

Automatic Modulation Recognition Method Based on Hybrid Model of Convolutional Neural Networks and Gated Recurrent Units

Xinyu Hao,¹ Yu Luo,² Qiubo Ye,¹ Qi He,¹
Guangsong Yang,^{1,3*} and Chin-Cheng Chen^{1**}

¹School of Ocean Information Engineering, Jimei University, Xiamen, Fujian 361021, China

²Guizhou Radio and Television Bureau, Guiyang, Guizhou 550000, China

³Fujian Shipping Research Institute, Xiamen, Fujian 361021, China

(Received June 9, 2021; accepted September 16, 2021; online published November 18, 2021)

Keywords: convolutional neural networks, global average pooling, gate recurrent units, automatic modulation recognition

With the application of various wireless communication technologies, the electromagnetic environment has become more complex, and the recognition of signal modulation has become increasingly difficult. In this paper, a hybrid model based on deep learning, which aims to quickly classify received modulated signals and help to plan spectrum resources, is proposed. The model is designed by considering the characteristics of convolutional neural networks (CNNs), global average pooling (GAP), gate recurrent units (GRUs), and other structures. Firstly, signal spatial features are extracted by convolution using a CNN, the dimension of the high-dimensional feature map is reduced by GAP, then the signal temporal correlation is extracted using GRUs. Finally, modulation modes are classified in the softmax layer to classify and recognize the modulation modes of the received signal. Experimental results show that the average recognition rate of the model was 60.64% and the maximum recognition rate was 90%. The proposed method not only improves the recognition performance but also enhances the interpretability of the network.

1. Introduction

There are different modulation methods in communication technology, and the purpose of automatic modulation classification (AMC) is to detect the received signal and obtain its modulation information for further processing. AMC relies on modulation recognition (MR) technology and has been widely applied in military and civil fields, such as electronic warfare, threat analysis,⁽¹⁾ interference identification, and spectrum management.

Traditional MR methods include likelihood-based (LB) and feature-based (FB) methods.⁽²⁻⁴⁾ The LB method obtains the final classification result by calculating and comparing the likelihood function of the received signal symbols in each modulation mode. It can achieve the

*Corresponding author: e-mail: gsyang@jmu.edu.cn

**Corresponding author: e-mail: 201761000018@jmu.edu.cn

<https://doi.org/10.18494/SAM.2021.3456>

maximum probability in the theory of correct classification when channel conditions are determined.⁽⁵⁾ However, this method has disadvantages of high computational complexity and the need for prior knowledge. The FB method extracts some features from the received signal and sends them to a classifier for scoring to obtain the final classification result. In the FB method, no prior knowledge is required, but a reasonable feature extraction method and a classifier are needed, and this method has the problem of limited recognition accuracy. It is difficult to extract features manually in a complex electromagnetic environment by using traditional methods, which leads to a low recognition rate.⁽⁶⁾ Some AMC methods have been proposed to solve these problems, such as convolutional neural networks (CNNs),⁽⁷⁾ convolutional long short-term deep neural networks (CLDNNs),^(8,9) and deep residual nets (ResNet),⁽¹⁰⁾ which automatically extract feature information from the received signal and send it to a classifier to obtain the classification result.

As a branch of machine learning, deep learning (DL) technology has achieved great success in image classification and speech recognition, providing favorable conditions for the development of AMC technology. O'Shea's group first proposed a recognition model based on a CNN⁽¹¹⁾ and obtained a loss function of 0.874 and an overall recognition rate of 66.9%, which demonstrated the effectiveness of CNNs in AMC. In Ref. 12, an automatic recognition system for cognitive radio waveforms based on the Choi–Williams time-frequency distribution transform and a CNN classifier was explored, which had a high recognition rate under high-power background noise but poor multisample classification ability. In Ref. 13, a novel two-step training method was proposed and transfer learning was introduced to improve the efficiency of retraining.⁽¹³⁾ In Ref. 14, a CNN algorithm was used to extract the features of modulation from the received signal and convert them into a 2D image, realizing the classification of modulation under different SNR conditions.

The CNN is a classical AMC algorithm with a simple system structure and strong recognition ability compared with traditional methods, but it also has disadvantages of limited fitting ability and poor timeliness. Because it is difficult to further improve the performance of models by using a single network or extracting single features, researchers began to use the CNN structure combined with other technologies. Zhang *et al.* proposed an AMC feature fusion scheme using a CNN, which achieved a classification accuracy of 92.5% at a signal-to-noise ratio (SNR) of -4 dB.⁽¹⁵⁾ In Ref. 16, a novel CNN framework based on spectrum analysis (SCNN) was designed for automatic modulation recognition (AMR). The received signal was converted into a spectrum image by the short-time discrete Fourier transform and used as the input of the SCNN. This approach achieved a high recognition accuracy but required a large amount of storage space and had a low training speed. In Ref. 17, the original I/Q data and the fourth-order cumulant (FOC) were combined to represent the modulated signal, and a deep CNN network and a deep long short-term memory (LSTM) network scheme were proposed to identify the modulated signal with a recognition rate of nearly 90%.⁽¹⁷⁾ Zhang *et al.*⁽¹⁸⁾ proposed a CNN–LSTM dual-stream structure in which each stream was composed of a CNN and LSTM. The features learned from the two streams mutually interacted in pairs to increase the diversity of features and improve performance. In Ref. 19, a HybridNet scheme that combined a CNN and a bidirectional gated recurrent unit (Bi-GRU) was proposed, allowing a network to capture the long-term dependence

of a signal and improve the network performance. The purpose of this study is to find an AMR method that integrates multiple DL models. Our proposed method can extract spatial features and the temporal correlation, reduce the number of learning parameters, improve the recognition accuracy and speed, and improve the recognition rate at a low SNR.

The rest of the paper is organized as follows. Section 2 introduces the two main structures of the CNN and GRU. The CNN–GRU hybrid network model is proposed in Sect. 3. The performance of the proposed model and other similar models is analyzed in Sect. 4. Finally, conclusions and future work are presented in Sect. 5.

2. CNN and GRU

2.1 CNN

The CNN is a feed-forward neural network used in image recognition and other fields. The convolution operation provides the CNN with a local receptive field and translation invariance. A typical CNN is composed of an input layer, a convolution layer, a pooling layer, a fully connected (FC) layer, and an output layer. The convolution layer is responsible for extracting image features, the pooling layer is responsible for sampling features to highlight the main features, and the FC layer is responsible for connecting all the incoming features and sending them to the classifier. Figure 1 shows a typical multilayer CNN structure.

2.2 LSTM and GRU

2.2.1 LSTM

LSTM is a time cycle neural network used to solve long-term dependence problems that cannot be solved by a general recurrent neural network (RNN). It can also solve the gradient

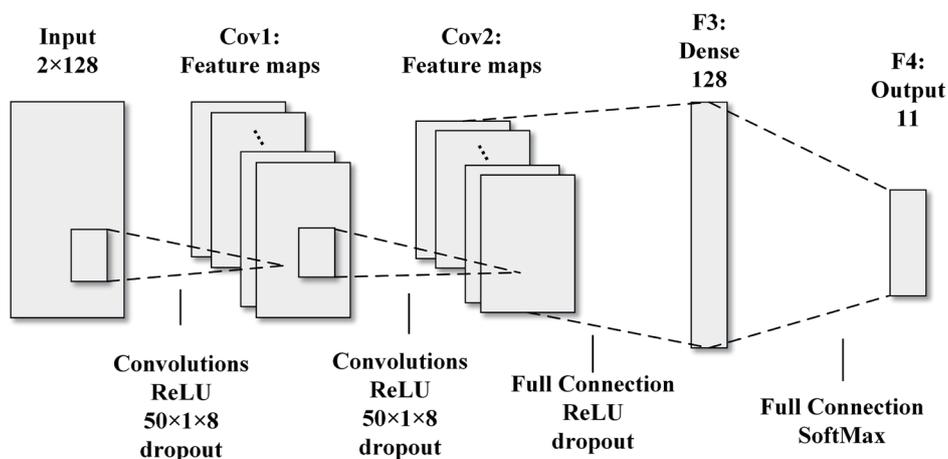


Fig. 1. CNN structure.

vanishing and gradient explosion problems. The superior performance of LSTM originates from the three “gates” in its structure, namely, the forget gate, input gate, and output gate. A typical single cell of LSTM is shown in Fig. 2. Several LSTM cells are connected to form the multilayer LSTM shown in Fig. 3.

In a cell of the LSTM network, the first step is to decide whether information from the previous timestamp should be kept or forgotten. The forget gate can filter (forget) some unimportant information from the previous cell. The equation for the forget gate is

$$f_t = s(W_f g[h_{t-1}, x_t] + b_f), \quad (1)$$

where f_t is the output of the forget gate, which is determined by the output h_{t-1} of the previous cell and the current input x_t of the current cell, and determines which data will be forgotten by the cell state. Moreover, W_f and b_f are the weight and bias, respectively.

The input gate is used to quantify the importance of the new information carried by the input and determines the information to be added to the cell state. The equations of the input gate are

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \quad (2)$$

$$\tilde{C}_t = \tanh(W_C g[h_{t-1}, x_t] + b_C), \quad (3)$$

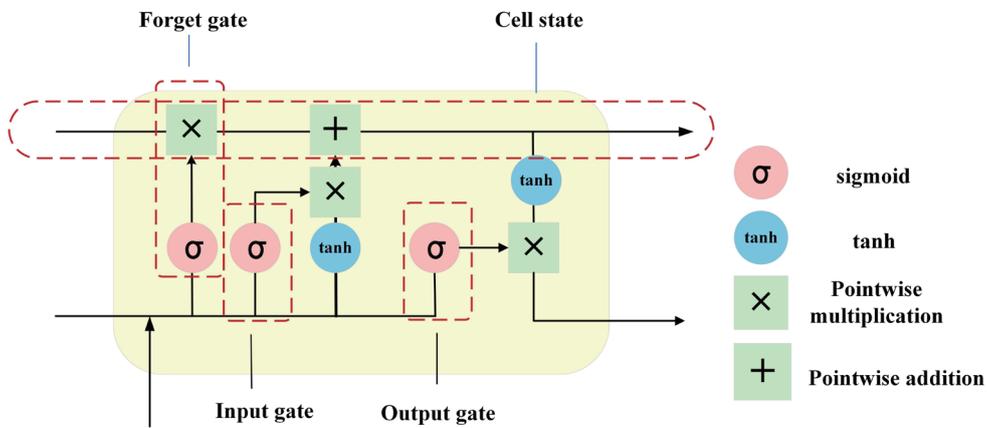


Fig. 2. (Color online) LSTM cell.

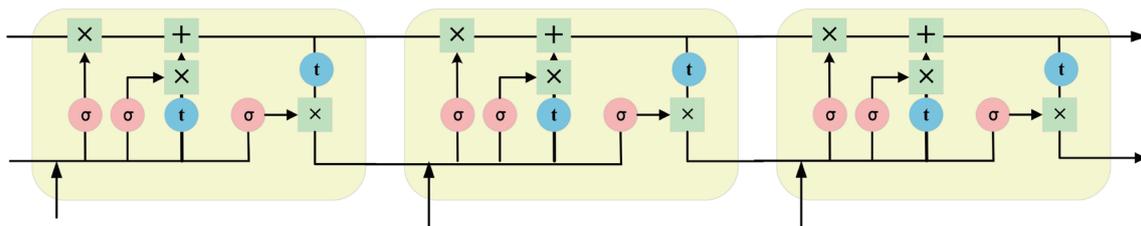


Fig. 3. (Color online) Multilayer LSTM model.

where i_t is the input of the input gate, which is determined by the output h_{t-1} of the previous cell and the input x_t of the current cell, and W_i and b_i are the weight and bias, respectively. \tilde{C}_t is the candidate cell state, in which some values will be added. W_C and b_C are the weight and offset, respectively.

Then, the cell state is updated as

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t, \quad (4)$$

where C_t is the cell state and f_t is the output of the input gate, which is used to determine the information to be forgotten by the old cell state. C_{t-1} is the old cell state, i_t is the output of the input gate, used to determine which information is input to the cell state, and \tilde{C}_t is the candidate cell state.

The output gate controls the output. The equation of the input gate is Eq. (5). The final output of the unit is determined by the output gate and cell state, as can be seen from Eq. (6).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

Here, o_t is the output of the output gate, the cell state acquires a value of -1 to 1 through the tanh function, h_t is the final output, determined by o_t and C_t , and W_o and b_o are the weight and bias, respectively.

2.2.2 GRU

A GRU is an evolved form of LSTM with a simpler structure of only two gates (Fig. 4). It combines the forget and input gates into a single update gate and uses a reset gate instead of an output gate. It also combines the cell and hidden states, so that it needs fewer training parameters

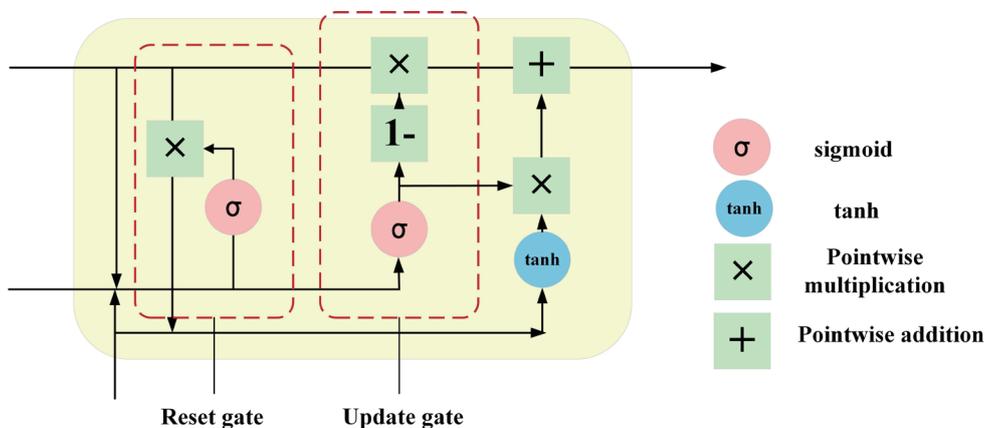


Fig. 4. (Color online) GRU.

to achieve the same effect as LSTM with faster implementation and a lower computational load.^(20,21)

The work processes of a GRU are described by Eqs. (7)–(10). The information input to the GRU can be selected as a candidate output by the reset gate, and then the update gate combines the input and candidate output to obtain the final output.

$$R_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \quad (7)$$

$$Z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \quad (8)$$

$$\tilde{H}_t = \tanh(W_h \cdot [x_t, (R_t * H_{t-1})] + b_t) \quad (9)$$

$$H_t = (1 - Z_t) * H_{t-1} + Z_t * \tilde{H}_t \quad (10)$$

Here, R_t and Z_t are the outputs of the reset and update gates, \tilde{H}_t is the candidate output, H_t is the final output of the cell, and W and b are weight and bias, respectively.

3. CNN–GRU Model for AMR

The CNN is mainly used in the field of image recognition; for wireless signal recognition, the time-domain information of the signal can also be constructed as a time-domain graph. In AMR, the CNN can learn the waveform characteristics of different modulation modes as the classification basis. It has already been demonstrated that a CNN can provide an additional 5–10% accuracy at a low SNR (<−2 dB).⁽²²⁾ Inspired by the CLDNN model, we also use a CNN as the first unit to obtain information in the proposed model to extract the features of the input signal.

Different from the CNN, a GRU is used to solve long-term dependence problems. For our AMR, the input data is the I/Q time-series signal before and after the sequences are correlated. Thus, we can find the correlation between them by using a GRU.

Global average pooling (GAP) can be used instead of the FC layer in a CNN.⁽²³⁾ Compared with an ordinary pooling layer, GAP does not take the value in the window but uses each feature map as the unit to calculate the mean value (or its window is the feature map). The difference between GAP and flattening is shown in Fig. 5. GAP can regularize the structure to prevent the model from overfitting. Compared with the FC layer, GAP makes the relationship between each category and the feature map more intuitive.⁽¹⁰⁾ At the same time, because there are no parameters in GAP to be trained or adjusted, the problem of overfitting is avoided and the training is faster.

On the basis of the complementary characteristics of the CNN and GRU, we propose an AMR network model, as shown in Fig. 6. The scheme combines a CNN with a GRU by using GAP as the connecting unit between them. This scheme reduces the dimension of the high-dimensional data output from the CNN.

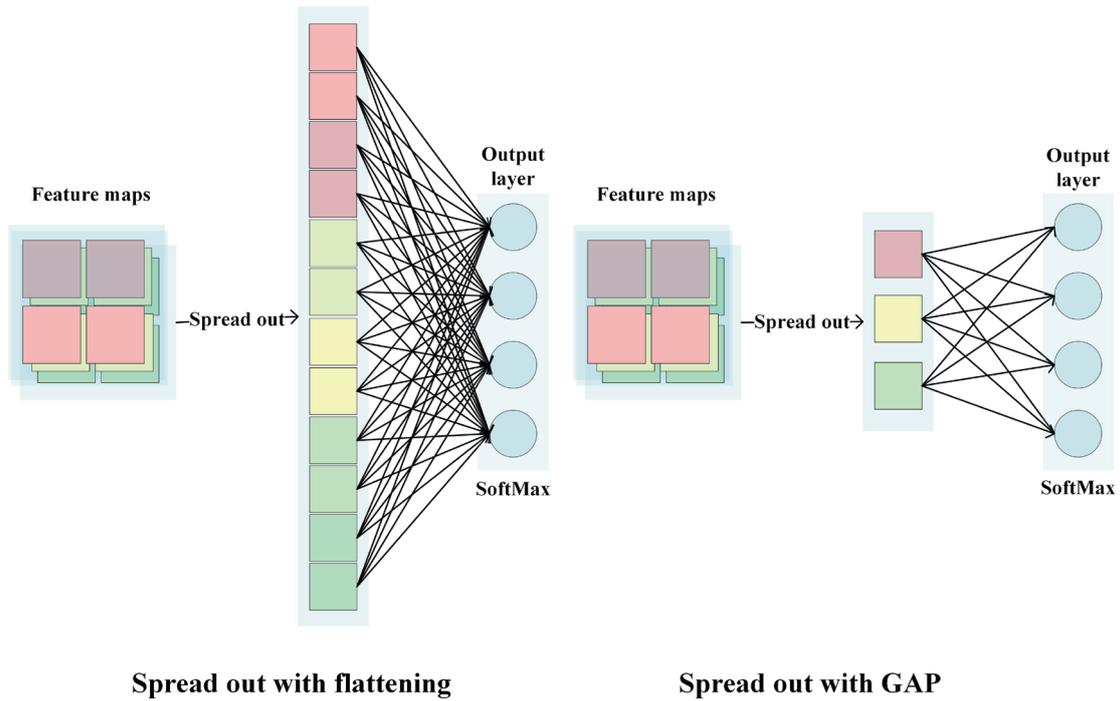


Fig. 5. (Color online) Schematic diagram of flattening and GAP operation process.

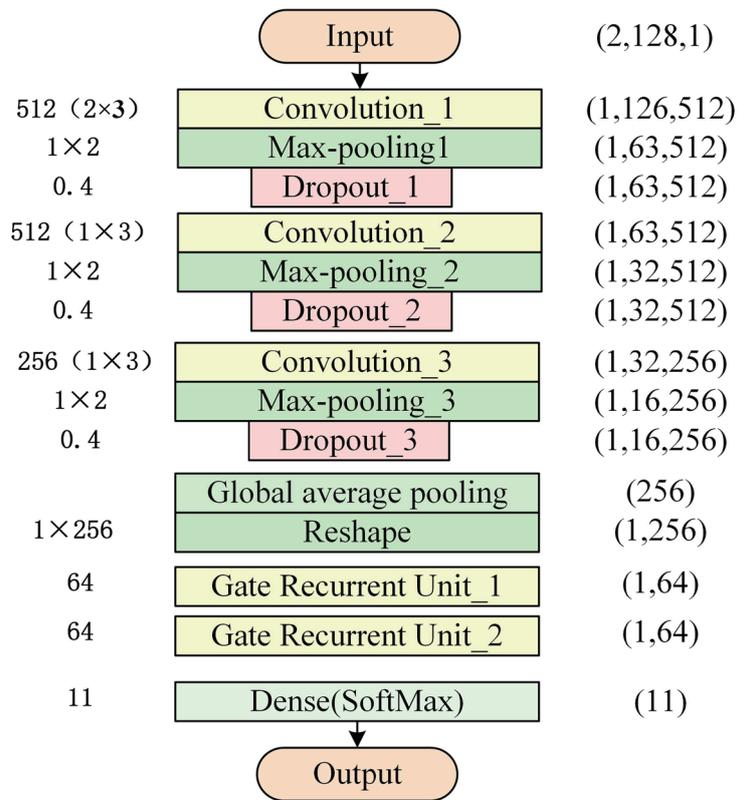


Fig. 6. (Color online) AMR network model based on CNN and GRU.

The model is composed of several parts to map the input I/Q signals into 11 modulation modes. The first part of the network consists of three convolution layers; the first layer adopts 512 filters with a size of 2×3 taps, the second layer adopts 512 filters with a size of 1×3 taps, and the third layer adopts 256 filters with a size of 1×3 taps.

Each convolution layer follows a max-pooling layer with a window of 1×2 and a dropout layer with a parameter of 0.4. The feature maps are sent from convolution and pooling to GAP, and each feature map is transformed to 2D then input into a GRU module for learning. The final results are sent to the FC layer with softmax as the activation function to classify the modulation mode of the input signal.

4. Experimental Results and Analysis

In this section, we evaluate the performance of our proposed scheme by using Keras and TensorFlow with the RadioML2016.10a and RadioML2016.04C datasets.

4.1 Experimental parameters and datasets

The RadioML2016.10a and RadioML2016.04C open datasets are provided by O'Shea's group⁽²⁴⁾ and generated by GNU Radio.⁽²⁵⁾ Several factors, such as the center frequency offset, sampling rate offset, additive Gaussian white noise, and multipath fading, are considered in these datasets to make these datasets realistic. The details of the two datasets are shown in Table 1.

We set the maximum number of iterations as 300, input 1024 pieces of data in each round, and use the early-stopping mode to avoid overfitting. If the loss function does not decrease after 12 consecutive training trials, the training is stopped.

4.2 Baseline

In this section, we introduce the baseline in our experiment.

1) We use the RML2016.10a dataset combined with the one-layer GRU scheme shown in Fig. 7, which was proposed by Hong *et al.*,⁽²⁶⁾ which consists of a one-layer GRU and two dense FC

Table 1
Contents of experimental datasets.

	RML2016.10a	RML2016.04C
Modulation modes	11 classes (WBFM, AM-SSB, AM-DSB, BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, and PAM4)	11 classes (WBFM, AM-SSB, AM-DSB, BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, and PAM4)
Length per sample	128	128
Number of samples	220000	162060
Signal dimension	2×128 per sample	2×128 per sample
Duration per sample	128 μ s	128 μ s
Sampling frequency	1 MHz	1 MHz
Samples per symbol	8	8
SNR (dB)	-20:2:18	-20:2:18

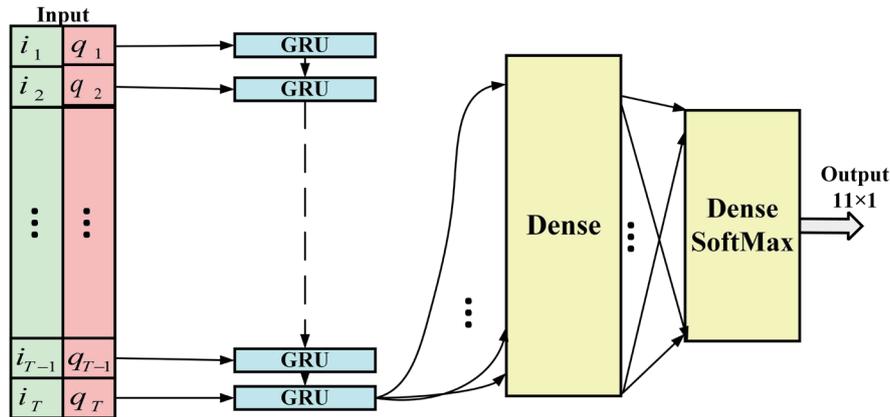


Fig. 7. (Color online) One-layer GRU network structure.

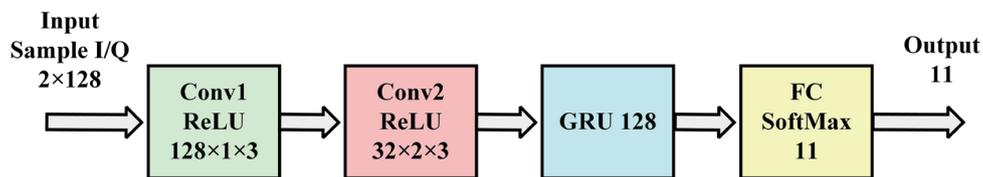


Fig. 8. (Color online) CNN-GRU network model.

layers. The GRU layer uses the ReLU activation function, and the final dense layer is the softmax output layer, which is used to calculate the classification probability of each modulation to achieve classification. RMSprop is used as the optimization algorithm of gradient descent.

2) We use the RML12016.04C dataset combined with the CNN-GRU scheme shown in Fig. 8, which was proposed by Tan *et al.*,⁽²⁷⁾ which consists of two convolution layers, one GRU layer, and one FC layer. The convolution layers use ReLU as the activation function and the FC layer uses softmax as the activation function to classify the received modulation signals. The convolution and GRU layers use dropout to avoid overfitting. Adam is used as the optimization algorithm of gradient descent.

4.3 Accuracy analysis

To evaluate the effectiveness of the proposed model, we compare the performance characteristics of different DL models and discuss the effects of different structures and parameters. The input signal is represented by the complex time-domain vector (baseband I/Q data) in the baseline scheme in Sect. 4.1 to reduce the computational complexity.

We compared the performance characteristics of our proposed model with those of five other models [ResNet,⁽²⁸⁾ Inception,⁽²⁸⁾ DenseNet,⁽²⁹⁾ CLDNN,⁽²⁸⁾ and CNN⁽¹¹⁾] that use a CNN and an LSTM-RNN. The experimental results in Fig. 9 show that our proposed model starts to outperform other models when SNR > -8 dB, the recognition rate is nearly 80% when SNR = -2 dB, and the maximum recognition rate of 89.65% is reached when SNR = 14 dB,

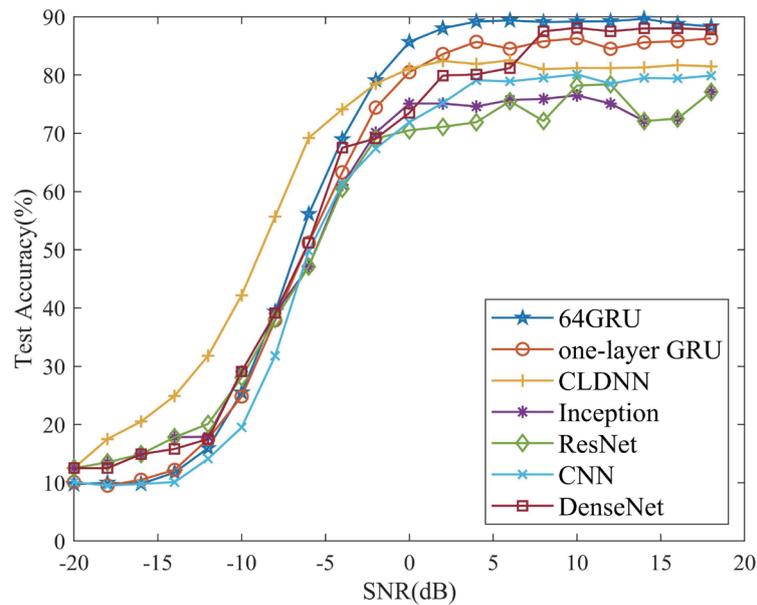


Fig. 9. (Color online) Comparison of test accuracies of different methods.

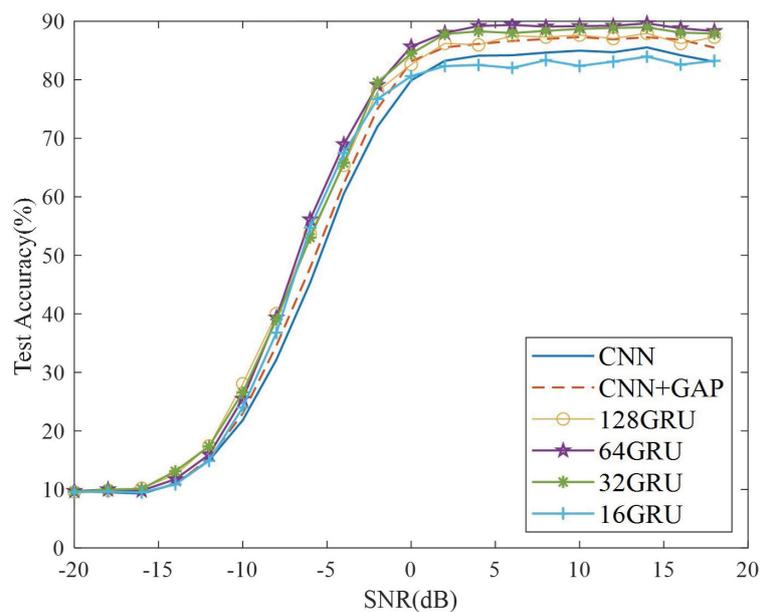


Fig. 10. (Color online) Accuracy with different numbers of units (one-layer GRU).

which is 3.5 and 10 percentage points higher than those of the baseline and original CNN models, respectively.

To obtain better performance, we studied the effects of different structures and parameters on the model performance. In Fig. 10, we can see that when the number of GRUs is greater than 32, the accuracy is better than CNN.

In Fig. 11, we first fixed the number of GRUs in the first layer to 64 and 128, then compared the recognition rate in different GRUs. The results show that the accuracy is low when using a

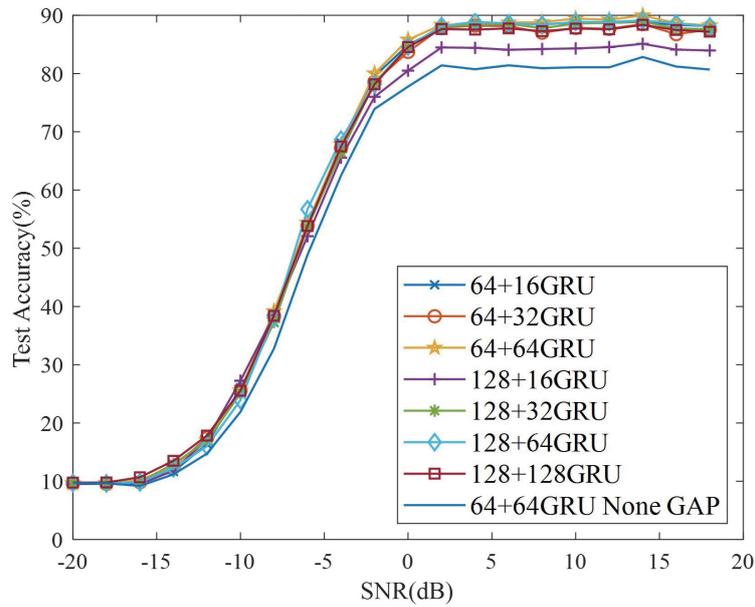


Fig. 11. (Color online) Accuracy with different numbers of units (two-layer GRU).

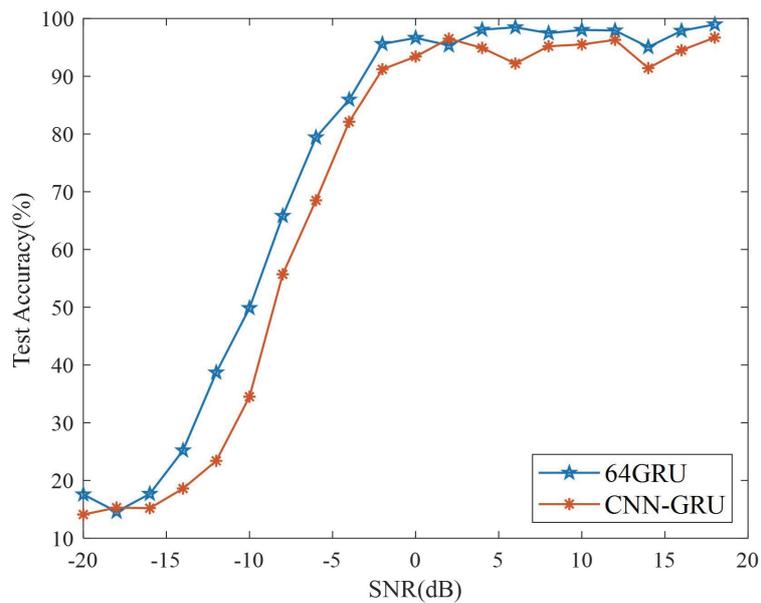


Fig. 12. (Color online) Test accuracy with different SNRs.

small number of GRUs (16 units), even if a multilayer GRU is used. When we add GAP, the accuracy improves by 3–7 percentage points. When the GRU structure with two layers and 64 units is used, the model achieves an average recognition rate of 60.64% and a maximum recognition rate of 90%.

In Fig. 12, we compare our proposed model with the model in Ref. 27 for the RML2016.04C dataset.⁽²⁷⁾ The experimental results show that the proposed model achieves better results at both high and low SNRs.

4.4 Confusion matrix analysis

We analyze the classification confusion matrix of several important SNR nodes in the network model training with 64 GRUs in one layer shown in Fig. 13.

We found that when the SNR is low (-20 dB), the model classifies all the modulation modes as AM-SSB. In this case, noise and other interference factors are mainly parts of the received signal, which makes the received signal close to Gaussian noise. The amplitude of the SSB modulation mode is lower than that of the DSB modulation mode in the time domain and is closer to Gaussian noise and similar to AM-SSB in the time domain. When $\text{SNR} > -8$ dB, a clear diagonal line is gradually formed in the confusion matrix. When $\text{SNR} > -2$ dB, a higher recognition rate can be achieved, and when $\text{SNR} = 14$ dB, the recognition rate is the highest. In Figs. 13(c) and 13(d), it is difficult to distinguish QAM16 and QAM64. This is because QAM16 and QAM64 essentially belong to the same modulation mode, and the latter includes the former

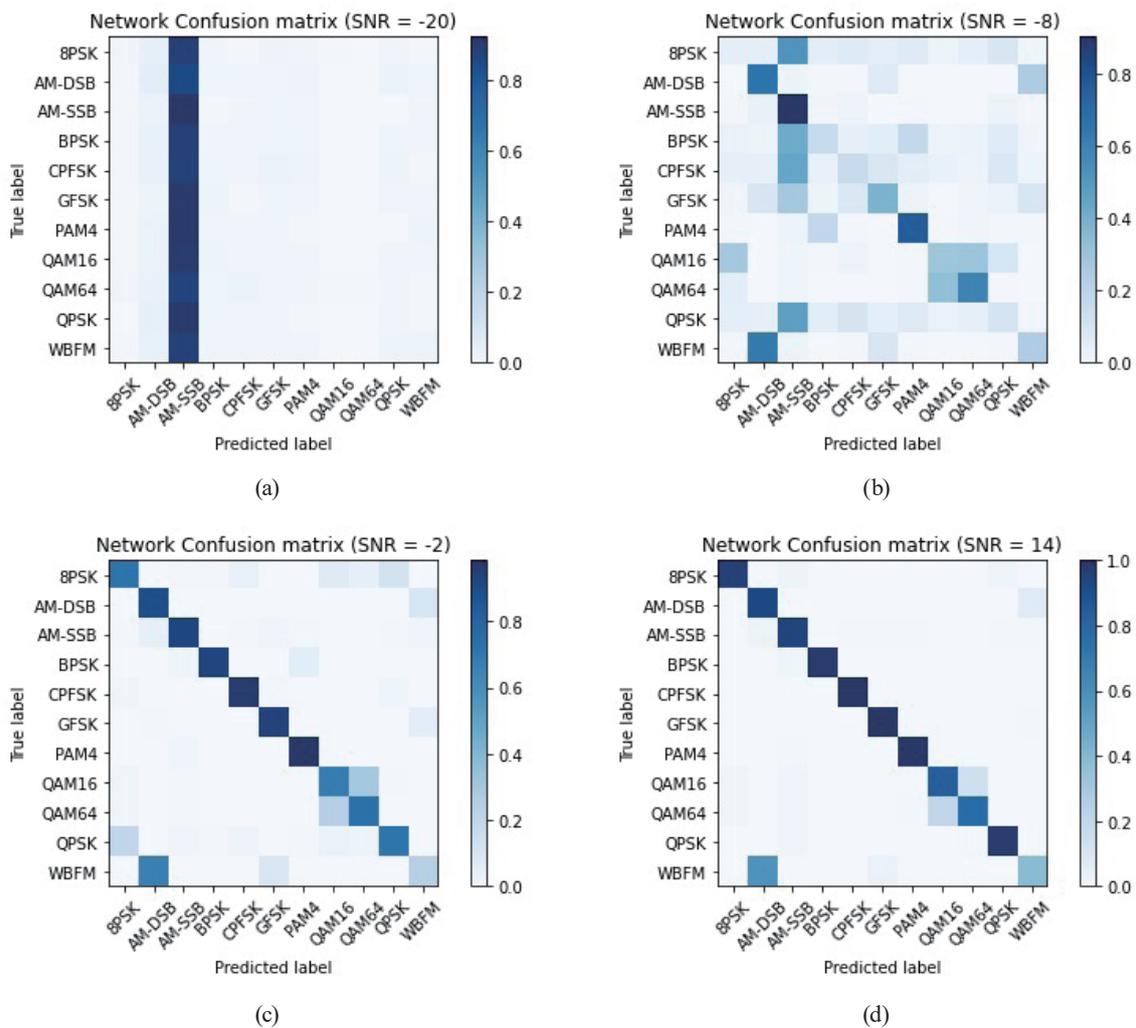


Fig. 13. (Color online) Confusion matrix of one-layer 64-unit GRU model.

in the constellation diagram. BPSK and WBFM are also difficult to distinguish owing to the received signal having a period of silence.

5. Conclusions

In this paper, we proposed an AMR model that uses different structures to extract and classify features with different dimensions automatically. To evaluate the performance of the proposed model, we automatically identified 11 common signal modulation modes in the public data provided by Ref. 24 and compared them with some existing methods. The experimental results show that the highest classification accuracy was improved from 80 to 90% and the comprehensive recognition rates on the RML2016.10a and RML2016.04C datasets were 60.64 and 73.2%, respectively.⁽²⁴⁾ We also explored the differences between the structure and parameters of the model.

With the development of computer and communication technologies, many wireless devices with different types of modulation will coexist in the same limited frequency band, and the electromagnetic environment will become more complex. To effectively manage frequency resources, it is necessary to identify signals in air. Wireless sensor networks can form a distributed monitoring network to detect, identify, monitor, and locate these signals to improve the recognition ability of a single sensor node. Our proposed method can be applied to such a sensor node by reducing the calculation time without decreasing the accuracy.

In our future work, we will continue to improve the performance of the model while also simplifying it as much as possible, so that the model requires fewer training parameters and has a higher speed. We also plan to find an adaptive model so that the appropriate method for recognition is automatically chosen according to the SNR to achieve a higher overall recognition rate.

Acknowledgments

The work was partially supported by the Natural Science Foundation of Fujian Province (No. 2021J01865 and No. 2021J01866) and the Open Project Fund of Fujian Shipping Research Institute.

References

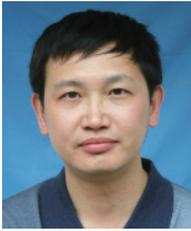
- 1 L. Liu and J. Xu: Proc. 2006 Int. Conf. Wireless Communications, Networking and Mobile Computing (IEEE, 2006) 1–5. <https://doi.org/10.1109/WiCOM.2006.157>
- 2 O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su: IET Commun. **1** (2007) 137. <https://doi.org/10.1049/iet-com:20050176>
- 3 T. R. Kishore and K. D. Rao: IEEE Trans. Aerosp. Electron. Syst. **53** (2017) 901. <https://doi.org/10.1109/TAES.2017.2667142>
- 4 H. Abuella and M. K. Ozdemir: Can. J. Electr. Comput. Eng. **39** (2016) 203. <https://doi.org/10.1109/CJECE.2016.2570250>
- 5 F. Hameed, O. A. Dobre, and D. C. Popescu: IEEE Trans. Wireless Commun. **8** (2009) 5884. <https://doi.org/10.1109/TWC.2009.12.080883>
- 6 Y. Xu, D. Li, Z. Wang, Q. Guo, and W. Xiang: Wireless Networks **25** (2019) 3735. <https://doi.org/10.1007/s11276-018-1667-6>

- 7 S. Zheng, P. Qi, S. Chen, and X. Yang: IEEE Access 7 (2019) 66496. <https://doi.org/10.1109/ACCESS.2019.2918136>
- 8 T. N. Sainath, O. Vinyals, A. Senior, and H. Sak: Proc. 2015 IEEE Int. Conf. Acoustics, Speech and Signal Processing (IEEE, 2015) 4580–4584. <https://doi.org/10.1109/ICASSP.2015.7178838>
- 9 X. Liu, D. Yang, and A. E. Gamal: Proc. 2017 51st Asilomar Conf. Signals, Systems, and Computers (IEEE, 2017) 915–919. <https://doi.org/10.1109/ACSSC.2017.8335483>
- 10 S. Huang, R. Dai, J. Huang, Y. Yao, Y. Gao, F. Ning, and Z. Feng: IEEE Internet Things J. 7 (2020) 7795. <https://doi.org/10.1109/JIOT.2020.2991052>
- 11 T. J. O'Shea, J. Corgan, and T. C. Clancy: Proc. Int. Conf. Engineering Applications of Neural Networks (Springer, 2016) 213–226. https://doi.org/10.1007/978-3-319-44188-7_16
- 12 M. Zhang, M. Diao, and L. Guo: IEEE Access 5 (2017) 11074. <https://doi.org/10.1109/ACCESS.2017.2716191>
- 13 F. Meng, P. Chen, L. Wu, and X. Wang: IEEE Trans. Veh. Technol. 67 (2018) 10760. <https://doi.org/10.1109/TVT.2018.2868698>
- 14 J. H. Lee, K. Kim, and Y. Shin: Proc. 2019 Int. Conf. Artificial Intelligence in Information and Communication (IEEE, 2019) 1–4. <https://doi.org/10.1109/ICAIIIC.2019.8669002>
- 15 Z. Zhang, C. Wang, C. Gan, S. Sun, and M. Wang: IEEE Trans. Signal Inf. Process. Networks 5 (2019) 469. <https://doi.org/10.1109/TSIPN.2019.2900201>
- 16 Y. Zeng, M. Zhang, F. Han, Y. Gong, and J. Zhang: IEEE Wireless Commun. Lett. 8 (2019) 929. <https://doi.org/10.1109/LWC.2019.2900247>
- 17 M. Zhang, Y. Zeng, Z. Han, and Y. Gong: Proc. 2018 IEEE 19th Int. Workshop on Signal Processing Advances in Wireless Communications (IEEE, 2018) 1–5. <https://doi.org/10.1109/SPAWC.2018.8446021>
- 18 Z. Zhang, H. Luo, C. Wang, C. Gan, and Y. Xiang: IEEE Trans. Veh. Technol. 69 (2020) 13521. <https://doi.org/10.1109/TVT.2020.3030018>
- 19 R. Lin, W. Ren, X. Sun, Z. Yang, and K. Fu: IEEE Access 8 (2020) 130314. <https://doi.org/10.1109/ACCESS.2020.3009471>
- 20 R-NET: Machine Reading Comprehension with Self-matching Networks: <http://www.microsoft.com/en-us/research/wp-content/uploads/2017/05/r-net.pdf> (accessed November 2020)
- 21 Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling: <https://arxiv.org/pdf/1412.3555> (accessed November 2020).
- 22 S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin: IEEE Trans. Cognit. Commun. Networking 4 (2018) 433. <https://doi.org/10.1109/TCCN.2018.2835460>
- 23 Network In Network: <https://arxiv.org/pdf/1312.4400> (accessed November 2020).
- 24 T. J. O'Shea, J. Corgan, and T. C. Clancy: Proc. 2016 1st Int. Workshop on Sensing, Processing and Learning for Intelligent Machines (IEEE, 2016) 1–5. <https://doi.org/10.1109/SPLIM.2016.7528397>
- 25 T. O'Shea and N. West: Proc. 6th GNU Radio Conf. (2016). <https://pubs.gnuradio.org/index.php/grcon/article/view/11/10>
- 26 D. Hong, Z. Zhang, and X. Xu: Proc. 2017 3rd IEEE Int. Conf. Computer and Communications (IEEE, 2017) 695–700. <https://doi.org/10.1109/CompComm.2017.8322633>
- 27 J. Tan, L. Zhang, Z. Zhong, and W. Yu: J. Phys. Conf. Ser. 1284 (2019) 012052. <https://doi.org/10.1088/1742-6596/1284/1/012052>
- 28 N. E. West and T. O'Shea: Proc. 2017 IEEE Int. Symp. Dynamic Spectrum Access Networks (IEEE, 2017) 1–6. <https://doi.org/10.1109/DySPAN.2017.7920754>
- 29 H. Ma, G. Xu, H. Meng, M. Wang, S. Yang, R. Wu, and W. Wang: IEEE Access 8 (2020) 78923. <https://doi.org/10.1109/ACCESS.2020.2988727>

About the Authors



Xinyu Hao received his B.E. degree from Ludong University, China, in 2019. He has been pursuing an M.E. degree in the Department of Information and Communication Engineering, Jimei University, Xiamen, China, since 2019. His major research interests include wireless communication, deep learning, and automatic modulation recognition. (xyhao@jmu.edu.cn)



Yu Luo is currently an engineer of the 645# wireless transmitting station of Guizhou Radio and Television Bureau. His current research interests include wireless communication and signal processing.



Qiubo Ye is a chair professor in the School of Information Engineering, Jimei University, Fujian, China. He received his Ph.D., Master's, and Bachelor degrees in electrical engineering from the University of Manitoba, Canada; North China Electric Power University; and Hefei University of Technology, China, respectively. He was a research scientist/project leader at Communications Research Centre, Canada; a visiting assistant professor at Rose-Hulman Institute of Technology, USA; and an R&D engineer at Zeland Software, Inc., USA. He was an adjunct research professor in the Electronics Department, Carleton University, Canada. His research interests include computational electromagnetics, antennas, EMC/EMI, wave propagation, wireless communication, and applications of AI in communications.



Qi He has been studying as a postgraduate in the School of Information Engineering, Jimei University, since 2019. Her main research interest is in modern communication technology.



Guangsong Yang obtained his Ph.D. degree from the School of Information Technology, Xiamen University. He was a visiting scholar at UC Davis, USA, from 2009 to 2010 and at Griffith University, Australia, from 2017 to 2018. He is currently a professor in the School of Information Engineering at Jimei University, Xiamen, China. His current research interests include wireless sensor networks and security and intelligent information processing. (gsyang@jmu.edu.cn)



Chin-Cheng Chen has been a professor at Jimei University, China, since 2017. He became a member of IEEE in 2011 and a senior member in 2016. He earned his M.S. degree in 2005 and recently earned his Ph.D. degree from the Department of Mechatronics Engineering, National Changhua University of Education. He has published 43 academic articles, some of which were published in IEEE Access, MPDI, etc., and owns three patents. His research interests include AIoT technology, machine learning, information security, and RFID applications. (201761000018@jmu.edu.cn).