Modulation Classification of VHF Communication System based on CNN and Cyclic Spectrum Graphs

Hao Wu, Yaxing Li, Yu Guo, Liang Zhou, and Jin Meng
National Key Laboratory of Science and Technology on Vessel Integrated Power System
Naval University of Engineering
Wuhan, Hubei, 430033, China
E-mail: wowhow@163.com

***Abstract*—Modulation classification is the technological basis of adaptive interference mitigation in communication system. This paper proposes a modulation classification method for very high frequency (VHF) signals, which is based on deep convolutional neural network (CNN) and cyclic spectrum graphs. First, the cyclic spectrum of VHF signals is analyzed. Then, a deep learning method based on CNN is proposed, down-sampling and clipping technologies are used for preprocessing cyclic spectrum images, parameters of the proposed neural network are optimized, and finally the modulation classification is realized. The experimental results show that, the proposed method has high modulation classification accuracy and less computation burden in low SNR.**

Keywords—modulation classification; low signal-to-noise ratio; deep learning; convolutional neural network; cyclic spectrum graph

I. INTRODUCTION

Modulation classification is the technological basis of adaptive interference mitigation in communication system, and it has been a hot research topic of military and civilian communication system in many years, especially in very high frequency (VHF) band. Previously, modulation classification has been mainly accomplished by likelihood-based methods [1,2] and feature-based methods [3,4]. In general, likelihood-based methods require specific parameters or prior information, and feature-based methods require the expert features of signal, such as instantaneous amplitude, phase, cumulants and constellation diagram. However, these traditional methods typically have high computational complexity and are limited in low SNR.

Recently, deep learning architecture [5] has been attracting increasing attention due to the successful applications in various fields, such as computer vision, natural language processing and economics. In [6], the authors propose a deep learning method based on convolutional neural networks (CNN), which can classify 11 analog and digital modulation signals. Based on [6], the authors in [7] propose three convolutional neural networks based on temporal IQ, amplitude/phase and frequency spectrum, respectively, and can realize interference recognition in ISM frequency band. The authors in [8] propose a modulation classification method based on deep neural network in frequency-selective fading channels. In [9], signal constellation diagram is introduced and two new CNN-based models, AlexNet and GoogLeNet, are used for modulation classification.

In this paper, we consider the modulation classification of 7 communication signals, including AM, FM, 2FSK, 4FSK, BPSK, QPSK and MSK, which are extensively used in VHF band. Based on the deep learning architectures, a multi-layer CNN model is built for addressing this issue. First, the theory of cyclic spectrum is given and the feasibility of modulation classification based on the two dimensional gray images of cyclic spectrum is analyzed. Then, a deep learning method based on cyclic spectrum and convolutional neural network is proposed, down-sampling and clipping technologies are used for preprocessing cyclic spectrum images, parameters of the proposed neural network are optimized, and finally the modulation classification is realized.

II. CYCLIC SPECTRUM ANALYSIS OF SIGNALS

Cyclic spectrum analysis is an important tool for analyzing the stationary characteristics of signals. Note that the cyclic spectrum of stationary noise is nearly zero at non-zero cyclic frequency, thus it has good anti-noise properties. For a modulated cyclostationary signal $x(t)$, the cyclic autocorrelation function $R_x^\alpha(\tau)$ can be written as

$$R_x^\alpha(\tau) = \lim_{\hat{T}\to\infty} \frac{1}{\hat{T}} \int_{-\hat{T}/2}^{\hat{T}/2} x(t+\frac{\tau}{2}) x^*(t-\frac{\tau}{2}) e^{-j2\pi\alpha t} dt \tag{1}$$

where $\hat{T}$ denotes the cycle, τ denotes the interval and α denotes the cycle frequency, respectively. Through Fourier transform, $R_x^\alpha(\tau)$ becomes

$$\begin{aligned} S_x^\alpha(f) &= \int_{-\infty}^{\infty} R_x^\alpha(\tau) e^{-j2\pi f\tau} d\tau \\ &= \lim_{\hat{T}\to\infty} \frac{1}{\hat{T}} X_{\hat{T}}(f+\frac{\alpha}{2}) X_{\hat{T}}^*(f-\frac{\alpha}{2}) \end{aligned} \tag{2}$$

where $S_x^\alpha(f)$ is the corresponding cyclic spectrum, and

$$X_{\hat{T}}(f) = \int_{-\hat{T}/2}^{\hat{T}/2} x(t) e^{-j2\pi ft} dt \tag{3}$$

is the spectrum of $x(t)$.

Fig. 1(a) and Fig. 1(b) present the cyclic spectrum of AM and BPSK modulated signals and the corresponding 2-D view of gray images of $f-\alpha$ plane, respectively, where the more white the pixels, the greater the values in gray images. It is obvious that the cyclic spectrum of AM and BPSK signals is different in details, especially in peak regions with the whitest pixels of gray image. Therefore, we can select the two-dimensional gray images of cyclic spectrum as the input data, and the deep CNN structure can be used to extract and learn multi-scale features of modulation signals automatically.

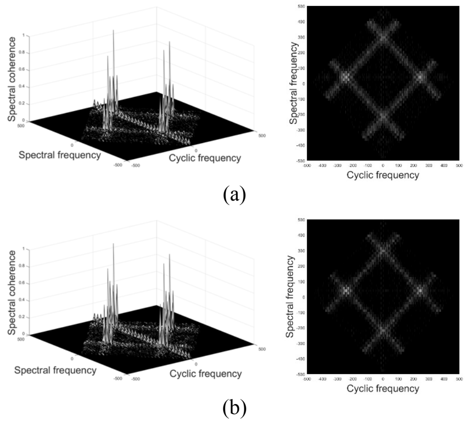


Fig. 1. Cyclic spectrum and the corresponding 2-D view. (a) AM signal (b) BPSK signal.

The original two-dimensional gray images of cyclic spectrum is usually large and has high resolution, CNN hardly converge in large samples and may cause huge burden on the computer storage. However, it is noted that the features of cyclic spectrum are sparse and the data is high redundancy, as shown in Fig. 1, the gray images of cyclic spectrum has the same value in some regions and can be sparsely represented effectively. Therefore, on the premise of unchanging the fundamental features, preprocessing the original gray images of cyclic spectrum is necessary and feature reduction is an important step.

First, we use the down-sampling technology. Assume that the original gray image is divided into K same-sized local receptive domains $\aleph_k$, $k = 1,\ldots,K$, the number of pixels in each domain is N, and all domains should be no overlap each other and the total number of pixels is NK. By down-sampling, all pixels in $\aleph_k$ is replaced by one single pixel $\overline{h}_k$, and the total number of pixels is reduced to K .In this paper, there are only 7 modulation signals need to be classified, and the structure of the proposed CNN can be relatively simple. To reduce parameter optimization and increase convergence speed, we use the classical LeNet-5 CNN model in [10] for reference and we set $K = 28$. Specifically, the pixel with maximum value in $\aleph_k$ is selected as $\overline{h}_k$, and we have

$$\overline{h}_k = \max_{i,j\in\aleph_k} h_{i,j}, \quad k = 1,\ldots,K \tag{4}$$

Then we use the clipping technology for further reducing the information redundancy of the gray images. The bottom right corner of the preprocessed images with 16×16 pixels are clipped as feature images.

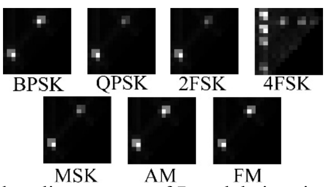


Fig. 2. Pre-processed cyclic spectrum of 7 modulation signals.

Fig. 2 shows the clipped gray images of 7 different modulation signals. It is obvious that the distribution and brightness of feature points of different signals are different. Thus, the pre-processed 16×16 pixels gray images of cyclic spectrum, as shown in Fig. 2, can be used as the dataset for deep learning.

III. The Structure of CNN

The structure of the proposed CNN is presented in Fig. 3, which includes convolutional layer C1, pooling layer S2, convolutional layer C3, pooling layer S4, fully-connected layer C5, fully-connected layer F6 and fully-connected layer O7. Based on the classic LeNet-5 model, we fine tune the size of convolutional kernels and feature maps for improving the convergence speed of CNN. The parameters of each layer are set as follows

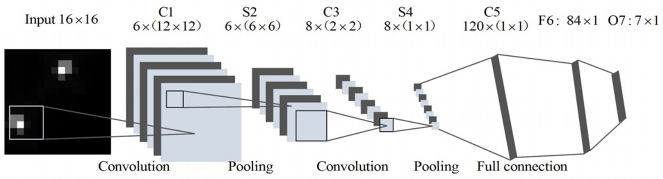


Fig. 3. Structure of the proposed CNN.

1) Convolutional layer C1. The size of convolutional kernel is 5×5, the number of feature maps is 6, the convolution step is 1, and the size of the output feature map is $6\times(12\times12)$.

2) Pooling layer S2. The output feature maps of layer C1 are pooled with 2×2 filters, where the maximum value is chosen from four values, and the size of the output feature map is $6\times(6\times6)$.

3) Convolutional layer C3. The size of convolutional kernel is 5×5, the number of feature maps is 8, the convolution step is 1, and the size of the output feature map is $8\times(2\times2)$.

4) Pooling layer S4. The output feature maps of layer C3 are pooled with 2×2 filters, and the size of the output feature map is $8\times(1\times1)$.

5) Fully-connected layer C5. This layer consists of 120 neurons and is fully connected with layer S4, and the size of the output feature map is $120\times(1\times1)$.

6) Fully-connected layer C6. This layer consists of 84 neurons and is fully connected with layer C5, and the size of the output feature map is 84×1.

7) Output layer O7. Output layer O7 is a fully-connected layer and consists of 7 neurons, which is corresponding to 7 modulation signals, and the size of the output is 7×1.

Convolution operation is the core of CNN. The input feature maps are convoluted with convolutional kernel, and then the results is obtained through nonlinear activation function, which can be expressed as follows

$$X_i^l = V\left(\sum_{i\in M_j} X_i^{l-1} * K_{ij}^l + b_j^l\right) \tag{5}$$

where X_i^{l-1} denotes the input feature map, X_i^l denotes the output feature map, $*$ denotes the convolution operation, b_j^l

denotes the bias, K_{ij}^{l} denotes the convolutional kernel, M_j denotes the set of input feature maps, and $V(\cdot)$ denotes the activation function. K_{ij}^{l} and b_j^{l} are the parameters to be optimized. Therefore, the training of CNN can be divided into two steps:

1) Forward propagation. For the multi-classification problem with N training samples, C classes and L layers, the cost function can be expressed by square error function

$$\begin{aligned} E &= \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{C}\left(o_k^n - y_k^n\right)^2 \\ y_k^n &= V\left(\sum_{i=1}^{L} W_{ij}x_i + b_j\right) \end{aligned} \tag{6}$$

where o_k^n denotes the true label of the nth image of the kth sample, y_k^n denotes the predicted label.

2) Back propagation. Calculating partial derivatives of the weight W and the bias b of E, where β denotes the learning rate, and the parameters are updated as

$$W_{ij} = W_{ij} - \beta\frac{\partial}{\partial W_{ij}}E(W,b), \quad b_j = b_j - \frac{\partial}{\partial b_j}E(W,b) \tag{7}$$

In practice, the signal in VHF band often last for a short time, which results in few labeled data. However, supervised learning is based on a large number of labeled data, and directly supervised learning on few labeled data is easy to cause overfitting. To address this problem, we use sparse filtering method to pre-train the CNN, and an unsupervised learning algorithm is proposed based on few labeled data.

Suppose the number of samples is N, the number of features is M, and the feature matrix can be obtained as follows

$$\mathbf{F} = \begin{bmatrix} f_1^{(1)} & f_1^{(2)} & \cdots & f_1^{(N)} \\ f_2^{(1)} & f_2^{(2)} & \cdots & f_2^{(N)} \\ \vdots & \vdots & \cdots & \vdots \\ f_M^{(1)} & f_M^{(2)} & \cdots & f_M^{(N)} \end{bmatrix} \tag{8}$$

where $f_j^{(i)}, i = 1,\ldots,N, j = 1,\ldots,M$ denotes the jth feature of the ith sample.

Sparse filtering method is efficient because it has only one hyperparameter M. Specifically, sparse filtering is based on optimizing the following properties of feature distributions: population sparsity, lifetime sparsity and high dispersal. For C1 layer of the proposed CNN model, the convolutional kernel is $\mathbf{W} \in \mathbb{R}^{6\times(5\times5)}$, the bias is $\mathbf{b} \in \mathbb{R}^{6\times1}$, and the feature matrix $\mathbf{F}$ can be written as

$$\mathbf{F} = \sqrt{\left[V\left(\mathbf{W}\tilde{\mathbf{H}} + repmat(\mathbf{b},1,L)\right)\right]^2 + \varepsilon} \in \mathbb{R}^{6\times L} \tag{9}$$

where $\mathbf{H}_i \in \mathbb{R}^{5\times5}, i = 1,\ldots,L$, $\tilde{\mathbf{H}} = \left[\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_L\right] \in \mathbb{R}^{(5\times5)\times L}$ denotes L samples of size 5×5 extracting from cyclic spectrum, and ε denotes smoothness coefficient. Firstly, the matrix $\mathbf{F}$ is normalized by rows

$$\tilde{\mathbf{f}}_j = \frac{\mathbf{f}_j}{\left\|\mathbf{f}_j\right\|_2}, j = 1,\ldots,M \tag{10}$$

Second, the matrix $\mathbf{F}$ is normalized by column

$$\hat{\mathbf{f}}^{(i)} = \frac{\tilde{\mathbf{f}}^{(i)}}{\left\|\tilde{\mathbf{f}}^{(i)}\right\|_2}, i = 1,\ldots,N \tag{11}$$

Then the optimization objective function can be written as

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{b}}\sum_{i=1}^{N}\left\|\hat{\mathbf{f}}^{(i)}\right\|_1 = \min_{\mathbf{W},\mathbf{b}}\sum_{i=1}^{N}\left\|\frac{\tilde{\mathbf{f}}^{(i)}}{\left\|\tilde{\mathbf{f}}^{(i)}\right\|_2}\right\|_1 \\ &\text{s.t. } \mathbf{F} = \sqrt{\left[g\left(\mathbf{W}\tilde{\mathbf{H}} + repmat(\mathbf{b},1,L)\right)\right]^2 + \varepsilon} \end{aligned} \tag{12}$$

where $\|\bullet\|_1$ denotes the L1-norm, $repmat(\mathbf{b},1,L)$ denotes the vector containing L copies of $\mathbf{b}$ in the row and column dimensions. The optimization problem can be solved by numerical solvers that implement BFGS, such as the "fminunc" function in MATLAB, and the optimized $\mathbf{W}_{opt}$ and $\mathbf{b}_{opt}$ of C1 layer can be obtained.

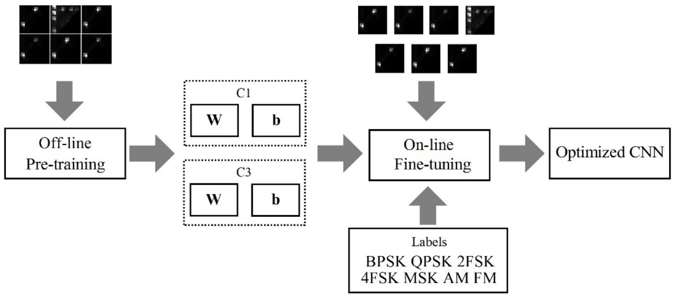


Fig. 4. Pre-training and fine-tuning steps based on sparse filtering.

Repeating the previous process layer by layer, we can also obtain the optimized $\mathbf{W}_{opt}$ and $\mathbf{b}_{opt}$ of C3 layer. Fig. 4 shows the pre-training and fine-tuning steps based on sparse filtering. Pre-train the proposed network with optimized $\mathbf{W}_{opt}$ and $\mathbf{b}_{opt}$, and fine-tune it with few labeled data, and finally we can use the well-trained CNN to recognize modulation formats.

IV. Simulation Results

In this paper, the hardware environment is Windows 7-64bits, Intel i7-8700@3.20GHz with 32GB RAM and NVIDIA GeForce GTX 1060-6GB, and the additive white Gaussian noise is introduced.

A. The Effect of Structures of CNN

First, we analyze the effect of parameters of CNN under supervised learning. 4 structures of CNN are designed with different convolutional kernels and feature maps, as shown in Table I. It is noted that the No.4 structure in Table I corresponds to the model in Fig. 3. All structures are tested for 50 times with the testing SNR=0dB, 5dB and 10dB, respectively, and the results are shown in Table II. It is obvious that the more the number of feature maps in convolutional layer, the higher the classification accuracy, but the more time consuming. Interestingly, the size of the convolutional kernel has little effect on the results. Therefore, we can choose No.3 and No.4 structures in Table I as the optimal CNN structure.

TABLE I. FOUR STRUCTURES OF CNN

No.	C1 layer		C3 layer	
	Kernel	*Features*	*Kernel*	*Features*
1	3×3	2	4×4	4
2	3×3	6	4×4	8
3	5×5	2	5×5	4
4	5×5	6	5×5	8

TABLE II. COMPARISON OF FOUR STRUCTURES

No.	Total accuracy (%)			Running time
	0dB	*5dB*	*10dB*	(s)
1	93.0	98.2	99.1	56.5
2	93.4	99.0	99.7	178.3
3	92.8	98.3	99.3	51.7
4	93.7	98.9	99.8	163.5

Table III shows the result of No.4 structure under SNR=5dB. The classification accuracies of all signals are over 98%, except for BPSK and QPSK, which are easily confused with MSK, and the total classification accuracy is 98.9%. Thus, it is indicated that the proposed method is little influenced by noise and is effective even in low SNR.

TABLE III. THE RESULT OF NO.4 STRUCTURE UNDER SNR = 5DB

True	Predicted (%)						
	BPSK	QPSK	2FSK	4FSK	MSK	AM	FM
BPSK	97.2	0	0	0	2.8	0	0
QPSK	0	97.8	0	0	2.2	0	0
2FSK	0	0	100	0	0	0	0
4FSK	0	0	0.8	99.2	0	0	0
MSK	0	1.8	0	0	98.2	0	0
AM	0	0	0	0	0	100	0
FM	0	0	0	0	0	0	100

B. The Performance of Unsupervised Pre-training

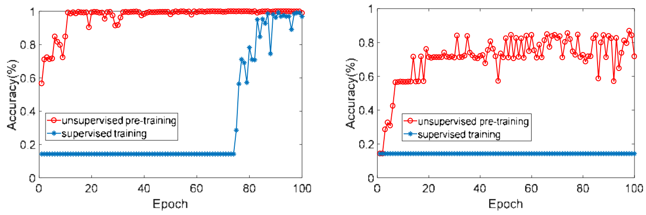
Then, we analyze the performance of unsupervised learning. For each modulation signal, 1000 unlabeled samples are randomly generated for unsupervised pre-training and the SNR is randomly varied in the range of 0-30dB, and 500 testing examples are generated and the SNR is 10dB.

1) Large labeled training sample scenario

For each modulation signal, 2000 labeled training samples are generated. Fig. 5(a) shows the performance of supervised training and unsupervised pre-training under large dataset. It can be seen that supervised learning converges only after 90 epoches, and the accuracy can reach 98%, while the unsupervised pre-training with supervised fine-tuning converges after 11 epoches, and the accuracy can reach 99.2%.

2) Small labeled training sample scenario

For each modulation signal, 200 labeled training samples are generated, Fig. 5(b) shows the performance of supervised training and unsupervised pre-training under small dataset. It can be seen that supervised learning cannot converge within 100 iterations, due to few labeled samples, while the unsupervised pre-training with supervised fine-tuning can converge after about 50 epoches, and the accuracy can reach 81%.



(a) (b)

Fig. 5. Performance comparison of unsupervised pre-training and supervised training. (a) large labeled training sample scenario (b) small labeled training sample scenario.

V. CONCLUSIONS

In this paper, a deep learning method based on cyclic spectrum and CNN is proposed, down-sampling and clipping technologies are used for preprocessing cyclic spectrum images, and the modulation classification is realized. Simulation results show that, the proposed method has high modulation classification accuracy and less computation burden in low SNR. Moreover, unsupervised pre-training has good performance in small labeled training sample scenario. Possible future research might concern the effects of carrier frequency and channel in practice.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grants 61801502 and 71801220.

REFERENCES

[1] W. Wen and J. Mendel, "Maximum-likelihood classification for digital amplitude-phase modulations", IEEE Transactions on Communications, vol. 48, no. 2, pp. 189-193, 2000.

[2] Y. Yang, Q. Liu and T. Soong, "A log-likelihood function based algorithm for QAM signal classification", Signal Processing, vol. 70, no. 1, pp. 61-71, 1998.

[3] K. Ho, W. Prokopiw and Y. Chan, "Modulation identification of digital signals by the wavelet transform", IEE Proceedings-Radar, Sonar and Navigation, vol. 147, no. 4, pp. 169-176, 2000.

[4] B. Mobasseri, "Digital modulation classification using constellation shape", Signal Processing, vol. 80, no. 2, pp. 251-277, 2000.

[5] G. Hinton, S. Osindero and Y. Teh, "A fast learning algorithm for deep belief nets", Neural Computation, vol. 18, no. 7, pp. 1527-1554, 2006.

[6] T. Oshea, J. Corgan and T. Clancy, "Convolutional radio modulation recognition networks", in International Conference on Engineering Application of Neural Networks, pp. 213-226, 2016.

[7] K. Merima, K. Tarik and M. Ingrid, "End-to-end learning from spectrum data: a deep learning approach for wireless signal identification in spectrum monitoring applications", IEEE Access, Mar. 2018.

[8] F. Khan, C. Lu and A. Lau, "Joint modulation format/bit-rate classification and signal-to-noise ratio estimation in multipath fading channels using deep machine learning", Electronics Letters, vol. 52, no. 14, pp. 1272-1274, 2016.

[9] S. Peng, H. Jiang and H. Wang, "Modulation classification based on signal constellation diagrams and deep learning", IEEE Transactions on Neural Networks and learning systems, pp. 1-10, Early Access, July 2018.

[10] Y. LeCun, L. Bottou and Y. Bengio, "Gradient-based learning applied to document recognition", Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.

