Physical Layer Identification and authentication of electronic devices

by

Gianmarco Baldini

Laurea Electronic Engineering, University La Sapienza (1993)

Submitted to the Department of Computer Science of University of Insubria, Varese, Italy
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

University of Insubria, Varese, Italy

December 2019

Author .....
Department of Computer Science of University of Insubria, Varese, Italy
December 4, 2019

Certified by .....
Claudio Gentile
Research Scientist, Google Inc.
Thesis Supervisor

Physical Layer Identification and authentication of electronic devices

by

Gianmarco Baldini

Submitted to the Department of Computer Science of University of Insubria,
Varese, Italy
on December 4, 2019, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

In this thesis, I have investigated the problem of identification and authentication of electronic devices through their physical layer intrinsic features or fingerprints. The concept is that small differences in the electronic components of electronic devices (e.g., smartphone) leave small but significant traces in the digital output generated by the electronic device (e.g., digital image or radio frequency emissions). Then, an analysis of the digital output provides the capability to identify and/or authenticate an electronic device from its digital output with a degree of accuracy, which is based on various factors including environmental effects. This research area has become more prominent in recent times due to the increasing computing power available for signal processing and analysis, which allows a more efficient and accurate extraction of the fingerprints. Even if there is considerable research in this area, which has proven the concept both with theoretical analysis and experimental results, there are still many aspects to be investigated both for the different types of electronic devices and for the analysis of the digital output through signal processing and machine learning techniques. The PhD activities have investigated various novel aspects in comparison to the existing literature. This thesis describes most of the results and describes the novelty in comparison to previous research literature. Three specific use cases were considered: identification of wireless devices, microphones and magnetometers.

Thesis Supervisor: Claudio Gentile
Title: Research Scientist, Google Inc.

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor Prof. Claudio Gentile for the superlative support of my Ph.D study and the related research activities. In particular, I am thankful for for his patience, motivation, and deep knowledge of machine learning. His guidance helped me in all the time of my PhD research activities, in providing guidance on drafting the publications, on the errors to avoid and in writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D study. Then, I would like to express my sincere gratitude to the academic staff of University of Insubria and in particular to rettore Prof. Alberto Coen Parisini and PhD coordinators Prof. Marco Donatelli and Prof. Barbara Carminati for their patience and understanding during my PhD studies. A sincere thanks also to the staff providing the PhD courses and in particular to Prof. Alberto Trombetta. I am also thankful to the management of the Joint Research Centre of Ispra to approve my PhD studies at my relatively old age. Last, but not least my deep gratitude to my family for the patience and lovely support during these three years of nightly and weekend efforts to pursue my research activities.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Identification and authentication are two important security functions. Identification is the capability to uniquely identify an entity (e.g., user, system or application) among other entities. Authentication is the capability to prove that an entity (e.g., user, system or application) is what it claims to be. Identification and authentication is extensively used in ensuring the security of Information and Communication Technologies (ICT) for the access and provision of services. For example, it is critical to distinguish between legitimate users (who are paying for a service) from other users (who may use the same services in an illegal way). Identification and authentication can be based on three main factors: information that an entity knows (such as a password), something that entity has (a smartcard) or something the entity is (biometric features). Identification and authentication can be based on each of these elements or a combination of these elements in the so-called multi-factor authentication, which is usually stronger than identification/authentication based on a single element.

Each of these elements has its own advantages/disadvantages as it is well known in literature [1],[2]:

- *Information that an entity knows* has the disadvantage that this information can be stolen from the entity or the entity can lose this information (e.g., forgetting it). In addition, many systems require to re-create information when the identification/authentication system is upgraded or periodically to increase security (e.g., re-issuing new passwords). The advantage is that this form of authentication has a relatively simple implementation and usability.

- *Something that an entity has* can also be stolen and used by another entity. There is also the problem of updating/re-issuing the âĂIJsomethingâĂİ when the identification/authentication system is upgraded. The advantage is that this form of authentication has a relatively simple implementation if it requires the generation and distribution of hardware or digital components (something to have), which can be cumbersome in some contexts (e.g., entities which do not have connectivity).

- *Something that the entity is* has the advantage that it cannot be lost as it is an

intrinsic feature of the entity (e.g., biometric features like a fingerprint). The disadvantage is that the extraction of the âĂIJsomethingâĂİ or its features can be a complex operation, which may require sophisticated equipment (e.g., DNA analysis)

A specific context is the identification/authentication of electronic devices, where an electronic device rather than a person must be identified or authenticated. This context has become increasingly important with Machine to (2) Machine (M2M) communication and provision of services without human intervention as the electronic device must be able to be identified and authenticated by the rest of ICT system or devices with which they must interact. The conventional method to identify/authenticate electronic device is through cryptographic means where the electronic device receives and stores cryptographic material (e.g., a private key) which is used to perform various security functions. On the other side, this approach has the disadvantages described above: the private key must be protected from being stolen and it is necessary to update it and obtain the associated public key when there is a change in the overall cryptographic system (e.g., because the algorithm generating the key is changed), which requires secure connectivity. There are also other potential threats to cryptographic systems described in the literature (e.g., eavesdropping) which are out of scope of this thesis.

The goal of this PhD thesis was to investigate an alternative way to identify and authenticate electronic devices using non-cryptographic means, which is based on the intrinsic features of the electronic device itself (something that the entity is).

The concept is the electronic devices have very small differences in their hardware components due to: a) the manufacturing process or b) the different composition of materials, which can be exploited to uniquely identify the electronic device. These differences are usually within the constraints of the standards on which the electronic devices are designed and they are not relevant enough to hamper the correct provision of the services by the electronic devices, but they can be used to uniquely identify the device. Some researchers have called these differences âĂŸfingerprintsâĂŹ like the human fingerprints or Radio Frequency DNA (RF-DNA) [3] in case of wireless

devices, which produce radio frequency emissions to transmit information on the basis of a wireless standard. Fingerprints can be identified by the digital output generated by the electronic device either when the device is providing services (e.g., radio frequency emission of a UMTS mobile phone when transmitting), or on the basis of a specific input (e.g., sensor patter noise in the pictures taken by a camera). The characterization of the digital output and the creation of the fingerprints is usually implemented through statistical analysis of the digital output itself. While there has been a considerable body of research in recent years on fingerprinting of electronic devices, there are still many research areas to further investigate and explore, which was the main focus of the doctoral research activities described in this thesis.

The fingerprints can be used in various applications, which are further described in Section 1.2:

1. multi-factor authentication where the fingerprints can be combined with conventional authentication methods based on cryptography,

2. fight against counterfeiting of electronic devices where proper and counterfeit devices can be distinguished on the basis of the fingerprints,

3. forensics analysis where the fingerprints can be used to match evidence from a crime scene or crime activity,

4. quality control of electronic devices to check if they deviate significantly from a blueprint.

The study of fingerprinting of electronic devices may require various capabilities and can involve many different areas of research:

- Knowledge of electronic devices to understand what type of meaningful digital output can be collected or generated on the basis of specific stimuli,

- signal processing to analyze and evaluate the digital output of an electronic device and extract the fingerprints,

- machine learning algorithms to classify the fingerprints and use them to distinguish the electronic devices,

- modeling of environmental effects, which may impact the fingerprint identification. For example, wireless propagation models when the fingerprints are based on radio frequency digital output because wireless propagation phenomena like fading or attenuation can impact the classification accuracy.

All these aspects are analyzed in detail in the following sections.

1.1 Problem Definition, operational requirements and metrics

1.1.1 Problem definition

The problem definition is how to identify and authenticate electronic devices on the basis of their digital output. As reported in the research literature, it is proven that the digital output (e.g., images from a camera or radio frequency signal of a wireless device when transmitting) from an electronic device includes specific characteristics (i.e., the fingerprints) related to the physical structure and materials of the device. As described in various surveys [4], [5], each electronic device including cameras, radio frequency communication systems, microphones, Inertial Measurement Unit (IMU) can be identified on the basis of the fingerprints. See figure 1-1 for a pictorial description of this concept. The fingerprints are generated by the small differences in the hardware components (e.g., filters, amplifiers, processors, transducers, generators and so on), how they are connected and the materials composing the components. For some sensors, the signal processing software can also generate fingerprints (see [6] for an example on how the processing software can impact fingerprinting in Global Navigation Satellite Systems (GNSS) receivers). Then, it is more challenging from a research point of view to determine the fingerprints related to the hardware components, which is the focus of this doctoral thesis.

As each electronic device has completely different hardware and software components, the process of extraction and analysis of the fingerprints from each type of device or component can be quite different. In theory, it would be possible to model the behavior, bias and tolerances of each specific electronic component (e.g., filter) and there is an extensive literature for each component which can be used for this purpose. On the other side, the complexity of electronic devices is that such a modeling approach has been demonstrated its validity for the creation of the fingerprints only in very specific cases and sensors [5], [4]. In the large majority of cases, an empirical approach must be adopted for the extraction of the fingerprints. Section 2 on the state of art provides more details on this specific aspect.

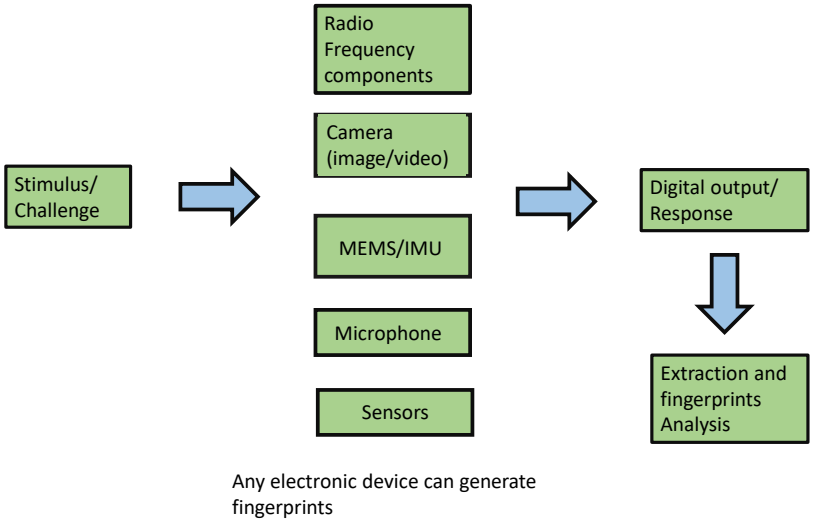


Figure 1-1: Any electronic device in a mobile phone can generate fingerprints

1.1.2 Operational Requirements

To be of practical use, the extraction of the fingerprints must fulfill a set of operational requirements and metrics, which are described in the following bullet list. These requirements/metrics are referred in the following subsections of this thesis.

1. The fingerprints must support a high classification (identification and/or au-

thentication) accuracy of the electronic devices,

2. The computing time for the generation and testing of the fingerprints must be limited,

3. Repeatability of the fingerprints: the fingerprints must be stable across different digital recordings taken from the electronic device in different times and environmental settings,

4. The classification accuracy must be robust against environmental effects including the ones introduced by the devices used to collect the fingerprints.

The requirements are often linked among themselves, thus generating trade-offs, which are dependent on the type of electronic device. For example, the computing time and the accuracy is often a trade-off as fingerprinting techniques which provide an higher accuracy often require more computing time or memory. Research literature has investigated various techniques to support the validation of the requirements identified above. During the research activities of the PhD thesis, a number of novel techniques to improve the state of art in comparison to existing literature have been investigated and published in peer-review papers. The results are described more in detail in the following Sections of this thesis (see Sections 3, 4 and 5).

1.1.3 Classification Metrics

Note: The thesis is focused exclusively on supervisioned learning in a closed labeled set. Then, the metrics for un-supervisioned learning are not discussed in this Section or the rest of thesis.

In this section we describe the classification metrics used in the rest of the thesis.

If A given class is taken as a reference class (usually called the "positive" class), then the following quantities are computed:

- T_p is the number of *true positive* matches, where the machine learning algorithm has correctly identified a sample (e.g., a collected Radio Frequency (RF) signal in our context) as belonging to the positive class;

- T_n is the number of *true negative* matches, where the machine learning algorithm has correctly identified a sample as *not* belonging to the positive class;

- F_p is the number of *false positive* matches, where the machine learning algorithm has mistakingly identified a sample as belonging to the positive class;

- F_n is the number of *false negative* matches, where the machine learning algorithm has mistakingly identified a sample as *not* belonging to the positive class.

The definitions above can be extended to the case of multiclass classification to provide the following classification metrics.

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}, \tag{1.1}$$

$$Recall = \frac{T_p}{T_p + F_n}, \tag{1.2}$$

$$FalsePositiveRate = \frac{F_p}{F_p + F_n}, \tag{1.3}$$

$$Precision = \frac{T_p}{T_p + F_p}, \tag{1.4}$$

$$F1score = 2 * Precision * Recall/Precision + Recall, \tag{1.5}$$

Another set of metrics is related to binary classification. The Receiver Operative Characteristics (ROC) is generated by plotting the True Positive Rate (TPR) (from equation 1.2) vs. False Positive Rate (FPR) (from equation 1.3) in a binary classifier system as its discrimination threshold is varied on the basis of the posterior probabilities (scores).

The Equal Error Rate (EER) corresponds to the condition on the ROC curve where TPR and FPR are equal. In this thesis, the value of the EER is calculated for the X-axis (i.e., the FPR). This metric is frequently used as a summary statistic

to compare the performance of various classification systems. In general, the lower is EER the better is the classification performance.

Finally, the confusion matrix is also used to show the results of the identification process. In the confusion matrix, each column of the matrix represents a predicted class while each row represents the actual class or viceversa as described in the specific figure.

1.2 Applications areas

1.2.1 Multi-factor authentication coupled with conventional cryptographic authentication

In this application area, the fingerprints complement cryptographic means of authentication: in addition to the authentication using keys (either with symmetric or asymmetric cryptography) the fingerprints can be used to add another level of security. If cryptographic keys are lost or compromised, fingerprints can be still used to authenticate the device.

1.2.2 Alternative to cryptographic authentication when it is not deployed or feasible because the key management process is expensive or complex

In some contexts, the deployment of a cryptographic infrastructure with deployment and installation processes (e.g., Public Key Infrastructure) is not a viable solution because of technical or economic factors. In addition, there are some scenarios where different systems need to coexist but they are not based on the same cryptographic infrastructure or not all the systems do have a cryptographic authentication system. One example is the detection of radio frequency emitters in a cognitive based scenario [7].

1.2.3 Fight against distribution of counterfeit products

Counterfeiting of electronic devices has an impact of billions of Euro on the economy of member states in Europe. Many techniques for the detection and identification of counterfeit electronic devices are destructive: the electronic device must be opened and disassembled to understand if the device is counterfeit or it contains counterfeit components. Instead, the analysis of the digital output can be used to detect counterfeit device and components without dissembling the device because counterfeit components have lower quality components with different fingerprints from proper components. As described in a recent study [8], counterfeit parts of an electronic device, including the integrated circuits (ICs), are not only clones, but also recycled, overproduced or (different) remarked components. This means that, in many cases, they are still components coming from the production line or from its suppliers but with lower quality, possibly posing problems of reliability and security of the devices, as already reported even in the automotive, aviation and military industry.

1.2.4 Detection of tampering of electronic devices

Electronic devices can be tampered for a number of reasons including infringement to regulations. For example, an electronic device (e.g., sensor) used to implement a regulation can be tampered to report false readings, which provide an economic benefit to the malicious user. The analysis of the digital output can be used to detect tampering in the sensor or the replacement of the entire sensor.

1.2.5 Quality Test

Fingerprints can be used for non-destructive testing as only the digital output can be analyzed but the testing system does not need to be disassembled. An example is [9] for amplifier acceptance testing.

1.2.6 Monitoring for aging effects

This application uses the fingerprints to detect aging effects in an electronic device, which can hamper its performance. The analysis of the digital output can show the ageing of the fingerprints, which in turn can give an indication that the electronic device can drift away from its functional operating characteristics

1.2.7 Forensics

This is one of the primary applications of the fingerprints concept. As the digital output of the electronic device contains the fingerprints of the device itself, it is possible to identify the source of a digital recordings, which is an important function in forensics studies. For example, a sound recording uploaded to the web can be used to track down the source of the recording (i.e., the microphone and the associated mobile phone).

1.2.8 Surveillance. Detection and identification of intruders in a device network

In a closed and monitored ICT system, it is possible to use the fingerprint concept to identify and authenticate devices on the basis of their fingerprints and then detect a foreign device, which does not belong to the authenticated set. For example, the authentication system of a wireless network can record the fingerprints of all the connected devices. When a malicious device is used to replace an already authenticated device, a check of the fingerprints can be used to detect the replacement.

1.2.9 Privacy attack when the fingerprints are used to track an electronic device (mobile phone) and then the person

This is negative aspect on the use of fingerprints, where they are used to track the device on the basis of the digital recordings. By tracking the device, it is possible to track also the user of the device, thus creating a privacy threat. For example, a

wireless device can be tracked in a multitude of other devices even if the user adopts mitigation privacy techniques like pseudonyms [10].

1.2.10 Identify an electronic device in a network, which is supposed to be covert for security reasons

This is also a negative aspect on the use of fingerprints, which can impact law enforcer operations. A covert device (e.g., a wireless microphone) could be installed by law enforcers for surveillance operations. Even if the wireless microphone is masqueraded as an existing device, its fingerprints will be different and it can be identified.

1.3 Structure of the thesis

The structure of the thesis is following:

- Section State of Art 2 provides an overview of the state of art on the identification and authentication of electronic devices (wireless devices, microphones and magnetometers) in the research literature. The focus is on the three main categories of electronic devices as these are the ones investigated in the PhD use cases.

- Section Use Case for Radio Frequency Physical Layer authentication 3 describes the results related to the fingerprinting of radio frequency wireless devices. This section also introduces the machine learning algorithms, which are used in the other use cases.

- Section Use Case for microphone identification 4 describes the results related to the fingerprinting of microphones.

- Section Use Case for magnetometer identification 5 describes the results related to the fingerprinting of mobile phones through magnetometers.

- Section Conclusions 6 provides the conclusions of the PhD thesis.

Chapter 2

State of Art on electronic device identification

2.1 Overview

This section provides an overview of the state of art on the identification and authentication of electronic devices for three specific types of devices: radio frequency wireless devices, microphones and magnetometers. In each of these areas, it is provided a high level description of the novel results produced by the activities in the PhD thesis in comparison to literature.

2.2 Radio Frequency components

The concept to identify wireless devices through the specific characteristics of their RF emissions has been widely investigated by the research community in recent years. As described in the Introduction section 1, the concept is based on physical differences in the hardware components of the wireless communication front end, which generates small but significant and reproducible features in the RF emissions. Even if such differences are usually within the boundaries defined by the wireless standard (e.g., 802.11) and they do not usually impact the wireless communication performance, they can be significant enough to identify or authenticate the wireless devices. Different terms are used in literature for the same concept: it is called Specific Emitter Identification (SEI) in [11], RAI in [12] and RF-DNA in [13]. This thesis uses the generic term PLIA and it is used in a generic way for the other uses cases as well.

The main challenge in this use case is to identify and extract the specific features from the digitized RF emissions and select the ones providing the optimal identification or authentication accuracy. Different wireless standards may have slightly different features, even if common elements are present in most of the wireless standards (especially if they are based on a similar design like 802.11a for WiFi and 802.11p for DSRC). In literature, this concept was applied to many different wireless standards including WiFi [14], ZigBee [15], WiMAX [13], GSM [16], DSRC at 5.9 GHz [10] and others.

Many wireless standards (e.g., GSM, UMTS, WiFi, WiMAX, DSRC) share a

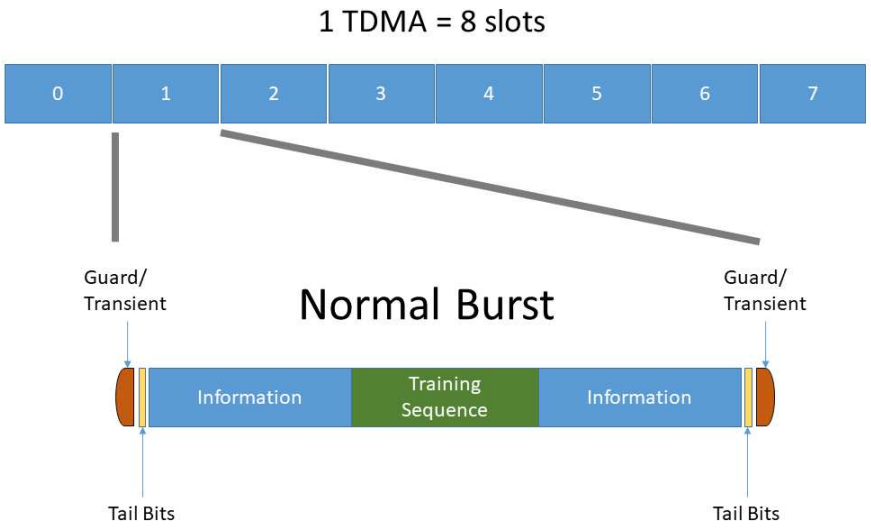


Figure 2-1: GSM Physical Layer representation in Frame, Slots and Burst structure

common aspect. The content transmitted between two or more wireless devices is transmitted in *bursts*, which are repeated many times in a second. Regardless of the content transported by the burst (e.g., voice, data) each burst includes also the features specific to the device. Then, it is possible to use the bursts to identify the device. The advantage in this use case is that even a short time (e.g., seconds) used to collect the signal can contain many bursts (the exact number depends on the wireless standard), which can be used to create a large sampling set for the machine learning classification. Usually a burst is composed by specific segments. To simplify, we show an example for the GSM wireless standard. Even if it is an old wireless standard, it represents quite well the concept of the burst structure, which can be present in more sophisticated wireless communication standards as well.

The burst in the GSM standard is used to transport the information (e.g., voice, data). There are also signalling GSM bursts used for the management functions of the GSM network, but they are not used in this use case, because they are transmitted with less frequency than the information (i.e., voice or data) bursts. One potential issue for PLIA in this use case, is the presence of content information, which can introduce a bias in the classification process. The reason is that the classification

process could classify the content (e.g., what the person is saying at the phone) rather than the GSM phone itself, which is the main objective.

Two possible approaches are possible:

- The specific parts of the content (information field in Figure 2-1) should be removed from the collected signal in space. Only the invariant parts like the midamble and the guards/transients portions of the burst should be used as they are invariant to content.

- The system is configured to send only test data: a specific sequence of bits, which is always the same in all the bursts.

While, the first option seems complex, it is actually quite straightforward to implement as the burst definition is quite precise in the wireless standards technical specifications and it can be extracted easily from the burst. The second option requires the possibility to control the configuration of the wireless communication systems, which is not always possible to achieve. This option would be easier to implement in local Wireless Local Area Network (WLAN) (e.g., WiFi) or specific Internet of Things (IoT) wireless systems where the researcher can have access to the configuration setup.

An example of the first option is shown in Figure 2-2, where the content information has been extracted from the bursts generated by 12 GSM mobile phones (one burst from each phone).

From the Figure, it is visually clear that each phone introduces specific fingerprints in the burst and in particular in the transient phase rather than the mid-amble. These specific fingerprints are highlighted by the circles in Figure 2-2. The main reason is that the transient part of the burst is where the radio frequency components of the wireless communication device are working out of the normal operative range and the non-linearities of the components (which are usually correlated to the fingerprints) appear more frequently in the burst [4] and [17].

In the use case of radio frequency, there is an interesting trade-off between the use of the different parts of the bursts shown in Figure 2-2. It is noted in [18] that

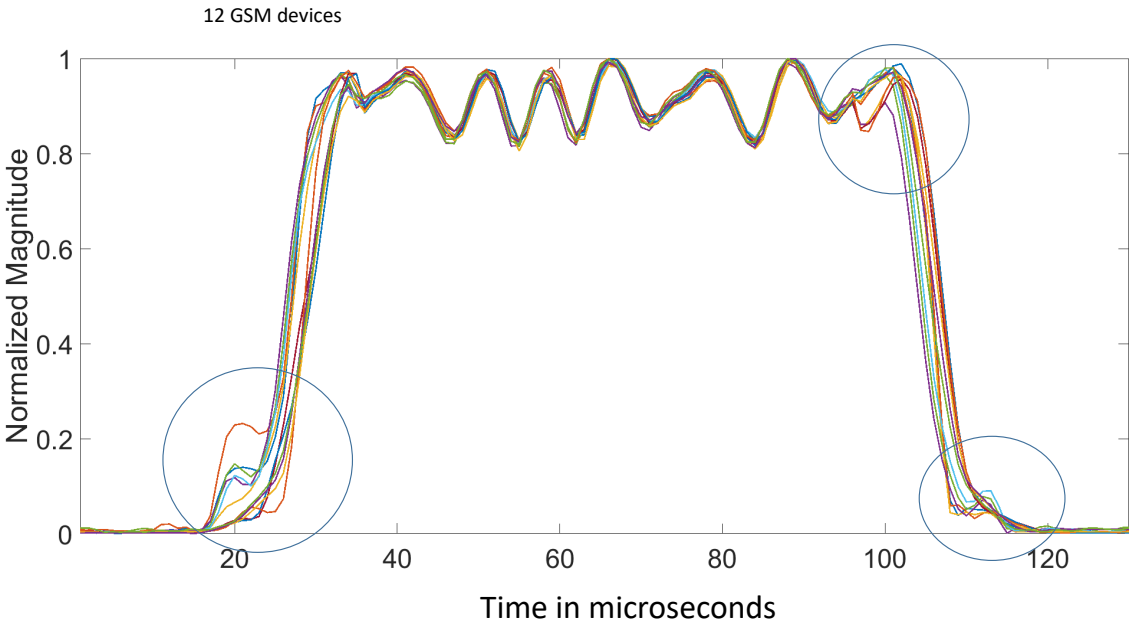


Figure 2-2: Example of a burst signal collected from 12 GSM phones

the capture and digitization of the transient signal requires very high oversampling rates and sophisticated and expensive receiver architectures. In fact, thanks to the increasing computing power of the Analog Digital Converter (ADC) circuitry, this problem has somewhat become less important today. In contrast to the transient signal, the steady-state signal portion can be much longer than the transient part of the burst, thus providing more information for classification purposes. Depending on the wireless standard, the receiver used to collect the signal in space and the shape of the burst, one of the options can be used. Another factor to take in consideration is that signal processing techniques also make assumptions on the stationarity or non-linearity of the signal to be analyzed. As the transients are strongly non-linear, this aspect imposes an additional constraint on the methodology to process the signals. For this reason, many authors prefer to use only the non-transient part of the signal. For example, the authors in [11] have used the Hilbert Huang Transform (HHT), which is a combination of EMD and Hilbert spectrum analysis. The former is a sifting process to decompose any signal into an Intrinsic Mode Function (IMF), while the latter offers the time-frequency distribution, referred to as the Hilbert spectrum, by performing the Hilbert transform on each IMF. In [11], the authors have demonstrated

an improved performance of the application of HHT in comparison to 1D Time and Frequency domains. Another example is the application of Variational Mode Decomposition (VMD), which is an evolution of Empirical Mode Decomposition (EMD) for the classification of Internet of Things and DSRC wireless communication devices using only the non-transient parts in [19].

Since a digitized RF signal is indeed a time series, various strategies have been used to implement PLIA and many techniques are based on research of time series classification. The time series obtained for different wireless devices can be compared either directly (e.g., euclidean distance and KNN classifiers) or using other approaches. A common strategy is to extract statistical features from the RF signal and then use a machine learning algorithm to classify the obtained set of features and correlate them to the identity of the wireless device. There is an extensive literature on the selection of different statistical features for PLIA including variance, entropy, skewness, kurtosis and others [13], [14]. The extraction of statistical features from the RF signal has the benefit of the dimensionality reduction, which improves the classification time but it may decrease the accuracy if the statistical features are not selected properly.

The analysis of the state of art has shown that three main areas have received limited attention. The first area is related to the application of Deep Learning to PLIA. The second area is the analysis of the impact of bias introduced by the signal receiver (used to collect and process the signal from the devices to be identified), which impacts the portability of the fingerprints from one receiver to another. The third area is the impact of the IQ imbalances of the receiver, which can be caused by aging effects, on the PLIA performance.

2.2.1 Application of CNN to PLIA

Regarding the first area, Deep Learning has been used successfully in recent times in different domains, it was not applied to this context at the time of the start of the PhD work. The only known result for the application of Deep Learning to PLIA

is in [20], where a Deep Convolution Neural Network (CNN) was applied to the identification of 7 WiFi wireless devices. The authors in [20] apply the CNN to the time representation (i.e., the original baseband representation) of the digitized signal in space collected from the transmitting devices by treating the real and imaginary parts as two separate input channels.

On the other side, CNN has been historically applied to images (see one of the first applications of CNN in 1995 to biomedical images in [21]). Then a logical step was to investigate the application of methods to transform the original time series of the signal collected from the wireless devices to images. Such transform has already shown its advantages in combination to non-Deep Learning machine learning algorithms in various papers like [22] and [23] where a Gabor transform was used to generate a time-frequency representation of the signal on which a feature selection process was implemented. In these papers, the application of machine learning algorithms to the time-frequency based approach provided substantial better results than the direct application of the same algorithms to the time representation.

Different transforms and imaging techniques were experimented and the results are described in section 3 and they were also published in [24]. One category of imaging techniques was based on the application of Time Frequency Analysis (TFA) to the original time series thus obtaining a bi-dimensional representation (time and frequency), which is used as an image to the input of the CNN. Another category of imaging techniques is to use the concept of Recurrence Plots and similar techniques as described in the subsequent section of this thesis. The application of these new transforms was novel in this context and the results show a significant improvement in comparison to the application of non-Deep Learning machine learning algorithms.

Novel advancement in comparison to literature: the author of the thesis has applied imaging techniques including time-frequency transforms in combination to CNN to the problem of PLIA, which was never attempted before. The description of the application of the approach and the results are described in section 3. The results of the investigation on this approach have been published in [24].

2.2.2 Problem of fingerprint portability

Regarding the second area related to the bias introduced by the receiver (i.e., the device used to collect the signal in space from the wireless devices to be classified), preliminary studies have shown that difference in the receivers have an impact on the *portability* of the fingerprints collected from one system to another, which poses a critical challenge to the practical implementation of PLIA because fingerprints collected with one system could be only used by the same system. The great majority of the papers addressing PLIA in radio frequency (i.e., RF-DNA, Emitter Identification, radiometric identification) use the same RF receiver to collect the signal in space both for training and testing. Only few papers have investigated the issue of portability as described in the following paragraph.

In [25], the authors analyze the problem that the RF fingerprints generated through one RF receiver cannot be used to identify the same wireless device by means of another RF receiver. This phenomenon is due to the fact that each RF receiver introduces a bias which degrades the RF fingerprints of the emitting device. As a consequence, RF emissions of the same wireless device taken from different RF receivers will actually generate different fingerprints for the same wireless devices. While the authors concluded in [25] that fingerprints are not portable from one receiver to another, they did not propose a solution to this problem. In a similar way, the authors in [26] have investigated the portability between high-end and low-end receivers and they have found out that the portability problem is also present because of the bias introduced in the low-end receivers. The authors also did not propose a solution to this problem.

As this is a critical problem for the practical application of fingerprints and no solution was found at the time of the PhD activities, it was decided to identify and find potential solutions to address the portability issue.

Novel advancement in comparison to literature: the author of the thesis has defined and proven with experimental data a new method to mitigate and resolve the

problem of portability of the fingerprints. The description of the application of the approach and the results are described in section 3. The results of the investigation on this approach have been published in [27].

2.2.3 Impact of IQ imbalance to classification accuracy

The third area is related to a specific aspect of RF receivers, which can hamper the practical application of the PLIA. This is based on the IQ imbalance of a RF wireless device, which can grow considerably with the aging of the device. The focus is on the IQ imbalance of the RF wireless receiver used to collect the RF signals in space.

The IQ imbalance is a phenomenon appearing in Direct Down Conversion (DDC) RF receivers, which transforms the RF-signal directly down to base-band. As described in [28], due to temperature dependencies, production imperfections and aging, the analog components in the I-path and Q-path can not be perfectly matched. The IQ-imbalance is a serious issue, which can degrade the receiver performance and impact the PLIA because the bias introduced by the IQ imbalance can degrade the classification performance. In recent literature, IQ imbalances have been used in relation to PLIA only as an instrument to identify and authenticate wireless devices rather than to assess the impact on identification. In [29] the authors use the IQ imbalances as the specific features of the wireless devices to be identified, rather than as a negative disturbance introduced by the receiver. In a similar way, the authors of [30] have used IQ imbalances to identify wireless transmitter, and extensive investigation has been performed using a CNN. The research angle in this PhD thesis is complementary to papers cited above.

Novel advancement in comparison to literature: the author of the thesis has investigated the impact of the IQ imbalance in the radio frequency receiver to the classification accuracy of wireless devices. The description of the application of the approach and the results are described in section 3. The results of the investigation on this approach have been published in [31].

2.3 Microphones

In recent years, the problem of how to identify the source of an audio recording has been addressed, with a considerable attention to the mobile phone as recording system. The authors in [32] proposed a pioneering work in this field, where a set of audio steganalysis-based features to cluster both the microphone and the environment have been used. This work has been extended in [33], wherein a combination of statistical features and unweighted information fusion have been employed to improve the accuracy in the classification. As for other cases of PLIA, we make a distinction between inter-model and intra-model classification. In inter-model classification, the devices to be identified belong to different models and brands (e.g., one HTC One mobile phone and one Samsung S5 mobile phone). In intra-model classification, the devices to be identified are of the same model (e.g., all of type HTC One mobile phone). In general, intra-model classification is much more difficult than inter-model classification because the devices are built with the same component models (e.g., filter, amplifiers) and they have the same design, while different models are usually built with different components and different design, thus improving the discriminating capability.

In most of the earliest works only the inter-model classification has been considered. More recently, in [34] and [35] the authors also addressed the intra-model classification task through the application of K Nearest Neighbor (KNN) and Gaussian mixture model (GMM). Furthermore, a comparison of various features showed that Mel-Frequency Cepstrum Coefficient (MFCC) provided the best accuracy results [35].

In [36] the Power Spectrum Density (PSD) of speech-free audio recordings is used to train a Support Vector Machine (SVM) classifier for microphone identification of a mobile phone. The speech-free audio recordings are detected using Audacity software and the PSD is calculated using a periodogram. The authors in [37], employed MFCC

coefficients of the non-speech segments of the voice recordings in combination with SVM and GMMs to classify the microphone. The method exhibited promising results but it also showed substantial sensitivity to additive noise. In the paper [38] a speech recording it is used for device recognition based on sparse representation.

Alternatively, in more recent works the microphones were stimulated using non-voice recordings such as in [39] and [40]. In particular, the authors in [39] found that the frequency response curve extracted from sample recordings can be a robust fingerprint to characterize the recording device. The application of SVM is proposed to perform the classification over 31 mobile phones. In [40], the authors proposed a speaker-to-microphone authentication protocol by leveraging the frequency response of a speaker and a microphone from two IoT wireless devices as the acoustic hardware fingerprint.

The application of Deep Learning to the identification and authentication of microphones was not attempted in literature on non-voice recordings.

Novel advancement in comparison to literature: the author of this thesis has investigated the application of CNN to the identification and authentication of a large set of *non-voice* recordings based on 34 mobile phones where the phones were stimulated with a repetition of specific sound stimuli. The application of CNN for microphone identification using non-voice recordings have not been applied in literature yet. The description of the application of the approach and the results are described in section 4. The results of the investigation on this approach have been published in [41].

2.4 Magnetometers

As described in the introduction, the identification of mobile phones through their built-in components has been extensively investigated by researchers for different elec-

tronic components: the internal digital camera [42], the RF transmission components for various communication standards (e.g., GSM, WiFi) as described in [43],[11], the microphones [44], [45] and the accelerometers [46] and [47].

In recent times, the mobile phones have been equipped with magnetometers of increasing recording capability. Magnetometers are mainly used as a compass to measure the direction of an ambient magnetic field, in this case, the Earth's magnetic field. Then, magnetometers in combination with other sensors can be used to improve the navigation and localization both outdoor and indoor as buildings do also contain ferromagnetic material, which is sensed by the magnetometer [48]. On the other side, the magnetometer has also specific characteristics which can be used to identify and authenticate the mobile phone where it is installed. From this point of view, the research activity on this field is quite limited; presumably because mass market mobile phones with magnetometers have been recently introduced but also because magnetometer fingerprints pose specific challenges and they are more difficult to extract in comparison to the fingerprints of other sensors.

The only paper at the time the PhD research activities started in this field (i.e., 2016) was the study performed by the authors in [49] where the authors used the magnetometers to pair two mobile phones among themselves. As in the similar use cases described in this PhD thesis, the method exploited the fingerprints in the magnetometers for identification and authentication. The authors used a very limited set of magnetometers and for the specific objective of pairing the mobile phones. Instead, the focus of the PhD research was to focus on the identification and authentication of the magnetometer and the mobile phone on its own with a specific stimulus.

Novel advancement in comparison to literature: the author of this thesis has investigated the identification and authentication of mobile phones using the built-in magnetometers, when they are stimulated by a specific and repeatable magnetic field. The description of the application of the approach and the results are described in section 5. The results of the investigation on this approach have been published in [50].

Note that after the publication of [50], other attempts to identify and authenticate the magnetometers have also been proposed in the literature. For example, the author of this thesis together with other researchers have also used another way to stimulate the magnetometers in [51] while other authors have used magnetometers in combination with the other IMU sensors in [52] but the earth magnetic field was used (which may be not be applicable for supervised classification because of the absence of golden truth absolute value of Earth magnetic flux density behaviour in the time of acquisition of time series) instead of a specific and repeatable pattern of magnetic flux density. Even more recently, the authors in [53] have used a similar approach to ours (citing [50]), which shows that the interest in this technique is increasing.

Chapter 3

Case Study of Radio Frequency Physical Layer authentication

3.1 Overview

This use cases focuses on the identification and authentication of wireless devices on the basis of the radio frequency signal transmitted by the device themselves while transmitting. As described before in section 2, the identification is possible because of the small differences in the RF front end of the wireless devices (e.g., amplifiers, filters) which are inserted in the signal transmitted in space. In comparison to other uses cases, wireless propagation effects like noise, fading or the bias introduced by the RF systems (e.g., a wireless receiver) used to collect the signal transmitted by the wireless devices can negatively impact the classification performance. The PhD research activities focused on the novel aspects in comparison to literature identified in section 2.

3.2 Materials

Two sets of wireless devices were used in the research activities and publications: IoT wireless devices and GSM mobile phones respectively described in section 3.2.2 and section 3.2.3.

In addition to the wireless devices to be identified, receivers were used to collect the RF signal transmitted by the devices as described in section 3.2.1.

3.2.1 Receivers used to collect the RF signal

The RF signals transmitted by the wireless devices are collected using a low cost Universal Software Radio Peripheral (USRP) Software Defined Radio (SDR) receiver of type N210, equipped with a XCVR2450 front end frequency locked to a Global Positioning Systems (GPS) disciplined (the SDR receiver was equipped with a u-blox NEO6Q GPS receiver) 10 MHZ reference to ensure frequency stability and repeatability in the collection of RF observables [54]. The RF signals were sampled by the SDR with a sampling rate of 5 MHz/sec.

For the study on the portability of the fingerprints on the IoT devices, two ad-

ditional USRP SDR receivers were used to compare the fingerprints of each specific wireless device among three different receivers (see the definition of IoT-DS3 below). The specific details on the test bed and the signal receivers are provided in section 3.6.2.

3.2.2 IoT data sets

The first set of composed by eleven(11) nRF24LU1+ devices. The number of devices used in the test bed is comparable with the number of devices used in literature: 9 ZigBee devices in [55], 8 Noise radars in [56] and 10 Zigbee devices in [57]. This wireless device is an Ultra Low Power (ULP) device transmitting at 2.4GHz in the Industrial, Scientific and Medical (ISM) band. It includes a 2.4GHz RF transceiver core, 8-bit CPU, full-speed USB 2.0 device controller, and embedded Flash memory. These wireless devices have been programmed to build a MySensors network. MySensors is a free and open source DIY (do-it yourself) software framework for wireless IoT devices allowing devices to communicate using radio transmitters. This data set was created with the wireless devices transmitting in test mode and repeating the same data sequence. As a consequence the entire burst could be used for classification because the content is always the same (the test sequence). Three different sets of data were collected with these devices:

1. One set of data was created with only 9 devices with smaller bursts (burst length 512). This data set will be called *IoT-DS1* in the rest of this thesis.

2. The second set of data was based on all the 11 devices and a longer bursts (burst length 7104). This data set will be called *IoT-DS2* in the rest of this thesis.

3. The third set of data was based on the collection of 11 devices from three different radio frequency receivers. This data set will be called *IoT-DS3* in the rest of this thesis.

3.2.3 GSM data set

The second set is composed by the 12 GSM mobile phones. In this case, the GSM wireless standard was used in normal communication mode with two persons talking at the phone. Then, the content (e.g., voice) must be removed from the burst before proceeding to the classification. This data set is called *GSM-DS4* in the rest of this thesis.

3.3 Methodology

The overall methodology workflow is described in Figure 3-1. This methodology is generic for the different research activities and the published papers related to this use case.

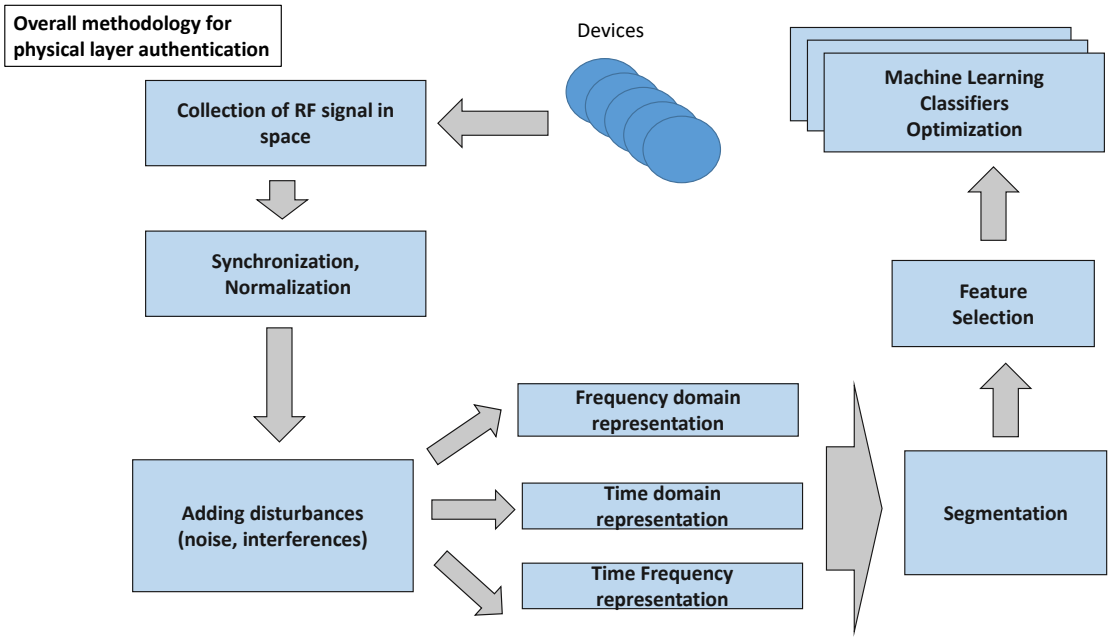


Figure 3-1: Generic Methodology

1. Collection of the RF signal from the IoT wireless devices. The RF signal from the wireless devices was collected using a SDR USRP type N200 receiver configured with a sampling rate calibrated for the specific data set (e.g., 5 MHz for the GSM data set (GSM-DS4) and 10 MHz for the three IoT data sets).

The USRP SDR receiver is equipped with a XCVR2450 front end locked to the GNSS (with an u-blox NEO6Q GPS receiver) and disciplined to 10 MHz reference clock to ensure repeatability in the collection of RF observables [54].

2. *Synchronization and normalization.* The real-valued signal samples were sampled directly in In-phase and Quadrature components (IQ) format and then synchronized and normalized offline to extract the bursts of traffic associated to each payload.

3. *Signal transformation* In this step, the signal is transformed in a way that it is more convenient for the subsequent step of device classification. Because, different transform were used in the various activities related to this PhD, the specific section 3.4 describes in detail the specific transforms.

4. *Classification with Machine Learning.* In this step, classification is performed using different machine learning algorithms based on the samples created in the previous step. Because different machine learning algorithms have been used, the specific section on machine learning 3.5 is used to describe the different algorithms.

5. *Addition of environmental effects.* In this step, environmental effects are added to the data sets to emulate difficult environmental conditions. The most common is the addition of Additive White Gaussian Noise (AWGN) as the performance of the different techniques is usually compared using decreasing values of Signal Noise Ratio (SNR) as the presence of increasing level of AWGN make the fingerprints more difficult to be distinguished. Other effects are described in the specific subsections.

3.4 Signal transforms

In this section, the different transforms used in the results section 3.6 are presented and discussed.

3.4.1 Recurrence Plots

RP is a visualization tool that aims to explore a multidimensional phase space trajectory through a 2D representation of its recurrences. The idea is to identify at which points some trajectories return to a previous. The concept of RP has been used in many different domains [58]. RP can be formulated as:

$$R_{i,j} = \theta(\epsilon - \|\overrightarrow{s}_i - \overrightarrow{s}_j\|), i, j = 1, ..., N \tag{3.1}$$

where N is the number of considered states $\overrightarrow{s}_i$, ϵ is a threshold distance, $\|.\|$ is a distance measurement (i.e., Euclidean norm in this case) and θ the Heaviside function.

3.4.2 Fast Fourier transform

The Fast Fourier Transform for a discrete signal x(k) of N samples is calculated as:

$$Y(k) = \sum_{j=1}^{N} x(j) W_N (j-1)(k-1) \tag{3.2}$$

where

$$W_N = e^{-2j\pi/N} \tag{3.3}$$

3.4.3 Short-time Fourier transform

The Short Time Fourier transform (STFT) is used to analyze how the frequency content of a nonstationary signal changes over time. Because the burst of a wireless communication system is a nonstationary signal, the STFT can be applied to the problem of RF fingerprinting.

The STFT of a signal is calculated by sliding an analysis window of length M over the signal and calculating the discrete Fourier transform of the windowed data.

We define the signal in time (the digitized collected signal in space from a RF emitter) $x(t)$ and a real even window w(x) whose Fourier Transforms are S(f) and

W(f) respectively. To obtain the localized spectrum of x(t) at time $\tau = t$, multiply the signal by the window w(x) centered at time $\tau = t$, obtaining the following equation:

$$x_w(t, \tau) = x(\tau)w(\tau - t) \tag{3.4}$$

if we apply the Fourier Transform (FT) on τ, we obtain the STFT, represented by the following equation:

$$F_x^w(t, f) = \mathcal{F}_{\tau \to f} \{x(\tau)w(\tau - t)\} \tag{3.5}$$

3.4.4 Wigner Ville transform

The Wigner-Ville distribution provides a high-resolution time-frequency representation of a signal.

For a continuous signal x(t), the Wigner-Ville distribution (or transform) is defined as:

$$WVD_x(t, f) = \int_{\infty}^{\infty} x(t + \tau/2)x * (t - \tau/2)e^{-j2\pi f}\tau d\tau \tag{3.6}$$

while for a discrete signal x(k) of N samples, the Wigner-Ville distribution is defined as:

$$WVD_x(n, k) = \sum_{m=-N}^{N} x(n + m/2)x * (n - m/2)e^{-j2\pi km/N} \tag{3.7}$$

3.4.5 Continous Wavelet Transform

The principle of wavelets is to consider an orthogonal basis of functions that is given by one shape that is scaled and translated. Wavelets are wave-like transients that can be interpreted as sinusoids of short duration and they are generally defined as:

$$\psi_{t,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{1 - \tau}{s}\right) \tag{3.8}$$

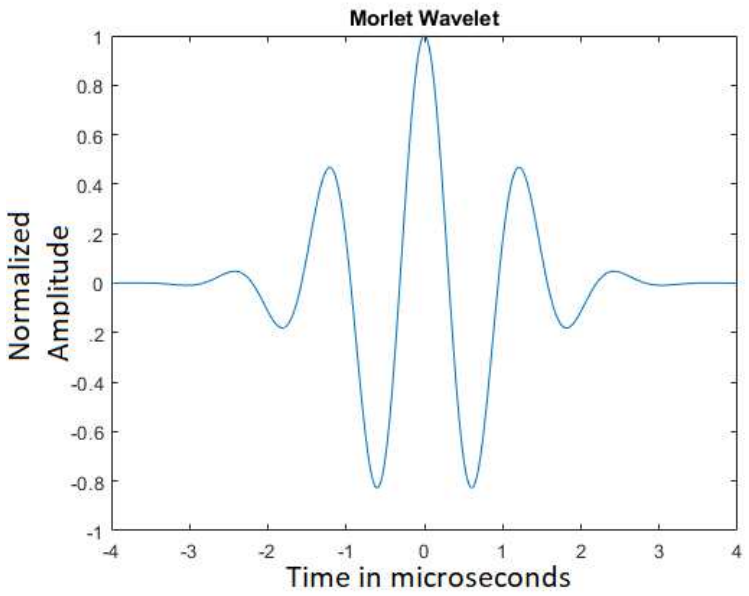


Figure 3-2: Shape of the Morlet wavelet

where τ and s are the translation and scale (dilation) parameters, respectively. The function ψ is the basis function called the mother wavelet.

For example, the Morlet wavelet is given by the following function:

$$\psi = e^{x^2/2} cos(5x) \tag{3.9}$$

A pictorial description of the Morlet wavelet is shown in figure 3-2.

The decomposition of a signal on such a basis are called wavelet transforms (WTs) and they are localized equivalents of the Fourier Transform (FTs). In fact, there is a fundamental connection between wavelet and Fourier transform: Wavelet can be seen as a Fourier transform localised in space.

The wavelet transform (WT) is defined in a similar manner except that the chosen *elementary messages* are wavelet instead of sinusoids. Using the wavelets as a basis, a time domain signal $x(t)$ can be represented as:

$$x(t) = \frac{1}{c_\psi} \int_\infty^\infty \int_0^\infty \psi_x^\psi(\tau, s) \frac{1}{\sqrt{s}} \psi \left(\frac{t-\tau}{s} \right) \frac{ds}{s^2} d\tau \tag{3.10}$$

where the coefficients $\psi_x^{\psi}(\tau, s)$ are given by:

$$\psi_x^{\psi}(\tau, s) = \int_{\infty}^{\infty} x(t) \frac{1}{\sqrt{s}} \psi^* \left(\frac{t-\tau}{s} \right) dt \tag{3.11}$$

The coefficient c_ψ is a constant, which depends on the wavelet used. $\psi_x^{\psi}(\tau, s)$ is the Continuous Wavelet Transform (CWT) of the signal x(t), which was used in this study to transform the original digitized signal collected from the radio frequency emitters.

Wavelets are notoriously good for:

- approximating non stationary signal
- data compression (image, sound, video)
- being computationally efficient

3.4.6 General Linear Chirplet Transform

The General Linear Chirplet Transform (GLCT) is a novel Time-Frequency (TF) analysis introduced in [59] and it is an extension of the Linear Chirplet Transform (LCT). GLCT has some specific features, which makes it suitable for its application in physical layer identification and authentication of IoT devices based on transients. As described in [59], GLCT is able to represent well multi-component signals with distinct non-linear features (as in the transient of a burst) and it is able to miminize the sensitiveness to noise. GLCT belongs to the family of the parameterized Time Frequency Analysis (TFA) methods, which strives to obtain a high TF resolution by identifying the inherent features (i.e., the fingerprints in this context) of the signal to be analyzed and constructing the window functions. The original Chirplet Transform (CT) was initial designed in [60] and it was refined by other authors to improve its representation of strongly non-linear signals as in this context (e.g., the transient of the bursts). In particular, the authors in [61] proposed an adaptive TFA method to select the best width and chirp rate using the maximum likelihood estimation. As described in [59], the methods described above and other methods presented in

literature have still shortcomings for the representation of non-linear signal, and the authors of [59] proposed a new TFA to address such shortcomings, which is called GLCT and it is described below. The implementation of the GLCT used in the PhD studies is based on the STFT. If $s(x)$ is the initial signal collected in space, the STFT can be represented as:

$$s(t', w) = \int w(u - t')s(u)e^{-iwu}du \tag{3.12}$$

where w(u-t') is the window to truncate the signal. to eliminate the influence of the modulated element, a demodulated operator should be introduced, which is time-variant in this context of non-linear signals. Then, Eq. (2) becomes:

$$S(t', w) = \int w(u - t')s(u)e^{-jwu} \cdot e^{-ic(t')(u-t')^2/2}du \tag{3.13}$$

where $e^{-ict'(u-t')^2/2}$ is the demodulated operator. If the demodulated operator is consistent with the modulated element, the Instantaneous Frequency (IF) of the signal will reach its maximum, which will generate a sharp TF representation with the desiderable properties for device identification (e.g., robustness to noise). Unfortunately, such operator is difficult to identify especially in non-linear signals. One simplification is to aproximate the operator in the following equation, which is the LCT:

$$S(t', w) = \int w(u - t')s(u)e^{-jwu} \cdot e^{-ic(u-t')^2/2}du \tag{3.14}$$

Unfortunately, even in this simplified form, the IF characteristics of the the signal are usually not know a priori (e.g., especially when there are multi-components like the fingerprints of the device), which makes the term $e^{-ic(u-t')^2/2}$ hard to determine. The approach proposed by the authors in [59] for the definition of the GLCT is to introduce a parameter α, which introduces a rotation in the TF plane as:

$$\alpha = \arctan\left(\frac{2 \cdot T_s}{F_s} \cdot c\right) \tag{3.15}$$

Then, Eq. (4) can be rewritten as:

$$S(t', w) = \int w(u - t')s(u)e^{-jwu} \cdot e^{-itan(\alpha)\frac{F_s}{2 \cdot T_s(u-t')^2}/2} du \tag{3.16}$$

If the parameter α has N values, the TF plan can be divided averagely in $N + 1$ sections using the following expansion:

$$\begin{aligned} \alpha = -\pi/2 + \pi/(N+1), -\pi/2 + 2 \cdot \pi/(N+1)... \\ -\pi/2 + N \cdot \pi/(N+1) \end{aligned} \tag{3.17}$$

For the purpose of using GLCT to obtain an optimal representation of the signal in the TF space for device identification, two hyperparameters must be empirically determined: the window w and the parameter N. The optimization of these two parameters for the specific case of IoT classification is described in section 3.6.1.

3.5 Machine learning

3.5.1 Common considerations for optimization

To mitigate the problem of overfitting, a K-fold approach is used in all the machine learning algorithms. In K-fold, the dataset is divided into k groups or folds of equal size. The first fold is kept for testing and the model is trained on k-1 folds. The process is repeated K times and each time a different fold or a different group of data points are used for validation. Depending on the data set, different values of K were used with (in general K=10).

The following sub-sections describe the specific machine learning algorithms used to produce the classification results. Each machine learning algorithm has a specific set of hyperparameters to tune. As recommended in literature, the hyperparameters were identified for each fold, then the final hyperparameters for classification were chosen to be the mean of the identified hyperparameters value or with a majority rule. See for example section 5 for an example of tuning of the SVM hyperparameters in the magnetometers classification use case and in particular sub section 5.4.1.

3.5.2 Classification with Convolutional Neural Networks

This section uses the notation introduced by [62]. CNN has the objective to use the spatial information between the pixels of an image, thus they are based on discrete convolution. With appropriate transform, they can also be applied to time series transformed in images, which is the case where they are used here.

We assume to have a grayscale image defined as:

$$I : 1, ..., n_1 \times 1, ..., n_2 \to C \subseteq \Re(i, j) \mapsto I_{i,j} \tag{3.18}$$

such that the image I can be represented by an array $n_1 \times n_2$.

Given a filter $K \in \Re^{2h1+1 \times 2h_2+1}$ the discrete convolution of the image I with filter K is given by:

$$(I * K)_{r,s} := \sum_{u=-h_1}^{h_1} \sum_{v=-h_2}^{h_2} K_{u,v} I_{r+u,s+v} \tag{3.19}$$

where the convolutional filter K is an array of $2h1 + 1 \times 2h_2 + 1$

There can be different types of filters. A common type of filter is the Gaussian filter defined like:

$$\left(K_{G(\sigma)}\right)_{r,s} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{r^2+s^2}{2\sigma^2}} \tag{3.20}$$

where σ is the standard deviation of the Gaussian Distribution.

A CNN is composed by different layers, which are briefly described here:

Convolutional Layer

If we indicated with l a convolutional layer, it accepts $m_1^{(l-1)}$ feature maps from the previous layer. Each map has size $m_2^{l-1} \times m_3^{l-1}$. For l=1, the layer is the input layer, which accepts the image I of size $n_1 \times n_2$. The output of a convolutional layer is simply m_1^l feature maps of size $m_2^l \times m_3^l$. The i feature map, which is denoted as $Y_i^{(l)}$ is computed by the following formula:

$$Y_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{m_1^{l-1}} K_{i,j}^{(l)} * Y_j^{(l-1)} \tag{3.21}$$

where B_i is a bias and K is the convolutional filter, which connects the j^{th} feature map in layer (l-1) with the i^{th} feature map in layer l.

Rectification

If the layer l is the rectification layer, its input includes $m_1^{(l-1)}$ feature maps of size $m_2^{(l-1)} \times m_3^{(l-1)}$. Then, the absolute value for each component of the feature maps is:

$$Y_i^{(l)} = \left| Y_i^{(l)} \right| \tag{3.22}$$

where the absolute value is computed point wise such that the output consists of $m_1^{(l)} = m_1^{(l-1)}$ feature maps unchanged in size.

Pooling

Subsampling from each layer to the next one can be useful to increase robustness to noise and distortions [63]. In the PhD research activities, it was used a a max pooling layer to perform down-sampling by dividing the input into rectangular pooling regions, and computing the maximum of each region. It was also used a stride with a specific step size for traversing the feature maps (and the input for l=1) vertically and horizontally. The stride is specified as a vector of two positive integers [a b], where a is the vertical step size and b is the horizontal step size. The stride must be obviously larger than the respective pooling dimensions otherwise the pooling regions overlap.

Fully Connected Layer

If l is a fully connected layer, the layer expects a $m_1^{(l-1)}$ feature maps of size $m_2^{(l-1)} \times m_3^{(l-1)}$ as input and the i^{th} unit in layer l computes:

$$y_i^{(l)} = f(z_i^{(l)} \tag{3.23}$$

with

$$z_i^{(l)} = \sum_{j=1}^{m_1^{(l-1)}} \sum_{r=1}^{m_2^{(l-1)}} \sum_{s=1}^{m_3^{(l-1)}} w_{i,j,r,s} \left(Y_j^{(l-1)}\right)_{r,s} \tag{3.24}$$

where $w_{i,j,r,s}$ indicates the weight, which connects the unit at position (r,s) in the j^{th} feature map of layer (l-1) with the i^{th} unit in layer l.

The objective is to minimize the error function. Different functions can be used. One of the most common, which has been used in our PhD activities is to the cross-entropy:

$$E(w) = \sum_{n=1}^{N} E_n(w) = \sum_{n=1}^{N} \sum_{k=1}^{C} t_{n,k} log(y_k(x_n, w)), \tag{3.25}$$

where $t_{n,k}$ is k^{th} entry of the target value t_n.

The objective is to minimize the $E_n(w)$ with respect to the network weights w. There are different techniques to achieve this. The necessary criterion can be written as:

$$\frac{\partial E_n}{\partial w} = \nabla E_n(w) \stackrel{!}{=\!=} 0 \tag{3.26}$$

Due to the complexity of the error E_n, a closed-form solution is usually not possible and we use an iterative approach. One of the methods is based on the gradient descent, which is motivated by the idea to take a step in the direction of the steepest descent, that is the direction of the negative gradient, to reach a minimum.

The weight update is:

$$\Delta w[t] = -\gamma \frac{\partial E_n}{\partial w[t]} = -\gamma E_n(w[t]) \tag{3.27}$$

In the standard gradient descent algorithm, the gradient of the loss function, $\nabla E_n(w)$, is evaluated using the entire training set, and the standard gradient descent

algorithm uses the entire data set at once.

By contrast, at each iteration the stochastic gradient descent algorithm evaluates the gradient and updates the parameters using a subset of the training data. A different subset, called a mini-batch, is used at each iteration. The full pass of the training algorithm over the entire training set using mini-batches is one epoch.

Other optimization algorithms seek to improve network training by using learning rates that differ by parameter and can automatically adapt to the loss function being optimized.

The other algorithm used in our activities is the root mean square propagation (RMSProp). It keeps a moving average of the element-wise squares of the parameter gradients:

$$v_l = \beta v_{l-1} + (1 - \beta) \left[\nabla E_n(w[t])\right]^2 \tag{3.28}$$

Then the optimization step becomes:

$$\Delta w[t] = \frac{\gamma \nabla E_n(w[t])}{\sqrt{v_l} + \epsilon} \tag{3.29}$$

where the division is performed element-wise. ϵ is a small constant added to avoid division by zero.

There are different hyperparameters, which can be tuned in the classification process. The following parameters have been mostly the focus of the optimization.

- l number of layers,
- optimization algorithm (e.g., RMSProp)
- the size of the feature maps n_1 and n_2
- stride

3.5.3 Classification with Support Vector Machines

Support Vector Machines were introduced by Vladimir Vapnik and colleagues [64]. The basic idea is that Support Vector Machine classifies data by finding the best hyperplane that separates all data points of one class from those of the other class. SVM was originally defined for binary-classification problems but various techniques have been proposed to extend it to multiclassifier problems. The best hyperplane for an SVM means the one with the largest margin between the two classes, where margin is the maximal width of the slab parallel to the hyperplane that has no interior data points. SVM is based on the concept of support vectors, which are the data points that are closest to the hyperplane and represent the boundaries.

A hyper plane in an n-D feature space can be represented by the following equation:

$$f(\mathbf{x}) = \mathbf{x}^T\mathbf{w} + b = \sum_{i=1}^{n} x_i w_i + b = 0$$

Dividing by $||\mathbf{w}||$, we get

$$\frac{\mathbf{x}^T\mathbf{w}}{||\mathbf{w}||} = P_{\mathbf{w}}(\mathbf{x}) = -\frac{b}{||\mathbf{w}||}$$

indicating that the projection of any point $\mathbf{x}$ on the plane onto the vector $\mathbf{w}$ is always $-b/||\mathbf{w}||$, i.e., $\mathbf{w}$ is the normal direction of the plane, and $|b|/||\mathbf{w}||$ is the distance from the origin to the plane. Note that the equation of the hyper plane is not unique.

The n-D space is partitioned into two regions by the plane. Specifically, a mapping function is defined as $y = sign(f(\mathbf{x})) \in \{1, -1\}$,

$$f(\mathbf{x}) = \mathbf{x}^T\mathbf{w} + b = \begin{cases} > 0, & y = sign(f(\mathbf{x})) = 1,\ \mathbf{x} \in P \\ < 0, & y = sign(f(\mathbf{x})) = -1,\ \mathbf{x} \in N \end{cases}$$

Any point $\mathbf{x} \in P$ on the positive side of the plane is mapped to 1, while any point $\mathbf{x} \in N$ on the negative side is mapped to -1. A point $\mathbf{x}$ of unknown class will be classified to P if $f(\mathbf{x}) > 0$, or N if $f(\mathbf{x}) < 0$.

The learning problem in SVM is that given a set K training samples from two linearly separable classes P and N:

$$\{(\mathbf{x}_k, y_k), k = 1, \cdots, K\}$$

where $y_k \in \{1, -1\}$ labels $\mathbf{x}_k$ to belong to either of the two classes it is necessary to find a hyper-plane in terms of $\mathbf{w}$ and b, that linearly separates the two classes.

For a decision hyper-plane $\mathbf{x}^T\mathbf{w} + b = 0$ to separate the two classes $P = \{(\mathbf{x}_i, 1)\}$ and $N = \{(\mathbf{x}_i, -1)\}$, it has to satisfy:

$$y_i(\mathbf{x}_i^T\mathbf{w} + b) \geq 0$$

for both $\mathbf{x}_i \in P$ and $\mathbf{x}_i \in N$. Among all such planes satisfying this condition, it is necessary to find the optimal one H_0 that separates the two classes with the maximal margin (the distance between the decision plane and the closest sample points).

The goal is to minimize the norm $||\mathbf{w}||$. Now the problem of finding the optimal decision plane in terms of $\mathbf{w}$ and b can be formulated as:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} = \frac{1}{2}||\mathbf{w}||^2 \qquad \text{(objective function)} \\ \text{subject to} \quad & y_i(\mathbf{x}_i^T\mathbf{w} + b) \geq 1, \quad \text{or} \quad 1 - y_i(\mathbf{x}_i^T\mathbf{w} + b) \leq 0, \qquad (i = 1, \cdots, m) \end{aligned}$$

Since the objective function is quadratic, this constrained optimization problem is called a quadratic program (QP) problem. This QP problem can be solved by Lagrange multipliers method to minimize the following

$$L_p(\mathbf{w}, b, \alpha) = \frac{1}{2}||\mathbf{w}||^2 + \sum_{i=1}^{m} \alpha_i(1 - y_i(\mathbf{x}_i^T\mathbf{w} + b))$$

with respect to $\mathbf{w}$, b and the Lagrange coefficients $\alpha_i \geq 0$ $(i = 1, \cdots, \alpha_m)$.

In our studies, a soft margin was used, because the context is such that the two classes (or more classes) in the PLIA are usually not linearly separable (e.g., due to noise). Then, the condition for the optimal hyper-plane can be relaxed by including an extra term:

$$y_i(\mathbf{x}_i^T\mathbf{w} + b) \geq 1 - \xi_i, \quad (i = 1, \cdots, m)$$

For minimum error, $\xi_i \geq 0$ should be minimized as well as $||\mathbf{w}||$, and the objective function becomes (for 1-norm soft margin problem):

$$\begin{aligned} \text{minimize} \quad & \mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{m}\xi_i^k \\ \text{subject to} \quad & y_i(\mathbf{x}_i^T\mathbf{w} + b) \geq 1 - \xi_i, \quad \text{and} \quad \xi_i \geq 0; \quad (i = 1, \cdots, m) \end{aligned}$$

Here C is a regularization parameter (soft margin) that controls the trade-off between maximizing the margin and minimizing the training error. Small C tends to emphasize the margin while ignoring the outliers in the training data, while large C may tend to overfit the training data. C is also called Box Constraint in the rest of this thesis.

C is one of the hyperparameter which must be optimized in the application of SVM. The other hyperparameter is based on the kernel representation of SVM as described in the following paragraphs. The application of the kernel trick means a transformation of the data into another dimension that has a clear dividing margin between classes of data.

In our studies, different types of kernels have been applied, but in particular the Gaussian kernel, which is defined in the following equation where γ is the scaling factor:

$$K(\mathbf{x}, \mathbf{z}) = e^{-\gamma||\mathbf{x}-\mathbf{z}||^2}$$

To summarize, the hyperparameters, which are tuned in the classification process during the PhD research activities were:

- hyperparameter linked to the chosen kernel: scaling factor γ for RBF,
- regularization parameter (soft margin) constant C (i.e., box constraint)

3.5.4 Classification with KNN

The KNN machine learning algorithm is based on a search algorithm, which lets you find the k closest points in X to a query point or set of points Y. kNN-based algorithms are widely used as benchmark learning rules because of their relative simplicity, which makes it easy to compare the results from other classification techniques (like SVM and CNN used in these PhD studies).

The "closest" is based on a distance metric, which can be of different types: Euclidean distance, Hamming distance, Jaccard distance, Chebychev distance and Minkowski distance.

The hyperparameters, which are tuned in the classification process during the PhD research activities were:

- K factor in the KNN,
- distance function. For example euclidean distance, manhanattan distance or Chebychev distance,

3.6 Results

This section describes the results for the different studies, which were described as novel in section 2. This section is divided in two main subsections: the application of CNN to the problem of PLIA, which is conducted separately for the GSM data set and the IoT data sets.

3.6.1 On the application of CNN to the RF data set

GSM data set

The proposed CNN model for PLIA in the GSM data set GSM-DS4 is illustrated in Figure 3-4. It is made up of three convolution layers and one fully connected layer because it achieved the optimal balance between computing time and performance. Three of the techniques described in section 3.4.5 are used on this data set and

the results were published in [24] and they are reproduced here. In particular, to transform the original time series derived from the digitized signals collected from the GSM wireless devices, it was used the Recurrence Plots (RP) (described in section 3.4.1), the CWT (described in section 3.4.5) and the STFT (described in section 3.4.3). Then, the images were used as an input to the CNN and the techniques are respectively called RP-CNN, STFT-CNN and CWT-CNN. It is noted that the original time series is complex (as a modulated wireless communication signal can be represented as a complex signal). Then, an initial feature analysis process was applied to verify which parts of the signal were optimal for the classification. The following combinations were tried: amplitude, phase and amplitude+phase as a complex signal can be represented as $x(t) = a * e^{j\theta(t)}$. Then, it was found that the amplitude component provided the best classification accuracy across all the techniques. In a similar way, some transforms (e.g., CWT) provided a complex output. It was also empirically found that the amplitude component of the complex transform output provided the best classification accuracy.

The comparison among the imaging techniques is shown in figure 3-3. From the initial representation of the GSM burst with the amplitude, three different images are generated: the first one is based on the application of the recurrence plots, the second one is based on the application of the CWT and the third one is based on the application of STFT. As the application of CWT and STFT generates a complex two dimensional matrix, both the amplitude and phase components are used to generate the image. As shown in 3-3, the amplitude component of the transform is stacked above the phase component for CWT and STFT representations.

A description of the CNN architecture used to produce the results in Figure 3-5 is shown in Figure 3-4. The parameters of the first input layer and convolution layer are adapted to optimize the classification performance of each technique (a cross-validation on a 4-fold was used). For instance, for RP-CNN, the input size was 114*114, as exemplified in Figure 3-4. This input is based on the application of the RP algorithm on the initial time series length (a burst of size 130). For CWT, the input size was 80*130, while for STFT, the input size was 16*129. The pooling size

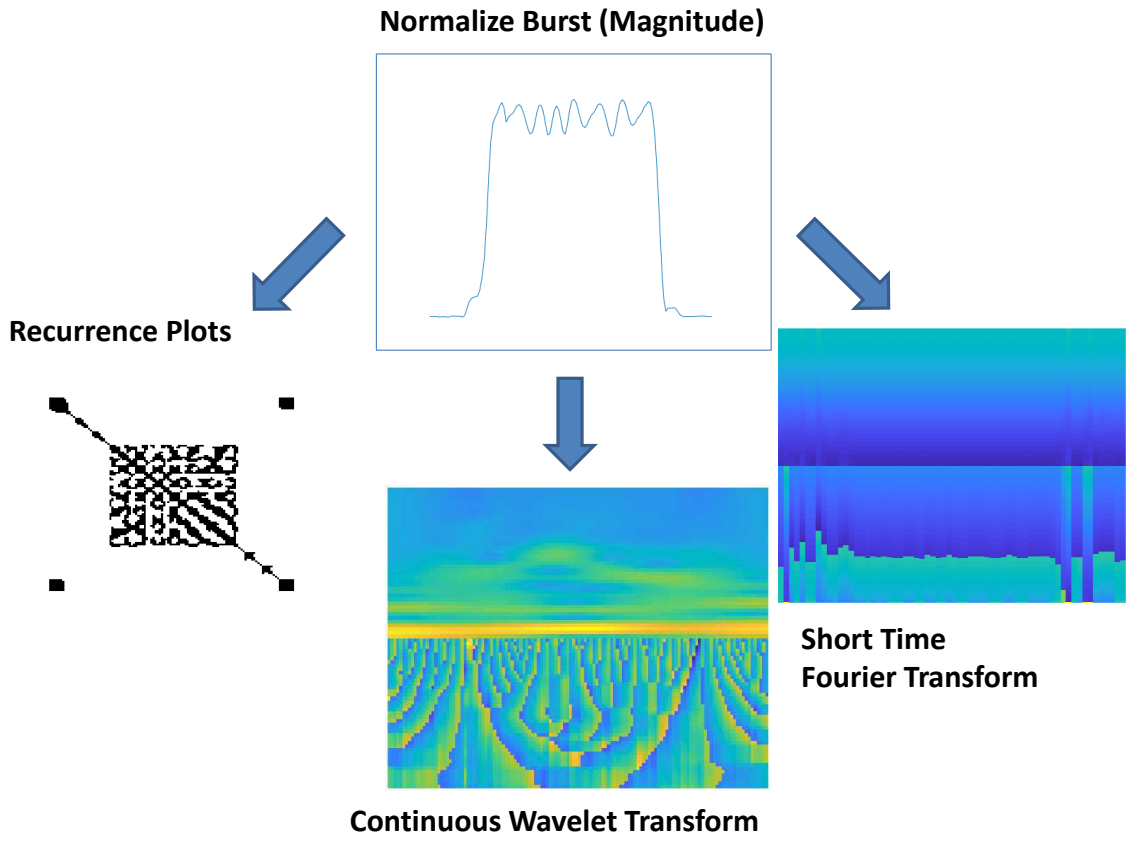


Figure 3-3: Application of the imaging techniques

was set to 4, while the size of the second convolution layer was set to 10*10.

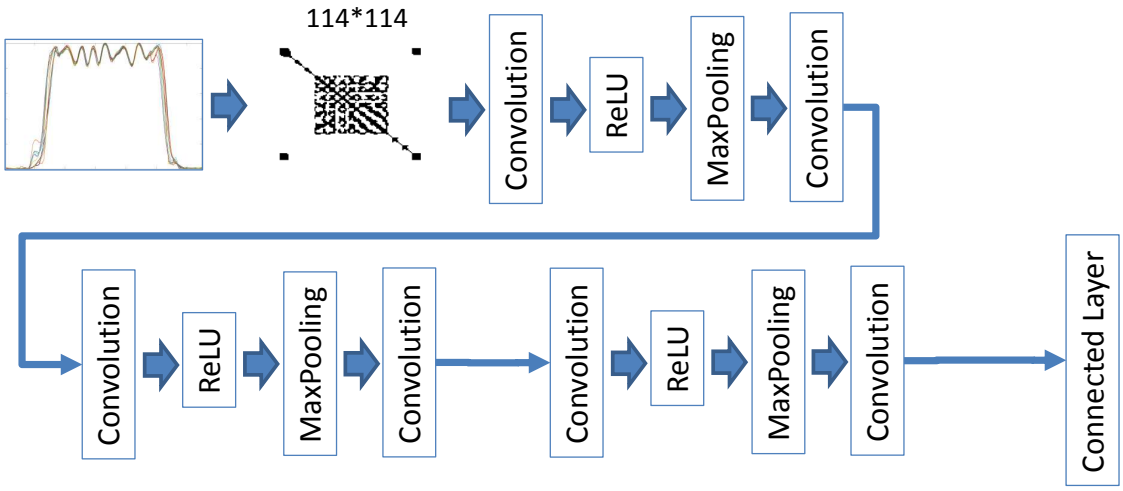


Figure 3-4: The proposed 3-stage CNN architecture for RP-CNN.

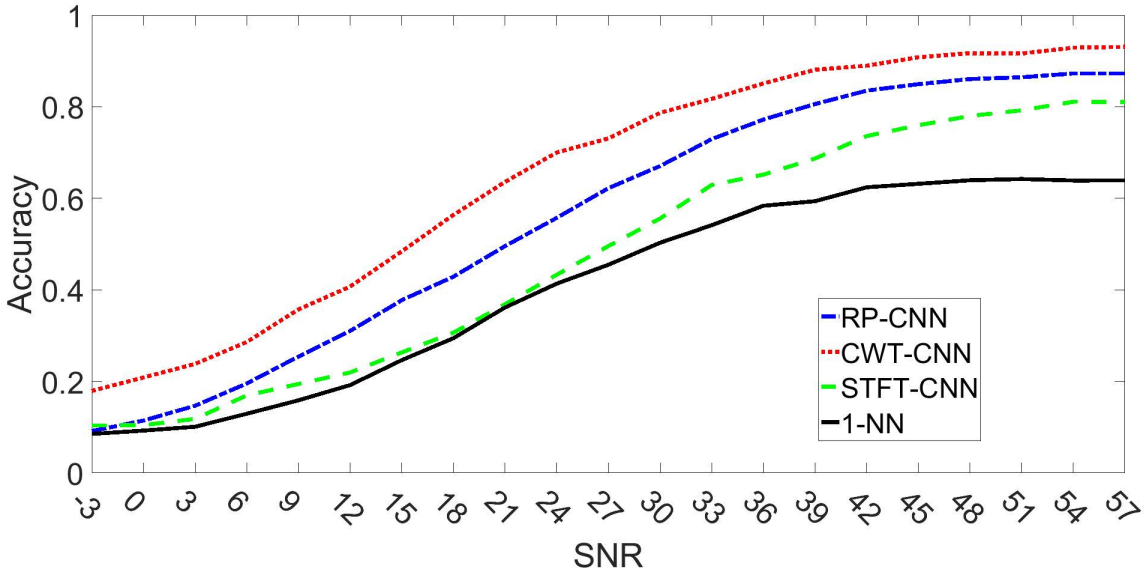


Figure 3-5: Identification accuracy in the presence of AWGN, as function of SNR. SNR is expressed in dB.

The results presented in figure 3-5 show the identification accuracy as a function of SNR(dB) for all the three representations RP-CNN, STFT-CNN and CWT-CNN. It is also provided the identification accuracy by applying KNN with K=1 to the original time series (i.e., amplitude of the GSM burst). It can be seen that CWT-CNN provides the highest identification accuracy for all the values of SNR.

IoT data set

Building on the results presented in 3.6.1 where CNN has been successfully applied to the physical layer authentication of wireless devices, more recent TFAs were investigated in combination with CNN on the IoT data set. In particular, it was explored the application of TFAs, which are particularly adapt to classify the transients of the bursts, which are strongly stationary and non-linear. In particular, it was investigated the application of GLCT, which is described in section 3.4.6 and it was compared with the application of CWT, which was also used in the GSM data set as described in section 3.6.1.

As the focus of the research was on the classification of IoT devices using the transient part of the burst, only the transients sections of the IoT-DS1 were used to

perform the experimental evaluation. The original transients for the 9 IoT devices used for classification are shown in Figure 3-6. It can be seen that each IoT device produce a slightly different transient shape. These differences are the fingerprints, which can be exploited for the classification. The application of the GLCT to a single transient from an IoT device (i.e., IoT device 3) is shown in Figure 3-7.

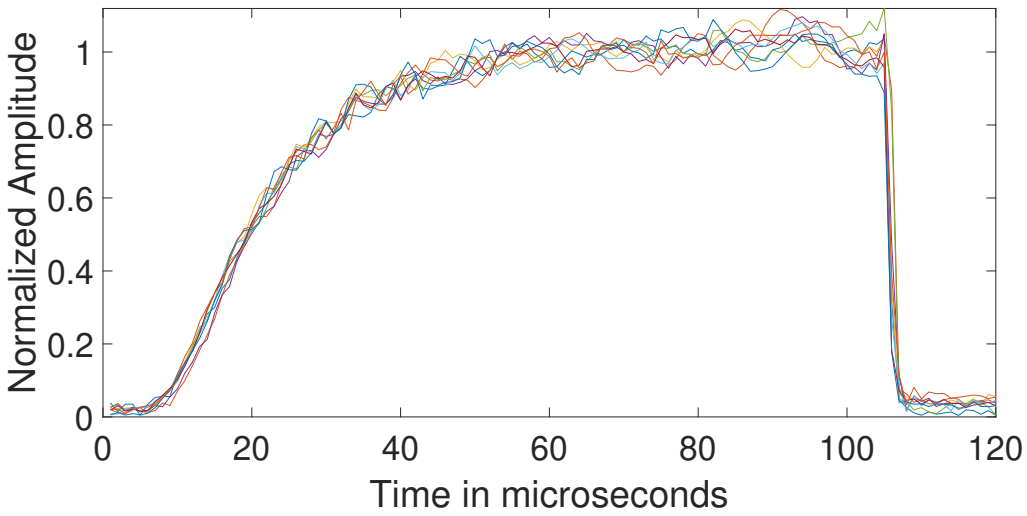


Figure 3-6: Transients of the bursts for the 9 IoT devices

In particular, the GLCT has been used and compared with other transforms and the conventional machine learning like SVM and KNN for decreasing values of the SNR against CWT used in the previous section for the GSM data set.

With a similar convention of the previous GSM data set, the different techniques are called respectively GLCT-CNN (combination of the GLCT transform with CNN), CWT-CNN (combination of the CWT transform with CNN), T-CNN (time domain with CNN), T-SVM (time domain and SVM) and T-KNN (time domain and KNN). Each of the machine learning algorithms (i.e., CNN, SVM and KNN) have been optimized regarding the choice of the value of the hyperparameters described in section 3.5.

For the purpose of using GLCT to obtain an optimal representation of the signal in the TF space for device identification, two hyperparameters must be empirically determined: the window w and the parameter N, which is indicated as the number

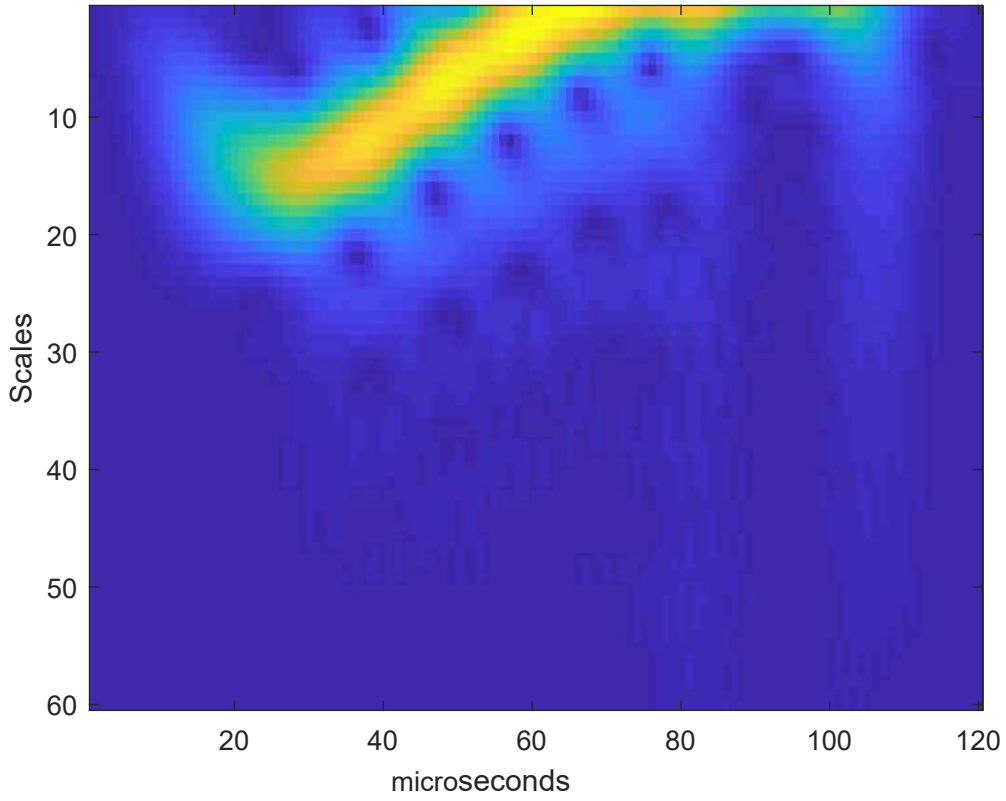


Figure 3-7: GLCT of the burst transient (X axis in microseconds)

of used chirplets. The optimal values are identified on the metric of the identification accuracy.

Figure 3-8 presents the classification accuracy by varying the size of the window (as number of samples) and for different values of SNR in dB by using GLCT-CNN. The optimal window size w is 30, which is used in the subsequent results.

Then, different values of the parameter N are chosen. It was empirically found that the value of N does not have a significant impact on the classification accuracy like the w window size and the SNR related curves are not clear distinguishable. Then, the classification accuracy for specific values of SNR is presented in Figure3-9. While it is clear that higher values of N provide a better classification accuracy, the value of $N = 14$ is chosen, because increasing values of N (e.g., N=16) do not present significantly better results and higher values of N would increase the computation complexity anyway.

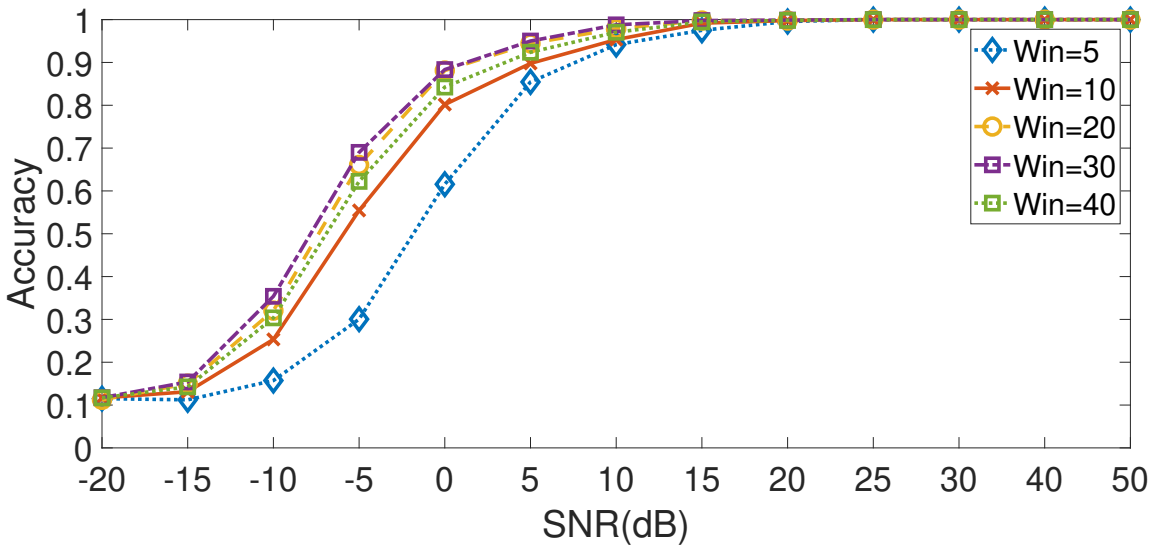


Figure 3-8: Accuracy results based on the window size of the GLCT and using GLCT-CNN for classification.

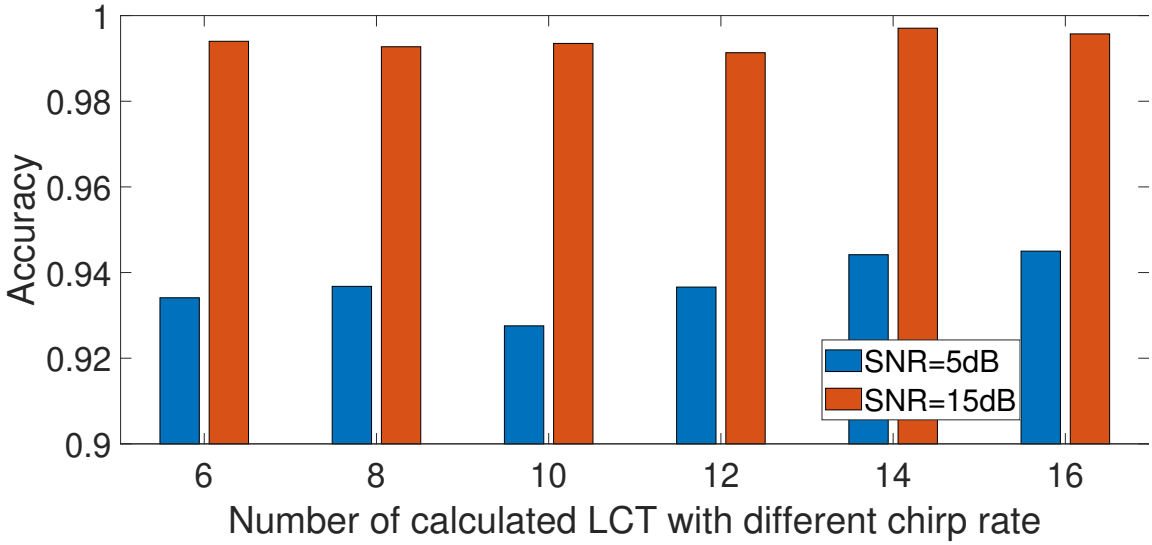


Figure 3-9: Comparison of the number of used chirplets in GLCT and using GLCT-CNN for classification.

The results are shown in figure 3-10, where it can be seen that the previous results for the GSM data set GSM-DS4 are also confirmed for the IoT data set as the TFA (both GLCT and CWT) in combination with CNN is significantly more performant than the basic CNN-time domain combination. In addition, it is demonstrated that GLCT is slightly more robust than CWT especially for lower values of the SNR expressed in dB. Then, this transform is preferable to CWT when the classification of

the wireless devices must be performed in noisy environments. This advantage comes at a greater computational cost of the GLCT transform in comparison to the CWT transform.

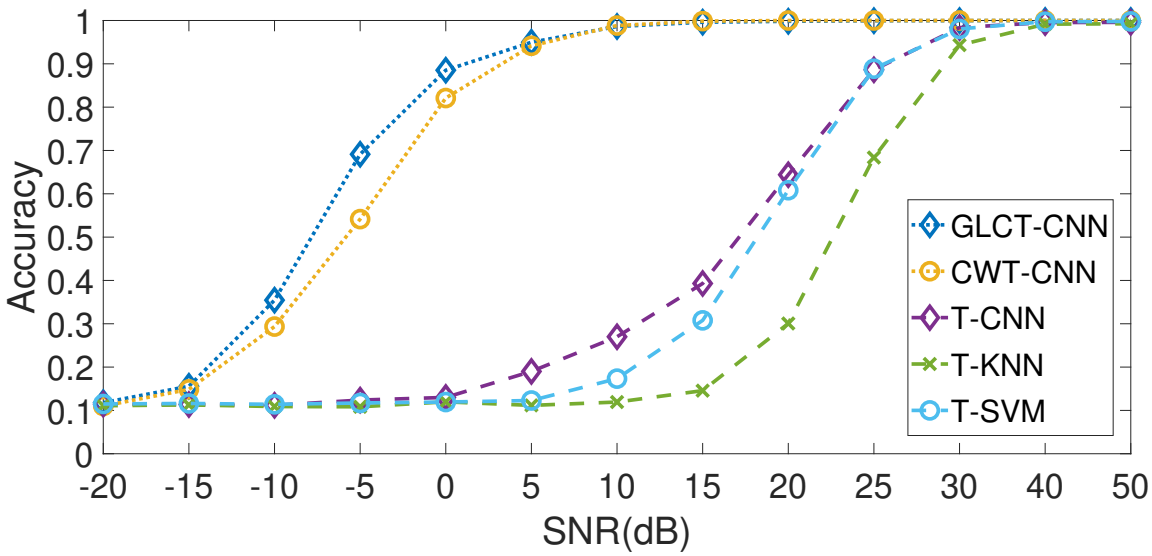


Figure 3-10: Comparison of the different techniques using CNN and different transforms for the IoT data set

3.6.2 Bias introduced by the receiver used to collect the RF signal in space

Analysis of the impact on fingerprints portability due to the bias introduced by the measurement instrument used to collect the signal

Note: Most of the text and figures of this section are extracted from the published paper [27], which received the best paper award at IEEE NMTS 2018 conference.

One issue presented in literature and still unresolved before this study is the lack of portability of the RF fingerprints. In the typical RF fingerprinting scenario, RF emissions of the wireless devices are collected by one RF receiver (called A), which converts the emissions in digital format, from where the fingerprints are extracted. The lack of portability issue is due to the fact that if another RF receiver (called B) is used to collect the fingerprints, this new receiver introduces a bias, which degrades the RF fingerprints of the wireless device taken with the first receiver A. As a consequence,

RF emissions of the same wireless device taken from different RF receivers A and B will generate different fingerprints for the same wireless device. This issue strongly limits the applicability of RF fingerprinting for security applications, because different RF receivers cannot be used to perform identification and the fingerprints are not portable from one receiver A to another receiver B. This thesis proposes a novel approach to mitigate the portability issue. The approach is based on two techniques: a) the removal of the bias introduced by RF receivers in the frequency domain through the use of one golden reference. The golden reference is used to generate a calibration function, which is then applied to the RF emissions collected by different RF receivers from any other wireless device. b) the identification of a space in the frequency domain (i.e., range of frequencies) where the portability issue is less present. This second technique is implemented by using a sliding window in the frequency domain and evaluating the impact of the portability issue in each window. The approach is validated against a set of 10 IoT wireless devices (plus the golden reference) and with 3 RFreceivers. This is the data set IoT-DS3 indicated in section 3.6.1

The experimental evidence demonstrates that this method is able to alleviate the portability issue at the cost of a minor degradation in identification accuracy. Two metrics were chosen to evaluate the performance of the approach: the overall identification accuracy and the average distance among the samples taken from the three receivers for the same wireless device under test. In an ideal situation, there should be no impact to the identification accuracy obtained with the single receivers and the distance among samples of different receivers should be equal to zero for the same wireless device under test. The approach works in the frequency domain to identify empirically the segment in the frequency domain where the two metrics are optimized.

The overall methodology is depicted in 3-11.

1. The first step is to collect the RF signal in space emitted by eleven (11) different Nordic devices and using three different RF receivers (data set IoT-DS3). The RF emissions are collected in sets of 600 bursts per device.

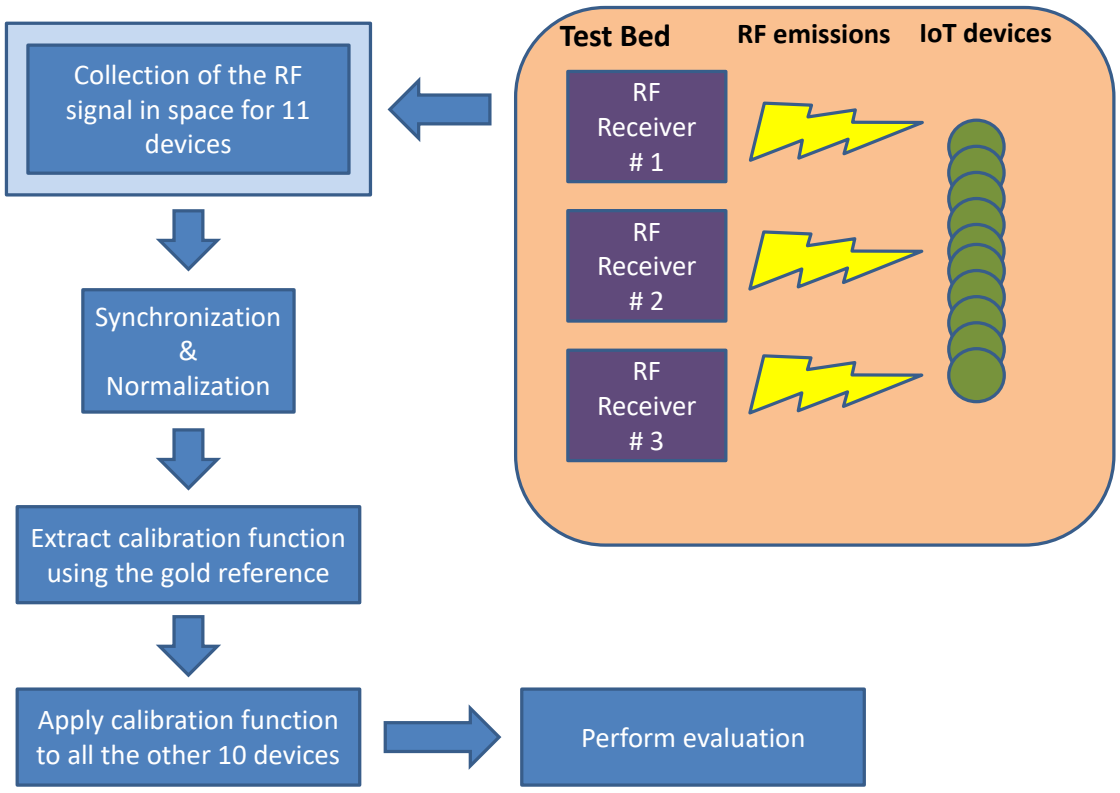


Figure 3-11: Overall methodology

2. The second step is to synchronize and normalize the collected bursts.

3. A dimensionality reduction is performed by extracting various statistical features (variance, skewness, kurtosis and entropy) from the bursts

4. The synchronized and normalized bursts (expressed using the statistical features) for the golden reference from all the three RF receivers are processed to generate the calibration function.

5. The calibration function is applied to the synchronized and normalized bursts (expressed using the statistical features) of all the other wireless devices used in the test bed (i.e., 10 wireless devices).

6. The performance of the calibration function is evaluated using the specific metrics described below. In particular the distance among the burst is calculated

and the identification accuracy is also calculated using SVM with a Radial Basis Function (RBF)

The metrics to evaluate the performance of this approach are:

1. The distance between the frequency domain representation of the bursts for all the three receivers D_c. The distance is calculated as the sum of all the distances, which should be minimized because a large distance means that the bursts from the three RF receivers are easily distinguishable while this should not be the case because the bursts are taken from the same wireless device. In other words, a large distance implies a lack of portability.

2. The identification accuracy of the wireless devices among themselves Acc_p. Even if the portability issue is mitigated and resolved, this result would not be useful if the resulting accuracy is greatly degraded.

A simplified model in the frequency domain of the relation between the signal received at each receiver, the transmitted signal from the RF device to be identified is described in the following equation:

$$R_i(f) = HR_i(f) * P_{ij}(f) * HE_j(f) * S(f) \tag{3.30}$$

where $R_i(f)$ are the digital sequences provided by each receiver i. HR_i are the transfer function in the frequency domain for each receiver. The transfer function includes various elements of the a RF receiver like the front end, the ADC, low pass filters and so on. $P_{ij}(f)$ represents the transfer function for the wireless propagation environment and the antennas between receiver i and emitter j. HE_j represents the transfer function of the j_{th} IoT transmitter. $S(f)$ is the digital signal to be transmitted, which is always the same series of bits in our case as the IoT device has been configured to transmit the same test sequences. The RF fingerprints are represented by HE_j, which is the quantity we want to determine, but in the portability problem, the HR_i quantities are unknown, which does not allow to solve the equation in closed form.

The main idea for this approach to define the calibration function to address the portability issue is to identify the segments in the frequency domain of the received signal $R_i(f)$, which are not affected by the bias of the RF receiver or they are at least constant. The frequency domain is chosen because the bias introduced in the receiver is usually localized in specific regions of the frequency domain representation of the wireless signal (i.e., the radio frequency spectrum). As a consequence, the approach tries to resolve a simple optimization problem where the value of metric D_c should be minimized and the value of metric Acc_p should be maximized.

A sliding window approach is used in the frequency domain to identify the optimal segments in the frequency domain. The results (which are described in detail in [27]) show that the approach is effective in identifying the optimal segment in the frequency domain where the bias is reduced at the small cost of loss in the identification accuracy among the 10 wireless devices due to the fact that only a segment of the overall frequency representation of the signal is used (98.3% of accuracy instead of the baseline 99.9677%).

A pictorial description of the positive results of the approach is provided in the following scatter plot figures in 3 dimensions, which are chosen to be the most relevant using the RelieFF selection algorithm ([65]). The clusters in figure shows the extracted features from the 400 bursts of a specific wireless device (the first device in 10 devices data set used for classification) as "seen" from the three different wireless receivers.

Figure 3-12 is the baseline case where the whole burst in the frequency domain is used and no segment optimization is applied.

It can be seen that the same wireless device appears as three different separate clusters (the green, blue and red clouds) because of the bias introduced by the receivers.

Figure 3-13 is the case where the optimization process has been applied. It can be seen that the clusters are now confused among others as it should be because it is the same wireless device, which is analyzed.

The results are more clear when the distance of the centroids of the clusters are calculated across all the 10 devices in the data set for different portion of the frequency

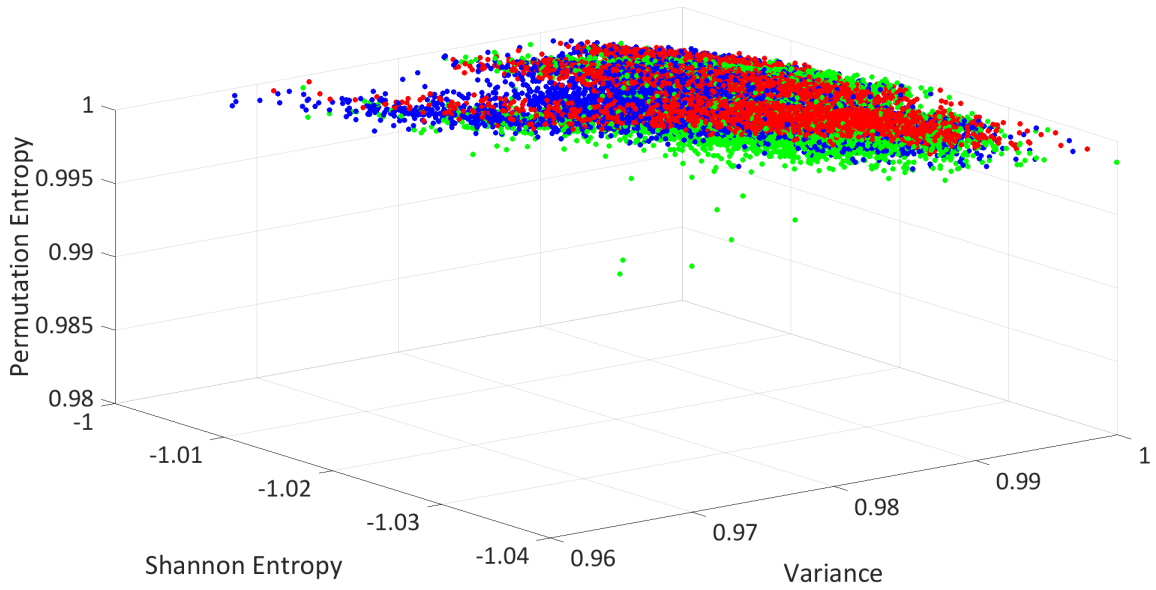


Figure 3-12: 3D scatterplots for the baseline case

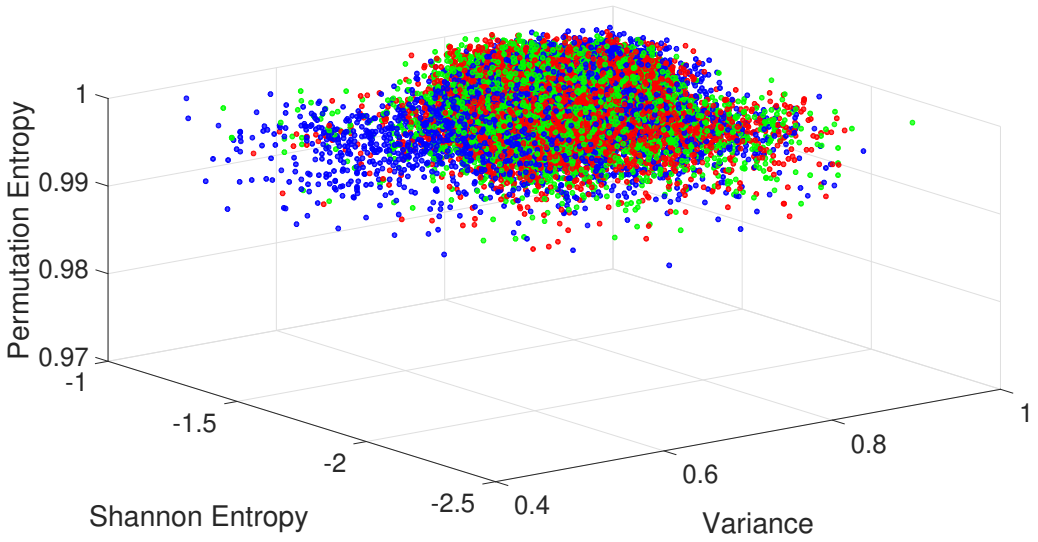


Figure 3-13: 3D scatterplot in the optimized case

domain representation of the burst (the segment index). The baseline is when the entire frequency domain representation is used. The distance among the centroids of the clusters on the basis of the window index are visible in Figure 3-14, which also provides the selection of the best segment index Window Index=3 (which is used to create Figure 3-13). We can see that the distance between the optimal case and the baseline is significant.

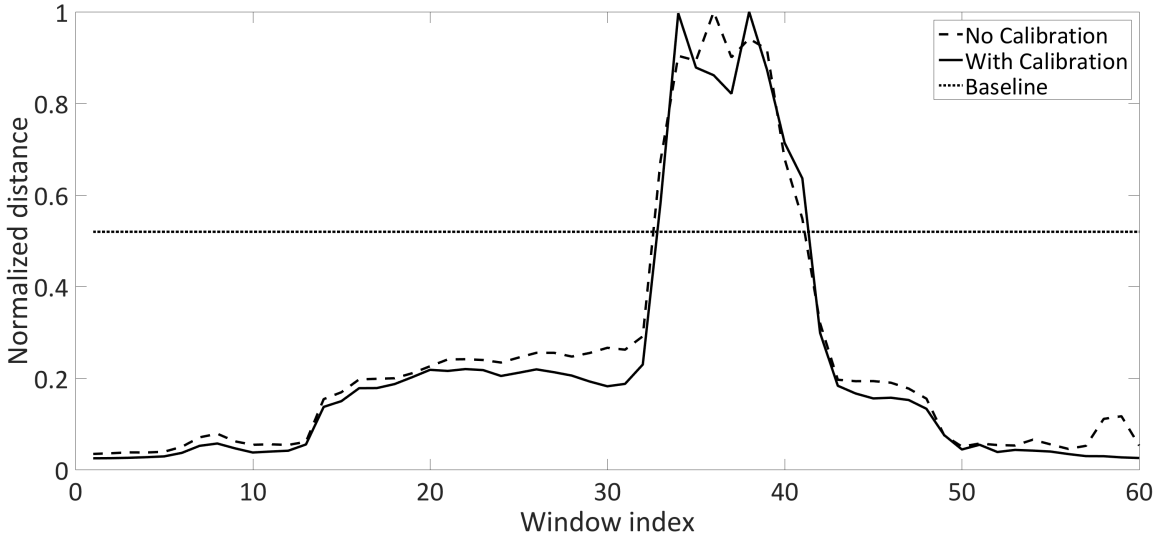


Figure 3-14: Distance among the clusters with applied calibration and no calibration. Baseline is also shown in green dotted line

The Figure 3-14 shows that for this specific data set the contribution of the calibration is limited even if it is still positive (i.e., where the calibration curve is almost always below the non-calibration curve). For example, at window sizes 30 and 35, 60, the calibration provides a significant benefit. On the other side, for this specific data set, the optimal window is $W = 3$, where the calibration has a positive but limited impact. Because it is known a priori which window is the optimal one (because it depends on the location of the fingerprints in the time and/or frequency domain), it is suggested to adopt both methods to mitigate the portability problem.

Analysis of the impact introduced by the IQ imbalance in the RF receiver

Most of the text and figures described in this section are extracted from the published paper [31].

The IQ imbalance is a phenomenon appearing in DDC RF receivers, which transforms the RF-signal directly down to base-band. The RF receiver used in our test bed (an USRP type N200 receiver) is based on a DDC architecture. For that purpose, the LO is set to the carrier frequency $\omega_{LO} = \omega_{RF}$ of the wanted channel. As described in [28], due to temperature dependencies, production imperfections and aging, the analog components in the I-path and Q-path can not be perfectly matched. The IQ-

imbalance is a serious issue, which can degrade the receiver performance and it can impact the emitter identification process, because both the amplitude and the phase of the Local Oscillator (LO) signal of the I and Q paths will differ from the optimal values resulting in phase (φ) and gain (g) imbalances, which can be described using the following equation:

$$s_{LO}(t) = g_1 \cdot cos(\omega_{LO} + \varphi_1) - jg_2 \cdot sin(\omega_{LO} + \varphi_2) \tag{3.31}$$

The overall schema of the IQ imbalance with the different paths for I_{BB} and Q_{BB} in a DDC are shown in Figure3-15 with the Low Pass Filter (LPF) and the ADC.

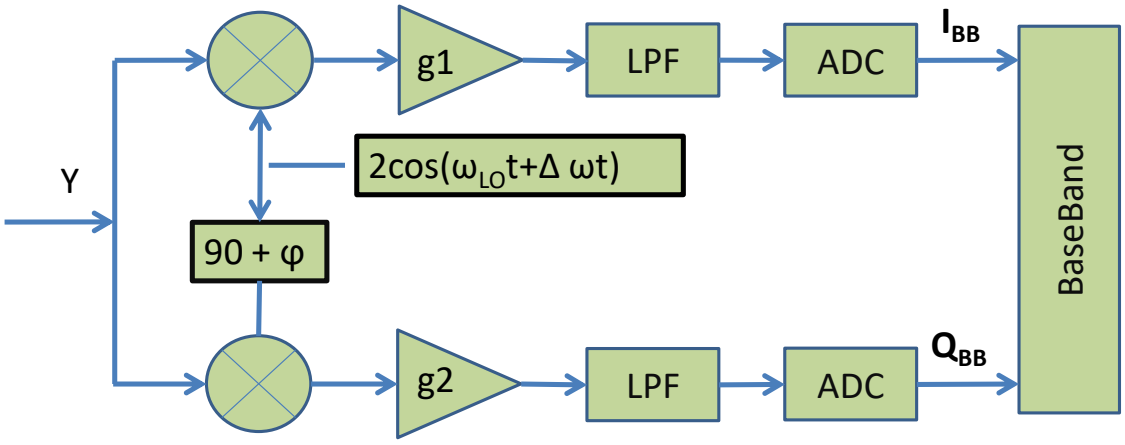


Figure 3-15: Schema of the RF receiver with different paths for I and Q

In this report, we take into consideration different values of g_1 and g_2 for the gain IQ imbalance and φ_1 and φ_2 for the phase IQ imbalance. In the initial step, a range of realistic values on IQ imbalances are used. These range of values are extracted from literature [30, 29]: IQ imbalance from 1dB to 10dB of Gain imbalance where g_1 is set to a fixed value of 0 and g_2 which progressively changes from 1dB to 10db or alternatively from -1dB to -10dB (in steps of 1dB), while keeping the phase imbalance to 0. Then, phase imbalance is modified while keeping the IQ gain imbalance to 0: φ_1 is set to a fixed value of -5 degrees and φ_2 increases incrementally in step of 1 degree to 5 degrees or alternatively with φ_1 set to a fixed value of 5 degrees and φ_2, which decreases to -5 degrees.

The impact of such increasing values of IQ imbalances on PLIA are evaluated using different representations of the RF signal using the IoT data set described in section 3.2. The different representations used in the study were the Time domain (the original representation of the digitized signal in space), the Frequency domain (described in section 3.4.2), the CWT (described in section 3.4.5 and the Wigner Ville transform (described in section 3.4.4). The reason for evaluating different types of representations was to investigate if the application of a transform could mitigate the negative impact of the IQ imbalance on the physical layer authentication of the wireless devices.

For this analysis, we adopted the KNN algorithm for classification (see section 3.5.4 for a description of the KNN algorithm) and it was purposely decided to select a choice of hyperparameters of K=1 and distance metric equal to Euclidean distance as the focus on the research was on the evaluation of the impact of IQ imbalance and the mitigation techniques based on different transforms, rather than the goal to obtain an optimal classification (which could be obtained with more sophisticated machine learning algorithms). A 10-fold was used for classification.

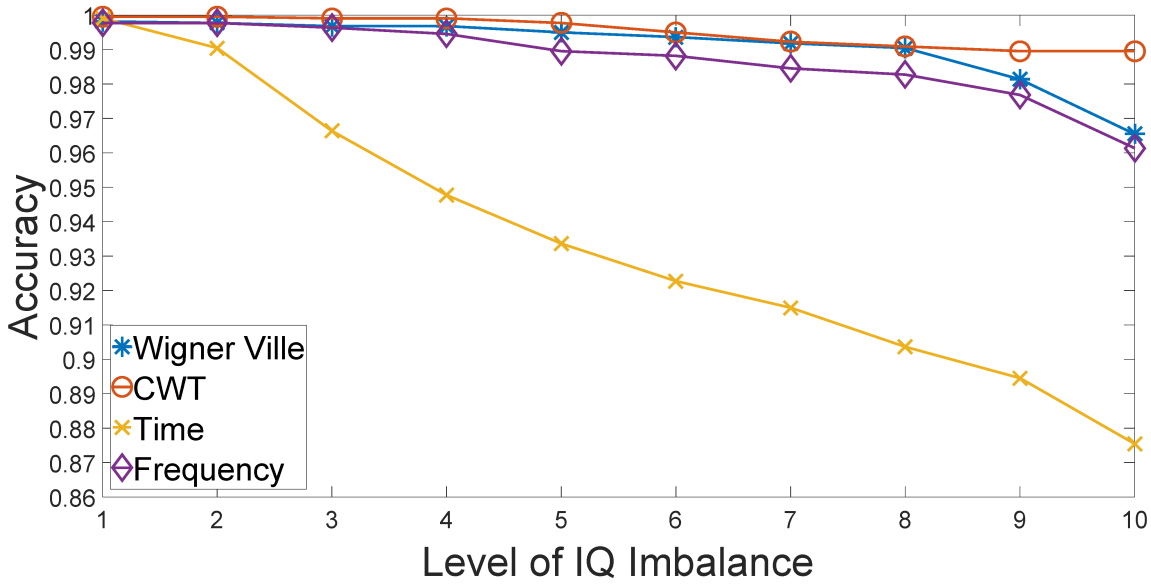
Figures 3-16a and 3-16b show the comparison (in terms of identification accuracy) of the different representations of the signal (Time Domain, Frequency Domain, Wigner Ville and CWT) using the KNN machine learning algorithm with $K = 1$ and the Euclidean distance for the different levels of IQ imbalance. Figure 3-16a shows the results for the negative values of the IQ imbalance in the range $IQ_g = -1 : -10$ dB and $IQ_{ph} = -1 : -10$ degrees, while figure 3-16b shows the results for the positive values of the IQ imbalance $IQ_g = 1 : 10$ dB and $IQ_{ph} = 1 : 10$ degrees.

The results show that the positive values of the IQ imbalance (figures (a)) for this specific data set provides slighter worst accuracy values than the negative values (figures (b)). In all cases and as expected, the presence of IQ imbalance degrades the classification performance, with a loss which can be quite significant for some representations. For example, the Time domain representation has a loss in classification accuracy of 8.9% for IQ_g=10 dB and IQ_{ph}=10 degrees.

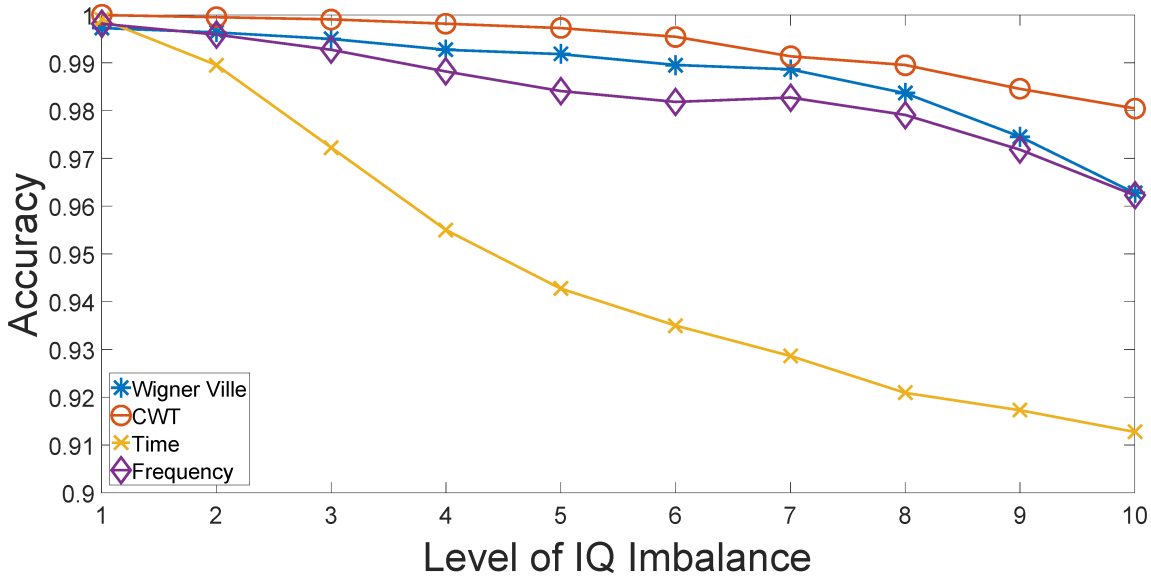
From all the figures, it is clear that the CWT representation outperforms the other

representations for increasing values of the IQ imbalance for all the various sets. The difference in performance of CWT in comparison to the other representations increases with the values of IQ imbalances. The results seem to indicate that the greater amount of information provided by CWT is able to mitigate the effect of IQ imbalances. The superior performance of physical layer authentication using CWT in the presence AWGN has also been recently demonstrated in [24]. The Wigner Ville distribution is quite sensitive to IQ imbalances, and especially to phase imbalances.

The conclusion of this analysis is that the presence of IQ imbalances in the RF receiver used to collect the RF signal in space from IoT wireless devices do have a significant negative effect on classification accuracy. This issue is part of the more general problem of portability of the RF fingerprints. On the other hand, the provided results seem to indicate that the adoption of specific representations, like CWT, can mitigate this problem at least for IQ imbalances.



(a) Range $IQ_g = -1 : -10$ dB and $IQ_{ph} = -1 : -10$ degrees.



(b) Range $IQ_g = 1 : 10$ dB and $IQ_{ph} = 1 : 10$ degrees.

Figure 3-16: Comparison of classification accuracy for different representations in the presence of IQ imbalance. $MF_{phase} = 1$.

Chapter 4

Case Study of microphone identification

4.1 Overview

Note: some of the material presented in this section is extracted from the paper [66].

In this section, we report on the results of the identification/authentication of smartphones using the intrinsic physical properties of the built-in microphone of the smartphone. The possibility to identify a microphone on the basis of features extracted from audio recordings is well known in literature but it is mostly used in forensics studies and is usually relies on human voice recordings (as described in section 2.3). Instead, the study conducted in the PhD thesis focused on the identification and authentication approach for mobile phones by stimulating the built-in microphone with non-voice sounds at different frequencies.

The advantage of using microphones for identification/authentication in comparison to other components in the mobile phone like the camera [42] or the voice is the possibility to control the stimulus, which is applied to the microphone from an external device. In this way it is easier to create an extensive challenge/response space as described in the Figure 4-1. This is more complex for the other components like a camera [42], where the recorded image can be totally random (based on the collected visual context) or the radio frequency fingerprints where the wireless standards may impose specific constraints. In a similar way, the common method (used in forensics) to identify the microphone on the base of the voice recordings has the similar problem that it is not possible to create a specific stimuli to define a challenge/response space. In this scenario, it is possible to create a wide space of challenges (the external stimuli) against related responses (the fingerprints in the recordings), where the microphone (and then the mobile phone) can be uniquely identified by the fingerprints. An attacker would need to record the microphone stimuli and the response as well, which is quite difficult because: a) the sound stimuli degrades significantly with the increasing distance between the emitter and the microphone, b) the attacker would need to have access to the audio recordings and know the process to extract the fingerprints (e.g., the type of statistical features used).

A potential application scenario is provided in Figure 4-1.

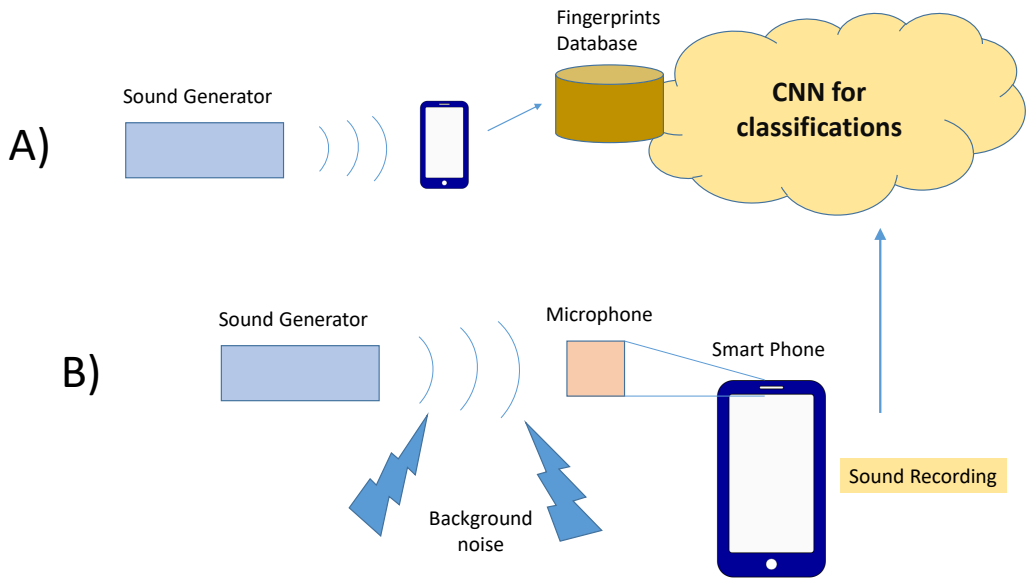


Figure 4-1: Potential application scenario for microphone based identification

In an initial first phase (identified with A in the Figure 4-1) the microphone of a smartphone is stimulated by a sound generator. The sound recording is collected and stored in a database of fingerprints, which is accessible by a cloud application which is used in the in-field identification and authentication phase (identified with B in the Figure 4-1). In the second phase B, the microphone is stimulated by the same sound used in the first phase A. After that, the audio recording is sent to a cloud application which uses a CNN algorithm to identify the source microphone in the fingerprints database in case of identification. In the authentication scenario, a phone with claimed identity of the phone P_i will be compared with the recording in the fingerprint database associated to P_i.

4.2 Materials

A set of 34 phones have been used to collect the audio recordings. This dataset is much larger than other datasets used in most of the literature, and is comparable in size to the large dataset recently used in [39]. However, compared to [39], our collection of mobile phones includes a larger number of phones of the same model, so

Mobile phones	Quantity
HTC One	3
Samsung Galaxy S5	3
Samsung ACE	25
Sony Experia	3
Total	**34**

Table 4.1: List of mobile phones used in our experiments.

as to properly address the more difficult problem of intra-model classification. The smartphones we used are a mixture of recent and old phones and the list is shown in Table 5.1.

A tone of 1KHz was generated with MATLAB and amplified through a high quality Onkyo amplifier. The tone was reproduced 800 times in 8 different days, while each mobile phone was in recording mode. The recording was stored in Pulse Code Modulation (PCM) raw format at 44.1 KHz. The microphone sensitivity and the level of the amplifier was adjusted to avoid the saturation phenomenon in the audio recordings. The position of the phones was always the same relative to the amplifier. The microphone was placed on a plastic absorber to minimize the impact of vibrations from the supporting surface.

4.3 Methodology

The goal of this section is to describe the overall methodology of the proposed classification method. The complete workflow is presented in Figure 4-2, and the single steps are detailed as follows.

- Each of the 34 mobile phones was stimulated by an audio signal with a duration of 2 seconds at a frequency of 1KHz generated by a sound source and amplified by an amplifier. The audio signal increases to a maximum value and then decreases to zero amplitude. As described before, the audio signal was repeated 800 times in different days with a period of 2 seconds, and one second separation between one audio signal to the next.

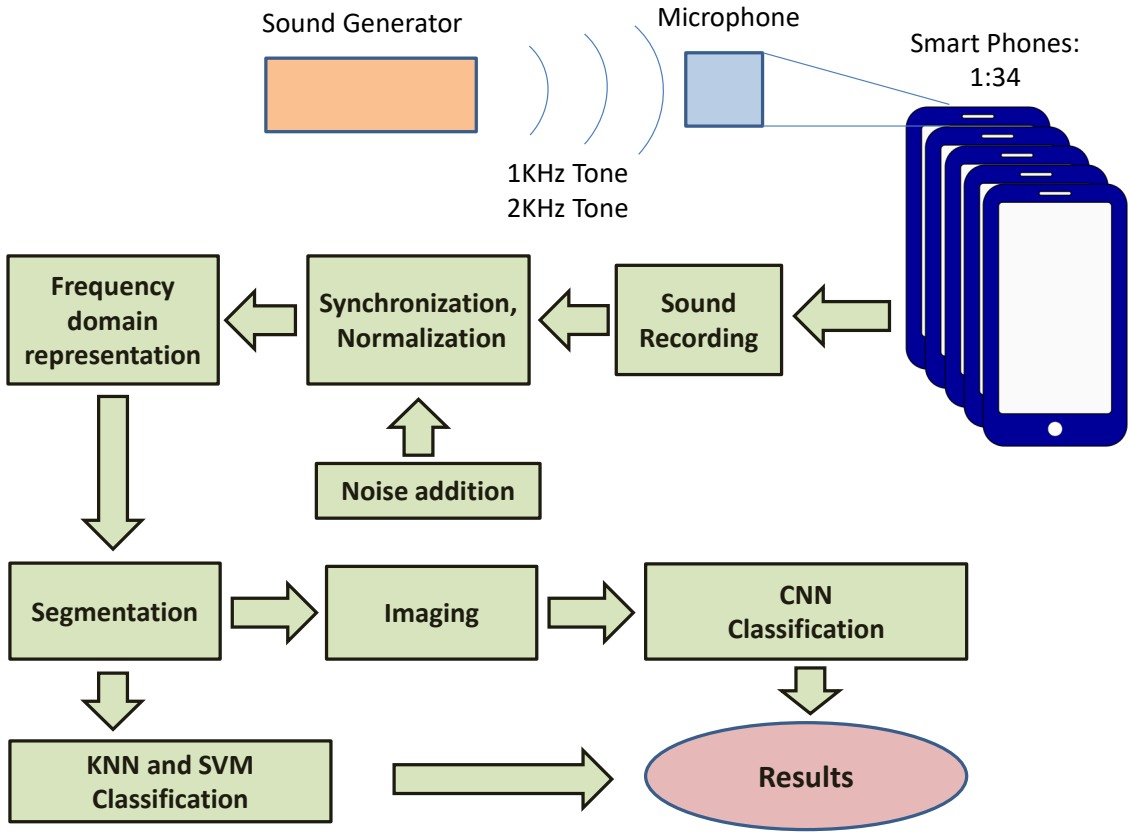


Figure 4-2: Overall methodology for microphone identification using the proposed CNN-based approach.

- The audio recordings from each mobile phone were first normalized and then synchronized using the first phone as a reference. In particular, Pearson's correlation was employed in such a way that the recording of each phone was shifted and correlated with the recording of the reference phone (i.e., the appropriate shift value was obtained through the maximum value of the correlation).

- AWGN was added to the audio recordings with different values of SNR, in order to simulate the presence of attenuation due to path loss.

- A Fast Fourier Transform (FFT) was applied to the sound recordings to generate the frequency response of each microphone. The transformation to the frequency domain was applied because empirical evidence has shown better identification accuracy over signals in the time domain. These empirical findings

are consistent with the recent findings on microphone authentication described in [39] and [40].

- A tailored CNN was then used for identification. The CNN architecture is similar to the one used for the RF use case and presented in section 3. Details on the CNN architecture are shown in sub-section 3.5.2. To make a comparison, we have also used the SVM and KNN machine learning algorithms as presented in the RF use case 3 and subsections 3.5.3 and 3.5.4 respectively.

- It was empirically found out that not all the segments are relevant for classification and that the optimal segment for classification is between 0 to 3000 Hz, which can be used as input to the network. This optimal segment is based on the sampling rate of 44100 Hz. With different sampling rates, the boundaries of the optimal segment should be revised. The (empirically proven) reason is that most (if not all) of the fingerprints of the microphone electronic components are in that frequency range.

The scheme of the adopted CNN architecture is shown in Figure 4-3. The optimized values of other hyper-parameters are reported in Figure 4-3 as well and are also given in Table 4.2. As CNN is usually applied to images, we converted the frequency representation of the digitized microphone recording just by reshaping the frequency vector to a matrix. The CNN network is made up of three convolutional layers followed by max pooling to reduce the size. All convolutional layers use the rectified linear unit (ReLU) as activation function. A softmax layer with as many units as the number of microphones to be identified (34 in our experiments) is attached. The softmax layer is aimed at producing the probability of each sample being classified into each class.

4.4 Results

This section presents the results on the application of CNN to the microphone classification and it is divided in two subsections.

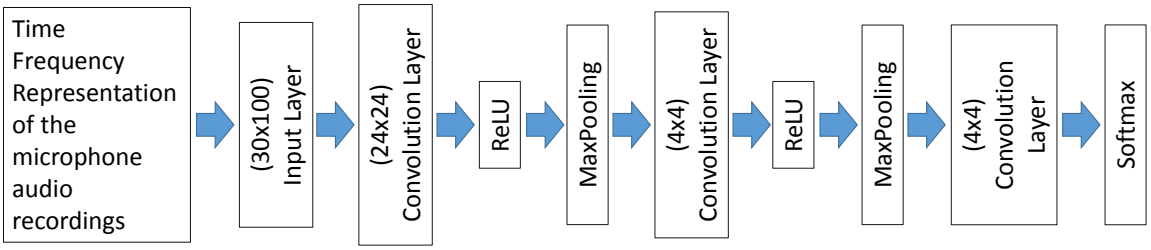


Figure 4-3: The proposed CNN architecture for microphone identification

The first subsection is specific for the optimization of the hyparameters, while the second presents the results in presence of AWGN.

4.4.1 Optimization of the hyperparameters for machine learning

The initial step was to optimize the hyparameters of the three machine learning algorithms used to perform the classification: CNN, SVM and KNN.

For all algorithms, a 4-fold approach was used for classification, where 25% of the dataset was used for test, and 75% was used for training and validation (9/10 of which used for training and 1/10 for validation, so that the validation set is 7.5% of the entire dataset). The overall classification process was then repeated 20 times, each time with different training and test sets. For each iteration, the hyperparameters were identified. Then, the final hyperparameter used for classification was the average of the identified optimal values or the function (e.g., solver) which scored the optimal classification results. Finally, all parameter optimizations were performed with $SNR = 50$ dB because this is the SNR level for a practical application of microphone identification.

The KNN algorithm was optimized using the single hyper-parameter K (in the range K=1 to 30) and the distance metrics (among Euclidean distance, Manhattan distance, Mahalanobis distance and Minkowski distance with d=3).

The SVM algorithm was optimized using the regularization parameter C, the kernel function between linear, polynomial and RBF and the associated parameters (e.g., γ in RBF) (see section 3.5.3 and the reference [67]).

The optimization of the proposed CNN was performed on the number of layers (already described in Figure 4-3), the solver used to train the network and the input size of the first convolutional layer, as these parameters were identified as those of statistical significance. The number of epochs for the CNN was limited to 100 as it was found out that this number of epochs is enough to make the algorithm converge. In other words, the epoch when the loss function on the validation set reaches its minimum is usual below 100.

The identified optimal parameters and functions (e.g., distance metric) for CNN, SVM and KNN are provided in Table 4.2.

Table 4.2: Hyper-parameters used for each machine learning algorithm.

Algorithm	**Hyperparameters and optimized values**
CNN	Solver RMSProp, First convolutional layer 24*24, Epochs=100
KNN	K=1 Euclidean Distance
SVM	RBF with C = 2^{12} and $\gamma = 2^6$

4.4.2 Classification accuracy in presence of Additive White Gaussian Noise

In this section, we present the results where we have progressively added AWGN to the original signals, causing decreasing values of SNR (from 50 dB down to -10 dB). In Figure 4-4 we show the performance accuracy for different values of SNR and we compare the CNN method with KNN and SVM. Figure 4-4 shows that the performance of the proposed CNN approach is better than KNN and SVM especially for lower values (less than 20 dB) of SNR. At $SNR = 10$ dB, it can be seen that CNN achieves a classification accuracy of 80% against a 40% of SVM and around 10% of KNN. This shows that CNN delivers a more robust classification method in the presence of noise. It is noted that SVM also provides a very high classification accuracy (which is consistent with the findings in [39] and [40]) for high SNR values (greater than 30 dB). This means that for security applications and scenarios where the presence of background noise is limited, a SVM based approach could also be

considered a valid alternative.

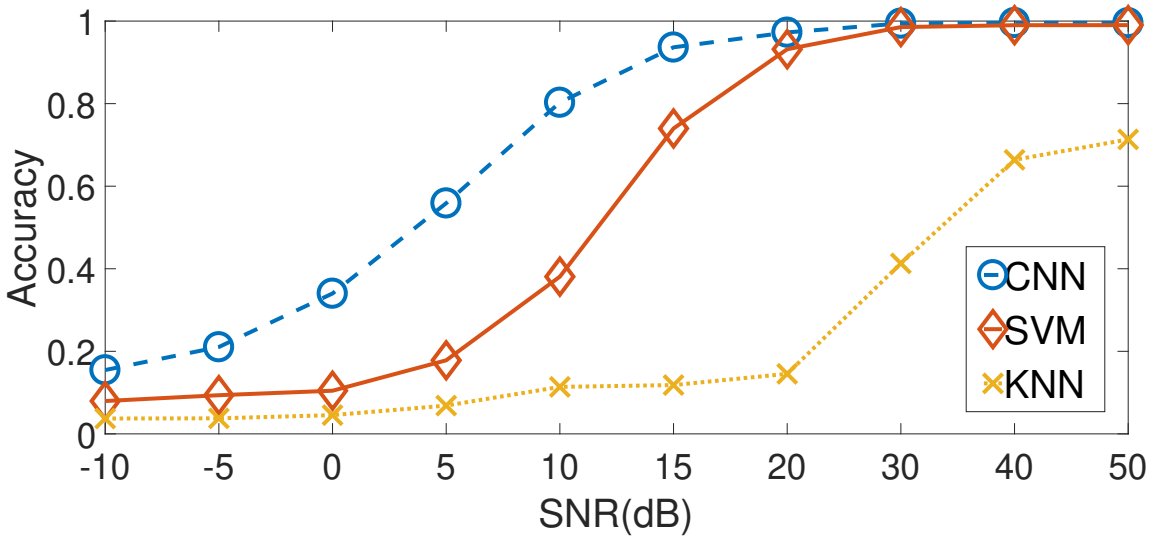


Figure 4-4: Accuracy on the dataset composed by 34 microphones. Comparison among CNN, KNN and SVM

Because accuracy only provides a limited view of the classification results, the results in Figure 4-4 are complemented by confusion matrices (in the form of heatmaps) for different values of the SNR in Figure 4-5. The Figure depicts the proposed predicted and true labels related to all the 34 evaluated microphones: a) CNN with $SNR = 15$ dB, b) CNN with $SNR = 0$ dB, c) SVM with $SNR = 15$ dB and d) KNN with $SNR = 15$ dB (results for SVM and KNN at $SNR = 0$ dB are not shown because the low classification accuracy does not provide meaningful information).

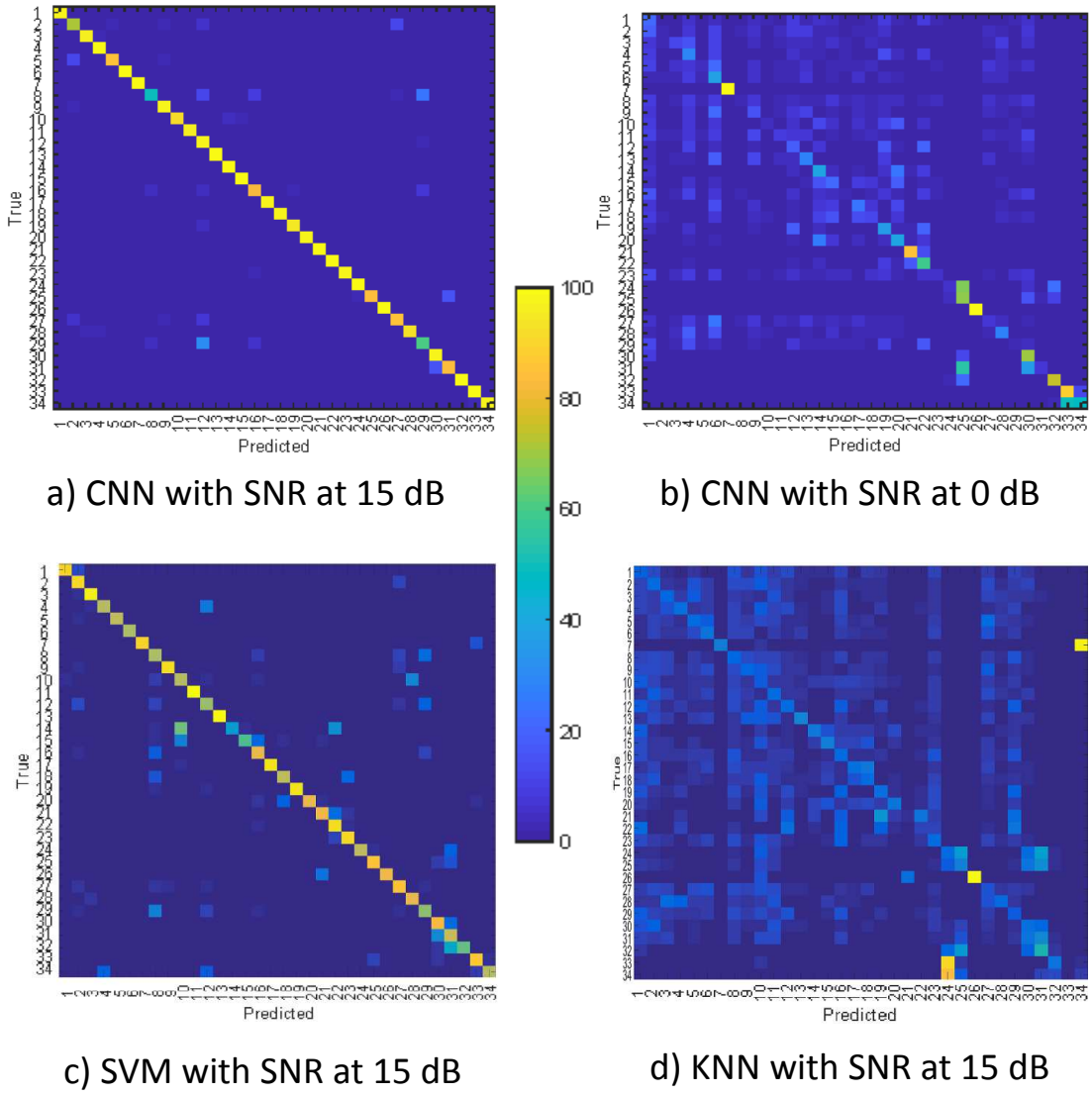


Figure 4-5: Confusion matrices on the 34 microphones obtained by the proposed CNN method at SNR = 15 dB and SNR = 0 dB (in a) and b) respectively). In the bottom row SVM and KNN results (in c) and d) respectively) at SNR =15 dB are reported for comparison. The heatmap values are percentages (e.g., 80 = 80%) of correct classifications for each class on the number of testing samples.

Chapter 5

Case Study of magnetometer identification

5.1 Overview

In this use case, it was investigated the identification of mobile phones through their built-in magnetometers. These electronic components have started to be widely deployed in mass market phones in recent years. As for other components of the mobile phone (e.g., camera or microphones), they can be exploited to uniquely identify mobile phones due their physical differences, which appear in the digital output generated by them. At the time of performing this research activity, there were no other reported attempts in the research literature to identify the mobile phone using the magnetometers. One of the possible reasons is because magnetometers started to appear only recently in mobile phones.

Even if there are some similarities with other types of sensors or components in the mobile phone, the magnetometers has some unique characteristics, which requires special configurations. In particular the magnetometer is characterized by an hysteris loop.

The exploitation of the magnetometer has an identification or authentication system can be described by the following figure 5-1, where a potential application scenario is shown.

In this scenario, the magnetometer in the mobile phone is stimulated by a specific magnetic field. The magnetic field stimulus is recorded by the mobile phone using the Application Programming Interface (API) (e.g., an android API) to access magnetometers recordings. An Android application, which performs the same function can also be used. In our experiments AndroSensor was used but any other android application can also be used. Then, the magnetic recordings are stored in a storage area for future comparison, where fingerprints can be extracted. In the identification/authentication phase (or testing phase from a machine learning point of view) other fingerprints are collected using the same stimulus and they are compared to the pre-stored fingerprints. As in the other cases of PLIA the objective is to determine the specific fingerprints with optimize the identification/authentication accuracy and minimize the classification processing time. As in the case of the microphones,

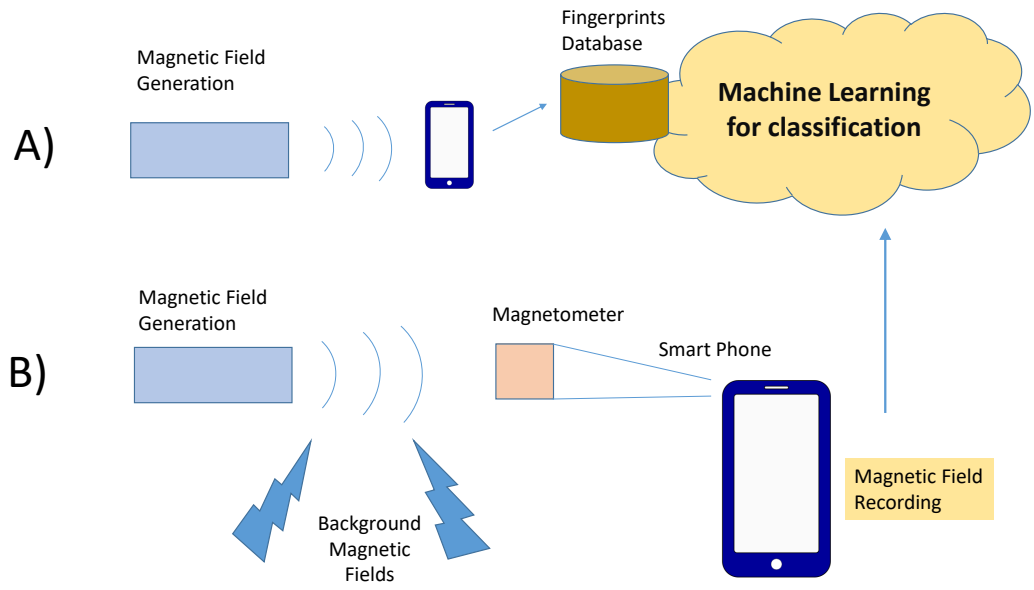


Figure 5-1: Application scenario for identification and authentication of mobile phones using the magnetometers

different stimuli can be used to generate the audio recordings and then embed the fingerprints. Then, it is possibile to create a wide space of challenges (the external stimuli)/responses (the fingerprints in the recordings), where the magnetometer (and then the mobile phone) can be uniquely identified by the fingerprints. An attacker would need to record the magnetometer stimuli and the response as well, which is quite difficult because: a) the magnetic field stimuli degrades significantly with the distance between the emitter and the magnetometer, b) the attacker would need to have access to the magnetic field recordings and know the process to extract the fingerprints (e.g., the type of statistical features used).

The advantage of using magnetometers for identification/authentication in comparison to other components in the mobile phone like the camera [42] or the RF wireless components is the possibility to control the stimulus, which is applied to the magnetometer from an external device. In this way it is easier to create an extensive challenge/response space. In this sense, the advantages are similar to the microphone described in the use case 4 with the additional advantage that a noisy environment from the acoustic point of view would not impact the magnetometers stimulus. A

related disadvantage is also true: the presence of specific magnetic fields (e.g., produced by an electric engine) can have an adverse effect. The magnetic field generated by the earth is relatively weak and stationary to significantly impact the magnetic field stimulus. The possibility to generate a large challenges/response space is more complex and difficult for the other components like a camera [42], where the recorded image can be totally random (based on the collected visual context) or the radio frequency fingerprints where the wireless standards may impose specific constraints.

There are different ways to stimulate the magnetometer. In this use case, the magnetometer was stimulated with a very simple movement of the mobile phone against a magnetic material. Materials and test bed set-up is described in subsection 5.2. The author has also investigated with other researchers the case where the magnetic stimulus was created with a specific magnetic field generator in [51], but in this thesis only the first case is described as it is simpler to implement (any magnetic object can be used and the stimulus can be created by sweeping the magnetometer in front of it). The output of the research of this use case has been published in [50].

5.2 Materials

Portions and figures of this section are extracted from the paper [50] published by the author in MDPI Sensor.

The test bed is made up of a platform which rotates the mobile phone under test, and a fixed magnet positioned on the edge of the rotating platform. When the mobile phone passes in front of the fixed magnet, the built-in magnetometer is stimulated and its digital output is recorded and analyzed. For each mobile phone, the experiment is repeated over six different days to ensure consistency in the results. A total of 10 phones of different brands and models or of the same model were used in our experiment. This test bed configuration has been created to simulate a practical scenario where an user can pass the mobile phone in front of a magnetic field. In this scenario, for simplicity the magnetic field is static and the stimuli is created by the motion of the mobile phone where the magnetometer is located.

A rotating platform is used for the definition of the motion pattern. The test setup is illustrated in Figure 5-2, where a mobile phone is installed on a cost-effective rotating platform and a magnetic element (an iron cube) is positioned at one extreme of the test bed. The rotating platform rotates the mobile phone with a specific motion pattern. The built-in magnetometer is stimulated by the magnetic element when it passes over it. The magnetic perturbation is collected and analyzed using an android application installed in the mobile phone. As described before, in this experiment, we have used the AndroSensor application but any other application, which can record the digital output from the magnetometer can be used. The application was configured to record the magnetometer digital output with a sampling time of 0.05s. The motion pattern used in our experiment is configured as follows: +120rpm then -120rpm for 4s, +150rpm then -150rpm for 3s, +180rpm then -180rpm for 2s. The mobile phone is kept for 60 seconds before the start of the motion pattern in a fixed position in front of the magnet. Each mobile phone was subject to this motion pattern. A total of 10 mobile phones were used in the experiments. Table 5.1 shows the brand and models of the phones used in the experiment. We note that three phones were of the same brand and model (i.e., HTC One) while the other phones were of different brands and models.

In each measurement campaign, each mobile phone is subject to 25 repetitions of the motion pattern. This experimental campaign was executed during six different days (even at the distance of a week), so as to ensure that the fingerprints are stable over time. As a consequence, we have a total of 25*6 = 150 motion patterns or responses.

5.3 Methodology

The process to collect the magnetic field recordings and extract the fingerprints is based on the following steps:

1. Each mobile phone is mounted on the test bed platform and submitted to the rotations as described in the Section 5.2.

Mobile phone model	Number of devices
Sony M4 Aqua	1
Huawei P8	1
Sony Xperia X	1
Samsung Galaxy S7	1
Huawei Mate 8	1
HTC One A9	1
LG G4	1
HTC One	3

Table 5.1: List of Mobile Phones used in our experiment.

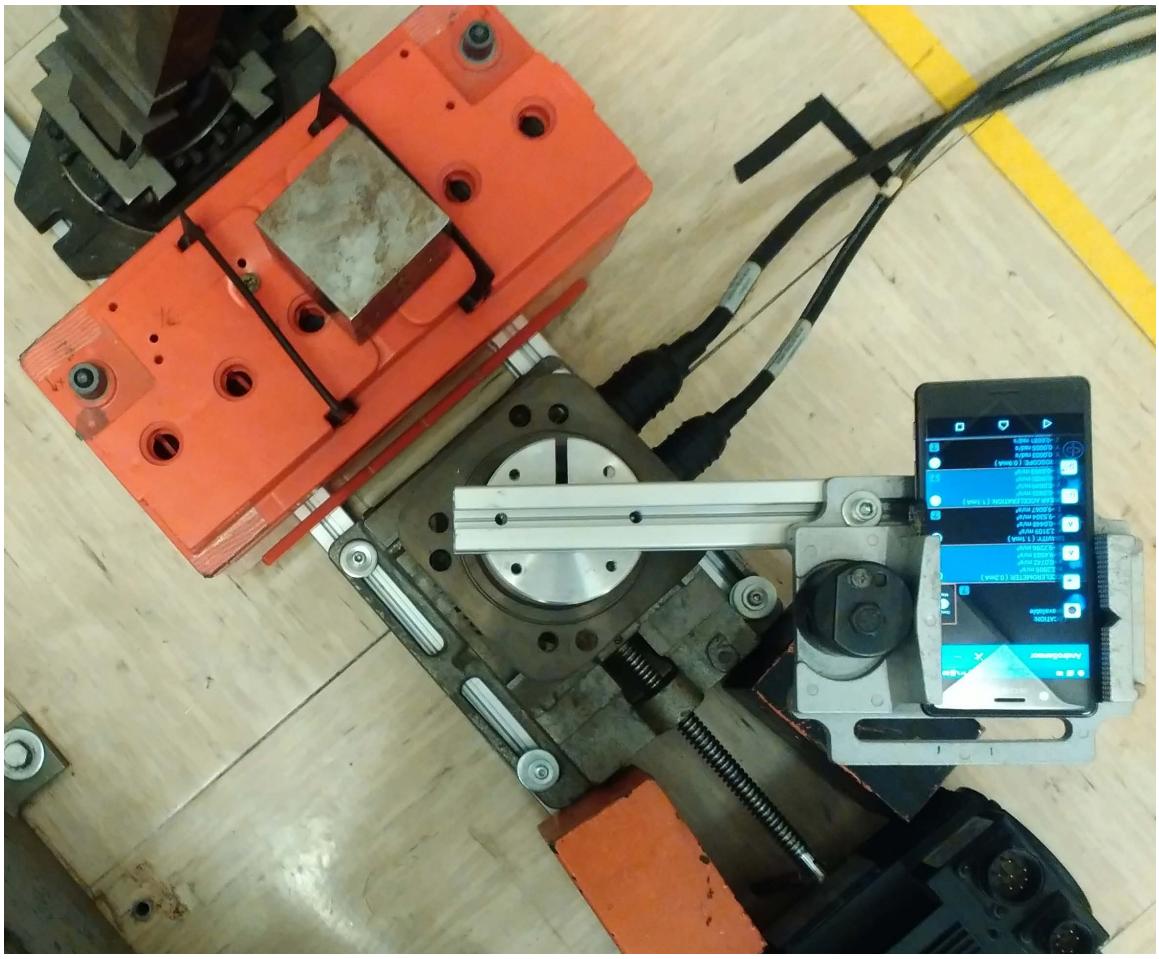


Figure 5-2: Image of the test setup used to collect the data.

2. Data is synchronized and normalized. This is an important step, since unsynchronized/unnormalized data can introduce a severe bias in the classification. After synchronization, the data are normalized. The normalization is carried out by applying the Root Mean Square (RMS) to each single response for each

individual mobile phone.

3. Statistical features are extracted from the magnetic recordings. The statistical features are based on similar features used in the literature for dimensionality reduction in PLIA. Table 5.2 shows the statistical features (the number represents the identifier of the feature) used in the experiment. These features are applied both in the time domain and the frequency domain after a FFT is applied. We note that another hyperparameter is the direction of the axis of the magnetometer, which is identified as X,Y and Z.

4. SVM machine learning algorithm is used to classify the mobile phones on the basis of the extracted statistical features (see Section 3.5.3 for a description of the SVM machine learning algorithm). Classification performance is evaluated using 3-fold cross validation. Each collection of statistical fingerprints (one for each mobile phone) is divided into three blocks, each having 50 fingerprints per block. Two blocks from each device are used for training and one block is *held out* for classification. The training and classification process is repeated three times until each of the three blocks has been *held out* and classified. Thus, each block of statistical fingerprints is used once for classification and twice for training. Final cross-validation performance statistics are calculated by averaging the results over all folds.

5. As described in Section 3.5.1, the optimization process was repeated for each of the three folds on the training set only. Finally, the overall process was repeated 50 times. While, this can be a time consuming process, it mitigates the risk of high variance in the results, and provides a good evaluation of the relevance of the statistical features. The results of the optimization are shown in Section 5.4.

Feature Name	Time Domain	Frequency domain (Phase)	Frequency domain (Amplitude)
Shannon Entropy	1	7	13
Log Energy Entropy	2	8	14
Variance	3	9	15
Standard Deviation	4	10	16
Skewness	5	11	17
Kurtosis	6	12	18

Table 5.2: Statistical features and the related identifier used for PLIA

5.4 Results

5.4.1 Optimization of the hyperparameters for machine learning

As described in Section 3.5.3 and from literature in [67], the SVM algorithm must be optimized on the C parameter (the so-called box constraint parameter), allowing the SVM user to control the weight of the classification errors during training, and the Kernel function, which is used to define the shape of the computed hyperplane. Various kernel functions are available in the literature including linear, polynomial and RBF. A comparison was performed among linear, polynomial and RBF and RBF provided the best results. As described in Section 3.5.3, the γ parameter in RBF must be optimized as well.

Various techniques can be used to optimize these values. In this paper, we adopt the grid approach with a set of exhaustive exponential values to the base 2, from 2^0 to 2^6 for the scaling factor γ, and from 2^0 to 2^{11} for the parameter C.

These values were calculated for the magnetomer in the X axis.

This process was repeated for all the 50 repetitions and the three folds. The final result of the SVM parameter optimization effort is shown in Figure 5-3 for parameter γ and in Figure 5-4 for parameter C. In the figures, the three different colors represent the three different folds.

On the basis of the selected features and the identified optimal values for C and

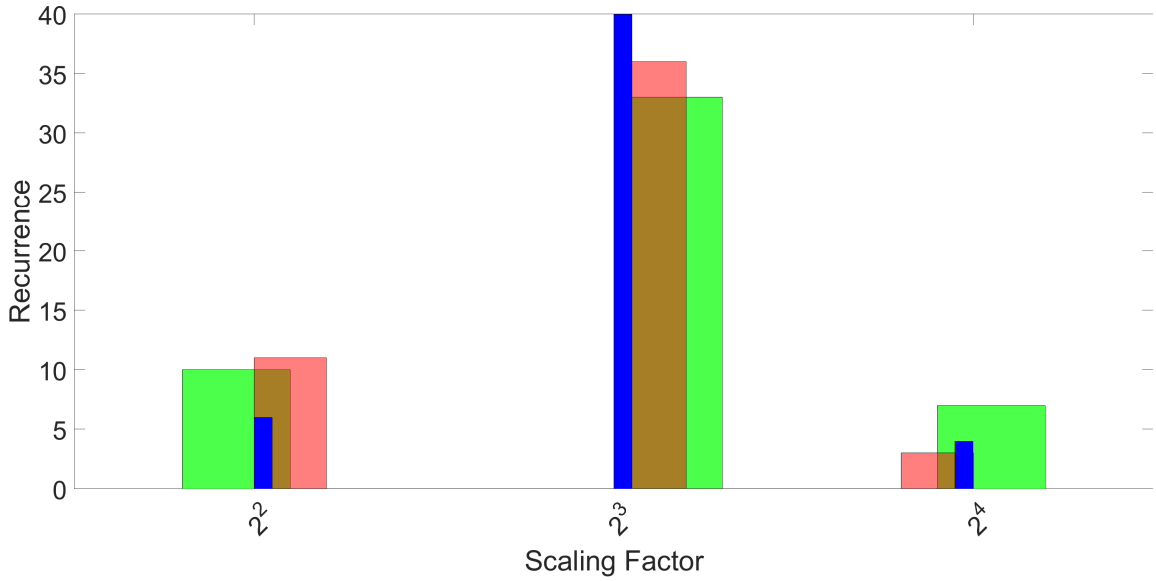


Figure 5-3: Histograms of the recurrence of the best performing scaling factor γ for 50 repetitions and three folds. The three colors represent the three folds.

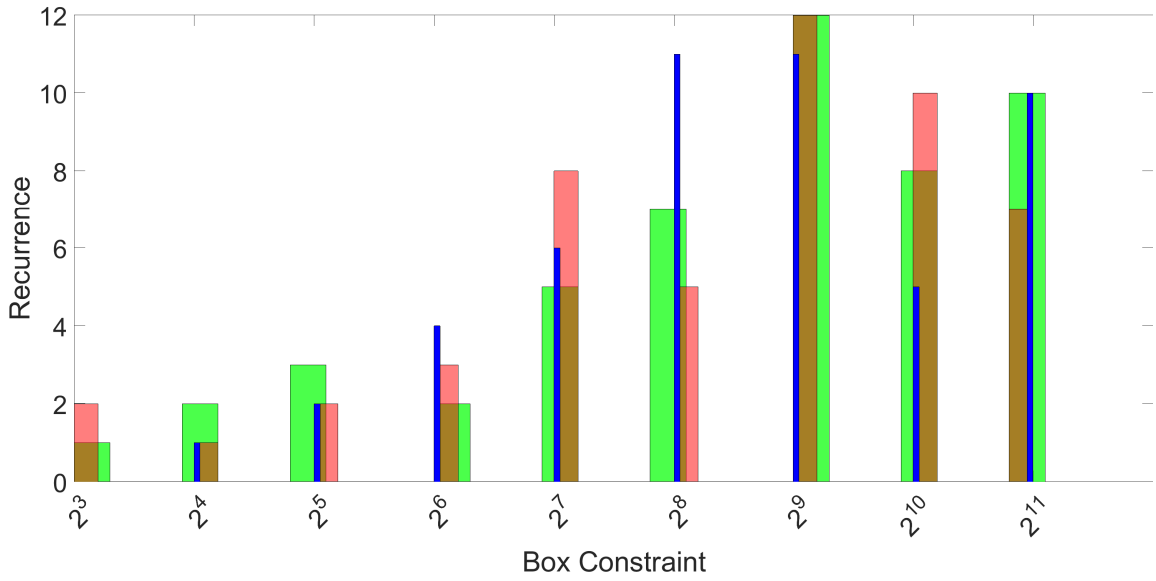


Figure 5-4: Histograms of the recurrence of the best performing box constraint parameter C for 50 repetitions and three folds. The three colors represent the three folds.

γ, the machine learning classification with SMV was performed and the results are provided in the next subsection.

5.4.2 Classification results

It was investigated both the identification scenario where it is estimated the probability of identifying one phone among others and the authentication scenario where a phone is compared against another phone.

Both SVM and KNN have been used to perform the classification and the results are presented in table 5.3.

Machine Learning algorithm	**Overall Accuracy**
SVM (X direction) , $\gamma = 2^3$ and $C = 2^9$	70.61 %
KNN with Number Neighbours = 3 (optimal value in a range from 1 to 10)	66.39 %

Table 5.3: Comparison among the different machine learning algorithms for the digital output generated by the Magnetometer on the X axis. Accuracy values have been averaged over the 50 repetitions.

As it can be seen, SVM provides a better identification accuracy in comparison to KNN, but it is relatively low in comparison to the results obtained in the other use cases (microphones and radio frequency fingerprints). This may due to the simpler design of the magnetometers in comparison to the microphones and radio frequency components, which allow the generation of less complex and distinguishable intrinsic features.

It was investigated if inter-model accuracy is higher than intra-model accuracy as it should be expected because the components are more distinguishable among models.

The inter-model accuracy is calculated as the average classification accuracy when including only one HTC mobile phone (i.e., phone identifiers from 1 to 8). The intra-model accuracy is computed when operating only with the data set of three HTC mobile phones (i.e., phone identifiers from 8 to 10).

Table 5.4 shows the identification for inter-model and intra-model for the three different axis using SVM using the optimal values identified in the previous section.

Inter-Model	
Magnetometor Axis	Overall Accuracy
X-axis	82.52 %
Y-axis	89.35 %
Z-axis	94.32 %
Intra-Model	
Magnetometor Axis	Overall Accuracy
X-axis	48.37 %
Y-axis	48.10 %
Z-axis	53.5 %

Table 5.4: Average overall accuracy for inter-model and intra-model classification using SVM for different axis of the magnetometer.

Even if a simplification is applied because, the optimization was conducted only for the X axis, it can be justified by the consideration that the intrinsic features (i.e., fingerprints) in the magnetomer are always physically the same for all the three axis.

From this result, it can be seen that the inter-model classification is quite high and it can be used for a practical application of the mobile phone identification using magnetometers when the phone are of different brands and models.

Then, it was evaluated the possibility to combine the signals from the three different magnetometers axis to obtain an higher classification accuracy.

By combining all the three axis together, it was obtained a resulting overall identification accuracy of 85.08 %, with an inter-model accuracy of 98.07 %, and an intra-model accuracy of 54.15 %. These accuracy values are higher than considering each axis in isolation. Specifically, there is a significant improvement (almost 4%) for inter-model accuracy, as compared to the best result of the single axis (magnetometer in the Z direction — see Table 5.4), and a slight improvement in intra-model accuracy.

The results for identification accuracy can also be confirmed by performing binary classification (i.e., authentication) separating two phones of different models (inter-model authentication) and two phones of the same model (intra-model authentication).

Figure 5-5 shows the binary classification between one Sony Experia X and one

Samsung Galaxy S7 where it can be seen that an high classification accuracy is obtained for inter-model scenario. In particular, the Y axis provides the best authentication accuracy among all the three axis. On the other side of the coin, the combination of all axis provides an even higher authentication accuracy (and consequently a lower EER), which confirms the previous results for identification accuracy.

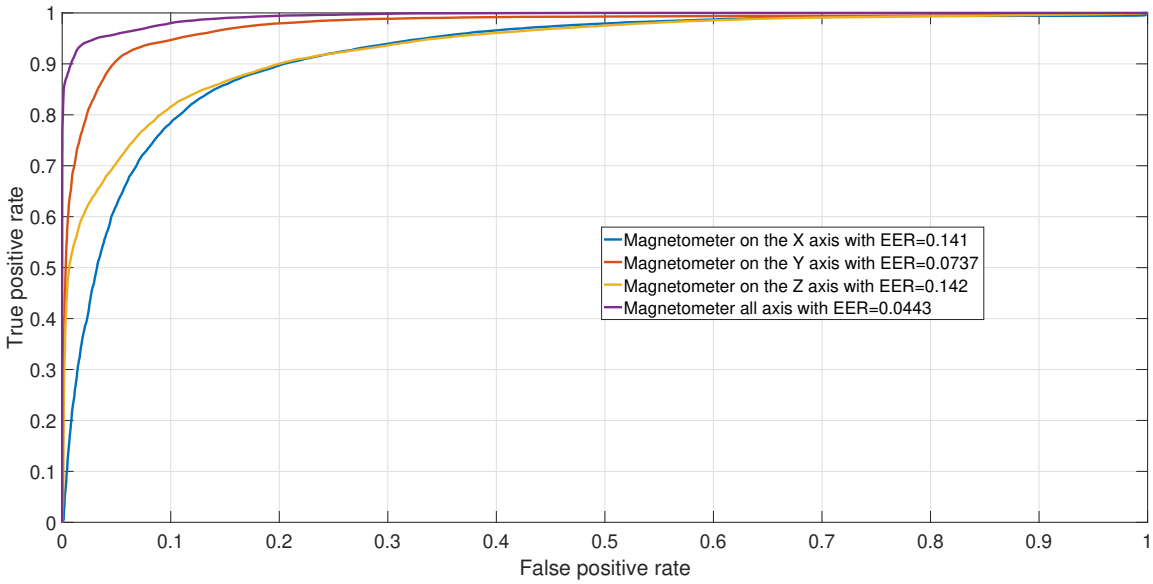


Figure 5-5: ROC achieved by SVM in binary classification between Sony Experia X and Samsung Galaxy S7. Results have been averaged over the 50 repetitions.

Figure 5-6 shows the binary classification between one HTC One 2 and one HTC One 3 where it can be seen that the classification accuracy is lower for the intra-model case than the inter-model case as expected. In this case, the combination of all axis does not provide a significant gain in comparison to the results for each specific axis. This result is also consistent with the previous result for the identification accuracy, where the combination of all axis did not provide a significant improvement in comparison to the identification obtained for each axis.

In a practical application of mobile phone identification based on the fingerprints of the built-in magnetometers, it is well possible that the distance between the mobile phone and the magnetic element stimulating the magnetometer can vary. Changes in distance and orientation will definitely impact the SNR. Different distances and different values of SNR can be simulated by adding AWGN to the collected magne-

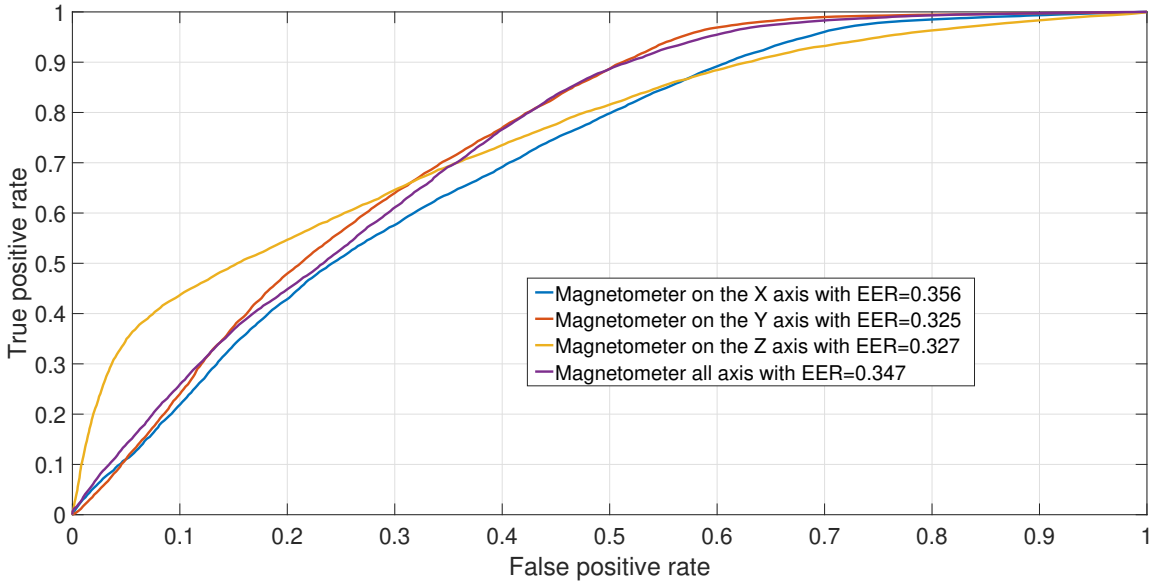


Figure 5-6: ROC achieved by SVM in binary classification between HTC One 2 and HTC One 3. Results have been averaged over the 50 repetitions.

tometers responses.

Figure 5-7 shows the ROCs for binary classification between Sony Xperia X and Samsung Galaxy S7 for decreasing values of SNR. The associated value of the EER is shown in the legend. As expected, a low value of SNR results in almost random choice identification (e.g., the green curve) because the machine learning algorithm is not able to leverage very noisy signals.

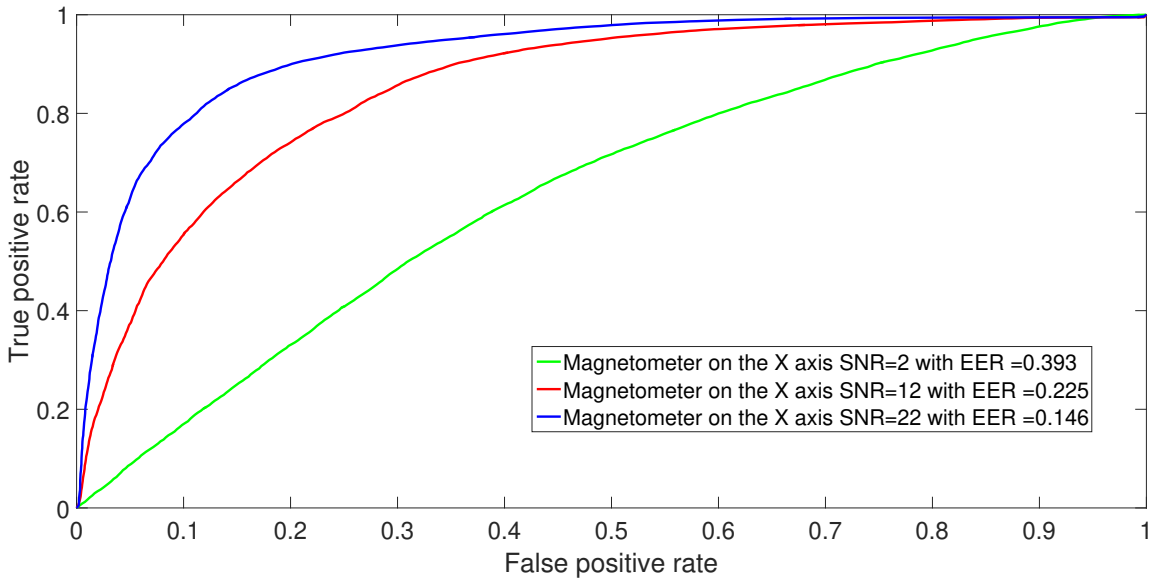


Figure 5-7: ROC achieved by SVM in binary classification between Sony Experia X and Samsung Galaxy S7 using the X-axis for decreasing values of SNR. Again, these curves are obtained after averaging over 50 repetitions.

Chapter 6

Conclusions

This thesis has investigated the physical layer authentication of electronic devices for three different types of electronic devices using different techniques both at the level of signal processing and machine learning. In all cases, only supervised learning has been investigated as this area alone presented numerous topics for novel investigation in comparison to literature. An extensive literature review has been conducted and specific gaps have been identified to propose novel results in different areas including the analysis of the bias introduced by the system (e.g., RF receiver) collecting the signal, the novel combination of specific signal processing techniques together with Convolutional Neural Networks and the application of physical layer authentication to new types of electronic devices like the magnetometers. The results of the PhD activities have been proposed and accepted for publication in journals in the Science Citation Index Extended (SCI-E) and IEEE conferences.

Bibliography

[1] A. Ometov, S. Bezzateev, N. Mäkitalo, S. Andreev, T. Mikkonen, and Y. Koucheryavy, "Multi-factor authentication: A survey," *Cryptography*, vol. 2, no. 1, p. 1, 2018.

[2] M. Masdari and S. Ahmadzadeh, "A survey and taxonomy of the authentication schemes in telecare medicine information systems," *Journal of Network and Computer Applications*, vol. 87, pp. 1–19, 2017.

[3] M. D. Williams, M. A. Temple, and D. R. Reising, "Augmenting bit-level network security using physical layer rf-dna fingerprinting," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pp. 1–6, IEEE, 2010.

[4] G. Baldini and G. Steri, "A survey of techniques for the identification of mobile phones using the physical fingerprints of the built-in components," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1761–1789, 2017.

[5] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Communications Surveys Tutorials*, vol. 18, pp. 94–104, Firstquarter 2016.

[6] D. Borio, C. Gioia, G. Baldini, and J. Fortuny, "Gnss receiver fingerprinting for security-enhanced applications," in *Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+), Portland, OR, USA*, pp. 12–16, 2016.

[7] S. U. Rehman, K. W. Sowerby, and C. Coghill, "Radio-frequency fingerprinting for mitigating primary user emulation attack in low-end cognitive radios," *IET Communications*, vol. 8, no. 8, pp. 1274–1284, 2014.

[8] U. Guin, K. Huang, D. DiMase, J. Carulli, M. Tehranipoor, and Y. Makris, "Counterfeit integrated circuits: A rising threat in the global semiconductor supply chain," *Proceedings of the IEEE*, vol. 102, pp. 1207–1228, Aug 2014.

[9] M. Lukacs, P. Collins, and M. Temple, "Device identification using active noise interrogation and rf-dna" fingerprinting" for non-destructive amplifier acceptance testing," in *2016 IEEE 17th Annual Wireless and Microwave Technology Conference (WAMICON)*, pp. 1–6, IEEE, 2016.

[10] G. Baldini, R. Giuliani, and E. Cano Pons, "An analysis of the privacy threat in vehicular ad hoc networks due to radio frequency fingerprinting," *Mobile Information Systems*, vol. 2017, 2017.

[11] J. Zhang, F. Wang, O. A. Dobre, and Z. Zhong, "Specific emitter identification via hilbert-huang transform in single-hop and relaying scenarios," *IEEE Transactions on Information Forensics and Security*, vol. 11, pp. 1192–1205, June 2016.

[12] S. Dolatshahi, A. Polak, and D. L. Goeckel, "Identification of wireless users via power amplifier imperfections," in *2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*, pp. 1553–1557, Nov 2010.

[13] D. R. Reising, M. A. Temple, and M. E. Oxley, "Gabor-based RF-DNA fingerprinting for classifying 802.16e WiMAX mobile subscribers," in *Computing, Networking and Communications (ICNC), 2012 International Conference on*, pp. 7–13, Jan 2012.

[14] W. C. Suski II, M. A. Temple, M. J. Mendenhall, and R. F. Mills, "Using spectral fingerprints to improve wireless network security," in *IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference*, pp. 1–5, IEEE, 2008.

[15] T. J. Bihl, K. W. Bauer, M. A. Temple, and B. Ramsey, "Dimensional reduction analysis for physical layer device fingerprints with application to zigbee and z-wave devices," in *Military Communications Conference, MILCOM 2015 - 2015 IEEE*, pp. 360–365, Oct 2015.

[16] D. R. Reising, M. A. Temple, and M. J. Mendenhall, "Improved wireless security for gmsk-based devices using rf fingerprinting," *International Journal of Electronic Security and Digital Forensics*, vol. 3, no. 1, pp. 41–59, 2010.

[17] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Communications Surveys Tutorials*, vol. 18, pp. 94–104, Firstquarter 2016.

[18] P. Scanlon, I. O. Kennedy, and Y. Liu, "Feature extraction approaches to RF fingerprinting for device identification in femtocells," *Bell Labs Technical Journal*, vol. 15, pp. 141–151, Dec 2010.

[19] G. Baldini, G. Steri, R. Giuliani, and F. Dimc, "Radiometric identification using variational mode decomposition," *Computers & Electrical Engineering*, vol. 76, pp. 364–378, 2019.

[20] K. Merchant, S. Revay, G. Stantchev, and B. Nousain, "Deep learning for RF Device Fingerprinting in Cognitive Communication Networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, pp. 160–167, Feb 2018.

[21] S.-C. B. Lo, H.-P. Chan, J.-S. Lin, H. Li, M. T. Freedman, and S. K. Mun, "Artificial convolution neural network for medical image pattern recognition," *Neural networks*, vol. 8, no. 7-8, pp. 1201–1214, 1995.

[22] M. D. Williams, S. A. Munns, M. A. Temple, and M. J. Mendenhall, "Rf-dna fingerprinting for airport wimax communications security," in *Network and System Security (NSS), 2010 4th International Conference on*, pp. 32–39, Sept 2010.

[23] D. R. Reising, M. A. Temple, and J. A. Jackson, "Authorized and rogue device discrimination using dimensionally reduced rf-dna fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 10, pp. 1180–1192, June 2015.

[24] G. Baldini, C. Gentile, R. Giuliani, and G. Steri, "Comparison of techniques for radiometric identification based on deep convolutional neural networks," *Electronics Letters*, vol. 55, no. 2, pp. 90–92, 2018.

[25] S. U. Rehman, K. W. Sowerby, S. Alam, and I. Ardekani, "Portability of an rf fingerprint of a wireless transmitter," in *Communications and Network Security (CNS), 2014 IEEE Conference on*, pp. 151–156, Oct 2014.

[26] H. Patel, M. A. Temple, and B. W. Ramsey, "Comparison of high-end and low-end receivers for RF-DNA fingerprinting," in *2014 IEEE Military Communications Conference*, pp. 24–29, Oct 2014.

[27] G. Baldini, R. Giuliani, C. Gentile, and G. Steri, "Measures to address the lack of portability of the rf fingerprints for radiometric identification," in *New Technologies, Mobility and Security (NTMS), 2018 9th IFIP International Conference on*, pp. 1–5, IEEE, 2018.

[28] M. Mailand, R. Richter, and H.-J. Jentschel, "IQ-imbalance and its compensation for non-ideal analog receivers comprising frequency-selective components," *Advances in Radio Science*, vol. 4, no. D. 1, pp. 189–195, 2006.

[29] L. J. Wong, W. C. Headley, and A. J. Michaels, "Emitter identification using cnn iq imbalance estimators," *arXiv preprint arXiv:1808.02369*, 2018.

[30] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "Oracle: Optimized radio classification through convolutional neural networks," *arXiv preprint arXiv:1812.01124*, 2018.

[31] G. Baldini, R. Giuliani, and C. Gentile, "An assessment of the impact of iq imbalances on the physical layer authentication of iot wireless devices," in *2019 Global IoT Summit (GIoTS)*, pp. 1–6, IEEE, 2019.

[32] C. Kraetzer, A. Oermann, J. Dittmann, and A. Lang, "Digital audio forensics: a first practical evaluation on microphone and environment classification," in *Proceedings of the 9th workshop on Multimedia & security*, pp. 63–74, ACM, 2007.

[33] C. Kraetzer, M. Schott, and J. Dittmann, "Unweighted fusion in microphone forensics using a decision tree and linear logistic regression models," in *Proceedings of the 11th ACM workshop on Multimedia and security*, pp. 49–56, ACM, 2009.

[34] A. Das, N. Borisov, and M. Caesar, "Do you hear what i hear?: Fingerprinting smart devices through embedded acoustic components," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 441–452, ACM, 2014.

[35] R. Aggarwal, S. Singh, A. K. Roul, and N. Khanna, "Cellphone identification using noise estimates from recorded audio," in *2014 International Conference on Communication and Signal Processing*, pp. 1218–1222, April 2014.

[36] V. Pandey, V. K. Verma, and N. Khanna, "Cell-phone identification from audio recordings using PSD of speech-free regions," in *Electrical, Electronics and Computer Science (SCEECS), 2014 IEEE Students' Conference on*, pp. 1–6, IEEE, 2014.

[37] C. Hanilçi and T. Kinnunen, "Source cell-phone recognition from recorded speech using non-speech segments," *Digital Signal Processing*, vol. 35, pp. 75–85, 2014.

[38] L. Zou, Q. He, and J. Wu, "Source cell phone verification from speech recordings using sparse representation," *Digital Signal Processing*, vol. 62, pp. 125 – 136, 2017.

[39] D. Luo, P. Korus, and J. Huang, "Band energy difference for source attribution in audio forensics," *IEEE Transactions on Information Forensics and Security*, vol. 13, pp. 2179–2189, Sep. 2018.

[40] D. Chen, N. Zhang, Z. Qin, X. Mao, Z. Qin, X. Shen, and X. Li, "S2M: A lightweight acoustic fingerprints-based wireless device authentication protocol," *IEEE Internet of Things Journal*, vol. 4, pp. 88–100, Feb 2017.

[41] G. Baldini, I. Amerini, and C. Gentile, "Microphone identification using convolutional neural networks," *IEEE Sensors Letters*, 2019.

[42] J. Lukas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, pp. 205–214, June 2006.

[43] V. K. Khanna, "Remote fingerprinting of mobile phones," *IEEE Wireless Communications*, vol. 22, pp. 106–113, December 2015.

[44] R. Buchholz, C. Kraetzer, and J. Dittmann, "Microphone classification using fourier coefficients," in *Information Hiding*, pp. 235–246, Springer, 2009.

[45] C. Hanilci, F. Ertas, T. Ertas, and . Eskidere, "Recognition of brand and models of cell-phones from recorded speech signals," *IEEE Transactions on Information Forensics and Security*, vol. 7, pp. 625–634, April 2012.

[46] G. Baldini, G. Steri, F. Dimc, R. Giuliani, and R. Kamnik, "Experimental identification of smartphones using fingerprints of built-in micro-electro mechanical systems (mems)," *Sensors*, vol. 16, no. 6, p. 818, 2016.

[47] T. Van Goethem, W. Scheepers, D. Preuveneers, and W. Joosen, "Accelerometer-based device fingerprinting for multi-factor mobile authentication," in *Engineering Secure Software and Systems*, pp. 106–121, Springer, 2016.

[48] A. Serra, D. Carboni, and V. Marotto, "Indoor pedestrian navigation system using a modern smartphone," in *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, pp. 397–398, ACM, 2010.

[49] R. Jin, L. Shi, K. Zeng, A. Pande, and P. Mohapatra, "Magpairing: Pairing smartphones in close proximity using magnetometers," *IEEE Transactions on information forensics and security*, vol. 11, no. 6, pp. 1306–1320, 2015.

[50] G. Baldini, F. Dimc, R. Kamnik, G. Steri, R. Giuliani, and C. Gentile, "Identification of mobile phones using the built-in magnetometers stimulated by motion patterns," *Sensors*, vol. 17, no. 4, p. 783, 2017.

[51] G. Baldini, G. Steri, I. Amerini, and R. Caldelli, "The identification of mobile phones through the fingerprints of their built-in magnetometer: An analysis of the portability of the fingerprints," in *2017 International Carnahan Conference on Security Technology (ICCST)*, pp. 1–6, IEEE, 2017.

[52] I. Amerini, R. Becarelli, R. Caldelli, A. Melani, and M. Niccolai, "Smartphone fingerprinting combining features of on-board sensors," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 10, pp. 2457–2466, 2017.

[53] B. Perez, M. Musolesi, and G. Stringhini, "Fatal attraction: identifying mobile devices through electromagnetic emissions," in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, pp. 163–173, ACM, 2019.

[54] E. Research, "USRP N200 N210 Networked series specifications," 2016. [Online accessed 22-December-2016].

[55] C. K. Dubendorfer, B. W. Ramsey, and M. A. Temple, "An rf-dna verification process for zigbee networks," in *MILITARY COMMUNICATIONS CONFERENCE, 2012-MILCOM 2012*, pp. 1–6, IEEE, 2012.

[56] M. W. Lukacs, A. J. Zeqolari, P. J. Collins, and M. A. Temple, "âĂIJrf-dnaâĂİ fingerprinting for antenna classification," *IEEE Antennas and Wireless Propagation Letters*, vol. 14, pp. 1455–1458, 2015.

[57] T. J. Bihl, K. W. Bauer, and M. A. Temple, "Feature selection for rf fingerprinting with multiple discriminant analysis and using zigbee device emissions," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 8, pp. 1862–1874, 2016.

[58] H. Yang, "Multiscale recurrence quantification analysis of spatial cardiac vectorcardiogram signals," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 2, pp. 339–347, 2010.

[59] G. Yu and Y. Zhou, "General linear chirplet transform," *Mechanical Systems and Signal Processing*, vol. 70-71, pp. 958 – 973, 2016.

[60] S. Mann and S. Haykin, "The chirplet transform: physical considerations," *IEEE Transactions on Signal Processing*, vol. 43, pp. 2745–2761, Nov 1995.

[61] H. K. Kwok and D. L. Jones, "Improved instantaneous frequency estimation using an adaptive short-time fourier transform," *IEEE Transactions on Signal Processing*, vol. 48, pp. 2964–2972, Oct 2000.

[62] D. Stutz, "Understanding convolutional neural networks," 2014.

[63] K. Jarrett, K. Kavukcuoglu, Y. LeCun, *et al.*, "What is the best multi-stage architecture for object recognition?," in *2009 IEEE 12th international conference on computer vision*, pp. 2146–2153, IEEE, 2009.

[64] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[65] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of relieff and rrelieff," *Machine learning*, vol. 53, no. 1-2, pp. 23–69, 2003.

[66] G. Baldini, I. Amerini, and C. Gentile, "Microphone identification using convolutional neural networks," *IEEE Sensors Letters*, pp. 1–1, 2019.

[67] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.