International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 41, No. 6, December, 2024, pp. 2895-2908
Journal homepage: http://iieta.org/journals/ts

2ERNN: Elman Residual Recurrent Neural Network for Spectrum Sensing in the Cognitive Radio Network

K. Danesh[1*], R. Dharani[2]

[1] Department of ECE, SRM Institute of Science and Technology, Rampuram Campus, Chennai 600089, India
[2] Department of CSE, SIMATS Engineering, Chennai 602105, India

Corresponding Author Email: danesh.kn1@gmail.com

https://doi.org/10.18280/ts.410610

Received: 14 September 2023
Revised: 13 March 2024
Accepted: 12 July 2024
Available online: 31 December 2024

Keywords:
primary user, secondary users, spectrum sensing, Cyclostationary Feature Detection, Quadrature Phase Shift Keying

ABSTRACT

Spectrum sensing (SS) is a fundamental aspect of cognitive radio networks (CRNs), enabling efficient utilization of the radio frequency spectrum by dynamically identifying and exploiting available spectrum opportunities. Effective identification of main user (PU) signals is essential for maximizing spectrum use by secondary users (SU) in CRNs. However, existing SS techniques suffer from high false alarm rates and missed detections, which degrade the efficient utilization of the spectrum. Therefore, to overcome high false alarms and missed detections, this work proposes a sensing model based on deep learning (DL) classification. In this work, initially, the Quadrature Phase Shift Keying (QPSK) signal is randomly generated, and this modulated signal is subjected to Additive White Gaussian Noise (AWGN). Next, this signal is fed into a Cyclostationary Detector (CD), within which numerous functions are evaluated as part of Cyclostationary Feature Detection (CFD). These functions include Fast Fourier Transform (FFT), Auto-correlation, and Sliding window analysis to determine the characteristics of the signal. Subsequently, the proposed classifier, Elman Residual Recurrent Neural Network (2ERNN), is trained using the features to detect the presence of PU signals in low SNR environments. The weights of the 2ERNN are updated using the optimization technique known as Battle Royale Optimization (BRO) to enhance detection efficiency. Finally, the performance of the proposed system is evaluated by comparing it to other approach using classification and detection performance criteria. The proposed model is simulated on MATLAB and achieved better accuracy of about 98.2% and less False positive rate (FPR) value of 0.01 respectively. Further, the classification based on the cyclostationary features provides better information about signal characteristics even at less SNR value. Overall, the proposed sensing model offers a comprehensive solution for real-time spectrum sensing applications, combining efficiency, robustness, adaptability, optimization, high accuracy, and information-rich features to ensure reliable performance in dynamic and challenging environments.

1. INTRODUCTION

Increasing demand for radio spectrum in wireless communication is driven by recent breakthroughs in wireless communication technology. Thus in recent years, the CRN network gains more attention due to its capability of alleviating under-utilization of assigned radio spectrum [1]. The CRN networks permit an unlicensed user (secondary user) to gain the unused licensed spectrums of licensed users (primary user) aimed at satisfying SU data transmission [2]. Therefore, SU must be cognizant of the existence of PU traffic on the channel in order to prevent any disruptive interference.

The prime reason for spectrum scarcity in communication is the static allocation of spectrum [3]. In a static spectrum allocation scheme, a fixed spectrum is allocated to licensed users for a prolonged period. Hence, static allocation of spectrum is not effective for permitting access to all devices in the network [4]. Moreover, only a few portions of the spectrum are used and the remaining portions are unused or rarely used in static allocation. The inability of spectrum sharing among users will cause unnecessary service interruption. Thus to overcome the spectrum shortage, dynamic spectrum allocation is suggested. SS is a periodic scrutinizing manner of spectrum utilization to detect the interference of PU [5]. In CRN, SS is very needy for noticing PU activity and detecting spectrum utilization [6].

SS process enhances the CRN to sense, learn and examine its environment. The spectrum information facilitates SU communication through vacant channels without interrupting the PU data transmission [7]. If a specific spectrum is underutilized by the PU at a specific time period, then the SU can acquire that spectrum. Moreover, the cooperation among several SU facilitates the detection of the licensed band's status [8]. The accuracy of SS is improved by optimized parameters. SS enhances the detection of spectrum holes in CRN [9]. Apart from numerous advantages, the SS consumes more energy and the detection accuracy of SS is degraded by wireless channel fading, noise interference, barriers, etc. The

SS method is categorized as cooperative detection and non-cooperative detection [10]. Moreover, the cooperative detection approach can be implemented either centrally or distributedly, while the non-cooperative detection is classified as matching filter detection, energy detection (ED), and CFD [11]. In transmitter detection methods, the signals are detected via local observations. By the transmitter detection method, the weak signals from the primary unit are also detected. In centralized SS, a central device or server gathers the data from cognitive devices and then controls it directly or conveys it to cognitive radio. In distributed SS, the cluster node decides the part of the spectrum that they want to use [12].

The ED scheme finds out the signals based on the transmitted energy thus it needs a range of threshold for detecting whether the signal is present by comparing the threshold with the detected spectrum [13]. If the detected signal is lesser than the threshold, then it is considered as signal not present. The MFD is an optimum method for detecting the presence of PU by matching its impulse response with the received signal [14]. The CFD method has the ability to differentiate the PU signals from noise and interference. Because of the noise reduction ability, the cyclostationary can withstand in low SNR region [15]. The process of cyclostationary is random in nature and has periodical statistics [16]. The circumstances of cyclostationary can occur in modulation and coding phases but also can be developed to aid the channel estimation process [17]. The CFD provides the benefits of being utilized in blind context [18, 19]. Because of the information based on PU's is not available the major aim is to introduce the effective approach for the extraction of cyclostationary [20].

1.1 Motivation

One of the challenges in CRN is SS and the major aim is to find the spectrum availability and the presence of the PUs. Spectrum holes can be determined by detecting the presence of PUs. In the literature works, the machine learning (ML) models are used in the CRN, which used new concepts to research but these algorithms are more prone to overfitting issues. In wireless communication deep learning (DL) models are used in multiple input-multiple output (MIMO), SS, resource allocation and signal modulation recognition. Hence in this work, SS based on the DL model is introduced and the aim of this research is to enhance spectrum utilization with less interference. This work focuses only on the SS based on cyclostationary features and DL classifier Elman Residual Recurrent Neural Network (2ERNN) and the weight is updated using Battle Royale optimization (BRO). This avoids the local optima and improves the convergence speed thereby achieving better performance. Proposing an enhanced spectrum sensing network (SS) model in and data generation for cognitive radio network (CRN). The major contributions of the developed work are:

·***Enhanced Spectrum sensing network:*** The network is developed based on the combination of the Resnet 50 and ERNN. The Resnet 50 has the capability of analysing the information in radio frequency spectrum collected from the spectrum sensing devices and differentiates the collected signals as PU, SU, noise and interference. The variations in the spectrum and the spatial characteristics are also identified by this network. Similarly the ERNN captures the temporal dependencies and the pattern present in the radio spectrum that provides the information such as signal strength, frequency occupancy and interference pattern. The hybridized usage of these networks helps in achieving high robustness, flexibility and improved detection performance.

·***Battle Royale optimization (BRO):*** The weights initiated by the spectrum sensing network is optimized using this optimization, where the weight space are efficiently explored by this optimization and the global optimal solution is determined. The parallelism in this optimization helps in enhanced robustness and the continuous adaptability helps in dynamic spectrum sensing.

·To compare the performance of the created technique to current works, evaluate metrics such as detection probability, false detection probability, and missed detection probability.

This article is structured as follows: Section 2 discusses recent literature; Section 3 explains the developed scheme; Section 4 presents the implementation results; Section 5 provides a thorough discussion; and Section 6 concludes with the overall conclusion.

2. RELATED WORK

Spectrum sensing is a fundamental aspect of CRN systems, allowing them to autonomously detect available spectrum bands. Traditional methods often rely on statistical metrics such as the eigenvalue ratio or frequency domain entropy to discriminate between signal and noise. However, these methods can be limited by noise power uncertainty and may struggle to adapt to new signal types. To address these challenges, recent research has explored deep learning-based approaches for SS, considering it as a classification problem. By normalizing received signal power and leveraging deep neural networks, these methods aim to improve robustness and adaptability. Training the models with diverse signal types and noise data enables them to generalize well to untrained signals, while transfer learning strategies further enhance performance for real-world scenarios. Some of the literature works based on SS in CRN is presented in this section.

2.1 Ensemble model

Ahmad [21] developed an Ensemble Classifier Detector (ECD) for SS in CRN. Initially, a dataset was generated, which was required to train the ML classifier. The Signal-to-Noise Ratio (SNR) was varied from -5dB to -15dB, and four modulations were utilized. Subsequently, cyclostationary features were extracted using the FFT accumulation method (FAM). The ensemble classifier was then trained, utilizing the AdaBoost and decision trees algorithm.

2.2 Deep learning model

Solanki et al. [22] introduced the DL model "DLSenseNet," which utilized the received structural information of signals for SS. The primary objective was to select SS to precisely identify under-utilized spectrum and exploit it to enhance spectrum utilization. Experimentation was conducted on the Radio ML2016.10b database, with the SNR value varying from -20dB to 18dB. The database was partitioned into training, validation, and testing sets, and outcomes were analyzed. Nasser et al. [23] employed an Artificial Neural Network (ANN) to evaluate Hybrid SS (HSS). A dataset based on test statistics was developed, emphasizing identification of whether the licensed user was active or absent. The study highlighted the crucial role of SS in CRN and utilized cutting-edge approaches to obtain the most effective neural networks. The efficiency of the developed ANN-HSS outperformed traditional ANN methods and demonstrated its capacity for detecting Primary User (PU) signals. Xie et al. [24] utilized a DL model, for SS. Initially, the CNN model was employed to extract energy and correlation features generated from sensing data. Subsequently, the correlation and energy feature series for multiple sensing periods were input into LSTM, enabling understanding of the PU activity pattern and promoting detection probability.

Soni et al. [25] proposed SS based on LSTM. The CR model utilized activity patterns of PU, such as on and off time duration and duty cycle, to enhance performance. Verification of the proposed SS was conducted on spectrum data obtained from various radio methodologies via experimental setups, demonstrating better detection and classification performance. Peng et al. [26] proposed a robust SS framework using a DL model. Signals received by SUs were, input into CNN. Transfer learning (TL) was incorporated to provide robustness. TL was reliable without labeled target data, and fine-tuning was robust for transfer into various domains, depending on whether QPSK or Gaussian modulation was the source or target. Liu et al. [27] utilized a DL model, Deep Neural Network (DNN), for SS. DNN was employed for offline design, with thresholding used for online decisions. Sample covariance matrix was input into CNN, and a CM-CNN detection technique was designed. Extensive experimentation demonstrated superior results. Xie et al. [28] introduced SS for CRN based on CNN. Features of PU signal presence and noise were extracted, including energy and cyclostationary features. Extracted features were pre-processed and used as training input. Test data was then employed to determine PU presence, achieving a detection probability of approximately 0.5 at -20dB.

Ibrahim [17] developed a novel CNN-LSTM detector for spectrum sensing by employing CNNs that extracted energy-correlation features from covariance matrices generated by sensing data. Shah and Koo [29] combined the Short-Time Fourier Transform (STFT) with CNN to introduce a novel STFT-CNN method for spectrum sensing. This method leverages the time-frequency domain information inherent in signal samples, exploiting the rich spectral and temporal characteristics present in the data. The method effectively learned and extracted discriminative features from the STFT representations, that provided versatility in handling various primary users' signals without the need for prior information, making it adaptable to dynamic and diverse signal environments.

2.3 Clustering model

Reliable SS employing K-nearest neighbor (KNN) Sensing reports were classified into proper sensing classes, and the present sensing report was then classified to determine PU presence or absence. Local decisions were integrated at a fusion center using a novel decision combination method. Experimental outcomes demonstrated superior detection performance and spectral exploitation ability compared to traditional OR rules. Giri and Majumder [30] introduced an unsupervised deep SS (UDSS) for achieving better detection performance. UDSS did not require prior knowledge such as statistical Cyclostationary Modulation (CM) and noise power. UDSS required only a small quantity of samples obtained in the absence of PU signals, with experimental outcomes proving the efficiency of the introduced UDSS model. Test vectors obtained from computed eigenvalues were classified into available and unavailable channel classes by conducting clustering in 2D space. KFCM model created two clusters on test vectors: one for checking vacant channel status and the other for checking active status. a novel threshold expression method based on an online learning algorithm to increase SS accuracy in CRN and reduce overall error probability. The online learning algorithm selected the optimal threshold value necessary for determining PU presence or absence. Energy Detection (ED) and Matched Filter Detection (MFD) were briefly discussed, and performance was verified against fading and non-fading channels for low SNR, demonstrating improvements in SS performance.

Table 1. Merits and challenges of the related literature works

Authors and Citation	Methods	Merits	Challenges
Ahmad [21]	ECD	Achieved better probability of detection	Has high computational complexity
Solanki et al. [22]	DLSenseNet	Provided better detection than other sensing approaches	Takes more time for training due to the complex structure
Nasser et al. [23]	ANN	Achieved almost zero FAR	Requires more processing time
Xie et al. [24]	CNN- LSTM	Achieved better results in energy and correlation features	Complicated structure
Soni et al. [25]	LSTM	Achieved better classification accuracy	Takes more time for execution
Shah and Koo [29]	KNN	Achieved better detection performance	High FAR value
Peng et al. [26]	TL	This method was more robust	Pre-trained models may converge
Liu et al. [27]	DNN	This model was scalable, reliable and achieved better results in low SNR	Has overfitting issues
Ibrahim [17]	OL algorithm	Achieved better ED and MFD performance in low SNR	High FAR in some cases
Xie et al. [28]	UDSS	This model required only less amount of samples obtained in the absence of PU	Has high computational complexity
Giri and Majumder [30]	KFCM	Achieved better ROC values	Has high dimensional features
Han et al. [31]	CNN	Achieved better detection probability of about 0.5 in -20 dB	Need large trained data

Table 1 presents merits and related researches. Even though, many methods can be used for SS, these methods lead to issues of missed detection, poor performance and high false alarm rate. Some of the ML models in SS achieved better detection performance; the cyclostationary as an input feature drastically affects the robustness when the noise power is high. Further, for training large data, overfitting issues occur. Further, these models have computational complexity which leads to increases with increasing layers. Hence a new approach 2ERNN is essential for CRN which hybrid the advantage of ResNet-50 and ERNN is presented in this work. A detection approach is proposed in this work, which is more encouraging than the abovementioned SS techniques and has the best probability detection of PU at less SNR values.

2.4 Problem formulation

The present SS methods for CRN need the utilization of conventional ML or DL model features. The emergence of communication system to the 5G and 6G requires DL trained models since the conventional models take more time and require prior knowledge about the spectrum and PU. Developing an efficient and robust SS is a complex process because of error rate, computational cost and complexity. Further, there is need of improvement on the basis of probability and accuracy detection with less complexity. Hence, for addressing these issues, this work presents a DL model which can learn the features and detects the PU.

The proposed research, tackle the shortcomings highlighted in the literature by introducing a novel approach that connects the combined strengths of ERNN and ResNet-50 architectures, optimized using the BRO algorithm. By integrating ERNN with ResNet-50, we aim to mitigate the challenges associated with missed detection, poor performance, and high false alarm rates encountered in conventional SS methods. The utilization of ERNN allows for capturing temporal dependencies in the data, which enhances robustness, particularly in scenarios with high noise power where cyclostationary features may be less effective. Moreover, the incorporation of ResNet-50 helps address overfitting concerns by facilitating effective feature extraction from large datasets while mitigating computational complexity through residual connections. The BRO optimization algorithm tune the parameters, optimizing its performance, and overcoming convergence challenges associated with traditional optimization methods. As a result, our proposed approach offers a more promising solution for SS, exhibiting superior probability of detection for primary users even at lower SNR, thereby advancing the state-of-the-art in CRN.

3. PROPOSED METHODOLGY

In this work, the Elman Residual Recurrent Neural Network (2ERNN) is utilized for performing SS. When compared to the conventional ML models, the DL models have better learning capacity. Hence, this work uses the DL model for detecting the presence of PU. Initially, the cyclostationary features are extracted and the 2ERNN is trained for distinguishing two hypotheses: HO,H1.

The numerical analysis proved that the proposed 2ERNN outperformed the other DL model and has the better capacity to identify the PU at very small SNR value.

3.1 System model and problem formulation

Spectrum sensing permits the SUs to study about the radio environment by determining the PU signals presence by one or multiple approaches. Since the primary receivers locations are not known to the CRN as there is no signal between CRN and PUs. When CRN has PU and many SUs, for any SU, to detect the PU signal, the following hypotheses are used for the received signal:

$$y(n) = \begin{cases} u(n) & H_0 : PU\ isnotpresent \\ u(n) + hs(n) & H_1 : PU\quad present \end{cases} \tag{1}$$

where, n is number of samples, $u(n)$ is a PU signal that is cyclosationary signal and h is an amplitude channel gain and $s(n)$ is a noise signal (AWGN). The detection of PU signal is carried out by one of the SS methods for deciding two hypotheses. The sensing detection is evaluated using.

$$\begin{cases} if \quad t \geq \alpha \quad H_1 \\ if \quad t < \alpha \quad H_0 \end{cases} \tag{2}$$

where, t is a test statistic and α is a threshold value. This value is set on the basis of probability of false alarm. Threshold (α) is computed by the Eq. (3):

$$\alpha = \max \frac{I(\lambda)}{\sqrt{(I^2(\lambda)/N}} \tag{3}$$

where, $I(\lambda)$ is a correlation coefficient and N is length of observation.

The abovementioned hypothesis is a classical detection issue and a threshold is used for differentiating these hypotheses. When the strength of signal changes with time, detector must be suitable to that condition. Hence, to address this developed model uses an optimization based DL algorithm.

When PU is not present, SU can able to access a PU channel or else it can't able to access to channel. Initially the dataset is generated using QPSK and AWGN noise at SNR of -10. Then, by FFT the cyclostationary features are extracted. At last, the classifier 2ERNN-BRO is trained by the features which are extracted. Finally, the presence of the PU is determined using this classifier. That is the presence and absence of signals is based on the output obtained from the output layers of 2ERNN. This classifier selects the category with the largest value as the last classification outcome.

The developed system is shown in Figure 1. This system has training and prediction stages. In training, features are retrieved from generated data and fed into the optimised DL model. For CS estimation, FFT is utilised. The detection stage detects PU.

3.2 Cyclostationary Feature Detection (CFD)

CFD uses the cyclostationary features of a received signal to identify the PU in spectrum. Generally, the cyclostationary signal presence is detected by cyclic auto correlation function (CAF) and cyclic spectrum. The CAF is utilized to show the cyclic frequency hide in a cyclostationary signal. PU signal and noise are cyclostationary and stationary signals.Hence, both primary signal and noise are differentiated by extracting the various features of the cyclostationary.

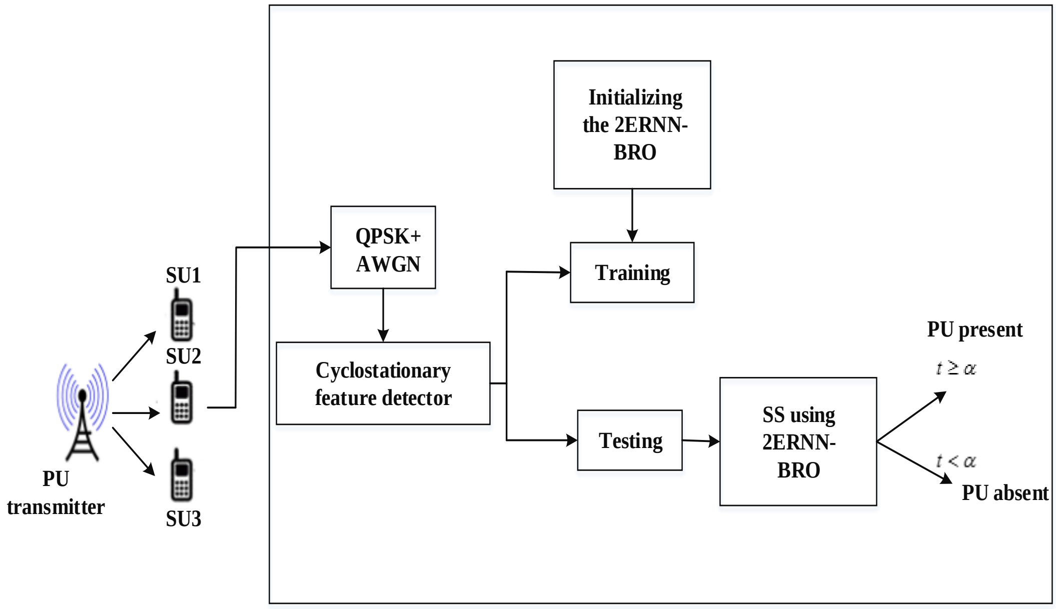


Figure 1. Developed system architecture

When an autocorrelation function that change with time $S_y(t,\tau)$ of zero mean $y(t)$ is periodic on the basis of time t for a parameter τ, and it is called as second order cyclostationary signal. It is defines as:

$$S_y(t,\tau) = E\left\{ y\left(t+\frac{\tau}{2}\right) y^*\left(t-\frac{\tau}{2}\right)\right\} \tag{4}$$

Then, Fourier series is used for decomposing this function and it is given by:

$$S_y(t,\tau) = \sum_{k=1}^{N} S_y^{k\gamma_0}(\tau) e^{j2\pi k\gamma_0 t} \tag{5}$$

where, $\gamma_0 = 1/T$ denotes the basic cyclic frequency with T. The CAF is denoted using the Fourier coefficients $S_y^{k\gamma_0}(\tau)$ and it is given as:

$$S_y^{k\gamma_0}(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} S_y(t,\tau) e^{-j2\pi k\gamma_0 t} dt \tag{6}$$

Then the below equation is used for estimating the coefficients in (5):

$$S_y^{k\gamma_0}(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} S_y(t,\tau) e^{-j2\pi k\gamma_0 t} dt \tag{7}$$

where, the time duration taken to evaluate CAF is given as t_0. When the cyclic frequency is given as $\gamma = k\gamma_0$, the Fourier transform (FT) of CAF is represented as:

$$R_y^{\gamma}(u) = \int_{-\infty}^{\infty} S_y^{\gamma}(\tau) e^{-j2\pi u\tau} d\tau \tag{8}$$

$S_y^{\gamma}(\tau)$ is approximated and given as:

$$S_y^{\gamma}(\tau) = \lim_{t_0\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} y\left(t+\frac{\tau}{2}\right) y^*\left(t-\frac{\tau}{2}\right) e^{-j2\pi\gamma t} dt \tag{9}$$

Further, the Eq. (8) is written by $e^{-j2\pi(\gamma/2)(t+\tau/2)} = 1$:

$$S_y^{\gamma}(\tau) = \lim_{t_0\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} y\left(t+\frac{\tau}{2}\right) e^{-j2\pi(\gamma/2)(t+\tau/2)} \times y^*\left(t-\frac{\tau}{2}\right) e^{-j2\pi(\gamma/2)(t+\tau/2)} dt \tag{10}$$

The cyclic spectrum is also called as spectral correlation function (SCF) which is obtained when $y(t+\tau/2)e^{-j2\pi(\gamma/2)(t+\tau/2)}$ is represented as $x(\tau)$ in Eq. (9) and FT is applied.

$$R_y^{\gamma}(u) = F\left\{S_y^{\gamma}(\tau)\right\} \tag{11}$$

where, $F\{S_y^{\gamma}(\tau)\}$ is a Fourier Transform (FT).

$$R_y^{\gamma}(u) = \lim_{T\to\infty} \frac{1}{T} F\left\{x(\tau) \times x^*(-\tau)\right\} \tag{12}$$

$$R_x^{\gamma}(u) = \lim_{T\to\infty} \frac{1}{T} Y_W\left(u+\frac{\gamma}{2}\right) Y_T\left(u-\frac{\gamma}{2}\right) \tag{13}$$

where, Y_W is a result achieved by g applying FT to rectangular window with W with signal $y(t)$ and it is represented by:

$$Y_W(u) = \int_{-T/2}^{T/2} y(t) e^{-j2\pi ut} dt \tag{14}$$

For a signal which doesn't reveal cyclostationarity, then CS is under the threshold level for every $\gamma=0$. Anyhow, if $\gamma=0$, CS minimizes to conventional ACF and PSD (Power Spectral density). The cyclic frequency γ is related to the signal carrier frequency.

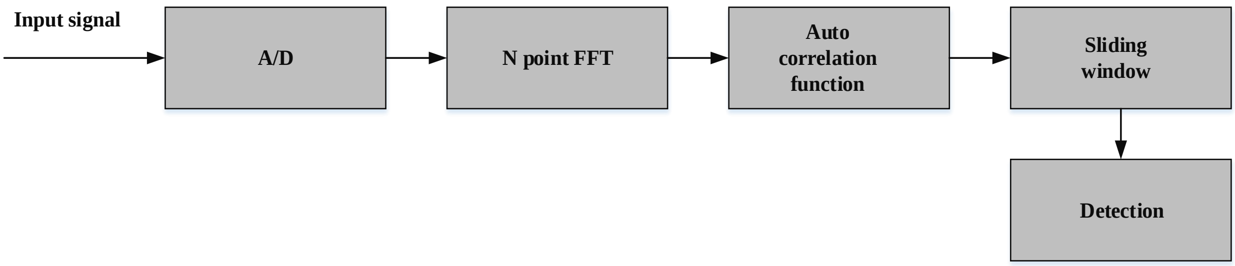


Figure 2. Implementation of CFD

3.3 Implementation of CFD

The analysis of random signals is on the basis of ACF and spectral density. In addition, the cyclostationary signals shows correlation among spectral components because of the spectral redundancy that occur by modulated signal periodicity. Figure 2 depicts the implementation and the procedure of CFD is explained below.

1. Initialize the carrier and cyclic frequency, window and FFT size.
2. Cyclic features are demonstrated by sliding window and this window is useful in effective SS detection.
3. FT of these windowed signals is carried out to continue the calculation in a frequency domain.
4. The SCF is calculated for every frame and then normalized using taking its mean.
5. By the use of SCF, PU will be detected.

3.4 Classification using proposed 2ERNN-BRO

There are two phases in the process of making ResNet50 and ERNN combined model. Initially, ResNet50 is used for extracting the input data features from CFD and the obtained output feature is given as input to ERNN. Finally, this hybrid model is used for finding the presence of PU. This hybrid introduced for effective classification of Pus under various SNR scenarios. ResNet-50 is used to predict the PUs and ERNN is used for classifying the Pus on the basis of user's activities statistical features. That is 2ERNN is trained for distinguishing between the hypotheses. In 2ERNN, output node values are the two probability values and it is summed to one. That is, the output layer of 2ERNN has two nodes and it predict the hypothesis as H_0 or H_1. The working process of 2ERNN is explained below:

The ResNet50 is a fifty layer residual network and the size of the convolutional layer is 3×3 filters and the input size is 224×224. ResNet50 overcome the problems like vanishing gradient issue, degradation issue and network optimization. It develops a residual connection among layers meaning that is layer output is a convolution of its input plus its input. The features in deep residual are extracted from the final convolutional block output of a ResNet-50 and it is pre-trained on ImageNet. It is made by 5 convolutional blocks and is followed by identity block. The ResNet convolutional blocks are different from conventional CNNs duet the new shortcut connection among the input and output of all blocks. This shortcut connection in ResNet provides better optimization and less complexity. This feature of ResNets makes the system train faster and is computationally less expensive than VGG. It has several groups of identical layers as shown by various colors. The curve arrow shows an identity blocks that is used to show the use of prior layers in the following layers. The first layer has 64 filters with a 7×7 kernal size. It is followed by a maxpooling layer with size of 3×3. The final layer of ResNet50 is hybrid with Elman Recurrent Neural Network (ERNN). The ERNN consists of an input, hidden and output layer. Every layer has one or more neurons that propagate information from one layer to another layer by calculating non-linear function of the weighted sum of inputs. Figure 3 depicts the structure of 2ERNN.

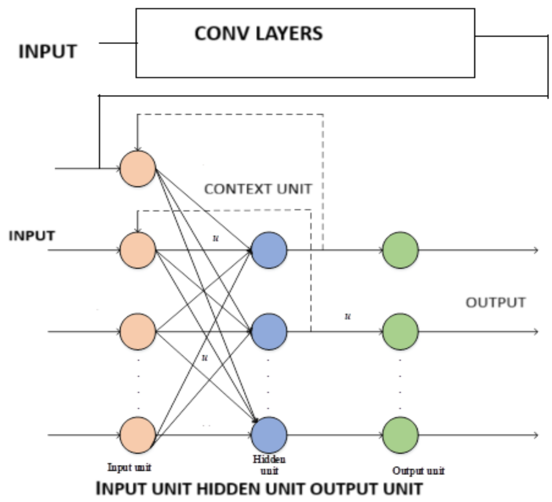


Figure 3. Structure of 2ERNN

Let m and n are the neurons in input and hidden layer. let y_{it} (i=1, 2, ..., m) is a set of input neurons vector at time t and x_{t+1} is a network output at a time t+1 and z_{kt} (k=1, 2, ..., n) is a hidden neurons output at time t. The node that links the input and output layer is given as u.

The hidden layer phase is represented as input of every neuron in a hidden layers and it is represented by:

$$net_{kt}(l) = \sum_{i=1}^{n} uy_{it}(l-1) + \sum_{k=1}^{m} c_k u_{kt}(l) \tag{15}$$

$$u_{kt}(l) = z_{kt}(l-1), \; j = 1,2....m \,, \; k = 1,2....n \tag{16}$$

where, c_k is weight that link the node k in a hidden layer and u_{kt} is a recurrent layer neurons.

The hidden neurons output is given as:

$$\begin{aligned} z_{kt}(l) &= g_H(net_{kt}(l)) \\ &= g_H\left(\sum_{i=1}^{n} uy_{it}(l-1) + \sum_{k=1}^{m} c_k u_{kt}(l)\right) \end{aligned} \tag{17}$$

The sigmoid function of hidden layer is chosen as an activation function and it is given as:

$$g_H(y) = \frac{1}{(1+e^{-y})} \tag{18}$$

where, y is the hidden layer input.

The hidden layer output is given as:

$$x_{t+1}(l) = g_T\left(\sum_{k=1}^{n} v_k z_{kt}(l)\right) \tag{19}$$

where, g_T is an identity map as an activation function.

3.4.1 BRO optimization

To enhance the system performance the optimization technique Battle Royale Optimization (BRO) is used in this work. Most of the optimization is influenced by the nature of the animals and birds. The algorithm BRO [26] is inspired by a game called "battle royale". This algorithm is a population based technique in which every individual is defined by a player who wants to move to the safety place and live. The optimization draws inspiration from the survival-of-the-fittest principle observed in nature, particularly in the context of competitive interactions. The algorithm is based on the concept of simulating the dynamics of a battle royale game, where individuals compete for survival within a confined arena until only the fittest individuals remain.

i) Population initialization

In this 2ERNN- BRO algorithm, every best position in BRO indicates the possible solution. In the initial epoch, biases and best weights are initialized with BRO and those weights are given to 2ERNN. Then, the weights in 2ERNN are computed and in the next epoch, BRO updates the weights with best solution obtained and BRO search for the best weights till the last epoch of the network is reached (probability of detection) achieved. That is 2ERNN is trained and optimized with BRO. Every search agent (player) is selected for representing the initial solution. The position of every player is adjusted on the training by maximizing the fitness function. Here, detection probability P_d is utilized as the fitness function for BRO. Like other metaheuristic approaches, this optimization also initialized by random population and it is equally distributed in a problem space.

ii) Fitness evaluation

In BRO, a population of candidate solutions, or individuals, represents potential solutions to the optimization problem at hand. These individuals are initialized randomly or using predefined strategies, forming the initial population. Each individual is associated with a fitness value, which quantifies its quality or suitability with respect to the optimization objective.

$$fitness\ function = \max(p_d) \tag{20}$$

iii) Degree of damage

The optimization process in BRO proceeds through iterations, known as generations or rounds, mimicking successive stages of the battle royale game. During each generation, individuals engage in competitive interactions, simulating battles where weaker individuals are eliminated while stronger ones survive and proceed to the next round. Every individual aims to attack the closest soldier by shooting a weapon. Hence, the soldier in better positions makes damage to the nearby soldier. If one soldier is attacked by another soldier, then the degree of the damage is increased by 1. These relationships are mathematically represented by the following expression:

$$zj.D = zj.D + 1 \tag{21}$$

where, $zj.D$ is the degree of damage of j^{th} soldier. Also, soldiers need to interchange their position once they face damage, hence attack the enemy from the opposite side. Therefore, for exploitation the soldier who is damaged moves to a place around the better and previous position. These interactions are computed by:

$$z_{dam,d} = z_{dam,d} + r(z_{best,d} - z_{dam}, d) \tag{22}$$

where, r is range from 0 and 1. $z_{dam,d}$ is the soldier who is damaged having dimension d. In addition, when damaged can attack the enemy in the further iteration, $zj.D$ is reset to 0. The soldier comes to problem space once they killed is expressed by the following equation.

$$z_{dam,d} = r(UL_d - LL_d) + LL_d \tag{23}$$

where, UL_d upper limit and LL_d are lower limit with a dimension d and the limit ranges from 0 to 1. Moreover, for entire iteration Δ the achievable search space of problem starts to decrease to better solution. $\Delta = log_{10}(max\ circle)$ is the starting value and $\Delta = \Delta + round\left(\frac{\Delta}{2}\right)$ where max circle is the higher number of generation. This relationship provides both exploitation and exploration. Therefore, the lower and upper limits are given by the following expressions:

$$LL_d = z_{best,d} - Std(\overline{z_d}) \tag{24}$$

$$UL_d = z_{best,d} - Std(\overline{z_d}) \tag{25}$$

where, $Std(\overline{z_d})$ is the standard deviation of the entire population having dimension d and $z_{best,d}$ is the best solution. Therefore, when UL_d and LL_d exceeds the actual limit, then it positioned to an original UL_d and LL_d. Moreover, to focus on elitism, the best soldier determined in every iteration is kept and regarded as elite. Finally, the weight updating by BRO improves the efficiency and detection probability of the system. Throughout the optimization process, individuals continuously evolve and adapt to their environment, undergoing genetic changes and interactions with other individuals. This dynamic interplay between exploration and exploitation enables BRO to effectively search for high-quality solutions to the optimization problem, leveraging the principles of survival and competition observed in nature.

The flowchart of 2ERNN-BRO is given in Figure 4. Steps to identify the presence of PU determined by 2ERNN-BRO are explained below.

Step 1: Initialize the Features from CFD and hyperparameters of 2ERNN and the number of population.

Step 2: Initialize the neurons in layers.

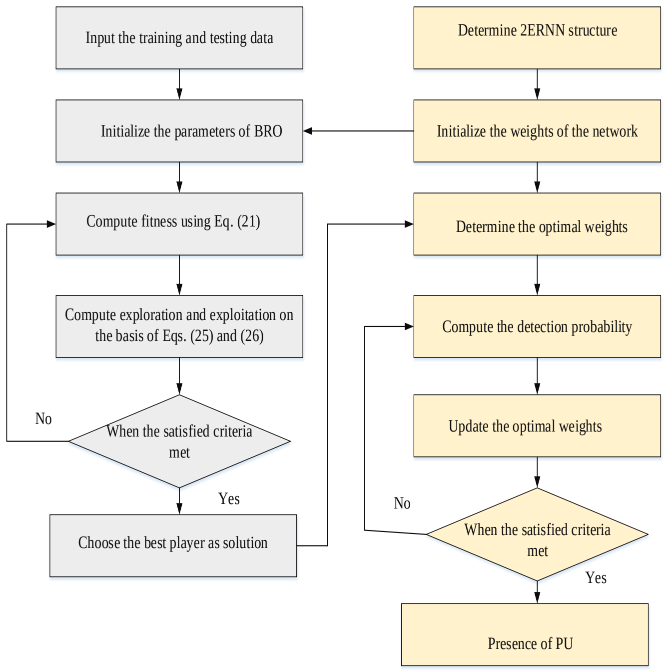


Figure 4. Flowchart of 2ERNN-BRO

Step 3: Estimate the fitness function and every players in the population has weights needed by the network. The fitness is calculated by the Eq. (22).

Step 4: Arrange the players based on the fitness function for generating the position of the population.

Step 5: Compute fitness and update the player's position.

Step 6: Continue until the conditions are reached or recalculate fitness.

Step 7: once the satisfied criteria is met, the optimal weights are given to 2ERNN for training and finding the presence of PU.

The advantages of using this algorithm is that the algorithm is resilient to getting stuck in local optima due to its dynamic and competitive nature, which allows it to explore a wide range of solutions effectively. It has the ability to converge towards global optima efficiently by promoting healthy competition among solutions, leading to faster convergence rates and adapt to different problem domains and optimization objectives by adjusting its parameters and strategies, making it versatile for a variety of applications. It can be easily parallelized, enabling efficient utilization of computational resources and scalability to handle large-scale optimization problems. The competitive nature of BRO encourages the maintenance of diverse solutions, preventing premature convergence and promoting a thorough exploration of the search space.

4. RESULTS

This section provides a performance analysis and discussion of the SS in the CRN of the proposed scheme. The complete implementation was executed on a system equipped with 16 GB of RAM and an Intel Core i7 CPU operating at a speed of 3.0 GHz. The proposed method is implemented using MATLAB 20a. QPSK is compared with other modulations such as BPSK and 16QAM based on their classification and detection performance.

4.1 Dataset generation

Dataset comprises of signal and noise samples. Initially, the dataset is generated by the QPSK modulation and added with AWGN noise. Every sample of signal has 64 symbols and the oversampling ratio is 8; hence the length of every signal is 512. Based on the value of SNR the power signal is adjusted. Then AWGN and signal added and then the SNR value is varied from -20dB to +20dB having the interval of 2dB. For the QPSK modulation, 1000 samples at every SNR are generated as training data and 500 samples are utilized as the test data. Finally, the generated signals are stored in a dataset.

4.2 Performance evaluation

For evaluating the performance of SS approaches, a various have been developed, like detection probability p_d, probability of missed detection p_{md} and probability of false detection p_{fd}. Further to compare the performance of the proposed 2ERNN, the classification measures such as accuracy, specificity, F-score, sensitivity, and precision values are compared with the existing classifiers. The formulas for calculating these measures are represented below:

Detection probability: p_d is the higher the p_d, the higher the PU detection and it is represented as:

$$p_d = P(H_1 / H_1) \tag{26}$$

Probability of false detection: p_{fd} is a probability that the SU reveals the presence of PU when the spectrum is idle.

$$p_{fd} = P(H_1 / H_0) \tag{27}$$

Probability of missed detection: p_{md} is a probability that the SU reveals the absence of PU signal when the spectrum is occupied:

$$p_{md} = P(H_0 / H_1) \tag{28}$$

where, H_0 presence of PU and H_1 is the absence of PU signal.

Accuracy: accuracy is the portion of correctly determined results.

$$A = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{29}$$

Sensitivity: It provides the positive proportions that are exactly determined.

$$Se = \frac{T_p}{T_p + F_n} \tag{30}$$

Specificity: It provides the negative proportions that are exactly determined.

$$Sp = \frac{T_n}{T_n + F_p} \tag{31}$$

F1-score: The ensemble average of sensitivity and precision.

$$F1-score = \frac{2T_p}{2T_p + F_p + F_n} \quad (32)$$

Precision: The proportion of predicted positives which are genuine positive.

$$P = \frac{T_p}{T_p + F_p} \quad (33)$$

False positive rate (FPR): negatives portions that are identified incorrectly.

$$FPR = \frac{F_p}{T_N + F_p} \quad (34)$$

where, T_p, F_p, F_n and T_p are the True positive, False positive, False negative and True negative.

Table 2 represents the hyper parameter setting of the proposed model.

Figure 5 illustrate the performance comparison of the proposed 2ERNN with various machine learning classifiers such as ResNet, ERNN, DBN, DNN, and CNN. The performance of the now available classifiers is implemented and assessed using the same dataset generation method as described earlier. Moreover, the system's performance is empirically shown in terms of likelihood of detection, missed detection, and erroneous detection. Furthermore, the evaluation of the modulation scheme of QPSK is conducted in comparison to BPSK and QAM modulation schemes.

Table 2. Hyperparameter setting

Parameters	Value
Batch size	32
Epoch	100
Dropout ratio	Nil
Learning rate	0.1

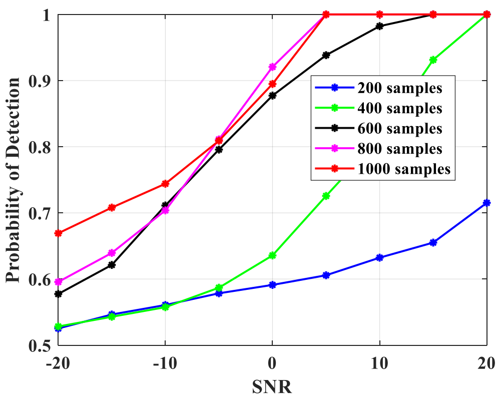


Figure 5. Probability of detection vs SNR by varying samples

P_d is a basic requirement for the output analysis and performance of system. Figure 5 depicts the detection probability by varying the SNR from -20 dB to +20dB and the performance is shown in the ROC curve. The samples are varied from 200 samples to 1000 samples. It is proved that the P_d and SNR are directly proportional to each other. When the SNR value is increased, P_d also increases. It is observed from the figure that CFD has a high probability of detection. The probability of detection for SNR from -20 dB to +20dB for sample 400 is 0.55, 0.56, 0.57, 0.62, 0.73, 0.81, 0.93 and 1.

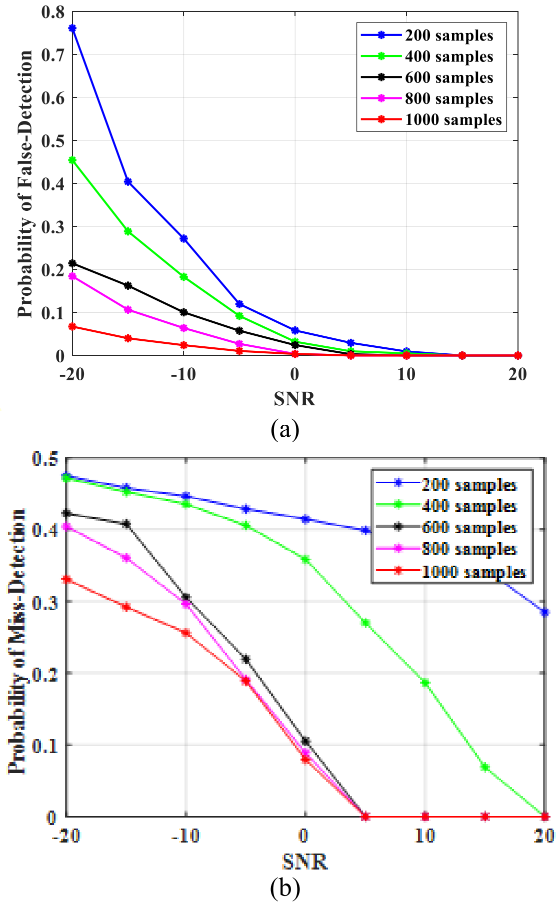
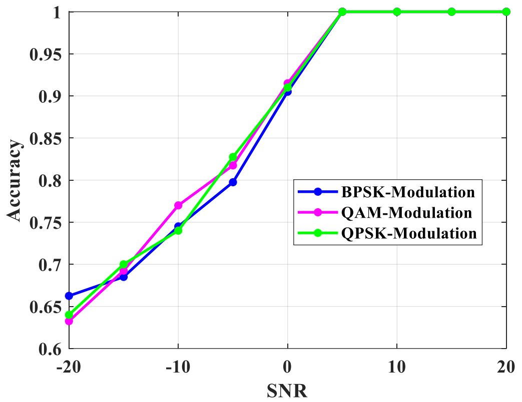


Figure 6. Performance of (a) false detection (b) missed detection

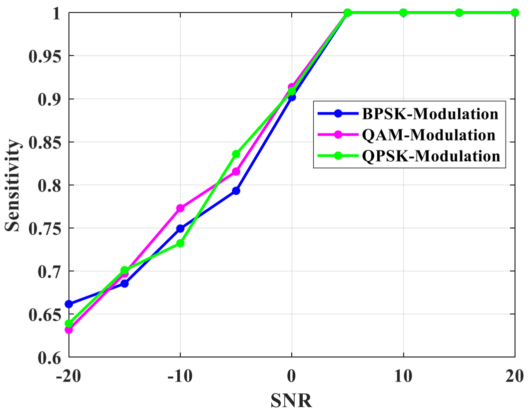
Figure 6 shows the probability of false detection and probability of missed detection vs SNR by varying samples. p_{fd} is probability when PU is detected mistakenly. That means no PU in a desired spectrum however it is detected by the system. It must be less for a system's better performance and better results. when the SNR value is increased, the false detection will be less. Figure 6(b) represents the probability of p_{md} vs SNR by varying samples from 200 to 1000. p_{md} is a probability of missing detection or PU missing during the spectrum detection. It must be less as much as possible. It is seen from the figure that when the SNR is increased, the probability of missed detection is low. Hence, PU will not reappear on the transmission of SU, therefore no interference occur due to SUs to PUs. Particularly, missed detection is very low from the SNR range of +10dB to +20dB which is 0 and hence, the system performance is better.

Figure 7 represents the performance of various modulation techniques with respect to various measures like accuracy, sensitivity, specificity, precision, F score and FPR. The SNR is varied from -20dB to +20dB and the ROC curve is plotted for three modulation schemes like QPSK, BPSK and QAM. From the graph it is observed that when compared to other two modulations, QPSK modulation has better performance.

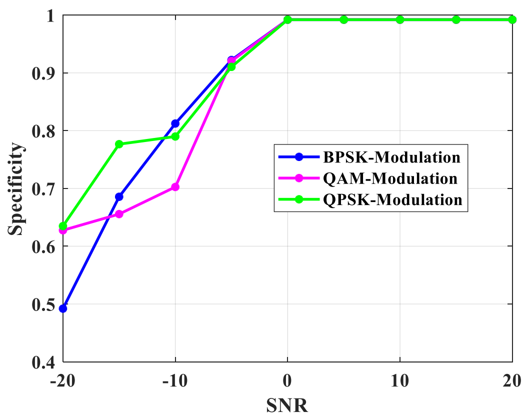
Hence, it is proved that QPSK modulation can be used for high data rate applications therefore improves the spectrum utilization.



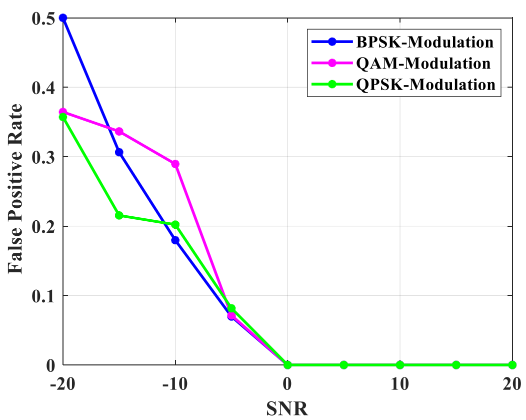
(a)



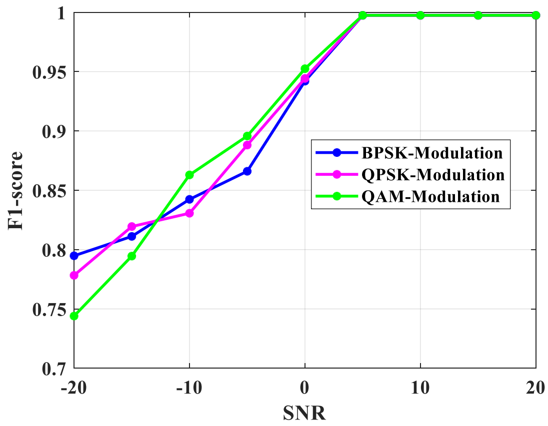
(b)



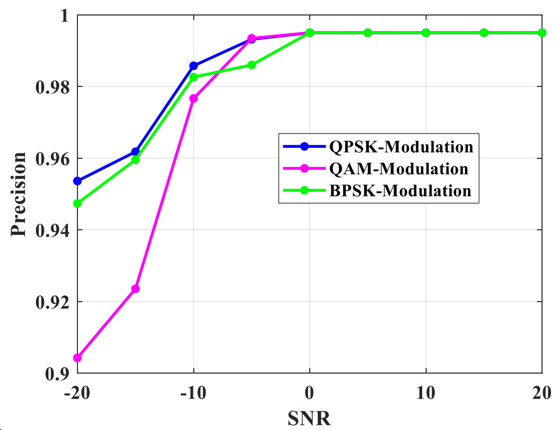
(c)



(d)

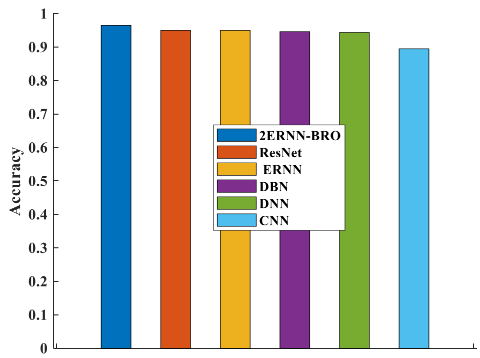


(e)

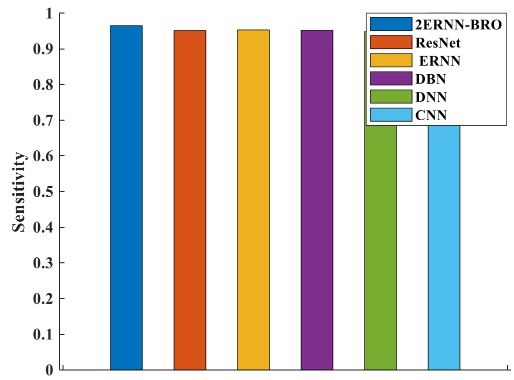


(f)

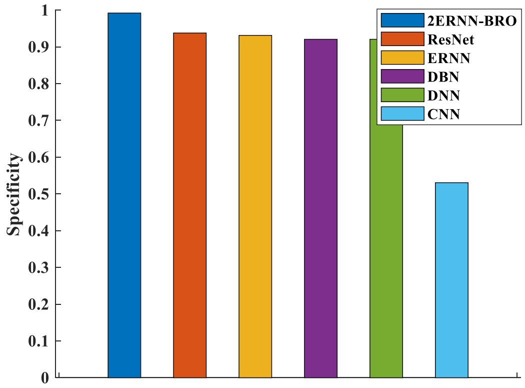
Figure 7. Performance comparison (a) Accuracy; (b) Sensitivity; (c) Specificity; (d) FPR; (e) F1score; (f) Precision



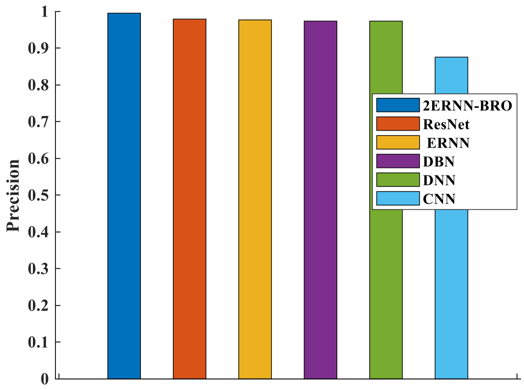
(a)



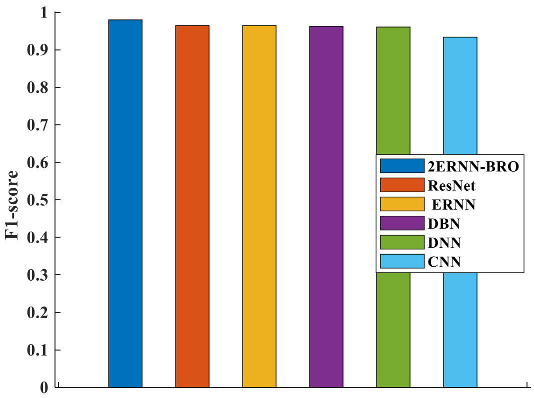
(b)



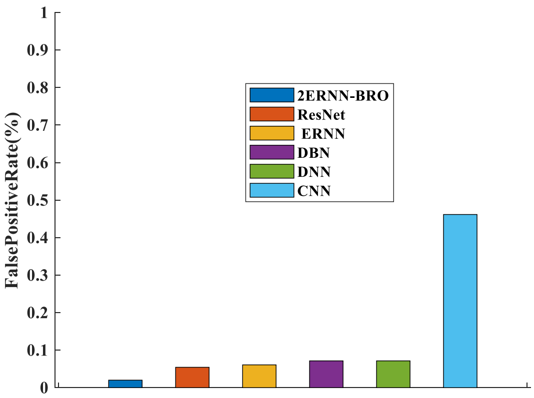
(c)



(d)



(e)



(f)

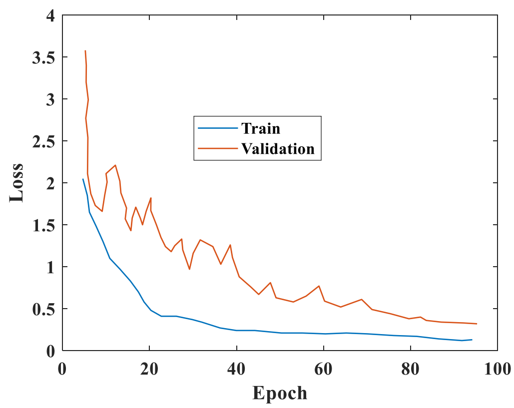
Figure 8. Performance comparison of various classifiers (a) accuracy; (b) sensitivity; (c) specificity; (d) Precision; (e) F1score; (f) FPR

In Figure 8, the proposed classifier 2ERNN-BRO performance is compared with the existing classifiers like Convolutional Neural Network), Deep Neural Network, Deep Belief Network, ResNet and ERNN. From the result, it is observed that the developed classifier achieved better performance. The accuracy of proposed 2ERNN-BRO achieved an accuracy of 98.2% which is 1.1% better than ResNet50, ERNN, DBN and DNN and 7.33% better than CNN. Further, the proposed model achieves less FPR value of about 0.01 and the FPR value of ResNet50 and ERNN are 0.05, FPR value of DBN and DNN are 0.07 and finally for CNN the FPR value is 0.49. The developed classifier achieved better performance because of the optimal weight selection by BRO. Other models achieves less performance due to the high computational complexity and overfitting problems.

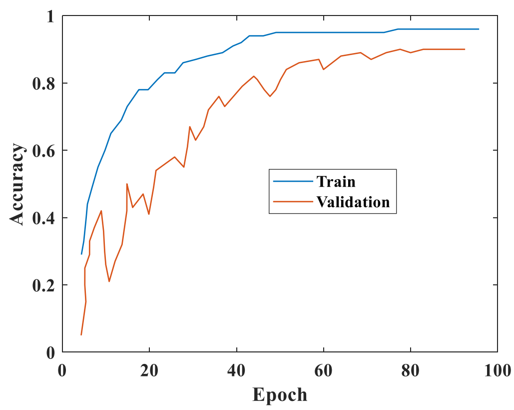
Table 3 shows the comparison of the training time of various approaches like 2ERNN-BRO, DBN, DNN, CNN, ECD and SVM (Support Vector Machine). From the table, it is observed that the training time of the developed classifier is 2.2447 s only. But the traditional methods take more than 2 s to complete the training process. Hence, it is proved that the proposed model 2ERNN-BRO significantly trains faster than the existing methods.

Table 3. Comparison of training time

Methods	Training Time (s)
DBN	3.3913
DNN	8.8680
CNN	23.083
ECD [21]	2.5519
SVM [21]	63.565
2ERNN-BRO	**2.2447**



(a)



(b)

Figure 9. (a) Epochs vs loss of the training and validation set; (b) Accuracy vs epochs of training and validation

Figure 9(a) displays the losses incurred during training and validation. The validation loss curve closely mirrors the corresponding training loss function, indicating the generalisation capabilities of 2ERNN. An increase in the number of epoch values results in a decrease in loss values. Figure 9(b) displays the graph of accuracy against the number of epochs for both training and validation. Analysis of the graph reveals a positive correlation between the epoch value and accuracy. Specifically, when the epoch value is set to 80, the training and validation performance achieves an accuracy of 96% and 80% correspondingly.

4.3 Comparison with other recent research works

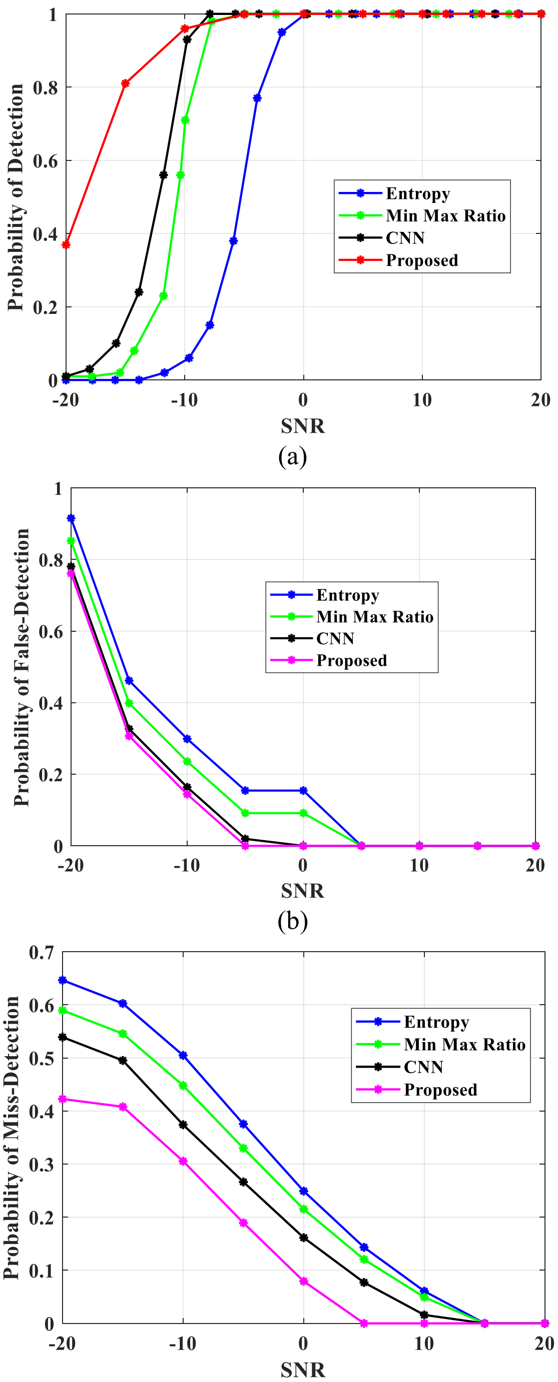


Figure 10. Comparison of (a) Probability of detection; (b) Probability of false detection; (c) Probability of missed detection

The existing SS models like min-max ratio [31], CNN [32] and entropy [33] are considered for the comparison. The min-max ratio model carries out the decision by computing the Eigen-values and it is efficient to noise power uncertainly.

From the result it is observed from the Figure 10 that the proposed model achieved better detection probability and less false detection and missed detection. However, the performance of other models is degraded. Particularly, in low SNR, performance is better for the model. This performance is achieved due to the better optimal weight selection by BRO optimization.

5. DISCUSSION

Several approaches have been introduced in the past like ED, MFD and CFD. The performance of SS can be increased by proper threshold value selection. This research provides a brief insight into several methods utilized in CNR, their classification and techniques. A vast review of related works is given in this work with their challenges. Then these challenges in the related works facilitate in improving the performance of the proposed model. In this work, initially the system model and problem formulation of CRN in SS is discussed. Then the CFD and the extracted features given to 2ERNN is discussed. Then, the weights of 2ERNN are updated by the BRO algorithm. The major aim of this research is to compare the performance of SS with other approaches. Further, the classification performance of 2ERNN-BRO is compared with other DL classifiers and other recent research works and achieved better results in low SNR. That is the proposed SS outperformed other existing SS models like min-max ratio, CNN and entropy. Further, the computational complexity is compared with other approaches and the proposed method complete the training process in less computation time. This indicates that the proposed model takes less time for the overall process and has no computational complexity. Finally, the epoch vs loss and epoch vs accuracy of the training and validation set is given. Loss is measured on training and validation and their evaluation is how better the model is performing for these two sets.

6. CONCLUSIONS

The detection capability of a signal of PU in a low SNR environment is necessary for CRN. To overcome the drawbacks of traditional SS approaches like computational cost and detection of PU in low SNR, a hybrid classifier 2ERNN-BRO is proposed in this work. Initially the dataset is generated by QPSK modulation and AWGN noise is added to it. Then SNR is varied from -20dB to +20 dB and the modulations like BPSK and QAM are compared. The cyclostationary features are extracted and the classifier2ERNN-BRO is trained and tested. Finally, the detection probability p_d, p_{md} and p_{fd} and the classification results are compared. For SNR=10, the probability of detection is 0.99 which shows the efficiency of the system. From the experimental results and ROC curves, it is proved that the modulation QPSK perform well than other two modulations. This model has the ability to detect the presence of PU at less SNR. In the future remarkable improvements in spectrum sensing approach will be investigated by using the most effective signal modulation schemes using dynamic threshold selection. Further, instead of AWGN, the noises like pink noise and colored noise will be considered for the performance evaluation. model offers a powerful and effective solution for spectrum sensing in practical applications. The robustness, adaptability, accuracy, efficiency, and ability to extract information-rich features make it well-suited for a wide range of real-world scenarios.

REFERENCES

[1] Ali, A., Hamouda, W. (2016). Advances on spectrum sensing for cognitive radio networks: Theory and applications. IEEE Communications Surveys & Tutorials, 19(2): 1277-1304. https://doi.org/10.1109/COMST.2016.2631080

[2] Gupta, M.S., Kumar, K. (2019). Progression on spectrum sensing for cognitive radio networks: A survey, classification, challenges and future research issues. Journal of Network and Computer Applications, 143: 47-76. https://doi.org/10.1016/j.jnca.2019.06.005

[3] Ejaz, W., Shah, G.A., Hasan, N.U., Kim, H.S. (2015). Energy and throughput efficient cooperative spectrum sensing in cognitive radio sensor networks. Transactions on Emerging Telecommunications Technologies, 26(7): 1019-1030. https://doi.org/10.1002/ett.2803

[4] Arjoune, Y., Kaabouch, N. (2019). A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions. Sensors, 19(1): 126. https://doi.org/10.3390/s19010126

[5] Althunibat, S., Di Renzo, M., Granelli, F. (2015). Towards energy-efficient cooperative spectrum sensing for cognitive radio networks: An overview. Telecommunication Systems, 59: 77-91. https://doi.org/10.1007/s11235-014-9887-2

[6] Haghighat, M., Sadough, S.M.S. (2014). Cooperative spectrum sensing for cognitive radio networks in the presence of smart malicious users. AEU-International Journal of Electronics and Communications, 68(6): 520-527. https://doi.org/10.1016/j.aeue.2013.12.010

[7] Xue, D., Ekici, E., Vuran, M.C. (2014). Cooperative spectrum sensing in cognitive radio networks using multidimensional correlations. IEEE Transactions on Wireless Communications, 13(4): 1832-1843. https://doi.org/10.1109/TWC.2014.022714.130351

[8] Hamza, D., Aïssa, S., Aniba, G. (2014). Equal gain combining for cooperative spectrum sensing in cognitive radio networks. IEEE Transactions on Wireless Communications, 13(8): 4334-4345. https://doi.org/10.1109/TWC.2014.2317788

[9] Eappen, G., Shankar, T.J.P.C. (2020). Hybrid PSO-GSA for energy efficient spectrum sensing in cognitive radio network. Physical Communication, 40: 101091. https://doi.org/10.1016/j.phycom.2020.101091

[10] Pandit, S., Singh, G. (2017). Spectrum sensing in cognitive radio networks: Potential challenges and future perspective. In: Spectrum Sharing in Cognitive Radio Networks, Springer, Cham. https://doi.org/10.1007/978-3-319-53147-2_2

[11] Claudino, L., Abrao, T. (2017). Spectrum sensing methods for cognitive radio networks: A review. Wireless Personal Communications, 95: 5003-5037. https://doi.org/10.1007/s11277-017-4143-1

[12] Niranjan, M., Khanai, R. (2016). Cognitive radio spectrum sensing: A survey. In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, pp. 3233-3237. https://doi.org/10.1109/ICEEOT.2016.7755301

[13] Alom, M.Z., Godder, T.K., Morshed, M.N. (2015). A survey of spectrum sensing techniques in cognitive radio network. In 2015 International Conference on Advances in Electrical Engineering (ICAEE), Dhaka, Bangladesh, pp. 161-164. https://doi.org/10.1109/ICAEE.2015.7506821

[14] Surampudi, A., Kalimuthu, K. (2016). An adaptive decision threshold scheme for the matched filter method of spectrum sensing in cognitive radio using artificial neural networks. In 2016 1st India International Conference on Information Processing (IICIP), Delhi, India, pp. 1-5. https://doi.org/10.1109/IICIP.2016.7975334

[15] Liu, Y., Zhong, Z., Wang, G., Hu, D. (2015). Cyclostationary detection based spectrum sensing for cognitive radio networks. Journal of Communications, 10(1): 74-79. http://doi.org/10.12720/jcm.10.1.74-79

[16] Amrutha, V., Karthikeyan, K.V. (2017). Spectrum sensing methodologies in cognitive radio networks: A survey. In 2017 International Conference on Innovations in Electrical, Electronics, Instrumentation and Media Technology (ICEEIMT), Coimbatore, India, pp. 306-310. https://doi.org/10.1109/ICIEEIMT.2017.8116855

[17] Ibrahim, D. (2020). Spectrum sensing in cognitive radio networks: Threshold optimization and analysis. EURASIP Journal on Wireless Communications and Networking, 1: 1-19.

[18] Jang, W.M. (2014). Blind cyclostationary spectrum sensing in cognitive radios. IEEE Communications Letters, 18(3): 393-396. https://doi.org/10.1109/LCOMM.2014.012714.132507

[19] Nasser, A., Mansour, A., Yao, K.C., Abdallah, H. (2017). Spectrum sensing for half and full-duplex cognitive radio. Spectrum Access and Management for Cognitive Radio Networks, pp. 15-50. https://doi.org/10.1007/978-981-10-2254-8_2

[20] Mansour, A., Mesleh, R., Aggoune, E.H.M. (2014). Blind estimation of statistical properties of non-stationary random variables. EURASIP Journal on Advances in Signal Processing, 2014: 1-18. https://doi.org/10.1186/1687-6180-2014-21

[21] Ahmad, H.B. (2019). Ensemble classifier based spectrum sensing in cognitive radio networks. Wireless Communications and Mobile Computing, 2019(1): 9250562. https://doi.org/10.1155/2019/9250562

[22] Solanki, S., Dehalwar, V., Choudhary, J. (2021). Deep learning for spectrum sensing in cognitive radio. Symmetry, 13(1): 147. https://doi.org/10.3390/sym13010147

[23] Nasser, A., Chaitou, M., Mansour, A., Yao, K.C., Charara, H. (2021). A deep neural network model for hybrid spectrum sensing in cognitive radio. Wireless Personal Communications, 118(1): 281-299. https://doi.org/10.1007/s11277-020-08013-7

[24] Xie, J., Fang, J., Liu, C., Li, X. (2020). Deep learning-based spectrum sensing in cognitive radio: A CNN-LSTM approach. IEEE Communications Letters, 24(10): 2196-2200. https://doi.org/10.1109/LCOMM.2020.3002073

[25] Soni, B., Patel, D.K., López-Benítez, 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
[26] Peng, Q., Gilman, A., Vasconcelos, N., Cosman, P.C., Milstein, L.B. (2019). Robust deep sensing through transfer learning in cognitive radio. IEEE Wireless Communications Letters, 9(1): 38-41. https://doi.org/10.1109/LWC.2019.2940579
[27] Liu, C., Wang, J., Liu, X., Liang, Y.C. (2019). Deep CM-CNN for spectrum sensing in cognitive radio. IEEE Journal on Selected Areas in Communications, 37(10): 2306-2321. https://doi.org/10.1109/JSAC.2019.2933892
[28] Xie, J., Fang, J., Liu, C., Yang, L. (2020). Unsupervised deep spectrum sensing: A variational auto-encoder based approach. IEEE Transactions on Vehicular Technology, 69(5): 5307-5319. https://doi.org/10.1109/TVT.2020.2982203
[29] Shah, H.A., Koo, I. (2018). Reliable machine learning based spectrum sensing in cognitive radio networks. Wireless Communications and Mobile Computing, 2018(1): 5906097. https://doi.org/10.1155/2018/5906097
[30] Giri, M.K., Majumder, S. (2021). Eigenvalue-based cooperative spectrum sensing using kernel fuzzy c-means clustering. Digital Signal Processing, 111: 102996. https://doi.org/10.1016/j.dsp.2021.102996
[31] Han, D., Sobabe, G.C., Zhang, C., Bai, X., Wang, Z., Liu, S., Guo, B. (2017). Spectrum sensing for cognitive radio based on convolution neural network. In 2017 10th International Congress on Image and Signal Processing, Biomedical Engineering and Informatics (CISP-BMEI), pp. 1-6. https://doi.org/10.1109/CISP-BMEI.2017.8302117
[32] Zhang, Y., Zhang, Q., Wu, S. (2010). Entropy-based robust spectrum sensing in cognitive radio. IET Communications, 4(4): 428-436. https://doi.org/10.1049/iet-com.2009.0389
[33] Radhi, N., Aziz, K., Abbas, S., AL-Raweshidy, H. (2011). Cyclostationary detection in spectrum pooling system of undefined secondary users. In the Seventh International Conference on Wireless and Mobile Communications, pp. 266-270.

NOMENCLATURE

PU	Primary user
SU	Secoundry user
ED	Energy Detection
t	Test Static
BRO	Battle Royal Optimization
CFD	Cyclostationar Frequency Detection
CS	Cyclic Spectrum
CAF	Cyclic Auto Correlation Function
SCF	Spectral correlation Function
PSD	Power Spectral Density
ERNN	Elman Residual Neural Network
VGG	Visual Geometry Group
FT	Fourier Transform

Greek symbols

α	Thereshold $I(\lambda)$
$I(\lambda)$	Correlation co efficient
γ_0	Basic cyclic frequency
Δ	Available search space

Subscripts

Zj.D	Degree of damage
Pd	Detection Probability
Pfd	False Detection
Pmd	Missied Detection
Tp	True positive
Tn	True Negative
Fp	False Positive
Fn	False Negative
H0	Hypothesis false
H1	Hyposthesis True
Ck	Weight of a Node in hidden layer
Ukt	Reccurrent Layer Neuron
Y	Input of hidden layer
gh	Hidden Layer output