IEEE ComSoc | IEEE COMPUTER SOCIETY | IEEE Signal Processing Society | VTS | IEEE Transactions on Machine Learning in Communications and Networking

Received 17 October 2023; revised 1 March 2024 and 4 June 2024; accepted 18 June 2024.
Date of publication 21 June 2024; date of current version 27 June 2024.

The associate editor coordinating the review of this article and approving it for publication was B. Di.

Digital Object Identifier 10.1109/TMLCN.2024.3417806

Physical Layer Spoof Detection and Authentication for IoT Devices Using Deep Learning Methods

DA HUANG (Member, IEEE), AND AKRAM AL-HOURANI (Senior Member, IEEE)

School of Engineering, RMIT University, Melbourne, VIC 3000, Australia

CORRESPONDING AUTHORS: D. HUANG (da.huang@ieee.org) AND A. AL-HOURANI (akram.hourani@rmit.edu.au)

This work was supported by Australian Government through the Automotive Engineering Graduate Program.

ABSTRACT The proliferation of the Internet of Things (IoT) has created significant opportunities for future telecommunications. A popular category of IoT devices is oriented toward low-cost and low-power applications. However, certain aspects of such category, including the authentication process, remain inadequately investigated against cyber vulnerabilities. This is caused by the inherent trade-off between device complexity and security rigor. In this work, we propose an authentication method based on radio frequency fingerprinting (RFF) using deep learning. This method can be implemented on the base station side without increasing the complexity of the IoT devices. Specifically, we propose four representation modalities based on continuous wavelet transform (CWT) to exploit tempo-spectral radio fingerprints. Accordingly, we utilize the generative adversarial network (GAN) and convolutional neural network (CNN) for spoof detection and authentication. For empirical validation, we consider the widely popular LoRa system with a focus on the preamble of the radio frame. The presented experimental test involves 20 off-the-shelf LoRa modules to demonstrate the feasibility of the proposed approach, showing reliable detection results of spoofing devices and high-level accuracy in authentication of 92.4%.

INDEX TERMS Physical layer security/authentication, IoT, LoRa, RF fingerprinting, spoof detection.

I. INTRODUCTION

DEVICE authentication is of great importance for secure communications of the Internet of Things (IoT) networks. A reliable authentication method should perform two essential functions: detecting malicious spoofing (rogue devices) and authenticating legitimate devices. The conventional authentication relies on software-based approaches such as cryptographic protocols [1] or credential exchange methods [2]. In more recent literature, efforts have also been seen in utilizing packet-level features obtained from network flow [3], [4] to profile IoT devices. However, these methods often require all devices with access to the network or transport layer to accommodate relevant protocols, making them less favorable for resource-constrained IoT devices.

Recent research explores approaches like radio frequency fingerprinting (RFF) as an alternative or add-on to existing software-based methods for additional security in device authentication. While software approaches consider features within the digital domain, RFF mainly focuses on analog ones. RF fingerprints originate from physical layer imperfections due to component-level defects such as manufacturing errors or aging. As a result, every imperfect transmitter (Tx) differs from the others even when they are of the same model and make. Accordingly, the imperfect Tx components introduce impairments such as frequency offsets, phase noises and IQ imbalance to the signal during transmission [5]. Depending on the defects of interest, the corresponding impairments can be exploited from the received signal through proper signal processing and used as device-identifiable fingerprints. As these impairments depend solely on a device's intrinsic characteristics, they are device-specific and challenging to replicate. Therefore, RFF offers the advantage of fingerprinting devices directly without the need for manual credential generation. Unlike other physical layer approaches such as physical unclonable functions (PUF) [6], RFF does not require any hardware or software modifications during

or after manufacturing to make a device eligible. This characteristic makes RFF favorable for IoT applications that utilize resource-constrained end devices (e.g., sensors without access to the transport layer or above) or are challenging for post-deployment modifications.

Literature has covered the extraction of transmitter fingerprints from both the transient stage and steady-state [7] for authentication. Conventionally, manual extraction of fingerprints is essential [8]. However, this process becomes challenging when the root source of fingerprints involves a mix of imperfections at different stages of the RF chain. The emergence of deep learning opens new possibilities for fingerprint utilization with minimal manual intervention. In previous literature, efforts are seen in introducing deep learning models such as convolutional neural network (CNN) [9], [10], [11], [12] and long short-term memory (LSTM) [13], [14] to offer device authentication solutions for RF devices.

RFF can also be utilized for detecting rogue or spoofing devices, especially when incorporated with specific deep learning networks. The generative adversarial network (GAN), a sub-category of adversarial neural networks (ANN), has shown effectiveness in applications of various domains, including medical [15] and industrial [16] anomaly detection. GANs excel in automatically extracting features and only require true class samples during the training process. The exploration of GANs for spoof detection in wireless applications has been observed in multiple fields, such as for GPS signal [17] or spacecraft telemetry data [18].

In this study, we present a framework for spoof detection and authentication, where we utilize continuous wavelet transform (CWT) to extract cross-domain (tempo-spectral) physical layer fingerprints. Accordingly, deep learning tools like GAN and CNN are respectively utilized for spoof detection and device authentication. To obtain practical data for training and testing, we explore LoRa modulation due to its wide availability and popularity for IoT applications. The RF fingerprints are extracted from the preamble section of the LoRa radio frame. This frame portion is independent of the payload and thus remains consistent across different transmission schemes (configurations). In summary, the main contributions of this work are as follows:

- We propose a framework for spoofing detection and authentication utilizing GAN and CNN, respectively. For efficient fingerprint exploitation, we further propose four cross-domain signal representation modalities based on CWT: *rawCWT*, *stackCWT*, *CWTd*, and *stackCWTd*. Each modality adopts a slightly different pre-processing techniques to emphasize potential device fingerprints from different perspectives.
- We evaluate our framework under line-of-sight (LoS) and non-line-of-sight (NLoS) channels using an experimental setup built with off-the-shelf LoRa IoT modules. Given that only the preamble portion of a LoRa signal is used, the proposed framework is compatible with both LoRa physical layer (PHY) and LoRaWAN standards. The results show that the proposed framework achieves robust performance under NLoS environments without channel compensation.
- We open-sourced the collected dataset[1] and the MATLAB realization of GANomaly.[2]

The paper is structured as follows: Section II provides a review of relevant literature, Section III provides the preliminary knowledge on the techniques of interest, Section IV details the methodologies, Section V explains the experimental setup, Section VI and VII present the experimental results, and lastly Section VIII concludes the work.

II. RELATED WORKS

While a great number of anomaly detection protocols implemented on the packet level aim at detecting irregular network traffic behaviors from multiple consecutive instances of packets [3], [4], RFF methods rather focus on authenticating the transmitting hardware directly per transmission basis. Such approaches only utilize physical layer features and do not require analysis of packet attributes. In addition, in contrast to software-based authentication methods relying on cryptographic [1] or mutual authentication [2], physical layer approaches do not require both end devices to have a similar level of architecture complexity. Although software methods offer a greater degree of freedom in algorithm complexity, RFF's potential for asymmetry design and single-transmission detection helps reduce the computational burden for resource-constrained IoT devices, allowing more design flexibility.

The source of device-specific RF fingerprints that are commonly used for device identification can be broadly categorized into: (i) transient or (ii) steady state fingerprints. The transient fingerprints refer to the short modulation-irrelevant portion at the beginning of the transmission when a transmitter, or its power amplifier (PA), is turned on to transmit. Features like energy spectral coefficients [8] and higher order statistical (HOS) measures [19] can serve as efficient RF fingerprints from this region. However, capturing the transient signals requires a receiver (Rx) with a high bit-precision analog-to-digital converter (ADC). Meanwhile, fingerprints from the steady state are more commonly available as typical RF devices spend a much longer period in the steady state. Popular steady state impairments, such as carrier frequency offset (CFO) [13], in-phase/quadrature (IQ) imbalance [10], and PA nonlinearity [20], can be considered as the source(s) of device fingerprints. However, since isolating a specific impairment source can be challenging, some studies consider the combined effects of multiple impairments as united RF fingerprints [11], [12], [14]. In these approaches, extraction of certain impairments is not required, and it is up to the classifier to decide what deterministic fingerprints can be learned from a mixture of multiple impairment sources. Though some

[1] https://bit.ly/49A311C
[2] https://bit.ly/49VBQOH

efforts have been made to utilize channel state information (CSI) as device fingerprints [21], it is not considered an intrinsic RF fingerprint but rather depends on the environment and location of devices [22], making it unsuitable for dynamic applications.

Within the existing body of knowledge, most literature centers around well-known modulation schemes like quadrature amplitude modulation (QAM) and phase shift keying (PSK), which differ from the modulation scheme used in LoRa. As a form of frequency shift keying (FSK), the linear chirp spread spectrum (CSS) adopted in LoRa differs from conventional QAM and PSK by lacking a clear constellation. This characteristic of FSK signals leads to challenges in utilizing specific impairment sources (e.g., CFO and IQ imbalance). Hence, a relatively complex feature engineering framework might be necessary to exploit deterministic fingerprints, as demonstrated in study [33]. Furthermore, the linear CSS LoRa signals occupy a broader bandwidth than conventional modulation schemes and operate in a different frequency band, influencing propagation behaviors and raising doubts about the applicability of existing methods for LoRa. Additionally, the packet structure of the LoRaWAN standard [34] is simpler than those within TCP/IP architectures, limiting the complexity of software-based authentication algorithms that can be implemented in the application layer.

Consequently, implementing new authentication and spoofing detection methods, or re-evaluating the compatibility of existing techniques, for LoRa signals is necessary. However, to the best of the authors' knowledge, there are only limited studies focusing on the authentication and spoofing detection of LoRa IoT devices. In a prior work [29], sLoRa was introduced by authors proposing the use of combined CFO and link signature (estimated link variations) for LoRa device authentication. Two-dimensional signal representations can also be introduced as additional modalities with embedded fingerprints. For instance, DCTF is utilized in [26] to transform LoRa signals into 2D images. Accordingly, the clustering centers of the DCTF are extracted as device fingerprints, and the Euclidean distance is employed to authenticate LoRa transmitters based on the distances between the clustering centers of each device. In other attempts, the short-time Fourier transform (STFT) spectrograms of the LoRa preamble are adopted as the signal representation in [13] and [27]. While CNN functions as the classifier in [13], it is only employed for feature extraction in [27], where an additional k-nearest neighbor (k-NN) classifier is introduced for authentication. In another work with a similar essence, authors in [28] feed STFT spectrograms of LoRa preambles into a deep fractional scattering network (DFSNet) for feature extraction and train 1D-CNNs to classify the obtained feature vectors.

By training only on true class samples, GANs can effectively mitigate the insufficiency in the training set due to the difficulty in preparing sufficient data for all potential anomalous scenarios, known as the data imbalance problem in anomaly detection [35], [36]. Originally designed for synthetic data generation, various adaptations of GANs have been employed to tackle spoofing detection. For instance, utilization of fully connected GANs to detect spoofs is observed in global navigation satellite system (GNSS) [30] as well as generic QPSK signals [14]. Alternatively, GAN built based on CNN structure serves as another popular candidate. Authors in [17] utilize GAN built on 1D-CNN to detect spoofing in raw GNSS IQ samples. In other instances, researchers in [18] have customized the GAN architecture, integrating long short-term memory (LSTM) networks with 2D-CNN to extract temporal fingerprints, thereby adapting the model to sequential data. Notably, GANomaly is utilized in [31] to detect spoofing for ZigBee devices. The approach employs a 2D image representation technique known as differential constellation trace figure (DCTF) to derive combined fingerprints arising from IQ imbalance, I/Q channel direct-current (DC) offset and CFO. It's worth noting the growing interest in utilizing few-shot learning for anomaly detection, as seen in literature handling both packet [37], [38] and physical layer level features [39], [40]. A significant advantage of few-shot learning lies in its learning ability given limited labeled training samples. Although few-shot learning is a strong candidate for authentication, it remains susceptible to data imbalance when introduced for spoofing detection, as providing comprehensive scenarios in its support set remains challenging.

For LoRa systems, the application of GAN in spoofing detection remains relatively limited. Instead, in [27], the authors utilize a CNN + k-NN architecture for spoofing detection. They claim that spoofing inputs are clustered separately from legitimate ones after passing through the trained classifier. In another study [32], gated recurrent units (GRUs) are introduced in a federated learning setup to detect anomalies in industrial LoRa systems by monitoring variations in CFO. However, since CFO has limited resilience to channel variations, additional mechanisms are required to continuously track CFO changes, resulting in increased complexity.

Wavelet transform, an additional well-regarded method for time-frequency analysis, can also be introduced to create 2D representations. The potential of utilizing wavelet transform for device authentication has been explored in previous literature. For instance, authors in [41] and [42] assess the potential of STFT, CWT, and recurrence plots (RPs) for transmitter classification and evaluate their resilience against IQ imbalance degradation on the receiver side. The authors determine that CWT exhibits greater resilience against undesired receiver degradation while ensuring decent classification accuracy. However, the application of wavelet transforms in LoRa contexts remains underexplored.

Table 1 and Table 2 provide a summary of reviewed literature. In the context of LoRa, existing works mainly concentrate on capturing combined features using techniques with fixed resolution, such as STFT. On the other hand, this study aims to assess the viability of various modalities

TABLE 1. Reviewed literature on RFF for transmitter authentication.

Literature	Signal Type	Fingerprint sources	Signal Representation	Classifier
[8]	Wi-Fi (OFDM)	Turn-on transient	Energy spectral coefficients	Probabilistic neural network (PNN)
[23]	Bluetooth	Turn-on transient	HOS features	Linear vector support machine (LVSM)
[20]	16-QAM	PA nonlinearity	Density trace figure (DTF)	CNN
[24]	Wi-Fi (OFDM)	IQ imbalance	IQ imbalance parameter combination	k-NN
[10]	Generic QAM signals	IQ imbalance	Density trace plot (DTP)	CNN
[11]	ZigBee (OQPSK)	IQ imbalance + CFO + DC offset	DCTF	CNN
[12]	Wi-Fi (m-PSK)	Mixture of impairments	Differential contour stellar	CNN
[25]	Satellite (4-QAM alike)	Mixture of impairments	Histogram	CNN
[26]	LoRa	Mixture of impairments	Clustering centers of DCTF	Euclidean distance
[13]	LoRa	Mixture of impairments + estimated CFO	IQ samples	Multilayer perceptron (MLP)
			Fast Fourier transform (FFT)	LSTM
			Spectrogram	CNN
[27]	LoRa	Mixture of impairments	Channel independent spectrogram	CNN + k-nearest neighbors (k-NN)
[28]	LoRa	Mixture of impairments	Spectrogram	DFSNet + CNN
[29]	LoRa	CFO + link signature	Estimated CFO and link signature	SVM
This work	LoRa	Inter-symbol transients + IQ imbalance	Scalogram, stacked scalogram	CNN

TABLE 2. Reviewed literature on spoofing transmitter detection using physical layer features.

Literature	Signal Type	Signal Representation	Classifier
[17]	Satellite	raw IQ	GAN built based on 1D-CNN
[30]	Satellite	Doppler detection matrix	fully connected GAN (built on dense layers)
[31]	ZigBee	DCTF	GANomaly (built based on 2D-CNN)
[18]	Telemetry data	Multivariate matrix	ST-GAN (built on 2D-CNN + LSTM)
[27]	LoRa	Channel independent spectrogram	2D-CNN + k-NN
[32]	LoRa	Fine-grain CFO	GRU
This work	LoRa	Scalogram, stacked scalogram	GANomaly (built based on 2D-CNN)

generated using multi-resolution transform CWT for authentication and spoofing detection. In particular, the stacked modalities produce a more compact representation than traditional spectrograms. Additionally, we tackle the literature gap in ANN-assisted spoofing detection for LoRa applications by exploring the potential of a specific GAN variant.

III. PRELIMINARIES

This section provides a brief theoretical background of CWT and LoRa, the focus technology in this work, as well as their associated settings used in this work.

A. CONTINUOUS WAVELET TRANSFORM

Unlike other time-frequency analysis methods like STFT, CWT offers an advantageous capability in multi-resolution analysis. This is facilitated by utilizing a filter bank consisting of wavelets with different scales. This inherent flexibility enables the customization of CWT to meet specific application requirements by selecting appropriate parameters. In this work, we adopt the generalized Morse wavelet as the mother wavelet, whose Fourier transform is obtained as follows:

$$\Psi_{P,\gamma}(\omega) = U(\omega) c_{P,\gamma} \omega^{\frac{P^2}{\gamma}} e^{-\omega^{\gamma}}. \tag{1}$$

Within (1), $U(\omega)$ is the unit step and $c_{P,\gamma}$ is the normalizing constant. One can adjust two additional parameters, time-bandwidth product P^2 and symmetry parameter γ, to customize the shape of the mother wavelet. In this work, we select the typical value of $\gamma = 3$ to minimize demodulate skewness and $P^2 = 90$ to enhance frequency localization. Moreover, we choose 16 voices per octave (i.e., 16 intermediary scales between consecutive integer values of a) for better estimation resolution.

B. LoRa MODULATION

The LoRa signals utilize the CSS as their dedicated modulation schemes. Without loss generality, a typical LoRa signal is expressed as a standard analytical linear chirp [43] as

$$c(t) = A\exp\left(-j2\pi \int_0^t \left[\frac{B}{2} - \left(\frac{Bx + m_{\mathrm{i}}}{T_{\mathrm{sym}}}\right)_{\mathrm{mod}_B}\right] \mathrm{d}x\right), \tag{2}$$

where A denotes the amplitude, T_{sym} is the symbol period and $m_{\mathrm{i}} \in \{0, 1, \ldots, T_{\mathrm{sym}}B - 1\}$ is the symbol value of index i. Further, the symbol period is determined by bandwidth B and spreading factor SF as $T_{\mathrm{sym}} = 2^{\mathrm{SF}}/B$.

Before transmission, the LoRa Tx handler appends an uncoded, payload-independent preamble portion to the beginning of each LoRa PHY packet. As specified in the current standards [34], this segment consists of eight upchirps and 4.25 synchronization symbols, as depicted in Fig. 1. The synchronization symbols vary based on the chosen synchronization words but remain constant when a network type (e.g., public or private) is selected. As the duration of this portion is only determined by the choice of B and SF, it can serve as a suitable reference sequence during synchronization.

For better integrality, we focus on the first twelve symbols within this signal portion (i.e., we discard the last

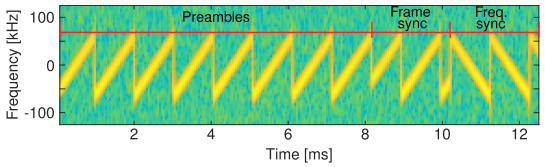

FIGURE 1. Spectrogram of the packet portion added by a LoRa Tx handler (SF = 7, B = 125 kHz).

0.25 symbol). Empirically, the targeted portion can be obtained by segmenting the first $N_s \times (2^{SF} f_s/B)$ samples from a synchronized received signal, where f_s is receiver's sampling frequency and $N_s = 12$ is the number of symbols.

IV. METHODOLOGY

This work considers a base station as the receiver (Rx) and is tasked to authenticate the end-node IoT devices (i.e., Tx). Meanwhile, the incoming traffic may originate from malicious transmitters attempting to gain access by faking their identities through relaying or forging valid transmissions. Noteworthy, this work does not consider scenarios where legitimate Tx devices are compromised and hijacked by adversaries.

The based station has, inherently, better access to resources (processing and energy) and thus could perform more complicated tasks, such as spoof detection and authentication. From this perspective, the proposed framework does not require any additional processing on the Tx side. As illustrated in Fig. 2, the proposed RFF framework is implemented on the receiver (i.e., base station) side. It operates on the baseband physical layer waveform and therefore requires no additional hardware for data acquisition. Given that no changes need to be made to other devices within the network, they are omitted from the figure for simplicity.

In this framework, spoofing detection and authentication take place before the demodulation and decoding of the physical layer payload. Thus, it operates independently of packet analysis and any subsequent applications. Upon completing RFF, only validated transmissions proceed for further processing, while any suspicious ones are discarded (i.e., access denied). Overall, there are a few steps required at the base station, which include (i) **Pre-processing** that conducts frame segmentation, compensation of unwanted effects and power normalization, (ii) **Representaion (modality) generation** that transforms the signal from raw RF domain to more suitable representation for deep learning, and (iii) **Deep learning** phase. During the deep learning phase, the training data is transmitted to the backbone server to facilitate training. Once completed, a copy of the trained network is stored at the base station for local access. The RFF authentication process follows a sequential order. The representation of a PHY transmission is initially subjected to the trained GAN for spoofing detection before being processed by the trained CNN for authentication.

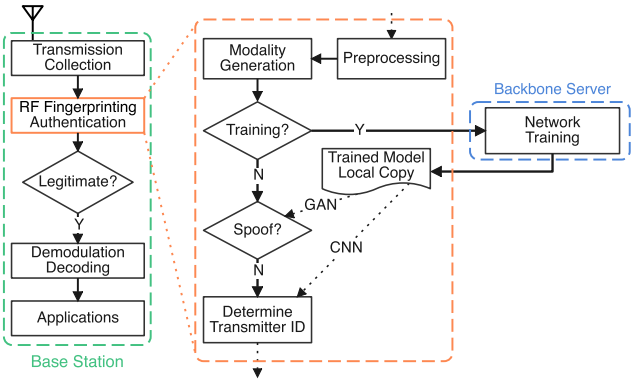

FIGURE 2. Overall framework diagram at the base station (receiver side) for spoof detection and authentication.

More detail of these steps are explained as follows:

A. PRE-PROCESSING

In a typical IQ receiver, the captured IoT signal is first down-converted by mixing it with two carrier signals offset by 90°. After applying a low-pass filter (LPF), the two signals are sampled using two different analog-to-digital converters (ADCs) with sampling frequency f_s, resulting in two baseband sample streams $r_i[n]$ and $r_q[n]$. The analytical received signal is then represented in a complex form as $r[n] = r_i[n] + jr_q[n]$. This stream is further processed following the steps illustrated below. For simplicity, we reuse the same symbol $r[n]$ to represent the signal stream and use "$\Leftarrow$" to denote the updating process of $r[n]$ after certain processing. Note although this work primarily considers LoRa signal portions configured as described in Section III-B, the utilized techniques can potentially be applied to other signals or packet preambles with repetitive patterns.

1) SYNCHRONIZATION AND SEGMENTATION

This step encompasses three stages: (i) coarse synchronization based on energy detection, (ii) fine synchronization based on matched filtering, and (iii) extracting the signal portion of focus. Energy detection can be achieved by setting a threshold to extract the signal portion with energy above the threshold. This will give a rough indication of the radio frame location. The segment is then match-filtered with an ideal reference sequence $h[n]$ of length L generated at Rx. In this work, we select the LoRa preamble as outlined in Section III-B as $h[n]$. Accordingly, we conduct the match-filtering in the frequency domain as

$$n^\star = \underset{n}{\text{argmax}}\ \mathcal{F}^{-1}\left(\sum_{k=1}^{k=N} \mathcal{F}(h^*[n-k])\mathcal{F}(r[k])\right), \qquad (3)$$

where $n^\star$ refers to the starting point of the detected portion, $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ respectively denote the Fourier transform and inverse Fourier transform operation, and N is the length of $r[n]$. An illustrated example of the synchronization steps applied on the received signal is presented in Fig. 3. Lastly,

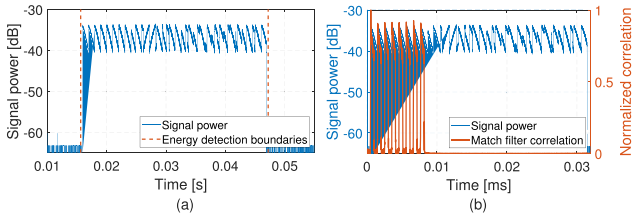

FIGURE 3. Synchronization examples of a LoRa signal. (a) Energy detection when choosing the −10 dB bandwidth power as the threshold, and (b) matched filtering following the energy detection for fine synchronization.

the signal portion of interest can be easily segmented from the Rx frame by extracting L samples starting from position $n^\star$.

2) CFO COMPENSATION

After down-conversion, the theoretical baseband signal is centered around 0 Hz. Nonetheless, the presence of CFO shifts the signal spectrum along the frequency axis. Although CFO can be considered a potential RF fingerprint, it is affected by factors such as variations in ambient temperature as well as Rx hardware degradation. As such, it is difficult to isolate the CFO portion that only originates from the Tx devices. Therefore, we introduce CFO compensation to resolve this ambiguity and ensure consistent performance. We adopt the method presented in [13] for CFO compensation, starting by calculating the instantaneous frequency of $r[n]$ as

$$f_{\mathrm{r}}[n] = \frac{1}{2\pi}\frac{d\phi[n]}{dn}, \tag{4}$$

where $\phi[n] = \mathrm{atan2}(r_{\mathrm{q}}[n], r_{\mathrm{i}}[n])$ denotes the instantaneous phase of $r[n]$. Accordingly, the average frequency of $r[n]$ is obtained as

$$\overline{f_{\mathrm{r}}} = \frac{1}{L}\sum_{n=1}^{L} f_{\mathrm{r}}[n]. \tag{5}$$

Since an ideal LoRa preamble should give $\overline{f_{\mathrm{ideal}}} = 0$ as it is centered around 0 Hz, the coarse CFO is obtained straightaway as $\delta f_{\mathrm{coarse}} = \overline{f_{\mathrm{r}}}$, and the corresponding signal after coarse compensation is updated as follows,

$$r[n] \Leftarrow r[n]e^{-j2\pi n T_{\mathrm{s}} \delta f_{\mathrm{coarse}}}. \tag{6}$$

The resolution of coarse CFO compensation is subject to the precision of $f_{\mathrm{r}}[n]$ estimation and hence requires a finer compensation to mitigate the residuals. The fine CFO compensation benefits from the repetitive structure of the preamble. As such, the residual CFO is estimated as

$$\delta f_{\mathrm{fine}} = -\frac{1}{2\pi T_{\mathrm{s}} L_{\mathrm{sym}}} \angle\left(\sum_{n=1}^{L-L_{\mathrm{sym}}} r[n]r^*[n+L_{\mathrm{sym}}]\right), \tag{7}$$

where L_{sym} is the symbol length, T_{s} is the sampling period, and $\angle(\cdot)$ returns the phase of the variable. Lastly, CFO compensated signal is given as

$$r[n] \Leftarrow r[n]e^{-j2\pi n T_{\mathrm{p}} \delta f_{\mathrm{fine}}}. \tag{8}$$

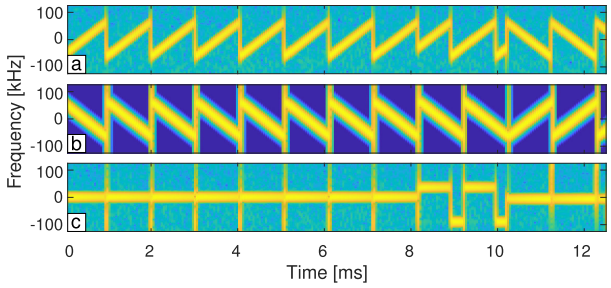

FIGURE 4. Example of de-chirping a LoRa signal: (a) original LoRa signal, (b) ideal basic upchirps and downchirps generated for de-chirping, and (c) LoRa signal after de-chirping (SF = 7, B = 125 kHz).

3) NORMALIZATION

Due to the influence of radio channels, signals arrive at the Rx with varying power levels. To compensate for this variation, $r[n]$ is normalized based on its root mean square (RMS), ensuring unity power. For more comprehensive training and testing purposes, we inject a controlled additive white Gaussian noise (AWGN) to simulate different levels of signal-to-noise ratio (SNR) levels.

4) DE-CHIRPING

This is an optional step that could be applied to a typical chirp modulated signal, such as LoRa. In this step, a sequence of basic chirps are employed to de-chirp the incoming signal to produce a simple frequency shift keying (FSK) representation [43], [44]. We adopt this principle and formulate a sequence consisting of basic upchrips and downchirps, as the example depicted in Fig. 4, to de-chirp $r[n]$. The underlying assumption is that an ideal de-chirped $r[n]$ predominantly comprises frequency components centered around 0 Hz, except for those from synchronization symbols.

B. SIGNAL REPRESENTATION METHODS

To improve the detectability, the CWT is utilized to transform the raw pre-processed $r[n]$ into a two-dimensional representation. When applying the CWT to complex signals (with negative frequency components), two normalized scalograms are produced, one containing CWT estimations for wavelets of positive scales and another for the negative. Since both of these scalograms are generated based on the same complex signal, merging them into a single scalogram matrix can be achieved by flipping the negative frequency scalogram and then concatenating it with the positive one, as illustrated in Fig. 5.

Although the resulting scalogram can be directly utilized as a 2D representation for training the neural network, if de-chirping is applied prior to the CWT transformation, any fingerprint stemming from physical layer impairments manifests as non-DC deviations.

Another optional step that can be employed is to stack the repeated symbols in the preamble. This will improve the SNR and allow a simpler signal representation for deep

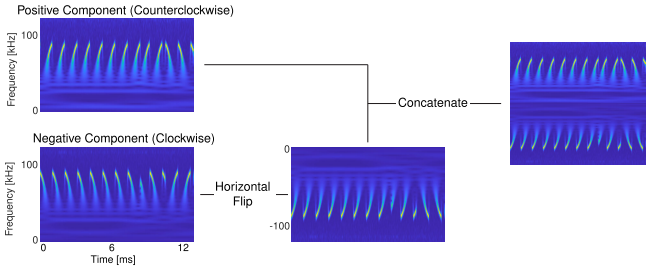

FIGURE 5. Illustration example on concatenating negative/positive frequency CWT scalograms of a complex signal into a single matrix.

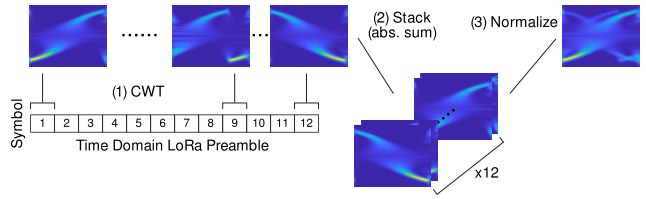

FIGURE 6. Illustration example on the generation of a *stackCWT* representation, when considering $N_s = 12$.

learning. With known SF and B, individual LoRa symbols can be directly segmented from the synchronized time domain signal. We then apply CWT to each symbol independently and overlap the yielded scalograms to produce a stacked modality as demonstrated in Fig. 6. The resulting stacked scalogram S_s can be obtained as follows

$$S_s[a,b] = \frac{1}{N_s}\sum_{i=1}^{N_s} |S_i[a,b]|\,, \tag{9}$$

where S_i is the scalogram of the i^{th} individual symbol, a and b are the wavelet scales and shifting parameter. The range of a is subject to the selection of filter banks, and the maximum value of b is equivalent to the number of samples within a signal segment, which, in this particular case, is further given as $b = L_{sym}$.

By exploring various combinations, we propose four potential signal representations (modalities), namely: (a) the original scalogram *rawCWT*, which is similar to the representations used in [41] and [42], (b) the stacked scalogram *stackCWT*, (c) the de-chirped scalogram *CWTd*, and (d) the de-chirped and stacked scalogram *stackCWTd*. A sample showcasing these four possible modalities is displayed in Fig. 7. It is important to note that the pseudo-colors are used for illustrative purposes only, and the actual scalogram is saved as a 2D matrix normalized to the range [0,1]. The 2D representations are used due to their efficiency in presenting spatiotemporal information on how the signal varies during propagation.

In the context of LoRa, both transient and steady state RF impairments can be considered as potential features embedded in the proposed modalities. While previous literature has utilized impairments like CFO as one potential fingerprint,

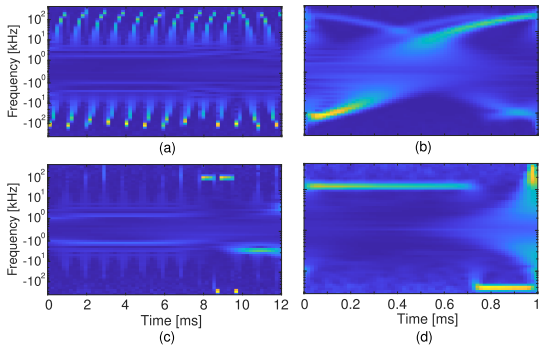

FIGURE 7. Examples of the four proposed CWT modalities: (a) *rawCWT*, (b) *stackCWT*, (c) *CWTd*, and (d) *stackCWTd* (SF = 7, B = 125 kHz, SNR=20 dB).

we compensate such relatively stochastic impairment during pre-processing to ensure better robustness. Consequently, our focus centers on two specific impairments (fingerprint sources): 1) transient imperfections resulting in random frequency components appearing in the low-frequency range and regions around inter-symbol transitions, and 2) IQ imbalance leading to varying frequency change patterns, impacting the density centers and gradients of different LoRa symbols.

C. DEEP LEARNING

The proposed framework considers detecting malicious spoofing and authenticating legitimate devices sequentially by utilizing different deep learning algorithms. We employ GANomaly, a variant of GAN, for spoofing detection alongside CNN for authentication. The GAN block is favorable as it addresses several limitations in CNNs trained for classification tasks. A CNN trained for classification invariably seeks to classify inputs to known classes, even if the input comes from an unknown class (spoofing device). Consequently, leaving a sole CNN insufficient for effective spoofing detection. Moreover, GANs are good candidates for handling data imbalance problems, allowing for decent performance when only data from limited scenarios are available during training.

A typical GAN network comprises two key components: a generator network G and a discriminator network D. The training process of a GAN is adversarial, with G and D being trained simultaneously and competing against each other to improve their own performance. The G learns to produce synthetic data that deceives D, while D aims to differentiate between the data generated by G and genuine inputs. By trained solely using legitimate (true) samples, a trained GAN exclusively learns the distribution of true samples and is proved to produce poor reconstructions when presented with spoof (rogue) samples [45].

As illustrated in Fig. 8, the architecture of GANomaly incorporates several components. Its generator G starts with a typical conventional deep convolutional GAN (DCGAN) structure named generator decoder G_D and is extended by two additional blocks: generator encoder G_E and E. The discriminator D follows the structure typical of a DCGAN

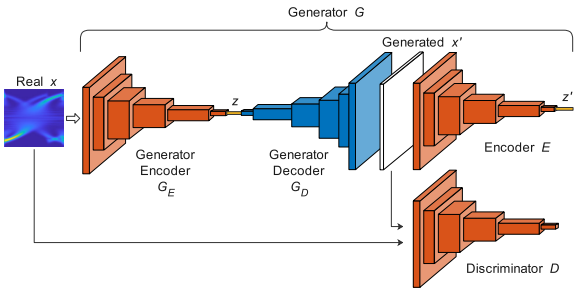

FIGURE 8. Architecture of the utilized GANomaly for spoof detection.

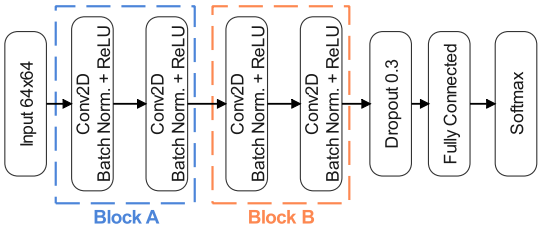

FIGURE 9. Architecture of the CNN to accommodate authentication, with an input size [64 x 64 x 1] used in this work.

discriminator. The principle of GANomaly largely follows a general GAN, while the main difference lies in its training aims at minimizing the following objective (loss) function

$$L = w_{\text{adv}} L_{\text{adv}} + w_{\text{con}} L_{\text{con}} + w_{\text{enc}} L_{\text{enc}}, \tag{10}$$

where w_{adv}, w_{con} and w_{enc} are adjustable weighting parameters, L_{adv}, L_{con} and L_{con} denotes the adversarial loss, contextual loss and encoder loss, respectively. More specifically,

$$L_{\text{adv}} = \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{x}}(\mathbf{x})} || f(\mathbf{x}) - \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{x}}(\mathbf{x})} f(G(\mathbf{x})) ||_2 \tag{11}$$

calculates the feature statistics matching error between real $\mathbf{x}$ and reconstructed $\mathbf{x}'$ at D's intermediate layer $f(\cdot)$ using Euclidean norm,

$$L_{\text{con}} = \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{x}}(\mathbf{x})} || \mathbf{x} - G_{\text{D}}(G_{\text{E}}(\mathbf{x})) ||_1 \tag{12}$$

measures the pixel level difference between $\mathbf{x}$ and $\mathbf{x}'$ through the Manhattan norm, and

$$L_{\text{enc}} = \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{x}}(\mathbf{x})} || G_{\text{E}}(\mathbf{x}) - E(G_{\text{D}}(G_{\text{E}}(\mathbf{x}))) ||_2 \tag{13}$$

measures the Euclidean distance between $\mathbf{z}$ and reconstructed $\mathbf{z}'$ within the latent space. Since the implementation and training principle of GANomaly is not a contribution of this work, we encourage the readers to refer to the original literature [46] for more detail.

The key hyperparameters of the GANomaly architecture used in this work are summarized in Table 3, where certain layers like input and output layers are omitted for simplicity. Diffrent from the original publication, we slightly modify hyperparameters such as the number of layers, size of the filters (kernels) and the number of channels for better performance. Once trained, only the generator G is used for spoof detection. When passing a sample through G, an anomaly score is calculated to assess the likelihood of spoofing by measuring the reconstruction error between latent representations z and z'. For arbitrary input $\tilde{\mathbf{x}}$, the anomaly score $A(\tilde{\mathbf{x}})$ is obtained as

$$A(\tilde{\mathbf{x}}) = || G_{\text{E}}(\tilde{\mathbf{x}}) - E(G_{\text{D}}(G_{\text{E}}(\tilde{\mathbf{x}}))) ||_1. \tag{14}$$

All $A(\tilde{\mathbf{x}})$s obtained from the test batch are further normalized to be in a range of [0,1].

Given that the sources of impairment interact in a nonlinear manner, manually determining learnable patterns remains challenging. Based on this consideration, we choose deep learning methods over conventional machine learning. Such an approach helps reduce the complexity of post-processing by minimizing the need for manual feature engineering. Furthermore, as evaluated in our previous work [10], for inputs presented using 2D modalities instead of time series, a network based on LSTM is generally more complex (e.g., with more total learnable parameters) and harder to train. Therefore, we employ a CNN with architecture illustrated in Fig. 9 to perform device authentication. The CNN shares the same input size as the GANomaly network and is structured using a multi-block design, where the convolutional layers within each block share identical hyperparameters. Batch normalization and ReLU activation are applied to all 2D convolutional layers. A dropout layer with a 30% drop rate is adopted accordingly to prevent overfitting, followed by one fully connected (hidden) layer prior to output layers. Notably, no pooling layers are utilized, and the feature map's dimension reduction is achieved by setting a stride of 2 for all convolutional layers. Hyperparameters like each convolutional block's kernel sizes and filter numbers are optimized via Bayesian optimization during training. The tuned hyperparameters for CNNs trained for the best-performing modalities are summarized in Table 5.

V. EXPERIMENTAL SETUP

This section summarizes the configurations of the experimental setup.

A. EXPERIMENTAL HARDWARE

We use two different types of commercially available LoRa shields from two different manufacturers representing our devices under test (DUT). Specifically, as outlined in Fig. 10, we acquired ten LoRa PHY shields (Semtech SX1276 chipset) from Duinotech and another ten LoRaWAN shields (Semtech SX1262 chipset) from Dragino. Both shields are mounted onto the same Arduino Uno and linked to MATLAB to function as transmitters. Additionally, we employ an ADALM-PLUTO software-defined Radio (SDR) from Analog Devices as the receiver due to its configuration flexibility. The Rx actively monitors transmissions centered at f_c = 916.5 MHz (as part of the license-free ISM band in Australia)

TABLE 3. GANomaly hyperparameters.

Generator Encoder[1] (G_E)		Generator Decoder[2] (G_D)		Encoder[1] (E)		Discriminator[1] (D)	
Layers	Parms.	Layers	Parms.	Layers	Parms.	Layers	Parms.
Conv_2D	5×5-8	Transpose Conv_2D	5×5-128	Conv_2D	5×5-8	Dropout	0.3
Conv_2D	5×5-16	Transpose Conv_2D	5×5-64	Conv_2D	5×5-16	Conv_2D	3×3-8
Conv_2D	5×5-32	Transpose Conv_2D	5×5-32	Conv_2D	5×5-32	Conv_2D	3×3-16
Conv_2D	5×5-64	Transpose Conv_2D	5×5-16	Conv_2D	5×5-64	Conv_2D	3×3-32
Conv_2D	5×5-128	Transpose Conv_2D[3]	5×5-1	Conv_2D	5×5-128	Conv_2D	3×3-64

[1] Batch Normalization and LeakyReLU activation. [2] Batch Normalization and ReLU activation.
[3] As a special case, this layer uses Batch Normalization and Tanh activation.

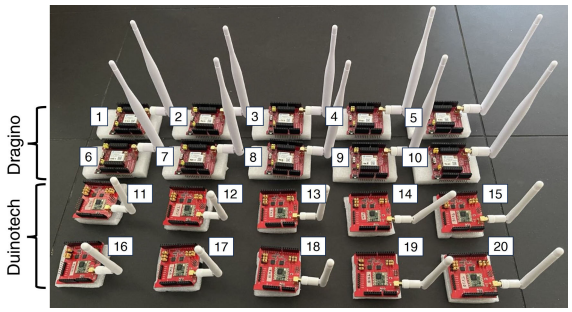

FIGURE 10. Experimental LoRa devices under test, where 1st and 2nd row is Dragino LA66 LoRaWAN Shield, while 3rd and 4th row belongs to Duinotech XC4392 LoRa Shield.

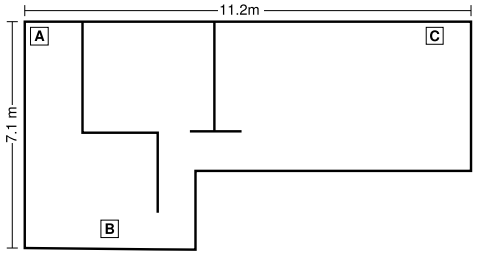

FIGURE 11. Approximate floor plan of the experimental room, where "A" is the fixed Rx position, "B" is the LoS Tx area, and "C" is the NLoS Tx area.

with a sampling frequency of $f_s = 1$ MHz. For robust and consistent performance, the same SDR is consistently used with its auto-gain controller (AGC) disabled.

B. DATA ACQUISITION

The data collection occurs within a residential room whose rough floorplan is illustrated in Fig. 11, where Rx is placed at a fixed location *A* while the Txs are placed at either location *B* or *C* depending on desired channel conditions, i.e., LoS or NLoS. The dataset of LoS and NLoS transmissions are collected independently, where the distance between two end devices in LoS scenarios is approximately 7m, and 11m for the NLoS scenarios. Owing to the operating frequency band and the adoption of CSS, LoRa signals are more resilient against noise and losses. As such, they have better performance under low signal-to-noise-ratio conditions, which allows the operation in deep indoor environments, urban canyons, and even underground locations. Hence, the slight difference in the transmission distance will not cause significant variation in the received signals. In the meantime, the power normalization techniques outlined in Section IV-A help ensure consistent signal power and a controllable SNR range, further retaining the validity of the collected data under both scenarios.

As indicated in [13], oscillators can be affected by chip heating, leading to fluctuations in CFO. To ensure the uniformity of data collection, all data collection is performed under an air-conditioned environment in an attempt to maintain consistent ambient temperature.

All Tx shields are configured with SF=7, B=125 kHz and with a transmit power P_{tx}=5 dBm. Given that our investigation centers on the signal portion appended by the Tx handler, only random bits are generated as the payload and the same payload is reused by all DUTs for all transmissions. Upon the acquisition of $r[n]$, we only retain the preamble portion while the payload is discarded. In total, approximately 5000 samples per Tx DUT per channel scenario are collected across multiple days within a month to account for possible environmental variations.

Lastly, all the key parameters and configurations used in this work to implement the experimental setup are summarized in Table 4 for easy reference.

TABLE 4. Table of key parameters.

Symbol	Description	Value
$f_s = 1/T_s$	Rx sampling frequency	1 [MHz]
f_c	Tx & Rx carrier frequency	916.5 [MHz]
P_{tx}	Tx transmit power	5 [dBm]
γ	CWT symmetry parameters	3
P^2	CWT time-bandwidth product	90
SF	LoRa spreading factor	7
B	LoRa Bandwidth	125 [kHz]
$T_{sym} = 2^{SF}/B$	LoRa symbol period	1 [ms]
$L_{sym} = T_{sym} f_s$	LoRa symbol length	1000
N_s[1]	Number of symbols	12
L^2	Length of reference sequence	12250

[1] Within the signal portion of interest.
[2] When considering 12.25 symbols ($12.25 \times L_{sym}$).

C. TRAINING SETUP

This work's processing and training are conducted using MATLAB on a desktop PC with an Intel Core i7 10700KF

CPU and a single NVIDIA RTX GeForce 2070s GPU. To balance the computational complexity and the amount of learnable RF fingerprints, we adopt an input size of $[64 \times 64 \times 1]$ for all generated representations and for both GANomaly and CNN networks. Independent sets of GAN and CNN are separately trained for each of the proposed signal representations. To ensure fair comparisons, the hyperparameters of their corresponding networks are individually optimized, following the same manner described in Section IV-C. For GANomaly training in particular, we only involve samples from the same DUT at a time. This setup simulates a countermeasure against the relay (i.e., man-in-the-middle) attack, where all transmissions not from the legitimate device are considered spoofing. To evaluate behavior differences across DUTs, we train multiple copies of GANomaly while considering one different DUT as the legitimate class each time (i.e., we conducted an iteration on the legitimate DUT). For both deep learning algorithms, each training round runs for a maximum of 50 epochs with a minibatch size set at 256 and uses the Adam optimizer as the solver.

During the training phase, 80% of the involved samples are randomly selected as the training set, while the remaining 20% are reserved for testing, i.e., they are not shown to the classifier during the training phase. Moreover, the training phase involves data with SNR levels of 0 dB and 10 dB as a mean of providing relatively adequate variations within the training set while maintaining acceptable signal quality. Samples with other SNR levels are used only for testing unless specified otherwise.

VI. MAIN RESULTS AND DISCUSSION

Results presented in this section utilize the dataset acquired from all 20 Tx DUTs under the NLoS channel and are transformed into the *stackCWT* representation. This modality is selected as it reflects the best performance under a practical scenario. In addition, Section VII provides supplement results that cover the performance comparison between the proposed modalities and representations introduced in existing literature.

A. SPOOF DETECTION

Given that GANomaly only outputs a normalized numerical value (i.e., the anomaly score) for any arbitrary input, a threshold must be selected to facilitate decision-making. Whereas the determination of appropriate thresholds hinges on the specific design requirements. Here we employ the area under the curve (AUC) derived from the receiver operating characteristic (ROC) curve as a metric to evaluate the trained network's overall performance.

The ROC curve maps the true positive rate (TPR) against the false positive rate (FPR) across various decision thresholds. These are two competing metrics, and, for example, the rise of TPR leads to increased FPR, which is an undesirable effect. In such case, AUC can capture the joining performance of these two metrics and is calculated by integrating the entire FPR range, as illustrated in Fig. 12. In addition, we compute

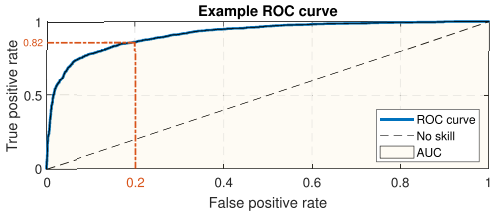

FIGURE 12. Visualization example of the utilized metrics (AUC and TPR).

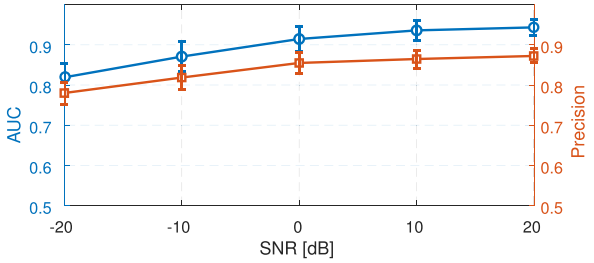

FIGURE 13. SNR sweeping results of spoof detection: mean and standard deviation of AUC and precision.

the precision, obtained as TPR/(TPR + FPR), at the single selected decision threshold that returns FPR=0.2 to serve as an additional metric. Although this particular threshold might not return the highest TPR, it limits the occurrence of Type 1 errors (i.e., false alarms) and ensures robust system behavior.

The trained model is tested using samples across the full SNR range, and the outcome of this SNR sweep is depicted in Fig. 13, where the line charts present the average metric values calculated among all 20 Tx DUTs and the error bars show the standard deviation of corresponding metrics. The standard deviation is introduced to visualize the performance variations between different Tx devices.

Based on the results in Fig.13, GANomaly demonstrates an adequate overall spoof detection performance, especially for SNR larger than 0 dB. For instance, the highest average AUC of 0.946 and the highest average precision of 0.882 are both achieved at SNR=20 dB. In addition, performance increment as SNR level increases can be observed in both curves in the form of increased metric values and gradual reduction of the standard deviation. This behavior is as expected, given low random noise effect in the signal. Similarly, the relatively worse performance in the low SNR region is limited by the high random variation in the signal that restricts the detectability of potential RF fingerprints, such as the short transients between adjacent symbols.

B. AUTHENTICATION

The authentication performance is verified using the CNN presented in Figure 9 (from Section IV-C) and trained following hyperparameters summarized in Table 5 (from Section VII-B). We attempt to train a single CNN to authenticate all Tx DUTs, given it is impractical to determine the manufacturer group of a specific Tx before conducting

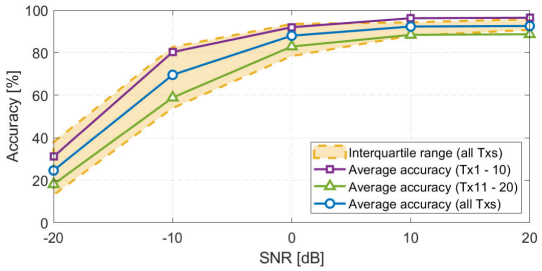

FIGURE 14. Average authentication accuracy at various SNRs. Accuracy is calculated either among all Tx DUTs or among Txs from the same manufacturer.

authentication. As such, the worst-case accuracy that could be possibly obtained is 5% (i.e., 1/20), equivalent to the performance of random guess, good classification results should far exceed this baseline.

The average authentication accuracy obtained from the SNR sweep is summarized in Fig. 14. The plot shows both the average accuracy across all 20 Transmitter Tx DUTs and the accuracy for devices from the same manufacturer. The shaded area represents the interquartile range (IQR), encompassing results distributed between the 25^{th} and 75^{th} quartiles (i.e., the middle half). The results demonstrate the feasibility of the proposed approach with an overall authentication accuracy surpassing 90% for SNRs$\geq$0 dB. The performance convergence is found at around SNR=10 dB, where an average accuracy of 92.4% is achieved among all Tx DUTs. When considering only the Dragino Tx DUTs (Tx1 - 10), the average accuracy at SNR=10dB reaches 96.1%. Furthermore, the IQR is also found to converge as SNR increases, indicating a more consistent performance across all Tx DUTs at relatively higher signal power. This performance increment is achieved as potential fingerprints, such as instantaneous inter-symbol transients, become more distinguishable as the impact of noise diminishes at higher SNR levels.

Furthermore, to better compare device-wise performance, the trained CNN is tested against a subset with samples from all 20 Tx DUTs at an SNR of 10 dB. The corresponding results are presented as a confusion matrix (CM) in Fig. 15. In line with the spoof detection results, primarily affected by Tx19 and Tx20, the Duinotech DUTs exhibit a relatively worse overall authentication performance than Dragino DUTs. This insufficiency implies that the process of fingerprint development appears to be an independent process for each device. Alternatively, the decisive fingerprints might require extra processing steps to be more comprehensively exploited.

However, it is noteworthy that, despite the variation in authentication accuracy among different devices, no devices are misclassified into the wrong manufacturer family. This observation indicates some decisive fingerprints might be manufacturer or chipset-oriented. Lastly, the results indicate that the proposed framework can efficiently exploit RF fingerprints from 2D cross-domain representations for the purpose of spoofing detection and authentication. There is a high potential that the same framework can be utilized for signals of other modulation schemes if presented using 2D modalities (e.g., STFT spectrograms of QAM or PSK signals). While these hypotheses are not thoroughly examined within this work, it might serve as a potential direction for future research.

VII. SUPPLEMENTARY RESULTS

This section presents additional results that support the experimental setups selected for Section VI. In particular, this section presents performance comparisons of different modalities and channel conditions. The additional tests in this section use a subset containing samples of six Txs randomly selected from the complete dataset obtained in Section V-B. This subset comprises data from five days, labeled as D1 to D5, with approximately 1000 samples per DUT. To assess the robustness of the trained networks, we train the deep learning algorithms exclusively on samples from D1 to D3 and test their performance on samples from D4 and D5. To simplify the testing scenario, the tests primarily focus on samples with a consistent SNR level of 10 dB.

A. IMPACT OF CHANNEL CONDITIONS

To validate the impact of channel condition, we conducted a comparison using CNNs trained and tested respectively using LoS and NLoS data, both transformed into the *rawCWT* representation. By following the same training setup outlined in Secion V-C, the obtained results, presented as CMs in Fig. 16, indicate that despite the NLoS scenario yielding a slightly worse overall accuracy, the discrepancy between the performance achieved under two channel conditions is not notably substantial.

This resilience can be attributed to a few factors. Firstly, adopting CSS modulation distributes the signal's spectrum across the entire bandwidth B, making the waveform less susceptible to frequency-selective fading that affects only certain frequency components. Secondly, preambles with duration at the millisecond level reduce the degree of impact by small-scale fading induced by multipath effects which is measured in a few nanoseconds range. Moreover, CWT facilitates visualization of channel-affected signals from various viewpoints, owing to its multi-resolution analysis ability.

Given that NLoS channels are more common in practical wireless access network applications, all subsequent testing will solely focus on data collected from NLoS scenarios.

B. MODALITY COMPARISONS

Lastly, we compare the potential modalities introduced in Section IV-B. For this evaluation, we concentrate on data acquired from the NLoS channel with CFO compensation in place. Furthermore, we compare our proposed modalities against the STFT spectrogram introduced in [13] and the channel-independent STFT (*ch-ind STFT*) technique presented in [27], along with a de-chirped variant of the

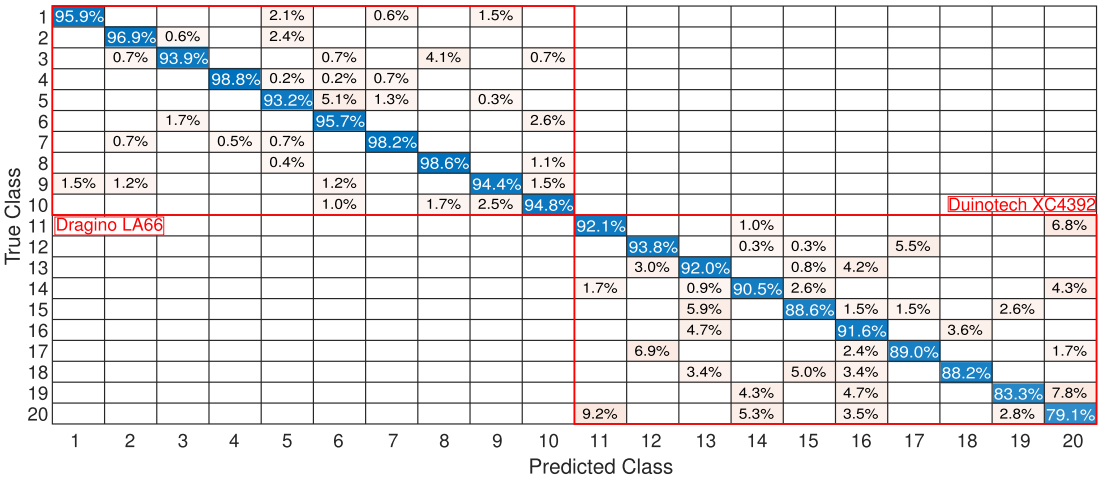

FIGURE 15. Authentication performance CM among all 20 Tx DUTs (10 Dragino LA66 + 10 Duinotech XC4392), measured at SNR=10 dB.

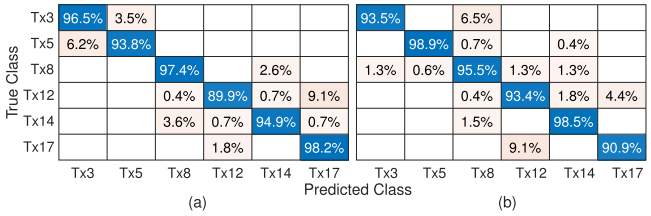

FIGURE 16. Resulting CMs when tested against the presence of the (a) LoS channel and (b) NLoS channel, with CFO compensation.

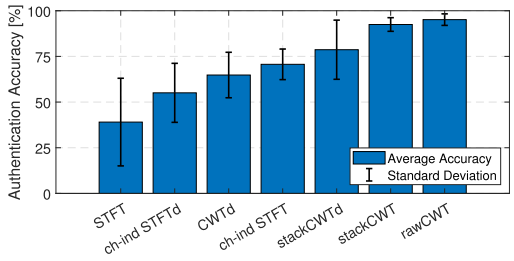

FIGURE 17. Average authentication accuracy and corresponding standard deviation when adopting different representation modalities.

channel-independent STFT (*ch-ind STFTd*). Notably, the CNN of each modality is independently trained, with hyperparameters separately optimized.

The average authentication results of all tested modalities are presented in Fig. 17 along with the standard deviation showcasing the performance difference across DUTs. Observed from the presented results, all the de-chirped methods exhibit worse performance than the representations generated directly from the original signal. This phenomenon can be attributed, at least in part, to factors such as the restricted range of interpretable frequencies in CWT due to limited available wavelet scales. This limitation particularly impacts low-frequency components, especially those around 0 Hz, that need to be detected using wavelets of larger scales. In the case of STFT, the performance discrepancy might be attributed to the fixed window size, which fails to effectively capture both high and low-frequency components simultaneously. Nonetheless, modalities that utilize CWT achieve respectable overall accuracy compared to STFT due to their inherent multi-resolution capability.

TABLE 5. CNN hyperparameters of best-performed modalities.

rawCWT		stackCWT	
Layers	Parms.	Layers	Parms.
Conv_2D_A	7×7-32	Conv_2D_A	3×3-16
Batch Norm. + ReLU	-	Batch Norm. + ReLU	-
Conv_2D_B	3×3-64	Conv_2D_B	5×5-32
Batch Norm. + ReLU	-	Batch Norm. + ReLU	-
Approx. Learnables[1]	128.1k	Approx. Learnables[1]	51.3k

[1] The amount of learnable is calculated based on 20 total classes.

From Fig. 17, we note that both *rawCWT* and *stackCWT* exhibit decent performance. Given their similar performance, a comparison of their respective CNN hyperparameters is detailed in Table 5 for further insights. Notably, the network optimized for *stackCWT* is found to be less complicated (i.e., characterized by fewer total learnable parameters) while delivering equivalent performance. Accordingly, we compare several operating metrics between the two modalities and summarize the results in Table 6. The results present the average relative difference between the two modalities, where a negative number indicates the metric value of *stackCWT* is less than the measurement for *rawCWT* by a certain percentage, and vice versa. Note the comparison is conducted under the setup explained in Section V-C. Based on the comparison, it is clear that despite the training and prediction of *stackCWT*'s model being relatively more time-consuming, it excels in faster modality generation and significantly reduces the memory space required to achieve a similar level

TABLE 6. Performance metrics of stackCWT (relative to rawCWT).

Metric	Relative percentage [%]
Modality generation time [s]	-11.43
Modality storage size [KBs]	-18.07
Training converge iterations[1]	14.39
Trained model storage size [KBs]	-60.02
Trained model static memory size [KBs] (After loaded into MATLAB)	-64.53
Classification time [s/input]	15.02

[1] Measured as the first iteration that achieves training accuracy $\geqslant 90\%$.

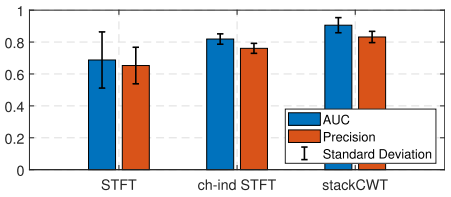

FIGURE 18. Average spoofing detection performance and corresponding standard deviation when adopting different representation modalities.

of performance. Based on these observations, we select *stackCWT* as the representation of focus in Section VI.

In addition, we compare the spoofing detection capabilities of *stackCWT* with modalities employed in [13], [27]. Utilizing the metrics outlined in Section VI-A, we visualize the average results obtained from iterating through six Txs in Fig. 18. The analysis demonstrates that *stackCWT* achieved the highest overall detection accuracy among all three modalities. However, compared to *ch-ind STFT*, its slightly higher standard deviation also indicates relatively worse robustness across various devices.

VIII. CONCLUSION

This work presented a spoof detection and authentication framework for IoT devices enabled by deep learning. Four data representation modalities were introduced to exploit device-identifiable RF fingerprints based on the uncoded preamble of the RF frame. To achieve spoof detection, we adopted the GANomaly architecture, while the CNN architecture was employed for performing authentication. As a practical example of IoT systems, we implemented the framework on commercially available LoRa modules. Based on the testing results, we showed that continuous wavelet transform (CWT) outperforms the short-time Fourier (STFT) representation, particularly when using the introduced stacked 2D modality *stackCWT*, a sequence of CWT snapshots. Under an adequate signal-to-noise ratio (SNR), a high accuracy of 92.4% was achieved for successful authentication, and an area under the curve (AUC) of 0.946 was achieved for spoof detection when involving 20 testing devices. Accordingly, the proposed framework shows a significant potential for practical applications in IoT networks. In addition, the testing results also indicate a dependency on the module manufacturer, which is naturally expected given the dependency on the fabrication tolerances and performance stability. Future work will evaluate the framework under dynamic radio channels and explore other potential classifier architectures and fingerprint representations.

REFERENCES

[1] M. A. Ferrag, L. A. Maglaras, H. Janicke, J. Jiang, and L. Shu, "Authentication protocols for Internet of Things: A comprehensive survey," *Secur. Commun. Netw.*, vol. 2017, pp. 1–41, 2017.

[2] L. Kane, V. Liu, M. McKague, and G. R. Walker, "Network architecture and authentication scheme for LoRa 2.4 GHz smart homes," *IEEE Access*, vol. 10, pp. 93212–93230, 2022.

[3] J. Zhao, Q. Li, J. Sun, M. Dong, K. Ota, and M. Shen, "Efficient IoT device identification via network behavior analysis based on time series dictionary," *IEEE Internet Things J.*, vol. 11, no. 3, pp. 5129–5142, Feb. 2024.

[4] Y. Meidan et al., "N-BaIoT—Network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Comput.*, vol. 17, no. 3, pp. 12–22, Jul. 2018.

[5] N. Xie, Z. Li, and H. Tan, "A survey of physical-layer authentication in wireless communications," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 1, pp. 282–310, 1st Quart., 2021.

[6] J. Lee et al., "PUFTAP-IoT: PUF-based three-factor authentication protocol in IoT environment focused on sensing devices," *Sensors*, vol. 22, no. 18, p. 7075, Sep. 2022.

[7] A. Jagannath, J. Jagannath, and P. S. P. V. Kumar, "A comprehensive survey on radio frequency (RF) fingerprinting: Traditional approaches, deep learning, and open challenges," *Comput. Netw.*, vol. 219, Dec. 2022, Art. no. 109455.

[8] M. Köse, S. Tascioglu, and Z. Telatar, "RF fingerprinting of IoT devices based on transient energy spectrum," *IEEE Access*, vol. 7, pp. 18715–18726, 2019.

[9] D. Huang, A. Al-Hourani, K. Sithamparanathan, W. S. T. Rowe, L. Bulot, and A. Thompson, "Deep learning methods for device authentication using RF fingerprinting," in *Proc. 15th Int. Conf. Signal Process. Commun. Syst. (ICSPCS)*, Sydney, NSW, Australia, Dec. 2021, pp. 1–7.

[10] D. Huang, A. Al-Hourani, K. Sithamparanathan, and W. S. T. Rowe, "Deep learning methods for IoT device authentication using symbols density trace plot," *IEEE Internet Things J.*, vol. 11, no. 10, pp. 18167–18179, May 2024.

[11] L. Peng, J. Zhang, M. Liu, and A. Hu, "Deep learning based RF fingerprint identification using differential constellation trace figure," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 1091–1095, Jan. 2020.

[12] J. Li, Y. Ying, C. Ji, and B. Zhang, "Differential contour stellar-based radio frequency fingerprint identification for Internet of Things," *IEEE Access*, vol. 9, pp. 53745–53753, 2021.

[13] G. Shen, J. Zhang, A. Marshall, L. Peng, and X. Wang, "Radio frequency fingerprint identification for LoRa using deep learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2604–2616, Aug. 2021.

[14] D. Roy, T. Mukherjee, M. Chatterjee, E. Blasch, and E. Pasiliao, "RFAL: Adversarial learning for RF transmitter identification and classification," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 2, pp. 783–801, Jun. 2020.

[15] J. Song et al., "End-to-end automatic differentiation of the coronavirus disease 2019 (COVID-19) from viral pneumonia based on chest CT," *Eur. J. Nucl. Med. Mol. Imag.*, vol. 47, no. 11, pp. 2516–2524, Oct. 2020.

[16] K. Liu, A. Li, X. Wen, H. Chen, and P. Yang, "Steel surface defect detection using GAN and one-class classifier," in *Proc. 25th Int. Conf. Autom. Comput. (ICAC)*, Sep. 2019, pp. 1–6.

[17] D. Roy, T. Mukherjee, A. Riden, J. Paquet, E. Pasiliao, and E. Blasch, "GANSAT: A GAN and satellite constellation fingerprint-based framework for GPS spoof-detection and location estimation in GPS deprived environment," *IEEE Access*, vol. 10, pp. 45485–45507, 2022.

[18] J. Yu, Y. Song, D. Tang, D. Han, and J. Dai, "Telemetry data-based spacecraft anomaly detection with spatial–temporal generative adversarial networks," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, 2021.

[19] Y. Yang, A. Hu, and J. Yu, "A practical radio frequency fingerprinting scheme for mobile phones identification," *Phys. Commun.*, vol. 55, Dec. 2022, Art. no. 101876.

[20] Y. Li, Y. Ding, G. Goussetis, and J. Zhang, "Power amplifier enabled RF fingerprint identification," in *Proc. IEEE Texas Symp. Wireless Microw. Circuits Syst., Making Waves Texas (WMCS)*, May 2021, pp. 1–6.

[21] K. St. Germain and F. Kragh, "Physical-layer authentication using channel state information and machine learning," in *Proc. 14th Int. Conf. Signal Process. Commun. Syst. (ICSPCS)*, Dec. 2020, pp. 1–8.

[22] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, "Machine learning for the detection and identification of Internet of Things devices: A survey," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 298–320, Jan. 2022.

[23] A. Aghnaiya, Y. Dalveren, and A. Kara, "On the performance of variational mode decomposition-based radio frequency fingerprinting of Bluetooth devices," *Sensors*, vol. 20, no. 6, p. 1704, Mar. 2020.

[24] H. Yuan et al., "Stable nonlinear and IQ imbalance RF fingerprint for wireless OFDM devices," 2021, *arXiv:2104.10397*.

[25] G. Oligeri, S. Sciancalepore, S. Raponi, and R. D. Pietro, "PAST-AI: Physical-layer authentication of satellite transmitters via deep learning," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 274–289, 2023.

[26] Y. Jiang, L. Peng, A. Hu, S. Wang, Y. Huang, and L. Zhang, "Physical layer identification of LoRA devices using constellation trace figure," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, Dec. 2019, Art. no. 223.

[27] G. Shen, J. Zhang, A. Marshall, and J. R. Cavallaro, "Towards scalable and channel-robust radio frequency fingerprint identification for LoRa," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 774–787, 2022.

[28] T. Zhang, P. Ren, D. Xu, and Z. Ren, "DFSNet: Deep fractional scattering network for LoRa fingerprinting," in *Proc. GLOBECOM IEEE Global Commun. Conf.*, Dec. 2022, pp. 4897–4902.

[29] X. Wang, L. Kong, Z. Wu, L. Cheng, C. Xu, and G. Chen, "SLoRa: Towards secure LoRa communications with fine-grained physical layer features," in *Proc. 18th Conf. Embedded Netw. Sensor Syst.*, Nov. 2020, pp. 258–270.

[30] J. Li, X. Zhu, M. Ouyang, W. Li, Z. Chen, and Q. Fu, "GNSS spoofing jamming detection based on generative adversarial network," *IEEE Sensors J.*, vol. 21, no. 20, pp. 22823–22832, Oct. 2021.

[31] Z. Chen, L. Peng, A. Hu, and H. Fu, "Generative adversarial network-based rogue device identification using differential constellation trace figure," *EURASIP J. Wireless Commun. Netw.*, vol. 2021, no. 1, Dec. 2021, Art. no. 72.

[32] S. Halder and T. Newe, "Radio fingerprinting for anomaly detection using federated learning in LoRa-enabled industrial Internet of Things," *Future Gener. Comput. Syst.*, vol. 143, pp. 322–336, Jun. 2023.

[33] M. Ezuma, F. Erden, C. K. Anjinappa, O. Ozdemir, and I. Guvenc, "Detection and classification of UAVs using RF fingerprints in the presence of Wi-Fi and Bluetooth interference," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 60–76, 2020.

[34] Semtech. (2019). *RP2-1.0.3 LoRaWAN® Regional Parameters*. [Online]. Available: https://lora-alliance.org/resource_hub/rp2-1-0-3-lorawan-regional-parameters/

[35] K. Merchant and B. Nousain, "Securing IoT RF fingerprinting systems with generative adversarial networks," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Nov. 2019, pp. 584–589.

[36] I. Ullah and Q. H. Mahmoud, "A framework for anomaly detection in IoT networks using conditional generative adversarial networks," *IEEE Access*, vol. 9, pp. 165907–165931, 2021.

[37] Y. Ouyang, B. Li, Q. Kong, H. Song, and T. Li, "FS-IDS: A novel few-shot learning based intrusion detection system for SCADA networks," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[38] C. Lu, X. Wang, A. Yang, Y. Liu, and Z. Dong, "A few-shot-based model-agnostic meta-learning for intrusion detection in security of Internet of Things," *IEEE Internet Things J.*, vol. 10, no. 24, pp. 21309–21321, Nov. 2023.

[39] C. Liu et al., "Overcoming data limitations: A few-shot specific emitter identification method using self-supervised learning and adversarial augmentation," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 500–513, 2024.

[40] Z. Cai, W. Ma, X. Wang, H. Wang, and Z. Feng, "The performance analysis of time series data augmentation technology for small sample communication device recognition," *IEEE Trans. Rel.*, vol. 72, no. 2, pp. 574–585, Jun. 2023.

[41] G. Baldini, C. Gentile, R. Giuliani, and G. Steri, "Comparison of techniques for radiometric identification based on deep convolutional neural networks," *Electron. Lett.*, vol. 55, no. 2, pp. 90–92, Jan. 2019.

[42] 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 *Proc. Global IoT Summit (GIoTS)*, Jun. 2019, pp. 1–6.

[43] B. Al Homssi, K. Dakic, S. Maselli, H. Wolf, S. Kandeepan, and A. Al-Hourani, "IoT network design using open-source LoRa coverage emulator," *IEEE Access*, vol. 9, pp. 53636–53646, 2021.

[44] K. Dakic, B. Al Homssi, A. Al-Hourani, and M. Lech, "LoRa signal demodulation using deep learning, a time-domain approach," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Apr. 2021, pp. 1–6.

[45] T. Schlegl, P. Seebóck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Proc. Int. Conf. Inf. Process. Med. Imag.* Cham, Switzerland: Springer, May 2017, pp. 146–157.

[46] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "GANomaly: Semi-supervised anomaly detection via adversarial training," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, Dec. 2018, pp. 622–637.