

Research Article

SPECTRUMNET: Cooperative Spectrum Monitoring Using Deep Neural Networks

M. Suriya ¹ and M. G. Sumithra ²

¹Sri Eshwar College of Engineering, Coimbatore 641202, Tamil Nadu, India

²Dr.N.G.P. Institute of Technology, Coimbatore 641048, Tamil Nadu, India

Correspondence should be addressed to M. Suriya; suriya.m@sece.ac.in

Received 15 September 2022; Revised 14 December 2022; Accepted 15 December 2022; Published 22 December 2022

Academic Editor: Atsushi Mase

Copyright © 2022 M. Suriya and M. G. Sumithra. 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 monitoring is one of the significant tasks required during the spectrum sharing process in cognitive radio networks (CRNs). Although spectrum monitoring is widely used to monitor the usage of allocated spectrum resources, this work focuses on detecting a primary user (PU) in the presence of secondary user (SU) signals. For signal classification, existing methods, including cooperative, noncooperative, and neural network-based models, are frequently used, but they are still inconsistent because they lack sensitivity and accuracy. A deep neural network model for intelligent wireless signal identification to perform spectrum monitoring is proposed to perform efficient sensing at low SNR (signal to noise ratio) and preserve hyperspectral image features. A hybrid deep learning model called SPECTRUMNET (spectrum sensing using deep neural network) is presented. It can quickly and accurately monitor the spectrum from spectrogram images by utilizing cyclostationary features and convolutional neural networks (CNN). The class imbalance issue is solved by uniformly spreading the samples throughout the classes using the oversampling method known as SMOTE (Synthetic Minority Oversampling Technique). The proposed model achieves a classification accuracy of 94.46% at a low SNR of -15 dB, which is an improvement over existing CNN models with minor trainable parameters.

1. Introduction

Due to the prevailing spectrum scarcity problem, recent technologies have emerged to innovate new approaches to efficient spectrum usage and management. The Federal Communications Commission (FCC) is a standard body which is responsible for managing and licensing the allocation of spectrum for all the commercial and noncommercial operations in the United States. These regulatory agencies typically use the fixed spectrum access (FSA) strategy to assign distinct areas of the available radio spectrum to diverse applications [1]. Only approved users, also known as licensed users or primary users (PUs), have the right to use the allotted spectrum under such a fixed and exclusive spectrum allocation policy. Regardless of how busy the designated spectrum is, other users, also known as unlicensed or secondary users (SUs), are not permitted to

use the spectrum. When a band of frequencies designated to a primary user is not used by that user at a certain time and location, it generates “spectrum holes.” Nowadays, researchers’ primary goal is to enhance spectrum usage and propose a novel standard to optimize the existing wireless spectrum. As a result, Mitola coined and suggested the cognitive radio (CR) technology [2] that can intelligently sense the available spectrum and increase spectrum usage by utilizing the idle spectrum called dynamic spectrum access (DSA), which enables the use of available spectrum opportunistically.

The process of sensing the radio frequency spectrum for signal occupancy is called spectrum monitoring, which is an essential spectrum management function [3] used by spectrum managers in identifying occupied and unoccupied frequency bands. If the channel is idle, then all SUs start data transmission and the information about the channel

state (idle or busy) is sent to the fusion center (FC) [4]. It is essential to combine the sensing results of numerous known and unknown SUs to enhance the detection performance, and this process is known as cooperative spectrum sensing (CSS) [5].

Spectrum monitoring is becoming more crucial in commercial, governmental, and military applications as the number of linked devices is increasing rapidly with the development of fifth-generation (5G), sixth-generation (6G), and beyond cellular networks [6]. The outcome of spectrum monitoring is used to assign frequencies efficiently, prevent incompatible usage, and discover sources of harmful interference. A spectrum monitoring system will help to find and eliminate unauthorized or unlicensed interference signals in cognitive radio networks (CRNs). An effective technique to identify and accurately predict the interference problem's cause is done by continuously scanning the spectrum for patterns of undesired signal activity. Spectrum monitoring is used, in addition to interference detection, to assess spectrum occupancy, find white spaces, etc.

Figure 1 outlines the key features associated with cognitive radio spectrum monitoring, and they are as follows:

- (1) Sense: Observe the spectrum continuously for signal occupancy
- (2) Collaborate: Gain collective observation and knowledge from other devices
- (3) Decide: Make a decision and adapt to its current environment
- (4) Act: Anticipate events for future decisions

The radio spectrum is getting increasingly congested due to the increased number of applications in wireless electronic devices that consume more bandwidth. Wireless innovations, including mobile phones, intelligent electronics, and IoT devices, are now a significant force behind commercial development in the business sector. Newer video-based applications need large amounts of wireless bandwidth to deliver the essential mission-critical performance in military and public safety applications. Despite having a significant economic impact, the RF spectrum is a restricted, finite resource, and its access has become increasingly expensive in recent years. The standard regulations body FCC (Federal Communications Commission) conducted an auction for 700 MHz spectrum in 2009, bringing in \$19.5 billion, while the AWS-3 spectrum auction in 2014 brought in \$44.5 billion [7]. Policymakers can use the valuable information gathered from spectrum monitoring to identify unused frequency bands that can be transferred through auctions or repurposed over policy changes. Making informed decisions on spectrum policy and planning, in particular, depends on data from stations monitoring the spectrum continuously over a long period.

The identification of illegal users abusing the expensive spectral resource, the detection of interference, and the assurance of spectral mask compliance are all achieved by spectrum monitoring, which is also crucial for enforcement purposes. Deep learning techniques are highly efficient for

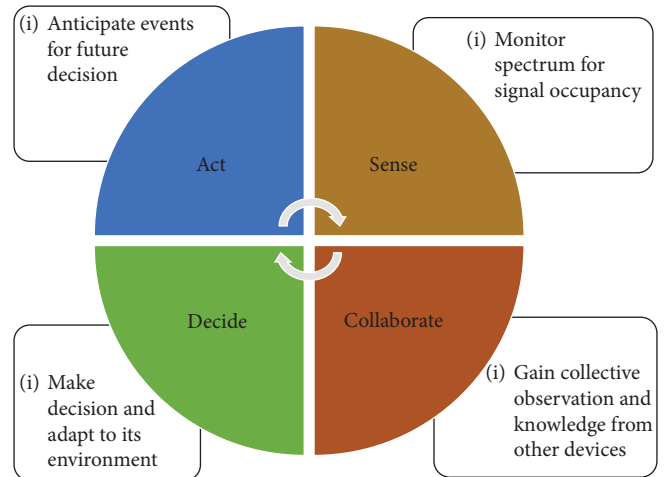


FIGURE 1: Overview of spectrum monitoring.

signal identification in wireless communication networks since big data spectrum datasets need intelligent signal processing algorithms.

This work considers two types of signal classes LTE and WiFi spectrogram signal data denoting the primary and secondary users, respectively. A deep learning model is created to accomplish signal identification that learns the input features from the spectrogram images.

The contribution addressed by this paper is as follows.

- (i) We use cyclostationary detection to extract signal features initially for performing sensing at a low signal-noise ratio (-15 dB to 15 dB) [8] and retain hyperspectral information
- (ii) We train a deep-learning modulation recognition model by building a spectrogram-based convolutional neural network called SPECTRUMNET from big spectrum data of spectrogram signals to perform quick and accurate signal classification
- (iii) We use the SMOTE approach, which addresses the problem of class imbalance in the dataset and resolves the overfitting issue
- (iv) The efficiency of SPECTRUMNET has been assessed using unbalanced and balanced datasets compared with other deep learning spectrum sensing algorithms.

In the rest of the work, the relevant survey and its research are elaborated on in Section 2. Section 3 elaborates the overview of the proposed model, and the SPECTRUMNET model for spectrum monitoring is presented under Section 4. Section 5 analyzes the extensive experimental findings, while Section 6 concludes the work with potential directions.

2. Literature Survey

Deep learning has recently received much attention from researchers working on cognitive radio networks' spectrum sensing research. Several works have been proposed for

efficient RF signal classification, focusing on various challenges of dynamic spectrum allocation. Few research works reflecting the proposed work are elaborated in this section.

2.1. Cyclostationary-Based Feature Detection Methods. Traditionally, the process of signal identification was performed using signal processing tools like cyclostationary-based feature detection. Further signal identification was performed using traditional machine learning algorithms such as the decision tree model, support vector machine, k-nearest neighbor, and artificial neural networks [9]. However, all the conventional techniques need time-consuming characteristics extraction that necessitates a substantial degree of technical and domain expertise. In reference [10], the author first performed signal detection and preprocessing using data from the cycle frequency domain profile (CDP), then identified and classified signals with low signal-to-noise ratio. Then, using a pattern matching approach, the hidden Markov model (HMM) was developed to analyze the retrieved signal characteristics for classification. Authors in reference [11] have incorporated cyclostationary features to perform fast spectrum sensing using the concept of change detection, and they also addressed the computation time and memory issues utilizing their work. According to the study in reference [12], cyclostationary signatures are an effective technique for solving network rendezvous difficulties in Long Term Evolution (LTE) advanced networks and beyond. Instead of the more straightforward periodogram-based detectors, they have suggested an autocorrelation function (AF)-based detector. The author explored the cyclostationary feature of IEEE 802.11 (WiFi) signals, which is caused by their underlying OFDM frame structure, in reference [13]. The team has studied its relevance to resolving the signal-selective direction estimation (SSDE) problem by analyzing the cyclostationary characteristics of WiFi signals and deriving the spectral correlation function (SCF).

2.2. Deep Learning Methods for Signal Classification. In reference [14], Dong Han et al. have developed a convolution neural network (CNN)-based spectrum sensing technique by taking into account the environment when there is low SNR and examining the primary user (PU) detection rate. The cyclostationary feature and energy feature are first extracted to train the CNN. They state that at roughly 0.5 in -20 dB, CNN generated a greater detection probability than cyclostationary feature detection (CFD).

Cooperative spectrum sensing (CSS) was studied by the author in reference [15], who also suggested deep cooperative sensing (DCS), which is based on CNN. By taking into account the geographical and spectral correlation, the authors have made it possible to learn the sensing results of the secondary users using CNN.

The study in reference [16] used a hybrid CNN Long-Short-Term Memory (LSTM) networks based detector called CNNLSTM to extract the correlation of energy features from the covariance matrices produced by the data from sensing

using CNN. The author then used the LSTM to train it on the pattern of PU activity and increase the detection probability by feeding the energy-correlation characteristics for various sensing periods as input.

The author in reference [17] used deep learning networks to conduct spectrum sensing of signals from OFDM signals. The team's initial suggestion was autoencoder-based spectrum sensing, which allows users' actions to be classified by extracting hidden characteristics from OFDM signals, particularly under low SNR settings.

With an emphasis on efficient methods to optimize the energy of distributed CSS, the team in reference [18] examined the use of deep learning algorithms for wireless communication systems. The team has created a deep learning framework to increase the overall system energy efficiency by combining reinforcement learning with graphical neural networks.

An LSTM based-automatic modulation classification is shown in work in reference [19]. The author states that their approach effectively categorizes modulation signals with various symbol rates. To overcome the consequences of uncertainty in the noise power, the author in reference [20] introduced a spectrum sensing approach using deep learning classification. For real-world signals, the team has improved performance using transfer learning techniques.

A 3-layered convolutional neural network was used in the study [21] to compare the effectiveness of three alternative approaches for SPN-43 radar detection. The research team claims their model is better than the LSTM-based recurrent neural network. This research used approximately 14,000 spectrograms in the 3.5 GHz band.

The author in reference [22] has proposed a multiclass classification problem to accurately predict spectrum scenarios. By testing with deep neural networks (DNNs), CNNs, and LSTM to identify signals with varying levels of SNR, they have stated that deep learning has improved sensing.

From the literature, it can be seen that there are several methods for classifying signals using machine learning and deep learning. However, preserving hyperspectral features and class imbalance is still an issue. We proposed a cyclostationary feature-based detection to resolve this issue, and the resulting reduced features are utilized by the CNN model using fewer parameters. We used the SMOTE technique to overcome the data class imbalance and correctly predict the presence of the primary user.

3. Proposed Spectrum Monitoring Model

Figure 2 depicts the proposed spectrum monitoring model's overview. It is a two-stage procedure; in the first phase, the PU signal's cyclostationary information is used to extract its features [23]. Then, the proposed SPECTRUMNET model performs signal identification utilizing the extracted features as input after preprocessing the imbalanced dataset in the second step. The detailed process of step 1 and step 2 are elaborated under Sections 3.1 and 3.2.

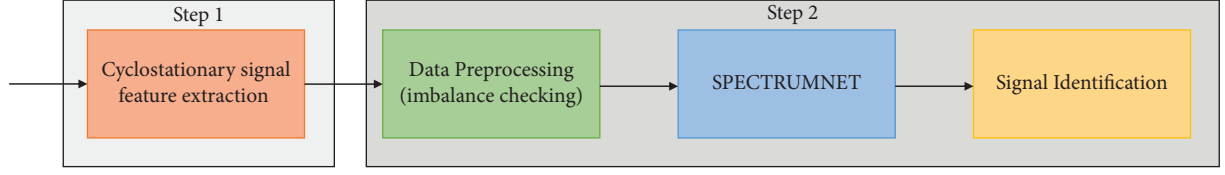


FIGURE 2: Overview of the proposed spectrum monitoring model.

3.1. Cooperative Sensing Using Cyclostationary Feature Detection. Cyclostationary feature detection is a sensing technique for detecting PU transmissions by processing cyclostationary features of the received signals. One PU and several SUs comprise a cognitive radio network (CRN) scenario that may be modeled as a binary hypothesis testing where the SUs perform cooperative spectrum sensing-based spectrum monitoring. Traditional CSS schemes include both centralized and distributed CSS. In our method, we consider the centralized CSS, where each node accumulates local sensing data and transmits it to fusion center (FC), which then gives a decision feedback based on the fusion rules. The centralized cooperative spectrum sensing uses the fusion rules like AND, OR, and majority-based methods at the fusion center for performing decision making [24]. In equations (1) and (2), the essential hypotheses H_0 and H_1 are considered as follows.

H_0 : Power of PU absent at time “ t .”

H_1 : Power of PU present at the time “ t .”

$$H_0: x(t) = n(t), \quad (1)$$

$$H_1: x(t) = h(t) + n(t), \quad (2)$$

where $t=0, 1, \dots, N-1$, where N represents the total samples of the received signal over time, $x(t)$ denotes the signal received by the SU at time “ t ,” $h(t)$ indicates the signal transmitted by the PU at time “ t ,” and $n(t)$ represents the additive white Gaussian noise (AWGN) with variance σ_n^2 .

The noise and signal features are extracted from the features of the cyclostationary signal for PUs. The SU’s cyclic autocorrelation function may be expressed as equation (3) if SU receives the signal $h(t)$.

$$R_x^\alpha = \frac{1}{T_0} \int_0^{T_0} R_x(t, \tau) e^{-j2\pi\alpha\tau} d\tau, \quad (3)$$

where $R_x^\alpha(\tau)$ = autocorrelation function, α = cyclic frequency, and T_0 = cycle period.

The power spectral density and autocorrelation are used to compute the spatial correlation as denoted in equation (4).

$$S_x^\alpha(f) = \int_{-\infty}^{\infty} R_x^\alpha(\tau) e^{-j2\pi f\tau} d\tau, \quad (4)$$

where $S_x^\alpha(f)$ = power spectral density.

(i) auto-correlation function, (ii) spectral correlation function, and (iii) energy function of the extracted signals are computed as in equations (5)–(7), respectively, using the binary hypothesis-testing function

$$R_x^\alpha = \begin{cases} R_{x,0}^\alpha, & H_0 \\ R_{x,1}^\alpha, & H_1 \end{cases}, \quad (5)$$

where $H_0 = R_{x,0}(t, \tau) e^{-j2\pi\alpha\tau}$ and $H_1 = R_{x,1}(t, \tau) e^{-j2\pi\alpha\tau}$.

$$S_x^\alpha(f) = \begin{cases} S_{x,0}^\alpha(f), & H_0 \\ S_{x,1}^\alpha(f), & H_1 \end{cases}, \quad (6)$$

where $H_0, H_1 = \rho_n^2 \sigma(\alpha)$, since noise is the white Gaussian noise.

The extracted signals’ energy feature is obtained as in equation (7).

$$E_{e,x} = \begin{cases} E_{e,0}, & H_0 \\ E_{e,1}, & H_1 \end{cases}, \quad (7)$$

where $H_0 = \sum_{t=1}^N (n(t))^2$ and $H_1 = \sum_{t=1}^N (h(t) + n(t))^2$.

The preprocessed data can be used to create the train and test dataset after the extracted features, allowing the SPECTRUMNET model to identify signals [25].

3.2. Deep Learning Model for Signal Classification. Deep learning approaches have been proposed for signal identification in wireless communication networks as a result of their impressive performance for various applications [26]. The primary steps involved in the deep learning-based sensing model for signal classification are as follows:

- (1) The relationship among set of inputs and outputs X and Y , respectively, is expressed mathematically by the function F , as shown in equation (8).

$$F: X \longrightarrow Y. \quad (8)$$

- (2) Input $X \in \mathbb{R}^{m \times n}$ represents set of different observations which is represented as in equation (9).

$$X = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_m^T \end{bmatrix}. \quad (9)$$

- (3) Then, m denotes the size of sample and $x_i \in \mathbb{R}^n$ in equation (10) denotes “ n ” features (or) labels of every i^{th} observation called feature vector that contains

$$x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T, \quad (10)$$

where $i = 1, 2, \dots, m$

- (4) The output $Y \in \mathbb{R}^m$, denotes the target or output corresponding to the m inputs x_i , denoted as in equation (11).

$$Y = [y_1, y_2, \dots, y_m]^T. \quad (11)$$

- (5) The training dataset S is constructed from m observed pair of XY (input output) as denoted in equation (12)/

$$s = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}, \quad (12)$$

where each pair (x_i, y_i) is known as the training factor obtained from the spectrogram to generate F .

4. Spectrum Monitoring Using SPECTRUMNET

The process of monitoring and PU spectrogram signal classification is performed using the proposed SPECTRUMNET model. The proposed SPECTRUMNET uses a CNN model to extract the unique signal features while improving the accuracy of PU classification. Figure 3 represents the proposed system model's overall flow, which comprises four main stages: data imbalance checking, balancing dataset by oversampling using SMOTE, splitting the balanced dataset and performing classification using SPECTRUMNET, and generating the signal classification output.

4.1. Dataset Description and Imbalance Checking. The spectrogram dataset used for this study consists of customized dataset of reference [27] into two classes, class 0 means LTE absent (or) Wi-Fi present and class 1 means LTE present (or) Wi-Fi absent. The sample images of class 0 and class 1 are shown in Figure 4.

The LTE-8 FDD (frequency division duplex) frequency band consisting of 880–915 MHz uplink and 925–960 MHz downlink is considered for data generation and simulation. Similarly, Wi-Fi 802.11b signals that are using ISM 2.4 GHz band with OFDM (orthogonal frequency-division multiplexing) modulation at a sampling rate of 20 MS/s are generated. The generated signals are configured with signal power and noise power based on power spectral density (PSD) and signal-to-noise ratio (SNR), respectively. A signal power from -40 dBm to -120 dBm and an SNR ranging between -15 dB and 15 dB is used to generate totally 10020 LTE and 8004 Wi-Fi samples. Ideally, there might be an equal number of observations among the classes. However, it is typical for the courses in the training set to be unbalanced with wireless signals [28]. Compared to WiFi signals, LTE transmissions may have a wider bandwidth, but the background noise is still present. An imbalance in the number of observations per class might be harmful to learning since learning is biased in favor of the dominating classes.

The distribution of the total images in the dataset before SMOTE is outlined in Table 1, which makes it abundantly evident that the dataset is class-imbalanced. By replicating the minority class randomly to match the majority class, the

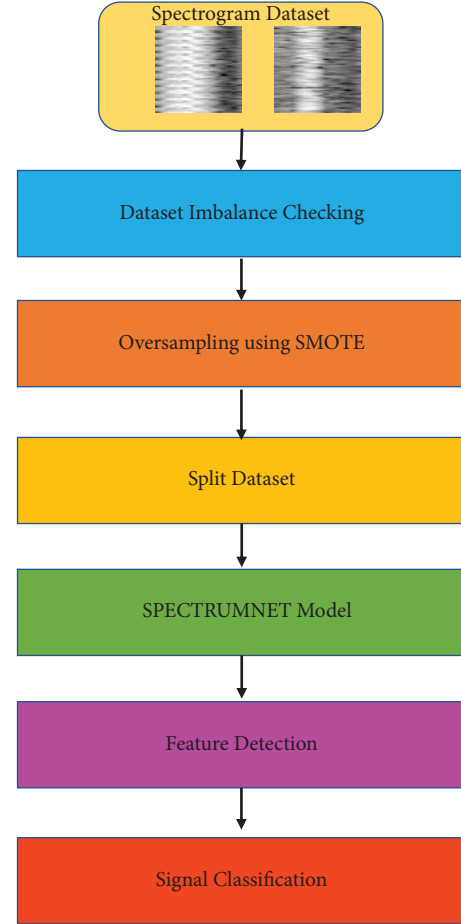


FIGURE 3: Workflow of spectrum monitoring using SPECTRUMNET.

SMOTE approach has been utilized to resolve the issue of the class imbalance problem. Utilizing SMOTE has advantages such as the ability to limit overfitting and decrease knowledge loss. Table 2 displays the dataset distribution after the SMOTE approach was expanded to 20040 samples, with 10020 pictures per class. Then, the dataset is split into 60% training, 20% validation, and the remaining 20% as a testing set, as shown in Table 3. The learning of the optimal parameters is improved by the use of normalized images.

4.2. The SPECTRUMNET Architecture. Our proposed SPECTRUMNET architecture is shown in Figure 5. Here, a spectrogram with a 128×128 image size serves as the network's input. The images are loaded into a CNN once the dataset has been preprocessed and normalized, which extracts discriminating factors to identify the LTE or WiFi signal. In our work, a CNN model created from scratch is used to classify signals. The proposed architecture consists of a convolutional layer with ReLU (rectified linear unit) activation function, a single max-pooling layer, two SPECTRUMNET blocks, two dropout layers, and two dense layers, and one SoftMax classification layer.

The convolutional layer consists of 64×64 filter sizes of 16 filters. Using the pooling layer after each convolution

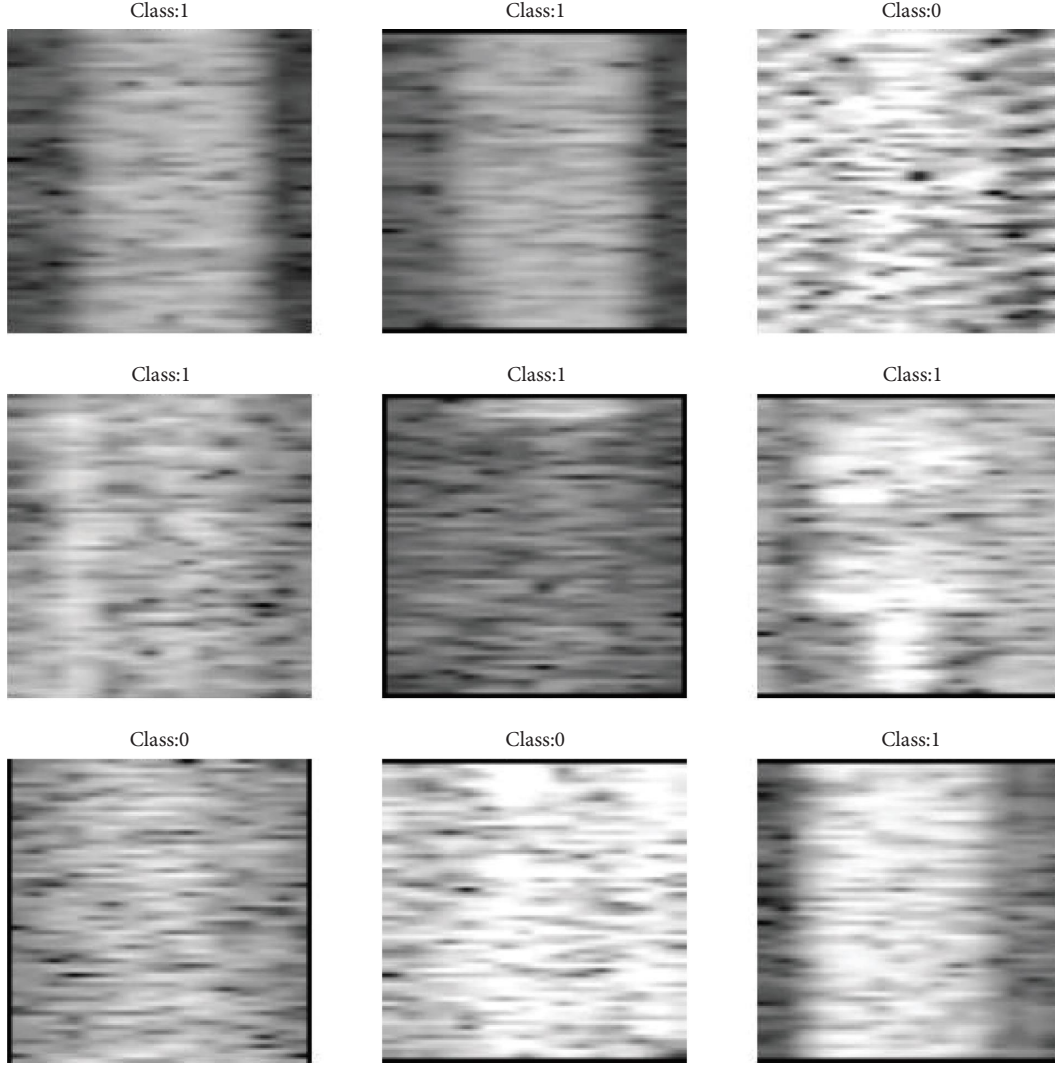


FIGURE 4: Sample images of class 0 and class 1 from the dataset.

TABLE 1: Dataset distribution before SMOTE.

Class type	Before SMOTE
Class 0	8004
Class 1	10020
Total	18024

TABLE 2: Dataset distribution after SMOTE.

Class type	After SMOTE
Class 0	10020
Class 1	10020
Total	20040

reduces the spatial domain, thus minimizing the feature dimensionality by maintaining the spatial features. In our model, max-pooling layers are used as computed by equation (13), where the input is x , the window size is p , and s denotes the stride value.

$$MP = \text{Floor}\left(\frac{x - p}{s} + 1\right). \quad (13)$$

Each SPECTRUMNET block comprises two convolutional layers that are layered on the other, with ReLU activation, batch normalization, and a max-pooling layer. In SPECTRUMNET, the filters 32, 64, and 256 are employed to extract the discriminative features. If the value is positive, the activation function gives the direct output, else it is zero. The activation function manages the vanishing gradient problem, which allows quicker learning and performance of networks than other activation functions. The activation function is given in (3).

$$F = \max(0, x). \quad (14)$$

In the proposed model, a regularization method called batch normalization is employed in the SPECTRUMNET block to overcome the overfitting problem. During training, specific neurons present in the hidden layer are randomly removed using the dropout layer. The optimal dropout levels

TABLE 3: Number of images after splitting the dataset into train, test, and validation sets.

Class	Without SMOTE				With SMOTE			
	Training	Testing	Validation	Total	Training	Testing	Validation	Total
0	5125	1603	1276	8004	6436	1993	1591	10020
1	6410	2002	1608	10020	6389	2015	1616	10020

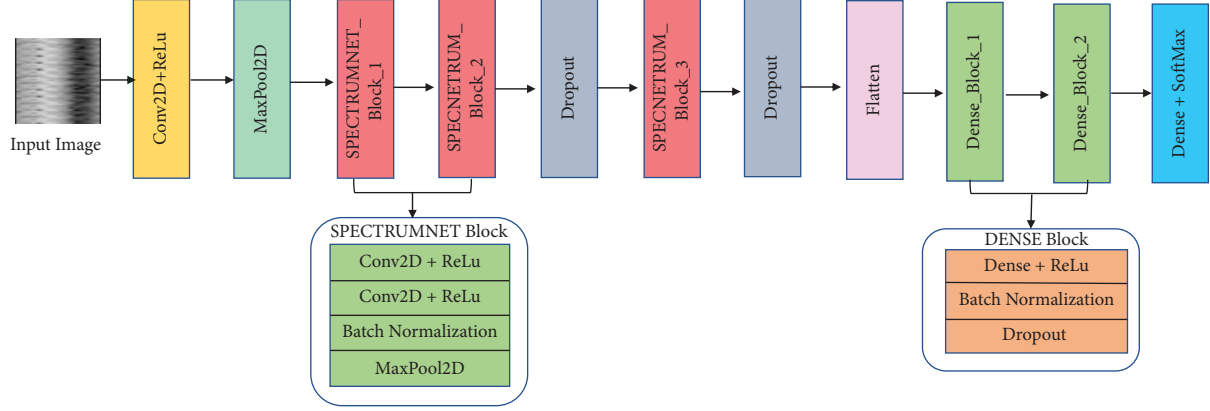


FIGURE 5: Architecture of SPECTRUMNET for signal classification.

of 0.5 are used in the first dense layer and optimal dropout levels of 0.2 are used in the second dense layer.

The flattened layer is used after the convolutional layer in the proposed architecture to perform dimensionality reduction. The dense layer does mathematical operations similar to the artificial neural network. It receives input from the flattened layer and processes it. The proposed method employs two dense layers, with each neuron in the lower part of the layer linked to each neuron in the dense layer. Then, the probability $P(Y_i = \kappa | x; \theta)$ for $\kappa \in \{0, 1\}$ is calculated using SoftMax in which the number of neurons denotes the total output classes, where θ denotes model parameters, x denotes the input spectrogram, and $k=1$ denotes the existence of LTE signals.

Figure 5 illustrates the proposed SPECTRUMNET architecture for classifying the signals with SPECTRUMNET block and DENSE block representation. The proposed model architecture details of the SPECTRUMNET are provided in Table 4.

4.3. Evaluation Metrics. The accuracy, precision, recall, $F1$ score, and AUC for the proposed model are evaluated using the confusion chart. The model outcomes are shown in the confusion chart, using which the metrics for efficiency are evaluated.

4.3.1. Classification Accuracy. The accuracy denotes the model efficiency by predicting the actual positive and negative values as denoted in equation (15).

$$\text{Accuracy} = \frac{TP + TN}{(TP + TN + FP + FN)}. \quad (15)$$

TN, TP, FN, and FP denote true negative, true positive, false negative, and false positive, respectively. The result is

TP if both the predicted and actual labels are regular. The result is TN when the algorithm outputs an abnormal image as abnormal. The output is FP when the model prediction is standard, but the actual label is abnormal, and if it predicts a standard output as abnormal, it is FN.

4.3.2. Precision. The metric precision (PR) denotes the number of valid positive observations over the total optimistic predictions as denoted in equation (16). A good model is one in which the precision is 1.

$$PR = \frac{TP}{(TP + FP)}. \quad (16)$$

4.3.3. Recall. Recall (REC), sometimes referred to as sensitivity, measures how well the classifier can find all positive samples. (14) is used to compute the recall.

$$REC = \frac{TP}{(TP + FN)}. \quad (17)$$

4.3.4. $F1$ Score. The value of the $F1$ score measures how well the recall and precision values are balanced and is computed using equation (15).

$$F1 \text{ score} = \frac{(2 \times PR \times REC)}{(PR + REC)}. \quad (18)$$

4.3.5. Support. The number of real instances of the class in the given dataset is referred to as support. Unbalanced support in the training data may be a sign of structural flaws

TABLE 4: SPECTRUMNET model architecture details.

Layer (type)	Output shape	Parameters
conv2d (Conv2D)	(None, 64, 64, 16)	448
max_pooling2d (MaxPooling2D)	(None, 32, 32, 16)	0
SPECTRUMNET Block_1	(None, 16, 16, 32)	4768
SPECTRUMNET Block_2	(None, 8, 8, 64)	18752
Dropout (dropout)	(None, 8, 8, 64)	0
SPECTRUMNET Block_3	(None, 4, 4, 256)	148736
dropout_1 (dropout)	(None, 4, 4, 256)	0
Flatten (flatten)	(None, 4096)	0
DENSE block	(None, 128)	524928
DENSE block	(None, 64)	8512
SoftMax layer	(None, 2)	130
Total parameters	706,274	
Trainable parameters	705,186	
Nontrainable parameters	1,088	

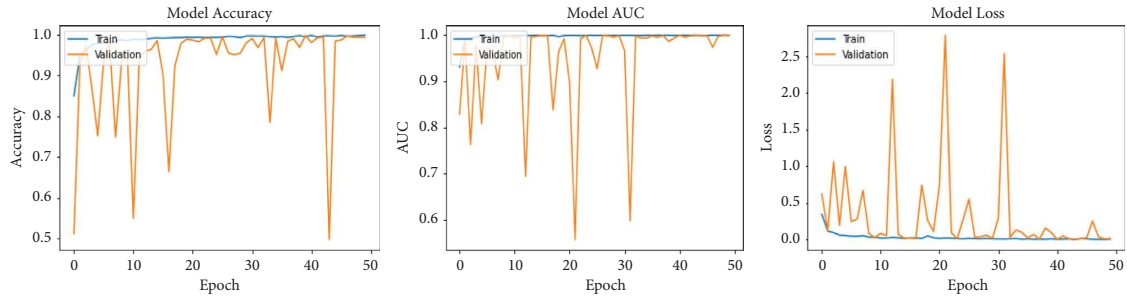


FIGURE 6: Model accuracy, AUC, and loss of training without SMOTE.

in the classifier's reported scores and may point to the need for stratified sampling or rebalancing.

4.3.6. Macro Average. The macro average is computed using the arithmetic mean of all the per-class $F1$ scores. This method treats all classes equally regardless of their support values.

4.3.7. Micro Average. The micro average metric considers a balanced dataset and evaluates the overall performance regardless of the class. Micro averaging computes a global average $F1$ score by counting the sums of the TP, FN, and FP.

4.3.8. Weighted Average. The weighted average function computes $F1$ score for each label and returns the average considering the proportion for each label in the dataset.

4.3.9. Samples Average. The samples average function computes $F1$ score for each instance and returns the average.

4.3.10. Matthews Correlation Coefficient. The Matthews correlation coefficient (MCC), increasing accuracy for the classes of different sizes, was utilized to evaluate the effectiveness of binary class prediction [29]. Equation (16) denotes the Matthews correlation coefficient determined using the confusion matrix.

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN))}}. \quad (19)$$

4.3.11. Optimizer Function. Adam is utilized in this study as the optimizer function to update the network's weights and biases and lower the error function, determined using back-propagation. The AdaGrad and RMS prop optimizer functions are combined to create the Adam optimizers, whose name is imitative of adaptive moment estimation. An extended form of stochastic gradient descent is employed by using the training data to update the network's weights iteratively. For various

parameters, the Adam optimizer has a unique learning rate. The Adam optimizer's Algorithm 1 is outlined below:

The number of iterations has been specified at the start of the algorithm. The gradients are computed in step 1, and the moving averages are calculated exponentially using the moving averages formula in steps 2 and 3 for x and y , respectively. To perform bias correction, the estimator must now be corrected using the \hat{x} and \hat{y} equations. The

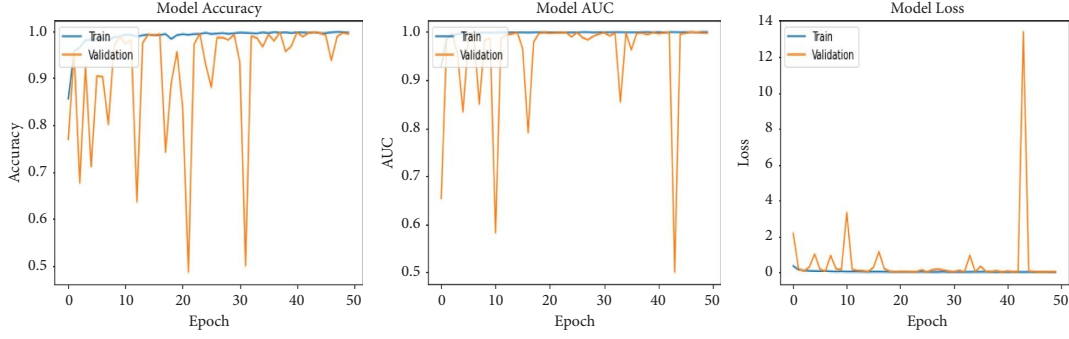


FIGURE 7: Model accuracy, AUC, and loss of training with SMOTE.

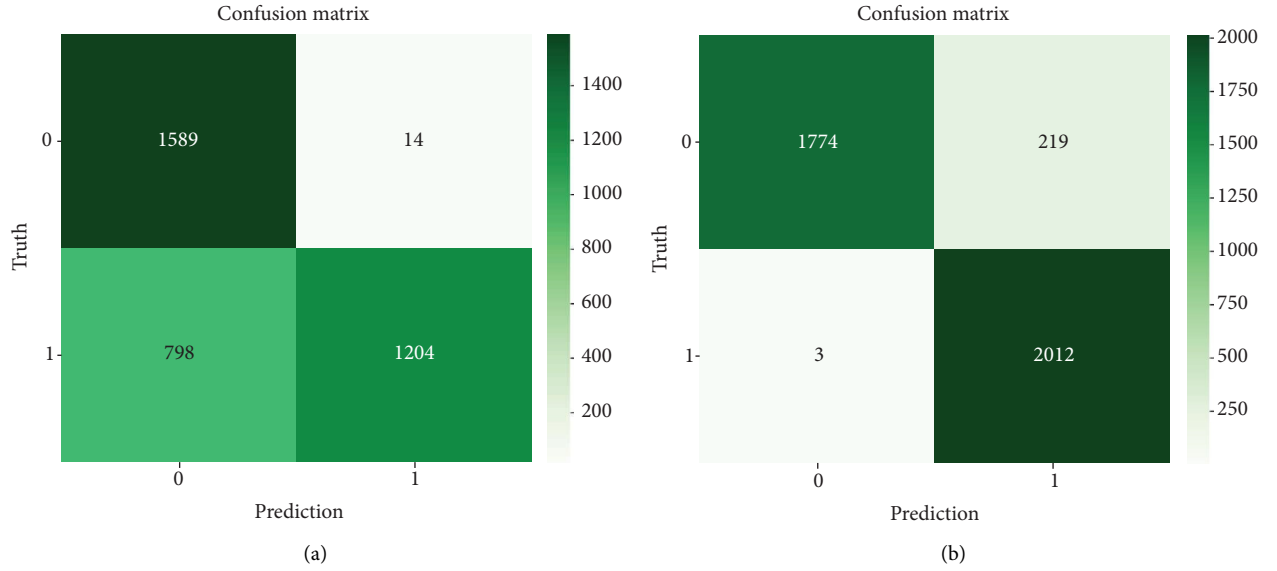


FIGURE 8: Confusion matrix of SPECTRUMNET (a) without SMOTE and (b) with SMOTE.

Adam optimizer algorithms in the last step update the network's weight indicated as z .

5. Experimental Result and Discussion

The proposed SPECTRUMNET model is trained using the Adam optimizer with the hyperparameter of 50 epochs, with a batch size of 16 and 0.001 as initial learning rate. The proposed model has been trained under two scenarios, one without SMOTE and the other with SMOTE. The area under curve (AUC) classification for each epoch is done to check whether the model successfully differentiates between the positive and negative classes.

Figures 6 and 7 illustrate the training curves for model accuracy, AUC, and model loss without and with SMOTE. The value of AUC indicates if the model classifies between positive and negative classes correctly. The model without SMOTE approach achieves an overall training accuracy of 78.09% and 77.67% as validation accuracy, which is not optimal due to the imbalance in the dataset and overfitting. The SMOTE approach results in 95.86% overall training and 95.01% validation accuracy. Once the model is trained, the testing is done with a test dataset. The model is unaware of

the images in the testing dataset since they are not used during the training process. Finally, the confusion matrix is obtained from the model testing.

Figures 8(a) and 8(b) depict the resulting confusion matrix obtained for SPECTRUMNET architecture without and with SMOTE to correctly classify LTE and WiFi signals in order to predict primary users. The confusion matrix indicates how well the model performed on the train images. It is evaluated over the (i) 10020 images from LTE and (ii) 8004 images from WiFi. Figures 9 and 10 depict the resulting ROC and precision-recall curve obtained for the model without and with SMOTE, respectively, which are used to determine the model's effectiveness.

Tables 5 and 6 show the SPECTRUMNET model's experimental classification and individual performance analysis reports. We have evaluated the performance of the model with respect to both imbalanced and balanced datasets without SMOTE and with SMOTE, respectively, as mentioned in Section 4.1. Tables 5 and 6 show the SPECTRUMNET model's experimental classification and individual performance analysis reports, respectively. Thus, the proportion of accuracy alone would be ineffective in assessing model performance. Since all classes in the

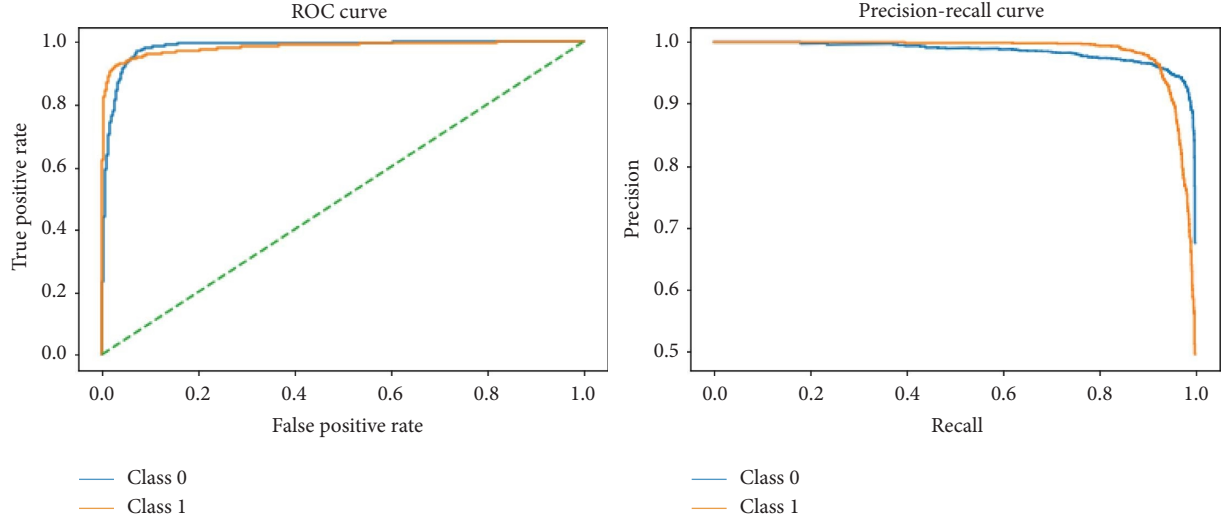


FIGURE 9: ROC curve and precision-recall curve without SMOTE.

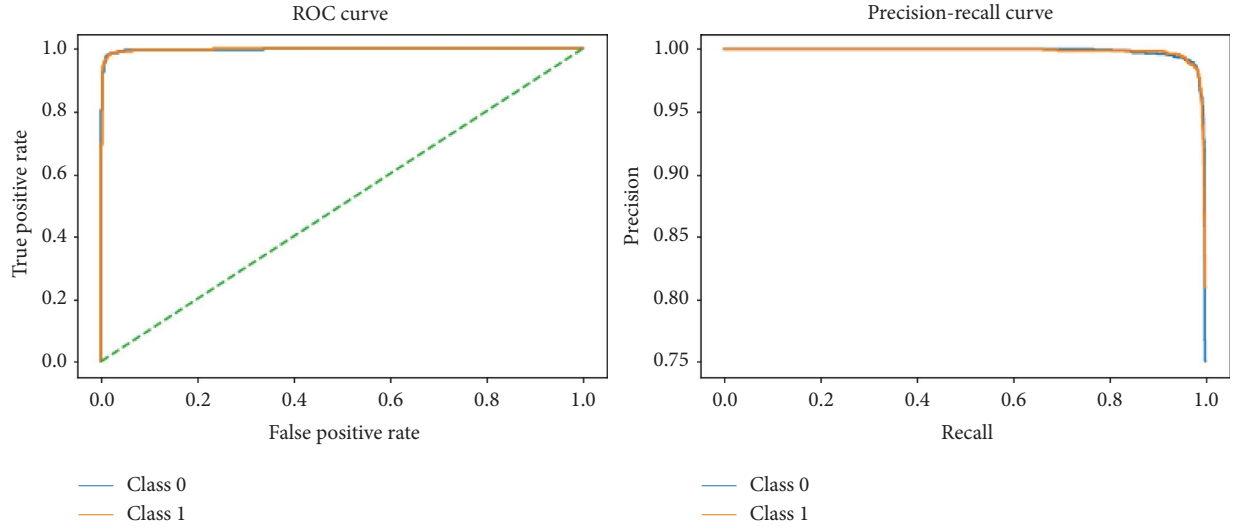


FIGURE 10: ROC curve and precision-recall curve with SMOTE.

```

for each  $t$  in range (num_iterations):
     $w$  = compute gradient ( $a, b$ )
     $x = \text{beta\_1} * x + (1 - \text{beta\_1}) * w$ 
     $y = \text{beta\_2} * y + (1 - \text{beta\_2}) * \text{power}(y, 2)$ 
     $x\_hat = m / (1 - \text{np. power}(\text{beta\_1}, t)) + (1 - \text{beta\_1}) * w / (1 - \text{np. power}(\text{beta\_1}, t))$ 
     $y\_hat = y / (1 - \text{np. power}(\text{beta\_2}, t))$ 
     $z = z - \text{step\_size} * x\_hat / (\text{np. sqrt}(y\_hat) + \text{epsilon})$ 

```

ALGORITHM 1: Adam optimizer algorithm.

spectrogram dataset are equally important, we determine the metrics for each class separately using the averaging methods (macro, micro, weighted, and samples) using the support values with SMOTE and without SMOTE, to describe overall the performance. The result of averaging metrics obtained without SMOTE is only maximum 77%, whereas the value of average metrics using SMOTE is 94%.

Promising results can be seen with the testing dataset's precision, recall, and F1 score for LTE and WiFi classes. During testing, the SPECTRUMNET model obtains a test accuracy of 94.46% with SMOTE and 77.48% without SMOTE. The proposed model achieves an area under the ROC curve of 94.43% with SMOTE and 79.63% without SMOTE.

TABLE 5: Classification report of SPECTRUMNET.

	Without SMOTE				With SMOTE			
	Precision	Recall	F1 score	Support	Precision	Recall	F1 score	Support
0	0.67	0.99	0.8	1903	1	0.89	0.94	1993
1	0.99	0.6	0.75	2002	0.9	1	0.95	2015
Micro avg	0.77	0.77	0.77	3905	0.94	0.94	0.94	4008
Macro avg	0.83	0.8	0.77	3905	0.95	0.94	0.94	4008
Weighted avg	0.84	0.77	0.77	3905	0.95	0.94	0.94	4008
Samples avg	0.77	0.77	0.77	3905	0.94	0.94	0.94	4008

TABLE 6: Performance analysis of SPECTRUMNET.

	Without SMOTE (%)	With SMOTE (%)
Training accuracy	78.09	95.86
Validation accuracy	77.67	95.01
Testing accuracy	77.48	94.46
Balanced accuracy score	79.63	94.43
Matthew's correlation coefficient	62.27	89.44

TABLE 7: Comparison of SPECTRUMNET with other models.

Network	Signal type	Accuracy	Precision	Recall	F1 score
CNN [29]	WiFi	81.53	0.7	1	0.55
	LTE		0.8	0.7	0.6
CM-CNN [30]	WiFi	92.89	0.63	0.85	0.72
	LTE		0.8	1	0.9
CLDNN [31]	WiFi	93.05	0.56	0.8	0.6
	LTE		0.33	1	0.5
SPECTRUMNET (without smote)	WiFi	77.48%	0.67	0.99	0.8
	LTE		0.99	0.6	0.75
SPECTRUMNET (with smote)	WiFi	94.46%	1	0.89	0.94
	LTE		0.9	1	0.95

Experimental results show that the SPECTRUMNET model classifies two classes with a minimum of 705,186 trainable parameters and performs better with the accuracy, precision, recall, and F1 score compared to all the other models. Table 7 outlines the results of the model, where it is compared against various CNN models from the literature, such as CNN [29], CM-CNN [21], and CLDNN [31]. On the provided dataset, the SPECTRUMNET model performs well. The model agrees perfectly with SMOTE, as evidenced by Matthew's correlation coefficient of 89.44% with SMOTE and 62.27% without SMOTE.

6. Conclusion

The deep neural network was used to study and implement a monitoring radio environment for performing spectrum detection to conduct spectrum handover. The suggested SPECTRUMNET deep learning algorithm model was used to provide image feature extraction using cyclostationary signal-based cooperative spectrum sensing. A novel CNN architecture model is proposed in this research to classify spectrum signals. A customized RF signal dataset with severe class imbalance is used to build the classification model, and the SMOTE technique has been applied to overcome this

problem. Compared to existing CNN techniques, the proposed model attained an overall accuracy of 94.46% with 96% AUC when tested using testing data consisting of two classes. As a result, it can identify the signal at a low SNR of -15 dB while predicting spectrum holes with great accuracy. In the future, the SPECTRUMNET model will be implemented using an FPGA board for real-time deployment, focusing on terrestrial applications as a standalone framework for monitoring spectrum and improving spectrum handover.

Data Availability

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

Disclosure

This research was performed as a part of author's research work for the publication.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] F. CC. Docket no.03-322, "Notice of proposed rulemaking and order," vol. 63, 2003.
- [2] J. Mitola, "Cognitive radio for flexible mobile multimedia communications," in *IEEE International Workshop on Mobile Multimedia Communications*, vol. 27, pp. 3–10, IEEE, 1999.
- [3] I. F. Akyildiz, W.-y. Lee, M. C. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 40–48, April 2008.
- [4] A. Haldorai and U. Kandaswamy, "Cooperative spectrum handovers in cognitive radio networks," in *Intelligent Spectrum Handovers in Cognitive Radio Networks*, pp. 1–18, Springer, Cham, 2019.
- [5] K. B. Letaief and W. Zhang, "Cooperative spectrum sensing," in *Cognitive Wireless Communication Networks*, pp. 115–138, Springer, Boston, MA, USA, 2007.
- [6] F. A. Bhatti, M. J. Khan, A. Selim, and F. Paisana, "Shared spectrum monitoring using deep learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 4, pp. 1171–1185, 2021.
- [7] "Weblink," <https://www.fcc.gov/engineering-technology/policy-and-rules-division/general/radio-spectrum-allocation>.
- [8] P. Jagadamba, "Signal to Noise Ratio(SNR) improvement of atmospheric signals using Variable windows," *Signal & Image Processing International Journal*, vol. 3, no. 5, pp. 91–109, 2012.
- [9] E. Axel, G. Leus, E. G. Larsson, and H. V. Poor, "Spectrum sensing for cognitive radiostate-of-the-art and recent advances," *IEEE Signal Processing Magazine*, vol. 29, no. 3, pp. 101–116, 2012.
- [10] K. Kim, I. A. Akbar, K. K. Bae, J. S. Um, C. M. Spooner, and J. H. Reed, "Cyclostationary approaches to signal detection and classification in cognitive radio," in *Proceedings of the 2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pp. 212–215, IEEE, Dublin, Ireland, 2007, April.
- [11] H. Li, "Cyclostationary feature-based quickest spectrum sensing in cognitive radio systems," in *Proceedings of the 2010 IEEE 72nd Vehicular Technology Conference-Fall*, pp. 1–5, IEEE, Ottawa, ON, Canada, 2010, September.
- [12] P. D. Sutton, B. Özgül, and L. E. Doyle, "Cyclostationary signatures for LTE advanced and beyond," *Physical Communication*, vol. 10, pp. 179–189, 2014.
- [13] C. Du, H. Zeng, W. Lou, and Y. T. Hou, "On cyclostationary analysis of WiFi signals for direction estimation," in *Proceedings of the 2015 IEEE International Conference on Communications (ICC)*, pp. 3557–3561, IEEE, London, UK, 2015, June.
- [14] D. Han, G. C. Sobabe, C. Zhang et al., "Spectrum sensing for cognitive radio based on convolution neural network," in *Proceedings of the 2017 10th International congress on Image and Signal Processing, Biomedical Engineering, and Informatics (CISP-BMEI)*, pp. 1–6, IEEE, Shanghai, China, 2017, October.
- [15] W. Lee, M. Kim, and D. H. Cho, "Deep cooperative sensingcooperative spectrum sensing based on convolutional neural networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 3005–3009, 2019.
- [16] J. Xie, J. Fang, C. Liu, and X. Li, "Deep learning-based spectrum sensing in cognitive radioa CNN-LSTM approach," *IEEE Communications Letters*, vol. 24, no. 10, pp. 2196–2200, 2020.
- [17] Q. Cheng, Z. Shi, D. N. Nguyen, and E. Dutkiewicz, "Sensing OFDM signals a deep learning approach," *IEEE Transactions on Communications*, vol. 67, no. 11, pp. 7785–7798, 2019.
- [18] H. He and H. Jiang, "Deep learning-based energy efficiency optimization for distributed cooperative spectrum sensing," *IEEE Wireless Communications*, vol. 26, no. 3, pp. 32–39, 2019.
- [19] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018.
- [20] S. Zheng, S. Chen, P. Qi, H. Zhou, and X. Yang, "Spectrum sensing based on deep learning classification for cognitive radios," *China Communications*, vol. 17, no. 2, pp. 138–148, 2020.
- [21] W. M. Lees, A. Wunderlich, P. J. Jeavons, P. D. Hale, and M. R. Souryal, "Deep learning classification of 3.5-GHz band spectrograms with applications to spectrum sensing," *IEEE transactions on cognitive communications and networking*, vol. 5, no. 2, pp. 224–236, 2019.
- [22] O. Omotere, J. Fuller, L. Qian, and Z. Han, "Spectrum occupancy prediction in coexisting wireless systems using deep learning," in *Proceedings of the 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pp. 1–7, IEEE, Chicago, IL, USA, 2018, August.
- [23] A. Wilfred and O. Or, "A review of cyclostationary feature detection-based spectrum sensing technique in cognitive radio networks," *E3 Journal of Scientific Research*, vol. 4, no. 3, pp. 041–047, 2016.
- [24] D. Teguig, B. Scheers, and V. Le Nir, "Data fusion schemes for cooperative spectrum sensing in cognitive radio networks," in *Proceedings of the 2012 Military Communications and Information Systems Conference (MCC)*, pp. 1–7, IEEE, Gdansk, Poland, 2012, October.
- [25] K. Tekbyk, Ö. Akbunar, A. R. Ekti, A. Görçin, G. K. Kurt, and K. A. Qaraqe, "Spectrum sensing and signal identification with deep learning based on spectral correlation function," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10514–10527, 2021.
- [26] S. M., "Machine learning and quantum computing for 5G/6G communication networks - a survey," *International Journal of Intelligent Networks*, vol. 3, pp. 197–203, 2022.
- [27] F. Paisana, "Dataset for GLOBECOM Paper "Spectrum Monitoring for Radar Bands Using Deep Convolutional Neural Networks," Harvard Dataverse, 2018.
- [28] L. Li, H. Jiang, and H. He, "Imbalanced learning for cooperative spectrum sensing in cognitive radio networks," in *Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, IEEE, Waikoloa, HI, USA, 2019, December.
- [29] C. Liu, J. Wang, X. Liu, and Y. C. Liang, "Deep CM-CNN for spectrum sensing in cognitive radio," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2306–2321, 2019.
- [30] K. Tekbiyik, A. R. Ekti, A. Görçin, G. K. Kurt, and C. Keçeci, "Robust and fast automatic modulation classification with CNN under multipath fading channels," in *Proceedings of the 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pp. 1–6, IEEE, Antwerp, Belgium, 2020, May.
- [31] S. Ramjee, S. Ju, D. Yang, X. Liu, A. E. Gamal, and Y. C. Eldar, "Fast Deep Learning for Automatic Modulation Classification," Article ID 05850, 2019.