IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 31 May 2024, accepted 12 June 2024, date of publication 19 June 2024, date of current version 27 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3416754

RESEARCH ARTICLE

Radar Signal Recognition Based on CSRDNN Network

ZHENG ZHANG[1], CHUAN WAN[1], YI CHEN[1], FANG ZHOU[1], XIAOFEI ZHU[2], WENCHAO ZHAI[1], AND DAYING QUAN[1], (Member, IEEE)

[1]School of Information Engineering, China Jiliang University, Hangzhou 310018, China
[2]Xi'an Research Institute of High Technology, Xi'an 710025, China

Corresponding author: Chuan Wan (wc0706@cjlu.edu.cn)

This work was supported in part by the National College Student Innovation and Entrepreneurship Training Program under Grant 202310356017, in part by the National Natural Science Foundation of China under Grant 62261014, and in part by Zhejiang Provincial Key Research and Development Program under Grant 2022C01144.

ABSTRACT It is essential to achieve the high-accuracy recognition of low probability of intercept (LPI) radar signals in modern electronic warfare. However, under low signal-to-noise ratio (SNR), the recognition accuracy of the LPI radar signals is relatively low. In this paper, a novel radar signal recognition method based on Convolutional Stacked Recurrent Deep Neural Network (CSRDNN) is proposed. Firstly, we design a Convolutional Neural Network (CNN) to expand the feature space of input time domain signals, the features extracted by CNN were then used as inputs of the Stacked Recurrent Neural Networks (SRNN) module. In the SRNN module, we sequentially stack GRU, LSTM, and BGRU, enabling the model to better handle the short-term and long-term dependence of signal features and effectively solve asynchronous problems in unidirectional RNN networks. Subsequently, a Fully Connected Deep Neural Network (FCDNN) was employed to accomplish the recognition task. In addition, we design a training algorithm composed of the Nesterov-Adaptive Moment Estimation (Nadam) algorithm and the CosineAnnealing Learning Rate (LR) adjustment strategy to improve the training efficiency of the model. The experimental results demonstrate that the proposed model has higher recognition accuracy at low SNR compared to other models, with an overall recognition accuracy of 92.96% at −4 dB.

INDEX TERMS Low probability of intercept, radar signal recognition, stacked recurrent neural networks, activation function, training algorithm.

I. INTRODUCTION

Due to the emergence and application of various new radar systems and electromagnetic warfare systems, the battlefield electromagnetic environment has become increasingly complicated [1], [2]. The pulse signal received by the wide band receiver in the same period increases sharply, causing the loss and overlap of the pulse signal to become more serious [3].

Some traditional low probability of intercept (LPI) radar signal recognition algorithms are based on pulse description word (PDW), which only extract the low-level features of the signal [4]. However, it is difficult to use these low-level features to achieve high-accuracy signal recognition, especially under low signal-to-noise ratio (SNR) [5]. The inter-pulse features are more stably separable than the low-level features extracted from the PDW. Extracting the inter-pulse feature of the LPI radar signal directly can not only increase the size of the feature parameter space but also improve the anti-distortion ability. Besides, the loss of signal characteristics will be significantly reduced. Therefore, researchers began to pay more attention to the characteristics of radar signals and continually proposed corresponding signal recognition methods [6], [7].

In the field of radar signal processing, the internal and external scholars commonly select Time-Frequency Analysis(TFA) methods to obtain the time-frequency domain, time-domain autocorrelation, transform domain, and higher-order statistical features of intra-pulse signals. Hu et al. [8]

The associate editor coordinating the review of this manuscript and approving it for publication was Cheng Hu.

introduced a LPI radar signal recognition method based on pulse accumulation. The effect of pulse accumulation was improved via time domain alignment iteration method (TAIM). Then, the time-frequency images (TFIs) of the accumulated signals were fed into the deep residual network for recognition. When the SNR is at −6 dB, this method exhibits a recognition accuracy that is 7% higher compared to directly identifying traditional single-pulse signals. Gupta and Rai [9] utilized wavelet ridges and high-order statistics to extract signal features. Simulation experiments demonstrated that the overall average recognition accuracy of the fuzzy support vector machine (SVM) classifier used in this algorithm can reach over 80%. In contrast, Huynh-The et al. [10] applied Wiener filtering and Choi-William distribution (CWD) to extract TFIs and constructed a combinatorial recognition algorithm through a multi-layer perceptron network. However, the feature extraction implementation was very cumbersome. And the model training also required numerous sample intra-pulse signals, which made it difficult to satisfy the requirements of the practical application in real electromagnetic environment.

In the past few years, image processing and deep learning techniques have made great breakthroughs in various recognition tasks, and these techniques have gradually been used in radar signal recognition. Park et al. [11] proposed a convolutional neural network (CNN), which is composed of a main classifier and several sub-classifiers. In addition, a two-step CNN scheme was proposed to select and recognize the CWD TFIs. It was shown that the recognition model was with high recognition accuracy and low computational cost. Wang et al. [12] introduced an Asymmetric Dilated Convolution Coordinate Attention Residual network (ADCCA-ResNeXt). Firstly, they learned morphological features from TFIs and then realized signal classification through the ADCCA-ResNeXt network. When the SNR is −8 dB, the recognition accuracy is about 97.94%. Lay and Charlish [13] presented a multilayer feed-forward neural network equipped with a supervised backpropagation training algorithm for signal recognition. Remarkably, the neural network maintains excellent recognition performance even when signals were corrupted by noise.

Recurrent neural network (RNN) [14] model has gained significant approvement and application in the field of image classification and recognition, particularly in radar signal recognition, owing to its exceptional ability to model time domain data, effectively handle long-term and short-term dependencies, and adapt to sequences of variable lengths. However, most existing methods proposed by researchers primarily focus on processing radar signal image samples. Compared with utilizing the original signal sequences as input data, these image-based methods evidently lack physical significance. Additionally, these methods typically classify different reference objects by repetitively stacking single RNN networks [15], [16], [17], or concatenating them with other non-RNN networks (such as attention mechanisms) [18], without considering the integration of three different RNN networks (Gated Recurrent Units (GRU), Long-Short Term Memory (LSTM), and Bidirectional Gated Recurrent Units (BGRU)). The Convolutional, Long Short-Term Memory, Deep Neural Network (CLDNN) [19] network model is a classic RNN model often used for radar signal recognition. However, this model still has some drawbacks, including high computational complexity and the occurrence of gradient vanishing problems.

Based on traditional CLDNN model, Wei et al. [20] chose to use the original signal sequence as the input of the model, and changed the single-layer LSTM in the CLDNN model to dual-layer LSTM. This improved the recognition accuracy of the model when the SNR is below 0 dB. In contrast, Sun et al. [21] replaced the LSTM network in CLDNN with BGRU to directly recognize 8 time domain signals. Compared with the traditional CLDNN model, the above two methods showed some improvements in recognition accuracy. Nevertheless, these two methods overlook the potential benefits of concatenating different RNNs, which restricts the model's expressive power and flexibility in capturing diverse sequence patterns to a certain extent. Therefore, it becomes challenging to effectively capture all sequence patterns, leading to limited recognition accuracy under low SNRs. Therefore, the appropriate integration of different RNNs is considered to better solve the above problems.

To address the limitations of single-type RNN networks and the problem of low recognition accuracy for LPI radar signals under low SNRs, this paper proposes a method based on Convolutional Stacked Recurrent Deep Neural Network (CSRDNN). This approach aims to automatically recognize different LPI radar signals. The CSRDNN model incorporates an improved training algorithm to enhance its performance. Concretely, the CSRDNN model consists of three modules: CNN, Stacked Recurrent Neural Networks (SRNN) module, and Fully Connected Deep Neural Networks (FCDNN). In addition, the improved training algorithm is composed of the Nesterov-accelerated Adaptive Moment Estimation (Nadam) algorithm and the Cosine Annealing learning rate (CosineAnnealingLR) adjustment strategy. Different from the recognition methods based on TFIs, our proposed approach directly extracts signal features from the time domain signal, which effectively improves the real-time performance of feature extraction.

In summary, the main contributions of this work can be summarized as follows:

(1) Distinguished from most recognition methods that first convert the time domain signal into TFIs, and then send the TFIs into the network for classification, we propose a time domain recognition method based on the CSRDNN model, achieving high-accuracy recognition of 12 radar signals.

(2) We have devised a novel SRNN module, which consists of GRU, LSTM, and BGRU. This module provides a more reliable solution for capturing short-term and long-term dependencies in signal features and addressing the asynchronous issues that occur in unidirectional RNNs. By

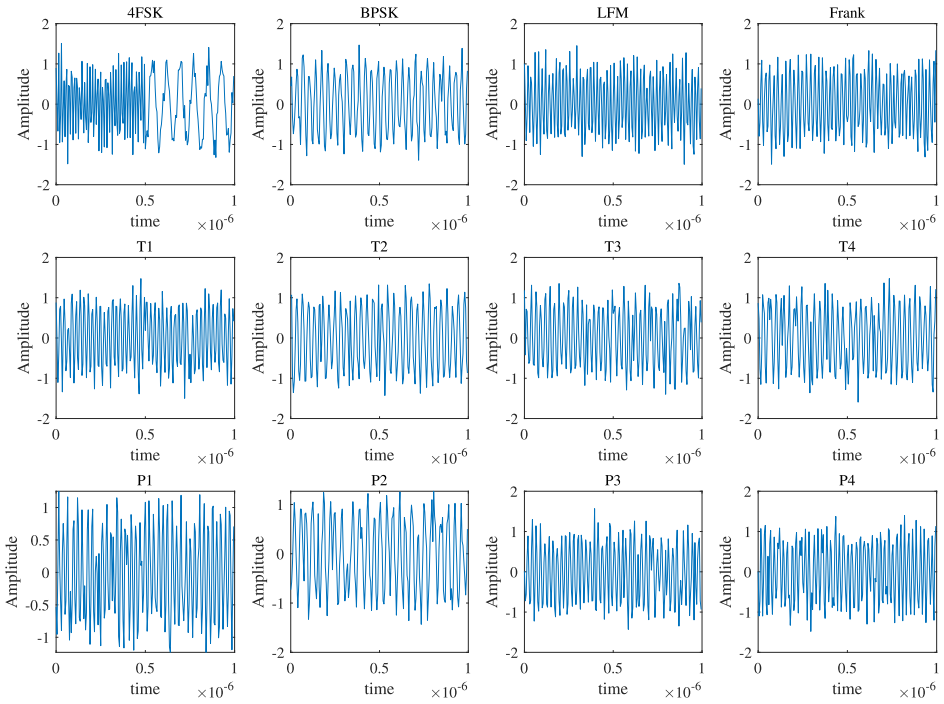

FIGURE 1. Time domain waveforms of 12 basic LPI radar signals.

concatenating the strengths of these different RNN units, it effectively addresses the deficiencies of single-type RNN networks.

(3) An improved training algorithm is introduced with the aim of improving the training efficiency of the CSRDNN model.

This paper is organized as follows. In Section II, a concise overview of the general mathematical model of LPI radar signals is presented. Section III delineates the structures of GRU, LSTM, and BGRU units, along with the Nadam training algorithm. The specific network framework and the comprehensive training process of the CSRDNN model are detailed in Section IV. Section V presents the experimental results and analysis. Finally, the conclusion of this paper is drawn in Section VI.

II. SIGNAL MODEL

We have considered 12 basic LPI radar signals: linear frequency modulation (LFM), 4-level frequency shift keying (4FSK), binary phase shift keying (BPSK) [22], multi-phase code signals (Frank, P1, P2, P3, and P4) [23], and multi-time code signals (T1, T2, T3, and T4) [24]. Figure 1 displays the time domain waveforms of these radar signals.

Normally, the radar signal received by the radar radio interception receiver is composed of the ideal signal and noise. Its general time domain model can be described as

$$r(t) = s(t) + n(t) = A \times e^{j\phi(t)} + n(t) \tag{1}$$

where $r(t)$ is the received signal and $s(t)$ is the desired signal. $n(t)$ means the white Gaussian noise (WGN) with an average value of 0 and a variance of $\boldsymbol{\sigma}^2$. A and $\phi(t)$ represent the amplitude and phase of the desired signal, respectively.

III. RNN NETWORKS AND MODEL TRAINING ALGORITHM

The SRNN module we have designed is a crucial component of the CSRDNN model. It is primarily concatenated by three RNN networks: LSTM, GRU and BGRU. Thus, in order to facilitate an understanding of the role of SRNN module and its training algorithms, this section will offer an overview of these three RNN networks and the Nadam algorithm.

A. LSTM NEURAL NETWORK

LSTM [25] is a specific variant of RNN structure used in the field of deep learning and artificial intelligence. This network is known for its ability to effectively handle gradient-related issues during training, such as vanishing and exploding gradients. It consists of three gates: an input gate, a forgetting gate, and an output gate. The structure of the LSTM is visually depicted in Figure 2.

In Figure 2, the activation value for the forget gate and the input gate are respectively denoted by $\boldsymbol{f}_t$ and $\boldsymbol{i}_t$. $\tilde{\boldsymbol{C}}_t$ stands for the candidate cell state, while $\boldsymbol{C}_t$ signifies the cell state. $\boldsymbol{h}_t$ means the hidden states of LSTM unit and $\boldsymbol{x}_t$ represents the input information.

B. GRU NEURAL NETWORK

GRU is also a variant of RNN, featuring a more streamlined structure and lower computational complexity compared to LSTM. It comprises two fundamental components, namely

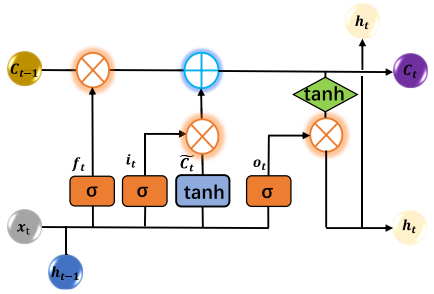

FIGURE 2. The LSTM unit structure.

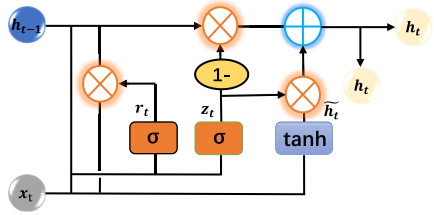

FIGURE 3. The GRU unit structure.

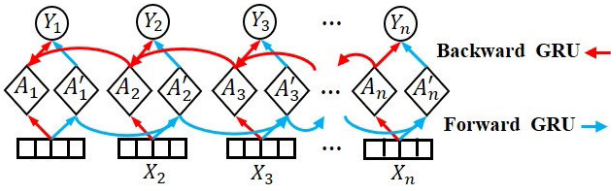

FIGURE 4. Workflow of BGRU.

the update gate and the reset gate, which play pivotal roles in controlling the information flow within the network. The detailed structure of GRU can be seen in Figure 3.

In Figure 3, $\mathbf{z}_t$ and $\boldsymbol{r}_t$ represent the activation values of update gate and reset gate, respectively. $\widetilde{\boldsymbol{f}}_t$ is the candidate hidden state, while $\boldsymbol{h}_t$ is the hidden state of the GRU unit. $\boldsymbol{x}_t$ denotes the input information.

C. BGRU NEURAL NETWORK

BGRU [26] is an enhanced variant of the GRU model that includes both a forward GRU and a backward GRU. The BGRU obtains its final output by weighting the outputs of the forward GRU and backward GRU at the same moment. The workflow of BGRU is depicted in Figure 4.

In Figure 4, $\boldsymbol{A}_n$ and $\boldsymbol{A}'_n$ are the forward and backward GRU, respectively. $\boldsymbol{X}_n$ represents the input information, while $\boldsymbol{Y}_n$ denotes the outputs of BGRU.

D. NADAM GRADIENT DESCENT ALGORITHM

The Nadam algorithm [27] is a gradient descent algorithm that combines the Nesterov momentum estimate and Adam (Adaptive Moment Estimation) algorithm. This algorithm can raise the convergence speed of the model and avoid the problem of gradient estimation error effectively. As we all know, first-moment estimation (FME) and second-moment estimation (SME) are two fundamental parameters in the Adam and Nadam algorithms. The parameters can be used to constrain gradients and adjust learning rates. The update equations for FME and SME at t-th iteration are as follows

$$\begin{cases} \boldsymbol{m}_t = \mu \times \boldsymbol{m}_{t-1} + (1-\mu) \times \boldsymbol{g}_t \\ \boldsymbol{n}_t = v \times \boldsymbol{n}_{t-1} + (1-v) \times \boldsymbol{g}_t^2 \end{cases} \tag{2}$$

where μ is the momentum coefficient in the range of [0, 1]. v signifies the exponential decay rate, whose value is also in the range of [0, 1]. $\boldsymbol{g}_t$ is the gradient at the time.

Based on Equation (2), the Adam algorithm combines traditional momentum estimation methods to obtain the following iterative formulas for each parameter

$$\begin{cases} \hat{\boldsymbol{m}}_t = \dfrac{\boldsymbol{m}_t}{1-\mu^t} = \left(\dfrac{\mu \times \boldsymbol{m}_{t-1}}{1-\mu^t} + \dfrac{(1-\mu) \times \boldsymbol{g}_t}{1-\mu^t} \right) \\ \hat{\boldsymbol{n}}_t = \dfrac{\boldsymbol{n}_t}{1-v^t} = \left(\dfrac{v \times \boldsymbol{n}_{t-1}}{1-v^t} + \dfrac{(1-v) \times \boldsymbol{g}_t^2}{1-v^i} \right) \\ \boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \alpha \times \dfrac{\hat{\boldsymbol{m}}_t}{\sqrt{\hat{\boldsymbol{n}}_t} + \lambda} \\ \boldsymbol{g}_t = \nabla_{\theta_t} \times f\left(\boldsymbol{\theta}_t - \alpha \times \mu \times \dfrac{\hat{\boldsymbol{m}}_{t-1}}{\sqrt{\boldsymbol{n}_{t-1}} + \lambda} \right) \end{cases} \tag{3}$$

where $\hat{\boldsymbol{m}}_t$ and $\hat{\boldsymbol{n}}_t$ are the bias correction estimations of current FME and SME, respectively. $\boldsymbol{\theta}_t$ denotes the learning rate parameter at the current time, while α signifies the learning rate, with a value range from 0 to 1. To avoid the occurrence of excessively large α, a random positive value is represented by λ. $\nabla_{\boldsymbol{\theta}_t}$ represents the partial derivative with respect to $\boldsymbol{\theta}_t$. $f\left(\boldsymbol{\theta}_t - \alpha \times \mu \times \frac{\hat{\boldsymbol{m}}_{t-1}}{\sqrt{\boldsymbol{n}_{t-1}}+\lambda} \right)$ is the loss function at the t-th iteration.

Similarly, based on Equation (2) and (3), the traditional momentum estimation methods in the Adam algorithm are replaced by Nesterov momentum estimation methods in the Nadam algorithm. Additionally, $\hat{\boldsymbol{m}}_t$ and $\hat{\boldsymbol{g}}_t$ are utilized to replace $\hat{\boldsymbol{m}}_{t-1}$ and $\boldsymbol{g}_t$ used in Adam algorithm, respectively. Then the update rules for the Nadam algorithm are obtained as follows

$$\begin{cases} \hat{\boldsymbol{m}}_t = \dfrac{\boldsymbol{m}_t}{1 - \prod_{i-1}^{t+1} \mu_i} \\ \hat{\boldsymbol{g}}_t = \dfrac{\boldsymbol{g}_t}{1 - \prod_{i-1}^{t+1} \mu_i} \\ \boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \dfrac{\alpha}{\sqrt{\hat{\boldsymbol{n}}_t} + \lambda} \left(\dfrac{\mu \times \hat{\boldsymbol{m}}_t}{1-\mu^t} + \dfrac{(1-\mu) \times \hat{\boldsymbol{g}}_t}{1-\mu^t} \right) \end{cases} \tag{4}$$

where μ_t represents the momentum coefficient of the i-th sample. From Equation (4), it can be observed that the Nadam algorithm imposes stronger constraints on the learning rate and exerts a more direct influence on gradient updates.

IV. THE LPI RADAR SIGNAL RECOGNITION METHOD BASED ON THE CSRDNN MODEL

A. THE STRUCTURE OF CSRDNN MODEL

The proposed CSRDNN model is generally designed for achieving high-accuracy recognition of LPI radar signals

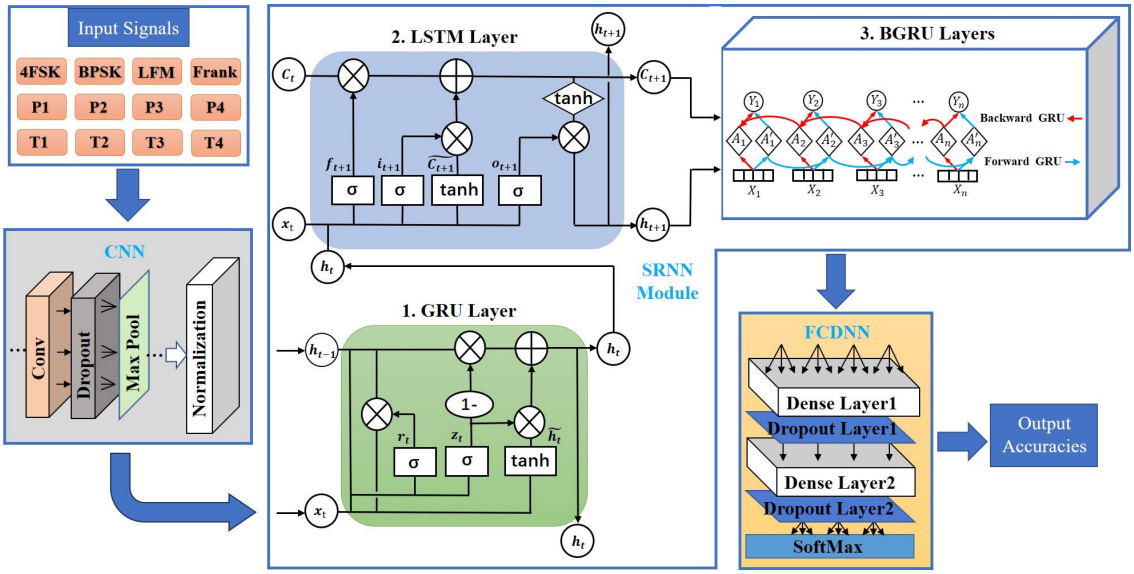

FIGURE 5. Radar signal recognition implementation framework flow.

under low SNR. The overall structure and workflow diagram of CSRDNN are shown in Figure 5.

As shown in Figure 5, the time domain data of 12 radar signals are first inputted into the CNN module. Through the different layers of the CNN, feature extraction, risk reduction of overfitting, and downsampling processing of the signals are performed. This whole process not only helps reduce redundant information but also enables the extraction of highly discriminative features from radar signals, effectively enhancing the model's generalization ability.

Following the above processing steps and performing batch normalization as the final step in the CNN, the resulting feature outputs are fed into the first part of the SRNN module, the GRU network, which specializes in learning short-term dependencies within sequential data. The outputs are then transmitted to the second part of the SRNN module, the LSTM network, which can further capture the long-term information. Subsequently, the outputs of LSTM pass through the last part of the SRNN module, the BGRU network, which is used to address the temporal asymmetry problems in unidirectional RNN network layers and enhance the integrity of information. Finally, in the FCDNN, the dense layers are utilized to perform one-dimensional serialization for the multi-dimensional feature matrix, and the features are mapped through the "SoftMax" layer, thus achieving the final recognition results of radar signals.

Specifically, as shown in Figure 6, the CNN module of the CSRDNN model built in this paper is mainly composed of three "convolution + pooling" layers and one batch normalization layer. Each "convolution + pooling" layer contains a convolution layer, a dropout layer, and a maxpool layer, where the discard rate of each dropout layer is taken as 0.2. The related parameter settings are provided in Figure 6. The first "convolution + pooling" layer sets the number

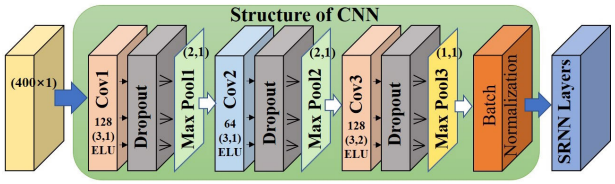

FIGURE 6. Structure of CNN.

of output channels in the convolutional layer to 128, and employs a convolutional kernel with a size of 3 and a stride of 1, accompanied by a maximum pooling layer with a window size of 2 and a stride of 1. The second "convolution + pooling" layer sets the number of output channels in the convolutional layer to 64, and utilizes a convolutional kernel with a size of 3 and a stride of 1, paired with a maximum pooling layer with a window size of 2 and a stride of 1. In the third "convolution + pooling" layer, we set the number of output channels in the convolutional layer to 128, and employ a convolutional kernel with a size of 3 and a stride of 2, accompanied by a maximum pooling layer with a window size of 1 and a stride of 1. Finally, we incorporate a batch normalization layer at the end of the "convolution + pooling" combination. This layer normalizes the feature data for each mini-batch, resulting in more stable feature distributions and accelerating the convergence of the network. The above designs for CNN structure can effectively extract useful features from the input data for subsequent classification or other tasks.

B. DESIGN ADVANTAGES OF CSRDNN MODEL

Compared with traditional CLDNN network and various improved RNN networks, the proposed CSRDNN model in this paper makes the following improvements and innovations:

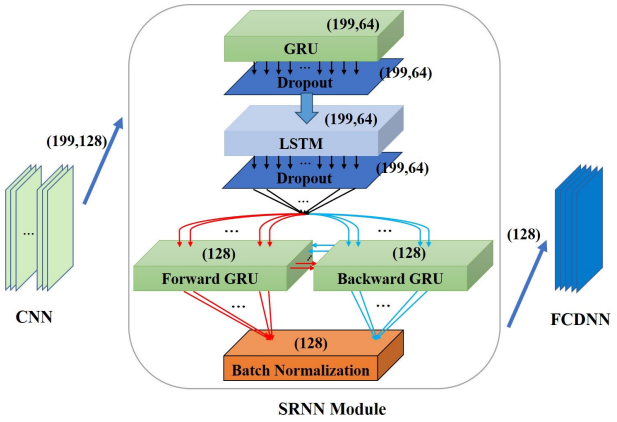

FIGURE 7. Structure of the SRNN module.

(1). The CSRDNN model has introduced a notable module called SRNN, which aims to replace the double-layer LSTM in the traditional CLDNN model. This SRNN module is a fusion of three RNN networks: GRU, LSTM, and BGRU. The GRU network can selectively update or forget the state information of the current moment by using the update door and the reset door. It is good at learning short-term dependencies. The LSTM network, by contrast, controls the flow of information through forget, input, and output gates, which can better capture the long-term dependencies in time domain data. Last but not least, the BGRU network combines forward and backward propagation to comprehensively capture contextual information in sequential data. We fused these three networks to construct the SRNN module. Concatenating these three networks in a reasonable form not only improves the capacity to comprehend input dynamics but also enhances the model's flexibility and expressiveness. By fully exploiting the complementary advantages between them, the gradient distortion problem can be better solved and the computational complexity will be reduced.

(2). To ensure the completeness of the extracted information and enhance the network's fitting capability, we linearly integrate GRU, LSTM, and BGRU in the sequence of GRU->LSTM->BGRU according to the structural characteristics of each type of RNN network. Meanwhile, we have taken into account the size of the dataset, the complexity of the network, and the experimental results under different channel numbers, eventually determining to set the output channel of all three network layers to 64. The detailed structure of the SRNN module is illustrated in Figure 7.

Notably, the linear integration approach of GRU, LSTM, and BGRU that we designed has certain advantages over other connection methods:

(a). It forms a progressive sequence information extraction and modeling process, thereby enhancing the expressive power of the model. The gated mechanism of GRU and LSTM allows the network to selectively update and ignore inputs at each time step, realizing learning short-term and long-term dependencies in the time sequence respectively, and enhancing the model's ability to represent input data. The BGRU layer considers both past and future context at each time step, effectively improving the model's ability to capture bidirectional dependencies in the sequence;

(b). This enables gradients to propagate more easily between network layers. In the backpropagation process, this kind of linear arrangement is conducive to better flow of gradients within the model, improving the stability of the training process.

(3). In this paper, we replaced the activation functions in CNN and FCDNN with Exponential Linear Units (ELU) [28] and Leaky Rectified Linear Units (LeakyReLU) [29], respectively. We also adopted the He Initialization method [30], which is suitable for network layers with activation functions of ReLU and its variants, as the initialization approach for the CSRDNN network. Traditionally, activation functions like ReLU, tanh, and sigmoid have been used in CNNs and their improved versions for training time domain data, with random parameter initialization being a common practice. In comparison, our strategy of adopting ELU and LeakyReLU activation functions and the He Initialization method offers several advantages:

(a). Better handling of gradient vanishing: Traditional activation functions such as ReLU, tanh, and sigmoid result in nearly zero derivatives when input values are large or small, leading to gradient vanishing. The derivatives of ELU and LeakyReLU are non-zero, which helps mitigate this problem;

(b). Faster convergence rate: Due to the smoother derivatives of ELU and LeakyReLU, the learning speed is faster, which accelerates the convergence of the model;

(c). Improved generalization performance: Using ELU and LeakyReLU activation functions in combination can reduce the risk of overfitting, thereby enhancing the model's generalization ability;

(d). Preserving the variance of activation values: According to research by Kaiming He et al. on parameter initialization, the He Initialization method helps maintain the variance of activation values approximately equal across each layer, allowing the network to learn more effectively during training.

C. MODEL TRAINING PROCEDURE

Based on the aforementioned CSRDNN model, training is conducted following the specific steps outlined below:

Step 1: Data preprocessing. The data preprocessing phase involves reading the time domain signal data for recognition and randomly dispersing it. Subsequently, the data is divided into training and test sets in 8:2.

Step 2: Optimization algorithm selection. To expedite model convergence and enhance its robustness, Nadam is selected as the optimization algorithm for the model. The initial learning rate is set to 0.001, and the maximum training epochs are set to 100.

Step 3: Learning rate adjustment. A cosine annealing learning rate adjustment strategy is adopted to dynamically adjust the learning rate. After the initial heating phase linearly increases the learning rate to the preset value of 0.001, the

TABLE 1. Parameters of the LPI radar signals.

Signal Types	Types of parameters	Range of parameters
ALL	Sampling frequency f_s	$1(f_s = 200\,\text{MHz})$
4FSK	Fundamental frequency f_h	$U(1/80, 1/2)$
BPSK	Carrier frequency f_c	$U(1/8, 1/4)$
	Barker codes N_c	{7,11,13}
LFM	Carrier frequency f_c	$U(1/8, 1/2)$
	Bandwidth B	$U(1/16, 1/8)$
	Barker codes N_c	{7,11,13}
FRANK	Carrier frequency f_c	$U(1/8, 1/4)$
	Samples of frequency stem M	$[4, 8]$
P1-P4	Carrier frequency f_c	$U(1/8, 1/4)$
	Samples of frequency stem M	$[4, 8]$
T1-T4	Carrier frequency f_c	$U(1/8, 1/4)$
	Number of segments K	$[4, 5]$

learning rate decreases nonlinearly according to a cosine function.

Step 4: Model updating. A policy is established to retain only the best-performing model. Whenever there is an improvement in the accuracy of the testing set recognition in the subsequent epochs, the original model is overwritten.

Step 5: Introduction of early stopping mechanism. To prevent overfitting during model training, an early stopping mechanism is introduced. If the accuracy of the test set does not increase after 20 epochs of training, the training is prematurely terminated. Otherwise, the model concludes training after 100 epochs.

Step 6: Network training. Once the network model structure is finalized, the preprocessed data is fed into the network for training. Finally, the best-performing model along with the relevant experimental data obtained during training is saved.

V. ANALYSIS OF SIMULATION RESULTS

In this part, we verify the effectiveness and reliability of the proposed radar signal recognition method, and a series of experiments are provided. Section V-A details the dataset design. Sections V-B contrasts the effects of different learning rate strategies and model training algorithms. Section V-C compares the influence of different radar signal forms on recognition accuracy. Section V-D gives the compare results of the model with four widely used signal recognition methods. Section V-E compares the signal recognition performance of CSRDNN model on Rayleigh fading channel and AWGN channel. Section V-F shows the confusion matrix for the 12 radar signals at −4 dB SNR.

All network models in this paper are constructed using TensorFlow as the backend of the Keras framework. The experiments are performed on computer hardware with the following specifications: 11th Gen Intel(R) Core (TM) i5-11400H @ 2.70GHz and 16GB RAM, GPU: NVIDIA Ge-Force RTX3050.

A. DATASET DESIGN

In this work, we generated 300 analog signals for each LPI radar signal at 2 dB steps based on the 12 typical LPI radar signals mentioned in Section II, combined with the parameter settings of the various LPI radar signals in Table 1. Each analog signal had a length of 400, resulting in time domain data for the LPI radar signals. Subsequently, we performed autocorrelation calculations on the obtained time domain data to produce the corresponding autocorrelation domain data. Before training the data, we divided it into training and test sets in a ratio of 8:2. Finally, we read and trained the processed data.

B. LEARNING RATE ADJUSTMENT STRATEGY AND TRAINING ALGORITHM COMPARISON

1) LEARNING RATE ADJUSTMENT STRATEGY COMPARISON

To select the optimal learning rate adjustment strategy for training the CSRDNN model, this study retained a constant neural network architecture and parameter configuration while contrasting the impact of four different learning rate adjustment strategies on the model's recognition capacity. These strategies including ReduceLROnPlateau, LRScheduler, ExponentialLR, and CosineAnnealingLR [31], [32], [33]. During the experiments, we systematically recorded the performance of the optimal model trained with each learning rate strategy in terms of training loss, test loss, training accuracy, test accuracy, and the inference time. The results are presented in Table 2.

As can be seen from Table 2, the inference time range of the four algorithm strategies ranges from 40s to 43s, just a little different. However, by comparing training loss and test accuracy, it is evident that the CosineAnnealingLR outperforms the other strategies. Furthermore, from the perspective of training accuracy and test loss, the CosineAnnealingLR still demonstrates decent performance. Therefore, based on the above analysis, this experiment decides to adopt CosineAnnealingLR strategy as the adjustment method of learning rate in model training.

2) TRAINING ALGORITHM COMPARISON

In order to emphasize the advantages of integrating the Nadam training algorithm with the cosine annealing learning rate adjustment strategy, we employed three training methods for the model in this experiment, including the Adam

TABLE 2. Comparison of 4 learning rate adjustment strategies.

Strategy	Test Loss	Train Loss	Test Accuracy/%	Train Accuracy/%	Inference Time/s
LRScheduler	0.353	0.551	82.6	87.2	43
ReduceLROnPlateau	**0.197**	0.552	83.2	**93.2**	42
ExponentialLR	0.313	0.546	83.5	88.8	**40**
CosineAnnealingLR	0.198	**0.527**	**84.2**	93.1	41

TABLE 3. Comparison of different training algorithms.

Strategy	Test Loss	Training Loss	Test Accuracy/%	Training Accuracy/%	Inference Time/s
Nadam	0.676	0.203	82.8	92.9	39
Adam	0.690	0.198	83.1	**93.2**	**35**
Nadam with CosineAnnealingLR	**0.620**	**0.197**	**84.0**	93.0	41

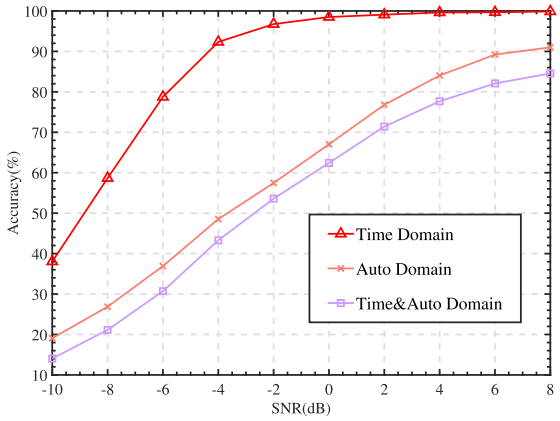

FIGURE 8. The recognition accuracy of 3 different forms of input signal.

training algorithm, the Nadam training algorithm, and the integration of the Nadam training algorithm with the CosineAnnealingLR strategy. All other parameters remained constant. Table 3 presents the results corresponding to the optimal models obtained from each training strategy, including training loss, test loss, training accuracy, test accuracy, and inference time.

From the results in Table 3, we can observe that compared to both the Nadam and Adam algorithms, the method concatenating the Nadam algorithm with the CosineAnnealingLR yielded the lowest training and test losses, along with the highest training accuracy. Therefore, we adopt the integration of the Nadam algorithm with the CosineAnnealingLR for model training.

C. IMPACT OF INPUT RADAR SIGNAL FORM ON RECOGNITION ACCURACY

To validate the effectiveness of selecting time domain radar signals for recognition, this study trained three forms of signals, including time domain, autocorrelation domain, and dual-channel signals combining time domain and autocorrelation domain. Subsequently, the signals of these three forms were input into the CSRDNN network, and the experimental results are shown in Figure 8.

Based on the results in Figure 8, the following conclusions can be drawn:

(1). Using only time domain signals for network training can achieve a high recognition accuracy, indicating that time domain signals already contain sufficient useful information for the classifier to make accurate predictions.

(2). The amount of useful information in the autocorrelation domain data is much less than that in the time domain data.

(3). Since the number of radar signal samples in this experiment is 3600, using dual-channel fusion features of time domain and autocorrelation domain for signal recognition easily introduces redundant information, leading to model overfitting and further degrading the recognition performance of the network.

In summary, this experiment effectively demonstrates the feasibility of selecting radar time domain data for classification recognition.

D. COMPARISON AND ANALYSIS OF NETWORK PERFORMANCE

1) ANALYSIS OF THE PERFORMANCE OF EACH COMPONENT IN THE GRU_LSTM_BGRU

In this section, we conducted ablation experiments to investigate the contributions of individual network components to the performance of the proposed SRNN model (i.e., GRU_LSTM_BGRU). Specifically, we conducted experiments on seven network objects, including GRU, LSTM, BGRU, GRU_LSTM, GRU_BGRU, LSTM_BGRU, and GRU_LSTM_BGRU, which are based on the same CNN and FCDNN structure. Figure 9 presents the recognition accuracy of the aforementioned seven networks under different SNR conditions.

From Figure 9, we can draw the following conclusions:

(1). The BGRU network achieves significantly higher accuracy in the recognition task compared to GRU and LSTM. When the SNR is -4 dB, the recognition accuracy of BGRU reaches 90.71%. This indicates that the BGRU is more effective than the unidirectional GRU and LSTM in handling sequence data. Furthermore, this observation provides strong evidence for the importance of learning sequence context information in recognition tasks.

(2). Additionally, when comparing the recognition accuracies of GRU, LSTM, and GRU_LSTM models, it is found

TABLE 4. Comparison of each model's complexity.

Model	FLOPs [G^1]	Learnable Parameters [M^2]	Inference Time [s]
CSRDNN	**1.08**	**0.18**	**30**
CTDNN	0.73	0.55	11
MobileNetV1	0.41	1.55	23
MobileNetV2	0.32	1.54	51
Improved CLDNN	1.05	0.14	21

[1][G] represents one billion floating-point operations per second.
[2][M] represents the unit of millions of parameters.

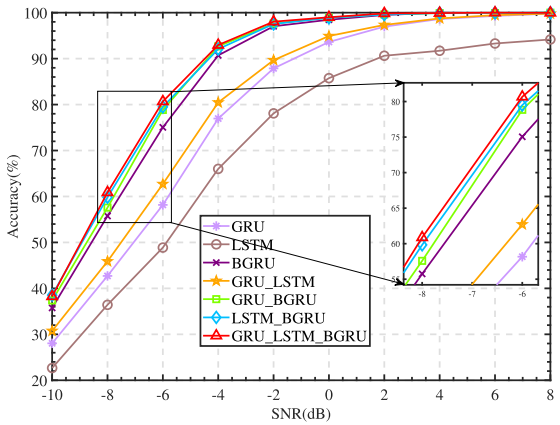

FIGURE 9. The ablation experiment about GRU_LSTM_BGRU.

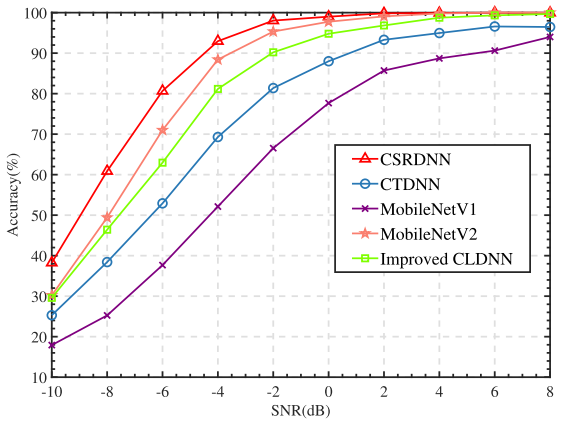

FIGURE 10. The recognition accuracies of different methods.

that the performance of the GRU_LSTM model is surpasses that the GRU and LSTM separately. When the SNR is −4 dB, the recognition accuracy of the GRU_LSTM reaches 80.46%, outperforming the GRU by 3.5% and the LSTM by 14.46%. This suggests that concatenating GRU and LSTM leads to improved model performance.

(3). Lastly, compared to the GRU_BGRU and LSTM_BGRU models, the GRU_LSTM_BGRU model exhibits slightly higher recognition accuracies. When the SNR is −6 dB, the recognition accuracy of GRU_LSTM_BGRU reaches 80.67%, which is 1.83% and 1.17% higher than that of the GRU_BGRU and LSTM_BGRU models, respectively. It is indicates that concatenating GRU, LSTM, and BGRU further enhances the model's performance. This enhancement is not only attributed to the advantage of BGRU in capturing bidirectional information in sequences, but also to the better comprehension and utilization of complex patterns in sequences achieved by integrating GRU and LSTM.

In summary, GRU_LSTM_BGRU achieves the best performance in terms of signal recognition accuracy, particularly in low SNR conditions. When the SNR reaches −4 dB, the recognition accuracy is as high as 92.96%.

2) PERFORMANCE COMPARISON OF DIFFERENT SIGNAL RECOGNITION ALGORITHMS

To further demonstrate the effectiveness of the proposed CSRDNN model, four typical existing signal recognition algorithms were selected for comparison under the same conditions. These models include the improved CLDNN [20], Convolutional Temporal Delay Neural Network (CTDNN) [34], MobileNetV1 [35], and MobileNetV2 [36].

So as to fairly compare the performance of different methods, we provide the computational complexity (FLOPs), space complexity (number of learnable parameter), and time complexity (inference time required for training one epoch) of different models in Table 4. Meanwhile, the recognition accuracy results of different models are shown in Figure 10.

Based on the computational results in Table 4, it can be observed that MobileNetV2 has the lowest computational complexity, while the improved CLDNN exhibits the lowest space complexity, and CTDNN demonstrates the lowest time complexity. Our proposed CSRDNN, on the other hand, possesses the highest computational complexity, reaching 1.08 G. Simultaneously, it also showcases a relatively higher time complexity, requiring 30 s of inference time. However, it is noteworthy that its space complexity is comparatively low, being only 0.18 M.

From Figure 10, it can be seen that the proposed CSRDNN model exhibits excellent robustness to interference and noise, demonstrating significantly higher recognition accuracy under various SNR conditions. Specifically, when the SNR at −4 dB, the recognition accuracy of CSRDNN reaches 92.96%, outperforming CTDNN, MobileNetV1, MobileNetV2, and improved CLDNN by 23.67%, 40.83%, 4.54%, and 11.79%, respectively. In contrast, due to the limitations of their network structures, CTDNN, MobileNetV1, MobileNetV2, and improved CLDNN are more sensitive to

TABLE 5. Parameter setting for the Rayleigh fading channel.

Parameters	Sample Rate/MHz	Multichannels Delay/ns	Multichannels Attenuation/dB	Max Doppler Shift/Hz
Range	200	delay1:0 delay2: $\{1,51,101,\ldots,951\}$ delay3: $\{1,51,101,\ldots,951\}$ delay4: $\{1,51,101,\ldots,951\}$	attenuation1: $\{-10,-8,\ldots,0\}$ attenuation2: $\{-10,-8,\ldots,0\}$ attenuation3: $\{-10,-8,\ldots,0\}$ attenuation4: $\{-10,-8,\ldots,0\}$	[5, 400]

noise and other interference factors, leading to a significant decline in recognition performance under the same conditions.

In conclusion, considering the computational complexity and recognition performance of the CSRDNN network model, it remains highly feasible for actual applications.

E. CHANNEL SIMULATION COMPARISON EXPERIMENT

In order to more accurately assess the performance of the proposed model under real-world signal transmission scenarios, yet confronted with the considerable challenge of acquiring LPI radar real data that meets our requirements, we have implemented a series of measures to emulate the impacts of real-world environmental factors on signal propagation.

Specifically, considering the multipath effect and fading of signal propagation in real environment, we explore the recognition performance of CSRDNN model and improved CLDNN model over Rayleigh fading channel [37], and compare these with the aforementioned models on AWGN channels. The comparison results are presented in Figure 11. In the configuration of the Rayleigh fading channel, we adopted the following parameter settings: the number of channel paths is 4, the sampling frequency of the channel is 200MHz, the maximum Doppler shift range is [5Hz, 400Hz], and the multipath fading values are within the range of {−10dB, −8dB, …, 0dB}. Additionally, in the setting of channel delays, we set the delay of the main path as 0 to represent direct propagation with no delay. Furthermore, the delay values of the other three subpaths range from {1ns, 51ns, 101ns, …, 951ns}, simulating the multipath propagation effects with different delay scenarios. The above settings for Rayleigh fading channels are shown in Table 5.

Subsequently, we separately input the generated radar signals with AWGN and the signals over the Rayleigh fading channel into the CSRDNN and improved CLDNN models for signal recognition experiments.

The experimental results shown in Figure 11 demonstrate that compared with the recognition accuracy of CSRDNN on AWGN channel, the recognition performance decreases over Rayleigh fading channel. However, CSRDNN model still has better recognition performance than the improved CLDNN model on the same type of channel. The above results verify the reliability of CSRDNN method in practical application.

We also observe that both CSRDNN model and improved CLDNN model have lower recognition accuracy on Rayleigh fading channels. This may be attributed to the introduction of fading channels, which brings about fading, multipath effects, and time delay spread, among other interferences. These interferences cause the received signals to become blurred, resulting in differences in signal phase, amplitude, and temporal characteristics. Consequently, these factors make it more challenging to accurately distinguish between different signals, which reduces recognition accuracy.

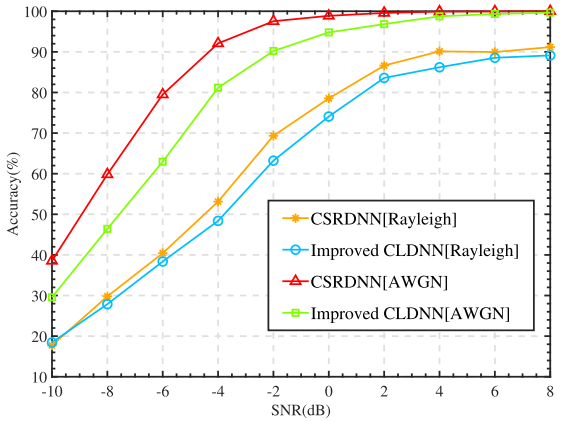

FIGURE 11. Comparison of signal recognition accuracy in Rayleigh channel and AWGN channel.

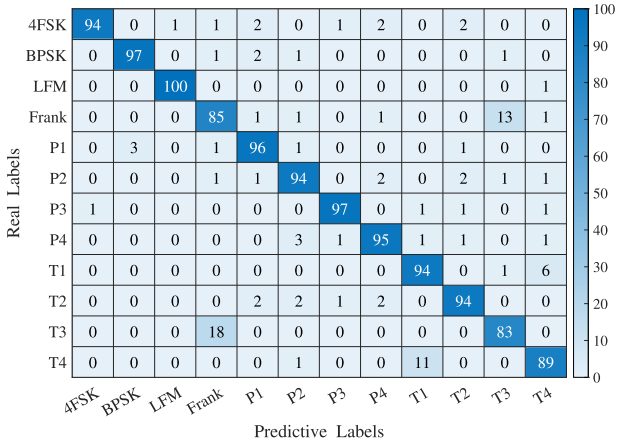

FIGURE 12. Confusion matrix of the CSRDNN (SNR = −4 dB).

F. CONFUSION MATRIX OF THE CSRDNN NETWORK MODEL

In order to analyze the recognition performance of the proposed CSRDNN network under different type of signals, we have generated a confusion matrix depicting the

CSRDNN network's performance under −4 dB SNR in Figure 12.

From Figure 12, we can observe that in the case of strong noise interference with a SNR of −4 dB, some different types of radar signals with similar characteristics are easily confused. For instance, the recognition accuracy of the Frank code is 85%, with 13% being incorrectly recognized as the T3 code. Similarly, the recognition accuracy of the T3 code is only 83%, with 18% being incorrectly recognized as Frank code. In addition, there is also a certain degree of confusion and misjudgement in the T1 and T4 codes. Obviously, the confusion between radar signals with similar characteristics has highlighted the challenges of signal recognition brought by strong noise interference and the lack of obvious differences in signal structure.

VI. CONCLUSION

In this paper, we have presented a CSRDNN-based method that significantly enhances the recognition performance of LPI radar time domain signals. The method comprises a high-performance CSRDNN model consisting of three core network modules (CNN, SRNN, and FCDNN) designed to capture relevant features from sequential data. This model effectively strengthens the extraction capability of temporal features from radar signals. Additionally, we introduce an improved training algorithm integrating the Nadam algorithm and Cosine Annealing learning rate adjustment strategy. This algorithm effectively enhances the convergence speed and recognition efficiency of the CSRDNN model. Experimental results demonstrate that the CSRDNN network achieves a high radar signal recognition accuracy of 92.96% under −4 dB SNR, outperforming other 6 networks in the ablation experiment and 4 widely used signal recognition algorithms, including improved CLDNN, CTDNN, MobileNetV1, and MobileNetV2. Our future work will primarily focus on improving the feature extraction performance of radar signals, and optimizing neural network structures to enhance the recognition capabilities of radar signals.

REFERENCES

[1] P. Sharma, K. K. Sarma, and N. E. Mastorakis, "Artificial intelligence aided electronic warfare systems-recent trends and evolving applications," *IEEE Access*, vol. 8, pp. 224761–224780, 2020.
[2] Z. Pan, S. Wang, M. Zhu, and Y. Li, "Automatic waveform recognition of overlapping LPI radar signals based on multi-instance multi-label learning," *IEEE Signal Process. Lett.*, vol. 27, pp. 1275–1279, 2020.
[3] T. Chen, L. Liu, and X. Huang, "LPI radar waveform recognition based on multi-branch MWC compressed sampling receiver," *IEEE Access*, vol. 6, pp. 30342–30354, 2018.
[4] G. Qiang, T. Long, W. Xinliang, Q. Liangang, and S. Wenming, "Deinterleaving of radar pulse based on implicit feature," *J. Syst. Eng. Electron.*, vol. 34, no. 6, pp. 1–13, Feb. 2023.
[5] Y. Zhao, H. Feng, K. Jiang, and B. Tang, "Information fusion for radar signal sorting with the distributed reconnaissance receivers," *Remote Sens.*, vol. 15, no. 15, p. 3743, Jul. 2023.
[6] W. Si, C. Wan, and Z. Deng, "Intra-pulse modulation recognition of dual-component radar signals based on deep convolutional neural network," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3305–3309, Oct. 2021.
[7] T. Wang and H. Jiang, "Real-time analysis of intra-pulse characteristics based on instantaneous frequency," in *Proc. 15th Int. Congr. Image Signal Process., Biomed. Eng. Informat. (CISP-BMEI)*, Nov. 2022, pp. 1–6.
[8] Z. Hu, D. Hu, J. Wang, and J. Huang, "A pulse accumulation method for LPI radar emitter signal recognition," in *Proc. 4th Int. Conf. Intell. Control, Meas. Signal Process. (ICMSP)*, Jul. 2022, pp. 875–880.
[9] A. Gupta and A. A. Bazil Rai, "Feature extraction of intra-pulse modulated LPI waveforms using STFT," in *Proc. 4th Int. Conf. Recent Trends Electron., Inf., Commun. Technol. (RTEICT)*, May 2019, pp. 742–746.
[10] T. Huynh-The, V.-S. Doan, C.-H. Hua, Q.-V. Pham, T.-V. Nguyen, and D.-S. Kim, "Accurate LPI radar waveform recognition with CWD-TFA for deep convolutional network," *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1638–1642, Aug. 2021.
[11] D.-H. Park, J.-H. Bang, J.-H. Park, and H.-N. Kim, "A fast and accurate convolutional neural network for LPI radar waveform recognition," in *Proc. 19th Eur. Radar Conf. (EuRAD)*, Sep. 2022, pp. 89–92.
[12] X. Wang, G. Xu, H. Yan, D. Zhu, Y. Wen, and Z. Luo, "LPI radar signals modulation recognition based on ACDCA-ResNeXt," *IEEE Access*, vol. 11, pp. 45168–45180, 2023.
[13] B. Lay and A. Charlish, "Classifying LPI signals with transfer learning on CNN architectures," in *Proc. Sensor Signal Process. Defence Conf. (SSPD)*, Sep. 2020, pp. 1–5.
[14] Wolf, "Recurrent nets for the storage of cyclic sequences," in *Proc. IEEE International Conference Neural Networks*, vol. 1, Jul. 1988, pp. 53–60.
[15] S. G. Bhatti and A. I. Bhatti, "BiLSTM based phase modulation detection of radar emitters," in *Proc. CIE Int. Conf. Radar (Radar)*, Dec. 2021, pp. 3272–3276.
[16] Y. Jiang, W. Sheng, D. Cheng, L. Xiang, R. Song, and W. Jiang, "Individual recognition of big data radar digital waveform based on long short-term memory network," in *Proc. IEEE 3rd Int. Conf. Inf. Technol., Big Data Artif. Intell. (ICIBA)*, vol. 3, May 2023, pp. 857–862.
[17] S. R. Borra, V. R. Kaneti, V. Tejaswini, S. Anitha, and R. Vinaykumar, "Fruit type classification using stacked bi-directional long short term memory," in *Proc. Int. Conf. Evol. Algorithms Soft Comput. Techn. (EASCT)*, Oct. 2023, pp. 1–4.
[18] M. Pan, A. Liu, Y. Yu, P. Wang, J. Li, Y. Liu, S. Lv, and H. Zhu, "Radar HRRP target recognition model based on a stacked CNN-Bi-RNN with attention mechanism," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, Feb. 2022, doi: 10.1109/TGRS.2021.3055061.
[19] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4580–4584.
[20] S. Wei, Q. Qu, H. Su, M. Wang, J. Shi, and X. Hao, "Intra-pulse modulation radar signal recognition based on CLDN network," *IET Radar, Sonar Navigat.*, vol. 14, no. 6, pp. 803–810, Jun. 2020.
[21] Y. Sun, R. Tian, and X. Wang, "Emitter signal recognition based on improved cldnn," *Syst. Eng. Electron.*, vol. 43, no. 1, pp. 42–47, Jun. 2021.
[22] Y. Dong, "Implementable phase-coded radar waveforms featuring extra-low range sidelobes and Doppler resilience," *IET Radar, Sonar Navigat.*, vol. 13, no. 9, pp. 1530–1539, Jul. 2019.
[23] P. S. Tan, J. Jakabosky, J. M. Stiles, and S. D. Blunt, "Higher-order implementations of polyphase-coded FM radar waveforms," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 6, pp. 2850–2870, Dec. 2019.
[24] L. Gao, P. Qin, H. Li, S. Wang, H. Sun, and Y. Lu, "Transfer learninng in polytime codes signal recognition," in *Proc. IEEE 4th Int. Conf. Signal Image Process. (ICSIP)*, Jul. 2019, pp. 91–95.
[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[26] K. Kawakami, "Supervised sequence labelling with recurrent neural networks," Ph.D. dissertation, Technical Univ. Munich, Munich, Germany, 2008.
[27] T. Dozat, "Incorporating Nesterov momentum into Adam," in *Proc. Int. Conf. Learn. Represent.*, Feb. 2016, pp. 1–20.
[28] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," 2015, *arXiv:1511.07289*.
[29] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. Int. Conf. Learn. Represent.*, Jun. 2013, vol. 30, no. 1, p. 3.
[30] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
[31] H. W. Oh, W. S. Jeong, and S. E. Lee, "Evaluation of posit arithmetic on machine learning based on approximate exponential functions," in *Proc. 19th Int. SoC Design Conf. (ISOCC)*, Oct. 2022, pp. 358–359.

[32] L. Xu, D. Niu, Y. Yang, A. Wang, Z. Jin, Y. Dong, and C. Zhang, "Adaptive transient stepping policy on reinforcement learning," in *Proc. Int. Symp. Electron. Design Autom. (ISEDA)*, May 2023, pp. 46–51.
[33] Z. Liu, "Super convergence cosine annealing with warm-up learning rate," in *Proc. CAIBDA ; 2nd Int. Conf. Artif. Intell., Big Data Algorithms*, Jun. 2022, pp. 1–7.
[34] S. Ying, S. Huang, S. Chang, Z. Yang, Z. Feng, and N. Guo, "A convolutional and transformer based deep neural network for automatic modulation classification," *China Commun.*, vol. 20, no. 5, pp. 135–147, May 2023.
[35] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[36] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.
[37] N. Belwal, G. S. Jethi, Pradeep. K. Juneja, and V. Joshi, "Investigating the impact of Doppler shift performance of multipath Rayleigh fading channel," in *Proc. IEEE 5th Int. Conf. Comput. Commun. Autom. (ICCCA)*, Oct. 2020, pp. 473–476.

ZHENG ZHANG is currently pursuing the B.S. degree in communications engineering with China Jiliang University, Hangzhou, China. His research interests include modulation classification and digital signal processing.

CHUAN WAN received the Ph.D. degree from Nanjing University of Science and Technology, Nanjing, Jiangsu, China, in 2021. She is currently with the School of Information Engineering, China Jiliang University, China. Her research interests include pattern synthesis, array signal processing, and intelligent wireless sensing.

YI CHEN received the B.S. degree in electronics and information engineering from Xiamen University Tan Kah Kee College, Zhangzhou, China, in 2023. He is currently pursuing the M.S. degree with China Jiliang University, Hangzhou, China. His research interests include radar signal recognition and modulation classification.

FANG ZHOU received the B.S., M.S., and Ph.D. degrees in electronics engineering from Xidian University, Xi'an, China, in 2006, 2009, and 2018, respectively. From 2011 to 2022, she was a Researcher with Guilin University of Electronic and Technology. Since 2023, she has been an Associate Professor with China Jiliang University. Her research interests include filter bank design, graph signal processing, and signal recognition.

XIAOFEI ZHU received the Ph.D. degree from Xi'an Research Institute of High Technology, Xi'an, China, in 2011. Since 2015, he has been an Associate Professor with Xi'an Research Institute of High Technology. His main research interests include hardware design and radar signal processing.

WENCHAO ZHAI received the Ph.D. degree from Xidian University, Xi'an, China, in 2017. He is currently with the College of Information Engineering, China Jiliang University, Hangzhou, China. His main research interests include wireless communications, communication signal processing, signal detection, and joint radar and communications.

DAYING QUAN (Member, IEEE) received the B.S. degree in automatic control and the M.S. degree in information and communication engineering from Xidian University, Xi'an, China, in 2001 and 2004, respectively. He was with Hangzhou Silan Microelectronics Company Ltd., from 2004 to 2008. He was with Eastern Communications Company Ltd., and had been working on professional communication systems, from 2009 to 2014. In 2014, he joined China Jiliang University. He is currently working on wireless signal processing and intelligent wireless sensing.