set, and  $N$  the total number of sub-networks. The global loss function may not be obtained directly at the central node, and the transmission network needs to be used to send the loss of each sub-network to the central node.

Different from conventional training, MAML uses the query set to update the main network parameters with the gradient of the current task sub-network, instead of directly using the gradient of the support set to update. This means that MAML pays more attention to recognizing the potential of the network, rather than its performance based on local samples. Task-driven training method of MAML gives the recognition network the ability to acquire prior knowledge, and focuses on the characteristics of the recognition network's potential for unknown jamming recognition, which maximizes the training benefits of each jamming signal sample, and is more in line with the difficulties faced by few-shot learning problem.

### B. Distributed Gradient Descent

The ultimate goal of federated learning is to find a global network parameter  $w^*$  that minimizes the global loss function  $loss(w)$  by

$$w^* = \arg \min loss(w). \quad (17)$$

We use the distributed gradient descent method [29]- [31] to minimize the global loss function, and set the local model parameter of each sub-node to  $w_i(t)$ , where  $t = 0, 1, 2, \dots, N$  represents the number of training iterations. When  $t = 0$ , all sub-network parameters are initialized to the same parameter  $w_i(0)$ . When  $t > 0$ ,  $w_i(t)$  is calculated based on the parameters of the previous iteration and the local loss function. In

## Algorithm 2 Distributed network training based on federated learning

- 1: Construct a distributed network for jamming recognition, establish multiple sub-networks  $Submodel_i$ , where  $i$  represents the  $i$ -th sub-network, and initialize  $w^f$ ,  $w_i(0)$  and  $\tilde{w}_i(0)$  as the same parameters;
- 2: For the  $i$ -th network, use (18) to perform a partial update to obtain  $w_i(t)$ , where  $t$  represents the number of partial updates;
- 3: Judge whether the current partial update times is an integer multiple of  $\tau$ . If yes, proceed to Step 4 and Step 5; otherwise proceed to Step 6;
- 4: Send all sub-networks to the central node, perform global aggregation and send global parameters to the sub-nodes  $\tilde{w}_i(t) = w(t)$ ;
- 5: Obtain all loss functions and use (19) to update the output network parameters;
- 6: Update all  $\tilde{w}_i(t)$  for all sub-nodes, and let  $\tilde{w}_i(t) = w_i(t)$ ;
- 7: Repeat from Step 2 and Step 6 until the training is over and get the final global network parameters  $w^f$ .

this way, the process of gradient descent on the local data set and parameters is locally updated. The central node performs global aggregation after several local updates as

$$w(t) = \frac{\sum_{i=1}^N D_i w_i(t)}{D}. \quad (18)$$

In the each training iteration, the sub-network will perform a local update, and then a global aggregation step may be performed.  $\tilde{w}_i(t)$  represents the parameters of the sub-network at node  $i$  after possible global aggregation. In the  $t$  iteration, if there is no global aggregation, then  $\tilde{w}_i(t) = w_i(t)$ ; if global aggregation is performed, then  $\tilde{w}_i(t) \neq w_i(t)$ . For the  $i$ -th sub-network, the local update rule is

$$w_i(t) = \tilde{w}_i(t-1) - \eta \nabla loss_i(\tilde{w}_i(t-1)). \quad (19)$$

After obtaining the global loss function, the central node uses the global loss function as a criterion to judge the current global parameters, and updates the output parameters as

$$w^f = \arg \min_{w \in \{w^f, w(t)\}} loss(w). \quad (20)$$

Federated learning repeats the above procedures continuously to protect the privacy of the sub-network samples [32]- [34]. Assuming that the sub-node network performs a global aggregation after  $\tau$ -step local update, the final output model parameter is  $w^f$ , the training steps of federated learning are summarized in **Algorithm 2**.

In conclusion, the main procedures of the proposed communication jamming intelligent recognition method based on few-shot learning is summarized in **Algorithm 3**.

## VI. NUMERICAL RESULTS AND DISCUSSION

In this section, we conduct simulation to validate the performance of the proposed intelligent recognition scheme. In our simulation, we employed Python for network construction and training. The training environment is Python 3.6,

---

**Algorithm 3** Communication jamming intelligent recognition based on few-shot learning

---

- 1: Use (11) to intelligently representation the received communication jamming signal, and extract its time-frequency distribution, FRFT and constellation diagram as network input;
  - 2: Build a distributed network and a sub-network model based on model-agnostic meta-learning;
  - 3: Train the distributed network by using federated learning, and use (19) to obtain the global optimal output model  $w^f$  to complete the jamming type recognition.
- 

the operating system is Linux, the computing framework is Pytorch, and GPU (GeForce GTX 3080TI) has been used for training acceleration. Seven types communication jamming signals, including single-tone jamming, multi-tone jamming, noise band jamming, noise FM jamming, LFM jamming, BPSK narrowband jamming, and BPSK wideband jamming were considered [35]- [36]. Under the assumption of perfect sampling synchronization, the sampling frequency is 10 MHz and the carrier frequency and initial phase of each type of jamming are randomly set. The number of tones of multi-tone jamming is 4, the bandwidth factor of Noise band jamming is randomly set from 0.1 to 0.7 (the bandwidth factor is the ratio of the jamming bandwidth to the bandwidth of the receiver), frequency modulation coefficient of FM jamming is randomly set from 0.125 Hz/s to 0.933 Hz/s, the modulation slope of LFM jamming is set randomly among 1.953 9.766 Hz/s and 0.35 roll-off coefficient is used for BPSK jamming. Each jamming signal was generated with 100 samples at each jamming-to-noise ratio (JNR) as training data, and 400 samples were generated as test data. In this paper, the JNR is defined as [37]

$$G_{JNR} = 10 \lg(P_J/P_N), \quad (21)$$

where  $P_J$  is the jamming signal power,  $P_N$  stands for the noise power.

#### A. Recognition Performance Based on Sub-network Learning

The dense block growth rate is set to 8. The parameters are initialized with "kaiming".<sup>1</sup> The MAML main network learning rate is 0.0002. The subtask learning rate is 0.04. The batch size is set to 105. The performance of various jamming recognition is shown in Fig. 8, which shows that the recognition rate of various types of jamming increases with the JNR. When the JNR is 2 dB, the recognition rate of various types of jamming reaches more than 90%. When the JNR is increased to 4 dB, the recognition rate of various types of jamming is close to 100%. Thus, the MAML-based sub-network model proposed in this paper can effectively identify various types of communication jamming under small sample conditions.

In order to evaluate the influence of the dense block growth rate on the recognition performance of the sub-network, the

growth rates are set to 2, 4, 8, 16, and 32 while other parameters remain unchanged. As shown in Fig. 9, the average recognition rate improves with the increase of the growth rate. The performances of the growth rates of 4, 8, and 16 are close to each other, which has a significant performance improvement over the growth rate of 2. The performance of the growth rate of 8 is the best. When the growth rate increases to 32, there is certain performance improvement at low JNR, but significant performance degradation occurs at higher JNR. Thus, increasing the growth rate of dense blocks can improve network performance to certain extent, but an excessively high growth rate will cause over-fitting problem.

To examine the influence of frequency offset changes on the recognition performance, the frequency offset is set to 10 kHz, 20 kHz, 40 kHz, respectively. In Fig. 10, it is easy to see that the jamming recognition accuracy curve under different frequency offsets increases with JNR. The accuracy rate rises to 90% when the JNR is 1 dB, and the accuracy rate is close to 100% when JNR is 3 dB. From the above, we can see that our proposed identification method is robust to frequency offset.

To study the influence of Rayleigh fading channels on the recognition performance, the jamming signal channel is set as Rayleigh fading channel [38], [39] and the Doppler frequencies are set to 10 Hz, 20 Hz, and 30 Hz, respectively. From Fig. 11, we can see that the jamming recognition accuracy curve over Rayleigh fading channels increases with the JNR increase. When JNR is 1dB, the accuracy rate rises to 80%. When the JNR is 4 dB, the accuracy rate is close to 90%, so the proposed method is effective over Rayleigh fading channels.

TABLE I  
PERFORMANCE COMPARISON OF DIFFERENT COMMUNICATION JAMMING RECOGNITION METHODS.

Methods	-4dB	0dB	4dB	8dB	12dB
Proposed	84.64%	97.21%	99.25%	99.29%	99.18%
[12]	83.54%	86.74%	95.46%	95.84%	96.30%
[14]	84.00%	97.10%	98.43%	98.14%	99.14%

In order to verify the influence of the number of training samples on the recognition performance, the JNR is 10dB and the number of training samples for each type of jamming signal is set to 25, 50, 100, 200, 400, respectively. A comparative experiment is performed on the MAML-based sub-network model, the recognition performance of the proposed method, [12] and [14] are shown in Fig. 12. One can observe that the performances of the three recognition methods all increase with the increase of the number of training samples. When the sample size is 25, [14] is best, the performance of the proposed method is slightly lower than that of [14], and the performance of [12] is the worst. When the number of training samples is 100 or 200, the performance of the proposed method is better than [12] and [14]. When the sample size is 400, the performance of the three methods is relatively close.

Next, we compare recognition performances of different recognition methods, the proposed method in this paper is compared with the methods in [12] and [14]. The communication jamming types are single-tone jamming, multi-tone jamming, LFM jamming, partial frequency band jamming,

<sup>1</sup>Pytorch framework uses the kaiming normal distribution to initialize the convolutional layer parameters by default.

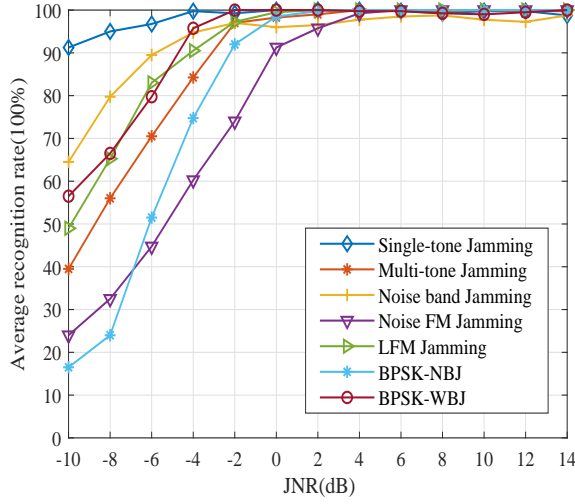


Fig. 7. Recognition performance of sub-network model based on MAML.

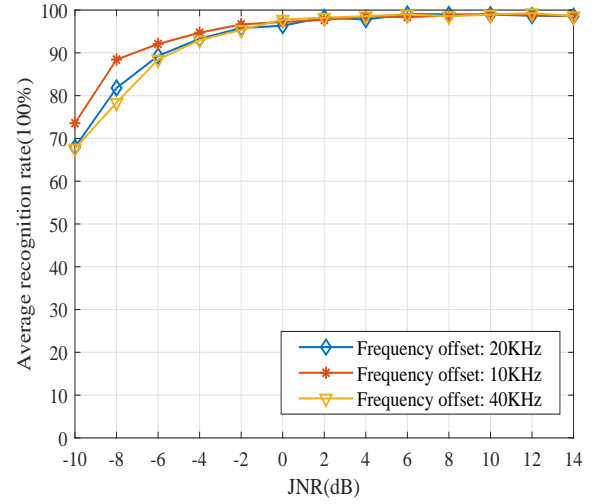


Fig. 9. Recognition performance of sub-network with different frequency offsets.

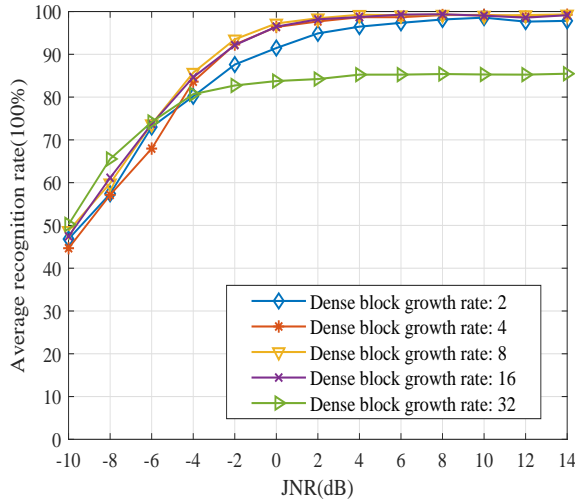


Fig. 8. Recognition performance of sub-network with different growth rates.

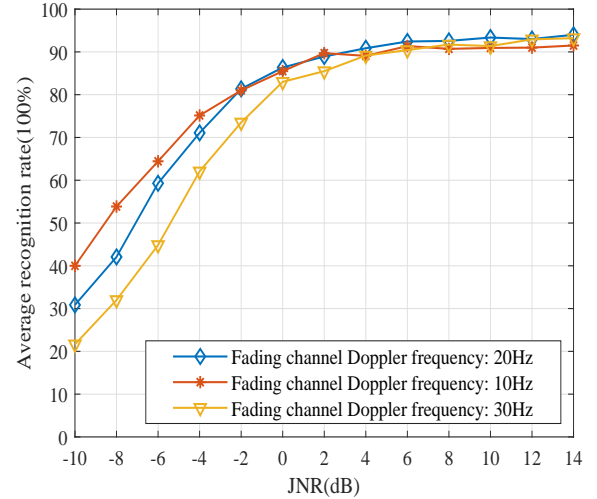


Fig. 10. Recognition performance of sub-network over Rayleigh fading channels.

and noise FM jamming. The number of training samples for each signal under each JNR is 400, and the comparison results are shown in Table I. As can be seen from the Tab. 1, The recognition performance of the proposed method is higher than that in [12] and [14] at low JNR. At high JNR, the recognition performance of the proposed method is higher than that of the other two methods. The proposed method has better performance under different JNRs. In the feature extraction section, the computational complexity of the proposed method and [14] are both  $O(N^2 \log_2 N)$ , and the computational complexity of [12] is  $O(N \log_2 N)$ . The computational complexity of the proposed method is higher than that of [12], but the recognition performance is better than that of [12]. Under the same GPU hardware acceleration conditions, the offline training time of the MAML-based sub-network model is 913.87s, the offline training time of [12] is 1009.51s, and the offline training time of [14] is 3656.42s. From the above, we can be seen that the training time of the

sub-network model based on MAML is slightly lower than that of [12], and the training time of [14] is much higher than that of other methods. Overall, the proposed method has better recognition performance than [12] and [14] under small samples condition.

### B. Recognition Performance Based on Distributed Learning

The number of sub-nodes is 3. The global aggregation interval is 4. The recognition performance of the distributed network is presented in Fig. 13. As shown in Fig. 13, we can observe that the recognition performance of various jamming signals improves with JNR. When JNR=0 dB, the recognition performance reaches more than 90%, and the recognition performance is close to 100% when JNR=4 dB. The recognition performance of BPSK narrowband jamming and BPSK wideband jamming is close, the performance of noise FM jamming is poor at low JNR, and the performance of single-

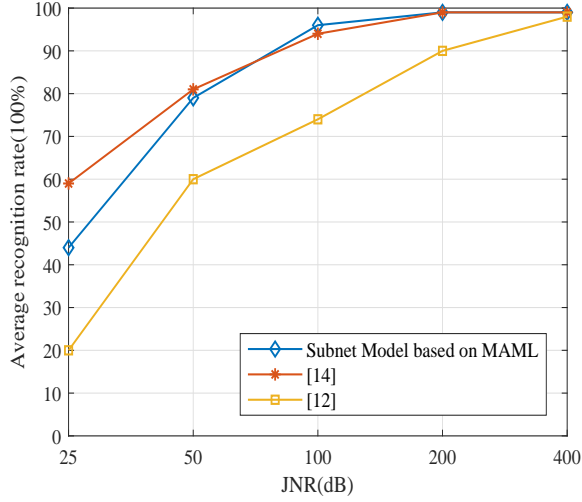


Fig. 11. Recognition performance comparison based on sub-network with different sample sizes.

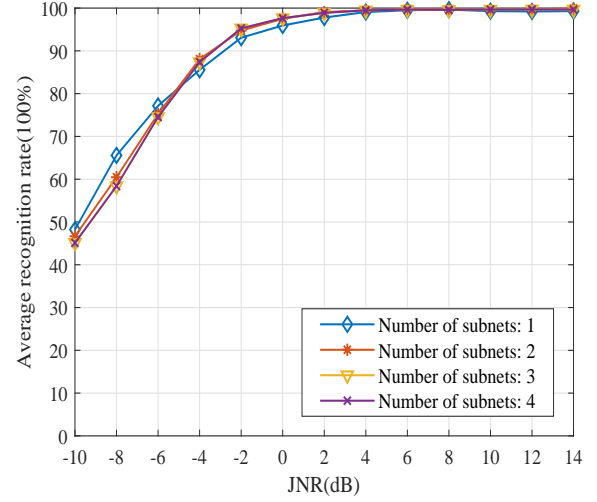


Fig. 13. Recognition performance of distributed network with the number of sub-networks.

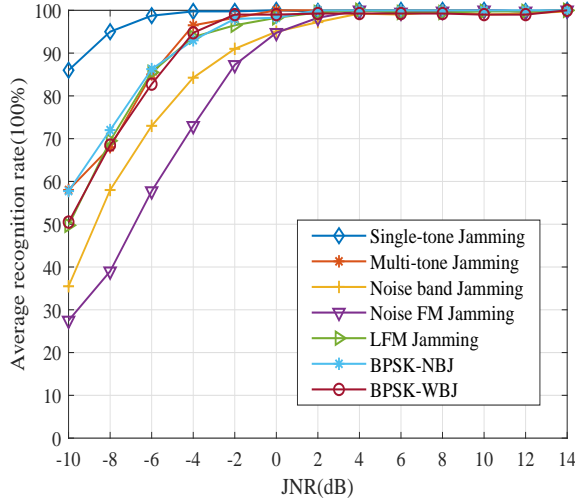


Fig. 12. Recognition performance based on distributed network.

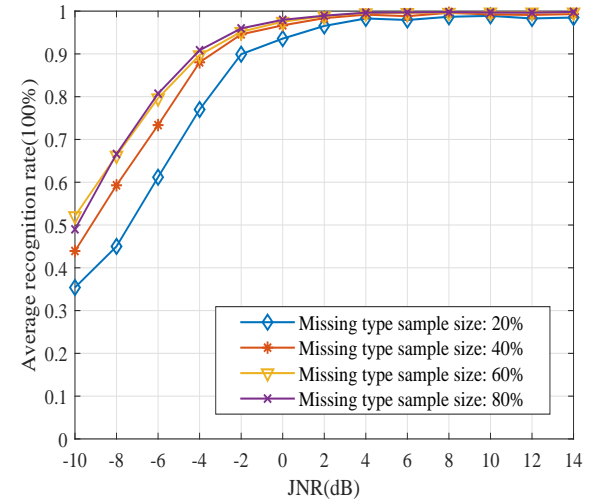


Fig. 14. Recognition performance of distributed network with different sample imbalance.

tone jamming recognition is the best. We can see that the distributed network can indirectly learn the samples of all sub-training sets, and could effectively recognize communication jamming signals. Compared with the recognition performance of sub-network in Fig. 8, the distributed network not only improves the recognition performance, but also enhances the stability of the network model.

In order to evaluate the recognition performance of the number of sub-networks based on distributed network, the number of sub-nodes is set to 1, 2, 3, and 4 for simulations, and the results are shown in Fig. 14. From Fig. 14, it can be found that when the number of sub-networks is 2, the performance is improved to compared with only one sub-network. When the number of sub-networks is 3 and 4, the performance is very close, and the performance is slightly better than the number is 2. From the above, the increase number of sub-networks will improve the recognition performance to a certain extent, but the performance improvement is limited when the number

of sub-networks increases to a certain number.

To study the impact of the imbalance in the number of jamming samples on the network performance, the number of child nodes is 3, and six of the seven jamming types are not replaced as missing types. Each node selects two missing types, and sets the number of missing type training samples to 20%, 40%, 60%, and 80% of the non-missing categories, respectively. The simulation results are shown in Fig. 15. In the case of unbalanced samples, the recognition performance increases with the JNR, and recognition rate is reaching 90% when JNR=2 dB. When the missing type is only 20% of the normal sample size, the recognition performance is significantly lower than others. When the sample size is 40%, the recognition performance is slightly lower than the performance when the sample size is 60% and 80%. From Fig. 15, we can see that the distributed network has the ability to adapt to the imbalance of the sample.

## VII. CONCLUSIONS

To carry out FSL of communication jamming signals in distributed networks, we have proposed an intelligent recognition framework in this paper. Through intelligent representation of communication jamming signals, we have obtained smooth pseudo Wegener-willie distribution, fractional Fourier transform and constellation diagram. We have introduced a sub-network model based on model-independent meta-learning and a distributed network training method based on federated learning. Finally, extensive simulation studies have been carried out to verify the effectiveness of the proposed framework. Simulation results have shown that our proposed framework exhibits an excellent recognition performance for FSL based distributed networks, whilst significantly outperforming the existing methods. In addition, our proposed method is robust and effective with the different frequency offsets and fading channels.

## REFERENCES

- [1] O. Shmuel, A. Cohen, O. Gurewitz, "Multi-antenna jamming in covert communication," *IEEE Transactions on Communications*, vol. 69, no. 7, pp. 4644-4658, July, 2021.
- [2] X. Chen, Z. Chang, N. Zhao, Y. Chen, F. Yu, A. Nallanathan. "Multi-antenna covert communication via full-duplex jamming against a warden with uncertain locations," *IEEE Transactions on Wireless Communications*, to appear, 2021.
- [3] P. Yang, Y. Xiao, M. Xiao, J. Zhu, S. Li, W. Xiang, "Enhanced receive spatial modulation based on power allocation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 6, pp. 1312-1325, Oct. 2019.
- [4] H. Yang, Z. Xiong, J. Zhao, D. Niyato, Q. Wu, H. Poor, M. Tornatore, "Intelligent reflecting surface assisted anti-jamming communications: a fast reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1963-1974, Mar. 2021.
- [5] Y. Shi, X. Lu, Y. Niu, Y. Li, "Efficient jamming identification in wireless communication: using small sample data driven naive bayes classifier," *IEEE Wireless Communications Letters*, vol. 10, no. 7, pp. 1375-1379, July, 2021.
- [6] Q. Qu, S. Wei, S. Liu, J. Liang, J. Shi, "JRNet: jamming recognition networks for radar compound suppression jamming signals," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 15035-15045, Dec. 2020.
- [7] M. Liu, K. Yang, N. Zhao, Y. Chen, H. Song, F. Gong, "Intelligent signal classification in industrial distributed wireless sensor networks-based industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 4946-4956, July, 2021.
- [8] M. Liu, G. Liao, N. Zhao, H. Song and F. Gong. "Data-driven deep learning for digital classification in industrial cognitive radio networks," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3412-3421, May. 2021.
- [9] Y. Liu, X. Yuan, Z. Xiong, J. Kang, X. Wang, D. Niyato. "Federated learning for 6G communications: challenges, methods, and future directions," *China Communications*, vol. 17, no. 9, pp. 105-118, Sept. 2021.
- [10] J. An, K. Yi, B. Tian, Q. Yu, "A modulation recognizer with ITD-based features," in *Proc. 2011 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, Oct. 2011, pp. 1-5.
- [11] X. Yang, T. Fu, Y. Wang, "Wireless communication jamming recognition based on lightweight residual network," in *Proc. 2020 IEEE 3rd International Conference on Electronics Technology (ICET)*, June, 2020, pp. 1-5.
- [12] T. O'Shea, J. Corgan, T. Clancy, "Convolutional radio modulation recognition networks," *Engineering Applications of Neural Networks*, vol. 629, pp. 213-226, Aug. 2016.
- [13] N. West, T. O'Shea, "Deep architectures for modulation recognition," in *Proc. 2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, May, 2017, pp. 1-6.
- [14] W. Li, J. Wang, L. Li, G. Zhang, Z. Dang, S. Li, "Intelligent anti-jamming communication with continuous action decision for ultra-dense network," in *Proc. 2019 IEEE International Conference on Communications (ICC)*, July, 2019, pp. 1-7.
- [15] S. Rahman, S. Khan, F. Porikli, "A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning," *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5652-5667, Nov. 2018.
- [16] G. Koch, R. Zemel, R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learning Workshop*, Dec. 2015, pp. 1-8.
- [17] S. Chopra, R. Hadsell, Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, Dec. 2005, pp. 539-546.
- [18] M. Norouzi, D. Fleet, R. Salakhutdinov, "Hamming distance metric learning," in *Proc. 25th International Conference on Neural Information Processing Systems*, vol. 1, Dec. 2012, pp. 1061-1069.
- [19] H. Pho, S. Lee, V. Tuyet-Doan, Y. Kim, "Radar-based face recognition: one-shot learning approach," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6335-6341, Mar. 2021.
- [20] H. Pho, S. Lee, V. Tuyet-Doan, Y. Kim, "The effect of narrow-band interference on wideband wireless communication systems," *IEEE Transactions on Communications*, vol. 53, no. 12, pp. 2139-2149, Dec. 2005.
- [21] S. Khare, V. Bajaj, U. Acharya, "SPWVD-CNN for automated detection of schizophrenia patients using EEG signals," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 2507409-2507409, Apr. 2021.
- [22] F. Liu, L. Wang, J. Xie, "Directional modulation via weighted fractional fourier transform at A way to enhance security," in *Proc. 12th European Conference on Antennas and Propagation (EuCAP 2018)*, Dec. 2018, pp. 1-5.
- [23] C. Finn, P. Abbeel, S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 1the 34th International Conference on Machine Learning*, vol. 70, Mar. 2017, pp. 1126-1135.
- [24] D. Shi, W. Gan, B. Lam, K. Ooi, "Fast adaptive active noise control based on modified model-agnostic meta-learning algorithm," *IEEE Signal Processing Letters*, vol. 28, pp. 593-597, Mar. 2022.
- [25] A. Madan, R. Prasad, "B-small: a bayesian neural network approach to sparse model-agnostic meta-learning," in *Proc. 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May, 2021, pp. 1-5.
- [26] S. Wang, T. Tuor, T. Salonidis, et al. "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205-1221, June, 2019.
- [27] Y. Gao, L. Liu, B. Hu, T. Lei, H. Ma, "Federated region-learning for environment sensing in edge computing system," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2192-2204, Oct. 2020.
- [28] J. Shi, H. Zhao, M. Wang, Q. Tian, "Signal recognition based on federated learning," in *2020 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Aug. 2020, pp. 1-46.
- [29] M. Zhang, X. Liu, J. Liu, "Convergence analysis of a continuous-time distributed gradient descent algorithm," *IEEE Control Systems Letters*, vol. 5, no. 4, pp. 1339-1344, Oct. 2021.
- [30] X. Cao, L. Lai, "Distributed gradient descent algorithm robust to an arbitrary number of byzantine attackers," *IEEE Transactions on Signal Processing*, vol. 67, no. 22, pp. 5850-5864, Nov. 2021.
- [31] E. Ozfatura, S. Ulukus, D. Gündüz, "Distributed gradient descent with coded partial gradient computations," in *Proc. 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2019, pp. 1-5.
- [32] W. Zhang, Q. Lu, Q. Yu, Z. Li, Y. Liu, S. Lo, S. Chen, X. Xu, L. Zhu, "Blockchain-based federated learning for device failure detection in industrial IoT," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5926-5937, Apr. 2021.
- [33] J. Tan, Y. Liang, N. Luong, D. Niyato, "Toward smart security enhancement of federated learning networks," *IEEE Network*, vol. 35, no. 1, pp. 340-347, Jan. 2021.
- [34] H. Zheng, H. Hu, Z. Han, "Preserving user privacy for machine learning: local differential privacy or federated machine learning?," *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 5-14, Aug. 2021.
- [35] J. Zhang, K. Chen, "Analysis of Some Problems of Projectile-carried Communication Jamming based on Partial-band Jamming," in *2020 International Symposium on Computer Engineering and Intelligent Communications (ISCEIC)*, Aug. 2020, pp. 1-4.
- [36] F. Yang, F. Sha, Q. Song, J. Wang, "A novel method to evaluate the effectiveness of communications jamming," in *2007 International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications*, Aug. 2007, pp. 1-4.

- [37] Y. Tang, Z. Zhao, X. Ye, S. Zheng, L. Wang, "Jamming recognition based on AC-VAEGAN," in *2020 15th IEEE International Conference on Signal Processing (ICSP)*, vol. 1, Dec. 2020, pp. 1-4.
- [38] A. Sergienko, P. Apalina, "Noncoherent reception of signals in fast fading Rayleigh channel using generalized likelihood ratio test," in *2021 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF)*, July, 2021, pp. 1-5.
- [39] B. Li, J. Hou, X. Li, Y. Nan, A. Nallanathan, C. Zhao, "Deep sensing for space-time doubly selective channels: when a primary user is mobile and the channel is flat rayleigh fading," *IEEE Transactions on Signal Processing*, vol. 64, no. 13, pp. 3362-3375, July 2016.