IEEE ComSoc IEEE Communications Society | IEEE COMPUTER SOCIETY | IEEE Signal Processing Society | VTS | IEEE Transactions on Machine Learning in Communications and Networking

Received 26 October 2023; revised 23 February 2024; accepted 5 April 2024.
Date of publication 11 April 2024; date of current version 24 April 2024.

The associate editor coordinating the review of this article and approving it for publication was J. Zhang.

Digital Object Identifier 10.1109/TMLCN.2024.3387430

STTMC: A Few-Shot Spatial Temporal Transductive Modulation Classifier

YUNHAO SHI, HUA XU, ZISEN QI, YUE ZHANG, DAN WANG, AND LEI JIANG

Information and Navigation College, Air Force Engineering University, Xi'an 710043, China
CORRESPONDING AUTHOR: Z. QI (qizisen@163.com)

ABSTRACT The advancement of deep learning (DL) techniques has led to significant progress in Automatic Modulation Classification (AMC). However, most existing DL-based AMC methods require massive training samples, which are difficult to obtain in non-cooperative scenarios. The identification of modulation types under small sample conditions has become an increasingly urgent problem. In this paper, we present a novel few-shot AMC model named the Spatial Temporal Transductive Modulation Classifier (STTMC), which comprises two modules: a feature extraction module and a graph network module. The former is responsible for extracting diverse features through a spatiotemporal parallel network, while the latter facilitates transductive decision-making through a graph network that uses a closed-form solution. Notably, STTMC classifies a group of test signals simultaneously to increase stability of few-shot model with an episode training strategy. Experimental results on the RadioML.2018.01A and RadioML.2016.10A datasets demonstrate that the proposed method perform well in 3way-Kshot, 5way-Kshot and 10way-Kshot configurations. In particular, STTMC outperforms other existing AMC methods by a large margin.

INDEX TERMS Automatic modulation classification, few-shot learning, graph network.

I. INTRODUCTION

AUTOMATIC modulation classification (AMC) refers to the process of distinguishing the modulation types of wireless communication signals. In cooperative communication scenarios, identifying modulation scheme is the primary condition for accurate information recovery. In non-cooperative communication scenarios, knowledge of the modulation types of intercepted signals is crucial for precise interference guidance, electromagnetic spectrum monitoring, and object detection.

Classical AMC methods can be divided into two categories: likelihood-based (LB) methods and feature-based (FB) methods. LB methods obtain the classification result by calculating the likelihood ratio function. However, acquiring prior knowledge of the received signal in non-cooperative communication is challenging. Additionally, maximizing the likelihood ratio function to estimate unknown parameters is computationally intensive. Therefore, FB methods are preferable for practical usage. FB methods address AMC by using handcrafted features designed based on domain knowledge. They extract features through preset calculation rules such as high-order cumulants, instantaneous amplitude, and power spectral density. These features are then fed into a classifier to determine the modulation type. However, with the rapid development of modern communication technology, the number of communication devices has risen sharply, and modulation types change rapidly. Facing these practical challenges, traditional FB methods expose serval shortcomings such as difficulty in designing hand-crafted features, low processing speed, and deterioration in recognition performance. In complex electromagnetic environments, FB methods are outperformed by other methods.

A. MOTIVATIONS

Recently, Deep Learning (DL) models have achieved remarkable success in various domains, including object detection [1], natural language processing [2], and classification tasks [3]. With the integration of powerful optimization tools, DL-based AMC models have emerged. These methods utilize neural networks to automatically extract features, thereby eliminating the laborious process of feature design. Driven by large-scale data, DL-based AMC models offer serval advantages, such as exceptional environmental generalization, the ability to learn and extract effective features, minimal

information loss during signal processing, and stable recognition performance.

Despite some breakthroughs in current research on DL-based AMC models, certain challenges have become increasingly apparent as their practical application continues to expand. Most existing DL-based AMC methods rely heavily on massive training samples. However, acquiring a sufficiently number of training samples for practical applications, especially in non-cooperative scenarios, is challenging. When training samples are scarce, the performance of DL-based models significantly deteriorates. Consequently, achieving effective AMC under small sample conditions remains a critical challenge in DL-based AMC methods. Some researches have attempted to introduce semi-supervised learning methods and transfer learning methods to reduce the reliance on labeled samples. Although these models decrease the dependency on labeled samples, they still require a substantial number of unlabeled samples, which does not fundamentally address the small sample problem.

Few-shot learning (FSL) is designed to address the problem of small sample classification using only a few samples. It involves adapting a model to recognize new classes that were not present during the training process, using only a limited number of samples from these new classes. Recently, serval attempts have been made to apply FSL techniques to AMC task, but there is still considerable room for improvement. The lack of support data and the imbalance in SNR conditions make it challenging to develop a stable classifier. Additionally, most existing few-shot models solely on a few labeled training samples for prediction, overlooking the potentially useful information contained in unlabeled test samples within a task.

Therefore, we propose a graph-based few-shot transductive prediction model. This model predicts the labels of the entire testing set directly, rather than predicting each signal independently. This approach not only enhances global accuracy and stability but also taps into the latent information contained in test signals. Besides, the proposed model achieves few-shot classification through an episode training strategy which is as same as meta learning.

B. RELATED DEEP LEARNING WORKS

The related deep learning works are categorized into four groups based on required training samples and network formats, so as to demonstrate the saturation of traditional DL-based AMC methods and the urgency of AMC under small samples. Broadly, these categories are supervised learning-based models, semi-supervised and unsupervised learning-based models, transfer learning-based methods and few-shot learning-based models.

1) SUPERVISED LEARNING-BASED MODELS

Supervised learning-based AMC models, which require extensive labeled samples, are currently the most widely studied in the field of AMC. O'Shea et al. [4] were almost the first researchers that introduced a Convolution Neural Network (CNN) into AMC, employing VGG-net as backbone for feature extraction. Additionally, they incorporated residual structure into the 2d-CNN to enhance classification performance [5]. The RadioML.2016.10A and RadioML.2018.01A datasets they built also have become the most commonly used datasets in AMC research. Dong et al. [6] proposed an efficient and lightweight AMC model that initially employs a phase estimator to counteract phase offset. Subsequently, the I-channel, Q-channel and IQ-channel are fed into three different convolution blocks in parallel for feature extraction. Furthermore, this model utilizes group convolution to reduce algorithm complexity. Ke and Vikalo [7] served Long Short-Term Memory (LSTM) as the backbone and designed an LSTM denoising autoencoder for AMC. They employed the last hidden state vector of LSTM as the extracted features, which is then connected to a fully connected (FC) network for modulation classification. Zhang et al. [8] designed an efficient Residual Shrinkage Convolutional Neural Network (RSCNN) for noise suppressing, comprising convolution layers, batch normalization layers, shrinkage function, and the ReLU activation function. The features extracted are concatenated and fed into a FC network for modulation classification. Other significant contributions [9], [10], [11] have also designed state-of-the-art AMC models based on CNN and Recurrent Neural Network (RNN) variants.

In various networks, CNN and RNN are the most basic network structures. This prominence stems from the CNN's proficiency in extracting spatial feature and the RNN's effectiveness in temporal modeling. Consequently, scholars have increasingly sought to complement the advantages of these two classical structures, which became the main consideration for following researches [12], [13], [14], [15], [16], [17] Huang et al. [12] proposed a hybrid Gated recurrent residual Network (GrrNet) for AMC, comprising a Resnet-based feature extractor, fusion module, and GRU-based classification module. Chang et al. [13] developed a hierarchical classification head capable of integrating outputs from each layer for the final prediction. Their model sequentially concatenates convolution blocks, Bidirectional Gated Recurrent Units (BiGRUs) and FC layers in a series way for feature extraction. Furthermore, Chang et al. [14] recently designed a new model named MLDNN. This model utilizes a CNN-block, BiGRU-block and SAFN-block to simultaneously extract features from I/Q and A/P sequences. It then incorporates a multi-task learning head method for final classification.

In addition to the aforementioned representative works, other researchers have attempted to introduce some novel theories into AMC domain, such as Transformer [18], [19], Neural Architecture Search (NAS) [20], Gradient-weighted Class Activation Mapping (Grad-CAM) [21], and Attention Mechanism (AM) [22], [23].

Overall, the supervised learning-based AMC models have been extensively and thoroughly studied. Consequently, focusing on marginal performance enhancements may not justify the associated costs and efforts.

2) SEMI-SUPERVISED AND UNSUPERVISED LEARNING-BASED MODELS

Compared with supervised learning-based models, semi-supervised and unsupervised learning-based models can substantially decrease the need for labeled samples. These methods leverage a combination of both labeled and unlabeled samples, or exclusively utilize unlabeled samples, for network training.

Dong et al. [24] utilized labeled samples to compute cross-entropy loss and unlabeled samples to calculate Kullback-Leibler (KL) divergence. They then jointly optimized these two types of losses to train the network. Yunhao et al. [25] introduced a spatiotemporal autoencoder for AMC, capable of not only realizing unsupervised feature extraction but also enabling semi-supervised classification. Xie et al. [26] designed an improved autoencoder to compress and reconstruct the transformed spectrogram of signals, thereby enabling efficient learning from signals without label information. Following unsupervised pretraining, they utilized a peak clustering algorithm for class determination.

3) TRANSFER LEARNING-BASED MODELS

Transfer learning leverages labeled dataset from source domain with similar distribution to quickly generalize on unlabeled dataset from target domain. This approach transfers useful knowledge from the source domain to the target domain through joint training, enhancing the model's adaptability and efficiency in target domain.

Deng et al. [27] designed a Generative Adversarial Network (GAN)-based transfer learning model for AMC, namely AMRMIDAN. They employed GAN as the backbone to jointly train signal from both the source domain and target domain, thereby achieving successful generalization on unlabeled signals in the target domain. Zhou et al. [28] proposed a transfer learning-based AMC model that utilizes a domain discriminator to align the source and target domains into a unified classification domain. Bu et al. [29] introduced an adversarial-based transfer learning approach to AMC, which mitigates the distribution difference between the original source domain and the downsampled target domain. Deng et al. [30] utilized a domain adversarial neural network to transfer multimodal information from the source domain to the target domain, enhancing the model's adaptability across different domains.

4) FEW-SHOT LEARNING-BASED MODELS

FSL bears resemblance to transfer learning, but it uniquely requires a few labeled samples from the target domain, as opposed to a large number of unlabeled samples. FSL accomplishes rapid generalization in the target domain by constructing analogous few-shot tasks in the source domain, thereby efficiently adapting to new classes with a few labeled samples.

At present, there exists serval well-performed FSL methods, such as Relation Network (RN) [31], and Prototypical Network (PN) [32]. The RN measures the relationship between different samples by calculating the similarity score between feature vectors, enabling the network to learn to distinguish different types. The idea of the PN is to create a prototype representation of each class and classify signals according to the Euclidean distance between the prototype and the test signal.

More recently, research on few-shot AMC has begun to emerge. Zhou et al. [33] introduced an RN into AMC, achieving a classification accuracy of up to 93.2% under 5way-5shot condition in the RadioML.2016.10A dataset. Che et al. [34] proposed a few-shot AMC method named STHFEN, consisting of four sub-modules: the spatial feature extractor, the temporal feature extractor, the Euclidean distance-based classifier, and the hybrid inference module. The core concept of STHFEN for few-shot AMC is akin to that of the PN. Wang et al. [35] introduced a few-shot AMC framework named MsmcNet, which includes IQF modules, 1D-SFP modules, and a classifier. The core idea of MsmcNet for achieving few-shot AMC is to determine the number of 1D-SFP sub-modules using a graph convolution network, significantly reducing the network's search space. Wang et al. [36] employed an episode-based training strategy, utilized the testing set to calculate the attention weight to enhance features separability, and ultimately achieving signal classification through a similarity comparison network.

However, most existing few-shot signal classification models directly apply method from few-shot image classification, which may not be entirely appropriate. Signal classification is profoundly influenced by the Signal-to-Noise Ratio (SNR), which can lead to an excessive presence of low SNR signals in randomly sampled tasks. This makes it challenging for the model to learn effective features and can result in unstable training performance.

C. CONTRIBUTIONS

In this paper, a novel few-shot AMC model is designed. Considering the insufficient feature diversity in traditional DL-based AMC models and the instability in existing few-shot AMC models, our proposed few-shot AMC method combines parallel multi-scale feature extraction network and a graph model-based transductive learning. The key contributions of this paper are summarized as follows:

1) Inspired by the realization that features derived from different input formats possess complementary representational capabilities, we have designed a hybrid parallel spatiotemporal feature extraction module. This module adequately excavate multi-stream signal features, incorporating complex-value convolution and dilated convolution for the feature extraction of complex signals and polar constellation diagrams. Additionally, we have devised various convolutional kernels for feature extraction and aggregation at different scales.

2) Recognizing that the majority of existing few-shot signal classification models experience significant instability during training due to SNR condition imbalances in randomly

sampled tasks, we introduce the graph model-based transductive learning method into few-shot AMC. This approach enables the classifier to predict all test signals within a task simultaneously, aiming to maximize overall classification accuracy. Specifically, our model can propagate labels from labeled signals to unlabeled signals by constructing a unidirectional graph and computing a closed-form solution, effectively leveraging the information contained in unlabeled test signals. To the best of our knowledge, this is the first implementation of a graph model-based transductive learning method in signal classification.

3) To verify the effectiveness of our model, extensive experiments were conducted on the RadioML.2018.01A and RadioML.2016.10A datasets. The experimental results demonstrate that our model outperforms other state-of-the-art methods. Notably, in both 5way-5shot and 10way-5shot under RadioML.2018.01A, our model's classification accuracy surpasses that of other transfer learning model and few-shot learning models by approximately 3%-12%.

The reminder of this paper is organized as follows: Section II details the signal model and problem definition considered in this paper. Section III introduces a few-shot AMC model named Spatial Temporal Transductive Modulation Classifier (STTMC) based on feature extraction module and graph network module. Section IV presents numerical experiments with different setting and provides a comprehensive discussion of the results. Finally, Section V summarizes our model and prospects the future work.

II. SIGNAL MODEL AND PROBLEM DEFINITION

In this section, the signal model used in this paper and the problem definition of few-shot AMC are illustrated.

A. SIGNAL MODEL

Wireless signals, after channel transmission, can generally be represented by the following equation:

$$r(n) = A(n)e^{j(\omega n+\theta)}x(n) + \sigma(n), n = 0, 1, \ldots..N-1 \quad (1)$$

where $x(n)$ represents the modulated signal, $\sigma(n)$ is the additive Gaussian Noise, $A(n)$ denotes the channel gain, ω and θ are the frequency offset and phase offset respectively, and $r(n)$ is the unknown modulated signals at the receiver. Typically, the received signal is preprocessed into I/Q sequences, which can be expressed as follow,

$$I = \{real[r(n)]\}_{n=0}^{N-1}, Q = \{imag[r(n)]\}_{n=0}^{N-1} \quad (2)$$

B. PROBLEM DEFINITION

Generally, there exist three datasets in an FSL problem: a training set, a support set and a testing set. The support set shares the same label space with the testing set, whereas the training set has its own label space. If support set S contains K labeled samples for each of the C categories, the FSL task is referred to as a Cway-Kshot task. Assuming that a model is trained with a support set only, we can in principle obtain a classifier for the testing set. However, the classification accuracy of such a classifier is definitely unsatisfactory. FSL aims to perform meta-learning on the training set, acquiring quick adaptability through a multitude of training tasks T_{train} sampled from the training set. This adaptability is then extend to test tasks T_{test}, enabling the recognition of unseen categories in the support set. In short, T_{train} is used to simulate T_{test}, facilitating the extraction of transferrable knowledge. This process allows the model to achieve improved performance on the testing set.

III. SIGNAL TRANSDUCTIVE PROPAGATION NETWORK

In this section, we illustrate the proposed model in detail. The main structure of STTMC is described in Fig.1, from which we can see that STTMC is composed of two primary modules: a feature extraction module and a graph network module. The former designs a spatiotemporal parallel feature extractor to empower the model with strong feature extraction ability. The latter constructs a graph network-based transductive few-shot classifier for predicting labels in the testing set. It is worth emphasizing that Fig.1 illustrates the forward propagation process within a task, and the complete training process encompasses multiple tasks, aligning with the FSL training strategy. Each task, also known as an episode, collectively forms an epoch.

A. FEATURE EXTRACTION MODULE

As shown in Fig.1, feature extraction module employs a parallel structure, named Parallel Multi-scale Spatiotemporal Network (PMSN), to excavate complementary features from the I/Q sequence and the Accumulated Polar Constellation Diagram (APCD) simultaneously.

For the I/Q sequence input, we design a hybrid series network to extract temporal features. Considering the impact of attenuation and channel fading, which result in an incomplete orthogonal relationship between I-channel and Q-channel, the information in these channels is inherently different. Based on the above considerations, we process the I/Q signal separately.

As shown in Fig.2, the I-channel and Q-channel are first inputted into three 1D convolution blocks with kernel sizes of (1,5), (1,8), and (1,11), respectively. The IQ-channel is processed through three complex-value convolution operators with varying kernel sizes. The real part and imaginary part of the complex kernel are set to (1,5), (1,8), and (1,11), enabling the extraction of multidimensional features. Subsequently, the features extracted from the I-channel, Q-channel and IQ-channel are concatenated. This concatenated output is then fed into a GRU, following a flattening operation, for further feature aggregation. In the aforementioned convolution block, the ReLU activation function is utilized, and the number of filters is set to 16. The max-pooling size for each block is (1,4), aiding in feature suppression. Additionally, the GRU comprises 64 hidden units, and the 3-layer FC network has dimensions of 256, 128, and 64. The output of the final FC layer is regarded as the temporal features.

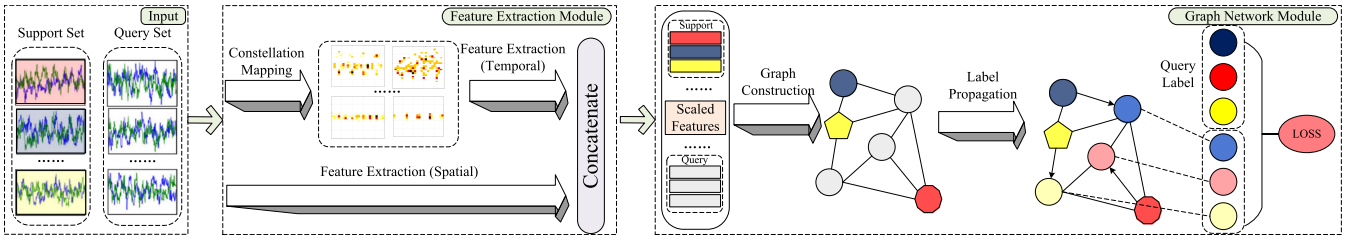


FIGURE 1. Few-shot automatic modulation classification procedure of the proposed STTMC.

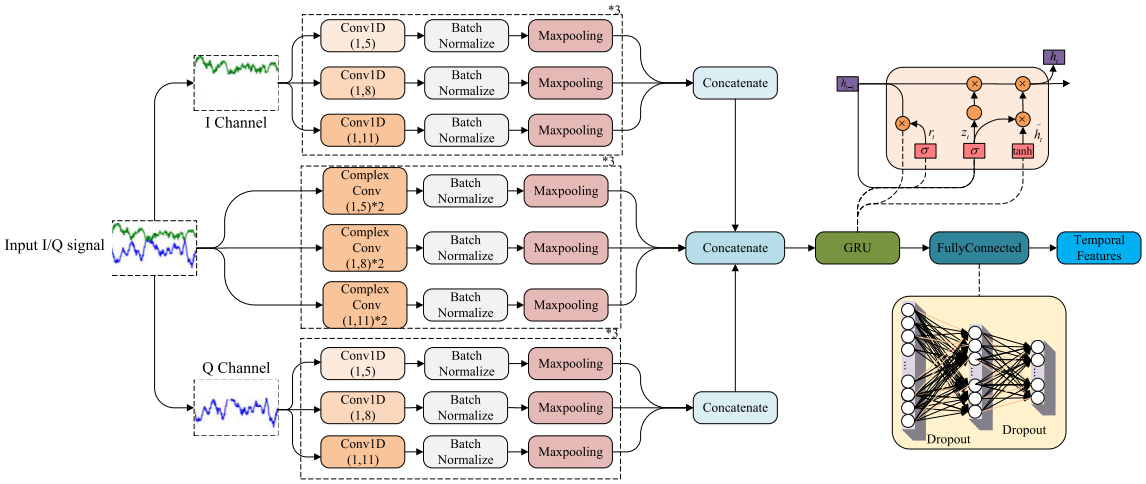


FIGURE 2. The detailed structure of the Temporal feature extractor.

In previous studies, complex signals were typically divided into real parts and complex parts for processing. However, this approach often resulted in the loss of the complex algebraic structures inherent in the signal, and the phase information might not be consistently preserved throughout the network. To address this issue, we introduce complex-value convolution to process IQ signals directly [37]. This method allows for the retention of the signal's complex structure and phase information. The operation of the complex-value convolution operator is illustrated in Fig.3. The complex-value convolution kernel is composed of a real part, denoted as K_{real}, and an imaginary part, denoted as K_{real}. The output of this convolution process is computed using the following equation:

$$\begin{bmatrix} Re(Output) \\ Im(Output) \end{bmatrix} = \begin{bmatrix} K_{imag} \times S_{real} + K_{real} \times S_{imag} \\ K_{real} \times S_{real} - K_{imag} \times S_{imag} \end{bmatrix} \quad (3)$$

where S_{real} and S_{imag} represent the real part and imaginary part of complex signal, respectively.

Beyond the I/Q sequence input, we also utilize the APCD to transform temporal sequences into images, thereby enhancing feature diversity. The constellation diagram is particularly effective in reflecting spatial features of signals, providing an insightful and intuitive visualization of the signal structure. Unlike traditional constellation mapping, APCD first applies L2 normalization to the amplitude A and normalizes phase φ. This approach aligns with the fundamental principle of modulation, which involves encoding messages into the amplitude or phase of the signal

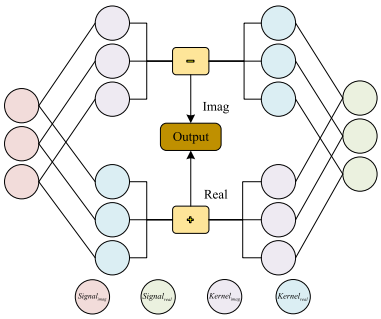


FIGURE 3. Principle of complex-value convolution.

waveform. The conversion from I/Q to A/φ precisely captures this inherent characteristic. By highlighting internal differences between modulation modes, the polar domain can also achieve lower training overhead in feature extraction compared to I/Q domain. Therefore, APCD maps the normalized I/Q sequences to A/φ sequences, increasing the separability between different modulation types. This is employed using the following formulas:

$$A_n = \sqrt{I_n^2 + Q_n^2} \quad (4)$$

$$\varphi_n = \arctan(Q_n / I_n) \quad (5)$$

where I_n and Q_n represent n-th point in I/Q sequence. After converting from the I/Q domain to the A/φ domain, the A/φ sequences are transformed into APCD as inputs for the feature extraction sub-module [38]. Compared to traditional constellation diagrams generated directly from I/Q

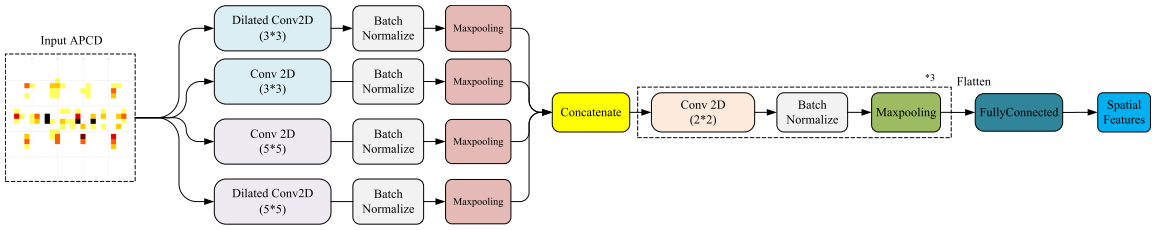


FIGURE 4. The detailed structure of the spatial feature extractor.

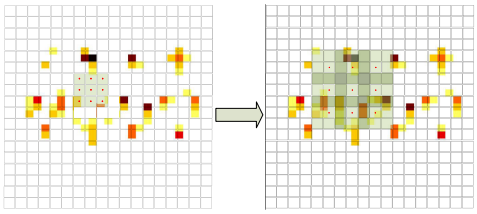


FIGURE 5. Principle of dilated convolution.

sequences, APCDs in the polar domain exhibit stronger classification capabilities, thanks to their resistance to amplitude deterioration. The process of constructing APCDs involves the following steps:

Firstly, the range of the radius axis r_0, r_1, the range of the theta axis θ_0, θ_1, and the image resolution of the two axes p_r, p_θ are determined. Then, each signal point is mapped onto a grid-like image P with coordinates (i, j). The transformation method is described by the following equations:

$$\Delta g_r = (r_1 - r_0)/p_r \tag{6}$$
$$\Delta g_\theta = (\theta_1 - \theta_0)/p_\theta \tag{7}$$
$$i = \lfloor (r\,[n] - r_0)/\Delta g_r \rfloor \tag{8}$$
$$j = \lfloor (\theta\,[n] - \theta_0)/\Delta g_\theta \rfloor \tag{9}$$

The pixel value is set to 1 if any symbol is mapped to the point. Finally, the historical information is accumulated by the following formula,

$$p(i, j) = p(i, j) + 1 \tag{10}$$

It is also worth mentioning that in constructing APCD, we intentionally control the coordinate range to 36×36, further compressing the size of the input image to avoid high computational complexity. To extract the global features from the APCD, we introduce dilated convolution to enhance the perception of global information. As shown in Fig.4, the APCD is firstly processed by four convolution groups with different kernels: dilated conv2D with kernel sizes (3,3) and (5,5), and traditional conv2D with kernel sizes (3,3) and (5,5). Each convolution block has 16 filters and utilizes ReLU as the activation function. The features with different scales are then concatenated in the channel dimension and re-fed into a series of convolution blocks for further feature aggregation. The filter number and max-pooling size in these three convolution blocks are set to 8 and (2,2), respectively. Finally, the output of the convolution block is connected to a two-layer FC network with dimensions 128 and 64. The final 64 units are regarded as the final spatial features.

Dilated convolution is an innovative adaptation of the standard convolutional kernel, incorporating a dilation factor to enlarge the receptive field. The principle of dilated convolution is shown in Fig.5 from which we can observe that it allows the receptive field of the convolution kernel to expand exponentially, while keeping the number of training parameters constant. The expansion of the receptive field is particularly beneficial for connecting distant points in the constellation diagram, making it especially suitable for feature extraction from APCDs. By incorporating dilated convolution, the model can capture broader spatial relationships within the signal data, ensuring that even features that are spatially separated in the APCD can be effectively integrated into the analysis.

After processing through the PMSN, spatial and temporal features are extracted, each with a shape 1×64. These feature are then concatenated in a parallel manner to form a spatial-temporal hybrid feature with a shape of 2×64, ready for use in subsequent module.

B. GRAPH NETWORK MODULE

Next, we present the graph network module in Fig.1, which is used for modulation scheme decision-making through the extracted features. The graph network is employed to depict topological structure composed of nodes and edges. To build a graph within a few-shot task, we first consider the extracted features from PMSN as node values. Before calculating edge values, a zoom factor is obtained by Zoom Factor Calculation Network (ZFCN) for better adaption to different samples. The structure of the ZFCN, as shown in Fig.6, involves a series of convolution blocks with different kernels for further feature extraction and dimension reduction, followed by a 3-layer FC network with 32, 16, and 1 neurons to calculate the zoom factor.

After preprocessing with ZFCN, we create an adjacency matrix on the testing set and support set to measure the similarity of different signals, forming the edges matrix. The similarity between different features is obtained using the following formula:

$$W_{ij} = \exp(-\frac{1}{2} d(\frac{f(x_i)}{\rho_i}, \frac{f(x_j)}{\rho_j})) \tag{11}$$

where x_i and x_j are different input signals, $f(\cdot)$ represents the feature extractor, $d(\cdot, \cdot)$ denotes the distance measure (e.g.,

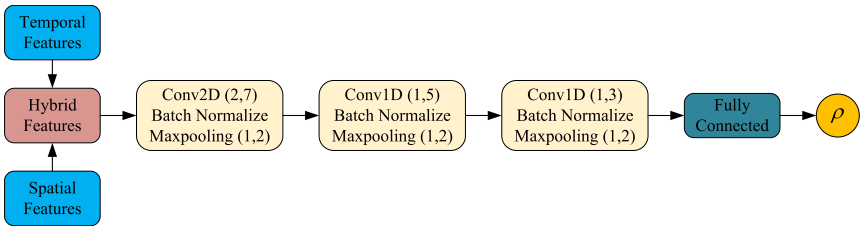


FIGURE 6. Structure of zoom factor calculation network.

Euclidean distance) and ρ is the zoom factor calculated by ZFCN.

With the graph constructed, the correlation between any two samples within a task can be measured, aiding in the pursuit of a global optimal solution rather than a local one.

As mentioned earlier, we consider the elements of the edges matrix as the weights of an undirected graph. Assuming a nonnegative matrix Y represents sample labels with a shape $(C*K+T)*C$, where C and K derive from Cway-Kshot, T represents the testing number. Let $Y_{ij}=1$ if signal x_i belongs to support set and the corresponding label $y_i=j$, otherwise $Y_{ij}=0$. Taking the binary Y as the initial value, labels of the testing set are propagated iteratively according to graph theory using the following formula [39]:

$$F_{t+1}=\alpha UF_t+(1-\alpha)Y \tag{12}$$

where α controls the amount of propagated label information, and F_t represents the predicted labels of the testing set at time t. We can also find from Eq.(12) that each point not only obtains information from its neighbors but also retains initial information. U represents the initialization matrix calculated by $U=M^{-1/2}WM^{1/2}$, where M denotes a dialog matrix whose (i,i)-element equals to the sum of the i-th row of W.

Through continuous iteration, each unlabeled signal is assigned to the class that received the most information, achieving class prediction. According to Limit Theory, the progression F_t can be further calculated by the following formulas:

$$\lim_{t\to\infty}F_{t+1}=\lim_{t\to\infty}(\alpha UF_t+(1-\alpha)Y) \tag{13}$$

$$\lim_{t\to\infty}F_{t+1}=\alpha\lim_{t\to\infty}UF_t+(1-\alpha)Y \tag{14}$$

$$F^*=\alpha UF^*+(1-\alpha)Y \tag{15}$$

$$F^*=\frac{(1-\alpha)}{I-\alpha U}Y \tag{16}$$

F^* can be seen as an approximate closed-form solution for labels of the testing set. In this paper, we directly use this solution as the predicting result without iterations, significantly improving computational efficiency. The final prediction result is then obtained using the softmax function.

$$P(y_i=j|x_i)=\frac{\exp(F^*_{ij})}{\sum_{j=1}^{N}\exp(F^*_{ij})} \tag{17}$$

where y_i represents the predicted label for i-th signal in the union of support and testing set. F^*_{ij} denotes the j-th component of the predicted label from label propagation.

Finally, the classification loss between predicted results and true labels from the union of the support set and the testing set is calculated using cross-entropy function:

$$L=\sum_{i=1}^{N*K+T}\sum_{j=1}^{N}-1(\hat{y_i}==j)\log P(y_i=j|x_i) \tag{18}$$

where $\hat{y_i}==j$ represents the true label of signal and $1(x)$ denotes the binary function. If the argument x is true, $1(x)=1$, otherwise $1(x)=0$.

Notably, the adaptive moment estimation (Adam) optimizer is used for updating network. The pseudocode of STTMC is outlined in Algorithm 1.

Algorithm 1 Workflow of STTMC

INPUT:**C** way, **K** shot, epochs, episodes, Testing Number **T**
for i=1:epochs
Select **C** classes from training set as support set randomly.
for j=1:episodes
$S=\{(x_n,y_n)\}_{n=1}^{K}$ Extract **K** signals from support set with **C** classes randomly.
$Q=\{(x_m,y_m)\}_{m=1}^{M}$ Extract **T** signals from same **C** classes and guarantee $S\cap Q=\emptyset$.
Calculate S_{APCD}, Q_{APCD} by (3)-(9).
Extract Feature $(S_{Spatial},Q_{Spatial})$ and Feature $(S_{Temporal},Q_{Temporal})$ through PMSN.
Calculate zoom factor ρ by ZFCN.
Construct W by (11).
Predict label $y_{prediction}$ of Q by (16)-(17).
Calculate loss between $y_{prediction}$ and y_{true} by (18).
Update parameters by $Adam(L)$.
End for
End for
PREDICTION: Use **C**way-**K**shot labeled signals from testing set to predict unlabeled signals.

To sum up, the decision-making process of STTMC primarily relies on information propagation within a graph, achieving global optimization within a task. Unlabeled signals in testing set are extensively utilized for

decision-making. Moreover, a closed-form solution is calculated for decision-making, which not only eliminates iterations but also reduces the need for training a FC network (traditional classification head), thereby decreasing the number of trainable parameters and facilitating the actual deployment of the model.

IV. EXPERIMENTS

In this section, we evaluate the proposed few-shot AMC model. Firstly, we explore the convergence performance of our model in various settings, then analyze the network structure through numerous ablation tests. We also investigate the performance of our model under different ways and shots. Lastly, we compare our module with serval state-of-the-art AMC approaches in the same settings.

A. DATASET

All experiments are conducted on the RadioML.2018.01A and RadioML.2016.10A datasets, which are generated in a laboratory environment. The RadioML.2018.01A dataset includes 24 modulation types listed as follows: OOK, 4ASK, 8ASK, BPSK, QPSK, 8PSK, 16PSK, 32PSK, 16APSK, 32APSK, 64APSK, 128APSK, 16QAM, 32QAM, 64QAM, 128QAM, 256QAM, AM-SSB-WC, AM-SSB-SC, AM-DSB-WC, AM-DSB-SC, FM, GMSK, OQPSK. The data dimension of RadioML.2018.01A is 2*1024. The SNR for each modulation scheme ranges from -20 to 30 dB with an interval of 2 dB. The dataset contains a total of 255,5904 signals, with 4096 samples per SNR for each modulation scheme. For evaluation, we consider SNR from -20 to 20dB.

The RadioML.2016.10A dataset comprises 11 modulation types of complex-valued I/Q samples with a length of 128, including WBFM, AM-DSB, AM-SSB, BPSK, CPFSK, GFSK, 4-PAM, 16-QAM, 64-QAM, QPSK and 8PSK. The samples are generated for 20 different SNR levels from -20dB to 18dB in 2dB steps. The dataset has a total of 220,000 signals, with each mode having 1000 samples per SNR.

B. EXPERIMENTS SETTING

As discussed in Section.II.B, the training process for FSL differs from traditional deep learning algorithm. It adopts an episode-based method to optimize the network. The core concept of FSL training method is to construct a task set through repeated sampling from the training set, simulating the real test task. In each epoch, a series of training tasks $T^{train} = \{T_1^{train}, T_2^{train}, \ldots\ldots T_N^{train}\}$ are randomly sampled from the training set, endowing the model with few-shot classification ability through extensive training tasks. Each task T_i^{train} is called an episode. After training the network with T^{train}, we can directly adapt to the test tasks T^{test}. In each task T, there exists a support set S with labeled signals and a testing set Q with unlabeled signals. Moreover, S contains K signals of C types is called Cway-Kshot problem.

In this paper, we primarily explore the following four conditions on the RadioML.2018.01A: 5way-5shot, 5way-10shot, 10way-5shot and 10way-10shot. The 5way includes: FM, GMSK, 32QAM, 16APSK and 32PSK. The 10way includes: FM, GMSK, OQPSK, BPSK, 8PSK, AM-SSB-DC, 4ASK, QPSK, OOK and 16QAM. We use the remaining (24-C) categories in the RadioML.2018.01A dataset for training, and then directly test the aforementioned 5 or 10 categories.

Additionally, we conduct 3way-3shot experiment on the RadioML.2016.10A. The 3way includes two groups: group1 comprises BPSK, AM- SSB and 8PSK, while Group2 includes QAM16, PAM4 and QAM64. We use the remaining 8 categories in the RadioML.2016.10A dataset for training, and then directly test the aforementioned 3 categories.

Different from the traditional predicting method, STTMC predicts all test signals simultaneously to seek global optimization within a task. Therefore, the number of testing sample in each episode is a key parameter. In this paper, we set the number of testing signals T to 30, unless otherwise specified. The label information control factor α is set to 0.99.

For network training and inference, an Nvidia P5200 GPU is used, and the CPU is Intel(R) Xeon(R) E-2186M. The training epochs are set to 300 with a learning rate of 0.001. In each epoch, the number of episodes is set to 100 and employ the Adam optimizer.

C. CONVERGENCE PERFORMANCE

In this sub-section, we explore the convergence performance of our model under different conditions, focusing on the following experiments: 5way-5shot, 5way-10shot, 10way-5shot, and 10way-10shot. Fig.7(a)-7(f) display the loss and accuracy curves of our model. From the experiment result, we observe that all curves exhibit high variability initially. This is attributed to the mixing of training signals with different SNR during training process, which may result in all sampled signals are of poor quality in an episode, leading to the degradation of model performance. However, our model converges stably in various situations after sufficient optimization. Interestingly, we can also note that the validation accuracy and loss sometimes outperform those in the training process. This unusual phenomenon is due to the uncertainty of the SNR of the training samples extracted by few-shot model in each episode.

Furthermore, we observe that the impact of the K value on our model is not obvious. In our experiment, the performance of 5-shot and 10-shot seems to have little difference. In other words, our model demonstrates good adaptability to various few-shot conditions and performs well even with only 5-shots required.

D. ABLATION TESTS

Following the complete training of our model in Section.IV.C, unseen classes in the testing set can be directly predicted using the transductive method with only Cway-Kshot samples. The test process still follows the Cway-Kshot strategy. It is also worth mentioning that The samples with varying SNR in test tasks are randomly selected, and their average value is taken as the final result.

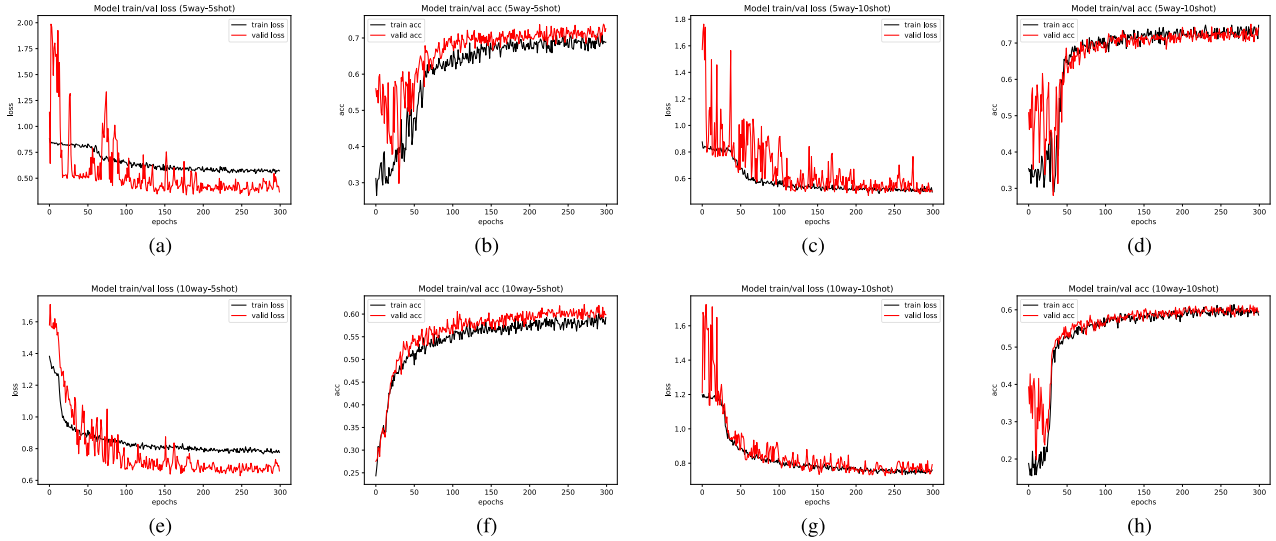


FIGURE 7. The training and validation accuracies and losses of STTMC under 5way-Kshot and 10way-Kshot. (a)Loss curve under 5way-5shot (b) Accuracy curve under 5way-5shot (c)Loss curve under 5way-10shot (d) Accuracy curve under 5way-10shot (e)Loss curve under 10way-5shot (f) Accuracy curve under 10way-5shot (g)Loss curve under 10way-10shot (h) Accuracy curve under 10way-10shot.

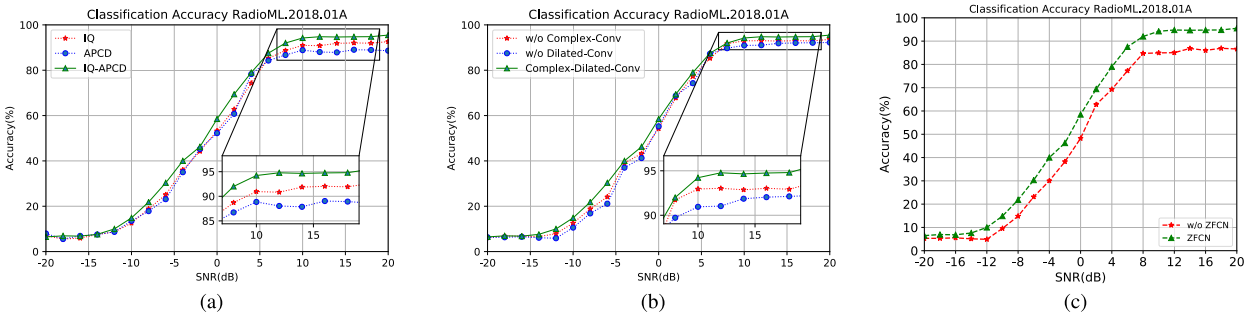


FIGURE 8. Comprehensive ablation analysis.(a) Ablation test with different input.(b) Ablation test with different convolution kernel.(c) Ablation test of ZFCN.

To verify the advantages of our proposed network structure, we conduct serval ablation tests comparing classification accuracy. In this section, we select 10way-5shot as the base experimental setting. First, we compare classification accuracy using different inputs for the proposed module. We employ IQ, APCD, and IQ-APCD as input, respectively, and the simulation result is shown in Fig.8(a). From the result we can conclude that the average top-10 classification accuracy of IQ-APCD is 3% and 6% higher than that of IQ and APCD, respectively. This indicates that hybrid parallel features extracted from different aspects are complementary.

Secondly, we explore the effectiveness of complex-value convolution and dilated convolution. In this test, we replace these specialized kernels with traditional conv2d to observe the impact on performance. The result, as depicted in Fig.8(b), reveals that average top-10 classification accuracy using complex-dilated-conv is 2% and 3% higher than that of without-complex-conv and without-dilated-conv, respectively. This finding substantiates that both complex-conv and dilated-conv contribute positively to feature extraction. It also demonstrates that the features extracted by these different convolution kernels are distinct and non-redundant, each contributing uniquely to the overall performance of the model.

Finally, to validate the necessity of the ZFCN, we conduct a comparative analysis by deactivating the ZFCN. Instead, we directly utilize the extracted feature to calculate metric matrix W. The simulation result is shown in Fig.8(c), from which we can see that the performance degrades seriously after removing ZFCN. The average accuracy gap between is up to 7.54%. We believe that if we do not introduce a zoom factor, it would be difficult to measure different feature vectors at the same scale, which directly affects the rationality of the metric matrix and leads to heavily performance degradation.

E. CWAY-KSHOT EXPERIMENTS

In this part, we analyze the impact of varying shot values on the overall average classification accuracy of the model. We test different shot values from 1 to 10 under both 5-way and 10-way. From Fig.9 we can see that even there is only

TABLE 1. Average classification accuracy on RadioML.2018.01A with different testing number.

	RadioML.2018.01A 10way-5shot									
Testing Number	5	10	15	20	25	30	35	40	45	50
Average accuracy	49.17%	51.23%	52.36%	52.97%	53.97%	**54.58%**	54.11%	55.02%	54.73%	54.70%

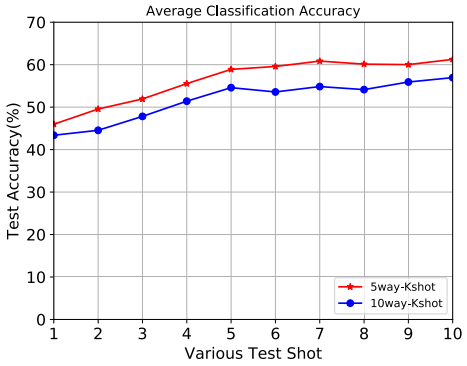


FIGURE 9. Average accuracy with different shot-value.

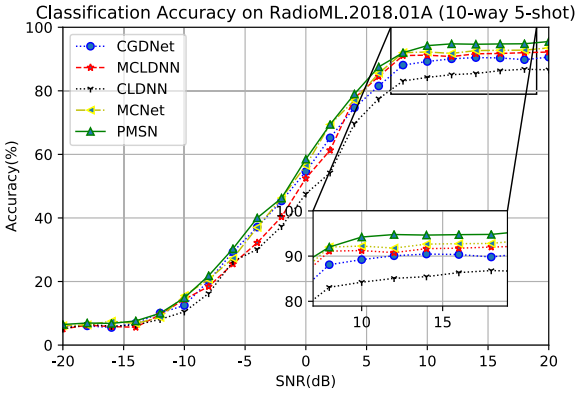


FIGURE 10. Comparison of different feature extraction network.

1 sample for training, our model still can achieve 45.9% classification accuracy in 5way and 43.4% in 10way. As the shot value increases, the performance of our model is gradually improved. Notably, when the shot value reaches 5, the performance stabilizes, oscillating around 60% for 5way and 54% for 10way. Overall, after conducting extensive experiments across different shot values, we conclude that our model exhibits good stability and can adapt well to various conditions. Notably, the number of testing samples is consistently set to 30 during these training process.

Besides, we investigate the impact of the number of testing samples on our model. We use 10way-5shot as baseline, altering the number of testing samples from 5 to 50 in increments of 5. The simulation result is shown in Table. 1. It can be observed that the average accuracy is up to 49.17% when the number of testing samples is equal to 5, which is quite satisfactory in a few-shot condition. As the number of testing samples increases, the performance of our model tends to be better. This enhancement is attribute to the transductive method's advantage of predicting all signals simultaneously to achieve optimal global classification performance. However, beyond 30 testing samples, the performance of our model tends to stabilize. Consequently, we set the number of testing samples to 30, which is also computational economical for each episode.

F. COMPARISON EXPERIMENTS

In this subsection, we conduct a comprehensive comparison with other advanced models to further demonstrate the superiority of our model. Initially, we replace our feature extraction module, PMSN, with other typical feature extraction networks, namely CGDNet [40], MCLDNN [41], CLDNN [42] and MCNet [43]. To maintain compatibility with the ZFCN module, we configure the final output of all comparison networks as a FC layer with 128 neurons, which is then reshape into 2×64 for ZFCN. All comparative simulations are carried out under the 10way-5shot condition. The result, as shown in Fig.10, indicate that PMSN has a stronger feature extraction capability than other models. It is noteworthy that the compared models only utilize IQ sequences as input, whereas PMSN employs hybrid inputs to enhance feature diversity.

Furthermore, it is also important to investigate other few-shot methods. We select serval representative few-shot methods for comparison, namely CNN-AMC(Transfer Learning) [44], Prototypical Network [32], AMCRN [33], Relation Network [31], IAFnet [36], and STHFEN [34]. We mainly compare 5-way and 10-way conditions with varying shot values. For CNN-AMC, we utilize the training set for pretraining and then fine-tuned the top layer with the testing set. For RN and PN, we use our feature extraction module as the backbone, replacing the graph network module with a relation metric module and a prototypical construction module. Besides, it is also worth mentioning that the query number is set to 30, and other parameter settings are keep consistent across all few-shot models.

The simulation result of average classification accuracy is shown in Fig.11. From the histogram, it is evident that our model achieves better average classification accuracy under various way and shot values. Specifically, CNN-AMC significantly underperforms other Meta Learning-based methods. This is because transfer learning primarily addresses few-shot problems by transferring existing knowledge to new tasks. However, when there are significant differences in the target task, transfer learning may not transfer useful knowledge effectively, leading to performance degradation. Meta learning appears to be a better choice for quickly adapting to new classification tasks. Compared with other few-shot methods, STTMC still maintains certain advantages. Notably, the average classification advantages of STTMC under 10way are better than those under 5way, further confirming our

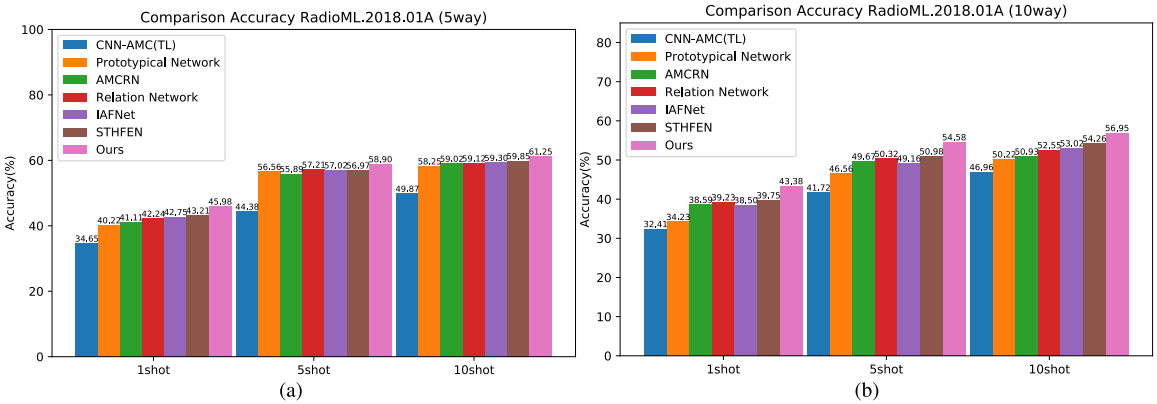


FIGURE 11. Average classification accuracy of STTMC and other FSL methods under different shot value in RadioML.2018.01A. (a) 5way (b) 10way.

TABLE 2. Comparison of computational complexity.

Method	Capacity (number of parameters)	Flops
CGDNet	241,159	8,973,526
MCLDNN	525,652	19,417,584
CLDNN	365,239	12,787,017
MCNet	221,841	7,762,216
Prototypical Network	732,703	28,575,417
Relation Network	517,942	19,940,767
STTMC	339,059	12,172,218

hypothesis that simultaneous prediction is beneficial for improving the overall performance. This is because the more categories to be predicted, the larger the graph to be constructed, leading to the richer information contained in the graph.

After comparing STTMC with other models in terms of recognition performance, we further explore the computational complexity. From Table. 2, it is clear that STTMC has the strongest feature extraction ability with relatively moderate trainable parameters and flops. Additionally, we find that the capacity of PN-based and RN-based few-shot methods are significantly greater than few-shot transductive learning method, and their classification performance is also inferior to few-shot transductive learning method.

G. SPECIFIC CLASSIFICATION PERFORMANCE

To further demonstrate the specific classification performance of STTMC, we analyze the classification accuracy for each modulation type. Fig.12 presents the classification accuracy for individual modulation types under the RadioML.2018.01A dataset in both 10way-5shot and 5way-5shot scenarios. Additionally, Fig.13 shows the classification accuracy for individual modulation types under the RadioML.2016.10A dataset in the 3way-3shot scenario.

1) 10WAY-5SHOT

The classification results in Fig.12(a) show that the accuracy for each modulation type can exceed 90% when the SNR is above 8dB. Signals with simpler waveform and structure, such as FM and BPSK, are easier to classify even at low SNR levels. Fig.12(b) and Fig.12(c) display the confusion matrices of our module at 10dB and 0dB, respectively. At 10dB, it can be observed that other five modulation modes are accurately classified, with minor errors in 8PSK, AM-SSB-DC, 4ASK, QPSK, and 16QAM, and the overall average accuracy can achieve 94%.

When the SNR drops to 0bB, our model tends to misclassify some confusing modulation types, yet the average accuracy remains at a respectable 59%, which is quite satisfactory for a 10way-5shot scenario. It is also note that the confusing signals are also prone to misclassification even with sufficient training samples. We suppose that it is not feasible to achieve accurate signal recognition using only a few samples at low SNR levels. The significant randomness of signals under low SNR makes it challenging to learn general features that can be used for modulation classification with limited samples.

2) 5WAY-5SHOT

The accuracy curves in Fig.12(d) reveal that, except for some confusion between 32QAM and 32PSK, the other three modulation types achieve 100% accurate classification when the SNR exceeds 10dB. Further investigation of the confusion matrices in Fig.12(e) and Fig.12(f) shows that confusion primarily occurs between these two modulation types as the SNR degrades. The likely cause of this confusion is the high similarity between the constellations of these two modulation types at low SNR levels, which poses challenges to the feature extraction of our model.

3) 3WAY-3SHOT

To further validate the effectiveness of STTMC, we conduct two 3way-3shot experiments on the RadioML.2016.10A dataset. The simulation result indicates that STTMC still adapts well to the RadioML.2016.10A dataset. In group 1, as shown in Fig.13(a)-(c), STTMC accurately recognizes AM-SSB, BPSK, and 8PSK when the SNR is above 0dB. However, with a continuing decrease in SNR, both BPSK and 8PSK tend to be misclassified as AM-DSB. This couple of confusion is also occurred in many

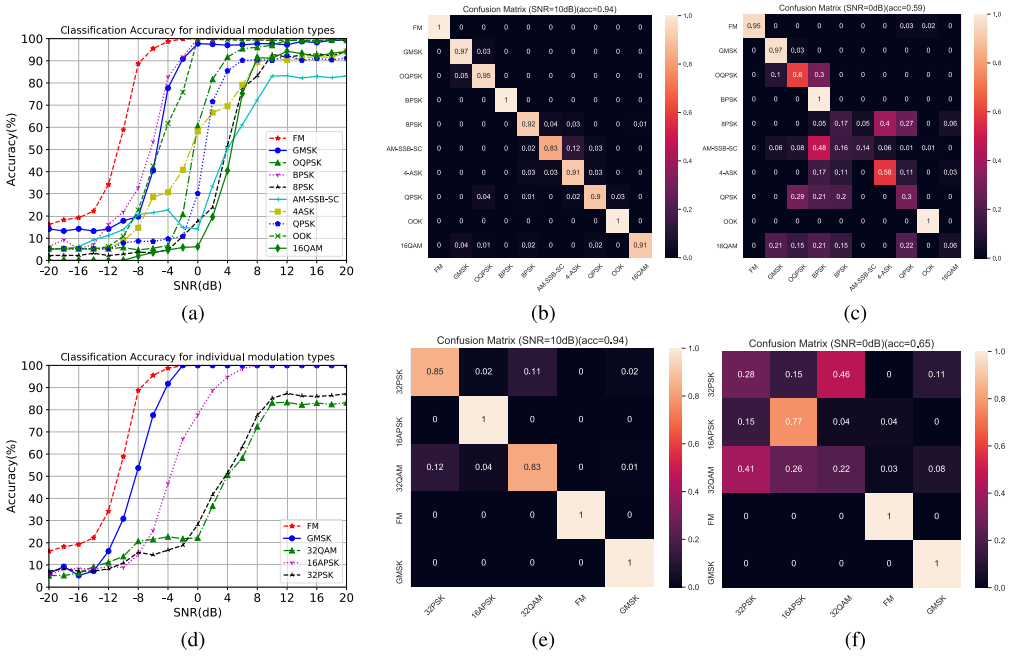


FIGURE 12. **Classification performance of RadioML2018.01A under different experiment settings. (a) Individual modulation types per SNR under 10way-5shot. (b) Confusion matrix of RadioML.2018.01A at 10dB under 10way-5shot. (c) Confusion matrix of RadioML.2018.01A at 0dB under 10way-5shot. (d) Individual modulation types per SNR under 5way-5shot. (e) Confusion matrix of RadioML.2018.01A at 10dB under 5way-5shot. (f) Confusion matrix of RadioML.2018.01A at 0dB under 5way-5shot.**

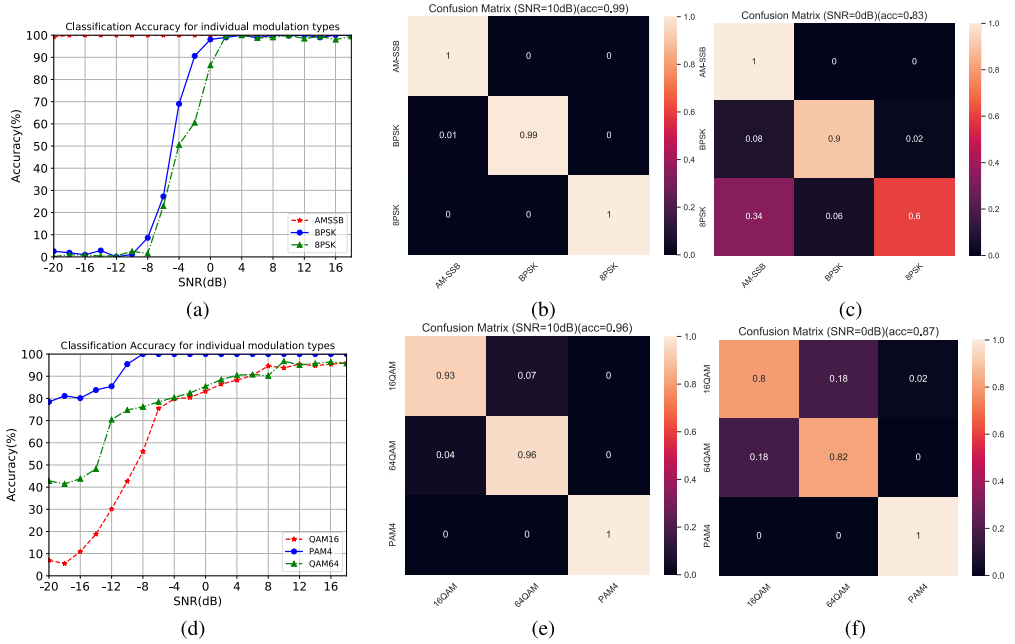


FIGURE 13. **Classification performance of RadioML2016.10A under different experiment settings. (a) Individual modulation types per SNR under 3way-3shot of RadioML.2016.10A-group1. (b) Confusion matrix of RadioML.2016.10A-group1 at 10dB under 3way-3shot. (c) Confusion matrix of RadioML.2016.10A-group1 at 0dB under 3way-3shot.(d) Individual modulation types per SNR under 3way-3shot of RadioML.2016.10A-group2. (e)Confusion Matrix of RadioML.2016.10A-group2 at 10dB under 3way-3shot. (f) Confusion matrix of RadioML.2016.10A-group2 at 0dB under 3way-3shot.**

DL-based AMC models with sufficient samples, which might be attributed to the analog modulation characteristics of AM-DSB, resembling noise distribution. In group 2, STTMC effectively distinguishes PAM4 and high-order QAM modes. As depicted in Fig.13(d)-(f), the average classification accuracy exceeds 87% when the SNR is above 0dB, and reaches

96% at an SNR of 10dB. It is noteworthy that distinguishing (16QAM, 64AQM) is challenging for many DL-based AMC models. Therefore, the classification performance of STTMC, achieved with only three training signals per class, is impressively superior.

V. CONCLUSION

In this paper, we have introduced the graph model-based transductive learning method into few-shot AMC originally. Our model, namely Spatial Temporal Transductive Modulation Classifier (STTMC), is composed of two primary modules: a feature extraction module and a graph network module. Notably, STTMC employs complex-value convolution, dilated convolution and a hybrid structure for better feature extraction. A significant aspect of STTMC is its use of a closed-form solution calculated by a graph, to predict all test signals simultaneously rather than individually, thereby improving the overall classification performance of the model. The experimental results demonstrate that STTMC achieves higher classification accuracy compared to serval state-of-the-art few-shot models and a transfer learning model. However, we also observed that STTMC's performance rapidly degrades at low SNR levels, particularly for certain confusing signals. In the future, there are two potential avenues for further improvement. On one hand, a denoising module could be integrated into feature extraction module to mitigate noise interference. On another hand, the training strategy of meta learning could be optimized. Specifically, the training strategy could be designed to intentionally construct tasks that are tailored to adapt to the classification of low SNR signals in advance, rather than relying on random sampling methods. These enhancements could significantly improve the model's robustness and effectiveness, particularly in challenging low SNR scenarios.

REFERENCES

[1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[2] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing [review article]," *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 55–75, Aug. 2018.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[4] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. Int. Conf. Eng. Appl. Neural Netw.* Cham, Switzerland: Springer, 2016, pp. 213–226.

[5] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.

[6] B. Dong et al., "A lightweight decentralized-learning-based automatic modulation classification method for resource-constrained edge devices," *IEEE Internet Things J.*, vol. 9, no. 24, pp. 24708–24720, Dec. 2022.

[7] Z. Ke and H. Vikalo, "Real-time radio technology and modulation classification via an LSTM auto-encoder," *IEEE Trans. Wireless Commun.*, vol. 21, no. 1, pp. 370–382, Jan. 2022.

[8] L. Zhang, X. Yang, H. Liu, H. Zhang, and J. Cheng, "Efficient residual shrinkage CNN denoiser design for intelligent signal processing: Modulation recognition, detection, and decoding," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 97–111, Jan. 2022.

[9] Z. Chen et al., "SigNet: A novel deep learning framework for radio signal classification," *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 2, pp. 529–541, Jun. 2022.

[10] C. Hou, G. Liu, Q. Tian, Z. Zhou, L. Hua, and Y. Lin, "Multisignal modulation classification using sliding window detection and complex convolutional network in frequency domain," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 19438–19449, Oct. 2022.

[11] H. Zhang, M. Huang, J. Yang, and W. Sun, "A data preprocessing method for automatic modulation classification based on CNN," *IEEE Commun. Lett.*, vol. 25, no. 4, pp. 1206–1210, Apr. 2021.

[12] S. Huang et al., "Automatic modulation classification using gated recurrent residual network," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7795–7807, Aug. 2020.

[13] S. Chang, R. Zhang, K. Ji, S. Huang, and Z. Feng, "A hierarchical classification head based convolutional gated deep neural network for automatic modulation classification," *IEEE Trans. Wireless Commun.*, vol. 21, no. 10, pp. 8713–8728, Oct. 2022.

[14] 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.

[15] W. Zhang, X. Yang, C. Leng, J. Wang, and S. Mao, "Modulation recognition of underwater acoustic signals using deep hybrid neural networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 8, pp. 5977–5988, Aug. 2022.

[16] Z. Zhang, H. Luo, C. Wang, C. Gan, and Y. Xiang, "Automatic modulation classification using CNN-LSTM based dual-stream structure," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13521–13531, Nov. 2020.

[17] F. Zhang et al., "An efficient deep learning model for automatic modulation recognition based on parameter estimation and transformation," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3287–3290, Oct. 2021.

[18] J. Cai, F. Gan, X. Cao, and W. Liu, "Signal modulation classification based on the transformer network," *IEEE Trans. Cognit. Commun. Netw.*, vol. 8, no. 3, pp. 1348–1357, Sep. 2022.

[19] Q. Zheng, P. Zhao, H. Wang, A. Elhanashi, and S. Saponara, "Fine-grained modulation classification using multi-scale radio transformer with dual-channel representation," *IEEE Commun. Lett.*, vol. 26, no. 6, pp. 1298–1302, Jun. 2022.

[20] X. Zhang et al., "NAS-AMR: Neural architecture search-based automatic modulation recognition for integrated sensing and communication systems," *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 3, pp. 1374–1386, Sep. 2022.

[21] L. Huang et al., "Visualizing deep learning-based radio modulation classifier," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 47–58, Mar. 2021.

[22] Z. Liang, M. Tao, L. Wang, J. Su, and X. Yang, "Automatic modulation recognition based on adaptive attention mechanism and ResNeXt WSL model," *IEEE Commun. Lett.*, vol. 25, no. 9, pp. 2953–2957, Sep. 2021.

[23] T. Wang et al., "Deep learning based modulation recognition with multi-cue fusion," *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1757–1760, Aug. 2021.

[24] 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.

[25] S. Yunhao et al., "ConvLSTMAE: A spatiotemporal parallel autoencoders for automatic modulation classification," *IEEE Commun. Lett.*, vol. 26, no. 8, pp. 1804–1808, Aug. 2022.

[26] C. Xie, L. Zhang, and Z. Zhong, "Few-shot unsupervised specific emitter identification based on density peak clustering algorithm and meta-learning," *IEEE Sensors J.*, vol. 22, no. 18, pp. 18008–18020, Sep. 2022.

[27] W. Deng, X. Wang, Z. Huang, and Q. Xu, "Modulation classifier: A few-shot learning semi-supervised method based on multimodal information and domain adversarial network," *IEEE Commun. Lett.*, vol. 27, no. 2, pp. 576–580, Feb. 2023.

[28] Q. Zhou, R. Zhang, Z. Jing, and X. Jing, "Semi-supervised-based automatic modulation classification with domain adaptation for wireless IoT spectrum monitoring," *Frontiers Phys.*, vol. 11, p. 252, Mar. 2023.

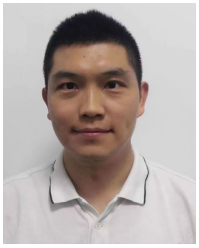
[29] K. Bu, Y. He, X. Jing, and J. Han, "Adversarial transfer learning for deep learning based automatic modulation classification," *IEEE Signal Process. Lett.*, vol. 27, pp. 880–884, 2020.

[30] W. Deng, Q. Xu, S. Li, X. Wang, and Z. Huang, "Cross-domain automatic modulation classification using multimodal information and transfer learning," *Remote Sens.*, vol. 15, p. 3886, Aug. 2023.

[31] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.
[32] J. Wang and Y. Zhai, "Prototypical Siamese networks for few-shot learning," in *Proc. IEEE 10th Int. Conf. Electron. Inf. Emergency Commun. (ICEIEC)*, Jul. 2020, pp. 178–181.
[33] Q. Zhou, R. Zhang, J. Mu, H. Zhang, F. Zhang, and X. Jing, "AMCRN: Few-shot learning for automatic modulation classification," *IEEE Commun. Lett.*, vol. 26, no. 3, pp. 542–546, Mar. 2022.
[34] J. Che, L. Wang, X. Bai, C. Liu, and F. Zhou, "Spatial–temporal hybrid feature extraction network for few-shot automatic modulation classification," *IEEE Trans. Veh. Technol.*, vol. 71, no. 12, pp. 13387–13392, Dec. 2022.
[35] Y. Wang, J. Bai, Z. Xiao, H. Zhou, and L. Jiao, "MsmcNet: A modular few-shot learning framework for signal modulation classification," *IEEE Trans. Signal Process.*, vol. 70, pp. 3789–3801, 2022.
[36] H. Wang, B. Wang, and Y. Li, "IAFNet: Few-shot learning for modulation recognition in underwater impulsive noise," *IEEE Commun. Lett.*, vol. 26, no. 5, pp. 1047–1051, May 2022.
[37] Y. Tu, Y. Lin, C. Hou, and S. Mao, "Complex-valued networks for automatic modulation classification," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 10085–10089, Sep. 2020.
[38] C.-F. Teng, C.-Y. Chou, C.-H. Chen, and A.-Y. Wu, "Accumulated polar feature-based deep learning for efficient and lightweight automatic modulation classification with channel compensation mechanism," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15472–15485, Dec. 2020.
[39] Y. Liu et al., "Learning to propagate labels: Transductive propagation network for few-shot learning," 2018, *arXiv:1805.10002*.
[40] J. N. Njoku, M. E. Morocho-Cayamcela, and W. Lim, "CGDNet: Efficient hybrid deep learning model for robust automatic modulation recognition," *IEEE Netw. Lett.*, vol. 3, no. 2, pp. 47–51, Jun. 2021.
[41] J. Xu, C. Luo, G. Parr, and Y. Luo, "A spatiotemporal multi-channel learning framework for automatic modulation recognition," *IEEE Wireless Commun. Lett.*, vol. 9, no. 10, pp. 1629–1632, Oct. 2020.
[42] N. E. West and T. O'Shea, "Deep architectures for modulation recognition," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Mar. 2017, pp. 1–6.
[43] T. Huynh-The, C. Hua, Q. Pham, and D. Kim, "MCNet: An efficient CNN architecture for robust automatic modulation classification," *IEEE Commun. Lett.*, vol. 24, no. 4, pp. 811–815, Apr. 2020.
[44] F. Meng, P. Chen, L. Wu, and X. Wang, "Automatic modulation classification: A deep learning enabled approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10760–10772, Nov. 2018.

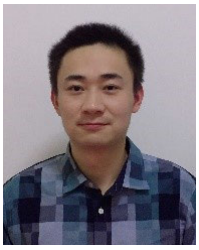
YUNHAO SHI received the B.S. degree in information engineering from Xidian University, Xi'an, China, in 2018, and the M.S. degree in cyberspace security from Air Force Engineering University, Xi'an, in 2020, where he is currently pursuing the Ph.D. degree with the Information and Navigation College. His research interests include deep learning, signal processing, and few-shot learning.

HUA XU received the B.S. and M.S. degrees in communication engineering from Air Force Engineering University, Xi'an, China, in 2001, and the Ph.D. degree in communication signal processing from Information Engineering University, Zhengzhou, China, in 2005. He is currently a Professor with Air Force Engineering University. His research interests include communication signal processing, blind signal processing, and communication countermeasures.



ZISEN QI received the B.S., M.S., and Ph.D. degrees in communication engineering and communication signal processing from Air Force Engineering University, Xi'an, China, in 2011. He is currently an Associate Professor with Air Force Engineering University. His research interests include communication systems and array antenna.

YUE ZHANG received the B.S. degree in electronic information science and technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2012, the M.S. degree in computational intelligence from Sheffield University, Sheffield, U.K., in 2013, and the Ph.D. degree in communication and information systems from Xidian University in 2018. He was an Associate Professor with the Unmanned System Research Institute, Northwestern Polytechnical University, and currently a Post-Doctoral Researcher with Air Force Engineering University. His research interests include machine learning, multi-agent reinforcement learning, deep reinforcement learning, game theory, the Internet of Things, intelligent transportation systems, and big data.



DAN WANG was born in Anhui, China, in 1990. He received the M.S. degree in information and communication engineering from the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, China, in 2013, and the Ph.D. degree in information and communication engineering from the Institute of Information and Navigation, Air Force Engineering University (AFEU), Xi'an, China, in 2022. He is currently a Lecturer with the Radar and Signal Processing Laboratory, Institute of Information and Navigation, AFEU, and also with the Collaborative Innovation Center of Information Sensing and Understanding. His research interests include signal processing and information countermeasure technology.

LEI JIANG received the B.S., M.S., and Ph.D. degrees in communication engineering and communication signal processing from Air Force Engineering University, Xi'an, China, in 2005. He is currently an Associate Professor with Air Force Engineering University. His research interests include communication systems, electronic countermeasures, and pattern recognition.