IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2018.DOI

A Novel Attention Cooperative Framework for Automatic Modulation Recognition

SHIYAO CHEN, YAN ZHANG,(Member, IEEE), ZUNWEN HE,(Member, IEEE), Jinbo Nie, and Wancheng Zhang,

[1]School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China. (e-mail: {schen,zhangy,hezunwen,niejinbo,zhangwancheng}@bit.edu.cn)

Corresponding author: Yan Zhang (e-mail: zhangy@bit.edu.cn).

The authors acknowledge the support from National Natural Science Foundation of China (No. 61871035, No. 61201192), and Ericsson company.

ABSTRACT Modulation recognition plays an indispensable role in the field of wireless communications. In this paper, a novel attention cooperative framework based on deep learning is proposed to improve the accuracy of the automatic modulation recognition (AMR). Within this framework, a convolutional neural network (CNN), a recurrent neural network (RNN), and a generative adversarial network (GAN) are constructed to cooperate in AMR. A cyclic connected CNN (CCNN) is designed to extract spatial features of the received signal, and a bidirectional RNN (BRNN) is constructed for obtaining temporal features. To take full advantage of the complementarity and relevance between the spatial and temporal features, a fusion strategy based on global average and max pooling (GAMP) is proposed. To deal with different influence levels of the signal feature maps, we present the attention mechanism in this framework to realize recalibration. Besides, modulation recognition based on deep learning requires numerous data for training purposes, which is difficult to achieve in practical AMR applications. Therefore, an auxiliary classification GAN (ACGAN) is developed as a generator to expand the training set, and we modify the loss function of ACGAN to accommodate the processing of the actual in-phase and quadrature (I/Q) signal data. Considering the difference in distribution between generated data and real data, we propose a novel auxiliary weighing loss function to achieve higher recognition accuracy. Experimental results on the dataset RML2016.10a show that the proposed framework outperforms existing deep learning-based approaches and achieves 94% accuracy at high signal to noise ratio (SNR).

INDEX TERMS Automatic modulation recognition (AMR), attention mechanism, convolutional neural network (CNN), generative adversarial network (GAN), recurrent neural network (RNN).

I. INTRODUCTION

AUTOMATIC modulation recognition (AMR), referring to the identification of the modulation type of the received signal, is essential for reducing the protocol overhead and ensuring the reliable performance of the communication system in non-cooperative scenarios. On the one hand, it is of considerable significance to achieve AMR to promote spectrum efficiency and transmission reliability for the link adaptation in next-generation communications [1]. On the other hand, obtaining the modulation types of hostile signals to develop the interfering or anti-interfering strategy is an essential application in the field of military communications.

Maximum likelihood (ML) hypothesis testing methods based on decision theory and statistical pattern recognition (PR) methods based on feature extraction are two primary categories of AMR solutions [2]. The former is based on the likelihood function, and AMR is completed by comparing the likelihood ratio with an appropriate threshold theoretically. ML methods have the best performance according to the Bayesian minimum misjudgment cost criterion and are applied in [3]–[5]. However, the calculation of the statistics is complex, and some prior probability information is required. In most information interception scenarios, modulation recognition has to be completed in a blind manner [2], i.e., there is no prior information that can be utilized. In contrast, the latter is less subject to prior information and

has a lower complexity. These PR methods based on feature extraction can achieve sub-optimal performance and broad applicability when adequately designed.

For PR methods, AMR can be regarded as a multi-pattern classification problem with multiple parameters. The process can be divided into two stages, extracting features and training classifiers. A variety of features were extracted and employed in [6]–[18], containing amplitude with phase and carrier frequency [6], instantaneous features [7], high-order statistical features [8], [9], cyclic spectrum parameters [10], [11], bispectrum features [12], wavelet features [13], [14] and constellation diagram [15], [16]. For the choice of the training classifier, the classifiers based on machine learning like support vector machine (SVM) in [6], [13], [17], [18], decision tree in [7], [8], [14], k nearest neighbor (KNN) in [10], compressive sensing in [12], genetic algorithm in [15], and neural network (NN) in [9], [11], [16], were widely used due to their robustness, self-adaption, and nonlinear processing ability [19]. However, the performance of these PR methods largely depends on empirical feature extraction due to the limited capacity of classifiers [19]. For PR methods, feature design relies on an empirical judgment. For specific signals, if the empirical feature design is inappropriate, the performance of classification will be greatly degraded.

In recent years, deep learning has performed well in various tasks due to its outstanding deep feature extraction capabilities. In the field of wireless communications, Gui *et al*. proposed a novel and effective deep-learning-aided non-orthogonal multiple-access (NOMA) system, in which several NOMA users with random deployment are served by one base station [20]. Authors in [21] focused on channel estimation and direction-of-arrival (DOA) estimation, and a novel framework that integrates the massive multiple-input multiple-output (MIMO) into deep learning was proposed. To avoid the limitations of empirical feature selection engineering, many researchers also have applied deep learning to AMR. O'Shea *et al*. used CNN to extract features from in-phase and quadrature (I/Q) data and identified modulation schemes in [22]. The result showed that CNN outperformed the traditional machine-learning-based classifiers. Inspired by the excellent performance of CNN in image processing, the authors in [23]–[28] characterized the signal in the form of an image to achieve AMR. CNN-based methods mentioned above only considered the spatial features of the signal, and the temporal features were ignored. In [29], the authors utilized the signal temporal features extracted from the unidirectional RNN. This method only considered the forward temporal features of the signal. However, the temporal features of the signal should be contextually bidirectionally correlated. Authors in [30] and [31] applied Convolutional Long short-term Deep Neural Networks (CLDNN) as the optimal architecture and achieved an accuracy approximately 88.5% at high signal to noise ratio (SNR). Nevertheless, accuracy and complete feature set is still needed and the rate of accuracy still has room for improvement. In addition, the performance of deep learning-based methods rely heavily on a large amount of data, which are difficult to collect due to the cost and time consumption. It is crucial to utilize the collected samples efficiently to improve the recognition accuracy.

In this paper, we propose a novel attention cooperation framework from the perspective of feature completeness and sample sufficiency to effectively realize AMR. For feature completeness, the spatial and temporal features of signals are extracted by CNN and RNN, respectively. Then, these features are fused by the global average and max pooling (GAMP) strategy to achieve final classification. For sample sufficiency, a generative adversarial network (GAN) is designed for data augmentation to provide adequate sample support.

The main contributions of this paper are summarized as follows:

(1) A novel framework that combines CNN, RNN, and GAN is proposed to realize AMR cooperatively. The attention mechanism is employed in this framework to improve the efficiency of the features. Based on the data set RML2016.10a, it is shown that the proposed framework is superior to the existing deep learning-based methods.

(2) Considering the spatial and temporal features of the signal to achieve feature completeness. For the spatial features, a cyclic CNN (CCNN) is designed to achieve the fusion of different levels of abstract features in different update stages. For the temporal features, a one-layer bidirectional RNN (BRNN) is designed to perform full mining of the signal context temporal information. The performance of AMR is promoted by adequate extraction and efficient reuse of the signal spatial-temporal features.

(3) We propose the GAMP strategy to capture the intrinsic correlation between temporal and spatial features and achieve feature fusion, and proved by experiments that this mechanism is better than the simple concatenation on recognition accuracy. In order to expand training data, an ACGAN is introduced to this framework, and we modify the loss function to accommodate the processing of the actual I/Q signal data.

(4) A new auxiliary weighing loss function is proposed to measure the influence of the generated data on the classification model. The auxiliary classification accuracy of ACGAN is exploited to automatically score the weight of the generated data so that the recognition performance is optimized by indirectly changing the distribution of the training data.

The remainder of this paper is organized as follows. In Section II, details of the proposed attention cooperative framework are described. In Section III, the experimental setting is introduced. The results are shown in Section IV. Section V concludes this paper.

II. ATTENTION COOPERATIVE FRAMEWORK

In this section, we introduced the operating mechanism of the attention cooperative framework, which is illustrated in Figure 1. The data flow in the direction of the arrows, i_t and q_t represent the in-phase component and quadrature component of the tth sampled point, respectively. The original dataset only contains the actual collected data. GAN is first

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

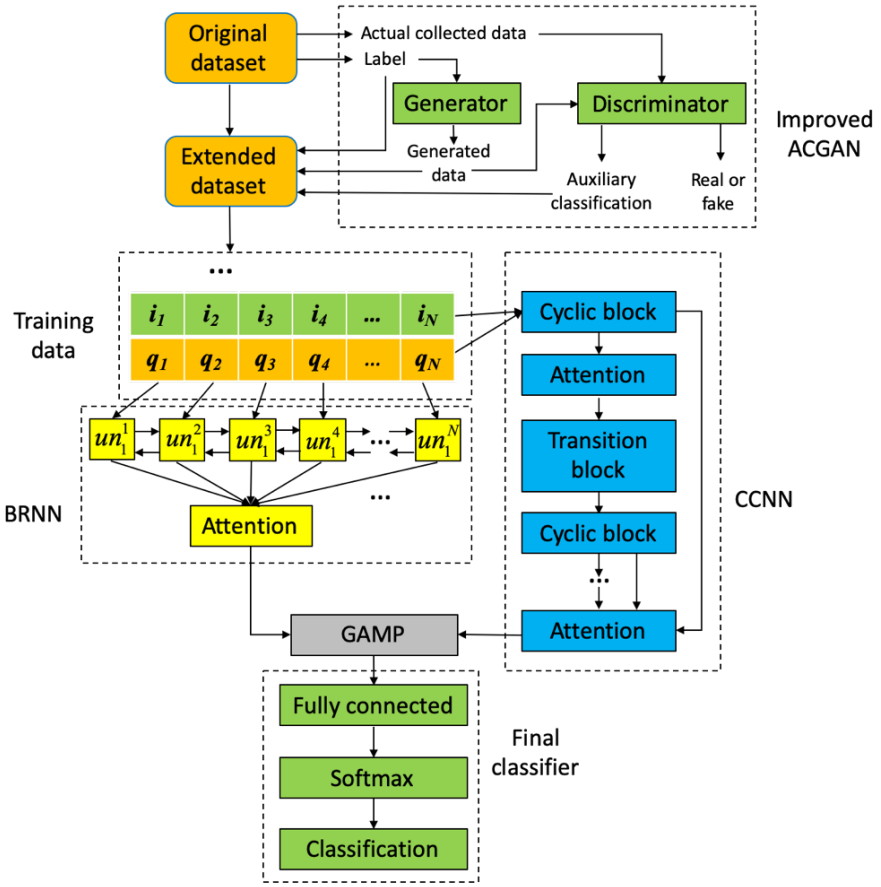

FIGURE 1. Architecture of attention cooperative framework.

trained to complete data augmentation. The hybrid data in the extended dataset, which consists of the actual collected data and generated data, is delivered to the BRNN containing one layer on time dimension to capture the global temporal features. The spatial features are extracted by the CCNN, which contains three cyclic blocks. The attention mechanism is employed with diverse forms to facilitate the effectiveness of the features. Then, GAMP incorporates the feature maps to realize the fusion. The reconstructed feature maps are transmitted to the fully connected layer to make the final classification. The detailed descriptions of the framework are introduced as follows.

A. I/Q DATA AUGMENTATION BASED ON THE ACGAN

For modulation recognition, the insufficiency of signal data has a negative influence on the signal feature analysis. Notably, the deep learning method requires a large amount of training data as support. Adequate training data is beneficial to enhance the generalization performance of the classification model to further improve the classification accuracy. Therefore, a GAN is designed to complement the training data. For actual collected data, the process of data augmentation can be regarded as the expansion of data scale by the samples. When the generator can approach the real sample distribution infinitely, the generated data has a significant probability of containing the useful features required for the classification so that the samples provided by GAN can be used to expand the scale of the training set. GAN is composed of a generator and a discriminator. The generator maps the vectors in the randomly distributed noise space to the target space to establish a distribution model. The goal of the discriminator is to distinguish between the real sample obeying the actual spatial distribution and the fake sample produced by the generator. The two networks are iteratively optimized with minimax countermeasures.

Fundamentally, the purpose of the data augmentation is to assist in classification tasks to improve accuracy. For generated data, it is necessary to add label restrictions to highlight their specific attributes. Therefore, we exploit an ACGAN to achieve data augmentation. The ACGAN intro-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

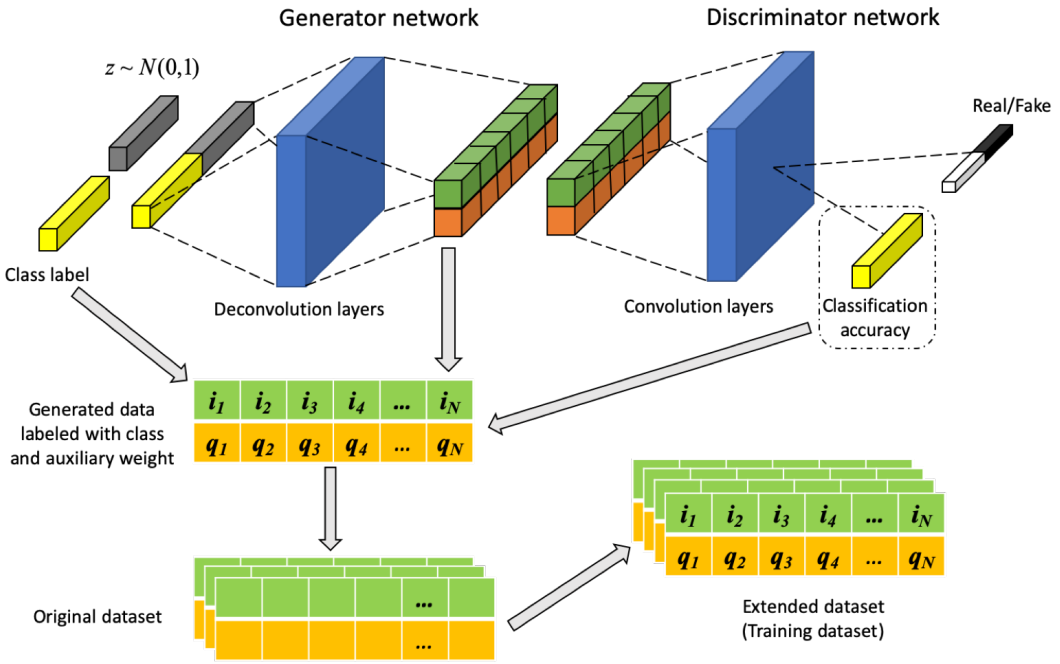

FIGURE 2. Signal data augmentation with the ACGAN.

duces conditional information into the input of the generator. Besides judging the real or fake, an auxiliary classifier is added to the discriminator to give the category estimation. The ability to generate specific data based on labels makes ACGAN suitable for data augmentation of the modulation classification problem.

The process of data augmentation is illustrated in Figure 2. First, the label and the data initialized with the standard Gaussian distribution are submitted to the generation network, and the generated signal data are obtained through a series of deconvolution operations. The generated data are delivered to the discriminating network, and the discriminating network ultimately gives a prediction of the data source and the modulation type probability. After iterative training, the network reaches the Nash equilibrium point. The discriminator cannot distinguish the source of the data, and the auxiliary classification accuracy tends to be stable. Then, we hybrid the generated data labeled with class and auxiliary weight to the original dataset. Finally, the extended dataset will be utilized to train the classification model in subsequent procedures.

However, the network is prone to gradient disappearance when the actual collected signal data are used to train the ACGAN. On one hand, compared with image data, the distribution of the two-dimensional signal data labeled with modulation is more stochastic. On the other hand, from the perspective of the model, for the real data distribution and the generator distribution, an optimal segmentation surface can separate them in the high-dimensional space. If the neural network corresponding to discriminator can fit the segmentation surface infinitely, there is an optimal discriminator which gives a constant probability (1 or 0) on the support set of the real data distribution and the generated data distribution, causing the gradient of the generator to disappear. In response to this problem, we improve the adaptability of the loss function of the ACGAN.

The objective function of the original ACGAN has two parts: the log-likelihood of the correct source, L_S, and the log-likelihood of the correct class, L_C [32]. The log-likelihood of the correct source can be expressed as

$$L_S = \mathop{\mathrm{E}}_{x\sim P_r}[\log(D_S(x))] + \mathop{\mathrm{E}}_{\tilde{x}\sim P_g}[\log(1 - D_S(\tilde{x}))] \quad (1)$$

where P_r is the real data distribution and $x \sim P_r$. P_g is the model distribution implicitly defined by $\tilde{x} = G(z), z \sim p_z$. The input z to the generator is sampled from noise distribution p_z [33]. $D_S(x)$ represents the source probability of sample x given by discriminator. $\mathrm{E}[\cdot]$ denotes the expectation operator. It is obvious that the objective function is presented in the form of binary cross entropy which easily leads to discriminator sensitivity. Inspired by [33] and [34], we replace the binary cross entropy with Wasserstein entropy with gradient penalty as the loss function of discriminator to predict real or fake. Therefore, the optimization problem can be redefined as

$$\begin{aligned} L'_S = &\mathop{\mathrm{E}}_{\tilde{x}\sim P_g}[D_S(\tilde{x})] - \mathop{\mathrm{E}}_{x\sim P_r}[D_S(x)] + \\ &\lambda \mathop{\mathrm{E}}_{\hat{x}\sim P_{\hat{x}}}\left[\left(\|\nabla_{\hat{x}} D_S(\hat{x})\|_2 - 1\right)^2\right] \end{aligned} \quad (2)$$

where $P_{\hat{x}}$ is implicitly defined as sampling uniformly along straight lines between pairs of points sampled from the real data distribution P_r and the generated data distribution P_g [34], and $\hat{x} \sim P_{\hat{x}}$, $x \sim P_r$, $\tilde{x} \sim P_g$. $\|\nabla[\cdot]\|_2$ denotes

the $l_2 - norm$ of the calculated gradient vector and λ is the penalty factor. Different from the original ACGAN, discriminator is limited in the set of 1-Lipschitz functions, which is implemented by gradient clipping in the model. Wasserstein entropy reduces the discriminator sensitivity to distribution differences by limiting the bounds of the network parameters so that the training stability can be enhanced. On this basis, the gradient penalty with factor λ is added to further improve the convergence of the model by solving the centralized problem of parameter distribution caused by weight clipping. For the modulated signal data, we employ the improved loss function for the ACGAN, which finally solves the problem that the gradient disappears in the process of training.

B. AUXILIARY WEIGHING LOSS FUNCTION

In this proposed framework, an ACGAN is designed to achieve data augmentation. While exploiting the commonality of generated data with real data, inherent differences in support of classification should also be taken into account. In the case that the amount of data is limited since the generated data cannot fully represent the real data for the classification task, we propose an auxiliary weighing loss function to balance the influence of the generated data to improve the performance of the classification model.

When a GAN is trained to converge, the discriminator is considered to be indistinguishable from the original signals and the generated signals. However, as the training data with category labels, in addition to the commonality to be distinguished from noise, features derived by classification should be paid more attention. Therefore, ACGAN is designed to indirectly increase the inter-class difference of the generated data by adding an auxiliary classification function to the discriminator, which can be used as a quantitative indicator.

As is shown in Figure 2, we labeled the generated data with auxiliary weight besides the class label. As the optimization target of the classification problem, the inter-class difference directly reflects the influence of training data. Therefore, a new auxiliary weighing loss function is proposed to balance the effect of the real data and the generated data.

The traditional cross entropy function can be expressed as

$$H(p,q) = -\sum_{c} p(c)\log q(c), \tag{3}$$

where p and q stand for the actual and predicted probability value respectively. c represents the class. Cross-entropy calculates the distance between the two probability distributions which describes the difficulty of expressing the probability distribution p through the probability distribution q. In order to weigh different source of data, the auxiliary classification accuracy of ACGAN is utilized as the quantification of the influence factor. The proposed loss function can be expressed as

$$\begin{aligned} H(p,q) &= -\alpha \sum_{c} p(c)\log q(c), \\ \alpha &= \begin{cases} 1 & p(x) \in p_{\text{real}}(c) \\ m & p(x) \in p_{\text{generated}}(c) \end{cases} \end{aligned} \tag{4}$$

where α is the influence factor and m denotes the auxiliary classification accuracy of the discriminator. This proposed loss function is actually designed to realize the data-level attention mechanism to facilitate AMR.

C. TEMPORAL FEATURE EXTRACTION BASED ON THE BRNN

The modulated signal can be viewed as the time-series data, and thus its global temporal features are indispensable for AMR. Therefore, we design a BRNN to perform time-series analysis on modulated signals to extract effective features for classification. Different from the previous work using the simple classical RNN to extract and classify time series [29], the proposed framework simultaneously performs forward and reverse processing on the signal, which is shown in Figure 3. Each hidden node containing two units outputs a two-dimensional data, which represents the information captured from the previous and subsequent sampling points of the current timestamp. For the observation of the modulation type of the signal, the context information before and after the observation point is valid and worthy to be comprehensively analyzed. A one-layer LSTM-based BRNN model is designed in the proposed framework to capture the overall temporal features of the modulated signal adequately.

The signal vector at timestamp t can be denoted as

$$\mathbf{s_t} = [i_t, q_t], \{t = 1, ..., N\} \tag{5}$$

where the $i_t$and q_t are the in-phaseïijĹI) and quadrature (Q) components. For the first layer, the process is defined as

$$\overrightarrow{h}_t^{(1)} = \sigma(\vec{\mathbf{U}}\mathbf{s_t} + \overrightarrow{V}^{(1)}\overrightarrow{h}_{t-1}^{(1)} + \overrightarrow{b}^{(1)}) \tag{6}$$

$$\overleftarrow{h}_t^{(1)} = \sigma(\overleftarrow{\mathbf{U}}\mathbf{s_t} + \overleftarrow{V}^{(1)}\overleftarrow{h}_{t+1}^{(1)} + \overleftarrow{b}^{(1)}) \tag{7}$$

The arrow indicates the direction of the process. $\overrightarrow{h}_t^{(1)}$and $\overleftarrow{h}_t^{(1)}$ denote the calculation result of forward and backward processe of the tth hidden node of the first layer. $\mathbf{U}$ and V denote the linear parameters of the input signal sample point and the output of the previous node, respectively. b is the bias and σ denotes the non-linear transformation which is performed by an activation function such as ReLU, sigmoid and tanh. For the BRNN in this proposed framework, the tanh function is chosen. The output value range is -1 to 1 and the average value is fixed at 0, which facilitates the management of the subsequent layers. The state value of the hidden node $h_t^{(a)}$ in the ath layer at timestamp t can be expressed as

$$\mathbf{h_t^{(a)}} = [\overrightarrow{h}_t^{(a)}, \overleftarrow{h}_t^{(a)}] \tag{8}$$

where $\overrightarrow{h}_t^{(a)}$and $\overleftarrow{h}_t^{(a)}$ denote the calculation result of forward and backward proceses of the tth hidden node of the ath

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

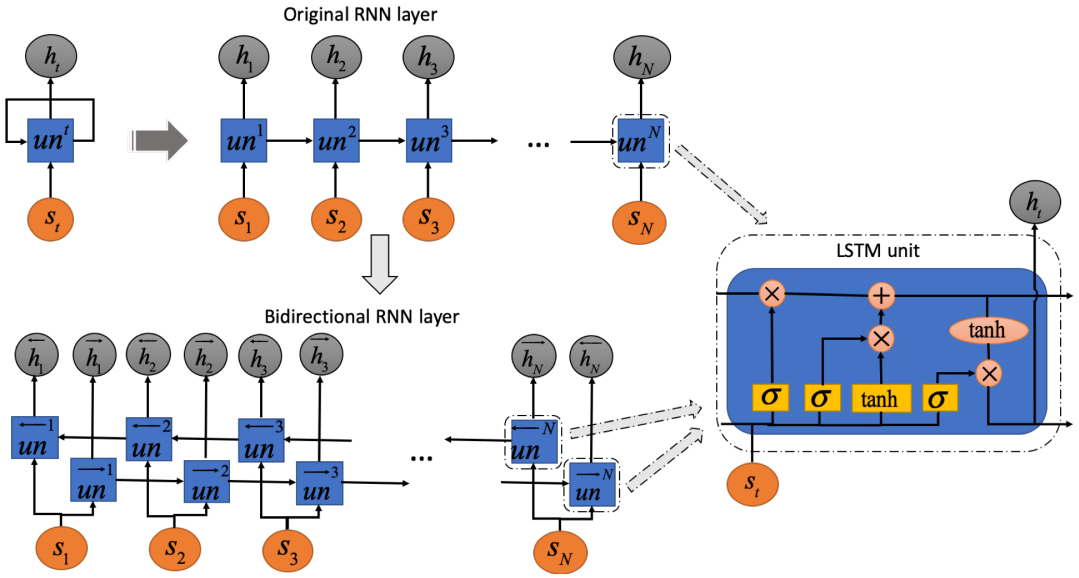

FIGURE 3. The comparison of the original RNN and the BRNN.

layer, respectively. The concatenate method is chosen here instead of simply summing, which ensures a non-linear interaction between the forward and backward information of the sequence modulated signal in subsequent process. The computing process of ath $(a > 1)$ layer can be defined as

$$\overrightarrow{h}_t^{(a)} = \sigma(\overrightarrow{\mathbf{W}}^{(\mathbf{a})} \mathbf{h_t^{(a-1)}} + \overrightarrow{V}^{(a)} \overrightarrow{h}_{t-1}^{(a)} + \overrightarrow{b}^{(a)}) \quad (9)$$

$$\overleftarrow{h}_t^{(a)} = \sigma(\overleftarrow{\mathbf{W}}^{(\mathbf{a})} \mathbf{h_t^{(a-1)}} + \overleftarrow{V}^{(a)} \overleftarrow{h}_{t+1}^{(a)} + \overleftarrow{b}^{(a)}) \quad (10)$$

where $\mathbf{W}$ is the matrices representing the linear relation of the hidden nodes of previous layer at current time. The parameters represented by $\mathbf{W}$, V, b are shared throughout the corresponding layer to realize the recurrent. Processed by a one-layer BRNN, the overall temporal features of the modulated signal data are extracted.

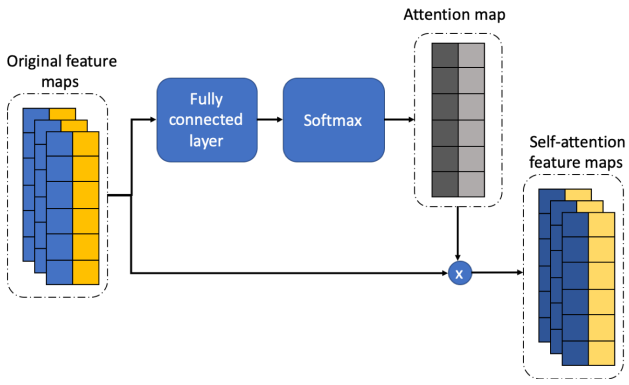

FIGURE 4. Soft self-attention mechanism.

Nevertheless, the influences of the vectors in the output feature map on the modulation type discrimination are actually different. Therefore, we introduce the self-attention mechanism to recalibrate the features to extract valid information. The soft attention is chosen due to the global receptive field and continuous differentiability, which benefits the gradient computation. The process of self-attention is illustrated in Figure 4. The attention map is obtained by a linear transformation of the processed feature sequence, and nonlinear transformation and normalization are performed by the softmax function. The calculated attention map is multiplied by the feature sequence to achieve temporal self-attention.

D. SPECIAL LOGICAL FEATURE EXTRACTION BASED ON THE CCNN

Similar to the image data, a logic relationship exists in spatial points of the two-dimensional matrix formed by the I/Q signal. In order to obtain the information contained in the local spatial features which are useful for identifying the modulation types, we design a CNN based on the cyclic structure to make full use of the abstract features of different layers. The basic structure of the cyclic block is shown in Figure 5.

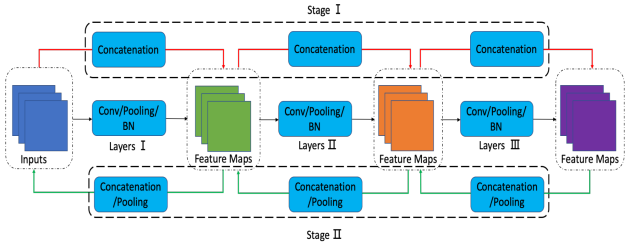

FIGURE 5. Basic structure of the the cyclic block.

In the first stage, the transmission process of the feature

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

maps is similar to the residual structure. The output features calculated by convolution, pooling, and batch normalization are concatenated with the input features to improve the efficiency of information dissemination between different layers and to enhance the feature reuse. The process of the first stage can be expressed as

$$o_l = g[H_l(o_{l-1}), o_{l-1}] \tag{11}$$

where o_l denotes the output of the lth $(0 < l \leq 3)$ layer in the first stage. Especially, if $l = 1$, o_0 denotes the inputs of the first layer. H_l and g represent the calculation of the lth layer and the nonlinear operation, respectively. Considering the features of higher-level output are more abstract and refined, we design a cyclic structure for the network, which introduces the second stage of feature propagation. This process is to propagate the features of the high-level output forward. In the second stage, the original map geometry is preserved while merging the feature maps by concatenation and pooling operations which can be expressed as

$$o_l^{'} = g^{'}[H_l(o_{l-1}^{'}), o_{l+1}] \tag{12}$$

where $o_l^{'}$ refers to the output of the lth $(0 < l \leq 2)$ layer and $g^{'}$ represents the nonlinear operation in the second stage. Especially, if $l = 0$, $o_0^{'}$ denotes the inputs of the first layer in the second stage which can be denotes as

$$o_0^{'} = g^{'}[o_1] \tag{13}$$

The cyclic feedback structure in the second stage refines the convolution kernel of the previous layer with higher-level abstract information, which influences the spatial attention. The result calculated by the last layer of the second stage will be delivered to the attention block for further processing. We design this cyclic structure so that the features of the hierarchy can be fully interacted to extract information which contributes to classification effectively.

As for the attention block, because the influence ranks are different between the channels inside each feature map, we exploit squeeze and excitation (SE) attention mechanism, which is shown in Figure 6 to achieve recalibration. First, performing global average pooling on the feature maps which can be expressed as

$$z_c = \frac{1}{K \times J} \sum_{k=1}^{K} \sum_{j=1}^{J} u_c(k, j) \tag{14}$$

where z_c denotes the initialization weight value of cth channel u_c, and K and J represent the width and height of the feature map, respectively. Then, the feature weight vector is generated after operating scaling and nonlinear transformation defined as the activation function sigmoid in the channel dimension, which controls the weight parameters value between 0 and 1. In brief, the global average pooling performs initial extraction of the feature weight parameters, and the scaling operation models the correlation between channels. The corresponding channel feature is multiplied by the weight vector to complete SE attention mechanism.

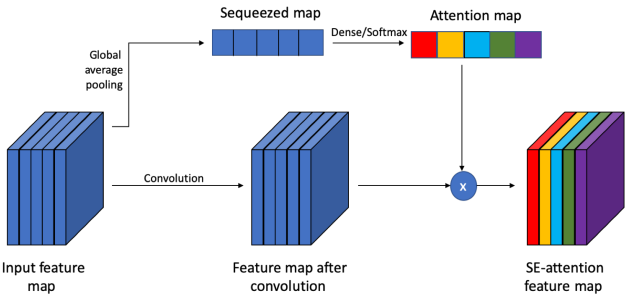

FIGURE 6. The procedure of SE mechanism in attention block.

Then, the recalibrated features will be transferred to the transition block, which is illustrated in Figure 7. We employ a 1×1 convolution to construct the bottleneck structure for channel dimensionality reduction, combined with the pooling operation to complete the feature map compression. The transition module is used for parameter reduction to improve the computational efficiency of the network.

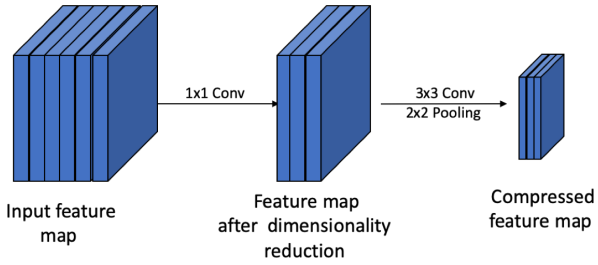

FIGURE 7. The basic structure of the bottleneck in transition block.

Considering the computational complexity and network performance, we build three cyclic modules. The overall CCNN architecture is shown in Figure 8. The output of each cyclic block is pooled to simplify the parameters. After the concatenation is completed, the integrated features are delivered to the attention module to obtain an output feature map of the network. Therefore, each block can directly access the gradient information to accelerate network convergence. In the pre-training phase, the feature map will continue through a fully connected layer, and the classification probability will be calculated by softmax. Actually, for a trained CCNN in the framework, the feature map is the final output of the spatial features.

E. SPATIAL-TEMPORAL FEATURE FUSION BASED ON THE GAMP STRATEGY

To utilize the correlation between the temporal and the spatial features, we propose the GAMP strategy for spatial-temporal feature fusion, whose mechanism is illustrated in Figure 9. First, the output of the BRNN is converted to structurally consistent with the output feature map of the CCNN through a fully connected layer. Then, based on simple concatenation, we first perform a global average pooling (GAP) and a max-pooling (MP) operation in channel dimension and get the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

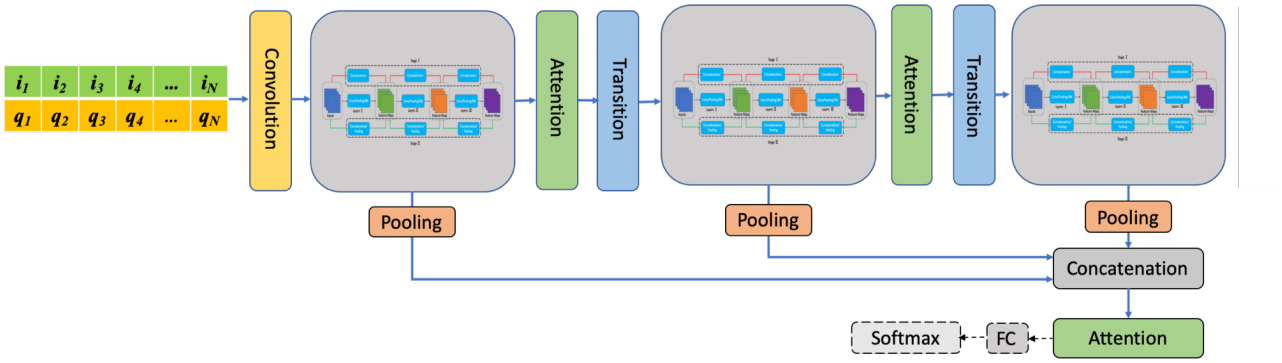

FIGURE 8. The architecture of the CCNN.

integration feature map. Next, inspired by the SE structure, the max-pooling and global average pooling are operated along the channel dimension and acquire their weight maps. Then, the attention map is calculated by averaging the sum of the weight maps. The reconstructed feature map is obtained by multiplying the attention map with the integration feature map, which can provide adequate information for AMR. As is shown in Figure 1, the output of GAMP is delivered to the final classifier to get the recogniiton results.

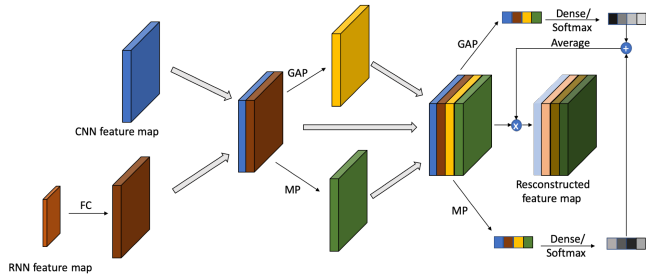

FIGURE 9. The fusion mechanism of the GAMP strategy.

III. EXPERIMENTAL SETTING

In this section, we present the dataset description, the hyperparameter configuration, and the parameter learning. The tricks of improving performance are also discussed.

A. DATASET DESCRIPTION

RadioML2016.10a dataset [22] is used as the basis for model performance verification. This dataset contains eleven different digital and analog modulation formats, including BPSK, QPSK, 8PSK, QAM16, QAM64, CPFSK, GFSK, 4PAM, WBFM, AM-DSB and AM-SSB, corresponding 220K sequences for 128 complex-valued baseband I/Q samples, which are collected at a sampling rate 1 M/s and 4 samples per symbol from the signals that pass through a wireless channel with the effects of multipath fading, sample rate offset, and center frequency offset [17]. It is widely used in evaluating AMR performance such as in [22], [27], [29]–[31]. The samples are taken with 2 dB interval within the range from -20 dB to 18 dB [22], and are processed as a matrix with the size of 2×128, where the in-phase and quadrature parts of the signal samples are separated.

B. HYPERPARAMETER CONFIGURATION AND PARAMETER LEARNING

For the optimizer configuration, the optimizer for the CCNN is based on Nesterov momentum method. The momentum parameter is set as 0.9, which corresponds to a maximum gradient update speed of ten times that of the gradient descent algorithm. The initial learning rate is set as 0.1. The BRNN, the ACGAN, and the final classifier of the whole framework employ Adam optimizer. For the BRNN, the learning rate is set as 0.001. The first-order and second-order moment parameters of the BRNN are set as 0.9 and 0.999, respectively. For the ACGAN, the learning rate of the generator and discriminator is set as 0.0001, the first-order moment parameter is set as 0.5, and the second-order moment parameter is set as 0.9. For the final classifier, the parameter setting is the same as BRNN.

For the parameter learning algorithms, the CCNN, the ACGAN, and the final classifier are trained through backpropagation (BP) algorithms. The BRNN is trained with backpropagation through time (BPTT) algorithms.

As for iterations, the ACGAN is trained first and converges after 20,000 training iterations. The extended training set is delivered to the classification module containing the CCNN and the BRNN. The CCNN and the BRNN are first trained separately and converge after 200 and 50 training iterations, respectively. According to the reconstructed feature map, the final classifier after the GAMP fully converges, i.e., the classification accuracy of testing data is no longer improved, after 20 training iterations.

The proposed framework is built and trained with Tensorflow deep learning library on Ubuntu 16.04 with an Nvidia GeForce GTX 1080 Ti GPU.

C. TRICKS OF IMPROVING PERFORMANCE

In the process of building the framework, we investigate some tricks which can effectively facilitate the training process or improve the classification accuracy.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

1) Zero padding

In consideration of the characteristics of CNN, zero padding has the effect of maintaining boundary information. For I/Q modulated signal, zero padding is performed in two dimensions of the row and column in the CCNN. The operation actually takes into account the retention of the head and the tail specialities, and correlations of in-phase and quadrature components.

2) Batch normalization

Batch normalization (BN) normalizes the first and second moments of the data so that the data still has zero mean and unit variance after it is processed through the network layers. In order to fully utilize the nonlinear expression ability of activation functions, the result of BN is multiplied by a scaling factor and added with bias before being passed to the nonlinear units. The scaling factor and bias are learned by the network. BN is employed in the CCNN, the BRNN, and the final classifier in the proposed framework to accelerate the convergence speed of the network and improve the classification accuracy.

3) Dropout

While the fitting ability is improved, the dropout is adopted to take into account the generalization ability. According to the different characteristics of the network structures, the dropout rate is set as 0.8, 0.3, and 0.5 in the CCNN, the BRNN, and the ACGAN of the proposed framework, respectively.

4) Weight decay

To avoid over-fitting of the training data, a regularization term, which can be expressed as the $l_2 - norm$ of the network weight vector, is added to the loss function. Intuitively, weight decay makes the model prefer to learn the weight with a smaller $l_2 - norm$, while the decay factor quantifies the degree of preference. In the proposed framework, the decay factor is set as 0.0001 to balance the fitting and generalization ability.

5) MSRA initialization

In the proposed framework, the ReLU activation function is introduced into the CCNN for nonlinear transformation. Corresponding to the characteristics of the ReLU activation function, in order to avoid the gradient disappearing, MSRA initialization is used for initializing weight values of the CCNN. The weight distribution after initialization is a Gaussian distribution. The MSRA initialization method effectively improves the classification accuracy in the proposed framework.

6) Leaky ReLU

Unlike the classical ReLU function, leaky ReLU assigns a non-zero slope to a negative region. We use leaky ReLU as an activation function in the ACGAN to avoid the instability of the training process caused by gradient disappearance and accelerate convergence.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed framework from different perspectives.

A. SIMULATION RESULTS OF BRNN

In our framework, RNN is introduced to extract the temporal characteristics of signals, and the bidirectional structure is designed to obtain the context information fully. In this section, BRNN is compared with the unidirectional structure. The effect of the number of BRNN layers on classification accuracy is also studied.

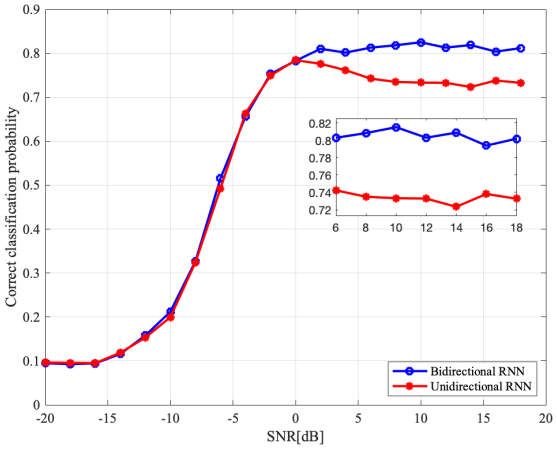

FIGURE 10. The comparison of the BRNN and the unidirectional RNN.

As is shown in Figure 10, the recognition performance of BRNN and unidirectional RNN below 0 dB is approximately consistent. When SNR is above 0 dB, the average recognition accuracy of BRNN is 0.6 higher than that of unidirectional RNN. The reason is that the bidirectional transmission structure enables the contextual temporal information of the signal to be fully extracted to obtain complete temporal features.

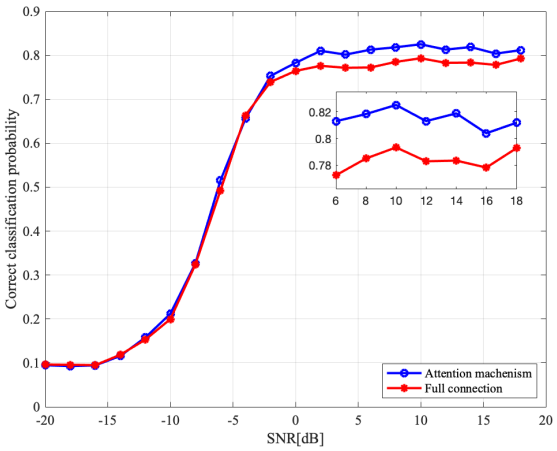

FIGURE 11. The comparison of the self-attention and the full connection.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

Moreover, the self-attention mechanism is introduced in BRNN as mentioned in Section II.B. Figure 11 illustrates the performance comparison between the attention mechanism and full connection. When SNR is above 0 dB, the recognition accuracy of the attention mechanism is about 1.5% higher than that of the full connection, because the self-attention mechanism captures the internal correlation of features and improves the precision of temporal feature representation.

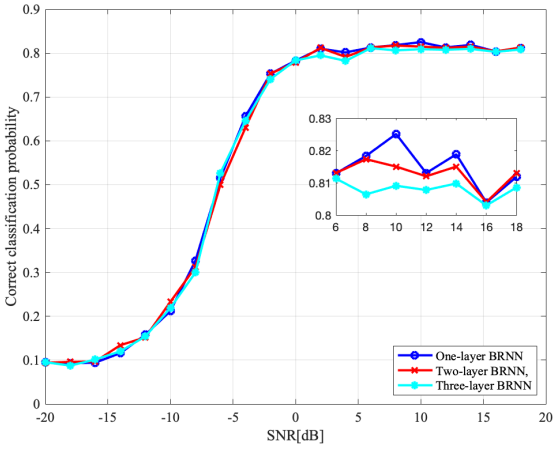

FIGURE 12. The comparison of the BRNN with different layers.

In Figure 12, we analyze the influence of the BRNN layer number on the recognition accuracy. As the number of BRNN layer increases from one to three, the recognition accuracy is not improved. For three-layer BRNN, the recognition accuracy even decreases, which is propably due to the over-fitting. Increasing the layer number also brings additional computational overhead. Therefore, in the proposed framework, a one-layer BRNN is constructed to extract temporal features.

B. SIMULATION RESULTS OF CCNN

CNN is introduced to extract the spatial features of the signal in this framnework. The cyclic block structure is designed to increase the information flows in the CCNN. In order to prove the superiority of the cyclic structure, it is compared with the residual structure and the densely connected structure.

As is shown in Figure 13, when SNR is above 0 dB, the cyclic structure outperforms the other two structures, which are only built with forward connections. The reason is that the feature reusability is improved since the cyclic structure establishes forward and backward connections between every two layers. The two-stage data propagation implements the loop feedback operation, which achieves the feature refining and spatial attention mechanism.

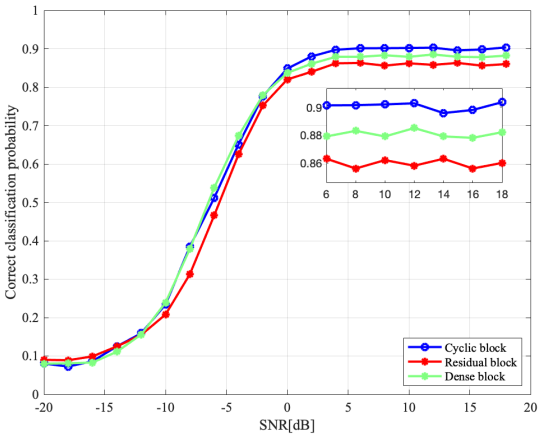

FIGURE 13. The comparison of the cyclic block and the structures only built with forward connections.

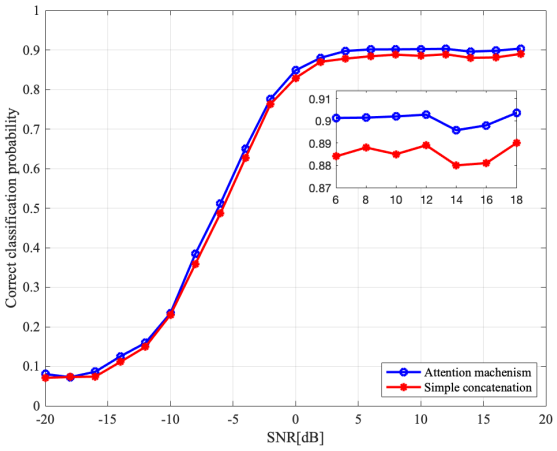

FIGURE 14. The comparison of the SE attention and simple concatenation.

We introduce the SE attention mechanism in CCNN to achieve feature recalibration. Figure 14 illustrates the performance comparison between SE-attention mechanism and the simple concatenation. Under the condition of high SNR, the recognition accuracy of the attention mechanism is about 1% higher than that of simple concatenation. The reason is that the feature recalibration completed by attention mechanism strengthens the discrimination of the output feature maps.

C. SIMULATION RESULTS OF GAMP

GAMP is proposed to fully fuse and recalibrate the spatial and temporal features of the signal. As a feature fusion strategy, we compare GAMP with simple concatenation and apply the final recognition accuracy as the performance indicator.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

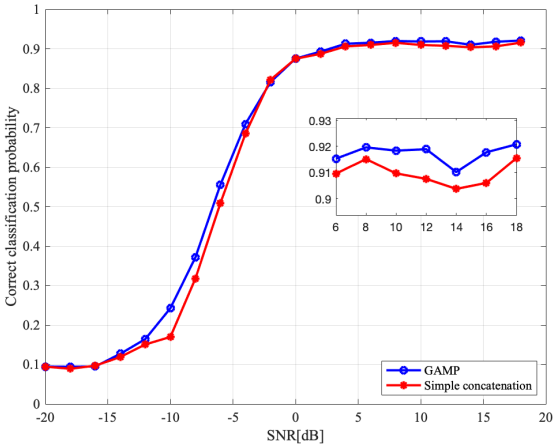

FIGURE 15. The comparison of the GAMP strategy with simple concatenation.

As is shown in Figure 15, under the condition that the SNR is -14 dB to -4 dB, the recognition accuracy of GAMP is higher than that of simple concatenation. GAMP also provides a 0.007 improvement when the SNR is above 6dB. The reason is that the pooling operation which is combined with the attention mechanism allows GAMP to facilitate the interaction of spatial and temporal features to extract sufficient information for AMR.

D. SIMULATION RESULTS OF IMPROVED ACGAN AND AUXILIARY WEIGHING LOSS FUNCTION

In the overall attention cooperative framework, temporal features are extracted by the BRNN and spatial features are extracted by the CCNN. After the temporal and spatial features are fused with the GAMP strategy, the recognition results are obtained by the final classifier. On this basis, we introduce ACGAN to expand the training data and proposed the auxiliary weight loss function to balance the influence of generated data and actual collected data to futher enhance the recognition performance. Different proportions of training data are studied to explore the effects of data augmentation. The influence of auxiliary weighing loss function on the final classification accuracy is also studied.

TABLE 1. Recognition accuracy improvement of the ACGAN at 6 dB.

Original data proportion	Accuracy	With generated data proportion	Accuracy	With auxiliary weighing loss	Accuracy
50%	74.53%	+50%	82.16%	✓	83.08%
60%	78.25%	+40%	85.48%	✓	86.35%
70%	83.44%	+30%	87.09%	✓	87.92%
80%	86.67%	+20%	88.37%	✓	89.18%
100%	91.53%	+0%	91.53%	-	-
100%	91.53%	+10%	91.85%	✓	92.24%
100%	91.53%	+30%	92.38%	✓	92.85%
100%	91.53%	+50%	**92.89%**	✓	**93.66%**
100%	91.53%	+70%	92.67%	✓	93.41%
100%	91.53%	+90%	92.14%	✓	92.95%

As is illustrated in Table 1, when the SNR is 6 dB, the expansion of generated data to the training set improves the recognition accuracy. However, due to the existence of an objective difference between the generated data and the actual collected data. When the generated data volume exceeds 50% of the actual collected data volume, the recognition accuracy is no longer improved. On this basis, an auxiliary weighing loss function is proposed to balance the influence of the two kinds of data, and the recognition accuracy is improved to 93.66%. The reason is that the increase in the amount of training data can facilitate the generalization of the classification model in a specific range to improve classification accuracy, and the auxiliary weighing loss function modifies the data influences to further enhance the performance of AMR.

E. PERFORMANCE COMPARATION WITH EXISTING WORKS

The proposed framework is compared with the existing works, which represent AMR techniques based on deep learning.

1) CNN based on VGG architecture. VGG architecture, contains series of narrowing convolutional layers followed by fully-connected layers and is terminated with a dense softmax layer for classification, which is leveraged in [22].
2) GoogleNet based on inception structure. The inception structure is used to extract multi-scale information using convolution kernels of different scales in the same layer, and the exported feature maps are concatenated in channel dimension. Essentially, features are extracted and retained in different receptive fields. The 1×1 convolution kernel is used to reduce the number of parameters which is studied in [30].
3) ResNet based on shortcut structure. The residual module can be implemented by attaching a shortcut connection to the forward neural network. The shortcut connection is equivalent to simply performing the equivalent mapping without generating additional parameters. The residual structure solves the problem of performance degradation due to the deepening of the network layers which is studied in [31]
4) Classical RNN based on gated recurrent unit (GRU) structure. RNN with GRU as the basic structure is widely used in the field of natural language processing (NLP). The time-series properties of modulated signals make the application of RNN in AMR reasonable. A two-layer classic RNN based on GRU is used for AMR in [29].
5) CLDNN (Convolutional, Long Short-term memory, fully connected Deep Neural Networks). CLDNN, which is originally used by Google for natural machine translation models, demonstrates superior performance in the field of speech recognition. The basic structure of CLDNN is the cascade of CNN and LSTM units. Some previous work like [30] and [**?**] tried to use CLDNN for AMR.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2966777, IEEE Access

6) CNN-CSCD (CNN based on cyclic spectra (CS) and constellation diagram (CD)). A two-branch CNN model is developed in [27]. The features learned from CS and CD is fused to achieve AMR.

At the same experimental condition, the recognition accuracy results of the proposed framework and other comparative techniques with well-tuned parameters are shown in Figure 16.

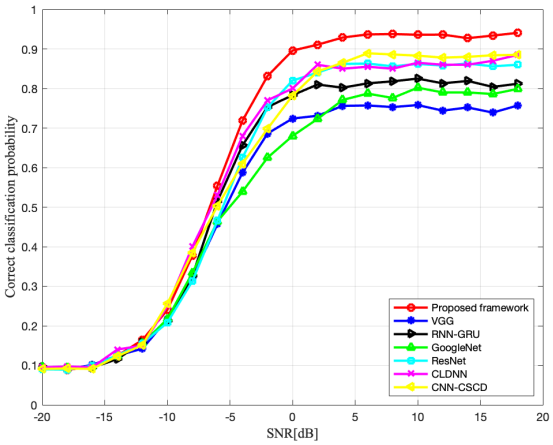

FIGURE 16. Recognition performance comparison versus SNR.

The recognition accuracy of the proposed framework is similar to other structures at low SNR stages. When the SNR is above -8 dB, the recognition accuracy curve has a significant upward trend. When the SNR is 0 dB, the recognition accuracy of the proposed framework is up to 90.1%, the GoogleNet is 68.1%, the VGG is 72.4%, the CNN-CSCD is 78.2%, the RNN-GRU is 78.3%, the CLDNN is 80.2%, and the ResNet is 82.4%. Moreover, the accuracy of the proposed framework at 18 dB is 94.1%, which is 75.7%, 79.9%, 81.2%, 86.3%, 88,4%, and 88.5% of the VGG, the GoogleNet, the RNN-GRU, the ResNet, the CNN-CSCD and the CLDNN, respectively. When the SNR is above 0 dB, the accuracy of attention cooperative framework outperforms the CLDNN by about 5.5%. The reason is that the spatical and the temporal features of the signal are extracted by the CCNN and the BRNN, and are fully fusioned by the GAMP strategy in the proposed framework to guarantee feature completeness. The training set is extended by the ACGAN, and is balanced with auxiliary weighing loss function to realize sample sufficiency. In addition, attention mechanism is employed in the proposed framework to enhance the effectiveness of the features so that the accuracy of AMR is further improved.

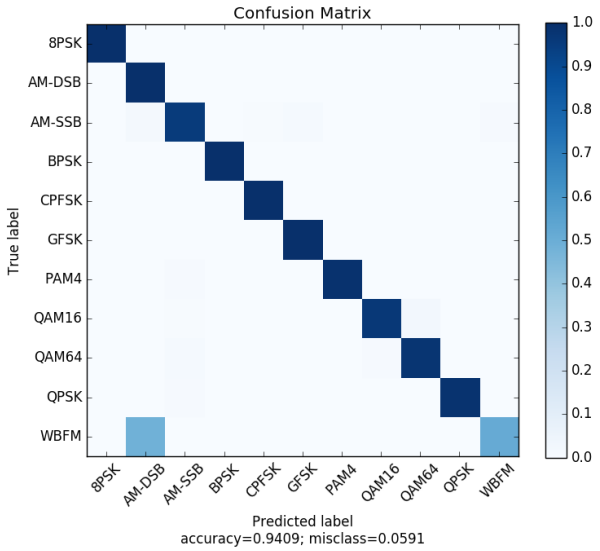

FIGURE 17. Confusion matrix of the proposed framework (SNR = 18 dB).

Figure 17 shows the confusion matrix of the proposed framework when the SNR is 18 dB. The distinction between AM-DSB and WBFM is difficult by the small observation window and low information rate with frequent silence between words of the data in RML2016.10a [22]. In [22], it is noted that QAM16 and QAM64 are confused because they share common points in constellations, which suffers from short-time observations. However, as is shown in Figure 17, the confusion severity of QAM16 and QAM64 are prominently reduced. The reason is that the proposed framework extracts the spatial and the temporal features together so that the periodic inner trends corresponding to the modulation types are captured more accurately.

V. CONCLUSIONS

A novel attention cooperative framework was proposed to improve the modulation recognition accuracy. An improved ACGAN was designed to achieve data augmentation, and an auxiliary weighing loss function was proposed to balance the influences of the training data. The CCNN and the BRNN with attention mechanisms were constructed to extract the spatial and the temporal features of the signal. The GAMP strategy was proposed to export the spatial-temporal correlation feature map, which provided more effective information for AMR. We utilized dataset RML2016.10a to demonstrate the performance of the proposed framework. The BRNN with the self-attention mechanism was compared with the original RNN and showed the advantages in the temporal feature extraction. The layer numbers of the BRNN was also studied to present a trade-off between the accuracy and the network complexity. The CCNN consturcted with the cyclic structure and the SE attention mechanism was demonstrated to be superior to the residual structure and the densely connected structure in the spatial feature extraction. For the spatial-temporal feature fusion, the GAMP strategy was compared with the simple concatenation and showed considerable supe-

riority. Moreover, we introduced the ACGAN and modified its loss function to accommodate the I/Q data augmentation. Different proportions of training data were studied and an appropriate data proportion was obtained. The auxiliary weighing loss function brought a further improvement of the recognition accuracy. In addition, we compared the attention cooperative framework with several existing works based on deep learning. The recognition accuracy showd that the developed framework outperformed existing deep learning-based techniques and showed significant potential for AMR.

REFERENCES

[1] B. Li, S. Li, J. Hou, J. Fu, C. Zhao and A. Nallanathan, "A Bayesian approach for adaptively modulated signals recognition in next-generation communications," *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4359–4372, Aug. 2015.

[2] O. A. Dobre, A. Abdi, Y. Bar-Ness and W. Su, "Survey of automatic modulation classification techniques: classical approaches and new trends," *IET Commun.*, vol. 1, no. 2, pp. 137–156, Apr. 2007.

[3] M. Abdelbar, W. H. Tranter and T. Bose, "Cooperative cumulants-based modulation classification in distributed networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 3, pp. 446–461, Sept. 2018.

[4] D. Zhu, V. J. Mathews and D. H. Detienne, "A likelihood-based algorithm for blind identification of QAM and PSK signals," *IEEE Trans. Wireless Commun.*, vol. 17, no. 5, pp. 3417–3430, May. 2018.

[5] W. Chen, Z. Xie, L. Ma, J. Liu and X. Liang, "A faster Maximum-Likelihood modulation classification in flat fading non-Gaussian channels," *IEEE Commun. Lett.*, vol. 23, no. 3, pp. 454–457, Mar. 2019.

[6] S. Hassanpour, A. M. Pezeshk and F. Behnia, "Automatic digital modulation recognition based on novel features and support vector machine," in *Proc. 12th Int. Conf. Sig. Imag. Technol. Int. Syst. (SITIS)*, Naples, 2016, pp. 172–177.

[7] E. Moser, M. K. Moran, E. Hillen, D. Li and Z. Wu, "Automatic modulation classification via instantaneous features," in *Proc. Natl. Aerosp. Electro. Conf. (NAECON)*, Dayton, OH, 2015, pp. 218–223.

[8] W. Su, "Feature space analysis of modulation classification using very high-order statistics," *IEEE Commun. Lett.*, vol. 17, no. 9, pp. 1688–1691, Sept. 2013.

[9] Z. Zhang, Z. Hua and Y. Liu, "Modulation classification in multipath fading channels using sixth-order cumulants and stacked convolutional auto-encoders," *IET Commun.*, vol. 11, no. 6, pp. 910–915, Apr. 2017.

[10] S. Jin, Y. Lin and H. Wang, "Automatic modulation recognition of digital signals based on Fisherface," in *Proc. IEEE Int. Conf. Softw. Quali. Reliab. Secur. Compan.*, Prague, 2017, pp. 216–220.

[11] J. Ma and T. Qiu, "Automatic modulation classification using cyclic correntropy spectrum in impulsive noise," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 440–443, Apr. 2019.

[12] Y. Chen, J. Liu and S.T. Lv, "Modulation classification based on bispectrum and sparse representation in cognitive radio," in *Proc. IEEE 13th Int. Conf. Commun. Technol.*, Jinan, 2011, pp. 250–253.

[13] Z. Wu, S. Zhou, Z. Yin, B. Ma and Z. Yang, "Robust automatic modulation classification under varying noise conditions," *IEEE Access*, vol. 5, pp. 19733–19741, 2017.

[14] Y. Lv, Y. Liu, F. Liu, J. Gao, K. Liu and G. Xie, "Automatic modulation recognition of digital signals using CWT based on optimal scales," in *Proc. IEEE Int. Conf. Comput. Inform. Technol.*, Xi'an, 2014, pp. 430–434.

[15] N. Ahmadi and R. Berangi, "Modulation classification of QAM and PSK from their constellation using Genetic Algorithm and hierarchical clustering," in *Proc. Int. Conf. Inform. Commun. Technol. Theory Appl.*, Damascus, 2008, pp. 1–5.

[16] F. Wang, Y. Wang and X. Chen, "Graphic constellations and DBN based automatic modulation classification," in *Proc. IEEE 85th Veh. Technol. Conf. (VTC Spring)*, Sydney, NSW, 2017, pp. 1–5.

[17] F. Yang, L. Yang, D. Wang, P. Qi and H. Wang, "Method of modulation recognition based on combination algorithm of K-means clustering and grading training SVM," *China Commun.*, vol. 15, no. 12, pp. 55–63, Dec. 2018.

[18] X. Zhang, T. Ge and z. Chen, "Automatic modulation recognition of communication signals based on instantaneous statistical characteristics and SVM classifier," in *Proc. IEEE Asia-Pacific Conf. Anten. Propag. (APCAP)*, Auckland, 2018, pp. 344–346.

[19] T. Wang, C. Wen, H. Wang, F. Gao, T. Jiang and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017.

[20] Guan Gui, Hongji Huang, Yiwei Song, and Hikmet Sari, "An Effective NOMA Scheme based on Deep Learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440-8450, Sept. 2018.

[21] H. Huang, J. Yang, H. Huang, Y. Song and G. Gui, "Deep Learning for Super-Resolution Channel Estimation and DOA Estimation Based Massive MIMO System," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549-8560, Sept. 2018.

[22] T. O'Shea and J. Hoydis, "An introduction to deep Learning for the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.

[23] R. Li, L. Li, S. Yang and S. Li, "Robust automated VHF modulation recognition based on deep convolutional neural networks," *IEEE Commun. Lett.*, vol. 22, no. 5, pp. 946–949, May 2018.

[24] D. Wang et al., "Modulation format recognition and OSNR estimation using CNN-based deep learning," *IEEE Photon. Technol. Lett.*, vol. 29, no. 19, pp. 1667–1670, Oct. 2017.

[25] B. Tang, Y. Tu, Z. Zhang and Y. Lin, "Digital signal modulation classification with data augmentation using generative adversarial nets in cognitive radio networks," *IEEE Access*, vol. 6, pp. 15713–15722, 2018.

[26] Y. Lin, Y. Tu, Z. Dou and Z. Wu, "The application of deep learning in communication signal modulation recognition," in *Proc. IEEE/CIC Int. Conf. Commun. China,* Qingdao, 2017, pp. 1–5.

[27] H. Wu, Y. Li, L. Zhou and J. Meng, "Convolutional neural network and multi-feature fusion for automatic modulation classification," *Electron. Lett.*, vol. 55, no. 16, pp. 895–897, Aug. 2019.

[28] W. Xu, Y. Wang, F. Wang and X. Chen, "PSK/QAM modulation recognition by convolutional neural network," in *Proc. IEEE/CIC Int. Conf. Commun. China,* Qingdao, 2017, pp. 1–5.

[29] D. Hong, Z. Zhang and X. Xu, "Automatic modulation classification using recurrent neural networks," in *Proc. IEEE 3rd Int. Conf. Comput. Commun. (ICCC)*, Chengdu, 2017, pp. 695–700.

[30] N. E. West and T. O'Shea, "Deep architectures for modulation recognition," in *Proc. IEEE Int. Symp. Dyna. Spect. Acce. Netw.*, Piscataway, NJ, 2017, pp. 1–6.

[31] X. Liu, D. Yang and A. E. Gamal, "Deep neural network architectures for modulation classification," in *Proc. 51st Asilo. Conf. Sig. Syst. Comput.*, Pacific Grove, CA, 2017, pp. 915–919.

[32] Odena. A, Olah. C, Shlens. J.(2017). arXiv preprint. "Conditional image synthesis with auxiliary classifier GANs." [Online].Avaliable: https://arxiv.org/abs/1610.09585

[33] Martin. A, Soumith. C, Léon. B.(2017). arXiv preprint. "Wasserstein GAN." [Online].Avaliable: https://arxiv.org/abs/1701.07875

[34] Ishaan. G, Faruk. A, Martin. A, Vincent. D, Aaron. C.(2017). arXiv preprint. "Improved training of Wasserstein GANs." [Online].Avaliable: https://arxiv.org/abs/1704.00028v3