This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

Shared Spectrum Monitoring using Deep Learning

Farrukh A. Bhatti, M. Jaleed Khan, Ahmed Selim, *Member, IEEE* and Francisco Paisana, *Member, IEEE*

***Abstract*—Shared spectrum usage is inevitable due to the ongoing increase in wireless services and bandwidth requirements. Spectrum monitoring is a key enabler for efficient spectrum sharing by multiple radio access technologies (RATs). In this paper, we present signal classification using deep neural networks to identify various radio technologies and their associated interferences. We use Convolutional Neural Networks (CNN) to perform signal classification and employ six well-known CNN models to train for ten signal classes. These classes include LTE, Radar, WiFi and FBMC (Filter Bank Multicarrier) and their interference combinations, which include, LTE+Radar, LTE+WiFi, FBMC+Radar, FBMC+WiFi, WiFi+Radar and Noise. The CNN models include, AlexNet, VGG16, ResNet18, SqueezeNet, InceptionV3 and ResNet50. The radio signal data sets for training and testing of CNN-based classifiers are acquired using a USRP-based experimental setup. Extensive measurements of these radio technologies (LTE, WiFi, Radar and FBMC) are done over different locations and times to generate a robust dataset. We propose a novel (spectrogram) representation called the Quarter-spectrogram (Q-spectrogram) that squeezes temporal and frequency information for input to CNN models. While considering classification accuracy, model complexity and prediction time for a single input Q-spectrogram (image), ResNet18 (CNN model) gives the best overall performance with 98% classification accuracy. While SqueezeNet (CNN model) offers the lowest model complexity which makes it very suitable for resource-constrained radio monitoring devices and also offers the least prediction time of 110 msec. Moreover, we also propose a simple WiFi classification scheme that buffers several WiFi Q-spectrograms and then makes a decision about WiFi's presence and also gives a quantified measure of WiFi traffic density.**

***Index Terms*—Spectrum monitoring, spectrum sharing, convolutional neural network, deep learning, radio access technologies classification, interference classification, heterogeneous networks.**

I. Introduction

Shared spectrum usage has enormous potential for delivering higher data rates in the rapidly evolving radio landscape with severe spectrum congestion. Although shared spectrum usage is a rather old concept in the context of cognitive radio but it never invoked interest of equipment manufacturers or operators despite having regulator's (FCC) backing. It is only in the recent half-decade that equipment manufacturers (Nokia, Qualcomm etc.) and operators (Verizon, AT&T etc.) have focused their efforts on shared spectrum technologies, such as, LTE-U (LTE-Unlicensed) and CBRS (Citizens Broadband Radio Service), to exploit their commercial potential.

There are three primary frequency bands that are designated for unlicensed operation worldwide, i.e., 900 MHz, 2.4 GHz and 5 GHz, of these the later two are used for spectrum sharing. Moreover, 3.5 GHz (CBRS) band is being opened up for shared usage in USA. Several wireless technologies are emerging that rely on spectrum sharing - a brief overview of these is presented in Section III. Interference management and fair coexistence are the major concerns in spectrum sharing. For instance, LTE-U devices in the 5 GHz band may interfere with the terrestrial radar systems (both military and weather) that operate in many regions including Europe and North America. Therefore, detecting and avoiding interference to and from radars is critical for efficient utilization of this shared band and radio spectrum monitoring is vital for a distributed channel management scheme.

Spectrum monitoring comprises three main aspects; 1) modulation recognition, 2) wireless technology (or RAT) identification and 3) wireless interference identification. Modulation recognition is a well explored research area [1], where the receiver differentiates between various modulation schemes, such as, M-Phase Shift Keying (M-PSK) or M-Quadrature Amplitude Modulation (M-QAM), etc., where M is the modulation order. Wireless technology identification refers to identification of technologies, such as, LTE, WiFi etc. Wireless interference idenfication is the task of identifying the co-existing wireless technologies that operate at the same carrier frequency. In the context of spectrum monitoring, modulation recognition is not explicitly required because the RATs including WiFi, LTE, Zigbee etc., utilize multi-carrier modulation schemes like Orthogonal Frequency Division Multiplexing (OFDM), that are modulated by M-PSK or M-QAM. Therefore, a spectrum monitoring device would perceive an OFDM signal rather than the underlying modulation scheme, such as, M-QAM. For the spectrum sharing scenarios in heterogeneous networks, RAT idenfication and interference idenfication have great significance and this is the focus of our paper.

Coexistence with WiFi is an active area of research to leverage additional spectrum for cellular or other services (more details in section III). Different operators/services may compete for this unlicensed spectrum. However, ensuring fairness with WiFi users is a prime concern. Knowing WiFi traffic density is very helpful for the contenders, as it gives an estimate of how much wireless capacity can be leveraged from WiFi spectrum. In case WiFi traffic density is very high, then it may not even be worth sharing this spectrum while maintaining fairness.

In this work we use CNN to classify different radio technologies, i.e., Radar, LTE, WiFi and FBMC, also called the

Farrukh A. Bhatti is with WiSP Lab, Department of Electrical Engineering, Institute of Space Technology, Islamabad, Pakistan, e-mail: farrukh.aziz@ist.edu.pk.

M. Jaleed Khan is with iVISION Lab, Electrical Engineering Department, Institute of Space Technology, Pakistan, e-mail: mjk093@gmail.com.

Ahmed Selim is with CONNECT Research Centre, Trinity College Dublin, Ireland, e-mail: selimam@tcd.ie.

Francisco Paisana is with Software Radio Systems, Dublin, Ireland, e-mail: francisco.paisana@softwareradiosystems.com.

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

single classes. Moreover, we also classify different interference combinations of these radio technologies, i.e., LTE+Radar, LTE+WiFi, FBMC+Radar, FBMC+WiFi and WiFi+Radar, also called the mixed classes. We also classify Noise as a distinct class, which indicates the absence of all radio technologies or their interferences. Moreover, we present a simple scheme to distinguish between pure noise and the intermittent presence of noise in a WiFi transmission. We have also included FBMC as it is a candidate waveform for future 5G networks and it may be used in scenarios where higher throughput is required with lower adjacent channel interference and where higher complexity can be afforded. We have considered all possible interference scenarios of these radio technologies, as signified by the discussion in section III, except for LTE+FBMC which is an unlikely scenario.

Contributions

The specific contributions of this paper are as follows.

- We propose a novel technique for representing a wireless signal as an input image for CNN. This is called a Quarter-Spectrogram, which can squeeze more information along time and frequency as compared to a typical spectrogram. This is particularly useful for WiFi signals, whose traffic density estimation requires greater information.
- We perform an elaborate comparison of the well-known CNN models (commonly used in computer science applications) using an extensive dataset of over-the-air wireless signals captured at two different locations. CNN-based radio signal classification has been done by several researchers, however, this is the first time that a comprehensive comparison has been performed while considering important CNN performance metrics, i.e., classification accuracy, model complexity and prediction time, that are critical for their practical use in wireless systems.
- For WiFi signal we devise a simple scheme for classification and its traffic estimation. Several technologies rely on coexistence with WiFi, therefore, knowing the WiFi traffic density is very helpful for efficient and fair spectrum sharing in such scenarios.

This paper is organized as follows. Section II presents the related work and section III gives an overview of the technologies that utilize unlicensed band. System model is described in section IV, experimental setup is explained in section V and CNN is discussed in section VI. Results are presented in section VII and finally conclusions are drawn in section VIII.

II. Related Work

Several recent studies have considered radio signal classification using machine learning techniques, especially CNNs. In [2], modulation classification is performed on 8 modulation schemes using a convoloutional long short-term deep neural network (CLDNN) while using a simulated (synthetic) dataset [3]. However, they do not consider wireless technology identification or interference identification. In [4], modulation recognition has been done on simulated signals using a CNN model having a few layers. In [5], 24 modulation schemes are classified while directly feeding (In-phase & Quadrature) IQ samples to CNN and residual neural network. It achieves good performance on high SNR over-the-air datasets while initially training on simulated dataset. However, this work only deals with modulation recognition and does not consider wireless technology identification or interference identification. In [6], modulation recognition and wireless technology identification has been done using a CNN2 (proposed in [4]). However, separate CNN models were trained for modulation recognition (eight schemes) and wireless technology identification (three technologies). In [7], received signal strength indicator (RSSI) of four wireless technologies is used for classification. It is argued that RSSI is related to a signal's modulation scheme and MAC and it therefore helps in signal classification even at sub-Nyquist sampling rate. Although this scheme is simple and does not rely on machine learning, but it is ineffective for the case of interfering signals as the RSSI distribution gets overwhelmed and no longer remain distinctive.

In [8], the authors have classified four wireless technologies, i.e., IEEE 802.11b/g/n, 802.15.4, 802.15.1 and Bluetooth Low Energy, using supervised learning (SL) techniques for implementation in commercial off-the-shelf (COTS) IoT devices. The SL techniques include, Classification Trees, Random Forest of Classification Trees and Multi-class Support Vector Machine. This work performs interference identification with low sensing times ($\sim 400\mu s$) but at relatively higher levels of interference to noise ratio (INR), i.e., at INR $>$ 0dB. RSSI-based traces with CNN are also used in [9] to classify WiFi beacon, different classes of WLAN traffic, BLE iBeacon and microwave oven. Although the authors term these as interference signals, however, they use individual signals only. In [10], Long Short Term Memory (LSTM) CNN is used for modulation recognition while deep learning is used for technology classification in a distributed sensor network. However, interference identification is not considered in this work. In [11], CNN has been used to identify radar signals in the presence of WiFi and LTE. However, they do not identify the underlying interference.

In their work on wireless technology classification [12], Fontaine *et al.* have classified LTE, WiFi and DVB-T technologies using decision trees, neural networks and CNN. They use different features including RSSI, IQ samples and spectrograms. The overall best performance is exhibited by IQ-based CNN, however, it is also computationally intensive. This work only considers three classes and does not consider interference classification. In [13], CNN has been used to classify LTE, WiFi, mLTE-U (LTE with muting, proposed in [14]) and LTE+WiFi while using IQ-based and FFT-based spectrograms. This work considers three technologies and one interference signal and still gives inferior results than our work (described in results section).

Henarejos *et al.* [15] used LSTM for interference classification while considering DVB-S2 as the base technology and adding different interferences to it, i.e., LTE, UMTS and GSM. Their results show interference at different signal-to-interference (SIR) levels, however, they do not vary the SNR of the base technology which is equally critical. In [16], technology classification is done while considering non-standardized

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

technologies that differ in their MAC. A semi-supervised learning (SSL) is performed with deep auto-encoder and the results are compared with a deep neural network where the former achieves 70% accuracy using 10% data. The authors claim that the approach is useful for unknown environments and helps in efficient labeling of the data. In [17], WCDMA, LTE and GSM technologies have been classified using Spectral Correlation Function (SCF) and Support Vector Machine (SVM). The authors use simulated data to analyze performance at lower SNRs. Schepper *et al.* [18] utilize deep learning to estimate WiFi traffic pattern using a synthetic dataset. They classify the traffic while considering the transport protocol, traffic pattern w.r.t. duty cycle, i.e., 25%, 50% and 75% and the data rate. They train separate CNN models for estimating these three traffic properties. However, their results do not indicate performance w.r.t. SNR and they do not consider interference either.

III. Technologies Utilizing Unlicensed Spectrum

This section presents an overview of the present and upcoming technologies that rely on shared spectrum access in 3.5 GHz or 5 GHz unlicensed spectrum. It signifies the need for radio technology identification and interference classification for such scenarios.

LTE-U : The envisioned LTE-Unlicensed (LTE-U) works by augmenting a carrier's LTE service by employing the unlicensed spectrum in the 5 GHz band. To mitigate collisions with the WiFi users, it uses dynamic channel selection and adaptive duty cycling of LTE. It does not require LBT (listen-before-talk) before accessing channel every time. Qualcomm's CSAT (Carrier Sensitive Adaptive Transmission) technique is a well-known LTE-U implementation. This may end up being used in the regions where LBT is not mandatory, such as, US, India and South-Korea.

LAA : The LAA (License Assisted Access) requires LBT implementation to ensure fair coexistence with the WiFi; this is a regulatory requirement in several regions including Europe and Japan. Since this is a shared band without exclusive rights, fair coexistence with other technologies, especially WiFi, is of fundamental concern. This widely debated issue has resulted in quite a few variants of coexistence technologies that can suit the specific regulatory requirements of different regions. In [19], it is stated a WiFi node should perform LBT using Clear Channel Assessment (CCA) and determine the availability of the spectrum band, i.e., whether the channel is clean or already occupied by other operators or radar.

LWA : LWA stands for LTE-WLAN Aggregation. Unlike LTE-U and LAA, which require specific hardware/software implementation for unlicensed LTE transmission to coexist with WiFi, LWA configures the network to utilize both LTE and WiFi networks simultaneously. The LWA technology can split a single data flow while accounting for both channel conditions to use either LTE, WiFi or both for all applications. Its major difference with respect to LTE-U and LAA is that in this case the LTE does not compete with WiFi, rather the WLAN connections seamlessly increase the capacity while utilizing the best available combination. For the network operator, LWA also reduces network deployment costs as compared to LTE-U and LAA.

MulteFire: This technology operates entirely in the unlicensed band. This allows convenient deployment by anyone anywhere similar to WiFi and it doesn't require the licensed spectrum as an anchoring service. End users install MulteFire access points in lieu of WiFi access points to provide LTE coverage. It uses a protocol similar to LAA that also conforms to LBT.

CBRS : Citizen Broadband Radio Service (CBRS) has been adopted by FCC in USA for shared wireless broadband in 3550-3700 MHz band. It utilizes a Spectrum Access System (SAS) which is a cloud based highly automated database and controller that coordinates the frequency usage in the CBRS spectrum. This is a three-tiered service that includes Incumbent Access (IA), Priority Access License (PAL) and General Authorized Access (GAA). The IA tier (includes U.S. Navy radars and fixed satellite earth stations) has priority over the other two tiers, hence, ideally it should not receive interference from the lower two tiers. Similarly, PAL has priority over GAA. It is envisaged that the PAL tier could include private LTE networks, also known as Mobile Virtual Network Operators (MVNOs). They would be given 10 MHz licenses within 100 MHz of the total 150 MHz spectrum. This could also be used for carrier aggregation by mainstream mobile network operators. The GAA tier can access 80 MHz spectrum within the CBRS band without requiring a license and while maintaining the lowest priority. For GAA services the key requirement before actual deployment is the Environmental Sensing Capability (ESC) that allows a device to monitor other radio signals at its location, such as the naval radars. For inland GAA users naval radars is not an issue but for the large number of users inhabiting the coastal areas including densely populated metropolitan, such as, New York, Boston and Los Angeles etc., this is an extremely important issue. ESC systems detect and communicate the presence of an incumbent user to the SAS to facilitate shared spectrum access within the 3.5 GHz band without causing interference. Recently in Oct 2018, FCC modified key points of the CBRS licensing regime established in 2015, which also includes expanding the size of the license areas from census tract to counties [20]. This will further aid in implementation of CBRS band for a variety of use cases including 5G-capable mobile and fixed wireless applications and industrial IoT applications.

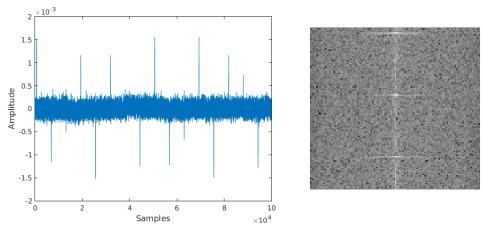
5G NR-SS: 3GPP has recently completed the first 5G NR (New Radio) standard. Following this, the work to introduce shared/unlicensed spectrum has also started which is called 5G NR-SS (5G New Radio-Shared Spectrum) [21]. The initial work is focusing on creating an NR-based LAA and an NR-based 5G MulteFire. Several advancements are expected in its development, including, support for a flexible framework to cope with more complicated spectrum sharing situations and spectrum monitoring capability.

Interference Identification in Heterogeneous Environments

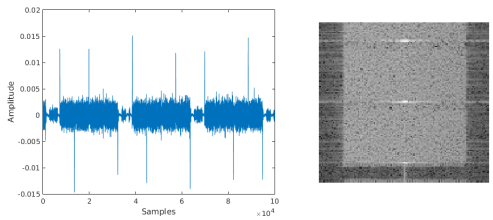
The above mentioned deployment scenarios clearly indicate the importance of sensing various types of signals including interference estimation in heterogeneous environments. A

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking



(a) Radar signal.



(b) Radar and LTE signals.

Fig. 1. Time domain signals with corresponding spectrograms.

conventional sensing scheme such as energy detection has limitations in detecting radar pulses in the heterogeneous environments mentioned above. Radar pulses appear for a very short duration of time, thus their total energy over an observation interval is quite low, especially if the received signal level is low to begin with, as is often the case.

Signal detection using conventional methods, such as energy detection, is difficult in heterogeneous environments where different signals might potentially overlap in time and frequency. This is a likely possibility in the above mentioned scenarios, where signals such as LTE may try to coexist with radars operating in shared spectrum. For such systems it is very important to classify the signals for better assessment of the radio channel. Knowing about the interference can significantly help in improved sharing of the spectrum. Fig. 1 shows a time-domain radar signal along with its spectrogram, where the x-axis of the spectrogram represents the frequency and the y-axis represents the time. If energy detection is employed in this case, then the energy of a few radar pulses would submerge in the overall signal. For example in Fig. 1a, even though the amplitude of the radar pulses appears to be quite high as compared to the noise samples, this still gives a signal-to-noise ratio (SNR) of -11.5dB. The problem gets further compounded in Fig. 1b, where an LTE signal overlaps with the radar pulses. Since LTE signal samples extend for a significant time duration, therefore, in this case the high amplitude radar pulses have an insignificant energy as compared to LTE, therefore they cannot be detected reliably. On the other hand the spectrograms in both the cases give a reasonably clear visual indication of the thin radar pulses along with noise and LTE signals in the background, as shown in Fig. 1a and Fig. 1b respectively. In Fig. 1b the radar gain is increased for clear distinction with LTE signal. For this reason there has been a growing interest in image-based signal identification or classification in wireless communication.

IV. System Model

For our system model we assume a radio monitoring system (RMS) at a fixed location that receives signals from several communication systems in its surroundings. It either receives signals of a single radio technology or from two different radio technologies that creates interference at the RMS. If it does not receive any signal then it only perceives internal noise.

A. LTE and WiFi

LTE and WiFi are OFDM-based technologies and they mainly differ in MAC layer. Therefore, the discrete-time baseband transmitted OFDM signal can be written as [22]

$$x[i] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \sum_{m=-\infty}^{\infty} s_{m,n} g[i - mN] e^{j\frac{2\pi}{N}ni}, \quad (1)$$

where, i is the sample index, $g[i-mN]e^{j\frac{2\pi}{N}ni}$ is the synthesis basis function and n denotes the subcarrier index. The data transmitted on the nth subcarrier of the mth OFDM symbol is expressed as $s_{m,n}$ which is a complex symbol from a M-QAM (Quadrature Amplitude Modulation) constellation, where, M is the modulation order, N represent the total number of subcarriers in an OFDM symbol, $\frac{1}{\sqrt{N}}$ is the power normalization factor and $g[.]$ is the rectangular window function with its time domain coefficients defined as $g[i] = \begin{cases} \frac{1}{\sqrt{T}} & \text{if} \quad |i| \leq \frac{T}{2} \\ 0 & \text{if} \quad |i| > \frac{T}{2} \end{cases}$. $T = \frac{1}{\Delta f} = NT_s$ is the OFDM symbol duration, T_s is the sampling interval and Δf is the subcarrier spacing. $x[.]$ is the output of an N-point Inverse Fast Fourier Transform (IFFT) of $s_{m,n}$. To eliminate interference, a Cyclic Prefix (CP) of length L_{cp} is added to the OFDM symbol whose length is equal or greater than the channel delay spread. The addition of CP transforms $x[.]$ with length L_x to $x_{cp}[.]$ with length $L_x + L_{cp}$. After passing through the channel, the received signal $y[.]$ at the RMS is expressed as

$$y[i] = x_{cp}[i] * h[i] + w[i], \quad (2)$$

where, $*$ represents convolution operation and $h[.]$ represents the discrete wireless channel. $w[.]$ is the receiver's internal noise, often modeled as additive white Gaussian noise (AWGN), $w \sim \mathcal{N}\left(0, \sigma_n^2\right)$, where σ_n^2 is the noise variance. However, in practical receivers such as USRPs, noise is not purely white due to filtering and due to the presence of DC offset which is a consequence of direct conversion architecture of the receiver.

B. FBMC

The discrete-time baseband transmitted FBMC/OQAM (Orthogonal Quadrature Amplitude Modulation) signal can be written as [23]

$$x[i] = \sum_{n=0}^{N-1} \sum_{m=-\infty}^{\infty} a_{m,n} g[i - mN/2] e^{j\frac{2\pi}{N}n(i-D/2)} e^{j\phi_{m,n}}, \quad (3)$$

where, $g[i - mN/2]e^{j\frac{2\pi}{N}n(i-D/2)}e^{j\phi_{m,n}}$ is the synthesis basis function and $a_{m,n} \in \{s^I_{m,n}, s^Q_{m,n}\}$ is a real-valued symbol

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

TABLE I
SIGNAL SOURCE PARAMETERS

Technology	Source	Parameters		Location
Radar	USRP N210	Pulse Repetition Interval	$1msec$	Iris testbed
		Pulse Width	10 μsec	
		Sampling rate	12.5 MS/sec	
LTE	Commercial	Frequency	0.796 GHz, 1.7775 GHz, 2.1275 GHz	Iris testbed, WiSP lab
		Bandwidth	10 MHz, 15 MHz, 12.5 MHz	
	USRP X310 (SRS-LTE)	Frequency	3.7 GHz	Iris testbed
		Bandwidth	6 MHz	
FBMC	USRP N210	Frequency	3.7 GHz	WiSP lab
		Bandwidth	6 MHz	
WiFi	Commercial	Frequency	2.467 GHz, 5.66 GHz	Iris testbed, WiSP lab
		Bandwidth	20 MHz, 15 MHz	

which is either the real or imaginary part of the input QAM symbol $s_{m,n}$. While $g[i]$ represents the well localized prototype filter. $D/2$ is the delay term that depends on the length of the prototype filter, i.e., $D = KN - 1$ and K is the length of poly phase filter. The phase term $\phi_{m,n}$ ensures that the phase shift of $\pm\pi/2$ is between adjacent pulse amplitude modulation (PAM) symbols along the time frequency axis and is given as $\phi_{m,n} = \frac{\pi}{2}(m+n)$. The received FBMC signal is similar to (2) where $x[i]$ in (3) replaces $x_{cp}[i]$.

C. Radar

The discrete-time radar baseband signal is expressed as [11]

$$x[i] = \sum_{m=-\infty}^{\infty} A_m(p[i - m\Delta t_m]).e^{(2\pi jn\frac{f_m^c}{f_0})}. \tag{4}$$

The received baseband radar signal with channel and noise effects is expressed as

$$y[i] = \left(\sum_{m=-\infty}^{\infty} A_m(p[i - m\Delta t_m] * h[i])\right).e^{(2\pi jn\frac{f_m^c}{f_0})} + w[i], \tag{5}$$

where, m is the pulse index, n is the sample index and $p[.]$ represents the Pulse Width (PW) and Intra-Pulse Modulation (IPM). f^c is the pulse centre frequency and f^0 is the carrier frequency. The pulse amplitude A_m undergoes intense variations over time as the radar antenna rotates. The Pulse Repetition Interval (Δt_m) is the time lapse between two successive pulses that arrive at the receiver. The radio channel $h[.]$ causes dispersion of the pulse shape $p[.]$ over time.

D. Interference and Noise

The interference between two radio technologies as received by the RMS can be expressed as

$$y[i] = x_a[i] * h_a[i] + x_b[i] * h_b[i] + w[i], \tag{6}$$

where $a \neq b$ and $a, b \in \{\text{LTE, WiFi, FBMC, Radar}\}$. $h_a[.]$ and $h_b[i]$ are the discrete and independent wireless channels experienced by $x_a[.]$ and $x_b[.]$ respectively. When the RMS does not receive any signal then it only perceives internal noise, i.e.,

$$y[i] = w[i]. \tag{7}$$

E. Q-spectrogram

The received time-domain baseband signal $y[.]$ is transformed into a spectrogram by taking Fast Fourier Transform (FFT) of successive samples, i.e.,

$$\boxed{y[.] \xrightarrow{\mathcal{FFT}} Y[.],} \tag{8}$$

where $Y[.]$ is a single FFT trace of length K. The FFT traces are also averaged out to improve contrast (5 traces were averaged out). The averaged FFT traces are stacked along the time domain to create a spectrogram of size $K \times K$, i.e., K rows and K columns. For brevity $S_{K\times K}$ is expressed as S. The FFT size determines the frequency resolution of the signal, thus a higher FFT size gives more information about the contours in frequency domain. However, doubling the FFT size, which is typically a power of 2 (e.g., 32, 64, 128 etc.), quadruples the size of the spectrogram. For example a 64×64 spectrogram has 4096 frequency bins, whereas, a 128×128 spectrogram has 16384 frequency bins. On the other hand the larger sized spectrogram also makes it more complex to feed into the CNN model, as it would significantly increase the model's complexity. Therefore, a trade-off has to be made between preserving frequency resolution and the resulting complexity of the CNN model.

To deal with this issue we created a custom spectrogram which is called the Quarter Spectrogram or Q-spectrogram. The Q-spectrogram squeezes a higher resolution FFT spectrogram into a smaller size which is quarter of the original size and hence the name Quarter Spectrogram. Since the energy information along the frequency axis is symmetric about the DC or the centre frequency, therefore, we mainly extract the left portion of S. The creation of a Q-spectrogram (image) can be expressed as follows.

$$S = \sum_{x=1}^{K}\sum_{y=1}^{K} S(x,y), \tag{9}$$

$$S_1 = \sum_{x=49}^{63}\sum_{y=1}^{K} S(x,y) \frown \sum_{x=65}^{71}\sum_{y=1}^{K} S(x,y) \frown \sum_{x=15}^{24}\sum_{y=1}^{K} S(x,y) \tag{10}$$

$$S_1^{lower} = \sum_{x=1}^{\frac{K}{4}}\sum_{y=1}^{\frac{K}{2}} S_1(x,y), \tag{11}$$

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

$$S_1^{upper} = \sum_{x=1}^{\frac{K}{4}} \sum_{y=\frac{K}{2}+1}^{K} S_1(x,y), \tag{12}$$

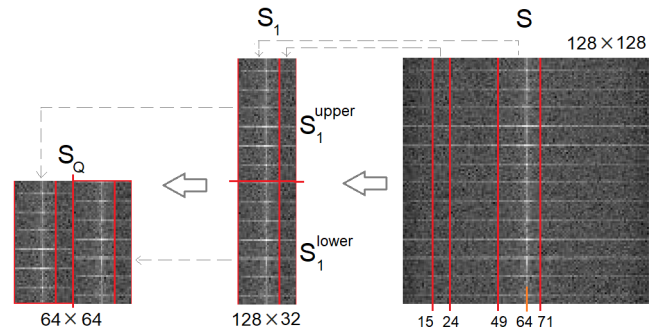
$$S_Q = S_1^{lower} \frown S_1^{upper}, \tag{13}$$

where, S_1 is formed by horizontally concatenating two vertical portions of S and $\frown$ represents the concatenation operation. These two portions span from (frequency) index 15 to 24 and 49 to 71, where the index 64 is eliminated to minimize DC component. S_1 has dimensions $K \times K/4$. S_1^{lower} and S_1^{upper} are the lower and upper halves of S_1 respectively, each with dimensions $K/2 \times K/4$. Finally, S_Q is the Q-spectrogram which is formed by horizontally concatenating S_1^{lower} and S_1^{upper} and has dimensions $K/2 \times K/2$.

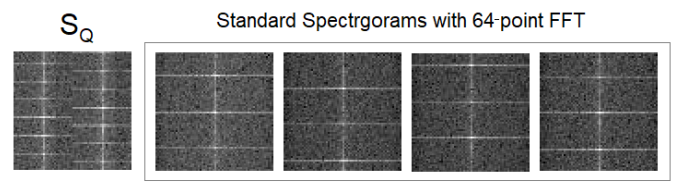
Fig. 2a shows step-by-step creation of Q-spectrogram. At first a standard spectrogram S of size 128×128 is created by taking a 128-point FFT. S_1 is formed by concatenating two portions of S as indicated by red lines and dulling the DC component indicated by the orange line. Since the DC component spans more than one bin, therefore it is not completely discarded. A red horizontal line bisects S_1 in the middle, thus creating S_1^{lower} and S_1^{upper} as indicated. S_1^{lower} and S_1^{upper} are then placed side-by-side, i.e., horizontally concatenated to form the Q-spectrogram, S_Q. The resulting Q-spectrogram is reduced to a size of 64×64. Fig. 2b shows four standard spectrograms (64-point FFT) created from the same signal as S_Q and both contain 13 radar pulses. Therefore, S_Q is able to squeeze the same amount of information along the time-domain as the four standard spectrograms, each of the same size. Although some of the unique information on the left side of S is discarded in creating S_Q, index 24 to 49 in Fig. 2a, however, we observed for all the signals used in our work that this does not affect their distinguishing characteristics.

By training the CNN model with substantial data, as we did, the model is able to explore deep features from the Q-spectrograms and can thus reliably classify various signals as also indicated by the performance (in Results section). The important insight is that even while considering a subset of frequency indices, and by feeding more information along the time domain, the CNN model reliably learns distinguishable features, given the training dataset is large enough. Index 15 to 24 were selected after visually inspecting the 128-point FFTs of all the radio signals as they cover the outer region of the spectrogram, whereas, index 49 to 71 cover the mid region. It may not make a substantial difference if frequency indices are varied slightly, however, it will require model re-training.

128-point FFT gives better frequency resolution compared to 64-point FFT, moreover, the middle and edge portions along frequency axis capture vital information along the frequency axis and finally squeezing the information along the time axis results in more conspicuous visual features on time-frequency grid as compared to a standard spectrogram of similar size created using 64-point FFT. Capturing more information along the time-axis is particularly useful for bursty signals like Radar and WiFi where the information sometimes appear intermittently along the time axis.



(a) Step-by-step creation of Q-spectrogram.



(b) Q-spectrogram comparison with standard spectrograms.

Fig. 2. Q-spectrograms.

F. Deep Learning

Deep learning (DL) is a sub-category of machine learning. It extends on the concept of artificial neural networks that has a single hidden layer between the input and the output layers. DL on the other hand has multiple hidden layers that allows it to perform complex tasks, such as multi-class classification on larger input images. In our problem, DL finds a mathematical function, $\mathcal{F}$, that expresses the relationship between a set of inputs $\mathbf{X}$ and a set of outputs $\boldsymbol{Z}$, i.e.,

$$\mathcal{F} : \mathbf{X} \rightarrow \boldsymbol{Z}. \tag{14}$$

The inputs, $\mathbf{X} \in \mathbb{R}^{j \times k}$, represents a set of input vectors denoted as

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_j^T \end{bmatrix}, \tag{15}$$

where, j is the size of the dataset, $k = (\frac{K}{2} \times \frac{K}{2})$, $\mathbf{x}_i$ is a column vector of k measurements or data points $S_i(x,y)$, derived from the ith Q-spectrogram, S_Q, by vertically concatenating all the columns of S_Q. For every $\mathbf{x}_i \in \mathbf{X}$, $\mathcal{F}$ predicts $\mathbf{z} \in \boldsymbol{Z}$, where $\mathbf{z}$ represents L class labels, i.e.,

$$\mathbf{z} = [z_1, z_2, ...z_L]^T. \tag{16}$$

V. Experimental Methodology

The main steps involved in this deep learning experiment are wireless signal acquisition, data set creation, model training and model testing. Fig. 3 shows the experimental setup used for transmitting, receiving, recording and processing the radio signals.

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

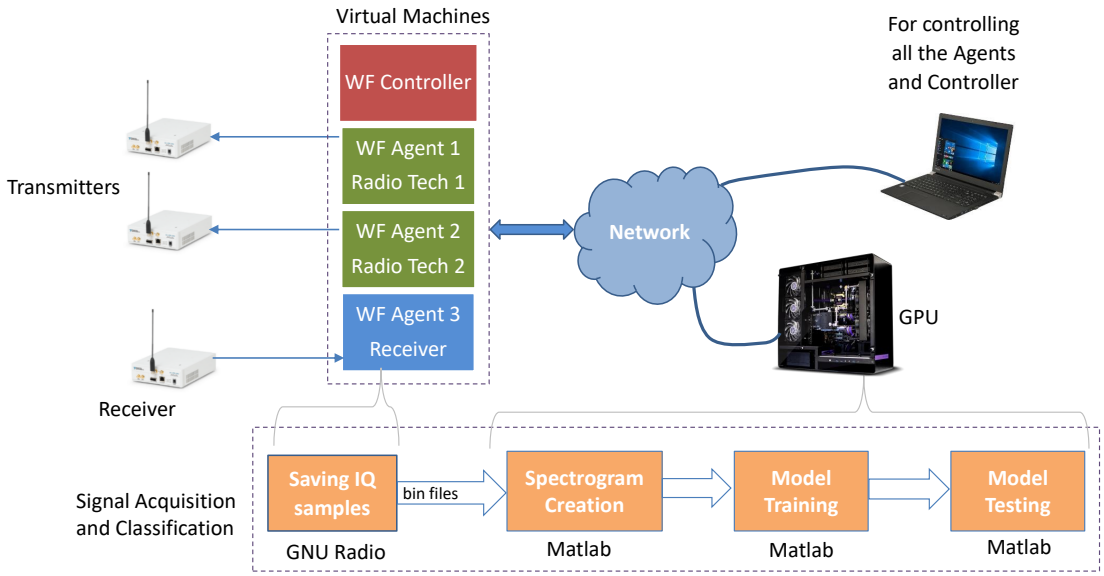


Fig. 3. Experimental setup.

A. Wireless Signals Acquisition

We acquired over-the-air wireless signals while using the Universal Software Radio Peripheral (USRP) devices. The experimental measurements were performed at the Iris testbed, CONNECT Centre, Trinity College Dublin (TCD) and at Wireless & Signal Processing (WiSP) lab, Institute of Space Technology, Islamabad. Four wireless technologies were used, i.e., radar, LTE, FBMC and WiFi. We created a diverse dataset of these wireless technologies that comprised of commercial broadcast as well as USRP generated signals in the lab, across multiple locations and times with different USRP devices, as shown in Table I. A diverse dataset is critical for robust model training using CNN, as explained later.

We used different models of USRPs, i.e., N210, B210 and X310. This was done for experimental ease while working in the testbed and does not effect the dataset creation. The radar pulses were generated using a USRP-based radar emulator that was developed in TCD. The parameters used for the radar emulator are shown in [24]. SRS LTE software suite was used for LTE transmission in the lab, which is compliant with the LTE standard (Rel-8) [25]. Moreover, commercial LTE signals were also captured at Iris testbed and WiSP lab. FBMC signal was generated using USRP [26], [27]. FBMC has been included as it is a potential candidate for future 5G networks. WiFi measurements were also done at several times of the day both at Iris testbed and WiSP lab. This helped in acquiring WiFi signals with varying traffic densities, as explained later. Interference combinations of these four wireless technologies were created by simultaneously transmitting two of these signals at the same carrier frequency and recording with another USRP or by receiving individual signals separately and adding them offline using MATLAB.

The transmitters and receiver were instantiated using WiSHFUL architecture [28] that allows easier control and management of multiple USRPs over a network and also supports dynamic on-the-fly reconfiguration of parameters such as transmit power, frequency etc. WiSHFUL stands for Wireless Software and Hardware platforms for Flexible and Unified radio and network controL [29], [30]. It is a framework that offers Unified Programming Interfaces (UPIs) for PHY and MAC control of various radio testbeds, e.g. USRPs connected to a local area network. A WiSHFUL controller was initialized on a virtual machine (VM) created on a server, whereas, WiSHFUL agents were initialized on other VMs. Fig. 3 shows WiSHFUL controller and agents instantiated on VM server rack and were accessed by a computer connected to the network (WF is a short form for WiSHFUL). Alternatively, each of the controller and agents can be instantiated on a separate computer, however, VMs allow economy of hardware resources as multiple VMs can be created on a single server depending on its computational speed. While doing measurements for an interfering scenario, e.g., LTE+Radar, four VMs were created that instantiated a WiSHFUL controller, a WiSHFUL agent for LTE transmission, a WiSHFUL agent for Radar transmission and a WiSHFUL agent for the reception of LTE and Radar signals centered on the same carrier frequency.

B. Dataset Creation

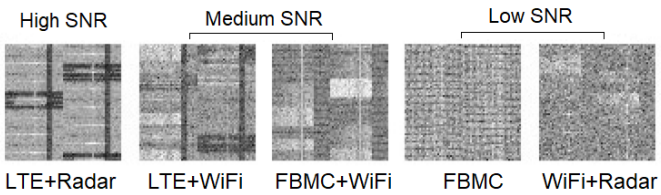


Fig. 4. Q-spectrograms of different technologies/interference at different SNRs.

The dataset was created by generating Q-spectrograms from the recorded wireless signals. The Q-spectrograms of a few

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

representative technologies and interferences considered in this work, at different SNR levels, are shown in Fig. 4. Only three SNR levels, i.e., high, medium and low are shown for visual demonstration. More Q-spectrogams of WiFi and noise can also be seen in Fig. 13. As can be observed, the visual features have been condensed in a much smaller image that aids in learning of the CNN. For the LTE signal, the thick vertical lines, shown in dark grey color, represent the spectrum edges. The Q-spectrogram also contains more information along the temporal domain which is particularly useful for radar pulses and WiFi traffic density estimation as explained in the results section. The dataset images were created in Matlab from the IQ samples gathered from the receiver USRP, as also shown in Fig. 3.

The SNR of the dataset is varied by varying the gain value of the receiver USRP. Alternatively, this can be done by varying the gain value of the transmitter USRP as well. The SNR is computed as

$$\text{SNR} = 10\log_{10}(\frac{E_S}{E_N}). \tag{17}$$

E_{Rxd} is the energy of the received signal computed using K samples (complex baseband) and E_N is the energy of pure noise also computed from K samples. E_S is the pure signal energy which is computed by subtracting the two, i.e., $E_S = E_{Rxd} - E_N$. However, while recording the commercial broadcast signals, the lower SNRs cannot be achieved easily by varying the USRP gain alone. Therefore, AWGN was also added to the received signal files to achieve the desired SNR levels in some cases. However, this does not effect the integrity of signals. This is particularly useful while dealing with SNR combinations of mixed classes, e.g., LTE+Radar. In this case, it is very difficult to achieve various SNR combinations while varying the receiver USRP gain and therefore AWGN was added to achieve the desired SNR levels. Moreover, for mixed classes, the individual signal at a particular SNR is normalized with its standard deviation before being added to the other signal.

C. Model Training and Testing

Fig. 5 shows training loss and accuracy versus the number of iterations for all the CNN models. Of all the models, ResNet18 converges first, thus it takes the least training time. The loss function is a very important indicator for neural networks and it measures the inconsistency between the predicted classes and the actual classes. Cross-Entropy (CE) loss is computed after Softmax activation; also called the Softmax Loss. The CNN has L output neurons (for L classes) that are gathered in a vector s with probability scores. The ground truth vector t is a one-hot vector with one correct class and $L-1$ incorrect classes. The Softmax function is expressed as $f(s_i) = \frac{e^{s_i}}{\sum_j^L e^{s_j}}$ and the CE loss is computed as

$$\boxed{Loss = -\sum_{i=1}^{L} t_i \log f(s_i).} \tag{18}$$

During training, the model tries to minimize the loss function by adjusting the model parameters including weights and biases, etc. Accuracy depicts the percentage of correctly predicted classes. A single iteration involves a forward pass of the inputs through the network, computation of loss function, backward pass through the network and adjustment of weights to minimize the loss function. Fig. 5 shows how loss decreases and accuracy increases during the training of the CNN models and the inverse relationship between accuracy and loss can be observed clearly. The dataset contains 5000 images per class at different SNR levels, out of which 4000 images are used for training and 1000 images are used for testing. While training each model, 3200 images of each class are used to train the model while the remaining 800 images are used for testing after every 5 epochs during the training process. The mean image of the data set was computed and subtracted from each of the Q-spectrogram image before it was passed on to the CNN. This normalizes the dataset before multiplying with weights and adding biases to the input data in the network for causing activations, which are then back propagated with the gradients to train the model. Mean subtraction ensures that the image features have similar ranges and the gradients do not go out of control. The Q-spectrograms spanned a wide range of SNR to enable robust training.

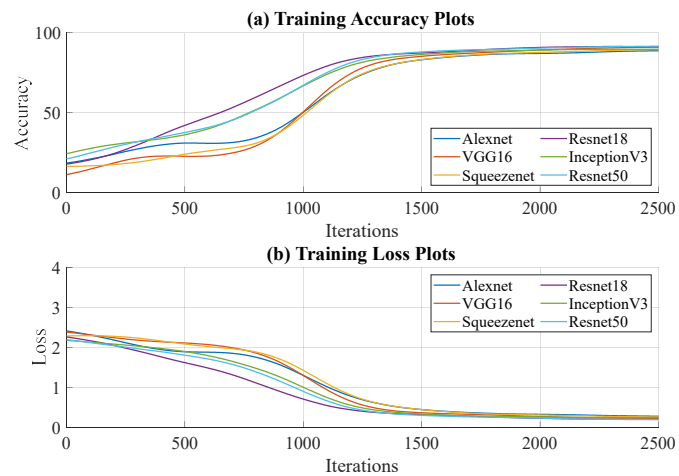


Fig. 5. No. of iterations vs accuracy and loss during training of CNN models.

The hyper-parameters selected for training the CNN models include a mini-batch size of 10, max epochs of 30, Stochastic Gradient Descent with Momentum (SGDM) as an optimizer, momentum of 0.9, initial learn rate of 0.01 and piece wise learning rate with drop factor of 0.2 applied after every 5 epochs. MATLAB R2018b is used for all the experiments on a workstation with 64 GB RAM, Intel 2.3 GHz, 64-bit processor with 18 cores. NVIDIA GPU with 12 GB memory and compute capability of 3.5 is used for training the CNN models. Each CNN model is trained with the same set of hyper-parameters and training/testing data in order to compare the training and validation performance of each model on the same scale and select the most efficient model for our classification task. The training accuracy and loss graphs in Fig. 5 show how smoothly each model converges and the iterations it takes to converge.

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

D. WiFi Traffic Density Estimation

WiFi has a contention-based MAC protocol, therefore, its spectrogram contains spectrum lines with lot of variation along the time axis. Even when several users are connected to a WiFi access point (AP) on the same channel, the data traffic varies significantly over time which results in WiFi signal spectrograms interleaved with pure noise spectrograms. This can be seen in Fig. 13a where the WiFi Q-spectrograms can be observed at the beginning and the at the end, bounded by the black line, whereas the noise Q-spectrograms are present in the middle. The WiFi spectrum traces within a Q-spectrogram also vary, but even if there is a single trace observed in a Q-spectrogram, it is considered as a WiFi Q-spectrogram. A few noise Q-spectrograms in the middle of signal Q-spectrograms does not indicate the absence of WiFi signal. When the WiFi AP is heavily loaded with user traffic then the ratio of signal Q-spectrograms and pure noise Q-spectrograms is higher, conversely, when the traffic load is low this ratio is less. We define this ratio as WiFi traffic density

$$D_{WiFi} = \frac{N_{signal}}{N_{noise}}, \tag{19}$$

where, N_{signal} is the number of WiFi signal Q-spectrograms and N_{noise} is the number of noise Q-spectrograms per 100 Q-spectrograms.

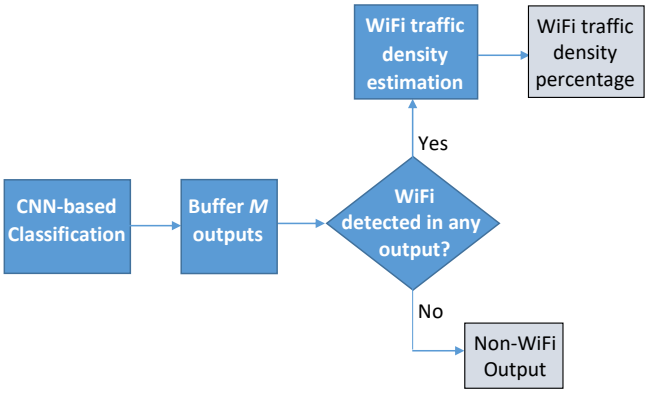


Fig. 6. Signal classification scheme.

Considering this temporal pattern of WiFi, we propose a simple scheme for signal classification as shown in Fig. 6. We buffer M outputs of the CNN model and check if their is any WiFi detection. If yes, then the classification scheme computes WiFi traffic density D_{WiFi}. If no WiFi signal is detected by the CNN model in M successive Q-spectrograms then the scheme outputs the non-WiFi class. If their is a mixed WiFi class, such as, WiFi+LTE, still D_{WiFi} is computed. In this case, we typically observed WiFi+LTE Q-spectrograms interlaced with LTE Q-spectrograms. M was selected by analyzing the low density WiFi signal, where a signal Q-spectrogram was typically observed in M Q-spectrograms. We used $M = 50$. This simple scheme efficiently detects WiFi signal and its density through observation over time. WiFi density is a very useful information for any service that may try to coexist in the WiFi spectrum. If the WiFi density is low then sharing of WiFi spectrum can be advantageous otherwise it may not be feasible. Our WiFi datasets were acquired at two different locations, i.e., CONNECT centre and WiSP Lab and over several times of the day to capture signal sets with varied traffic densities.

VI. CNN Models

CNN is extensively used in computer vision for image classification, object detection and speech recognition. With sufficient training and by using a diverse data set, CNN can learn robust features which makes it a good choice for performing signal classification in the presence of interference. A CNN architecture comprises several types of layers each having a different purpose. Convolutional layers perform 2D convolutions on the input image or feature map to create activations for various possible features, such as, edges, blobs, curves etc. Each convolutional layer is followed by a Rectified Linear Unit (ReLU) layer which introduces non-linearity to the system, which is often applied to the output feature maps of a convolutional layer, as the convolutional layer only performs linear operations. Pooling layers perform dimensionality reduction by retaining only dominant features. Dropout layers drop a random set of activations in the preceding layer to prevent over-fitting of the model. Several groups of the aforementioned layers are stacked up in a CNN architecture and the result of the whole process feeds into a fully connected layer that drives the final classification decision using Softmax.

We have trained and evaluated six well-known pre-trained CNN models, presented in Table II, in order to choose the most suitable model for our wireless technology and interference classification problem. These models are pre-trained using a million images of 1000 different objects from the ImageNet database used in the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) [31]. Fine-tuning a pre-trained CNN is usually much easier and faster than designing and training a CNN from scratch. AlexNet [32], shown in Fig. 7, is one of the initial CNN models that started the deep learning revolution in computer vision by winning the ImageNet ILSVRC-2012 competition by a huge margin. The CNN architectures used in this work are well-known and their pre-trained versions are publicly available for transfer learning. Further details of each architecture are available in [32], [33], [34], [35], [36].

The AlexNet architecture comprises five convolutional layers (CONV) followed by three fully connected layers as shown in Fig. 7. The input layer takes an image of 64×64 pixels as input. The first two convolutional layers extracting low-level features are followed by max-pooling layers for dimensionality reduction of the tensors. The third, fourth and fifth convolutional layers are connected directly. The fifth convolutional layer is followed by a max-pooling layer, the output of which goes into a series of two fully connected layers. The second fully connected layer feeds into a soft-max classification layer with 10 classes. Each convolutional layer is followed by a ReLU non-linearity, and the first and second convolution+ReLU layers are followed by a local normalization layer. The output of each feature extraction or dimensionality reduction layer is a cube of feature maps

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

TABLE II
PRE-TRAINED CNN MODELS USED IN THIS WORK

Pre-trained Model	Depth*	Training Parameters (Millions)	Size (MBs)
AlexNet	8	61	227
VGG16	16	138	515
ResNet18	18	11.7	44
SqueezeNet	19	1.24	4.6
InceptionV3	48	23.9	89
ResNet50	50	25.6	96

*Highest number of sequential convolutional or fully connected layers among all paths from the input layer to the classification layer.

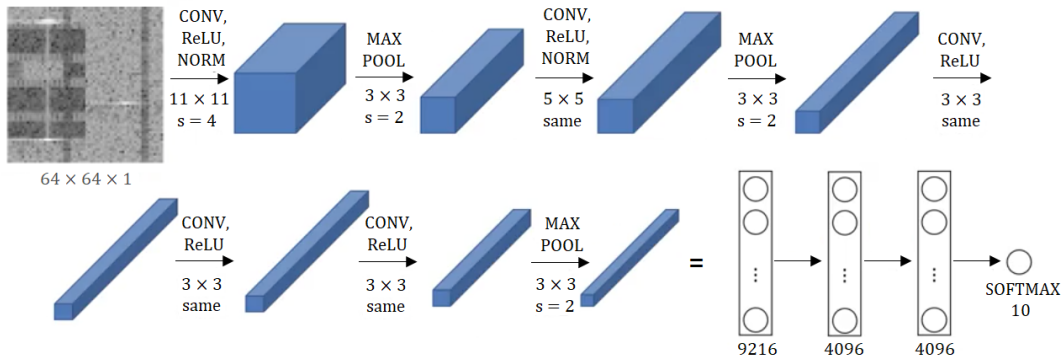


Fig. 7. AlexNet fine-tuned to our data.

represented by blue blocks in the Fig. 7. Moreover, in Fig. 7, s represents the stride.

VGG [33] comprises 16 layers and uses smaller convolutional filters in order to learn more complex features. VGG has high computation and memory requirements as compared to other pre-trained CNNs due to the large number of filters in convolutional layers. The concept of skip connections or residual connections was incorporated in ResNet architecture [34], which helped the problem of vanishing gradient caused by increasing number of layers and made it convenient to build deeper models. The residual blocks in ResNet learn the residual instead of learning a function, which simplifies and improves training of deeper networks. InceptionV3, the most recent version of GoogleNet [35], uses inception blocks employing batch normalization, image distortions and RMSprop for improved feature extraction and convolutions of smaller sizes to drastically reduce the learn parameters. SqueezeNet [36] architecture uses 1×1 filters instead of 3×3 filters in AlexNet to reduce computational complexity and achieve AlexNet-level accuracy on ImageNet with 50x fewer parameters. Prior to transfer learning, we adjust the input image size of each CNN to match the Q-spectrogram size, and the sizes of final fully connected layer and softmax classification layer to match the output classes, i.e., 1×10.

Usually the physical layer characteristics and operating frequency band for a specific radio technology are standardized and then they do not change over a short span, as is the case with LTE, WiFi, DTV etc. Therefore, once a deep learning model is trained for certain coexistence scenario, then this will be valid for a reasonably long duration. Moreover, 5G utilizes software-defined networking (SDN) and network function virtualization (NFV), that can be used to update or fine-tune the deep learning models on edge devices in the network [37].

VII. RESULTS

In [38], a CNN has been used to classify between different primary user channel access patterns, but in this work we have focused on technology classification and their interferences and used MAC patterns to determine WiFi traffic density. Our models shows very good classification performance overall as it has been trained with a dataset that spans over a wide range of SNRs. The classification performance is expressed in terms of accuracy which is expressed as,

$$A = \frac{N_{correct}}{N_{total}} \times 100 \tag{20}$$

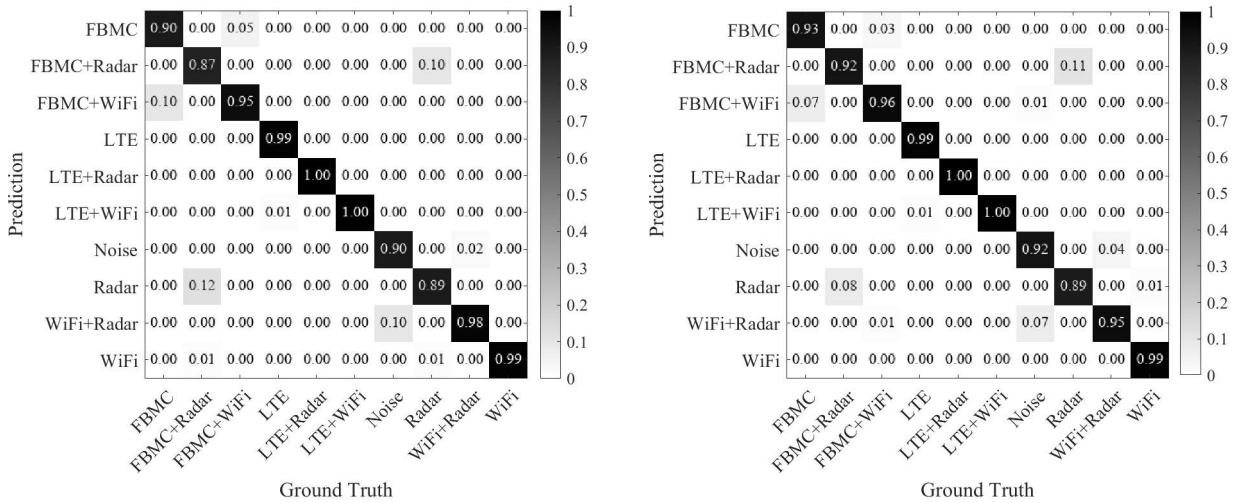
where, $N_{correct}$ is the number of Q-spectrograms that are classified correctly and N_{total} is the total number of Q-spectrograms in the test dataset.

Fig. 8 shows the confusion matrices generated from the test data set at for the six CNN models. It can be observed that all the models give 100% classification accuracy for LTE-based classes, i.e. LTE, LTE+Radar and LTE+WiFi. All the models with slight variation exhibit relatively higher percentage of miss-classification for Radar and WiFi+Radar. This may happen because of thin radar pulses that may be confused with short time signal burst of WiFi. Since WiFi is a bursty signal, a WiFi Q-spectrogram sometimes only contain one or two spectral lines that appear similar to radar pulses. The FBMC-based classes also have a relatively higher percentage of miss-classification for all CNN models.

Since ResNet18 gives the overall best performance, the subsequent results are based on this model. Fig. 9 shows

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking



(a) AlexNet

(b) InceptionV3

(c) ResNet18

(d) ResNet50

(e) SqueezeNet

(f) VGG16

Fig. 8. Confusion Matrices of CNN models at 5dB SNR.

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

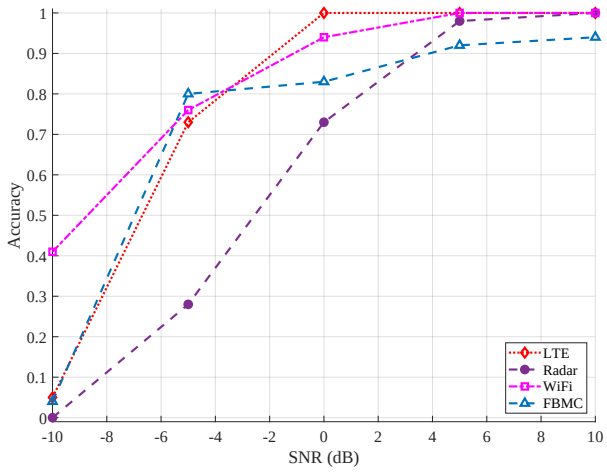


Fig. 9. Classification accuracy vs SNR for single classes with ResNet18.

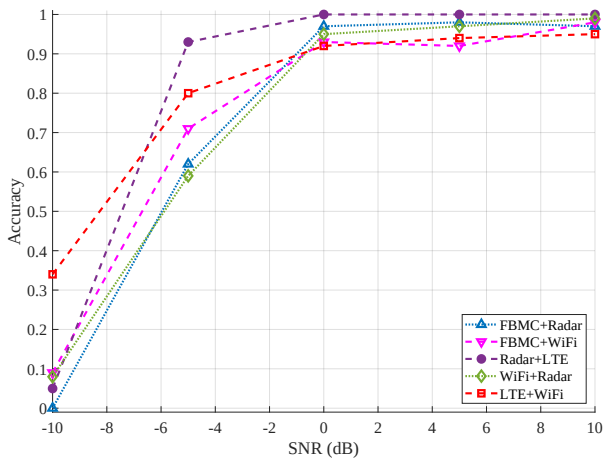


Fig. 10. Classification accuracy vs SNR for mixed classes with ResNet18.

the performance of ResNet18 CNN model for single classes. WiFi is classified with highest accuracy among the single classes, even at low SNR (-10dB) the accuracy is 40%. Radar is classified with the lowest accuracy. This possibly happens due to the more obvious visual signature of WiFi, whereas, Radar only has thin pulses that are overwhelmed by noise at low SNR. The classification accuracy for WiFi is moderately inferior to LTE, as it is a bursty signal and its appearance varies from thin pulses to more conspicuous traces as shown later in Fig. 13. The overall accuracy for FBMC is also relatively lower.

Fig. 10 shows the classification performance of ResNet18 model for mixed classes. The classification accuracy for a particular SNR value implies same SNR for both the signals of a mixed class. Above 0dB all the classes are classified with more than 90% accuracy. Overall Radar+LTE is classified with highest accuracy at higher SNRs (>-5dB), however, at -10dB LTE+WiFi gives relatively better accuracy of 35%. In interference scenarios, the relative SNR of the interfering signals can vary, e.g., in case of LTE and Radar interference in the 3.5GHz band, either of the two signals can have a relatively higher or lower SNR or their SNRs may be equal.

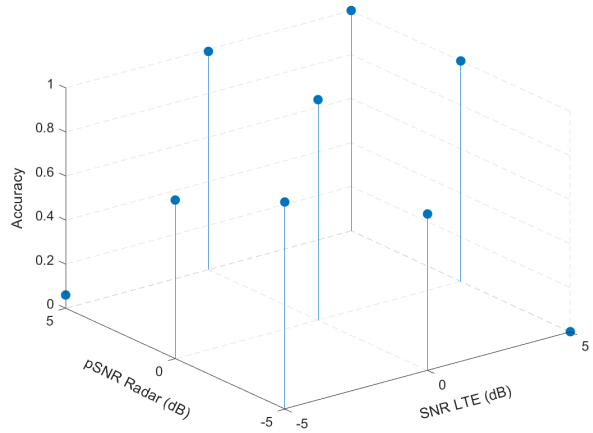


Fig. 11. Classification accuracy for LTE+Radar with different SNR combinations.

Fig. 11 shows the classification accuracy of LTE+Radar for several combinations of relative SNRs of both signals. It is observed that the classification accuracy varies significantly with the variation in relative SNR. When the individual SNR of LTE and Radar is higher than 0dB then the accuracy is approximately 100%. However, when the SNR of either signals (Radar or LTE) is 5dB and that of the other is -5dB then the accuracy drops to almost zero. When the two SNRs are same yet at the lowest level (-5dB), the accuracy remains quite high at 94%. When the SNR of LTE and Radar is -5 and 5dB respectively, the classification accuracy is 6% and the model miss-classifies to WiFi+Radar (56%) and Radar (38%). Since WiFi is also an OFDM signal, therefore, at lower signal level and in the presence of Radar pulses, the model confuses LTE+Radar with WiFi+Radar. Moreover, the model also sometimes confuses it with pure Radar signal while completely missing out the background LTE.

Fig. 12 shows the complexity, accuracy and prediction time comparison of the six CNN models used for signal classification. The x-axis represents the time required by a trained CNN model to classify a single input Q-spectrogram image. The circle size depicts the computational complexity of the model which depends on the number of training parameters which in turn is related to the number of layers and the number of convolutional kernels. The y-axis represents the training accuracy of the models. The white dot in the middle of each circle represents the corresponding accuracy. The highest overall accuracy of 98% is achieved by ResNet50, whereas VGG16 has the highest complexity and InceptionV3 requires the longest prediction time of 400 msec. While considering accuracy, prediction time and complexity, ResNet18 gives the best overall performance.

Performance Comparison: We compare our results for ResNet18 with that of [13], Fig. 6. Their proposed CNN, while utilizing FFT-based (standard) spectrogram and classifying four classes (LTE, WiFi, mLTE-U and WiFi+LTE), gives a classification accuracy of 80% at 0dB SNR (lowest considered SNR). While using IQ-based spectrogram, the same model gives an even inferior accuracy of 65%. Whereas our fine-tuned ResNet18 model, for mixed classes, gives an accuracy

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

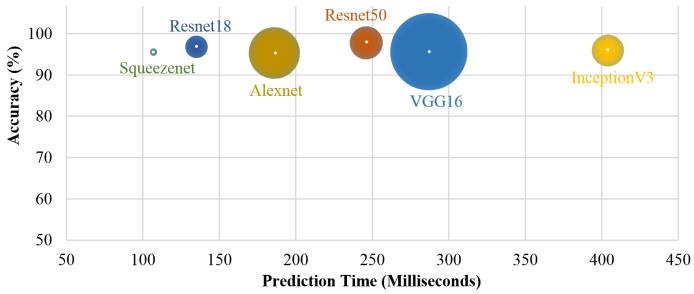
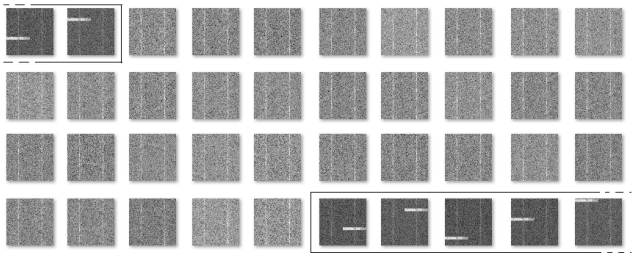
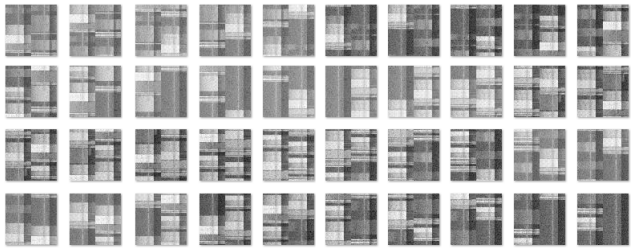


Fig. 12. Accuracy and complexity comparison of CNN models.

>90% at 0dB SNR, where LTE+WiFi is classified with 92% accuracy. For single classes, our fine-tuned ResNet18 model gives an accuracy of 100% and 93% at 0dB SNR, for LTE and WiFi respectively. Therefore, our fine-tuned ResNet18 model performs significantly better than [13].



(a) Low density traffic.



(b) High density traffic

Fig. 13. WiFi traffic patterns.

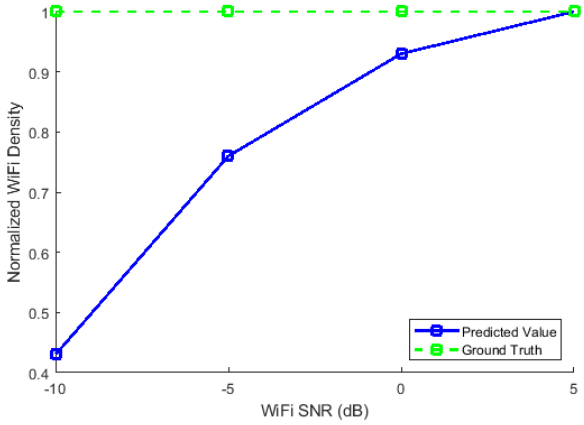


Fig. 14. WiFi traffic density vs SNR.

Fig. 14 shows the density estimation of pure WiFi signal at different SNR levels. At higher SNR level (5dB), the normalized WiFi density (D_{WiFi}) is correctly estimated. However, as the WiFi SNR degrades, the ResNet18 model starts to miss-classify some of the WiFi Q-spectrograms as Noise, therefore, the density estimation reflects a lower value than the ground truth. For this simulation the ground truth of a high SNR WiFi signal was determined and then noise was added in the same signal and the classifier outputs were compared with the ground truth to compute the normalized WiFi density.

Performance comparison for WiFi traffic estimation: In [18], WiFi traffic estimation has been done by feeding spectrograms (short-term Fourier Transform) or time-frequency images to a CNN. However, their CNN model only classifies between different WiFi traffic patterns, i.e., low, medium, high and constant traffic, without interference. The first three correspond to duty cycles of 25%, 50% and 75% respectively. With synthetic dataset they achieve an accuracy of 99% using spectrograms, however, while testing their model with real-world WiFi traffic, they achieve an accuracy of 59.1%. Their results do not present the performance w.r.t. SNR either. In [39], logarithmic power spectral density feature is used to train a CNN for WiFi identification. At 0dB SNR their system identifies WiFi signals with an accuracy of 74% which is also an inferior performance. They do not perform traffic estimation either. Therefore, our fine-tuned ResNet18 model performs significantly better, as it classifies WiFi as well as its interference combinations with an accuracy > 90% at 0dB SNR and subsequently computes the WiFi traffic density.

VIII. Conclusions

Spectrum sharing in 3.5 GHz and 5 GHz bands create enormous opportunities for provisioning of high speed broadband services for public use. However, this also brings in technical challenges that include interference management and fair usage of shared spectral resources. Radio monitoring is a vital technology that can help address these challenges by identifying (or classifying) the RATs or their associated interferences that may frequently arise in such a complex heterogeneous radio landscape. In this work, we presented an experimental evaluation of a CNN-based radio signal classification scheme. Six well-known CNN models were used to perform radio signal classification of ten classes that include, LTE, Radar, FBMC, WiFi and their interference combinations, i.e., LTE+Radar, LTE+WiFi, FBMC+Radar, FBMC+WiFi, WiFi+Radar and Noise. Overall highest classification accuracy of 98% was achieved by ResNet18 model, whereas, the SqueezeNet model offered the lowest complexity and the shortest prediction time with slightly lower classification accuracy. An efficient signal representation called the Q-spectrogram was proposed that squeezes temporal and frequency information into a smaller size input image for CNN models. This is particularly useful for WiFi and Radar whose bursty signaling patterns require comparatively longer observation duration. We also presented a simple classification framework that computes WiFi traffic density, which is extremely useful for those services that aim to coexist with WiFi in 5 GHz band.

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Horizon 2020 Program under grant agreement n645274 (WiSHFUL project). Guidance and support of Prof. Luiz A. DaSilva is also gratefully acknowledged.

REFERENCES

[1] R. Zhou, F. Liu, and C. W. Gravelle, "Deep Learning for Modulation Recognition: A Survey With a Demonstration," *IEEE Access*, vol. 8, pp. 67 366–67 376, 2020.

[2] J. Gao, X. Yi, C. Zhong, X. Chen, and Z. Zhang, "Deep learning for spectrum sensing," *IEEE Wireless Communications Letters*, vol. 8, no. 6, pp. 1727–1730, 2019.

[3] T. O'Shea and N. West, "Radio Machine Learning Dataset Generation with GNU Radio," *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016.

[4] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Engineering Applications of Neural Networks*, 2016, pp. 213–226.

[5] T. J. OÂShea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, Feb 2018.

[6] M. Kulin, T. Kazaz, I. Moerman, and E. De Poorter, "End-to-End Learning From Spectrum Data: A Deep Learning Approach for Wireless Signal Identification in Spectrum Monitoring Applications," *IEEE Access*, vol. 6, pp. 18 484–18 501, 2018.

[7] W. Liu, M. Kulin, T. Kazaz, A. Shahid, I. Moerman, and E. De Poorter, "Wireless Technology Recognition Based on RSSI Distribution at Sub-Nyquist Sampling Rate for Constrained Devices," *Sensors*, vol. 17, no. 9, 2017.

[8] S. Grimaldi, A. Mahmood, and M. Gidlund, "Real-Time Interference Identification via Supervised Learning: Embedding Coexistence Awareness in IoT Devices," *IEEE Access*, vol. 7, pp. 835–850, 2019.

[9] S. Yi, H. Wang, W. Xue, X. Fan, L. Wang, J. Tian, and R. Matsukura, "Interference Source Identification for IEEE 802.15.4 wireless Sensor Networks Using Deep Learning," in *IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2018, pp. 1–7.

[10] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, Sep. 2018.

[11] 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 *GLOBECOM*, Dec 2017, pp. 1–6.

[12] J. Fontaine, E. Fonseca, A. Shahid, M. Kist, L. A. DaSilva, I. Moerman, and E. D. Poorter, "Towards low-complexity wireless technology classification across multiple environments," vol. 91, p. 101881, 2019.

[13] V. Maglogiannis, A. Shahid, D. Naudts, E. De Poorter, and I. Moerman, "Enhancing the Coexistence of LTE and Wi-Fi in Unlicensed Spectrum Through Convolutional Neural Networks," *IEEE Access*, vol. 7, pp. 28 464–28 477, 2019.

[14] V. Maglogiannis, D. Naudts, A. Shahid, and I. Moerman, "An adaptive LTE listen-before-talk scheme towards a fair coexistence with Wi-Fi in unlicensed spectrum," *Telecommunication Systems*, pp. 701–721, 2018.

[15] P. Henarejos, M. Ã. VÃ¡zquez, and A. I. PÃ©rez-Neira, "Deep Learning For Experimental Hybrid Terrestrial and Satellite Interference Management," in *IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2019, pp. 1–5.

[16] M. Camelo, A. Shahid, J. Fontaine, F. A. P. de Figueiredo, E. De Poorter, I. Moerman, and S. Latre, "A semi-supervised learning approach towards automatic wireless technology recognition," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–10.

[17] K. Tekbiyik, Ã. Akbunar, A. R. Ekti, A. GÃ¶rÃ§in, and G. Karabulut Kurt, "MultiÂdimensional wireless signal identification based on support vector machines," *IEEE Access*, vol. 7, pp. 138 890–138 903, 2019.

[18] T. De Schepper, M. Camelo, J. Famaey, and S. LatrÃ©, "Traffic classification at the radio spectrum level using deep learning models trained with synthetic data," *International Journal of Network Management*, vol. 30, no. 4, 2020.

[19] B. Chen, J. Chen, Y. Gao, and J. Zhang, "Coexistence of LTE-LAA and Wi-Fi on 5 GHz with corresponding deployment scenarios: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 7–32, 2017.

[20] Federal Communications Commission (FCC), "Promoting Investment in the 3550-3700 MHz Band," *GN Docket no 17-258*, Oct. 2018. [Online]. Available: https://www.federalregister.gov/documents/2018/12/07/2018-25795/promoting-investment-in-the-3550-3700-mhz-band

[21] 5G New Radio Spectrum Sharing. [Online]. Available: https://www.qualcomm.com/news/onq/2018/01/03/what-can-we-do-5g-nr-spectrum-sharing-isnt-possible-today

[22] B. Farhang-Boroujeny, "Ofdm versus filter bank multicarrier," *IEEE Signal Processing Magazine*, vol. 28, pp. 92–112, 2011.

[23] L. Zhang, P. Xiao, A. Zafar, A. u. Quddus, and R. Tafazolli, "Fbmc system: An insight into doubly dispersive channel impact," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 3942–3956, 2017.

[24] J. F. Santos, F. Paisana, N. J. Kaminski, J. M. Marquez-Barja, and L. A. DaSilva, "Context-aware radar modeling framework," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2015, pp. 113–122.

[25] [Online]. Available: https://github.com/srsLTE/srsLTE

[26] J. J. H. Almeida, C. Akamine, and P. B. Lopes, "An implementation on GNURadio of a new model to ISDB-Tb using FBMC," *ISEL Academic Journal of Electronics, Telecommunications and Computers*, 2018.

[27] P. Vijayakumar and S. Malarvizhi, "Reconfigurable Filter Bank Multicarrier Modulation for Cognitive Radio Spectrum Sharing - A SDR Implementation," *Indian Journal of Science and Technology*, 2016.

[28] P. Ruckebusch, S. Giannoulis, E. De Poorter, I. Moerman, I. Tinnirello, D. Garlisi, P. Gallo, N. Kaminski, L. DaSilva, P. Gawlowicz, M. Chwalisz, and A. Zubow, "A unified radio control architecture for prototyping adaptive wireless protocols," in *European Conference on Networks and Communications (EuCNC)*, June 2016, pp. 58–63.

[29] Wishful github repository. [Online]. Available: https://github.com/wishful-project

[30] Wishful project. [Online]. Available: http://www.wishful-project.eu/

[31] O. Russakovsky, J. Deng, and H. Su, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[33] A. Z. Karen Simonyan, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[35] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[36] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and $<$0.5MB model size," 2016.

[37] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 22, no. 2, pp. 869–904, 2020.

[38] F. Paisana, A. Selim, M. Kist, P. Alvarez, J. Tallon, C. Bluemm, A. Puschmann, and L. DaSilva, "Context-aware cognitive radio using deep learning," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2017, pp. 1–2.

[39] T. Ohtsuji, T. Takeuchi, T. Soma, and M. Kitsunezuka, "Noise-tolerant, deep-learning-based radio identification with logarithmic power spectrum," in *IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2021.3071149, IEEE Transactions on Cognitive Communications and Networking

Farrukh Aziz Bhatti received his Ph.D. from The University of Auckland, New Zealand, in 2014, for his work on spectrum sensing for cognitive radios. Since 2014, he is an Assistant Professor in the department of electrical engineering at the Institute of Space Technology, Islamabad, where he is also the director of Wireless and Signal Processing (WiSP) Lab and core member of the iVISION Lab. From 2017-18 he was a Research Fellow at CONNECT Centre, Trinity College Dublin. His research interests include 5G networks, software-defined radios, application of machine learning in wireless communication, biomedical signal processing and computer vision.

Muhammad Jaleed Khan is a PhD student in Artificial Intelligence at the Data Science Institute, National University of Ireland Galway. He is also an adjunct researcher at iVISION Lab, Institute of Space Technology, and a committee member at Pakistan Pattern Recognition Society. His research interests include multimedia event processing, neurosymbolic reasoning, deep learning, computer vision and hyperspectral image analysis.

Ahmed Selim received the Ph.D. degree in wireless communications with Trinity College Dublin in 2012. He was with Alcatel-Lucent Bell-Labs, Ireland, for a year. Since 2014, he has been a Research Fellow with Trinity College Dublin. His research interests include digital signal processing, machine learning techniques for wireless communications, networks management, and computer graphics.

Francisco Paisana received his Masters Degree in electrical engineering and computers from Instituto Superior Tecnico, Portugal in 2012. He subsequently completed his PhD in wireless communications at Trinity College Dublin in 2016. He is currently a senior developer at Software Radio Systems. His research interests include software-defined radio, spectrum sharing, machine learning and digital signal processing.

2332-7731 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on May 15,2021 at 03:48:11 UTC from IEEE Xplore. Restrictions apply.