IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ARTIFICIAL INTELLIGENCE FOR PHYSICAL-LAYER WIRELESS COMMUNICATIONS

Received May 6, 2019, accepted May 14, 2019, date of publication May 22, 2019, date of current version June 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2918136

Fusion Methods for CNN-Based Automatic Modulation Classification

SHILIAN ZHENG[1], (Member, IEEE), PEIHAN QI[2], SHICHUAN CHEN[1], AND XIAONIU YANG[1]

[1]Science and Technology on Communication Information Security Control Laboratory, Jiaxing 314033, China
[2]State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China

Corresponding author: Xiaoniu Yang (yxn2117@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61871398, and in part by the China Postdoctoral Science Foundation under Grant 2018M631122.

ABSTRACT An automatic modulation classification has a very broad application in wireless communications. Recently, deep learning has been used to solve this problem and achieved superior performance. In most cases, the input size is fixed in convolutional neural network (CNN)-based modulation classification. However, the duration of the actual radio signal burst is variable. When the signal length is greater than the CNN input length, how to make full use of the complete signal burst to improve the classification accuracy is a problem needs to be considered. In this paper, three fusion methods are proposed to solve this problem, such as voting-based fusion, confidence-based fusion, and feature-based fusion. The simulation experiments are done to analyze the performance of these methods. The results show that the three fusion methods perform better than the non-fusion method. The performance of the two fusion methods based on confidence and feature is very close, which is better than that of the voting-based fusion.

INDEX TERMS Modulation classification, deep learning, fusion, convolutional neural network, residual network, wireless communications, cognitive radio.

I. INTRODUCTION

Automatic modulation classification has a wide range of applications in wireless communications [1], [2]. For example, in software-defined radio-based communication [1]–[3], by identifying the modulation type of the received signal, the receiver can demodulate the signal by using the corresponding demodulation algorithm. In this way, the transmission signal does not need to include additional control information for informing the receiver of the modulation type it used, which is beneficial to reduce the protocol overhead. In cognitive radio [4], [5], automatic modulation classification can be used to assist in detecting the primary user signal. When the primary user is discovered, the cognitive user can vacate the current channel, thereby avoiding harmful interference to the primary user. In spectrum monitoring [6], [7], by identifying signals in a wide frequency band, interference signals or illegal users can be found, so that measures can be taken to ensure the security of wireless communications.

Traditional automatic modulation classification methods can be divided into two categories: likelihood-based methods and feature-based methods [2]. The likelihood-based methods [8]–[12] calculate the likelihood function of the received signal and compare it with a certain threshold to make decision. Although the likelihood-based method can minimize the error rate, its computational complexity is high, and it cannot adapt to unknown channel conditions and mismatch between transmitter and receiver (such as clock frequency deviation) [2]. The feature-based methods [13]–[17] calculate certain features of the received signal, such as mean, standard deviation and kurtosis of the normalized centered amplitude, absolute normalized instantaneous frequency, higher order moments, higher order cumulants, cyclic moments, cyclic cumulants of the received signal. The computational complexity of these methods is relatively low, but the selection of features relies too much on manual analysis. It is very difficult to find features that can adapt to non-ideal conditions and distinguish between multiple modulation types. Therefore, automatic modulation classification is a very challenging task, especially in non-cooperative scenarios where we do not have prior information on the received signals [1].

In recent years, deep learning has benefited from neural network units such as multiple hidden layers and nonlinear

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

activation, and has been able to learn more deep-level information hidden in the data, showing excellent performance in many tasks, such as image classification, machine translation, and natural language processing [18]–[23]. In recent years, deep learning was used for several areas in wireless communications [24] and radio signal processing [6], such as beamforming [25], channel estimation [26]–[29], RF fingerprinting [30],[31], non-orthogonal multiple access (NOMA) scheme [32], [33], sparse signal regulation and recovery [34], [35], resource allocation [36], and localization [37], [38]. In modulation classification, deep learning has been adopted and better performance than feature engineering-based methods has been obtained. For example, [39] used convolutional neural networks (CNNs) for modulation recognition, and experimental results show that it obtains performance close to feature-based expert systems. To further improve performance, the deep residual network and long short time memory (LSTM) are used for modulation classification, respectively [40], [41]. As expected, these methods yield better classification accuracy than the simple CNN and the traditional feature-based methods. Furthermore, the authors in [42] used two CNN models, AlexNet and GoogLeNet, for modulation classification. The authors in [43] compared the performance of various neural network models in modulation classification and reduced the training complexity from the perspective of reducing the input signal dimension. The authors in [44] proposed a two-step training to improve the efficiency of training CNN for modulation classification. The authors in [45] proposed a deep learning-based method, combined with two CNNs trained on different datasets, to achieve higher accuracy. In addition, generative adversarial nets (GANs) were also considered for modulation classification [46]–[48]. Overall, the biggest difference between the modulation classification method based on deep learning and the traditional modulation classification methods is that features are automatically learned by the adopted neural network, which avoids the feature design process and can adapt to the requirements under non-cooperative scenario.

As can be seen from the above, CNN is a neural network model widely used in modulation classification. In most cases, in CNN-based modulation classification methods, the signal input size supported by the CNN is often fixed. However, the duration of the actual radio signal burst is uncertain. When the signal length is greater than the CNN input length, how to make full use of the complete signal burst to improve the classification accuracy is a problem to be considered. In this context, this paper proposes several fusion methods to solve this problem. The signal is first segmented and then fused based on the classification results, confidence, or intermediate features of these segmented signals. Specifically, the contributions of the paper are as follows:

- Three fusion methods for CNN-based modulation classification are presented.
  (1) Voting-based fusion, that is, voting on the classification results, and the category with the largest number of votes is used as the final classification result.
  (2) Confidence-based fusion, that is, linear fusion of the confidence vector, and the category with the highest average confidence is used as the classification result.
  (3) Feature-based fusion, that is, the input features of the softmax layer is linearly fused, and the category with the highest average value is taken as the classification result.
- The performance of the fusion methods are analyzed through simulation experiments. In the experiments, the performance of different signal segmentation overlapping ratios, different fusion methods, and different signal lengths are compared and analyzed.

The rest of this paper is organized as follows. In Section II, we introduce the CNN-based automatic modulation classification by presenting two CNN structures. In Section III, we present three fusion methods in detail. In Section IV we discuss the simulation results, and finally in Section V we summarize the paper.

II. CNN-BASED MODULATION CLASSIFICATION

A. PROBLEM MODEL

Modulation classification can be expressed as a classification problem with M modulations. The received signal can be expressed as

$$r(t) = \alpha(t)e^{j(2\pi f_o t+\theta_o(t))}s(t) + n(t),$$

where $s(t)$ is the complex baseband envelope of the transmitted signal, $\alpha(t)$ is the impulse response of the transmitted wireless channel, $\theta_o(t)$ and f_o are the carrier phase and frequency offsets due to disparate local oscillator and Doppler effect caused by motion, $n(t)$ is the additive white Gaussian noise (AWGN). The aim of any modulation classifier is to give out $\Pr(s(t) \in M(i)|r(t))$ given the received signal $r(t)$, where $M(i)$ represents the i-th modulation. For simplicity, the received signal is usually represented by its in-phase and the quadrature (IQ) components. The real and imaginary parts of $r(t)$ represent the IQ components, respectively. The subsequent modulation classification is based on this IQ representation.

B. CNN-BASED METHODS

CNNs [49] are a special type of artificial neural network designed primarily for computer vision applications. Compared with traditional neural networks, CNN has two distinct features. First, based on the assumption that adjacent pixels of the image are highly correlated, the CNN uses weight sharing, and the same filter is used for the input to obtain the output of the corresponding channel. Second, the CNN uses a pooling operation to give it a degree of translation invariance and reduces the computational complexity of deeper layers by downsampling.

Most CNN structures are inspired by LeNet [50]. Classical CNNs often contain four basic layers: convolutional

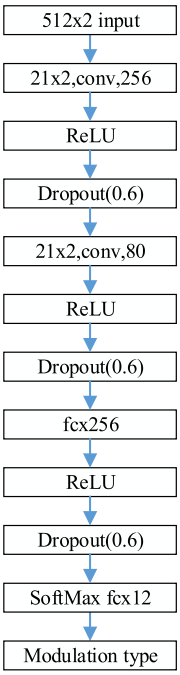


FIGURE 1. CNN layout (CNN1). "conv" stands for convolutional layer; the number before "conv" indicates the size of the convolution kernel, and the following number indicates the number of convolution kernels; "ReLU" indicates rectification linear activation; "Dropout" indicates the Dropout layer, the number in parentheses after that indicates the probability of Dropout; "fc" indicates the full connection Layer, the number after that represents the number of neurons; "SoftMax" represents the SoftMax layer; the final output is the category, and the category label is One-Hot encoded. A batch normalization layer is also included between the convolutional layer and the non-linear activation layer, which is not shown in the figure for the sake of simplicity.

layer, normalized layer, nonlinear activation layer and pooling layer. This paper considers two CNN structures. The first CNN structure is similar to the CNN in [39], [40]. The network structure is shown in Fig. 1 (hereinafter referred to as CNN1), which combines the IQ components of the received signal into two columns of a matrix as the input of the CNN. The number of signal sampling points is fixed at 512.

The second CNN structure considered in this paper is the residual network structure similar as the resnet used in [40] but with much more layers. As the traditional convolutional neural network going deeper, the training accuracy will gradually become saturated and then deteriorate. As the number of layers is further increased, the training error will become larger and larger, that is, the degradation problem will occur. Residual network is proposed in [51] to solve the training problem of deep network. Let $H(\mathbf{x})$ denote an intrinsic map of a series of cascaded layers that need to be fitted, where $\mathbf{X}$ is the input to the first of these layers. If it is assumed that multiple nonlinear layers can approximate complex functions, then it is equivalent to assuming that these nonlinear layers can approximate the residual function $H(\mathbf{x}) - \mathbf{x}$ (assuming the input and output have the same dimensions). Therefore, unlike the way in which conventional methods expect these layers to approximate function $H(\mathbf{x})$, the residual network expects these layers to approximate the residual function $F(\mathbf{x}) = H(\mathbf{x}) - \mathbf{x}$. The original function becomes $H(\mathbf{x}) = F(\mathbf{x}) + \mathbf{x}$. Although the two forms are equivalent, the learning difficulty will be very different. Consider the building blocks defined below:

$$\mathbf{y} = F(\mathbf{x}, \{w_i\}) + \mathbf{x},$$

where $\mathbf{x}$ and $\mathbf{y}$ represent the input and output. An implementation structure is shown in Fig. 2, which is completed by direct connection and addition.

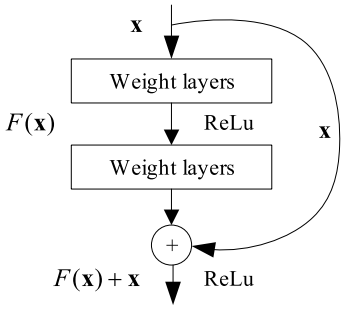


FIGURE 2. The residual module [51].

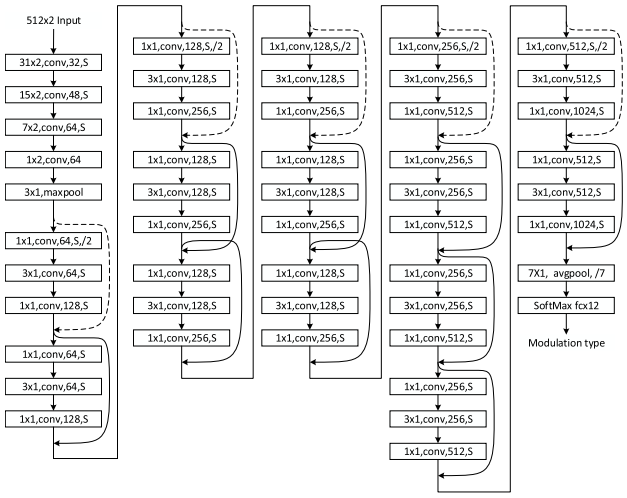


FIGURE 3. Deep residual network layout (CNN2). In the figure, "conv" represents the convolutional layer; the number before "conv" represents the size of the convolution kernel, and the following number indicates the number of convolution kernels; "S" indicates that the convolution contains padding so that the input and output are of the same size, and "/2" indicates that the downsampling factor is 2, which means the output size is reduced to half of the input size; "maxpool" represents the maximum pooling; "avgpool" represents the average pooling; "/7" indicates that the downsampling factor is 7; "fc" represents the fully connected layer, and the number after that represents the number of neurons; "SoftMax" represents the SoftMax layer; the final output is the category. All activations are using ReLU. A batch normalization layer is also included between the convolutional layer and the ReLU layer, which is not shown in the figure for the sake of simplicity.

Based on the residual module, the second CNN structure used in this paper is shown in Fig. 3 (hereinafter referred to as CNN2).

The networks are trained using a cross entropy loss function, i.e.,

$$\ell = -\frac{1}{S}\sum_{i=1}^{S}\log\left(o_{M(i)}\right),$$

where S represents the number of training samples and $o_{M(i)}$ represents the predicted probability that the ith sample belongs to class $M(i)$.

III. FUSION METHODS

A. SIGNAL SEGMENTATION

As indicated earlier, the input length of the convolutional neural network is often fixed, and the length of the signal to be classified may be much larger than this length. In order to make full use of the signal information and improve the classification accuracy, the signal can be divided into multiple segments of length L. Each segment is sent to the CNN, and the final decision result is obtained by the corresponding fusion method. Let the signal sequence be $x(n)$ of length N. One of the segmentation methods is to slide at interval P, and select each signal segment of length L, thereby obtaining

$$y_i(m) = x((i-1)P + m),$$

where $m = 0, 1, 2, \ldots, L-1$, $i = 1, 2, \ldots, \lfloor (N-L+1)/P \rfloor$, and $\lfloor a \rfloor$ denotes the largest integer not greater than a. For convenience, let $I = \lfloor (N-L+1)/P \rfloor$. The CNN takes $y_i(m)$ as input for modulation classification, and the final classification decision result is obtained according to the fusion method. This paper presents three methods of fusion: voting-based fusion, confidence-based fusion, and feature-based fusion.

B. FUSION METHODS

The fusion methods presented in this paper consider the output of different layers in CNN. As shown in Fig. 4, different signal segments are input to the CNN, and different outputs are obtained at these layers. By fusing the classification results, confidence vectors or features, we can obtain the corresponding fusion methods. These three methods will be described in detail next.

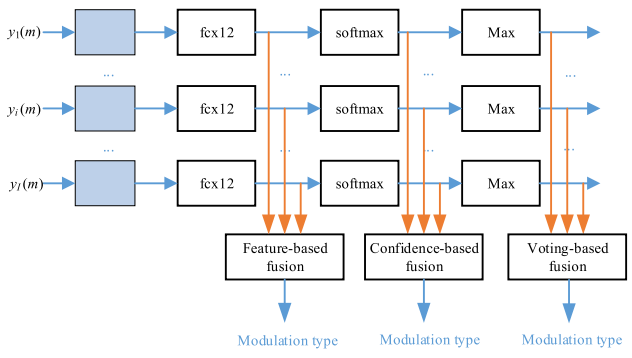


FIGURE 4. Schematic diagram of fusion methods.

1) VOTING-BASED FUSION

Voting is a common fusion mechanism that uses the principle of "majority wins". We use $y_i(m)$ as the input of the convolutional neural network and we denote the classification results as $o_i \in [1, 2, \ldots, M]$. According to o_i, we get

$$v_{ij} = \begin{cases} 1, & \text{if } o_i = j \\ 0, & \text{else} \end{cases}$$

Then the number of times the classification result is the j-th modulation is obtained as

$$c_j = \sum_{i=1}^{I} v_{ij}.$$

The final decision is made as

$$r_{\text{Voting}} = \arg\max_{1\le j\le M} c_j.$$

2) CONFIDENCE-BASED FUSION

Confidence-based fusion uses the confidence vector for linear fusion. First, input $y_i(m)$ into the convolutional neural network to obtain a confidence vector (the output of the softmax layer of the CNN) as $\boldsymbol{p}_i = [p_{i,1}, p_{i,2}, \ldots, p_{i,M}]$, where $p_{i,j}$ represents the confidence that $y_i(m)$ is classified as the j-th modulation type. Then calculate the average value of the confidence of the j-th modulation type,

$$\bar{t}_j = \sum_{i=1}^{I} p_{i,j}.$$

Finally, the subscript corresponding to the maximum value of $\bar{t}_j$ is obtained as the final modulation classification fusion result:

$$r_{\text{Confidence}} = \arg\max_{1\le j\le M} \bar{t}_j.$$

3) FEATURE-BASED FUSION

Feature-based fusion uses the activations of a specific layer for linear fusion. The feature layer considered in this paper is the previous layer of softmax. First, the feature vector (the input of the softmax layer of the CNN) is obtained as $\boldsymbol{g}_i = [g_{i,1}, g_{i,2}, \ldots, g_{i,M}]$ when $y_i(m)$ is served as the input, and then the average value of the j-th dimension is calculated:

$$\bar{g}_j = \sum_{i=1}^{I} g_{i,j}.$$

Finally, the subscript corresponding to the maximum value of $\bar{g}_j$ is obtained as the final modulation classification fusion result:

$$r_{\text{Feature}} = \arg\max_{1\le j\le M} \bar{g}_j.$$

Fig. 5 shows a schematic diagram of feature-based fusion, in which fcx12 is used as the feature layer.

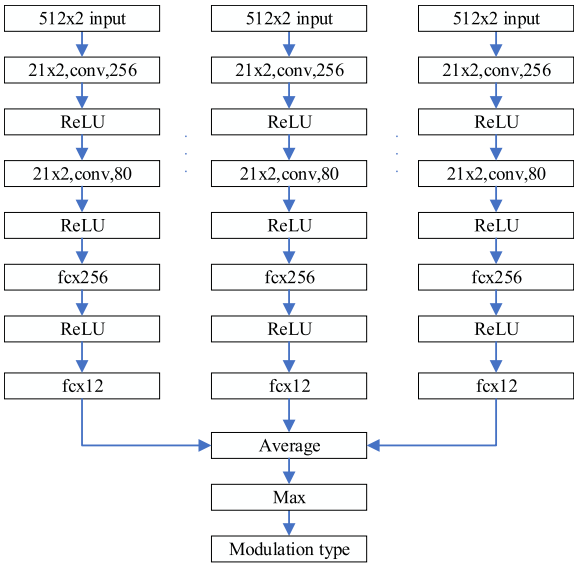


FIGURE 5. Feature fusion diagram (CNN1). Since the dropout is invalid during the reasoning process, the dropout layers are not shown in the figure.

IV. SIMULATION RESULTS

A. DATA GENERATION

In this paper, signals of various modulations are generated by simulation. The simulation considers several nonideal effects of a real communication system, including pulse shaping, carrier phase and frequency offsets and noise. 12 modulations are considered, namely BPSK, QPSK, 8PSK, OQPSK, 2FSK, 4FSK, 8FSK, 16QAM, 32QAM, 64QAM, 4PAM, and 8PAM. The original bit stream to be modulated is generated in a random manner to ensure that the transmitted symbols have equal probability. Raised cosine pulse-shaping filter is used and the roll-off factor is randomly chosen within the range 0.2 to 0.7. The phase offset is randomly selected, and the carrier frequency offset (normalized to the sampling frequency) is randomly chosen within the range -0.1 to 0.1. The signal-to-noise ratio (SNR) ranges from -20 dB to 30 dB with an interval of 2 dB. A thousand samples were generated as training data and the same number of samples were used as validation data for each modulation at each SNR. Each signal sample contains 64 symbols, and the oversampling rate is 8, so the number of sampling points of each signal sample is 512. In the fusion experiment, the number of sampling points of each signal burst ranges from 512 to 4096.

B. SIMULATION RESULTS

1) TRAINING OF THE NETWORKS

As previously mentioned, 1000 samples are generated for each SNR and each modulation type. The samples of all SNRs and all modulation types constitutes the training set. The validation set is generated in the same way. During the training, the mini-batch size is set to 128. The training uses the stochastic gradient descent (SGD) method with momentum, and the momentum factor is 0.9. The number of training epoch of CNN1 and CNN2 is 60 and 20, respectively. Validation is performed every 1500 iterations. Fig. 6 shows the training progress of CNN1 and CNN2. Note that CNN2 takes less iterations for convergence of training.

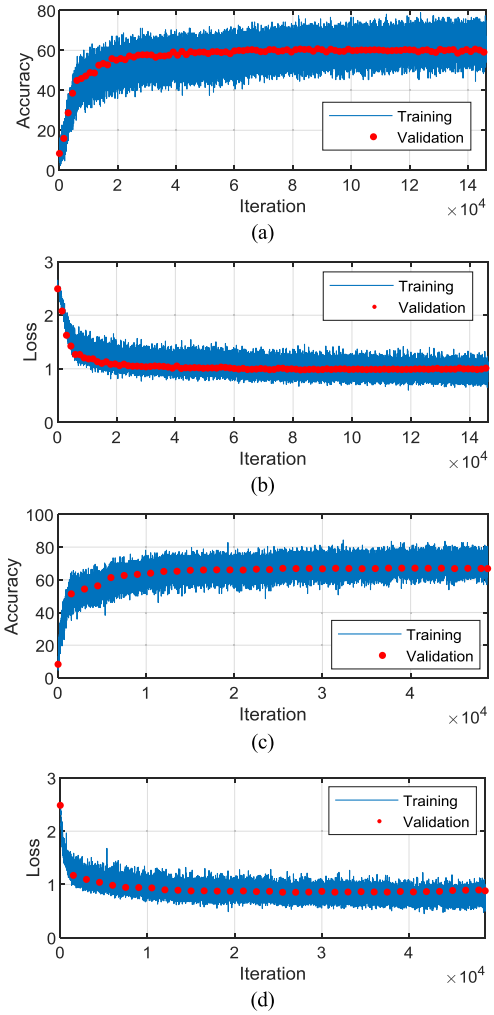


FIGURE 6. Training progress. (a) Training and validation accuracy of CNN1, (b) training and validation loss of CNN1, (c) training and validation accuracy of CNN2, and (d) training and validation loss of CNN2.

2) BASIC CLASSIFICATION RESULTS

The classification accuracy performance of 12 modulation types using CNN1 and CNN2 is given as a benchmark for performance comparison of fusion methods. Fig. 7 shows the classification performance of the trained model at different SNRs. It can be seen that CNN2 performs better than CNN1 in modulation classification accuracy. Especially when the SNR is greater than 10 dB, the classification accuracy difference between the two methods is about 6.5%. The reason why the two CNN models with different performances are used is because we want to observe the performance improvement of the fusion methods for different CNNs.

3) COMPARISON OF DIFFERENT FUSION METHODS

We now compare the performance of the three fusion methods presented in Section III. Different from the signal length

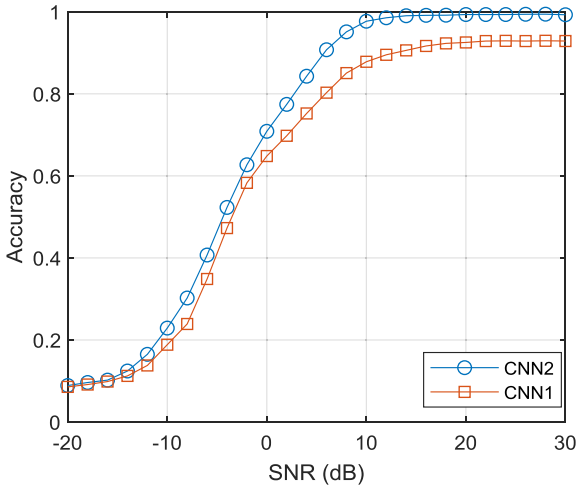


FIGURE 7. Modulation classification accuracy of CNN1 and CNN2.

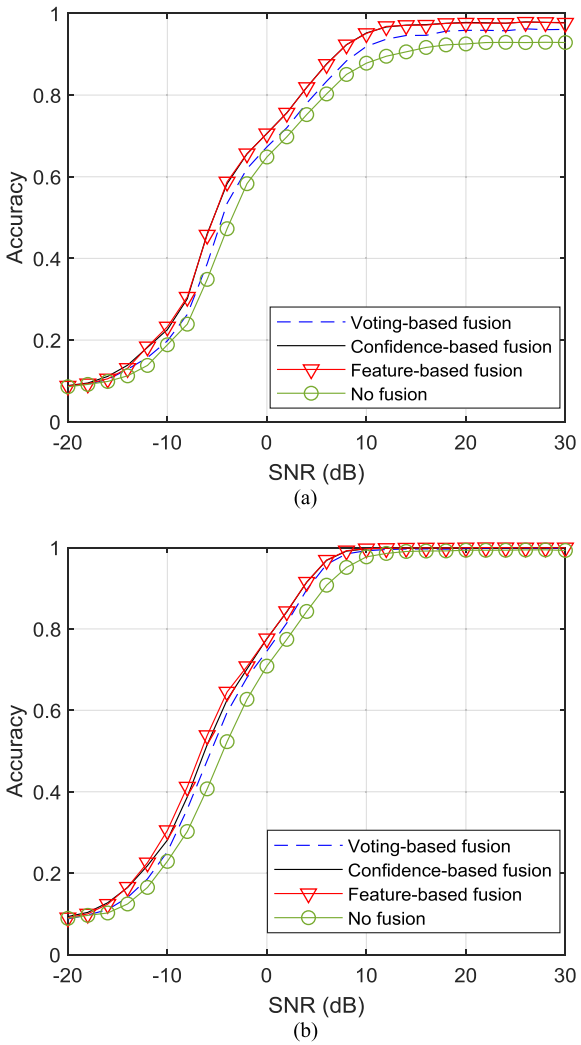


FIGURE 8. Performance comparison of different fusion methods. (a) CNN1, and (b) CNN2.

used in the previous experiment, the signal length here is 2048 sampling points. When the signal is segmented, there is no overlap between adjacent signal segments, i.e. $P = 512$.

Fig. 8 shows the performance curves for the three fusion methods. Whether it is CNN1 or CNN2, the performance of the three fusion methods is significantly better than the non-fusion method. The performance of confidence-based fusion is very close to that of feature-based fusion, which is better than voting-based fusion. The performance gains under CNN1 is much more obvious. It can also be said that fusion can bring greater performance gains to poor performance networks.

4) EFFECT OF DIFFERENT OVERLAPPING RATIOS

Since the signal is segmented, there may be overlaps between adjacent two signal segments. Fig. 9 and Table 1 show the performance at different overlap ratios. For the sake of simplicity, the performance of the feature-based fusion method is given here, and the results of other fusion methods are similar. There is almost no difference in performance at different overlapping ratios. Therefore, considering the computational complexity, a non-overlapping signal segmentation approach is a better choice.

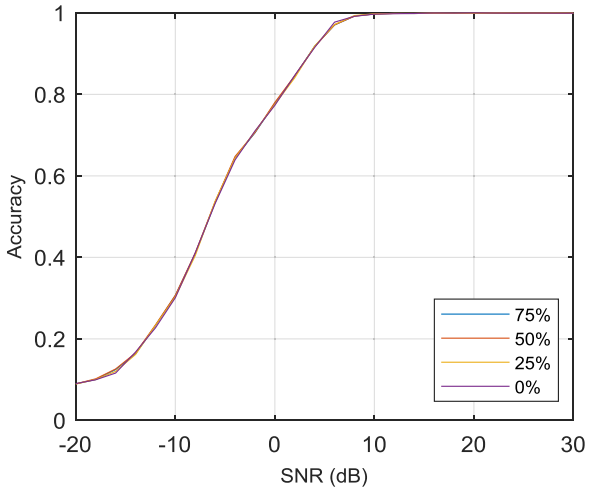


FIGURE 9. Fusion performance under different overlapping ratios.

TABLE 1. Fusion performance under different overlapping ratios.

SNR (dB)		-20	-10	0	10	20
Accuracy	0%	0.0892	0.3010	0.7732	0.9970	0.9995
	25%	0.0892	0.3032	0.7755	0.9975	0.9995
	50%	0.0897	0.3071	0.7781	0.9973	0.9996
	75%	0.0898	0.3064	0.7803	0.9974	0.9996

5) PERFORMANCE AT DIFFERENT SIGNAL LENGTHS

Fig. 10 shows the performance at different signal lengths, using a feature-based fusion approach. The oversampling rate is 8. It can be seen from the figure that, as expected, as the signal length increases, the classification accuracy obtained by the fusion method of this paper is improved. At high SNR, the fusion performance gains of CNN1 is more obvious.

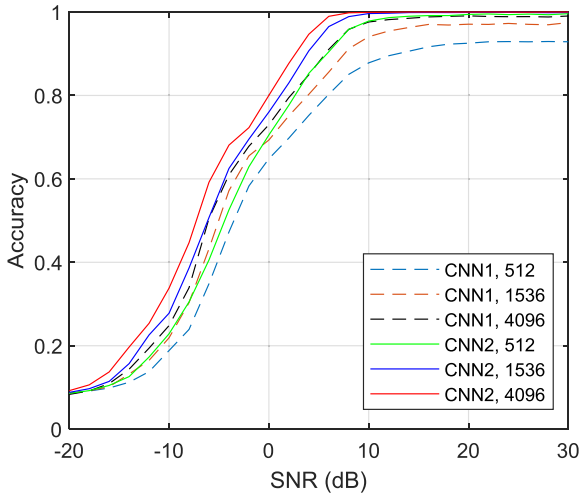


FIGURE 10. Fusion performance under different signal lengths.

6) PERFORMANCE WITH VARYING SNR

The characteristics of the communication channel may cause the parameters such as the SNR to be continuously changed over time. Fig. 11 shows the classification accuracy of the feature fusion method in the case of SNR variation, in which the CNN1 is used. The signal SNR varies over the interval $[\gamma - \delta, \gamma + \delta]$, where γ is the average SNR and δ is the maximum SNR variation. As can be seen from the figure, the SNR variation has little effect on the fusion performance. When the average SNR is low, the increase in the amount of SNR variation is even beneficial to the improvement of the classification accuracy. This is mainly because the increase in the amount of SNR variation means that the probability of occurrence of a higher SNR signal segment increases, and this high SNR signal segment contributes to the classification accuracy of the fusion result.

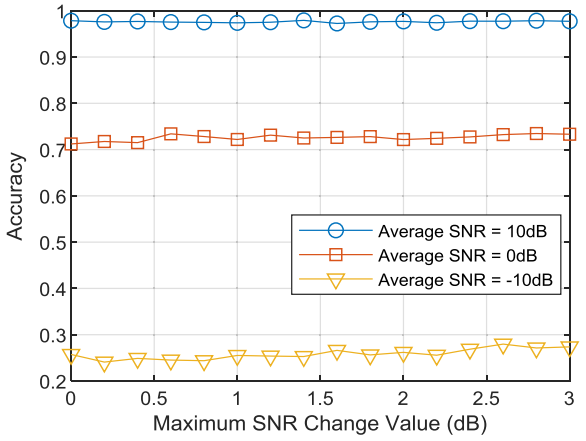


FIGURE 11. Fusion performance with varying SNR.

V. CONCLUSION

We have studied the fusion method when the signal length is larger than the designed CNN input length. The methods we have proposed include voting-based fusion, confidence-based fusion, and feature-based fusion. The simulation results show that the accuracy of the modulation classification of the three fusion methods is better than that of the non-fusion method, and the performance of the latter two fusion methods is very close, which is better than the first one. It should be noted that although the fusion methods of this paper consider the case of a single classification node, they can also be extended to multi-node classification fusion. We will further study the modulation classification fusion strategy in the multi-node cooperation scenario [52] to further improve the accuracy of modulation classification.

REFERENCES

[1] Z. Zhu and A. K. Nandi, *Automatic Modulation Classification: Principles, Algorithms and Applications*. Hoboken, NJ, USA: Wiley, 2015.
[2] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: Classical approaches and new trends," *IET Commun.*, vol. 1, no. 2, pp. 137–156, Apr. 2007.
[3] T. Ulversøy, "Software defined radio: Challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 4, pp. 531–550, Nov. 2010.
[4] S. Haykin and P. Setoodeh, "Cognitive radio networks: The spectrum supply chain paradigm," *IEEE Trans. Cogn. Commun. Netw.*, vol. 1, no. 1, pp. 3–28, Mar. 2015.
[5] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 116–130, 1st Quart., 2009.
[6] S. Zheng, S. Chen, L. Yang, J. Zhu, Z. Luo, J. Hu, and X. Yang, "Big data processing architecture for radio signals empowered by deep learning: Concept, experiment, applications and challenges," *IEEE Access*, vol. 6, pp. 55907–55922, 2018.
[7] S. Rajendran, R. Calvo-Palomino, M. Fuchs, B. Van den Bergh, H. Cordobes, D. Giustiniano, S. Pollin, and V. Lenders, "Electrosense: Open and big spectrum data," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 210–217, Jan. 2018.
[8] A. Polydoros and K. Kim, "On the detection and classification of quadrature digital modulations in broad-band noise," *IEEE Trans. Commun.*, vol. 38, no. 8, pp. 1199–1211, Aug. 1990.
[9] B. F. Beidas and C. L. Weber, "Asynchronous classification of MFSK signals using the higher order correlation domain," *IEEE Trans. Commun.*, vol. 46, no. 4, pp. 480–493, Apr. 1998.
[10] P. Panagiotou, A. Anastasopoulos, and A. Polydoros, "Likelihood ratio tests for modulation classification," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct. 2000, pp. 670–674.
[11] W. Wei and J. M. Mendel, "Maximum-likelihood classification for digital amplitude-phase modulations," *IEEE Trans. Commun.*, vol. 48, no. 2, pp. 189–193, Feb. 2000.
[12] J. L. Xu, W. Su, and M. Zhou, "Likelihood-ratio approaches to automatic modulation classification," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 41, no. 4, pp. 455–469, Jul. 2011.
[13] A. Swami and B. M. Sadler, "Hierarchical digital modulation classification using cumulants," *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 416–429, Mar. 2000.
[14] S. S. Soliman and S.-Z. Hsue, "Signal classification using statistical moments," *IEEE Trans. Commun.*, vol. 40, no. 5, pp. 908–916, May 1992.
[15] B. G. Mobasseri, "Digital modulation classification using constellation shape," *Signal Process.*, vol. 80, no. 2, pp. 251–277, 2000.
[16] D. Grimaldi, S. Rapuano, and L. De Vito, "An automatic digital modulation classifier for measurement on telecommunication networks," *IEEE Trans. Instrum. Meas.*, vol. 56, no. 5, pp. 1711–1720, Oct. 2007.
[17] S. Majhi, R. Gupta, W. Xiang, and S. Glisic, "Hierarchical hypothesis and feature-based blind modulation classification for linearly modulated signals," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 11057–11069, Dec. 2017.
[18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.
[19] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[20] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, Apr. 2017.

[21] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[22] A. Ioannidou, E. Chatzilari, S. Nikolopoulos, and I. Kompatsiaris, "Deep learning advances in computer vision with 3D data: A survey," *ACM Comput. Surv.*, vol. 50, no. 2, p. 20, 2017.

[23] R. Socher, Y. Bengio, and C. D. Manning, "Deep learning for NLP (without magic)," in *Proc. Tutorial Abstr. ACL*, 2012, p. 5.

[24] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cognit. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.

[25] H. Huang, W. Xia, J. Xiong, J. Yang, G. Zheng, and X. Zhu, "Unsupervised learning-based fast beamforming design for downlink MIMO," *IEEE Access*, vol. 7, pp. 7599–7605, 2019.

[26] X. Yan, F. Long, J. Wang, N. Fu, W. Qu, and B. Liu, "Signal detection of MIMO-OFDM system based on auto encoder and extreme learning machine," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 1602–1606.

[27] T. Wang, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based CSI feedback approach for time-varying massive MIMO channels," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 416–419, Apr. 2019.

[28] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and doa estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.

[29] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," Jun. 2017, *arXiv:1706.01151*. [Online]. Available: https://arxiv.org/abs/1706.01151

[30] K. Merchant, S. Revay, G. Stantchev, and B. Nousain, "Deep learning for RF device fingerprinting in cognitive communication networks," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 160–167, Feb. 2018.

[31] Z. Zhang, X. Guo, and Y. Lin, "Trust management method of D2D communication based on RF fingerprint identification," *IEEE Access*, vol. 6, pp. 66082–66087, 2018.

[32] G. Gui, H. Huang, Y. Song, and H. Sari, "An effective NOMA scheme based on deep learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.

[33] M. Liu, T. Song, and G. Gui, "Deep cognitive perspective: Resource allocation for NOMA-based heterogeneous IoT with Imperfect SIC," *IEEE Internet Things*, vol. 6, no. 2, pp. 2885–2894, Apr. 2019.

[34] Y. Li, X. Cheng, and G. Gui, "Co-robust-ADMM-net: Joint ADMM framework and DNN for robust sparse composite regularization," *IEEE Access*, vol. 6, pp. 47943–47952, 2018.

[35] S. Wu, A. G. Dimakis, S. Sanghavi, F. X. Yu, D. Holtmann-Rice, D. Storcheus, A. Rostamizadeh, and S. Kumar, "Learning a compressed sensing measurement matrix via gradient unrolling," Jun. 2018, *arXiv:1806.10175*. [Online]. Available: https://arxiv.org/abs/1806.10175

[36] M. Liu, J. Yang, T. Song, J. Hu, and G. Gui, "Deep learning-inspired message passing algorithm for efficient resource allocation in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 641–653, Jan. 2018.

[37] Y. Lin, H. Tao, Y. Tu, and T. Liu, "A node self-localization algorithm with a mobile anchor node in underwater acoustic sensor networks," *IEEE Access*, vol. 7, pp. 43773–43780, 2019.

[38] X. Wang, L. Gao, and S. Mao, "CSI phase fingerprinting for indoor localization with a deep learning approach," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1113–1123, Dec. 2016.

[39] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. Int. Conf. Eng. Appl. Neural Netw.*, 2016, pp. 213–226.

[40] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.

[41] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Trans. Cognit. Commun. Netw.*, vol. 4, no. 3, pp. 433–445, Sep. 2018.

[42] S. Peng, H. Jiang, H. Wang, H. Alwageed, Y. Zhou, M. M. Sebdani, and Y.-D. Yao, "Modulation classification based on signal constellation diagrams and deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 718–727, Mar. 2019.

[43] S. Ramjee, S. Ju, D. Yang, X. Liu, A. EI Gamal, and Y. C. Eldar, "Fast deep learning for automatic modulation classification," Jan. 2019, *arXiv:1901.05850*. [Online]. Available: https://arxiv.org/abs/1901.05850

[44] F. Meng, P. Chen, L. Wu, and X. Wang, "Automatic modulation classification: A deep learning enabled approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10760–10772, Nov. 2018.

[45] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-driven deep learning for automatic modulation recognition in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 4074–4077, Apr. 2019.

[46] Y. Tu, Y. Lin, J. Wang, and J.-U. Kim, "Semi-supervised learning with generative adversarial networks on digital signal modulation classification," *Comput. Mater. Continua*, vol. 55, no. 2, pp. 243–254, 2018.

[47] B. Tang, Y. Tu, Z. Zhang, and Y. Lin, "Digital signal modulation classification with data augmentation using generative adversarial nets in cognitive radio networks," *IEEE Access*, vol. 6, pp. 15713–15722, 2018.

[48] M. Sadeghi and E. G. Larsson, "Adversarial attacks on deep-learning based radio signal classification," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 213–216, Feb. 2019.

[49] Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, and V. K. Asari, "The history began from AlexNet: A comprehensive survey on deep learning approaches," Mar. 2018, *arXiv:1803.01164*. [Online]. Available: https://arxiv.org/abs/1803.01164

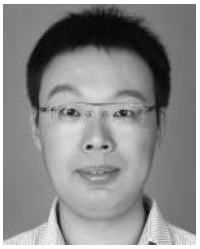
[50] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 119–130, Nov. 1998.

[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[52] M. Abdelbar, W. H. Tranter, and T. Bose, "Cooperative cumulants-based modulation classification in distributed networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 3, pp. 446–461, Sep. 2018.

SHILIAN ZHENG (M'11) was born in Wencheng, Zhejiang, China, in 1984. He received the B.S. degree in telecommunication engineering and the M.S. degree in signal and information processing from Hangzhou Dianzi University, Hangzhou, China, in 2005 and 2008, respectively, and the Ph.D. degree in communication and information system from Xidian University, Xi'an, China, in 2014.

He is currently an Associate Researcher with the Science and Technology on Communication Information Security Control Laboratory, Jiaxing, China. He holds more than ten patents. His research interests include cognitive radio, spectrum management, compressed sensing, and deep learning-based radio signal processing.



PEIHAN QI was born in Henan, China, in 1986. He received B.S. degree in telecommunications engineering from Chang'an University, Xi'an, in 2006, the M.S. degree in communication and information system, and the Ph.D. degree in military communication from Xidian University, in 2011 and 2014, respectively, where he has been an Associate Professor with the School of Telecommunications Engineering, since 2018. His research interests include compressed sensing, spectrum sensing in cognitive radio networks, and high-speed digital signal processing.

SHICHUAN CHEN was born in Pan'an, Zhejiang, China, in 1976. He received the B.S. degree in electronic circuits and systems from the University of Electronic Science and Technology of China, Chengdu, China, in 1998.

He is currently an Associate Researcher with the Science and Technology on Communication Information Security Control Laboratory, Jiaxing, China. He holds seven patents. His research interests include radio signal processing, analysis and recognition, big data for radio signals, and deep learning algorithms for radio signals.

XIAONIU YANG is currently the Chief Scientist with the Science and Technology on Communication Information Security Control Laboratory, Jiaxing, China. He is also an Academician of the Chinese Academy of Engineering, a Fellow of the Chinese Institute of Electronics, and a Ph.D. Supervisor of Xidian University, Xi'an, China. He published the first software radio book in China, *Software Radio Principles and Applications* [China: Publishing House of Electronics Industry, 2001 (in Chinese)] along with C. Lou and J. Xu. He holds more than 40 patents. His current research interests include software-defined satellite, big data for radio signals, and deep learning-based signal processing.