



HAL
open science

Understanding Radio Frequency Fingerprint Identification With RiFyFi Virtual Databases

Alice Chillet, Robin Gerzaguet, Karol Desnos, Matthieu Gautier, Elena Simona Lohan, Erwan Nogues, Mikko Valkama

► **To cite this version:**

Alice Chillet, Robin Gerzaguet, Karol Desnos, Matthieu Gautier, Elena Simona Lohan, et al.. Understanding Radio Frequency Fingerprint Identification With RiFyFi Virtual Databases. IEEE Open Journal of the Communications Society, 2024, pp.1-1. 10.1109/OJCOMS.2024.3414858 . hal-04617950

HAL Id: hal-04617950

<https://inria.hal.science/hal-04617950>

Submitted on 19 Jun 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Understanding Radio Frequency Fingerprint Identification with RiFyFi Virtual Databases

Alice CHILLET[†], Robin GERZAGUET[†], Karol DESNOS^{*}, Matthieu GAUTIER[†],
Elena Simona LOHAN[‡], Erwan NOGUES[§], Mikko VALKAMA[‡],

[†] Univ Rennes, CNRS, IRISA, firstname.name@irisa.fr

^{*} Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, France, firstname.name@insa-rennes.fr

[‡] Faculty of Information Technology and Communication Sciences, Tampere University, Tampere, Finland

[§] DGA

Abstract—This paper proposes to explore the Radio Frequency Fingerprint (RFF) identification with a virtual database generator. RFF is a unique signature created in the emitter transmission chain by the hardware impairments. These impairments may be used as a secure identifier, as they cannot be easily replicated for spoofing purposes. In recent years, the RFF identification has mainly relied on Deep Learning (DL), and large databases are consequently needed to improve identification in different environmental conditions. In this paper, we introduce RiFyFi, a virtual database generator and explore individually the impairment impact on the classification accuracy to highlight the most relevant impairment. Different transmission scenarios are then explored, such as the impact of the data type (being a preamble or a payload) and the data size. Design rules of real database are finally drawn for the different scenarios. We found out that the power amplifier imperfections play the biggest role in RFF accuracy and that average accuracy levels of 94% can be reached when combining the various hardware impairments at the transmitter.

Index Terms—Radio Frequency Fingerprint, Deep Learning, Database, RF impairments models.

I. INTRODUCTION

In recent years, the massive introduction of low-power Internet of Things (IoT) devices has increased the sensitivity of wireless networks to attacks. Hence, a secure authentication has been the topic of many researches [4], particularly the Radio Frequency Fingerprint (RFF) identification has been largely studied [11], [18], [18], [24], [26].

The purpose of RFF is to uniquely identify a device by recognizing defects in the signals it emits. This identification is based on the singularity of the hardware, the transmitter component impairments create unique electromagnetic distortions in the transmitted signal [5], and those distortions are used to differentiate devices. This identification solution addresses some security issues of IoT and can be used for non-spoofing authentication [12]. In most telecommunication standards, identification methods are based on the meta-data communication protocol that gives an address to enable the authentication, for example the Media Access Control (MAC) address, but those solutions can be counterfeited [39]. The RFF identification can be combined with a MAC address identification to improve and ensure identification without spoofing. [A] For example, the RFF must correspond to the MAC address to validate authentication.

To identify a device thanks to its RFF, most related works use the signal in time domain and classification methods to discriminate among the different potential emitter candidates. The State of the Art (SoA) proposes two families of classification methods, the first one is called model-based or parametric method, and the second is the Deep Learning (DL) based solution. The parametric methods use some features that describe the RFF [5], [22], [31]. This method requires RFF knowledge to choose and compute some feature estimators used for classification. The quality of the estimators can hinder the accuracy of parametric RFF methods.

Recently, with the explosion of the use of DL, many research works have been focused on this second family of classification solutions. The supervised DL techniques use labeled signals from different transmitters during the training phase and learn how to recognize the source of the different signals. Many DL architectures exist, in particular Convolutional Neural Network (CNN) is used to extract and classify RFFs [3], [35], [36], [40].

While DL techniques present promising results, there is a strong need for a large and robust database [18]. In the literature, databases are composed of raw signals from different devices captured with one or many Software Defined Radio (SDR). Each signal in the database is labeled to its emitter. During the learning phase, the network has to learn the RFF in the signals to estimate the labels and improve the classification. However, ensuring RFF learning requires an unbiased database [25]. A bias is created when a systematic error is introduced during the database creation. For example a systematic error can be a different power at emission for each transmitter, the propagation channel [3], [15], a MAC address in data, or the quality of emitters in the sense of the amount of distortion induced by the device hardware. These biases could impact positively or negatively the identification, by helping or distracting the network to classify.

Most existing RFF identification works use experimental data to explore RFF identification [3], [15], [36]. However, these works are limited by the lack of flexibility of their database. As a consequence, they cannot explore RFF identification in many dimensions, they focus their work on one aspect and it is difficult to combine the different contributions

because the databases are not comparable. To address those issues we propose a new database generator based on a virtual model of both impairments and transmission scenarios. The **Radio Frequency Fingerprint Virtual Database Generator (RiFyFi_VDG)** allows exploring RFF identification and studying the impact of the quality of the device in a countermeasure context. The databases created by the generator are composed of numerous WiFi-like signals that come from SoA RFF models.

The core contributions of this paper are as follows:

- We present an open-source RFF Identification Framework called RiFyFi_IdF coupled with an open-source Virtual Database Generator based on complex based band models of RFF which four impairments models and different scenarios of transmission called RiFyFi_VDG. **[To Complete/To do]**
- We model the transmitter quality, and study each impairment independently in order to define the limit dispersion to classify devices.
- We study all impairments together in different transmission contexts in function of transmitters quality and then highlight the most significant impairments for RFF identification.
- We study the impact on classification performance of the size of the database in different transmission contexts.

The rest of this paper is organized as follows. In section II, transmission and RFF models are presented, with a focus on their intricacies and pitfalls in an RFF identification context, and some related works are introduced. Section III introduces the framework and the transmission models in particular RFF models used to create the virtual database. Then the section IV describes precisely the database creation and scenario used in this paper. Then, the section V presents the individual effect of the impairments. Finally, section VI presents the interest of a virtual database is discussed. The conclusion of the paper is in section VII.

II. DATABASE CHALLENGES FOR RFF IDENTIFICATION

A. RF transmission and RFF models

1) *RF transmission*: Before the transmission, the source information data is converted into symbol sequences. Most wireless communication protocols generate symbol frames composed of three parts as detailed in Figure 1. The symbol sequences are composed of three different parts: the preamble (Pre) which contains synchronization information that is always the same for all the packets, the MAC addresses (MAC) of the emitter and the payload data (Pay). The symbol sequences are converted into IQ samples and transmitted via the RF chain (Tx, Figure 2). A Digital Analog Converter (DAC) transforms the signal into the analog domain to yield $x(t)$. The Local Oscillator (LO) modulates it at carrier frequency f_{RF} , and the Power Amplifier (PA) amplifies the signal, creating $x_{ant}(t)$ for transmission via the antenna.

2) *RFF definition and issues*: The RFF of a transmitter is a unique signature created by the hardware components of the transmission chain. This is neither controlled nor done on



Fig. 1. Transmission frame in wireless communications.

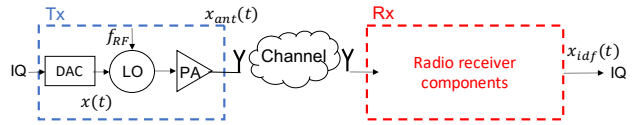


Fig. 2. Transmission and reception chain.

purpose, but it is the result of small flaws in the manufacturing process. This signature, called the RFF of the transmitter Tx, is denoted $\mathcal{F}_{\text{RFF}_{\text{Tx}}}$ and could be used in wireless communications to identify the device which has emitted the signals. The emitted signal could be modeled by:

$$x_{\text{ant}}(t) = \mathcal{F}_{\text{RFF}_{\text{Tx}}}(x(t)), \quad (1)$$

$$x_{\text{ant}}(t) = \mathcal{F}_{\text{PA}} \circ \mathcal{F}_{\text{LO}}(x(t)), \quad (2)$$

where \circ represents the function composition, and \mathcal{F} represents a component behaviour, including its distortion. The nested functions of (2) show the impact of each component and the difficulty of extracting features and modeling the transmission with RFFs. The LO adds distortions related to frequency, phase, and gain and the PA impacts the gain.

In Figure 2, the channel block represents the wireless communication environment defined by the noise, interference signals, and the multi-path and fading channels that could impact the signal. The propagation channel is modeled by $\mathcal{F}_{\text{channel}}$. The red block Rx represents the receiver with its components (not detailed in the model) and its distortion function called the RFF of the receiver, but investigating its impact is beyond the scope of this paper. The received signal x_{idf} can thus be expressed as:

$$x_{\text{idf}}(t) = \mathcal{F}_{\text{RFF}_{\text{Rx}}} \circ \mathcal{F}_{\text{channel}} \circ \mathcal{F}_{\text{RFF}_{\text{Tx}}}(x(t)). \quad (3)$$

In an experimental study, $\mathcal{F}_{\text{channel}}$ changes in function of the position and the relative position of transmitters and receiver(s) and of the noise level; these also influence the received power. All of these disturbances make difficult the Tx RFF identification. This issue is largely studied in the SoA and will be called channel or environmental condition impact in the rest of the paper. The environmental conditions represent an important bias in database construction for RFF identification DL solutions.

In conclusion, the database challenges are multiple because of the large design space with many communication protocols and the type and the number of transmitters. The next subsection presents the related works where the community explores the large design space of databases for RFF, such as the communication standard, the type of devices (emitters and receiver(s)), the part of the signal used to identify, the environmental condition of transmission and the data pre-processing such as transformation, slicing, and multi-packets.

TABLE I
SUMMARY TABLE OF DATABASES FO RFF IDENTIFICATION

Reference	Year	Database	Is data public ?	Protocol	Numbers of emitters	Additional informations or Contribution
Experimental Databases						
Hall et al. [13]	2003	Own	No	Bluetooth	10 transmitters	Exploit the phase to detect transient signal
Riyaz et al. [33]	2018	Own	No	WiFi	5 B210	CNN outperforms alternate Machine Learning
Sankhe et al. [35], [36]	2019	ORACLE	Yes	WiFi	16 X310	Add control impairments with feed-back driven to increase differentiability for bit-similar device
		DARPA	No	WiFi	140 DUTs (phones)	
Morin et al. [25]	2019	Own	Yes	WiFi	21 N2932	Generate data to minimise the role of the unwanted channel
Peng et al. [27]	2019	Own	No	ZigBee	54 DUTs	Adopt 4 novels modulation-based features effectives in ZigBee node classifying
Jian et al. [20]	2020	DARPA	No	WiFi ADS-B	5000 DUTs 5000 DUTs	Investigate 2 CNNs for RFF identification under different environmental scenarios
Soltani et al. [40]	2020	DARPA	No	WiFi	50, 250, 500, 5000 DUTs	Study the interest of data augmentation
Al-Shawacka et al. [3]	2020	Own	Yes	WiFi	13 N210 and 7 X310	Evaluate the impact of the wireless channel on CNN-based fingerprinting algorithms
		DARPA	No	WiFi, ADS-B	100 to 10000 DUTs	
Shen et al. [38]	2021	Own	No	LoRa	10 DUTs	Use a transformer as classifier, improve low SNR classification accuracy with data augmentation
Elmaghbbub et al. [7]	2021	Own	Yes	LoRa	25 Pycom devices	
Reus-Muns et al. [32]	2020	Own	No	WiFi, 4G, 5G	4 USRP X310	Incorporate the triplet loss with the deep CNN
Hanna et al. [16]	2022	WiSig	Yes	WiFi	174 USRPs	Create 4 datasets for RFF identification varying, days, number of emitters, receivers and signals
Jagannath et al. [17]	2022	Own	Yes	Bluetooth	10 DUTs	2 days to do generalization
Chillet et al. [6]	2023	WiSig	Yes	WiFi	6 USRPs	Use Tangled Program Graph as a classifier Compare training duration to SoA CNN
Elmaghbbub et al. [1]	2023	Own	Yes	WiFi	50 Pycom devices	2 datasets outdoor and indoor
Elmaghbbub et al. [8]	2023	Own	Yes	Wifi	50 Pycom devices	4 datasets: wire, wireless, different locations
Simulation based Databases						
Soltani et al. [40]	2020	Own	Yes	WiFi	10 transmitters	Model only IQ imbalance impairment
Zhang et al. [43]	2021	Own	No	LoRa	50/200 transmitters	Uniformly and randomly distributed IQ imbalances and PA nonlinearities
Chillet et al.	2024	RiFyFi_VDG	Yes	WiFi	N_{Tx}	Virtual database generator Model IQ imbalance, PA, PN, CFO

B. Related Works

Recently, the number of contributions on RFF identification has increased [18], mainly due to the number of applications such as authentication for security in IoT systems. The different applications lead to different identification scenarios. For example, Guo et al. [12] use the term "1 to 1 authentication" to verify if the RFF of a device matches its MAC address. A second context defines a list of authorized devices and unauthorized devices, only based on the RFF or by combining RFF and address. Hanna et al. [14] formulate the problem of recognizing authorized transmitters and rejecting new transmitters.

In the literature, two families of RFF identification methods are investigated: parametric and DL techniques. The paramet-

ric method uses feature extractors combined with a classic machine-learning solution. This method exploits the intrinsic and unique nature of the impairments to identify the device. For example, PARADIS [5] uses IQ samples and defines different metrics: the IQ origin offset the frequency error and SYNC correlation, to characterize the "radiometric" signature of a device. The use of such parametric methods is strongly limited by the knowledge about the transmission chain, protocol, modulation, and the superposition of impairments.

Recently, the number of RFF identifications by classification methods has exploded with the advent of DL [3], [12]. In particular, supervised DL is massively used in RFF classification, as it automatically learns how to classify radio transmitters by recognizing complex patterns from labeled signals. DL tech-

niques include CNN [19], [23], [35], [36], Long Short-Term Memory (LSTM) [34], Recurrent Neural Network (RNN) [40] and transformers architectures [37], [38]. CNNs are the most popular solution and different architectures have been explored such as AlexNet-inspired with 12M parameters, distributed on 8 convolutional layers and 3 fully connected layers, or CNNs with less layers, but more parameters [35], [36]. Numerous CNN are used for RFF without real comparison between them [18], [19], [23] except in [40].

DL solutions achieve good results, but the classification accuracy of such methods dramatically depends on the database used to train the network. Therefore, there is a strong need for large and robust databases composed of raw labeled signals [18] from different transmitters to ensure RFF recognition in many environmental conditions. Since 2019, the SoA on RFF identification with DL has increased and presents different databases to experiment RFF identification. A selection of recent papers on RFF identification is presented in Table I. These databases are separated into two types: experimental-based and simulation-based as shown in Table I. They are created for different wireless protocols which depend on the author's research affinities, presented in column 5 in Table I such as WiFi [16], [35], [36] and LoRa [38], [41]. Column 4 "Is data public?" gives information on public access and column 6 "Numbers of emitters" and 7 gives respectively information about the number of emitters and additional information or contribution of the paper. The nature of the signals and the frame structure impact the results analysis deduced by the authors and it is therefore difficult to compare different works. For example, [35], [38] uses only the preamble data while in ORACLE [36] the frame consists of a MAC address field with always the same address and a random payload. Jian et al. propose to slice the signals containing the MAC address to be resilient to MAC address spoofing [19].

Databases can be separated into two types: experimental-based and model-based. In the literature, experimental signals are mostly created with IoT Device Under Test (DUTs) or Software Defined Radio (SDR) platforms for both transmitters and receivers. The work of Zhang et al. [44] shows that the transmitter and receiver type is important in RFF identification because the capacity to discriminate two devices is linked to the RFF difference between two devices. For example, a Universal Software Radio Peripheral (USRP) X310, a high-hand quality device, is produced with low variability components, resulting in minimal RF front-end variations between two devices. Sankhe et al. [35] show that X310 is more difficult to separate, compared to B210. Moreover, [31] studies the receiver impact on the classification capacity, a receiver could be sensitive to one emitter. To avoid some classification problems as channel or receiver impacts, some authors propose to pre-process data before using the neural network, [35] proposes an under-complete demodulation that aims at removing only the channel effect from raw IQ samples, without compensating the device imperfections. The pre-processing used in RFF identification are synchronization, domain transform, or equalization [22], and some authors add a CFO compensation

because the CFO is sensitive to temperature variation and reduces the system stability [38], [43]. The next subsections present different families of databases used in the SoA of RFF identification. First, the real databases are introduced private and public databases, and finally, we focus on virtual databases.

1) *Real Private Database*: The largest existing database for RFF identification was created by DARPA in 2020. This database, used by authors from Northeastern University in Boston, is a private one and is used in many papers [3], [19], [20], [35], [40]. This database is composed of two datasets, one with WiFi signals, with 5117 DUTs, and an average of 166 transmissions for each device. The second dataset is composed of ADS-B signals from 5000 DUTs and an average of 76 transmissions for each device [40]. This database offers the possibility to train the network with a large number of devices. Unfortunately, this database is only available to researchers with US government sponsors.

Peng et al. [27], [28] designed a large RFF database for ZigBee standard. They use 54 DUTs as transmitters and one USRP as a receiver to create the database. They performed ten measurements for each ZigBee device at different locations with line-of-sight transmissions. Their database is only used for their work.

Exploring RFF identification needs a database adapted to the scenario with metadata. Thus for many papers in the literature, the authors create their own database, with few devices [13], [32], [33], [38] but they never give an open access to their database. Consequently, the reproducibility of experimental results is not possible.

2) *Real Public Database*: The University of Boston creates its own databases for experiments in papers [33], [35], [36]. First, they created a database with 5 USRPs B210 transmitters with different distances varying between 2 and 50 ft [33]. Then in 2019, they created the ORACLE database with 20 USRPs X310 transmitters [36]. They suggested introducing software-controlled impairments at the transmitter side to enhance identification robustness. This recommendation arises from the fact that the X310 transmitters are produced with low variability components, resulting in minimal RF front-end variations between two devices.

Hanna et al. proposed a new public database for RFF [16] called WiSig. WiSig is constructed with many signals and with information on how signals have been captured as transmitter location and the type of transmitter used (Atheros). They provide a large-scale WiFi dataset captured by 41 USRPs with 20 MHz bandwidth from different references. The signals come from 174 WiFi transmitters over four different days of captures performed over a month. The authors have created different databases with many transmitters (150), many receivers (32), and many signals (1000 for each transmitter). They present WiSig as a RFF database to explore the identification in a static environment with different types of transmitters and different numbers of transmitter/receiver/signal/days.

In the same way, Al-Shawabka et al. present in [3] a public database for RFF. This database is composed of 4 datasets,

each of them is composed of 20 transmitters, 12 B210 and 8 X310, and one fixed receiver. They first explore the best pre-processing and then they explore the impact of antenna and channel with both wired and wireless communications in an anechoic chamber.

Morin et al. [25] work on unbiased database creation, they leverage FIT/CorteXlab anechoic chamber to capture signals and control the propagation environment as well as the interference profile, which enables the full control of the generated datasets. To increase channel variations and to reduce the possibility for the receiver to learn from the channel properties, the MultiRx setup is proposed where they merge the signals observed from several devices acting alternatively as identification receivers. However, this combination of signals creates confusion between the channel effect and receiver effect, which cannot be studied separately.

Jagannath et al. present in 2022 [17] a new public dataset which includes emissions from 10 COTS IoT emitters (2 laptops and 8 commercial chips) that are captured with a USRP X300 device. The dataset is split into two: Day1 and Day2 each of which is recorded in a different time frame, location, and testbed setup to enable critical generalization test of the trained DL model.

Elmaghub proposes different WiFi datasets [1], [8] composed of 50 Pycom devices. They create outdoor and indoor scenarios, and wired and wireless scenarios on different days, and static or dynamic propagation channels. They captured the first two minutes of transmissions using the USRP B210 at a sample rate of 45 MSps. The captured signals were then digitally down-converted to the baseband and stored as I/Q samples on a computer. To avoid any data dependency on the identity of the WiFi transmitter, all transmitters were configured to broadcast the same packets, which include the same spoofed MAC address and a payload of zero-bytes.

3) *Public and Virtual Database*: Soltani et al. [40] propose to simulate 10 virtual transmitters to create a custom dataset and study the impact of multiplying the number of channels seen by the network during the training phase. However, they decided to model only IQ mismatch because of the complexity of modeling many RF impairments.

Zhang et al. present [43], a model-based database with 4 impairments models. They work on a comprehensive study of RF impairments modeling to address the need for the design of a robust RFF identification protocol. Their model includes LO imperfections, IQ gain and phase imbalance, and PA nonlinearities. They study the impact of individual and overall impairments in different configurations and define a robust RFF identification protocol when RF impairments cannot be reconfigurable or customized to help the identification. Their work focuses on the estimation and calibration of the CFO and the calibration of the IQ imbalance of the receiver.

C. Database Challenges

As a conclusion, the design space of RFF database is largely explored by the community. However, it is difficult to design a good training database related to an application context. Real databases cannot provide the flexibility, reproducibility,

and exhaustivity we need to understand and ensure that the network is currently learning the RFF, and creating a real database is a long process. Virtual databases are therefore really useful to study RFF identification scenarios and design a robust RFF identification protocols [43]. However the authors only give access to the final database [40], the community misses therefore a generic virtual database generator. This work addresses the need for the design of a robust database to perform RFF identification with DL. We propose a virtual database generator that creates a database based on the scenario description to study the DL RFF identification process in different scenarios and explore database design space such as the number of signals.

III. RiFYFi SYSTEM OVERVIEW

A. RiFYFi Identification Framework

In this section, we present our flexible framework for RFF identification coded in Julia language. Julia [9] is a high-level language, efficient in abstraction and execution, with many DL and telecom libraries [21]. The framework is composed of i) a virtual database generator which allows flexible exploration of many RFF scenarios and ii) a classification stage based on DL. The classification part is trained before being used as a classifier. The global framework presented in Figure 3 offers flexibility to use different classifiers in the classification stage or to use different virtual or real databases. With the virtual database generator, the framework takes as input the description of the database, called scenario parameters: number of signals per transmitter, the number of transmitters, size of the sequences, transmission context, and the information about network and training: the name of the architecture network, the number of training iteration called epoch.

The databases created by the generator offer flexibility to explore transmission contexts such as frame symbols, RFF parameters, and environmental conditions such as channel. The SoA shows some difficulties with channel variation [3] and concludes with the need to have a robust system for channel variation. The application context will determine the properties of a system that can be defined as robust. For applications where the time window between training and identification is narrow, generalization is not expected to be a problem as the channel will remain static, especially when considering motionless devices. On the contrary, an application with motion devices requires more generalization to be able to classify devices in different locations.

B. RFF Identification with Deep Learning

In recent years, DL has been massively used for classification for RFF as it could learn automatically how to classify transmitter transmitter [3], [35], [36], [40]. Figure 4 describes the classification procedure in the framework. This stage takes as an input the database and the name of the architecture network. The database has been previously created, labeled and split in Training and Test set. During the training part, the network takes signals from the training set grouped in batches as input. The labels of the signals in the batch are predicted

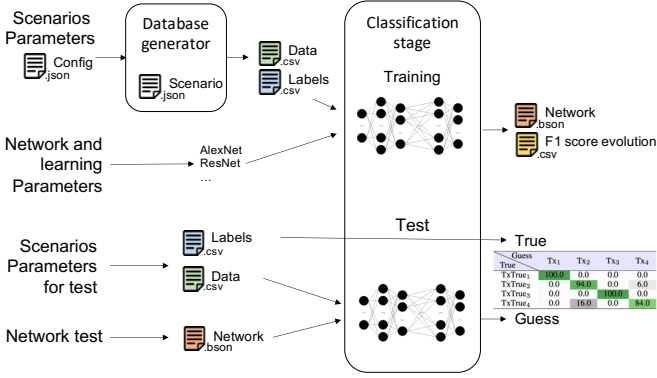


Fig. 3. Framework flow.

and compared with the true labels using cross-entropy as the loss function to apply the back-propagation. This process is repeated for each batch and each epoch. At the end of the training, the framework saves the network status in .bson file and saves the performance evolution during the training phase in a .csv file.

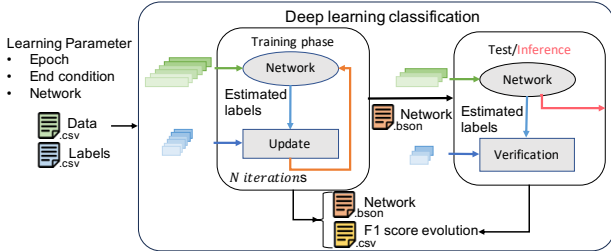


Fig. 4. DL classification in framework.

The network is considered as an independent block from the database with fixed inputs and outputs. The input size depends on the size of the raw input signal. The output depends on the number of classifying devices, the outputs of the network are the probability of belonging to a class. From these probabilities, the classification prediction is done.

To evaluate the network classification performance, two metrics are used: the F1 score and the accuracy. The accuracy is calculated by counting the number of correct predictions out of the total number of classifications. The F1 score is calculated on the batch sequences as follows:

$$F1 = \mathbb{E}_{c \in \mathcal{C}} \left(\frac{2}{\frac{1}{P(c)} + \frac{1}{R(c)}} \right), \quad (4)$$

$$\text{with } \begin{cases} P(c) = \frac{t_p(c)}{t_p(c) + f_p(c)} \\ R(c) = \frac{t_p(c)}{t_p(c) + f_n(c)} \end{cases}$$

where $\mathbb{E}[\cdot]$ stands for the expectancy operator applied here on all the classes $c \in \mathcal{C}$. $P(c)$ is called the precision for the class c and is a function of the number of true positives $t_p(c)$ and false positives $f_p(c)$. $R(c)$ is the recall for the class c and is function of $t_p(c)$ and the false negatives $f_n(c)$.

C. [A] Proposed Virtual Database and Radio Models

The database generator creates virtual databases to allow the exploration of impairments, database design, and learning models. Creating a virtual database requires wireless communication models, hardware impairments models, and wireless propagation channel models. This section describes models implemented in this database.

1) *Symbols*: First of all, the wireless communication model between an emitter and a receiver requires creating a signal for transmission. We consider that the binary sequence is modulated by QAM symbols and then followed by an Orthogonal Frequency-Division Multiplexing (OFDM) modulation with subcarrier-based pilot insertion. OFDM modulation is massively used in standard communication, particularly in the RFF database as it is shown in Table I and the signal varies greatly in amplitude which makes it interesting for the analysis of non-linear imperfections.

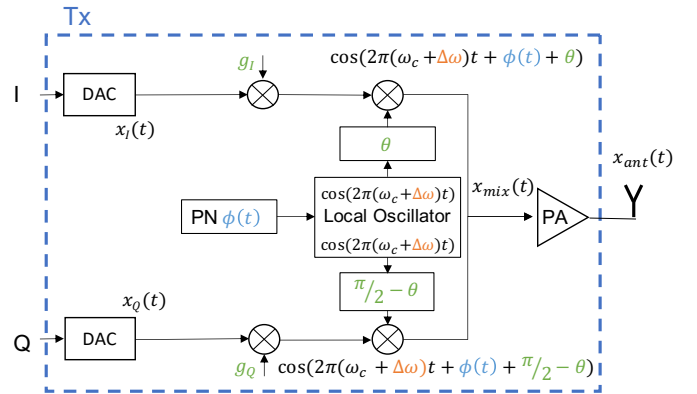


Fig. 5. Transmitter chain architecture and impairments.

2) *Transmitter impairment models*: The impairments modeling is described in the Figure 5 and is based on SoA models. We consider here a classic Zero Intermediate Frequency [A ?] modulation stage with I/Q modulation. The signal is multiplied by a carrier generated from a LO and different impairments occur in the transmission chain. As Zhang et al. [43] this study is focused on the main features: CFO impairments, gain and phase IQ imbalance, Phase Noise (PN), and Power Amplifier (PA) nonlinearity. The model of each impairment is described hereafter.

Before the LO, the analytical signal is modeled as:

$$\underline{x}(t) = x_I(t) + jx_Q(t), \quad (5)$$

where x_I and x_Q represent the real part and imaginary part of the complex signal. The LO allows modulating the signal to a carrier frequency noted f_c , this modulation may create three different impairments. The first one is called CFO, then the LO is polluted by a phase noise and an imbalance between the two branches, called IQ imbalance.

CFO impairments: The LO modulates the signal at the ideal carrier frequency, f_c . However, CFO impairments introduce

a frequency offset Δf , resulting in the effective carrier frequency, f_0 , noted as:

$$f_0 = f_c + \Delta f. \quad (6)$$

For the sake of brevity, the models are expressed in terms of angular frequency with $\omega_* = 2\pi f_*$. With such impairment, the modulated signal $x_{mix}(t)$ is expressed as a gain and phase error by:

$$x_{mix}(t) = x_I(t) \cos((\omega_c + \Delta\omega)t) - x_Q(t) \sin((\omega_c + \Delta\omega)t), \quad (7)$$

which can be equivalently written as

$$\begin{aligned} \underline{x}_{mix}(t) &= \underline{x}(t)e^{j(\omega_c + \Delta\omega)t} \\ x_{mix}(t) &= \Re(\underline{x}_{mix}(t)) \end{aligned} \quad (8)$$

All complex variables will be underlined in the rest of the paper.

IQ imbalance impairments: In the presence of imbalance, the LO can be expressed according to the Figure 5 in the form:

$$\begin{aligned} \underline{X}_{LO}(t) &= g_I \cos(\omega_0 t + \theta) + jg_Q \cos(\omega_0 t + \frac{\pi}{2} - \theta), \\ \underline{X}_{LO}(t) &= g_I \cos(\omega_0 t + \theta) - jg_Q \sin(\omega_0 t - \theta), \end{aligned} \quad (9)$$

where θ is the phase impairment considering balanced, and g_I and g_Q the gain impairments. The expression can be simplified as done by Valkama et al. [42]:

$$\begin{aligned} \underline{X}_{LO}(t) &= K_1 e^{-j\omega_0 t} + K_2 e^{j\omega_0 t}, \\ \text{where } K_1 &= \frac{g_I e^{-j\theta} + g_Q e^{j\theta}}{2}, \quad K_2 = \frac{g_I e^{j\theta} - g_Q e^{-j\theta}}{2}. \end{aligned} \quad (10)$$

The signal $\underline{x}_{mix}(t)$ at the output of the LO in the presence of IQ imbalance could be expressed:

$$\begin{aligned} \underline{x}_{mix}(t) &= \underline{x}(t) \times \underline{X}_{LO}(t), \\ &= \underline{x}(t)K_1 e^{-j\omega_0 t} + \underline{x}(t)K_2 e^{j\omega_0 t}. \end{aligned} \quad (11)$$

In our model, as it is often done in the SoA, a balanced IQ mismatch is considered with $g_I = g_Q = \frac{g}{2}$.

Phase noise impairments: The PN has been modeled in the literature with different models, like Gaussian, Wiener or Lorentz and we focus on the Wiener model as it is a commonly used case in the literature to model free oscillator [44]. The LO PN $\phi(t)$ may be modeled [30]:

$$\phi(t) = \sqrt{c}B(t), \quad (12)$$

where $B(t)$ denotes a standard Wiener process and parameter c describes the LO quality called diffusion rate [30]. $B(t)$ is defined as $B(t_2) - B(t_1)$ with t_1 and t_2 correspond to the duration that forms the noise of variance $\sqrt{t_2 - t_1}\mathcal{N}(0, 1)$. In the rest of the paper, we consider the digital Wiener PN model parameterized by its state noise variance σ_ξ^2 [10].

Considering all impairments described from now, the output of the LO could be expressed by:

$$\underline{x}_{mix}(t) = \underline{x}(t)K_1 e^{-j(\omega_0 t + \phi(t))} + \underline{x}(t)K_2 e^{j(\omega_0 t + \phi(t))}. \quad (13)$$

Power amplifier impairments without memory: At the end of the transmission chain, the PA amplifies a low-power signal to a higher-power one. To model the memoryless nonlinear effect of the power amplifier in our system, the Saleh model used in SoA [43] is chosen. The non-linearity is modeled as amplitude/amplitude (AM/AM) denoted $A(t)$ and amplitude/phase (AM/PM) distortions denoted $\xi(t)$.

$$\begin{aligned} A(t) &= \frac{\alpha_{AM} |\underline{x}_{mix}(t)|}{1 + \beta_{AM} |\underline{x}_{mix}(t)|^2}, \\ \xi(t) &= \frac{\alpha_{PM} |\underline{x}_{mix}(t)|^2}{1 + \beta_{PM} |\underline{x}_{mix}(t)|^2}, \end{aligned} \quad (14)$$

where $|\cdot|$ denoted L1 norm. $\alpha_{AM}, \alpha_{PM}, \beta_{AM}, \beta_{PM}$ are the parameters of Saleh model [43].

Finally, the signal $\underline{x}_{PA}(t)$ after the PA is modeled as

$$\underline{x}_{PA}(t) = A(t)e^{j(\angle \underline{x}_{mix}(t) + \xi(t))}, \quad (15)$$

where \angle represent the angle of $\underline{x}_{mix}(t)$.

Power amplifier impairments with memory: The power amplifier can be modeled by memory model, the signal $\underline{x}_{PAM}(t)$ after the PA is modeled as [29]:

$$x_{PAM}(t) = \sum_{\substack{p=1 \\ p \text{ odd}}}^P f_p(t) * (|\underline{x}_{mix}(t)|^{p-1} \underline{x}_{mix}(t)) \quad (16)$$

where P is the nonlinearity order of the model and $f_p(t)$ denotes the p th-order response of the polynomial model.

Conclusion: All these impairment models are implemented in RiFyFi. the value of each parameter as well as the similarities between devices are discussed in the next sections.

IV. PRACTICAL USE: FROM MODEL TO SCENARIOS

A. Simulator overview

In this section, the implementation and use of the virtual database generator are characterized, to create interesting databases to analyze. The generator offers the possibility to explore easily RFF identification scenarios, thanks to parametrization.

For all database creations, it is possible to choose:

- the number of transmitters N_{Tx} ,
- the similarities or dissimilarities between RFF emitters,
- the number of signals per device N_{signals} , in the database and used to train our network,
- the size of a signal $ChunkSize$
- the transmission symbols scenario
- the fingerprint scenario
- the channel scenario

As the SoA shows, the RFF identification conditions are multiples such as the frame of data used to identify the transmitter [19], the level of noise [20], the number of signals, the number of propagation channel views, the number of transmitters and the quality of them. Therefore, exploring these different scenarios with a single framework seems interesting and could help in designing a real database.

In RiFyFi, a signal *scenario* consists of key parameters that can be separated into 3 aspects: symbol generation, RFF,

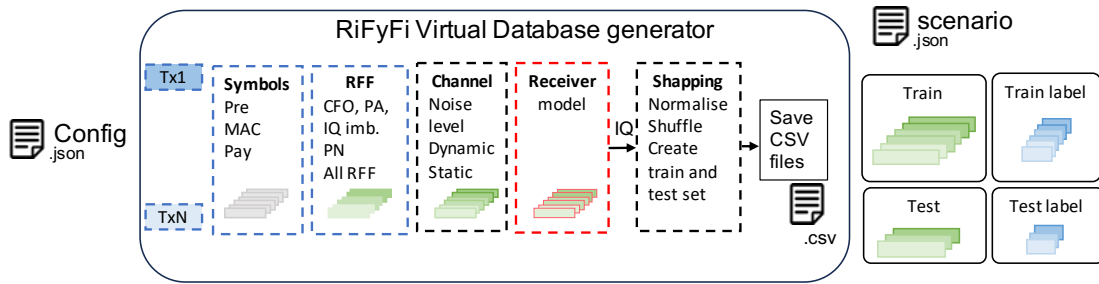


Fig. 6. Parametric database generator chain.

and propagation channel. The first one, the symbol, represents the part of the signal used for identification: Preamble, MAC address, or Payload (Pre, MAC, Pay). This is represented by the *Symbols* block in Figure 6. Then, the *RFF* block defines the transmitter impairments considered: CFO, PN, PA, IQ imbalance or all impairments. After the transmitter model description, the *channel* block defines environmental conditions, such as noise or channel model. Finally, it is possible to add a receiver model with its own RFF. Note that in this paper the impact of the receiver will not be explored and is left for future studies. In the end, a large matrix of IQ samples of size $(ChunkSize, 2, N_{signals} \times N_{Tx})$ is created and passes through a shaping block where data is shuffled and split to create, training and test sets. Then the database and the labels are saved in 4 CSV files to be used by the network. The labels matrix is composed of 2 dimensions $(N_{Tx}, N_{signals})$. The matrix contains only 0 and 1. The 1 allows us to attribute the signal to the corresponding transmitter.

In the context of this work, is followed the conventional approach found in the literature, which involves feeding the network with complex IQ signals, as it is done in [36] and [15]. The size of the input signal, varying in the SoA, is explored at the end of this paper.

B. Symbols Scenario

A *signal* is a sequence of *ChunkSize* complex IQ samples. In the first part of this paper 256 IQ samples are considered corresponding to the input size of the network [6], [16]. The baseline of the signal is created with OFDM symbols composed of 548 IQ samples with an FFT size of 512 and a cyclic prefix size of 36. This database generator can simulate different scenarios: Preamble, MAC address, and Payload.

Creating a Preamble-based database requires generating the same sequence of symbols for all emitters. It can be a specific data sequence or a special sequence such as Zadoff-Chu sequences. The second possibility is to generate a unique sequence for each transmitter. This scenario is close to a MAC address scenario where the signal contains the MAC address. The last possibility is to generate different sequences for each transmission, this scenario is called Payload where the identification can only be done through payload.

To simulate these scenarios, the generator creates a sequence called a Burst composed of 64 *signals* of 256 IQ samples, this burst is repeated to complete the $N_{signals}$. The preamble is the

same for each burst and each transmitter. In the MAC address scenario, a different burst is defined for each transmitter. Finally, in the Payload scenario, all bursts are different.

C. Fingerprint Scenario

Contrary to the symbols scenario, the fingerprint scenarios are not all realistic, but they offer exploration possibilities. First of all, it is possible to activate one or multiple impairments, to combine their effects. Seven scenarios are created and explored:

- CFO: only CFO impairment
- Imbalance: only gain and phase IQ imbalance impairment
- PN: PN
- PA: Power amplifier impairment with Saleh model
- PA_memory: PA with measured memory model
- All_impairments: CFO, imbalance, PA with Saleh model and PN.

All scenarios are studied in this paper. First, in section V impairments are independently studied, and then the most realistic scenario is addressed: All_impairments.

The impairment models described in Section III-C2 are parametric models. Each impairment is defined by one or multiple parameters. The gap between the values of an impairment parameter for two transmitters determines the RFF similarity between two transmitters. Other papers propose to create a grid to make sure that the space between two impairment values is sufficient. In Zhang paper [43], the impairments follow a uniform random distribution within an interval. This paper presents the study of the impact of the space between two impairment values with different similarity percentages around a mean value.

D. DL Network

The flexible nature of the framework offers the possibility to explore different network architectures or other classification solutions. The network chosen in this paper has been proposed and studied for RFF identification in [35], which is one of the first contributions on RFF with DL solution and public database. It is a CNN inspired by AlexNet, with 4 convolutional layers, each layer is composed of two blocks of 128 filters size 7×1 and 5×1 and a max-pooling stage. The chosen activation function is ReLu and the optimizer is Adam with γ learning rate. After the 4 convolutional layers, the CNN has 3 fully connected layers with 256 nodes, 128

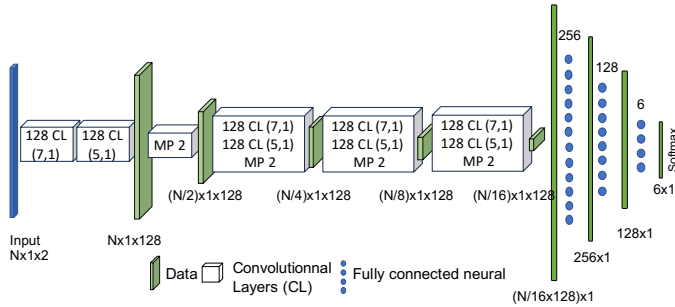


Fig. 7. Deep Learning Architecture

TABLE II
MEAN VALUE CHOSEN FOR IMPAIRMENT PARAMETERS.

Impairment	Parameters	Mean value
CFO	Δf	300 Hz
Imbalance	g_Q	1.5 dB
	θ	2.5°
PN	σ_ϕ^2	10^{-7}
PA	α_{AM}	2.1587
	β_{AM}	1.1517
	α_{PM}	4.0033
	β_{PM}	9.104

and the number of classes (in this case 6 classes). After the two first fully connected layers a dropout layer is added with $dr \in [0; 1]$. In input, the network takes complex-based band signals without pre-processing. These signals are split into two rows I and Q, and the input size N corresponds to $ChunkSize$. For $N = 256$ as done in [6], [16], this architecture is presented in Figure 7 and has 1,232,774 parameters. In Figure 7, the notation corresponds to $N_{Filters}CL(a, b)$ for Convolutional Layers with (a,b) the size of the filter, and MP 2 for the Maxpooling 2.

V. INVESTIGATION OF THE INDIVIDUAL IMPACT OF IMPAIRMENTS

In this section, impairments are separately studied with different confidence intervals around a fixed mean value, inspired by the SoA [43] and defined in Table II. [To Complete/To do] The confidence interval is a metric to modelized the quality of transmitters. The RFF identification complexity depends on the similarities between the RFF transmitters. For a given number of transmitters, a large interval reduces the similarity between two transmitters. However a small interval increases the RFF similarity, and, therefore, it makes the identification difficult. For this study, some learning parameters are empirically adjusted upstream for each impairment in order to compare them in favorable [To Complete/To do] situations. The parameters are specifically the dropout and the learning rate. [A]For this study, we chose the number of transmitters as a function of the number of parameters we have to explore. The results presented in this section are obtained by means of 5 different seeds, the different colors evaluated the performance.

TABLE III
MEAN F1 SCORE EVOLUTION DURING TRAINING PHASE FOR DIFFERENT CFO IMPAIRMENTS, $\gamma = 10^{-4}$

F1 score at	20 epochs		50 epochs		100 epochs		315 epochs	
	Train	Test	Train	Test	Train	Test	Train	Test
5%	98%	98%						
2%	93%	91%	98%	98%				
1%	53%	51%	92%	89%	97%	95%		
0.5%	48%	46%	52%	47%	57%	56%	98%	87%

A. CFO

To study the CFO impairment, we set the mean value at 300Hz and create different similarities with $p\%$ for two transmitters with the following CFO values:

$$\Delta f_{Tx1} = \bar{\Delta f}(1 - p\%), \quad (17)$$

$$\Delta f_{Tx2} = \bar{\Delta f}(1 + p\%). \quad (18)$$

with $p = 5\%$, 2% , 1% and 0.5% . The CFO values of both transmitters for each similarity scenario p are given in Appendix (Table XV). Figure 8 presents the F1 score evolution during the training phase and Table III summarizes the results with the mean F1 score obtained during the training phase on the training set and test set at different epochs. For the next impairments, and for the sake of conciseness we only use the tables to present results.

Results are obtained with a learning rate $\gamma = 10^{-4}$ and no dropout and they show that narrowing the impairment interval between two transmitters increases the network's difficulty in learning how to distinguish between these transmitters. Nevertheless, this is compounded by the fact that numerous studies have demonstrated the instability of the CFO, which further exacerbates the situation we will not focus on this. The study of CFO highlights the link between RFF transmitters similarity scenario and the capacity of the network to separate transmitters.

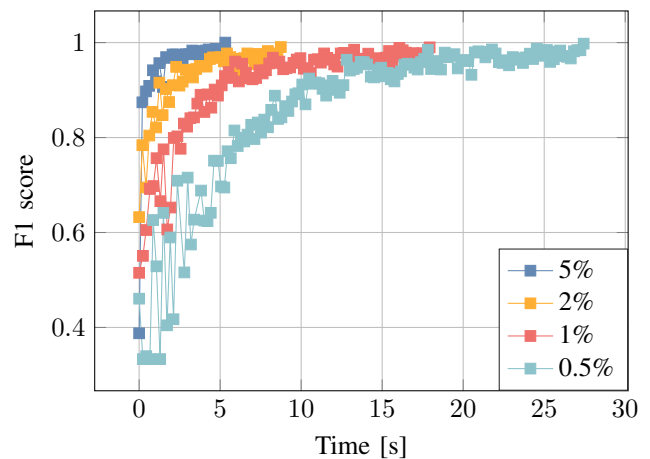


Fig. 8. Time evolution of the F1 score of the different CFO impairments values.

TABLE IV
MEAN F1 SCORE EVOLUTION DURING TRAINING PHASE FOR DIFFERENT IQ IMBALANCE IMPAIRMENTS, $\gamma = 10^{-4}$

F1 score at	85 Epochs		130 Epochs		210 Epochs	
	Train	Test	Train	Test	Train	Test
g: 10% [0°;5°]	92%	88%				
g: 10% [1°;4°]	66%	61%	95%	80%		
g: 10% [2°;3°]	44%	42%	48%	44%	84%	48%
g: 5% [0°;5°]	94%	91%				
g: 5% [1°;4°]	69%	66%	89%	79%		
g: 5% [2°;3°]	58%	49%	67%	48%	96%	50%
g: 3% [0°;5°]	90%	87%				
g: 3% [1°;4°]	70%	64%	86%	76%		
g: 3% [2°;3°]	58%	49%	58%	48%	86%	53%
g: 1% [0°;5°]	63%	55%	84%	55%		
g: 1% [1°;4°]	30%	28%	69%	55%	91%	51%
g: 1% [2°;3°]	12%	12%	13%	13%	70%	18%

B. IQ imbalance

As it is done with the CFO, we explore different similarity configurations for IQ imbalance, defined by g_Q, g_I and θ with $g_Q = -g_I$ and $\theta \in [\theta_{min}; \theta_{max}]$ with 4 transmitters.

$$g_{QTx1} = g_{QTx3} = \bar{g}(1 - p\%) \quad (19)$$

$$g_{QTx2} = g_{QTx4} = \bar{g}(1 + p\%) \quad (20)$$

$$\theta_{Tx1} = \theta_{Tx2} = \theta_{max} \quad (21)$$

$$\theta_{Tx3} = \theta_{Tx4} = \theta_{min} \quad (22)$$

The two impairments, gain and phase have been explored together by testing all combinations, with p equal 10%, 5%, 3% and 1% and the interval $[\theta_{min}; \theta_{max}]$ takes $[0^\circ; 5^\circ]$, $[1^\circ; 4^\circ]$ and $[2^\circ; 3^\circ]$. The Table XVI in the Appendix presents gain and phase values for each transmitter. The Results are obtained without dropout and a learning rate at 10^{-4} .

Table IV presents F1 score values at different times for the different impairment combinations. Comparing the first rows of results with a 10% similarity scenario shows that increasing the phase similarities from $[0^\circ; 5^\circ]$ to $[1^\circ; 4^\circ]$ increases the time required for the network to converge by increasing the number of epochs. Then, comparing the first row of the 10% similarity scenario and the first row of the 3% one, shows a slight difference in F1 score value at the same time. Moreover in combination with $[2; 3]$ and gain over 3%, results show an over-learning on training data as it stops around 50% on Test data. The analysis of the confusion matrix in Table V, under 10% and $[2^\circ; 3^\circ]$ similarity conditions, reveals an effective classification of transmitters 1 with 89% of correct classification. However, it exhibits confusion between Tx1 and Tx3, as well as Tx2 and Tx4. In summary, a 1° gap between two transmitters is insufficient for clear differentiation.

When the IQ imbalance gain is set below 1%, during training, the network tends to over-learn and stops at around 50%. This is confirmed by the confusion matrix in Table VI, which highlights a classification issue, as the network only seems to recognize two classes.

The study of IQ imbalance shows a decrease in convergence speed when the similarity between impairments decreases for gain and phase with a limit for recognizing devices.

TABLE V
CONFUSION MATRIX FOR TEST DATA FOR IQ IMBALANCE IMPAIRMENT, G: 10% AND $[2; 3]$ COMBINATION.

Guess \ True	Tx ₁	Tx ₂	Tx ₃	Tx ₄
TxTrue ₁	89.0	0.0	11.0	0.0
TxTrue ₂	0.0	52.0	0.0	48.0
TxTrue ₃	82.0	0.0	18.0	0.0
TxTrue ₄	0.0	48.0	0.0	52.0

TABLE VI
CONFUSION MATRIX FOR TEST DATA FOR IQ IMBALANCE IMPAIRMENT, G: 1% AND $[2; 3]$ COMBINATION.

Guess \ True	Tx ₁	Tx ₂	Tx ₃	Tx ₄
TxTrue ₁	67.0	0.0	18.0	15.0
TxTrue ₂	66.0	0.0	17.0	17.0
TxTrue ₃	67.0	0.0	19.0	14.0
TxTrue ₄	65.0	0.0	18.0	17.0

C. Phase Noise

To study the PN 4 different PN for 4 different transmitters are chosen, and the value of the PN is presented in Table XVIII in Appendix. [A]The PN is a particular impairment because it is a noise so it is difficult to find the specific difference between transmitters only based one PN. The experiments are done with different learning rates and dropout but the results are constant, after around 500 epochs the network has reached 99% in training data but stayed at 27% in test. The results show that the network learns to separate the signals from different transmitters for the training set, but it over-learns and it is not able to separate the transmitters in the testing set. Due to its variability and because it alters preamble quality, PN cannot be a relevant impairment to be able to separate the transmitters even with large difference in noise statistics.

D. PA model

To study the PA effect, two types of model introduced in section III-C2 are used. The first one is the Saleh model, and the second is the memory model based on real coefficient. This memory model is presented by [2]. The memory model contains around 100 emitters models but we extract 2 groups of 4 PA models to better stress the impact of closed PA configurations "MM close" and distinct PA configurations "MM far". For the Saleh model, the different parameters are presented in Table XVII for different impairment similarities expressed as: For α_{AM} and α_{PM}

$$\alpha_{Tx_i} = \alpha_{Tx_{i+2}} = \alpha(1 - p\%) \text{ with } i = 1, \quad (23)$$

$$\alpha_{Tx_i} = \alpha_{Tx_{i+2}} = \alpha(1 + p\%) \text{ with } i = 2. \quad (24)$$

For β_{AM} and β_{PM}

$$\beta_{Tx_i} = \beta_{Tx_{i+1}} = \beta(1 + p\%) \text{ with } i = 1, \quad (25)$$

$$\beta_{Tx_i} = \beta_{Tx_{i+1}} = \beta(1 - p\%) \text{ with } i = 3. \quad (26)$$

Different experiments are done and present instability of the network during the training phase. To reduce this problem the

TABLE VII

MEAN F1 SCORE EVOLUTION DURING TRAINING PHASE FOR DIFFERENT POWER AMPLIFIER IMPAIRMENTS, $\gamma = 10^{-5}$ DR = 0.25

F1 score at	160 epochs		500 epochs		970 epochs	
	Train	Test	Train	Test	Train	Test
5%	98%	99%				
2%	74%	70%	96%	95%		
1%	28%	26%	80%	59%	95%	76%
0.5%	24%	23%	50%	38%	81%	49%
0.3%	25%	23%	44%	24%	87%	23%
MM far	86%	87%	91%	92%	93%	93%
MM close	33%	31%	57%	49%	74%	54%

dropout is put at 0.25 and the learning rate is decreased at 10^{-5} . Table VII presents the F1 score value during training for the train dataset and test dataset. This shows a decrease in convergence speed when the similarity between impairments increases and for $p \leq 0.5\%$ the network overlearns on training data. The use of the memory model allows us to show the flexibility of our framework in particular the interest of the generator is to use any RFF parametric models. Finally, it shows that the results obtained with the Saleh model are realistic in terms of convergence speed.

E. Conclusion of individual impairment effects

The investigation of the individual impact of impairment reveal the link between the RFF impairments similarity and the capacity of the network to classify several devices. The impairments are not all relevant, [A] in particular the PA and IQ imbalance seems to be interesting. This study shows the importance of tuning learning parameter to adapt the network to the data.

VI. [A] CONGLOMERATE SCENARIOS STUDY

In this section, different transmission scenarios with all impairments are studied with $N_{Tx} = 6$ transmitters and $p\%$ interval, for IQ imbalance, CFO, PN and PA. For the PN two variances of state noise center around 10^{-7} and 10^{-4} are explored. The values chosen for each parameter for the 6 transmitters are calculated following (29), the Table XIX in Appendix presents the parameter values calculated for $p = 5\%$ similarity. Except for θ , the parameter values P_{Txk}^p for device $k \in [1, N_{Tx}]$ are computed as:

$$P_{Txk}^p = P_{min}^p + k \frac{(P_{max}^p - P_{min}^p)}{N_{Tx}}, \quad (27)$$

$$\text{with } P_{min}^p = \text{Mean Value}(1 - p\%), \quad (28)$$

$$P_{max}^p = \text{Mean Value}(1 + p\%), \quad (29)$$

with P_{min}^p the minimum of the impairment parameter in the $p\%$ similarities scenario and P_{max}^p the maximum. Four different similarity scenarios are studied in this paper, 5%, 3%, 2% and 1%. For θ parameters, P_{Txk}^p follows (18) but P_{min}^p and P_{max}^p depend of the similarity scenario. For $p = 5\%$ and 3%, we set:

$$P_{min}^p = 0^\circ \quad P_{max}^p = 5^\circ, \quad (30)$$

while for $p = 2\%$ and 1%, we set:

$$P_{min}^p = 1^\circ \quad P_{max}^p = 4^\circ. \quad (31)$$

A. Preamble scenario

1) *How close can the RFF of 6 devices be ?*: This section addresses the convergence speed of the CNN in preamble scenarios with all impairments and different contexts. The databases are composed of 6 emitters with 1000 signals per emitters. Each database is split into 90% and 10% to create training and test sets, respectively. Table IX presents the F1 score values during training for both training and test sets, during the training phase for the different similarity scenarios. The training is ended when the network obtains a F1 score of 98% on the training set. First, at 5% similarity, two state noise variances of the PN scenarios are studied, 10^{-7} and 10^{-4} . The results show that increasing the state noise variance increases classification and generalization difficulties. At 30 epochs the network as reached 98% for test in 10^{-7} scenario, but for 10^{-4} the network obtain 93% on test. The results are interesting and present good performance for both PN scenarios with worst result in test for 10^{-4} as the first study shows the PN was not relevant but could disturb the network by adding noise and making the identification difficult. For the rest of the paper, phase noise is set around 10^{-7} .

Then different similarity scenarios: 5%, 3%, 2% and 1% are compared. The network has no difficulty in classifying the 6 transmitters in the 5% scenario. As the similarity increases, the network needs more time to learn and classify the devices. The complexity of the classification problem increases as device impairments become closely situated, making it more challenging for the network to distinguish between them. To solve this issue it is possible to change some learning parameters such as the learning rate and add dropout to avoid overfitting. In the 2% similarity scenario, these changes improve the F1 score in the test but not enough. Furthermore, in the case of the 1% similarity scenario, the test F1 score remains at approximately 18%, close to random value $1/N_{Tx}$. This suggests that the network struggles to learn RFF due

TABLE VIII

MEAN F1 SCORE EVOLUTION DURING TRAINING PHASE FOR PREAMBLE AND DIFFERENT SIMILARITY SCENARIOS, $\gamma = 10^{-4}$ DR = 0

F1 score at	30 epochs		60 epochs		280 epochs	
	Train	Test	Train	Test	Train	Test
5% and PN 10^{-4}	96%	93%				
5% and PN 10^{-7}	98%	98%				
3%	48%	45%	96%	85%		
2%	32%	25%	43.6%	19%	85%	30%
1%	31%	17%	45%	17%	84%	18%

TABLE IX

MEAN F1 SCORE EVOLUTION DURING TRAINING PHASE FOR PREAMBLE AND DIFFERENT SIMILARITY SCENARIOS, $\gamma = 10^{-5}$ DR = 0.25

F1 score at	280 epochs		500 epochs		1100 epochs	
	Train	Test	Train	Test	Train	Test
2%	42%	17%	73%	30%	97%	45%
1%	41%	16%	76%	19%	99%	17%

TABLE XIV
TIME REQUIRED FOR THE NETWORK TO REACH 98% OF F1 SCORE ON TRAINING DATA, IN PAYLOAD CONTEXT.

Number of signals	5%	10%
900	27s	25s
4500	160s	110s
9000	8min	220s
45000	45min	19min
900000	2h	38min
4500000	+10h	2h12

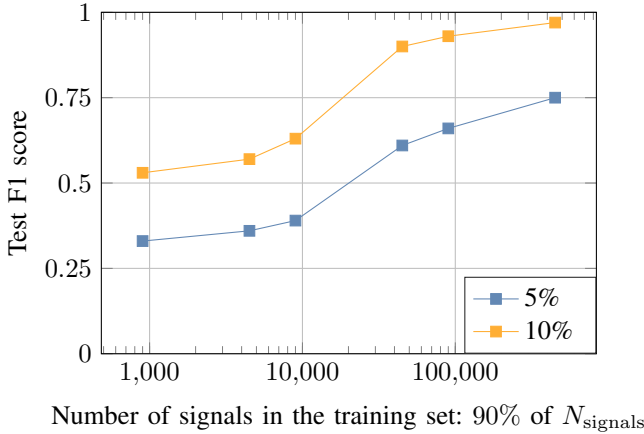


Fig. 10. F1 score obtained in test in function of the number of signals used to train the network when training has reached 98% of F1 score.

required to obtain robust RFF learning in the Payload scenario is studied. Such a scenario represents non-correlated data and it is complex for the network, for example with 900 signals at 5% similarity the F1 score in the test stays around 30% compared to the Preamble situation where the network achieves 98% in the test. In the Payload scenario, the network overfits on training data, to avoid this issue the number of signals used in the train is explored. The results are obtained with a learning rate at 10^{-4} and no dropout. Figure 10 presents the F1 score obtained in the Test set when the network has reached 98% of the F1 score on the training set, obtained for 2 different similarity scenarios. In blue, we represent the 5% similarity scenario database, and yellow represents the 10% similarity scenario. Table XIV completes the results by adding the time of training to reach 98% of the F1 score.

The Table and Figure present great improvement in F1 score in the test when the number of signals is increased. [A] However the Table presents the time required to achieve the different training and the time convergence speed increase because of the number of data seen in an epoch.

VII. CONCLUSION

This paper introduces a virtual database generator based on wireless transmission and RFF models included in a flexible framework for RFF identification. This work proposes an exploration of database design for RFF identification with DL considering the similarity between the RFF of transmitters, the transmission scenario, and the number of signals.

Our analysis showed the impact of similarity between RFF transmitters on the network convergence speed and the F1 score performance in a preamble context. For a 2% similarity, decreasing the learning rate and adding dropout helped the network to improve the classification in test from 30% to 45% and augmented increasing the number of signals permitted to achieve 65% of accuracy.

A very large number of signals per transmitter is needed when the RFF similarities are strong between transmitters or in a Payload context consequently having similar RFF devices can be a countermeasure to avoid RFF identification. The virtual database generator can help to pre-assess the needed database design. The authors commit to release this generator as an open-source tool immediately after acceptance of the paper.

VIII. APPENDIX

Below the parameter tables used in the various above-mentioned scenarios are presented.

TABLE XV
CFO VALUES FOR DIFFERENT SIMILARITY SCENARIOS

300Hz	Tx1	Tx2
CFO 5%	285	315
CFO 2%	294	306
CFO 1%	297	303
CFO 0.5%	298.5	301.5

TABLE XVI
GAIN AND PHASE IMPAIRMENTS VALUES FOR DIFFERENT IQ IMBALANCE

	p%	Tx1	Tx2	Tx3	Tx4
Gain	g: 10%	1.350	1.650	1.350	1.650
	g: 5%	1.425	1.575	1.425	1.575
	g: 3%	1.455	1.545	1.455	1.545
	g: 1%	1.485	1.515	1.485	1.515
Phase	[0°;5°]	5°	5°	0°	0°
	[1°;4°]	4°	4°	1°	1°
	[2°;3°]	3°	3°	2°	2°

TABLE XVII
VALUES OF IMPAIRMENTS FOR DIFFERENT PA IMPAIRMENTS,

p%	parameter	Tx1	Tx2	Tx3	Tx4
5%	α_{AM}	2.051	2.267	2.051	2.267
	β_{AM}	1.209	1.209	1.094	1.094
	α_{PM}	3.803	4.203	3.803	4.203
	β_{PM}	9.559	9.559	8.649	8.649
2%	α_{AM}	2.116	2.202	2.116	2.202
	β_{AM}	1.175	1.175	1.129	1.129
	α_{PM}	3.923	4.083	3.923	4.083
	β_{PM}	9.286	9.286	8.922	8.922
1%	α_{AM}	2.137	2.180	2.137	2.180
	β_{AM}	1.163	1.163	1.140	1.140
	α_{PM}	3.963	4.043	3.963	4.043
	β_{PM}	9.195	9.195	9.013	9.013
0.5%	α_{AM}	2.148	2.169	2.148	2.169
	β_{AM}	1.157	1.157	1.146	1.146
	α_{PM}	3.983	4.023	3.983	4.023
	β_{PM}	9.150	9.150	9.058	9.058
0.3%	α_{AM}	2.152	2.165	2.152	2.165
	β_{AM}	1.155	1.155	1.148	1.148
	α_{PM}	3.991	4.015	3.991	4.015
	β_{PM}	9.131	9.131	9.077	9.077

TABLE XVIII
PHASE NOISE

Tx1	Tx2	Tx3	Tx4
10^{-5}	10^{-4}	10^{-6}	10^{-7}

REFERENCES

- [1] Adl-id: Adversarial disentanglement learning for wireless device fingerprinting temporal domain adap author=Elmaghbbub, Abdurrahman and Hamdaoui, Bechir and Wong, Weng-Keen, journal=arXiv preprint arXiv:2301.12360, year=2023.
- [2] Mahmoud Abdelaziz, Lauri Anttila, Alberto Brihuega, Fredrik Tufvesson, and Mikko Valkama. Digital pre-distortion for hybrid mimo transmitters. *IEEE Journal of Selected Topics in Signal Processing*, 12(3):445–454, 2018.
- [3] Amani Al-Shawabka, Francesco Restuccia, Salvatore D’Oro, Tong Jian, Bruno Costa Rendon, Nasim Soltani, Jennifer Dy, Stratis Ioannidis, Kaushik Chowdhury, and Tommaso Melodia. Exposing the Fingerprint: Dissecting the Impact of the Wireless Channel on Radio Fingerprinting. In *2020 INFOCOM IEEE Conference on Computer Communications*, pages 646–655, Toronto, ON, Canada.
- [4] Fadele Ayotunde Alaba, Mazliza Othman, Ibrahim Abaker Targio Hashem, and Faiz Alotaibi. Internet of things security: A survey. *Journal of Network and Computer Applications*, 2017.
- [5] Vladimir Brik, Suman Banerjee, Marco Gruteser, and Sangho Oh. Wireless device identification with radio-metric signatures. In *2008 Proceedings of the 14th ACM international conference on Mobile computing and networking - MobiCom 08*, page 116, San Francisco, California, USA. ACM Press.
- [6] Alice Chillet, Baptiste Boyer, Robin Gerzaguët, Karol Desnos, and Matthieu Gautier. Tangled program graph

TABLE XIX
VALUES OF IMPAIRMENTS FOR DIFFERENT ALL IMPAIRMENTS,

Parameter 5%	Tx1	Tx2	Tx3	Tx4	Tx5	Tx6
Imbalance g	1.425	1.455	1.485	1.515	1.545	1.575
Imbalance θ	0.000	0.017	0.035	0.052	0.070	0.087
CFO Δf	285	291	297	303	309	315
PN $\sigma^2 10^{-7}$	0.950	0.970	0.990	1.01	1.03	1.05
PN $\sigma^2 10^{-4}$	0.950	0.970	0.990	1.01	1.03	1.05
PA α_{AM}	2.051	2.094	2.137	2.180	2.223	2.267
PA β_{AM}	1.094	1.117	1.140	1.163	1.186	1.209
PA α_{PM}	3.803	3.883	3.963	4.043	4.123	4.203
PA β_{PM}	8.649	8.831	9.013	9.195	9.377	9.559

for radio-frequency fingerprint identification. In *2023 Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (IEEE PIMRC 2023)*, 2023.

- [7] Abdurrahman Elmaghbbub and Bechir Hamdaoui. LoRa device fingerprinting in the wild: Disclosing RF data-driven fingerprint sensitivity to deployment variability. *IEEE Access*, 9:142893–142909, 2021.
- [8] Abdurrahman Elmaghbbub and Bechir Hamdaoui. Eps: Distinguishable iq data representation for domain-adaptation learning of device fingerprints. *arXiv preprint arXiv:2308.04467*, 2023.
- [9] Kaifeng Gao, Gang Mei, Francesco Piccialli, Salvatore Cuomo, Jingzhi Tu, and Zenan Huo. Julia language in machine learning: Algorithms, applications, and open issues. *Computer Science Review*, 37:100254, 2020.
- [10] Robin Gerzaguët, Laurent Ros, Fabrice Belvèze, and Jean-Marc Brossier. On the performance of digital adaptive spur cancellation for multi-standard radio frequency transceivers. *Digital Signal Processing*, 33:83–97, 2014.
- [11] Hanqing Gu, Lisheng Su, Weifeng Zhang, and Chuan Ran. Attention is needed for rf fingerprinting. *IEEE Access*, 11:87316–87329, 2023.
- [12] Xinghao Guo, Zhen Zhang, and Jie Chang. Survey of Mobile Device Authentication Methods Based on RF Fingerprint. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs)*, pages 1–6, Paris, France, April 2019. IEEE.
- [13] Jeyanthi Hall, Michel Barbeau, and Evangelos Kranakis. Detection of Transient in Radio Frequency Fingerprint using Signal Phase. page 6.
- [14] Samer Hanna, Samurdhi Karunaratne, and Danijela Cabric. Open set wireless transmitter authorization: Deep learning approaches and dataset considerations. *IEEE Transactions on Cognitive Communications and Networking*, 7(1):59–72, 2020.
- [15] Samer Hanna, Samurdhi Karunaratne, and Danijela Cabric. WiSig: A Large-Scale WiFi Signal Dataset for Receiver and Channel Agnostic RF Fingerprinting. Technical Report arXiv:2112.15363, arXiv, 2022.
- [16] Samer Hanna, Samurdhi Karunaratne, and Danijela Cabric. Wisig: A large-scale wifi signal dataset for receiver and channel agnostic rf fingerprinting. *IEEE Access*, 10:22808–22818, 2022.

- [17] Anu Jagannath and Jithin Jagannath. Embedding-Assisted Attentional Deep Learning for Real-World RF Fingerprinting of Bluetooth. 9 2022.
- [18] Anu Jagannath, Jithin Jagannath, and Prem Sagar Patantashetty Vasanth Kumar. A Comprehensive Survey on Radio Frequency (RF) Fingerprinting: Traditional Approaches, Deep Learning, and Open Challenges. *arXiv:2201.00680 [cs]*, 2022.
- [19] Tong Jian, Bruno Costa Rendon, Andrey Gritsenko, Jennifer Dy, Kaushik Chowdhury, and Stratis Ioannidis. Mac id spoofing-resistant radio fingerprinting. In *2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 1–5. IEEE, 2019.
- [20] Tong Jian, Bruno Costa Rendon, Emmanuel Ojuba, Nasim Soltani, Zifeng Wang, Kunal Sankhe, Andrey Gritsenko, Jennifer Dy, Kaushik Chowdhury, and Stratis Ioannidis. Deep Learning for RF Fingerprinting: A Massive Experimental Study. in *2020 IEEE Internet of Things Magazine*, 3(1):50–57.
- [21] Corentin Lavaud, Robin Gerzaguët, Matthieu Gautier, and Olivier Berder. AbstractSDRs: Bring down the two-language barrier with Julia Language for efficient SDR prototyping. *IEEE Embedded Systems Letters*, pages 1–1, 2021.
- [22] Yun Lin, Jicheng Jia, Sen Wang, Bin Ge, and Shiwen Mao. Wireless Device Identification Based on Radio Frequency Fingerprint Features. In *2020 IEEE International Conference on Communications (ICC)*, pages 1–6, Dublin, Ireland.
- [23] Kevin Merchant, Shauna Revay, George Stantchev, and Bryan Nousain. Deep learning for rf device fingerprinting in cognitive communication networks. *IEEE journal of selected topics in signal processing*, 12(1):160–167, 2018.
- [24] Noemí Miguélez-Gómez and Eduardo A. Rojas-Nastrucci. Rf fingerprinting: Hardware-trustworthiness enhancement in the hardware trojan era: Rf fingerprinting-based countermeasures. *IEEE Microwave Magazine*, 24(11):35–52, 2023.
- [25] Cyrille Morin, Leonardo Cardoso, Jakob Hoydis, Jean-Marie Gorce, and Thibaud Vial. Transmitter Classification With Supervised Deep Learning. *arXiv:1905.07923 [cs, eess]*, 2019.
- [26] Braeden P. Muller, Lauren J. Wong, William H. Clark, and Alan J. Michaels. A real-world dataset generator for specific emitter identification. *IEEE Access*, 11:110023–110038, 2023.
- [27] Linning Peng, Aiqun Hu, Junqing Zhang, Yu Jiang, Jiabao Yu, and Yan Yan. Design of a Hybrid RF Fingerprint Extraction and Device Classification Scheme. *IEEE Internet of Things Journal*, 6(1):349–360, February 2019.
- [28] Linning Peng, Junqing Zhang, Ming Liu, and Aiqun Hu. Deep Learning Based RF Fingerprint Identification Using Differential Constellation Trace Figure. *IEEE Transactions on Vehicular Technology*, 69(1):1091–1095, January 2020.
- [29] Jaakko Pihlajasalo, Dani Korpi, Mikko Honkala, Janne MJ Huttunen, Taneli Riihonen, Jukka Talvitie, Alberto Brihuega, Mikko A Uusitalo, and Mikko Valkama. Deep learning ofdm receivers for improved power efficiency and coverage. *IEEE Transactions on Wireless Communications*, 2023.
- [30] Thierry Pollet, Mark Van Bladel, and Marc Moeneclaey. BER sensitivity of OFDM systems to carrier frequency offset and wiener phase noise. *IEEE Transactions on communications*, 43(2/3/4):191–193, 1995.
- [31] Saeed Ur Rehman, Kevin Sowerby, and Colin Coghill. Analysis of receiver front end on the performance of RF fingerprinting. In *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 2494–2499, Sydney, Australia.
- [32] Guillem Reus-Muns, Dheryta Jaisinghani, Kunal Sankhe, and Kaushik R Chowdhury. Trust in 5g open rans through machine learning: Rf fingerprinting on the powder pawr platform. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*, pages 1–6. IEEE, 2020.
- [33] Shamnaz Riyaz, Kunal Sankhe, Stratis Ioannidis, and Kaushik Chowdhury. Deep learning convolutional neural networks for radio identification. *IEEE Communications Magazine*, 56(9):146–152, 2018.
- [34] Debashri Roy, Tathagata Mukherjee, Mainak Chatterjee, Erik Blasch, and Eduardo Pasilliao. Rfal: Adversarial learning for rf transmitter identification and classification. *IEEE Transactions on Cognitive Communications and Networking*, 6(2):783–801, 2019.
- [35] Kunal Sankhe, Mauro Belgiovine, Fan Zhou, Luca Angioloni, Frank Restuccia, Salvatore D’Oro, Tommaso Melodia, Stratis Ioannidis, and Kaushik Chowdhury. No Radio Left Behind: Radio Fingerprinting Through Deep Learning of Physical-Layer Hardware Impairments. in *2020 IEEE Transactions on Cognitive Communications and Networking*, 6(1):165–178.
- [36] Kunal Sankhe, Mauro Belgiovine, Fan Zhou, Shamnaz Riyaz, Stratis Ioannidis, and Kaushik Chowdhury. ORACLE: Optimized Radio Classification through Convolutional neural networks. In *2019 INFOCOM IEEE Conference on Computer Communications*, pages 370–378, Paris, France.
- [37] Guanxiong Shen, Junqing Zhang, Alan Marshall, and Joseph Cavallaro. Towards Scalable and Channel-Robust Radio Frequency Fingerprint Identification for LoRa. *arXiv:2107.02867 [eess]*, July 2021.
- [38] Guanxiong Shen, Junqing Zhang, Alan Marshall, Mikko Valkama, and Joseph Cavallaro. Radio Frequency Fingerprint Identification for Security in Low-Cost IoT Devices. *arXiv:2111.14275 [eess]*, 2021.
- [39] Yong Sheng, Keren Tan, Guanling Chen, David Kotz, and Andrew Campbell. Detecting 802.11 mac layer spoofing using received signal strength. In *2008 IEEE INFOCOM The 27th Conference on Computer Communications*.
- [40] Nasim Soltani, Kunal Sankhe, Jennifer Dy, Stratis Ioan-

- nidis, and Kaushik Chowdhury. More Is Better: Data Augmentation for Channel-Resilient RF Fingerprinting. in *2020 IEEE Communications Magazine*, 58(10):66–72.
- [41] J. Toonstra and W. Kinsner. Transient analysis and genetic algorithms for classification. In *IEEE WESCANEX 95. Communications, Power, and Computing. Conference Proceedings*, volume 2, pages 432–437, Winnipeg, Man., Canada, 1995. IEEE.
- [42] Mikko Valkama, Markku Renfors, and Visa Koivunen. Blind i/q signal separation-based solutions for receiver signal processing. *EURASIP Journal on Advances in Signal Processing*, 2005(16):1–11, 2005.
- [43] Junqing Zhang, Roger Woods, Magnus Sandell, Mikko Valkama, Alan Marshall, and Joseph Cavallaro. Radio Frequency Fingerprint Identification for Narrowband Systems, Modelling and Classification. *IEEE Transactions on Information Forensics and Security*, 16:3974–3987, 2021.
- [44] Zhen Zhang, Aiqun Hu, Wei Xu, Jiabao Yu, and Yang Yang. An artificial radio frequency fingerprint embedding scheme for device identification. *IEEE Communications Letters*, 26(5):974–978, 2022.