

PAPER • OPEN ACCESS

Spectrum Sensing for Cognitive Radio Based on Feature Extraction and Deep Learning

To cite this article: Yue Geng *et al* 2022 *J. Phys.: Conf. Ser.* **2261** 012016

View the [article online](#) for updates and enhancements.

You may also like

- [Comparison of Single User with Multi User Cooperative Spectrum Sensing for Energy Detector at Low SNR Wall](#)
Kavita Bani and Vaishali Kulkarni
- [A Comprehensive Survey on Effective Spectrum Sensing in 5G Wireless Networks through Cognitive Radio Networks](#)
P Ramakrishnan, P T Sivagurunathan and Dr.N Sathishkumar
- [A two-stage spectrum sensing scheme based on energy detection and a novel multitaper method](#)
Pei-Han Qi, , Zan Li et al.



The Electrochemical Society
Advancing solid state & electrochemical science & technology

242nd ECS Meeting

Oct 9 – 13, 2022 • Atlanta, GA, US

Early hotel & registration pricing
ends September 12

Presenting more than 2,400
technical abstracts in 50 symposia

The meeting for industry & researchers in

BATTERIES
ENERGY TECHNOLOGY
SENSORS AND MORE!



Register now!



**ECS Plenary Lecture featuring
M. Stanley Whittingham,**
Binghamton University
Nobel Laureate –
2019 Nobel Prize in Chemistry



Spectrum Sensing for Cognitive Radio Based on Feature Extraction and Deep Learning

Yue Geng^{1,*}, Jingyi Huang², Jianxin Yang³, Sen Zhang⁴

¹School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore

²College of Physics and Optoelectronic Engineering, Shenzhen University, Shenzhen, 518061, China.

³School of Mechanical Engineering & Automation, Beihang University, Beijing, 100191, China.

⁴School of Measurement-Control and Communication Engineering, Harbin University of Science and Technology, Harbin, 150080, China.

YGENG001@e.ntu.edu.sg

Abstract. In cognitive radio, spectrum sensing is used to determine whether the primary user is using the spectrum based on the signal received on a specific frequency band, thereby determining whether the secondary user can use the spectrum. The main problem faced by spectrum sensing is how to identify the existence of the primary signal under the condition of low signal-to-noise ratio (SNR). Compared with traditional technologies, deep learning methods can identify the features of input data more efficiently and accurately. Based on convolutional neural network (CNN), This paper regard spectrum sensing as a binary classification problem. In the method we proposed, different features of received are extracted, and a dataset of feature matrices obtained under different SNRs is constructed for the training of the CNN network. Experiment results show that under the condition of low signal-to-noise ratio, the performance of our method is improved compared with the traditional method, and the combination of different features can improve the sensing accuracy.

1 Introduction

For the past few years, the increasing demand for higher data rates requires more spectrum resources. However, with the rapid development of wireless communication technology such as 5G, the spectrum resources become rarer and rarer [1]. To further improve spectrum utilization, cognitive radio technology has been proposed [2]. It is a method that can let secondary users (SU) reuse the spectrum in an unlicensed way while licensed primary users (PU) are inactive. Thus, unused spectrum resources can be utilized. For this purpose, SUs needs to frequently detect the state of the spectrum occupation of the PUs and feedback the result of spectrum sensing. To avoid interference, SUs has to identify the existence of the primary signal under the condition of a low signal-to-noise ratio (SNR) to know whether it is available or not [3].

Tradition spectrum sensing methods use different signal features, including energy detection [4,5], pilot detection [6,7], cyclostationary detection [8,9], and so on. The problem is that these methods can't achieve good performance when the SNR is relatively low. Compared with traditional methods, deep learning and CNN has more powerful capabilities in extracting spatial features of the input. With the



convolutional layer, CNN can extract deeper features of the input and achieve higher recognition accuracy. Many deep learning methods have been used for spectrum sensing. In [10], the authors used received signal sequence as the input of the CNN and determine whether the received signal contains primary signal. The problem was that the method didn't extract signal features, and the signal sequences will be hard to recognized when the SNR is low. In [11], the authors used power spectrums of the received signal as the input of the CNN, while the power of signal will be severely affected when the SNR is low. In [12], the authors used both energy and cyclostationary feature as the input of the CNN. The problem of this method was that cyclostationary features can be represented more comprehensively. A spectrum sensing method based on feature extraction and CNN is proposed in this paper. In the work, a received signal sequence is represented by energy, power spectrum and cyclostationary feature. When a SU sense the spectrum, it samples the received signal and extract the three features. Then, the three features are combined and a feature matrix is built. The matrix is fed in the CNN. By the convolution operation, the three features can be calculated simultaneously, and the signal can be analyzed more comprehensively.

The rest of the paper is organized as follows. In Section II, how to extract different signal features and build feature matrix is introduced. In Section III, a CNN network is constructed and models are trained, then the result is analyzed. In section IV, whether the combination of three features is better than signal feature for spectrum sensing based on CNN is tested. Finally, the conclusion is made in Section V.

2 Feature Extraction

Spectrum sensing can be considered as a binary classification problem:

$$\begin{cases} H_0 = n(t) \\ H_1 = s(t) + n(t) \end{cases} \quad 0 \leq t \leq T \quad (1)$$

where T represents sampling time. H_0 represents received signal only contains noise $n(t)$, and H_1 represents received signal contains primary signal $s(t)$. Primary signals are always bandpass modulated signals, including ASK, PSK, and so on. In our method, SUs will sample the received signal and extract the energy feature, power spectrum feature and cyclostationary feature respectively. To combine the three features, we use two 64×1 vector to present energy feature and power spectrum feature, one 64×2 matrix is used to represent cyclostationary feature. Finally, the three features are combined to construct a 64×4 feature matrix to present a received signal sequence. The steps to build the matrix are as follows.

2.1 Energy feature

To extract the energy feature of the received signal sequence, the signal sequence is evenly divided into 64 segments, and the average energy of each segment is calculated. The test statistic for energy feature extractor is expressed by:

$$T(y) = \frac{1}{N} \sum_{n=1}^N |y(n)|^2 \quad (2)$$

where, $y(n)$ is the signal samples and N is the number of signal samples in one segment. Then the energy feature vector can be expressed as $T_{energy} = (T_1, T_2, \dots, T_{64})^T$. In the work, the vector is used to express the energy feature of a received signal sequence.

2.2 Power spectrum feature

Signal power is another characteristic of the signal, and the power spectral density can reflect the power distribution of the signal. Assume that the finite-length received signal sequence is $x(n)$, the estimation of power spectral density can be expressed by:

$$S_x(k) = \frac{1}{N} |X(k)|^2 = \frac{1}{N} |FFT[x(n)]|^2, \quad k = 0, 1, \dots, N-1 \quad (3)$$

where, $FFT[x(n)]$ is the Fourier transform of the received signal sequence $x(n)$. Since the period of $FFT[x(n)]$ is N , the calculated power spectrum estimate takes N as the period. By using the Welch power spectrum density estimator in MATLAB, the power spectrum of a signal samples sequence can be obtained. One of the results is shown in Figure 1.

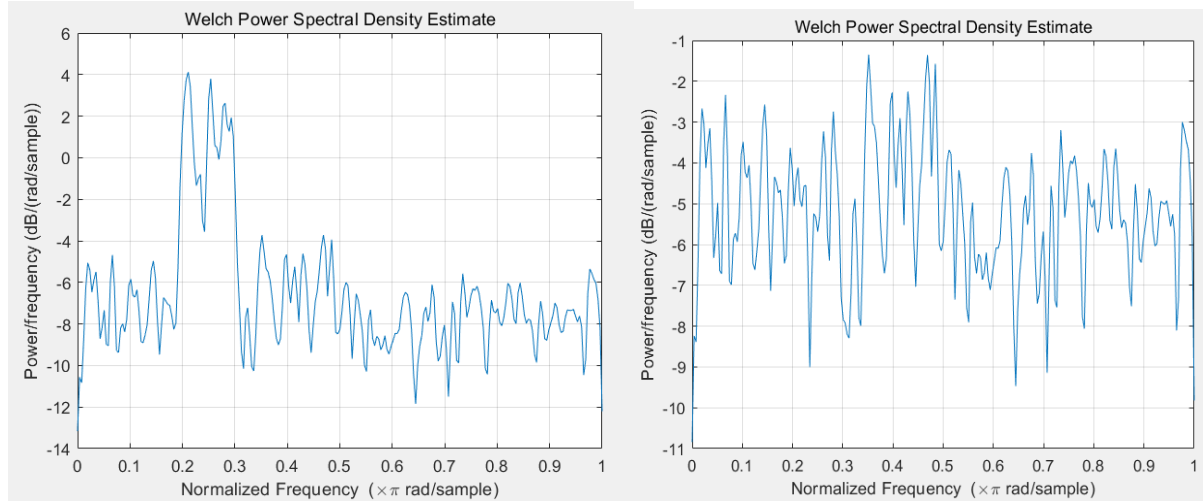


Figure 1. The result of Welch power spectrum density estimator of two received signal sequences. Left is the power spectrum density of the signal sequence which contains primary signal and SNR is 0. Right is the power spectrum density of the signal sequence which only contains noise.

Then, the spectrum is sampled for 64 times and another 64×1 power spectrum feature vector $P_{power} = (P_1, P_2, \dots, P_{64})^T$ is obtained.

2.3 Cyclostationary feature

Modulated signals have periodic autocorrelation. The autocorrelation function is expressed by:

$$R_x(t, \tau) = E\{x(t)x(t - \tau)^*\} \quad (4)$$

where, τ is a constant lag. Autocorrelation function is periodic in time t only for T_0 . It can be expressed by:

$$R_x(t, \tau) = R_x(t + T_0, \tau) \quad (5)$$

Then, the Fourier transform of autocorrelation function can be expressed by:

$$R_X(\alpha, \tau) = F(R_x(t, \tau)) \quad (6)$$

where, α is cyclic frequency. Then the spectral domain transform can be calculated through 2-D transform in frequency and cyclic frequency:

$$S_x(\alpha, f) = F(R_X(\alpha, \tau)) \quad (7)$$

Cyclostationary feature of a signal sequence can be get through the communication toolbox in MATLAB. One of the cyclostationary feature of the signal samples sequence is shown in Figure. 2.

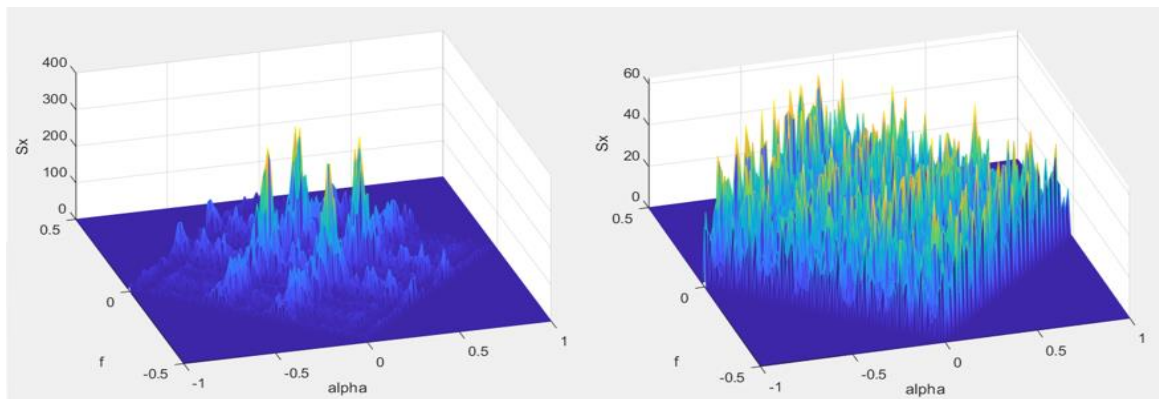


Figure 2. The cyclostationary feature in spectral domain of two received signal sequences. Left is the cyclostationary feature of the signal sequence which contains primary signal and the SNR is 0. Right is the cyclostationary feature of the signal sequence which only contains noise.

It can be found that the feature values are mainly focused on the straight line corresponding to $\alpha = 0$ and $f = 0$. To express the cyclostationary feature, the α and f are set as 0 respectively, then the spectral correlation is sampled for 64 times and two 64×1 cyclostationary feature vectors CS_α and CS_f is obtained. Finally, through feature extraction, the 64×4 feature matrix of a received signal sequence is obtained. The format of the matrix is expressed in Table 1. As shown in Table 1, the first column is the energy feature vector, the secondary column is the power spectrum vector, the third and fourth column is the cyclostationary feature vectors.

Table 1. The format of the feature matrix of a received signal sequence.

Features Samples	T_{energy}	P_{power}	CS_α	CS_f
1	T_1	P_1	$CS_{\alpha 1}$	$CS_{f 1}$
2	T_2	P_2	$CS_{\alpha 2}$	$CS_{f 2}$
...
64	T_{64}	P_{64}	$CS_{\alpha 64}$	$CS_{f 64}$

3 Deep Learning Based System Model

In cognitive radio system, a SU determines whether the spectrum is idle or busy depending on the received signal, then decides whether it can reuse the spectrum. The system structure proposed in this paper can be shown in Figure 3.

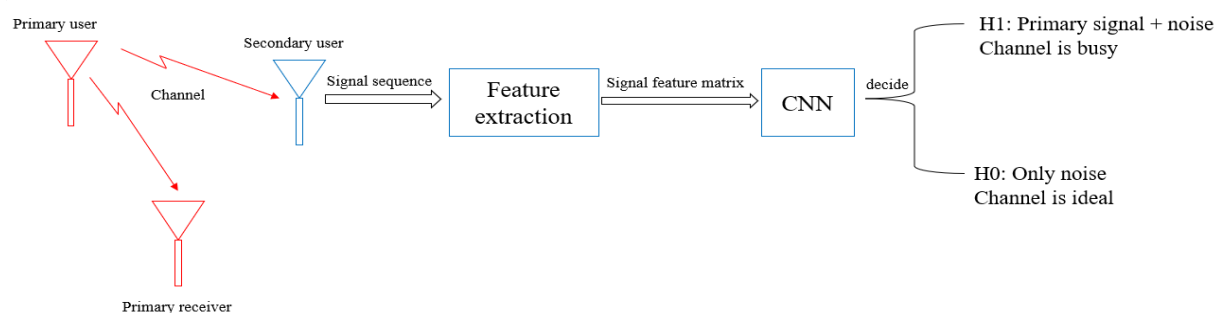


Figure 3. The structure of feature extraction and CNN based spectrum sensing system

To sense the spectrum, the SU samples the received signal. Then the signal samples sequence is fed in the feature extractor to generate the feature matrix corresponding to the signal sequence. Then the matrix is fed in CNN model. The CNN model will do binary classification to decide whether the received signal contains primary signal or not.

3.1 CNN network architecture

CNN has achieved good results in areas such as image classification tasks [13]. In our method, we also consider spectrum sensing as a binary classification task. To evaluate whether the feature matrix can be used to train CNN model and do spectrum sensing, a CNN network is constructed in the work and models are trained, then the performance is evaluated. The CNN network can be termed as “SenseNet”, and the structure of the network is shown in Figure 4.

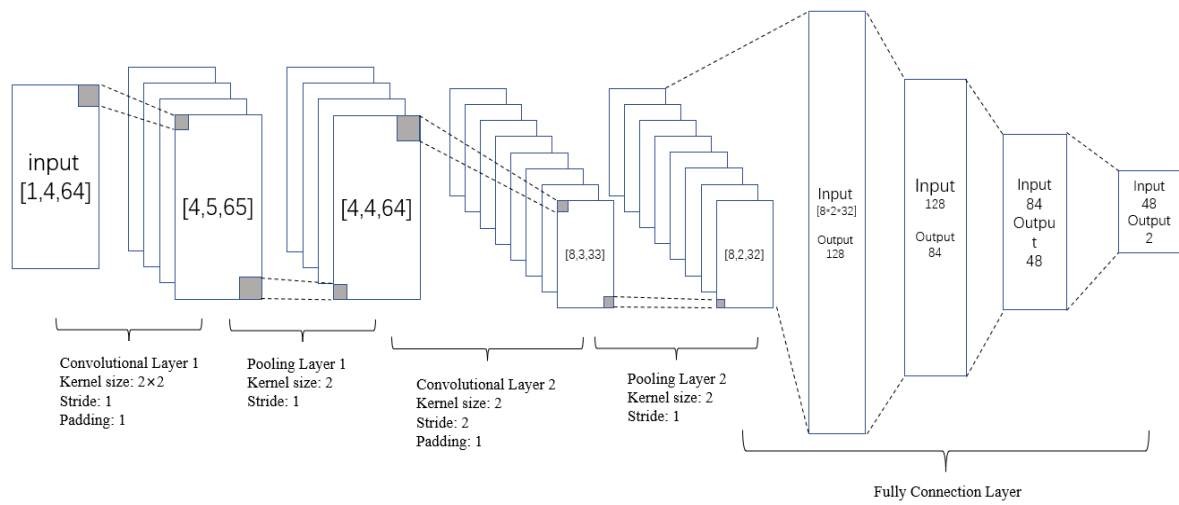


Figure 4. The structure of the “SenseNet”.

As shown in Figure 4, The format of the input layer is equal to the format of the 64×4 feature matrix formed by the extracted signal features. The two convolution layer uses four 2×2 convolution kernels and Zero-padding is 1, stride is set as 1 and 2, correspondingly. 8 different feature maps are obtained in the end of two convolution layers. The operations of the convolution layer can be expressed by:

$$X(l) = f(W(l) \otimes X(l-1) + b(l)) \quad (8)$$

where, $X(l)$ is the output of the l convolutional layer. $W(l)$ is the weight of the l convolutional layer. $b(l)$ is the bias of the l layer. \otimes is convolution operation, and f is activate function. In the method, Sigmoid is used as activate function. Sigmoid function can be expressed by:

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (9)$$

The sigmoid function image is an S-shaped curve with a value range between (0,1), as shown in Figure 5. Sigmoid function can increase the nonlinearity of the network and map an input to the interval of (0,1).

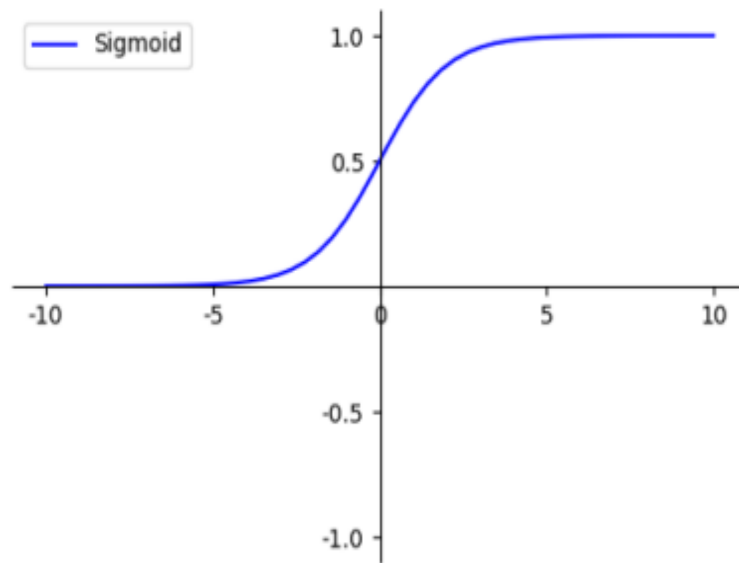


Figure 5. Sigmoid activation function.

Pooling layer can remove unimportant data and further reduce parameter need to be trained. Max pooling layer is added after the convolutional layer, the kernel size is 2 and the stride is 1. Finally, four 2×32 feature maps are obtained after two convolutional later and two pooling layer.

Four fully connection layers are added in the end of the CNN. Fully connection uses Softmax as activate function. Each neuron in each layer of the fully connection layer is fully connected with all neurons in the previous layer, and the data features extracted by the convolution are analyzed and integrated, and the final prediction classification evaluation result is output. Finally, two outputs are set. The two outputs correspond to H_0 and H_1 .

3.2 Dataset generation

To generate a dataset to train the network, In MATLAB, two kinds of bandpass modulated signals QPSK and 8PSK are simulated, the carrier frequency is 10kHz. The pulse shaping filter used is raised cosine filter with roll-off factor of 0.5. The signal passes through a multi-path Rayleigh fading channel, path delays vector is [0s, 0.001s]. The noise is assumed to be Gaussian white noise. It is set that the signal passes through AWGN channel, the SNR changes from - 20 to 5.

In the method, it is assumed that SU samples the signal for 1280 times. After passing through the Rayleigh channel and AWGN channel, the received signal is sampled 1280 times, and the sample sequence is fed in feature extractor to generate feature matrixes. The train dataset contains two type “signal 0” and “signal 1”. The “signal 0” only contains the feature matrixes corresponding to noise samples, and the “signal 1” contains the feature matrixes corresponding to the signals including primary signal and noise. Every signal type contains more than 5000 data. To evaluate the performance of the model under different SNRs, the test dataset that contains two types of data under different SNRs is generated, and every dataset corresponding to one SNR contains more than 1000 data.

3.3 Experiment results

The model is trained on the train dataset, and the trained model is tested on the test dataset. Based on PyTorch library, the training process can be expressed in Figure. 6.

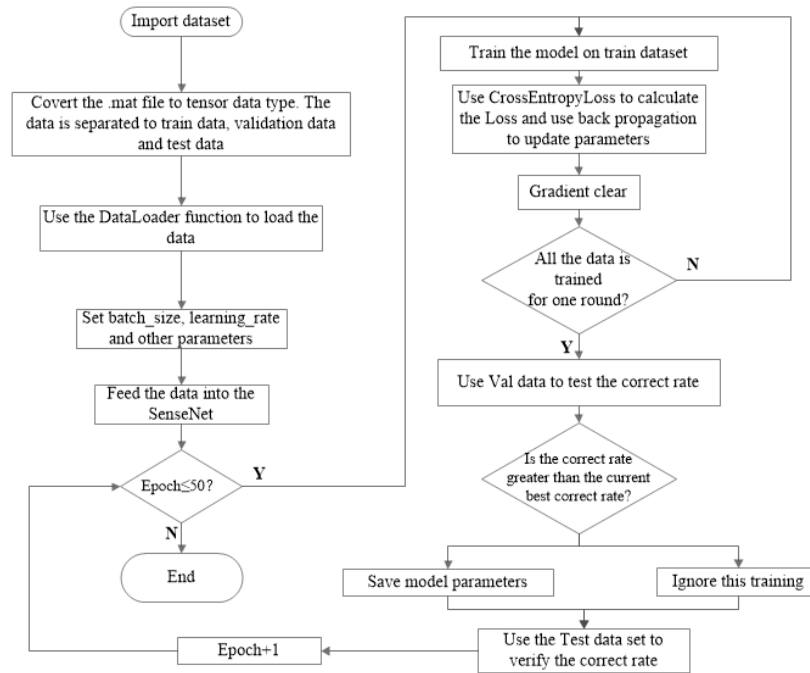
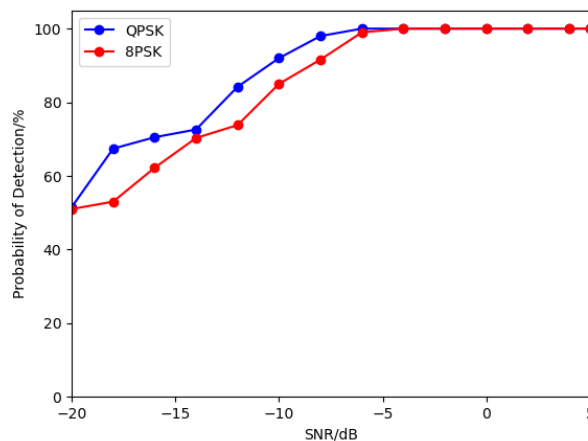


Figure 6. The process of model training. Dataset and DataLoader class provided in PyTorch is used to load data. The network is set to be trained for 50 epochs, and the model with best sense accuracy is saved.

Then, the test dataset is used to evaluate the performance of the spectrum sensing model. The performance of spectrum sensing is evaluated through two parameters probability of detection (P_d) and probability of false alarm (P_f), which can be expressed as:

$$P_d = P(\text{decide } H_1 | H_1) \quad P_f = P(\text{decide } H_1 | H_0) \quad (10)$$

To evaluate the performance of the model by the two parameters, for QPSK and 8PSK signal sequences, P_d and P_f are calculated under different SNRs. The results are shown in Figure. 7.



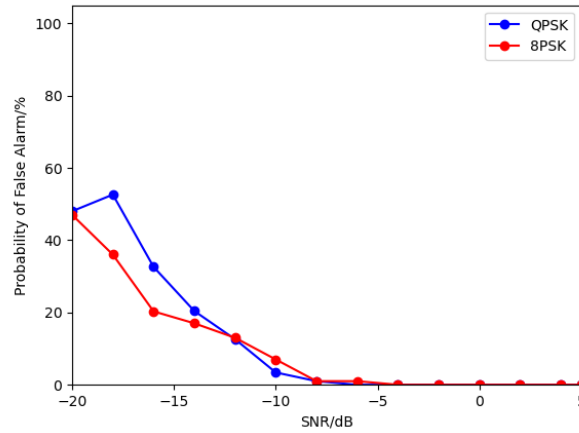


Figure 7. The sense performance of the model for QPSK and 8PSK signals.

As shown in Figure 7, with the increase of SNR, P_d increases and P_f decreases. When the SNR is larger than -10, P_d will exceed 80% and P_f will be less than 10%. Compared with traditional spectrum sensing method [4-9], this method has better performance when the SNR is relatively low. Through the experiment results, it can be found that for both signal types, when the SNR is small (< -14), the performance of the model is relatively bad. The probability of detection is not high enough while the probability of false alarm is not low. While if SNR is large enough (> -10), the performance is good. The probability of detection is high (> 80) and the probability of false alarm is almost 0. The results verifies that the feature extraction can be used for spectrum sensing based on deep learning.

4 Feature Types Comparison

To further evaluate whether the combination of different feature types can improve the sensing accuracy, the data containing only energy and power features and the data containing only cyclostationary feature to train the models is used to train model respectively. Every type of the two data is 64×2 matrix. The structure of the network is the same, while the input format is changed from 64×4 to 64×2 .

4.1 Experiment results

Used the same method proposed in Section III, the test results for the two types of the data are obtained. The test results are shown in Figure. 6 and Figure 7.

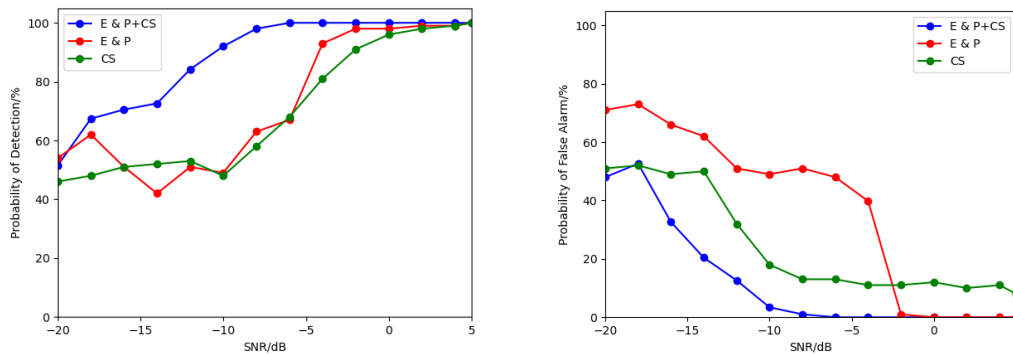


Figure 8. The performance corresponding to different data types for QPSK signal. Where, E & P refers to energy feature with power feature, CS refers to cyclostationary feature.

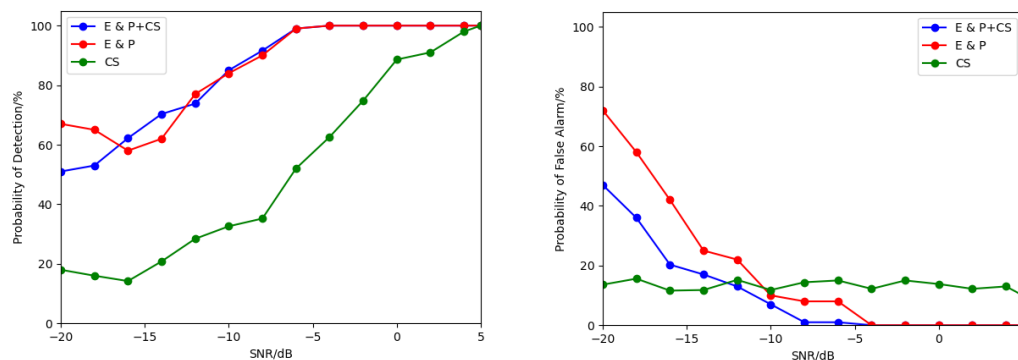


Figure 9. The performance corresponding to different data types for 8PSK signal. Where, E & P refers to energy feature with power feature, CS refers to cyclostationary feature.

4.2 Result discussion

As shown in Figure 8 and Figure 9, through the experiment results, it can be found that use the combination of the three feature types to build the data and train model can achieve better sense accuracy. The results show that when the SNR is lower than -6dB, the model that trained on the energy and power feature can't achieve good performance, that is while the P_d is high, the P_f is also high. It means that the model can't recognize the signal type accurately, each type is likely to be recognized as another type.

Compared with energy and power features, when cyclostationary feature is used to do spectrum sensing, the P_f is relatively low. However, the P_d is not high enough when the SNR is lower than 0dB. it can be found that by combining all the kinds of features, the CNN model can make tradeoff between different features, which means the model can reduce P_f and improve P_d to a certain extent. The results verify that by combining several types of features, CNN model can learn received signal sequence more comprehensively and make more accurate decision.

5 Conclusion

In this paper, for spectrum sensing, a method based on feature extraction and deep learning is proposed. In the work, a novel feature extractor is proposed and the feature matrix extracted is used to train network. Finally, the model trained can be used to do spectrum sensing. The experiment results show that the method can achieve better performance than traditional spectrum sensing methods. To further test whether the method combining more types of signal features can improve sense performance, different data types corresponding to different combinations of features is generated and different models are trained. The experiment results verifies that the performance of the method that combines different features is better than that only use one or two kinds of feature. Based on the method, more kinds of signal features and advanced CNN structure can be explored to improve the performance of spectrum sensing models.

6 References

- [1] Ali A, Hamouda W. Advances on spectrum sensing for cognitive radio networks: Theory and applications[J]. IEEE communications surveys & tutorials, 2016, 19(2): 1277-1304.
- [2] Ahmad W S H M W, Radzi N A M, Samidi F S, et al. 5G technology: Towards dynamic spectrum sharing using cognitive radio networks[J]. IEEE Access, 2020, 8: 14460-14488.
- [3] Arjoune Y, Kaabouch N. A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions[J]. Sensors, 2019, 19(1): 126
- [4] Saad M A, Mustafa S T, Ali M H, et al. Spectrum sensing and energy detection in cognitive networks[J]. Indonesian Journal of Electrical Engineering and Computer Science, 2020, 17(1): 465-472.

- [5] Sarala B, Devi S R, Sheela J J J. Spectrum energy detection in cognitive radio networks based on a novel adaptive threshold energy detection method[J]. *Computer Communications*, 2020, 152: 1-7.
- [6] Salahdine F. Spectrum sensing techniques for cognitive radio networks[J]. *arXiv preprint arXiv:1710.02668*, 2017.
- [7] Bharathy G T, Rajendran V, Tamilselvi T, et al. A Study and Simulation of Spectrum Sensing Schemes for Cognitive Radio Networks[C]//2020 7th International Conference on Smart Structures and Systems (ICSSS). IEEE, 2020: 1-11.
- [8] Ye Z, Grosspietsch J, Memik G. Spectrum sensing using cyclostationary spectrum density for cognitive radios[C]//2007 IEEE Workshop on Signal Processing Systems. IEEE, 2007: 1-6.
- [9] Jang W M. Blind cyclostationary spectrum sensing in cognitive radios[J]. *IEEE communications letters*, 2014, 18(3): 393-396.
- [10] Gao J , Yi X , Zhong C , et al. Deep Learning for Spectrum Sensing[J]. *IEEE Wireless Communications Letters*, 2019, 8(6):1727-1730.
- [11] Zheng S, Chen S, Qi P, et al. Spectrum sensing based on deep learning classification for cognitive radios[J]. *China Communications*, 2020, 17(2): 138-148.
- [12] Dong H , Sobabe G C , Zhang C , et al. Spectrum sensing for cognitive radio based on convolution neural network[C]// 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI). IEEE, 2018.
- [13] Li Z, Liu F, Yang W, et al. A survey of convolutional neural networks: analysis, applications, and prospects[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.