Modulation Recognition Based on Denoising Bidirectional Recurrent Neural Network

Ruiyan Du
Northeastern University - Qinhuangdao Campus
Fulai Liu
Northeastern University - Qinhuangdao Campus
Jialiang Xu (jialiang9102@163.com)
Northeastern University https://orcid.org/0000-0001-6453-0527
Fan Gao
Northeastern University
Zhongyi Hu
Northeastern University
Aiyi Zhang
Northeastern University

Research Article

Keywords: Modulation recognition, Denoising bidirectional recurrent neural network, Deep learning, Dilated convolution

Posted Date: April 13th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-325033/v1

Noname manuscript No.
(will be inserted by the editor)

Modulation Recognition Based on Denoising Bidirectional Recurrent Neural Network

Ruiyan Du[1,2] · **Fulai Liu**[1,2*] · **Jialiang Xu**[2*] · **Fan Gao**[2] ·
Zhongyi Hu[2] · **Aiyi Zhang**[2]

Received: date / Accepted: date

Abstract Modulation recognition is an important research area in wireless communication. It is commonly used in both military and civilian domains, such as spectrum detection and interference identification. Most existing modulation recognition algorithms have a better recognition performance at high signal noise ratio (SNR). However, when SNR decreases to -10 dB or even lower, such as the battlefield and disaster areas and other harsh environment, the recognition rate may decrease dramatically. In order to solve this problem, a modulation recognition algorithm based on denoising bidirectional recurrent neural network (DBRNN) is proposed. Firstly, the state memory ability of the signal reconstruction layer in the network is utilized to learn the temporal correlation of the modulated signal, the reconstruction of the received signal is completed and the noise in the modulated signal is suppressed. Then, the reconstructed signal is encoded and decoded by the feature reconstruction layer, in which the feature of reconstructed signal is compressed and reconstructed, thereby the influence of noise can be further reduced. Finally, the reconstructed features are identified and classified by the fully connected layer. Simulation results demonstrate that the proposed network can effectively suppress the noise in the signal. Compared with other existing algorithms, the proposed method has higher recognition accuracy in the low SNR environment.

Keywords Modulation recognition · Denoising bidirectional recurrent neural network · Deep learning · Dilated convolution

1 Introduction

Modulation recognition, as an important technology of wireless communication, is commonly used in military and civilian domains. In military applications, it can be employed for signal confirmation, spectrum detection, emitter interception and so on. In the civilian domains, its applications include signal verification, cognitive radio, interference identification, etc. There are different modulation types of signals, and their features are also different. The modulation recognition problems can be resolved by recognizing the signal features, and feature-based classification methods are popular.

In [5], several algorithms based on signal phase, frequency and amplitude feature are used for modulation recognition. Although the computational complexity of these algorithms are very low, the recognition rate is significantly influenced by noise. It may decrease dramatically in the low SNR environment. High-order statistical-based algorithms [6], [7], [10], [15], such as instantaneous features and high-order cumulants, have excellent anti-noise performance and the computational complexity is relatively low. However, these methods require manually extracted features and the performance of the algorithms are unsatisfied in the low SNR environment.

Deep learning is an effective technique to extract various complex features from the original data automatically. It has been widely used in modulation recognition because of its excellent self-learning ability and nonlinear mapping ability. It can be combined with the simple features to discover more complex features automatically by multiple nonlinear transformations. Because the neural network has the ability of nonlinear mapping, it can solve the difficult classification problem, such as modulation recognition in the low SNR environment. In [12], O'Shea applies deep learning to modulation recognition for the first time and discusses the critical importance of good datasets for model learning, testing, and evaluation. In [13], a modulation recognition algorithm based on deep neural network is proposed. The algorithm uses particle swarm optimization algorithm to optimize the number of neural network nodes,

The third author made the same contribution as the first author
[1] Engineer Optimization & Smart Antenna Institute, Northeastern University at Qinhuangdao, Qinhuangdao, China.
· [2] School of Computer Science and Engineering, Northeastern University, Shenyang, China.
[*]Corresponding author: Fulai Liu (fulailiu@126.com) Jialiang Xu (jialiang9102@126.com).

and adaptively selects the optimal number of hidden nodes in the network, which improves the recognition accuracy of modulation recognition in multipath interference environment. In [14], a deep belief network (DBN) to classify the signal modulation mode is proposed. Compared with spectral correlation function, a modulation recognition algorithm based on high order cumulant neural network is proposed in [15], which extracts the sixth order cumulant of the signal to construct a new feature parameter as a feature input of the neural network. The simulation results demonstrate that the algorithm has good recognition performance and robustness in frequency offset and multipath effect environments. In [16], a deep neural network (DNN) based on high order cumulants is employed to improve the recognition rate, and the overall success rate of the method is over 99% at the -2dB SNR. A convolutional neural networks (CNN) based on constellation diagrams is designed to recognize modulation mode that is difficult to distinguish such as 16 quadratic-amplitude modulation (QAM) and 64 QAM [17]. A modulation recognition algorithm based on convolutional neural network is proposed in [18]. The algorithm uses the constellation diagram of modulation signal and convolutional neural network to get better recognition performance. A genetic backpropagation neural network (BPNN) is investigated in [18], genetic algorithm (GA) is used to design the architecture of BPNN to find the best value for the number of hidden layers and the number of neurons in each layer. This approach eliminates the human factor and improves the efficiency and accuracy of network. However, the recognition rate of the existing deep learning methods such as [12]− [18] in the low SNR environment is lower. When SNR is less than -10 dB, the recognition rate is less than 30% and some even less than 10%. With the increasing complexity of modern communication environment, the signal is subject to more and more interference during transmission [19]. In the actual channel, the signal power will gradually fade with the transmission distance, resulting in SNR reduce to -10dB or even lower [20]. In the electronic detection, the signal received by the detector may be sent by the sidelobe of the transmitter, resulting in the received signal with a low SNR [21]. In modern electronic warfare, the communication signal has low power for communication security, and multiple types of noise are often superimposed during wireless propagation, resulting in SNR decreased significantly [22].

In order to improve the recognition rate of modulation recognition technology in the low SNR environment. A denoising bidirectional recurrent neural network is proposed in this paper. The rest of the paper is organized as follows. The data set is described in Section 2. Section 3 introduces the proposed method. Section 4 gives some simulation results. Finally, the conclusion is summarized in Section 5.

2 Data Model

Assuming that the received signal is given by

$$x(n) = s(n) + g(n) \tag{1}$$

where $x(n)$ represents the received signal, $g(n)$ denotes additive white Gaussian noise having zero mean and variance $\sigma_g^{(2)}$, and $s(n)$ is given by:

$$s(n) = ae^{i(\omega_0 nT+\theta_n)} \sum_{j=-\infty}^{j=\infty} s(l)\rho(nT + j\tau + \varepsilon_T) \tag{2}$$

where $s(l)$ represents the input sequence, a denotes the signal amplitude, ω_0 stands for angular frequency offset constant, $\rho(.)$ is channel effect, τ represents symbol spacing, ε_T stands for timing jitter and θ_n denotes phase jitter.

3 Algorithm Formulation

In this section, a new network structure called DBRNN is proposed. Figure 1 shows DBRNN network structure. The network consists of signal reconstruction layer, feature reconstruction layer and fully connected layer. Firstly, the signal reconstruction layer receives the input signal, which is processed by the signal reconstruction layer to obtain the reconstructed signal. The signal reconstruction layer can suppress the noise in the signal, thus reducing the influence of noise features on feature extraction, facilitating the feature reconstruction layer to extract the signal features accurately. Then, the feature reconstruction layer processes the reconstructed signal and generates the feature reconstruction signal. It can compress and reconstruct the features of the reconstructed signal to make the signal purer. Finally, the feature reconstruction signal is identified and classified by the fully connected layer. The specific functions of each layer will be described as follows.

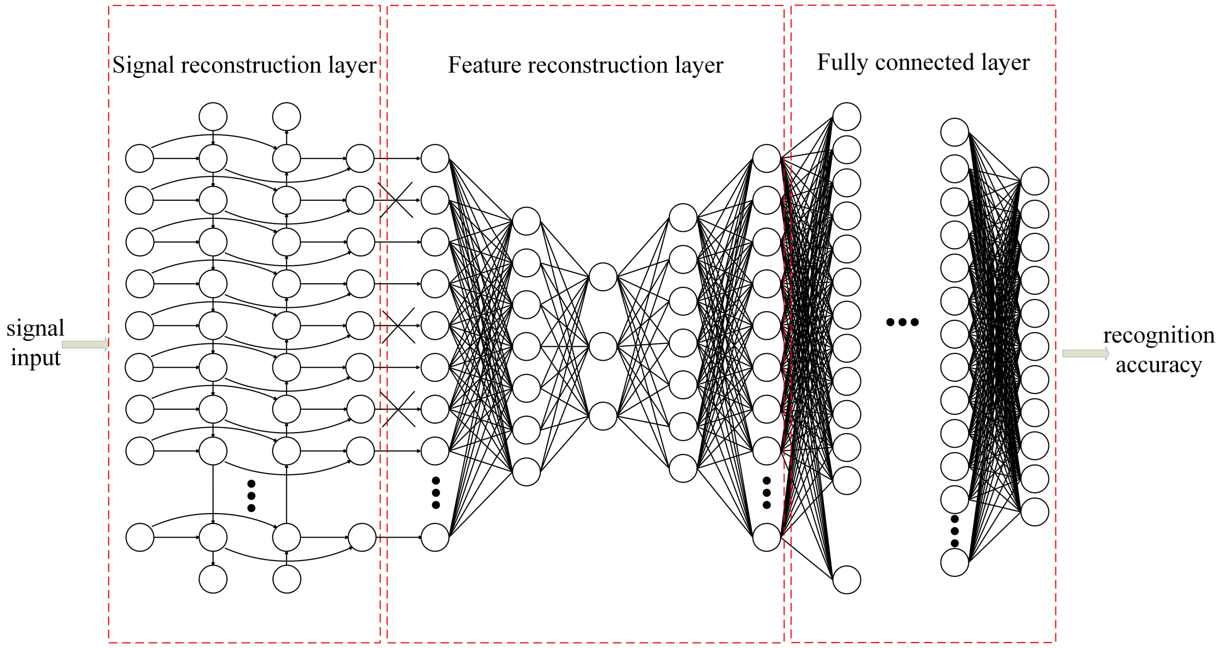

Figure 1. DBRNN network structure

3.1 Signal reconstruction layer

Firstly, the signal reconstruction layer is used to preprocess the received signal so as to reduce the interference of noise. The signal reconstruction layer is composed of a series of bidirectional cyclic neurons. Bidirectional cyclic neurons can not only receive information from other circulating neurons, but also receive their own information, thereby forming a circular network structure, which has the short-term memory function and is suitable for processing time series. The modulated signal can be thought as a time series, moreover the before and the after signal have strong temporal correlation. The temporal correlation of the received signal is learned by the signal reconstruction layer, which captures the effective information of time series and removes the invalid information, therefore the denoising of the received signal is completed. The neurons of the signal reconstruction layer are shown in Figure 2.

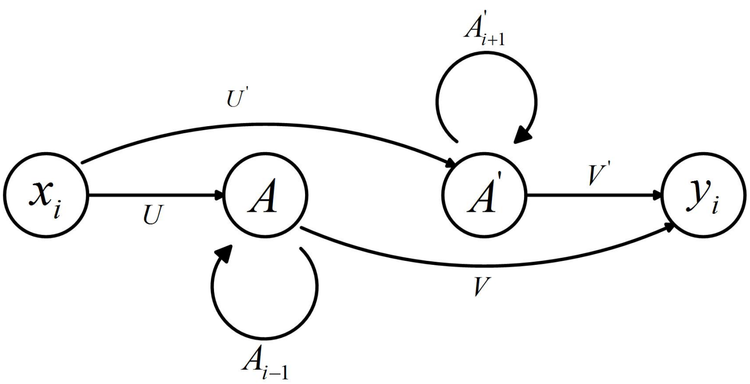

Figure 2. The neurons of the signal reconstruction layer

It can be seen from Figure 2 that the structure of the neurons in the signal reconstruction layer is different from the traditional neuron structure. On the basis of the traditional neuron structure, a delay memory unit is added, it records the state of the next neuron.

The amplitude of the received signal is used as the input of the network. Since the amplitude of the received signal varies widely in the low SNR environment, and it has a relatively discrete characteristic, the input of the network is transformed by one hot encoding, and the encoded vector $\boldsymbol{x}$ is used as the input to the network. The process of the input layer to the hidden layer can be expressed as

$$A_i = f\left(\boldsymbol{Ux} + \boldsymbol{W}A_{i-1}\right) \tag{3}$$

where $\boldsymbol{U}$ represents the weight matrix from the input layer to the hidden layer, $\boldsymbol{x}$ denotes the encoded vector, and $\boldsymbol{W}$ stands for the weight matrix between the previous state A_{i-1} and the current state A_i , which is the weight matrix of its own recursion. $f(\cdot)$ is sigmoid activation function. After the current state A_i is obtained, the output of the network is calculated, and the result is given by

$$\boldsymbol{y} = g\left(\boldsymbol{V}A_i\right) \tag{4}$$

where $\boldsymbol{V}$ denotes the weight matrix from the hidden layer to the output layer, and $g(\cdot)$ stands for tanh activation function, which can be expressed as:

$$g\left(x\right) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{5}$$

In the signal reconstruction layer, because of the temporal correlation of signal, signal is propagated not only from the input layer to the hidden layer, but also from the hidden layer to the input layer. The reverse process of network from the input layer to the hidden layer can be expressed as follow:

$$A_i{}' = f\left(\boldsymbol{U}'\boldsymbol{x} + \boldsymbol{W}'A_{i+1}\right) \tag{6}$$

where $\boldsymbol{U}'$ represents the weight matrix from the hidden layer to the input layer, and $\boldsymbol{W}'$ stands for the weight matrix between the future state A_{i+1} and the current state A_i'.

According to (2) and (5), the output vector of the signal reconstruction layer can be expressed as

$$\begin{aligned} y_i &= g\left(\boldsymbol{V}A_i + \boldsymbol{V}A_i'\right) \\ &= g\left(\boldsymbol{V}f\left(\boldsymbol{U}x_i + \boldsymbol{W}A_{i-1}\right) + \boldsymbol{V}'f\left(\boldsymbol{U}'x_i + \boldsymbol{W}'A'{}_{i+1}\right)\right) \\ &= g\begin{pmatrix} \boldsymbol{V}f\left(\boldsymbol{U}x_i + \boldsymbol{W}\left(f\left(\boldsymbol{U}x_{i-1} + \boldsymbol{W}A_{i-2}\right)\right)\right) + \dots \\ +\boldsymbol{V}'f\left(\boldsymbol{U}'x_i + \boldsymbol{W}'\left(f\left(\boldsymbol{U}'x_{i+1} + \boldsymbol{W}'A_{i+2}'\right)\right)\right) \end{pmatrix} \end{aligned} \tag{7}$$

where x_i represents the network input at ith time and y_i stands for the network output at ith time.

In the back propagation process of signal reconstruction layer, the partial derivative of parameter $(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W})$ is calculated respectively.

Assume that the derivative of hidden state is $\delta^{(t)}$ at time t, the partial derivative of the function L regard as the parameter V can be expressed as

$$\frac{\partial L}{\partial V} = \sum_{t=1}^{\tau}\left(\hat{y}^{(t)} - y^{(t)}\right)\left(A^{(t)}\right)^T \tag{8}$$

where L represents the mean square error loss function, $\hat{y}^{(t)}$ indicates the expected output of the network, and $y^{(t)}$ stands for the real output of the network.

The partial derivative of the function L regard as the parameter U can be expressed as

$$\frac{\partial L}{\partial U} = \sum_{t=1}^{\tau}\operatorname{diag}\left(1 - \left(A^{(t)}\right)^2\right)\delta^{(t)}\left(x^{(t)}\right)^T \tag{9}$$

where $x^{(t)}$ represents the input data of the network at t time and $A^{(t)}$ represents the state A_i at t time.

The partial derivative of the function L regard as the parameter W can be expressed as

$$\frac{\partial L}{\partial W} = \sum_{t=1}^{\tau}\operatorname{diag}\left(1 - \left(A^{(t)}\right)^2\right)\delta^{(t)}\left(A^{(t-1)}\right)^T \tag{10}$$

where $A^{(t-1)}$ represents the state A_{i-1} at t time.

The noise of the received signal is reduced by the signal reconstruction layer, this process provides the basis for the subsequent network layer processing.

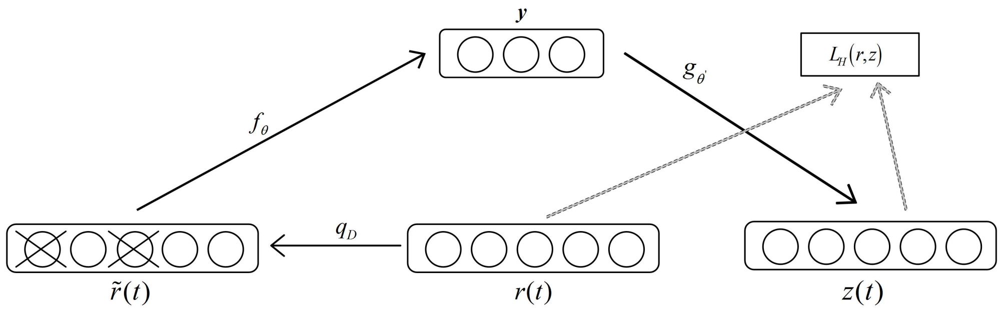

Figure 3. The structure of the feature reconstruction layer

3.2 Feature reconstruction layer

After the signal reconstruction layer, the received signal has been reconstructed. And the amplitude, frequency, phase and other effective information of the signal have emerged from the noise. However, the noise in the received signal has not been completely removed. The residual noise affects the internal feature representation of the signal, resulting in poor representativeness of the signal feature and low recognition accuracy. In order to obtain the more effective features from the reconstructed signal, the feature reconstruction layer is used to reconstruct the signal feature in this section. The structure of the feature reconstruction layer is shown in Figure 3.

First, the reconstructed signal $r(t)$ is mapped to the corrupted signal $\tilde{r}(t)$ by the random damage process q_D ,which can be expressed as

$$\tilde{r}(t) = q_D(r(t)) \tag{11}$$

The random damage process q_D randomly sets the sampling points of $r(t)$ to zero, and this process randomly removes some sampling points that may have large noise, thereby the signal $\tilde{r}(t)$ whose a part of amplitude value is 0. Then, $\tilde{r}(t)$ and $r(t)$ are combined to form the training sample pair $(r(t), \tilde{r}(t))$ as an input to the feature reconstruction layer.

As shown in Figure 1, the feature reconstruction layer encodes the reconstructed signal, so as to the features of the reconstructed signal is learned and compressed to a set of smaller feature vectors. Then through the decoding process, the feature vector is reconstructed into more representative signal features. The encoding and decoding process of the feature reconstruction layer will be described as follow.

In the process of encoding, one hot encoding is applied to $(r(t), \tilde{r}(t))$, so that it can be transformed into $\boldsymbol{x} = (x_1, x_2, x_3, \ldots x_n)$ that the value of one position is 1 and the value of the other positions value is 0. $\boldsymbol{x}$ is processed by the first nonlinear layer, and the process can be expressed as

$$\boldsymbol{h}_{\boldsymbol{k}}^{(1)} = f(\boldsymbol{W}_1 \boldsymbol{x} + \boldsymbol{b}_1) \tag{12}$$

where k represents the length of feature vector. $\boldsymbol{W}_1$ represents the weight matrix, and b_1 denotes the bias vector. $f(\cdot)$ represents sigmoid activation function. The next layer feature is a compressed representation of the previous layer feature.

After the input signal encoding, the signal feature coding $\boldsymbol{h}_k^{(1)}$ is further compressed. The process can be expressed as

$$\boldsymbol{h}_{\boldsymbol{l}}^{(2)} = f\left(\boldsymbol{W} \boldsymbol{h}_{\boldsymbol{k}}^{(1)} + \boldsymbol{b}\right) \tag{13}$$

where l represents the length of the feature vector, in order to compress the signal feature to a smaller length, let $l < k$.

According to (12) and (13), the input signal $(r(t), \tilde{r}(t))$ is compressed into the feature vector $\boldsymbol{y}$. The vector $\boldsymbol{y}$ retains the original features of the input signal as much as possible, and randomly removes the noise in the signal.

Then the feature vector $\boldsymbol{y}$ is decoded, and the decoding process of $\boldsymbol{y}$ is the reconstruction process of signal feature. By reconstructing the feature vector $\boldsymbol{y}$ multiple times, a more representative signal feature is obtained. The first reconstruction process in the decoding phase can be expressed as

$$\hat{\boldsymbol{h}}_{\boldsymbol{k}}^{(1)} = f\left(\boldsymbol{W}'\boldsymbol{y} + \boldsymbol{b}'\right) \tag{14}$$

where $\boldsymbol{W}'$ represents the weight matrix corresponding to the first decoding. $\boldsymbol{b}'$ is the bias to the first decoding. $\hat{\boldsymbol{h}}_{\boldsymbol{k}}^{(1)}$ is the first reconstructed feature vector. The process of decoding and reconstructing features after the second layer can be expressed as

$$\hat{\boldsymbol{h}}_{\boldsymbol{l}}^{(1)} = f\left(\boldsymbol{W}'\hat{\boldsymbol{h}}_{\boldsymbol{k}}^{(1)} + \boldsymbol{b}'\right) \tag{15}$$

In the process of feature reconstruction, a few original features are used to recover the noiseless features, feature vector maps from the lower dimensional space to the higher dimensional space.

Finally, the vector $\hat{\boldsymbol{x}}$ is obtained by the reconstruction process multiple times, which can be expressed as

$$\hat{\boldsymbol{x}} = f\left(\boldsymbol{W}'\hat{\boldsymbol{h}}_{\boldsymbol{l}}^{(1)} + \boldsymbol{b}'\right) \tag{16}$$

where $\hat{\boldsymbol{x}}$ represents the reconstructed feature vector obtained by the reconstructed signal encoding and decoding process.

The error function of backpropagation can be expressed as

$$L_H(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \sum_{n=1}^{N} ||\boldsymbol{x} - \hat{\boldsymbol{x}}||^2 + \eta\rho\left(\boldsymbol{h}^{(n)}\right) \tag{17}$$

where the error function adopts the mean square error loss function, η represents the penalty factor, $\rho(\cdot)$ denotes the sparse metric function, and the sparse metric function can be specifically expressed as

$$\begin{gathered} \rho\left(h^{(n)}\right) = \sum_{j=1}^{p} KL(\rho^*||\hat{\rho}_j) \\ \hat{\rho}_j = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{h}^{(n)} \end{gathered} \tag{18}$$

where ρ^* represents a constant, $\hat{\rho}_j$ is the average activity value of the jth neuron in the middle layer, and $KL(\rho^*||\hat{\rho}_j)$ denotes the KL distance between ρ^* and $\hat{\rho}_j$, which is used to measure the difference between ρ^* and $\hat{\rho}_j$.

3.3 Fully connected layer

After reconstructing the signal feature, each input signal has a new feature representation, and the reconstructed feature are identified and classified by the classifier. In this section, the 14-layer neural network will be used to identify and classify the reconstructed signal feature. The network structure is shown in Figure 4.

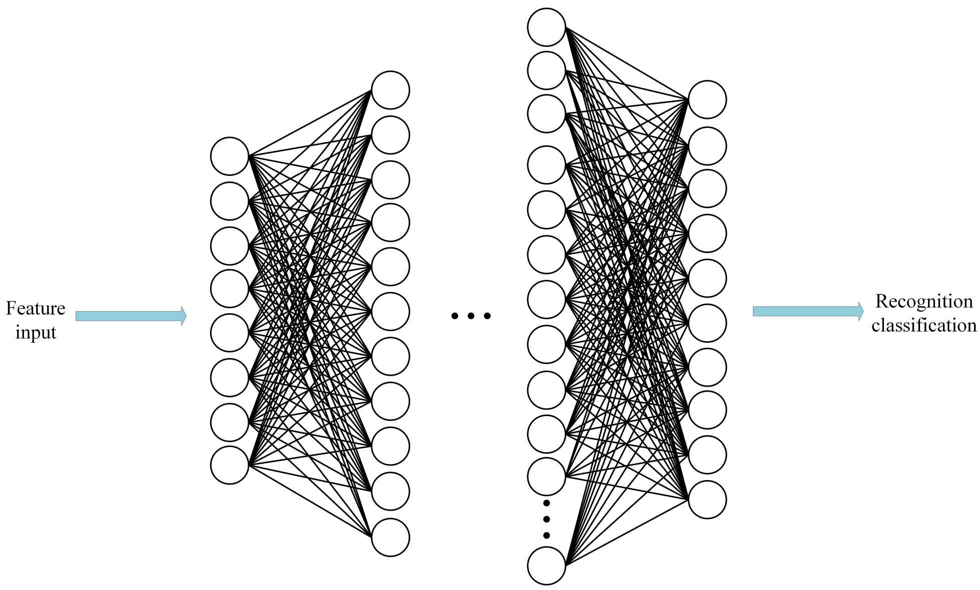

Figure 4. 14-layer neural network

The 14-layer neural network is a fully connected BP neural network. The input vector of the fully connected layer is the output vector of the feature reconstruction layer. After the feature is input to the fully connected network, the process of the fully connected network can be expressed as

$$n_j = \sum_{i=1}^{n} w_{ji} x_i + \theta_j \qquad b_j = f(n_j) \tag{19}$$

where i is input neuron serial number, j is output neuron serial number, x_i represents the reconstructed feature, w_{ji} denotes the weight of the hidden layer, θ_j is the bias of the the hidden layer, b_j represents the output feature of the hidden layer, and $f(\cdot)$ stands for the relu activation function.

In the last layer of the network, the output of the network needs to be identified, so the last layer of the network uses the softmax function. The error between the actual output and the expected output is calculated. The process can be expressed as

$$E = \frac{1}{2}\sum_{k=1}^{q} (O_k - y_k)^2 + \Omega(\omega) \tag{20}$$

where $\boldsymbol{O}_k$ represents the expected output of the kth neuron, $\boldsymbol{y}_k$ represents the actual output of the kth neuron, and $\Omega(\omega)$ denotes the regularization term.

According to the back propagation of the network, continuously optimize the network weights until the network training is completed. The received signals through the signal preprocessing, feature reconstruction and signal recognition, the classification results are obtained.

Summary of the Proposed Method

The procedure of the proposed DBRNN can be summarized as follows.

Step.1 According to (2)∼(9), the signal reconstruction layer is used to mine the temporal correlation information of the signal, the received signal is reconstructed and the noise in the signal is suppressed;

Step.2 According to (10)∼(12), the reconstructed signal feature is encoded, and the feature is mapped into a set of extremely small feature vector $\boldsymbol{y}$, which retain the noiseless feature;

Step.3 According to (13)∼(17), the reconstructed feature vector $\boldsymbol{y}$ is decoded, $\boldsymbol{y}$ maps from the low dimensional space to the high dimensional space, and the signal feature $\hat{\boldsymbol{x}}$ is obtained, thereby the signal reconstruction process is completed;

Step.4 According to (18)∼(19), the back propagation algorithm is used to train the neural network, the reconstructed signal feature $\hat{\boldsymbol{x}}$ is reconstructed and modulation recognition of the received signal is completed.

4 Simulation Results

In this section, the proposed algorithm is verified by simulation. This section shows the processing effect of the signal reconstruction layer, compares the performances of BRNN-DAE with the performances of DAE and BPNN respectively in the different SNR environments, shows the confusion matrix of DBRNN algorithm in the different SNR environments, and shows the recognition rate curve between the proposed method and the existing modulation recognition algorithms in the different SNR environments. The modulated signal data set used in this paper is generated by matlab software. The data set contains 11 modulation types: 2ASK, 2PSK, 2FSK, 4PSK, AM, SSB, DSB, VSB, FM, 16QAM, 64QAM. The signal symbol is generated randomly by matlab, and the noise of signal is Gaussian White Noise. The range of signal to noise ratio(SNR) is -20 dB to 6 dB, and the interval is 1 dB. Each modulated signal generates 40 sampling points at different SNR, a total of 15400 samples. Among them, 12320 samples are used to train the network model and 3080 samples are used to test the model. Each sample carries a label, which is a vector that length is 11, the value of one position is 1 and the value of the other positions value is 0. For example, if the modulation type of a sample is 2ASK, the sample label is $(1,0,0,0,0,0,0,0,0,0,0)$. The Euclidean distance of the same sample label is 0. The Euclidean distance of the different sample label is 1. This makes it more convenient to calculate the loss function or the accuracy. The software environment used in the training model is Python 3.6 and Tensorflow-gpu 1.3.0. The hardware environment is CPU Intel® Xeon® E5-2560, GPU NVIDIA Tesla K80, RAM 64GB.

Figure 5 shows the relationship between the received signal and the reconstructed signal. The solid line indicates the original signal with SNR= -10 dB. The dotted line represents the modulated signal is processed by the signal reconstruction layer. It is obvious from the Figure 5 that the received signal has no obvious signal feature in the low SNR environment. The amplitude, frequency, phase and other features of the modulated signal have been submerged in the noise and cannot be directly observed. If the original received signal is used as an input to the neural network, the recognition accuracy of the modulation recognition will be greatly reduced. Compared with the original received signal, the signal is processed by the signal reconstruction layer has relatively obvious signal feature. Because the

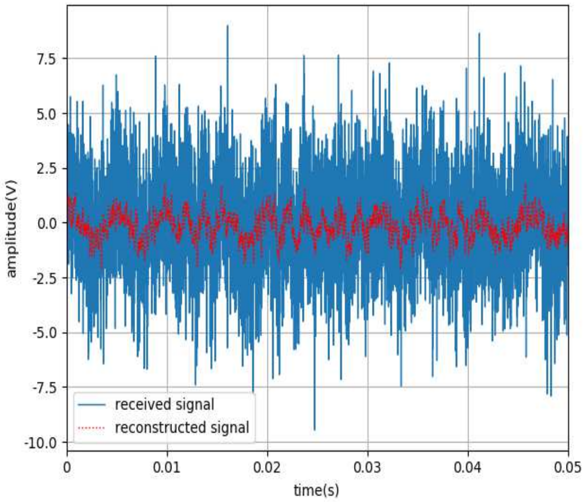

Figure 5. Received and reconstructed signal

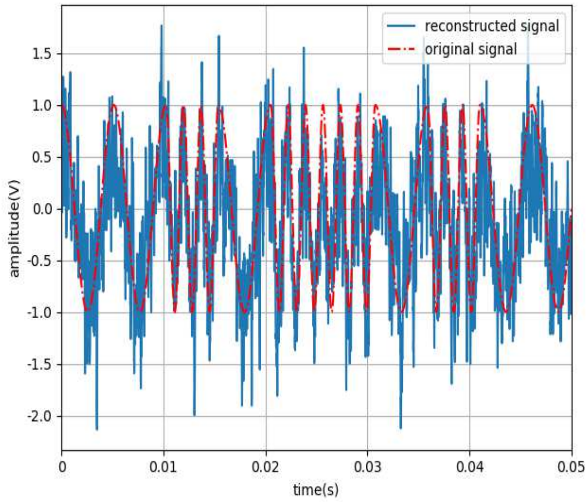

Figure 6. Reconstructed and original signal

noise in the reconstructed signal is reduced, thereby the amplitude, frequency and phase information of the signal are clearly displayed.

Figure 6 shows the modulated signal processed by the signal reconstruction layer. The reconstructed signal has obvious signal feature. It has two different carrier frequencies and the signal amplitude is basically constant within 0.05 second, so the modulation type of the signal shown in Figure 6 is FSK.

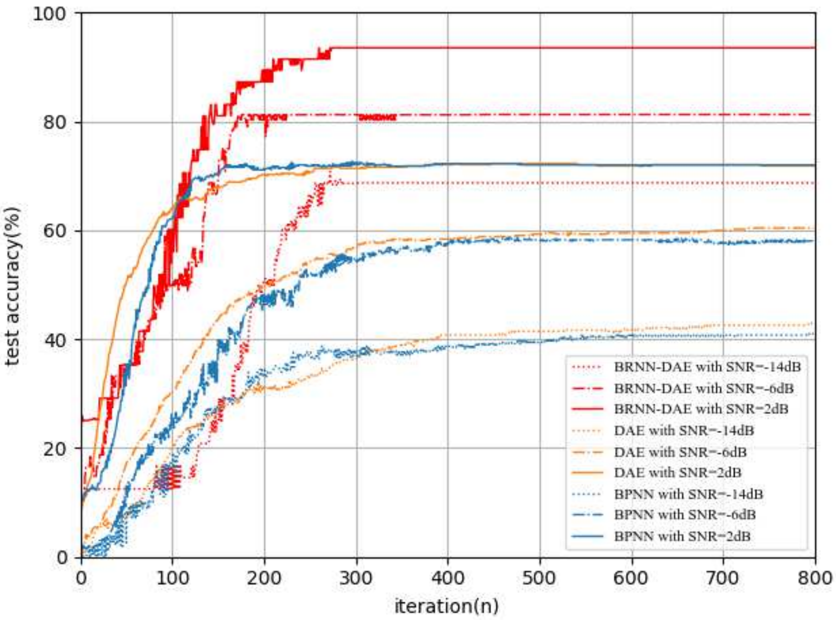

Figure 7. Test accuracy versus iterations

Figure 7 shows the relationship between iteration times and test accuracy in -14 dB, -6 dB, and 2 dB SNR environments. It can be seen from the figure 7 that with the increase of the iterations, the test accuracy of the algorithm gradually increases, and when the iterations reaches a certain number, the accuracy of the algorithm always fluctuates within a small range. Figure also shows the curve of test accuracy in three different SNR environments. When the iterations is the same, the test accuracy increases as the SNR increases. When the SNR = -14dB, the test accuracy of the proposed algorithm is close to 70%. When the SNR = -6dB, the test accuracy of the proposed algorithm exceeds 80%. This proves that compared with other algorithms, the proposed algorithm has better recognition accuracy in the low SNR environment.

Figure 8 shows the relationship between iteration times and network loss in -14 dB, -6 dB, and 2 dB SNR environments. It can be seen from the figure that with the increase of the iterations, the loss of the network gradually

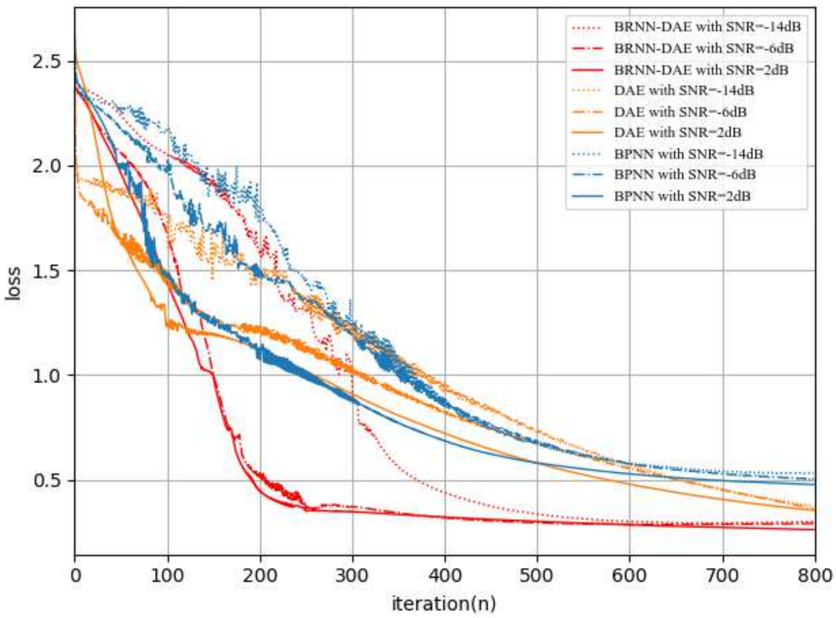

Figure 8. The test error of network versus iterations

decreases and tends to be stable. When the SNR = 2dB and - 6dB, the loss of the network is stable when the iteration exceed 300. When the SNR = - 14dB, the loss of the network is stable when the iteration exceed 600. With the increase of SNR, the loss of network decreases gradually. When the iterations of the network is constant, the higher the SNR, the smaller the network loss. When the SNR = - 14dB and the network loss is stable, the network loss value is about 0.25. It can be seen that the proposed algorithm has low network loss in the low SNR environment.

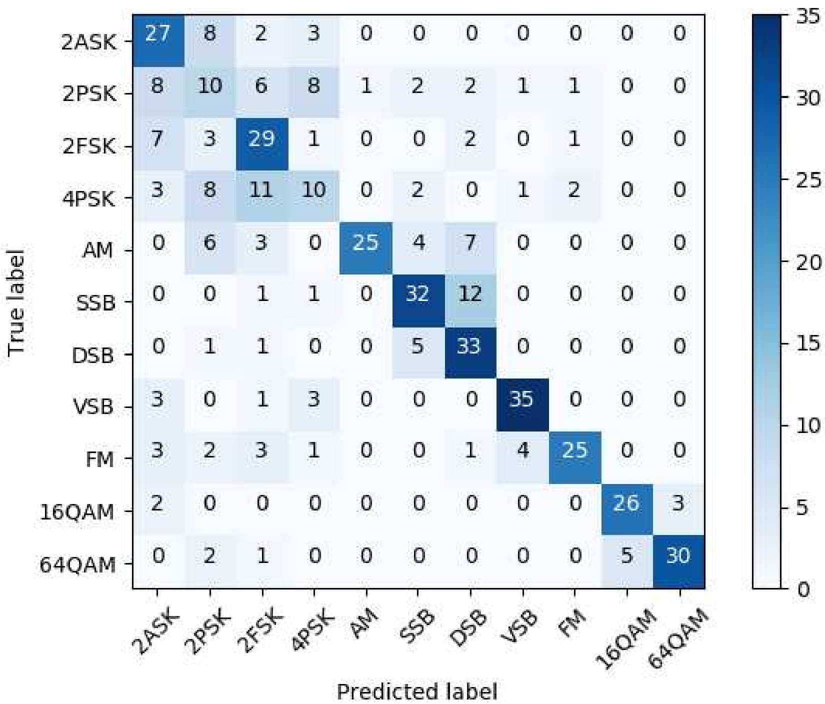

Figure 9. Confusion matrix with SNR= -14dB

Figure 9, Figure 10 and Figure 11 show the confusion matrix of DBRNN algorithm in different SNR environments. It can be seen from the figure that when SNR is - 14dB, according to the data distribution of the confusion matrix, the data located in the diagonal is more than the data outside the diagonal, and the sum of the numbers off the diagonal is

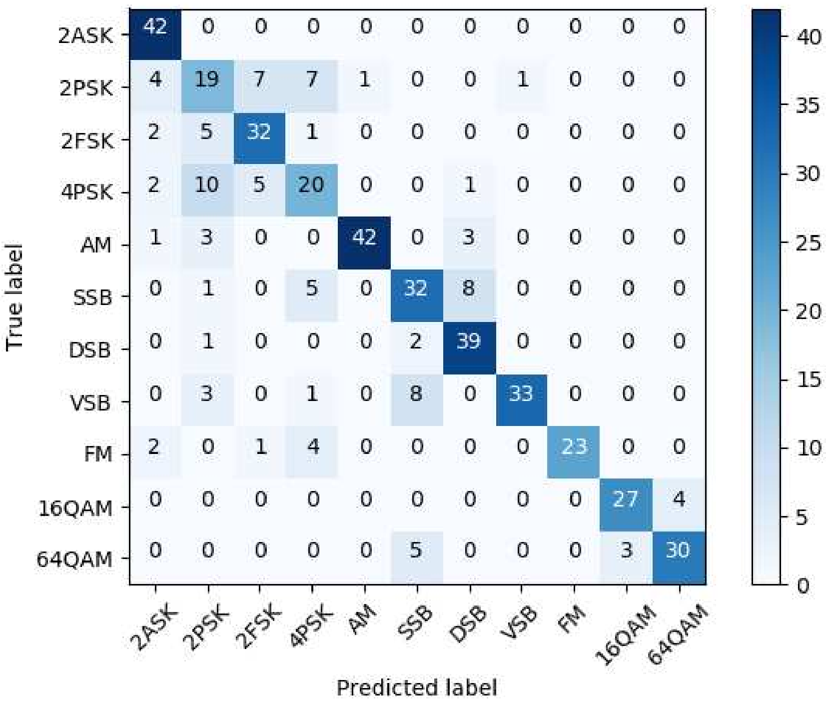

Figure 10. Confusion matrix with SNR= -6dB

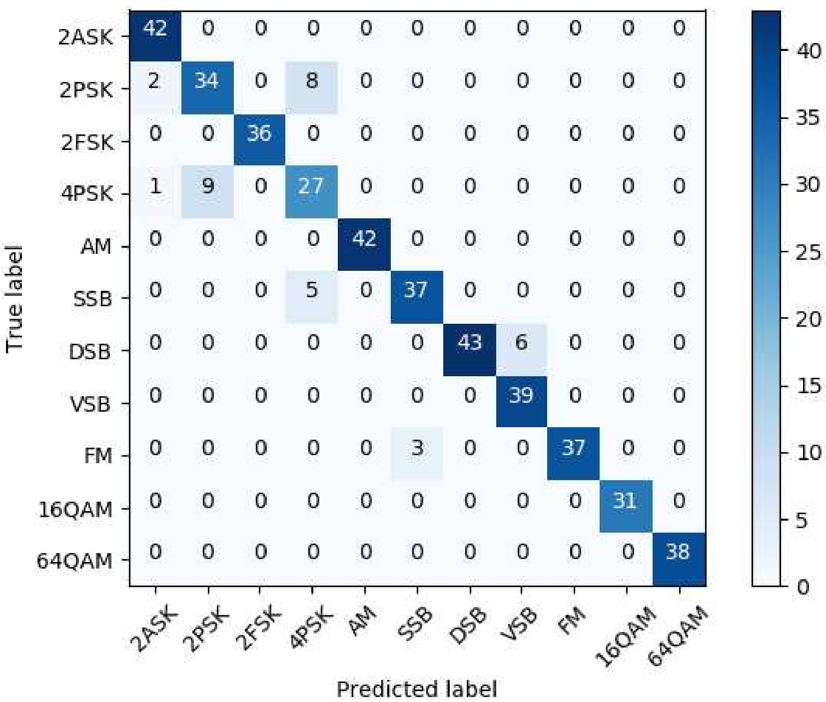

Figure 11. Confusion matrix with SNR= 2dB

smaller, and the recognition error of each modulation signal is about 1 2. From the data distribution, the recognition accuracy is more than 65%. Compared with the obfuscation matrix with the SNR = - 14dB, when the SNR = - 6dB, as the amount of data on the diagonal of the obfuscation matrix increases, the amount of data off the diagonal decreases, and the amount of recognition errors of each modulation type also decreases, which proves that with the increase of SNR, the recognition accuracy of the proposed algorithm increases, and the recognition accuracy is above 80%. When the SNR = 2dB, except for a few data distributed outside the diagonal, the rest of the data are all correctly distributed on the diagonal. It can be concluded that the recognition accuracy of the proposed algorithm is close to 95%, which further proves that the proposed algorithm has a high recognition rate in the low SNR environment.

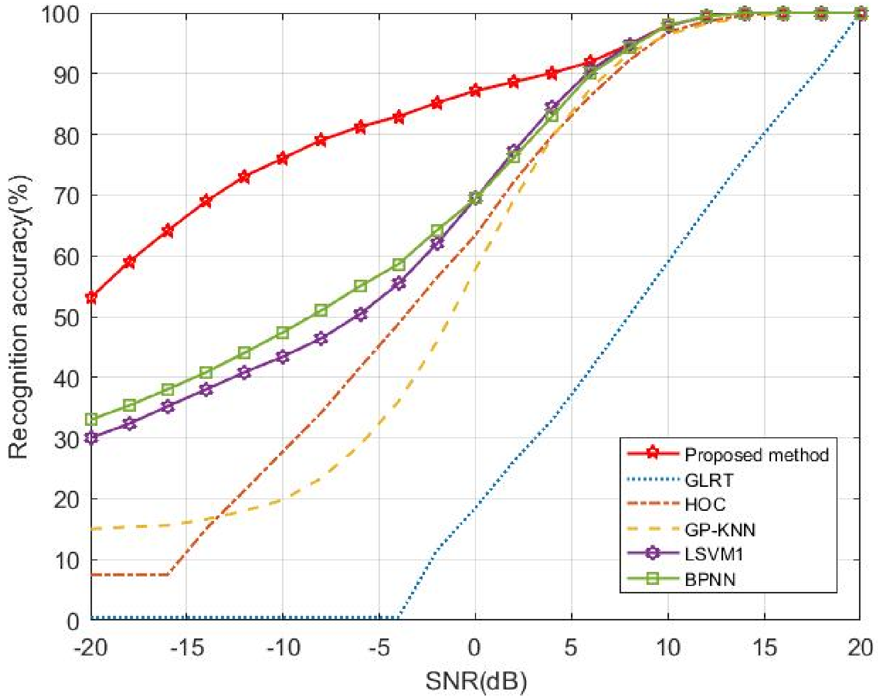

Figure 12. Recognition accuracy versus SNR

Figure 12 shows the recognition rate curve between the proposed method and the existing modulation recognition algorithms in the different SNR environments. In addition to the curves of DBRNN algorithm, the figure also includes the curves of generalized likelihood ratio test (GLRT) [23], high-order cumulants (HOC) [24], k -nearest neighbors (KNN) [10], linear support vector machine (LSVM) and backpropagation neural network (BPNN) classifiers [25]. In GLRT, assume that the prior probabilities of the eleven modulated signals are the same, the threshold of the classified classification is set to zero, and the probability of correct classification is the average of 1000 independent experiments. In HOC, the mean and variance of the modulating signal statistic are calculated in each Monte Carlo experiment. The fourth-order cumulants and the optimal threshold of the signal are calculated by mean and variance, which are compared and identified. In KNN, the genetic operator crossover has a probability of 90% and the probability of mutation is 10%. 10,000 sampling points are generated for different SNR values. The 10,000 sampling points are tested using the optimal tree and the results are summarized. In LSVM, the amplitude, phase, real part, and imaginary part of the signal are respectively calculated to identify the modulated signal. In BPNN, the two layers neural network is adopted and 50 nodes are used in the hidden layer. Compared with other algorithms, the proposed algorithm has higher recognition accuracy in the low SNR environment. When the SNR exceeds - 18dB, the recognition accuracy of the proposed algorithm is more than 60%. Compared with the existing modulation recognition algorithm based on BPNN, the recognition rate is improved by about 20%. Compared with other existing algorithms in the low SNR environment, when the SNR is greater than - 10dB, the recognition rate of the proposed algorithm has reached more than 70%, which proves that the proposed algorithm has better recognition performance in the low SNR environment.

5 Conclusions

Most existing modulation recognition algorithms have low recognition rate in the low SNR environment. In order to solve this problem, a modulation recognition algorithm based on DBRNN is proposed in this paper. Firstly, the state memory ability of the signal reconstruction layer in the network is used to learn the temporal correlation of the modulated signal, the reconstruction of the received signal is completed and the noise in the modulated signal is suppressed. Then, the reconstructed signal is encoded and decoded by the feature reconstruction layer, in which the feature of reconstructed signal is compressed and reconstructed, thereby the influence of noise can be further reduced.

Finally, the reconstructed features are identified and classified by the fully connected layer. Simulation results show that the proposed network can effectively suppress the noise in the signal. Compared with other existing algorithms, the proposed method has higher recognition accuracy in the low SNR environment.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No.61971117, by the Natural Science Foundation of Hebei Province (Grant No. F2020501007).

References

1. Huan C Y , Polydoros A, Likelihood methods for MPSK modulation classification. IEEE Transactions on Communications. 43(234),0-1504(1995).
2. Wei W , Mendel J M, A new maximum-likelihood method for modulation classification. IEEE Conference on Signals, Systems, Computers. 1132-1136(1995).
3. Wei W , Mendel J M, Maximum-likelihood classification for digital amplitude-phase modulations. IEEE Transactions on Communications. 48(2), 189-193(2000).
4. Chavali, Silva D, Maximum-Likelihood Classification of Digital Amplitude-Phase Modulated Signals in Flat Fading Non-Gaussian Channels. IEEE Transactions on Communications. 59(8), 2051-2056(2011).
5. Nandi A K, Azzouz E E, Algorithms for automatic modulation recognition of communication signals. IEEE Transactions on Communications. 46(4), 431-436(1997).
6. Deng H, Doroslovacki M, Mustafa H, Automatic digital modulation classification using instantaneous features. IEEE International Conference on Acoustics, Speech, and Signal Processing. Orlando, pp. IV-4168-IV-4168, (2002),
7. Swami A, Sadler B M, Hierarchical digital modulation classification using cumulants. IEEE Transactions on Communications. 48(3), 416-429(2000).
8. Teng X, Tian P, Yu H, Modulation classification based on spectral correlation and SVM. International Conference on Wireless Communications. 1-4(2008).
9. Chen P, Liu S, An Improved DAG-SVM for Multi-class Classification. IEEE International Conference on Natural Computation. 460-462(2009).
10. Orlic V D , Dukic M L, Automatic modulation classification algorithm using higher-order cumulants under real-world channel conditions. IEEE Communications Letters. 13(12), 917-919(2009).
11. Liu Y, Simeone O, Haimovich A M, Modulation classification via gibbs sampling based on a latent dirichlet bayesian network. IEEE Signal Processing Letters. 21(9), 1135-1139(2014).
12. OSHEA T J, WEST N, Radio machine learning dataset generation with gnu radio. Proceedings of the GNU Radio Conference. 1(2016).
13. Shi W, Liu D, Cheng X, Li Y, Zhao Y, Particle swarm optimization-based deep neural network for digital modulation recognition. IEEE Access. 7, 104591-104600(2019).
14. Mendis G J , Wei J , Madanayake A, Deep learning-based automated modulation classification for cognitive radio. IEEE International Conference on Communication Systems. (2016).
15. Wen X, Sheng H, Chao Y, Peng Z, Deep learning in digital modulation recognition using high order cumulants. IEEE Access. 7, 63760-63766(2019).
16. Tang B, Tu Y, Zhang S, Digital Signal Modulation Classification with Data Aug-mentation Using Generative Adversarial Nets in Cognitive Radio Networks. IEEE Access. 1-1(2018).
17. Ali A , Yangyu F, Automatic Modulation Classification Using Deep Learning Based on Sparse Autoencoders With Nonnegativity Constraints. IEEE Signal Processing Letters. 24(11), 1626-1630(2017).
18. Yu W, Miao L, Data driven deep learning for automatic modulation recognition in cognitive radios. IEEE Transactions on Vehicular Technology. 68(4), 4074-4077(2019).
19. Wallayt, Younis, Imran, Automatic Modulation Classification for Low SNR Digital Signal in Frequency-Selective Fading Environments. Wireless Personal Communications. 84(3), 1891-1906(2015).
20. Ivan A, Jafar W, Abdul Sadah, Thamir R. Recognition of QAM Signals with Low SNR Using a Combined Threshold Algorithm. IETE Journal of Research. 61(1), (2015).
21. Baher S, Kastantin R. Automatic Modulation Classification of LFM and Polyphase-coded Radar Signals. Radio Engineering. 26(4), 1118-1125(2017).
22. Park C , Kim D Y. A novel robust feature of modulation classification for reconfigurable software radio. IEEE Transactions on Consumer Electronics. 52(4), 1193-1200(2006).
23. Li L , Ding Y , Zhang J K , Blind Detection with Unique Identification in Two-Way Relay Channel. IEEE Transactions on Wireless Communications. 11(7), 2640-2648(2012).
24. Swami A , Sadler B M, Hierarchical digital modulation classification using cumulants. IEEE Transactions on Communications. 48(3), 416-429(2000).
25. Han L , Xue H , Gao F, Low Complexity Automatic Modulation Classification Based on Order Statistics. IEEE Vehicular Technology Conference. (2017).

Figures

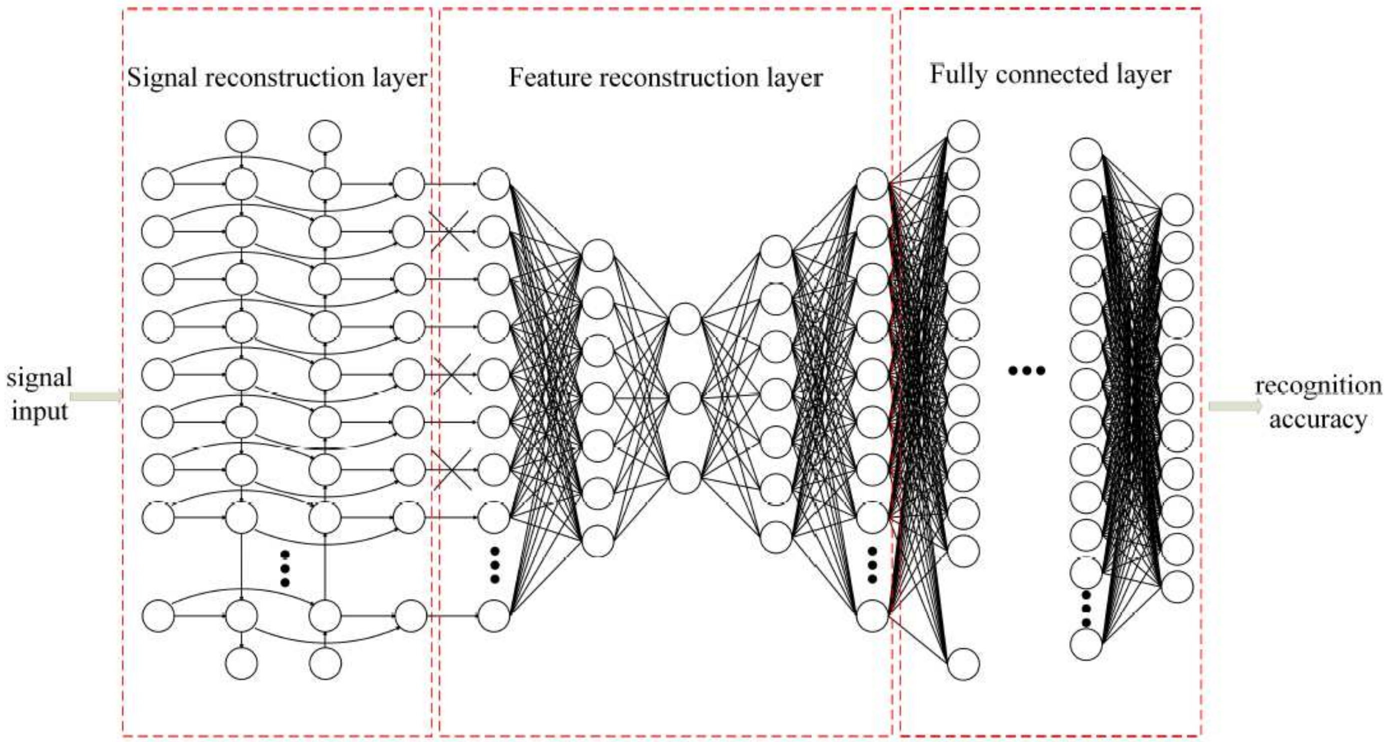

Figure 1

DBRNN network structure

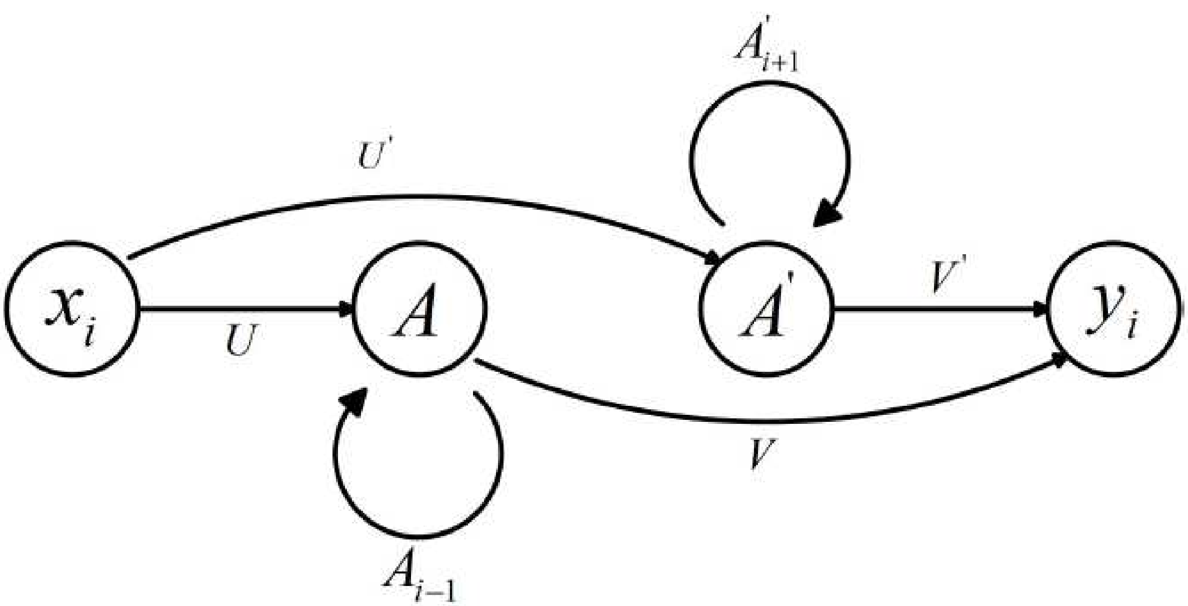

Figure 2

The neurons of the signal reconstruction layer

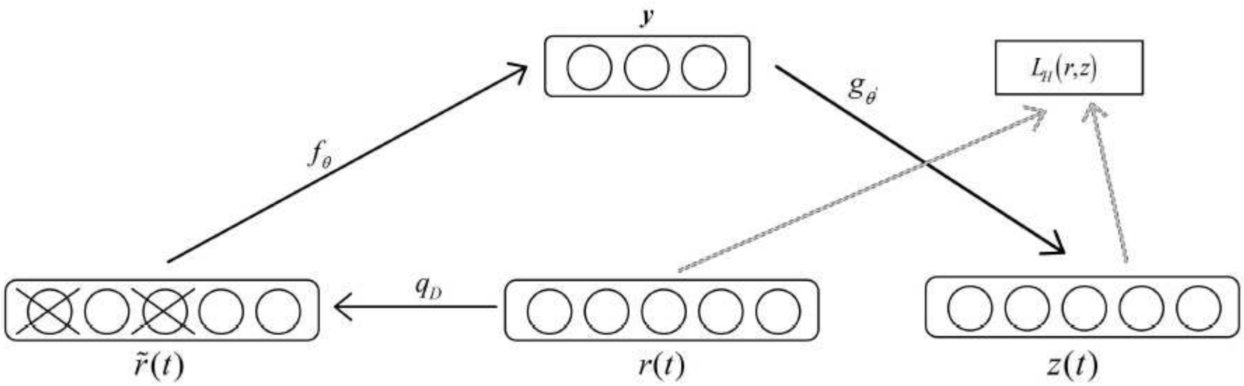

Figure 3

The structure of the feature reconstruction layer

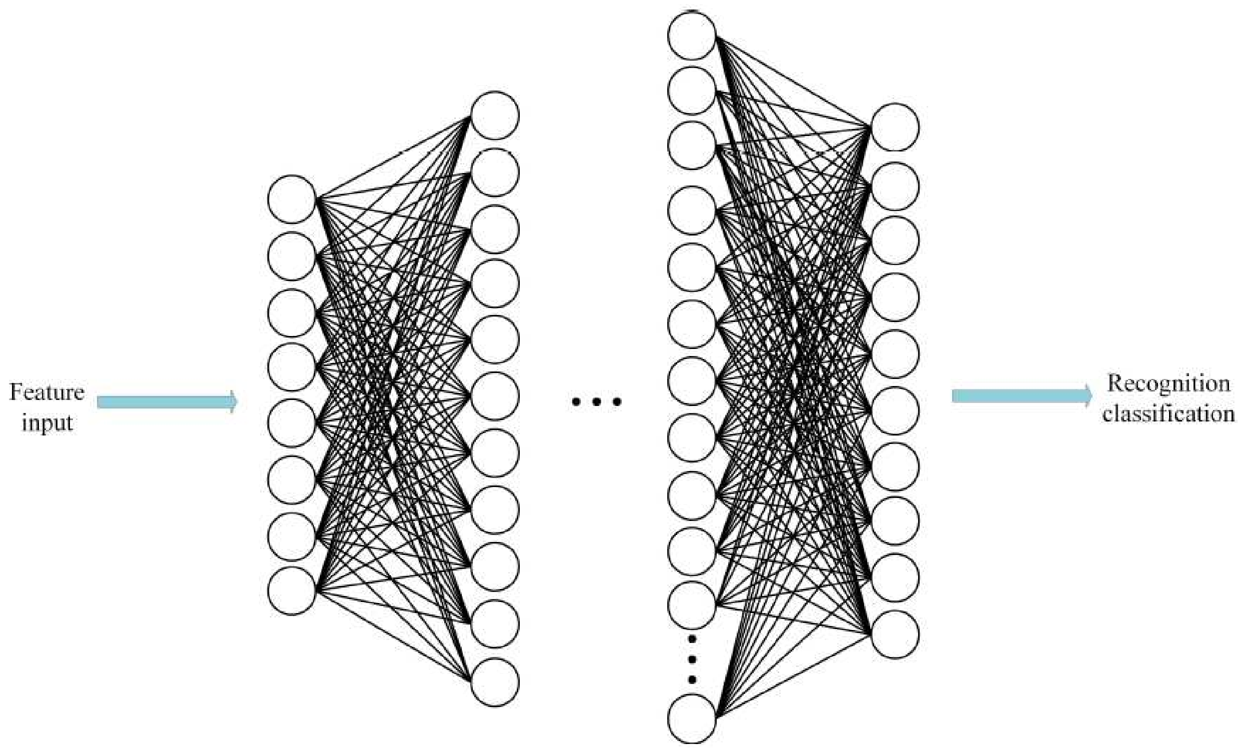

Figure 4

14-layer neural network

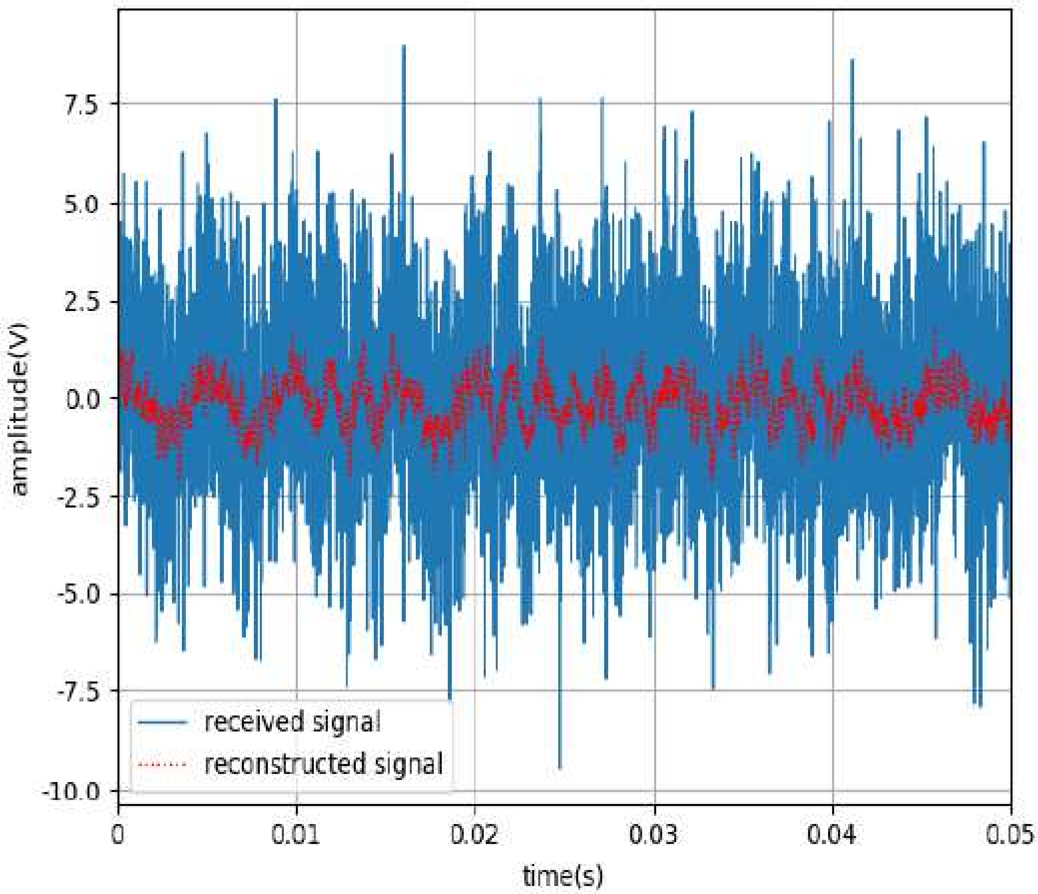

Figure 5

Received and reconstructed signal

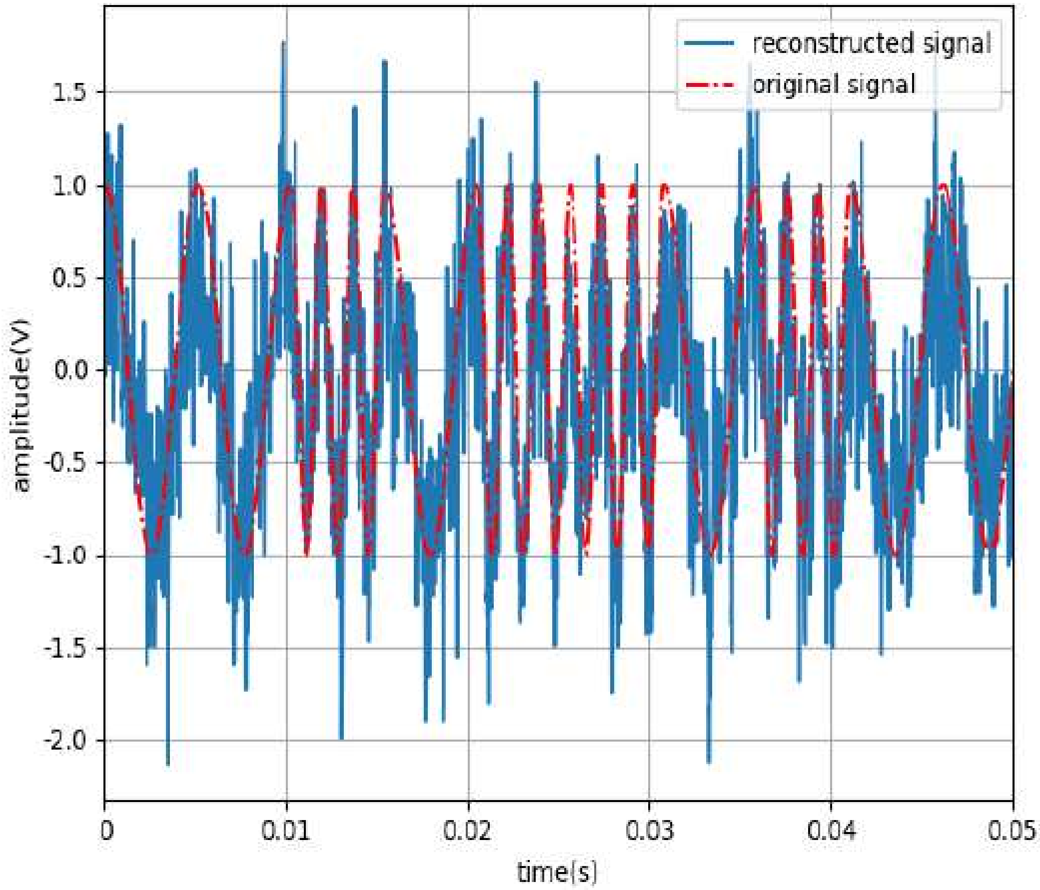

Figure 6

Reconstructed and original signal

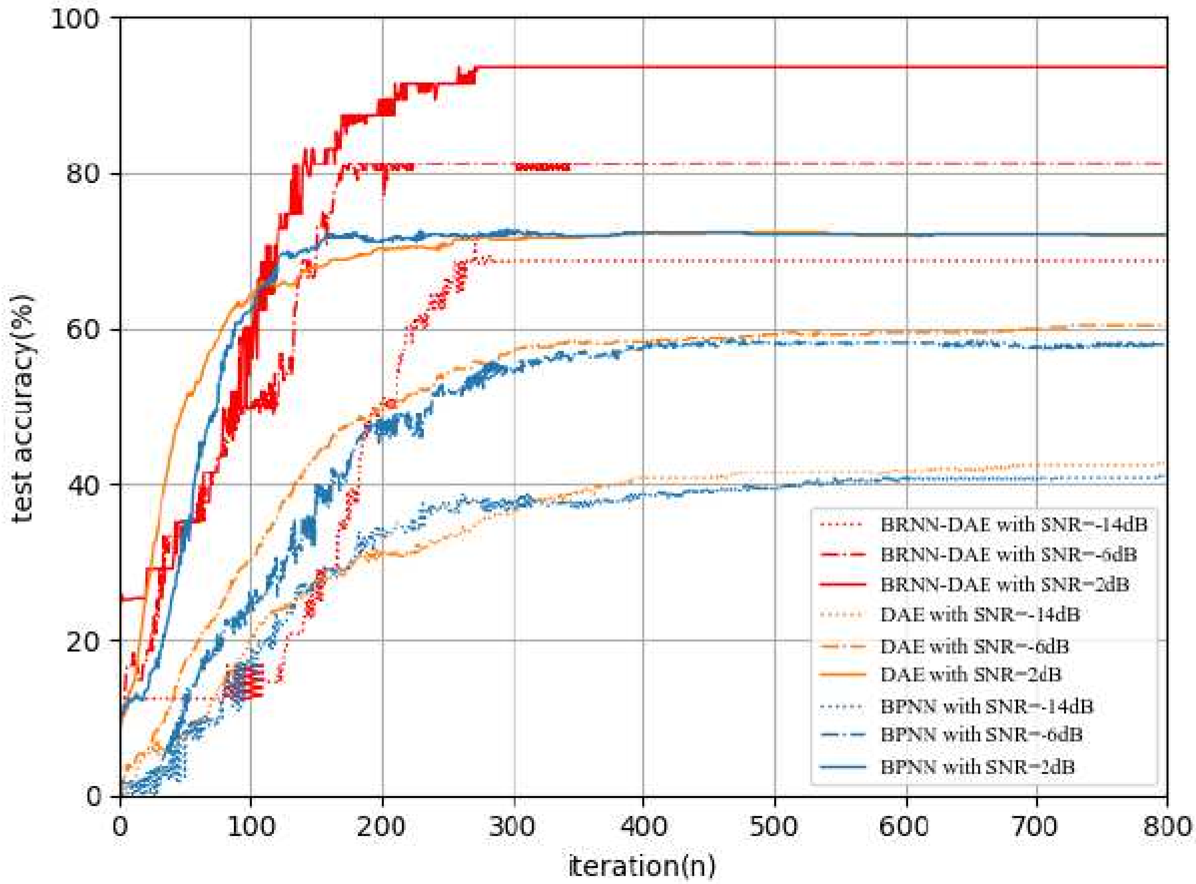

Figure 7

Test accuracy versus iterations

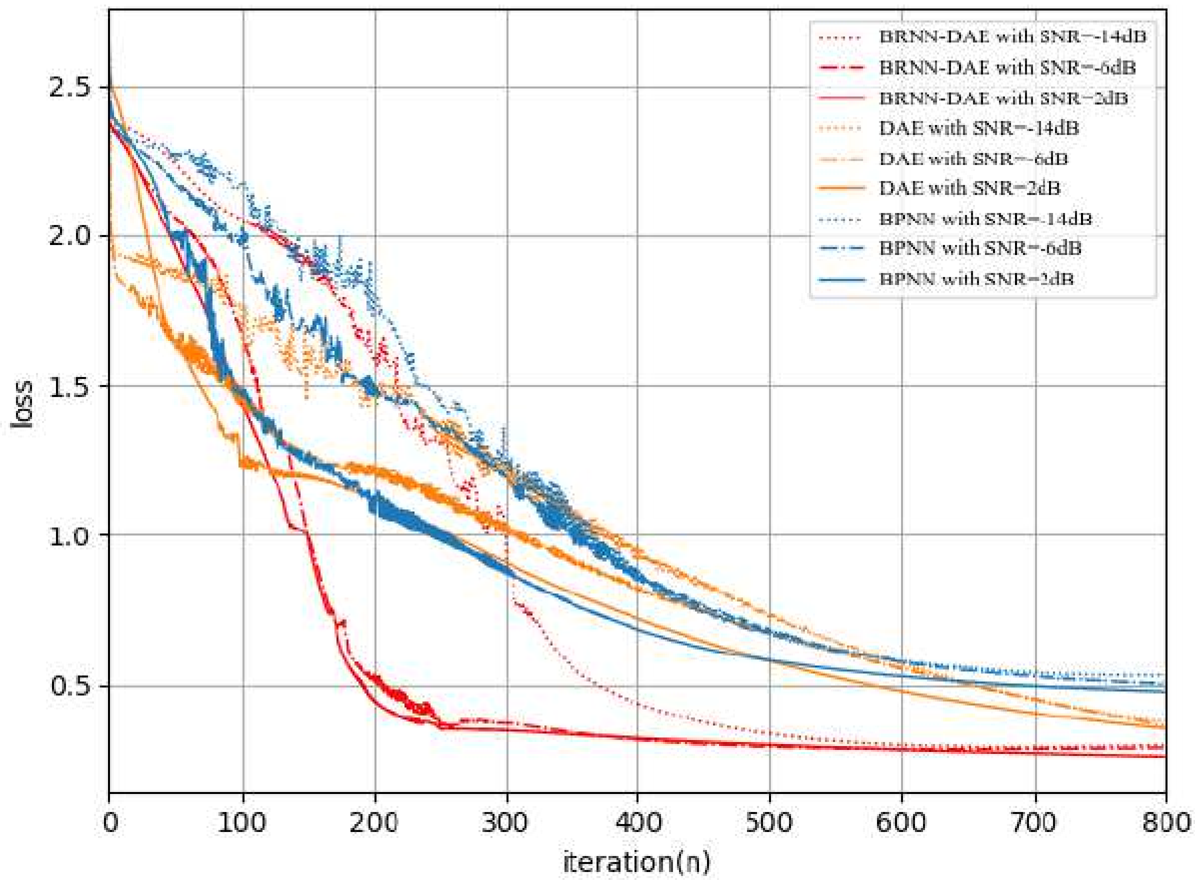

Figure 8

The test error of network versus iterations

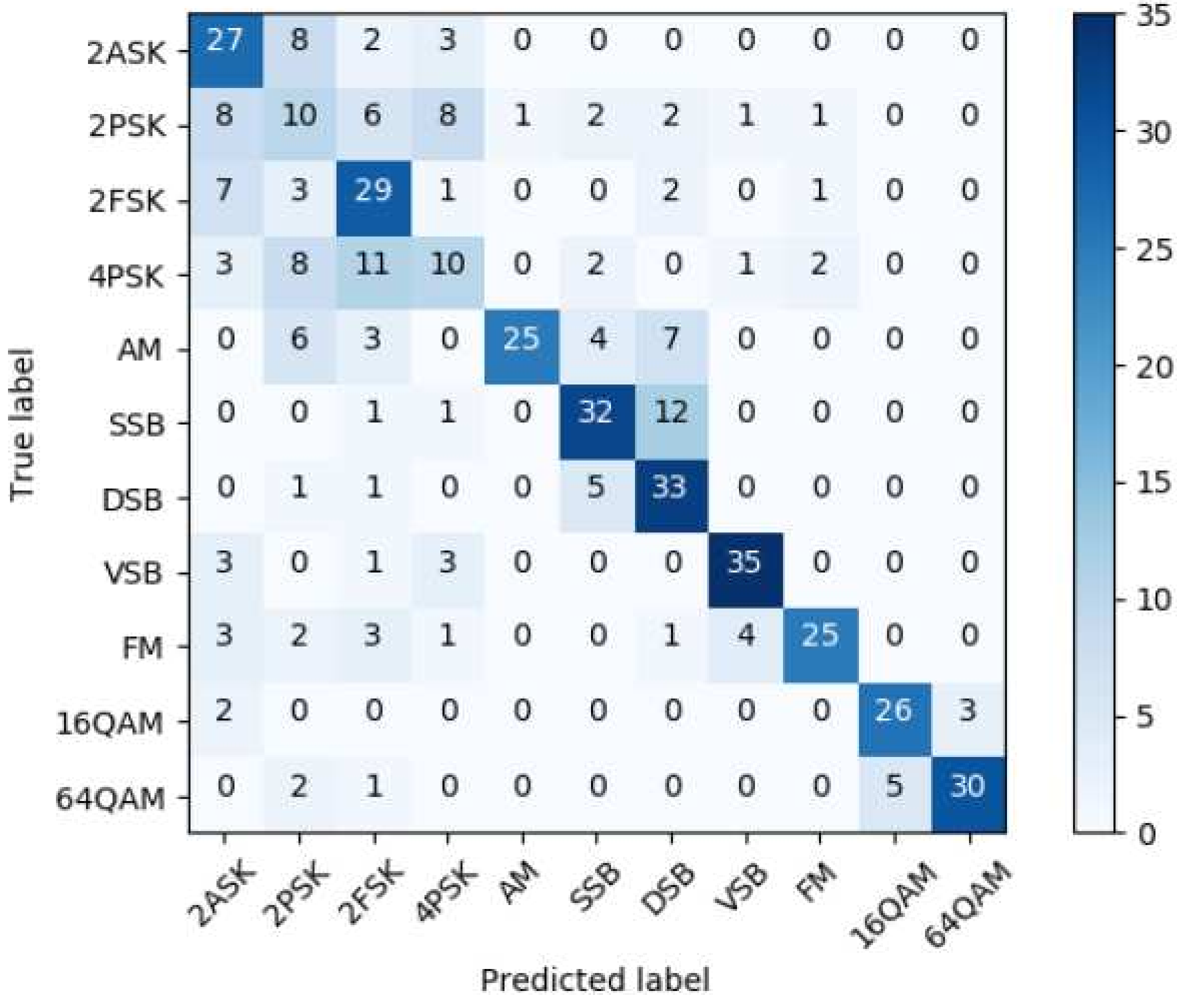

Figure 9

Confusion matrix with SNR= -14dB

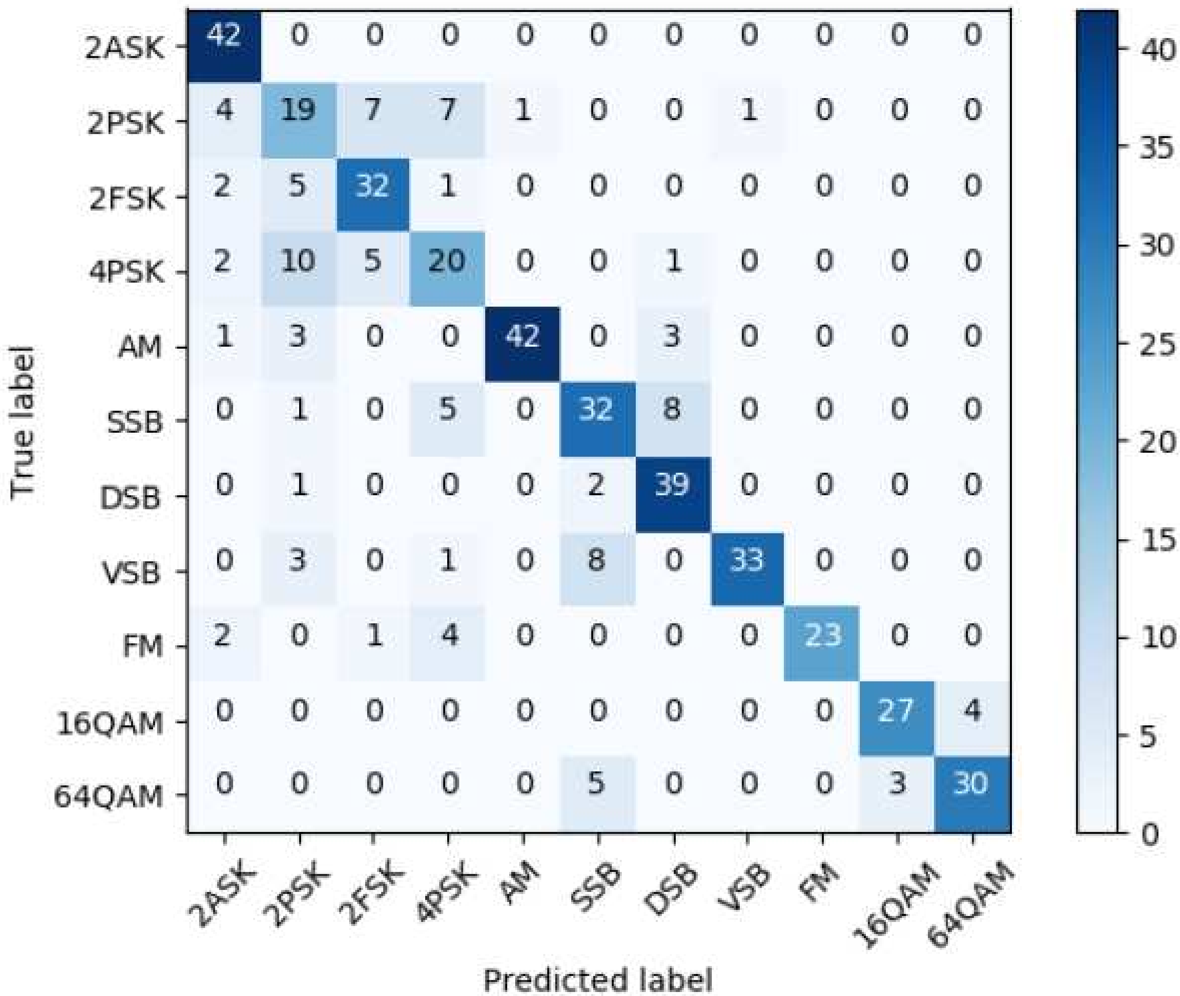

Figure 10

Confusion matrix with SNR= -6dB

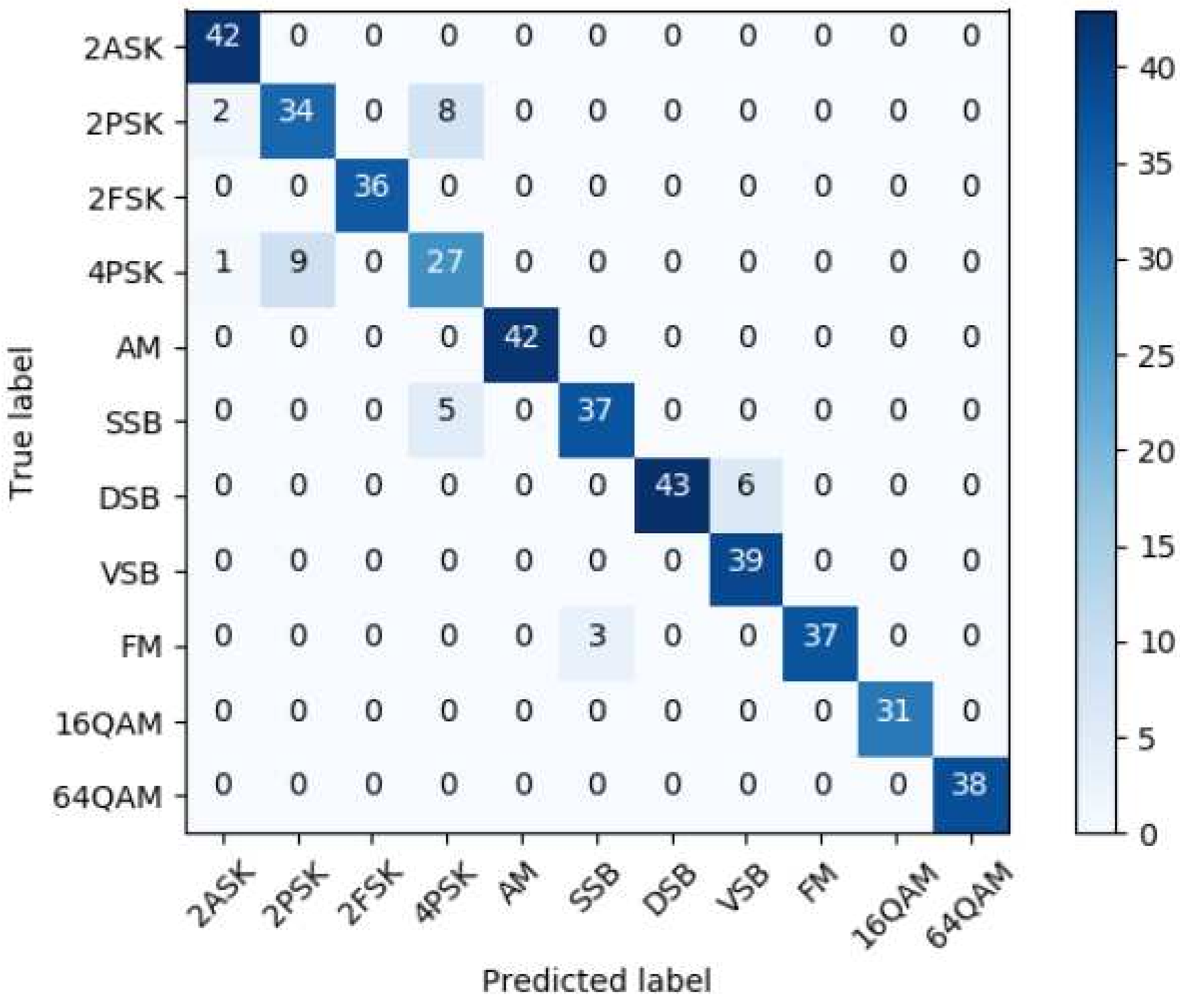

Figure 11

Confusion matrix with SNR= 2dB

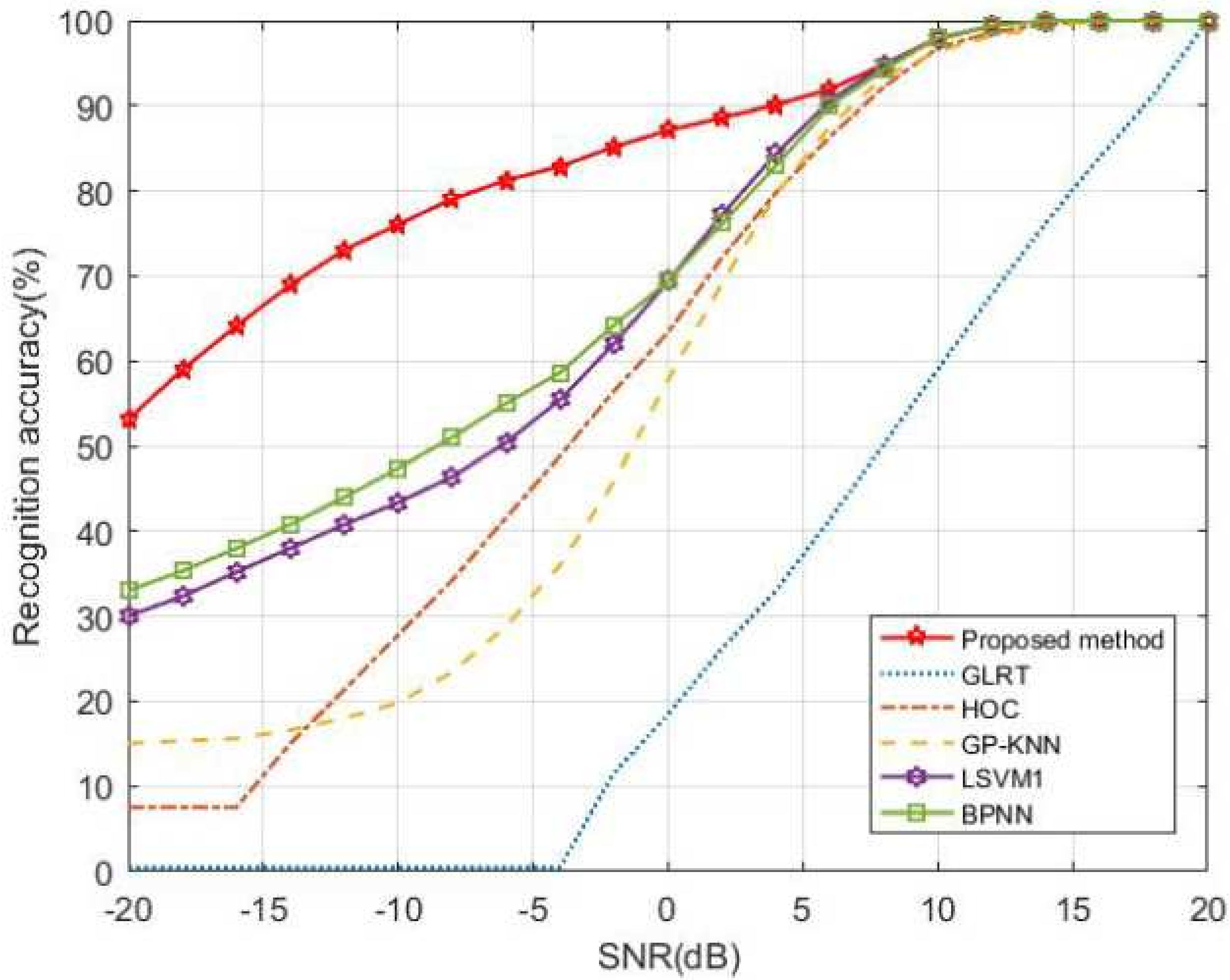

Figure 12

Recognition accuracy versus SNR

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- wpcModulationRecognitionBasedonDBRNN.rar