Peer-to-Peer Networking and Applications (2021) 14:3235–3249
https://doi.org/10.1007/s12083-021-01169-4

Deep learning application for sensing available spectrum for cognitive radio: An ECRNN approach

S. B. Goyal[1] · Pradeep Bedi[2] · Jugnesh Kumar[3] · Vijaykumar Varadarajan[4]

Received: 10 February 2021 / Accepted: 20 April 2021 / Published online: 7 June 2021
© Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

Spectrum sensing (SS) is a concept of cognitive radio systems at base transceiver stations that can find the white space i.e. licensed spectrum owned by primary users (PU), for transmission over a wireless network without any channel interference. The cognitive radio network is designed to overcome the problem of the limited radio frequency spectrum as most of the applications are dependent on wireless devices in 5G. The major concern that arises here is the detection of spectrum availability. The traditional approaches can solve this issue but consume a large amount of time and prior information about PU and spectrum. The objective of this paper is to give a solution to resolve such issues. In this paper, we have used the learning capabilities of deep learning algorithms such as Convolution neural network (CNN) and Recurrent neural network (RNN) for spectrum sensing without prior knowledge of PU. The proposed model is termed ensemble CNN and RNN (ECRNN) to learn the features of spectrum data and predict the spectrum availability at base transceiver stations in 5G. The simulation result of the ECRNN showed the improvement of accuracy of the system with a reduction in losses that occurred during the false alarm of prediction as well as an improvement in the probability of detection. ECRNN had analyzed PU statistics and result in better spectrum sensing. This paper also supported multiple SUs that would increase the speed of spectrum sensing and data transmission over the available limited spectrum at the same time.

Keywords Cognitive radio · 5G · Spectrum sensing · Deep learning · Probability of detection

1 Introduction

The advancement of 5G technologies and modern wireless communication systems had led to the scarcity of spectrum resources [1]. From different studies, it has been reported that there is a variation of spectrum usage from 7% to 34%. So, to overcome the scarcity of the limited spectrum resources, Cognitive radio (CR) appeared as a potent approach that can balance the trade-off of demand and availability of spectrum resources [2, 3]. The main concept of CR is to reuse the

This article is part of the Topical Collection on *Special Issue on Cognitive Models for Peer-to-Peer Networking in 5G and Beyond Networks and Systems*
Guest Editors: Anil Kumar Budati, George Ghinea, Dileep Kumar Yadav and R. Hafeez Basha

✉ S. B. Goyal
drsbgoyal@gmail.com

Pradeep Bedi
pradeep.kcu21@gmail.com

Jugnesh Kumar
jugnesh@rediffmail.com

Vijaykumar Varadarajan
v.varadarajan@unsw.edu.au

[1] Faculty of Information Technology, City University, Malaysia, Petaling Jaya, Malaysia

[2] Department of Computer Science & Engineering, Lingayas Vidyapeeth, Faridabad, India

[3] St Andrews Institute of Technology and Management, Gurgaon, India

[4] Department of CSE, University of New South Wales, Kensington, Australia

available unused frequency bands. These are also termed white spaces or spectrum holes. This method also ensures that there is no interference in the spectrum of licensed users [4]. The licensed user is technically termed a primary user (PU) whereas unlicensed users are termed as a secondary user (SU) (Fig. 1). The CR technology allows SU to access the available unused spectrum frequency bands in a non-interfering way to PU [5]. This makes spectrum sensing highly robust and efficient. An intelligent, multi-dimensional, adaptive, and wireless communication device that learns from its experience, plans, and determines future behavior to meet customer needs, can be described simply as a cognitive radio [6]. Cognitive radio has two major characteristics. One is the cognitive capacity that collects the information from its radio environment is the skill of cognitive radio technology. The second is reconfigurability that makes it possible to dynamically program the cognitive radio according to the radio environment necessary.

The four key functions of Cognitive Radio [7] are spectrum sensing, management, sharing, and mobility. Radio is continually looking for the unused bandwidth known as the void in the spectrum. This cognitive radio property is known as spectrum sensing. Once the spectrum holes are located, the available hole or channel is chosen by the cognitive antenna. This cognitive radio property is referred to as spectrum management. As long as the primary user does not require it, the property of cognitive radio to delegate the spectrum holes to secondary users is called spectrum sharing. It is the property where, when a licensed (primary) user is identified, the cognitive radio (CR) vacates the channel.

One of the aspects of 5G transmission is spectrum sensing for fast data transmission and utilization of limited spectrum

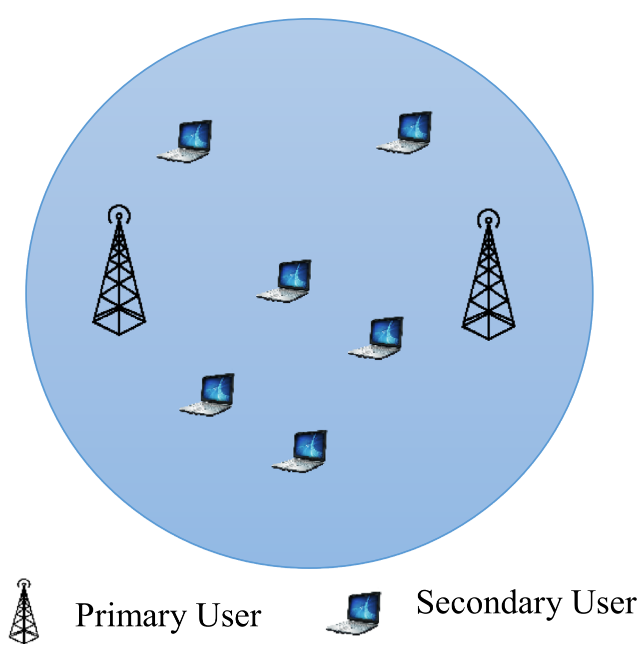


Fig. 1 An Example of Cognitive Radio Network

band. Empty spectrum was utilized for the elimination of congestion created due to traffic of a large amount of data. An efficient spectrum sensing algorithm is integrated with current 5G technologies. There is no way for disruption or delay of communication. Radio frequencies only can obtain spectrum sensing in cognitive radio [8]. To make the idea of cognitive radio performance, witnessing a licensed user's unused spectrum is important. Thus, the primary user is sensed to enable the mobility of the SU's channel in another part of the spectrum; if the primary user initiates the transmission. Efficient hardware is needed with minimal error. The detection threshold is the key. The intervention in the worst-case scenario should be considered. Future study of the spectrum and decisions rely on the right sensing of the primary consumer. This is known as the dynamic management of the spectrum.

Parametric and non-parametric schemes are two categories for spectrum sensing (SS). In the condition of parametric sensing, there is a need for prior PU activity information. Whereas in non-parametric schemes there is no need for any prior information. Therefore, non-parametric SS is preferred over parametric SS [9]. There are some conventional non-parametric (SS) techniques, for example, matched filter, cyclostationary, and energy detection are commonly used to their low computational complexity [10]. The matched-filter detection is used when the CR has previous information about PU. In this condition, a matched-filter can be considered to be the best detection technique. It's precise since the signal-to-noise ratio (SNR) is maximized. The matched filter coincides with the time version of the received signal. The primary user presence is calculated by a contrast between the final output of the corresponding filter and the specified threshold. Therefore, the matched filter will work weakly if this information is not correct. Similarly, a spectrum sensing technique that can distinguish the modulated signal from the additive noise is the implementation of a Cyclostationary function detector. Cyclostationary is a signal, provided it has a normal mean and autocorrelation. The identification of cyclostationary features will differentiate PU signal from noise and use the information present in the PU signal that is not present in the noise at a very low Signal to Noise Ratio (SNR). Due to its low computational and application complexity, energy detection is the most common means of spectrum sensing. No prior information about the primary users is required by the receivers. An energy detector (ED) essentially considers the primary signal as noise and, depending on the energy of the signal detected, determines the presence or absence of the primary signal. Even though these conventional SS methods have low computational complexity but these low detection rate with increasing communication advancements.

With the advancement of communication technologies from 5G to 6G, it is needed not only to adapt to the changing environment but also to adapt its hardware [11]. The current spectrum sensing techniques for cognitive radio network

(CRN) requires the adoption of artificial intelligence or machine learning features. The journey of the communication system towards 6G needs deep learning trained transmission control because the traditional approaches consume a large amount of time and need prior information about PU and spectrum. So, to resolve issues of the traditional approach in CRN, this paper presented an application of computational intelligence algorithms (machine learning or deep learning) due to their learning ability so that they can learn the features of spectrum data and predict the spectrum availability at base transceiver stations in 5G.

1.1 Scope of the research

This paper is focused to design a blind spectrum sensing algorithm with the application of deep learning for cognitive radio (CR) system. The main scope of this paper is to mitigate the limitations of existing spectrum sensing algorithms for PU misdetection and to allow interference-free sensing of the spectrum. Based on the properties of the input data covariance matrix, this paper formulates the application of deep learning for spectrum sensing techniques. This paper employs the approach to detect PU activities in a blind state in which the sensing unit doesn't have prior knowledge about the PU activities or channel state. To overcome the limitations of practical spectrum sensing, aggregation of the most advanced method is expected. The data covariance matrix has different descriptive features such as energy, eigenvalues, etc.

It can be noted that the CNN model can learn the 2-D structured input data matrix. It has the powerful capability to extract correlation features from input covariance matrices. Whereas at the same time RNN can extract temporal features and can find time-shifted correlation features from the covariance matrix. In this paper, we propose the hybrid ensemble approach of CNN and RNN to extract energy correlation as well as temporal correlation to learn the PU's activities and pattern for spectrum sensing.

1.2 Key contributions of research

The key contributions of this paper are as follows:

In this paper, a state-of-the-art about spectrum sensing in cognitive radio is discussed along with detection techniques and associated challenges. Related works of researchers are also focused in this paper to explore their advantages and limitations for further improvement.

- We have proposed an ensemble deep learning model that supports a non-linear function termed ensemble CNN and RNN (ECRNN) to test the presence of PU in data samples.
- Further, we have also conducted simulation analysis under different test conditions to prove the efficiency of the proposed model concerning existing models.

1.3 Organization of Paper

The remaining section of this paper are illustrated to be as follows: Section 2 describes related works about spectrum sensing or detection in cognitive radio networks. In Section 3 paper illustrates the problem statement summarized from existing works. Section 4 gives a descriptive overview of the system model. Section 5 gives information about the performance parameters used. Finally, in Section 6 conclusion, limitations and future research scope are discussed.

2 Related work

In CRN, one of the major research topics for industrial application is spectrum sensing as the demand for high-speed data transmission is increasing day by day. The major function of spectrum sensing technologies is to sense the availability of spectrum. In the last few years, there is the development of different techniques for spectrum sensing for different scenarios such as blind, semi-blind, and non-blind. One of the approaches for the blind scenario was proposed by [9] termed as maximum to average eigenvalue ratio detector (MAER) and arithmetic to the geometric mean detector (AGM) [10] in which there is no need for known noise power. Similarly, in [12], a maximum eigenvalue detector (MED) and generalized likelihood ratio test-based signal subspace eigenvalues detector (GLRT-SSE) [13] was developed for the semi-blind scenario. This is termed semi-blind because there is a need for known noise power. Whereas in the condition of non-blind network scenario, sensing samples are needed for the detection process of PU. It has been reported in [14] that in non-blind conditions, there is the transition of PU from the silent state to transmission state and remains in the same period for the entire process. The Hidden Markov Model [15], had resolved the issues related to PU for such activities. In the current research area, machine learning or deep learning is also proposed for spectrum sensing. The local spectrum sensing quality can be improved by introducing the concept of cooperative spectrum sensing (CSS) whose function is to combine the local sensing information. In [16], the application of deep reinforcement learning (DRL) was adopted to classify the SU signals and resolved the CSS issues by reducing the signaling of SUs. Another deep learning approach such as long short-term memory (LSTM) [17] and convolutional neural networks (CNNs) [18] was proposed to detect available spectrum by learning the correlation between the energy of PU signals. In [19] hierarchical CNN model was proposed to learn co-relation between the energy of PU signals as well as the pattern of PU activities recorded from previous sensing data to enhance future sensing performance. It should be noted that CNN has shown up its capabilities to learn spatial features extracted from signals. At the same time, LSTM had shown up their capabilities for extraction of temporal features from energy correlation samples.

In [20] a combined CNN-LSTM detector was used. The energy correlation features are extracted from multiple sensing inputs and PU activity pattern was learned. The detection probability was increased by analyzing PU activities. The limitation of CNN-LSTM is that its computational complexity is somehow dependent on its input. In [21] spectrum sensing was proposed using LSTM which established the temporal correlation from spectrum data. The PU activity is also exploited to improve the performance of CR. The PU activities such as off period and the duty cycle is used as statistics to train the LSTM network. The detection process and classification accuracy were improved in terms of training time and execution time. The drawback of [21] was observed that it doesn't support multiple PU and SU scenarios which are considered to be a generic scenario. In [22] efficiency of DL is presented for spectrum sensing. But still in these DL algorithms learning process is generally based on a single feature that degrades performance in the noisy scenario. Furthermore, in [23] spectrum sensing is performed using two-autoencoder for OFDM scenario that gives better performance over traditional OFDM. In [24] CNN model is used for cooperative spectrum sensing (CSS) for multiple secondary users in a cognitive radio network (CRN) by using spectral and spatial correlation of each sense. In [25] deep reinforcement learning was used to explore the spectrum sensing issues in CRN. Even though these existing deep learning algorithms improve the detection performance that needs prior statistical knowledge. These methods are vulnerable to noise uncertainty.

3 Problem statement

The main working principle of spectrum sensing techniques to sense the available spectrum at base transceiver stations and to check whether the primary user is present or not. So, this arises an issue to track all channel statistics, spectrum characteristics to predict the available spectrum with high probability. During the last decade, there are much research presented, the most used statistic is the covariance matrix that contains different discriminative detection features. The key problem associated with spectrum sensing traditional techniques there is a requirement of prior knowledge about both PU signal and noise then only optimal performance is achieved. In traditional non-cooperative detection methods such as energy detection or cyclostationary detection algorithm there arise the problem of hidden terminal that generally occurs when cognitive radio is shadowed due to very low SNR values and detection methods cannot SNR sense the PU's presence. Designing an effective and robust spectrum sensing technique is a quite challenging task due to the level of complexity, accuracy, computational cost, error rate, etc. These performance parameters create a trade-off between the spectrum sensing technique and its requirements. Therefore, to resolve these issue that arises a need for prior knowledge about primary users, computational intelligence algorithms showed up their efficiency. But still, there is a need to improvise their performance in terms of probability and accuracy of detection with reduced complexity. So, this paper had adopted a deep learning approach for detection and classification of statistics as PU and SU.

4 Methodology

4.1 System model

In this model, we have considered a multi-antenna scenario of cognitive radio, as shown in Fig. 2. This figure illustrates multi-antenna (A_m) with observation vector (V_n) for spectrum sensing. The spectrum sensing problem is formulated on the following hypothesis, Eqn (1):

$$\begin{aligned} H_0 &: Y_n = U_n \\ H_1 &: Y_n = h_n X_n + U_n \end{aligned} \tag{1}$$

Where, H_0 represents the hypothesis of absence of PU i.e., PU is silent whereas H_1 represents the hypothesis of the presence of PU i.e., PU is in an active state. X_n and Y_n represents the PU transmitted signal vector as well as the received signal vector. $h_n \in C_m$, that represents the channel index between PU and SU. U_n represents the received noise. In some scenarios, it may suffer some path loss or fading. As per the signal vector, we can design the decision statistics that detect PU state to be H_1 in test statistics (T) based on decision threshold (D_s). If the $T > D_s$ then it will represent the presence of PUs otherwise, the PUs are absent. As illustrated in Fig. 2, the conventional framework for spectrum sensing in which transmitted signals are sampled together and the further test statistic is calculated for decision making. The CR will collect all signal vectors from multiple SUs system and further features associated with a signal vector such as energy, covariance matrix, co-relation, etc. to design the decision statistics methods such as ED [7, 26], MED [12], CM-CNN [18], CAV [26], etc. Based on a threshold value, test statistics will compare and finally decide the presence of PU. Hence, it can be stated that test statistics have importance for detection performance improvement. So, in this paper, we have focused on the deep learning model to design decision statistics to show its efficiency over existing techniques.

4.2 CNN-based framework for Spectrum sensing

We have adopted the deep learning approach and termed it as ensemble CNN and RNN (ECRNN). As compared to machine learning, deep learning showed up its proficiency of great learning capacity. Another issue with the machine learning problem is the overfitting problem that is resolved by deep learning.

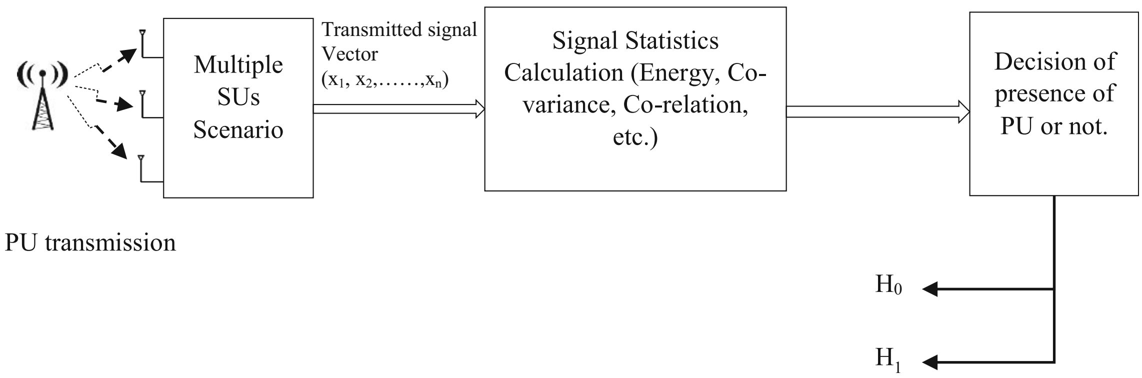


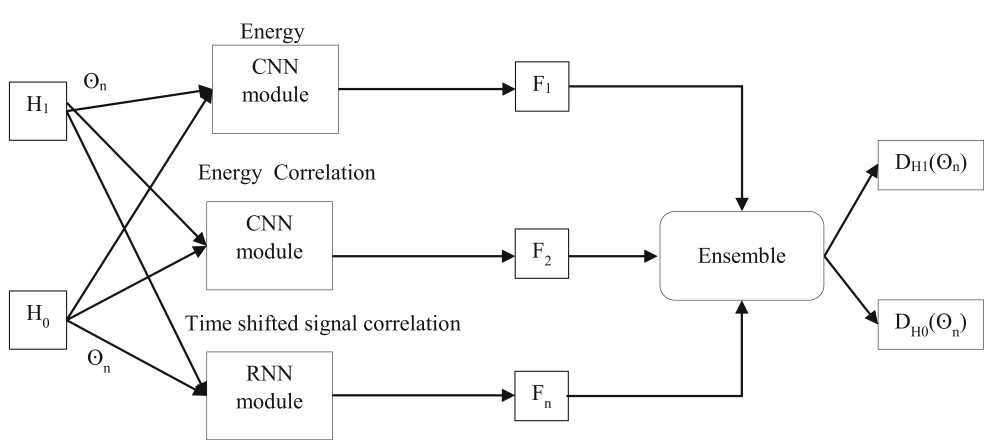
Fig. 2 Conventional framework for spectrum sensing

Therefore, we have adopted deep learning for PU presence from previous signal statistics. While training there is a requirement of labeled data, even for the deep learning (DL) approach. In this paper, we have taken $Y = \{(x_1,l_1), (x_2,l_2), (x_3,l_3), \ldots\ldots\ldots,(x_N,l_N)\}$ where Y is termed as a training set having training data of size 'N' with input data, x_N and l_N represents the labeled data. The PU presence is represented by Y. As it is observed that with the increased size of training input, the computational complexity increases. For PU sensing, sampling statistics may contain redundant data, because it may be from the same distribution source. Therefore, there is a requirement to pre-processing the input data before the start of the training process. The energy correlation and cyclostationary correlation are the two most important features that are applied in this paper.

In this paper, we have proposed an ensemble deep learning approach using CNN, as illustrated in Fig. 3. In this, two inputs, sample covariance matrix ($\odot_n$) are fed into two CNN layers and one RNN layer respectively, as covariance matrix is considered to be the complex mathematical problem that contains real and imaginary parts. CNN and RNN layers are illustrated in Figs. 4 and 5 respectively. Here RNN is used for time-shifted correlation feature extraction because it can work effectively in time series data. In the case of the H_0 hypothesis, the feature information such as energy is given in diagonal elements of the real part of the matrix whereas, in the case of the H_1 hypothesis, feature information is scattered. The difference between features of H_0 and H_1 is enough for the learning process of CNN. The training covariance matrices ($\odot_n$) of both hypotheses are fed into three layers of CNN. Then each layer works on three different feature vectors out of the input covariance matrix. In this architecture, three features are considered, energy, correlation, and time-shift signal correlation, individually and lastly their decisions are ensembled together to make a final decision of either presence of PU, $D_{H_1}(\odot_n)$ or absence of PU, $D_{H_0}(\odot_n)$, such that $D_{H_0}(\odot_n) + D_{H_1}(\odot_n) = 1$ where D stands for decision parameters of CNN.

The convolution component in our spectrum sensing structure consists of three sub-blocks. Each sub-block also consists of a convolution layer a leaky rectified linear unit (LReLU) layer, which is also linked together in tandem. The retrieved spatial features of input data are fed into a 2D convolution layer. Each filter is set to 3×3 in the convolution layer. The convolution layer depth for the basic i_{th} sub-block is set to C_i. To keep the result as same as that of the input set the stride to one and use the zero paddings. The LReLU layer activation layer complements non-linearity to the CNN. The convolution layer is linear that cannot classify non-linear data without the presence of LReLU. The fully connected layer classifies the function by obtaining the results of extraction of the function. At last, a fully connected (FC) layer is applied that performs classification process taking

Fig. 3 The network structure of ensemble CNN



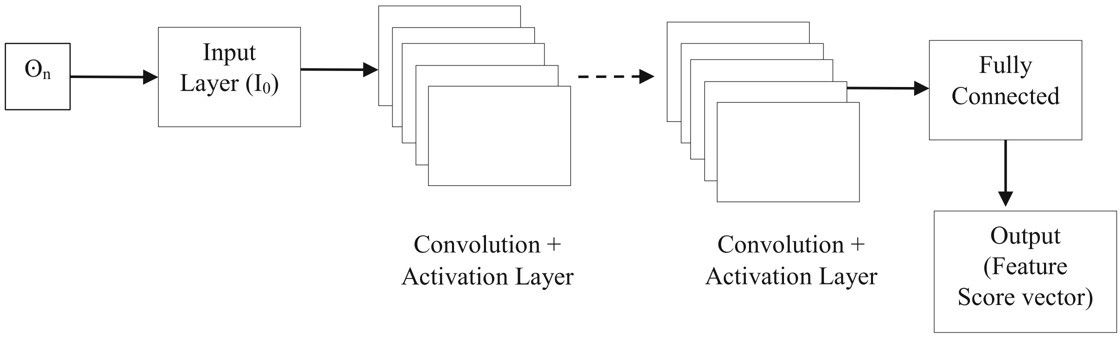


Fig. 4 CNN Architecture

input from the output of the previous convolution layer. The performance of the FC layers is then integrated into the ensemble classification system. By applying the ensembling approach final decision about the presence of PU or absence of PU is taken with the boosting function. Indexes 1 and 0 will present the presence and absence of PU respectively. The performance of CNN can be decreased there is no information about the presence of SU that makes the learning process difficult. Here, many CNN models with multiple SU permutations can be trained simultaneously that can achieve the highest accuracy. A permutation operation is performed for the correct order of the SU index that can be found in a data array to boost SS efficiency such that the sensing result of neighboring SU is located close to one another. The trained model can then be used to evaluate H_1 state or H_0 state based on different detecting outcomes. While preparation for the spectrum sensing process can lead to computational overhead, the conclusion of the final sensing result can, as shown later in the performance assessment, be carried out with relatively low overhead so that the operation of our proposed system in real-time is feasible.

4.3 Network training and complexity analysis

4.3.1 Network training

When dealing with the offline based training modules the unlabelled samples are accumulated and constructed to bring about the formation of training data set, (X,L) = {(x_1,l_1), (x_2,l_2), (x_3,l_3),…,(x_N,l_N)}.The *(X, l)* is the training sample in the equation and the value of the example persisting in it. While taking into account only a single example in this set, (x,l) then they in it is indicative of the input value provided to the neural network for the training purpose. The value y as an input can be a raw observation vector or can also be utilized in the form of the test statistic that has been derived from the observational vector. The X and L are indicative of the collections comprising the data associated with x and data associated with l respectively. The architectural design for the training has been done by utilizing the ensembling of CNN and RNN architecture to extract the features from the training set. The study concludes the ECRNN training requires to be dealing with the classification problems as the spectrum identification and sensing is a binary testing challenge. Therefore, the (x_N,l_N) being a single part of the set, the label for it can be encoded as one vector, Eqn (2):

$$l_N = \begin{Bmatrix} [1]^N, & H_1 \\ [0]^N, & H_0 \end{Bmatrix} \tag{2}$$

The Training process of ECRNN shall maximize the likelihood, $L(\odot)$, based on Eqn (3).

$$L(\odot) = P(L|X;\odot) = \prod_{k=1}^{k} \left(D(\odot)_{H_1}(x_N)\right)^{l_N} \left(D(\odot)_{H_o}(x_N)\right)^{1-l_N} \tag{3}$$

In terms of log-likelihood:

$$\begin{aligned} l(\odot) &= \log L(\odot) \\ &= \sum_{n=1}^{N} l_N \log D(\odot)(x_N) + (1{-}l_N)\log(1{-}D(\odot)(x_N)) \end{aligned} \tag{4}$$

This can be used for maximizing the cost function, C_f. The posterior probability enhancement P(L|X), can only be achieved by the optimal $\odot$ evaluation that forms the key objective for the proposed model training process.

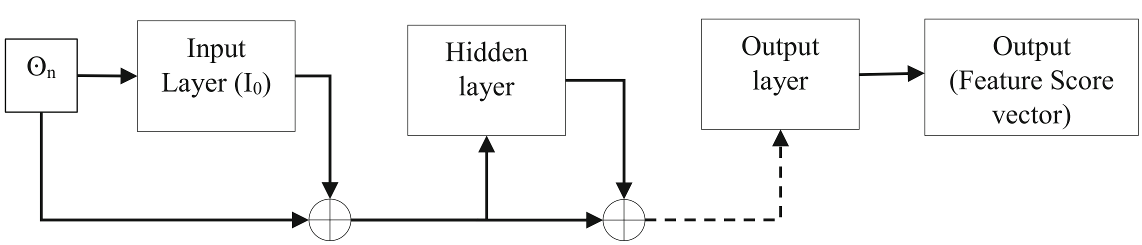


Fig. 5 RNN Architecture

$$C_f = \max(\mathrm{P(L|X)}, \odot) \tag{5}$$

The derivation of a well-trained ECRNN model is achieved by continuously updating the ECRNN network parameters via another backpropagation algorithm of calculation that is dependent on the cost function achieved the well-trained network is represented as Eqn (6):

$$D^*_{\odot}(x) = \begin{bmatrix} D(\odot)^*_{|H_1}(x) \\ D(\odot)^*_{|H_0}(x) \end{bmatrix} \tag{6}$$

The expression comprises of the well-trained CNN network having input as x which is indicated by $D^*_{\odot}(x)$. The expression $D(\odot)^*_{|H_1}(x)$ depicts the class score for H_1 or H_0. These can be used to derive the posterior probabilities associated with two hypotheses, Eqn (7):

$$\begin{aligned} H_1 &: P\ (H_1|x) = D(\odot)^*_{|H_1}(x) \\ H_0 &: P\ (H_0|x) = D(\odot)^*_{|H_0}(x) \end{aligned} \tag{7}$$

When the system if completely and efficiently trained with respective parameters, we can say that the training process is converged as well as "well trained". On referring to the Bayes theorem Eqn (8) [28]:

$$\begin{aligned} P(x|H_1) &= \frac{P(H_1|x).\ P(x)}{P(H_1)} = \frac{D(\odot)^*_{|H_1}(x).\ P(x)}{P(H_1)} \\ P(x|H_0) &= \frac{P(H_0|x).\ P(x)}{P(H_0)} = \frac{D(\odot)^*_{|H_0}(x).\ P(x)}{P(H_0)} \end{aligned} \tag{8}$$

Where, $P(x|\ H_1)$ = conditional probability, $P(H_i)$ = prior probability of H_i, and $P(x)$= marginal probability. $P(x|\ H_1)$ and $P(x|\ H_0)$ are calculated and the conclusion is drawn that the NP is indicative of the optimum statistic for the test which is the likelihood ratio (LR).

4.3.2 Neyman Pearson detection

To maximize the probability of detection (P_d) for a given PFA, we decide H_1 if

$$P_d = \frac{P(x|H_1)}{P(x|H_0)} > D_s \tag{9}$$

The derivation of the ECRNN has been made as Eqn (10), x utilizing the above equations.

$$\begin{aligned} L_{ECRNN}(x) &= \frac{D(\odot)^*_{|H_1}(x)}{D(\odot)^*_{|H_0}(x)}\ .\ \frac{P(H_0)}{P(H_1)} \\ &= \frac{D(\odot)^*_{|H_1}(x)}{D(\odot)^*_{|H_0}(x)}\ \geqslant D_s \end{aligned} \tag{10}$$

The D_s is the threshold value selected that is derived by the false alarm constraint and the $L_{ECRNN}(x)$ is the test statistic framework indicating the ECRNN. The ECRNN testing framework helps in acquiring posterior probabilities for two distinct hypotheses by training the data set of (X,L). However, it has been found that the training process generates posterior probabilities associated expressions that were not suitable for testing the samples that command the requirement of the conditional probability-based derivation of the ECRNN during the detection process. To achieve this $P(x|\ H_1)$ and $P(x|\ H_0)$ are being derived as the conditional probability that utilizes the Bayes' hypothesis for derivation. The process follows the attaining ECRNN that lays on the NP theorem. Further, the decision-making process shall inculcate comparison with a detection threshold (D_s). The threshold value can even be determined with a method referred to as the Monte Carlo process that aids in achieving the P_d required. The training process is performed using Algorithm as shown below:

Algorithm 1 Training algorithm

1: Procedure Training parameter (Max_epochs, Batch_size, X, y, α)
2: for i = 1:Max_epochs do
3: Input[Y={(X_1,l_1), (X_2,l_2), (X_3,l_3), ………,(X_N,l_N)}]
4: Feature Vector(F_v) ← extract(X)
5: Train(DNN) ← F_v
6: If likelihood ratio (L_r) > threshold (λ), {Maximize Probability of detection (P_d)}
7: End if
8: End for

4.3.3 Testing process

The test data that is to be utilized during the detection based on the test framework is represented as $\widetilde{X}$ for a single as well as multi SU system that aims at achieving this data as a set of unlabelled samples. The ECRNN is trained for $\widetilde{X}$ samples of the collected data and further the ECRNN steps are processed for the test samples, this is denoted by the Eqn (11):

$$L_{ECRNN}\left(\widetilde{X}\right) = \frac{D(\odot)^*_{|H_1}(x)}{D(\odot)^*_{|H_0}(x)} \geqslant D_s \tag{11}$$

The inherent comparison with the threshold value that has been preset previously, can bring about the decision-making process after achieving the test statistic. It has also been found that currently, an existing algorithm such as DL-based sensing of the spectrum has the capability of completely replacing the system with the neural network for end-to-end analysis and detection. The work in this process shall not comprise of the provision to define the threshold for attaining the P_d. The ECRNN based schemes for spectral identification hold within itself the framework for determining the current practical threshold value during the function other than other frameworks, whose objective of to

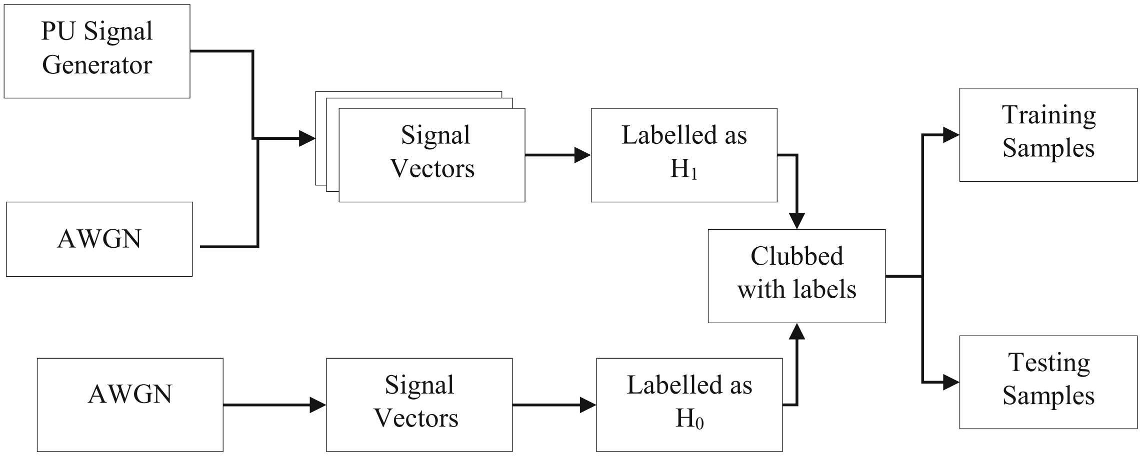


Fig. 6 Dataset Preparation for ECRNN

keep updating the threshold value to achieve the desired P_d. The complete algorithm of ECRNN (called Specturum Sensing algorithm using ECRNN) is given below :

Algorithm 2 Spectrum Sensing using ECRNN

```
Initialization
Set epoch =1
Start Training (X,L)
Feature Vector(Fv) ← extract(X)
Train(DNN) ← Fv
While epoch<max_epochs
do
Update Ds
epoch =epoch+1
End for
Set Ds
Input Test (X̃)
Feature Vector (F̃V) ← extract(X̃)
Compare with Ds
Identify PU activity
End
```

4.3.4 Complexity analysis

While training any network, the complexity of CNN for processing one data sample is evaluated to be as in Eqn (12) [29]:

$$O\left(\sum_{p=1}^{P} N_{k,p-1} S_{k,p}^{2} n_{c,p} O_{k,p}^{2}\right) \tag{12}$$

where P = Number of convolution layer with $N_{k,\,p-1}$ to be the number of input channels. With $N_{k,\,p}$ as number of convolutional kernels for p_{th} kernel with the spatial size of $S_{k,p}^2$ that generates $O_{k,p}^2$ of the output feature map. We have designed the CNN layer of ECRNN with two convolution layers and taken the input of real and imaginary data of size (S × S × 1). The CNN stride is set to 1 to reduce the computational complexity. While for the RNN network, the computational complexity is dependent on the number of neurons and internal parameters of the network. This can be illustrated in Eqn (13):

$$O(n_i) \tag{13}$$

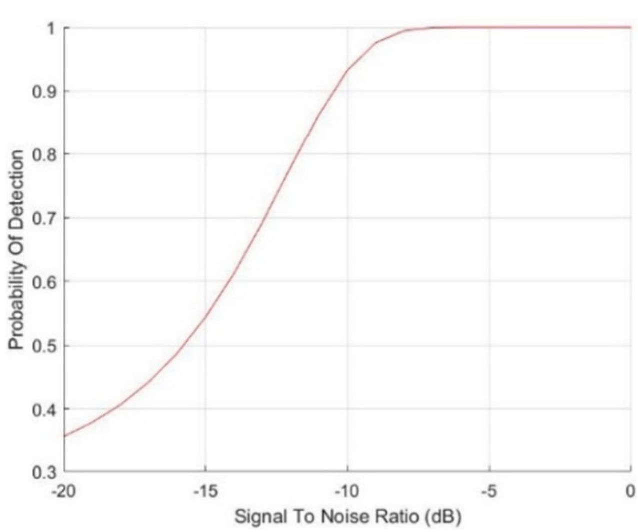


Fig. 7 P_d versus SNR

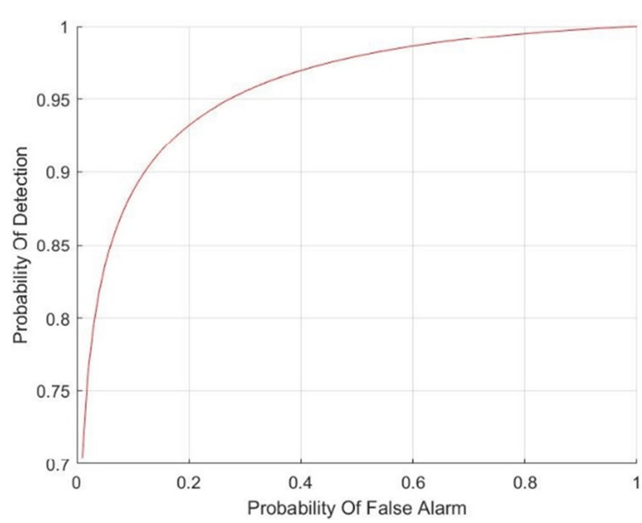


Fig. 8 P_f versus P_d

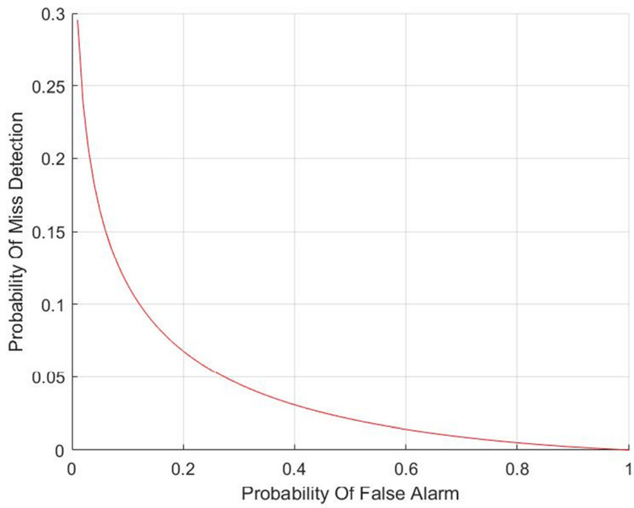


Fig. 9 P_f versus P_m

Where n_i is the number of neurons present in hidden layers. Therefore, the complexity of ECRNN for one data sample is represented as in Eqn (14):

$$O\left(\sum_{p=1}^{P} N_{k,p-1} S_{k,p}^{2} n_{c,p} O_{k,p}^{2} n_i\right) \tag{14}$$

4.3.5 Dataset preparation

In this subsection, the dataset required for training the proposed ECRNN model is prepared. The spectrum data is used for training and test validation purposes. The data is captured through a simulation setup (Fig. 6). The clean PU signal is generated from the generator and its spectrum power is measured as σ_x^2.

The Additive white Gaussian noise (AWGN), n, is added to achieve a required signal-to-noise ratio (SNR). This noise is added to PU signal for timestamp t.

$$X = [x_1, x_2, \ldots x_t]^T \tag{15}$$

For this study, approx. 5000 data samples are generated in the SNR range − 15 dB to +5 d B having equal number of PU signal and AWGN signals. The generated dataset is divided into 2 sets 70% training and 30% testing samples.

5 Results and discussions

In this section, the simulation setup of ECRNN is presented. In our implementations, we have utilized the MATLAB platform for training and testing scenarios. The training is performed with different data samples having two classes i.e. H_1 and H_0. The individual CNN or RNN are trained on different signal features and their results are ensembled together to generate the final result.

For training, the model simulation was performed with 10,000 data samples in which 7000 data samples are used for training and 3000 data samples are used for testing. The learning rate was set to be 0.0003 and 64 sample patches are used. The performance metrics are used to show the relationship between the probability of false alarm rate (P_f), probability of detection (P_d), and probability of misdetection (P_m). The variation of P_d concerning P_f is also observed. The training process is performed using Algorithm 1. While Algorithm 2 is used to test the data samples for the presence of PU.

5.1 Relation between P_f and D_s

Theorem 1 In spectrum sensing, theoretically, the probability of false alarm (P_f) is related to decision threshold (D_s) value as following in Eqn (16):

$$P_f = (1 - D_s)^{M-1} \tag{16}$$

Proof When there is the presence of noise in the channel, the Cumulative Distribution Function (CDF) of D_s is evaluated as Eqn (17):

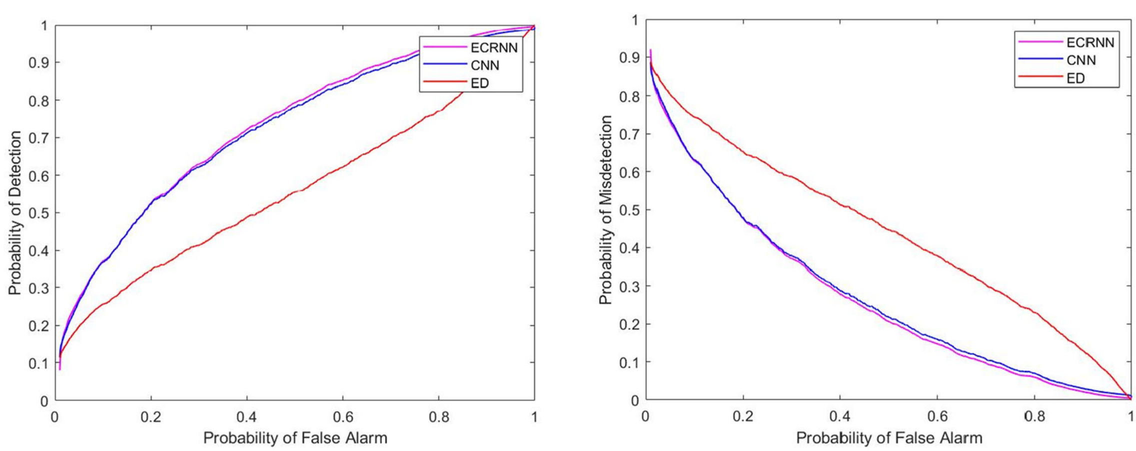


Fig. 10 ROC Curve for single SU

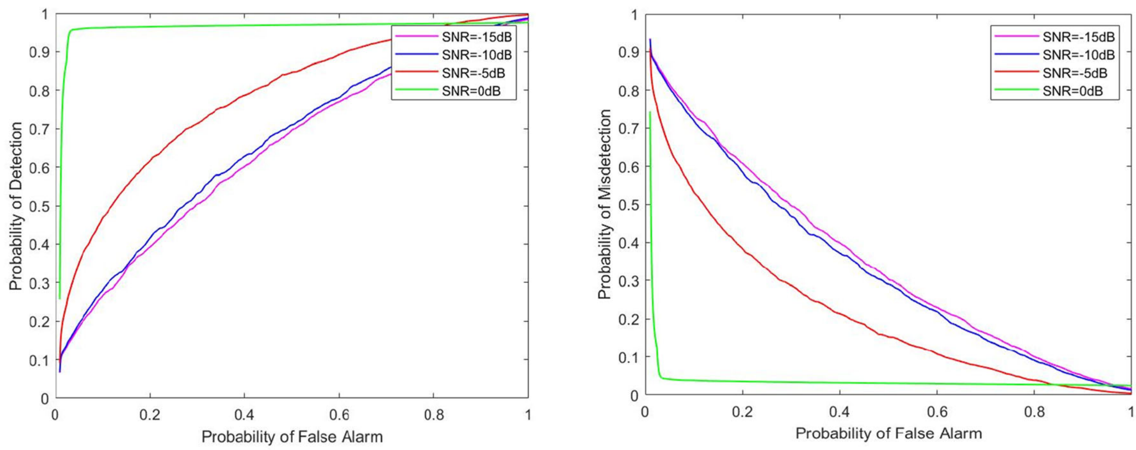


Fig. 11 ROC Curve varying SNR for single SU

$$F_d = 1-(1-d)^{M-1}, 0\leq d\leq 1 \tag{17}$$

Where P_f is represented as Eqn (18):

$$P_f = P_r[d\geq D_s|H_o] = 1-F_d \tag{18}$$

Where H_o represents the absence of PU.

By substituting the value of F_d into Eqn (19), it has been proved the relationship between a false alarm and decision threshold. So, the threshold can be computed as:

$$D_s = P_f^{\frac{1}{M-1}} \tag{19}$$

5.2 Relation P_m and P_d

Theorem 2 For t cyclostationary detection, the Probability of misdetection (P_m) is calculated as in Eqn (20):

$$P_m = 1-P_d \tag{20}$$

Proof: Probability of detection (P_d) is calculated as in Eqn (21):

$$P_d = P_r\Big[[d\geq D_s|H_1]\tag{21}$$

Where,

Q(.) = q-function.

d = signal-to-noise ratio (SNR) at the receiver and H_1 represents the presence of PU.

The probability of misdetection(P_m) is calculated as in Eqn (22):

$$P_m = 1-P_d \tag{22}$$

Figure 7 represents the graph of Probability of detection (P_d) concerning SNR whereas Fig. 8 represents the graph of P_f concerning the probability of detection P_d. The figure shows that with an increasing number of samples (NoS) the P_d increases. Similarly, Fig. 9 represents the probability of misdetection (P_m) with respect P_f. The figure concludes that with increasing samples the P_m decreases.

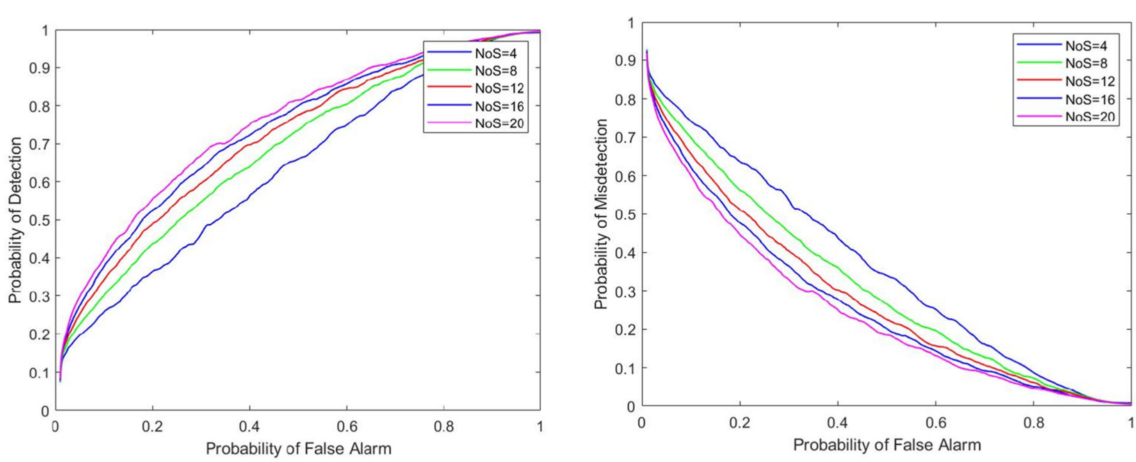


Fig. 12 ROC Curve varying NoS for single SU

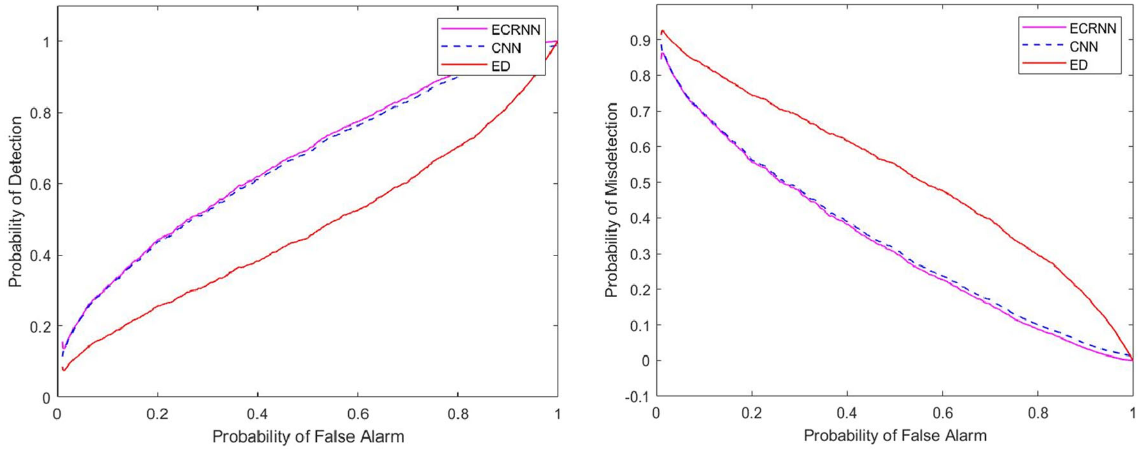


Fig. 13 ROC Curve for multiple SUs

5.3 Performance parameters

While simulating the proposed ECRNN model, the performance parameters used here are Receiver Operating Characteristics (ROC) for P_d against P_f for single and multiple PU. The ROC curve represents the area to show the relationship between P_d and P_f. The area increases with increased model performance. Three scenarios are created 1st is to observe at fixed SNR, the second with variable SNR, and the third is variable NoS. The performance of ECRNN is compared with CNN [19] and ED. Another parameter used to evaluate the performance of the proposed ECRNN are computational time and error rate. The time represents the total execution time for performing training as well as testing simulation. The error rate represents the mean square error (MSE) that occurred during training. This is evaluated by finding the mean of the squared difference between target and reconstructed value. MSE is calculated as in Eqn (23).

$$loss_{mse} = \frac{\sum_{i=1}^{N}\left((x_t - x_r)\right)^2}{N} \tag{23}$$

Where, x_t = target value, X_r = reconstructed value, N = Number of samples.

5.4 Result analysis

Figure 10 illustrates the comparative ROC curve for different spectrum sensing methods. The figure is plotted for SNR = -15 dB. For comparison, the NoS taken is 20. In the comparison of ECRNN with other techniques such as ED [7] and CNN [30] the training data and scenario are kept the same. This simulation was performed for single PU and single SU and trained accordingly. As each module contains different features and input to CNN is single column so, we have taken a 1-D CNN network. The model was created, trained, and tested on the MATLAB platform using the deep learning library. We can analyze from the graph plot that ECRNN gives a better result as compared to other sensing techniques even at SNR of −15 dB. Due to the ensembled architecture of ECRNN, it gives better performance because it combines the combined results from different features while other existing

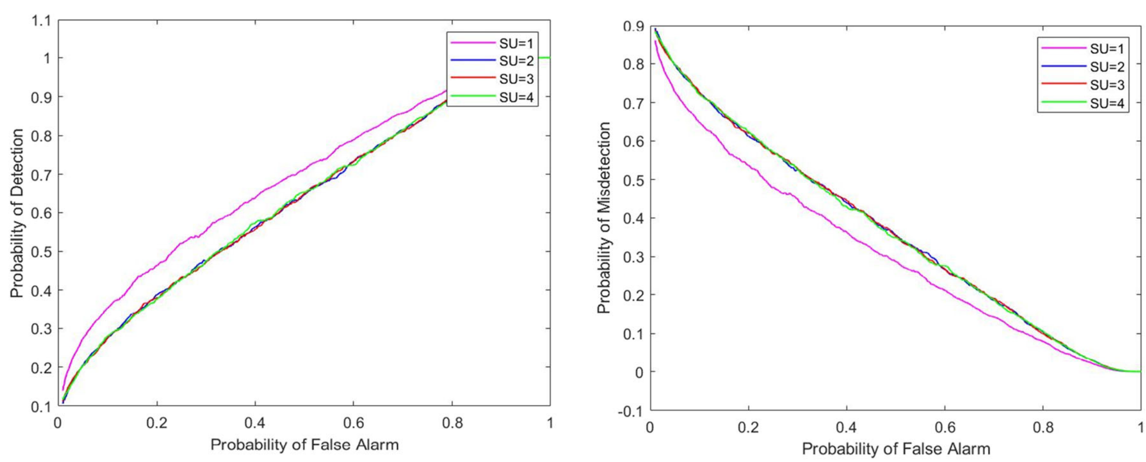


Fig. 14 ROC Curve varying SNR for multiple SUs

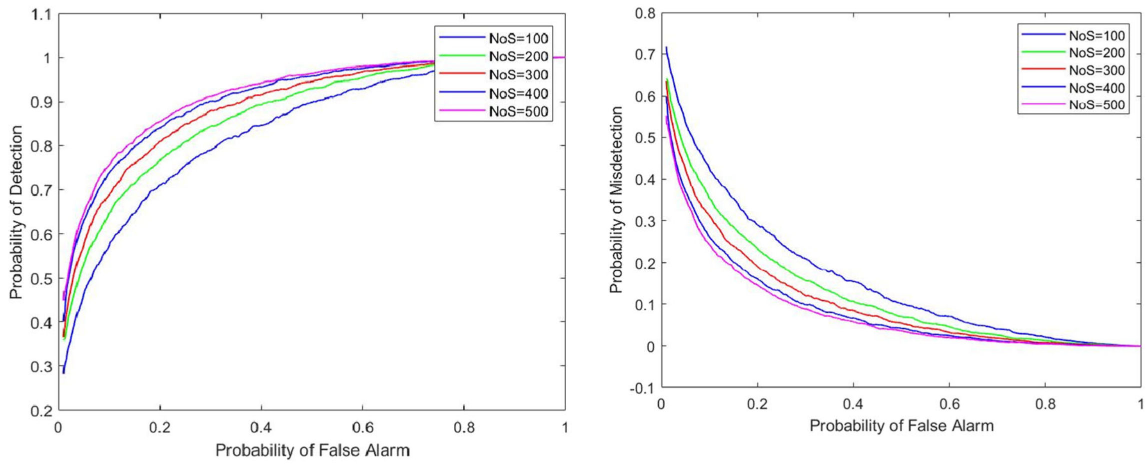


Fig. 15 ROC Curve varying NoS for multiple SUs

techniques give results on a single feature such as energy correlation. Figure 9 represents the ROC for the probability of detection (Pd) concerning false alarm (Pf) as well as ROC for the probability of misdetection (Pm) concerning P_f. Figure 11 represents the ROC curve for the P_d concerning P_f as well as the ROC curve of P_m concerning P_f. In this scenario, a comparison was performed with varying SNR values from 0 dB to -15db. The graph illustrates that with increasing SNR the P_d decreases and P_m increases. Figure 12 represents the ROC curve for the P_d concerning P_f as well as the ROC curve of P_m concerning P_f. In this scenario, the comparison was performed with varying data samples with -15 dB SNR. The graph illustrates that with an increasing sample the P_d increases and P_m decreases. Figure 13 illustrates the comparative receiver operating characteristics (ROC) curve for different spectrum sensing methods. The figure is plotted for SNR = -15 dB for multiple SUs scenarios and the NoS taken is 20. The graph represents the ROC for the probability of detection (P_d) concerning false alarm (P_f) as well as ROC for the probability of misdetection (P_m) concerning P_f. Figure 14 represents the ROC curve for the P_d concerning P_f as well as the ROC curve of P_m concerning P_f. In this scenario, a comparison was performed with a varying number of SU, and the values of SNR are -15db. The graph illustrates that with an increasing number of SU the P_d decreases and P_m increases. Figure 15 represents the ROC curve for the P_d concerning P_f as well as the ROC curve of P_m concerning P_f. In this scenario, a comparison was performed with varying data samples with -15 dB SNR under multiple SUs scenario. The graph illustrates that with an increasing sample the P_d increases and P_m decreases.

Similarly, Table 1 represents the computational time evaluated in seconds for training and testing samples using the ECRNN algorithm. The algorithm is implemented in MATLAB and executed on a PC with an Intel Core i5 3.71GHz CPU and 2 GB Nvidia graphics with 8GB RAM. In summary of existing work, the proposed method achieves the optimal solution concerning detection. Even though the ECRNN had achieved optimal solution but still there is needed to reduce the computational complexities. If this model is parallelly executed on GPU, then it would be very much helpful to reduce computational complexity. Similarly, in Table 2 error rate is evaluated for the detection process and it can be inferred that ECRNN achieved less training error as compared to the CNN model.

6 Conclusion

This paper is dedicated to spectrum sensing problems using the application of CNN models. For this ensemble, CNN and RNN technique is developed and termed as ECRNN and presented over single and multiple user scenarios. In the first

Table 1 Computational Time Analysis (in seconds)

Algorithms	ECRNN
For Single SU	85.97 s
For Multiple SUs	166.14 s
For Varying SNR	314.26 s
For Varying Samples	297.011 s

Table 2 Error Evaluation

Algorithms	ECRNN	CNN [15]
For Single SU	0.0004320	0.0094
For Multiple SUs	0.000220	0.0125
For Varying SNR	0.000275	0.0115
For Varying Samples	0.0004378	0.0098

scenario, a single SU is considered under a varying NoS and varying SNR. Whereas in the second scenario, multiple SUs was considered with varying number of samples, SU and SNR. For training energy, correlation and time-shifted correlation was considered to be as a feature vector and individual DL model was trained and their results are ensembled together to give the final result. The detection of test data samples was performed using an ensemble approach which results in the optimal solution. The simulation results were performed and performance was evaluated by ROC curve analysis as well as time complexity and error rate. The result analysis showed better performance concerning the CNN model as well as the traditional ED model.

In this paper, we provide a theoretical analysis of the advantages of ECRNN over other methods. Then simulation experiments are performed for the probability of detection concerning variable SNR and showed up its robustness as well as scalability. The results have shown that the proposed CM-CNN method could achieve almost the same performance as that of the optimal E-C detector whether the PU signals are independent or correlated.

The limitation of this work is that with increasing SU there is a decrease in detection performance which needed to be optimized. These limitations can be improved in the future by deciding the optimal number of SU that can be handled. In the future, this work will also be enhanced with a path fading channel scenario along with noise.

References

1. Lundén J, Koivunen V, Poor HV (2015) Spectrum exploration and exploitation for cognitive radio: recent advances. IEEE Signal Process Mag 32:123–140
2. Wellens M, Mähönen P (2009) Lessons learned from an extensive spectrum occupancy measurement campaign and a stochastic duty cycle model. In: 2009 5th international conference on Testbeds and research infrastructures for the development of networks and communities and workshops, TridentCom 2009. https://doi.org/10.1109/TRIDENTCOM.2009.4976263
3. Mitola J, Maguire GQ (1999) Cognitive radio: making software radios more personal. IEEE Pers Commun 6:13–18. https://doi.org/10.1109/98.788210
4. Haykin S (2005) Cognitive radio: brain-empowered wireless communications. IEEE J Sel Areas Commun 23:201–220. https://doi.org/10.1109/JSAC.2004.839380
5. López-Benítez M, Casadevall F (2011) Modeling and simulation of time-correlation properties of spectrum use in cognitive radio. In: proceedings of the 2011 6th international ICST conference on cognitive radio oriented wireless networks and communications, CROWNCOM 2011. Pp 326–330. https://doi.org/10.4108/icst.crowncom.2011.246158
6. Haykin S, Thomson DJ, Reed JH (2009) Spectrum sensing for cognitive radio. Proc IEEE 97:849–877. https://doi.org/10.1109/JPROC.2009.2015711
7. Urkowitz H (1967) Energy detection of unknown deterministic signals. Proc IEEE 55:523–531. https://doi.org/10.1109/PROC.1967.5573
8. Yücek T, Arslan H (2009) A survey of spectrum sensing algorithms for cognitive radio applications. IEEE Commun Surv Tutorials 11:116–130. https://doi.org/10.1109/SURV.2009.090109
9. Wang P, Fang J, Han N, Li H (2010) Multiantenna-assisted spectrum sensing for cognitive radio. IEEE Trans Veh Technol 59:1791–1800. https://doi.org/10.1109/TVT.2009.2037912
10. Chen X, Zhang H, MacKenzie AB, Matinmikko M (2014) Predicting spectrum occupancies using a non-stationary hidden markov model. IEEE Wirel Commun Lett 3:333–336. https://doi.org/10.1109/LWC.2014.2315040
11. Letaief KB, Chen W, Shi Y, Zhang J, Zhang YJA (2019) The roadmap to 6G: AI empowered wireless networks. IEEE Commun Mag 57:84–90. https://doi.org/10.1109/MCOM.2019.1900271
12. Zeng Y, Choo LK, Liang YC (2008) Maximum eigenvalue detection: theory and application. In: IEEE International Conference on Communications. pp. 4160–4164. https://doi.org/10.1109/ICC.2008.781
13. Zhang R, Lim TJ, Liang YC, Zeng Y (2010) Multi-antenna based spectrum sensing for cognitive radios: a GLRT approach. IEEE Trans Commun 58:84–88. https://doi.org/10.1109/TCOMM.2010.01.080158
14. Saleem Y, Rehmani MH (2014) Primary radio user activity models for cognitive radio networks: a survey. J Netw Comput Appl 43:1–16
15. Nguyen T, Mark BL, Ephraim Y (2013) Spectrum sensing using a hidden bivariate markov model. IEEE Trans Wirel Commun 12:4582–4591. https://doi.org/10.1109/TWC.2013.072513.121864
16. Sarikhani R, Keynia F (2020) Cooperative Spectrum sensing meets machine learning: deep reinforcement learning approach. IEEE Commun Lett 24:1459–1462. https://doi.org/10.1109/LCOMM.2020.2984430
17. Lees WM, Wunderlich A, Jeavons PJ, Hale PD, Souryal MR (2019) Deep learning classification of 3.5-GHz band spectrograms with applications to spectrum sensing. IEEE Trans Cogn Commun Netw 5:224–236. https://doi.org/10.1109/TCCN.2019.2899871
18. Liu C, Wang J, Liu X, Liang YC (2019) Deep CM-CNN for Spectrum sensing in cognitive radio. IEEE J Sel Areas Commun 37:2306–2321. https://doi.org/10.1109/JSAC.2019.2933892
19. Xie J, Liu C, Liang YC, Fang J (2019) Activity pattern aware Spectrum sensing: a CNN-based deep learning approach. IEEE Commun Lett 23:1025–1028. https://doi.org/10.1109/LCOMM.2019.2910176
20. Xie J, Fang J, Liu C, Li X (2020) Deep learning-based Spectrum sensing in cognitive radio: a CNN-LSTM approach. IEEE Commun Lett 24:2196–2200. https://doi.org/10.1109/LCOMM.2020.3002073
21. Soni B, Patel DK, Lopez-Benitez M (2020) Long short-term memory based Spectrum sensing scheme for cognitive radio using primary activity statistics. IEEE Access 8:97437–97451. https://doi.org/10.1109/ACCESS.2020.2995633
22. Paisana F, Selim A, Kist M, Alvarez P, Tallon J, Bluemm C, Puschmann A, Dasilva L (2017) Context-aware cognitive radio using deep learning. In: 2017 IEEE international symposium on dynamic Spectrum access networks, DySPAN 2017. https://doi.org/10.1109/DySPAN.2017.7920784
23. Cheng Q, Shi Z, Nguyen DN, Dutkiewicz E (2019) Deep learning network based Spectrum sensing methods for OFDM systems. ArXiv. http://arxiv.org/abs/1807.09414
24. Lee W, Kim M, Cho DH (2019) Deep cooperative sensing: cooperative Spectrum sensing based on convolutional neural networks. IEEE Trans Veh Technol 68:3005–3009. https://doi.org/10.1109/TVT.2019.2891291

25. Zhang Y, Cai P, Pan C, Zhang S (2019) Multi-agent deep reinforcement learning-based cooperative Spectrum sensing with upper confidence bound exploration. IEEE Access 7:118898–118906. https://doi.org/10.1109/ACCESS.2019.2937108
26. Kim S, Lee J, Wang H, Hong D (2009) Sensing performance of energy detector with correlated multiple antennas. IEEE Signal Process Lett 16:671–674. https://doi.org/10.1109/LSP.2009.2021381
27. Zeng Y, Liang YC (2009) Spectrum-sensing algorithms for cognitive radio based on statistical covariances. IEEE Trans Veh Technol 58:1804–1815. https://doi.org/10.1109/TVT.2008.2005267
28. Oppenheim AV, Editor ANDREWS S, Brigham H, Adaptive Filters CROCHIERE G, Dudgeon R, HAMMING Digital Filters M, Haykin E, Haykin E, Array Signal Processing JAYANT E, Johnson ND, Dudgeon Kay Kay NA, Marple Mcclellan E, Mendel Oppenheim R, Oppenheim E, Oppenheim E, Young Oppenheim W, Rabiner G, Stearns T, Stearns D, Tribolet Vaidyanathan Widrow H, Kay SM (n.d.) PRENTICE H A L L SIGNAL PROCESSING SERIES Digital Signal Processing OPPENHEIM AND SCHAFER Discrete-Time Signal Processing Fundamentals of Statistical Signal Processing: Est imat ion Theory. Retrieved May 5, 2021, from http://wmn.prenhrll.com
29. He K, Sun J (2015) Convolutional neural networks at constrained time cost. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, pp. 5353–5360. https://doi.org/10.1109/CVPR.2015.7299173
30. Liu C, Liu X, Liang YC (2019) Deep CNN for Spectrum sensing in cognitive radio. In: IEEE International Conference on Communications. Institute of Electrical and Electronics Engineers Inc

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Pradeep Bedi received the B.Tech. degree in computer science and engineering from Uttar Pradesh Technical University (UPTU), Lucknow, India, in 2005 and M.Tech. in computer science and engineering from Guru Gobind Singh Indraprastha University (GGSIPU), Delhi, India, in 2009. He is GATE, UGC-NET qualified and currently pursuing Ph.D. He started his academic career from Mahatma Gandhi Mission's College of Engineering and Technology in 2005 and served various reputed colleges and universities in India and abroad. He has authored or co-authored over 15 technical papers published in national and international journals and conferences and also published 8 patents in India and abroad. He is a member of reputed professional bodies such as CSI, ACM etc. His research interests include applications of artificial intelligence, machine learning, deep learning and IoT in healthcare, automation etc.

Dr S B Goyal had completed PhD in the Computer Science & Engineering in 2012 from India and served many institutions in many different academic and administrative positions. He is holding 19+ years experience at national and international level. He has peerless inquisitiveness and enthusiasm to get abreast with the latest development in the IT field. He has good command over Industry Revolution (IR) 4.0 technologies like Big-Data, Data Science, Artificial Intelligence & Blockchain, computer networking, deep learning etc. He is the first one to introduce IR 4.0 including Blockchain technology in the academic curriculum in Malaysian Universities. He had participated as a speaker in the Bloconomic 2019 event on Blockchain. He had participated in many panel discussions on IR 4.0 technologies at academia as well as industry platforms. He is holding 19+ years' experience in academia at National & International level. He is serving as a reviewer or guest editors in many Journals published by Inderscience, IGI Global, Springer. He is contributing as a Co-Editor in many Scopus books. He is an IEEE Senior member since 2012. He had contributed in many Scopus/ SCI Journal/ conferences. Currently, Dr. Goyal is associated as a Director, Faculty of Information Technology, City University, MALAYSIA.

Dr. Jugnesh Kumar is working as a Director in St. Andrews Institute of Technology and Management, Gurugram, Haryana affiliated to the Maharshi Dayanand University, Rohtak, Haryana. He is an IT specialist in the area of machine learning, big-data. Dr. Kumar is received academic cum professional degrees MCA, M.Tech. and PhD(CSE). He is holding 20+ year administrative experience in addition to academic expertise in the area of Database System, Mobile Computing, Software Engineering, Computer Architecture, Java programming, Computer Networking, Deep Learning etc. He had organized many conferences/ events at India level. He had published many research papers/ book chapters in the Scopus Journals/ International Conferences/ Edited books. He had got the grant of International Patent for eight years from Australia in data analytics.

Prof Vijayakumar Varadarajan is currently an Adjunct Professor in School of Computer Science and Engineering, University of New South Wales, Sydney, Australia. He is also a Visiting Postdoc Scientist in Centro de Tecnologia, Federal University of Piauí, Brazil. He was a Professor and Associate Dean for School of Computing Science and Engineering at VIT University, Chennai, India. He has more than 18 years of experience including industrial and institutional. He also served as a Team Lead in industries like Satyam, Mahindra Satyam and Tech Mahindra for several years. He has completed Diploma with First Class Honors. He has completed BE CSE and MBA HRD with First Class. He has also completed ME CSE with First Rank Award. He has completed his PhD from Anna University in 2012. He has published many articles in national and international level journals/conferences/books. He is a reviewer in IEEE Transactions, Inderscience and Springer Journals. He has initiated a number of international research collaborations with universities in Europe, Australia, Africa, Malaysia, Singapore and North & South America. He had also initiated joint research collaboration between VIT University and various industries. He is also the Lead Guest Editor for few journals in Inderscience, Springer, Elsevier, IOS, UM and IGI Global. He also organized several international conferences and special sessions in USA, Vietnam, Africa, Malaysia and India including ARCI, IEEE, ACSAT, ISRC, ISBCC, ICBCC etc. His research interests include computational areas covering grid computing, cloud computing, computer networks, cyber security and big data. He received his university-level Best Faculty Award for 2015–2016. He is also a member of several national and international professional bodies including IFSA, EAI, BIS, ISTE, IAENG, CSTA, IEA etc.