

Research Article

Ensemble Classifier Based Spectrum Sensing in Cognitive Radio Networks

Hassaan Bin Ahmad 

Department of Electrical Engineering, Capital University of Science Technology, Islamabad 44000, Pakistan

Correspondence should be addressed to Hassaan Bin Ahmad; hassaanbinahmad@gmail.com

Received 25 June 2018; Revised 23 September 2018; Accepted 10 December 2018; Published 1 January 2019

Academic Editor: Zhou Su

Copyright © 2019 Hassaan Bin Ahmad. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Spectrum sensing is one of the most important and challenging tasks in cognitive radio. To develop methods of dynamic spectrum access, robust and efficient spectrum sensors are required. For most of these sensors, the main constraints are the lack of information about the primary user's (PU) signal, high computational cost, performance limits in low signal-to-noise ratio (SNR) conditions, and difficulty in finding a detection threshold. This paper proposes a machine learning based novel detection method to overcome these limits. To address the first constraint, detection is achieved using cyclostationary features. The constraints of low SNR, finding detection threshold, and computational cost are addressed by proposing an ensemble classifier. First, a dataset is generated containing different orthogonal frequency-division multiplexing signals at different SNRs. Then, cyclostationary features are extracted using FFT accumulation method. Finally, the proposed ensemble classifier has been trained using the extracted features to detect PU's signal in low SNR conditions. This ensemble classifier is based on decision trees and AdaBoost algorithm. A comparison of the proposed classifier with another machine learning classifier, namely, support vector machine (SVM), is presented, clearly showing that the ensemble classifier outperforms SVM. The results of the simulation also prove the robustness and superior efficiency of the detector proposed in this paper in comparison with a cyclostationary detector without machine learning as well as the classical energy detector.

1. Introduction

With the advancement in communication technologies, there is an ever increasing requirement of high data rates. Due to limited natural frequency spectrum when compared to the needs generated by an increasing number of high data-rate devices, it is evident that currently available static frequency allocation schemes are not enough. Consequently, techniques are needed, which are able to exploit the current spectrum in new ways. In order to overcome the challenges posed by spectral congestion, the concept of cognitive radio has emerged as an attractive field of research. It has the potential of opportunistically exploiting less occupied frequency bands [1, 2]. The primary functions of a cognitive radio include the sensing, management, and sharing of spectrum [3]. One main property of the cognitive radio is autonomy in exploiting unused local spectrum. Subsequently, spectrum sensing qualifies as the most important task for establishing cognitive radios and therefore remains till date an open research problem.

The term spectrum sensing means gaining awareness of real-time spectrum utilization and the presence of primary licensed users. In cognitive radio, the user having legacy rights for using a particular part of the spectrum is known as the primary user (PU), whereas the secondary user (SU), having lower priority, tries exploiting the spectrum as long as it causes no interference to signals of the primary users. As a result, the cognitive radio can be used by SUs for sensing the spectrum reliably and confirming the presence of a PU, in which case they switch to another part of the spectrum which is not being used.

1.1. Related Work and Motivation. There are several methods for sensing the spectrum, which have been developed recently, including energy detection (ED), waveform detection (WFD), eigenvalue based detection (EVD), and cyclostationary feature detection (CFD) [4, 5]. The method of energy detection estimates the incoming signal's power and compares it to a previously determined threshold, thereby

determining whether the PU is present or not. However, the performance of this method is significantly affected in conditions having low SNR [6]. Waveform detection method, with the highest reliability, correlates the waveforms of the reference signal and the received signal. The WFD method has a very high efficiency [7] but requires highly accurate information about the signal of the PU. In reality, however, the SUs have no information about the signal of the PU and therefore WFD cannot be used for blind detection. Eigenvalue detection method that was first proposed by Zeng and Liang [4] performs well in low SNR conditions but its high computational complexity is a disadvantage [8].

The method of cyclostationary feature detection proposed initially by Gardner [9, 10] models communication signals as cyclostationary signals [11]. Cyclostationary processes have periodical statistics and are random in nature [10]. The phenomenon of cyclostationarity can occur during coding stages or modulation but can also be introduced intentionally for aiding the processes of synchronization or channel estimation [6]. The CFD offers the advantage of being used in blind context [12, 13]. Since the information about the PU's signal is not available at all, the primary objective is to develop efficient methods of cyclostationary feature extraction [14, 15]. The cyclic spectrum of cyclostationary signals can be estimated using either the strip spectral correlation algorithm (SSCA) or the FFT accumulation method (FAM) [16].

In [17], cyclostationary statistical test is used to detect OFDM signals. Reference [18] uses FRESH filters and cyclostationarity for spectrum sensing. A comparison of energy detector (ED), hybrid energy detector (HED), eigenvalue based Roy Largest Root Test (RLRT), Hybrid Roy Largest Root Test (HRLRT), cyclostationarity test, and hybrid cyclostationarity test is presented in [19]. According to [19], hybrid cyclostationary test performs the best. In [20], a joint energy and cyclostationary method is proposed for blind spectrum sensing. In [21], another cyclostationary approach is presented for blind signal detection. This approach uses a crest factor to obtain a variable threshold.

To address the spectrum sensing task, recently, machine learning (ML) techniques have also been applied [22–25]. Spectrum sensing by employing support vector machine (SVM) is proposed in [22]. Spectrum sensing by using a combination of eigenvalue and SVM is proposed in [23]. Reference [24] presents spectrum sensing using artificial neural networks. An algorithm that uses a combination of covariance matrix and SVM for spectrum sensing is proposed in [25].

The use of ensemble learning methods is another practical way of achieving higher detection accuracy. Recently, ensemble classifiers have been used in many detection problems and have shown promising results [26–33]. Ensemble classifiers make use of multiple learning algorithms in order to achieve a prediction efficiency higher than any of their base learners [34–36]. These classifiers use divide-and-conquer tactics for improving base learner performance to solve a complex problem [37]. Depression detection in speech using ensemble method is proposed in [26]. In [27], ensemble classifier is used for network intrusion detection. In [29], ensemble

decision trees are used for electrocardiograph artifact detection. In [31], ensemble classifier is used for weather radar anomalous propagation echo detection. Ensemble classifier and AdaBoost [38] based islanding detection under smart grid environment is proposed in [33].

In this paper, a spectrum sensing technique based on cyclostationary features and ensemble machine learning is proposed. The use of ensemble classifier for spectrum sensing makes the proposed method different from the existing spectrum sensing methods because ensemble machine learning combines the outputs of weak learners to get the final output. These weak learners are computationally inexpensive and thus offer an advantage over complex learning techniques. None of the existing spectrum sensing techniques use ensemble machine learning as [17] used cyclostationary features and statistical test, [18] used FRESH filters, [19] incorporated hybrid algorithm involving statistics, [20] used energy and cyclostationary feature based learning algorithm, and [21] used a random variable based detection threshold. References [22–25] have used machine learning techniques such as SVM and neural networks for spectrum sensing but no kind of ensemble machine learning has been applied till date.

1.2. Main Contributions. A detection method is proposed in this paper, which is more promising than the above-mentioned spectrum sensing techniques and has a better detection probability. The proposed detector uses ensemble classifier and a signal's cyclostationary features for its detection. This research paper makes the following major contributions: (1) proposing dataset generation algorithm to train and evaluate the classifier, (2) adjusting FAM for estimating the intercepted signal's cyclic spectrum, and (3) proposing ensemble classifier based detector that uses decision trees and AdaBoost algorithm for classification. Furthermore, for performance validation of the proposed ensemble classifier based detector, an SVM and cyclostationary feature based detector is used. For comparison of these two classifiers, performance measures are presented based on accuracy, confusion matrix, receiver operating characteristics (ROC), area under the curve of the receiver operating characteristics (ROC-AUC), and other performance figures obtained from the confusion matrix. Additionally, the results of the simulations showing ROC curves compare the proposed ensemble classifier based detector with a cyclostationary detector without machine learning as well as the classical energy detection method to confirm its robustness and efficiency.

The remainder of the paper is arranged in the following sections. Section 2 presents the system model. Section 3 presents the proposed dataset generation algorithm. Section 4 gives a brief overview of the concept of cyclostationarity. In Section 5, FAM algorithm is presented, which is used for cyclic spectrum estimation. Proposed ensemble classifier based detector is presented in Section 6. Section 7 gives the performance comparison of the proposed classifier and SVM. The simulation results comparing the proposed detector with other techniques is presented in Section 8. Finally, conclusion and possibilities of future work are presented in the last section.

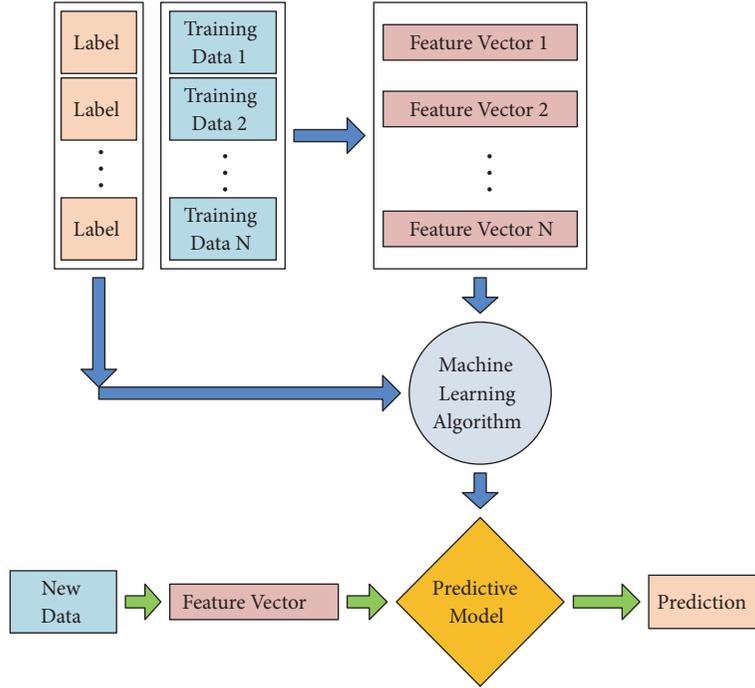


FIGURE 1: Basic machine learning work flow showing training and prediction phases.

2. Proposed System Model

To propose a system model, a cognitive radio system that operates in a dynamic spectrum access (DSA) environment and receives signal through a single antenna is considered. The objective is the identification of spectrum holes [3], which are areas of the spectrum where no other transmitting terminals are present in a particular frequency band. This information can be used for achieving higher spectrum utilization.

First, a signal detection model is set up with \mathbf{x} being the received vector having a length N and containing signal and noise as given by

$$\mathbf{x} = \mathbf{s} + \mathbf{n} \quad (1)$$

where the letters \mathbf{s} and \mathbf{n} stand for the signal and noise vectors, respectively. \mathbf{n} is represented as complex Gaussian, zero-mean, independent, and identically distributed having variance σ^2 , that is, $\mathbf{n} \sim CN(\mathbf{0}, \sigma^2 \mathbf{I})$. The objective is to identify the presence of a signal and therefore the following null and alternative hypotheses are considered:

$$H_0 : \mathbf{x} = \mathbf{n} \quad (2)$$

$$H_1 : \mathbf{x} = \mathbf{s} + \mathbf{n} \quad (3)$$

The above hypotheses represent a classical detection problem and a threshold should be determined to distinguish between the two hypotheses. As the signal strength varies with time, the detector should also adapt accordingly. To tackle this problem, the proposed system uses machine learning algorithm.

A basic machine learning work flow is shown in Figure 1. As shown in the figure, the work flow starts with the training data along with the labels. Labels are required by the machine learning algorithm to distinguish between different types of data. First, features are extracted from the available data. These features and the corresponding labels are provided to the machine learning algorithm block for training. After training the machine learning algorithm block generates a predictive model. This concludes the training phase. Now this generated predictive model is used for predicting new unknown data. In the prediction phase, features are extracted from new data. These features are, then, provided to the predictive model block for the prediction of final output.

System model is presented in Figure 2. The proposed system has two parts: training and detection (prediction). In the training part, first the cyclostationary features are extracted from the training dataset that contains the stored samples of signal and noise. Cyclic spectrum of signal plus noise is expressed as

$$S_{si+no}^\alpha(\nu) = \int_{-\infty}^{+\infty} R_{si+no}^\alpha(\tau) e^{-j2\pi\nu\tau} d\tau \quad (4)$$

and cyclic spectrum of noise is expressed using the following expression:

$$S_{no}^\alpha(\nu) = \int_{-\infty}^{+\infty} R_{no}^\alpha(\tau) e^{-j2\pi\nu\tau} d\tau \quad (5)$$

Then, using the extracted features, an ensemble classifier is trained and a classification model is obtained. As shown in the figure, the ensemble classifier block consists of n weak classifiers and AdaBoost algorithm. This model is then

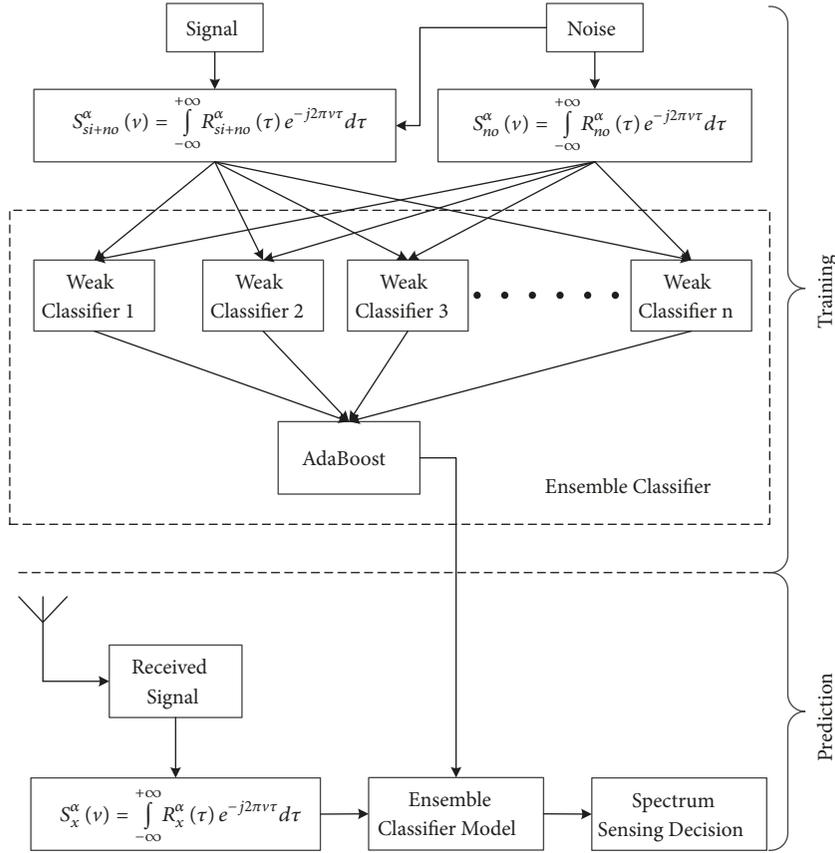


FIGURE 2: Proposed system model showing cyclostationary feature extraction and ensemble classifier based detector.

used for the detection of the PU's signal. In the detection part, cyclostationary features are extracted from the received signal and finally the trained classification model is used for signal detection in the presence of noise. Cyclic spectrum of received signal is expressed using the following expression:

$$S_x^\alpha(v) = \int_{-\infty}^{+\infty} R_x^\alpha(\tau) e^{-j2\pi v\tau} d\tau \quad (6)$$

3. Dataset Generation

Dataset contains orthogonal frequency-division multiplexing (OFDM) signals-with-noise and noise-only samples. As the supervised machine learning techniques need labeled data to distinguish between different categories, this generated data is also assigned labels.

The two categories of data mentioned above are labeled as Signal and Noise, respectively. Half of the elements in dataset are those where signal is present and the rest are only noise elements. Dataset generation algorithm is presented as Algorithm 1. First OFDM signals are generated with BPSK, QPSK, 16-QAM, and 64-QAM. Then signal power is adjusted depending upon the SNR value. Then white Gaussian noise (WGN) is added and also the *Signal* label is assigned. SNR is varied from -5dB to -15dB. Next noise-only signals are generated and *Noise* label is assigned. Finally we store the generated signals in the dataset. Dataset generation algorithm

is proposed so as to remove any bias from the training and validation of the classifier.

4. Cyclostationary Spectrum Analysis

In general, the presence of a cyclostationary signal can be detected using cyclic autocorrelation function (CAF) as well as cyclic spectrum (CS) [39]. The CAF can be used to reveal the cyclic frequencies concealed within a cyclostationary signal, whereas the CS is the equivalent of CAF in frequency domain [40].

4.1. Mathematical Background. If an autocorrelation function that varies with time, $R_x(t, \tau)$, of a zero-mean $x(t)$, is periodic with respect to time t for any parameter τ , then it is termed as a second-order cyclostationary signal, where

$$R_x(t, \tau) = E \left\{ x \left(t + \frac{\tau}{2} \right) x^* \left(t - \frac{\tau}{2} \right) \right\} \quad (7)$$

Consequently, Fourier series can be used to decompose this function.

$$R_x(t, \tau) = \sum_{k=1}^M R_x^{k\alpha_0}(\tau) e^{j2\pi k\alpha_0 t} \quad (8)$$

where $\alpha_0 = 1/T_0$ represents the fundamental cyclic frequency with T_0 being the hidden period. The rank of the

- 1: Iterations = 250
- 2: SNR value from -5dB to -15dB
- 3: ModType = BPSK (It stores the modulation type. It can have one of the four values BPSK, QPSK, 16-QAM and 64-QAM).
- 4: Generate OFDM signal
 - (i) T = OFDM(ModType)
 - (ii) S = Power Adjustment(T, SNR value) + WGN
 - (iii) Assign Signal label
- 5: Generate Noise
 - (i) N = WGN
 - (ii) Assign Noise label
- 6: Dataset = concatenate(Dataset, S, N)
- 7: Update ModType
- 8: Until all ModTypes go to step 4
- 9: Update SNR value
- 10: Until all SNR values go to step 3
- 11: Iterations = Iterations - 1
- 12: Go to step 2 if iterations > 0

ALGORITHM 1: Dataset generation algorithm.

last harmonic is denoted by M [41]. The CAF is represented by Fourier coefficients $R_x^{k\alpha_0}(\tau)$ [42].

$$R_x^{k\alpha_0}(\tau) = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} R_x(t, \tau) e^{-j2\pi k\alpha_0 t} dt \quad (9)$$

The following equation can also be used to estimate the coefficients in (9) [41]:

$$R_x^{k\alpha_0}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} R_x(t, \tau) e^{-j2\pi k\alpha_0 t} dt \quad (10)$$

where the duration of time taken for the evaluation of CAF is represented by T . Equation (9) is used, while the hidden periodicity is already known [41]. If the cyclic frequency is taken as $\alpha = k \cdot \alpha_0$, the CS that is the Fourier transform of CAF [42] takes the form

$$S_x^\alpha(\nu) = \int_{-\infty}^{+\infty} R_x^\alpha(\tau) e^{-j2\pi\nu\tau} d\tau \quad (11)$$

$R_x^\alpha(\tau)$ can be approximated as shown below and mentioned by [43, 44]:

$$R_x^\alpha(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} x\left(t + \frac{\tau}{2}\right) x^*\left(t - \frac{\tau}{2}\right) e^{-j2\pi\alpha t} dt \quad (12)$$

Equation (12) is rewritten using $e^{-j\pi\alpha(\tau/2 - \tau/2)} = 1$, as shown as follows:

$$R_x^\alpha(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} x\left(t + \frac{\tau}{2}\right) e^{-j2\pi\alpha(\alpha/2)(t+\tau/2)} \times x^*\left(t - \frac{\tau}{2}\right) e^{-j2\pi\alpha(\alpha/2)(t-\tau/2)} dt \quad (13)$$

CS, also known as spectral correlation function [45], can be obtained if $x(t + \tau/2)e^{-j2\pi\alpha(\alpha/2)(t+\tau/2)}$ is denoted by $y(\tau)$ in (13) and Fourier transform is applied:

$$S_x^\alpha(\nu) = F\{R_x^\alpha(\tau)\} \quad (14)$$

where $F\{\cdot\}$ represents the Fourier transform.

$$S_x^\alpha(\nu) = \lim_{T \rightarrow \infty} \frac{1}{T} F\{y(\tau) \times y^*(-\tau)\} \quad (15)$$

$$S_x^\alpha(\nu) = \lim_{T \rightarrow \infty} \left\{ \frac{1}{T} X_T\left(\nu + \frac{\alpha}{2}\right) X_T^*\left(\nu - \frac{\alpha}{2}\right) \right\} \quad (16)$$

where $X_T(\nu)$ represents the results obtained by applying Fourier transform to the product of a rectangular window having a width T with signal $x(t)$ and can be defined by

$$X_T(\nu) = \int_{-T_0/2}^{T_0/2} x(t) e^{-j2\pi\nu t} dt \quad (17)$$

$(1/T)X_T(\nu + \alpha/2)X_T^*(\nu - \alpha/2)$ is defined as the cyclic periodogram [16, 46, 47].

4.2. Estimation of Cyclic Spectrum. Estimation of cyclic spectrum can be obtained using frequency smoothing or time smoothing algorithms [48, 49]. The efficiency and reliability of time smoothing algorithms, however, are better compared to frequency smoothing algorithms [16, 45, 48]. If the observation time is taken as t , the time-smoothed cyclic periodogram can be used to estimate the cyclic spectrum as follows:

$$S_x^\alpha(\nu) \approx S_{x_{T_w}}^\alpha(t, \nu)_{\Delta t} = \frac{1}{\Delta t} \int_{t-\Delta t/2}^{t+\Delta t/2} S_{x_{T_w}}(u, \nu) du \quad (18)$$

where

$$S_{x_{T_w}}(u, \nu) = \frac{1}{T_w} X_{T_w}\left(u, \nu + \frac{\alpha}{2}\right) X_{T_w}^*\left(u, \nu - \frac{\alpha}{2}\right) \quad (19)$$

with the width of short-time FFT window denoted by T_w and the short-time Fourier transform (STFT) being

$$X_{T_w}(t, \nu) = \int_{t-T_w/2}^{t+T_w/2} x(u) e^{-j2\pi\nu u} du \quad (20)$$

Here, $\Delta f = 1/T_w$ represents the resolution of the spectral components generated by short-time Fourier transform

- 1: Divide $x[n]$ into P blocks with N_p samples per block.
Skip L data samples between two consecutive blocks. Fix the value of L to $N_p/4$.
Determine the values of N_p and P by:

$$N_p = 2^{\lceil \log_2(F_s/\Delta\nu-1) \rceil + 1}$$

$$P = 2^{\lceil \log_2(F_s/L\Delta\alpha-1) \rceil + 1}$$
- 2: Apply $w(n)$, known as the hamming window across each block.
- 3: Obtain the complex envelope $X_{T_w}(n, \nu)$ by computing N_p - points FFT of each block.
Multiply the output by $e^{-j2\pi(\alpha/2)nT_s}$, for downshifting $X_{T_w}(n, \nu)$ in the frequency: $X_{T_w}(n, \nu + \alpha/2)$
- 4: Compute the complex conjugates of $X_{T_w}(n, \nu + \alpha/2)$.
- 5: Estimate CS by taking product of $X_{T_w}(n, \nu + \alpha/2)$ with its conjugate.
- 6: Compute P - points FFT of the product.

ALGORITHM 2: FAM algorithm.

(STFT). Grenander's uncertainty condition must be taken into consideration for reliably estimating CS [45].

$$\Delta t \cdot \Delta f \gg 1 \quad (21)$$

Time smoothing algorithms that are most commonly used include the strip spectrum correlation algorithm (SSCA) and the FFT accumulation method (FAM) [16]. Since the computational efficiency of the FAM algorithm is higher than SSCA [45, 48], it is preferred over the other.

5. FFT Accumulation Method

If, for a signal $x(t)$, the discrete time version is taken as $x[n]$, the estimation of CS takes the form [21]

$$S_x^\alpha(n, \nu) = \frac{1}{N} \sum_{n=0}^{N-1} \frac{1}{N_p} X_{T_w}\left(u, \nu + \frac{\alpha}{2}\right) X_{T_w}^*\left(u, \nu - \frac{\alpha}{2}\right) \quad (22)$$

where the number of discrete samples that are observed in time Δt is represented by N and the total number of points that are contained within short-time discrete FFT is represented by N_p . $x[n]$'s discrete Fourier transform is denoted by X_{T_w} and is given by

$$X_{T_w}(n, \nu) = \sum_{k=-N_p/2}^{N_p/2-1} w(k) x(n-k) e^{-j2\pi\nu(n-k)T_s} \quad (23)$$

where $x(t)$'s sampling period is denoted by T_s , the data tapering window with a width $T_w = N_p T_s$ seconds is represented by $w(k)$, and $x(n)$'s complex demodulate is represented by the term $X_{T_w}(n, \nu + \alpha/2)$. FAM, derived through (22), works by dividing the bifrequency plane (ν, α) into small sections and then calculating the CS for each section.

The working sequence of FAM as outlined by [6, 46] is presented as Algorithm 2. In FAM algorithm, first $x[n]$, the sequence of input samples with N as its total length, is divided into P blocks with N_p samples contained within each block. L data samples between two consecutive blocks, each having N_p samples, are skipped. The value of L is fixed to $N_p/4$. By doing so, an acceptable balance between cycle aliasing, computational efficiency, and cycle leakage is achieved. The

values of N_p and P are determined in accordance with the desired frequency resolution $\Delta\nu$ and the desired cyclic frequency resolution $\Delta\alpha = 1/\Delta t$, respectively, as well as according to sampling frequency F_s by

$$N_p = 2^{\lceil \log_2(F_s/\Delta\nu-1) \rceil + 1} \quad (24)$$

$$P = 2^{\lceil \log_2(F_s/L\Delta\alpha-1) \rceil + 1} \quad (25)$$

where the integer part of a is denoted by $[a]$. Then $w(n)$, known as the hamming window, is applied across each block. Choosing hamming window allows reduced cycle leakage because it has low sidelobes and skirts [16].

Next, the complex envelope $X_{T_w}(n, \nu)$ is obtained by computing N_p points FFT of each block. Multiply the output by $e^{-j2\pi(\alpha/2)nT_s}$, for downshifting $X_{T_w}(n, \nu)$ in the frequency: $X_{T_w}(n, \nu + \alpha/2)$. Then the complex conjugates of $X_{T_w}(n, \nu + \alpha/2)$ are computed. The CS is estimated by taking product of $X_{T_w}(n, \nu + \alpha/2)$ with its conjugate. Finally P points FFT of the product, is computed to achieve smoothing. The stages of FAM algorithm are illustrated in Figure 3. Figure 4 shows the cyclic spectrum estimation of OFDM signal with QPSK modulation. Figure 5 shows the cyclic spectrum estimation of OFDM-QPSK signal with signal-to-noise ratio equal to -10dB. This figure shows how the cyclic spectrum is affected with degradation in SNR.

6. Proposed Ensemble Classifier Based Detector

Ensemble classifier consists of multiple weak classifiers and an algorithm to combine them. These weak classifiers are trained using the training dataset and a combined ensemble prediction model is generated. Ensemble classifiers also allow for updating the data sources of the weak classifiers, thereby eliminating the need for retraining [27]. They offer the advantage of achieving improved prediction results due to the diversity of weak classifier outputs, since each type of data may represent varied characteristics of the instance to be classified [27]. These classifiers are highly efficient in improving accuracy and reducing false alarms [27]. Figure 6 presents a summarized concept of the ensemble classifier.

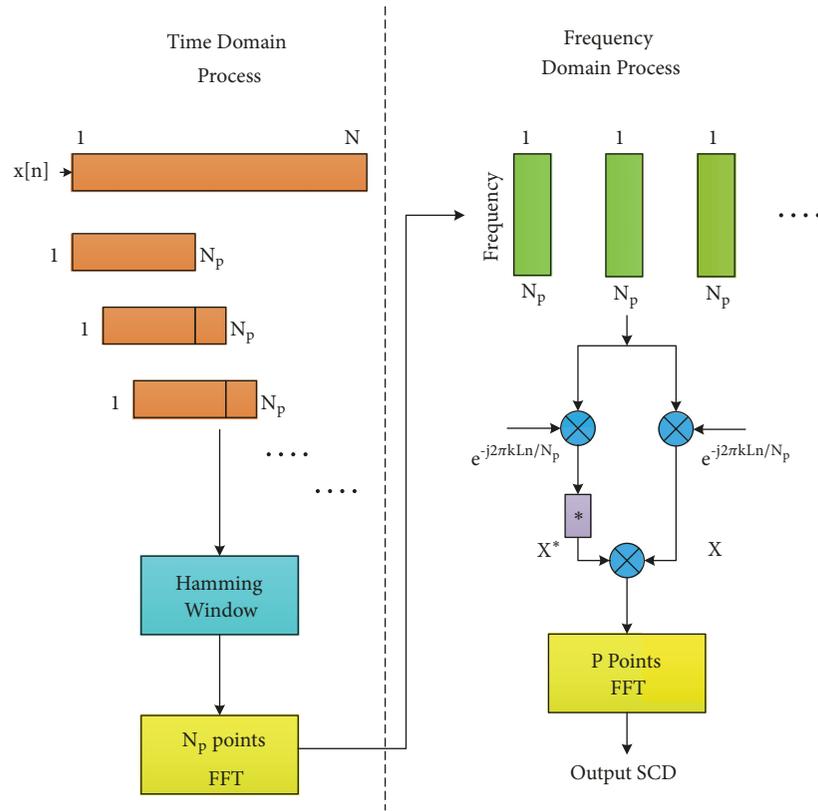


FIGURE 3: Implementation of FFT accumulation method.

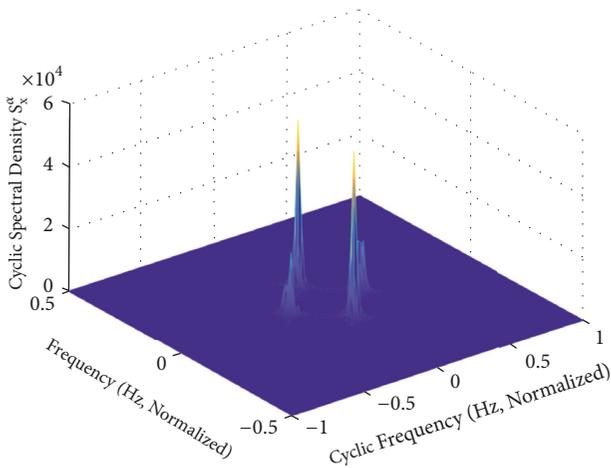


FIGURE 4: Cyclic spectral density of QPSK signal obtained using FAM algorithm.

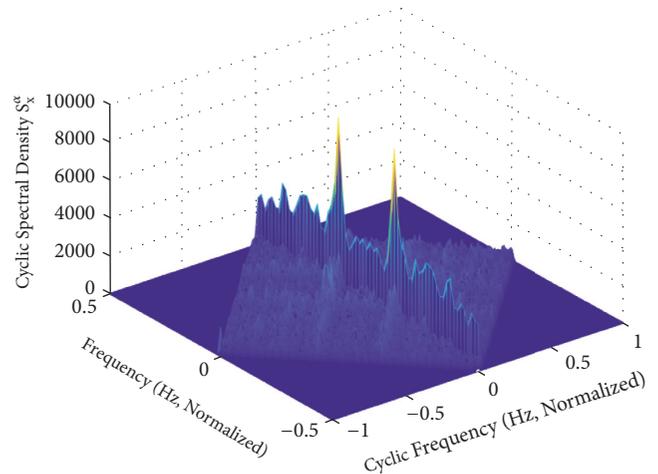


FIGURE 5: Cyclic spectral density of QPSK signal with SNR = -10dB, obtained using FAM algorithm.

The training dataset is given to each of the weak learners with the goal of generating their corresponding models. Each weak classifier predicts the class of the input objects, whereas final classification is obtained using the chosen combination algorithm that combines the outputs of the individual weak classifiers. Decision trees are mostly used as weak classifiers [50]. In the proposed model, decision trees have been selected as the base learners and AdaBoost algorithm has been

selected to combine these base learners to form an ensemble classifier.

6.1. Decision Trees. Decision trees [51, 52] are algorithms that can be used for feature vector classification. Decision trees work by breaking down complex decisions into a series of simpler ones, thereby making the interpretation of results easier [52]. The entire space is initially supposed to be a root

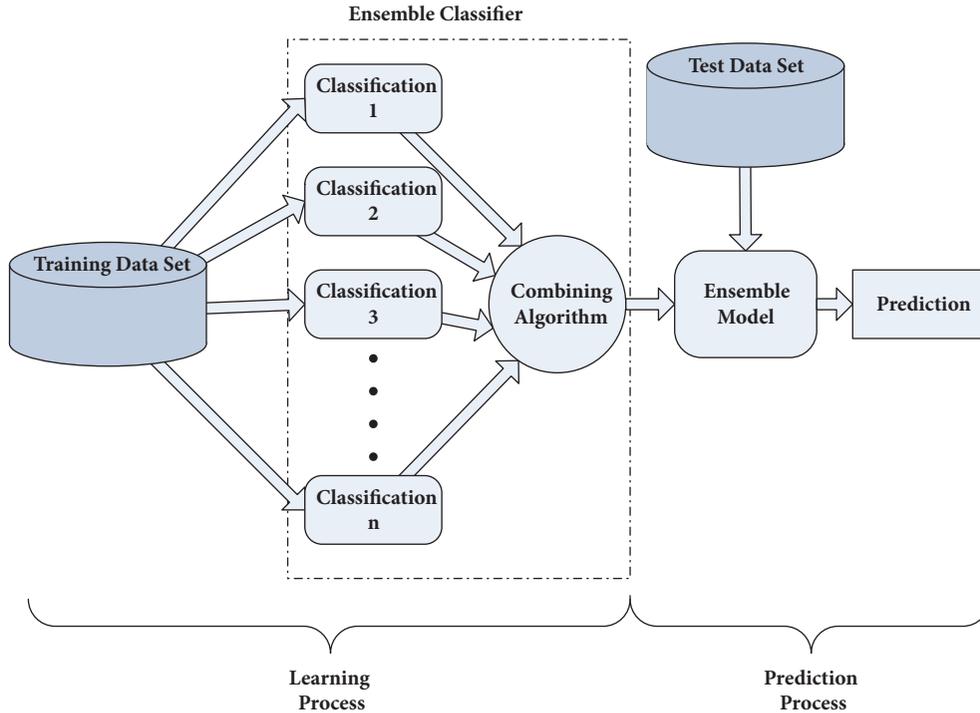


FIGURE 6: Block diagram of ensemble classifier showing the learning and prediction phases.

node. A predictor variable then creates two child nodes by splitting the root node. The child nodes that are created from the root node hold the purest data and further splits can be made. A node that does not split is called a leaf node, also known as the terminal node [53]. The configuration of the leaf nodes is used to make predictions by following the split decisions until arriving at a leaf node.

6.2. AdaBoost. Boosting is a method to combine weak classifiers for performance improvement by sequentially applying the algorithm to reweighted versions of the training dataset and voting for the sequence of classifiers based on weights. Although quite simple, boosting dramatically improves the performance of many algorithms [54]. One machine learning boosting meta-algorithm known as AdaBoost, which is the short form for Adaptive Boosting, is adaptive in nature because successive weak learners are directed to focus more on the misclassified instances of preceding classifiers. Even though the individual learners are weak, the final model converges to a strong learner as long as individual learners show better performance than can be obtained by random guessing. The use of AdaBoost in combination with weak learners such as decision trees makes it one of the best contemporary classifiers [50]. In combination with decision trees as weak learners, information about each sample's relative hardness is collected at each stage of AdaBoost and fed into the tree growing algorithm in order to make the later trees focus more on harder-to-classify instances. In machine learning problems, samples may contain numerous potential features and evaluating each one of them can lead not only to reduction in training and execution speed of the classifier

but also to reduced predictive power [55]. In contrast to SVMs and neural networks, the training process of AdaBoost considers only those features that lead to an improvement to the model's predictive power, thereby improving execution time by reducing dimensionality and omitting irrelevant features. This effect has also been observed for the proposed detector as can be seen in Section 7. Furthermore, AdaBoost is a specific method in which a boosted classifier is trained having a form as given by the following equation:

$$F_T(x) = \sum_{t=1}^T f_t(x) \quad (26)$$

with $f_t(x)$ representing a set of weak learners, taking an input object x and returning the class of the object as the output. In a machine learning problem with two classes, the absolute value of the weak learner's output represents the classification confidence, whereas the output sign predicts the class of the object. For all samples within the training set, an output hypothesis is produced, represented by $h(x_i)$. A weak learner is chosen for every iteration t , and a coefficient α_t is assigned to it in order for the resulting classifier, having t boost stages, to have a minimum cumulative training error E_t .

$$E_t = \sum_i E [F_{t-1}(x_i) + \alpha_t h(x_i)] \quad (27)$$

with $F_{t-1}(x)$ representing the resulting classifier that is boosted and is already built up to the preceding training stage, $f_t(x) = \alpha_t h(x)$, representing the choice of weak learner under consideration to be added to the end classifier and $E(F)$ representing an error function. Within the training set, each

1: Extract cyclostationary features of the signal using FAM Algorithm:

$$S_x^\alpha(n, \nu) = \frac{1}{N} \sum_{n=0}^{N-1} \frac{1}{N_p} X_{T_w} \left(u, \nu + \frac{\alpha}{2} \right) X_{T_w}^* \left(u, \nu - \frac{\alpha}{2} \right)$$

2: For feature vectors obtained in step 1 choose the weak classifier $h_t(x)$ for each t in $1 \dots T$ that minimizes the weighted error

$$\epsilon_t = \sum_{\substack{i=1 \\ h_t(x_i) \neq y_i}}^n w_{i,t}$$

3: Use ϵ_t to find out the weight

$$\alpha_t = \frac{1}{2} \ln \left(\frac{1 - \epsilon_t}{\epsilon_t} \right)$$

4: Add the chosen weak learner and the weight to ensemble

$$F_t(x) = F_{t-1}(x) + h_t(x)\alpha_t$$

5: Update the weights in the expression for ϵ_t for all i

$$w_{i,t+1} = w_{i,t} e^{-y_i \alpha_t h_t(x_i)}$$

6: Re-normalize the weights $w_{i,t+1}$ so that

$$\sum_i w_{i,t+1} = 1$$

ALGORITHM 3: Proposed ensemble classifier based detector.

sample is assigned a weight equal to the sample's current error, $E(F_{t-1}(x_i))$, at every iteration. These weights are then used to influence weak learner's training. For decision trees, weights are used to grow trees that favor splitting up the sets of high-weight samples.

For a dataset $\{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ having a class $y_i \in \{-1, 1\}$ being associated with each object x_i and a classifier set $\{k_1, k_2, \dots, k_L\}$ each having a classification $k_j(x_i) \in \{-1, 1\}$ as the output, the boosted classifier can be represented through a linear combination of weak classifiers after $m - 1$ iterations in the following form:

$$C_{m-1}(x_i) = \alpha_1 k_1(x_i) + \alpha_2 k_2(x_i) + \dots + \alpha_{m-1} k_{m-1}(x_i) \quad (28)$$

This is extended to a better boosted classifier on the m th iteration by adding a weak classifier multiple:

$$C_m(x_i) = C_{m-1}(x_i) + \alpha_m k_m(x_i) \quad (29)$$

The best choice for the weak classifier k_m and the associated weight α_m need to be determined for the above expression. In order to choose the best weak classifier, the total error of C_m , represented by E , defined as the cumulative exponential loss for each x_i , is found as follows:

$$E = \sum_{i=1}^N e^{-y_i C_m(x_i)} \quad (30)$$

If, for $m > 1$, the weights are assumed to be $w_i^{(1)} = 1$ and $w_i^{(m)} = e^{-y_i C_{m-1}(x_i)}$, then the expression for total error takes the form

$$E = \sum_{i=1}^N w_i^{(m)} e^{-y_i \alpha_m k_m(x_i)} \quad (31)$$

For points that are classified correctly by k_m , that is, $y_i k_m(x_i) = 1$, and for those classified incorrectly, that is, $y_i k_m(x_i) = -1$, the total error can be split as follows:

$$E = \sum_{y_i=k_m(x_i)} w_i^{(m)} e^{-\alpha_m} + \sum_{y_i \neq k_m(x_i)} w_i^{(m)} e^{\alpha_m} \quad (32)$$

$$E = \sum_{i=1}^N w_i^{(m)} e^{-\alpha_m} + \sum_{y_i \neq k_m(x_i)} w_i^{(m)} (e^{\alpha_m} - e^{-\alpha_m}) \quad (33)$$

Since only $\sum_{y_i \neq k_m(x_i)} w_i^{(m)}$ is dependent on k_m , the weak classifier k_m , which minimizes the total error E , is the one that minimizes $\sum_{y_i \neq k_m(x_i)} w_i^{(m)}$ and has the lowest weighted error with the weights being $w_i^{(m)} = e^{-y_i C_{m-1}(x_i)}$. For the weak classifier k_m chosen previously, the weight α_m that minimizes the total error E is given by the following differentiation:

$$\frac{dE}{d\alpha_m} = \frac{d \left(\sum_{y_i=k_m(x_i)} w_i^{(m)} e^{-\alpha_m} + \sum_{y_i \neq k_m(x_i)} w_i^{(m)} e^{\alpha_m} \right)}{d\alpha_m} \quad (34)$$

This expression is set to zero and is solved for α_m to give

$$\alpha_m = \frac{1}{2} \ln \left(\frac{\sum_{y_i=k_m(x_i)} w_i^{(m)}}{\sum_{y_i \neq k_m(x_i)} w_i^{(m)}} \right) \quad (35)$$

If, for the weak classifier k_m , the rate of weighted error ϵ_m is expressed as

$$\epsilon_m = \frac{\sum_{y_i \neq k_m(x_i)} w_i^{(m)}}{\sum_{y_i=k_m(x_i)} w_i^{(m)}} \quad (36)$$

the expression for the weight α_m finally becomes

$$\alpha_m = \frac{1}{2} \ln \left(\frac{1 - \epsilon_m}{\epsilon_m} \right) \quad (37)$$

6.3. Algorithm of Proposed Ensemble Classifier Based Detector. Algorithm of the proposed ensemble classifier based detector using AdaBoost is presented in Algorithm 3. First the

		Predicted class	
		Signal	Noise
True class	Signal	True Positive (TP)	False Negative (FN)
	Noise	False Positive (FP)	True Negative (TN)

FIGURE 7: Confusion matrix.

cyclostationary features are extracted using cyclic spectral estimation. Then the detector is trained using these extracted features. For cyclic spectral estimation FFT accumulation method is used.

$$S_x^\alpha(n, \nu) = \frac{1}{N} \sum_{n=0}^{N-1} \frac{1}{N_p} X_{T_w} \left(u, \nu + \frac{\alpha}{2} \right) X_{T_w}^* \left(u, \nu - \frac{\alpha}{2} \right) \quad (38)$$

After obtaining S_x^α , the feature vector is populated. In the training phase, the feature vectors are also accompanied by their corresponding labels. In the detection phase, the feature vector is used to identify the presence of primary user. Feature vectors obtained are stored in the form $\{(x_1, y_1), \dots, (x_N, y_N)\}$ having a class $y_i \in \{-1, 1\}$ for each x_i . Let $h(x)$ represent weak learners, $E(f(x), y, i) = e^{-\gamma_i f(x)}$ represent an error function, and $w_{1,1}, w_{2,1}, \dots, w_{n,1}$ set to $1/n$ representing initial weights. A weak classifier $h_t(x)$ is chosen for each t in $1 \dots T$ which minimizes the weighted error ϵ_t . To find out the weight α_t for the weak classifier ϵ_t is used. Next the chosen weak learner and the weight are added to the ensemble as

$$F_t(x) = F_{t-1}(x) + h_t(x) \alpha_t \quad (39)$$

In the equation of ϵ_t , weights are updated. Finally the weights $w_{i,t+1}$ are renormalized so that their sum is equal to 1.

7. Classifier Performance Measures

To compare the performance of proposed ensemble classifier support vector machine (SVM) classifier is trained on the same extracted features. As mentioned in Introduction, SVM for spectrum sensing is proposed in [22, 23, 25]. The confusion matrix, which summarizes the number of correct and incorrect detections of instances for each event class as shown in Figure 7, is used as the basis for calculating various measures of classification efficiency. It consists of four metrics used for calculating the performance measures for a test set.

The following performance measures are calculated and used for a comparative evaluation of the selected classifiers:

- (1) True positive rate (*TPR*), also referred to as sensitivity, represents the fraction of correctly identified positives.

$$TPR = \frac{TP}{TP + FN} \quad (40)$$

- (2) True negative rate (*TNR*), also referred to as specificity, represents the fraction of correctly identified negatives.

$$TNR = \frac{TN}{FN + TN} \quad (41)$$

- (3) Positive predictive value (*PPV*), also referred to as precision, represents the fraction of positive results that are true positives.

$$PPV = \frac{TP}{TP + FP} \quad (42)$$

- (4) Negative predictive value (*NPV*) represents the fraction of negative results that are true negatives.

$$NPV = \frac{TN}{TN + FN} \quad (43)$$

- (5) False positive rate (*FPR*), also called fall-out, is the proportion of negatives that are incorrectly identified.

$$FPR = 1 - TNR = \frac{FP}{FP + TN} \quad (44)$$

- (6) False negative rate (*FNR*), also known as miss rate, represents the proportion of positives that are incorrectly identified.

$$FNR = 1 - TPR = \frac{FN}{FN + TP} \quad (45)$$

- (7) False discovery rate (*FDR*) represents the proportion of positive results that are incorrectly identified.

$$FDR = 1 - PPV = \frac{FP}{FP + TP} \quad (46)$$

- (8) Accuracy (*ACC*) represents the proportion of correctly identified results, both positives and negatives.

$$ACC = \frac{TP + TN}{TP + FN + FP + TN} \quad (47)$$

- (9) *F1* score represents the accuracy of classification and is the harmonic mean of positive predicted value (*PPV*) and true positive rate (*TPR*) having a value between 0 and 1.

$$F1 = \frac{2TP}{2TP + FN + FP} \quad (48)$$

- (10) *AUC* represents the area under the curve of the receiver operating characteristic curves.

TABLE 1: Performance comparison: ensemble classifier versus SVM.

Performance metric	ECD	SVM
Sensitivity/true positive rate (<i>TPR</i>)	0.932	0.876
Specificity/true negative rate (<i>TNR</i>)	0.973	0.967
Precision/positive predictive value (<i>PPV</i>)	0.9718	0.9633
Negative predictive value (<i>NPV</i>)	0.9347	0.8791
Fall-out/false positive rate (<i>FPR</i>)	0.027	0.033
False discovery rate (<i>FDR</i>)	0.0282	0.0367
Miss rate/false negative rate (<i>FNR</i>)	0.068	0.133
Accuracy (<i>ACC</i>)	0.9525	0.9126
<i>F1</i> score	0.9515	0.9170
Area under the curve (<i>AUC</i>)	0.98	0.97

		Predicted class	
		Signal	Noise
True class	Signal	TP = 932	FN = 68
	Noise	FP = 27	TN = 973

FIGURE 8: Confusion matrix of ensemble classifier.

Figure 8 shows confusion matrix of ensemble classifier. The testing dataset contains 2000 samples, out of which 1000 samples contain signal as well as noise, whereas the other 1000 were noise-only samples. Out of 1000 signal samples, ensemble classifier classifies 932 correctly and from 1000 noise samples it classifies 973 correctly.

Figure 9 shows confusion matrix of SVM. For the same dataset, SVM classifies 876 out of 1000 signal samples and 967 out of 1000 noise samples correctly. Table 1 contains a comparison between ensemble classifier and SVM, based on the performance measures mentioned above. According to these performance measures, ensemble classifier performs significantly better than SVM.

All the performance metrics are achieved with the exact same environment for both classifiers.

The computational cost of the proposed ensemble classifier is less than SVM in terms of training time, whereas the prediction speed of SVM is higher. Table 2 shows the comparison of ensemble classifier and SVM in terms of accuracy, training time, and prediction speed. The ensemble classifier has a better accuracy and trains significantly quicker than SVM but has slightly lower prediction speed.

However, the inherent parallel implementation ability of ensemble classifiers enables them to address the issue of

TABLE 2: Computational cost: ensemble classifier versus SVM.

Performance metric	ECD	SVM
Accuracy (<i>ACC</i>)	0.9525	0.9126
Training time	2.5519 sec	63.565 sec
Prediction speed	$\approx 28.57\mu\text{s}$	$\approx 13.15\mu\text{s}$

		Predicted class	
		Signal	Noise
True class	Signal	TP = 876	FN = 133
	Noise	FP = 33	TN = 967

FIGURE 9: Confusion matrix of SVM.

computational cost. As ensembles use multiple base classifiers, a parallel implementation can reduce the training as well as prediction times significantly. In [56], hardware acceleration of ensemble classifier based on decision tree is presented and targeted towards embedded applications. In [57], an FPGA based implementation is proposed for ensemble classifier using decision tree, which delivers a multifold improvement in speed. In [58], a graphic processing unit based implementation of decision tree ensembles is presented. This method also shows a significant reduction in processing time. Hence the implementation of decision tree ensembles on cognitive radio platform is becoming more realistic and this proposed solution can be a good candidate for spectrum sensing and other cognitive tasks in cognitive radio.

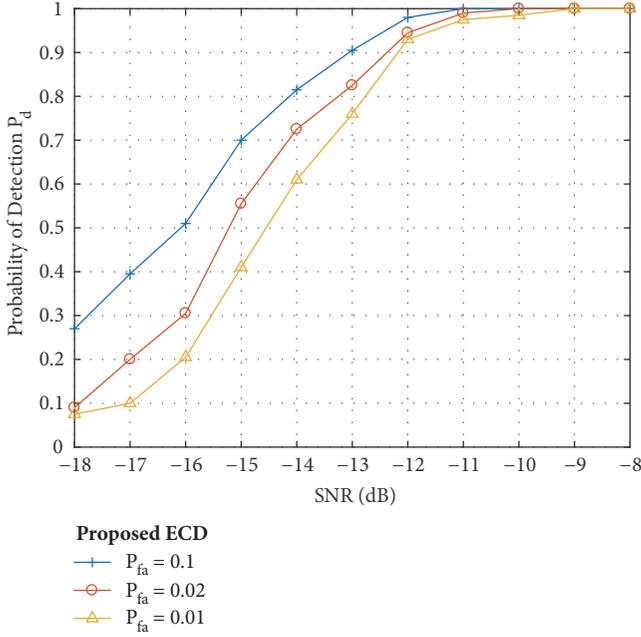


FIGURE 10: Probability of detection versus SNR keeping P_{fa} constant for a 64-QAM signal.

8. Simulation Results

Various OFDM signals, that is, 64-QAM, 16-QAM, BPSK, and QPSK, are selected for analyzing the proposed algorithm's performance. The results achieved for the 64-QAM and QPSK signals are presented in this paper. To compare the proposed ensemble classifier based detector (ECD) with non-machine-learning detectors, a cyclostationary detector and energy detector are considered. The crest factor based cyclostationary detector (CFCD) compared here is presented by [21]. This detector uses crest factor for threshold calculation. The following characteristics of the intercepted signal are considered: data frequency = 1000Hz, carrier frequency = 2000Hz, and sampling frequency = 8000Hz. $T_w = 20ms$, which represents the sliding window time duration, and $\Delta t = 100ms$, which is the time duration for the observation of the intercepted signal, are taken as detector parameters. As a result, the frequency resolution of the detector $\Delta\nu = 1/T_w = 500Hz$ and the cyclic frequency resolution of the detector $\Delta\alpha = 1/\Delta t = 10Hz$. Both the proposed detector and CFCD are simulated with the same parameters. To compare the proposed detector with energy detector, a window size of 100ms is used. The selection of simulation parameters is based on the existing research. Similar parameters are used in [17–19, 21, 24, 25]. These parameters are chosen in order to compare the results of the proposed technique with those of the existing techniques. Figure 10 depicts SNR versus probability of detection (P_d) curves for the OFDM 64-QAM signal, which clearly indicates that the proposed technique performs very well in detecting the signal in low SNR situations.

It is apparent from Figure 10 that the algorithm detects the signal within a Gaussian channel having an SNR value of

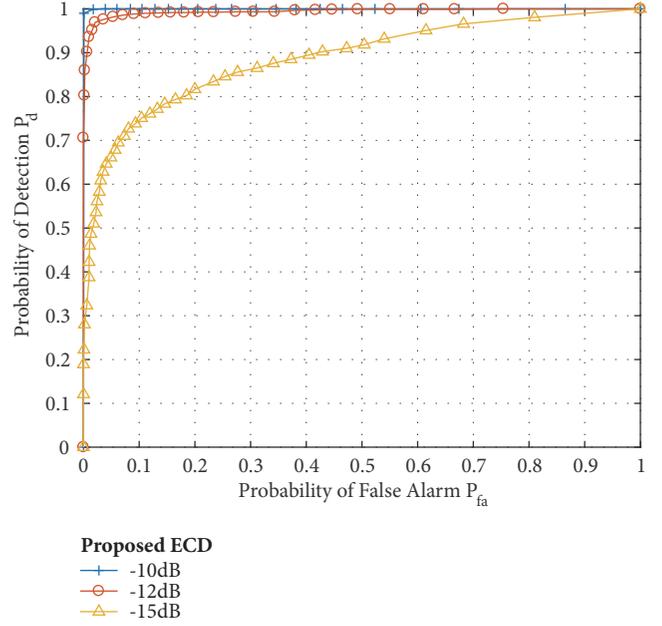


FIGURE 11: ROC curves of proposed ECD. Input consists of all types of signals.

-13dB with $P_d \geq 0.9$. In this case, the value of P_{fa} is fixed at 0.1. Furthermore, the proposed algorithm can successfully detect the signal of the primary user within a Gaussian channel having an SNR value of -12dB with $P_d \geq 0.93$ in the case where the value of P_{fa} is fixed at 0.01. It can also be inferred from Figure 10 that the signal of the primary user can be detected with ease having $P_{fa} \leq 0.1$ for $SNR \geq -11dB$.

Figure 11 shows ROC curves of proposed ensemble classifier detector for -10dB, -12dB, and -15dB. Input consists of the following signals: OFDM BPSK, OFDM QPSK, OFDM 16-QAM, and OFDM 64-QAM. This ROC plot shows an overall performance of the detector.

For an evaluation of the detector's robustness, the ROC curves are generated for various SNR values for OFDM 64-QAM as depicted in Figure 12. The results achieved previously are confirmed through these ROC curves. The relationship between P_d and P_{fa} for various SNR values can be understood in a better way using these curves. OFDM QPSK is the second type of signal chosen for simulations. In this case, for an SNR value of -10dB, the detector easily detects the signal with $P_{fa} = 0.001$ and $P_d = 0.99$. For an SNR value of -12dB, the signal is easily detected with $P_{fa} = 0.002$ and $P_d = 0.9$. For SNR values of less than -15dB, however, it becomes difficult for the proposed ensemble classifier based detector to optimally detect PU's signal as depicted in Figure 13.

Figure 14 shows the ROC curve comparison of cyclostationary ensemble classifier detector, cyclostationary SVM detector, and energy detector, when SNR is -15dB. It can be seen that ECD has the best performance. At $P_{fa} = 0.1$, P_d of ECD is 0.73, P_d of SVM is 0.6, and P_d of ED is 0.15. Figure 15 shows the comparison of ROC curve for ensemble classifier detector, crest factor cyclostationary detector, and energy detector, when SNR is -12dB. It can be seen clearly

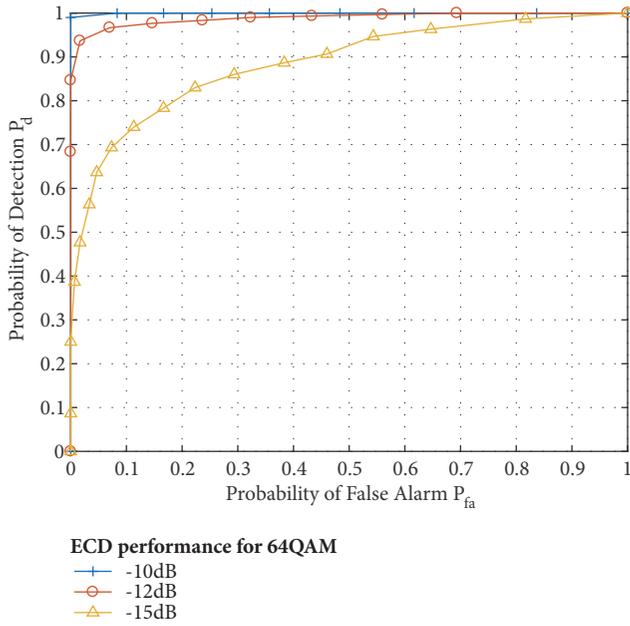


FIGURE 12: ROC curves showing ECD performance for OFDM 64-QAM signal for various SNR values.

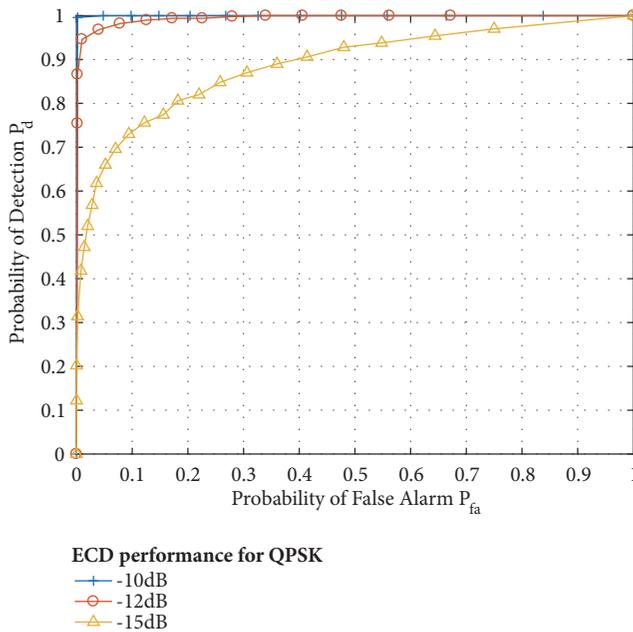


FIGURE 13: ROC curves showing ECD performance for OFDM QPSK signal for various SNR values.

that ECD performs much better than the competition. At $P_{fa} = 0.1$, P_d of ECD is 0.99, P_d of CFCD is 0.52, and P_d of ED is 0.19.

The robustness of ensemble classifier based detector (ECD) is compared to that of crest factor based cyclostationary detector (CFCD) and the classical energy detector (ED).

The same simulation parameters have been used for all techniques so that the comparison can be justified. OFDM

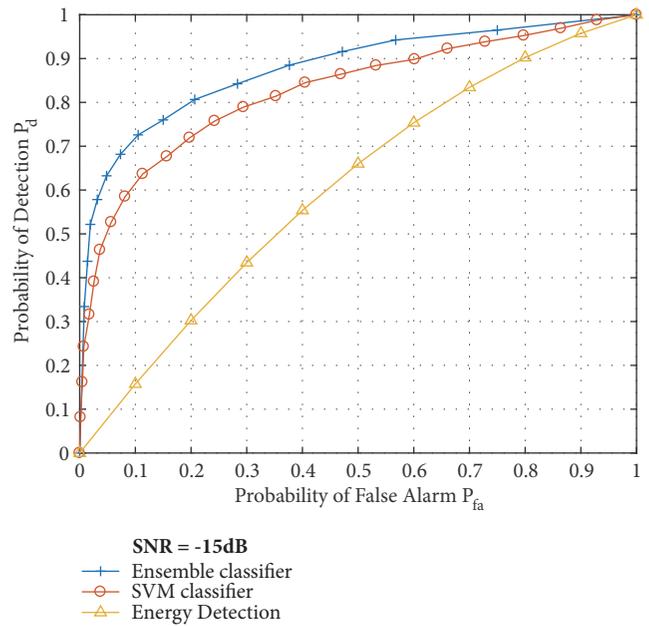


FIGURE 14: ROC curves of proposed ECD versus SVM versus ED.

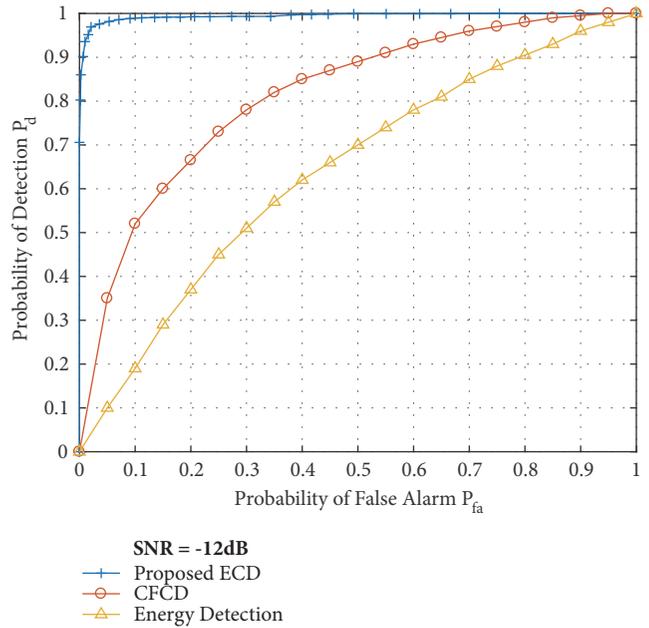


FIGURE 15: ROC curves of proposed ECD versus CFCD versus ED.

64-QAM is chosen as the type of signal. Probabilities of detection of ECD, CFCD, and ED are evaluated for SNR values from -18dB to 0dB for $P_{fa} = 0.1$ and the results of the simulation are presented in the Figure 16.

It can be observed that ECD detects the signal of the primary user within a channel having SNR = -13dB with $P_d = 0.9$ for a value of $P_{fa} = 0.1$, whereas the CFDC and ED detect the same signal with $P_d = 0.9$ and with SNR values of only -10dB and -3dB, respectively. P_d versus SNR curves presented in Figure -16 confirm that ECD has the capability of

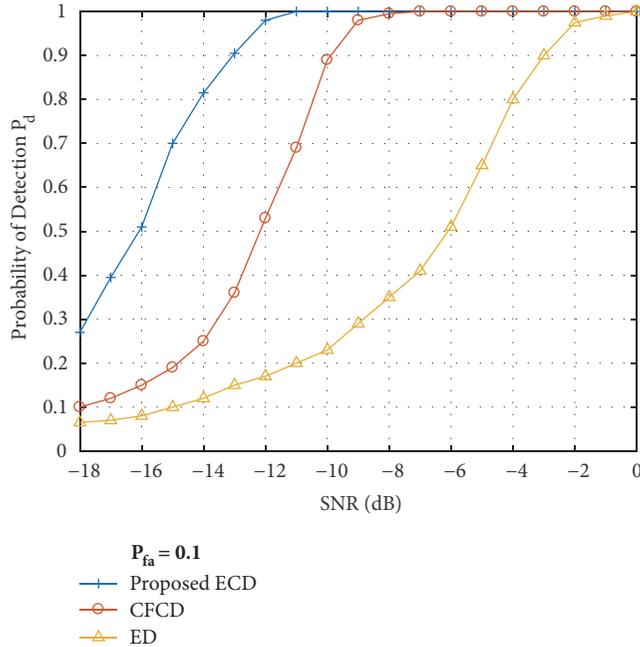


FIGURE 16: Probability of detection versus SNR keeping P_{fa} constant at 0.1 for a comparison of the three detectors.

detecting PU's signal even in very low SNR environments. It outperforms both CFCD and ED.

9. Conclusion and Future Work

The detection ability of the signal of the primary user in a low SNR environment is essential for cognitive radio. Several spectrum sensing techniques are already proposed but each of these techniques has its limitations. These limitations include (1) computational cost, (2) detection of primary user without prior knowledge of signal parameters, (3) detection of primary user in low SNR conditions, and (4) selection of detection threshold. In this research, a novel ensemble classifier and cyclostationary feature based signal detector (ECD) are proposed. Cyclostationary features of communication signals are extracted and used by the ensemble classifier for the detection of PU's signal. The proposed spectrum sensing technique addresses all the above-mentioned challenges.

In this research, first a dataset generation algorithm is proposed. In order to train the machine learning classifier a training dataset is required. This dataset is generated using the proposed dataset generation algorithm and contains all the combinations of signal and noise configuration in Gaussian channel with low SNR conditions. Signal-to-noise ratio is varied from -5dB to -15dB and signal modulations include BPSK, QPSK, 16-QAM, and 64-QAM.

Cyclostationary features are extracted using FFT accumulation method. Ensemble classifier is trained, which uses decision trees and AdaBoost algorithm. Moreover, for comparison of performance, SVM is also trained using the same extracted features. Classifiers are compared with the help of ROC curves and confusion matrix. Based on these

performance metrics, it is evident that the ensemble classifier outperforms SVM. Furthermore, based on the simulation results and ROC curves, the proposed ECD is also more efficient than crest factor cyclostationary detector (CFCD) and classical energy detector (ED).

The future work may include the following. The combination of cyclostationary features and ensemble classifiers for spectrum sensing is a new area, and, therefore, there is a need for further exploration of relevant theories. To name a few examples, the algorithm's computational complexity can be reduced further. In addition, analysis of the impact that multiple antennas would have on the detection probability can be analyzed. Future research can also address extraction of some new features and various combinations of different features to achieve even better performance. Other ensemble classifiers with different combining strategies and heterogeneous weak learners could also be potentially explored.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] J. Mitola III and G. Q. Maguire Jr., "Cognitive radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, 1999.
- [2] "Notice of proposed rule making and order: Facilitating opportunities for flexible, efficient, and reliable spectrum use employing cognitive radio technologies," Tech. Rep. 03-108, Federal Communications Commission, 2005.
- [3] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.
- [4] Y. Zeng and Y.-C. Liang, "Eigenvalue-based spectrum sensing algorithms for cognitive radio," *IEEE Transactions on Communications*, vol. 57, no. 6, pp. 1784–1793, 2009.
- [5] T. Yücek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, 2009.
- [6] V. Prithiviraj, B. Sarankumar, A. Kalaiyaran, P. P. Chandru, and N. N. Singh, "Cyclostationary analysis method of spectrum sensing for Cognitive radio," in *Proceedings of the 2nd International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace and Electronic Systems Technology, Wireless VITAE 2011*, pp. 1–5, Chennai, India, March 2011.
- [7] A. Nasser, A. Mansour, K. C. Yao, H. Charara, and M. Chaitou, "Efficient spectrum sensing approaches based on waveform detection," in *Proceedings of the 3rd International Conference on e-Technologies and Networks for Development, ICeND 2014*, pp. 13–17, Beirut, Lebanon, May 2014.

- [8] M. Z. Shakir, A. Rao, and M.-S. Alouini, "Generalized mean detector for collaborative spectrum sensing," *IEEE Transactions on Communications*, vol. 61, no. 4, pp. 1242–1253, 2013.
- [9] W. Gardner, "Cyclostationary processes," in *Introduction to Random Process with Applications to Signals and Systems*, pp. 323–402, McGraw-Hill, New York, NY, USA, 2nd edition, 1990.
- [10] W. Gardner, "An introduction to cyclostationary signals," in *Cyclostationarity in Communications and Signal Processing*, Chapter 1, pp. 1–81, IEEE PRESS, New York, NY, USA, 1993.
- [11] C. Tom, *Investigation and implementation of computationally efficient algorithm for cyclic spectral analysis [Master's thesis]*, Department of Electronics, Carleton University, Ottawa, Ontario, Canada, 1995.
- [12] W. M. Jang, "Blind cyclostationary spectrum sensing in cognitive radios," *IEEE Communications Letters*, vol. 18, no. 3, pp. 393–396, 2014.
- [13] A. Nasser, A. Mansour, K. C. Yao, and H. Abdallah, "Spectrum Sensing for Half and Full-Duplex Cognitive Radio," in *Spectrum Access and Management for Cognitive Radio Networks*, Signals and Communication Technology, Chapter 2, pp. 15–50, Springer, Singapore, 2017.
- [14] A. Mansour, R. Mesleh, and E.-H. M. Aggoune, "Blind estimation of statistical properties of non-stationary random variables," *EURASIP Journal on Advances in Signal Processing*, vol. 2014, no. 1, pp. 1–18, 2014.
- [15] A. Mansour, "The blind separation of non-stationary signals by only using the second order statistics," in *Proceedings of the Fifth International Symposium on Signal Processing and its Applications*, pp. 235–238, Brisbane, Qld., Australia.
- [16] R. S. Roberts, W. A. Brown, and H. H. Loomis, "Computationally Efficient Algorithms for Cyclic Spectral Analysis," *IEEE Signal Processing Magazine*, vol. 8, no. 2, pp. 38–49, 1991.
- [17] J. An, M. Yang, and X. Bu, "Spectrum Sensing for OFDM Systems Based on Cyclostationary Statistical Test," in *Proceedings of the 6th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, pp. 1–4, Chengdu City, China, September 2010.
- [18] H. Saggarr and D. Mehra, "Cyclostationary spectrum sensing in cognitive radios using fresh filters," in *Proceedings of the 1st ICEIT National Conference on Advances in Wireless Cellular Telecommunications: Technologies & Services*, pp. 1–6, New Delhi, India, 2011.
- [19] R. Garelo and Y. Jia, "Comparison of spectrum sensing methods for cognitive radio under low SNR," in *Proceedings of the 1st IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications, APWC '11*, pp. 886–889, Italy, September 2011.
- [20] M. Bkassiny, S. K. Jayaweera, Y. Li, and K. A. Avery, "Blind cyclostationary feature detection based spectrum sensing for autonomous self-learning cognitive radios," in *Proceedings of the IEEE International Conference on Communications, ICC 2012*, pp. 1507–1511, Canada, June 2012.
- [21] J.-M. Kadjo, K. C. Yao, and A. Mansour, "Blind detection of cyclostationary features in the context of Cognitive Radio," in *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2016*, pp. 150–155, Cyprus, December 2016.
- [22] D. Zhang and X. Zhai, "SVM-Based Spectrum Sensing in Cognitive Radio," in *Proceedings of the 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, pp. 1–4, Wuhan, China, September 2011.
- [23] O. P. Awe, Z. Zhu, and S. Lambotharan, "Eigenvalue and Support Vector Machine Techniques for Spectrum Sensing in Cognitive Radio Networks," in *Proceedings of the Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, pp. 223–227, Taipei, Taiwan, December 2013.
- [24] Y. J. Tang, Q. Y. Zhang, and W. Lin, "Artificial neural network based spectrum sensing method for cognitive radio," in *Proceeding of the 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, pp. 1–4, Chengdu, China, Sep 2010.
- [25] H. Xue and F. Gao, "A machine learning based spectrum-sensing algorithm using sample covariance matrix," in *Proceedings of the 10th International Conference on Communications and Networking in China, Chinacom '15*, pp. 476–480, Shanghai, China, August 2015.
- [26] Z. Liu, C. Li, X. Gao, G. Wang, and J. Yang, "Ensemble-based depression detection in speech," in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 975–980, Kansas City, MO, USA, November 2017.
- [27] V. Timcenko and S. Gajin, "Ensemble classifiers for supervised anomaly based network intrusion detection," in *Proceedings of the 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 13–19, Cluj-Napoca, Romania, September 2017.
- [28] T. Guo, P. Papadopoulos, P. Mohammed, and J. V. Milanovic, "Comparison of ensemble decision tree methods for on-line identification of power system dynamic signature considering availability of PMU measurements," in *Proceedings of the IEEE Eindhoven PowerTech, PowerTech '15*, pp. 1–6, Eindhoven, Netherlands, July 2015.
- [29] J. Moeyersons, C. Varon, D. Testelmans, B. Buyse, and S. Van Huffel, "ECG Artefact Detection Using Ensemble Decision Trees," in *Computing in Cardiology*, pp. 1–4, Rennes, France, Sep 2017.
- [30] Z. Wei and P. Zhang, "Empirical Study of Pedestrian Detection Algorithm Based on Ensemble Learning," in *Proceedings of the 13th IEEE International Symposium on Autonomous Decentralized Systems, ISADS '17*, pp. 175–180, Bangkok, Thailand, March 2017.
- [31] H. Lee and S. Kim, "Decision Tree Ensemble Classifiers for Anomalous Propagation Echo Detection," in *Proceedings of the 8th Joint International Conference on Soft Computing and Intelligent Systems and 17th International Symposium on Advanced Intelligent Systems, SCIS-ISIS '16*, pp. 391–396, Sapporo, Japan, August 2016.
- [32] J. L. Herrera, H. V. Figueroa, and E. J. Ramirez, "Deep fraud. A fraud intention recognition framework in public transport context using a deep-learning approach," in *Proceedings of the 2018 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, pp. 118–125, Cholula, Mexico, February 2018.
- [33] S. S. Madani, A. Abbaspour, M. Beiraghi, P. Z. Dehkordi, and A. M. Ranjbar, "Islanding detection for PV and DFIG using decision tree and AdaBoost algorithm," in *Proceedings of the 3rd IEEE PES Innovative Smart Grid Technologies Europe, ISGT Europe '12*, pp. 1–8, Berlin, Germany, October 2012.
- [34] D. Opitz and R. Maclin, "Popular ensemble methods: an empirical study," *Journal of Artificial Intelligence Research*, vol. 11, pp. 169–198, 1999.
- [35] R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits and Systems Magazine*, vol. 6, no. 3, pp. 21–45, 2006.

- [36] L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review*, vol. 33, no. 1-2, pp. 1–39, 2010.
- [37] Z.-H. Zhou, "Introduction," in *Ensemble Methods: Foundations and Algorithms*, Chapter 1, pp. 1–20, CRC Press, USA, 2012.
- [38] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, part 2, pp. 119–139, 1997.
- [39] A. Al-Dulaimi, N. Radhi, and H. Al-Rawashidy, "Cyclostationary detection in spectrum pooling system of undefined secondary users," in *Proceedings of The Seventh International Conference on Wireless And Mobile Communications*, pp. 266–270, Luxembourg, Belgium, June 2011.
- [40] C. Tom, "Cyclostationary spectral analysis of typical satcom signals using the fft accumulation method," Defence Research Establishment Ottawa (Ontario), Ottawa, Ontario, Canada, Dec 1995.
- [41] W. A. Gardner, "The spectral correlation theory of cyclostationary time-series," *Signal Processing*, vol. 11, no. 1, pp. 13–36, 1986.
- [42] W. A. Gardner and C. M. Spooner, "Signal Interception: Performance Advantages of Cyclic-Feature Detectors," *IEEE Transactions on Communications*, vol. 40, no. 1, pp. 149–159, 1992.
- [43] W. Brown, *On The Theory of Cyclostationary Signals [Ph.D. dissertation]*, University of California, Davis, Calif, USA, Sep 1987.
- [44] W. A. Gardner, "Exploitation of Spectral Redundancy in Cyclostationary Signals," *IEEE Signal Processing Magazine*, vol. 8, no. 2, pp. 14–36, 1991.
- [45] P. Pace, "Cyclostationary spectral analysis for detection of lpi radar parameters," in *Detecting and Classifying Low Probability of Intercept Radar*, pp. 513–525, Artech House, Norwood, MA, USA, 2009.
- [46] E. da Costa, *Detection and identification of cyclostationary signals*, Naval Postgraduate School, Calif, USA, March 1996.
- [47] R. S. Roberts and H. H. Loomis, "Parallel computation structures for a class of cyclic spectral analysis algorithms," *Journal of Signal Processing Systems*, vol. 10, no. 1, pp. 25–40, 1995.
- [48] W. A. Brown and H. H. Loomis, "Digital Implementations of Spectral Correlation Analyzers," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 703–720, 1993.
- [49] R. Robert, W. Brown, and H. Loomis, "A review of digital spectral correlation analysis: Theory and implementation," in *Cyclostationarity in Communications and Signal Processing*, W. A. Gardner, Ed., vol. 6, pp. 455–479, IEEE Press, New York, USA, Feb 1993.
- [50] L. Rokach and O. Maimon, "Decision forests," in *Data Mining with Decision Trees: Theory and Applications*, pp. 99–149, World Scientific, Singapore, 2nd edition, 2015.
- [51] M. Lu, C. L. P. Chen, J. Huo, and X. Wang, "Multi-stage decision tree based on inter-class and inner-class margin of SVM," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, SMC '09*, pp. 1875–1880, USA, October 2009.
- [52] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.
- [53] L. Rokach and O. Maimon, "Top-down induction of decision trees classifiers—a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 35, no. 4, pp. 476–487, 2005.
- [54] I. Mukherjee, C. Rudin, and R. . Schapire, "The rate of convergence of AdaBoost," *Journal of Machine Learning Research (JMLR)*, vol. 14, pp. 2315–2347, 2013.
- [55] G. F. Hughes, "On the Mean Accuracy of Statistical Pattern Recognizers," *IEEE Transactions on Information Theory*, vol. 14, no. 1, pp. 55–63, 1968.
- [56] R. Struharik, "Decision tree ensemble hardware accelerators for embedded applications," in *Proceedings of the 13th IEEE International Symposium on Intelligent Systems and Informatics, SISIS '15*, pp. 101–106, Subotica, Serbia, September 2015.
- [57] M. Owaida, H. Zhang, C. Zhang, and G. Alonso, "Scalable inference of decision tree ensembles: Flexible design for CPU-FPGA platforms," in *Proceedings of the 27th International Conference on Field Programmable Logic and Applications, FPL '17*, pp. 1–8, Ghent, Belgium, September 2017.
- [58] K. Jansson, H. Sundell, and H. Bostrom, "gpuRF and gpuERT: Efficient and Scalable GPU Algorithms for Decision Tree Ensembles," in *Proceedings of the IEEE International Parallel & Distributed Processing Symposium Workshops (IPDPSW)*, pp. 1612–1621, Phoenix, AZ, USA, May 2014.



Hindawi

Submit your manuscripts at
www.hindawi.com

