Deep Learning for Radar Signal Detection in the 3.5 GHz CBRS Band

Raied Caromi*, Alex Lackpour†, Kassem Kallas*, Thao Nguyen* and Michael Souryal*‡
*NIST, Communications Technology Laboratory, USA
{raied.caromi, kassem.kallas, thao.t.nguyen}@nist.gov
†Drexel University, USA
alackpour@drexel.edu

***Abstract*—This paper presents a comprehensive framework for generating radio frequency (RF) datasets, designing deep learning (DL) detectors, and evaluating their detection performance using both simulated and experimental test data. The proposed tools and techniques are developed in the context of dynamic spectrum use for the 3.5 GHz Citizens Broadband Radio Service (CBRS), but they can be utilized and expanded for standardization of machine learned spectrum awareness technologies and methods. In the CBRS band, environmental sensing capability (ESC) sensors are required to detect the presence of federal incumbent signals and trigger protection mechanisms when necessary. To support the development and evaluation of detection techniques for ESC sensors, we provide software tools for generation and augmentation of simulated radar datasets as well as baseline DL detectors that can be replicated, evaluated, and tested in a simulated or an experimental environment. We find that all the proposed detectors exceed ESC requirements for incumbent detection. The software tools, the pre-trained DL models and their configurations, and the experimental setup are made available in the public domain.**

***Index Terms*—3.5 GHz, CBRS, deep learning, environmental sensing capability sensor, machine learning, radar detection, RFML, RF dataset generation.**

I. Introduction

The success of recent auction in the 3.5 GHz Citizens Broadband Radio Service (CBRS) has paved a new path and created many opportunities for spectrum sharing among different stakeholders [1]. In this multi-tiered framework, federal incumbent radar systems, e.g., SPN-43, must be protected from harmful interference from lower tier commercial users [2]. An environmental sensing capability (ESC)—a network of sensors deployed along the coasts—is utilized to detect the presence of incumbent radar signal and to trigger protection alerts. In [3], the authors show that classical matched-filter detectors can provide feasible detection results in the presence of co-channel interference from commercial users as well as out-of-band emissions from adjacent-band radars. However, these detection techniques usually require full or partial knowledge of the radar waveforms, which might not be available, especially for advanced, classified radar systems.

Having been successfully applied to many areas, machine learning (ML) and deep learning (DL) techniques can offer appealing solutions to the radar detection problem. The survey in [4] and the tutorial in [5] provide numerous ML and DL applications for mobile and wireless networks. These techniques can be used for cognitive radios [6] and physical layer applications [7]. Higher-order statistics and cumulant features are proposed for signal detection and classification in [8]–[10]. Signal classification problem can also be solved by using support vector machine (SVM) and various DL techniques [11]–[14]. Furthermore, DL techniques are also utilized for signal detection in [15]–[20]. In [21], the authors use over 14 000 spectrograms collected in the 3.5 GHz band to evaluate the performance for SPN-43 radar detection of three methods including a classical energy detection, a convolutional neural network (CNN), and a long short-term memory recurrent neural network (LSTM RNN). In [22], ML-based radar detectors are proposed and evaluated using field-measured radar waveforms in the presence of out-of-band emissions and Long Term Evolution (LTE) interference signals. Finally, multiple deep learning models are studied and evaluated for ESC radar detection in [23]. Although field-measured radar waveforms are best suited for testing ML and DL algorithms in real world scenarios, they often lack the ground truth and are not widely available for general public due to operational security concerns of the incumbent radar systems.

Related to this paper, the IEEE Dynamic Spectrum Access Networks Standards Committee (DySPAN-SC) [24] recently created the IEEE 1900.8 Working Group (WG) to standardize the storage format for structured Radio Frequency Machine Learning (RFML) datasets and the interfaces that connect stages of the RFML model training pipeline. The IEEE 1900.8 standard will address use cases for RF signal detection, classification, and characterization as well as identification of RF emitters. The IEEE 1900.8 WG is considering adoption of the National Institute of Standards and Technology (NIST) radar waveform generator software as part of a reference workflow for generating RFML datasets.

The aim of this work is to provide an alternative to using classical detection techniques and limited field measurements data. We use radar detection in the 3.5 GHz CBRS band as a case study to present our RF data generation and DL detection techniques. Nevertheless, the techniques developed in this paper can be used for other cases of RF signal detection and augmentation of simulated RF data. Fig. 1 shows the workflow for using our dataset, tools, and models to develop

‡This paper is dedicated to the memory of Dr. Michael Souryal (1968–2020).

and test radar detectors for the CBRS band. The workflow demonstrates the flexibility of our framework. The experience can range from full development of the models and generating the data to only using our pre-trained models for inference. For example, the user can use our published dataset, or generate a new dataset and adds more testing waveforms as needed using our RF dataset generator. On the other hand, another user may choose to only use the practical implementation approach with experimental data followed by the performance evaluation step. In a different example, the user can start development by using our pre-trained models, or develop new models without the need to invest time and resources in generating a dataset. Hence, our framework is flexible since it expedites the development and testing without the need to go through all the steps in the process.

We present an in-house developed software tool that can generate various radar waveforms and augment these waveforms with interference to create applicable RF datasets [25]. Using the simulated datasets, we then train and test several DL detection techniques, as a baseline, to detect the presence of radar signal. We formulate the detection problem as a binary classification problem and utilize CNN as well as selected state-of-the-art classification networks to solve it. In addition, we select a few best DL models and analyze their detection performance, in terms of probability of detection and probability of false alarm. Furthermore, we perform a hands-on experiment to evaluate the performance of some detectors intended to be used in this band.

The remainder of this paper is as follows. In Section II, we describe the radar waveform generation software tool. Section III presents DL-based radar detection techniques. We discuss simulated detection performance results in Section IV and practical implementation tests in Section V. Finally, we summarize the paper in Section VI.

II. Radar Waveform Generation

The motivation behind the RF dataset generator tool is part of an effort by the Communications Technology Laboratory (CTL) at NIST to facilitate and support the use of machine learning in next generation wireless applications. One target application is to create curated RF signal datasets focusing on signals, schemes, systems, and environments for shared spectrum communications systems [26].

For developing and testing of the radar detectors in the 3.5 GHz CBRS band, the need for radar waveforms is inevitable. Field-measured radar waveforms have been collected, e.g., in [21], but they are not widely available due to operational security issues of the radar systems. To overcome this challenge, we design a radar waveform generator tool that enables us to generate different radar waveforms with randomized parameters, mix the waveforms with noise and/or interference, and output the results as RF datasets [25]. We develop the tool in MATLAB[1], and we utilize a graphical user interface (GUI) to simplify the selection of parameters and to automate the waveforms generation process.

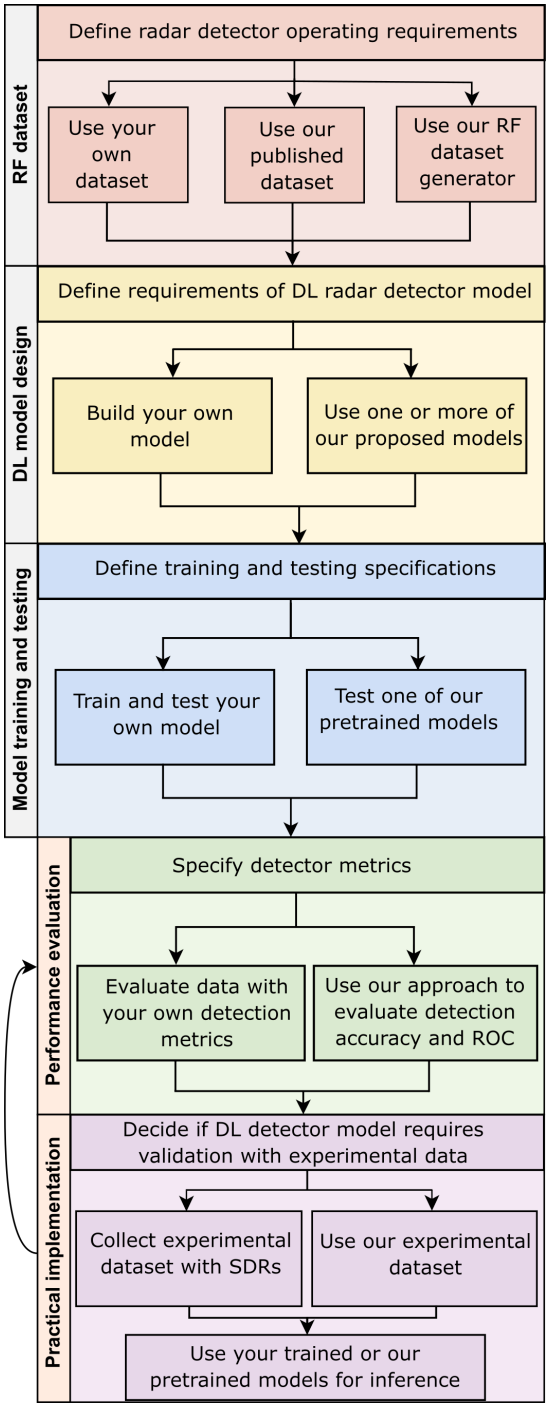

Fig. 1: DL radar detector development workflow.

The tool can be used to generate radar waveforms with parameters similar to what the National Telecommunications and Information Administration (NTIA) proposed in their

[1]Certain commercial equipment, instruments, or materials are identified in this paper to foster understanding. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

testing procedures for ESC sensor certification [27]. To the best of our knowledge, most of these radar waveforms have not been widely simulated or generated by commercial software tools and released in the public domain. As listed in Table 1 in [27], the radar signals are divided into five types depending on the type of pulse modulation and the set of parameter bounds for the signal. Specifically, they are called waveform bins and labeled as: P0N#1, P0N#2, Q3N#1, Q3N#2, and Q3N#3. Each bin includes a range of parameters that encompasses specific existing or anticipated future radar designs in the 3.5 GHz CBRS band. The signals in the first two bins, e.g., P0N#1 and P0N#2, are simple pulsed radar. In addition, bin P0N#2 signal is phase-coded. The radar signals in the other three bins, e.g., Q3N#1, Q3N#2, and Q3N#3, are linear frequency modulated with different sets of parameter bounds.

The waveform parameters include pulse modulation, pulse width, pulse repetition rate, chirp width, and number of pulses per burst. In addition to these parameters, our tool selects and randomizes the following parameters during the mixing process such as signal-to-noise ratio (SNR) range, noise power level, radar signal peak power level, start time of the radar signal, and the baseband center frequency of the radar signal. The peak power of the radar signal is used for SNR computation. The SNR and the WGN power levels are computed in a 1 MHz bandwidth that is centered at the peak of the radar signal in the frequency domain.

Fig. 2 shows the radar waveform generator GUI. A reference RF dataset was generated using this software and can be downloaded from https://doi.org/10.18434/M32116. The reference dataset consists of $40\,000$ waveforms. Half of the dataset includes waveforms with radar signals plus white Gaussian noise (WGN), and the other half includes waveforms with WGN only. At the time of writing this article, the testing procedures for ESC sensor certification in [27] only consider WGN as the interference source. We set the sampling rate of the waveforms to 10 MHz, which is equal to the CBRS channel bandwidth. In addition, we set the duration of the waveforms to 80 ms. We choose this value based on the longest possible duration when using the set of the parameters from Table 1 in [27]. Furthermore, these values for the sampling rate and the duration allow us to randomize the center frequency and the start time of radar signal, respectively. The randomization of the start time and center frequency of the radar enables us to generate more realistic detection scenarios. This is because the sensor does not have knowledge of these parameters in practice, and therefore acquires the waveforms at random time instances and frequencies with respect to the radar signals. The steps for generating the datasets and the full set of the parameters used to generate the reference dataset are documented in the software manual of the NIST simulated radar waveform and RF dataset generator [25].

III. Radar Detection Using Deep Learning

A. Radar Detection in the CBRS Band

Classical detection techniques can be used for detection of radar signals. In the ideal scenario with a known radar signal in WGN, a matched filter is the optimal detector [3]. However, this approach is not feasible for the CBRS band because the radar signal parameters vary over a wide range. Consequently, we need to match a large number of templates if matched filter is used. Therefore, we investigate the use of machine learning approaches for detection of the radar signals. The approach consists of training a deep learning model on a large dataset of waveforms. These waveforms must be representative of the signals in the CBRS band.

B. Deep Learning Detection Based on Binary Classification

The radar detection problem can be formed as a binary classification problem. We investigate three groups of DL classifiers based on the type of input. We modify the input space in the preprocessing step. Specifically, signal processing techniques are applied to the input space to emphasize the features of the signal of interest. In addition, the preprocessing step enables us to change and reduce the dimensions of the input space. Fig. 3 demonstrates the three input types of the classifiers. The waveform in this example is of pulsed radar type (bin P0N#1) and can be accessed from the reference RF dataset through Group 1, subset 7, and waveform number 144. The plot illustrates the same waveform in time domain, spectrogram with max-hold, and full resolution spectrogram along the same time axis.

In the following, we present different DL architectures for each input type. The models are implemented and trained using Keras [28] with Tensorflow backend. The code of the models, and a subset of the pre-trained models are released to researchers and developers and can be accessed at https://git.io/RadarDL. These models can be used for further development and baseline performance comparisons.

1) Classifiers with raw signal input: For this group of classifiers, we use the raw input signal with minimal preprocessing. Specifically, we use the magnitude of the signal as the input to the classifiers. Furthermore, we normalize the input signal in two steps. First, we subtract the mean and divide by the standard deviation of the signal. In the second step, we compute the magnitude and normalize the signal between 0 and 1. For classifiers with raw signal input, we design two architectures: CNN model-1 with one convolutional layer and CNN model-2 with two convolutional layers.
   - CNN model-1: The CNN model-1 is a simple network with one convolutional layer, followed by a max-pooling and two fully connected layers.
   - CNN model-2: The CNN model-2 consists of two convolutional layers, followed by a max-pooling and two fully connected layers.
2) Classifiers with spectrogram and max-hold input: For this group of classifiers, we use the spectrograms of the signals and apply a max-hold procedure over the time dimension of the spectrogram to reduce the size of the input. While this technique reduces the temporal resolution of the input, i.e., loss of information, the reduction in size results in a faster

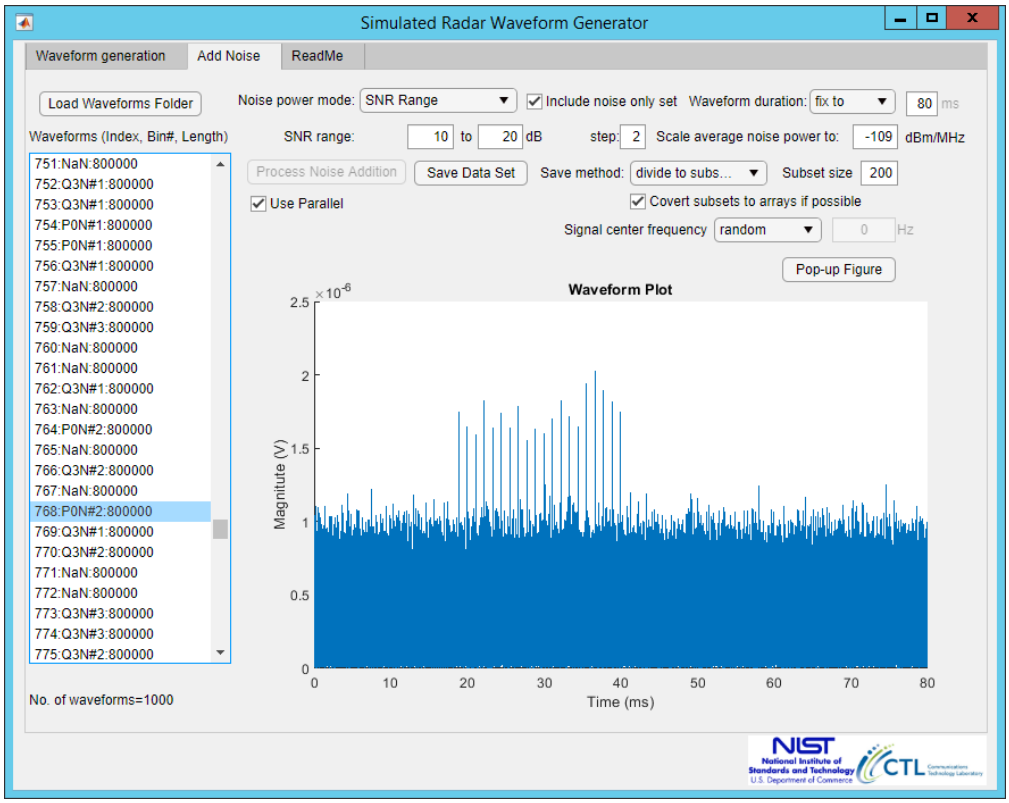

Fig. 2: Radar waveform generator GUI.

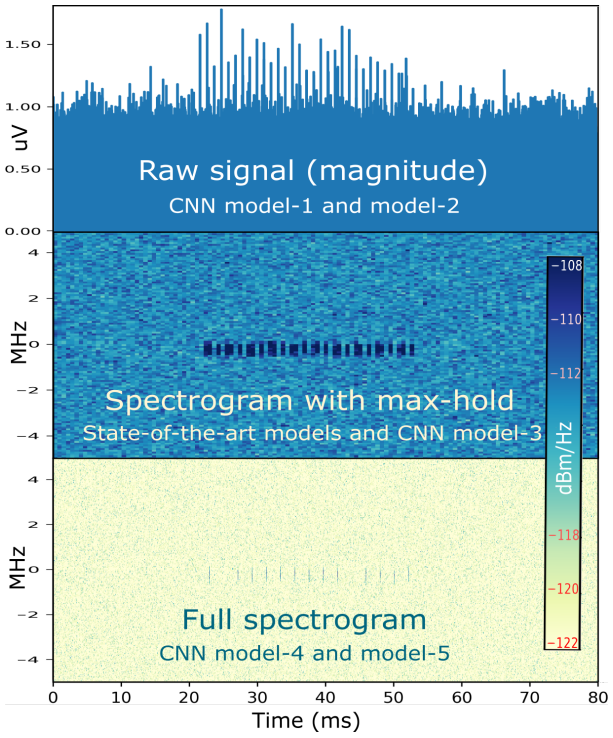

Fig. 3: Model input examples.

training of the model or lower computing resources. We generate the spectrogram using the following parameters: the size of fast Fourier transform (FFT) is 128, the length of the segments is 128, and the number of overlap points between the segments is 24. After generating the spectrogram, we compute the maximum over a window with a length of 60 points. The resulting spectrogram size is 128×128. Consequently, each point in the resulting spectrogram is mapped to 0.624 ms in time and 78.125 kHz in frequency. For training and inference, we normalize the spectrogram between 0 and 1.

- CNN model-3: For the spectrogram with max-hold input case, the proposed CNN architecture consists of three convolutional layers, and each one followed by a max-pooling layer. Then, we added a fully connected layer before the output layer.

In addition to our proposed architecture, we have trained the following state-of-art deep architectures [28]: ResNet50 with 50 layers, Xception with 71 layers, and MobileNetV2 with 53 layers. For these models, we set the size of the output layer to 1 and the activation to sigmoid function.

3) Classifiers with full spectrogram input: Similar to the spectrogram with max-hold, we compute the spectrogram of the input signal with following parameters: FFT length is 256, length of the segments is 256, and the number of overlap points between the segments is 24. For this case we use the full resolution of the spectrogram with the size of 256×3448 as an input to the models. Consequently, each point in the resulting spectrogram is mapped to 0.0232 ms in time and 39.0625 kHz in frequency. Furthermore, we normalize the spectrogram between 0 and 1 for training and inference.

We propose two CNN architectures for the full spectrogram

input case as follows:

- CNN model-4: The first proposed CNN architecture consists of three convolutional layers each followed by a max-pooling then, one fully connected layer before the output.
- CNN model-5: The second proposed CNN architecture input is a max-pooling layer. Then, we added three convolutional layers each followed by a max-pooling layer and a fully connected layer before the output.

IV. Detection Performance Results

In this section we describe the steps and procedures for training the models and analyzing the testing results.

A. Model Training and Accuracy Results

In order to choose the training hyper-parameters, we experimented with different optimizers, learning rates, batch sizes, and number of epochs. We used all the 40 000 waveforms in reference RF dataset to train and test the models. We divided the dataset to subsets of 35 %, 15 %, and 50 % for training, validation, and testing, respectively. The higher number of waveforms for testing is chosen in order to provide enough testing data points for further analysis of the results. Specifically, we used testing results to generate receiver operating characteristic (ROC). We used the training and validation subsets to train all the models. After training, we saved the trained model and the inference results of the testing subset. For the accuracy results, we tested the inference output of the models against a fixed threshold of 0.5 and made the binary decisions accordingly. Table I shows the testing accuracy of each model.

We observe from the accuracy results that models with raw signal magnitude input, i.e., CNN model-1 and CNN model-2, had the lowest performance. Adding additional convolutional layer in CNN model-2 degraded the accuracy slightly in comparison to CNN model-1. Models with spectrogram and max-hold input, i.e., CNN model-3, ResNet50, Xception, and MobileNetV2, performed very well. Among them, CNN model-3 and Xception models performed the best. However, we chose CNN model-3 instead of Xception model from this group because of its simplicity in comparison to the state-of-art image classification models. The results also suggest that radar signal detection with spectrogram does not require sophisticated classification models due to presence of unique signal features in the spectrogram.

Finally, models with the full spectrogram input, i.e., CNN model-4 and CNN model-5, provided comparable results to the models with spectrogram and max-hold input. This finding is expected since the models make use of all the information in the spectrogram. The performance of CNN model-5 was lower than CNN model-4 because the max-pooling layer at the input of CNN model-5 removes some of the spectral features from the input. Consequently, our favored model of all architectures is CNN model-4 since this model has the potential of providing good detection results even with a wider range of scenarios due to no information loss at the input.

TABLE I: Summary of Classification Accuracy.

Model	Accuracy	Number of Parameters
Input: Raw signal (magnitude), Input size: 800 000x1		
CNN model-1	**0.808**	275 706
CNN model-2	0.783	276 316
Input: Spectrogram with max-hold, Input size: 128x128		
CNN model-3	**0.944**	831 009
ResNet50	0.923	23 583 489
Xception	**0.949**	20 862 953
MobileNetV2	0.925	2 258 689
Input: Full spectrogram, Input size: 256x3448		
CNN model-4	**0.948**	429 737
CNN model-5	0.908	273 953

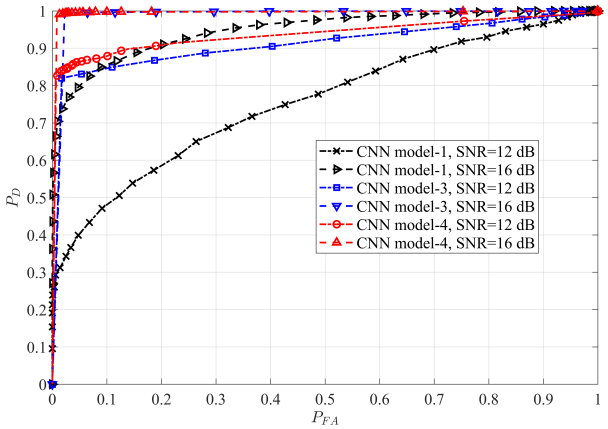

Fig. 4: ROC curves.

B. Detection Performance Analysis

In the following, we further analyze the results of selected best models from each group. Specifically, we generate ROC curves for CNN model-1, model-3, and model-4. To compute ROC curves, we vary a threshold over the inference output of the model and compute the average values for probability of false alarm (P_{FA}) and probability of detection (P_D) at each threshold step. Fig. 4 shows ROC curves for selected best models from each group. Both CNN model-3 and model-4 achieve near perfect detection ($P_{FA} = 0$ and $P_D = 1$) at 16 dB SNR. On the other hand, CNN model-1 shows lower performance at 16 dB SNR. However, further analysis on the performance of CNN model-1 showed the model requires 18 dB or higher SNR to achieve perfect detection. Recall that in the CBRS band, the NTIA requires that ESC sensors must achieve 99 % probability of detection within 60 s of incumbent radar onset at a level of −20 dB WGN power below radar pulse peak power [27]. Hence, CNN model-1 can still potentially be used in practice.

In addition, we observed that the behavior of the detector in all the models depends on the modulation type and the set of parameters for the radar pulse. Fig. 5 shows the ROC curves for each waveform bin over the entire test dataset, i.e., SNR range of 10 dB to 20 dB, for CNN model-4. Among the worst performing radar types is bin P0N#1. Nevertheless, the inadequate detection performance of this bin in Fig. 5 is only due to the lower SNR range. For instance, the detection performance for this bin is near perfect when the SNR is higher than 16 dB, which is 4 dB below the NTIA's requirement of 20 dB SNR.

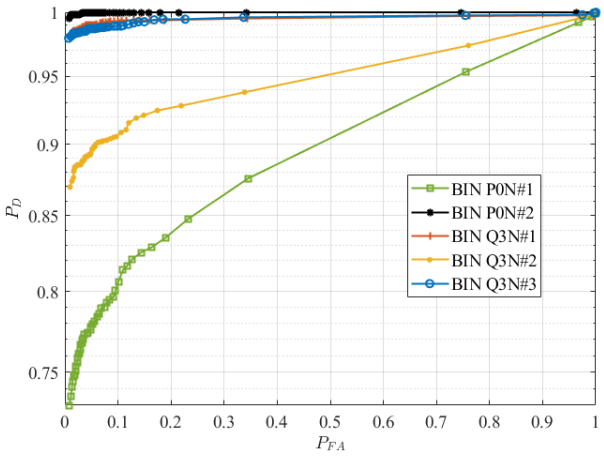

Fig. 5: ROC curves per pulse modulation for CNN model-4.

C. Additional Tests with Low SNR

We use the simulated radar waveform and RF dataset generator [25] to generate an additional testing dataset. The new dataset includes 10 000 waveforms and has similar parameters as the reference RF dataset except for the range of SNR. We set the SNR between 5 dB and 9 dB with a step size of 2 dB. We use our CNN model-4 that was trained on SNR levels over a range of 10 to 20 dB to evaluate the model's detection accuracy of the waveforms with the lower SNR range. We define the following performance measures:

- True negative rate (TNR): The rate of correct detections when the radar signal is absent.
- False positive rate (FPR): The rate of false detections when the radar signal is absent.
- False negative rate (FNR): The rate of false detections when the radar signal is present.
- True positive rate (TPR): The rate of correct detections when the radar signal is present.

Table II compares the accuracy results of the testing dataset (SNR range in $[10,\ 20]$ dB) against the lower SNR dataset (SNR range in $[5,\ 9]$ dB). As the SNR range decreases, the overall accuracy only degrades from 0.948 to 0.721, whereas the TPR degrades significantly from 0.922 to 0.471. Further analysis would be required to analyze the performance degradation for each SNR step in the new dataset. In addition, this test raises a question of whether retraining the model on the lower SNR dataset could improve the performance? However, addressing these questions is beyond the scope of this paper and will be addressed in future work.

TABLE II: Confusion Matrices for CNN model-4.

Target		0	1
	0	TNR=0.975	FPR=0.025
	1	FNR=0.078	TPR=0.922
		0	1

Predicted
Accuracy=0.948

(a) SNR range in $[10,\ 20]$ dB.

Target		0	1
	0	TNR=0.971	FPR=0.029
	1	FNR=0.529	TPR=0.471
		0	1

Predicted
Accuracy=0.721

(b) SNR range in $[5,\ 9]$ dB.

V. Experimental Results

This section describes how to evaluate radar detector performance using an experimental dataset that is collected using Software Defined Radios (SDRs). Our radar detector models are trained on simulated data samples created by our waveform generation software. It is convenient to train detector models on simulated radar samples because it requires significantly less effort than capturing and labeling RF data from actual radar systems and avoids the operational security constraints of working with military radars. Once trained, the detector's performance is evaluated with waveform samples that exhibit RF distortion and impairments caused by RF hardware imperfections. These impairments are imparted by passing a set of test waveforms through the analog RF front ends and digital baseband circuitry of sending and receiving SDRs. Our assumption is that the model's detection accuracy will be degraded by being trained on simulated data and tested on experimental data that exhibits RF impairments. But, the question remains, in what way and to what extent is the accuracy degraded? The following is a description of our experimental approach for evaluating a detector's performance using SDRs.

A. Method for Collecting Experimental Test Data

The radar test waveforms are created using the NIST radar waveform generator software described in Section II. The software is configured to generate 900 unique waveform samples over a SNR range of 10 to 20 dB with a fixed average noise power of 0 dBm/MHz. Each 80 ms waveform sample contains one of five possible radar waveform modulations summed with appropriately scaled WGN that results in a specified SNR level. The rational for creating and sending a composite waveform of signal plus noise is that it increases the confidence of obtaining a specified SNR level at the receiver. This is reinforced by sending composite waveforms where the average noise power is at least 40 dB greater than the thermal noise floor of the receiver, thereby ensuring that any change in SNR due to receiver's thermal noise floor is imperceptible. This approach is preferred over attempting to send a precisely scaled radar waveform that attains a specified SNR based on knowledge of the thermal noise floor of the receiver and the RF pathloss between the sending and receiving SDRs. Furthermore, each radar waveform sample is randomly offset in frequency to represent the expected misalignment between real-world radar channels and RF spectrum sensors. The experimental test dataset is augmented by adding 900 noise-only waveform samples for evaluation of the detector's FPR and TNR metrics.

As shown in Fig. 6, experimental waveforms are collected using two Universal Software Radio Peripherals (USRP) N210 SDRs [29] connected via a 2 meter RF coaxial cable with a 30 dB RF attenuator attached to the transmit RF port of the sending SDR. Each USRP is connected to its own host computer using a 100BASE-TX Ethernet interface. Custom waveform transmit and receive software is written using MATLAB's USRP Communication Toolbox support package. This SDR software package simplifies the design and operation of transmitting and receiving large batches of waveform samples. The MATLAB test script commands the Sending SDR Node to sequentially transmit 80 ms radar sample at 10 MSps

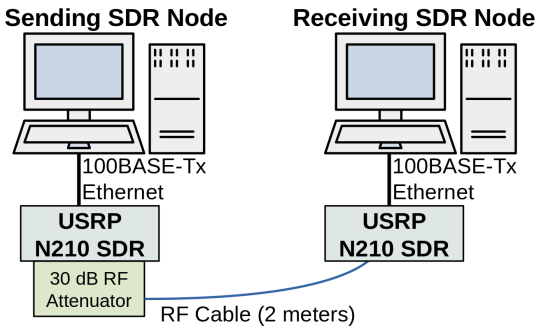

Fig. 6: Test equipment setup and connection diagram.

while another script commands the Receiving SDR Node to asynchronously collect the waveform samples at 10 MSps and store the data. A 80 ms RF muting period is inserted as a receiver sync signal prior to sending 200 consecutive 80 ms waveform samples. After collecting the entire 16 s of waveform samples, the receiver's data collection script detects the rising edge of the first received radar waveform sample and marks that sample index as the start of the test data. The subsequent 16 s of data samples are then reshaped into a 800 000 row by 200 column matrix of complex double-precision floating point numbers and saved in a MATLAB workspace file that includes the ground truth labels of the waveform samples. This workspace file is then loaded by post-processing scripts for evaluation of the detector's accuracy metrics. The original (simulated) and experimentally collected radar waveform datasets are available for download at the following URL: https://doi.org/10.5281/zenodo.4521678. This dataset archive includes the MATLAB scripts that were used to control USRP SDRs to send and collect the experimental waveform data.

B. Description of SDR RF Impairments

The USRP SDRs possess an analog RF front end that distorts the RF waveform on transmission and reception. Some distortion is caused by analog hardware imperfections, such as In-Phase/Quadrature (IQ) mixer imbalance, Local Oscillator (LO) phase noise, non-linear amplification of signals in the transmit and receive RF chains, and filter rolloff at the band edges due to the USRP's digital sample decimation filters. Another notable impairment is the frequency response "droop" observed at the upper and lower edges of the band being captured by the USRP. This spectral coloration is due to the use of computationally efficient digital Cascaded Integrator-Comb (CIC) decimation filters that possess a gradual transition band rolloff [30], [31]. Fortunately, the USRP N210 automatically cascades a halfband decimation filter when the sampling rate is set to 10 MSps. This secondary decimation filter has a steeper transition band than the CIC filter and reduces the extent of the droop at the band edges when cascaded with the CIC filter. Finally, we mitigated LO leakage within the capture band by setting the USRP's LO frequency offset to 10 MHz.

C. Analysis of Experimental Accuracy

We use the previously described CNN model-4 that was trained with simulated waveform samples to generate detection results for the experimental test dataset. The accuracy metrics shown in Table III were generated with a detection threshold of 0.5 over 1800 collected waveform samples where the SNR range is 10 to 20 dB. Despite training the model with simulated waveform samples, the detector's accuracy on experimental data degraded only from 0.944 to 0.874, or a percent change of 7.52 %. This limited reduction indicates that the CNN model-4 detector is fairly robust to waveform distortion caused by passing through SDRs with hardware impairments. Future work will attempt to mitigate this performance impact by adding experimentally collected samples to the training dataset. Finally, for a SNR of 20 dB, the detector's TPR of 100 % on the experimental test data exceeds the NTIA's ESC sensor certification criteria of attaining at least 99 % TPR within 60 s [27].

TABLE III: Table of Detection Performance Metrics.

	Simulated Data Detection Results		**Experimental Data Detection Results**	
SNR	**FNR**	**TPR**	**FNR**	**TPR**
10	0.253	0.747	0.247	0.753
12	0.131	0.869	0.15	0.85
14	0.027	0.973	0.048	0.952
16	0.007	0.993	0.034	0.966
18	0.007	0.993	0.014	0.986
20	0	1.0	0	1.0
Average	0.071	0.929	0.082	0.918
Average TNR	0.959		0.830	
Average FPR	0.041		0.170	
Accuracy	**0.944**		**0.874**	

We note that the RF impairments present in our laboratory experiments do not represent the harsh RF environment experienced by RF sensors when operating in real-world scenarios. In the case of the CBRS band, the type of RF interference is likely to be RF waveforms emitted by LTE or 5G New Radio (5G NR) cellular network radios. Other possible sources of interference are out-of-band emissions from other military radars that appear in the CBRS band [32], [33]. The impact of these real-world operating conditions can be characterized using an extension to our modular radar detector development workflow described in Fig. 1. The impact of RF multipath fading on the detector can be characterised by sending radar waveform samples through a RF mobile channel emulator and capturing an experimental test dataset using the techniques described in Section V-A. Similarly, detector degradation due to other-user interference can be studied by superpositioning a variety of cellular radio network signals onto the waveform samples in the training dataset.

VI. Conclusion

In this paper, we have presented a comprehensive framework for RF signal generation, detection, and evaluation. Specifically, we provided a functional and effective software tool that can generate different radar waveforms, add interference to these waveforms, and produce associated RF datasets. We also designed, trained, and tested several deep learning based detection models—tailored to specific input types and features—using the simulated RF datasets. We then identified the best models, one in each input group, and took a deep dive

analysis into their detection performance for different radar pulse modulations and at different SNR levels. In addition, we evaluated radar detector performance on the experimental test data that exhibit realistic RF signal impairments.

This work was largely motivated by a need to create curated RF signal datasets and baseline detection performance results for machine learning in next generation wireless applications. Although we mostly targeted the radar detection problem for spectrum sharing in the 3.5 GHz band, researchers and developers are encouraged to expand their own version of this work to other scenarios and spectrum bands by utilizing the tools and the approaches developed in this paper. For instance, four ESC operators have been tested and certified against the waveform bins described in [27] at the time of this writing. If requirements for other types of radar waveforms in the 3.5 GHz band become available, our waveform generation tool can be modified to incorporate the new waveforms.

ACKNOWLEDGMENT

The authors would like to acknowledge Dr. Kapil Dandekar of the Drexel Wireless Systems Laboratory for his advice and consultation on collecting the experimental results.

REFERENCES

[1] "Auction 105: 3.5 GHz," Federal Communications Commission, 2020. [Online]. Available: https://www.fcc.gov/auction/105

[2] "Citizens broadband radio service," 47 C.F.R. § 96, 2016. [Online]. Available: https://www.ecfr.gov/cgi-bin/text-idx?node=pt47.5.96

[3] R. Caromi, M. Souryal, and W. Yang, "Detection of incumbent radar in the 3.5 GHz CBRS band," in *Proc. IEEE GlobalSIP*, Nov. 2018.

[4] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 3, pp. 2224–2287, Third Quarter 2019.

[5] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3039–3071, Fourth Quarter 2019.

[6] M. Bkassiny, Y. Li, and S. K. Jayaweera, "A survey on machine-learning techniques in cognitive radios," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1136–1159, 2013.

[7] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, Dec 2017.

[8] A. Schmidt, C. Rügheimer, F. Particke, T. Mahr, H. Appel, and H. Kölle, "Kurtosis based approach for detection of targets in noise," in *2016 17th International Radar Symposium (IRS)*, May 2016, pp. 1–3.

[9] T.-T. Ng, S.-F. Chang, and Q. Sun, "Blind detection of photomontage using higher order statistics," in *2004 IEEE International Symposium on Circuits and Systems*, vol. 5, May 2004.

[10] C. M. Spooner, "Classification of co-channel communication signals using cyclic cumulants," in *Conference Record of The Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, vol. 1, Oct. 1995, pp. 531–536.

[11] M. Petrova, P. Mähönen, and A. Osuna, "Multi-class classification of analog and digital signals in cognitive radios using support vector machines," in *2010 7th International Symposium on Wireless Communication Systems*, Sep. 2010, pp. 986–990.

[12] M. M. Ramón, T. Atwood, S. Barbin, and C. G. Christodoulou, "Signal classification with an SVM-FFT approach for feature extraction in cognitive radio," in *2009 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, Nov 2009, pp. 286–289.

[13] T. J. O'Shea, 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.

[14] 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.

[15] Narengerile and J. Thompson, "Deep learning for signal detection in non-orthogonal multiple access wireless systems," in *2019 UK/ China Emerging Technologies (UCET)*, Aug 2019, pp. 1–4.

[16] D. Ke, Z. Huang, X. Wang, and X. Li, "Blind detection techniques for non-cooperative communication signals based on deep learning," *IEEE Access*, vol. 7, pp. 89 218–89 225, 2019.

[17] Z. Li, R. Liu, X. Lin, and H. Shi, "Detection of frequency-hopping signals based on deep neural networks," in *2018 IEEE 3rd International Conference on Communication and Information Systems (ICCIS)*, Dec 2018, pp. 49–52.

[18] H. Ye, G. Y. Li, and B. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, Feb 2018.

[19] C. Ha and H. Song, "Signal detection scheme based on adaptive ensemble deep learning model," *IEEE Access*, vol. 6, pp. 21 342–21 349, 2018.

[20] L. Fang and L. Wu, "Deep learning detection method for signal demodulation in short range multipath channel," in *2017 IEEE 2nd International Conference on Opto-Electronic Information Processing (ICOIP)*, July 2017, pp. 16–20.

[21] W. M. Lees, A. Wunderlich, P. J. Jeavons, P. D. Hale, and M. R. Souryal, "Deep learning classification of 3.5 GHz band spectrograms with applications to spectrum sensing," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 2, pp. 224–236, June 2019.

[22] R. Caromi and M. Souryal, "Detection of incumbent radar in the 3.5 GHz CBRS band using support vector machines," in *in Sensor Signal Processing for Defence (SSPD) Conference*, May 2019.

[23] S. Sarkar, M. Buddhikot, A. Baset, and S. K. Kasera, "Deepradar: A deep-learning-based environmental sensing capability sensor design for CBRS," in *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, New York, NY, USA, 2021.

[24] "IEEE DySPAN Standards Committee (DySPAN-SC) website," https://sagroups.ieee.org/dyspan/ieee-1900-8/, accessed: 2021-08-11.

[25] R. Caromi and M. Souryal, "Simulated radar waveform and RF dataset generator for incumbent signals in the 3.5 GHz CBRS band," National Institute of Standards and Technology, 2020. [Online]. Available: https://doi.org/10.18434/M32229

[26] T. Hall, R. Caromi, M. Souryal, and A. Wunderlich, "Reference datasets for training and evaluating RF signal detection and classification models," in *Proc. IEEE GLOBECOM Workshop on Advancements in Spectrum Sharing*, Dec. 2019.

[27] F. H. Sanders, J. E. Carroll, G. A. Sanders, R. L. Sole, J. S. Devereux, and E. F. Drocella, "Procedures for laboratory testing of environmental sensing capability sensor devices," National Telecommunications and Information Administration, Technical Memorandum TM 18-527, Nov. 2017. [Online]. Available: https://www.its.bldrdoc.gov/publications/details.aspx?pub=3184

[28] F. Chollet, "Keras," 2015. [Online]. Available: https://github.com/fchollet/keras

[29] "Ettus Research, USRP N210 product page," https://www.ettus.com/all-products/un210-kit/m, accessed: 2021-01-20.

[30] J. Malsbury and M. Ettus, "Simplifying FPGA design with a novel network-on-chip architecture," in *Proceedings of the Second Workshop on Software Radio Implementation Forum*, ser. SRIF '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 45–52. [Online]. Available: https://doi.org/10.1145/2491246.2491251

[31] "Why does my received spectrum "droop"?" accessed: 2021-08-11. [Online]. Available: https://witestlab.poly.edu/blog/why-does-my-received-spectrum-droop-at-the-edges/

[32] P. D. Hale, J. A. Jargon, P. J. Jeavons, M. R. Souryal, A. J. Wunderlich, and M. Lofquist, "3.5 GHz radar waveform capture at Point Loma: Final test report," National Institute of Standards and Technology, Technical Note 1954, May 2017. [Online]. Available: https://www.nist.gov/publications/35-ghz-radar-waveform-capture-point-loma

[33] ——, "3.5 GHz radar waveform capture at Fort Story: Final test report," National Institute of Standards and Technology, Technical Note 1967, Aug. 2017. [Online]. Available: https://www.nist.gov/publications/35-ghz-radar-waveform-capture-fort-story-final-test-report