PAST-AI: Physical-Layer Authentication of Satellite Transmitters via Deep Learning

Gabriele Oligeri, Savio Sciancalepore, Simone Raponi, and Roberto Di Pietro, *Senior Member, IEEE*

***Abstract*— Physical-layer security is regaining traction in the research community, due to the performance boost introduced by deep learning classification algorithms. This is particularly true for sender authentication in wireless communications via radio fingerprinting. However, previous research mainly focused on terrestrial wireless devices while, to the best of our knowledge, none of the previous work considered satellite transmitters. The satellite scenario is generally challenging because, among others, satellite radio transducers feature non-standard electronics (usually aged and specifically designed for harsh conditions). Moreover, the fingerprinting task is specifically difficult for Low-Earth Orbit (LEO) satellites (like the ones we focus in this paper) since they feature a low bit-rate and orbit at about 800 Km from the Earth, at a speed of around 25, 000 Km/h, thus making the receiver experiencing a down-link with unique attenuation and fading characteristics. In this paper, we investigate the effectiveness and main limitations of AI-based solutions to the physical-layer authentication of LEO satellites. Our study is performed on massive real data—more than 100*M* I-Q samples—collected from an extensive measurements campaign on the IRIDIUM LEO satellites constellation, lasting 589 hours. Our results show that Convolutional Neural Networks (CNN) and autoencoders (if properly calibrated) can be successfully adopted to authenticate the satellite transducers, with an accuracy spanning between 0.8 and 1, depending on prior assumptions. However, the relatively high number of I-Q samples required by the proposed methodology, coupled with the low bandwidth of satellite link, might prevent the detection of the spoofing attack under certain configuration parameters.**

***Index Terms*— Physical-layer security, satellite systems security, applications of artificial intelligence for security, wireless security.**

I. Introduction

PHYSICAL-LAYER authentication relies on detecting and identifying unique characteristics embedded in over-the-air radio signals, thus enabling the identification of the hardware of the transmitting source [1], [2]. Wireless Physical-layer authentication is also known as radio fingerprinting when referring to the challenge of both detecting and extracting features from the received signal (fingerprint), which can uniquely identify the transmitting source [3], [4].

Physical-layer authentication can significantly enhance the security and privacy of wireless channels in two adversarial scenarios: (i) spoofing; and, (ii) replay attacks. The former involves a rogue transmitting source attempting to impersonate a legitimate one, while the latter assumes the adversary being able to re-transmit previously eavesdropped messages [5]. Despite spoofing detection can be achieved by authenticating the transmitting source with standard cryptographic techniques (e.g., digital signatures), in many scenarios involving massive deployments (e.g., IoT), difficult to reach devices (e.g., satellites), or when the cryptography-induced overhead is considered excessive, digital signatures might be inefficient [6]. Alternative solutions could involve crowdsourcing, i.e., cross-checking context information to validate the transmitting source [7], [8]. Replay attacks can be even more difficult to detect, being dependent on specific protocol flaws: the adversary re-transmits encrypted information, which will be considered as valid if not timestamped. Both spoofing and replay attacks can be prevented if the receiver can authenticate the hardware of the transmitting source [9].

Many researchers have already undertaken the challenge of extracting unique fingerprints and developing effective detection algorithms to extract and match the fingerprints (see Sec. VII for an overview). The cited tasks have been mainly achieved by resorting to dedicated hardware at the receiver side, featuring high sampling resolution and better signal quality. Indeed, Software-Defined Radios (SDRs) played a major role as an enabling technology for radio fingerprinting. Specifically, SDRs provide both high-resolution bandwidth (thus exposing the features of the transmitting source) and high signal-to-noise ratio (thus facilitating the extraction of the features to the back-end algorithms). Unfortunately, radio noise still represents the major issue for all the state-of-the-art solutions. Indeed, the fingerprint of the transmitting source is mixed—drown, in many cases—with the noise of the radio channel. Therefore, discriminating between the needed features and the noise brings back the problem of developing effective algorithms to achieve the cited objective.

Recently, Convolutional Neural Networks (CNNs) have been used for radio fingerprinting in several scenarios, such as ADS-B, WiFi, and Zigbee [10], [11], [12], [13]. The idea behind the adoption of CNNs relies on exploiting their multidimensional mapping during the learning process to detect

Manuscript received 28 February 2022; revised 5 July 2022 and 19 September 2022; accepted 20 October 2022. Date of publication 3 November 2022; date of current version 7 December 2022. This work was supported by the National Priority Research Program (NPRP) Grants from the Qatar National Research Fund (a member of Qatar Foundation) under Grant NPRP12S-0125-190013, Grant NPRP-S-11-0109-180242, and Grant NPRP X-063-1-014. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ashish Khisti. *(Corresponding author: Savio Sciancalepore.)*

Gabriele Oligeri, Simone Raponi, and Roberto Di Pietro are with the Division of Information and Computing Technology (ICT), College of Science and Engineering (CSE), Hamad Bin Khalifa University (HBKU), Doha, Qatar (e-mail: goligeri@hbku.edu.qa; sraponi@hbku.edu.qa; rdipietro@hbku.edu.qa).

Savio Sciancalepore is with the Eindhoven Artificial Intelligence Systems Institute (EAISI), Eindhoven University of Technology, 5612 AZ Eindhoven, The Netherlands (e-mail: s.sciancalepore@tue.nl).

Digital Object Identifier 10.1109/TIFS.2022.3219287

and extract reliable radio fingerprints. However, all of the recent contributions took into account terrestrial links, only. Recent results [13] based on real measurements on terrestrial wireless links confirmed that the wireless channel significantly impacts the classification accuracy (up to 80%), thus confirming the need for more effective classification techniques. It is worth noting that no prior contribution has been made up to date to the physical layer authentication of satellite transmitters (in particular, the IRIDIUM constellation), given their intrinsic challenges [14]. Indeed, LEO satellites, which IRIDIUM constellation is part of, are characterized by unique features: the satellite transmitter is at around 800 Km from the earth and moves at about 7 Km/s with a pass duration of about 8 minutes [8]—involving a radio link (quality) that significantly changes over the time. Indeed, we observe that attenuation and multi-path fading can significantly change when the satellite is either on top of the receiver or far away, just over the horizon (before disappearing). Therefore, the noise affecting the satellite link makes radio fingerprinting in satellite a unique, more challenging scenario, requiring additional research.

Contribution. We provide the following contributions:

- We introduce PAST-AI, i.e., a set of methodologies to perform radio fingerprinting over LEO satellite links.
- We prove that Convolutional Neural Network (CNN) and autoencoders can be effectively adopted to fingerprint radio satellite transmitters.
- We propose two different classification scenarios, i.e., *intra-constellation satellite authentication* and *satellite authentication in the wild*, which fit the adopted classification algorithm and their assumptions.
- We provide several insights to properly calibrate the algorithm parameters, achieving overwhelming performance, i.e., an accuracy greater than 0.8 for the former scenario and average Area Under the Curve (AUC) equal to 1 for the latter (i.e., the vast majority of the satellites).
- We compare the adopted neural network, i.e., *ResNet-18*, with other neural networks typically used for the same problem, showing the existing trade-off between classification accuracy and training overhead.
- We experimentally demonstrate the limitations of physical-layer authentication via radio fingerprinting in satellite networks, showing through real data the impact of the bandwidth and the received signal strength on satellite links.

Paper organization. The rest of this paper is organized as follows. Sec. II introduces background notions; Sec. III illustrates the data acquisition campaign and the initial data processing; Sec. IV introduces the PAST-AI methodology; Sec. V focuses on the intra-constellation satellite authentication scenario; Sec. VI details the authentication scenario with minimal satellites' knowledge; Sec. VII reviews related work; and, finally, Sec. VIII tightens the conclusions.

II. Background and Adversary Model

In this section, we describe both background notions and the assumed adversary model.

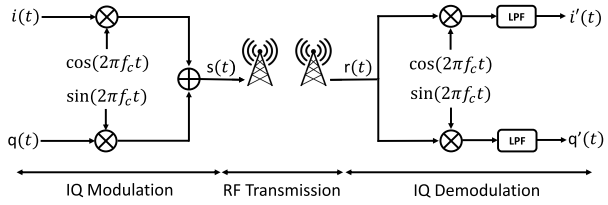


Fig. 1. Modulation and Demodulation of a digital signal represented by its phase $i(t)$ and quadrature $q(t)$ components.

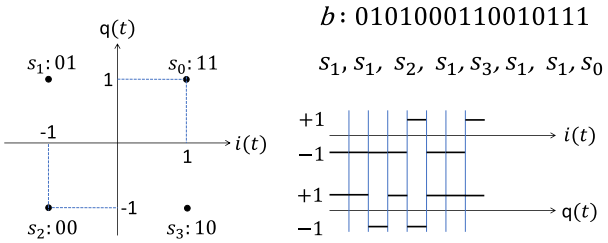


Fig. 2. QPSK modulation example: from bit sequence b to the in-phase $i(t)$ and quadrature $q(t)$ components.

A. I-Q (de)Modulation

Digital modulation schemes involve the processing of a (low frequency) baseband signal, i.e., a bit sequence $b_i \in \{0, 1\}$ with $i \in [1, N]$, to make it suitable for transmission in the RF spectrum (high frequency). While several techniques to achieve the aforementioned result are available, *I-Q modulation* is the most adopted due to practicality: efficient I-Q (de)modulators are available as inexpensive System on Chips (SoC). Fig. 1 shows the block diagram of a typical communication system involving I-Q modulation, RF transmission, and I-Q demodulation. A sequence of bits should be preliminary converted into *I-Q symbols*, i.e., $i(t)$ and $q(t)$ in Fig. 1. Different families of modulation schemes are possible, e.g., Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), or Phase Shift Keying (PSK), depending on how the sequence of bits is converted to the *in-phase* $i(t)$ and *quadrature* $q(t)$ components (recall Fig. 1).

As a toy example, we consider the Quadrature Phase Shift Keying (QPSK)—very similar to the one adopted by Iridium. QPSK maps pair of bits into (four) I-Q symbols, i.e, $\{1, 1\} \rightarrow s_0$, $\{0, 1\} \rightarrow s_1$, $\{0, 0\} \rightarrow s_2$, and $\{1, 0\} \rightarrow s_3$, as in Fig. 2. Note that the aforementioned mapping can be easily achieved by setting $i(t) = \{-1, 1\}$ and $q(t) = \{-1, 1\}$, as depicted in Fig. 2. For instance, the bit string $b : [0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1]$ becomes the sequence of symbols $[s_1, s_1, s_2, s_1, s_3, s_1, s_1, s_0]$, thus obtaining the in-phase $i(t)$ and quadrature $q(t)$ signal components. For the sake of completeness, we highlight that both $i(t)$ and $q(t)$ should be subject to other filtering stages and they cannot be directly used as mentioned in Fig. 1, since the sharp level changes will eventually cause $s(t)$ to have a very large bandwidth [15].

$i(t)$ and $q(t)$ components are *modulated* adopting an in-phase $(\cos 2\pi f_c t)$ and a quadrature $(\sin 2\pi f_c t)$ signal at the reference frequency f_c (*carrier*). The resulting signals are summed up to obtain the actual RF signal $s(t)$. Fig. 1 takes

into account also propagation phenomena, such as fading and attenuation, that may affect the received signal, and therefore $r(t) \neq s(t)$. The demodulation block is the reciprocal of the modulator: the received signal $r(t)$ is multiplied by both an in-phase and a quadrature signal at frequency f_c, and then, low-pass-filtered to remove the unwanted upper side-bands. The final result consists of $i'(t)$ and $q'(t)$, which can be arbitrarily different from the original $i(t)$ and $q(t)$ components. The greatest source of difference usually comes from RF propagation, which affects $i(t)$ and $q(t)$ so badly to make the symbol recovery impossible. When the *signal-to-noise ratio* is large enough, the symbols are evenly distributed and the information recovery becomes feasible. Further, there are also minor effects that introduce small offsets in the I-Q symbols. A typical example is constituted by impairments and biases introduced by small differences in the electronics components that, although being mass-produced by controlled and standardized assembly lines, are still characterized by imperfections at nano-scale, that affect the displacement of the symbols. The analysis introduced in later sections proves that the symbols' displacement is systematic, thus being at least theoretically possible to detect it, measure it, and eventually leverage it to identify the hardware generating it.

B. Autoencoders

Autoencoders are artificial neural networks whose goal is to learn an optimal representation (i.e., encoding) of a training set from which it is possible to accurately reconstruct the input data. Although it may seem trivial (i.e., the mere copy of the input data to the output may easily lead to an outstanding accuracy), to identify useful features, the internal function responsible for the research of good encoding candidates is usually constrained. For instance, the autoencoder may be forced to find an encoding smaller than the input data (i.e., undercomplete autoencoder). This unsupervised technique has been widely used for dimensionality reduction and feature learning, since it may be tuned to generate smaller encoding, similar to the original input. Recently, autoencoders are also being put to the forefront of generative modeling [16]. The more similar the output to the training set, the more likely the autoencoder represents input data. In case the encoding is (parametrically) smaller than the input data, the feature reduction phase is successful. Different autoeconder models are available, such as the regularized autoencoders, able to learn the most salient features of the data distribution [16], and variational autoencoders, providing a framework to learn deep latent-variable models and the corresponding inference models [17]. Autoencoders usually include four components: (i) an encoder, allowing to learn the features; (ii) a bottleneck, identified as the layer containing the encoding of the training set; (iii) a decoder, allowing the model to learn how to reconstruct the input data from the encoding; and, (iv) the reconstruction error function, measuring the performance of the model during training. Autoencoders have been applied for intrusion detection tasks [18], anomaly detection [19], and DDoS attack detection [20].

In this paper, we rely on autoencoders to perform the one-class classification task on the IRIDIUM satellites. The intuition is the following: starting from a distribution X, the reconstruction of input data drawn from X is easier (i.e., the error metric is reduced) than the reconstruction of input data drawn from any other distribution Y, with $Y \neq X$.

C. Convolutional Neural Networks

A CNN is a Deep Neural Network (DNN) having at least one convolutional layer, i.e., a layer performing convolutional operations. A convolutional operation, in turn, is the mathematical combination of two functions that produces a third function, being the expression of the change of shape caused by the application of one function to the other. For CNNs, a convolution consists of a slide of a parametric-sized filter (also known as operator) over the input representation. Being the filter smaller compared to the input representation, it is applied to different overlapping input portions, thus generating a feature map. Different filters allow to catch different patterns within the input representation (i.e., in case the input is represented as an image, operators can be used to highlight edges, corners, and possibly other patterns). A typical CNN is composed of three types of layers: (i) convolutional layers, to build the feature map of the input; (ii) pooling layers, to reduce the number of learnable parameters and discretize the input; and, (iii) fully-connected layers, to hold the high-level features found during the convolutions and to learn non-linear combinations of them. Nowadays, CNN applications can be found in handwriting recognition, face detection, behavior recognition, recommendation systems, speech recognition, image classification, and Natural Language Processing [21].

D. Transfer Learning

Until a few years ago, conventional ML algorithms have been designed to work in isolation, trained every time from scratch to solve specific tasks. However, training a network from scratch may be cumbersome, since the available datasets may not be rich enough to effectively capture the features. As a result, the resulting classifier could not generalize properly when applied in the wild. Conversely, transfer learning takes advantage of the knowledge learned while solving a task in a particular domain, to simplify the learning phase for a task in another domain. As highlighted in [22], transfer learning is particularly advantageous in situations where the model related to the source task has been trained on a training set bigger or approximately of the same dimension of the one of the destination task. In such situations, transfer learning helps mitigating overfitting, producing a more reliable model. In this paper, to perform multi-class classification on the IRIDIUM satellites, we used the Resnet-18 CNN, pre-trained on the popular ImageNet dataset. Resnet, introduced in 2015, proved to be very performant, since it is structured in a way to achieving deeper architectures with a reduced number of parameters [23]. We provide more details on these in Sec. V-A.

E. Iridium Satellite Constellation

The IRIDIUM satellite constellation was conceived in 1987 and first operated in 1993 by IRIDIUM SSC, founded by

Motorola [24]. The constellation is constituted by a set of Low-Earth Orbit (LEO) satellites, orbiting 800 km above the Earth surface, and arranged so that they can guarantee full Earth coverage at any time. The name of the satellite constellation is inspired by the originally-planned number of satellites, i.e., 77, coincident with the atomic number of the IRIDIUM chemical element. However, to minimize deployment costs while still guaranteeing Earth coverage, only 66 satellites are operational nowadays. IRIDIUM radio signals are transmitted in the L-band, in the frequency range $[1,616-1,626.5]$ MHz. On the ground, IRIDIUM subscribers can receive such signals as well as transmit by using dedicated mobile satellite devices, provided by companies such as Motorola and Kyocera. Today, IRIDIUM is mainly used on-board of vessels, to initiate and receive calls when located off-shore. In this context, starting from January 2020, the International Maritime Organization (IMO) has certified IRIDIUM as an approved Global Maritime Distress and Safety System (GMDSS) service provider for vessels. However, IRIDIUM transceivers are also used in the aviation, railway, and critical infrastructures domains, and recently they have received significant attention also in the emerging satellite-Internet of Things (IoT) application domain [25]. Each IRIDIUM satellite includes an array of antennas, hereby referred to as *beams*, that widens the transmission range of the satellite at the ground. Overall, each satellite has 48 beams and an additional antenna dedicated to the identification of the satellite. Note that the transmission power adopted by the *satellite* antenna is higher than the one used by the *beams*, so that any receiver that could decode the signal emitted by a beam can also receive the information about the satellite itself. Overall, two channels categories are available, i.e., *system overhead channels* and *bearer service channels*. In this paper, we focus our attention on one of the *system overhead channels*, i.e., the IRIDIUM Ring Alert (IRA) broadcast channel. It is a broadcast, unencrypted, downlink-only channel, operating at the center frequency of $1,626.27$ MHz, and used to deliver information useful for handover operations at the ground. IRA messages are characterized by a 12 bytes preamble, encoded according to the Binary-Phase Shift Keying (BPSK) modulation scheme, while the rest of the information (103 bytes) follows the Differentially-encoded Quadrature-Phase Shift Keying (DQPSK) modulation. Such information includes the ID of the satellite emitting the packet, the specific transmitting beam (the beam ID is 0 in the case the transmitter is the one identifying the satellite), the position of the satellite (expressed in latitude, longitude, and altitude), and further information used for handover, e.g., the Temporary Mobile Subscriber Identity (TMSI) of any user subject to handover. Note that IRA packets can have different sizes, depending on the amount of TMSIs included in the message, as well as the presence of additional specific paging information. Previous contributions [8] used the information included within IRA messages to reverse-engineer several system parameters of the IRIDIUM constellation, such as the speed of the satellites, the coverage at the ground, the arrangement of the beams, and the satellite pass duration. In this paper, we further extend those results, by providing hints on the time needed to *observe* a specific satellite, the distribution of I-Q samples, the effect of the noise, and the expected number of I-Q samples per satellite pass (see Sec. III. This information is instrumental to the scope of our work, i.e., the authentication of the IRIDIUM satellite at the physical-layer, by using raw I-Q samples.

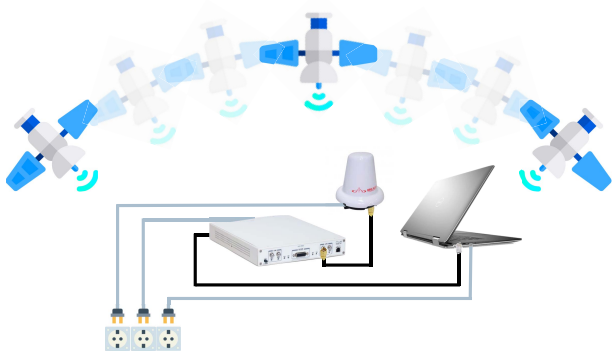


Fig. 3. Measurement Setup: we adopted an active (pre-amplified) Iridium antenna (Beam RST740) connected to a USRP X310 Software Defined Radio.

F. Adversary Model

We assume an adversary whose main objective is to spoof satellite transmissions. Generally speaking, satellite communications can be authenticated by resorting to traditional protocol level services. Nevertheless, authentication is not provided by all the satellite networks, e.g., as in the case of GPS and Galileo. Moreover, the secret material (enabling the authentication process) can be leaked, thus enabling a wide area to be under the control of the adversary—a LEO satellite can easily cover a country-wide area. This makes the satellite network a critical infrastructure, requiring multiple authentication layers. We assume our adversary can both transmit fresh messages and re-transmit previously recorded ones. Moreover, we assume the receiver cannot resort to any protocol level authentication service to verify the identity of the actual transmitter, being either the real satellite or a terrestrial transducer set up by the adversary.

In the remainder of the manuscript, we consider real data acquired through a large experimental acquisition campaign, and we will show that we are able to discriminate single satellites transmitting exactly the same data, even if they come from the same manufacturer and same satellite constellation. Thus, being able to discriminate the actual satellite in such a set, implicitly, we are also able to discriminate transmissions originated from a low-end terrestrial spoofing device, being this latter a weaker adversarial model.

III. IRIDIUM Data Acquisition and Processing

In this section, we first describe the equipment (hardware and software) adopted for our measurements. Then, we show how we reverse-engineered the architectural parameters of the IRIDIUM constellation and, finally, we introduce how we used the I-Q samples to authenticate the satellite transmitters.

A. Measurement Set-up

The measurement setup is illustrated in Fig. 3. The hardware used to acquire IRIDIUM signals consists of a dedicated

TABLE I
EXCERPT OF THE COLLECTED DATASET

Time (s)	Time (ms)	Satellite ID	Beam ID	Latitude	Longitude	I-Q Samples
1580712040	000000739	115	0	25.22	27.66	0.03+0.3j, ...
1580712040	000004519	115	0	25.72	27.66	0.02-0.4j, ...
1580712040	000005059	115	0	26.24	27.67	-0.07+0.8j, ...
1580712040	000005599	115	0	26.39	27.67	-0.2-0.4j, ...
1580712040	000008839	66	0	26.75	27.69	0.03+0.3j, ...
1580712040	000013159	66	0	26.90	27.69	0.03+0.3j, ...
1580712040	000013699	66	0	27.25	27.69	0.03+0.3j, ...

L-Band IRIDIUM antenna, connected to a general-purpose Ettus Research X310 SDR. The antenna is an IRIDIUM Beam Active Antenna, model RST740, commonly used by commercial IRIDIUM transceivers [26]. The antenna is connected through an SMA cable to the Ettus X310 SDR [27], integrating the UBX160 daughterboard [28]. In turn, the SDR is connected via Ethernet to a Laptop Dell XPS15 9560, equipped with 32GB of RAM and 8 Intel Core i7700HQ processors running at 2.80 GHz. On the software side, we used the well-known GNURadio development toolkit. Specifically, we adopted the *gr-iridium* module to detect and acquire IRIDIUM messages [29]. In addition, we used the *iridium-toolkit* tool to parse IRA messages [30]. In detail, we modified the *gr-iridium* module in a way to log the I-Q samples of all the *valid* IRIDIUM packets, i.e., the ones containing the 12 bytes BPSK-modulated preamble, typical of the IRIDIUM messages. For each of these packets, we logged the values of the I-Q samples after the filtering and synchronization performed by the Phased Locked Loop (PLL). Next, we used the *iridium-toolkit* tool to log only valid IRA packets. Our measurement campaign has been carried out in very harsh conditions, i.e., by exposing the IRIDIUM antenna out of the window of an apartment. This is a worst-case scenario, since part of the open sky is obstructed by the wall of the building, attenuating and deviating the signal coming from the satellites. Note that this is not a limitation of our study. Conversely, the high-level performance achieved in such a disadvantaged scenario paves the way for further improvement. Overall, we continuously acquired IRIDIUM signals for about 589 hours (24 days), gathering $102,318,546$ I-Q samples ($1,550,281$ per satellite, on average). An excerpt from the dataset is reported in Tab. I. Specifically, for each received IRA packet we log the reception timestamp on the SDR, both in seconds and in milliseconds, the satellite ID, the beam ID, the latitude, longitude, and altitude coordinates of the emitting satellite, and the raw I-Q samples included in the IRA packet. As recently discussed by the authors in [8], any IRIDIUM satellite is equipped with a total number of 49 radios, where 48 represent the radio of the beams and the remaining one reports the whole satellite ID, characterized by the beam numbered 0. For our work, we further restricted the analysis to *satellite* IRA packets, i.e., the one having beam ID 0. Finally, we implemented the classification algorithms (Convolutional Neural Network (CNN) and autoencoders) in MATLAB R2020a. The training, validation, and testing have been carried out by a server featuring 64 cores, 512GB RAM, and 4 GPUs Nvidia Tesla M40. The collected data are released open source [31].

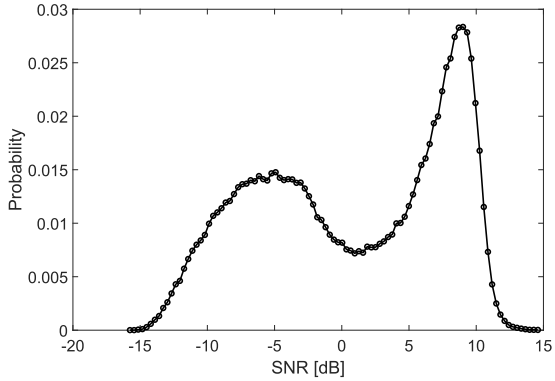


Fig. 4. Signal to Noise Ratio (SNR) computed on all the satellite I-Q samples.

B. Reverse-Engineering IRIDIUM Constellation Parameters

In this section, we derive important parameters of the IRIDIUM constellation, functional to the subsequent analysis. We consider the Signal-to-Noise Ratio (SNR) associated with the collected I-Q samples, the waiting time between two consecutive passes of a specific satellite and, finally, the number of I-Q samples that can be collected per single satellite pass.

1) Signal-to-Noise Ratio (SNR): We start the analysis by considering the quality of the collected samples, in terms of SNR. We considered the signal $\sqrt{i^2(t)+q^2(t)}$ and we computed the associated periodogram (over a sequence of 1,000 samples), while evaluating the SNR as the difference between the power level of the main component (first harmonic) and the power level of the other ones [32]. Figure 4 shows the probability mass function associated with the SNR values computed from all the collected satellite I-Q samples. We observe that there are SNR values significantly high, i.e., spanning between -15 dB and 15 dB: this is mainly due to our measurement set-up, featuring a pre-amplified ad-hoc antenna. Moreover, the SNR values are mainly grouped around two reference values, i.e., -5dB and 9dB. While the former are related to satellite trajectories close to the horizon, the latter ones are from the satellite trajectories crossing on top of the antenna.

2) Waiting Time Between Consecutive Satellite Passes: We also investigate in Fig. 5 the time an observer (on the ground) has to wait to see again the same satellite. We can

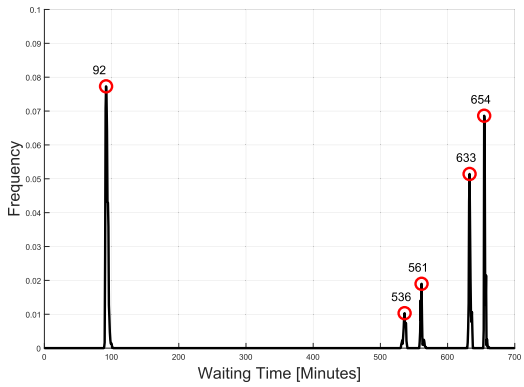


Fig. 5. Waiting time among consecutive satellite passes.

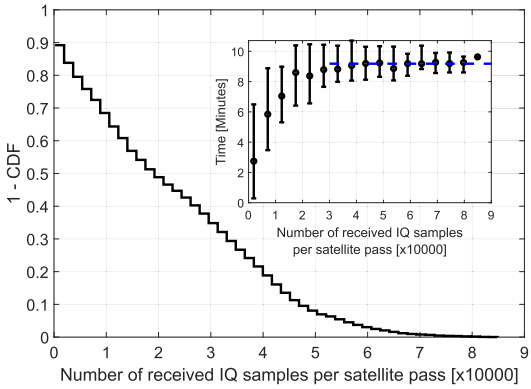


Fig. 6. Probability to experience at least x I-Q samples in a single satellite pass.

explain these results by recalling that a satellite can pass over a specific location in two directions, either north-south or south-north. Indeed, each satellite passes over the same location twice every 90 minutes: up to two consecutive passes can be detected from the same position. Subsequently, after a full Earth revolution, the satellite returns to the same location after about 560 minutes from the opposite direction. Higher waiting times (in Fig. 5), e.g., $560 + 90 \approx 650$ minutes, are due to passes that have not been detected by the receiver.

3) I-Q Samples Per Satellite Pass: Another important parameter for the subsequent analysis is the number of collected I-Q samples per satellite pass, i.e., the number of I-Q samples that can be collected by a receiver during a single satellite pass. Firstly, we consider the inverse cumulative distribution function associated with the number of received I-Q samples (N) per satellite pass, as depicted in Fig. 6, i.e., $P(N > x)$, where x represents a predefined value of I-Q samples. The overall trend is linear up to $50,000$ samples: it is worth noting a probability of 0.7 and 0.5 to have at least $10,000$ and $20,000$ samples per satellite pass, respectively.

The inset of Fig. 6 shows the time required to collect the I-Q samples. For instance, $10,000$ and $20,000$ I-Q samples can be collected by satellite passes lasting for 7 and 8 minutes, respectively. The satellite passes last maximum 9 minutes (median value of the maxima); during this period, we were able to collect between $30,000$ and $80,000$ I-Q samples. We explain this wide range of values due to the varying noise conditions during the measurement campaign. Finally, note the trend between 0 and $30,000$ I-Q samples, characterized by satellite pass length between 3 and 8 minutes. We consider these events to be associated with passes close to the horizon, where the satellite appears just for a short amount of time.

Note that one of the main challenges in physical-layer authentication is to distinguish between the intrinsic features of the transmitters' electronic components and the environmental effects affecting the communication link, such as the Doppler shift and ionosphere effects, to name a few.

In our measurement campaign, we took care of these aspects in two different ways. First, we acquired data (I-Q samples) from IRIDIUM satellites for a long period of time, lasting approx. 589 hours. Considering that a generic IRIDIUM satellite passes twice on the same location at least approx. every 654 minutes (see Fig. 5), we gathered at least 54 passes of the same satellite, on average, in different time windows. Such extended period is not continuous, but spans across multiple non-consecutive time periods, to make our data representative of a large time window and independent from the specific acquisition time (and atmospheric effects experienced by the communication link at that time).

Second, as for the independence of our data from the specific location of transmitter and receiver, note that LEO satellites continuously move while emitting signals, at a speed of approx. 800 km/h. Therefore, consecutive I-Q samples, transmitted even shortly after than the previous ones, experience different communication links, relative locations, and channel conditions, intrinsically experiencing the multitude of conditions needed to allow the classifier to reject temporary effects.

C. Transmitting-Source Authentication via I-Q Samples

Fig. 7 shows the received In-Phase $i'(t)$ and Quadrature $q'(t)$ components of $679,740$ samples gathered from the Satellite with ID 7. Note that the ideal I-Q constellation (recall Fig. 2) is significantly different from the one experienced in real down-link satellite communications. Red circles in Fig. 7 highlight the ideal positions of the I-Q samples and identify the four Cartesian quadrants adopted for the decision (recall Fig. 2), i.e., the received I-Q sample (black dot) is mapped to the corresponding red circle as a function of the Cartesian quadrant on which it lies. The received I-Q samples are affected by different phenomena that displace their original positions. As for the bit error rate, as long as the samples remain in their intended quadrants, the error rate remains zero. In this contribution, we are not interested in the link error rate; instead, we focus on the phenomena behind the I-Q samples' displacement. In general, a received (satellite) signal is affected by the following phenomena:

- *Fading.* Iridium satellites are LEO satellites, hence located at a height of approximately 780 Km, thus being affected by a significant signal attenuation. Note that Fig. 7 is the result of a post-processing amplification, where the samples are stretched to fit the Cartesian plane $[-1, 1] \times [-1, 1]$.
- *Multipath.* Multipath is caused by multiple replicas of the transmitted signal reaching out the receiver through different paths, summing up at the receiver with different

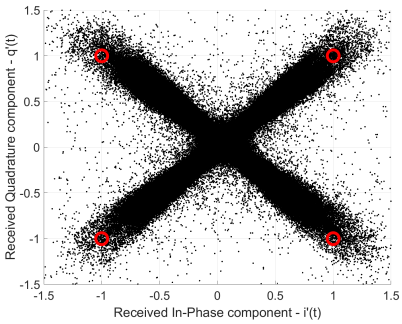


Fig. 7. Received In-Phase $i'(t)$ and Quadrature $q'(t)$ components of 679,740 samples from Satellite with ID 7.

phases. Since the phase shift is random, the attenuation can be arbitrarily large, causing a destructive interference that can significantly affect the signal quality.

- *Doppler shift.* Doppler shift represents the change of frequency (shift) of the received signal as a function of the relative speed between the transmitter and the receiver at the ground. The satellite scenario is particularly challenging, since the Doppler shift is maximum when the satellite is at the receiver's horizon, while becoming minimum at the receiver's zenith.
- *Hardware impairments.* Although mass-produced, any two radio transceivers are not identical. Indeed, the discrete components can be affected by small physical differences at the micro and nano scale (e.g. material impurity), which are reflected in variations of capacitance, resistance, and inductance, eventually leading to small signal artifacts and I-Q unbalances. While the cited imperfections do not affect communication performance, they make the transmitted signal unique, thus enabling the identification of the transmitting source. Unfortunately, such small I-Q unbalances are hidden by all the previously-discussed phenomena—each of them having a remarkable impact on the I-Q unbalancing. In the following, we will discuss an AI-based methodology to detect and extract such imperfections. We will prove our approach being robust to noise, and able to identify a specific satellite transmitter among the 66 that make up the Iridium constellation—thus enabling the physical authentication of the transmitting source.

D. I-Q Samples Pre-Processing

Noise represents a major challenge when the receiver aims at identifying the transmitting source via the I-Q unbalances. Over the years, several techniques have been developed to address the above issue, and the vast majority of them achieve great performance. Nevertheless, none of the mentioned techniques considered the satellite wireless channel. Indeed, recalling Fig. 7, we note that I-Q samples do not appear just around the ideal points (red circles), but they spread all over the I-Q plane. The "cross"-like shape can be explained by the lack of signal amplitude normalization in the demodulation chain [33]. We will prove that the aforementioned issue does not affect our solution, being effective also for small values of the SNR (like the ones of a satellite link).

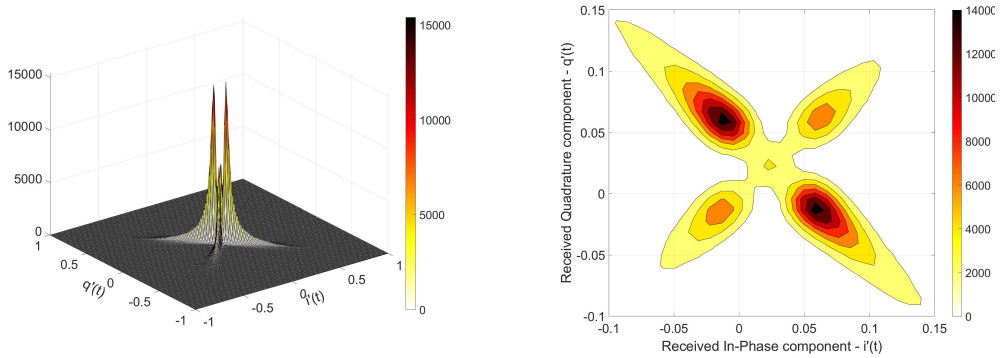
Our approach relies on applying state-of-the-art image pattern recognition techniques to synthetically generated *images of I-Q samples*. As previously discussed, hardware impairments generate (consistent, though low intensity) anomalies in the distribution of the I-Q samples. Therefore, our intuition is to discriminate between the noise and the anomalies by relying on the more powerful classifiers in the literature. The aforementioned methodology requires an effective representation of the I-Q samples in the image domain. Fig. 8 shows how we pre-processed the I-Q samples to represent them as images. In particular, we sliced the I-Q plane into 224×224 tiles (details on this will be clarified later on), and then we evaluated the deployment of different amounts of I-Q samples ($679,740$ from the satellite with ID=7 in Fig. 8). Subsequently, we computed the bivariate histogram over the aforementioned tiles, i.e., the number of I-Q samples belonging to the same tile. Finally, we mapped each value into a grey-scale, i.e., $[0, 255]$, constituting one pixel of our grey image. Therefore, pixels with higher values (white color) represent the tiles with a high number of I-Q samples, while pixels with small values (black color) represent tiles with no I-Q samples. Figure 8(b) highlights the spatial correlation and the features embedded in an image generated as the bi-variate histogram of I-Q samples—those characteristics will be leveraged later on by the deep learning algorithms for classification purposes.

IV. Satellite Authentication Methodologies

In this section, we describe the methodology adopted to authenticate satellite transmitters.

Specifically, we split the whole I-Q samples dataset in three subsets, i.e., *training* ($\mathcal{T}$), *validation* ($\mathcal{V}$), and *testing* ($\mathcal{S}$), each subset accounting for the 60%, 20%, and 20% of the whole dataset, respectively. Moreover, it is worth noting that the number of I-Q samples for each satellite is evenly distributed in each subset (i.e., the dataset is balanced by construction). Let us define $\mathcal{D}_s$ the subset of I-Q samples from satellite s, with $s \in C$ and $C = \{1, \ldots, 66\}$ being the set of satellites in the IRIDIUM constellation. Moreover, let $\mathcal{D}_s$ be the subset of I-Q samples from satellite s and $\mathcal{D}_s = \mathcal{T}_s \cup \mathcal{V}_s \cup \mathcal{S}_s$ where $\mathcal{T}_s$, $\mathcal{V}_s$ and $\mathcal{S}_s$ are the training, validation, and testing subsets associated with the I-Q samples from satellite s. We achieved physical-layer satellite-authentication along two dimensions:

- **Multi-class classification.** We aim at being able to correctly authenticate all the satellites in the constellation. This scenario represents the worst case, involving 66 equivalent classes. We assume prior knowledge on $\mathcal{T}_s, \forall s \in C$. Moreover, we assume the test subset $\mathcal{S}_x$ to be constituted by I-Q samples from the satellite constellation, i.e., $x \in C$—although we do not know to which satellite s the I-Q samples belong to.
- **Binary classification - One-vs-Rest.** We consider a candidate satellite s, and we combine the I-Q samples from all the remaining satellites, thus obtaining two classes: the class containing the reference satellite s, and the one constituted by all the I-Q samples of the remaining satellites, i.e., $C \setminus \{s\}$. Compared to the previous scenario, this one involves limited prior knowledge, i.e., only $\mathcal{T}_s$,



(a) Histogram of the I-Q samples.

(b) Contour plot (magnified) associated with the histogram of the I-Q samples.

Fig. 8. Image representation of I-Q samples.

TABLE II
CLASSIFICATION STRATEGIES

	Prior Knowledge	**Test Subset**
Multi-class	*All satellite* training subsets	*Any test subset* of satellites belonging to the constellation
One vs Rest	*Only the reference* training subsets	*Any test subset* of satellites belonging to the constellation

with s the reference satellite. Moreover, we assume $\mathcal{S}_x$ to be any test subset. Indeed, the algorithm adopted for this categorization returns a *similarity score*, e.g., root mean square, which is used to estimate the similarity of the test subset $\mathcal{S}_x$ against the reference training subset $\mathcal{T}_s$.

Tab. II summarizes our assumptions on the adopted categorization strategies. In the remainder of this paper, we refer to *intra-constellation satellite authentication* as the problem of identifying and authenticating a satellite by resorting to a multiclass classification tool (Sec. V), while we refer to *satellite authentication in the wild* when applying the one-vs-rest classification model (Sec. VI).

V. INTRA-CONSTELLATION SATELLITE AUTHENTICATION

In this section, we focus on the intra-constellation satellite authentication scenario. Specifically, Sec. V-A shows and motivates the deployed CNN, Sec. V-B reports details on the application of the described CNN to authenticate IRIDIUM transmitters, while Sec. V-C investigates the CNN classification performance on subsets of the satellite constellation.

A. Convolutional Neural Network Setup

In this paper, the multi-class classification task is supported by a Deep Convolutional Neural Network (DCNN) based on a Residual Network with 18 layers, i.e., *ResNet-18*. The original *ResNet-18* has its last fully connected layer composed of $1,000$ neurons (followed by a *softmax*), since it was pre-trained on *ImageNet*, a $1,000$-class dataset. Given that we want to classify 66 satellites, we replaced the last fully connected *softmax* layer with a fully connected layer of 66 neurons. Then, we transferred the set of parameters of the *ResNet-18* convolutional layers to the layers of our DCNN.

Note that *ResNet-18* was trained using different images than the ones created through our acquired I-Q samples. On the one hand, we observe that we took the conservative stance of combining common elements such as edges, blobs, patterns, borders, etc. from standard images with new elements typical of I-Q diagrams, with minimal intervention on the structure of the CNN—this guaranteeing a trade-off between training time and performance. On the other hand, training ResNet-18 from scratch might give better results. However, as highlighted in [22], transfer learning is particularly advantageous in situations where the model related to the source task has been trained on a training set bigger or approximately of the same dimension of the one of the destination task. In such situations, transfer learning helps mitigating overfitting, producing a more reliable model. In Sec. VI-D we also report the performance of other CNNs, justifying the selection of *ResNet-18*.

There are mainly two ways to perform transfer learning in deep neural networks: (i) the fine-tuning approach; and, (ii) the freezing layers approach [34]. The fine-tuning requires to retrain (i.e., unfreeze) the whole network parameters, with the classification errors coming from the new training set back-propagating to the whole network. The freezing layer approach, instead, leaves unchanged (i.e., frozen) most of the transferred feature layers. Generally speaking, when the dataset is small compared to the original one (i.e., the dataset on which the network was pre-trained), the freezing layers approach is suggested; otherwise, the fine-tuning approach is the most suitable. However, Yosinki et al. in [34] showed that the freezing layers approach may lead to a drop in performance, while the co-adaptation of the features re-learned with the fine-tuning approach prevents this effect. Since it has been observed that the lower layers of a CNN are able to detect features that are usually general for each image recognition task (e.g., curves and edges), and that fine-tuning allows preventing accuracy drops, here we rely on a combination of the two approaches. Indeed, instead of retraining the network from scratch (i.e., fine-tuning approach) or keeping the layers frozen (i.e., freezing layers approach), we fine-tune the layers of the network with a monotonically increasing learning rate: the deeper the layer in the CNN, the higher the learning rate. In this way, the parameters of the first layers can still detect

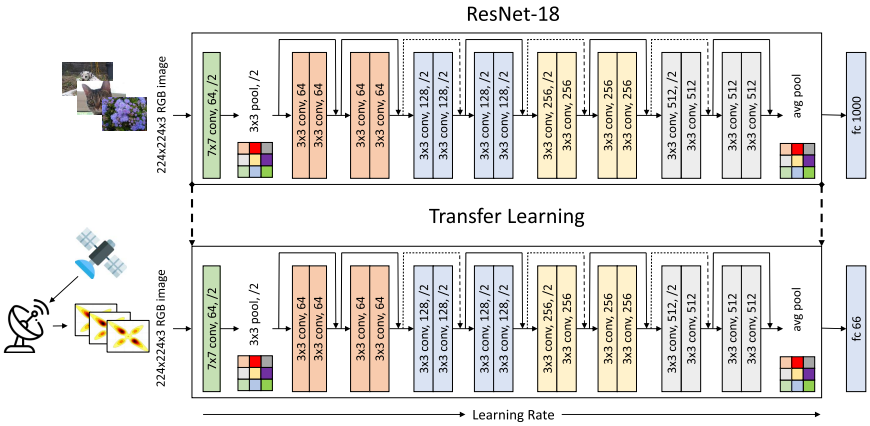


Fig. 9. Overview of the proposed architecture. *ResNet-18* pre-trained layers are transferred to our DCNN, with the replacement of the fully connected layer (i.e., from 1, 000 neurons to 66), and the fine-tuning with monotonically increasing learning rate.

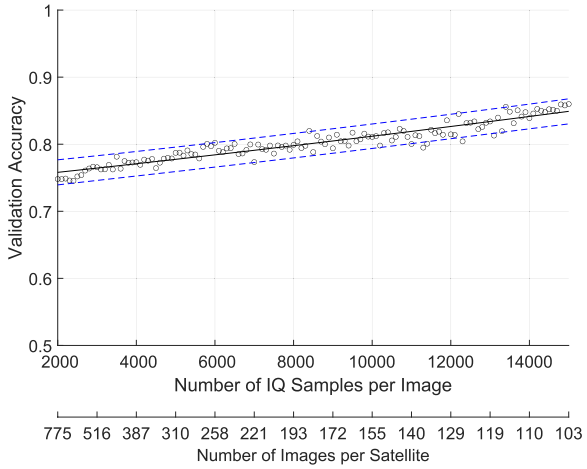


Fig. 10. Validation accuracy as a function of the number of I-Q samples per image (or number of images per satellite).

common features in images, and we opportunely tune the parameters of the deeper layers in a way to guarantee high accuracy. Fig. 9 summarizes the proposed architecture.

B. Satellite Authentication via CNN

In this section, we address the problem of authenticating a satellite by classifying the received I-Q samples. As discussed in Sec. III-D, I-Q samples are pre-processed and converted to 224×224 grey-scale images. Grouping the I-Q samples into images involves the following trade-off: on the one hand, increasing the number of I-Q samples enriches the information possibly conveyed by a single image; on the other hand, the number of available images is reduced, this latter one being the actual input for the classification algorithm (typically, classification techniques performs better as the size of its input increases). Fig. 10 shows the validation accuracy as a function of the number of I-Q samples per image (or the number of images per satellite). Each circle is the result of a single training and validation process, while varying the number of I-Q samples per image. We recall that, for each satellite, 60% of the subset have been used for training and 20% for validation. First, we notice that the validation accuracy increases significantly only by increasing the overall number of samples used to build the corresponding images. Conversely, training the CNN with images constructed using the number of samples contained in a single message leads to very low validation accuracy. This is a finding which leads us to conclude that the I-Q samples corresponding to specific pieces of information in the packet, such as the satellite identifier, are not affecting the validation accuracy of PAST-AI (see the findings recently reported by the authors in [35]). The number of I-Q samples per image is an important parameter, to be evaluated in conjunction with Fig. 6. Indeed, the number of I-Q samples per image should be matched to a single satellite pass. We could consider waiting for multiple satellites' passes, but this approach would involve long waiting times, i.e., at least 92 minutes for the satellite to appear again (recall Fig. 5. Therefore, as a reference parameter, we decided to consider 10, 000 I-Q samples per image (leading to 155 images per satellite), guaranteeing a validation accuracy of about 0.83. Note that the probability to experience at least 10, 000 I-Q samples is about 0.7. Based on the results in Fig. 10, we use 10, 000 samples per image and an overall number of 155 images. On the one hand, we notice that it is always possible to acquire more I-Q samples (IRA packets), and increase the number of available images while not reducing the number of samples per image. On the other hand, processing images composed by a higher number of samples would require additional processing power and computation time. Based on all the motivations discussed above, we selected 155 and 10, 000 as the most reasonable values for the number of images and the number of I-Q samples per image, respectively.

Testing. We run 30 iterations of the training, validation, and testing sequence by randomly choosing the images from the dataset. We computed the mean of the resulting confusion matrices from the testing procedure—results in Appendix. The confusion matrix is sorted according to the values in the diagonal, i.e., best performance (31) in the top left part of the matrix, being 31 images (20% of total 155 images per satellite) the size of the test set for each satellites' image.

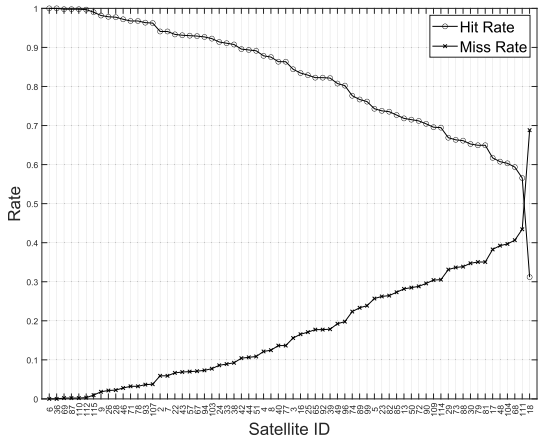


Fig. 11. Hit and Miss rates (mean values) for 30 runs of the CNN classification algorithm. For each run, we consider the whole training, validation, and testing procedures.

We define the *hit rate* as the ratio between the total number of hits (true positive) and the test subset cardinality, yielding:

$$\text{hit rate} = \frac{TP}{TP + FN}$$

We also define *miss rate* the ratio between the total number of misses (false negative) and the test subset cardinality, yielding:

$$\text{miss rate} = \frac{FN}{TP + FN}$$

Fig. 11 shows the hit and miss rates for each satellite, extracted from previous results (see Appendix). Note that 24 satellites (more than 36% of the constellation) experience a hit rate higher than 0.9, while only 4 satellites have a hit rate less than 0.5.

C. Authentication of Satellite Subsets

Driven by the results of Sec. V, we investigate the CNN classification performance on subsets of the satellite constellation. The intuition relies on removing satellites characterized by high miss rates, which are intrinsically difficult to classify, thus constituting a source of mis-classification for the remaining ones. Therefore, we systematically removed the worst satellites (in terms of hit rate) from the dataset, and we subsequently re-evaluated the performance of the classifier.

Fig. 12 shows the accuracy associated with the testing procedure as a function of the number of excluded satellites (the next satellite to be removed is the one with the poorest hit rate among the ones left). The analysis confirms that the image-based classification of I-Q samples is an effective solution. Indeed, CNN classification guarantees a baseline accuracy above 0.82, which can be made arbitrarily high by removing a few satellites—for instance, removing the worst 9 satellites, the accuracy is higher than 0.9.

VI. Satellite Authentication in the Wild

In this section, we tackle the challenge of authenticating a satellite with minimal prior knowledge, i.e., only one of its training subset. Our intuition is to train a model with a reference training subset, and to challenge it with a random test subset. We define a metric, i.e., *reproduction error*, and we estimate the deviation of a synthetically-generated subset from the original one. The reproduction error implies a threshold, under which all the samples are considered as belonging to the satellite to be authenticated.

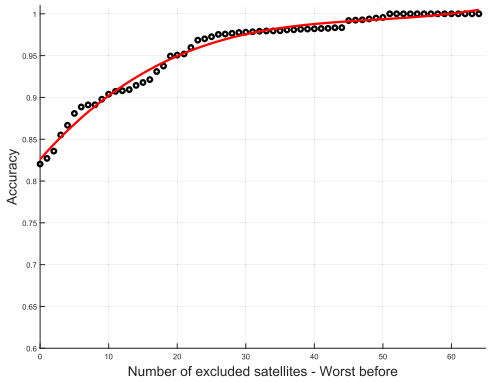


Fig. 12. Testing accuracy as a function of the number of excluded satellites. The removed satellites are the ones with worse performance in terms of hit rate.

The most suitable algorithms for this strategy are *autoencoders*. Indeed, after the training phase, the autoencoders are biased towards the training subset. Therefore, we expect that a synthetically-generated test subset will be characterized by a higher reproduction error, thus being discarded as not belonging to the satellite to be authenticated. We selected the reproduction error coincident with the mean square error (m.s.e.). In the remainder of this section, we first discuss the architecture of the deployed autoencoders (Sec. VI-A). Then, we consider two scenarios: One-vs-Rest (Sec. VI-B) and One-vs-One (Sec. VI-C). The former undertakes the challenge of authenticating the I-Q samples from a reference satellite when compared with I-Q samples coming from a set of sources (the other satellites from the constellation). The latter refers to the classification of I-Q samples coming from two different sources, i.e., the satellite to be authenticated and another (random) one from the constellation. We stress that our test subset is constituted by I-Q samples belonging to the IRIDIUM constellation, only. We consider this assumption the worst-case scenario, i.e., the test subset has the same characteristics as the training subset, in terms of technology, scenario, and noise pattern. Moreover, our solution is agnostic to both the content of the messages (bit-string) and the appearance order of the I-Q samples, since we collect and classify the I-Q samples independently of their mapping to the bit values.

A. Satellite Authentication via Autoencoders

We considered the MATLAB implementation of the *Sparse Autoencoder* to perform the *one-vs-rest* and *one-vs-one* classification. A sparse autoencoder is an autoencoder whose training involves a penalty (also known as sparsity penalty). Several previous works, such as [36], observed that classification tasks may see their performance considerably improved when the representations are learned in a way that encourages sparsity (e.g., by adding a regularizer to the cost function).

TABLE III
TRAINING OPTIONS OF OUR AUTOENCODER

Parameter	Value
HiddenSize	1,024
MaxEpochs	100
EncoderTransferFunction	*logsig*
DecoderTransferFunction	*logsig*
L2WeightRegularization	0.001
SparsityRegularization	1
SparsityProportion	0.05
LossFunction	*msesparse*
TrainingAlgorithm	*trainscg*
ScaleData	*true*

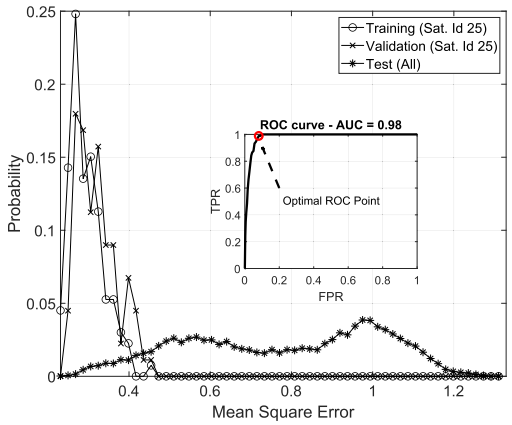


Fig. 13. Distribution of the m.s.e. for training, validation, and testing with autoencoders (*One-vs-Rest* scenario for satellite with ID 25). The inset highlights the ROC curve and the optimal point.

B. One-vs-Rest

In this section, we consider the *One-vs-Rest* scenario: the reference satellite (to be authenticated) versus the rest of the constellation. Fig. 13 resumes the results of our methodology for the case of the satellite with $s = 25$. We trained the autoencoder with the training subset, constituted by the 80% of the subset samples from satellite 25. Then, we used the trained autoencoder to generate a training subset and we estimated the m.s.e. between the two subsets, i.e., the original one and the generated one. The circles in Fig. 13 identifies the probability density function associated with the m.s.e. computed over the original training subset and the generated one. We performed the same procedure on the validation subset (remaining 20% of the samples from satellite 25), and we computed the probability density function associated with the m.s.e. between the original validation subset and the generated one, as depicted by the distribution identified by the crosses in Fig. 13. It is worth noting that the two distributions (the one associated with the training subset and one associated with the validation subsets) are characterized by the same m.s.e., in the range between 0.2 and 0.5.

We applied the same process to a test set, constructed by considering all the satellites from the IRIDIUM constellation, but the one with ID 25. We consider the previous one as the worst-case scenario, since we considered the I-Q samples

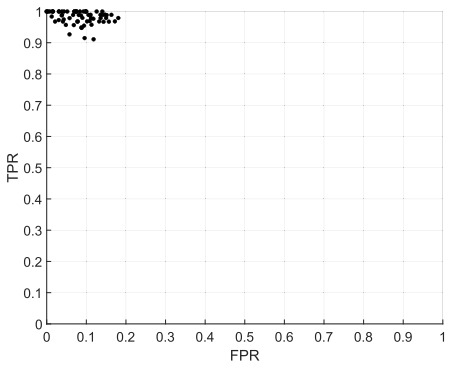


Fig. 14. Optimal operating points of the ROC curves for each satellite when testing (with autoencoders) one satellite against the whole constellation dataset (*one-vs-rest*).

originated from transceivers belonging to the same owner, all of them deployed within a short time delay, and hence very likely featuring the same hardware. Asterisks in Fig. 13 identifies the distribution associated with the m.s.e. computed between the generated test and training subset. The test subset is characterized by m.s.e. values in the range between 0.7 and 1.4, with only a few values less than 0.5. By defining a threshold thr in the range between 0.2 and 1.5, and assuming as legitimate the m.s.e. values less than thr, we can experience different False Positive (FP) and False Negative (FN) events. The trade-off between FP and FN can be evaluated by resorting to the associated ROC curve, as shown in in the inset of Fig. 13, where the True Positive Rate (TPR) is evaluated as a function of the False Positive Rate (FPR), with TPR and FPR being $\frac{TP}{TP+FN}$, and $\frac{FP}{FP+TN}$, respectively. In optimal conditions, i.e., $TPR = 1$ and $FPR = 0$, the AUC should be equal to 1; in our case, for the developed example related to the satellite with ID 25, we report an AUC of about 0.98. Finally, we considered the optimal point of the ROC curve, i.e., the best cut-off with the highest TPR and the lowest FPR, and we reported this value as the red circle in the inset of Fig. 13, with coordinates [0.048, 1]. We applied the aforementioned procedure for all the satellites in the constellation, thus evaluating the optimal operating point in the ROC curve for each of the investigated satellites, as depicted in Fig. 14. The 66 dots identifying the optimal operating points of the ROC curves (one per satellite) are very close to each other, and in turn, very close to the optimal point $TPR = 1$, $FPR = 0$. Finally, Fig. 15 shows the sorted AUC values for all IRIDIUM satellites. AUC values are characterized by very high values (greater than 0.93).

C. One-vs-One

In this section, we consider the *One-vs-One* scenario: the reference satellite (to be authenticated) versus each satellite in the constellation. We followed the same methodology of Sec. VI-B, by considering the generation of a training and test subset and their comparison in terms of m.s.e. values. Finally, we considered different thresholds, and we evaluated the AUC for each satellite pair in the IRIDIUM constellation. Indeed, for each considered reference satellite, we evaluated

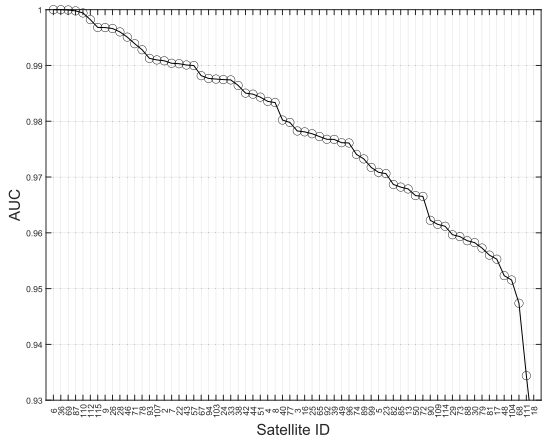


Fig. 15. AUC for each satellite in the constellation when performing One-vs-Rest classification.

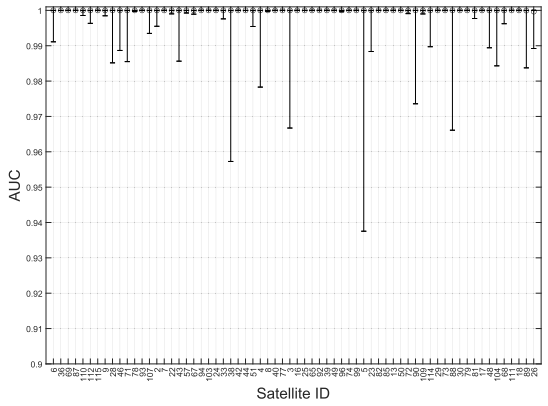


Fig. 16. Error-bars (quantile 5, 50, and 95) associated with the AUC for each satellite in the constellation, when performing *One-vs-One* classification.

66 classifications and the related AUC. Fig. 16 shows the error-bars (quantile 95, 50, and 5) associated with each considered reference satellite. We adopted the same order as before, i.e., satellites are sorted by performance (best on the left) considering the median value. We observe that the quantile 95 and the median are coincident and equal to 1 for almost all the satellites, while only a few satellites are characterized by a quantile 5 below 0.99. This is due to a few satellite-to-satellite classifications experiencing lower performance, but still characterized by AUC values greater than 0.96.

Note that, in line with related works on ADS-B signals fingerprinting, the satellite transmitters are already moving while emitting signals. Thus, our results already take into account the variability of the communication channel between the transmitter and the receiver, and the models we built are able to implicitly compensate such phenomena.

D. Discussion and Limitations

1) CNN Impact: Here, we discuss preliminary results about the impact of the network type on the classification accuracy. We considered three well-known CNNs: AlexNet, ResNet-18 (the one adopted in this paper), and Inception-v3. Table IV reports their most important parameters. Figure 17 shows the

TABLE IV

COMPARISON OF ALEXNET, RESNET-18, AND INCEPTION-V3 IN TERMS OF DEPTH, SIZE, NO. OF PARAMETERS, AND INPUT SIZE

Network	Depth	Size [MB]	Parameters [Milions]	Image Input Size
AlexNet	8	227	61	227x227
ResNet-18	18	44	11.7	224x224
Inception-v3	48	89	23.9	299x299

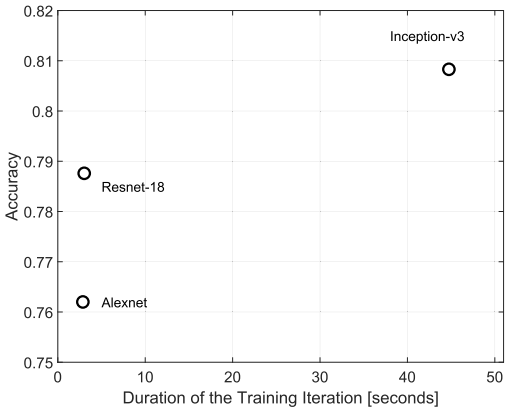


Fig. 17. Accuracy as a function of the duration of the training process for AlexNet, ResNet-18, and Inception-v3.

classification accuracy we obtained on our dataset, as a function of the duration of the training when considering the three CNNs. We observe that the network structure significantly affects the performance, although other parameters should be taken into account when selecting a specific network, such as the time for the training process and the size of the network, which might not fit the hardware requirements of a real deployment. The choice of ResNet-18 emerges as a trade-off between overhead (training time and network size) and accuracy, although more work is required on this topic in order to properly choose the network that perfectly fits the requirements of a real deployment, maximizing the accuracy.

2) Rogue Packets Injection: Finally, we study the problem of an adversary injecting less than $10,000$ I-Q samples, i.e., the number of I-Q samples constituting an image. Indeed, $10,000$ I-Q samples correspond to $20,000$ bits ($2,500$ bytes). The rationale behind the study in this section is that an adversary might successfully inject less information than $2,500$ bytes, while trying to pass the tests described in previous sections. Therefore, hereafter we provide the analysis about the behaviour of our solution under such assumption.
We consider the *One-vs-One* problem from Section VI-C, taking into account the satellites 6 and 36 (the ones reporting the best performances from Fig. 16). We trained the autoencoder with the images associated with the satellite 36, and subsequently, we generate a test set by mixing the I-Q samples coming from the same satellite (36) and another one, i.e., the satellite 6. We consider a *mixing factor* spanning from 0 (no I-Q samples coming from satellite 6) to 1 (all the considered I-Q samples are coming from satellite 6). Figure 18 shows the errorbars (quantile 0.01, 0.5, and 0.99) associated with the m.s.e. as a function of the mixing factor,

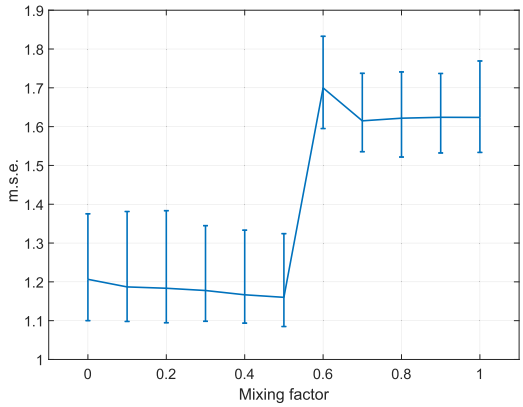


Fig. 18. Mixing I-Q samples coming from different satellites: mse of the reconstructed image versus the original one with a growing mixing factor, considering the satellite 36 as the reference and mixing samples from the satellite 6.

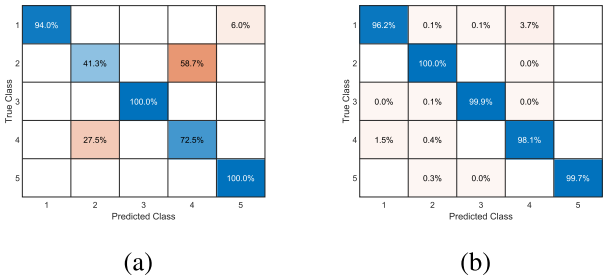


Fig. 19. Classification of 5 satellites, with data batches of 230 IQ samples (one satellite packet). We report the confusion matrix of (a) the approach in [37], and (b) PAST-AI.

where the m.s.e. has been computed considering the images from the original (mixed) test-set and the ones re-constructed adopting the model trained on the images generated from the satellite 36. The m.s.e. (quantile 0.99) is less than 1.4 when the mixing factor is less than 0.5, i.e., the illegitimate I-Q samples (the ones coming from satellite 6) are less than 5, 000. Conversely,when the mixing factor is greater than 0.5, the m.s.e. (quantile 0.01) is greater than 1.5 and our solution detects that the image (and the associated I-Q samples) are not coming from the purported satellite (36). We observed similar results also for other couples of satellites, independently from the performance of the single trained model. The analysis shows that, in the satellite scenario, physical-layer authentication is effective only when the adversary injects a number of I-Q samples higher than a half of the ones constituting the generated images—this value selected as a function of the desired accuracy (see Fig. 11). Considering the parameters of the IRIDIUM constellation, i.e., D-QPSK and 230 bits per packet (IRA messages), one image is constituted by 20, 000 bits (2, 500 bytes) or 87 packets.

The above considerations highlight that performing physical-layer authentication at packet level is not feasible, but multiple packets (87) should be considered. The detection process becomes not effective if the adversary injects a number of packets equal or less than 44. This result is a limitation of the physical-layer authentication process, specific of the satellite scenario. Indeed, the combination of the low SNR of the received signals and the reduced bit-rate of the communication link force the system to work on multiple packets, making the authentication of the single packet challenging.

3) Comparison: Some solutions in the literature consider raw IQ samples, i.e., they provide the IQ samples directly to the CNN, modified on purpose to accept IQ sequences despite images. We compare our solution with [37], focusing only on the best 5 satellites, i.e., the ones reporting the best performance from the table in Appendix. To guarantee a fair comparison, we considered the same batch size of IQ samples assumed in this paper, training both the neural network (AlexNet) in [37] and ours with sequences of 10,000 IQ samples. We report the results in Figure 19 (a) for [37] and Figure 19 (b) for PAST-AI. PAST-AI achieves a staggering accuracy of 0.98, against 0.81 of the competing solution.

VII. Related Work

Physical-layer authentication solutions based on the analysis of raw I-Q samples have gained significant popularity in the last years, and they have been adopted in several scenarios.

For instance, as for mobile cellular networks, the authors in [38] proposed *FBSLeuth*, a framework able to identify rogue 2G base stations through hardware impairments. To this aim, they used the Support Vector Machines (SVM) algorithm. In the same context, the authors in [39] relied on Differential Constellation Trace Figure (DCTF)-based features and CNNs to identify mobile phones. Similarly, the authors in [40] first distinguished among Commercial Off-The-Shelf (COTS) WiFi devices and SDRs emitting WiFi-compliant signals. Using a CNN operating on raw I-Q samples, they precisely identified the transmitter among 16 other SDRs. The authors further extended their work in [11], showing how the classification accuracy can reach over 99% by smartly removing the noise effects of the wireless channel. The impact of the wireless channel on wireless radio fingerprinting has been specifically studied in [13]. They evaluated the accuracy of CNN in several conditions, i.e., in an anechoic chamber, in the wild, and using cable connections, analyzing both WiFi and Automatic Dependent Surveillance - Broadcast (ADS-B) signals (employed in the aviation domain). They revealed that the wireless channel can severely affect the accuracy of the radio fingerprinting, degrading the classification accuracy up to the 85% in low-SNR regime. At the same time, they showed that equalizing I-Q data can slightly enhance the accuracy. By working on the same dataset, the authors in [41] confirmed that partial equalization of the samples can improve the accuracy. ADS-B signals were investigated also in [12]. Specifically, the authors compared three DNNs, with different number of hidden layers and nodes, and they showed that the performance slightly decreases when aircraft increases. Recently, the authors in [10] demonstrated that stacked autoencoders can be used to enhance I-Q fingerprinting, especially in low-SNR scenarios. They used 27 CC2530 micro-controllers, and they were able to distinguish each of them with accuracy over 90 % starting from 5 dB SNR. Autoencoders were successfully used also by the authors in [42] for WiFi devices fingerprinting and by the authors in [43], for anomaly detection

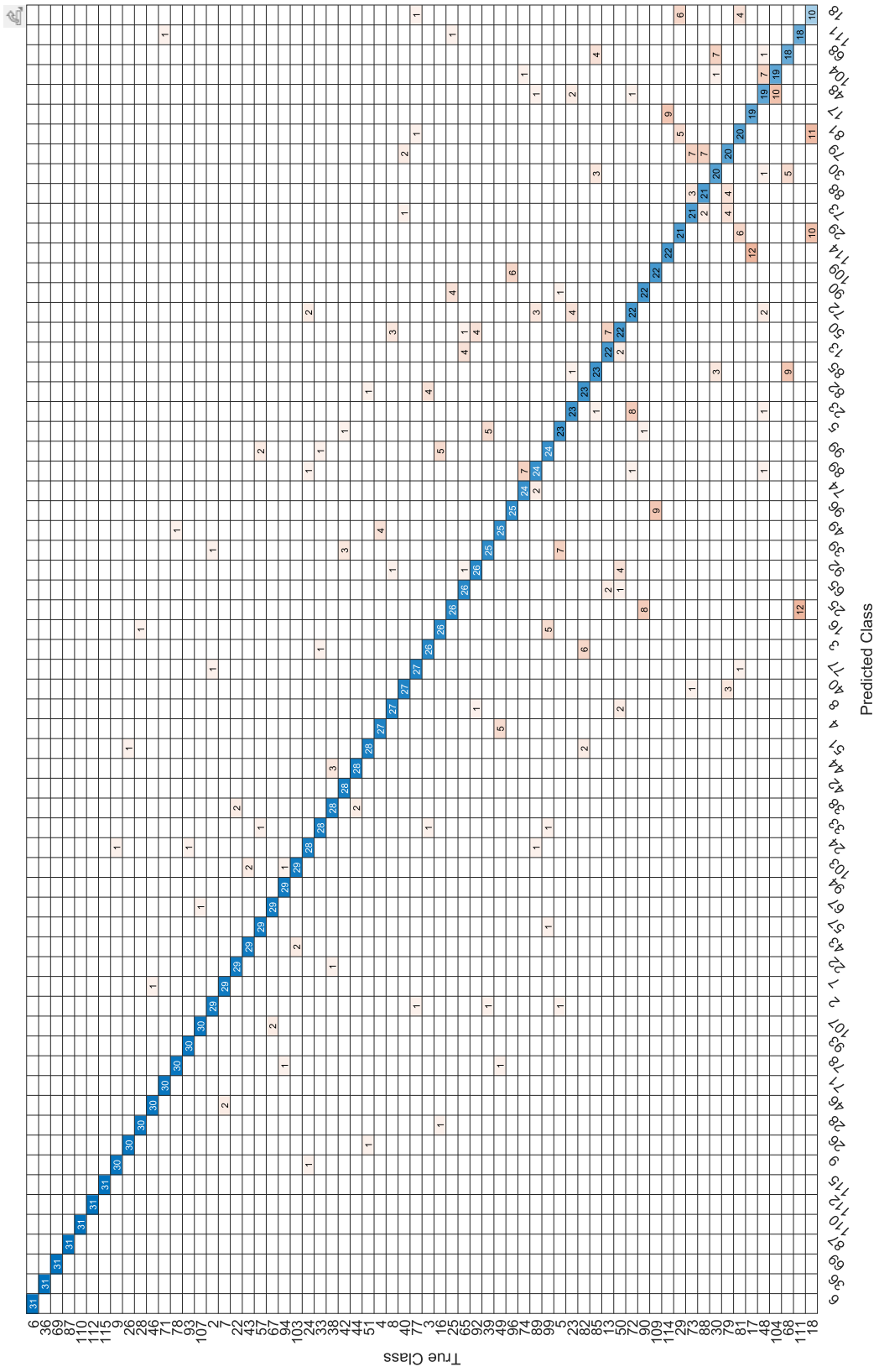


Fig. 20. Confusion matrix of the classification accuracy of the deployed autoencoder in the 1-vs-rest scenario.

of unknown transmitters. However, at the time of this writing, their application to satellites has not been explored.

Despite the large amount of research on I-Q fingerprinting [10], [44], [45], [46], the satellite scenario has not yet been considered. Indeed, being satellites located at a significant altitude, the signals are typically characterized by a low SNR, thus making the fingerprinting task challenging. At the time of this writing, the only contribution working on the fingerprinting

of satellites is [47]. The authors identify Global Positioning System (GPS) spoofing attacks by analyzing the received I-Q samples, using a statistical approach based on scores computed over characterizing Multi-Variate Normal (MVN) distributions. However, they extracted the I-Q samples after the I-Q demodulation at the Radio-Frequency (RF) front-end. Therefore, their solution does not act on raw I-Q samples and applies only to US GPS satellites. Finally, the authors focused on the detection of GPS spoofing attacks, and they distinguish SDRs from legitimate satellites, not the specific transmitting satellite. Conversely, in this paper, we identify the specific transmitting satellite, considering *raw* I-Q samples, before any demodulation operation. As a result, our methodology applies to a wider set of scenarios than spoofing attacks, and it is potentially applicable to all LEO satellites constellations adopting Phase Shift Keying (PSK) modulation techniques.

VIII. Conclusion

We presented PAST-AI, a methodology for physical-layer authentication of satellite transmitters that leverages the power of deep learning classifiers, such as CNNs and autoencoders, applied to the generated I-Q samples. To the best of our knowledge, we are the first to investigate the effectiveness and limitations of radio fingerprinting for the satellite domain—in particular, for LEO constellations—characterized by high attenuation, multi-path fading, strong Doppler effect, and short link duration. We investigated the challenges associated with two scenarios: (i) intra-satellite classification; and, (ii) satellite classification in the wild. We validated our methodology on a dataset generated from a real measurement campaign, involving more than 100M I-Q samples collected from the IRIDIUM constellation. We achieved an accuracy between 0.8 and 1, depending on the scenario. We also demonstrated experimentally the impact of multiple network classifiers, as well as the impact of both the high number of required I-Q samples per image and the limited bandwidth of satellite links on packet-level authentication capabilities. We believe that the novelty of the introduced scenarios, the detailed methodology, and our results will pave the way for future research.

Acknowledgment

The authors would like to thank the anonymous reviewers for their comments that helped improve the manuscript quality. The findings achieved herein are solely the responsibility of the authors. The authors also thank the Qatar National Library (QNL) for the support to the publication of this article.

Appendix

See Fig. 20.

References

[1] X. Wang, P. Hao, and L. Hanzo, "Physical-layer authentication for wireless security enhancement: Current challenges and future developments," *IEEE Commun. Mag.*, vol. 54, no. 6, pp. 152–158, Jun. 2016.

[2] L. Xiao, L. Greenstein, N. Mandayam, and W. Trappe, "Fingerprints in the ether: Using the physical layer for wireless authentication," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2007, pp. 4646–4651.

[3] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 94–104, 1st Quart., 2016.

[4] O. Ibrahim et al., "MAGNETO: Fingerprinting USB flash drives via unintentional magnetic emissions," *ACM Trans. Embedded Comput. Syst.*, vol. 20, no. 1, pp. 1–26, Dec. 2020.

[5] D. Schmidt, K. Radke, S. Camtepe, E. Foo, and M. Ren, "A survey and analysis of the GNSS spoofing threat and countermeasures," *ACM Comput. Surv.*, vol. 48, no. 4, pp. 1–31, May 2016.

[6] N. Soltanieh, Y. Norouzi, Y. Yang, and N. C. Karmakar, "A review of radio frequency fingerprinting techniques," *IEEE J. Radio Freq. Identificat.*, vol. 4, no. 3, pp. 222–233, Sep. 2020.

[7] G. Oligeri, S. Sciancalepore, O. A. Ibrahim, and R. Di Pietro, "Drive me not: GPS spoofing detection via cellular network: (Architectures, models, and experiments)," in *Proc. 12th Conf. Secur. Privacy Wireless Mobile Netw.*, May 2019, pp. 12–22.

[8] G. Oligeri, "GNSS spoofing detection via opportunistic IRIDIUM signals," in *Proc. WiSec*, 2020, pp. 42–52.

[9] X. Zhou, A. Hu, G. Li, L. Peng, Y. Xing, and J. Yu, "Design of a robust RF fingerprint generation and classification scheme for practical device identification," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Jun. 2019, pp. 196–204.

[10] J. Yu et al., "Radio frequency fingerprint identification based on denoising autoencoders," in *Proc. WiMob*, Dec. 2019, pp. 1–6.

[11] K. Sankhe et al., "No radio left behind: Radio fingerprinting through deep learning of physical-layer hardware impairments," *IEEE Trans. Cognit. Commun. Netw.*, vol. 6, no. 1, pp. 165–178, May 2020.

[12] X. Ying, J. Mazer, G. Bernieri, M. Conti, L. Bushnell, and R. Poovendran, "Detecting ADS-B spoofing attacks using deep neural networks," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Jun. 2019, pp. 187–195.

[13] A. Al-Shawabka et al., "Exposing the fingerprint: Dissecting the impact of the wireless channel on radio fingerprinting," in *Proc. IEEE INFOCOM*, Dec. 2020, pp. 646–655.

[14] P. Tedeschi, S. Sciancalepore, and R. D. Pietro, "Satellite-based communications security: A survey of threats, solutions, and research challenges," *Comput. Netw.*, vol. 216, Oct. 2022, Art. no. 109246.

[15] T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2001.

[16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[17] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," 2019, *arXiv:1906.02691*.

[18] C. Ieracitano, A. Adeel, F. C. Morabito, and A. Hussain, "A novel statistical analysis and autoencoder driven intelligent intrusion detection approach," *Neurocomputing*, vol. 387, pp. 51–62, Apr. 2020.

[19] S. Nazir, S. Patel, and D. Patel, "Autoencoder based anomaly detection for SCADA networks," *Int. J. Artif. Intell. Mach. Learn.*, vol. 11, no. 2, pp. 83–99, Jul. 2021.

[20] K. Yang, J. Zhang, Y. Xu, and J. Chao, "DDoS attacks detection with autoencoder," in *Proc. NOMS - IEEE/IFIP Netw. Operations Manag. Symp.*, Apr. 2020, pp. 1–9.

[21] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, Apr. 2017.

[22] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Dec. 2009.

[23] K. He et al., "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, 2016, pp. 770–778.

[24] S. R. Pratt, R. A. Raines, C. E. Fossa, and M. A. Temple, "An operational and performance overview of the IRIDIUM low earth orbit satellite system," *IEEE Commun. Surveys*, vol. 2, no. 2, pp. 2–10, 2nd Quart., 1999.

[25] IRIDIUM Corp. *IRIDIUM's Internet of Things—Connect to a World of IoT Possibilities*. Accessed: Sep. 13, 2022. [Online]. Available: https://www.iridium.com/solutions/iot/

[26] Beam Communications. *IRIDIUM Beam Active Antenna (RST740)*. Accessed: Sep. 13, 2022. [Online]. Available: https://www.beamcommunications.com/satellite/76-iridium-beam-active-antenna

[27] Ettus Research. *USRP X310*. Accessed: Sep. 13, 2022. [Online]. Available: https://www.ettus.com/all-products/x310-kit/

[28] *UBX160 Daughterboard*. Accessed: Sep. 13, 2022. [Online]. Available: https://www.ettus.com/product/details/UBX160

[29] C. C. C. München. *Gnuradio IRIDIUM Out of Tree Module*. Accessed: Sep. 13, 2022. [Online]. Available: https://github.com/muccc/gr-iridium
[30] *Simple Toolkit to Decode IRIDIUM Signals*. Accessed: Sep. 2019. [Online]. Available: https://github.com/muccc/iridium-toolkit
[31] G. Oligeri and S. Sciancalepore. *Physical Layer Data Acquisition of IRIDIUM Satellites Broadcast Messages*. Accessed: Sep. 13, 2022. [Online]. Available: https://data.mendeley.com/datasets/xcxspv8c2r/1
[32] Mathworks. *SNR—Signal to Noise Ratio*. Accessed: Sep. 13, 2022. [Online]. Available: https://nl.mathworks.com/help/signal/ref/snr.html
[33] S. Sciancalepore. *Weird Patterns in I/Q Values*. Accessed: Nov. 28, 2020. [Online]. Available: https://github.com/muccc/gr-iridium/issues/48#issuecomment-657152591
[34] J. Yosinski et al., "How transferable are features in deep neural networks?" in *Proc. NIPS*, vol. 2. Cambridge, MA, USA: MIT Press, 2014, pp. 3320–3328.
[35] S. Gopalakrishnan, M. Cekic, and U. Madhow, "Robust wireless fingerprinting via complex-valued neural networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.
[36] A. Makhzani and B. Frey, "K-sparse autoencoders," 2013, *arXiv:1312.5663*.
[37] J. Zhang, R. Woods, M. Sandell, M. Valkama, A. Marshall, and J. Cavallaro, "Radio frequency fingerprint identification for narrowband systems, modelling and classification," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3974–3987, 2021.
[38] Z. Zhuang et al., "FBSleuth: Fake base station forensics via radio frequency fingerprinting," in *Proc. ACM AsiaCCS*, 2018, pp. 261–272.
[39] S. Wang, L. Peng, H. Fu, A. Hu, and X. Zhou, "A convolutional neural network-based RF fingerprinting identification scheme for mobile phones," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Jul. 2020, pp. 115–120.
[40] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "ORACLE: Optimized radio clAssification through convolutional neuraL nEtworks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2019, pp. 370–378.
[41] T. Jian et al., "Deep learning for RF fingerprinting: A massive experimental study," *IEEE IoT Mag.*, vol. 3, no. 1, pp. 50–57, Dec. 2020.
[42] A. Gritsenko, Z. Wang, T. Jian, J. Dy, K. Chowdhury, and S. Ioannidis, "Finding a 'new' needle in the haystack: Unseen radio detection in large populations using deep learning," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Nov. 2019, pp. 1–10.
[43] S. Hanna, S. Karunaratne, and D. Cabric, "Open set wireless transmitter authorization: Deep learning approaches and dataset considerations," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 1, pp. 59–72, Mar. 2021.
[44] H. Jafari et al., "IoT devices fingerprinting using deep learning," in *Proc. IEEE MILCOM*, 2018, pp. 1–9.
[45] J. Bassey, D. Adesina, X. Li, L. Qian, A. Aved, and T. Kroecker, "Intrusion detection for IoT devices based on RF fingerprinting using deep learning," in *Proc. 4th Int. Conf. Fog Mobile Edge Comput. (FMEC)*, Jun. 2019, pp. 98–104.
[46] S. Balakrishnan et al., "Physical layer identification based on spatial–temporal beam features for millimeter-wave wireless networks," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1831–1845, 2020.
[47] M. Foruhandeh, A. Z. Mohammed, G. Kildow, P. Berges, and R. Gerdes, "Spotr: GPS spoofing detection via device fingerprinting," in *Proc. 13th ACM Conf. Secur. Privacy Wireless Mobile Netw.*, Jul. 2020, pp. 242–253.