(301) Communication Engineering

A study on Automatic Modulation Recognition using Convolutional Neural Network with Constellation Diagram

Yutao Liu† Toru Nishiyama† Shigeru Tomisato† Kazuhiro Uehara†
†Graduate School of Natural Science and Technology, Okayama University

1. Introduction

In process of digital signal application, modulation of communication signal is one of key parameters. It is necessary to know modulation methods of digital signals, when it is demodulated. Therefore, the study on automatic modulation recognition of digital signals has received extensive attention. Traditional systems have been studied since a long time ago, but there are some problems such as low efficiency and low accuracy [1]. According to previous studies, when the $E_b/N_0 >$ 0dB, the automatic modulation recognition accuracy is above 90% with using Artificial Neural Networks (ANN) [2], and when the $E_b/N_0 >$ 5dB, the automatic modulation recognition accuracy is above 85% with using Backpropagation (BP) algorithms [3]. However, when the $E_b/N_0 <$ 0dB, the above methods are almost impossible to recognite the modulation method.

In recent years, there are a lot of applications on field of image recognition for the Convolutional Neural Network (CNN), and they have achieved great recognition results [4]. And digital amplitude and phase modulation signals can be uniquely represented by signals' constellation diagram. Therefore, in order to achieve modulation recognition under low E_b/N_0 , we propose a new method for automatic modulation recognition based on CNN and use the trained model to classify with constellation diagram. After used the improved CNN structure and the appropriate hyperparameters, we have achieved high efficiency and high precision recognition of modulation method.

2. Digital communication system model

For this study, the digital communication system can be seen in Fig.1. First of all we generate the random signals with the equal probability distribution of 0 and 1. Then, the generated signals are modulated by different methods [5]. In order to be able to simulate broadband characteristics of the Additive white Gaussian noise, and to be able to show that the effect of Square Root Raised Cosine filter (SRRC), add 0 points between any two adjacent codewords in I and Q paths of the original signals.

Then in order to avoid interference between the adjacent signals in transmission, there ought be suitable signal waveforms in multiple code. In this study, we use SRRC filter at the sending end and Matched filter at the receiving end. According to Nyquist–Shannon sampling theorem [4], the accepted waveform in the communication system of this study is a raised cosine roll-off signal, and this process is implemented by SRRC filter and Matched filter. The filters at both ends are raised cosine roll-off filtered, and the combination of the two parts achieves raised cosine roll-off filtering. A SRRC filter is generally expressed as in (1).

$$H(f)=\begin{cases} 1 & if\ \ |f|<f(1-\alpha) \\ \left[\frac{1}{2}+\frac{1}{2}\sin\frac{\pi}{2f_N}\left(\frac{f_N-|f|}{\alpha}\right)\right]^{\frac{1}{2}} & if\ \ f(1-\alpha)\le|f|\le f(1+\alpha) \\ 0 & if\ \ |f|>f(1+\alpha) \end{cases} \tag{1}$$

Here, $f_N=\frac{1}{2T_s}=\frac{R_s}{2}$ is Nyquist frequency, α is roll-off factor. Assume that the frequency transfer function of Matched filter is $H(f)$, the time domain impulse response is $h(t)$.

And the filter input is the superposition of the sending signal and the noise, and a Matched filter is generally expressed as in (2).

$$x(t)=S(t)+n(t) \tag{2}$$

Here, $S(t)$ is signal, its spectral function is $S(f)$. $n(t)$ is the Gaussian white noise , its bilateral power spectral density is $n_0/2$. The filter output is expressed as in (3).

$$y(t)=\left[S(t)+n(t)\right]*h(t) \tag{3}$$

Then, in this study we use Additive white Gaussian noise (AWGN) channel to add Gaussian white noise in original signal. Here the parameter of AWGN is E_b/N_0 , thereinto E_b is the signal energy per bit, and N_0 is the noise spectral density.

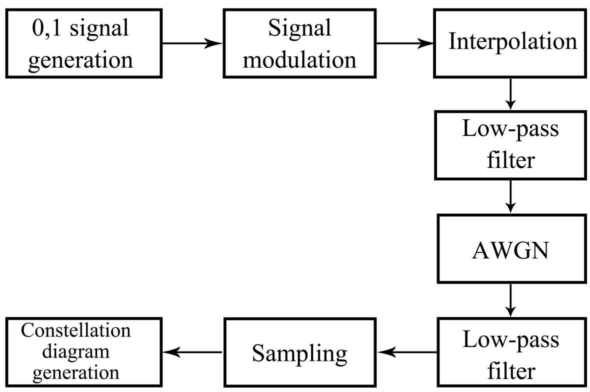

Fig.1 A digital communication system

And then since the number of signal points passing through Matched filter are more than the number of the original signal points, there ought be sampled in the signal. Finally, we use the sampled signal to generate constellation diagrams.

3. Convolutional Neural Network model

For this study, the CNN model can be seen in Fig.2. This study uses the CNN architecture based on Lenet architecture. Input layer inputs constellation diagrams of 528*700 pixels. After passing two convolutional layers, two activation function layers, two pooling layers, and two fully connected layers, finally the output layer outputs the result according to four classification methods. In the CNN architecture of this study, we use the Softmax function(5) on the output layer.

$$f(x)_j = \frac{e^{x_j}}{\sum_{k=1}^{K} e^{x_k}} \quad for\ j = 1, \ldots, K \tag{5}$$

Here, the sum of the function's outputs $f(x)$ is 1.

And we use ReLU function (6) to the activation function, and we use Log-likelihood cost function (7) to the loss function

$$f(x) = \begin{cases} x & if \quad x \geq 0 \\ 0 & if \quad x < 0 \end{cases} \tag{6}$$

$$f(a,y) = \sum_k y_k \log a_k \tag{7}$$

The following is the particular struture of each layer. Input layer: constellation diagrams of 528*700*3 pixels. Convolutional layer1: 130*173*8 feature maps generated by eight 11*11 cores, cores' stride are 4 and the activation function is ReLU. Pooling layer1: 65*86*8 feature maps generated by one 2*2 cores, cores' stride are 2. Convolutional layer2: 31*41*16 feature maps generated by sixteen 5*5 cores, cores' stride are 2 and the activation function is ReLU. Pooling layer2: 15*20*16 feature maps generated by one 2*2 cores, cores' stride are 2. Fully connected layer1: 22016 points converted from all pixels in the pooling layer2 and the activation function is ReLU. Fully connected layer2: 4 points converted from fully connected layer1. The softmax function on the output layer outputs four nodes, which stand for four classifications.

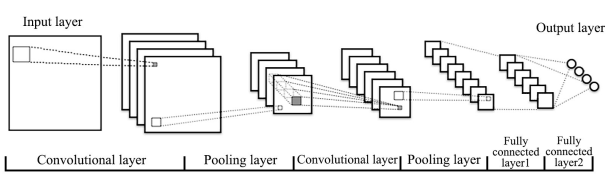

Fig.2 Convolutional Neural Network model

4. Simulation and results

The simulation parameters can be seen in Fig.3. For the digital communication system model, we use E_b/N_0 to represent the value of signal energy per bit-to-background noise power spectrum destiny ratio, and choose the range of E_b/N_0 from -15db to 5dB. And then we use four digital modulation methods: BPSK, QPSK, 8PSK and 16QAM. For the carrier frequency and the sampling frequency, we set to 2kHz and 8kHz. The sampling point is set to 1024. We superimpose AWGN channel, simulating constellation diagrams after signal processing, and obtain training sets and test sets.

For the CNN model, the first is generation of data sets. In order to obtain the CNN model, a large number of training sets are required for supervised training, and in order to verify the accuracy of the CNN model, the test set needs to be used for test verification. Both the training set and the test set are labeled constellation diagrams under different signal to noise ratio and different modulation methods. In this study, we choose 50 constellation diagrams for each 1dB, and there are a total of 1,050 constellation diagrams for each modulation method. Use these generated constellation diagrams as the training set. In each modulation method, 105 constellation diagrams are generated for each 1dB. Use these generated constellation diagrams as the test set. Then there are learning rate, miniBatchsize and maxEpochs on the training parameter. The miniBatchsize means the number of samples that will be propagated through the network and it is set to 25. And an epoch corresponds to a full pass of the data, maxEpochs is epoch size and it is set to 40. Learning rate can control training progress and it is set to 10^{-5}.

Simulation parameter	Value
E_b/N_0	-15dB~5dB
Modulation method	BPSK, QPSK, 8PSK, 16QAM
Carrier frequency	2kHz
Sampling frequency	8kHz
Sampling point	1024
Learning rate	10^{-5}
MiniBatchsize	25
MaxEpochs	40

Fig.3 Simulation parameter

Then we use the prepared training set to train with the CNN architecture of this study. Training process can be seen in Fig.4. The training process is represented by the loss function curve, the X-axis is the number of iterations, and the Y-axis is the training loss. As the number of training increasing, the training loss decreases continuously, and when the number of iterations achieving 16500, the convergence of the loss function is realized.

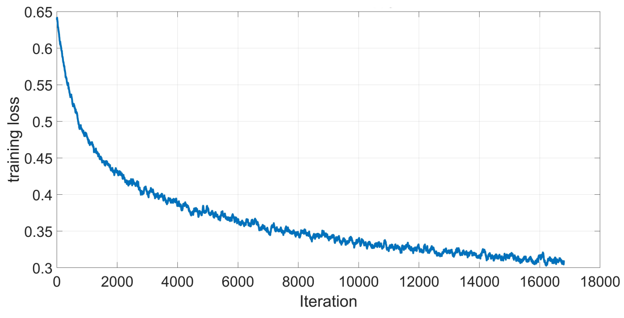

Fig.4 Training process

We use recognition accuracy as the result of this study. The description of recognition accuracy rate is as shown in (7).

$$\textit{Recognition accuracy} = \frac{\sum_{i=1}^{N} I\left(y_i == F(x_i)\right)}{N} \quad (7)$$

Here, $F(x_i)$ is the classification of experimental results, y_i is the true classification. Thereinto(8)

$$I\left(y_i == F(x_i)\right) = \begin{cases} 1, y_i = F(x_i) \\ 0, y_i \neq F(x_i) \end{cases} \quad (8)$$

When the E_b/N_0 < -10dB, the model has a recognition bias problem. At this time the recognition accuracy of the other three methods is low, and the BPSK recognition accuracy is significantly higher than the other three methods. This is because for the model of this study, since constellation diagrams of the signal are too scattered and almost the same under this stage of E_b/N_0, the recognition ability of the model is low, and it is easy to produce the recognition bias for a certain modulation method. Therefore, we do not consider the recognition accuracy of the model at the E_b/N_0 < -10dB, and the model cannot recognize the signal modulation method under this stage of E_b/N_0.

Recognition accuracy of signal modulation methods can be seen in Fig.5. We can find from the experimental results that four modulation methods can be recognited under low E_b/N_0, but there are some differentiations on recognizing ability for different modulation methods when E_b/N_0 is low. The reason is that different CNN models can find diffrernt characteristics from the same image, therefore different CNN models have different ability to recognize modulation methods. For this study's CNN model, there is the strongest recognizing ability on 16QAM method and the weakest recognizing ability on BPSK method, when E_b/N_0 < -5dB. Then when the $E_b/N_0 \geq$ -10dB, as with the improvement of the value of E_b/N_0, the recognition accuracy of four modulation methods is significantly improved. Especially when the $E_b/N_0 \geq$ -5dB, the recognition accuracy of all four modulation methods are higher than 90%.

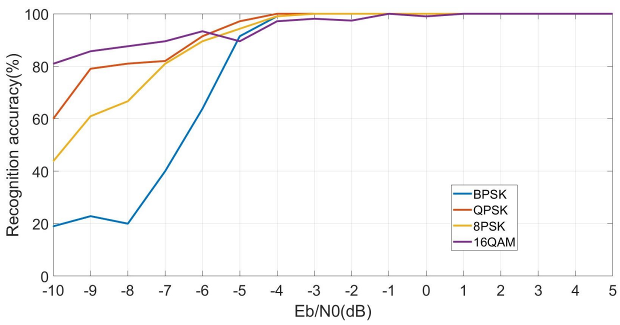

Fig.5 Recognition accuracy of four signal modulation methods

5. Conclusion

In this study, we proposed a method of high efficiency and high precision for automatic recognition signal modulation methods using CNN with constellation diagrams, and used four common digital modulation methods, BPSK, QPSK, 8PSK, 16QAM. We achieved modulation recognition under low E_b/N_0, and when signals of which the $E_b/N_0 \geq$ -5dB, the recognition accuracy of all four modulation methods are higher than 90%.

References

[1] A. Ameen, A. Khaled and E.-T. Mohamed, "Automatic modulation classification based on high order cumulants and hierarchical polynomial classifiers," *Physical Communication*, vol. 21, pp. 10-18, Sept. 2016.

[2] S. Kharbech, I. Dayoub and M. Zwingelstein-Colin, et al, "On classifiers for blind feature-based automatic modulation classification over multiple-input-multiple-output channels," *IET Communitication*, vol. 10, pp. 790-795, May. 2016.

[3] L. Han, H. Xue and F. Gao, et al, "Low complexity automic modulation classification based on order statistics," *IEEE Transactions on Wireless Communications*, vol. 16, pp. 400-411, Jan. 2017.

[4] G. Niu, G. Yao and Y. Yan, "Modulation Recognition Algorithm of Digital and Analog Mixed Signal Based on Neural Network," *Computer Engineering*, vol. 42, no. 4, pp. 101-104, Apr. 2016. (in Chinese)

[5] A. Afan, Y. Fan and S. Liu, "Automatic modulation classification of digital modulation signals with stacked autoencoders," *Digital Signal Processing*, vol. 71, pp. 108-106, Sept. 2017.