DEVELOPING NOVEL SPECTRUM OCCUPANCY PREDICTION AND PHYSICAL LAYER SECURITY TECHNIQUES

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF

ENGINEERING AND NATURAL SCIENCES

OF ISTANBUL MEDIPOL UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL, ELECTRONICS ENGINEERING AND CYBER SYSTEMS

By

Mehmet Ali Aygül

December, 2020

DEVELOPING NOVEL SPECTRUM OCCUPANCY PREDICTION AND PHYSICAL LAYER SECURITY TECHNIQUES
By Mehmet Ali Aygül
December, 2020

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Hüseyin Arslan (Advisor)

Prof. Dr. Hasan Fehmi Ateş

Asst. Prof. Dr. Ali Görçin

Approved by the Graduate School of Engineering and Natural Sciences:

Assoc. Prof. Dr. Yasemin Yüksel Durmaz
Director of the Graduate School of Engineering and Natural Sciences

Foreword

This thesis is written to finalize my master's degree in Electrical, Electronics Engineering and Cyber Systems department at Istanbul Medipol University. In this thesis, the increasing demand for the usage of wireless communication systems and some of the problems it brings are covered. These problems are spectrum scarcity and security. Besides that, novel techniques that employ machine learning for spectrum occupancy prediction and attack identification are proposed.

This thesis is achieved with the guidance and support of my advisor Prof. Dr. Hüseyin Arslan. I wish also to thank Prof. Dr. Hasan Fehmi Ateş and Assist. Prof. Dr. Ali Görçin for their valuable suggestions. I would like to thank Dr. Mahmoud Nazzal for his advice and for being supportive in every way. Finally, I would like to express my sincere thanks to my colleagues and my family for their love and encouragement.

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: MEHMET ALI AYGÜL

Signature :

Acknowledgement

I would like to express my deep and sincere gratitude to my advisor Prof. Dr. Hüseyin Arslan. I appreciate his guidance also, the motivation and support he provided me throughout my study. His immense knowledge shed light on my researches and vision. I also wish to thank Prof. Dr. Hasan Fehmi Ateş, Dr. Ali Görçin, Prof. Dr. Bahadır Kürşat Güntürk, and Dr. Ali Rıza Ekti. I highly appreciate that they agreed being on my committee and I am thankful for their invaluable suggestions, intellectual comments for my development, support and patience during my studies. I hope I will have the opportunity to benefit from their deep and unique knowledge, wide experience; and to also have the chance to contribute various works with them in the future as well.

I would also like to thank my lab-mates both in the CoSiNC group at IMU in Turkey and WCSP group at USF in the USA. Their productive, stimulating and fruitful discussions helped me in my studies. I would also like to express my special thanks to Dr. Mahmoud Nazzal, who has been supportive in every way. His guidance and support have always encouraged me to work harder. I will be ever grateful and be hopeful for future collaborations.

Last but not least, I would like to thank my family for their love and guidance in every stage of my life. I am thankful that they always encourage me to follow my dreams. I would also like to thank my sister for her limitless love and positive attitude while she kept believing in my success. My sister always gives me great support emotionally which I believe makes me stronger and more successful. I also wish to thank my dearest friend Zemzem Selin Oruç. I am thankful for her unlimited support and love in my life. I am thankful for the inspiration and constant trust she gave me in this long and challenging journey as she always supports and inspires me in every part of my life.

Contents

1 Introduction 1

1.1 Motivation . . . 1

1.2 Dissertation outline . . . 3

1.3 Publications . . . 4

2 Efficient Spectrum Occupancy Prediction Exploiting Multidimensional Correlations through Composite 2D-LSTM Models 5

2.1 System Model and Preliminaries . . . 8

2.1.1 Prediction with Autoregressive Model . . . 9

2.1.2 Prediction with Bayesian-Inference . . . 10

2.1.3 Prediction with Long Short-Term Memory . . . 10

2.1.4 Prediction with Convolutional Long Short-Term Memory . 11

2.2 The Proposed Technique for Spectrum Occupancy Prediction Exploiting Time, Frequency, and Space Correlations . . . 12

2.2.1 Motivations for Time, Frequency, and Space Correlation Exploitation and Problem Sub-Division 12

2.2.2 The Proposed Technique 14

2.2.3 A Note on Computational Complexity 16

2.3 Dataset Generation 18

2.3.1 Measurement Setup 18

2.3.2 Measurement Procedure and Geographical Locations . . . 19

2.4 Parameter Settings and Experimental Results 21

2.4.1 Hyperparameters of Deep Learning Models 22

2.4.2 Performance Evaluation and Discussion 25

2.5 Conclusions 27

3 Signal Relation-Based Physical Layer Authentication 29

3.1 System Model and Useful Information 31

3.1.1 Channel Model 32

3.1.2 Existing Channel-Based PLA 33

3.1.3 LS-Based Channel Estimation 34

3.2 The Proposed Technique 34

3.2.1 MMSE Signal Relation-Based Technique 35

3.2.2 LSTM Signal Relation-Based Technique 36

3.2.3 Hypothesis Test 36

3.3 Simulation Results and Discussions 37

3.3.1 Parameter Settings 37

3.3.2 Performance Evaluation and Discussions 39

3.4 Conclusions 43

4 Deep Learning-Assisted Detection of PUE and Jamming Attacks in Cognitive Radio Systems 44

4.1 Introduction 44

4.2 System Model and Preliminaries 46

4.3 The Proposed Technique for PUEA and JA Detection 49

4.3.1 The Proposed Technique 49

4.3.2 Computational Complexity Discussion 50

4.4 Simulations and Results 51

4.4.1 Parameter Setting 51

4.4.2 Performance Evaluations and Discussions 53

4.5 Conclusions 55

5 Future Directions 57

List of Figures

2.1 Spectrum occupancy prediction with (a) 1D data, (b) 2D data, (c) 3D data. 7

2.2 Spectrum subband (SB) occupancy modeling. 9

2.3 An example spectrum's (a) occupancy distributions and (b) correlation. 13

2.4 An illustrative motivation for space correlation. 14

2.5 The proposed technique for spectrum occupancy prediction-training stage. 16

2.6 The proposed technique for spectrum occupancy prediction-testing stage. 17

2.7 Picture of the environment for Taksim. 20

2.8 Satellite capture of the environment for Taksim. 20

2.9 Picture of the environment for Silivri. 21

2.10 Satellite capture of the environment for Silivri. 22

2.11 Composite 2D-LSTMs in spectrum occupancy prediction. 23

2.12 (a) Loss and (b) accuracy graphs for 2D-LSTM. 27

3.1 An illustration of a PLA system model that includes Alice (legitimate transmitter), Bob (receiver) and several spoofers (extremely intelligent illegitimate transmitters). 31

3.2 Illustration of signal relation-based PLA using MMSE and LSTM for (a) training stage, (b) testing stage. 35

3.3 The overall block diagram of the proposed LSTM model with an input layer, three LSTM layers, a fully connected layer, and an output layer. 38

3.4 The detection performance of legitimate user when exact L is known (L=J); (a) $N_p = 40$, (b) $N_p = 50$, (c) $N_p = 60$ pilot test symbols. 40

3.5 The detection performance of legitimate user ($N_p = 50$) when channel modeled as (a) undermodel ($L > J$), (b) overmodel ($L < J$). 41

3.6 The detection performance of legitimate user ($N_p = 50$) for different number of training pilot symbols (N) when exact L is known ($L = J$). 42

4.1 A visual illustration of a PU, attacker (PUE or jammer), and noise. 47

4.2 An illustration of the proposed technique's (a) training stage, (b) testing stage. 50

4.3 The proposed 1D-CNN model. 53

4.4 Comparison of the ED, CS, and DL-based techniques in PUEA detection. 54

4.5 Comparison of the ED, CS, and DL-based techniques in JA detection. 54

4.6 (a) Loss and (b) accuracy graphs for 1D-CNN. 56

List of Tables

2.1 Training and testing execution time (seconds) comparison of the tensor and 2D-based technique. 14

2.2 The number of handover attempts between main BS and neighboring BSs in one week. 19

2.3 Details of the signals used for the datasets. 22

2.4 The *precision* (π), *recall* (ψ), and F_1-score spectrum occupancy performances of ARM, BIF, 1D-LSTM, 2D-LSTM, ConvLSTM, and composite 2D-LSTMs (the proposed technique) for Taksim scenario. 26

2.5 The *precision* (π), *recall* (ψ), and F_1-score spectrum occupancy performances of ARM, BIF, 1D-LSTM, 2D-LSTM, ConvLSTM, and composite 2D-LSTMs (the proposed technique) for Silivri scenario. 26

2.6 Complexity analyses of ConvLSTM and the proposed technique for Taksim scenario. 26

2.7 Complexity analyses of ConvLSTM and the proposed technique for Silivri scenario. 27

4.1 AUROC values for PUEA detection by the proposed technique. . 55

4.2 AUROC values for JA detection by the proposed technique. . . . 55

ÖZET

YENİLİKÇİ SPEKTRUM DOLULUK TAHMİNİ VE FİZİKSEL KATMAN GÜVENLİK TEKNİKLERİ GELİŞTİRME

Mehmet Ali Aygül
Elektrik-Elektronik Mühendisliği ve Siber Sistemler, Yüksek Lisans
Tez Danışmanı: Prof. Dr. Hüseyin Arslan
Aralık, 2020

Kablosuz iletişim sistemleri, finansal işlemlerden sağlık kayıtlarına, eğlenceden işe, eğitimden seyahate hayatımızın her alanında yaygın olarak yer almaya başladı. Bu yaygınlaşma, spektrum kıtlığı ve güvenlik dahil olmak üzere birçok zorluğu da beraberinde getirdi. Son yıllarda, bu sorunları ele almak için makine öğrenmesi teknikleri kullanılmıştır. Ancak, bu tekniklerin çoğu ya iyi doğruluk sağlamakta ya da ölçeklenebilir bir karmaşıklık düzeyini korumakta başarısız olmaktadır.

Bu tezde, sırasıyla spektrum doluluk tahmini ve saldırı tanımlama için teknikler önerilmektedir. Spektrum doluluk tahmini, Türkiye'nin en büyük telekom operatörlerinden biri tarafından sağlanan bir veri seti ile yapılan simülasyonlarla doğrulanmıştır. Sonuçlar, önerilen tekniğin karmaşıklığını düşük tutarken spektrum doluluğunu yüksek doğrulukla tahmin edebileceğini göstermektedir. Benzer şekilde, aldatma, karıştırma ve birincil kullanıcı benzetim saldırısı gibi çeşitli saldırıların belirlenmesi için yüksek doğruluğa ve düşük karmaşıklığa sahip teknikler önerilmiş ve performansları değerlendirilmiştir.

Anahtar sözcükler: 5G ve ötesi, karıştırma saldırısı, birincil kullanıcı benzetim saldırısı, makine öğrenmesi, fiziksel katman güvenliği, spektrum doluluk tahmini, kablosuz haberleşme sistemleri.

ABSTRACT

DEVELOPING NOVEL SPECTRUM OCCUPANCY PREDICTION AND PHYSICAL LAYER SECURITY TECHNIQUES

Mehmet Ali Aygül
M.S. in Electrical, Electronics Engineering and Cyber Systems
Advisor: Prof. Dr. Hüseyin Arslan
December, 2020

Wireless communication systems have pervaded every aspect of our lives; from financial transactions to health records, from entertainment to work, from education to travelling. This extensive usage of wireless devices is accompanied by certain challenges including, but not limited to, spectrum scarcity and security. In recent years, machine learning techniques have been employed to address both problems. However, the majority of these techniques fail to either provide good accuracy or keep a scalable complexity level.

In this dissertation, techniques are proposed for spectrum occupancy prediction and attack identification, respectively. The former is validated via simulations on a data set provided by one of Turkey's largest telecom operators. The results show that the proposed technique can predict spectrum occupancy with high accuracy while keeping the complexity low. Similarly, accurate and low complexity techniques for the identification of various attacks such as spoofing, jamming, and primary user emulation are proposed, and their performance is evaluated.

Keywords: 5G and beyond, jamming attack, primary user emulation attack, machine learning, physical layer security, spectrum occupancy prediction, wireless communications systems.

Chapter 1

Introduction

1.1 Motivation

Wireless communication systems are not only limited to communication nowadays but are also used in many areas of daily life. However, accommodating exploding data traffic is among one of the biggest challenges for future wireless communication systems. In fifth-generation (5G) and beyond systems, data rates will be multiplied by ten compared to fourth-generation [1], and latency will go down to one millisecond or less [2]. Therefore, the ever-increasing demanding nature of higher rate communications causes an inherent gap with the scarcity of the available spectrum [3]. Cognitive radio (CR) is believed to be one of the key solutions to bridge this gap [4].

In CR systems, identifying spectrum opportunities is fundamental to efficiently use the spectrum. Spectrum occupancy prediction is a convenient way of revealing opportunities based on previous occupancies. Studies have demonstrated that usage of the spectrum has a high correlation over multidimensions which includes time, frequency, and space. Accordingly, recent literature uses tensor-based techniques to exploit the multidimensional spectrum correlation. However, these techniques share two main drawbacks. First, they are computationally complex.

Second, they need to re-train the overall model when no information is received from any base station for any reason.

Security threats are also another major challenge of 5G and beyond systems. Because of the broadcast feature of the wireless medium, an illegitimate node may try to hear the communication between legitimate parties (eavesdropping), change the message transmitted between two legitimate users (spoofing), and/or intentionally attempt to disrupt the communication between legitimate users (jamming). Therefore, security arises as a critical issue for next-generation wireless networks [5].

Spoofing attacks are one of the serious threats to communication security [6]. Most physical layer authentication techniques use channel information to prevent spoofing attacks. In such techniques, one must estimate the channel information for each authentication procedure. However, when the number of pilots decreases, authentication accuracy also decreases due to low channel estimation quality.

In addition to the aforementioned security threats, CR is naturally vulnerable to many security threats. Examples include eavesdropping, jamming attack (JA), and primary user emulation attack (PUEA) [7]. Amongst these, PUEAs and JAs are the most critical as they can prevent the exploitation of the spectrum by causing false alarms about spectrum occupancy. More specifically, a primary user emulator (PUE) can emulate the transmission characteristics similar to a (legitimate) PU while a jammer can generate intentional interference. In both cases, the effects of attacks lead to a faulty conclusion about the spectrum occupancy. Machine learning has been recently applied to the detection of these attacks. Still, the need for feature extraction required by machine learning techniques restrains the full exploitation of raw data.

1.2 Dissertation outline

In order to provide solutions for the aforementioned problems, the following contributions are done for each chapter.

- Chapter 2 proposes a technique to reduce the aforementioned drawbacks of the tensor-based techniques. The proposed technique uses composite two-dimensional (2D)-long short-term memory (LSTM) models. Extensive experimental results reveal a high detection performance with more robustness and less complexity attained by the proposed technique. The real-world measurements provided by one of the leading mobile network operators in Turkey validate these results.
- Chapter 3 proposes a novel signal relation-based authentication technique that relies on the detection of received signal symbols and does not require the estimation of channel information in the testing stage. It is noteworthy that the authentication performance of the proposed scheme remains at a good level. We develop two different solutions for the detection of received signal symbols, namely, minimum mean-square error and long short-term memory. Extensive simulation results show the main insights of the proposed signal relation-based authentication technique compared to the conventional channel-based authentication technique.
- Chapter 4 proposes one-dimensional deep learning as a framework for identifying PUEAs and JAs. Simulations show the ability of the proposed technique to detect these attacks with high performance.
- Future directions for spectrum occupancy prediction and physical layer security are given in Chapter 5.

1.3 Publications

The publications that are used along with the establishment of the chapters in this dissertation are itemized below.

- Chapter 2 is written premised on the publications in [8, 9].
- Chapter 3 is written premised on the publication in [10].
- Chapter 4 is written premised on the publications in [11, 12].

Chapter 2

Efficient Spectrum Occupancy Prediction Exploiting Multidimensional Correlations through Composite 2D-LSTM Models

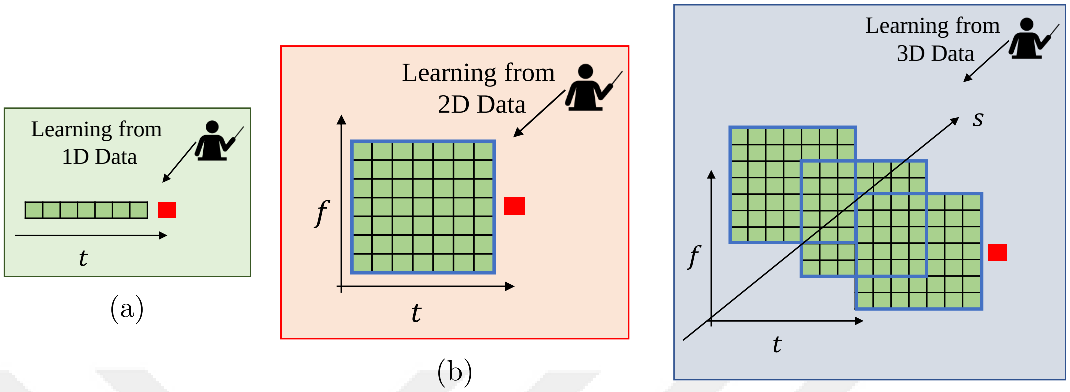
CR enables secondary users (SUs) to opportunistically use available spectrum bands (referred to as *spectrum holes*) unused by primary users (PUs) [13]. It is evident that this needs identifying spectrum usage states, in a process referred to as spectrum sensing. However, spectrum sensing requires continuous sensing. To eliminate the need for continuous spectrum sensing, spectrum occupancy prediction is used [14–17] where future occupancies are predicted based on previous occupancies. The spectrum occupancy prediction substantially saves the time, energy, and computation overheads required by continuous spectrum sensing [18].

There are various spectrum occupancy prediction techniques [19]. Early works used classical statistical prediction techniques to predict holes. Examples include predicting holes using an exponential moving average model-based technique [20], an autoregressive model (ARM)-based technique [21], and a Bayesian inference (BIF)-based technique [22]. More recently, machine learning (ML)-based techniques have been preferred for this problem such as shallow artificial neural networks [23] and wavelet neural networks [24].

Recent literature views that spectrum occupancy is a non-stationary process [25]. However, the aforementioned techniques may not always be capable of addressing this issue. This incapability has more strongly emerged with the diverse user types and increased user mobility envisaged in 5G and beyond. In view of these challenges, the non-stationary hidden Markov technique exploited the time-varying feature of PU behaviors [26] and deep learning (DL) techniques have been proposed as an advanced spectrum prediction framework for addressing this non-stationarity.

Advanced DL techniques, namely, convolutional neural networks (CNN) and long short-term memory (LSTM) are promising to exploit correlation with long lags. Accordingly, the phase and amplitude difference of data was used to train CNN classifiers for detecting the presence of radar signals with high accuracy [27]. Spectral and temporal correlations were used with LSTM models [28]. Moreover, the spectrum in a frequency hopping communication was predicted by an LSTM network [29]. This work was subsequently extended using the Taguchi approach [18]. Furthermore, deep neural networks, LSTM, and CNN-based models were designed. Then, their capabilities in spectrum occupancy prediction were compared [30]. In [8], time and frequency correlations were exploited to predict spectrum occupancy over real-world measurements.

Although the aforementioned approaches have been useful in the analysis of numerous cases, they consider correlations only in time, space, and/or frequency domain. However, these dimensions do not provide a detailed analysis of the non-stationary characteristics and multidimensional attributes of the wireless signals [31]. Jointly exploiting multidimensional (time, frequency, and space) correlations provides a promising perspective for spectrum prediction, as illustrated in Fig. 2.1. Recently, tensor analysis has been adopted as a framework to utilize multidimensional correlations for spectrum prediction. Along this line, [32] converted



(a)

(b)

(c)

Figure 2.1: Spectrum occupancy prediction with (a) 1D data, (b) 2D data, (c) 3D data.

spectrum prediction into a third-order tensor completion problem. This approach achieved one-day-long predictions with a reasonable error margin. Another work considers merging CANDECOMP/PARAFAC tensor decomposition with LSTM for prediction [33]. Besides, multidimensional correlations were utilized jointly with convolutional long short-term memory (ConvLSTM) for a long-term temporal prediction [34].

Although tensor models provide a powerful and rich representation of a three-dimensional (3D) dataset, they share two common drawbacks. First is their high processing time [35, 36]. Second, is that they assume 3D data can be provided at any time. However, sometimes, it is difficult to get information from all of the base stations (BS)s. For example, in the case of CR security threats (primary user emulation attack and jamming attack [7]), accurate information about spectrum occupancy cannot be provided from BSs. Also, in the case of a natural disaster, the information flow of some BSs can be cut. Therefore, such an assumption is not always valid or realistic.

To compensate for the aforementioned effects, more adaptive and flexible techniques should be developed. In this regard, step-based techniques [37], which divide the problem into smaller sub-problems, can be used. By the virtue of these techniques, the complexity load in one model can be divided at the expense of temporal globality [38]. An attractive advantage of this setting is that it is not required to re-train the whole model once any element (BS) in the model fails to provide its information. Alternatively, the BS whose information is missing temporarily is removed from the model, and only a simple end-classifier, which will give the last decision about spectrum occupancy is re-trained.

In this chapter, we exploit multidimensional correlations and propose the usage of composite 2D-LSTM models to divide the 3D spectrum occupancy prediction problem into smaller sub-problems. Extensive experimental results demonstrate that the proposed technique can predict spectrum occupancies with less complexity and small performance loss compared to tensor-based techniques. Besides, the performance of the proposed technique is superior to the one-dimensional (1D) and 2D-based techniques. Also, the proposed technique does not require complete re-training with the absence of data from any BS. These results are validated over real-world spectrum measurements provided by one of the leading mobile network operators in Turkey. These measurements are made in two scenarios; city center and village, to reflect different user density scenarios. The *precision*, *recall*, and F_1-score performance metrics are used in this validation. Moreover, the training and testing execution time is used as a computational complexity metric.

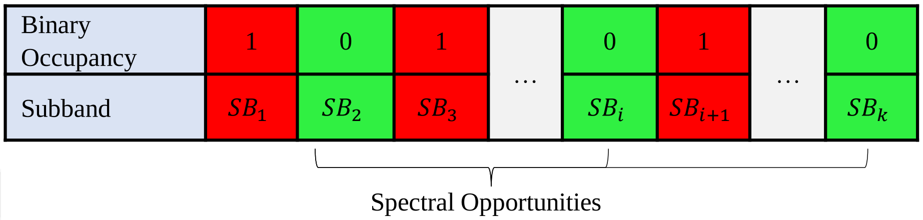
Organization: This chapter is organized as follows. Section 2.1 presents the system model and preliminaries. Section 2.2 details the proposed technique. The data generation and experiments conducted to evaluate the performance of the proposed technique are presented in Section 2.3 and Section 2.4, respectively. Finally, the chapter is concluded in Section 2.5.

2.1 System Model and Preliminaries

This chapter aims at predicting spectrum occupancy states over a given frequency range depending on the previous occupancies. For modeling the spectrum access, the heterogeneous spectrum access model [39] is adopted. This model is demonstrated in Fig. 2.2. More specifically, the spectrum is split into k contiguous frequency subbands. In this model, the absence (presence) of a PU means signifies a hole (occupied spectrum). The following hypotheses ($\mathcal{H}_0$ and $\mathcal{H}_1$) formally states these cases.

$$\boldsymbol{y} = \begin{cases} \boldsymbol{n}, & \mathcal{H}_0 : \text{there is no PU} \\ \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}, & \mathcal{H}_1 : \text{a PU is present}, \end{cases} \tag{2.1}$$

where $\boldsymbol{x}$ denotes PU transmitted signal, $\boldsymbol{n}$ represent the additive white Gaussian noise, channel matrix is represented by $\boldsymbol{H}$ and $\boldsymbol{y}$ denotes received signal. Besides, the works include state of the art techniques used for spectrum occupancy are briefly explained below.



Binary Occupancy	1	0	1	…	0	1	…	0
Subband	SB_1	SB_2	SB_3		SB_i	SB_{i+1}		SB_k

Figure 2.2: Spectrum subband (SB) occupancy modeling.

2.1.1 Prediction with Autoregressive Model

The ARM [21] is a linear predictor and it works as follows

$$\hat{s}_t = \sum_{i=1}^{r} \varphi_i y_{t-i} + \omega_t, \tag{2.2}$$

where r represent the model order, φ_i, $i = 1, 2, \ldots, r$, is the model parameter, $\hat{s}_t$ and ω_t denote predicted state at a future time instant t and white noise at time t, respectively, and y_{t-i} is the observation at time instant $t - i$.

Predicting future states requires tuning the ARM parameters first. This can be achieved with several techniques such as Yule-Walker equations or maximum likelihood estimation. Once these parameters are tuned, they are used along with the historical values to predict the future states $\hat{s}_t$.

2.1.2 Prediction with Bayesian-Inference

BIF is a prediction technique that uses the Bayes rule to update the probability distribution of a hypothesis when additional evidence data is learned. In this prediction technique, a posterior probability distribution $P(s|\boldsymbol{y})$ is derived by a CR user according to a Bayes rule as follows

$$P(s|\boldsymbol{y}) = \frac{P(\boldsymbol{y}|s) \cdot P(s)}{P(\boldsymbol{y})}, \tag{2.3}$$

where s denotes the spectrum occupancy. Afterward, the upcoming data is predicted by the derived posterior probability with the Bayes rule.

2.1.3 Prediction with Long Short-Term Memory

A more accurate data representation can be acquired by the use of multiple hidden layers in the deep architectures of DL models. The usage of multiple layers enables the model to magnify the intrinsic distinctive data features while suppressing the irrelevant information at each layer [40]. Thus, a primary advantage of DL is that it works directly on raw data. This means alleviating the human effort needed in any feature crafting/engineering. LSTM is an artificial recurrent neural network and it can be used as a DL model. This DL model is well-suited for handling grid-like data either in one, two, or multiple dimensions.

LSTM models have a memory block which includes cells and gates. This memory block makes the model capable to use long short-term dependencies [41]. The gates are categorized under three parts according to their practical functionalities. These are input gates, forget gates, and output gates. Their transition equations are shown as follows

$$\boldsymbol{i}_t = \sigma_g(\boldsymbol{W}_i\boldsymbol{y}_t + \boldsymbol{U}_i\boldsymbol{h}_{t-1} + \boldsymbol{b}_i), \tag{2.4}$$

$$\boldsymbol{f}_t = \sigma_g(\boldsymbol{W}_f\boldsymbol{y}_t + \boldsymbol{U}_f\boldsymbol{h}_{t-1} + \boldsymbol{b}_f), \tag{2.5}$$

$$\boldsymbol{o}_t = \sigma_g(\boldsymbol{W}_o\boldsymbol{y}_t + \boldsymbol{U}_o\boldsymbol{h}_{t-1} + \boldsymbol{b}_o), \tag{2.6}$$

$$\tilde{\boldsymbol{c}}_t = \sigma_c(\boldsymbol{W}_c\boldsymbol{y}_t + \boldsymbol{U}_c\boldsymbol{h}_{t-1} + \boldsymbol{b}_c), \tag{2.7}$$

$$\boldsymbol{c}_t = \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t + \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1}, \tag{2.8}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \sigma_h(\boldsymbol{c}_t), \tag{2.9}$$

where $\boldsymbol{i}_t$, $\boldsymbol{f}_t$, and $\boldsymbol{o}_t$ represents input, forget, and output gate's activation vectors, respectively and $\boldsymbol{y}_t$ represents input vector at time t, $\boldsymbol{h}_t$ represents hidden layer at t time step, $\tilde{\boldsymbol{c}}_t$ represents cell input activation vector, $\boldsymbol{c}_t$ denotes cell state vector, σ_g, σ_c, and σ_h denote sigmoid function, hyperbolic tangent function, and hyperbolic tangent function respectively, and biases, recurrent connections, and weights are denoted by $\boldsymbol{b}$, $\boldsymbol{U}$, $\boldsymbol{W}$, respectively.

The above equations show that an input gate controls how a new value flows into the memory. On the other hand, how much of the past information to keep in the memory is decided by a forget gate and an output gate determines the weighting of the values to be used for computing the output activation of the block. Finally, the model can learn how to represent information over multiple time scales since the values of the gating variables vary for each vector element [42].

2.1.4 Prediction with Convolutional Long Short-Term Memory

ConvLSTM is a type of LSTM cell. More specifically, convolution takes place within the LSTM cell, and matrix multiplication is replaced with the convolution operation. The application of convolution allows capturing the spatial features from the image or data grid. The convolutional structures can be observed in ConvLSTM in both the input-to-state transition and state-to-state transition. It has been applied to activation from previous timestamps and input of the current timestamp. Further details can be found in [43].

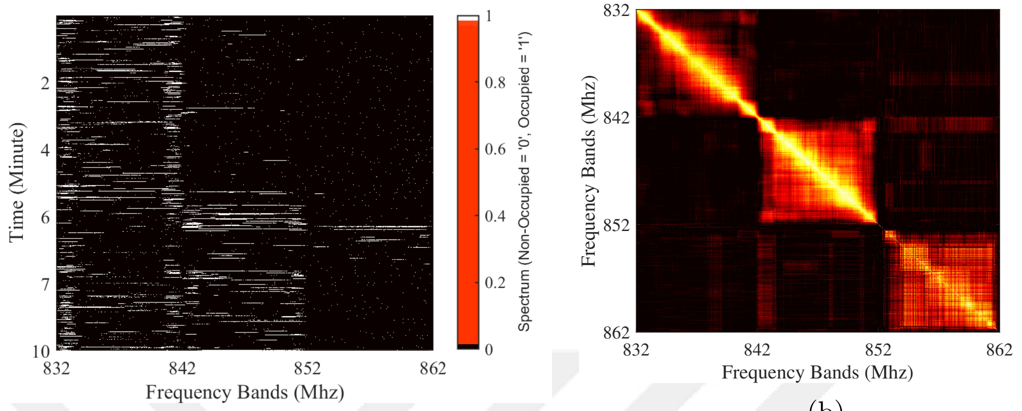
2.2 The Proposed Technique for Spectrum Occupancy Prediction Exploiting Time, Frequency, and Space Correlations

2.2.1 Motivations for Time, Frequency, and Space Correlation Exploitation and Problem Sub-Division

Time series prediction depends on correlation over time. Similarly, the inherently existing frequency and space correlation can be exploited. The following test quantifies this correlation across a time-frequency grid. First, a day-long record of spectrum measurements has been obtained according to the measurement setup detailed in [8]. Figure 2.3 (a) represents spectrum occupancy distributions for all frequency bands and time instants. Besides, the spectrum occupancy correlations were calculated for each frequency point. These results are plotted in Fig. 2.3 (b). We here note that in these figures, each 10 MHz band belongs to a different operator.

The correlations in both time and frequency are shown in Fig. 2.3 (a) where the color bar shows the state of spectrum occupancy. More specifically, the vertical (horizontal) lines represent the correlation in time (frequency). Correspondingly, the correlation coefficient is demonstrated by the block-correlation pattern in Fig. 2.3 (b). As seen in the figure, this correlation is high in the neighboring frequency bands of each operator. Therefore, the existence of strong correlations across both time and frequency can be observed for each operator individually.

In addition to the time and frequency correlations, the exploitation of spatial correlation is advantageous for the spectrum occupancy prediction problem, as



(a)

(b)

Figure 2.3: An example spectrum's (a) occupancy distributions and (b) correlation.

illustrated in Fig. 2.4. Here, BS_1 (main BS) is the BS that the spectrum occupancy prediction will be made on. It can be seen from this figure that when BS_1 is trained without any prior information from BS_2 and BS_3 (neighboring BSs), it decides on the future spectrum as non-occupied. However, BS_2 and BS_3 are full, and the user of BS_2 and BS_3 can occupy the spectrum of the BS_1 for the upcoming intervals, so the number of occupied bands will increase for BS_1. Exploiting the information from this dimension, spectrum occupancy can be predicted more reliably.

Motivated by the aforementioned existence of a multidimensional correlation, current literature uses tensor-based techniques for spectrum occupancy prediction. However, these techniques exhibit high degrees of computational complexity. Alternatively, a *divide-and-conquer* approach can be used. This decentralized approach divides the original problem into smaller sub-problems and solves each one individually. Then, it integrates the solutions to obtain an eventual holistic solution. The following experiment is conducted to investigate the validity of this idea. A dataset is collected according to the measurement setup for the Taksim area as detailed in Section 2.3. Then, the networks are trained with 2D and 3D datasets. We here note that hyperparameters of these networks are detailed in Section 2.4. Finally, a complexity comparison is made in terms of training and testing execution times. The results are listed in Table 2.1. This table shows that

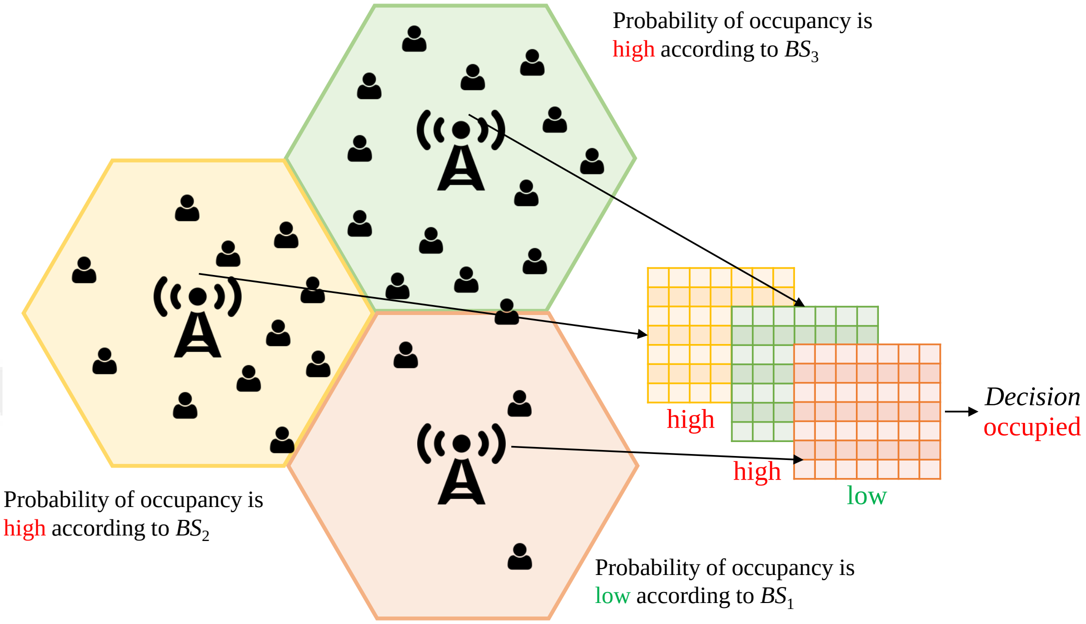

Probability of occupancy is high according to BS_3
Probability of occupancy is high according to BS_2
Probability of occupancy is low according to BS_1
high
high
low
Decision
occupied

Figure 2.4: An illustrative motivation for space correlation.

the computational complexity of the 3D-based technique is much higher than the 2D-based technique.

Table 2.1: Training and testing execution time (seconds) comparison of the tensor and 2D-based technique.

Technique	Execution Time (s)	
	Training	Testing
The tensor-based technique	608	2.8
2D-LSTM-based technique	57.8	0.7

2.2.2 The Proposed Technique

The proposed technique uses a learning-based strategy. This strategy consists of two stages which are referred to as training and testing. In the training stage, the dataset is collected and DL models are configured and trained. Then, in the testing stage, the spectrum occupancy prediction is performed. These stages are detailed below.

In the training stage, a set of spectrum measurements for training are collected by several BSs. Then, the binary occupancies of these measurements are acquired with the comparison of the measured received signal strength indicator (RSSI) with a specific threshold. After collecting the measurements, the dataset is obtained as follows. A sample of input is formed by a sliding window that sweeps the time and frequency [1]. Moreover, the output data of this input is the RSSI value in the medial frequency value for the upcoming time interval, which is to be predicted. According to this setting, RSSI measurements are stored in a 2D shape referring to time and frequency as an input dataset for each BS. Besides, their corresponding occupancies of the main BS are stored as an output dataset.

Once the training data is obtained, the DL model is configured to have a total number of q models, where q denotes the number of spatially-correlated BSs, and each model accepting as input a time-frequency occupancy grid. As mentioned earlier, this configuration simultaneously incorporates space correlation and time-frequency correlations. More specifically, each LSTM model is trained over a set of pairs of a given time-frequency occupancy grid and the corresponding ground-truth occupancy status of the main BS. Then, each DL model calculates a probability of occupancy (P). The occupancy probabilities of all q models are augmented to form an occupancy probability feature vector. Afterward, the occupancy probability vector, along with the true occupancy state form the training data pair of the end-classifier to incorporate space correlation. Finally, the end-classifier is trained over this data pair. These processes are presented in Fig. 2.5.

The run-time operation of the proposed technique is represented by the testing stage. In this stage, spectrum measurements for time, frequency, and space lags are used to predict the corresponding spectrum occupancy probability in the upcoming time instant. This is achieved by feeding the binary occupancies of the 2D dataset as a grid to the DL models already trained, which generate the occupancy as numeric values (probabilities). Then, analogous to the setting in the training stage, the predicted q grid probabilities are augmented in the shape of an occupancy probability vector. Finally, the predicted occupancy probability vector is fed to the end-classifier to yield the eventual occupancy prediction.

[1] For instance, if the length of the sliding window is set to 7, then each input will have a 7×7 matrix of binary occupancies. This matrix will sweep time and frequency axes.

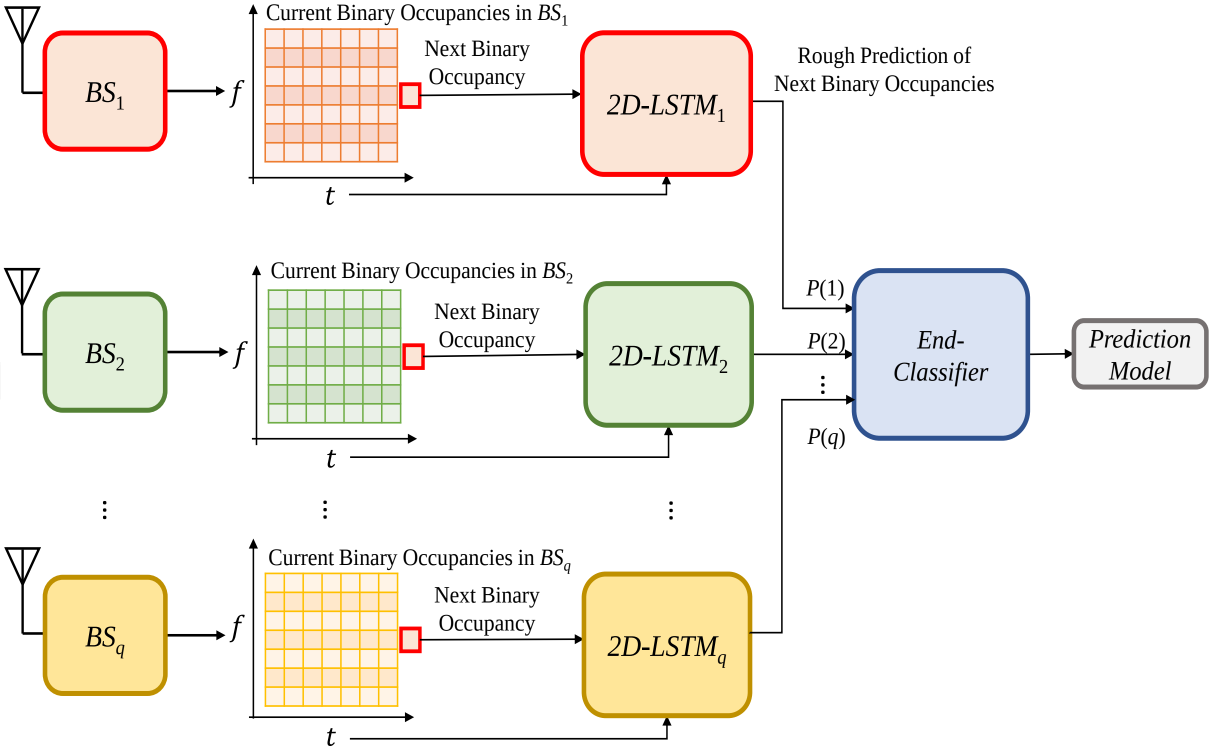


Figure 2.5: The proposed technique for spectrum occupancy prediction-training stage.

Figure 2.6 illustrates these processes.

2.2.3 A Note on Computational Complexity

The computational complexity of the proposed technique can be roughly quantified in terms of the required execution time for training and testing stages. For the training stage, the time computational complexity can be approximated as

$$T(n) = \mathcal{O}_l + \mathcal{O}_s, \tag{2.10}$$

where $\mathcal{O}_l$ and $\mathcal{O}_s$ represent time complexity of a 2D-LSTM model and an end-classifier, respectively. It is noted that although composite LSTM models (q number of LSTM models) are used in the proposed technique, their total execution times equals an LSTM model execution time since they are independent and can work in parallel.

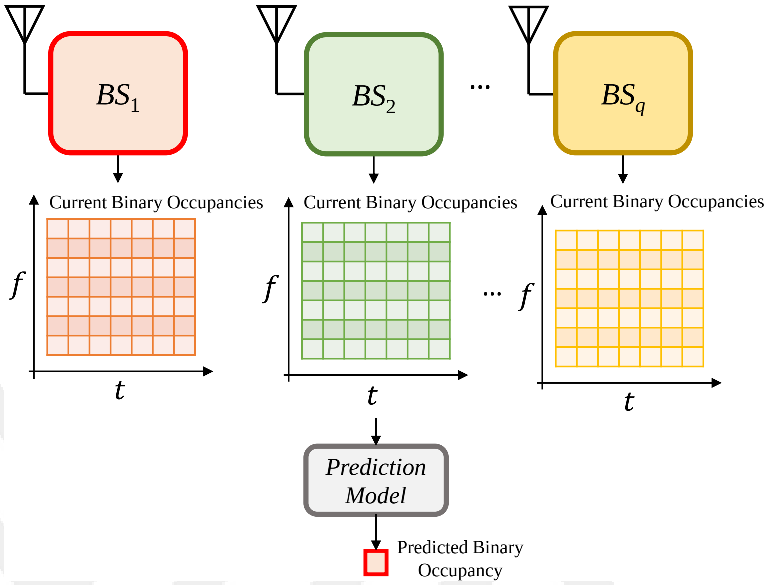

BS_1
BS_2
...
BS_q
Current Binary Occupancies
Current Binary Occupancies
Current Binary Occupancies
f
f
...
f
t
t
t
Prediction Model
Predicted Binary Occupancy

Figure 2.6: The proposed technique for spectrum occupancy prediction-testing stage.

The total number of parameters, p_l, in a standard LSTM network, including one cell in each memory block, neglecting the biases, is as follows [44]

$$p_l = 4n_c n_c + 4n_i n_c + n_c n_o + 3n_c, \tag{2.11}$$

where n_i, n_o, and n_c represent the number of input units, output units, and memory cells, respectively.

The computational complexity of an LSTM models per weight is $\mathcal{O}(1)$ in training stage [45]. Thus, training computational complexity per time step is $\mathcal{O}(p_l)$, and the total complexity of the LSTM models is $\mathcal{O}(p_l)$.

As an end-classifier, let us consider a two-layer neural network. The number of parameters for this classifier, p_s, is $lk + ml$, where k, l, and m denotes the number of neurons at the input, hidden, and output layers, respectively. The computational complexity of an end-classifier ($\mathcal{O}_s$) can be negligible since the number of parameters is very small compared to LSTM. Consequently, the overall complexity of the training stage is $\mathcal{O}(qp_l)$. Besides, the computational complexity of training per sample is approximately two times more as compared to the complexity of testing per sample [46].

2.3 Dataset Generation

2.3.1 Measurement Setup

The measurements of the uplink (UL) private frequency band belong to one of the leading mobile network operators in Turkey and were collected by directly measuring the RSSI of the spectrum as a function of time, frequency, and space. Afterward, the binary occupancies were obtained by thresholding the RSSI measurements, where “0” represents a hole, and “1” represents the occupied spectrum. The BSs measure noise and interference power on the traffic channels (physical UL shared channel). The minimum measurable value by the BSs is -121 dBm [47]. On the other hand, a margin of 3 dB is considered to account for any variations and unforeseen effects [48, 49]. Therefore, -118 dBm is set as a threshold overall in the chapter. In a certain frequency, if the measured RSSI is above this adopted threshold, then this band is considered as occupied and if it is below the threshold, then this band is considered as a hole.

The measurement is conducted on a physical resource block (PRB) basis. This means that each measurement addresses 180 kHz. The accumulated noise and interference power of each PRB is averaged during the recording period. The time resolution of the recording period is 15 minutes. The Comba ODI-065R17M18JJJ-G receiving antenna [50] measures received signals between 852-862 MHz, where the frequency-division duplex is considered as an operational mode. We here note that two neighboring BSs are selected to exploit space correlations. While choosing these neighboring BSs, the number of handover attempts in a one-week period prior to the time of measurement was taken into consideration [2]. We here note that channel quality between the BS and the user; and spatial distance between the BS and the user can be used to select the best neighboring BSs as well. However, we use the number of handovers since the number of handovers is beneficial as it reflects not only channel quality (power) but also the motion pattern of the users which was motivated in Fig. 2.4. The numbers of

[2] The BSs that have made the highest number of handover attempts with the main BSs between July 31, 2020, and August 07, 2020, were selected as neighboring BSs.

handover attempts of two BSs where the most handover attempts are made with the main BSs are given in Table 2.2.

Table 2.2: The number of handover attempts between main BS and neighboring BSs in one week.

BS	Area	
	Taksim	Silivri
BS_2	6429	3178
BS_3	4442	3138

2.3.2 Measurement Procedure and Geographical Locations

The measurements were taken in two regions in Istanbul-Turkey with varying levels of traffic, to investigate the generalizability of the proposed technique. These areas are detailed further below.

2.3.2.1 City Center (Taksim)

For the measurements with high user activity, data were collected simultaneously for the whole spectrum at Taksim Square, Istanbul. The measurements were started at midnight local Istanbul time (GMT+3) on August 07, 2020, and ended at midnight on August 14, 2020. It is worth mentioning that Taksim has one of the most crowded streets in the city, with contiguous buildings around. Taksim is a flat street within a high altitude overlooking high-rise areas in general. Also, it has a central metro station, and it is an important point for transportation. That makes the area very crowded around commute times and rush hours. Furthermore, there are many dining places which make it crowded also at lunch and dinner times. Figure 2.7 shows a picture of the environment for the Taksim area, which has been taken from the corresponding BS. Besides that, satellite captures of the area are shown in Fig. 2.8 to show the area with a wider perspective.

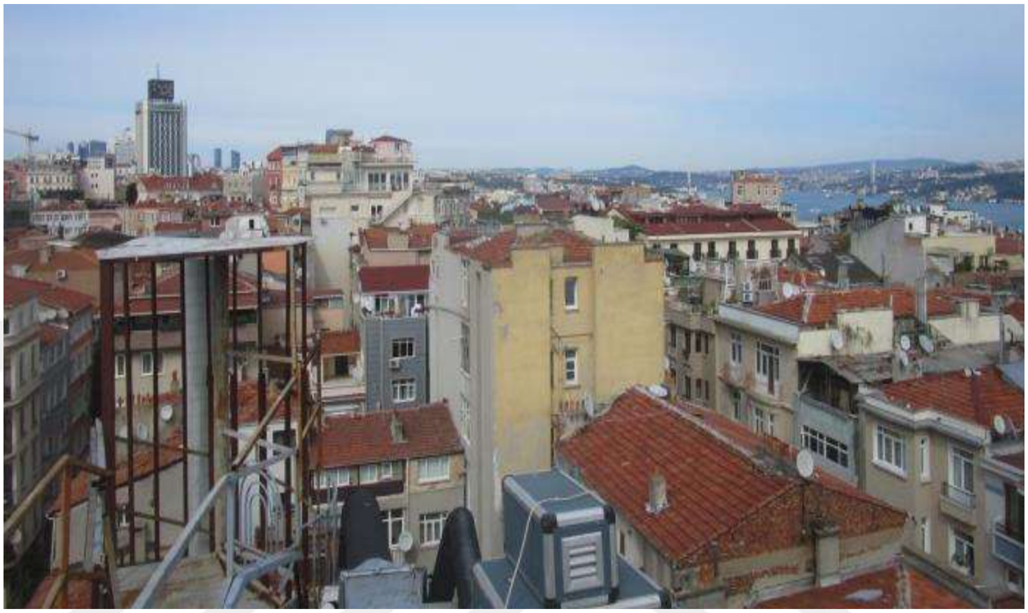


Figure 2.7: Picture of the environment for Taksim.

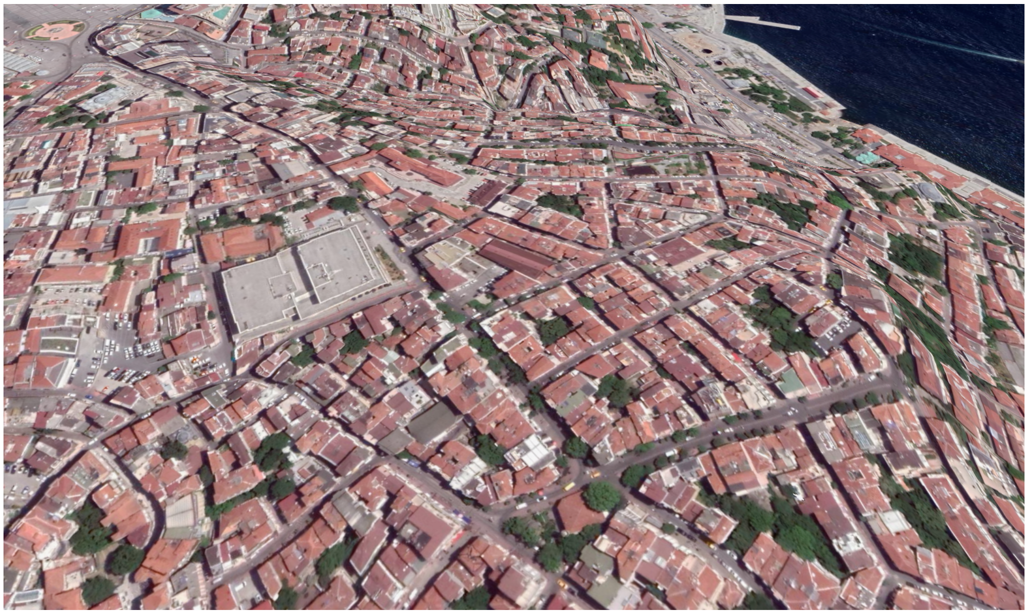


Figure 2.8: Satellite capture of the environment for Taksim.

2.3.2.2 Rural Area (Silivri)

For the measurements with low user activity, data were collected at Silivri where it is located in the rural parts of Istanbul. The measurements were made during the same time as the measurements taken at Taksim. The area is not as crowded as the city center. It has slightly wavy hills with a height of about 60 meters at most. Figure 2.9 shows a picture of the environment for the Silivri area that has been taken from the corresponding BS. Also, a satellite capture picture is given in Fig. 2.10.

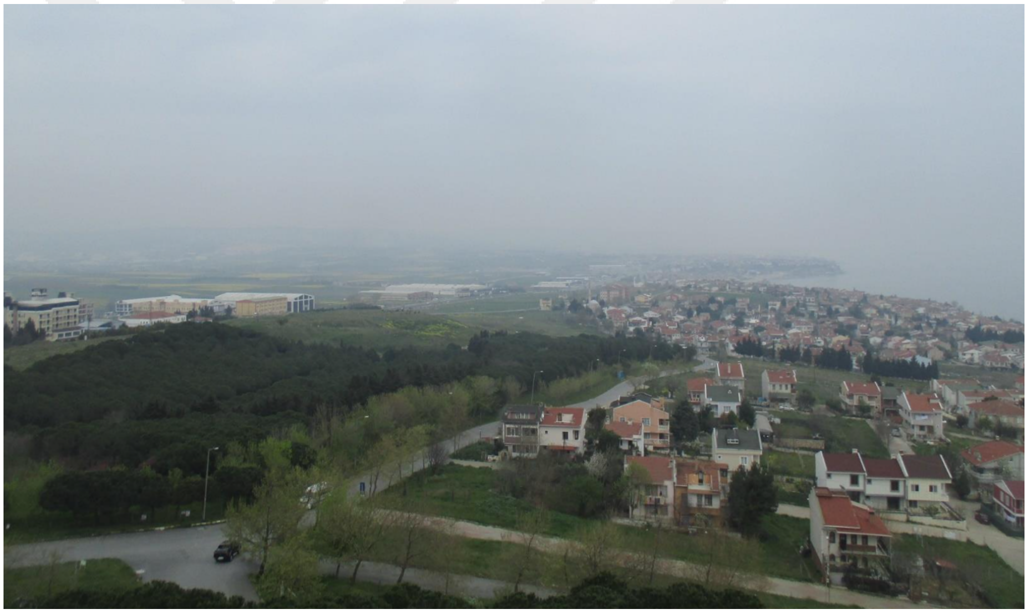


Figure 2.9: Picture of the environment for Silivri.

2.4 Parameter Settings and Experimental Results

The experimental setup used is given in Section 2.3. 180 kHz frequency resolution of the frequency bands within 852-862 MHz was used for the dataset. We here note that 20 kHz was used as a guard band between the subbands [51]. More details regarding the signals used for the datasets is given in Table 2.3. The

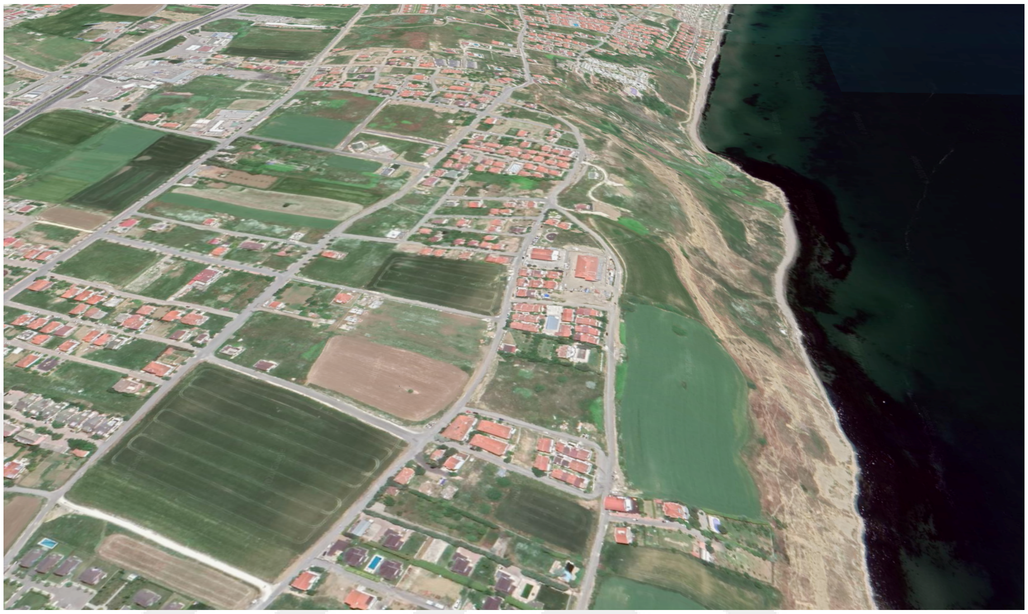


Figure 2.10: Satellite capture of the environment for Silivri.

frequency bands were measured every fifteen minutes for one week. Therefore, 201600 points were measured in total. The binary occupancies were assigned for the related frequencies according to the threshold. A lag value was empirically set to 7 for time and frequency correlations and set to 3 for space correlation according to the performance and generalization capability of the prediction models. The first five-day dataset is used for training, the next day dataset for validation, and the remaining for testing in all of the experiments. In other words, approximately the first 71.5% of the dataset was used for training, the next 14.25% for validation, and the remaining 14.25% for testing in all of the experiments.

Table 2.3: Details of the signals used for the datasets.

3GPP Band	Bandwidth	Frequency	Duplex	Technology
B20	10 MHz	852-862 Mhz (UL)	Frequency division duplexing	Long term evolution-Advanced pro (4.5G)

2.4.1 Hyperparameters of Deep Learning Models

All of the DL models for spectrum occupancy predictions were implemented by Keras [52], an open-source ML library under the Python environment. All of the models were trained and tested on an MSI computer with Intel® Core™ i7-7700HQ central processing unit (CPU) @ 2.80 GHz CPU, 16 GB RAM, GeForce GTX 1050 Ti graphical processing unit, and Windows 10 operating system. All parameters of the DL models were empirically set with the consideration of the performance and generalization capability of these models. DL models were individually trained for Taksim and Silivri datasets. However, the same hyperparameters were used for both since the effect of hyperparameters were negligible for scenarios. Three types of DL models; 1D-LSTM, ConvLSTM, and 2D-LSTM models, and an end-classifier are briefly described below. We note that three 2D-LSTM models (composite 2D-LSTM models) and an end-classifier are used in the proposed technique. The overall model that is used for the proposed technique is illustrated in Fig. 2.11.

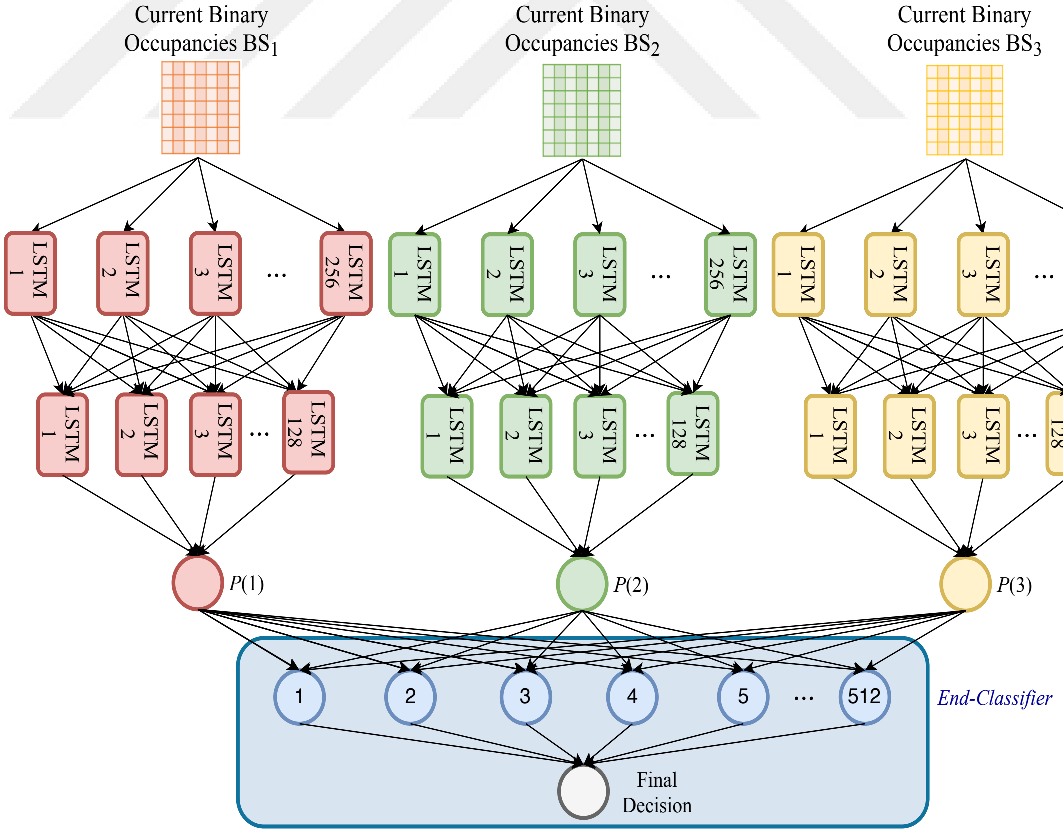


Figure 2.11: Composite 2D-LSTMs in spectrum occupancy prediction.

- *1D-LSTM*: This model uses two LSTM hidden layers and an output layer. Particularly, 256 and 128 hidden units are used in the first and second hidden layers, respectively. The rectified linear unit (ReLU) are used as activation functions. Afterward, the probability of the occupancies is calculated in the output layer which uses a sigmoid activation function with 1 unit. We here note that one unit is enough to represent the occupancies since there are only two classes (spectrum is occupied “1” or not “0”). In total, 461441 parameters are used. Finally, the model is trained with a batch size of 256 and 18 epochs. Efficient adaptive moment estimation (ADAM) is used with an optimum learning rate of 0.0001 during the training. Also, the logarithmic loss function is used for binary classification.

- *ConvLSTM*: The ConvLSTM model is used with 3D data as the state-of-the-art technique. The model includes two ConvLSTM layers, a flatten layer, and an output layer. In the first and second ConvLSTM layers, 256 and 128 units are used, respectively. Afterward, a flatten layer is used to prepare a vector for the output layer. Finally, an output layer is used with one unit. In the output layer, the sigmoid function is used. In total, 4142721 parameters are used. A batch size of 256 and 15 epochs are used to train the model. ADAM is used for adaptive learning rate optimization with an optimum learning rate of 0.00005. Besides, the logarithmic loss function is used for binary classification.

- *2D-LSTM*: This model uses two LSTM hidden layers and an output layer. More specifically, with ReLU activation functions 256 and 128 hidden units are used in the first and second LSTM hidden layers, respectively. Afterward, an output layer is used to calculate the probability of the occupancy. The sigmoid function is used in the output layer. In total, 467585 parameters are used. Finally, the DL model is trained with a batch size of 256 and 15 epochs. ADAM is used for adaptive learning rate optimization and the optimum learning rate in this model is found at 0.00005. Again, for binary classification, the logarithmic loss function is employed.

- *An end-classifier*: A standard two-layer feed-forward network [53] is used as an end-classifier. This classifier consists of a hidden layer and an output layer. The sigmoid functions are used as activation functions. The MATLAB Neural-Network-Toolbox “nprtool” [53] is used for implementation. Scaled conjugate gradient (trainsscg), and cross-entropy (crossentropy) are used for training and performance metrics, respectively. The number of hidden neurons is set to 512 while the number of output neurons is set to one. Therefore, 25600 parameters are used in total.

2.4.2 Performance Evaluation and Discussion

Six spectrum occupancy prediction techniques are compared. These prediction techniques are ARM, BIF, 1D-LSTM using only time correlation, 2D-LSTM using time and frequency correlations, ConvLSTM using multidimensions as a tensor, and the proposed technique using multidimensions.

The performance of a classifier model can be evaluated in terms of *precision* (π), *recall* (ψ), and F_1-score performance metrics. The quantification of the percentage of positive results that are actually positive is measured by the *precision* metric, the percentage information of true positives that are identified correctly as positive are quantified by the *recall*, and the overall measures for the accuracy of classifier models are given by the F_1-score since it gives the harmonic average of *precision* and *recall.* These metrics are defined as follows

$$\pi = \frac{\xi}{\xi + \upsilon}, \psi = \frac{\xi}{\xi + \mu}, F_1\text{-score} = 2 \times \frac{\pi \times \psi}{\pi + \psi}, \tag{2.12}$$

where ξ, υ, and μ represent the numbers of true positive, false positive, and false negative, respectively.

The results can be seen in Tables 2.4 and 2.5. These tables show that the proposed technique is superior to ARM, BIF, 1D-LSTM, and 2D-LSTM-based techniques, and its performance is close to the ConvLSTM technique. This is consistently true in terms of all quality metrics. Besides, the general performance (performance of all techniques) is better in the Taksim scenario since most of the bands are occupied and so the technique has a stronger tendency to predict the bands as occupied.

The complexity performance analysis based on the execution times of the training and testing stages is presented in Tables 2.6 and 2.7. We here note that the

Table 2.4: The *precision* (π), *recall* (ψ), and F_1-score spectrum occupancy performances of ARM, BIF, 1D-LSTM, 2D-LSTM, ConvLSTM, and composite 2D-LSTMs (the proposed technique) for Taksim scenario.

Technique	Measure		
	π	ψ	F_1-score
ARM	0.9210	0.9681	0.9440
BIF	0.9264	0.9738	0.9495
1D-LSTM	0.9602	0.9780	0.9690
2D-LSTM	0.9720	0.9732	0.9726
ConvLSTM	0.9760	0.9763	0.9762
Composite 2D-LSTMs	0.9727	0.9742	0.9735

Table 2.5: The *precision* (π), *recall* (ψ), and F_1-score spectrum occupancy performances of ARM, BIF, 1D-LSTM, 2D-LSTM, ConvLSTM, and composite 2D-LSTMs (the proposed technique) for Silivri scenario.

Technique	Measure		
	π	ψ	F_1-score
ARM	0.8863	0.8704	0.8783
BIF	0.9336	0.9130	0.9232
1D-LSTM	0.9465	0.9165	0.9312
2D-LSTM	0.9462	0.9216	0.9338
ConvLSTM	0.9479	0.9298	0.9388
Composite 2D-LSTMs	0.9476	0.9233	0.9353

value of testing complexity is given for the total computation complexity of all of the testing samples. It is evident that the proposed technique has smaller execution times and faster convergence in both training and testing stages.

Table 2.6: Complexity analyses of ConvLSTM and the proposed technique for Taksim scenario.

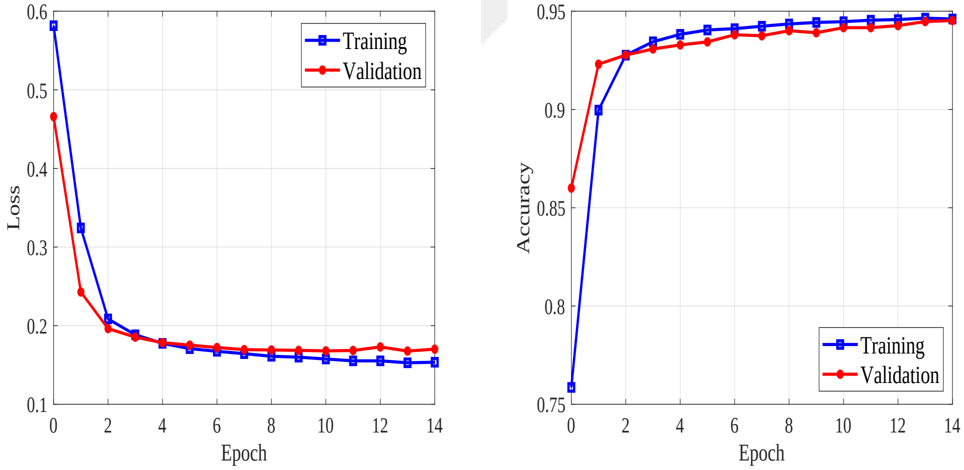
Technique	Execution Time (s)	
	Training	Testing
ConvLSTM	608.7	2.7
Composite 2D-LSTMs	58.1	0.7

The generalization capability of an ML model is an important success criterion; from an ML perspective, a trained model should not memorize the training samples. The quality of a trained ML classifier model is shown in Fig. 2.12 as in the training and testing losses and accuracies versus epochs for the spectrum

Table 2.7: Complexity analyses of ConvLSTM and the proposed technique for Silivri scenario.

Technique	Execution Time (s)	
	Training	Testing
ConvLSTM	610.3	2.9
Composite 2D-LSTMs	58.9	0.8

occupancy prediction when the Taksim dataset is used with the 2D-LSTM. The figure clearly shows that the accuracy of both training and test sets converge to similar values. These results validate the generalizability of the proposed model, as there is no overfitting observed. It is worth mentioning that loss and accuracy graphs are provided only for one 2D-LSTM to prevent repetition. Similar behavior is observed for the graphs of other models and end-classifiers.



(a)

(b)

Figure 2.12: (a) Loss and (b) accuracy graphs for 2D-LSTM.

2.5 Conclusions

This chapter has demonstrated the advantage of exploiting occupancy correlation over time, frequency, and space, for spectrum occupancy prediction. 3D exploitation was achieved using composite 2D-LSTM models that incorporate previous time and frequency spectrum measurements to predict the following spectrum occupancy. It has been shown that the accuracy of the 2D model can be improved by using the 3D model. Also, the proposed technique provides additional accuracy over the 2D model without incurring a substantial increase in complexity. Besides, only a simple re-training for the end-classifier is required when data is not completely available in the proposed technique. Extensive experimental results revealed performance improvements over classical prediction techniques and DL models that use time and joint time/frequency correlations. On the other hand, the performance of the proposed technique is close to the tensor-based technique, but at a less computational cost. These results were validated in terms of the *precision*, *recall*, and F_1-score performance metrics over real-world spectrum measurements. Besides, we conducted complexity analyses in terms of training and testing execution times. The reduction in computational complexity that is achieved by the proposed technique is demonstrated by these analyses. Moreover, this chapter has also shown the practicality of proposed DL-based and other well known spectral occupancy prediction techniques when applied to real-world measurements.

Chapter 3

Signal Relation-Based Physical Layer Authentication

Spoofing attacks are one of the serious threats to communication security [6]. To avoid these kinds of attacks, security is generally ensured through the upper layer techniques, like key-based cryptography [54]. However, the implementation of these techniques is quite hard, mainly when the computational and bandwidth resources are limited. Hence, with the aim to provide more robustness against such limitations, physical layer security (PLS) has attracted great interest in recent years [55]. Specifically, physical layer authentication (PLA) is one of the robust PLS techniques that is used for avoiding spoofing attacks. In line with this, several PLA works including secrecy capacity, channel-based authentication, wiretap code-based authentication, and radio frequency recognition have been reported in the current literature [56].

In particular, a considerable number of channel-based PLA techniques [57–64] have been studied since it is difficult to mimic the channel. Under these techniques, the channels of candidate users are observed. Then, they are compared with the pre-known legitimate user channels in order to determine whether the candidate user is legitimate or not. However, channel-based PLA techniques have some disadvantages. Firstly, the existing techniques rely on the estimation of the channel information for each authentication procedure that includes training and testing stages. Therefore, accurate channel estimation is required for both stages. For the accurate channel estimation, the number of pilots using the channel must be higher than a certain amount. However, in next-generation wireless networks, the number of pilots is expected to decrease, which reduces channel prediction quality and reduces authentication performance [65]. Secondly, the exact channel taps cannot be known in the real-time channel. For instance, when the channel is modeled with a higher number of taps than it actually has, these extra taps may deteriorate the authentication performance drastically.

To overcome these drawbacks, this chapter proposes a novel technique based on the detection of the received signal symbols. Specifically, a signal relation-based authentication technique is proposed which relies on the detection and comparison of the received signal symbols in the testing stage instead of detecting and comparing the channel itself. It is shown that such a technique works efficiently when the pilot number decreases and when the number of channel taps is overnumbered. In addition, to detect received symbols for authentication, this chapter develops two distinct solutions, which are called minimum mean-square error (MMSE) (due to being the most utilized algorithm in the signal detection [66]) and long short-term memory (LSTM) (due to being suitable for the time-forecasting analysis [67]).

Extensive simulation results show the effectiveness of the technique in terms of receiver operating characteristic (ROC) curves as the number of pilots decreases. Specifically, in the cases where the channel is modeled with a higher number of taps than its real tap number, our results reveal that such effectiveness is more pronounced. Besides, it is demonstrated that the LSTM-based solution can be more advantageous than the MMSE-based solution when the number of training pilot symbols increases.

The rest of this chapter is organized as follows. Section 3.1 introduces the system model and useful information. MMSE and LSTM-based signal relation-based solutions are provided in Section 3.2. Section 3.3 presents the simulation results and discussions. Finally, Section 3.4 concludes the chapter.

3.1 System Model and Useful Information

This chapter considers a PLA system, as illustrated in Fig. 3.1, consisting of Alice (legitimate transmitter), Bob (receiver) and several spoofers (extremely intelligent illegitimate transmitters), in which time-division duplex is considered as an operational mode. In this figure, $\boldsymbol{h}_A$ represents Alice's channel vectors and $\boldsymbol{h}_S$ represents spoofers' channel vectors where M is the arbitrary number of spoofers. In this system model, firstly, the frames are transmitted from Alice to Bob. Afterward, spoofers try to mimic Alice in order to fraud Bob. Therefore, Bob has to identify whether the incoming frames are from Alice or not. We assume that Alice and spoofers are located one-half wavelength away from each other (after a distance of approximately half a wavelength, the received signal rapidly decorrelates [58]) and the spoofers are not able to mimic the legitimate user's channel.

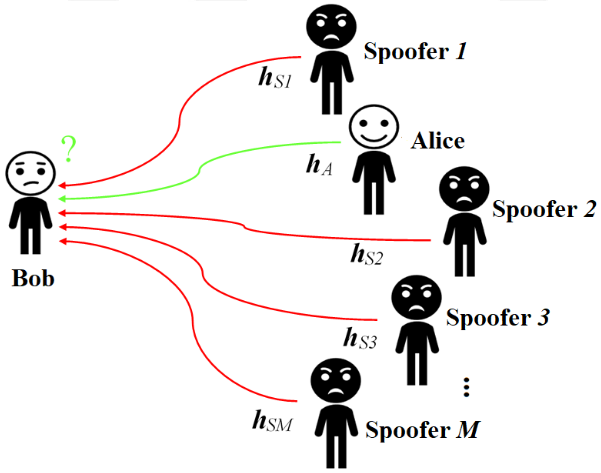


Figure 3.1: An illustration of a PLA system model that includes Alice (legitimate transmitter), Bob (receiver) and several spoofers (extremely intelligent illegitimate transmitters).

3.1.1 Channel Model

Digital symbols at Alice are transmitted as

$$x(t) = \sum_{k} p_k f(t - kT), \tag{3.1}$$

where p_k represents a sequence of symbols, $f(\tau)$ is the impulse response of transmit filter as a delay function (τ), and T signifies the symbol period. The transmitted signal passes through a wireless channel that can be modeled via discrete filter taps [68]. Then, in the existence of noise, the received signal can be written as

$$y(t) = \sum_{l=0}^{L-1} c(l)x(t - \tau(l)) + n(t), \tag{3.2}$$

where the number of channel taps is denoted by L, the lth complex channel coefficient is denoted by $c(l)$, and the delay is denoted by $\tau(l)$. The delays are supposed to be spaced evenly, i.e. $\tau(l) = lT/W$, where W is an integer. In general, W is set to 1 and 2 for the symbol spaced channel modeling and for fractionally spaced channel modeling, respectively. The noise term, $n(t)$, can be modeled as a white complex Gaussian noise. For the sake of simplicity, this technique focuses on symbol-spaced channel modeling.

The received signal at the receiver is filtered by a filter that is matched to the pulse shape, and sampled with the sampling period T_s, as formulated below

$$r_k = \int f^*(\tau)y(\tau + kT_s)d\tau, \tag{3.3}$$

where superscript '$*$' indicates complex conjugate. Then, by replacing (3.2) into (3.3), the received signal samples can be stated as

$$r_k = \sum_{j=0}^{J-1} h(j)b_{k-j} + z_k, \tag{3.4}$$

where $h(j)$ represent jth composite channel coefficient, which follows a Rayleigh distribution.

For a better understanding of the channel estimation techniquee, this technique uses vector and matrix forms. Considering S received samples total, the following column vectors are defined: $\boldsymbol{r} = [r_0 \; r_1 \ldots r_{S-1}]^T$, $\boldsymbol{h} = [h(0) \; h(1) \ldots h_{J-1}]^T$, and $\boldsymbol{z} = [z_0 \; z_1 \ldots z_{S-1}]^T$. Then, (3.4) can be rewritten as

$$\boldsymbol{r} = \boldsymbol{B}\boldsymbol{h} + \boldsymbol{z}, \tag{3.5}$$

where $\boldsymbol{B}$ is a $S \times J$ matrix so that its rows denote the different shifts of the transmitted sequence of symbols.

3.1.2 Existing Channel-Based PLA

Typical channel-based PLA techniques consider that initially Bob estimates and stores the channel vector from Alice which is the secret information between them at the initialization of the authentication process [62]. The corresponding stored channel vector of Alice is expressed as

$$\boldsymbol{h}_A = [h_{A,0} \; h_{A,1} \; h_{A,2} \; \ldots \; h_{A,N-1}]^T. \tag{3.6}$$

Here, subscript A represents that the channel belongs to Alice and N represents the number of samples that are uniformly sampled in each frame.

Following the initialization of the authentication process, when the first frame is received, Bob estimates the channel by using this frame as

$$\boldsymbol{h}_t(k) = [h_{t,0}(k) \; h_{t,1}(k) \; h_{t,2}(k) \; \ldots \; h_{t,N-1}(k)]^T. \tag{3.7}$$

In (3.7), subscript t represents transmitter to be authenticated and ($k = 1, 2, \ldots$) represents the frame index. Then, based on $\boldsymbol{h}_t(1)$ and $\boldsymbol{h}_A$, Bob has to identify if the first frame is coming from Alice or not.

If Bob determines that the first frame belongs to Alice, Bob should continue authenticating the second frame through the estimated channel vectors for both frames. In other words, by using $\boldsymbol{h}_t(1)$ and $\boldsymbol{h}_t(2)$, Bob determines if the second frame belongs to Alice or not. If it does not belong to Alice, an alarm should be given. Then, the authentication process is continued with the determination of whether the third frame (and so on) is from Alice based on $\boldsymbol{h}_t(2)$ and $\boldsymbol{h}_t(3)$ (and so on).

3.1.3 LS-Based Channel Estimation

LS is the most commonly utilized channel estimation algorithm, especially, in case of the presence of a training sequence [69]. The estimation of LS channel coefficients can be stated as

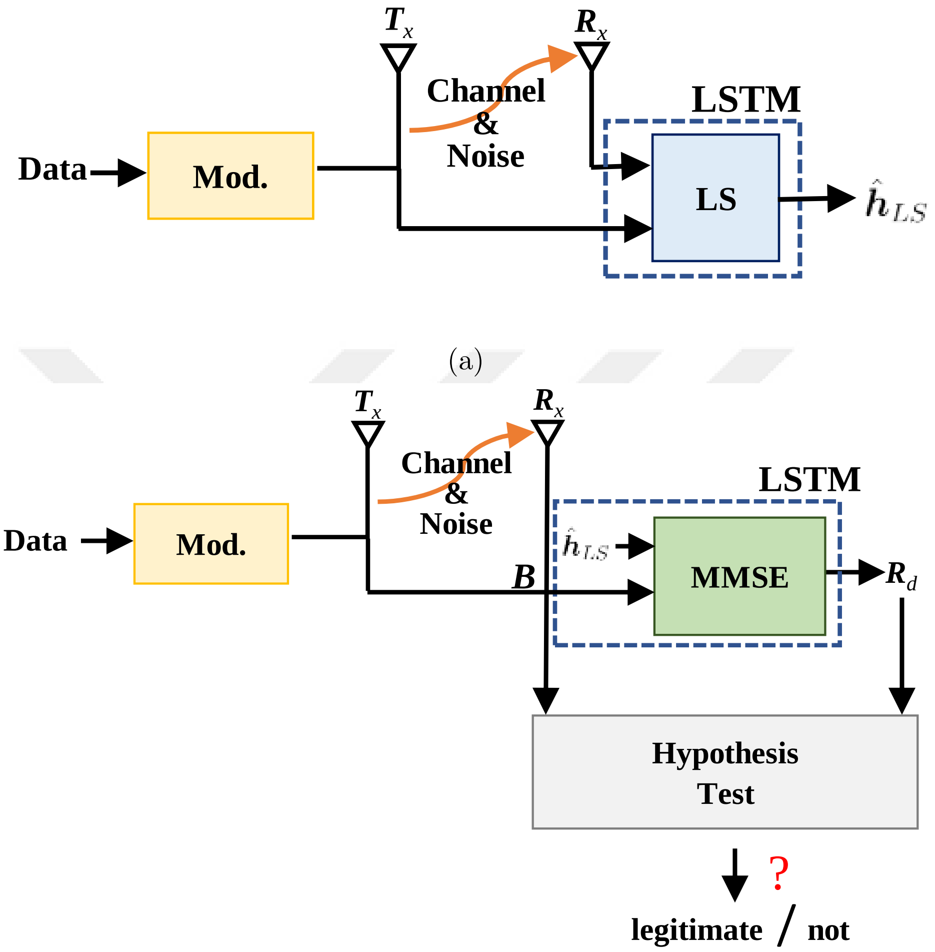
$$\hat{\boldsymbol{h}}_{LS} = \arg\min_{h} (\boldsymbol{r} - \boldsymbol{B}\boldsymbol{h})^{H} (\boldsymbol{r} - \boldsymbol{B}\boldsymbol{h}). \tag{3.8}$$

Then, by differentiating (3.8) according to each channel coefficient and setting the result to zero, the corresponding coefficient can be obtained in a closed-form expression as

$$\hat{\boldsymbol{h}}_{LS} = (\boldsymbol{B}^{H}\boldsymbol{B})^{-1}\boldsymbol{B}^{H}\boldsymbol{r}. \tag{3.9}$$

3.2 The Proposed Technique

This section introduces our proposed signal relation-based authentication technique. The corresponding technique is illustrated in Figs. 3.2 (a) and 3.2 (b), as a training and testing stage, respectively. As can be seen from these figures, firstly the transmitted data is modulated and transmitted symbols ($\boldsymbol{T}_x$) are obtained. Afterward, $\boldsymbol{T}_x$ is passed through a wireless channel and noise is added. At the receiver side, received signal symbols are obtained and are denoted by $\boldsymbol{R}_x$. Here, for the detection of received signal symbols, two different solutions, namely MMSE and LSTM, are developed. Note that, due to the difference between the proposed technique and the existing channel-based authentication techniques, $\boldsymbol{h}$ notation given in Section 3.1-B is revised as $\boldsymbol{r}$ notation, where $\boldsymbol{r}$ represents received signal symbols. These are also used in the hypothesis test to determine whether the user is legitimate or not. In line with this, in testing stage, detected received



(a)

(b)

Figure 3.2: Illustration of signal relation-based PLA using MMSE and LSTM for (a) training stage, (b) testing stage.

signal symbols from Alice, i.e. $\boldsymbol{r}_A$, and the received signal symbols, i.e., $\boldsymbol{r}_t$, are rephrased as $\boldsymbol{r}_A = [r_{A,0}\ r_{A,1}\ r_{A,2}\ \dots\ r_{A,N_p-1}]^T$ and $\boldsymbol{r}_t = [r_{t,0}\ r_{t,1}\ r_{t,2}\ \dots\ r_{t,N_p-1}]^T$, where N_p represents the number of pilot symbols. The steps of the proposed solutions are detailed below.

3.2.1 MMSE Signal Relation-Based Technique

At the receiver side, according to transmitted and received training symbols, the channel for Alice is estimated by using (3.9). In this equation, $\boldsymbol{r}$ denotes the training $\boldsymbol{R}_x$ symbols and $\boldsymbol{B}$ is obtained by signifying the different shifts of the $\boldsymbol{T}_x$ training symbols. Following the estimation of the channel in the training stage, to detect received test symbols, the MMSE signal detection algorithm is utilized. In this procedure, the inverse operation of (3.9) is applied using the estimated $\boldsymbol{h}$ from the training stage and $\boldsymbol{B}$ from the testing stage. Finally, to identify whether the user is legitimate or not, a hypothesis test according to $\boldsymbol{R}_x$ test symbols and detected test symbols ($\boldsymbol{R}_d$) is used.

3.2.2 LSTM Signal Relation-Based Technique

LSTM is a good match for time series forecasting problems. Thus, an LSTM-based signal relation solution is developed as an alternative to the aforementioned MMSE solution. Note that in LSTM signal relation, $\boldsymbol{B}$ and $\boldsymbol{r}$ are obtained through the same way as the MMSE algorithm. The proposed LSTM solution uses $\boldsymbol{B}$ as an input and $\boldsymbol{R}_x$ as an output. In this solution, firstly, the system is trained with training $\boldsymbol{B}$ as input and training $\boldsymbol{R}_x$ symbols as an output. Afterward, in the testing stage, the trained LSTM model is made a decision to get $\boldsymbol{R}_d$. Finally, as in the above solution, the hypothesis test is used to identify if the user is legitimate or not.

3.2.3 Hypothesis Test

To determine if the incoming frame is legitimate or not, as in many previous authentication works [58, 62], the proposed signal relation-based authentication technique is formulated as a sequence of hypothesis test problems. The first hypothesis test can be defined as

$$\begin{aligned} &\mathcal{R}_0 : \boldsymbol{r}_t(1) \text{ is from Alice}, \\ &\mathcal{R}_1 : \boldsymbol{r}_t(1) \text{ is not from Alice}, \end{aligned} \tag{3.10}$$

which can be reformulated as

$$\mathcal{R}_0 : \boldsymbol{r}_t(1) = \boldsymbol{r}_A, \mathcal{R}_1 : \boldsymbol{r}_t(1) \neq \boldsymbol{r}_A, \tag{3.11}$$

where $\boldsymbol{r}_A = \boldsymbol{r}_A(0)$.

The null hypothesis, $\mathcal{R}_0$, means that $\boldsymbol{r}_t(1)$ is from Alice, rather than spoofers. The alternate hypothesis, $\mathcal{R}_1$, means that $\boldsymbol{r}_t(1)$ is from spoofers, rather than Alice. Assume that, throughout the first hypothesis test, Bob continues authenticating the subsequent frames unless $\mathcal{R}_1$ is received and an alarm will be given out. Afterward, the second hypothesis can be stated as

$$\mathcal{R}_0 : \boldsymbol{r}_t(2) = \boldsymbol{r}_A(1), \mathcal{R}_1 : \boldsymbol{r}_t(2) \neq \boldsymbol{r}_A(1). \tag{3.12}$$

To sum up, the signal relation-based authentication can be fitted into a problem of determining if the subsequent signal vector $\boldsymbol{r}_t(i)$ of given $\boldsymbol{r}_A(i-1)$ signal vector between legitimate users is coming from Alice or not. This hypothesis test problem can be represented in a compact form as

$$\mathcal{R}_0 : \boldsymbol{r}_t = \boldsymbol{r}_A, \mathcal{R}_1 : \boldsymbol{r}_t \neq \boldsymbol{r}_A. \tag{3.13}$$

3.3 Simulation Results and Discussions

3.3.1 Parameter Settings

Illustrative simulations are conducted to show the performance of the signal relation-based authentication. Binary phase-shift keying (BPSK) is used as a modulation technique. All the simulations, except the last one, consider 1000 number of training pilot symbols from each 0, 5, and 10 dB signal-to-noise ratio (SNR) values. Therefore, the total number of pilot symbols that are used is 3000 for each training frame. As channel model, Rayleigh is used in which the number of channel taps is $L = 20$ and $\boldsymbol{E}[|\boldsymbol{h}|^2] = 1$. More specifically, correlated channel realizations are generated for the legitimate user and uncorrelated channel realizations are generated for spoofers based on the channel decorrelation concept [58]. The assumed correlation model of $\boldsymbol{h}_c$ can be formulated as: $\boldsymbol{h}_c = \rho\boldsymbol{h} + (1-\rho)\boldsymbol{h}_i$, where ρ is the correlation factor and $\boldsymbol{h}_i$ denotes an independent channel. Without loss of generality, we assume $\rho = 0.9$ [70]. The noises are modeled by $\mathcal{CN}(0, \sigma_N^2)$, i.e., zero-mean complex Gaussian samples with variance σ_N^2. For all simulations, 10000 samples of the legitimate user frames and 10000 samples of the spoofers frames are used and it is assumed that each spoofers' signal has the same parameters of the legitimate transmitter (Alice) signal. This means that extremely intelligent spoofers are considered. The only information unknown by the spoofers are the channel corresponding to Alice.

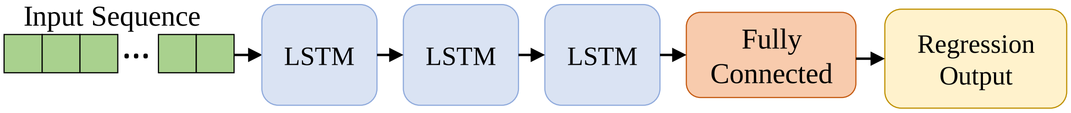


Figure 3.3: The overall block diagram of the proposed LSTM model with an input layer, three LSTM layers, a fully connected layer, and an output layer.

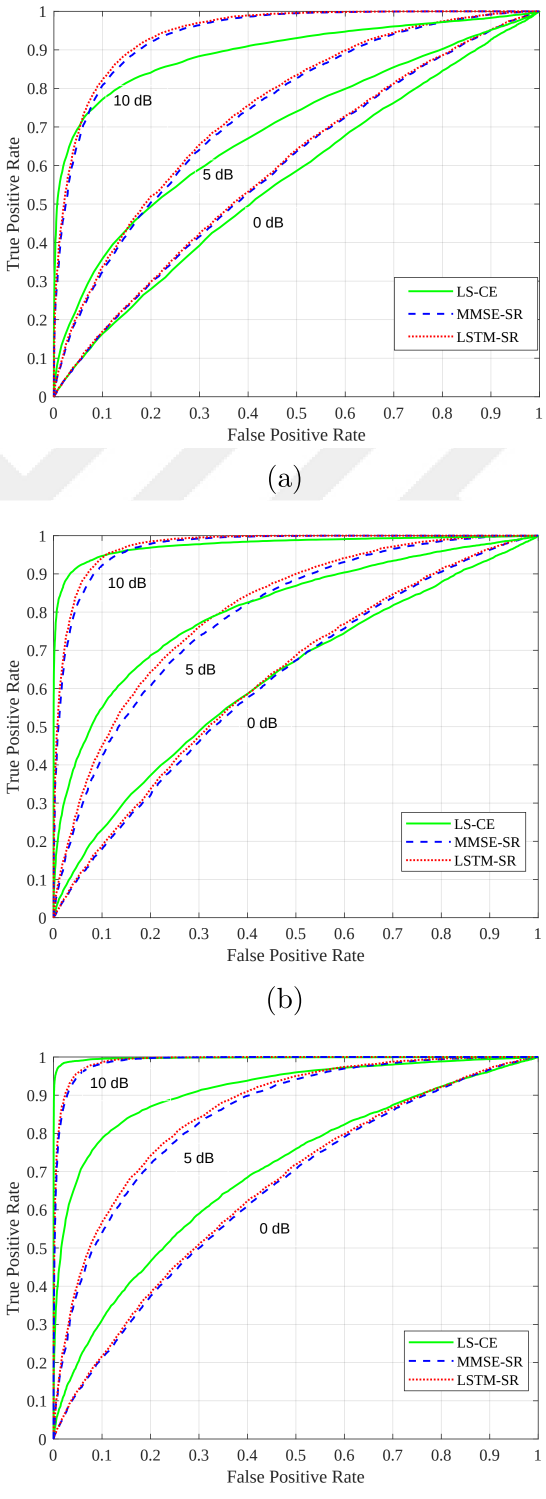
We develop an LSTM network that consists of a sequence input layer, three LSTM layer, a fully connected layer, and a regression output layer for sequence-to-one regression. The size of the sequence input layer is set as the number of features of the input data which is a conjectural number of channel taps (J). The number of hidden units for the three LSTM layers are assigned to 64, 128, and 256 units, respectively. To prevent overfitting, dropout regularization is inserted after the last LSTM layer with a dropout rate of 0.4. Afterward, a fully connected layer, which has 512 units, is added. Lastly, an output layer that predicts the single numerical value, which is the received symbol, is used. The model is fit by the utilization of an efficient adaptive moment estimation (ADAM) version of stochastic gradient descent and optimized with "mse" loss function. The LSTM model is trained with 10 epochs. All parameters are empirically tuned by considering the generalizability and performance of the proposed LSTM model. The overall block diagram for the proposed LSTM model is depicted in Fig. 3.3.

The performance of the techniques for detecting the legitimate user is graphically shown by the ROC curves. ROC is a probability curve which plots the true positive rate on the y-axis against the false positive rate on the x-axis at different classification thresholds. More items would be classified as positive when the classification threshold is decreased and therefore, false positives (FP) and true positives (TP) would be increased. The true positive rate is obtained by the proportion of correctly predicted as positive observations out of all positive observations (TP/(TP + FN)). The false positive rate is similarly acquired by the proportion of incorrectly predicted as positive observations out of all negative observations (FP/(TN + FP)).

3.3.2 Performance Evaluation and Discussions

In the performance comparison tests, three authentication solutions are compared, which are channel estimation-based PLA, MMSE signal relation-based PLA, and LSTM signal relation-based PLA. In the following results, these solutions are denoted by LS-CE, MMSE-SR, LSTM-SR, respectively.

In Fig. 3.4, ROC curves are plotted for SNR values between 0 dB and 10 dB with a step size of 5 dB. As represented in Fig. 3.4 (a), the proposed technique using MMSE and LSTM improve the user authentication quality compared with the channel-based authentication when $N_p = 40$, because when N_p is low, channel estimation cannot be performed properly and channel-based authentication does not authenticate users accurately since it is significant to know the channel coefficients for the authentication in channel-based PLA technique. However, when N_p increases, the channel estimation quality can be better and the performance of the channel-based PLA technique can be better than signal relation-based PLA. Figs. 3.4 (b) and 3.4 (c) show that when N_p is increased, the performance of all techniques improves. Besides, channel-based PLA performance improvement is higher than the proposed solutions. Therefore, these results indicate that the authentication accuracy of the proposed solutions will be higher than the



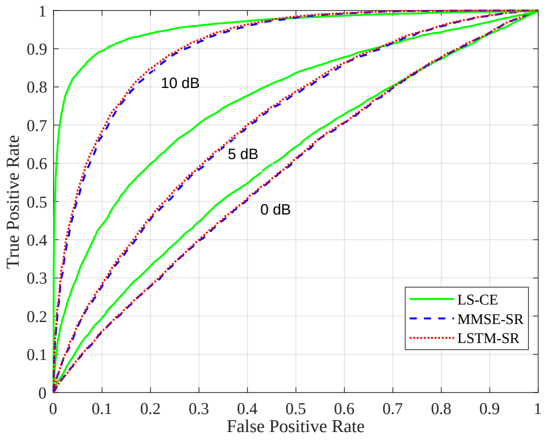
(a)

(b)

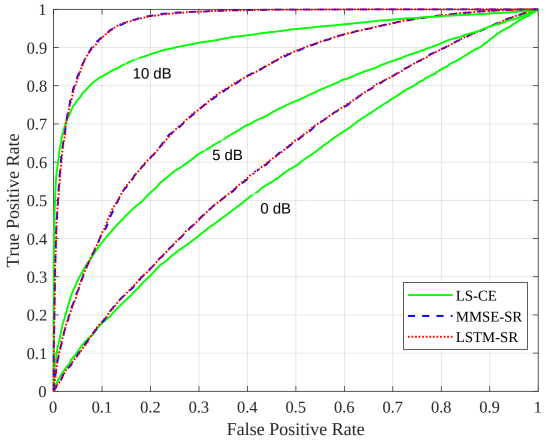
(c)

Figure 3.4: The detection performance of legitimate user when exact L is known (L=J); (a) $N_p = 40$, (b) $N_p = 50$, (c) $N_p = 60$ pilot test symbols.

channel-based PLA technique in the low pilot symbols case and the authentication accuracy of the channel-based PLA technique becomes high when the pilot number increases.



(a)



(b)

Figure 3.5: The detection performance of legitimate user ($N_p = 50$) when channel modeled as (a) undermodel ($L > J$), (b) overmodel ($L < J$).

In Fig. 3.5, we examine the case when the exact L is not known. Along this direction, letting $J = 15$, where $J < L$ (undermodel) and $N_p = 50$, ROC curves are plotted. Fig. 3.5 (a) shows that channel-based PLA is less affected by undermodel case than the proposed solutions. This happens because if the channel represented by undermodel, received symbols cannot be detected exactly true and

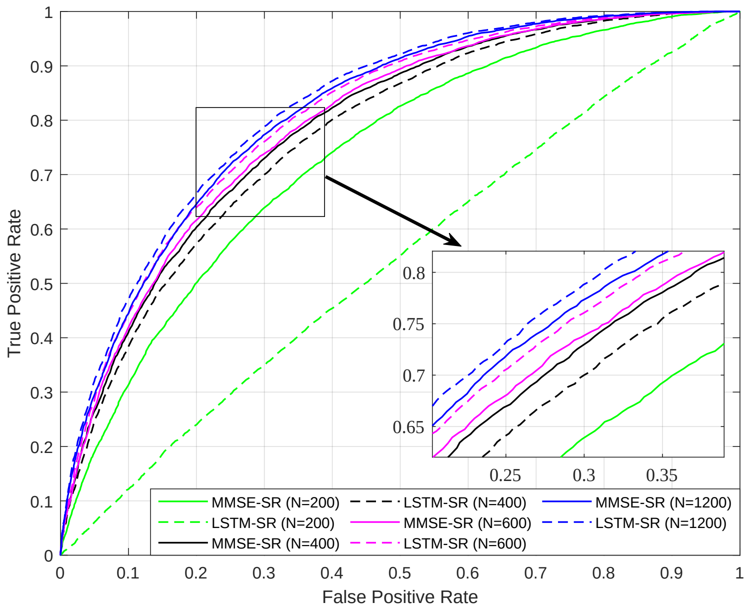


Figure 3.6: The detection performance of legitimate user ($N_p = 50$) for different number of training pilot symbols (N) when exact L is known ($L = J$).

detection of the received symbols correctly is more important in signal relation-based PLA solutions than the channel-based PLA one since signal relation-based solutions require each pilot symbol (that are estimated in the training stage) for the authentication. Moreover, by taking $J = 25$, where $J < L$ (overmodel) and $N_p = 50$, ROC curves are plotted. It is seen from Fig. 3.5 (b) that the channel-based PLA technique is more affected by overmodel case than the signal relation-based technique with MMSE and LSTM. This is because when the overmodel is used, the estimated channel coefficients refer directly to noise where $J > L$ and these misestimated channel coefficients are used for user identification in channel-based authentication techniques.

Finally, in Fig. 3.6, MMSE and LSTM solutions are compared through ROC curves, when the exact L is known, $SNR = 5$ dB and $N_p = 50$ for 200, 400, 600 and 1200 number of training pilot symbols (N). This figure shows that the performances of both techniques increase significantly when the number of training pilot symbols increases from 200 to 600. However, from ($N = 600$) to ($N = 1200$) and to higher values, the improvement is marginal. Besides, when the number of training pilot symbols is 200, the MMSE-based solution is superior to the LSTM-based solution. Then, as the number of training pilot symbols increases, the performance accuracy of the LSTM-based solution increases even more and performs better compared to the MMSE-based solution.

3.4 Conclusions

This chapter proposed a signal relation-based authentication technique to determine the legitimation of a user, by developing MMSE and LSTM signal detection solutions. With the proposed solutions, the channel information is not required for the testing stage, so they work even when the number of pilot symbols decreases. The effectiveness of these solutions was verified by extensive simulations. The simulation results showed that when the number of test pilot symbols is reduced, the MMSE and LSTM solutions are superior to existing channel-based PLA. Besides, it was shown that when the channel is modeled as an undermodel, the performance of the conventional technique is superior to signal relation-based PLA techniques. However, for the overmodel case, the performances of the proposed solutions are better than the conventional technique. Last, but not least, simulation results showed that when the number of training pilot symbols increases, the LSTM-based solution is superior to the MMSE-based solution.

Chapter 4

Deep Learning-Assisted Detection of PUE and Jamming Attacks in Cognitive Radio Systems

4.1 Introduction

Due to the broadcast nature of wireless communication, CR is naturally vulnerable to many security threats. Examples include eavesdropping, jamming attack (JA), and primary user emulation attack (PUEA) [7]. Amongst these, PUEAs and JAs are the most critical as they can prevent the exploitation of the spectrum by causing false alarms about spectrum occupancy. More specifically, a primary user emulator (PUE) can emulate the transmission characteristics similar to a (legitimate) PU while a jammer can generate intentional interference. In both cases, the effects of attacks lead to a faulty conclusion about the spectrum occupancy.

Several approaches have been developed for detecting PUEAs and JAs. Conventional approaches include cryptography-based techniques. However, these approaches suffer from key management and distribution issues in heterogeneous wireless communication networks [54]. To solve these issues, a simple energy detection (ED)-based technique [71] is proposed for identifying legitimate PUs. Another approach uses a Markov random field-based belief propagation framework based on ED for PUEA detection [72]. Despite the simplicity of ED-based techniques, they tend to create high levels of false alarm rates [7]. Another category of detection techniques is based on the exploitation of physical layer characteristics. Although these techniques are effective in detecting the CR security threats, more robust and intelligent techniques are still required to support diverse services in various scenarios [73].

Recent literature considers using machine learning (ML) as a mechanism to detect PUEAs and JAs. This usage is based on training a machine/network to identify such attacks, where training is conducted over features extracted from received signals [74]. Along this line, support vector machine-based approaches view attack detection as a classification process [75]. More recently, artificial neural networks are used along with ED and cyclostationary features. In this setting, ED is applied first to detect user existence. Then, cyclostationary features are used to distinguish between legitimate users and attacks. Likewise, for PUEA detection, a radio-frequency (RF) fingerprinting [76] is used to create RF-fingerprint profiles for each transmitter. [11] uses the convergence patterns of sparse coding as features for ML-based classification to identify such attacks.

Despite the witnessed success of ML-based attack detection techniques, they share a common drawback of requiring consecutive stages of feature extraction. More specifically, the numerical errors tend to propagate across these stages during both the training and usage of the ML network [77]. This restricts the full exploitation of raw data [78] and calls for developing ML techniques alleviating the need for feature extraction/crafting. Deep learning (DL) seems to be an ideal candidate for this objective.

In this chapter, a DL-based technique is proposed to detect the PUEA and JA without explicit feature extraction. This is achieved by designing and optimizing a one-dimensional convolutional neural network (1D-CNN) architecture. Numerical simulation results show that the performance of the proposed technique is superior to ML with feature extraction techniques in terms of detection accuracy.

4.2 System Model and Preliminaries

A standard CR environment that includes a PU node, an SU node, and an illegitimate node is considered in this chapter. This setting allows an SU node to opportunistically exploit the spectrum when there is an illegitimate node that can launch a PUEA or a JA. Fig. 4.1 conceptually illustrates such signals and the system model. Here, signals transmitted by a PUE (jammer) are both assumed to be structured (random). Indeed, we assume an extremely smart PUE that mimics a (legitimate) PU.

A transmitted signal can be demonstrated as $\boldsymbol{x} = \boldsymbol{A}s$, where $\boldsymbol{A}$ is an $N \times N$ coefficient matrix . Each component is denoted by $a_{i,j}$ with $i, j = 1, \dots N$, and $s = [s_1(t), \dots, s_N(t)]^T$ is the transmitted data vector. Any coordinate of s is $s_i(t) = \sum_{k=-\infty}^{\infty} d_k u(t - kT_s) e^{j2\pi f_{c,z} t}$, where the symbol duration is denoted by T_s, the center frequency is represented by $f_{c,z}$, digitally modulated data symbols are denoted by d, and the pulse shaping filter is $u(t)$, with $z = 1, 2, \dots, N$.

The signal sent by any node at the receiver can be represented as

$$\boldsymbol{y} = \boldsymbol{h}\boldsymbol{x} + \boldsymbol{n}, \tag{4.1}$$

where $\boldsymbol{h}$ is the general channel vector between any transceiver pair, and $\boldsymbol{n}$ is additive white Gaussian noise. Due to the spatial decorrelation concept, the channel, $\boldsymbol{h}$, between different transmitter-receiver pairs is different [79].

The system model used consists of a single-cell downlink multiple-input multiple-output system with a uniform linear antenna array (ULA) containing

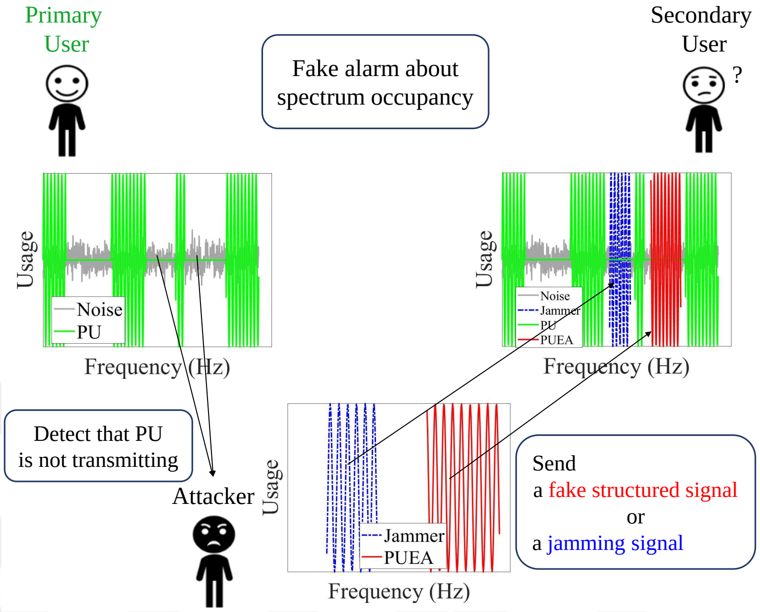


Figure 4.1: A visual illustration of a PU, attacker (PUE or jammer), and noise.

N_t antennas. The channel model assumed is based on geometry-based stochastic channel model (GSCM) [80]. It is noteworthy that this model promotes the channel sparsity assumption. Thus, performance comparison with [11] is valid and fair. The channel impulse response is denoted by $\boldsymbol{h} \in \mathbb{C}^{N_t}$ and is a narrowband block flat-fading channel. The general channel, $\boldsymbol{h}$, between the base station and any node with a single antenna can be expressed [80] as

$$\boldsymbol{h} = \sum_{i=1}^{N_c} \sum_{j=1}^{N_s} \boldsymbol{\alpha}_{ij} \boldsymbol{\beta}(\theta_{ij}), \tag{4.2}$$

where the number of scattering clusters and the number of sub-paths in each cluster can be presented by N_c and N_s, respectively, and $\boldsymbol{\alpha}_{ij}$ denotes the complex gain of the j-th propagation subpath in the i-th scattering cluster. The normalized array response at the UE can be represented by $\boldsymbol{\beta}(\theta_{ij})$, where the angle of arrival/departure of the j-th subpath in the i-th scattering cluster is denoted by θ_{ij}.

In case of ULA, we can model $\boldsymbol{\beta}(\theta_{ij})$ as

$$\boldsymbol{\beta}(\theta_{ij}) = \frac{1}{\sqrt{N_t}}[1,\ e^{jb\sin(\theta_{ij})},\ \ldots\ , e^{jb\sin(\theta_{ij})(N_t-1)}]^T, \tag{4.3}$$

where $b = 2\pi\frac{d}{\lambda_d}$, λ_d and d represent the propagation wavelength and antenna spacing, respectively. To this end, it is noted that the overall channel observed at a user with respect to different source nodes is different. Hence, it can be used to differentiate between different sources.

The proposed technique is based on a DL model. DL models use multiple hidden layers in the so-called deep architectures to achieve better data representation. Thanks to the use of deep layers, the intrinsic distinctive data aspects are magnified, whereas the irrelevant information is suppressed at each layer. This gives DL its greatest advantage; the ability to handle complex data streams in a rather simple manner [81].

1D-CNN has a simple and compact configuration making it feasible to be implemented with real-time and low-cost settings [82]. It is mainly used in data classification, anomaly detection, and recognition. Still, its application areas are increasing. CNN includes two stages, which are training and testing. Training includes adjusting convolutional layers where the feature vector is convolved with various filters for obtaining a convolved feature map. This can be given as

$$o = \sum_{p=1}^{u} w_p \boldsymbol{x}_k[p-1], \tag{4.4}$$

where w_p is the element at p-th row of the $u \times 1$ filter vector, and $\boldsymbol{x}_k$ denotes the elements of feature vector convolved by w_p.

4.3 The Proposed Technique for PUEA and JA Detection

The formal objective of the technique proposed in this chapter is to distinguish between the following hypotheses:

$$\boldsymbol{y} = \begin{cases} \boldsymbol{n}, & \mathcal{H}_0 : \text{there is no PU} \\ \boldsymbol{h}_{PU}\boldsymbol{x}_s + \boldsymbol{n}, & \mathcal{H}_1 : \text{a PU is present} \\ \boldsymbol{h}_i\boldsymbol{x}_s + \boldsymbol{n}, & \mathcal{H}_2 : \text{a PUE is present} \\ \boldsymbol{h}_i\boldsymbol{x}_n + \boldsymbol{n}, & \mathcal{H}_3 : \text{a jammer is present,} \end{cases} \tag{4.5}$$

where $\boldsymbol{y}$, $\boldsymbol{h}_{PU}$, $\boldsymbol{h}_i$, $\boldsymbol{x}_n$, and $\boldsymbol{x}_s$ denote the received signal, the channel corresponding to the legitimate PU, the channel corresponding to PUE or jammer, the (unstructured) jamming signal, and the (structured) signal, respectively. It is noted that the proposed technique can be set to differentiate PUEAs and JAs jointly. However, we opt to treat PUEA and JA detection as separate problems, as earlier suggested in [11].

4.3.1 The Proposed Technique

The proposed technique is composed of a training stage and a testing stage. The training stage is represented in Fig. 4.2-(a). As demonstrated in this figure, a set of received signals ($\boldsymbol{Y}_i$) and their corresponding classes ($\boldsymbol{c}_i$) are fed to DL to train the DL model. In this figure, $\boldsymbol{Y}_i$ ($\boldsymbol{c}_i$) consists of $\boldsymbol{y}_0$ (c_0), $\boldsymbol{y}_1$ (c_1), and $\boldsymbol{y}_2$ (c_2) for the case of PUEA detection, and $\boldsymbol{y}_0$ (c_0), $\boldsymbol{y}_1$ (c_1), and $\boldsymbol{y}_3$ (c_3) for the case of JA detection.

After classifier training, the testing stage represents the run-time operation of the proposed technique. This process is explained in Fig. 4.2-(b). In this stage, each incoming test signal is fed to the learned classifier. Then, the classifier decides on the hypothesis corresponding to the signal of interest.

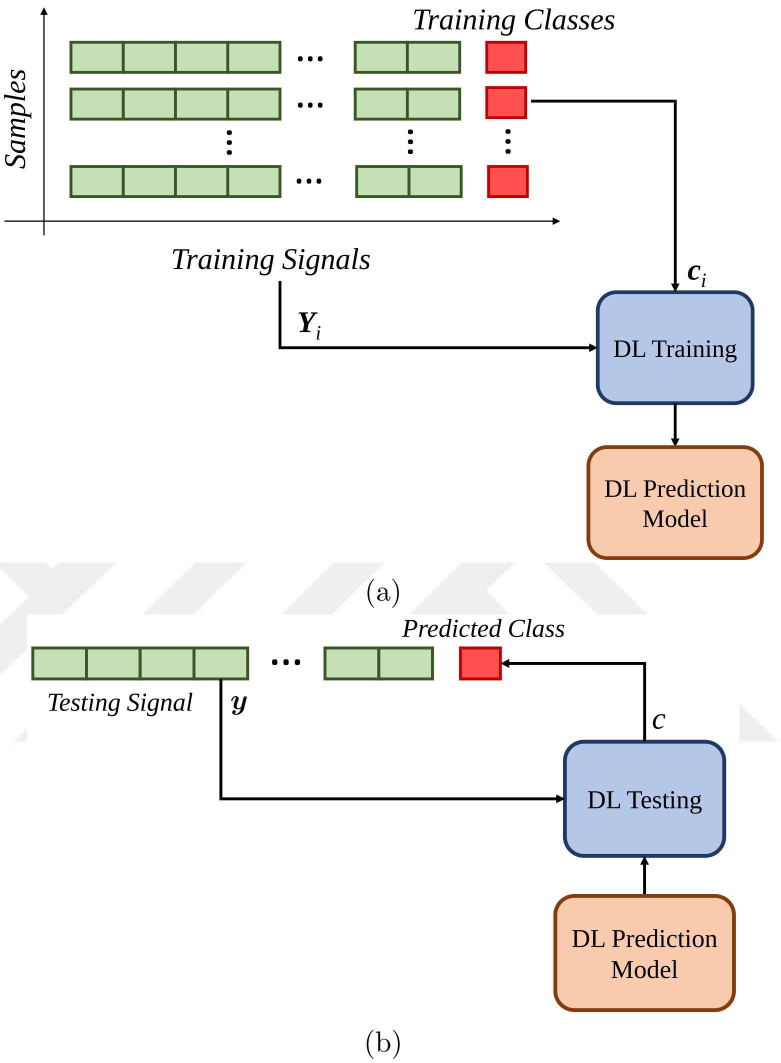


Figure 4.2: An illustration of the proposed technique's (a) training stage, (b) testing stage.

4.3.2 Computational Complexity Discussion

The computational complexity of the proposed technique can be roughly quantified in terms of the required execution time for training and testing in the adopted DL architecture. We use 1D-CNN networks due to their low computational requirements and the ability to learn challenging tasks with a relatively smaller number of hidden layers.

The theoretical time complexity for 1D-CNN [46] can be given as

$$\mathcal{O}(\sum_{l=1}^{b} q_{l-1}\rho_l^2 q_l m_l^2), \tag{4.6}$$

where the number of convolutional layers is represented by b, conventional layer index is represented by l, q_l represents the total number of filters used in l_{th} layer, the length of filter is represented by ρ_i, and the size of output is given by m_l. For the sake of simplicity, the time cost due to the fully connected layers is not involved in the formulation in (6) because they only take 5-10 % of the computational time.

The above-mentioned complexity formula applies to both training and testing stages, but they have different scales. The training stage per sample is roughly three times more complex as compared to the testing stage per sample because of one forward and two backward propagation [46].

4.4 Simulations and Results

4.4.1 Parameter Setting

The system presented in Section 4.2 is simulated under MATLAB environment. We assume a signal length of 100 with different data streams, and several modulation techniques. Namely, pulse amplitude modulation, phase-shift keying, quadrature amplitude modulation, and frequency-shift keying. Each received signal is obtained corresponding to a specific channel realization.

Channel coefficients are generated from PU and illegitimate nodes towards SU based on the GSCM channel model [80]. The considered cell consists of the multi-antenna primary node, multi-antenna illegitimate node, and a single antenna secondary node along with local and far scatterers. The angle of arrival (AoA) of the signals corresponding to local scatterers is dependent on the user location, while the AoA of the signals from far scatterers is fixed. Moreover, it is assumed that the AoA of the signals that are coming from local scatterers are concentrated around the location of the user.

Channel parameters are adjusted based on the spatial channel model presented in [83]. In this chapter, seven fixed location scattering clusters are assumed whose locations are randomly selected between 300 and 800 meters at the beginning of the simulation, and they are kept unchanged afterward. Four of the scatterers are considered as local scatters, while the remaining three are assumed to be far scatterers. Moreover, there are 20 effective sub-paths with a 4-degree angular spread in each cluster. Also, the values of azimuth angle, θ, range between $-\pi/2$ and $\pi/2$. Under the aforementioned assumptions, we generate channel realizations concerning PU and illegitimate node towards SU by considering them at different locations from SU.

The technique of [11], which is used for comparison purposes, is set to have a standard two-layer feed-forward network with a hidden layer and an output layer with sigmoid functions.

For the DL architecture, a 1D-CNN using Keras library in Python was employed. The CNN consists of 3 convolutional layers with a kernel size of 5. The number of units used in convolutional layers is 64, 64, and 32, respectively. In each hidden layer, rectified linear units (ReLU)s as activation functions are used. Then, the dense layer is used with 128 hidden units. Finally, the softmax activation function with 3 units is used to calculate the probability of each class in the output layer. Moreover, we employed adaptive moment estimation for adaptive learning rate optimization with a learning rate of 0.00009, and CNN is trained with 30 epochs. It should be noted that the aforementioned parameter is tuned empirically while considering the performance and generalizability of the proposed CNN model. The overall block diagram for the proposed CNN model is presented in Fig. 4.3.

In the training stage, 6000 samples are used from each class, and in the testing stage, 1000 samples are used from each class for each of the signal-to-noise ratio (SNR) values. SNR values within -5 dB to 15 dB with a step size of 5 dB are used

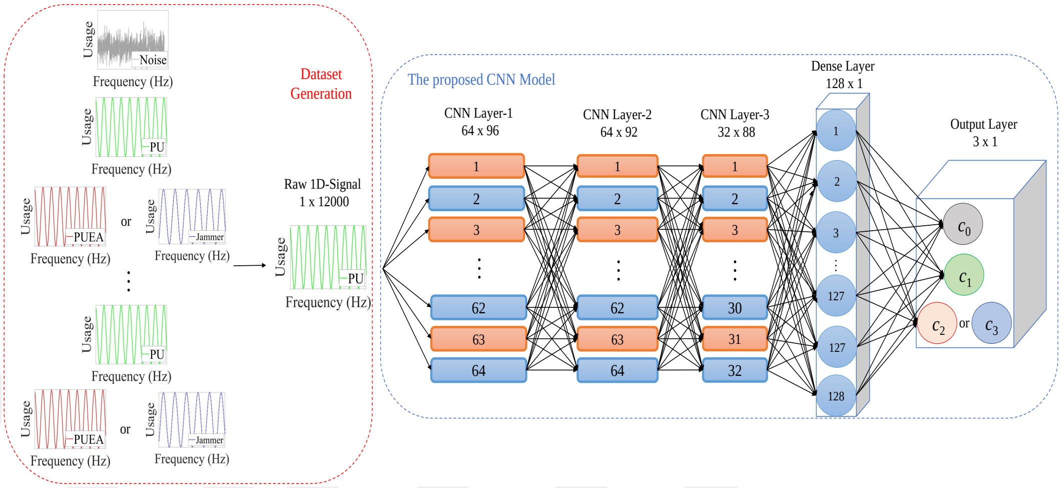


Figure 4.3: The proposed 1D-CNN model.

for the training of the neural network and DL model, in PUEA and JA detection.

4.4.2 Performance Evaluations and Discussions

We examine the performance of the proposed technique in terms of the receiver operating characteristic (ROC) quality metric. ROC is a curve that plots the true positive rate on the vertical axis versus the false positive rate. The capability of distinguishing for any model can be shown by ROC curves. The area under the receiver operating characteristics (AUROC) is also used as an additional quality metric.

For the case of JAs, the SNR of the non-structured signal is assumed to be equal to that of a legitimate PU. In the PUEA scenario, it is assumed that the PUE is extremely intelligent, and each signal of the PUE is considered to have precisely the same parameters with the PU signal. The channel that corresponds to the legitimate PU ($\boldsymbol{h}_{PU}$) is the only information unknown by the PUE.

Figures 4.4 and 4.5 show the performance comparison between ED as a benchmark technique, the technique of [11] (denoted by CS) as a state-of-the-art technique, and the proposed technique, respectively. It can be seen that

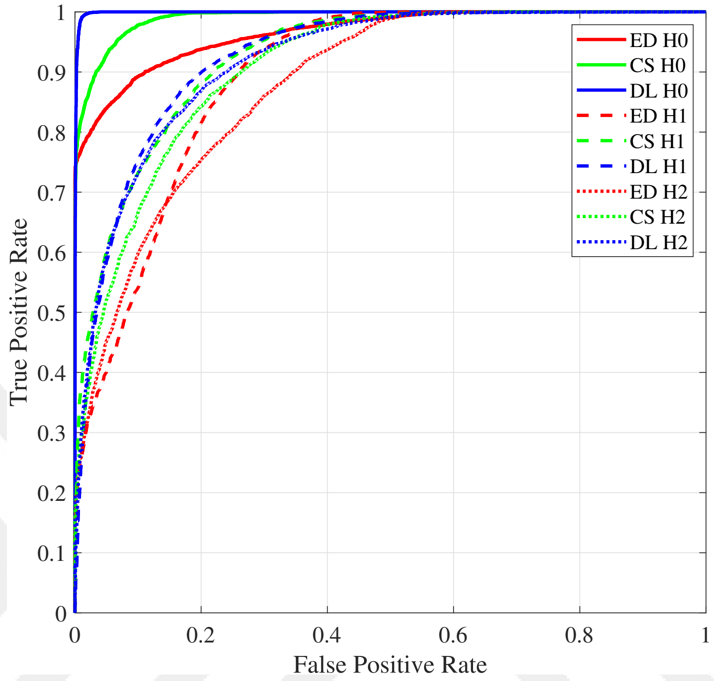


Figure 4.4: Comparison of the ED, CS, and DL-based techniques in PUEA detection.

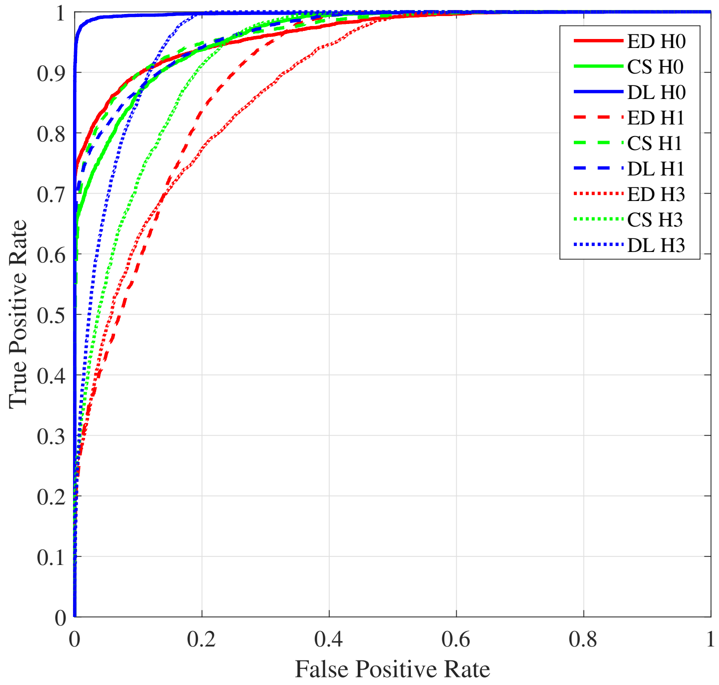


Figure 4.5: Comparison of the ED, CS, and DL-based techniques in JA detection.

the proposed technique is superior to the CS technique for both PUEA and JA. Besides, the proposed technique is more strongly superior compared to the ED technique. These results are also visible in terms of the AUROC values, as can be in Tables 4.1 and 4.2.

Table 4.1: AUROC values for PUEA detection by the proposed technique.

	PU	**Hole**	**PUEA**
ED	0.9192	0.8934	0.8784
CS	0.9901	0.9261	0.9084
DL	0.9989	0.9262	0.9192

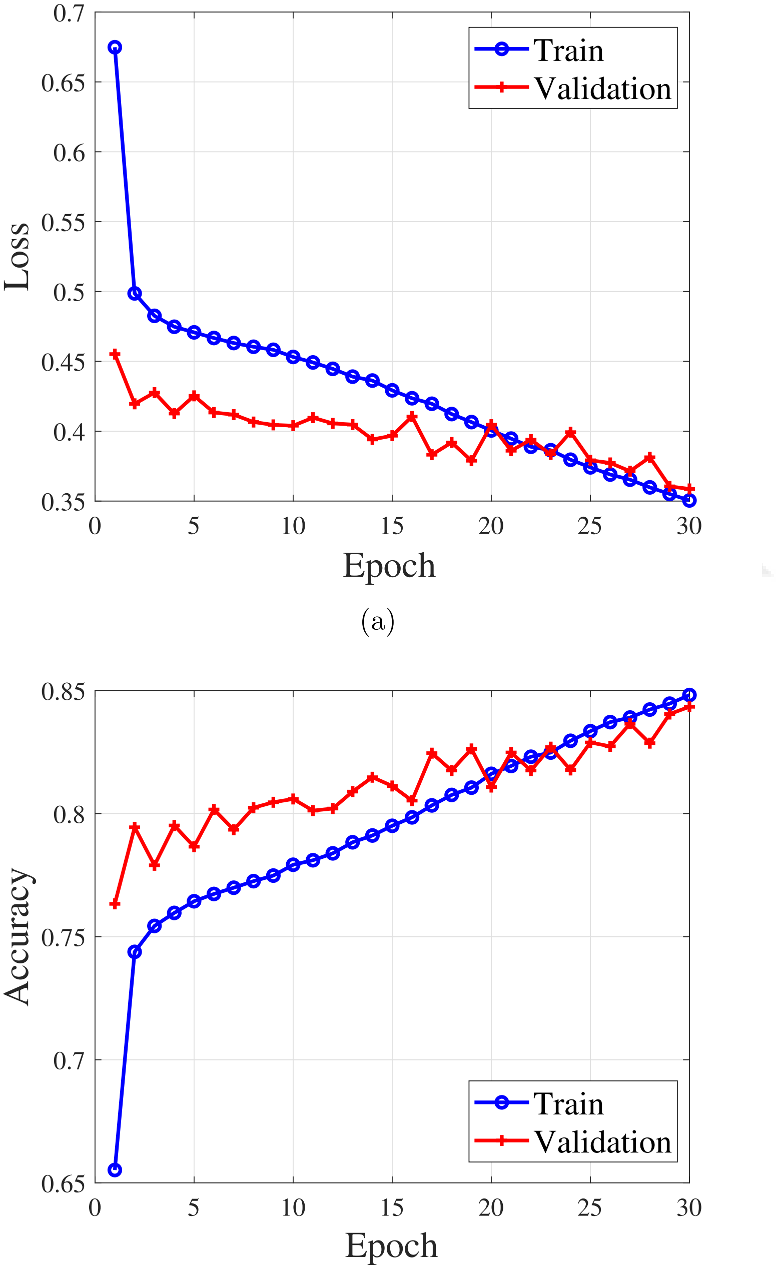
Table 4.2: AUROC values for JA detection by the proposed technique.

	PU	**Hole**	**JA**
ED	0.9654	0.9012	0.8866
CS	0.9620	0.9666	0.9305
DL	0.9973	0.9657	0.9582

For the PUEA detection, the training and testing losses and accuracies versus epochs are represented in Fig. 4.6. In these figures, it is shown that the accuracy of the training sets converges to that of the test set. By these results, it can be shown that there is no over-fitting, so the generalizability of the proposed model is realized. That means the training data is not memorized by the trained model. It is worth mentioning that the loss and accuracy graph is included only for the PUEA case to avoid repetition. Similar graphs are observed for JA detection, as well.

4.5 Conclusions

This chapter proposed using DL as means of PUEA and JA detection in CR systems. 1D-CNN was designed as a DL architecture for this purpose. The unique characteristics of this network alleviate the need for explicit feature extraction. Simulations validate that the proposed technique is superior to ML with feature extraction techniques. These results were validated in terms of ROC curves and values of AUROC curves, as quality metrics.



(a)

(b)

Figure 4.6: (a) Loss and (b) accuracy graphs for 1D-CNN.

Chapter 5

Future Directions

In this dissertation, contributions are made for spectrum occupancy prediction and physical layer security. Some future directions regarding these contributions are listed below.

- In Chapter 2, supervised learning-based techniques were used with the multidimensional datasets to predict spectrum occupancy. Although the successes of these techniques, they have some drawbacks; they require a large labeled dataset, they are sensitive to bias; they require retraining. Reinforcement learning can eliminate these drawbacks. Therefore, it can be used with the multidimensional dataset to predict spectrum occupancy as a future work.
- In Chapters 3 and 4, contributions were made for jamming, spoofing, and primary user attacks detection. To detect these attacks, the channel characteristics of legitimate users were used. As a future work, channel and radio-frequency characteristics can be used jointly with the proposed techniques. Also, these detection techniques can only identify the received signals whether they are legitimate or not (jammer, spoofer, primary user emulator). Prediction of these attacks is an important task that helps to consider what actions should be taken before they occur. Along this line, proposed techniques can be used to predict security attacks in future time slots as a future work.

Bibliography

[1] G. Ancans, V. Bobrovs, A. Ancans, and D. Kalibatiene, "Spectrum considerations for 5G mobile communication systems," *Procedia Comput. Sci.*, vol. 104, pp. 509–516, 2017.

[2] E. Dahlman, G. Mildh, S. Parkvall, J. Peisa, J. Sachs, Y. Selén, and J. Sköld, "5G wireless access: Requirements and realization," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 42–47, 2014.

[3] M. Amjad, M. H. Rehmani, and S. Mao, "Wireless multimedia cognitive radio networks: A comprehensive survey," *IEEE Commun. Surv. Tuts.*, vol. 20, pp. 1056–1103, Secondquarter 2018.

[4] X. Hong, J. Wang, C.-X. Wang, and J. Shi, "Cognitive radio in 5G: A perspective on energy-spectral efficiency trade-off," *IEEE Commun. Mag.*, vol. 52, no. 7, pp. 46–53, 2014.

[5] "5G vision." White Paper, Feb. 2015.

[6] M. H. Yılmaz and H. Arslan, "A survey: Spoofing attacks in physical layer security," in *Proc. IEEE 40th Local Comp. Net. Conf. (LCN)*, pp. 812–817, Oct 2015.

[7] A. G. Fragkiadakis, E. Z. Tragos, and I. G. Askoxylakis, "A survey on security threats and detection techniques in cognitive radio networks," *IEEE Commun. Surv. Tuts.*, vol. 15, no. 1, pp. 428–445, 2012.

[8] M. A. Aygül, M. Nazzal, A. R. Ekti, A. Görçin, D. B. da Costa, H. F. Ateş, and H. Arslan, "Spectrum occupancy prediction exploiting time and

frequency correlations through 2D-LSTM," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC2020-Spring)*, pp. 1–5, IEEE, 2020.

[9] M. A. Aygül, M. Nazzal, M. İ. Sağlam, D. B. da Costa, H. F. Ateş, and H. Arslan, "Efficient spectrum occupancy prediction exploiting multidimensional correlations through composite 2D-LSTM models," *Sensors*, vol. 21, p. 135, Jan. 2021.

[10] M. A. Aygül, S. Büyükçorak, D. B. da Costa, H. F. Ateş, and H. Arslan, "Signal relation-based physical layer authentication," in *IEEE Int. Conf. Commun. (ICC)*, pp. 1–6, IEEE, 2020.

[11] H. M. Furqan, M. A. Aygül, M. Nazzal, and H. Arslan, "Primary user emulation and jamming attack detection in cognitive radio via sparse coding," *EURASIP J. Wireless Commun. Netw.*, vol. 2020, no. 1, pp. 1–19, 2020.

[12] M. A. Aygül, H. M. Furqan, M. Nazzal, and H. Arslan, "Deep learning-assisted detection of pue and jamming attacks in cognitive radio systems," in *Proc. IEEE Veh. Technol. Conf. (VTC2020-Fall)*, IEEE, in press.

[13] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, pp. 201–220, Feb. 2005.

[14] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Commun. Surv. Tuts.*, vol. 11, pp. 116–130, First 2009.

[15] N. R. Banavathu and M. Z. A. Khan, "Optimization of n-out-of-k rule for heterogeneous cognitive radio networks," *IEEE Signal Process. Lett.*, vol. 26, pp. 445–449, March 2019.

[16] L. Arienzo and D. Tarchi, "Statistical modeling of spectrum sensing energy in multi-hop cognitive radio networks," *IEEE Signal Process. Lett.*, vol. 22, pp. 356–360, March 2015.

[17] Y. Arjoune and N. Kaabouch, "A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions," *Sensors*, vol. 19, no. 1, p. 126, 2019.

[18] L. Yu, J. Chen, G. Ding, Y. Tu, J. Yang, and J. Sun, “Spectrum prediction based on Taguchi method in deep learning with long short-term memory,” *IEEE Access*, vol. 6, pp. 45923–45933, 2018.

[19] M. H. Naikwadi and K. P. Patil, “A survey of artificial neural network based spectrum inference for occupancy prediction in cognitive radio networks,” in *Proc. 4th Int. Conf. Trends Electron. Inf. (ICOEI)*, pp. 903–908, IEEE, 2020.

[20] Z. Lin, X. Jiang, L. Huang, and Y. Yao, “A energy prediction based spectrum sensing approach for cognitive radio networks,” in *Proc. 5th Int. Conf. Wireless Commun., Netw. Mobile Comput. (WICOM)*, pp. 1–4, IEEE, 2009.

[21] Z. Wen, T. Luo, W. Xiang, S. Majhi, and Y. Ma, “Autoregressive spectrum hole prediction model for cognitive radio systems,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 154–157, May 2008.

[22] X. Xing, T. Jing, Y. Huo, H. Li, and X. Cheng, “Channel quality prediction based on bayesian inference in cognitive radio networks,” in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, pp. 1465–1473, Apr. 2013.

[23] V. K. Tumuluru, P. Wang, and D. Niyato, “Channel status prediction for cognitive radio networks,” *Wireless Commun. Mob. Comput.*, vol. 12, no. 10, pp. 862–874, 2012.

[24] A. A. Eltholth, “Spectrum prediction in cognitive radio systems using a wavelet neural network,” in *Proc. 24th Int. Conf. on Software, Telecommun. Comp. Net. (SoftCOM)*, pp. 1–6, IEEE, 2016.

[25] G. Ding, Y. Jiao, J. Wang, Y. Zou, Q. Wu, Y.-D. Yao, and L. Hanzo, “Spectrum inference in cognitive radio networks: Algorithms and applications,” *IEEE Commun. Surv. Tuts.*, vol. 20, no. 1, pp. 150–182, 2017.

[26] X. Chen, H. Zhang, A. B. MacKenzie, and M. Matinmikko, “Predicting spectrum occupancies using a non-stationary hidden Markov model,” *IEEE Wireless Commun. Lett.*, vol. 3, no. 4, pp. 333–336, 2014.

[27] A. Selim, F. Paisana, J. A. Arokkiam, Y. Zhang, L. Doyle, and L. A. DaSilva, “Spectrum monitoring for radar bands using deep convolutional neural networks,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, IEEE, 2017.

[28] K. Davaslioglu, S. Soltani, T. Erpek, and Y. Sagduyu, “DeepWiFi: Cognitive WiFi with deep learning,” *IEEE Trans. Mobile Comput.*, 2019.

[29] L. Yu, J. Chen, and G. Ding, “Spectrum prediction via long short term memory,” in *Proc. 3rd IEEE Int. Conf. Comp. and Commun. (ICCC)*, pp. 643–647, IEEE, 2017.

[30] O. Omotere, J. Fuller, L. Qian, and Z. Han, “Spectrum occupancy prediction in coexisting wireless systems using deep learning,” in *Proc. IEEE 88th Veh. Technol. Conf. (VTC-Fall)*, pp. 1–7, IEEE, Aug 2018.

[31] A. M. Hisham and H. Arslan, “Multidimensional signal analysis and measurements for cognitive radio systems,” in *2008 IEEE Radio and Wireless Symposium*, pp. 639–642, IEEE, 2008.

[32] J. Sun, J. Wang, G. Ding, L. Shen, J. Yang, Q. Wu, and L. Yu, “Long-term spectrum state prediction: An image inference perspective,” *IEEE Access*, vol. 6, pp. 43489–43498, 2018.

[33] I. Alkhouri, M. Joneidi, F. Hejazi, and N. Rahnavard, “Large-scale spectrum occupancy learning via tensor decomposition and LSTM networks,” in *Proc. IEEE Int. Radar Conf. (RADAR)*, pp. 677–682, IEEE, 2020.

[34] B. S. Shawel, D. H. Woldegebreal, and S. Pollin, “Convolutional LSTM-based long-term spectrum prediction for dynamic spectrum access,” in *Proc. 27th European Signal Process. Conf. (EUSIPCO)*, pp. 1–5, IEEE, 2019.

[35] A. Ioannidou, E. Chatzilari, S. Nikolopoulos, and I. Kompatsiaris, “Deep learning advances in computer vision with 3D data: A survey,” *ACM Comput. Surv. (CSUR)*, vol. 50, no. 2, pp. 1–38, 2017.

[36] A. S. Gezawa, Y. Zhang, Q. Wang, and L. Yunqi, “A review on deep learning approaches for 3D data representations in retrieval and classifications,” *IEEE Access*, vol. 8, pp. 57566–57593, 2020.

[37] S. Bassoy, *Self-organised multi-objective network clustering for coordinated communications in future wireless networks.* PhD thesis, University of Glasgow, 2020.

[38] K. Jaqaman, D. Loerke, M. Mettlen, H. Kuwata, S. Grinstein, S. L. Schmid, and G. Danuser, “Robust single-particle tracking in live-cell time-lapse sequences,” *Nature methods*, vol. 5, no. 8, pp. 695–702, 2008.

[39] B. Khalfi, B. Hamdaoui, M. Guizani, and N. Zorba, “Efficient spectrum availability information recovery for wideband DSA networks: A weighted compressive sampling approach,” *IEEE Trans. Wireless Commun.*, vol. 17, pp. 2162–2172, Apr. 2018.

[40] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, “Handwritten digit recognition with a back-propagation network,” in *Advances in Neural Information Process. Systems*, pp. 396–404, 1990.

[41] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[42] K. S. Tai, R. Socher, and C. D. Manning, “Improved semantic representations from tree-structured long short-term memory networks,” *arXiv preprint arXiv:1503.00075*, 2015.

[43] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, “Convolutional lstm network: A machine learning approach for precipitation nowcasting,” in *Advances Neural Inf. Process. Systems*, pp. 802–810, 2015.

[44] H. Sak, A. Senior, and F. Beaufays, “Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition,” *arXiv preprint arXiv:1402.1128*, 2014.

[45] H. Sak, A. W. Senior, and F. Beaufays, “Long short-term memory recurrent neural network architectures for large scale acoustic modeling,” *Proc. Annual Conf. Int. Speech Commun. Assoc., (INTERSPEECH)*, pp. 338–342, Sep. 2014.

[46] K. He and J. Sun, “Convolutional neural networks at constrained time cost,” in *Proc. of the IEEE Conf. Comput. Vision Pattern Recognition*, pp. 5353–5360, 2015.

[47] 3rd Generation Partnership Project (3GPP), *Evolved Universal Terrestrial Radio Access (E-UTRA); (Release 16)* , June 2020.

[48] K. A. Qaraqe, H. Celebi, A. Gorcin, A. El-Saigh, H. Arslan, and M. Alouini, “Empirical results for wideband multidimensional spectrum usage,” in *Proc. IEEE 20th Int. Symposium Personal, Indoor Mobile Radio Commun. (PIMRC)*, pp. 1262–1266, Sep. 2009.

[49] G. Zhang, J. Wang, and Y. Xiang, “Method, apparatus and system for spectrum prediction,” Apr. 2014. US Patent 8,687,516.

[50] Comba, *Outdoor directional quad-band antenna (ODI-065R17M18JJJ-G).*

[51] Arimas, “LTE guard band calculation.” https://arimas.com/lte-guard-band-calculation/. Accessed: 2020-08-17.

[52] F. Chollet, “Keras.” https://keras.io. Accessed: 2020-08-14.

[53] B. D. M.H. Beale, T. Hagan, “Neural network toolbox™ 7.” nnet.pdf, Sept. 2010.

[54] X. Wang, P. Hao, and L. Hanzo, “Physical-layer authentication for wireless security enhancement: Current challenges and future developments,” *IEEE Commun. Mag.*, vol. 54, pp. 152–158, June 2016.

[55] Y. Shiu, S. Y. Chang, H. Wu, S. C. Huang, and H. Chen, “Physical layer security in wireless networks: A tutorial,” *IEEE Wireless Commun.*, vol. 18, pp. 66–74, Apr. 2011.

[56] Y. Liu, H. Chen, and L. Wang, "Physical layer security for next generation wireless networks: Theories, technologies, and challenges," *IEEE Commun. Surv. Tuts.*, vol. 19, pp. 347–376, Firstquarter 2017.

[57] L. Xiao, L. Greenstein, N. Mandayam, and W. Trappe, "Fingerprints in the ether: Using the physical layer for wireless authentication," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 4646–4651, Jun. 2007.

[58] L. Xiao, L. J. Greenstein, N. B. Mandayam, and W. Trappe, "Using the physical layer for wireless authentication in time-variant channels," *IEEE Trans. Wireless Commun*, vol. 7, pp. 2571–2579, Jul. 2008.

[59] L. Xiao, L. Greenstein, N. B. Mandayam, and W. Trappe, "Channel-based spoofing detection in frequency-selective Rayleigh channels," *IEEE Trans. Wireless Commun*, vol. 8, pp. 2571–2579, Dec. 2009.

[60] J. K. Tugnait and H. Kim, "A channel-based hypothesis testing approach to enhance user authentication in wireless networks," in *Proc. IEEE Int. Conf. Commun. Syst. Netw. (COMSNETS)*, pp. 1–9, Jan. 2010.

[61] F. J. Liu, Xianbin Wang, and H. Tang, "Robust physical layer authentication using inherent properties of channel impulse response," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, pp. 538–542, Nov 2011.

[62] C. Pei, N. Zhang, X. S. Shen, and J. W. Mark, "Channel-based physical layer authentication," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 4114–4119, Dec 2014.

[63] F. Pan, H. Wen, R. Liao, Y. Jiang, A. Xu, K. Ouyang, and X. Zhu, "Physical layer authentication based on channel information and machine learning," in *Proc. IEEE Conf. Commun. Net. Security (CNS)*, pp. 364–365, Oct 2017.

[64] R.-F. Liao, H. Wen, J. Wu, F. Pan, A. Xu, Y. Jiang, F. Xie, and M. Cao, "Deep-learning-based physical layer authentication for industrial wireless sensor networks," *Sensors*, vol. 19, p. 2440, 05 2019.

[65] R. Chen, C. Li, S. Yan, R. A. Malaney, and J. Yuan, "Physical layer security for ultra-reliable and low-latency communications," *CoRR*, vol. abs/1906.08443, 2019.

[66] B. Zhou and Q. Chen, "A tutorial on minimum mean square error estimation," *Southwest Jiaotong Univ., Sichuan, China, Tech. Rep*, 2015.

[67] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, "Deep learning with long short-term memory for time series prediction," *IEEE Commun. Mag.*, vol. 57, no. 6, pp. 114–119, 2019.

[68] H. Arslan and G. E. Bottomley, "Channel estimation in narrowband wireless communication systems," *Wireless Commun. Mob. Comput.*, vol. 1, no. 2, pp. 201–219, 2001.

[69] R. A. Ziegler and J. M. Cioffi, "Estimation of time-varying digital radio channels," *IEEE Trans. Veh. Technol.*, vol. 41, pp. 134–151, May 1992.

[70] H. M. Furqan, J. M. Hamamreh, and H. Arslan, "Adaptive OFDM-IM for enhancing physical layer security and spectral efficiency of future wireless networks," *Wireless Commun. Mob. Comput.*, vol. 2018, 2018.

[71] F. Jin, V. Varadharajan, and U. Tupakula, "Improved detection of primary user emulation attacks in cognitive radio networks," in *Proc. IEEE Int. Telecommun. Net. Applications Conf. (ITNAC)*, pp. 274–279, Nov. 2015.

[72] Z. Yuan, D. Niyato, H. Li, J. B. Song, and Z. Han, "Defeating primary user emulation attacks using belief propagation in cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 30, pp. 1850–1860, Nov. 2012.

[73] Y. Wu, A. Khisti, C. Xiao, G. Caire, K. Wong, and X. Gao, "A survey of physical layer security techniques for 5G wireless networks and challenges ahead," *IEEE J. Sel. Areas Commun.*, vol. 36, pp. 679–695, Apr. 2018.

[74] D. Pu and A. M. Wyglinski, "Primary-user emulation detection using database-assisted frequency-domain action recognition," *IEEE Trans. Veh. Technol.*, vol. 63, pp. 4372–4382, Nov. 2014.

[75] S. Arul Selvi and M. Sundararajan, "SVM based two level authentication for primary user emulation attack detection," *Indian J. Sci. Technol.*, vol. 9, 2016.

[76] S. U. Rehman, K. W. Sowerby, and C. Coghill, "Radio-frequency fingerprinting for mitigating primary user emulation attack in low-end cognitive radios," *IEEE Institution Engr. Technol. (IET) Commun.*, vol. 8, no. 8, pp. 1274–1284, 2014.

[77] L. Robert, "Camera calibration without feature extraction," *Computer Vision Image Understanding*, vol. 63, no. 2, pp. 314–325, 1996.

[78] A. K. Sangaiah, *Deep Learning and Parallel Computing Environment for Bioengineering Systems.* Academic Press, 2019.

[79] J. M. Hamamreh, H. M. Furqan, and H. Arslan, "Classifications and applications of physical layer security techniques for confidentiality: A comprehensive survey," *IEEE Commun. Surv. Tuts.*, vol. 21, pp. 1773–1828, Secondquarter 2019.

[80] A. F. Molisch, A. Kuchar, J. Laurila, K. Hugl, and R. Schmalenberger, "Geometry-based directional model for mobile radio channels-principles and implementation," *European Trans. Telecommun.*, vol. 14, no. 4, pp. 351–359, 2003.

[81] K. Qureshi, *Cutting-Edge Evolutions of Information Technology: Artificial intelligence Machine Learning.* Booksclinic Publishing; 3 edition, 2019.

[82] S. Kiranyaz, T. Ince, O. Abdeljaber, O. Avci, and M. Gabbouj, "1-D convolutional neural networks for signal processing applications," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP)*, pp. 8360–8364, IEEE, 2019.

[83] J. Salo *et al.*, "Universal mobile telecommunications system (UMTS); spatial channel model for multiple input multiple output (MIMO) simulations," *Eur. Telecommun. Standards Inst., Nice, French, Tech. Rep. GT*, vol. 25, 2010.

DEVELOPING NOVEL SPECTRUM OCCUPANCY PREDICTION AND PHYSICAL LAYER SECURITY TECHNIQUES

ORIGINALITY REPORT

15%	10%	10%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

MATCH ALL SOURCES (ONLY SELECTED SOURCE PRINTED)

2%

★ mafiadoc.com

Internet Source

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		