IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 11, 2020, accepted April 2, 2020, date of publication April 6, 2020, date of current version April 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2986027

VGM-RNN: HRRP Sequence Extrapolation and Recognition Based on a Novel Optimized RNN

YIFAN ZHANG[1], FEI XIAO[1], FENGCHEN QIAN[1], AND XIANG LI[2]
[1]College of Information and Communication, National University of Defense Technology, Xi'an 710106, China
[2]College of Electronic Science, National University of Defense Technology, Changsha 410072, China

Corresponding author: Fengchen Qian (fchen_qian@163.com)

ABSTRACT High-Resolution Range Profile (HRRP) sequence has attracted academic attentions in the field of radar automatic target recognition (RATR) owing to its abundant spatial-temporal correlation between adjacent samples. However, it is difficult in the working state of radar to obtain complete HRRP sequence samples due to various internal and external factors such as ground clutter and systematic error, which poses an enormous challenge to radar target recognition. Therefore, it is crucial to repair the missing HRRPs based on the adjacent samples in the previous frames. In this paper, we discuss the extrapolate method of incomplete samples and propose an improved neural network algorithm named as Vanishing Gradient Mitigation Recurrent Neural Network (VGM-RNN). The lost samples in the sequence can be extrapolated by VGM-RNN, and the problem of vanishing gradient which is possessed in classical RNN can be effectively mitigated. The proposed method in this paper can be divided into two parts, as sample extrapolation and sequence recognition, in which sample extrapolation is the core method. Experimental results on Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset show that our proposed model exhibits higher accuracy and efficiency, as well as excellent anti-noise performance, compared with traditional methods. It is suggested that our proposed model can be effectively applied to radar system.

INDEX TERMS Automatic target recognition, recurrent neural network, high-resolution range profile sequence, vanishing gradient, sample extrapolation, sequence recognition.

I. INTRODUCTION

High resolution radar is widely used in more and more fields, especially in meteorological forecast [1], environmental monitoring [2], forest resources detection [3] and geological survey [4]. There are three kinds of data used for target recognition: one-dimensional High-Resolution Range Profile (HRRP) [5], [6], two-dimensional synthetic aperture radar (SAR) [7] and inverse synthetic aperture radar (ISAR) images [8], among which HRRP can effectively reflect energy and structure information of multiple scatters for a distributed target along the slant range direction with respect to a certain radar line-of-sight (RLOS) [9]. Furthermore, compared with SAR and ISAR image, HRRP has a lower requirement on the RLOS change, lower system complexity and higher computational efficiency [6]. Unfortunately, structure information of the targets contained in a single HRRP sample is limited. However, considering the rich spatial-temporal correlation between adjacent frames contained in sequential HRRP, it is utilized to improve the performance of radar target recognition [10]–[12].

In practical application of radar detection, as a result of equipment breakdown, artificial error and other mistakes, partial sequence samples will be inevitably lost. Besides, due to the motion interference of water waves, ground clutter and other disturbing environments [13], radar signals will be corrupted by noise, which will lead to the submergence of the separable information of targets. This is a bottleneck in the field of radar target recognition.

In order to solve this problem, the academic methods in recent years can be roughly summarized into three types. In the first class [14], the missing samples and samples with poor quality are directly be removed and deleted. Although simple and operable, this method will lead to the loss of some important information contained in the deleted samples. In the second class, the statistical filling methods are introduced to repair incomplete samples, such as mean filling, median value filling and so on [15]. Although the estimated

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

value is filled by these methods, the correlation information between sequence samples are lacked. In the third class, machine learning method [16]–[18] is applied in sample repairing. The common methods are Hidden Markov Model (HMM), Recurrent Temporal Restricted Boltzmann Machine (RTRBM) and RNN. The estimated values filled by these methods contain the sequence information of samples. Long Short Term Memory Network (LSTM), an improved model of RNN, is the most popular RNN network at present, which has been widely used in the fields of Natural Language Processing (NLP) and speech recognition [19], [20]. Considering their high computational complexity and insufficient processing capacity for high dimensional and long sequences, these methods are not selected to complete the HRRP sequence recognition task in this paper. As the most effective deep neural network in the above methods, RNN is widely utilized in sequence recognition, and the idea of back propagation is induced to optimize the network [21]. To be specific, the optimization task of RNN is to find the best weight matrix and minimize or maximize the value of loss function. To seeking extreme value, the gradient descent method is more suitable mathematically. In calculating process, the value of gradient update is usually less than 1. With the increase of RNN layers, the gradient update value will tends to zero. This phenomenon is known as gradient vanishment problem of RNN [22].

To solving the gradient vanishment problem of classical RNN, a novel optimization model for RNN is proposed in this paper, named Vanishing Gradient Mitigation RNN (VGM-RNN). The network mainly adopts three strategies to alleviate the disappearance of gradient: activation function replacement, gradient lifting, and Batch Normalization (BN). The concrete measures will be described in Section III. Furthermore, the recognition task of HRRP sequence can be divided into two steps: sample extrapolation and sequence recognition. In detail, we first extrapolate the missing samples and low-quality samples in the data set according to the sequence information to form a complete sample set, then the completed sample set will be used as training set to train the optimal parameter set, finally, the incomplete sequences collected in practice are used to evaluating the recognition performance of VGM-RNN.

The main contributions in this paper can be summarized as follows. Firstly, we creatively introduce the improved RNN to the incomplete HRRP sequence recognition task, which is more effective than traditional methods based on statistics method. Secondly, we extrapolate the missing data of the training set according to its spatiotemporal correlation before the recognition step, which outperforms traditional methods. In addition, we creatively put forward the strategies to mitigate the vanishing gradient problem, to be specific, we improve the ELU function and BN method based on the HRRP sequence recognition task, and creatively put forward the gradient lifting algorithm. Finally, combining the characteristics of HRRP sequence recognition, we derive the parameter updating formula of VGM-RNN and summarize its algorithm pseudo code. Targeted experimental results verified that the proposed method has a great contribution to mitigate the gradient vanishment, improve the recognition accuracy and enhance the robustness to noise.

The remained paper is organized as follows. In section II the background work of classical RNN and vanishing gradient problem are reviewed. In section III the architecture of the proposed VGM-RNN method is presented and the procedures of the HRRP sequence recognition based on VGM-RNN are shown. In section IV analysis and discussions of the experiments for different purposes are performed. Finally, conclusions and future works are drawn in Section V.

II. PRILIMINARIES

In this section, we first review the structure and mathematical principle of classical RNN. Then we analyze the mathematical reasons of vanishing gradient problem in sequential information processing, which is the basis of Section III.

A. RECURRENT NEURAL NETWORKS

The RNN is an essential member of the neural network family, which is used to process sequence data and has been widely used in the field of NLP. Similarly, HRRP sequences can also be processed by RNN due to high sequence structure similarity.

The structure of traditional neural networks, such as Deep Belief Network (DBN) [23], and RNN are shown in Figure 1. It is shown that the structure of RNN is similar to that of DBM, which is composed of input layer, several hidden layers and output layer [24]. The difference between them lies in the way to connect the multiple hidden layers. Specifically, the hidden layer of DBN is connected vertically while that of RNN is connected horizontally. Noting that the input of the hidden layer of RNN includes two parts: the input data and the last hidden layer, which means that the network will remember previous information and apply it to current calculation of $\hat{y}^{(t)}$.

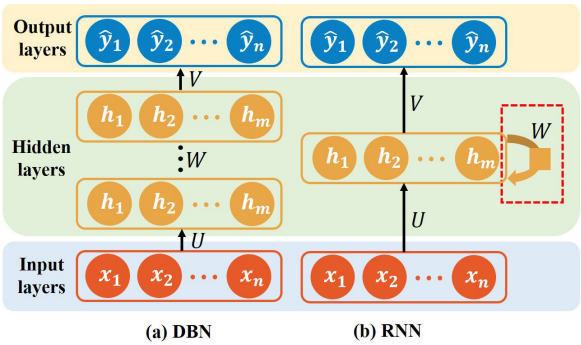

FIGURE 1. Graphical structure of DBN and RNN. Noting that the hidden layer of DBN is connected vertically while that of RNN is connected horizontally. Indicating that RNN can reflect the sequential information of input sample, while DBN cannot.

Figure 1 shows the folded structure of RNN and its unfolded structure is shown more detail in Figure 2. In the expanded RNN, we can see that the structure

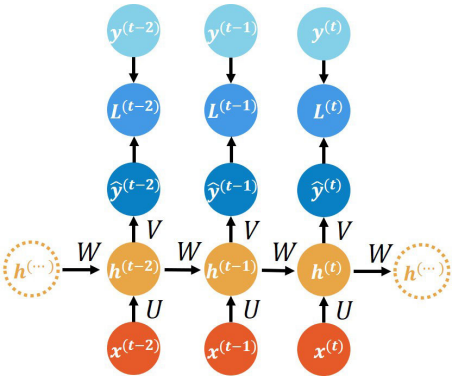

FIGURE 2. Graphical structure of expanded RNN. Noting that the circle in the figure is equivalent to the layer in the corresponding position in Figure 1(b).

contains the input unit which represented as $x = \{x^{(1)}, x^{(2)}, \cdots, x^{(t)}\}$, the hidden layer unit which expressed as $h = \{h^{(1)}, h^{(2)}, \cdots, h^{(t)}\}$, and the output layer unit represented as $\hat{y} = \{\hat{y}^{(1)}, \hat{y}^{(2)}, \cdots, \hat{y}^{(t)}\}$. Noting that there is a one-way flow of information throughout all the hidden nodes in the unfold structure, which represents the transfer of sequence correlation information in the model. In addition, the t in the upper right corner of the parameters represents the state of the t *th* time step. V, W and U are weight matrices, it is worth noting that the weights are shared, that is, all the values in W are equal, and so are V and U. Therefore, according to forward propagation algorithm, the value of hidden layer at time step t can be expressed as:

$$h^{(t)} = f\left(Ux^{(t)} + Wh^{(t-1)} + b\right) \tag{1}$$

where $f(\cdot)$ is the activation function, which is generally the sigmoid function. And b represents the bias value. Therefore, the final output of t *th* time step can be expressed as:

$$\hat{y}^{(t)} = \sigma\left(Vh^{(t)} + c\right) \tag{2}$$

where c is the bias value of *tth* output layer, and $\sigma(\cdot)$ is the tanh function in RNN.

Additionally, we also show the loss function $L^{(t)}$ and the training sample label set $y^{(t)}$ in the figure, which represents that the goal of RNN training process is to minimize the value of loss function, and the loss function represents the similarity between the output of RNN and the real label.

B. VANISHING GRADIENT PROBLEM

The purpose of training RNN is to get the optimal parameter set, while back-propagation through time (BPTT) algorithm is the most commonly used method for training RNN [25], and its core idea is based on gradient descent method. Therefore, calculating the gradient to each parameter is the key to BPTT algorithm. Concretely, BPTT algorithm adjusts the parameters in the direction of negative gradient of loss function, and the parameters are updated by:

$$\omega \leftarrow \omega + \Delta\omega \tag{3}$$

where ω represent the parameters to be updated. According to (1) and (2) that the gradient calculation of matrix W and U involves historical data. In detail, the partial derivative of loss function to W or U at t *th* time step needs to be traced back to all the information before t *th* time step. The general expression of the partial derivative of Loss function to W and U at time step t can be expressed as:

$$\frac{\partial L^{(t)}}{\partial W} = \sum_{k=0}^{t} \frac{\partial L^{(t)}}{\partial o^{(t)}} \cdot \frac{\partial o^{(t)}}{\partial h^{(t)}} \left(\prod_{j=k+1}^{t} \frac{\partial h^{(j)}}{\partial h^{(j-1)}}\right) \frac{\partial h^{(k)}}{\partial W} \tag{4}$$

$$\frac{\partial L^{(t)}}{\partial U} = \sum_{k=0}^{t} \frac{\partial L^{(t)}}{\partial o^{(t)}} \cdot \frac{\partial o^{(t)}}{\partial h^{(t)}} \left(\prod_{j=k+1}^{t} \frac{\partial h^{(j)}}{\partial h^{(j-1)}}\right) \frac{\partial h^{(k)}}{\partial U} \tag{5}$$

As shown in the above two equations, activation functions are nested in the loss function. If we put sigmoid in and take out the part of the middle multiplication, it can be shown as:

$$\prod_{j=k+1}^{t} \frac{\partial h^{(j)}}{\partial h^{(j-1)}} = \prod_{j=k+1}^{t} sigmoid' \cdot W_S \tag{6}$$

It is worth noting that the derivative of the activation function is nested in the above equations and appears in the form of cumulative multiplication. Unfortunately, as shown in Figure 3, the derivative value range of the commonly used sigmoid function is $(0, 0.25]$. Subsequently, multiple numbers smaller than 1 will be multiplied, causing that the gradient update value tend to zero, which will make the deep RNN network become a simple mapping that without parameters update and become meaningless.

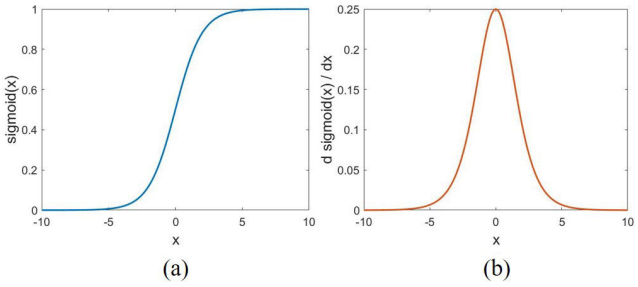

FIGURE 3. The graph of sigmoid and its derivative, which is the most commonly used activation function. Noting that the value range of (b) is limited.

The phenomenon described above is the problem of vanishing gradient, which will reduce the convergence step-size and ultimately increase the training time of the model or even make the model unable to converge to the optimal state [22]. Furthermore, the vanishing gradient problem will lead to the failure of RNN to effectively extrapolate the ideal missing HRRP samples, that is, there will be obvious errors between the estimated value and the real value of the missing HRRPs, which have a negative impact on the performance of HRRP sequence recognition.

III. THE PROPOSED METHOD

In this part, we introduce the detail strategies to mitigate the problem of vanishing gradient which we proposed, then show the two steps of our proposed method for sequence recognition, which are sample extrapolation and sequence recognition.

A. PROPOSED STRATEGIES TO MITIGATE THE VANISHING GRADIENT PROBLEM

In order to mitigate the problem of vanishing gradient, we integrate three targeted strategies to optimize classical RNN according to the characteristics of HRRP sequence recognition.

The first strategy we proposed is to optimize the activation function. Considering the value of gradient update tends to zero with the increase of RNN layers, which is the main cause of the vanishment of gradients, the activation function with better derivative performance should be selected to replace the traditional sigmoid function. Inspired by [26], the commonly used alternative activation functions are Rectified Linear Unit (ReLU), leakReLU and Exponential Linear Unit (ELU), the graph of the functions are as shown in Figure 4.

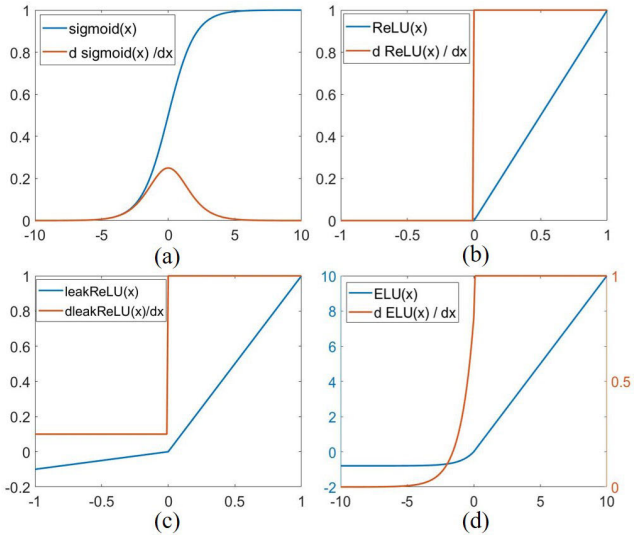

FIGURE 4. Graphs of alternative activation functions and their derivative functions. (a) sigmoid(x); (b) ReLU(x); (c) leakReLU(x); (d) ELU(x).

Figure (a) shows the sigmoid function, in which we can see that the derivative value range is limited, which is (0, 0.25]. In the deep network, the gradient will be vanished. The ReLU function in Figure (b) slows down the trend of vanishing gradient and increases the step-size of gradient updating, however, some neurons cannot be activated because the negative part of its derivative is always zero. The leakReLU function in Figure (c) solves the problems in ReLU, so that all neurons can be activated. In order to further improve the performance of the activation function, ELU is proposed in [26], which is similar to leakReLU. The difference is that the negative part of the derivative of ELU function is nonlinear and smooth, which will make the proposed model more robust to input changes or noise and achieve better optimization effect. Therefore, the method proposed in this paper utilizes the ELU function as the activation function to calculate the hidden layer node h, and the expression of ELU is as follows.

$$ELU\ (x) = \begin{cases} \alpha\ (e^x - 1)\,, & x \le 0 \\ x, & x > 0 \end{cases} \tag{7}$$

The vanishing gradient problem is mitigated to some extent by replacing the activation function mentioned above, but there is still a dilemma that the gradient update value is too small in theory, Therefore, more strategies are needed to be integrated to improve the performance of our method.

The second strategy we propose is gradient lifting, which corresponds to gradient clipping [27], which is proposed to mitigate the problem of exploding gradient. More detailed, the core idea of gradient lifting is to set a gradient threshold. With the increase of RNN layers, if the gradient $\nabla\omega$ exceeds the preset threshold value during the update process, then it will be lifted to a reasonable range by specific calculations. The algorithm pseudo code is shown in Table 1.

TABLE 1. The pseudo code of gradient lifting.

Algorithm 1. Pseudo code of gradient lifting
$\nabla\omega \leftarrow \frac{\partial L}{\partial \omega}$
if $\|\nabla\omega\| \le threshold$ then
$\nabla\omega \leftarrow \frac{\|\nabla\omega\|}{threshold}\nabla\omega$
end if

To further mitigate the problem of vanishing gradient, and to accelerate the convergence process, the third strategy proposed is Batch Normalization (BN) [28]. Specifically, BN map the input value distribution of each layer of neurons to the standard normal distribution with the mean value of 0 and the variance of 1. Therefore, a minor alteration in input data will lead to a sharp change in loss function and a larger gradient, so as to mitigate the vanishing gradient problem. Moreover, a larger gradient means a faster learning convergence speed and a faster training speed.

Three strategies (ELU + lifting + BN) are integrated from the algorithm level and code level to mitigate the problem of vanishing gradient in HRRP sequence recognition task, which provide a novel method in theory and programming level.

B. THE FLOW CHART OF VGM-RNN FOR SEQUENTIAL HRRP RECOGNITION

We synthesize the above three strategies to propose the VGM-RNN, and divide the procedures of radar HRRP sequence recognition based on VGM-RNN into two steps, which are sample extrapolation and sequence recognition. This section will introduce the detail process and the flow chart of the method proposed in this paper.

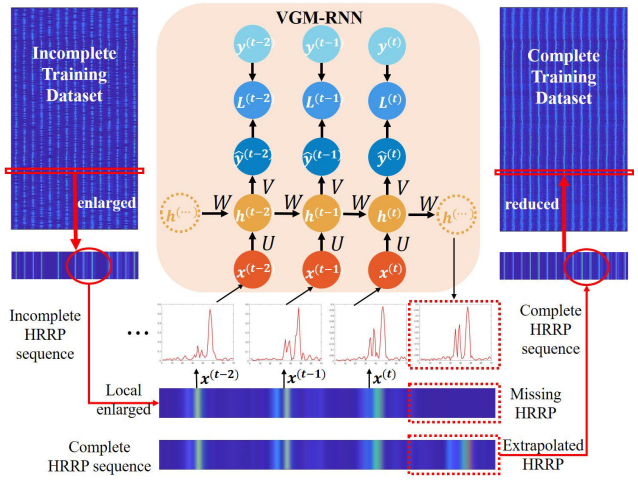

FIGURE 5. The diagram of sample extrapolation, which is the first step of the proposed model for sequential HRRP recognition.

In the step of samples extrapolation, the training sequences with missing HRRPs are chosen as input vectors. As shown in Figure 5, take a sample sequence circled in the red box on the left of the figure as an example, we decompose it into 15 independent HRRP samples (Compared with [17], we set the same sequence length.), and then input them into successive time steps in VGM-RNN. We record the missing samples as x^{t+1}, and multiple samples before x^{t+1} in the sequence as $x^{1:t}$. Hence, the state of the hidden layer node at each time can be calculated according to (1). Subsequently, missing samples can be extrapolated by VGM-RNN according to the spatial-temporal correlation between adjacent sample, and as shown in the red dotted box in the middle of the figure. Then, the complete sample set will be obtained after completing all the extrapolations of missing samples. The missing sample sequence will be replaced by extrapolated HRRPs and the new complete sequence sample set with the complete sequences will be recombined, which is shown in the right of the figure.

The sequence recognition step can be divided into two parts: training and testing. The purpose of training is to get the optimal parameter set $\omega = \{W, U, V\}$, which will be used in the test phase. In training step, the complete sequence sample set which extrapolated from the sample extrapolation step is utilized as the input vector, and the corresponding hidden layer node status at each time step can be calculated by (1). Then, the output of the model can be gained according to (2). Finally, the loss function is used to measure the difference between the model recognition result and the real label, and the value of loss function guides the updating of model parameters, that is $\omega = \{W, U, V\}$. In the proposed method, the loss function is defined as the cross entropy [29] between the output $\hat{y}(x_i)$ and label $y(x_i)$ of the sample, which is shown as follows.

$$Loss_{Cro} = -\sum_{i=1}^{n} y(x_i) \log\left(\hat{y}(x_i)\right) \tag{8}$$

where $y(x_i)$ and $\hat{y}(x_i)$ represent the probability that sample x belongs to the i *th* type of target. The difference is that $y(x_i)$ represents the sample label and the value range is 0, 1}, while $\hat{y}(x_i)$ represents the recognition result of the model and the value range is [0, 1]. Additionally, n represents the total number of sample types. Through the consecutive iteration of parameters, the value of loss function is minimized and the optimal parameter set is obtained.

In test step, in order to evaluate the robustness of our method to missing samples and the performance in practical applications, the extrapolation is not done for testing samples, and zero vectors are utilized to replace the missing samples. Testing set is decomposed into sequences of the same length as the training set, and to be input into the trained model. Then, the input vector and the optimal parameter set obtained by training step are calculated according to (1) and (2) to get the corresponding classified categories that we expect.

Therefore, the pseudo code of the proposed method can be summarized as:

IV. EXPERIMRNTS

To evaluate the proposed method, two groups of experiments with different purposes were performed on MSTAR dataset [30]. The first group aims to explore the parameters of VGM-GNN, the size of hidden layer and learning rate, to achieve the best extrapolation performance, while the purpose of the second group is to evaluate the performance of VGM-GNN in HRRP sequence recognition task, and to explore the anti-noise performance of the proposed method. All the experiments in this paper are implemented on two pieces of NVIDA RTX 2080Ti GPUs. More detailed, when the learning rate is set to 0.01, batch size is set to 128, the number of hidden layer nodes is set to 256, and set the max epoch to 50, the training time for three types of training data is about 13 hours.

This section will be divided into three subsections, the first subsection introduces the composition of MSTAR data set and the transformation of the data set for the experiment; the second subsection evaluates the impact of the proposed strategies in VGN-RNN on missing HRRP extrapolation; the third section explores the optimum model parameters to make the best extrapolation performance, and compares the HRRP sequence recognition and anti-noise performance of the proposed method with traditional methods.

A. DATASET

In order to make a better comparison with the traditional methods, Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset, which is widely used in the field of radar automatic target recognition tasks is chosen to provide the data needed in the experiments in this paper. However, the form of the data is changed according to the requirements of the experiments in this paper. This subsection mainly introduces the composition of the data set and its characteristics.

MSTAR data set is the standard data set for SAR target recognition. The data set is collected from synthetic aperture radar with resolution of $0.3m \times 0.3m$. The radar works in X

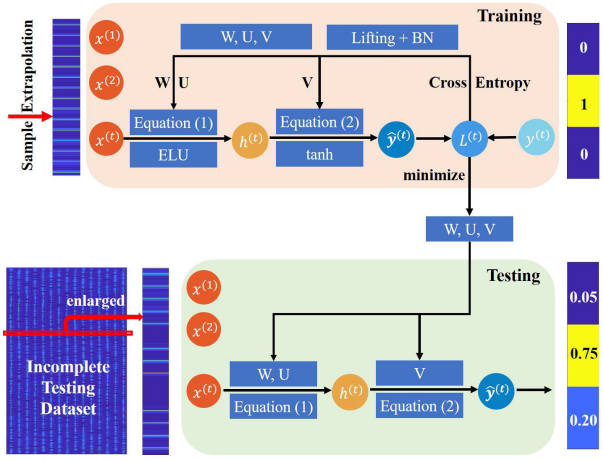

FIGURE 6. The diagram of sequence recognition, which is the second step of the proposed model for sequential HRRP recognition.

band and adopts HH polarization mode. MSTAR data set contains 10 kinds of static ground targets. In this paper, we use the widely used subset, which contains three kinds of targets with high similarity. In order to compare the recognition performance between this paper and other state-of-the-art literatures more conveniently, we convert two-dimensional SAR samples into one-dimensional HRRP sequence samples with the method consistent with literature [17]. The optical image, SAR image and single HRRP image under an azimuth of the three kinds of targets are shown in Figure 7.

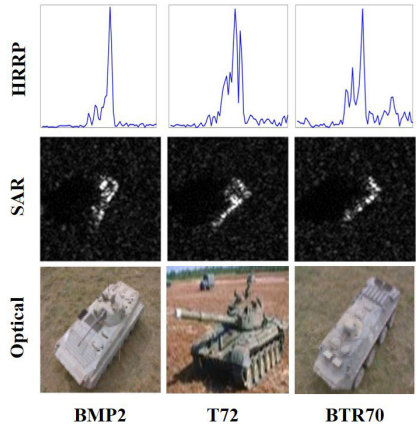

FIGURE 7. MSTAR dataset, the optical image, SAR image and single HRRP image under an azimuth of the three kinds of targets.

The three targets of the MSTAR data sets contain a standard training set and a test set, the sample type, model, quantity and other information are shown in Table 3.

It can be seen from the table that the training set and the test set contain three types of targets respectively, but it is worth noting that the depression angle of the training sample set is 17 degrees while that of the test sample set is 15 degrees. Besides, each sample in the training set contains only one model, but the BMP2 and T72 in the test set contain three models respectively, which is conducive to the detection of the generalization performance of the model, and which is very valuable for practical radar application scenarios.

TABLE 2. The pseudo code of the proposed method (VGM-RNN) for incomplete HRRP sequence recognition.

Algorithm 2. Pseudo code of the proposed VGM-RNN

 # Procedure 1: sample extrapolation

Input: incomplete training dataset: x_train_inc;
label matrix: y_train; hidden layer size: dim_h;
learning rate: λ; momentum: β; weight cost: η.

(1). Calculate $h^{(t)}$ according to Equation (1);
(2). Calculate $y^{(t)}$ according to Equation (2);
(3). Calculate $Loss_{Cro}$ according to Equation (8);
(4). Update the parameters: $\omega \leftarrow \omega + \Delta\omega$
where $\Delta\omega \leftarrow \beta\Delta\omega + \lambda(\nabla\omega - \eta\omega)$;
(5). Calculate $x^{(t+1)}$ according to Equation (1);
(6). Repeat step (1) to (5) on each training HRRP sequence.

Output: complete training dataset: x_train_com

 # Procedure 2: sequence recognition

Phase 2.1: training

Input: complete training dataset: x_train_com;
label matrix: y_train; hidden layer size: dim_h;
learning rate: λ; momentum: β; weight cost: η.

(7). Repeat step (1) to (4) for 50 epochs;
(8). Save the optimal $\omega = \{W, U, V\}$ for testing phase.

Output: optimal $\omega = \{W, U, V\}$

Phase 2.2: testing

Input: incomplete testing dataset: x_test_inc;
label matrix: y_test; hidden layer size: dim_h;

(9). Calculate $h^{(t)}$ according to Equation (1) with optimal W and U;
(10). Calculate $\hat{y}^{(t)}$ according to Equation (2) with optimal V;
(11). Output the type with the largest probability as the identification label: $Label = \max(y\_label)$

Output: The recognition label vector.

TABLE 3. Training and testing set of HRRPs for three targets.

Num.	Training Set	Size	Testing Set	Size
1	BMP2(Sn_C9563)	2330	BMP2(Sn_C9563)	1950
			BMP2(Sn_C9566)	1960
			BMP2(Sn_C21)	1960
2	T72(Sn_132)	2320	T72(Sn_132)	1960
			T72(Sn_812)	1950
			T72(Sn_S7)	1910
3	BTR70(Sn_C71)	2330	BTR70(Sn_C71)	1960
Sum	Training Set	6980	Testing Set	13650

We use the same method as [17] to convert SAR image into HRRP sequence with length $T = 15$. Take the first three HRRP samples as an example, the HRRP sequence of three types of targets are shown in Figure 8, which shows that the HRRP sequences of the three kinds of targets are highly similar, but their maximum amplitudes and the range cell index of the maximum echo intensity are slightly different.

In order to simulate the real working condition of radar, we randomly delete 20% of the samples (3 samples) in each sequence of the training set and the test set to simulate the data set with missing samples. Taking a sequence sample of BMP2 (SN_C9563) as an example, the sequential HRRPs after deleting 20% samples are shown in Figure 9.

The purpose of this section is to test the extrapolation performance for missing samples and to evaluate the recognition

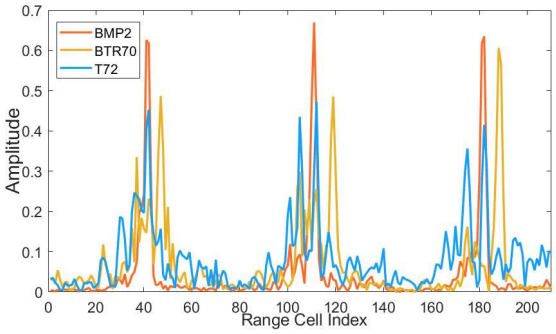

FIGURE 8. The first three samples in the HRRP sequence of three types of targets. Noting the slight difference of the maximum amplitudes and the corresponding range index cell.

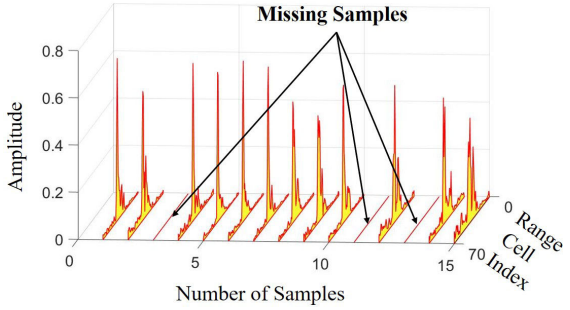

FIGURE 9. Example of HRRP sequence (BMP2 (SN_C9563)) with 20% missing samples, which is used to simulate the radar working condition.

performance on the test set with missing HRRPs of the proposed method.

B. EXPERIMENT 1: EVALUATING THE IMPACT OF PROPOSED STRATEGIES IN VGN-RNN ON MISSING HRRP EXTRAPOLATION

In this section, we will conduct experiments to verify the effectiveness of the extrapolation method of our model, which is measured by MSE, PSNR, and SSIM. Additionally, several latest algorithms such as RTRBM and RNN was taken as the baseline experimental groups to verify the effectiveness of the proposed method.

Before conducting the experiments, we analyze three indicators that can represent the extrapolation quality. Reconstruction error, the distance between the real samples and the extrapolated samples, is a significant index to measure the extrapolation performance of the model. Mean Square Error (MSE), Peak Signal to Noise Ratio (RSNR) and Structural Similarity Index (SSIM) are utilized to measure the reconstruction error [31], which is selected according to the characteristics of HRRP sequence recognition.

MSE represents the mean value, which is the error sum of squares, and which denotes the corresponding range cells of the estimated HRRP and the real HRRP sample. And smaller MSE corresponds to smaller estimation error. The MSE is calculated as follows.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - x_i'\right)^2 \tag{9}$$

where x_i and x_i' correspond to the values of real HRRP and extrapolated HRRP respectively, while N represents the total number of samples.

$$PSNR = 10 \times log_{10}\left(\frac{(2^n-1)^2}{MSE}\right) \tag{10}$$

The value of PSNR is related to SNR and is the logarithm of the mean square error of the real sample and the extrapolated HRRP sample, which is measured in dB. In the HRRP sequence recognition scenario, the smaller the sample estimation error is, the larger the PSNR value is.

Additionally, SSIM, structural similarity index, is a measure of similarity between two images. When x and y are utilized to represent the extrapolated HRRP and real HRRP sample respectively, SSIM can be calculated by:

$$SSIM = \frac{\left(2\mu_x\mu_y + c_1\right)\left(2\mu_{xy} + c_2\right)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{11}$$

where μ and σ denotes the mean value and standard deviation respectively, while c_1 (c_2) is a constant. It is worth noting that the range of structural similarity is 0 to 1, and when the two HRRP samples are exactly same, the value of SSIM is equal to 1.

Aiming at exploring whether the strategies to the vanishing gradient problem in VGM-RNN is effective, four other commonly used methods as the contrast experiments were designed and utilized in the experiments of this subsection. The first comparative experiment, the average method, calculates the average value of HRRP samples before the missing sample and takes it as the estimated sample value, which is utilized in the training process. The second method is similar to the first one, except that the average value is replaced by the mid-value. The third and fourth group of contrast experiments adopt the deep learning method, which are Recurrent Temporal Restricted Boltzmann Machine (RTRBM) and classical RNN respectively. The number of visible layer units of the two models is set to 70 according to MSTAR data, while the number of hidden layer units is set to 128, which is consistent with the setting of [17]. In order to eliminate the random errors in the experiment, the third and fourth group of contrast experiments and the methods proposed in this paper will be repeated for five times under the same parameter settings, and the final extrapolation sample is represented by the average of five extrapolation results.

Take a random sequence in BMP2 (SN_C9563) as an example. Figure 10 shows the extrapolation results of the lost sample according to different methods. The region of local amplification in the figure describes the performance of extrapolation values based on different methods at the peak echo intensity, in which the extrapolation values of each method contain errors with the real values. Nevertheless, our method still achieves the closest extrapolation results to the real sample.

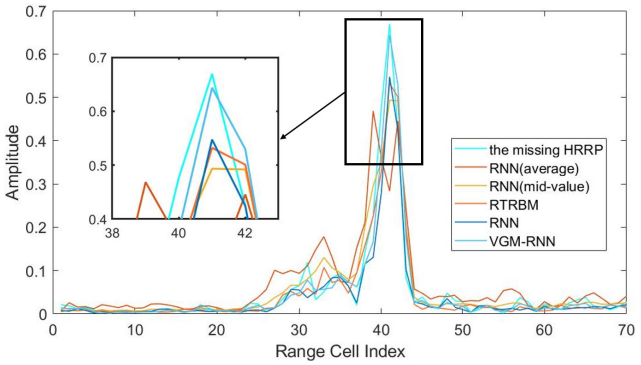

FIGURE 10. An example of the extrapolation results of the lost sample (BMP2 (SN_C9563)) according to different methods, and the detail of extrapolation performance is shown in the enlarged area in the figure.

TABLE 4. The extrapolation performance on five different methods.

Method	MSE	PSNR	SSIM
RNN (average)	0.1602	18.3113	0.6027
RNN (mid-value)	0.1306	20.3562	0.6186
RTRBM	0.1094	22.1274	0.6712
RNN	0.1055	22.4904	0.6754
VGM-RNN	**0.0939**	**23.6552**	**0.6901**

Table 4 shows the average error of five experimental methods on all training sets, and the error is measured by three indexes.

It can be seen in Table 4 that the superior recognition performance of VGM-RNN against the other four contrast methods. More detailed, our proposed model gets optimal performance on each index (the lowest value of MSE and highest value of PSNR and SSIM), which shows the strong ability to extrapolate the missing HRRP. The explanation for the result is that the proposed model effectively mitigate the vanishing gradient problem of classical RNN through the strategies of ELU + lifting + BN. Noting that the indexes of RNN and RTRBM are outperformed than other two contrast methods, which is owing to the neural networks can extract more and longer spatiotemporal correlations.

C. EXPERIMENT 2: COMPARING THE PERFORMANCE OF THE PROPOSED VGM-RNN WITH THE STATE-OF-THE-ART MODELS

In this section, we conduct experiments to evaluate the recognition performance of the proposed model with the baseline experiments. Experiments in this section consists of three experiments with different purposes. The purpose of the first one is to explore the optimal parameters in VGM-RNN, and the second experiment aims at comparing the recognition performance of the proposed method with comparative methods, additionally, the purpose of the third contrast experiment is to estimate the robustness of the model.

1) EXPLORING THE OPTIMAL SIZE OF HIDDEN LAYER AND THE VALUE OF LEARNING RATES

In Experiment 1, the number of hidden nodes in VGM-RNN is set to 128, which is consistent with [17], so as to explore

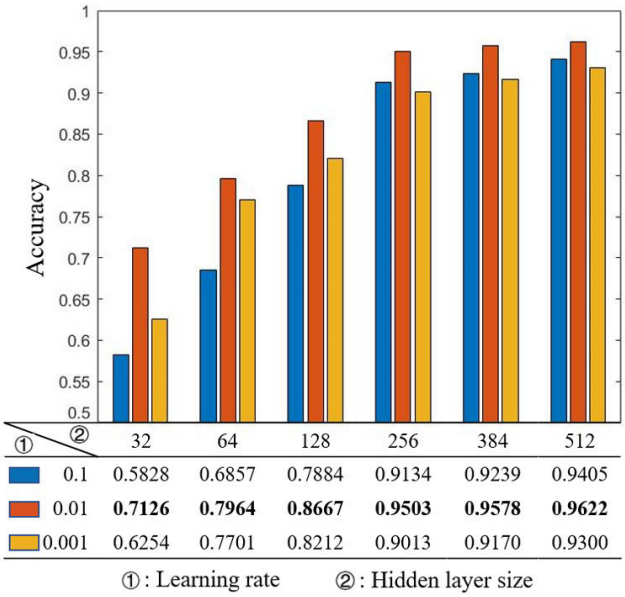

FIGURE 11. The recognition performance of VGM-RNN on the condition of different learning rate and hidden layer size. The table in the lower half of the figure is the specific quantitative description of the bar graph in the upper half.

the extrapolation performance of our proposed model under the same conditions.

In this part of the experiment, in order to explore the best recognition performance of VGM-RNN, we set different size of hidden layer and learning rate. The number of hidden nodes is set to 32, 64, 128, 256, 384 and 512 respectively, and the learning rate is set to 0.1, 0.01 and 0.001 respectively. Therefore, there are 18 different combinations in the experiment. The complete training sample set obtained from experiment 2 contains three different targets, which are used as training sequence samples. Meanwhile, the raw incomplete sample set, which contains three different types and seven different models, are used in test step. The length of the sample sequence remains 15.

We compare the recognition performance of each group under different parameter settings by comparing the recognition accuracy of each group of experiments, which can be expressed as [32]:

$$Acc = \frac{TP + TN}{TP + TN + FP + TN} \tag{12}$$

where TP means true positive, i.e. positive samples are recognized; TN means true negative, i.e. negative samples are recognized, both of which are correct cases. Correspondingly, FP and FN indicate the situation of error recognition, that is, FP indicates that positive samples are recognized as negative samples, and FN indicates that negative samples are recognized as positive samples.

The detailed recognition performance is shown in Figure 11.

It is obvious that no matter what the value of learning rate is, the recognition accuracy grows steeply and then gradually reaches to the maximum value with the increase of the hidden

layer size. It is easy to explain that he hidden layer is used to extract the deep high-dimensional features of the input data, and the more nodes in the hidden layer, the more separable features will be extracted. The extracted features are of great significance for the recognition task. Besides, the highest accuracy on 384 hidden nodes in the experiments is 95.78%, which is 1.30% higher than that of Attention-based RTRBM in [17]. However, it is worth noting that when the number of hidden layer nodes exceeds 256, the growth rate of accuracy is close to the highest level, that is to say, the number of hidden nodes is close to the saturation state, and increasing the hidden nodes has little help to improve the accuracy.

Additionally, learning rate has great impact on recognition performance. Specifically, the highest recognition rate was found in the experimental group with a learning rate of 0.01. It is easy to understand that the learning rate is related to the parameter iteration step size. The large learning rate represents the large iteration step and shows a faster iteration. However, it is difficult for the loss function to converge to the optimal value for the long iteration step, which is why the recognition rate of 0.1 learning rate is lower than the condition of 0.01. In addition, when the learning rate is very small, such as 0.001, the step length of iteration will be small, and it takes a long time to converge to the optimal value, so the learning rate of 0.001 performs worse than that of 0.01.

Based on the experimental results and the above analysis, the VGM-RNN with 256 hidden layer nodes and a learning rate of 0.01 achieves the best performance with the lowest computational complexity cost. Therefore, in the remaining experiments, the proposed model will also adopt such parameter configuration.

2) COMPARING THE PERFORMANCE OF THE PROPOSED VGM-RNN WITH THE STATE-OF-THE-ART MODELS

In order to evaluate the performance of the proposed method on the data sets with missing samples, we compare VGM-RNN with other classical and state-of-the-art deep learning methods. We set up 5 different baseline experiments, in which 3 different network models are conducted.

In the first group of contrast experiment, CRBM model was used and the parameters of the model consists with [17]. The second to fifth contrast experiments use RNN model, the differences among them lies in the estimation methods of missing samples in the sequence. Specifically, in the second contrast experiment, the missing samples are directly deleted and replaced by zero vector; then in the third control experiment, the missing samples are replaced by the average of the HRRPs before the missing HRRP; and in the fourth control experiment, the mid-value of the sequence before the missing sample is utilized to estimate the missing HRRP; in addition, the classical RNN is utilized as the fifth comparative experiment. It is worth noting that the number of hidden nodes in all experimental groups is set to 256, the learning rate is set to 0.01, and the model will automatically stop updating after iterating 50 epochs. The recognition results of VGM-RNN and the contrast experiments are described in Figure 12.

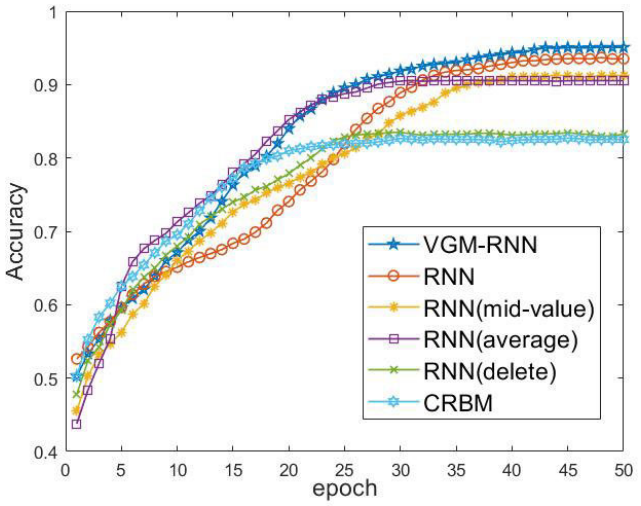

FIGURE 12. Recognition performance of VGM-RNN and the contrast experiments.

It can be seen from the figure that the recognition rates of different groups gradually increase with the iteration of parameters and converge to their highest values. However, it is worth noting that there is a big gap between the recognition accuracy of the contrast group using CRBM and RNN (delete) and that of other groups of experiments, specifically, the difference in accuracy between them was around 10% after 50 epochs. The reason for the result is that CRBM model failed to utilize the sequential information of HRRP samples, while RNN (delete) model does not estimate the missing samples, so the separable information is lost. Therefore, in the above two experiments, the separable features that can be extracted by the hidden layer of each model are limited, resulting in the accuracy far lower than that of other four groups.

In addition, the performance of other groups of experiments is similar. In order to further analyze the differences between them, we list the confusion matrix of accuracy of other four experiments on different kinds of targets in Table 5.

The confusion matrix in Table 5 accurately shows the recognition results of four methods for three types of targets. It is obvious that our proposed model achieves the best performance, which is 1.55% higher than the classical RNN, and which is primarily because the strategies we proposed mitigate the problem of vanishing gradient. Additionally, RNN with average or mid-value HRRP works poorly, mainly because the extrapolated samples based on the statistical method just contain limited spatiotemporal correlation information. More detailed, the misclassification of BMP2 lowers the average accuracy, the main possible reason is that we train the models only on BMP2(Sn_C9563), but test them on three models, Sn_C9563, Sn_C9566, and Sn_C21. Unfortunately, the similarity among the three types of BMP2 is relatively low. Conversely, the three models of T72 has a high similarity,

TABLE 5. Confusion matrix of accuracy of four methods on different types of targets for HRRP sequence recognition.

Method	RNN (average)			RNN (mid-value)			RNN (classical model)			VGM-RNN (ELU + lifting + BN)		
	BMP2	T72	BTR70	BMP2	T72	BTR70	BMP2	T72	BTR70	BMP2	T72	BTR70
BMP2	**0.8471**	0.0945	0.0584	**0.8418**	0.0872	0.0710	**0.8859**	0.0821	0.0320	**0.9105**	0.0702	0.0193
T72	0.0371	**0.9488**	0.0141	0.0347	**0.9550**	0.0103	0.0315	**0.9685**	0	0.0250	**0.9726**	0.0024
BTR70	0.0611	0.0189	**0.9200**	0.0517	0.0091	**0.9392**	0.0406	0.0082	**0.9512**	0.0302	0.0008	**0.9690**
Acc.		**0.9053**			**0.9120**			**0.9352**			**0.9507**	

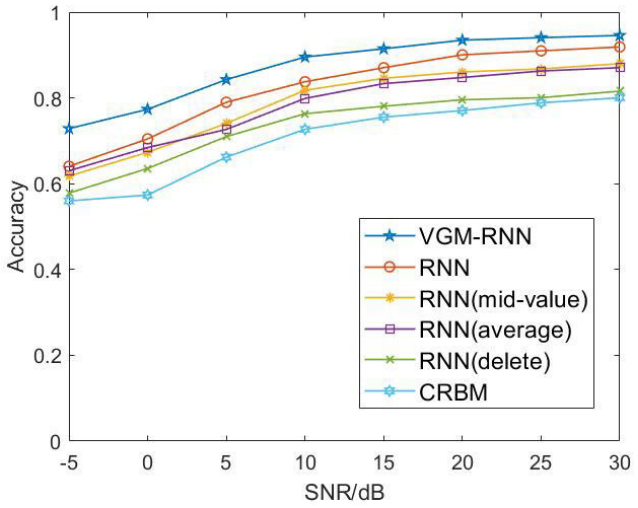

FIGURE 13. Recognition performance of the proposed method and five baseline models with different SNR.

which is the reason for the low misclassification rate of T72. Also, the probability of misclassifying BMP2 as T72 or T72 as BMP2 is much higher than that of BTR70, which is mainly because the high similarity between the sample sequences of BMP2 and T72, and the separable features between the two extracted from the input sequence is limited.

3) COMPARING THE ANTI-NOISE PERFORMANCE OF THE PROPOSED VGM-RNN WITH THE STATE-OF-THE-ART MODELS

In the real scenario of radar, the noise caused by ground clutter, electromagnetic interference, temperature and humidity will inevitably appear, so the anti-noise performance is the necessary ability for the model. Therefore, we add different intensity of Gaussian white noise to the test data to simulate the real environment. The SNR increasing from -5dB to 30dB were utilized to investigate the robustness of the proposed model. Similarly, 256 is set to be the number of hidden nodes of VGM-RNN, and the learning rate is set to 0.01. In addition, five groups of baseline experiments in the above experiment are used, and all the parameter settings remain unchanged. Figure 13 shows the recognition performance of the six models with different SNR.

It is obvious that as the decline of SNR, the performance of all recognition systems gradually reduces. Noting that our proposed method achieves better performance than other five comparative methods at all SNR levels, the recognition rate of VGM-RNN is more than 10% higher than that of RNN even at the lowest SNR (-5dB). More detailed, our proposed method shows high robustness when the SNR higher than 15dB, and the recognition accuracy is near 95% with lightly fluctuations, which is close to the average accuracy in Table 5 (95.07%), and which inflects the strong anti-noise ability of our proposed model. The reason is that our proposed strategies (ELU + lifting + BN) mitigate the problem of vanishing gradient effectively, and the model parameters were better optimized, which extract more separable spatiotemporal correlation information, so the model still perform well with low SNR. Considering the working environment of radar system, which were often corrupted by noise, our proposed method is a better choice.

V. CONCLUSION AND FUTURE WORKS

An efficient optimized RNN based HRRP sequence extrapolation algorithm was proposed and applied to incomplete HRRP sequence recognition. Compared with the reported methods, the proposed VGM-RNN has two compelling advantages. Firstly, the new step of missing sample extrapolation before the recognition step is proposed, which supplements the missing HRRP samples and complete the input information. Additionally, the strategies of ELU + lifting + BN is proposed to mitigate the problem of vanishing gradient, which greatly improves the performance of the classical RNN, and extracts more separable spatiotemporal correlation information from the input sequence.

Experiments verified the superiority of the proposed method clearly. More detailed, Experiment 1 indicates that the strategies proposed in VGM-RNN have a positive impact on the extrapolation of missing HRRP, while the Experiment 2 verifies the superior recognition performance and the strong robustness for noise of the proposed method. How to train a model to deal with HRRP sequences of different lengths is an appealing future topic. Finally, we suggest that the proposed method can be utilized for the real radar automatic target recognition tasks.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewer for their valuable comments on improving this article.

REFERENCES

[1] S. Duan, W. Yang, X. Wang, S. Mao, and Y. Zhang, "Forecasting of grain pile temperature from meteorological factors using machine learning," *IEEE Access*, vol. 7, pp. 130721–130733, 2019.

[2] A. Bhatta and A. K. Mishra, "GSM based hand-held commsense-sensor for environment monitoring," in *Proc. Int. Conf. Ind. Inf. Syst.*, Dec. 2018, pp. 360–364.
[3] R. Nelson, J. Boudreau, and T. G. Gregoire, "Estimating quebec provincial forest resources using ICESat/GLAS," *Can. J. Forest Res.*, vol. 39, no. 4, pp. 862–881, 2009.
[4] X. Luo, W. Zhou, W. Wang, Y. Zhu, and J. Deng, "Attention-based relation extraction with bidirectional gated recurrent unit and highway network in the analysis of geological data," *IEEE Access*, vol. 6, pp. 5705–5715, 2018.
[5] C. Zhao, X. He, J. Liang, T. Wang, and C. Huang, "Radar HRRP target recognition via semi-supervised multi-task deep network," *IEEE Access*, vol. 7, pp. 114788–114794, 2019.
[6] M. Pan, J. Jiang, Q. Kong, J. Shi, Q. Sheng, and T. Zhou, "Radar HRRP target recognition based on t-SNE segmentation and discriminant deep belief network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 9, pp. 1609–1613, Sep. 2017.
[7] J. H. Cho and C. G. Park, "Multiple feature aggregation using convolutional neural networks for SAR image-based automatic target recognition," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 12, pp. 1882–1886, Dec. 2018.
[8] Z. Liu, X. Wei, and X. Li, "Decoupled ISAR imaging using RSFW based on twice compressed sensing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 3195–3211, Oct. 2014.
[9] L. Du, H. Liu, and Z. Bao, "Radar HRRP statistical recognition: Parametric model and model selection," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1931–1944, May 2008.
[10] N. Su, F. Dai, H. Liu, and B. Zhang, "Three-dimensional absolute attitude reconstruction of a rigid body based on multi-station HRRP sequences," *IEEE Access*, vol. 8, pp. 27793–27806, 2020.
[11] P. Wang, L. Du, and H. Liu, "A spatiotemporal model for radar HRRP sequence recognition," in *Proc. IEEE Radar Conf.*, Washington, DC, USA, Dec. 2010, pp. 1005–1008.
[12] C. Da-qing, X. Su-dao, Z. Zhe, and W. Rong-chen, "Micro-motion analysis based on HRRP sequence," in *Proc. IET Int. Radar Conf.*, Xi'an, China, 2013, pp. 1–5.
[13] M. Farshchian, "Target extraction and imaging of maritime targets in the sea clutter spectrum using sparse separation," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 2, pp. 232–236, Feb. 2017.
[14] E. Serpedin, "Subsequence based recovery of missing samples in oversampled bandlimited signals," *IEEE Trans. Signal Process.*, vol. 48, no. 2, pp. 580–583, Feb. 2000.
[15] Y. Zhang, F. Xiao, and F. Qian, "A generative approach for incomplete high resolution range profile sequence recognition," in *Proc. IEEE ICET*, vol. 2, May 2019, pp. 557–563.
[16] L. Du, P. Wang, H. Liu, M. Pan, F. Chen, and Z. Bao, "Bayesian spatiotemporal multitask learning for radar HRRP target recognition," *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3182–3196, Jul. 2011.
[17] Y. Zhang, X. Gao, X. Peng, and X. Li, "Attention-based recurrent temporal restricted Boltzmann machine for radar high resolution range profile sequence recognition," *Sensors*, vol. 18, p. 1585, Dec. 2018.
[18] J. Liu, B. Chen, W. Chen, and Y. Yang, "Radar HRRP target recognition with target aware two-dimensional recurrent neural network," in *Proc. IEEE Int. Conf. Signal Process., Commun. Comput. (ICSPCC)*, Beijing, China, Sep. 2019, pp. 1–6.
[19] Z. Li, F. Yang, and Y. Luo, "Context embedding based on bi-LSTM in semi-supervised biomedical word sense disambiguation," *IEEE Access*, vol. 7, pp. 72928–72935, 2019.
[20] J. Tu, T. Huang, X. Liu, F. Gao, and E. Yang, "A novel HRRP target recognition method based on LSTM and HMM decision-making," in *Proc. 25th Int. Conf. Autom. Comput. (ICAC)*, Lancaster, U.K., Sep. 2019, pp. 1–6.
[21] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[22] S. Squartini, A. Hussain, and F. Piazza, "Preprocessing based solution for the vanishing gradient problem in recurrent neural networks," in *Proc. Int. Symp. Circuits Syst. (ISCAS)*, Bangkok, U.K., 2003, p. 5.
[23] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.
[24] H. Shao, "Delay-dependent stability for recurrent neural networks with time-varying delays," *IEEE Trans. Neural Netw.*, vol. 19, no. 9, pp. 1647–1651, Sep. 2008.
[25] M. Han, J. Xi, S. Xu, and F.-L. Yin, "Prediction of chaotic time series based on the recurrent predictor neural network," *IEEE Trans. Signal Process.*, vol. 52, no. 12, pp. 3409–3416, Dec. 2004.
[26] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," *Comput. Sci.*, to be published.
[27] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," Tech. Rep., 2012.
[28] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Int. Conf. Mach. Learn.*, 2015, pp. 1–5.
[29] G. M. J.-B. Chaslot, M. H. M. Winands, I. Szita, and H. J. van den Herik, "Cross-entropy for Monte-Carlo tree search," *ICGA J.*, vol. 31, no. 3, pp. 145–156, Sep. 2008.
[30] MSTAR. *Targets: T-72, BMP-2, BTR-70, SLICY*.[Online]. Available: http://www.mbvlab.wpafb.af.mil/public/MBVDATA.
[31] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. 20th Int. Conf. Pattern Recognit.*, Istanbul, Turkey, Aug. 2010, pp. 23–26.
[32] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

YIFAN ZHANG received the B.S. degree in intelligent science and technology from Xidian University, Xi'an, China, in 2016, and the M.S. degrees in communication engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2018. He is currently an Assistant with the College of Information and Communication, NUDT. His research interests include radar automatic target recognition, deep learning and data analysis.

FEI XIAO was born in Xi'an, Shanxi, China, in 1983. She received the master's and Ph.D. degrees in engineering from the National University of Defense Technology, China, in 2007 and 2012, respectively. She is currently an Associate Professor with the College of Information and Communication, National University of Defense Technology. She is also the author of more than 30 articles and four inventions. Her research interests include machine learning, data science, and data mining algorithms for spare samples.

FENGCHEN QIAN received the M.S. degree in image processing from Northwestern Polytechnical University, Xi'an, China, in 2004, and the Ph.D. degree in optics engineering from the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, China, in 2014. He is currently an Associate Professor and the Director of the Testing and Evaluation Center of Information and Communication Equipment, National University of Defense Technology. His research interests include designing, simulation, testing and evaluation of communication networks, and data processing.

XIANG LI received the B.S. degree from Xidian University, Xi'an, China, in 1989, and the M.S. and Ph.D. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 1995 and 1998, respectively. He is currently a Professor with the School of Electronic Science, NUDT. He is also a Professor and performs research on signal detection, automatic target recognition, and radar signal processing.