Radio Access Technology Characterisation Through Object Detection

Erika Fonseca*, Joao F. Santos*, Francisco Paisana[†], and Luiz A. DaSilva[‡]
*CONNECT / the telecommunications research centre
*Trinity College Dublin, Ireland, {*fonsecae, facocalj*}*@tcd.ie*
† Software Radio Systems, *francisco.paisana@softwareradiosystems.com*
[‡]Commonwealth Cyber Initiative, Virginia Tech, USA, *ldasilva@vt.edu*

***Abstract*—Radio Access Technology (RAT) classification and monitoring are essential for efficient coexistence of different communication systems in shared spectrum. Shared spectrum, including operation in license-exempt bands, is envisioned in the fifth generation of wireless technology (5G) standards (e.g., 3GPP Rel. 16). In this paper, we propose a Machine Learning (ML) approach to characterise the spectrum utilisation and facilitate the dynamic access to it. Recent advances in Convolutional Neural Networks (CNNs) enable us to perform waveform classification by processing spectrograms as images. In contrast to other ML methods that can only provide the class of the monitored RATs, the solution we propose can recognise not only different RATs in shared spectrum, but also identify critical parameters such as inter-frame duration, frame duration, centre frequency, and signal bandwidth by using object detection and a feature extraction module to extract features from spectrograms. We have implemented and evaluated our solution using a dataset of commercial transmissions, as well as in a Software-Defined Radio (SDR) testbed environment. The scenario evaluated was the coexistence of WiFi and LTE transmissions in shared spectrum. Our results show that our approach has an accuracy of 96% in the classification of RATs from a dataset that captures transmissions of regular user communications. It also shows that the extracted features can be precise within a margin of 2%, and can detect above 94% of objects under a broad range of transmission power levels and interference conditions.**

***Index Terms*—Dynamic spectrum access, signal detection, object detection, cognitive radio.**

I. Introduction

Spectrum monitoring is necessary for efficient coexistence in shared spectrum; it is also needed for regulators to be able to enforce spectrum policy and identify possible violations [1]. Most of the existing works on spectrum monitoring employ the Received Signal Strength Indicator (RSSI) or energy detection-based methods for detecting the presence of a signal in the channel of interest [1] [2]. However, these approaches may not be effective when multiple Radio Access Technologies (RATs) coexist in the same band [3]. In order to operate efficiently, i.e., in an interference-aware manner, wireless devices operating in shared spectrum must identify other radios and RATs present in the same band before communicating.

For example, 5G introduces, in 3GPP Rel. 16 [4], New Radio Unlicensed (NR-U), with New Radio operating in license-exempt spectrum and therefore required to coexist with other RATs. Even before that, Long-Term Evolution (LTE) in unlicensed spectrum (LTE-U) already must rely on contextual information about the spectrum usage to operate in shared spectrum [5] [6]. In this paper, we propose a solution to detect the presence of other RATs in the band of interest, classify them, and extract features of the transmission, such as centre frequency, bandwidth and duty cycle.

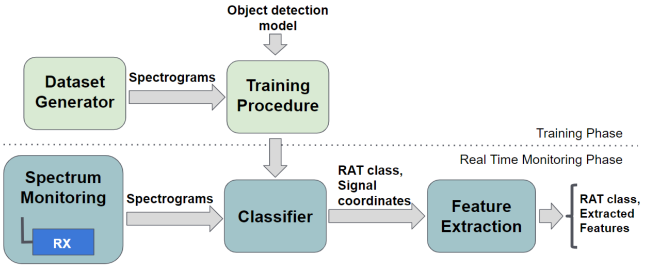


Fig. 1: Overview of the proposed approach for characterising RATs through object detection.

Our solution employs Deep Learning (DL), performing object detection directly on spectrograms, for the characterisation of different RATs in shared spectrum. We not only classify different RATs but also localise the signals in the frequency and time domains, as well as extract features, including centre frequency, bandwidth, frame and inter-frame duration. Our model works well under different levels of received signal strength, and in the presence of overlapping transmissions by multiple radios.

We have also created a dataset generator for training our DL model and evaluating the feature extraction on spectrograms. This was necessary due to limitations of publicly available datasets of commercial transmissions, which do not possess the variations in the frequency and/or time domains required to evaluate the robustness of the proposed approach. Our dataset generator automatically produces labelled bounding boxes within spectrograms, based on the input parameters used in the generation of the signals. The resulting labelled spectrograms are then used for training and validating our feature extraction model. Figure 1 illustrates the key components of our proposed approach.

The remainder of this paper is structured as follows. In

Section II, we provide an overview of related work on Machine Learning (ML) applied to spectrum monitoring. In Section III, we introduce our approach for the characterisation of RATs, composed of a RAT classifier and feature extraction components. In Section IV, we present our dataset generator. Then, in Section V, we evaluate the performance of our classifier and feature extraction under different channel and interference conditions. Finally, in Section VI, we present our concluding remarks and avenues for future work.

II. State-of-the-Art on ML-based RF Signal Classification

Some of the early work on spectrum monitoring has relied on techniques such as cyclostationary feature detection and energy detection [7, 8]. The focus tended to be on detecting the presence of a signal in the band of interest, rather than characterising the signals being detected. More recently, there has been renewed interest in modulation classification, driven by spectrum sharing in military bands, and, in the commercial arena, by the possibility of operating 4G and 5G in unlicensed bands, sharing the spectrum with other RATs such as WiFi and radar communications. Current military and commercial spectrum sharing can benefit from more sophisticated awareness of what other transmissions are present in the band, and what the characteristics of those transmissions are, than the earlier spectrum monitoring solutions were able to provide. This has motivated a number of ML-based solutions for signal classification.

In [9], the authors present several applications of Deep Learning (DL) algorithms for modulation recognition and channel decoding. Other works propose the use of Support Vector Machine (SVM) algorithms [10] or Genetic Programming with K-Nearest Neighbors (GP-KNN) [11] for modulation classification. However, both SVM and GP-KNN techniques are susceptible to frequency and phase offsets, which can compromise the signal classification accuracy under multipath, fading, or other real-world Radio Frequency (RF) impairments. Following works have focused on how to make ML-based models more robust to Signal-to-Noise Ratio (SNR) variations, capturing real-world RF impairments and generating more reliable RF signal classifiers. For example, in [12], the authors generate their dataset through over-the-air transmissions using Universal Software Radio Peripherals (USRPs), evaluating the accuracy of their classification model under different SNRs. The works of [13] and [14] also consider different SNRs when applying ML algorithms for modulation classification. In [13], the authors investigate the classification problem using dictionary learning, and in [14], the authors utilise SVMs. In [15], we characterise the performance achieved by a Convolutional Neural Network (CNN) model when identifying distinct modulations for different SNR levels using spectrograms.

The work of [16] applies computer vision methods to modulation transmission detection. They perform object detection on spectrograms using their own generated dataset. Their model detects the transmitted frames, but it does not classify them. In this paper, we apply object detection to perform not just the detection, but also the classification of different RATs and the identification of key features of the transmission. For this, we rely both on a dataset that we generated in the Iris Testbed [17] and on available datasets of commercial transmissions.

The coexistence between different RATs in shared spectrum requires more information about the surrounding wireless devices than simply the knowledge of their modulation schemes, e.g., QPSK or QAM. For example, different RATs may employ the same type of modulation and yet use different medium access schemes. In [18], the authors propose a long short term memory (LSTM) model for modulation classification in large distributed networks of low-cost sensor nodes. They conclude that a modulation model classifier is not always effective in classifying RATs. The works summarised above provide solutions in the field of signal classification with, predominantly, high accuracy in what they propose to do. However, most of them are focused on modulation classification and not on RAT classification. These works also do not exploit a scenario with varying degrees of interference and overlapping transmissions among the classified signals, which significantly increases the difficulty in correctly classifying these transmissions. In an environment where the spectrum is shared between multiple RATs and multiple access points belonging to the same RAT, e.g. in the ISM band, transmissions that occur in the same frequency channel can happen with full or partial overlap. Recognising these cases and localising them can lead to better interference and/or coexistence management mechanisms.

Our earlier work on RAT classification [3] focused on distinguishing between radar, LTE, and WiFi transmissions. In [19] we analyse different machine learning techniques for wireless technology classification with two different datasets, and check the ability of the model to generalise to unforeseen scenarios. These solutions are effective in performing RAT classification but do not provide a more detailed characterisation of the spectrum.

A new approach, presented for the first time in this paper, applies object detection on spectrograms, which allow us to classify and extract key features of the sensed transmissions. The advantage of applying object detection to the RAT characterisation problem is that this technique identifies the object independently of its location in the image, allowing the detection and classification of transmissions that do not occur in the centre of the band being observed, with different bandwidths, different duty cycles, etc.. To the best of our knowledge, our work is the first to detect and classify different RAT transmissions applying object detection techniques and to evaluate its performance under unfavourable noise and interference conditions. Our solution also provides feature extraction functionality, which can be used to build efficient dynamic coexistence mechanisms in shared spectrum. Table I summarises the results of our analysis.

III. RAT Characterisation

In this section, we describe our solution for RAT characterisation using object detection. Our approach consists of two

TABLE I: Qualitative evaluation of the features in ML-based RF signal classifiers.

Paper	Created Dataset	Techniques	Signal Classification	Transmissions Impairments	Overlapping Transmissions	Feature extraction
F. Paisana et al. [7]	Yes	Cyclostationary Detection	No	Yes	No	No
F.F. Digham et al. [8]	No	Energy Detection	No	No	No	No
L. Han et al. [10]	No	SVM	Modulation	No	No	No
M. Aslam et al. [11]	No	GP-KNN	Modulation	No	No	No
T. OShea et al.[12]	Yes	Gradient Tree	Modulation	Yes	No	No
K. Zhang et al. [13]	No	Dictionary Learning	Modulation	No	No	No
Z. Wu et al. [14]	No	SVM	Modulation	No	No	No
F. Paisana et al. [15]	Yes	CNN	Modulation	Yes	No	No
T. OShea et al. [16]	Yes	Object Detection	No	Yes	No	No
S. Rajendran et al. [18]	Yes	LSTM	Modulation	Yes	No	No
A. Selim et al. [3]	Yes	CNN	RAT	Yes	No	No
J. Fontaine et al. [19]	Yes	FNN, RForest, CNN	RAT	Yes	No	No
F. Erika et al. [this paper]	Yes	Object Detection	RAT	Yes	Yes	Yes

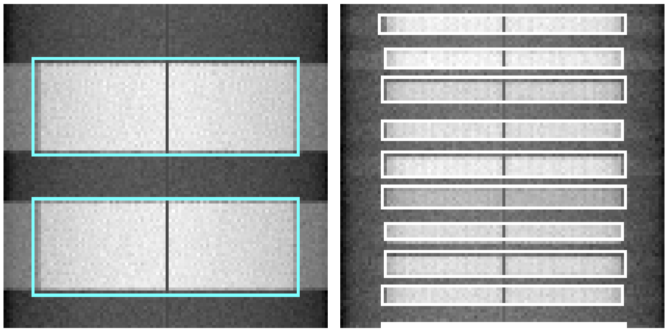
main components, an image-based RAT classifier, and post-processing feature extraction, as shown in Figure 1. In the following subsections, we describe each of these.

A. Image-based RAT Classifier

We developed a CNN-based classifier for recognising different RATs coexisting in shared spectrum. Our classifier can identify multiple RATs by directly applying object detection to spectrograms. The CNN must be trained and validated against target objects. Depending on the size of the neural network and the computing platform available, the training and validation of the CNN from scratch may take between hours and days. One way to reduce this time is by applying transfer learning, which relies on the partial reuse of a previously trained model (trained on a different set of tasks) for addressing a new task. This implies retraining an existing network, typically by fine tuning the weights from the hidden layers close to the output layer, to make the network more suitable to the new task. As such, the first layers, which are typically good at extracting basic features such as edge detection in computer vision tasks, are reused for the new task as well. Transfer learning significantly decreases the amount of data required for the training process and, consequently, the duration of the training process.

The application of transfer learning requires the choice of a previously trained network as a starting point. A broad range of pre-trained networks already exists; these are suitable for different problems, e.g., predictive text, speech recognition, and image object detection. For the spectrum sharing scenario, where it is necessary to dynamically assess how the spectrum is being occupied, we need a model that can provide acceptable classification accuracy in real-time. We also require a solution that can provide not just the classification of the object, but also its localisation in the image (as discussed later, we rely on this localisation information for feature extraction).

We employ the well-known object detection model You Only Look Once (YOLO) [20] as the starting point for our RAT classifier. YOLO is one of the most efficient solutions in the literature for real-time implementation of object detection. This model outputs both the class of the detected objects, as well as their position in the input image. Using weights and architecture from YOLO pre-trained on ImageNet [21], we modify the Softmax layer, which corresponds to the last layer before the output of the model. During the training process, the Softmax layer is explicitly optimised for the classification of LTE and WiFi waveforms. The architecture we adopted is presented in [22] and it has 19 convolution layers and 5 max-pooling layers. Moreover, our model can easily be extended for supporting more RATs, by retraining it with datasets that include new waveforms.



(a) LTE detection.

(b) WiFi detection.

Fig. 2: Spectrogram with the bounding boxes created by our ML-based signal classifier. The positions of the bounding boxes represent the detection of the frame, and the colour represents the classification, blue for LTE and white for WiFi.

The training itself requires the fine-tuning of parameters related to the learning rate and convergence of the classification, known as the hyperparameters. Hyperparameters are parameters chosen before the training process, for example, learning rate, optimiser, epochs, etc. We detail our choices for the hyperparameters below:

- Learning Rate: is the amount by which the weights in an ML model are updated. We set it to 10^{-5}; with this value, the model did not overfit and was able to learn the objects' characteristics.
- Epoch: is an iteration of the training process where the model is filled with all the elements of the training dataset. If a model is trained with too many epochs, it can overfit to the training data, while if a model uses too few epochs, it might not learn the necessary features to perform the classification. After testing several values, we set the number of epochs to 50,000.
- Mini-batch: is a part of the dataset used to update the network's weights. The first approaches in ML used the entire dataset to update the weights in the network; however, the work of [23] argues that this update should use a smaller part of the dataset, called a mini-batch. The

TABLE II: The mapping between the image position and the parameters of interest in time and frequency domains.

Parameters Time/Frequency	Position Mapping
I_t	$(t_2 - t_1)/(Y_{max} - Y_{min})$
I_f	$(f_2 - f_1)/(X_{max} - X_{min})$
b_w	$(x_{max} - x_{min}) * I_f$
f_c	$f_1 + (I_f * x_{min}) + (b_w/2)$
FD	$(y_{max} - y_{min}) * I_t$
CWT	$(t_2 - t_1) - (frame\_rate * f_{av})$
FI	$CWT/frame\_rate$

mini-batch approach can increase model performance when it uses batches with values between 2 to 32 [23]. In the early stages of the design process of our solution, we observed good performance when setting the mini-batch to 32.

- Optimiser: is the function that modifies the weights of each neuron with the purpose of minimising the loss function. The loss function indicates how close the output of the model is to the expected result. The main objective of the learning process is to optimise the loss function, making the predicted output closer to the expected one without over-fitting to the training data. We chose the optimiser Adaptive Moment Optimisation (Adam) because it has the feature of accelerating the search for the minimum value of the loss function and reducing oscillations.

After trained, our model produces the identification of the RAT (i.e., the result of the classification) and the coordinates of each frame detected in the spectrogram image. Figure 2 shows examples of LTE and WiFi frames detected, surrounded by bounding boxes: blue for LTE, white for WiFi. The four coordinates of each of these bounding boxes are used by the feature extraction component, discussed next.

Once our model is trained and validated, it can provide results on the fly, making it suitable for real time applications. Our classifier analyses frames in batches of three frames each, providing three outputs at the same time; this allows us to parallelise the classification task and use multiple cores in parallel. A trade-off that is important to consider is the implication of this design choice on real-time detection and RAT classification: the number of images analysed simultaneously cannot be too large, otherwise the model will not operate in real-time. In our implementation, we evaluated the classification speed using a computerwith Intel Core i7-6820HK CPU and GeForce GTX 1070 Mobile. With this commercial off-the-shelf computer, we are able to analyse three images in around 0.1ms with 2 classes and trained with a commercial transmission dataset (described later).

B. Post-processing Feature Extraction

Once the classification of the RAT is completed, the feature extraction component allows us to obtain additional information about the RATs present in a given channel. The spectrogram corresponds to a band of frequencies [f_1, f_2], collected during a time interval [t_1, t_2]. Then, we calculate the granularity that each pixel in the image represents in the time and frequency domains, as an increment value in time (I_T) and frequency (I_F), respectively. This mapping depends on the size of the spectrogram ($[X_{min}, X_{max}]$, $[Y_{min}, Y_{max}]$)[1].

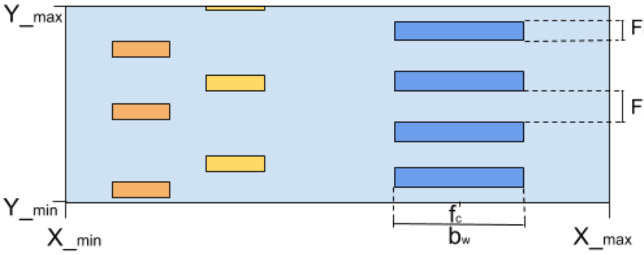


Fig. 3: Parameters representation in a spectrogram.

The trained model provides the corners of a rectangle that encloses a transmission frame, denoted by coordinates $x_{min}, x_{max}, y_{min}, y_{max}$. Given the coordinates of this rectangle, i.e., the bounding box, as well as the values of each time and frequency increment, we can localise the signals in the spectrum and in time. In order to calculate the bandwidth of the signal (b_w) and its centre frequency (f_c), we use the horizontal coordinates of the corners of the bounding box, translating them into their respective value in frequency. The Frame Duration (FD) of the signal is calculated in a similar manner, but now using the vertical coordinates of the corners of the bounding box. To calculate the average Frame Interval (FI), we must first calculate the average time the channel stays without a transmission (CWT), which is the total time represented in a spectrogram subtracted by the time that is occupied by frame transmissions. Then, the FI is given by CWT divided by the number of transmissions on the spectrogram. We summarise the formulas we use for extracting the features of different RATs in Table II, and illustrate the representation of the relevant values on a spectrogram in Figure 3.

IV. Dataset Generation

In the literature, several research efforts, as the ones mentioned in Table I, have raised the issue of RF dataset scarcity and had to face the challenge of creating an RF dataset before the development and validation of their ML-based classifier. The lack of labelled RF datasets inhibits the development and testing of new CNN-based solutions, and the manual collection of large RF datasets is time-demanding and error-prone.

For the testing and validation of our proposed solution, we have relied on datasets of LTE and WiFi transmissions collected over different locations in Belgium [24]. However, such datasets of commercial transmissions are not sufficient for the complete evaluation of our feature extraction component. To evaluate that component it is necessary to have the ground truth for the parameters of the transmissions, as this evaluation is related to the position of the signal in the spectrum and in time. In the case of commercial transmissions captured over the air, it is not possible to determine precisely the ground truth, and also it is not possible to vary the SNR

[1] Note that uppercase X and Y refer to the spectrogram, and lowercase x and y refer to the bounding box around a frame.

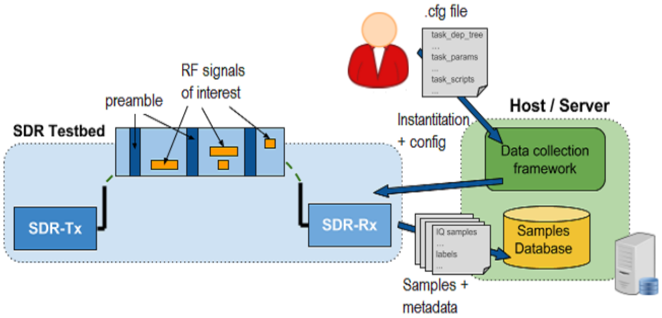


Fig. 4: Generating, collecting, and labelling RAT transmissions using our dataset generator.

of the transmission, its centre frequency or its bandwidth, for example. To tackle this issue, we designed and implemented a dataset generator for the creation of labelled RF datasets, based on waveforms that mimic the transmissions of LTE and WiFi radios. As we are transmitting and receiving the signals, we have full control and knowledge about parameters a priori so that we can generate the ground truth label of the transmissions. This allows the evaluation of the feature extraction that is essential for validating our solution. We relied on SRS LTE [25] for the generation of LTE signals and on a GNU Radioimplementation [26] for the generation of WiFi signals.

To produce a dataset that reflects real-world transmissions, the dataset must be collected over-the-air to produce samples that undergo RF impairments such as phase/frequency offsets, phase noise, amplifier non linearity, etc. The use of Software-Defined Radios (SDRs) enables unparalleled programmability and reconfiguration. While traditional radios implement their protocol stack in hardware, SDRs implement their protocol stack in software. Hence, a RAT upgrade or revision can be deployed through a simple software update, a capability leveraged to create our dataset generator.

Our dataset generator uses an SDR to generate waveforms of different signal strength and bandwidth. It automates the collection and labelling of over-the-air samples of the waveforms of different RATs. Figure 4 depicts the process of generating, collecting and labelling RF waveforms using our dataset generator. The user specifies the waveforms and their parameters. Then, our RF dataset generator creates signal traces with all the permutations of parameters, as well as transmitting, collecting, and labelling the signal traces.These RF waveform datasets can be used for training and testing of deep CNNs for signal classification and spectrum monitoring.

A. *Tree of tasks*

Our dataset generator allows the generation of datasets with different: (i) waveforms, e.g., WiFi, LTE, and Phase-Shift Keying (PSK) signals; (ii) waveform-specific features, e.g., modulation order and frame length, and DSP transformations, e.g., Frequency Offset (FO), soft gains, shape filtering, and multipath emulation; (iii) RF parameters, e.g., centre frequency, hardware gains. Each permutation of parameters and waveform types is translated into IQ signals that are transmitted over the air between SDRs. Then, the received IQ signals and the associated parameters are stored in data files for later access.

We developed a pipeline-based approach for generating traces of RF waveforms with different characteristics. The process is implemented as a graph of individual tasks, e.g., producing a waveform, setting the frame duration, and setting the transmission gain. Each task can be configured and run independently. Each of the task's parameters can be a list of different values, and the task generates respective output files for all the input values. The subsequent task receives a set of different input files from the previous task and performs its operation on all of them. Such a pipeline-based approach facilitates the extension and inclusion of new tasks, the parallelisation of tasks, and resuming from intermediate points.

B. *Synchronisation and Channel Estimation*

A compelling aspect of our dataset generator is the automatic labelling provided by it, as this is essential for the process of training an ML model. The labelling is created in different formats, including the Visual Object Classes (VOC) format that is used in object detection approaches. To provide automatic labelling, it is essential to keep the SDR transmitter and receiver synchronised so that the labels of their transmitted and received samples remain consistent. We accomplish the synchronisation and channel estimation through the periodic transmission of preambles.

The preamble used during dataset generation needs to display strong robustness to noise, so that it can collect samples at the low SNR levels that are generally required in RF signal/waveform classification use cases. We chose a preamble structure composed of several short Zadoff-Chu sequences with absolute phase shift by an m-sequence for coarse frequency, time offset estimation, and disambiguation, followed by a long Zadoff-Chu sequence for precise frequency offset estimation. For this study, we selected a preamble length of 1031 samples to guarantee robust synchronisation, with probability of preamble detection close to 1 even for values of SNR lower than -5 dB. However, if the user of the dataset generator needs to collect data in an environment under less favourable channel conditions, the length of the preamble can be increased at a cost of additional synchronisation time. The drawback of increasing the preamble size is the longer processing time spent by the framework finding the preamble in the collected samples. Our synchronisation allowed our preamble detector to cope with sampling rates of 20 MS/s in a modern CPU.Whenever preamble synchronisation fails, the generator triggers a retransmission.

V. Performance Evaluation

In this section, we evaluate our solution for RAT characterisation through object detection. First, we evaluate the detection and classification performance of our model for different RF waveforms under different channel conditions.

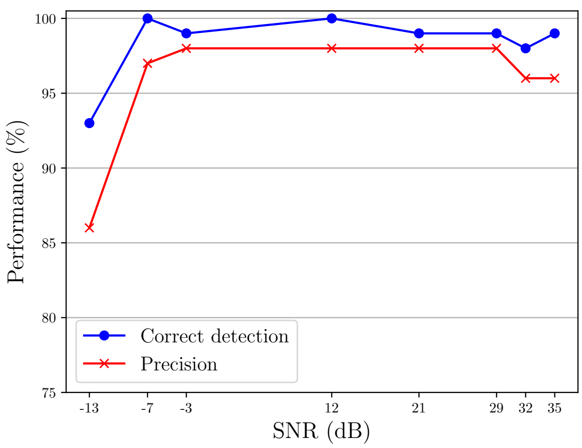


Fig. 5: Percentage of correctly detected objects and precision as a function of SNR.

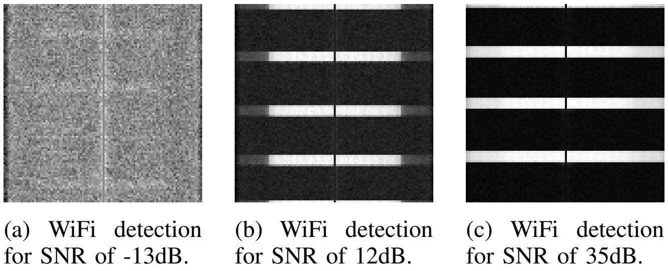
Next, we assess the feature extraction component of our solution. Then, we estimate the accuracy of the RAT classifier and feature extraction components using data from commercial transmissions.

A. Detection and Classification Performance

In this section, we evaluate the detection performance and classification accuracy of our model, and demonstrate its robustness in detecting and classifying RF waveforms under different SNR conditions and interference levels. We used the dataset generator described in the previous section to compose a dataset of images, i.e., spectrograms and labels, for two radio access technology classes, LTE and WiFi. This scenario resembles real world use cases of coexistence in unlicensed spectrum [27]. Moreover, our model can be extended, for instance, by increasing the diversity of the RATs included in the training dataset. Extending the training dataset might be useful in a scenario where a technology operating in the unlicensed spectrum might share it with Bluetooth or Zigbee, for example.

1) Performance of the Classifier Under Different SNRs: In this analysis, we evaluate the detection and classification performance of our solution under different SNR conditions. For this evaluation, we generated a dataset with different levels of transmission power, measuring the SNR at the receiver side. We used 400 images to train the model and adopted the configuration described in Section III, which empirically produced satisfying results. As explained in Section IV, our dataset generator has a minimum SNR threshold value for synchronisation of the preamble over-the-air. The measurements start with an SNR value of -13dB and go up to 35dB. Each spectrogram represents a 50ms time interval and a 20MHz band.

First, we are interested in assessing the ability of our model to detect the transmitted frames correctly. The object detection curve in Figure 5 shows the percentage of correctly detected frames as a function of SNR. Detection is around 98% for all SNR values tested, except -13 dB: at that SNR, the edges of the transmitted frames are not as sharp, as illustrated in Figure 6, resulting in a lower probability of detection. Energy detection proposed in [8] proved that at 20 dB the technique should detect transmission in 100% of the cases. An implementation of the energy detection used in 802.11 [26] showed that the minimum SNR to consider a transmission on the channel is 5dB, and its accuracy is close to 100% detection above this value. For higher SNRs, the energy detection solution presents better performance for detecting frames. In energy detection approaches, the detected energy has to be above a threshold for the channel to be considered busy. If the energy is lower than the threshold, it is then considered that there is no transmission at the channel. Once the SNR increases, the performance of this approach improves as the distance to the threshold raise. When energy detection is compared to our solution at lower SNRs, object detection can detect the presence of transmissions with better performance being limited to the sharpness of the frames on the image.



(a) WiFi detection for SNR of -13dB. (b) WiFi detection for SNR of 12dB. (c) WiFi detection for SNR of 35dB.

Fig. 6: Illustration of WiFi signals under different SNRs.

Next, we are interested in assessing our model's ability to classify the detected frames. The precision metric is commonly used in classification problems [28], and it represents the percentage of all detected frames that are correctly classified. The precision is shown in Figure 5 and varies from 86% for an SNR of - 13dB to 98% at SNR between -3 and 32dB. For the highest SNRs, 32 and 35dBs, we obtained an accuracy of 96%. It is worth mentioning that when the SNR is very high the leakage in the transmission also increases, as illustrated in Figure 6, which in our evaluation compromised 2% of classification accuracy.

Figure 5 shows that when the SNR is low, both the ability to detect the frame and to correctly classify it are impaired. Although the higher leakage does not influence the ability to detect the frames, it slightly affects the classification performance.

2) Interfering Transmissions Under Different SNRs: In this analysis, we evaluate the ability of our model to detect and classify frames under the effect of cross-technology interference. We consider two signals with the same bandwidth: the desired signal is an LTE transmission, and the interfering signal is a WiFi transmission. The desired signal is transmitted with an SNR of 29 dB, and the SNR of the interfering signal varies between 3dB to 35dB, both in the same centre frequency and with 20MHz of bandwidth. The spectrograms have the same characteristics mentioned in the

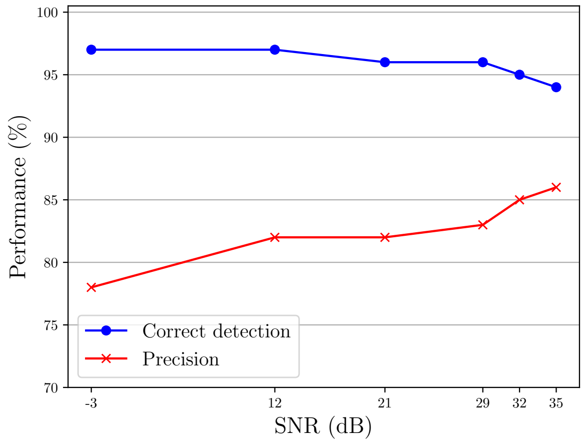


Fig. 7: Correct detection and precision per SNR of the interference signal.

previous section.

Figure 7 shows the results of our experiment. The model could detect the LTE frames 97% of the time, with this accuracy declining slightly as the SNR of the interfering WiFi transmission increases. The curve representing the precision of the model shows that it improves in classifying the frames once the SNR of the WiFi signal increases. This happens because when the interfering WiFi frames had lower SNR, the model had issues clearly classifying the transmissions as either LTE or WiFi. However, once the SNR of the WiFi is higher than the SNR of the LTE transmissions, the model is more successful on classifying them, achieving 86% of accuracy.

These results show that even in a scenario of strong cross-technology interference, our model is capable of detecting the frames and classifying different RATs, providing a reasonable characterisation of the environment. To the best of our knowledge, this is the first work to assess the performance of an ML model for RAT classification under the effect of interference with overlapping transmissions.

B. Feature Extraction

To evaluate the capabilities of our feature extraction component, we generated several datasets using different combinations of: transmission bandwidths, frame duration, inter-frame duration, and centre frequency. The average SNR of the transmissions in this evaluation is 29 dB. Figure 8 illustrates the accuracy in the feature extraction, for different transmission characteristics. In our experiments, the value of the I_f is 192.307KHz, which means that each pixel in the spectrograms accounts for a variation of 192.307KHz in the frequency domain. For example, if the calculated centre frequency is off by a single pixel, the computed value will deviate 192.307KHz from the correct centre frequency. The same applies in the time domain, where each pixel accounts for a variation of $I_t = 519\mu$s.

Figures 8a and 8b illustrate the accuracy in the extraction of frequency domain features. For all cases tested, the median deviation from the ground truth is at most 2%.

The results of the extraction of time-domain features are shown in Figures 8c and 8d. For these, the median deviation from the ground truth is at most 4%. Figure 8c illustrates that when the frame duration to be detected is smaller, the solution tends to have an average error higher than when the frame has a longer duration. This happens because it is harder to identify the precise size of smaller objects. As depicted in Figure 8d, the extraction of inter-frame duration shows similar accuracy. Our model detects a high percentage of the transmitted frames. Whenever the model fails to detect a frame, it assumes that the spectrum is empty for that period, increasing the extracted inter-frame duration. However, even in those cases, our model achieves a median deviation of less than 2 percent in all the cases.

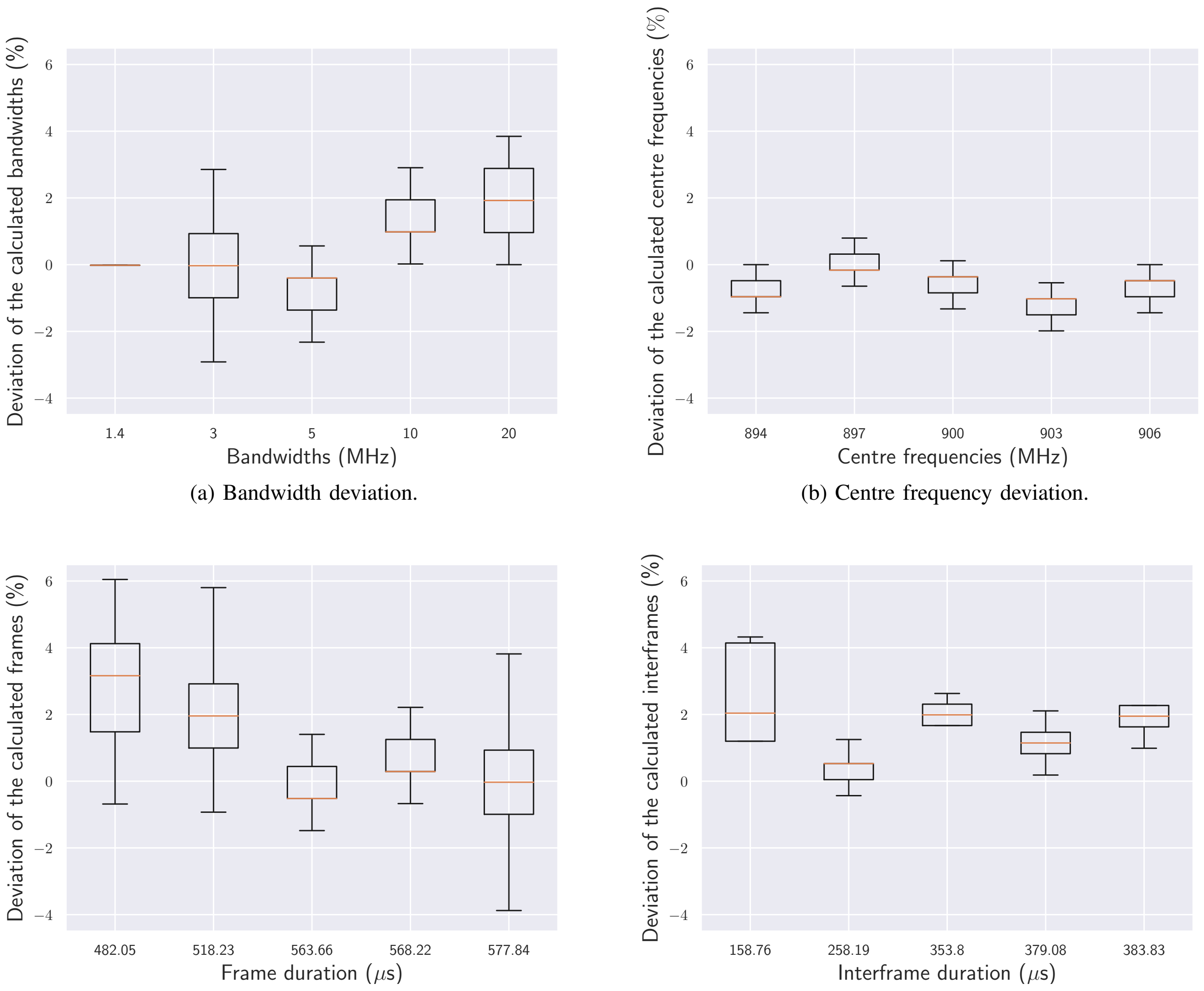
Considering the results discussed in this subsection, we can conclude that our model is capable of extracting the signal features with high precision. Moreover, if necessary for specific applications, a higher precision can be achieved by using higher-resolution spectrograms, i.e., smaller I_f and I_t values.

C. Performance Comparison Using Public Datasets

In this section, we evaluate our model using a publicly available dataset of commercial LTE and WiFi transmissions collected in Belgium. This evaluation is crucial because it shows that our model can work in real-world scenarios. First, we investigate the accuracy of our model as a function of the number of spectrograms in the training dataset. Then, to demonstrate the ability of our object detection model to classify commercial transmissions accurately, we compare our solution to the ones proposed in [19], which used the same publicly available dataset.

We start by analysing how the number of the samples (spectrogram images) affects the performance of the proposed model. The volume of training data can limit the application of ML, because ML techniques usually require a considerable amount of data to learn. For example, the work of [19] used more than 12 thousand images for training the CNN solution based on spectrograms. In this section, we assess the performance of our model, considering the volume of training data.

We assessed the detection performance of our model against the size of the training dataset until we achieve performance similar to state of the art.We repeated the training in an identical setup while only adjusting the number of spectrograms used: 2, 10, 20, 30, 40, 50, 100, 200, and 400. The training samples equally represent the LTE and WiFi classes, which means that if the training set has 20 spectrograms, 10 are from LTE transmissions and 10 are from WiFi transmissions.Figure 9 illustrates how accuracy scales with the number of spectrograms. The best accuracy achieved was 96% with 400 spectrograms. Hence, we limited the size of our training dataset to 400 images, as this volume of training data is sufficient for our model to achieve a comparable accuracy



(a) Bandwidth deviation.

(b) Centre frequency deviation.

(c) Frame duration deviation.

(d) Inter-frame duration deviation.

Fig. 8: Feature extraction deviation evaluation in time and frequency domain.

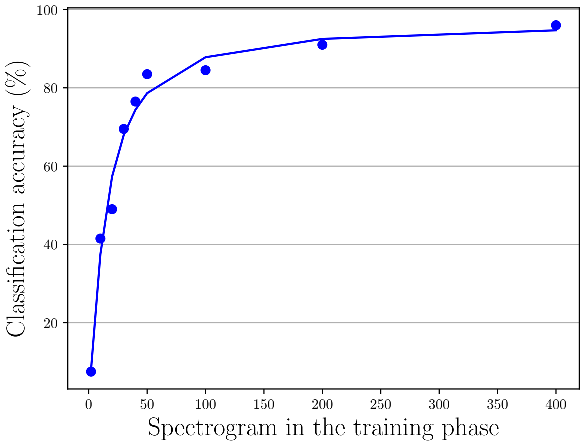


Fig. 9: Number of spectrogram in the training phase versus accuracy of the model.

to the CNN image-based solution presented in [19], while using a considerably lower number of training images (only 3.23% of the dataset size used in [19]). From Figure 9, we can estimate the number of labelled spectrograms required in training to achieve a certain accuracy.

We then compared the object detection-based classification solution presented in this paper against other RAT classification solutions in [19]. These solutions include a fully connected neural network (FNN), a Random Forest (RForest) [29], a CNN solution based on RSSI, a CNN solution based on IQ samples, and a CNN solution based on spectrograms. The results of this comparison are shown in Figure 10. The CNN-based solutions, including the solution presented in this paper, correctly identify the RAT with accuracy above 95%. The CNNs for IQ and image-based solutions achieve marginally better accuracy compared to our proposed solution. However, our solution provides additional information regarding spectrum usage that can enhance the efficient use of the spectrum.

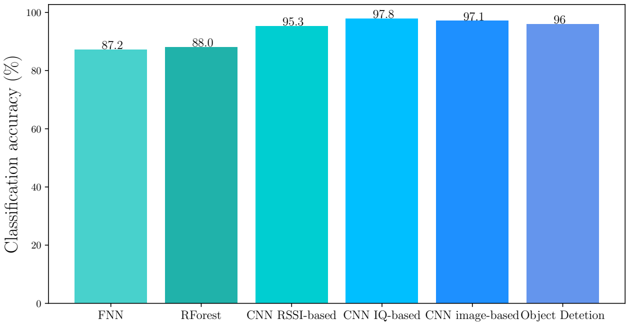


Fig. 10: Classification accuracy of different ML solutions.

VI. Conclusion

In this paper, we presented an ML-based classifier for RAT characterisation using object detection. To the best of our knowledge, this is the first work that evaluates the use of an object detection model for radio technology classification, under multiple interference conditions and employing real user data. Our proposed approach combines the application of object detection on spectrograms for classifying different RATs and a feature extraction component for further characterising the RATs. From spectrogram images, we can extract specific features from the RAT, e.g., inter-frame duration, frame duration, centre frequency, and signal bandwidth. To evaluate the classification accuracy of our model, we trained and classified spectrograms created using the public dataset [24] that was collected in different locations in Belgium. To evaluate the feature extraction component of our approach, we developed a prototype implementation of the RAT classifier in software defined radio. We trained our classifier with LTE and WiFi waveforms and showed its efficiency in detecting and classifying different RATs. Furthermore, we evaluated the resilience of our feature extraction component through transmissions over-the-air with different bandwidths and centre frequencies under distinct SNRs and with overlapping transmissions in the same band.

Our solution can be useful in spectrum monitoring applications and to facilitate the coexistence of different RATs in shared spectrum, as envisioned in 5G. However, we believe that there are improvements to be made in the generation of the labelled data from commercial transmissions. For instance, the process of manually labelling data is time consuming and error-prone.

The source code of the prototype implementation of our dataset generator can be found on GitHub [30] for local use with USRPs, or remote use in the Iris Testbed [17]. Our implementation is based on widely used frameworks, such as GNU Radio, for digital signal processing, and YOLO, for real-time object detection. This facilitates the use of our proposed approach by the community and enables further potential applications related to spectrum sensing.

Acknowledgements

The research leading to this work received funding from the European Horizon 2020 Program under the grant agreement No. 732174 (ORCA project). In addition, this work was partly funded by Science Foundation Ireland (SFI) and the National Natural Science Foundation of China (NSFC) under the SFI-NSFC Partnership Programme Grant Number 17/NSFC/5224.

References

[1] J. Peha, "Sharing Spectrum Through Spectrum Policy Reform and Cognitive Radio," *Proceedings of the IEEE*, vol. 97, no. 4, 2009.

[2] X. Ying *et al.*, "Incentivizing Crowdsourcing for Radio Environment Mapping with Statistical Interpolation," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2015.

[3] A. Selim *et al.*, "Spectrum Monitoring for Radar Bands Using Deep Convolutional Neural Networks," in *IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2017.

[4] 3rd Generation Partnership Project, "3GPP TR 38.889: Study on NR-based Access to Unlicensed Spectrum (Release 16)," 3GPP, Tech. Rep., December 2018, accessed: 2019-06-13.

[5] Qualcomm, "Qualcomm Research LTE in Unlicensed Spectrum: Harmonious Coexistence with WiFi," Qualcomm, Tech. Rep., June 2014, accessed: 2020-03-06.

[6] Wang, Xuyu *et al.*, "A Survey of LTE WIFI Coexistence in Unlicensed Bands," *GetMobile: Mobile Computing and Communications*, Jan. 2017.

[7] F. Paisana *et al.*, "An Alternative Implementation of a Cyclostationary Detector," in *International Symposium on Wireless Personal Multimedia Communications (WPMC)*. IEEE, 2012.

[8] F.F. Digham *et al.*, "On the energy detection of unknown signals over fading channels," in *IEEE International Conference on Communications*, vol. 5. IEEE, 2003.

[9] T. Wang *et al.*, "Deep Learning for Wireless Physical Layer: Opportunities and Challenges," *China Communications*, vol. 14, no. 11, 2017.

[10] L. Han *et al.*, "Low Complexity Automatic Modulation Classification Based on Order-Statistics," *IEEE Transactions on Wireless Communications*, vol. 16, no. 1, 2017.

[11] M. Aslam *et al.*, "Automatic Modulation Classification Using Combination of Genetic Programming and KNN," *IEEE Transactions on Wireless Communications*, vol. 11, no. 8, 2012.

[12] T. O'Shea *et al.*, "Over-the-Air Deep Learning Based Radio Signal Classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, 2018.

[13] K. Zhang *et al.*, "A Dictionary Learning Based Automatic Modulation Classification Method," *IEEE Access*, vol. 6, 2018.

[14] Z. Wu *et al.*, "Robust Automatic Modulation Classification under Varying Noise Conditions," *IEEE Access*, vol. 5, 2017.

[15] F. Paisana *et al.*, "Context-Aware Cognitive Radio Using Deep Learning," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2017.

[16] T. O'Shea *et al.*, "Learning robust general radio signal detection using computer vision methods," in *IEEE 51st Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2017.

[17] L. Doyle *et al.* , "Experiences from the Iris Testbed in Dynamic Spectrum Access and Cognitive Radio Experimentation," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2010.

[18] S. Rajendran *et al.*, "Deep Learning Models for Wireless Signal Classification With Distributed Low-Cost Spectrum Sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, 2018.

[19] J. Fontaine *et al.*, "Towards Low-Complexity Wireless Technology Classification Across Multiple Environments," vol. 91. Elsevier, 2019.

[20] J. Redmon *et al.*, "You Only Look Once: Unified, Real-Time Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[21] J. Deng *et al.*, "ImageNet: A Large-Scale Hierarchical Image Database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009.

[22] J. Redmon *et al.*, "YOLO9000: Better, Faster, Stronger," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[23] Masters, Dominic *et al.*, "Revisiting Small Batch Training for deep Neural Networks," *preprint arXiv:1804.07612*, 2018, accessed: 2019-06-13.

[24] Shahid, Adnam, "Technology Classification Dataset," https://github.com/ewine-project/Technology-classification-dataset, accessed: 2019-06-13.

[25] "Software Radio Systems," https://www.softwareradiosystems.com/tag/srslte/, accessed: 2019-05-21.

[26] Bloessl, Bastian, "IEEE 802.11 a/g/p transceiver for GNU Radio," https://github.com/bastibl/gr-ieee802-11, accessed: 2019-06-13.

[27] A. A. Alabdel Abass *et al.*, "WiFi/LTE-U Coexistence: An Evolutionary Game Approach," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 1, 2019.

[28] Powers, David Martin, "Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation," Tech. Rep., December 2011, accessed: 2020-02-22.

[29] Svetnik, Vladimir *et al.*, "Random Forest: a Classification and Regression Tool for Compound Classification and QSAR Modeling," *Journal of Chemical Information and Computer Sciences*, vol. 43, no. 6, 2003.

[30] Paisana, Francisco, "Dataset Generator of RF Waveforms," https://github.com/frankist/gr-specmonitor, accessed: 2019-01-18.