

# Spectrogram-Based Automatic Modulation Recognition Using Convolutional Neural Network

Sinjin Jeong, Uhyeon Lee, and Suk Chan Kim  
Department of Electrical and Computer Engineering  
Pusan National University  
Busan, Korea  
{sjjeong, u.h.lee, sckim}@pusan.ac.kr

**Abstract**—We study a system for classifying modulation types with spectrograms obtained through short-time Fourier transform. AWGN-based carrier modulated signals and their spectrograms are generated. In order to extract features from spectrogram automatically, we learned our convolutional neural network model with the generated data. Even at low SNRs, the performance is fairly good, but additional modulation type applications and comparisons with others in various environments are necessary.

**Keywords**—Spectrogram, Automatic Modulation Recognition, Convolutional Neural Network, Short-Time Fourier Transform

## I. INTRODUCTION

Automatic Modulation Recognition (AMR) is a technology that automatically classifies modulation types without prior knowledge of the received signal. It was first motivated by its application in military scenarios where electronic warfare, surveillance and threat analysis requires the recognition of signal modulations in order to identify adversary transmitting units, to prepare jamming signals, and to recover the intercepted signal. AMR is also most important for the application of link adaptation (LA). If the receiver can recognize the modulation type automatically, it is possible to transmit and receive data by changing modulation type according to the channel state without adding the modulation information to the signal frame. [1]

Most modern approaches to AMR rely on analyzing known features derived from advanced signal processing. These feature-based methods distinguish the modulation schemes from features obtained by Fourier transform, wavelet transform, higher order statistics, cyclostationary analysis, and template matching. [2] These methods while powerful, can suffer loss of generality when faced with dynamic RF environments. In recent years, deep neural networks (DNN) has been used in many other fields by showing remarkable results in computer vision, natural language processing, and speech recognition. DNN can automatically identify features that are difficult to artificially find because they consist of many layers and parameters. It has been shown an overwhelming performance compared to expert features, but it requires a lot of data. Fortunately, in the case of communications data can be generated relatively easily.

Recently, researches have been conducted to recognize the modulation schemes by automatically extracting features by applying convolutional neural network (CNN) and long short-term memory (LSTM). While these studies use baseband signals, we use carrier modulated signals with the assumption that there is no information about the center frequency. [3] [4] [5]

In this paper, we describe and analyze the overall simulation process and results of signal modeling, data generation, CNN architecture, and learning. AWGN-based carrier modulated signal is used and input data is generated by short-time Fourier transform (STFT) of this signal. The generated input data is labeled with modulation schemes and the entire data set is divided into training and test set. Our convolutional neural network architecture has 13 layers to recognize 7 modulation types including BFSK, 4FSK, 8FSK, BPSK, QPSK, 8PSK, and 16QAM. After explaining learning procedures of the CNN model, we analyze the test set results according to SNR and modulation types.

## II. SIGNAL MODEL & INPUT DATA

Let the received modulated signal  $r(t)$  be described as

$$r(t) = \text{Re}\{C(t)e^{j2\pi f_c t}\} + n(t)$$

where  $C(t)$  is the complex envelope of the modulated signal,  $n(t)$  is an additive white Gaussian noise,  $f_c$  is the carrier frequency, and  $\text{Re}\{\cdot\}$  denotes the real part. Pulse shaping, fading channel, and frequency offset can be applied to the signal model, but only the AWGN environment is considered for simplicity. The input data of the CNN model is generated through preprocessing of this signal.

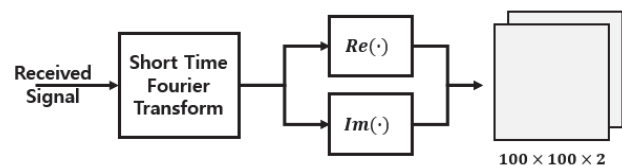


Figure 1. CNN Input Data Generation Process

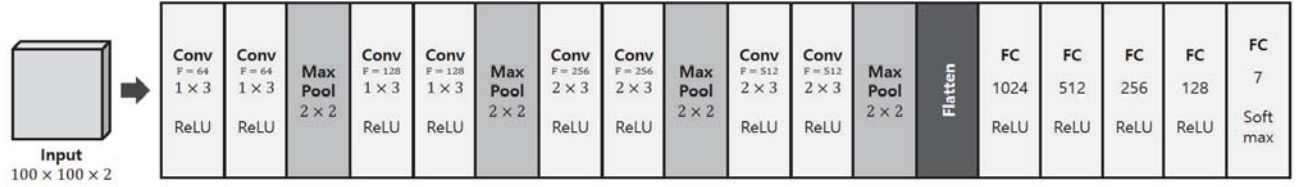


Figure 2. Proposed Convolutional Neural Network Architecture

Short-time Fourier transform (STFT) is a method that can check the frequency and phase characteristics over time by dividing the whole signal into several sections through a window and the performing fast Fourier transform (FFT) on each of the sections. We have adjusted this window length to symbol duration, and additional research is needed on the relationship between window length and symbol duration. The frequency range of the FFT was 5 Hz to 500 Hz at 5 Hz intervals. The input data generation process is shown in the Figure 1. Through the STFT of the received signal, a  $100 \times 100$  complex number array is created and divided into real and imaginary parts to generate a real number array of  $100 \times 100 \times 2$ . The final array is three dimensional and the dimensions have information about time, frequency, and phase, respectively. We used a CNN model to extract and identify the features of each modulation types from this array.

### III. CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE

CNN is a kind of DNN that is used for computer vision. Generally, it includes various convolution and pooling filter layers, and represents output values after fully-connected layers. The convolution filter recognizes a specific pattern from the input image and pooling layer extracts key feature values. Since the features found by the convolution filters are obtained automatically by learning CNN model with data, a lot of good data must be applied to learning to get good results. We generate signals and spectrograms which were applied only AWGN. The model learned with these data are likely to show poor results for data in new environments such as fading channels. Learning data generated in various environments can produce a robust model in such environments.

The proposed CNN architecture is shown in Figure 2. In general, CNN used in computer vision has a three-dimensional array as input. Color information (RGB) is added as a third dimension to a 2D array having a pixel values. We added phase information in the third dimension instead of RGB. Since there is no significant difference from the general CNN architecture used in computer vision, the preceding studies can be applied without difficulty. Our model consists of an input layer of  $100 \times 100 \times 2$ , 8 convolution layers (Conv), 4 max-pooling layers (Max Pool), 4 fully-connected layers (FC) and a 7-label output layer. The seven output labels refer to BFSK, 4FSK, 8FSK, BPSK, QPSK, 8PSK, and 16QAM, respectively. To prevent overfitting problems, we applied a dropout (dropout rate = 0.3) at fully-connected layer. Rectified linear unit (ReLU) is used as an activation function in most layers, and softmax is used

in the output layer. Convolution filters of  $1 \times 3$  and  $2 \times 3$  are used to focus on changes in signal over time. Hyperparameter tuning can improve performance by finding the optimal filter size and number of filters.

### IV. TRAINING & TEST PROCEDURES

Data for training and testing are generated by MATLAB. The parameters of the generated signal are shown in Table 1. We generate a total of 7,000 training data with uniformly distributed SNR from -10dB to 20dB. The 1400 test data were generated at each SNR with 3dB intervals.

CNN architecture modeling and learning were performed using Keras. Learning of the CNN model consists of three stages: forward-propagation, back-propagation, and parameter updating. Forward-propagation is used to calculate the outputs of each layers and loss value. Back-propagation is used to calculate the derivatives of the loss value respect to parameters. Parameters are updated in the direction that the loss is decreased with the calculated derivative values. The initial values of parameters were determined using Glorot initializer. We use categorical cross-entropy as a loss function and optimization is performed with Adam optimizer. The number of epochs to train the model is 99.

Table 1. Modulation parameters

Parameter	Value
Sampling frequency, $F_s$	1,000 Hz
Carrier frequency, $F_c$	100 Hz
Symbol rate, $R_s$	25 Hz
No. symbols, $N_s$	100

### V. RESULTS

The Figure 3 shows the confusion matrix at 2dB and -4dB SNR. At 2dB SNR, very high accuracy was shown, but at -4dB SNR it shows errors between QPSK and 8PSK. In the Figure 4, the accuracy of test set according to SNR is gradually decreased from 2dB SNR below. In this study, we have conducted experiments to classify seven digital modulation types. It is necessary to apply more modulation types and further studies on a neural network architecture suitable for each modulation types should be performed.

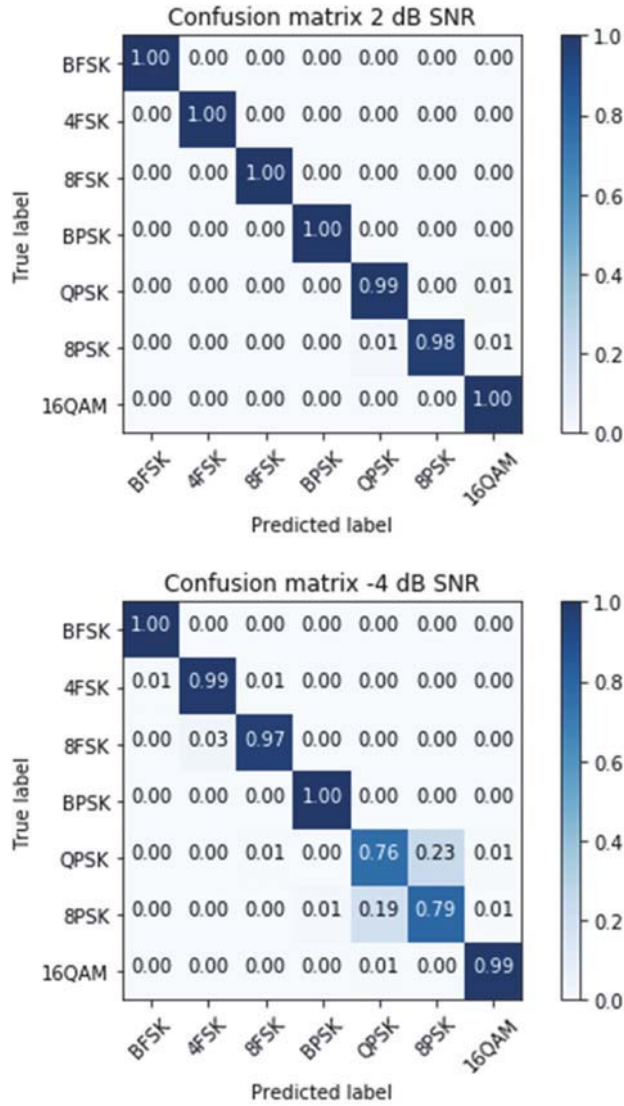


Figure 3. Confusion Matrix at 2dB and -4dB SNR

## VI. CONCLUSIONS & FUTURE WORK

Even at low SNRs, it has a high accuracy. Additional hyperparameter tuning and data learning can enhance performance. Because proposed CNN architecture is deep, high computational complexity can be a disadvantage. The structure should be simplified by repeating the structural change and verification. When the feature value is extracted from the

baseband signal, the frequency offset error due to the Doppler effect may degrade the performance. Since the method proposed in this paper finds pattern from spectrogram, it can be robust to frequency offset error. Therefore, it is necessary to analyze performance according to various environments and variables such as fading channel, pulse shaping, and STFT window length as well as frequency offset. Spectrogram is also expected to be able to distinguish multiple sub-carrier modulation such as OFDM. [6] In addition, by applying the CNN model used for object detection, two or more signal modulations can be simultaneously recognized.

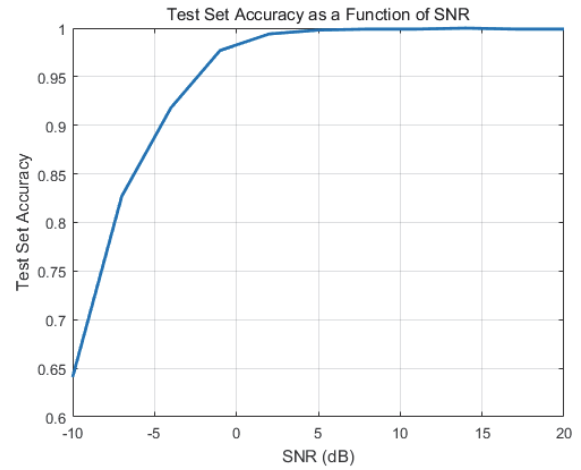


Figure 4. Test Set Accuracy as a Function of SNR

## REFERENCES

- [1] Zhu, Zhechen, and Asoke K. Nandi. *Automatic modulation classification: principles, algorithms and applications*. John Wiley & Sons, 2014.
- [2] Hazza, Alharbi, et al. "An overview of feature-based methods for digital modulation classification." *Communications, Signal Processing, and their Applications (ICCSPA), 2013 1st International Conference on*. IEEE, 2013.
- [3] Karra, Krishna, Scott Kuzdeba, and Josh Petersen. "Modulation recognition using hierarchical deep neural networks." *Dynamic Spectrum Access Networks (DySPAN), 2017 IEEE International Symposium on*. IEEE, 2017.
- [4] O'Shea, Timothy J., Johnathan Corgan, and T. Charles Clancy. "Convolutional radio modulation recognition networks." *International Conference on Engineering Applications of Neural Networks*. Springer, Cham, 2016.
- [5] West, Nathan E., and Tim O'Shea. "Deep architectures for modulation recognition." *Dynamic Spectrum Access Networks (DySPAN), 2017 IEEE International Symposium on*. IEEE, 2017.
- [6] Norouzi, S., A. Jamshidi, and A. R. Zolghadrasli. "Adaptive modulation recognition based on the evolutionary algorithms." *Applied Soft Computing* 43 (2016): 312-319.