Multi-modulation Recognition Using Convolution Gated Recurrent Unit Networks

TAN Jiyuan¹, ZHANG Limin², ZHONG Zhaogen³ and YU Wenlong⁴
¹,²,³Naval Aviation University, Yantai, Shandong 264001, China
⁴Naval Air Force 91213

E-mail address: 173160056@qq.com

Abstract. To solve the problems of long modulation and recognition time and complexity of convolutional neural networks, a modulation method based on CNN-GRU (Convolutional Neural Networks and Gated Recurrent Unit) is proposed. Firstly, the spatial features of the signal are extracted by CNN convolution operation, then the timing correlation of the signal is extracted by GRU and finally the recognition probability is output by using the softmax layer to achieve the purpose of multi-modulation recognition. The experimental results show that the recognition performance of the method is further improved under the condition of no prior information such as channel and noise, and 11 modulation categories such as 16QAM and 64QAM can be effectively identified, and the complexity of the method is low, which greatly saves. The training identification time has good engineering application value.

1. Introduction

Signal modulation recognition technology has always been a research hotspot in the field of wireless communication. It is widely used in military and civilian fields, such as electronic countermeasures, signal monitoring and information acquisition. It is the basis of cognitive radio and perceptual spectrum [1][2][3]. In recent years, modulation recognition technology has been greatly developed. Domestic and foreign scholars have done a lot of research on the identification of different modulation modes of signals and the debugging and identification of signals under different channel and noise conditions, and have obtained some practical results, including the largest. Likelihood method [4] [5], high-order cumulant method [5] [6], cyclic spectrum algorithm [7] and KS detection algorithm [1] [8].

With the development of communication technology, the types of modulation methods of communication signals are also diverse. How to effectively distinguish multiple modulation methods under different signal-to-noise ratios and different channel conditions is still a difficult problem in research. At present, the literature in this aspect is relatively more difficult. Less. N.Ghani proposed to classify the difference between the square spectrum peaks of the equal signals and the power spectrum [9]. This method can identify 10 modulation modes, but only for the AWGN channel. In 2007, K.Maliatsos et al. first introduced wavelet transform [10], firstly reduced the influence of noise by wavelet transform, and then extracted features for modulation recognition, which realized the recognition of multiple modulation methods of PSK, FSK, ASK and QAM. At the same time, the computational complexity is reduced, but the performance of the algorithm when the signal-to-noise ratio is lower than 10 dB is not analyzed. In recent years, deep learning has gradually been applied to the field of modulation recognition. Compared with traditional machine learning, deep learning does not need to know information about modulated signals, and has the advantage of automatically
learning features directly from original data. In 2016, Timothy J O'Shea et al. proposed the use of convolutional neural networks to classify 11 modulated signals [11]. The results show that this method is superior to traditional identification methods, and has a variety of recognition types and accuracy. Improve, but the recognition rate of QPSK is lower, and the algorithm takes longer to operate. In 2017, et al. proposed a Hierarchical Deep Neural Networks (H-DNN)-based recognition method [12], which solved the problem of low recognition rate of WBFM in the literature [11], but produced new ones. Problem, the recognition rate between 4-QAM and 16-QAM has been reduced. In 2018, Rundong Li et al. proposed robust automatic VHF based on deep convolutional neural network for modulation recognition [13]. This method improves the anti-noise performance and the robustness of the algorithm, but the recognition type is reduced and the recognition is common. Seven modulation methods have been added, and the network complexity has increased. For the first time in literature [14], convolutional long-term and short-term deep neural networks combined with CNNs and LSTM are applied to the field of modulation identification. Compared with CNNs identification network, CLDNN has improved recognition performance due to the increase of network depth.

In this paper, the convolutional neural network is combined with the gated loop unit to propose a signal multi-modulation method for the convolution-gated loop unit neural network. For the first time, this method combines CNN and GRU in the field of modulation identification, and reduces the number of convolutional layers. It is constructed into a simpler neural network to achieve the purpose of modulation recognition. The CNN-GRU method proposed in this paper has the following advantages: (1) the number of network layers and training parameters are reduced, and the recognition time is greatly reduced; (2) the prior identification information such as channel and noise is not needed, and the modulation recognition rate is improved.

2. network description

2.1. Convolutional Neural Network
CNNs is a feedforward neural network. Convolutional neural networks have the characteristics of translation invariance in image processing, so they have always been one of the core algorithms in the field of image recognition. Standard CNNs generally consist of an input layer, a convolutional layer, a pooled layer, a fully connected layer, and an output layer [15][16], as shown in Figure 1.

2.2. Gated Recurrent Unit
GRU is a special kind of Recurrent Neural Networks (RNN), which can use the previous input of prediction information together with several inputs later, so that the prediction will be more accurate. The network structure of the GRU is shown in Figure 2 below.
2.2.1. Forward propagation of GRU

\[ r_t = \sigma(w_r \cdot [h_{t-1}, x_t]) \]
\[ z_t = \sigma(w_z \cdot [h_{t-1}, x_t]) \]
\[ \tilde{h}_t = \tanh(w_c \cdot [r_t \cdot h_{t-1}, x_t]) \]
\[ h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \]
\[ y_t = \sigma(w_o \cdot h_t) \]

Where \[ [] \] indicates that two vectors are connected, and \(*\) indicates the product of the matrix.

2.2.2. Training process of GRU

From the formula in the forward propagation process, it can be seen that the parameters to be learned are \( w_r, w_z, w_o, w_h \). The first three parameters are spliced (because the latter vectors are also spliced), so you need to split them during the training:

\[ w_r = w_{rz} + w_{rh} \]
\[ w_z = w_{zr} + w_{zh} \]
\[ w_o = w_{oz} + w_{oh} \]

The input of the output layer is \( y'_t = w_t \cdot h_t \), and the output of the output layer is \( y''_t = \sigma(y'_t) \). After the final output is obtained, the loss of network transmission can be written. The loss of a single sample at a certain moment is \( E_t = \frac{1}{2}(y'_d - y''_d)^2 \), and the loss of a single sample at all times is \( E = \sum_{t=1}^{T} E_t \). The backward error propagation algorithm is used to learn the network, and the loss function is first obtained. Partial guidance for each parameter:

\[ \frac{\partial E}{\partial W_{w_r}} = \delta_{r_t} \cdot h_t, \quad \frac{\partial E}{\partial W_{w_z}} = \delta_{z_t} \cdot x_t, \quad \frac{\partial E}{\partial W_{w_o}} = \delta_{o_t} \cdot h_{t-1}, \quad \frac{\partial E}{\partial W_{w_h}} = \delta_{h_t} \cdot x_t \]

\[ \frac{\partial E}{\partial W_{w_{rz}}} = \delta_{r_t}, \quad \frac{\partial E}{\partial W_{w_{zh}}} = \delta_{z_t} \cdot x_t, \quad \frac{\partial E}{\partial W_{w_{oz}}} = \delta_{o_t} \cdot x_t, \quad \frac{\partial E}{\partial W_{w_{oh}}} = \delta_{h_t} \cdot h_{t-1} \]

Wherein, each intermediate parameter is \( \delta_{y'_t} = (y'_d - y''_d) \cdot \sigma' \).
\[
\delta_h = \delta_{r} W + \delta_{r+1} W_{\theta} + \delta_{r+1} W_{\text{h1}} r_{r+1} + \delta_{h+1} W_{\theta} + \delta_{h+1} (1 - z_{r+1})
\]

\[
\delta_{r} = \delta_{h} \cdot (h_{r-1}) \cdot \sigma
\]

\[
\delta_{r+1} = \delta_{h} \cdot z_{r} \cdot \phi
\]

\[
\delta_{r+1} = h_{r-1} \cdot [(\delta_{h} \cdot z_{r} \cdot \phi) W_{\text{h1}}] \cdot \sigma
\]

(4)

After calculating the partial derivatives of each parameter, the parameters can be updated, and the loss convergence is obtained by sequentially iterating.

3. Experimental data and CNN-GRU network

3.1. Experimental database

The database of modulation recognition is derived from the RML2016.04C data sets published by Timothy J O’Shea[17]. The data set is composed of I/Q two-way data and is one that includes each sampled data. In-phase and quadrature components, the dimension is \(2 \times 128\). The data sets are composed of 220,000 modulated data samples, including 11 modulation categories, including AM-DSB, AM-SSB, WBFM three analog modulations and BPSK, 8PSK, CPFSK. Eight digital modulations such as GFSK, 4QAM, 16QAM, 64QAM and QPSK. These data are approximately evenly distributed at 2dB intervals from SNR = -20dB to SNR = 18dB. These modulated data samples are obtained using real speech and text signals under channel conditions such as fading, multipath and additive white Gaussian noise.

3.2. Network structure

The CNN-GRU network used in this paper consists of two convolutional layers, one GRU layer and one fully connected layer. The two convolutional layers use the modified linear unit ReLU as the activation function, and the full connection layer uses the soft maximum function SoftMax as the activation function to classify the 11 kinds of modulation data, while the convolutional layer and the GRU layer use the missing output Dropout to prevent over-fitting phenomenon. At the same time, Adam is selected as the gradient descent optimization algorithm. The specific network structure is shown in Figure 3.

![CNN-GRU network structure](image)

Figure 3. CNN-GRU network structure

The input layer is the modulation data \(2 \times 128\) of the I/Q two-way dimension. The size of the first layer convolutional layer convolution kernel is \(1 \times 3\), and the number is 128. The size of the second convolutional layer convolution kernel is \(2 \times 3\), and the number is 32. The number of neurons in the GRU layer is 128, and the number of neurons in the last layer of the fully connected layer is 11, corresponding to the 11 modulation type.

3.3. Network implementation

First, the input data is processed, and the Reshape function in Keras is used to convert the \(2 \times 128\) dimension modulation data into \((\text{None}, 1, 2, 128)\) 4D tensors, which in turn represent (sample number, image height, image width, image channel number). As an input to the convolution layer. After convolution of the two convolutional layers, the output is \((\text{None}, 128, 1, 128)\) and the dimension is the 4D tensor of \(32 \times 128\). Since the input requirement of the GRU layer is 3D tensor, the Reshape function is used again to convert the 4D tensor to \((\text{None}, 128, 124)\), 32 as the dimension of the GRU
network input sequence, and 124 as the length of the GRU network input sequence to the GRU network. After the two-way operation of equations (1), (2), and (3), a 2D tensor (None, 128) is output. It is then passed to the last fully connected layer.

3.3.1. Learning of convolutional layers. In the convolutional layer, it is assumed that the input of the convolutional layer is \( X \in \mathbb{R}^{A \times B} \), where \( A \) represents the number of features of the input signal and \( B \) represents the number of input signals. Let \( x = [x_1, x_2, \ldots, x_n] \), where \( x_b \) represents the eigenvector of signal \( b \). The activation of the convolutional layer can be calculated as:

\[
 h_{j,k} = \theta(\sum_{b=1}^{B} W_{b,j}^T x_{b,k-1} + a_j)
\]  

(5)

Where \( h_{j,k} \) represents the convolutional network output of the \( j \)th feature map, \( W_{b,j}^T \) is the weight parameter of the \( j \)th convolver, \( s \) is the size of the convolver, and \( a_j \) is the offset of the \( j \)th feature map. After the input and convolver are weighted averaged, the output node value of the network is calculated by the nonlinear activation function \( \theta \). Through the formula (6), the convolution layer realizes the convolution of the feature map, and the corresponding output features are obtained. The nonlinear activation function \( \theta \) used in this paper is ReLU, and the specific expression is shown in formula (7).

\[
 \text{ReLU}(x) = \max(0, x)
\]  

(6)

The number of input channels of the convolutional layer is determined by the number of channels that input 4D tensors. The number of channels for the output tensor is determined by the number of output channels of the convolutional layer. The height and width of the output tensor are calculated as follows.

\[
 \text{height}_{\text{out}} = \frac{\text{height}_{\text{in}} - \text{height}_{\text{kernel}} + 2 \times \text{padding}}{\text{stride}} + 1
\]

\[
 \text{width}_{\text{out}} = \frac{\text{width}_{\text{in}} - \text{width}_{\text{kernel}} + 2 \times \text{padding}}{\text{stride}} + 1
\]  

(7)

Among them, \( \text{height}_{\text{out}} \) represents the height of the convolutional layer output tensor, \( \text{height}_{\text{in}} \) represents the width of the convolutional layer input tensor, \( \text{height}_{\text{kernel}} \) represents the height of the convolution kernel, \( \text{padding} \) represents the scanning mode, and the selection in this paper is valid. \( \text{stride} \) represents the number of convolution kernels, \( \text{width}_{\text{out}} \) represents the width of the output tensor, and \( \text{width}_{\text{in}}, \text{width}_{\text{kernel}} \) represents the width of the convolution layer input tensor and the width of the convolution kernel, respectively.

3.3.2. softmax layer. The full connection layer uses the softmax function, which can be used in the multi-classification process to map the inputs of multiple neurons to (0, 1), and each mapping can be viewed as an output probability for multi-classification. Assuming that the output of the previous layer is \( \{z_1, z_2, \ldots, z_n\} \), the softmax layer operation formula is as follows.

\[
 k_i = \frac{e^{z_i}}{\sum e^{z_i}}
\]  

(8)

Where \( k_i \) represents the probability that \( z_i \) represents the predicted result. Converting the output probability into an image display is a confusion matrix, which can intuitively reflect the recognition of each modulation mode.

4. Experimental results and analysis

4.1. Experimental environment

The experiment runs under Windows 7 system, the CPU is Intel(R) Core(TM) i5-4210U, without GPU participation, the memory is 12G. The deep learning library built by the network uses Keras, and the
back end selects TensorFlow. The network training is done 100 times. The EarlyStopping function is used to set the training loss function of the network to the monitoring value. When the monitoring value exceeds 5 times, the training will stop and the training result will be output. 50% of the data under the respective signal-to-noise ratios of the data sets are used as training data, and the remaining 50% of the modulated data is used as the detection data.

4.2. Recognition accuracy comparison

It can be seen from the comparison of the recognition rates from figure 4 that the recognition rate of the network proposed in this paper is higher than that of the CNN network. It can be seen from the identification confusion matrix of figure 5, figure 6, figure 7 and figure 8 that the improvement of the network recognition rate proposed in this paper is mainly due to the improvement of the recognition rate of 8PSK.

![Figure 4. Comparison of CNN-GRU and CNN2 recognition rates under RML2016.04C](image)

![Figure 5. CNN confusion matrix when SNR=0dB](image)

![Figure 6. CNN-GRU confusion matrix when SNR=0dB](image)

![Figure 7. CNN confusion matrix when SNR=18dB](image)

![Figure 8. CNN-GRU confusion matrix when SNR=18dB](image)
5. Conclusion

Based on the published data sets of RML2016.04C, this paper studies the identification of 11 modulation methods. Based on convolutional neural network and bidirectional long-term and short-term memory neural network, the proposed method establishes the CNN-GRU modulation recognition network, making full use of CNN's advantages in image processing and GRU's time-series correlation, and achieves 11 different modulation modes. The goal of the experimental results show that it greatly reduces the network complexity and the modulation recognition performance has been improved. The next step will be to study how to effectively improve the recognition performance of the algorithm when the signal-to-noise ratio is lower than -14dB.

Reference
[1] Wang F, Dobre O A, Chan C, et al. Fold-based Kolmogorov–Smirnov Modulation Classifier J. IEEE Signal Processing Letters, 2016,23(7):1003-1007.
[2] Marey M and Dobre O A. Blind Modulation Classification Algorithm for Single and Multiple-Antenna Systems Over Frequency-Selective Channels J. IEEE Signal Processing Letters, 2014, 21(9):1098-1102.
[3] Dobre O. Signal identification for emerging intelligent radios: classical problems and new challenges J. Instrumentation & Measureme-nt Magazine IEEE, 2015, 18(2):11-18.
[4] Hameed F, Dobre O A and Popescu D C. On the likelihood-based approach to modulation classification J. IEEE Transactions on Wireless Communications, 2009, 8(12):5884-5892.
[5] Abu-Romoh M, Aboutaleb A and Rezki Z. Automatic Modulation Classification Using Moments And Likelihood Maximization J. IEEE Communications Letters, 2018,1-1.
[6] Abdelbar M, Tranter W H and Bose T. Cooperative Cumulants-Based Modulation Classification in Distributed Networks J. 2018, PF(99):1-1.
[7] Wu T and Di Wa Huang G. MPSK Signal Modulation Recognition Algorithm Based on Joint Frequency Offset Estimation and Recurring Moment J. Instrument Technology and Sensors, 2014, 117 (12): 102-104.
[8] Wu H C, Saquib M and Yun Z. Novel Automatic Modulation Classification Using Cumulant Features for Communications via Multipath Channels J. IEEE Transactions on Wireless Communications, 2008, 7(8):3098-3105.
[9] Ghani N and Lamontagne R. Neural networks applied to the classification of spectral features for automatic modulation recognition C// Military Communications Conference, 1993. MILCOM ’93. Conference record. Communications on the Move. IEEE. IEEE, 1993:111-115 vol.1.
[10] Zhou H. Research on blind identification of digital modulated signals under low SNR D. University of Electronic Science and Technology of China, 2010.
[11] O’Shea T J, Corgan J and Clancy T C. Convolutional Radio Modulation Recognition Networks C// International Conference on Engineering Applications of Neural Networks. Springer International Publishing, 2016,213-226.
[12] Karra K, Kuzdeba S and Petersen J. Modulation recognition using hierarchical deep neural networks C// IEEE International Symposium on Dynamic Spectrum Access Networks. IEEE, 2017,1:1-3.
[13] Li R, Li L, Yang S, et al. Robust Automated VHF Modulation Recognition Based on Deep Convolutional Neural Networks J. IEEE Communications Letters, 2018, PP(99):1-1.
[14] Liu X , Yang D and Gamal A E . Deep Neural Network Architectures for Modulation Classification J. 2017.
[15] Li Y, Zhang T, Shan C, et al. Deep learning Convolutional neural networks from entry to proficiency M. Mechanical Industry Press, 2018: 4-5.
[16] Tensorflowai. Deep learning using Keras: (vi) LSTM and two-way LSTM explanation and practice [OL]. https://blog.csdn.net/fendouaini/article/details/80198994, 2018.12.
[17] T. J. O’Shea, https://radioml.com/datasets/radioml-2016-04-dataset/.