Detection of Co-existing RF Signals in CBRS using ML: Dataset and API-based Collection Testbed

Chinenye Tassie [*], Abdo Gaber[†], Vini Chaudhary [*], Nasim Soltani [*], Mauro Belgiovine[*], Michael Loehning[†], Vincent Kotzsch[†], Charles Schroeder[†] and Kaushik R. Chowdhury[*]

[*]Dept. of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA
[†]National Instruments Corporation, Austin, Texas, USA
[*]{tassie.c, vi.chaudhary, soltani.n, belgiovine.m}@northeastern.edu, krc@ece.neu.edu
[†]{abdo.gaber, michael.loehning, vincent.kotzsch, charles.schroeder}@ni.com

***Abstract*—Opening up of spectrum for shared use, such as the Citizen Radio Broadband Service (CBRS) band, offers unprecedented opportunities for allowing commercial operators to operate in frequencies otherwise reserved for federal use only. Specifically in the CBRS band, the challenge of detecting the highest priority incumbent radar reliably forces severe restrictions on the transmit power for operators deploying LTE networks. While Machine Learning (ML)-based solutions have demonstrated the potential for detecting weak radar signals in fully overlapping secondary signals, there exists a fundamental gap in porting these methods for practical, real-world conditions due to a key reason: There are no accessible datasets or even controlled methods to generate such datasets today over-the-air (OTA), where radar and LTE 'overlap' in a number of challenging SINR conditions. This paper makes three contributions: (i) It describes the first publicly available CBRS overlapping and non-overlapping LTE and radar OTA dataset in the 3.5 GHz band using an experimental testbed composed of software defined radios, (ii) It describes the first-of-its-kind open source Application Programming Interface (API) that can configure automatically multiple transmitters and receiver radios, synchronize them, remove the Tx local oscillators-induced artifacts, and carefully set their parameters such as sampling rates, center frequencies, and time duration for sample collection, ultimately resulting in high-fidelity data in the Signal Metadata Format (SigMF); (iii) It demonstrates the utility of the CBRS dataset by adapting the well-known ML model called 'You Only Look Once' (YOLO) for detecting and localizing the radar and LTE signals with near-perfect accuracy, pointing to the possibility that current FCC-mandated power thresholds can be lowered for cellular operators in the CBRS band.**

***Index Terms*—CBRS band, Data collection API, Incumbent radar, LTE, RF Signals dataset for ML**

I. Introduction

To alleviate the looming spectrum crunch, the Federal Communication Commission (FCC) is actively exploring the shared spectrum usage model, where frequencies exclusively reserved for federal use are now being opened for commercial cellular operators and unlicensed devices. One such example is the use of the Citizen Broadband Radio Service (CBRS) band (3.55-3.7 GHz), which allows incumbent radar users, Long-Term Evolution (LTE) operators and other unlicensed devices to share spectrum in a tiered access structure [1]. While desirable, the success of spectrum sharing depends on the key challenge of preventing interference to the incumbents from co-channel secondary users' signals. In the CBRS band, the FCC mandates deploying Environment Sensing Capability (ESC) sensors to detect radar signals in a myriad of complex spectrum overlap scenarios, including the worst case of the radar pulse being completely enveloped within the secondary LTE signal. In such diverse interference scenarios, classical signal processing techniques such as matched-filtering [2] often fail to provide good detection performance. Further, these require a prior knowledge of radar signal templates, which is not possible as per rules of the CBRS band.

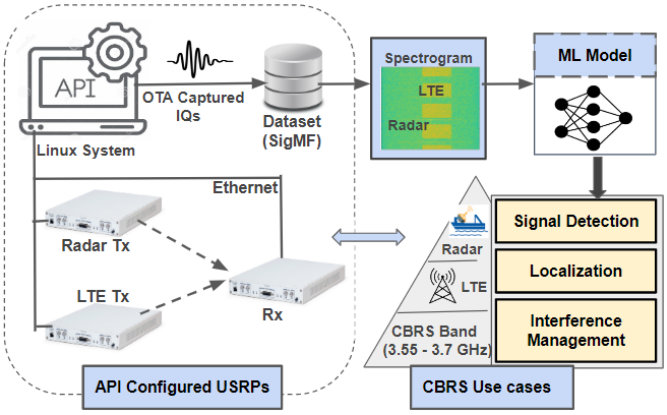

Fig. 1: Proposed API-based dataset collection testbed.

Machine Learning (ML) for solving wireless communication applications: Over the years, there has been an increasing interest in using ML for diverse wireless communication applications, such as radio fingerprinting [3], mmWave beam selection for vehicular scenarios [4], etc. More relevant to this paper, few recent works [5]–[7] use ML for radar waveform recognition and CBRS spectrum resource allocation in low interference scenarios showing low detection accuracy in the range of $[76.92 - 94.4]\,\%$. Additionally, these works do not estimate the bandwidth of the radar/interfering signals. To overcome these limitations, we use a deep learning-based approach YOLO, which detects and localizes radar and LTE signals with better performance accuracy even in high interference scenarios without requiring any prior knowledge.

While these ML approaches show great promise, the models are trained and tested on datasets (described later in Table I) that are not collected over-the-air (OTA) in the actual 3.5 GHz

Applications	Paper	Architecture	Dataset: Scenario	Access
Radar detection	Troglia *et al.* [7]	Federated Learning	Emulated radar, Modulated Signals	No
Co-channel interference	Caromi *et al.* [2]	Matched-Filter	Real radar captures, Synthetic LTE	Yes
Co-channel interference	Caromi *et al.* [6]	Deep Learning	SDR-based radar + AWGN (wired connection)	Yes
Localization, Unauthorized signal detection	Soltani *et al.* [9]	Deep Learning	Simulated radar, LTE, 5G	No
Localization	Sarkar *et al.* [10]	Deep Learning	Emulated radar, OTA LTE (synthetically added)	No

TABLE I: State-of-the-art techniques that use datasets capturing co-existing RF signals in the shared CBRS spectrum.

CBRS band. Thus, there remains a gap in their validation in representative environments, which cannot be fully bridged via simulation alone. To the best of our knowledge, there is no real radio frequency (RF) dataset that contains overlapping radar and LTE signals captured OTA in the CBRS band for rigorous testing in high and low interference scenarios.

Challenges in collecting OTA datasets for CBRS band: According to the FCC requirements, the ESCs must accurately detect 99% of radar transmissions for the scenarios with the radar peak power above -89 dBm/MHz and the aggregate interference (LTE+noise) power below -109 dBm/MHz, i.e., signal-to-interference-noise ratio (SINR) ≥ 20 dB [8]. Creating a representative dataset that satisfies these requirements is challenging for these reasons: It requires multiple software-defined radios (SDRs)-based transmitters (Txs) to transmit different OTA signals and precisely configure their parameters such as sampling rates, time duration, transmit antenna gains, and center frequencies. Most importantly, the presence of local oscillator (LO) offsets in the signals transmitted via SDRs needs to be corrected for every Tx, which otherwise could result in false alarm(s) during the detection of the signals. Additionally, the in-phase and quadrature (IQ) samples of the OTA collected overlapping signals should have accurate timestamps and properly documented metadata in standard-compliant formats for re-usability of the dataset by the research community.

APIs for collecting complex datasets: We develop and publicly release a Python-based Application Programming Interface (API) which simplifies and automates dataset recording campaigns by interfacing NI's Universal Software Radio Peripheral (USRP) devices via USRP Hardware Drivers (UHDs). It requires a single text-based configuration file in JavaScript Object Notation (JSON) or Yet Another Markup Language (YAML) format, to configure complex scenarios with multiple Txs and Rxs over the large ranges of parameters. It provides continuous playback of waveforms for each Tx while supporting different waveform formats such as TDMS (NI's Technical Data Management Streaming) and MAT (MATLAB). It stores datasets in the open source Signal Metadata Format (SigMF) [11] with comprehensive, human readable, and easy-to-parse JSON-based metadata files.

In Fig. 1, we show our proposed API-based CBRS dataset collection testbed, which configures the Tx SDRs for transmitting LTE and radar signals OTA, and the Rx SDR for collecting and saving IQ samples in SigMF format. Thereafter, spectrograms are created and fed to ML model (e.g., YOLO) for detection and localization of the interfering LTE and radar signals in the CBRS band.

Our contributions are as follows:

- We provide a 55.5GB real-world RF dataset which contains overlapping and non-overlapping radar and LTE signals with a controlled SINR range of $15-35$ dB. The signals were transmitted and recorded OTA at 3.5 GHz in an RF anechoic chamber using Ettus X310 SDRs.
- To collect this dataset, we develop a generic API-based solution that can be easily adapted for other, general purpose dataset generation needs. This API interfaces between Python and UHD drivers to provide unified control over SDRs configurations. We describe the API configuration process for the collection of an RF dataset containing the radar and LTE RF signals in the shared CBRS band ($3.55-3.7$ GHz).
- We demonstrate the efficacy of this real-world dataset in detecting and localizing RF signals in time-frequency space in the CBRS band using the framework designed in our previous work [9] for detecting signals in 3.5 GHz band using the 'You Only Look Once' (YOLO) object detection model.

II. Use Cases for CBRS Signal Dataset and API

In this section, we discuss how the dataset created for the CBRS band could benefit a broader set of activities related to ML-based solutions for wireless sensing. We also review existing approaches for these activities and prior datasets relevant to CBRS spectrum sensing in Table I.

A. Co-channel Interference Management

Identifying the presence of incumbent signal transmission (here, radar) overlapping in frequency with a lower priority signal (here, LTE) and timely informing such events to the spectrum access management system (SAS) has broad application in different licensed and unlicensed bands. Exciting research wherein ML algorithms identify available vacant spectrum, regulate the power of the interfering signals, and identify regions where LTE networks may operate, can usher in new paradigms for fully automated SAS. In [6] and [2], simulated/field-measured signals of shipborne radar with synthetically added interference (noise or LTE) are used for radar detection using signal processing and deep learning (DL) techniques. We believe that using real RF datasets that capture co-existing radar and LTE signals can comprehensively validate approaches like these, which attempt to solve interference detection problems using powerful ML techniques. It is difficult to obtain LTE/radar co-existence datasets from an existing ESC sensor deployment along the coast due to reluctance from vendors for sharing their data and also for security and privacy reasons (for example, it is illegal to track

the source radar geographical location). In all such cases, our datasets and the SDR-based data collection API can help accelerate research activities given the extensive ability to tune the system for different conditions.

B. Time-Frequency Localization

We define 'localization' as estimating temporal and spectral occupancy of the RF signals existing in shared spectrum bands. Such information, especially for overlapping signals, aids the SAS in taking decisions related to completely stopping the secondary transmission or shifting the center frequency of the interfering secondary signals. Recently, we used the well known DL YOLO architecture [12] to detect, classify, and localize overlapping weak radar and strong 5G/LTE signals in the CBRS band under high noise conditions using a fully simulated dataset in MATLAB [9]. In a seminal prior work, the authors in [10] localize radar signals in the CBRS band by generating a dataset that uses a signal generator for emulating radar pulses and synthetically adding OTA-captured LTE signals. However, this dataset is not publicly available. These examples suggest that OTA data collection is both challenging and necessary. Our API-based testbed lowers the barrier towards collection of diverse RF datasets by allowing configuration of all the scenario-specific variations in frequencies, gains, signals, and data collection cycles in just one configuration file. The API runs the SDRs without further human intervention to collect a comprehensive OTA dataset.

C. Unauthorized Signal Detection

Many spectrum sharing models, especially the one used in the CBRS band, have limited ability to detect opportunistic transmissions of unauthorized signals, e.g., LTE and 5G users that are not registered with the SAS to operate in the CBRS band. Such signals could either coincide with the incumbent signals or exist separately within the shared band. For identifying them, it is crucial to detect, classify, and localize all the ongoing transmissions and share this information with the SAS, which can compare them with the list of registered signals. In our recent work [9], we attempt the labeling of all transmissions (radar, authorized/unauthorized LTE/5G) in the CBRS band using a simulated dataset. We believe that the ML-based algorithms for the detection of unauthorized signals will greatly benefit from our real-world dataset containing LTE and radar signals co-existing at same/different frequencies.

III. RF Data Recording API for Dataset Generation

To test and validate ML models, different researchers use different simulation tools, channel models, and/or ray-tracing applications to generate datasets for their specific scenarios. Thus, available datasets are typically stored in heterogeneous formats and with different kind and completeness of metadata (scenario descriptions). This makes it quite difficult to generalize and compare the ML models and to adopt the datasets by the broader research community. Another important aspect is the augmentation of the AI/ML model training and validation

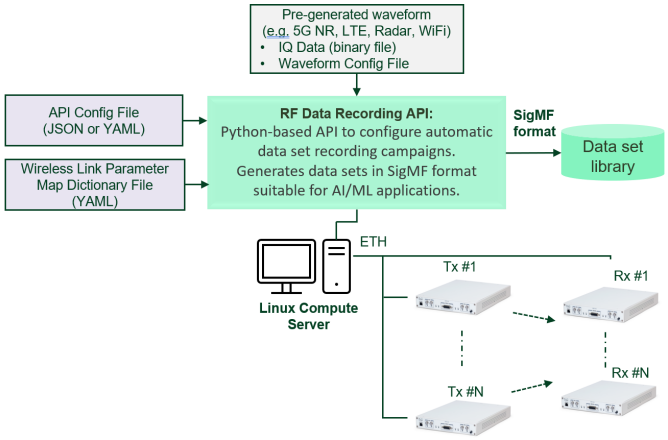

Fig. 2: RF data recording API reference architecture.

with real-world RF datasets. This can ultimately improve the robustness of the trained AI/ML algorithms in practical environments since these datasets include additional effects like RF impairments and real-world channel properties which are not covered by simulation environments. In addition, the underlying ML algorithms require large datasets to get trained for a wide variety of potential scenarios and the performance depends on the quality and representability of the datasets.

To the best of our knowledge, there is no existing dataset recording tool that provides a holistic consideration of the previous problems and challenges. The proposed RF Data Recording API is a free and open-source Python-based API to record real-world RF datasets in an easy and automated way utilizing NI's USRP platform. It has been created by NI in collaboration with Northeastern University and published under an open-source license on Github [13]. The generation of real-world datasets could be critical in determining how the algorithms perform in the real world. The recorded datasets are saved in the SigMF metadata format. SigMF is an open-source standard that specifies a way to describe sets of recorded digital signal samples with data properties and comprehensive scenario descriptions provided in JSON-based SigMF metadata files. This allows usage of the recorded datasets for various research areas such as ML for wireless communication. In addition, this will simplify the management of dataset libraries and encourage adoption by other researchers.

Fig. 2 shows the block diagram of the RF data recording API which interfaces NI's USRP SDRs that are connected to a Linux server. The Python-based data recording API allows to easily configure complex data recording campaigns by using just a single configuration file for setting all desired parameter values and variations for multiple connected SDRs. Data recording campaigns can be configured via a YAML or via a JSON configuration file. Functionally there is no difference between these two configuration formats. The YAML format is more verbose with inline comments closely attached to the respective parameter settings and therefore good for beginners. The JSON configuration file allows for very compact descriptions of complex configurations. As additional input the data recording API requires pre-generated IQ waveforms in a supported binary format together with corresponding text-based waveform configuration files. The latter contain the transmit

waveform configuration parameters, which will be added to the SigMF metadata of the generated recordings. Since different waveform generators usually apply different naming conventions for the waveform parameters, a Wireless Link Parameter Map Dictionary file in YAML format is used to define the mapping between waveform configuration parameters and the target SigMF metadata output fields. This approach allows to flexibly adapt to different waveform generators' naming schemes or to add new or self-defined waveform parameters by just modifying the Wireless Link Parameter Map Dictionary file. The number of Tx and Rx stations can be scaled with individual parameter configurations. Each Tx station can do a continuous waveform playback with an individual waveform. Several waveforms (e.g. 5G NR, LTE, Radar, WiFi) have been given as a template. To mitigate the Tx LO leakage/DC offset, the LO configuration can be used for that. It can be enabled or disabled via the API configuration file. For each IQ data record, data formatting and saving in SigMF format is done automatically. Each record has two files: the SigMF binary file where the actual data is stored, and the SigMF metadata file in a JSON format. The metadata file provides a description about the test scenario, the RF configuration and hardware information of both Tx and Rx stations, and the individual information of each Tx waveform. The API can be executed in several RF modes (Tx and Rx, Tx-only, and Rx-only).

IV. Dataset Collection using API & its Application for Localizing the CBRS Signals

A. Data Collection Setup and Dataset Description

Our dataset collection setup consists of three Ettus X310 USRPs, one each for the radar transmitter (incumbent user) and LTE (secondary user) signals in the CBRS band ($3.55 - 3.7$ GHz), and a third USRP for receiving IQ samples of these OTA signals. All USRPs have free-running oscillators; they are not synchronized. Due to the transmission restrictions in the CBRS band, we conduct the experiment in an indoor RF anechoic chamber as shown in Fig. 3.

We consider 3GPP standard-compliant time division duplex downlink LTE signals of bandwidths $5, 10, 15, 20$ MHz corresponding to different LTE test models (or E-UTRA models)- $1\_1, 1\_2, 2, 3\_1, 3\_2,$ and 3_3 which vary in number of resource blocks, types of modulation, etc.. We generate these signals using NI RFmx Waveform Creator [14] in TDMS format. We synthetically generate type 1 radar waveform using National Institute of Standards and Technology's simulated radar waveform generator [6] with parameters: bandwidth 2 MHz, sampling rate 20 MHz, pulse repetition rate 1010 pulses per second, and pulses per burst 10 pulses.

We configure center frequency, sampling rate and time duration of OTA signals' frame captured by the Rx radio in the API configuration file as 3600 MHz, 30.72 MHz, and 40 ms, respectively. Further, we consider two different scenarios of overlapping and non-overlapping radar and LTE signals in different frames by, respectively, setting same or different center frequencies of the Tx radios in the range $[3600 - 15.36, 3600 + 15.36]$ MHz. The former one captures interference scenarios between the radar and LTE signals,

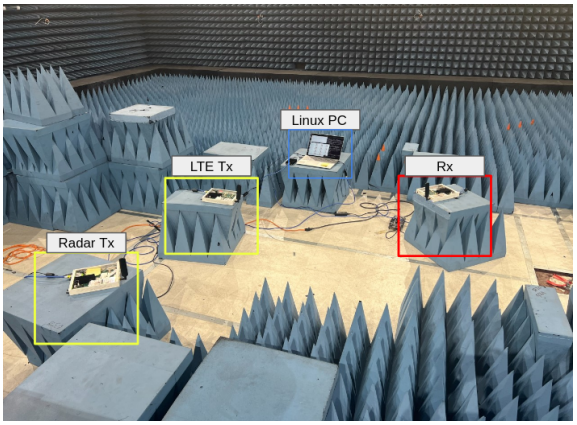

Fig. 3: Experimental setup of API-based dataset collection in RF anechoic chamber.

which is missing in all the publicly available CBRS datasets. Our dataset contains 3,360 frames capturing IQ samples of overlapping signals and 1,920 frames of non-overlapping scenarios with their SigMF metadata files. Most importantly, we change the distances and gains of the Tx and Rx radios to capture radar signals in high and low interference generated from the LTE+noise present in the background in the CBRS band. To quantify this, we explicitly calculate the value of SINR (= ratio of the peak radar power per MHz to the average noise and interference power per MHz in the CBRS band) for each frame and add it in the respective frame's metadata file. The range of SINR in the dataset is $[15, 35]$ dB. Since the FCC requires current ESC sensors to provide 99% radar detection accuracy for SINR values $\geq$ 20 dB, we consider the corresponding frames to represent low interference cases, while frames with the SINR values $<$ 20 dB to represent high interference cases. *This varied SINR feature makes our real OTA signals' dataset one-of-its-kind due to its potential to serve the ML applications that target to achieve radar detection in high interference from secondary users' LTE signals and background noise such as the work [9].* It is important to note that due to the limitation on distances required for achieving the FCC-mandated signals' powers and the noise of the receiver, the received power of our signals is higher, but the SINR requirements of the FCC are met. Lastly, while collecting our dataset, we correct the LO leakage that creates an undesired peak at the center frequency of the radio by enabling the LO offset removal in the API configuration file. The presence of this peak in the frames used to train the ML models could result in inaccurate detection and poor performance.

The resulting dataset contains 5,640 binary files and accompanying meta files occupying 55.5 GB. Our dataset can be accessed here: https://genesys-lab.org/CBRS or https://dx.doi.org/10.21227/zdg0-7242

B. YOLO for Signal Detection and Localization

To demonstrate the competence of our real-world OTA CBRS signals' dataset in training ML model to detect and localize LTE and radar signals, we use YOLO-based framework,

which we earlier used with a simulated CBRS radar and LTE signals dataset [9]. The YOLOv3 [12] is an object detection framework that uses spectrogram as input and provides the detected signals along with their labels, detection probabilities, and bounding boxes to specify time and frequency boundaries. It has several advantages over other ML-based models- 1) it looks at the entire image at test time, hence its predictions are informed by the global context in the image, and 2) it is faster than other neural network-based model (1000x faster than R-CNN and 100x faster than Fast R-CNN) because it makes predictions with one network evaluation. These features make YOLO viable for signal detection, classification, and time-frequency localization in the shared CBRS band.

Our work brings a new dimension to the data that is fed to the YOLO framework. By considering partially and completely overlapping signals (CBRS radar and LTE), we stress-test YOLO's signal identification/localization performance for shared spectrum applications such as interference detection and anomaly detection. To use our dataset in training the YOLO framework, we first generate spectrograms from the collected IQ samples of each frame of duration 40 ms and sampling rate 30.72 MHz. We shuffle our spectrograms and partition them $70\%, 20\%$, and 10% to create training, validation, and testing sets, respectively. Next, we create one label file for the signals in each frame required for running the YOLO framework. The label file contains the class numbers (0 for radar and 1 for LTE) and the relative placements of signals (their x center, y center, width, and height with respect to spectrogram's origin) present in the spectrogram of the considered frame. Thereafter, we train and test the YOLO model and observe that the radar and LTE signals are detected with high probabilities of detection, correct labels, and near accurate time-frequency boundaries in both the overlapping and non-overlapping scenarios, as shown in Fig. 4. The code used to demonstrate the results of YOLO on our dataset is downloaded and used "as is" from this publicly accessible and open source repository: https://github.com/ultralytics/yolov3.

To get more insights on the statistical performance of YOLO-based signal detection framework trained using our real-world CBRS signals' dataset, we compute object detection metrics-recall and precision. Recall is defined as ratio of number of true positives to the total number of signals in all the spectrograms in the test set, while the precision is computed as true positive count divided by total number of positives for that signal label [9]. The values of these metrics for both radar and LTE signals are provided in Table II. We observe that the recall for radar detection drops to 86.6% in case of our real-world dataset-based training, while it is 98% in case of the simulated dataset-based training for course signal detection in our work [9]. This drop in performance is attributed to the following reasons:

- The impairments present in wireless channel and radios are captured in our real dataset.
- The real dataset contains frames corresponding to high interference scenarios (SINR $15-35$ dB), which adds complexity in the detection of narrow radar signals.
- The LTE signals considered in the real dataset have varying pattern of resource block allocations. For instance, 5 MHz LTE signals corresponding to test models $1\_1, 1\_2, 3\_1, 3\_2,$ and 3_3 look like a contiguous solid block spanning entire frame duration, while that corresponding to the test model 2 is made of small solid blocks with the space between these blocks representing unused resource blocks. The former LTE signals make detection of overlapping narrow radar signals even more challenging as seen from the lower values of recall for the radar signals (rows in blue color) compared to that in the latter case (row in violet color) in the Table II.

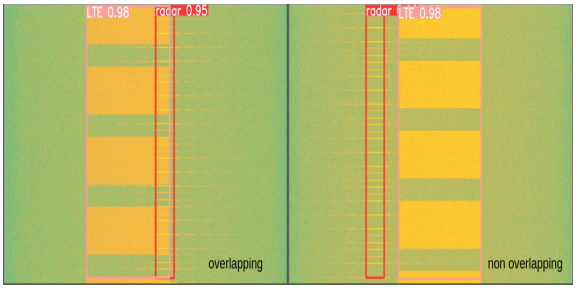

Fig. 4: Sample output spectrograms of YOLOv3 with bounding boxes around the detected radar and LTE signals in overlapping (left) and non-overlapping (right) scenarios during testing.

Category	Accuracy on test set Per-Test Model (TM)			
	LTE		RADAR	
	Precision	Recall	Precision	Recall
TM1_1	1	**0.99**	0.97	**0.87**
TM1_2	1	**1**	0.96	**0.85**
TM2	1	**1**	0.97	**0.90**
TM3_1	1	**0.99**	0.96	**0.87**
TM3_2	1	**0.99**	0.97	**0.86**
TM3_3	1	**0.99**	0.97	**0.86**
All TM	1	**0.99**	0.97	**0.86**

TABLE II: Performance metrics of YOLO-based detection of radar and LTE signals. The results are shown for various LTE test models and highlights that Radar recall is best when overlapping with LTE TM2.

This highlights the fact that the simulated datasets alone cannot train well the ML models in representative environments. Though the radar detection accuracy mandated by the FCC is 99%, note that we are not trying to further improve our YOLO-based framework [9] to achieve this accuracy with our current real-world dataset because the focus of this paper is to provide a challenging and representative OTA CBRS signals' dataset. In this context, robust ML-based approaches may help in detecting co-existence of radar and LTE signals with high accuracy. We discuss a few such directions in Section V-A.

V. Future Research Directions

A. Adapting the Dataset for Robust ML-based Sensing

ML-Model generalization to unseen environment settings: ML model trained on data collected in one environment often fails to perform well on similar data collected in new unseen environment. This unseen environment could result from the

use of different set of parameters (e.g., sampling rates, time duration, etc.) or different wireless channels (e.g., indoor/outdoor environment) for new dataset collection. Such ML model generalization problems are open for signal detection in the shared CBRS band as well. In this regard, our current dataset could serve as data collected in one environment and our API could be readily used for generating similar datasets in different environments. These can then be leveraged in ML techniques such as transfer/meta learning for robust signal detection in the shared CBRS spectrum use cases.

Multi-modal fusion for accurate signal detection: Fusing multi-modal data inputs (such as spectrograms and IQs) in ML techniques for robust radar detection (i.e., improved detection accuracy) is an open avenue for research. The IQ samples of our collected dataset can be readily used to create spectrograms for investigating ML fusion frameworks in this direction. Motivated by the work [15] that used IQ samples and cyclostationary features extracted from them to detect direct spread spectrum signals within LTE frames with better accuracy, we believe the IQ samples of our collected dataset can be post-processed to extract desirable features and used for exploring different multi-modal fusion frameworks [15].

B. Enhancements to Data Collection API

From an API architecture design perspective, we plan to study an event trigger-based dataset collection capability that could allow for precise synchronization and more complex traffic patterns involving multiple devices operating concurrently in the same physical environment. We will also explore how data collection framework, packaged as an rApp or xApp, can be integrated within OpenRAN (ORAN) to transmit and collect data through the radio interfaces and train ML models at the non-real time Radio Intelligence Controller (RIC).

VI. Conclusion

In this paper, we develop a Python-based API that simplifies and automates collection of any real-world RF datasets using NI's USRP platform. We adopt this API for generating one-of-its-kind RF dataset containing OTA radar and LTE signals, collected using SDRs in an RF anechoic chamber at frequencies corresponding to the shared CBRS spectrum band. This dataset consists of 5640 frames of IQ samples for duration 40 ms each, containing both overlapping and non-overlapping scenarios of the radar and LTE signals for a wide SINR range of $15-35$ dB, which can serve a variety of ML-based shared CBRS spectrum applications. We outline a few such use cases where this dataset and API can be used to accelerate research in this direction, while demonstrating their efficacy in the use case of localizing CBRS signals in time-frequency space using a well known ML model for object detection- YOLO.

Acknowledgement

The authors from Northeastern University are supported by the NSF CCRI RFDataFactory project CNS #2120447.

References

[1] "FCC Releases Rules for Innovative Spectrum Sharing in 3.5 GHz Band," https://docs.fcc.gov/public/attachments/FCC-15-47A1.pdf, accessed: May 2022.

[2] R. Caromi, M. Souryal, and W.-B. Yang, "Detection of incumbent radar in the 3.5 GHz CBRS band," in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2018, pp. 241–245.

[3] A. Al-Shawabka, F. Restuccia, S. D'Oro, T. Jian, B. Costa Rendon, N. Soltani, J. Dy, S. Ioannidis, K. Chowdhury, and T. Melodia, "Exposing the Fingerprint: Dissecting the Impact of the Wireless Channel on Radio Fingerprinting," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020, pp. 646–655.

[4] B. Salehi, G. Reus-Muns, D. Roy, Z. Wang, T. Jian, J. Dy, S. Ioannidis, and K. Chowdhury, "Deep Learning on Multimodal Sensor Data at the Wireless Edge for Vehicular Network," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 7, pp. 7639–7655, 2022.

[5] M. Tonnemacher, C. Tarver, V. Chandrasekhar, H. Chen, P. Huang, B. L. Ng, J. Charlie Zhang, J. R. Cavallaro, and J. Camp, "Opportunistic Channel Access Using Reinforcement Learning in Tiered CBRS Networks," in *2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2018, pp. 1–10.

[6] R. Caromi, A. Lackpour, K. Kallas, T. Nguyen, and M. Souryal, "Deep Learning for Radar Signal Detection in the 3.5 GHz CBRS Band," in *2021 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2021, pp. 1–8.

[7] M. Troglia, J. Melcher, Y. Zheng, D. Anthony, A. Yang, and T. Yang, "FaIR: Federated Incumbent Detection in CBRS Band," in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–6.

[8] 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*, pp. 18–527, 2017.

[9] N. Soltani, V. Chaudhary, D. Roy, and K. Chowdhury, "Finding Waldo in the CBRS Band: Signal Detection and Localization in the 3.5 GHz Spectrum," in *IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2022, pp. 4570–4575.

[10] 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*, 2021, pp. 56–68.

[11] "Signal Metadata Format," accessed: Oct. 2022. [Online]. Available: https://github.com/gnuradio/SigMF

[12] J. Redmon and A. Farhadi, "Yolov3: An Incremental Improvement," *arXiv*, vol. 1804.02767, pp. 1–6, 2018.

[13] NI and N. University, "RF Data Recording API repo on GitHub," https://github.com/genesys-neu/ni-rf-data-recording-api, accessed: Sep. 2022.

[14] "What is RFmx?" accessed: Oct. 2022. [Online]. Available: https://www.ni.com/en-ca/shop/wireless-design-test/application-software-for-wireless-design-test-category/what-is-rfmx.html

[15] D. Roy, V. Chaudhary, C. Tassie, C. Spooner, and K. Chowdhury, "ICARUS: Learning on IQ and Cycle Frequencies for Detecting Anomalous RF Underlay Signals," in *IEEE International Conference on Computer Communications (INFOCOM)*. IEEE, 2023, pp. 1–10.

Chinenye Tassie is pursuing a Ph.D. at Northeastern University. Her research interests are ML applications for wireless systems and receiver beamforming. **Abdo Gaber** received his PhD degree in Electrical Engineering from the University of Magdeburg, Germany, in 2015. Currently, he is a Principal Software Engineer at NI. His current research interests include machine learning / mmWave / and cell-free massive MIMO for 6G Wireless Comms. **Vini Chaudhary** received her Ph.D. degree (2021) in Electrical Engineering from Indian Institute of Technology Delhi, India. She is currently a post-doctoral research associate at Northeastern University. Her current research

interests include RF datasets design for ML in wireless communication, quantum communication networks, and signal processing.

Nasim Soltani is a PhD student at Northeastern University. Her research interests are machine learning applications for the physical layer of wireless systems.

Mauro Belgiovine is pursuing his Ph.D. at Northeastern University. His current research interests involve deep learning, wireless communication, and heterogeneous computing.

Michael Loehning is a Chief Engineer at NI in Dresden, Germany, where he focuses on long-term innovation research in the area of wireless communications and RF test. He holds a Ph.D. degree in Electrical Engineering, received from Dresden University of Technology in 2006.

Vincent Kotzsch received his PhD degree in Electrical Engineering from the Technical University of Dresden in 2012. Currently, he is a Chief Engineer at NI where he leads a 6G long-term innovation initiative with focus on AI/ML for RF wireless.

Charles Schroeder is a Fellow at NI in Austin, TX, where he is a member of a small team that directs NI' long-term research and innovation. He is a student of the innovation process and focuses his time in the area of wireless networks.

Kaushik Roy Chowdhury (M'09-SM'15) is a Professor at Northeastern University, Boston, MA. His current research interests involve systems aspects of networked robotics, machine learning for agile spectrum sensing/access, wireless energy transfer, and large-scale experimental deployment of emerging wireless technologies.