

# Deep Cooperative Spectrum Sensing Utilizing Recurrent Convolutional Neural Networks

Rahil Sarikhani (✉ [rahil.sarikhani@iauk.ac.ir](mailto:rahil.sarikhani@iauk.ac.ir))

Islamic Azad University Kerman Branch <https://orcid.org/0000-0003-4658-9730>

Farshid Keynia

Graduate University of Advanced Technology

---

## Research

**Keywords:** Cognitive Radio, Cooperative Spectrum Sensing, Convolutional Neural Network, Recurrent Neural Network

**Posted Date:** May 21st, 2020

**DOI:** <https://doi.org/10.21203/rs.3.rs-22343/v1>

**License:** © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

## RESEARCH

# Deep Cooperative Spectrum Sensing Utilizing Recurrent Convolutional Neural Networks

Rahil Sarikhani<sup>1\*</sup> and Farshid Keynia<sup>2†</sup>

\*Correspondence:

rahil.sarikhani@iauk.ac.ir

<sup>1</sup>Department of Computer

Engineering, Kerman Branch,

Islamic Azad University, Kerman,

Iran

Full list of author information is available at the end of the article

<sup>†</sup>Equal contributor

## Abstract

Cognitive Radio (CR) network was introduced as a promising approach in utilizing spectrum holes. Spectrum sensing is the first stage of this utilization which could be improved using cooperation, namely Cooperative Spectrum Sensing (CSS), where some Secondary Users (SUs) collaborate to detect the existence of the Primary User (PU). In this paper, to improve the accuracy of detection Deep Learning (DL) is used. In order to make it more practical, Recurrent Neural Network (RNN) is used since there are some memory in the channel and the state of the PUs in the network. Hence, the proposed RNN is compared with the Convolutional Neural Network (CNN), and it represents useful advantages to the contrast one, which is demonstrated by simulation.

**Keywords:** Cognitive Radio; Cooperative Spectrum Sensing; Convolutional Neural Network; Recurrent Neural Network

## 1 Introduction

Cognitive radio networks (CRN) are exceptional cases of networks which have been designed to utilize the frequency bands dynamically and efficiently. Two leading players of these networks are primary users (PU) who pay for the frequency bands and secondary users (SU) that utilize the vacant bands dynamically. Hence, determining the unoccupied bands is crucial for the SUs due to not interrupt the PUs. In order to detect the empty bands, SUs need to recognize the activity of the PUs, which is called spectrum sensing in the context [1–5].

Although the spectrum sensing is essential in each SU, the individual sensing results are highly disposed to the fading channel conditions and other destructive effects. Consequently, to improve the sensing quality, Cooperative Spectrum Sensing (CSS) is introduced in the literature. CSS utilizes the results of some of cooperating SUs to enhance the quality of sensing and develop the accuracy. However, the channel condition affects the optimal strategy of cooperating in CSS [1]. As an example, the SU that is close to the PU will detect the presence of PU which is very important to share this information with remote SUs to help them in detecting the presence of PU [3, 4]. Therefore, having a unique approach to combine the individual sensing results is very crucial in the accuracy of the CSS.

Accordingly, Combining the individual results become very popular among the researchers to have an optimal approach. Firstly, some simple procedures were introduced based on the hard decisions such as AND-rule, OR-rule, counting rule, and linear quadratic combining rule [6–8]. These are the most common cooperation approaches. Besides, there are some soft decision techniques which utilizes the detected energy levels in the SUs to develop the accuracy [9]. Further, the relay-based

cooperation is studied in [10, 11], too. Moreover, some efficient approaches based on the  $K$ -out-of- $N$  scheme has been introduced to determine the presence of the PU by the positive response of  $K$  SUs among  $N$  SUs [1, 2] in which the independence of the individual sensing is assumed. Moreover, multi-dimensional correlation in individual sensing was considered for the CSS in [3, 4]. Besides, machine learning-based approaches were introduced in [5] which improve the quality of the well-known procedures in the fusion center (FC). These techniques are categorized in unsupervised and supervised categories based on the training method. The authors of [5] studied K-means clustering and Gaussian Mixture Models (GMM) as the unsupervised approaches and Support Vector Machine (SVM) and K-nearest Neighbor (KNN) under the supervised approach. Further, Deep Learning (DL) was utilized in [12] to design a CSS algorithm based on DL, which needs the update of all the measurements in all SUs in each time slot. The main drawback of these approaches is the lack of historical data in the classification process. It has been expressed in [13, 14] that exploiting temporal data of PU states can improve the spectrum sensing relations. Hidden Markov Model (HMM) has been applied as the first employer of the temporal relation in [15], which is hugely dependent on the accuracy of statistical models.

In this paper, we have utilized DL [16] to improve the classification performance of CSS. In order to utilize the previous knowledge of the PU, we have proposed to employ Long-Short-Term-Memory (LSTM) layer. Utilizing temporal relations could improve the classification accuracy. The proposed neural network is composed of three consecutive convolutional blocks to employ the spatial features of different sensing nodes. Besides, each of the convolutional blocks consists of a convolutional layer, batch normalization layer, and rectified linear unit (RELU), which is succeeded by a max-pooling layer to decrease the size of sensing samples. Henceforth, the LSTM layer employs temporal features. Ultimately, the fully connected layer translate the resultant samples into two categories to be alleviated by softmax and classification layers. The most significant novelty of the paper is employing CNN and LSTM together to exploit the spatio-temporal information to improve the classification accuracy. Moreover, the proposed NN can learn the PU activity pattern without any prior information about the PU's activity distribution, such as dwell time and the probability of dwelling. Additionally, the proposed approach is totally model-free, and its accuracy is not in danger by the inaccurate assumptions. Finally, in low Signal-to-Noise Ratios (SNRs), it is more robust than other state-of-the-art approaches, and its performance is less affected than others.

The remainder of the paper is as follows: After system model in Section II, the DL based CSS algorithm is considered in Section III. Subsequently, the simulation results and concluding remarks are expressed to declare the applicability and performance merit of the proposed approach in Section IV and V.

## 2 Methods

### 2.1 System Model

We consider a CRN included of  $N_{SU}$  SU users which try to accompany sensing the spectrum holes to transmit the in-queue data. The positions of the SUs are randomly selected in each scenario. Moreover, some PUs are located randomly in the network

and working in the licensed network. Furthermore, we consider a wideband channel to be sensed with  $N_B$  bands and the total bandwidth of  $W$ .

The PUs can transmit in  $N_P$  bands arbitrarily, and none of the  $N_{SU}$  are aware of the utilized bands. The power of transmitting is fixed at  $P$ . Additionally, the Additive White Gaussian Noise (AWGN) is assumed where the power spectral density is  $N_0$  and  $z_i^j(n)$  is the noise of the  $i$ -th SU in the band  $j$  at the time of  $n$ . Hence, the SU broadcasts the request of cooperative sensing, and all the one-hop neighbors will respond to its appeal by their local sensing results, intermittently. The initiating SU, which requests for cooperative sensing is called *Fusion Center (FC)* in this phase. The FC will combine the local sensing results in the network.

As shown in Fig. 1, the FC begins the DL-based cooperative spectrum sensing and detect the presence or absence of the PU based on the surrounding one-hop neighbors. The number of one-hop neighbor surrounding the agent is  $N$ . In each stage of cooperative sensing, only  $C < N$  nodes are selected based on the reliance of the cooperation, and it would change based on the conditions of the channel and hidden PUs. To broadcast the fusion decision, the FC utilizes a reliable channel to transmit its cooperation consequence and inform other one-hop nodes about the last result. In addition, the channel is assumed to be unavailable if all of the available bands is detected to be occupied by the PU. The same as other references such as [5], the birth-death process is assumed for the simulation of the PU activity characterized by the birth rate  $r_b$  and death rate  $r_d$ . Hence, average probability of being active for the PU is assumed to be  $r_b/(r_b + r_d)$ .

Moreover, the fading and shadowing effect together with the hidden PU makes some SUs unable to detect the presence of the PUs by local sensing. The received signal model in each of the SU could be determined by

$$x_i^j(n) = \begin{cases} g_i^j(n)s^j(n) + z_i^j(n) & \mathbf{H}_1 \\ z_i^j(n) & \mathbf{H}_0 \end{cases} \quad (1)$$

where  $g_i^j(n)$  is the fading factor of the channel between the PU and  $i$ -th SU in frequency band  $j$  in time instant  $n$ , and  $s^j(n)$  is the transmitted sample in the  $j$ -th band. Furthermore, let  $y_{i,j}$  be the received energy samples from the SU  $i$  in the  $j$ -th band as follows

$$y_{i,j} = \frac{1}{N_{ED}} \sum_{n=1}^{N_{ED}} |x_i^j(n)|^2 \quad (2)$$

where  $N_{ED}$  is the number of samples exploited to calculate the received energy level. The SUs transmit their calculated energy to the FC where the received samples by the FC could be formulated by

$$\hat{y}_{i,j}(n) = h_i^j(n)y_i^j(n) \quad (3)$$

where  $h_i^j(n)$  is the channel gain between the  $i$ -th SU and the FC in the  $j$ -th frequency band in time instant  $n$ .

The FC collects all the measurements in the matrix  $\mathbf{Y}(n) = \{\hat{y}_{i,j}(n)\} \in \mathcal{R}^{N_{SU} \times N_B}$ . After collecting all the measurements, it will determine the value function using DL. Finally, the result of the cooperation will be broadcasted to all the neighbors. The proposed RCNN-based DL network for detecting the availability of the channel is expressed.

## 2.2 Proposed Recurrent Convolutional Neural Network

The proposed Recurrent CNN is depicted in Fig. 2. The proposed network is consist of three convolutional blocks which are followed by a max-pooling layer to reduce the amount of the data entering the LSTM layer. Then, a fully connected layer generates the concise samples for softmax and classification. Besides, each of the convolutional blocks is containing three layers called convolutional layer, batch normalization layer, rectifier linear unit (RELU). In the convolutional layer spatial convolution is performed in 2 dimensions to extract the spatial features of data. According to [17], the size of the spatial filter is set to  $3 \times 3$ , which is adequate for spatial feature extraction. Hence, the output of the convolution layer can be denoted by  $Y_{conv}(i, j) = \sum_{p=0}^2 \sum_{q=0}^2 X_{conv}(i + p - 1, j + q - 1) W_{conv}(i, j)$  where  $W_{conv}$  represents the weights and  $X_{conv}$  denotes the input of the convolutional layer. Besides, the input to the first convolutional block is the matrix of detected energies  $\mathbf{Y}(n)$ . Furthermore, padding is used to maintain the size of the input in the output of this layer and stride is set to 1. Then, each batch is normalized utilizing batch normalization layer. At the end of the convolutional block, the non-linearity is introduced by activation function RELU. This layer is added to handle non-linearity in the input data and enabling the proposed RCNN to classify the non-linear behavior [17]. This non-linear layer rejects the negative data and ferries the positive data without any manipulation.

At the end of each convolutional block, there is a max-pooling layer with a corresponding pool size of  $2 \times 2$  where the maximum is extracted. This layer decreases the computation complexity without remarkable performance degradation.

Then, the decreased samples are inserted in LSTM layer with 200 neurons. The number of neurons in LSTM layer is selected experimentally. At first, the number of neurons was 50, then increase it with steps of 20. Dramatic improvements were observed until 200 neurons while increasing neurons more than 200 was not much useful. In the LSTM layer, the output sequence and hidden sequence are calculated by

$$\begin{aligned} h_t &= \mathcal{H}(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \\ y_t &= W_{hy}h_t + b_y \end{aligned} \quad (4)$$

where  $\mathbf{x} = (x_1, x_2, \dots, x_t)$  is the input sequence, and  $\mathbf{h} = (h_1, h_2, \dots, h_t)$  and  $\mathbf{y} = (y_1, y_2, \dots, y_t)$  are the hidden sequence and output sequence, respectively. Additionally,  $W$  is the weight matrix and  $b$  is the bias where the subscript determines the position of the weight and bias. Moreover,  $\mathcal{H}$  is the function of the hidden layer which is responsible for the temporal feature extraction. The hidden function  $\mathcal{H}$  is the well-known LSTM layer function which exploits the memory cells to remember

the temporal information. The implementation of  $\mathcal{H}$  is organized as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (5)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (6)$$

$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (7)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (8)$$

$$h_t = o_t \tanh(c_t) \quad (9)$$

where  $i, f, o$ , and  $c$  are *input*, *forget*, *output*, and *cell* gates activation vectors, and  $\sigma$  is the logistic sigmoid function and all of them are the same size of the hidden vector  $h$ . Moreover, the matrices  $W_{ji}$  for  $j = h, s, c$  (the matrices from cell to gate) are diagonal matrices.

Henceforth, fully connected layer is used to introduce the abstract features to soft-max unit and classification. Additionally, in this layer, the weights are multiplied by the input data, and biases are added to the resultant data. Specifically, the output is obtained via  $\mathbf{W}_{FC}\mathbf{X}_{FC} + \mathbf{b}_{FC}$ , where  $\mathbf{W}_{FC}$  and  $\mathbf{b}_{FC}$  are the weights and biases, respectively and  $\mathbf{X}_{FC}$  is the input to the fully connected layer. Soft-max layer generate the classification input by utilizing exponential function as  $\frac{\exp(x_r)}{\sum_i \exp(x_i)}$ . Thus, the output is lower than 1 and more than 0. The classification layer determines the existence or absence of the PU in the network according to the abstract spatio-temporal features.

### 2.3 Training of the Proposed Network

To train the proposed network, a set of labeled data is generated. In each of the data ensemble, the channel state is changed until the averaging in different channel models is ascertained. Using these labeled data, the weights and biases of the proposed network are estimated to minimize the cross-entropy loss function. The cross-entropy is minimized utilizing stochastic gradient descent with Momentum (SGDM). In order to generate the labeled data, we have changed three main parameters, stochastically. These three parameters are channel conditions, positions of cooperating nodes, the node-wise permutation of gathered data in the input of the proposed NN. Each of these parameters has a significant effect on performance. Theoretically, position permutation will affect the fading coefficients which are encountered by SUs and the fusion center, which will affect the energy measurements report. Moreover, channel conditions will affect the fading factors, too. Additionally, the node-wise permutation of the gathered data will determine the order of the impact of the near or far node on the proposed network. Consequently, we consider all the sources of the stochastic behavior of the problem in the training phase to decrease the error of the classification network.

## 3 Results and Discussion

Here, we will evaluate the performance of the proposed algorithm to the other machine learning based approaches. All the simulations are done on MATLAB software. A single PU is assumed which is located randomly among number of SUs in an area limited by 500 meters in 2 dimensions. Furthermore, the number of frequency bands

is set to  $N_B = 16$  where each sub-bands occupies 1MHz of bandwidth. Moreover, the transmitter power is assumed to be  $P = 30$  dBm, and  $\beta = 3000$ ,  $\alpha = 3.8$ , and  $\sigma = 7.9$  dB [18]. Additionally, the training of the required coefficients in DL is performed based on adaptive moment estimation algorithm. Furthermore, spatially correlated fading is assumed for spatial correlation sensing which is modeled based on  $\rho_{A-B} = e^{(\frac{d_{A-B}}{d_{ref}})}$  with  $d_{ref} = 50$  m [19].

To evaluate the performance of the proposed algorithm, we have compared the proposed approach with other machine learning based cooperative spectrum sensing including SVM, and DL-based CSS in [12] approaches and HMM based CSS approach as the state-of-the-art schemes. Furthermore, the comparison metrics are the probability of miss detection ( $P_m$ ) to the probability of false alarm ( $P_f$ ) and probability of detection in different SNRs and the impact of the number of SUs in sensing error. Sensing error which is considered as the sum of probability of false alarm ( $P_f$ ) and the probability of the detection ( $P_d$ ). Additionally, the three probability metrics  $P_m$ ,  $P_f$ ,  $P_d$  are denoted by the following:

$$\begin{aligned} P_m &= Pr(\mathbf{H}_0|\mathbf{H}_1) \\ P_f &= Pr(\mathbf{H}_1|\mathbf{H}_0) \\ P_d &= Pr(\mathbf{H}_1|\mathbf{H}_1). \end{aligned} \quad (10)$$

In Fig. 3, the comparison of  $P_m$  to the corresponding  $P_f$  of the four mentioned schemes is provided. In this simulation number of SUs are 30, and the number of samples in each SU is  $N_{ED} = 128$ . Apparently, the proposed approach outperforms others in a higher probability of false alarms, while in a lower probability of false alarm, the result of the DL-based approach is comparable to the proposed approach. Furthermore, the HMM-based approach is utilizing the history of the PU activity while the decisions in two others are independent of history.

The impact of different SNRs on the probability of detection is depicted in Fig. 4. Apparently, the proposed approach is superior to the others in lower measured SNRs. While increasing the SNR will increase the  $P_d$ , and different sensing approaches getting closer for higher SNRs. The results of the simulation are averaged over 1000 independent runs.

Eventually, the sensing error is represented in Fig. 5, where several cooperating users and number of samples are changed. As depicted, the sensing error in both cases for the proposed approach is more reliable than others. Specifically, two initial results are the lowest sensing error and lowest fluctuations in the sensing error by changing the number of samples and users.

## 4 Conclusion

In this paper, the cooperative spectrum sensing is considered in CRN by utilizing the machine learning techniques, specifically Recurrent CNN. Here, we have proposed a Recurrent CNN based algorithm to sense the spectrum holes, cooperatively. In this algorithm, we have used the LSTM layer to accurately exploit the temporal correlation and convolutional layer to employ the spatial correlation. Hence, the proposed scheme utilizes spectrum spatio-temporal features to alleviate the cooperative sensing and improve the accuracy the sensing. Eventually, the simulations

represent the superiority of the proposed approach to the other state-of-the-art techniques in the literature.

#### Declarations

##### Abbreviations

AWGN	Additive White Gaussian Noise
CNN	Convolutional Neural Network
CR	Cognitive Radion
CSS	Cooperative Spectrum Sensing
DL	Deep Learning
FC	Fusion Center
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
KNN	K-nearest Neighbor
LSTM	Long-Short-Term-Memory
PU	Primary User
RNN	Recurrent Neural Network
ReLU	Rectified Linear Unit
SGDM	Gradient Descent with Momentum
SNR	Signal-to-Noise Ratio
SVM	Support Vector Machine
SU	Secondary User

##### Ethics Approval and Consent to Participate

Not applicable.

##### Consent for Publication

Not applicable.

##### Availability of Data and Material

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

##### Competing Interests

The authors declare that they have no competing interests.

##### Fundings

There is no source of funding to achieve the findings in this manuscript.

##### Author's contributions

The simulation results and the writing were prepared by Sarikhani, while all the works where supervised by Keynia.

##### Acknowledgements

Not applicable

##### Author details

<sup>1</sup>Department of Computer Engineering, Kerman Branch, Islamic Azad University, Kerman, Iran. <sup>2</sup>Department of Energy Management and Optimization, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology, Kerman, Iran.

##### References

1. A. H. Gazestani and S. A. Ghorashi, "Distributed diffusion-based spectrum sensing for Cognitive Radio sensor networks considering link failure," *IEEE Sensors J.*, vol. 18, no. 20, pp. 8617–8625, Oct. 2018.
2. J. So and W. Sung, "Group-based multibit cooperative spectrum sensing for cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 10193–10198, Dec. 2016.
3. D. Xue, E. Ekici, and M. C. Vuran, "Cooperative spectrum sensing in cognitive radio networks using multidimensional correlations," *IEEE Trans. Wireless Commun.*, vol. 13, no. 4, pp. 1832–1843, Apr. 2014.
4. A. W. Min and K. G. Shin, "An optimal sensing framework based on spatial RSS-profile in cognitive radio networks," in *Proc. of IEEE SECON*, Rome, Italy, Jun. 2009.
5. K. M. Thilina, K. W. Choi, N. Saquib, and E. Hossain, "Machine learning techniques for cooperative spectrum sensing in cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 11, pp. 2209–2221, Nov. 2013.
6. A. Ghasemi and E. S. Sousa, "Spectrum sensing in cognitive radio networks: The cooperation-processing tradeoff," *Wireless Commun. and Mobile Comput.*, vol. 7, no. 9, pp. 1049–1060, Nov. 2007.
7. E. C. Peh, Y.-C. Liang, Y. L. Guan, and Y. Zeng, "Optimization of cooperative sensing in cognitive radio networks: A sensing-throughput tradeoff view," *IEEE Trans. Veh. Technol.*, vol. 58, no. 9, pp. 5294–5299, Nov. 2009.



8. J. Unnikrishnan and V. V. Veeravalli, "Cooperative sensing for primary detection in cognitive radio," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 18–27, Feb. 2008.
9. Z. Quan, W. Ma, S. Cui, and A. H. Sayed, "Optimal linear fusion for distributed detection via semidefinite programming," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2431–2436, Apr. 2010.
10. G. Ganesan and Y. G. Li, "Cooperative spectrum sensing in cognitive radio - part I: Two user networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 6, pp. 2204–2213, Jun. 2007.
11. W. Zhang and K. B. Letaief, "Cooperative spectrum sensing with transmit and relay diversity in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4761–4766, Dec. 2008.
12. W. Lee, M. Kim, and D.-H. Cho, "Deep Cooperative Sensing: Cooperative Spectrum Sensing Based on Convolutional Neural Networks," *IEEE Trans. on Veh. Technol.*, vol. 68, no. 3, pp. 3005–3009, Mar. 2019.
13. S. Geirhofer, L. Tong, and B. M. Sadler, "Cognitive radios for dynamic spectrum access-dynamic spectrum access in the time domain: Modeling and exploiting white space," *IEEE Commun. Mag.*, vol. 45, no. 5, 2007.
14. Y. Saleem and M. H. Rehmani, "Primary radio user activity models for cognitive radio networks: A survey," *J. Netw. Comput. Appl.*, vol. 43, pp. 1–16, 2014.
15. H. Eltom, S. Kandeepan, Y. C. Liang, B. Moran, and R. J. Evans, "HMM based cooperative spectrum occupancy prediction using hard fusion," in *Proc. Commun. Workshops, Int. Conf. Commun. (ICC), Kuala Lumpur, Malaysia*, May 2016.
16. S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed lowcost spectrum sensors," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 3, pp. 433–445, 2018.
17. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
18. IEEE 802.20, "IEEE 802.20 channel models document," January 2007.
19. A. Algans, K. Pedersen, and P. Mogensen, "Experimental analysis of the joint statistical properties of azimuth spread, delay spread, and shadow fading," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 3, pp. 523–531, Apr. 2002.

# Figures

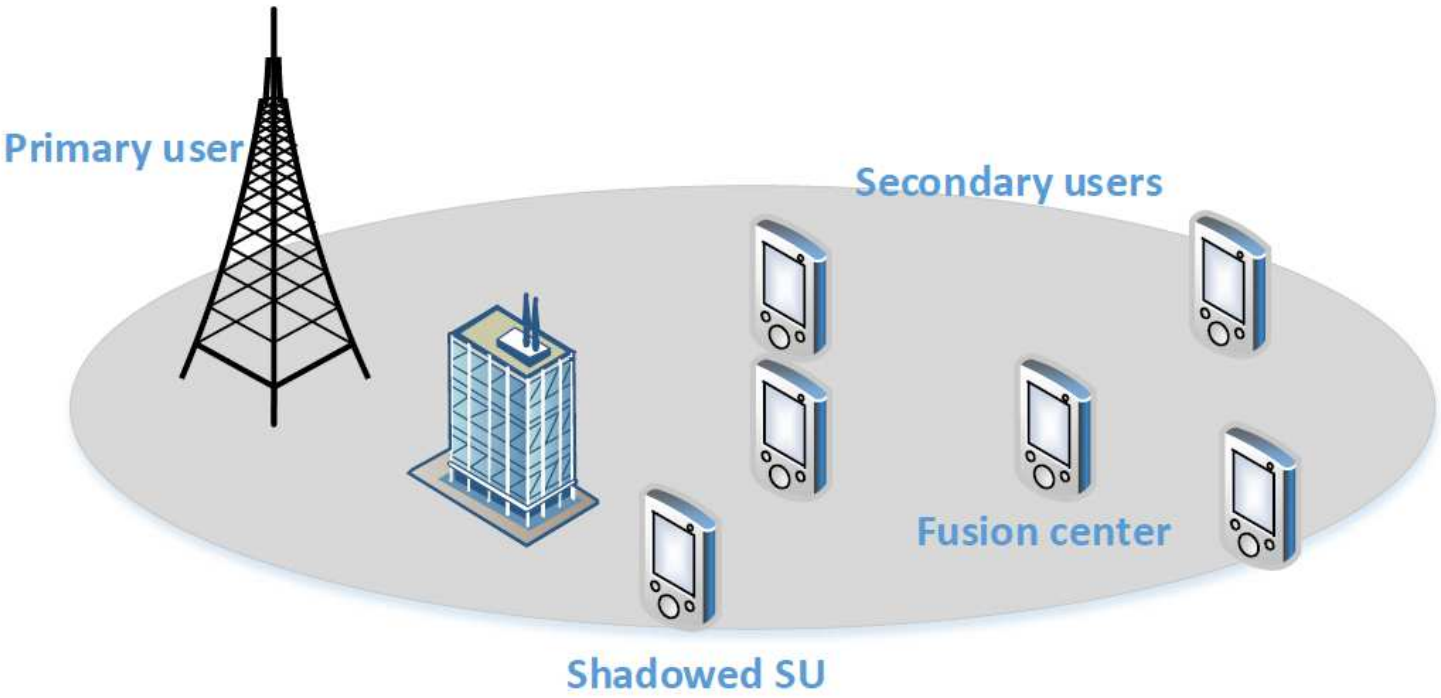


Figure 1

System model of the Cognitive Radio network

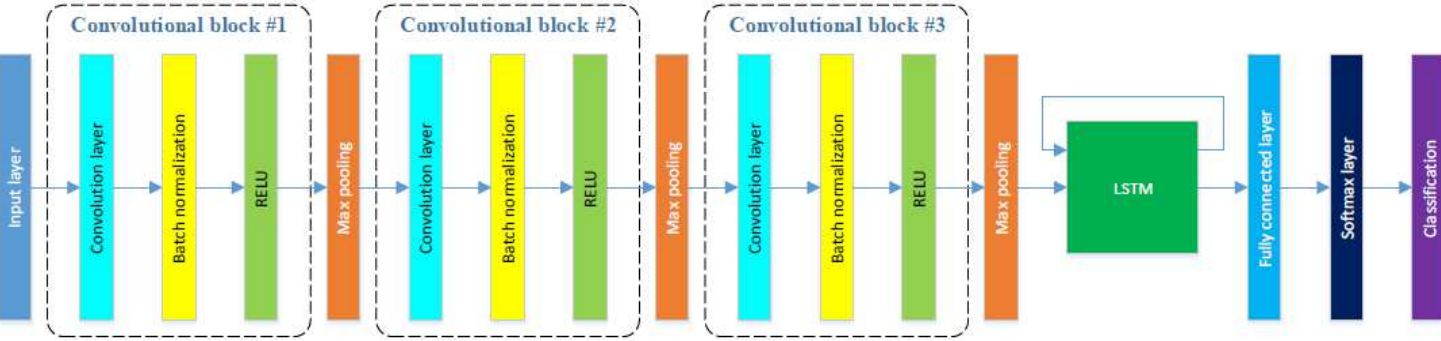
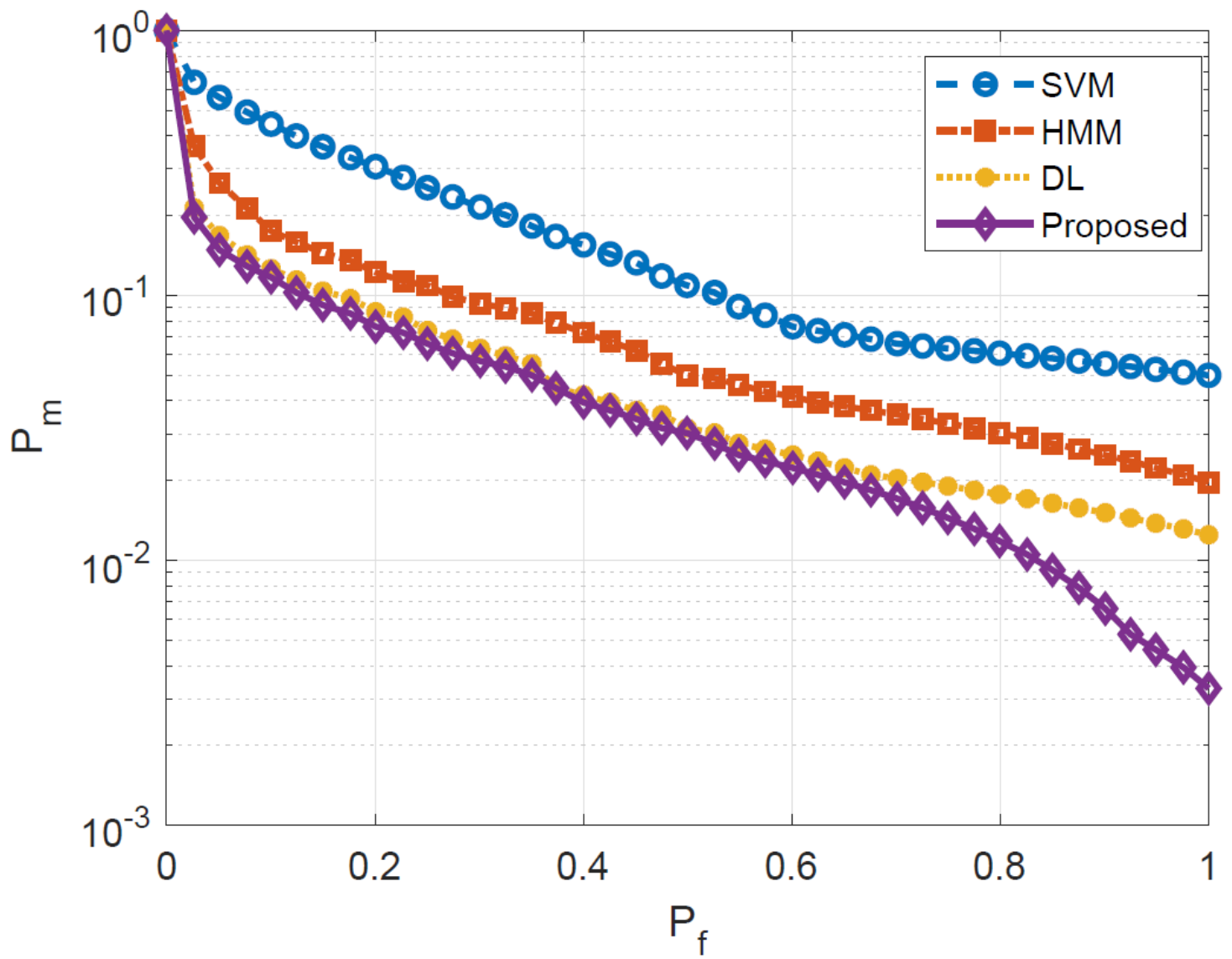


Figure 2

The proposed RCNN for CSS



**Figure 3**

The comparison of  $P_m$  vs.  $P_f$  for different types of procedures

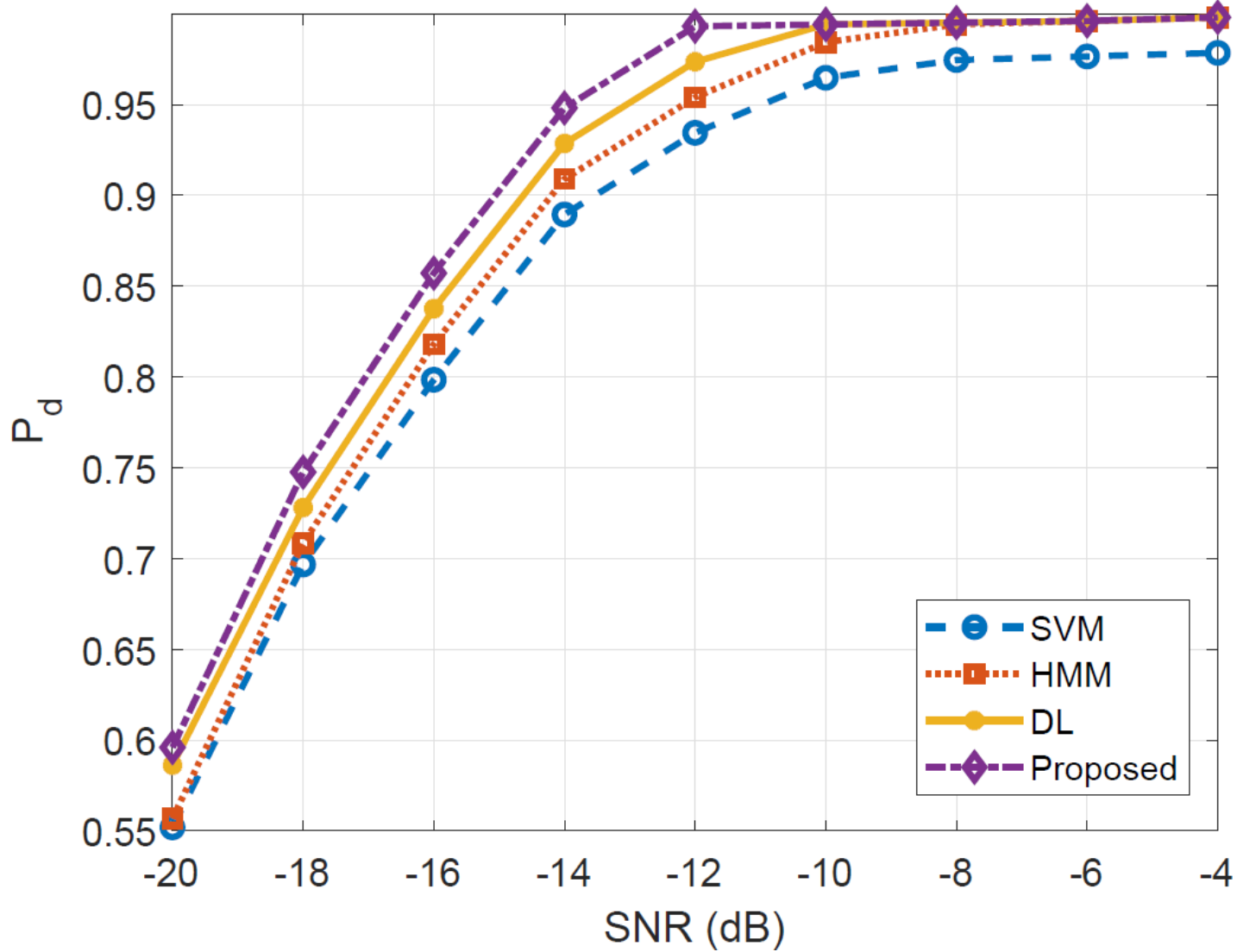


Figure 4

The comparison of  $P_d$  in different SNRs for different types of procedures

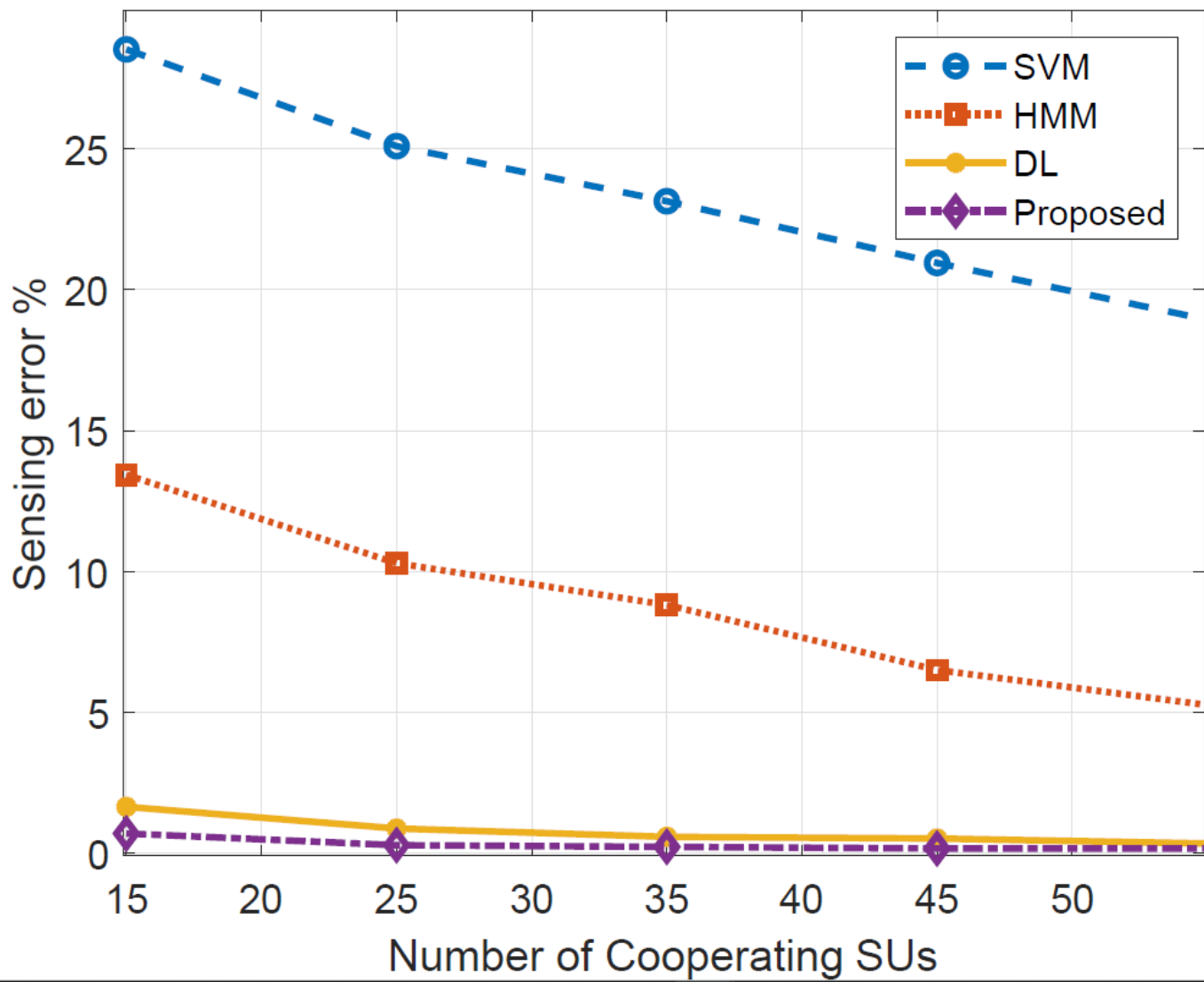


Figure 5

The comparison of sensing error for different types of procedures