IET Journals
The Institution of Engineering and Technology

An Investigation into the Impacts of Deep Learning-based Re-sampling on Specific Emitter Identification Performance

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Mohamed K. M. Fadul[1], *Donald R. Reising*[1*], *Lakmali P. Weerasena*[2]

[1] *Electrical Engineering Department, University of Tennessee at Chattanooga, 735 Vine Street, Chattanooga, TN, U.S.A.*
[2] *Department of Mathematics, University of Tennessee at Chattanooga, 735 Vine Street, Chattanooga, TN, U.S.A.*
* *E-mail: donald-reising@utc.edu*

Abstract: Increasing Internet of Things (IoT) deployments present a growing surface over which villainous actors can carry out attacks. This disturbing revelation is amplified by the fact that a majority of IoT devices use weak or no encryption at all. Specific Emitter Identification (SEI) is an approach intended to address this IoT security weakness. This work provides the first Deep Learning (DL) driven SEI approach that upsamples the signals after collection to improve performance while simultaneously reducing the hardware requirements of the IoT devices that collect them. DL-driven upsampling results in superior SEI performance versus two traditional upsampling approaches and a convolutional neural network only approach.

1 Introduction

It is anticipated that the number of deployed Internet of Things (IoT) devices will reach 75 billion by 2025 [1]. However, limited computational resources, high manufacturing costs, and scalability challenges associated with implementation and management of encryption has led most IoT devices ($\sim$70%) to weak employments or to forego it altogether [2]. This leaves the IoT devices and corresponding infrastructure exposed to abuse by nefarious actors [3]. Specific Emitter Identification (SEI) is a passive, physical layer means to secure IoT devices and the associated infrastructure [4, 5]. SEI exploits inherent, unique, and distinct features that are unintentionally imparted upon a radio's signals during their formation and transmission. These features are attributed to acceptable manufacturing discrepancies and differences present within the components, sub-systems, and systems that comprise the radio's Radio Frequency (RF) front-end. Although sufficient to discriminate radios of the same manufacture and model, SEI features do not interfere with a radio's normal operations. Recently, Deep Learning (DL) based SEI has been put forward, because it has been shown capable of learning discriminator features directly from the In-phase (I) and Quadrature (Q) values of collected, sampled signals [6]. This eliminates the need for handcrafted SEI that requires expert knowledge to carefully select signal transformations, feature generation and selection techniques, and other practices that maximize performance. Despite its successes, SEI performance is greatly influenced by the receiver's capabilities such as sampling rate. Typically, SEI makes use of high sampling rates to ensure the nuances of a given radio's signal features are sufficiently captured. However, this is problematic when considering IoT devices, because they may be incapable of generating sampling rates high enough to capture these nuances. If this occurs, then SEI performance will be degraded and lead to radios being misidentified, which can result in network access being granted to unauthorized IoT devices and users.

This work proposes the novel use of a conditional Generative Adversarial Network (cGAN) to increase the sampling rate of signals collected at lower sampling rates prior to DL-based SEI [7]. The goal is to improve SEI performance without increasing the hardware requirements (e.g., memory, clock rate) of the IoT devices that collect the signals. The use of a cGAN permits generation of higher sampled signals from the learned SEI feature distributions associated with each of the radios being identified while eliminating the need to learn Markov Chain parameters [8]. The proposed cGAN-based upsampling approach is compared with two traditional interpolation techniques–described in [9]–that are each applied to the same set of IoT device signals.

The remainder of this article is organized as follows: Sect. 2 provides a brief summary of related works and describes how they differ from the work presented in this article; Sect. 5 describes the signal of interest, the employed deep learning architectures, as well as descriptions of two, traditional interpolation techniques used to facilitate comparative assessment; Sect. 6 outlines the process developed to generate the results in Sect. 7; and the conclusion is presented in Sect. 8.

2 Related Works

A key motivating factor for DL-based SEI is the elimination of feature-engineering or domain-specific, expert knowledge. DL-based SEI has been well investigated by multiple efforts [6, 10–45]. The work in [34, 36] proposes semi-supervised SEI approaches based upon GANs. The authors of [34] extract the the received RF signals' features using a representation network and then train a Triple-GAN network using the extracted features. Feedback learning assists the representation network to learn more discriminative SEI features while the representation network finds a better raw data representation to improve the performance of the Triple-GAN and SEI classification system. The work in [36] performs SEI using a Information maximized GAN (InfoGAN) and applies Radio Frequency Fingerprint Embedding (RFFE) to integrate each received signal's bispectrum histogram associated features into the InfoGAN's training process. The work in [38] adopts a Complex-Valued Neural Network (CVNN) for IoT radio identification using complex-valued, baseband signals. It also uses Neural Network (NN) based compression to reduce the dimensionality of the baseband signals for a more computationally efficient CVNN implementation. The work in [20] presents an SEI approach that identifies radios using a Deep Residual Network (DRN) and grey-scale images of the received signals' Hilbert spectrum. A received signal's Hilbert spectrum is generated by decomposing each received signal into a finite number of intrinsic mode functions via the Hilbert-Huang Transform (HHT). The DRN learns discriminatory features from the input grey-scale images to enable identification of one radio over another. The compression system in [38] and DRN architecture in [20] aim to reduce the complexity of the SEI model, but not the frequency used to sample the baseband signals. These approaches differ from our approach in that none of the reviewed published efforts specifically reduce the SEI

hardware complexity by reducing the sampling rate at which an IoT device collects signals for subsequent SEI processing.

3 Problem Definition

Traditional IoT networks connect many devices and sensors which are assumed to send and receive, and process a few bytes per day at low data rates. This requirement is satisfied by designing the devices to measure and communicate a few values to match the IoT communication technology limits [46]. Satisfying that requirement becomes more challenging for high sampling rate and data intensive applications. The work in [20, 38] considers less computationally complex architectures and learning algorithms for a more efficient implementation of SEI in IoT devices. These efforts did not consider reducing the computational requirement by reducing the sampling rate at which an IoT device collects signals for subsequent SEI.

4 Objective

The goal of this work is to study SEI performance when an IoT device's sampling rate is reduced to decrease computational complexity. This work presents a DL-based approach that leverages a conditional GAN architecture to improve SEI performance at lower sampling rates. Our contributions to the current level of SEI understanding are as follows:

- Using adversarial training provided by conditional GAN architecture to upsample IoT waveform while preserving/improving SEI discriminative features.
- Analyzing the effect of low sampling rate on the performance of SEI to discriminate between IoT devices that communicate using IEEE 802.11a.
- Using CGAN-based architecture to improve the traditional low sampling rate SEI performance by an average of 4% when sampling rate is reduced by a factor of 8.

5 Background

This section provides the necessary background information on the signal of interest, employed deep learning algorithms as well as two interpolation techniques used to facilitate comparative assessment.

5.1 Signal of Interest

The results presented herein are generated using the IEEE 802.11a Wi-Fi preambles from our work in [47]. The use of IEEE 802.11a Wi-Fi signals is due to (i) its designation as a IoT communications standard [48], (ii) its use in other SEI work [6, 47, 49], and (iii) the fact that current and future communication systems are built upon the Orthogonal Frequency Division Multiplexing (OFDM) scheme [50]. The data set consists of 8,000 IEEE 802.11a Wi-Fi preambles collected from four Cisco AIR-CB21G-A-K9 radios using an Agilent spectrum analyzer at a sampling rate of 20 MHz.

5.2 Deep Learning Architectures

This section provides a brief description of each DL architecture used in this work.

5.2.1 Autoencoder (AE): An AE is a deep generative model that attempts to reconstruct the input data at its output layer [51]. An AE consists of an input, hidden, and output layer. Functionally, its comprised of two parts: an encoder and decoder. The encoder generates a representation (a.k.a., a code) of the input data. The encoder is similar to a Convolutional Neural Network (CNN), but without the fully connected layers. The decoder reconstructs the input using the encoder generated code [51, 52]. An AE does not perfectly copy the input data at its output, but learns how to approximately recover training-like input data under certain restrictions. This allows an AE to learn a good representation and useful properties of the input data [51]. When compared to traditional Multi-Layer Perceptron (MLP) networks, AEs learn a compressed representation from unlabeled, input data using an unsupervised scheme [53, 54].

This work uses a Convolutional AE (CAE) for its ability to handle multi-dimensional data such as the complex-valued signals that result from analog-to-digital conversion within the signal collection process. The collected signal's In-phase and Quadrature (IQ) values are restructured to form the first and second rows of a multi-dimensional input, respectively. The CAE's encoder consists of convolutional layers that extract the features and pooling layers to reduce the dimensionality of the resulting feature maps [52]. The hidden layer is constrained to be smaller than that of the input layer, thus forcing the CAE to learn the most efficient representation of the input data [51]. Its decoder upsamples and reconstructs the input using de-convolutional and un-pooling layers and the code from the hidden layer.

5.2.2 Convolutional Neural Networks (CNN): A CNN is constructed by prepending a feed-forward MLP network with one or more convolutional and pooling layers, which extract and resize multi-dimensional data such as images, or in our case, the IQ values associated with sampled signals [15, 53]. Each convolutional layer generates a feature map from its input, and an activation function performs a linear/nonlinear transformation for each node in the feature map. This work uses the Rectified Linear Unit (ReLU) as the activation function. The activated feature maps are passed to the pooling layer for dimensionality reduction. After one or more convolutional, activation, and pooling layers, dense layers extract higher-level features from the feature maps [6]. Finally, the the output layer assigns the learned features to one of the classes that represent each of the four radios.

5.2.3 Generative Adversarial Network (GAN): A GAN uses unsupervised or semi-supervised adversarial training to learn deep representations of the training data [8, 55]. A GAN learns these representations by jointly training two deep networks: the generator, G, and discriminator, D. The G learns the training data distribution to generate new data samples that minimize the D's probability of correctly determining the origin of a sample. The G can be represented by any deep generative network such as a CAE. The D aims to maximize the probability of correctly determining whether or not an input sample came from the training data or was produced by the G [8]. The D can be implemented using any discriminative network such as a CNN. If MLP-like networks are used for both the G and the D, then the entire system can be trained using back-propagation [8, 55].

If the input to the G is z with prior probability $P_z(z)$, then the function that maps the input to the output is given by $G(z;\ \theta_g)$, with θ_g being the parameters of G. The D's function is given as $D(x;\ \theta_d)$, and its output is a single value representing the probability that the input x is from the training data rather than the G [8]. The GAN's minimax optimization problem is represented by the objective function given by,

$$\min_G \max_D V(D,G) = E_{x\sim P_{\rm d}(x)}\{\log[D(x)]\} + E_{z\sim P_z(z)}\{\log[1-D(G(z))]\}. \quad (1)$$

where E is the expected value. During training, the optimum point is reached when the G perfectly recovers the training data distribution (i.e., $P_g{=}P_{\rm d}$). In this work, z is IEEE 802.11a Wi-Fi preambles that have been collected at a lower sampling frequency than desired for SEI processing (e.g., 5 MHz versus 20 MHz). The goal is to train the G to recover the higher frequency sampled preamble—from its lower sampled version—while preserving the unique SEI features that permit discrimination of a given radio from the set of known radios.

5.3 Interpolation Techniques

Interpolation techniques can be used to estimate function values for undefined points within intervals bounded by points with defined

Table 1 Generator, G, NN configurations for signal collection sampling frequencies of F_L=[2.5, 5, 10] MHz.

Dimensions				
Layer	**F_L = 10**	**F_L = 5**	**F_L = 2.5**	**Activation**
Input	4 X 160 X 2	4 X 80 X 2	4 X 40 X 2	none
Convolution 2D Transpose	3 X 4 X 64	3 X 3 X 64	3 X 3 X 64	relu
UpSampling2D	1 X 4	1 X 8	1 X 4	none
Convolution 2D	3 X 3 X 16	3 X 6 X 16	3 X 6 X 16	relu
UpSampling2D	none	none	1 X 2	none
Max Pooling 2D	1 X 2	1 X 2	none	none
Convolution 2D	1 X 4 X 8	1 X 4 X 8	1 X 4 X 8	relu
Dropout 2D	0.5	0.5	0.5	none
Convolution 2D	1 X 3 X 2	1 X 3 X 2	1 X 4 X 2	linear

values (a.k.a., measurements). Interpolation can be employed to upsample the collected IoT signals since they are composed of a sequence of points in time paired with sample values (a.k.a., measurements). In this work, the sampling frequency–of the IoT devices' signals–is performed using piece-wise Linear Approximation Interpolation (LAI) and Cubic-Spline Interpolation (CSI). The higher sampling frequency is expressed as,

$$F_H = V \times F_L, \tag{2}$$

where V is an integer associated with the increase applied to the lower sampling frequency F_L.

In piece-wise linear approximation, if the signal to be interpolated z is defined at some points $a = \tau_1 < \tau_2 < \cdots < \tau_n = b$, then the interpolating function f can be represented by pieces of linear functions bounded by the intervals $[\tau_k, \tau_{k+1}]$ [9]. On each interval $[\tau_k, \tau_{k+1}]$, the function f and signal z are exactly equal at the interval points, while f approximates z with a straight line connecting τ_k and τ_{k+1} within the interval [9].

In CSI, if the function values $z(\tau_1), z(\tau_2), \cdots, z(\tau_n)$ are defined for the interval points $\tau_1 < \tau_2 < \cdots < \tau_n$, the points between interval points can be estimated using a piece-wise cubic interpolating function f, where each piece P_i is at most a third-degree polynomial [9]. The pieces P_i of the cubic-spline interpolating function f satisfies the following conditions:

$$P_i(\tau_i) = z(\tau_i), \tag{3}$$

$$P_i(\tau_{i+1}) = z(\tau_{i+1}), \tag{4}$$

$$P_i'(\tau_i) = s_i, \tag{5}$$

$$P_i'(\tau_{i+1}) = s_{i+1}, \tag{6}$$

where $i = 1, \cdots, n-1$, and $s_1, \cdots, s_n$ are free parameters [9]. The conditions in equation (6) state that the cubic-spline interpolating function f must be continuous and have a continuous first derivative over the entire interval $[a, b]$.

6 Methodology

6.1 Data Preprocessing

Fig. 1 shows the overall process developed for data preprocessing and cGAN training. The cGAN is formed using a CAE and CNN for the G and D networks, respectively. A total of 2,000 preambles, X, are collected from each of the four Wi-Fi radios. Due to the F_H=20 MHz sampling rate, each preamble consists of 320 IQ samples and has like-filtered Additive White Gaussian Noise (AWGN) added to it to achieve a Signal-to-Noise Ratio (SNR) of 9 dB to 30 dB in increments of 3 dB between consecutive values. This process is repeated ten times per preamble to augment the training set and facilitate Monte Carlo analysis. For a given SNR and each radio, 1,600 preambles from each noise realization are randomly selected to form the training data set, X_H. The training data is copied and downsampled to one of three lower sampling rates of F_L=[2.5, 5, 10] MHz to form the G inputs, X_L. The preambles not chosen for the training data set are used to generate the results. All of the preambles are formatted in accordance with the procedure described in [49], which forms a two-dimensional (2D) tensor constructed from a preamble's IQ samples and their natural logarithm. Thus, the cGAN is learning the deep representation from the preambles' raw IQ samples, magnitude, and phase representations. Each 2D tensor is scaled column-wise using min-max normalization.

6.2 Deep Networks Construction, Training, and Usage

In cGAN, the mapping functions $G(z)$ and $D(x)$ are conditioned by the class label y; thus, the minimax optimization equation (1) is rewritten as,

$$\min_G \max_D V(D,G) = E_{x\sim P_{\rm d}(x)}\{\log[D(x|y)]\} + E_{z\sim P_z(z)}\{\log[1 - D(G(z|y))]\}, \tag{7}$$

where $P_{\rm d}(x)$ is the training data distribution learned by the G. The class label y is added to the G and D inputs using a hidden representation. This enables the cGAN to estimate a one-to-many generative function instead of the traditional one-to-one mapping. The hidden representation is generated using an embedding layer to generate a length 50 label vector for each of the four Wi-Fi radios. The resulting vectors are expanded using a Fully Connected (FC) layer with a linear activation function, so that they can be reshaped and added—as an extra channel—to each preamble's 2D representation. For X_H, the output of the FC layer is of size 1,280, reshaped into a 4×320 matrix, and appended to each preamble's 2D tensor along the third dimension to form a 4×320×2 representation. The only difference between X_L and X_H is the number of columns comprising the final three-dimensional tensor. The number of columns depends on the lower sampling rate being investigated. For lower sampling rates of [2.5, 5, and 10] MHz the number of columns is 40, 80, and 160, respectively. The NN architectures associated with the G and D are provided in Table 1 and Table 2, respectively. The F_L=2.5 MHz case's G is constructed using an additional upsampling layer in lieu of the max pooling layer—used in the other cases—to ensure its output dimension is consistent with the F_H=20 MHz case (i.e., 4×320×2). The cGAN is trained using backpropagation with a 256 tensor mini-batch, 1,000 epochs, and an alternating scheme in which the D is trained for a given G in k steps. The latter results in the best D for that particular G. The work in [8] treats k as a hyperparameter with a choice of one being the least computationally complex; thus, we set the value of k to one. The D's training utilizes forward- and backpropagation with the goal of maximizing $V(D,G)$ to achieve the highest probability for making a correct decision. After k steps, the G is trained using stochastic gradient descend to minimize the function $V(D,G)$, with the goal of working against the ability of the D to make a correct decision. This training process continues until $D(x)$=0.5 (i.e., the discriminator is guessing as to the origin of the input) everywhere or the total number of training iterations equals the empirically chosen value of 1,000.

Once cGAN training concludes for the selected sampling rate F_L, the resulting G—that provides the mapping function $G(z, \theta_g)$—is

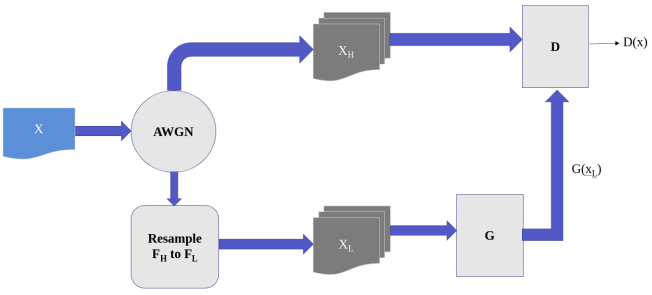


Fig. 1: Flowchart illustrating the process used to preprocess the collected signal data and train the cGAN to facilitate upsampling while preserving radio-specific SEI features.

Table 2 Configurations used to construct the discriminator, D, NN and CNN used for SEI.

Dimensions			
Layer	**Discriminator**	**RF Fingerprint Classifier**	**Activation**
Input	4 X 320 X 2	4 X 320 X 2	none
Convolution 2D	3 X 3 X 64	3 X 3 X 64	relu
Max Pooling 2D	2 X 2	2 X 2	none
Convolution 2D	3 X 3 X 32	3 X 3 X 32	relu
Max Pooling 2D	1 X 2	1 X 2	none
Convolution 2D	3 X 3 X 8	3 X 3 X 8	relu
Flatten	none	none	none
Dropout	0.5	0.5	none
Dense	128	128	relu
Dense	1	4	sigmoid / softmax

disconnected from the D and used to upsample preambles collected at rate F_L to the desired sampling rate of F_H. The second channel—introduced for cGAN training—is removed from the resulting upsampled preambles prior to SEI using a trained CNN. This process is illustrated in Fig. 2. The CNN is trained using (i) the 2D tensors that form the training set X_H, (ii) backpropagation, (iii) stochastic gradient descent optimization to minimize the categorical cross-entropy loss function, (iv) l_2 regularization to reduce overfitting, and (v) adjustment of the network's weights via Adam optimization [56]. The Adam optimizer's initial learn rate is set to 1×10^{-3}. The final layer of the CNN is a softmax activation that assigns the i^{th} preamble a label of Q_i according to

$$Q_i = \max_j(y_{ij}), \tag{8}$$

where j=[1, 2, 3, 4] and i=1, 2, . . . , 1,600 for a given radio, SNR, and noise realization.

Best possible SEI performance is achieved by training the CNN using preambles that are at an SNR equal to or lower than those comprising the test set. A grid search determined that training at SNR values of [9, 9, 9, 12, 15, 15, 15, 18] dB resulted in superior SEI performance when classifying test preambles at SNR values of [9, 12, 15, 18, 21, 24, 27, 30] dB, respectively.

7 Results

For lower sampling frequencies F_L=[2.5, 5, 10] MHz, Fig. 3 shows the average percent correct classification performance for the upsampled IEEE 802.11a Wi-Fi preamble tensors generated using the corresponding trained G and CNN at SNR values of 9 dB to 30 dB in 3 dB steps. These results are presented using solid lines and closed markers. In addition to these results, average percent correct classification performance is shown for the case when a CNN is trained and tested using tensors constructed directly from preambles sampled at the lower frequency F_L. These results are designated as the "CNN only" cases, included for comparative assessment, and presented using dashed lines and open markers. All of the presented results are generated using Tensorflow 2.0 running on NVIDIA Tesla K40m Graphics Processing Units (GPUs).

The presented cGAN-driven upsampling approach results in superior SEI performance over the "CNN only" approach for SNR values equal to or greater than 9 dB when F_L is equal to 2.5 MHz or 5 MHz and 12 dB for F_L equal to 10 MHz. When considering an average percent correct performance threshold of 90%, the "CNN only" case

Fig. 2: Flowchart illustrating the process used to preprocess the collected signal data and train the cGAN to facilitate upsampling while preserving radio-specific SEI features.

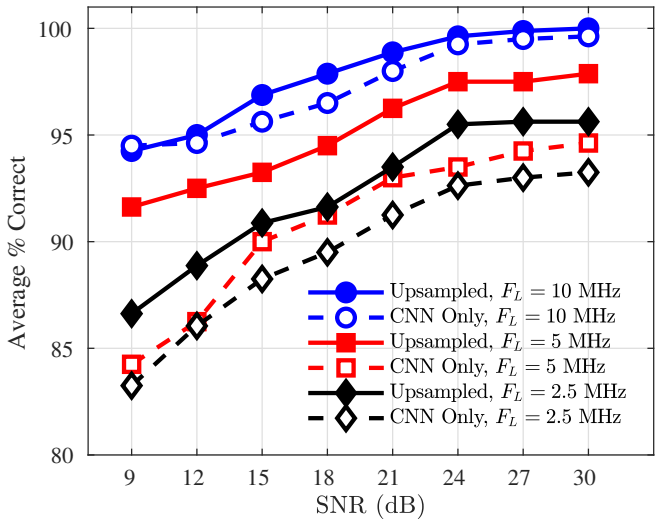


Fig. 3: Average percent correct classification performance results generated from IEEE 802.lla Wi-Fi preambles that are transmitted by four radios, collected at sampling rates of F_L equal to 2.5 MHz (◊), 5 MHz (□), or 10 MHz (◯), and upsampled to a frequency of 20 MHz using a cGAN prior to CNN classification (designated using solid lines and filled markers). The dashed lines and unfilled markers designate average percent correct classification results generated using the "CNN only" case in which training and testing is conducted using preambles collected at the sample frequencies F_L (i.e., cGAN-based upsampling is not conducted).

achieves or exceeds this threshold for all SNR values when using preambles sampled at a frequency F_L=10 MHz. However when using preambles sampled at rates of F_L=5 MHz or F_L=2.5 MHz the "CNN only" case meets or exceeds the threshold at SNR values greater than or equal to 15 dB and 21 dB, respectively. In contrast, when using preambles that are upsampled—using the corresponding trained G network—SEI performance meets or exceeds the 90% threshold for the lower sampling frequencies, F_L, of 5 MHz and 10 MHz at SNR values of 9 dB and higher. When the trained G is used to upsample preambles collected using a sampling frequency of F_L=2.5 MHz, then the 90% threshold is met or exceeded at SNR values of 15 dB and higher. Interestingly, the greatest SEI

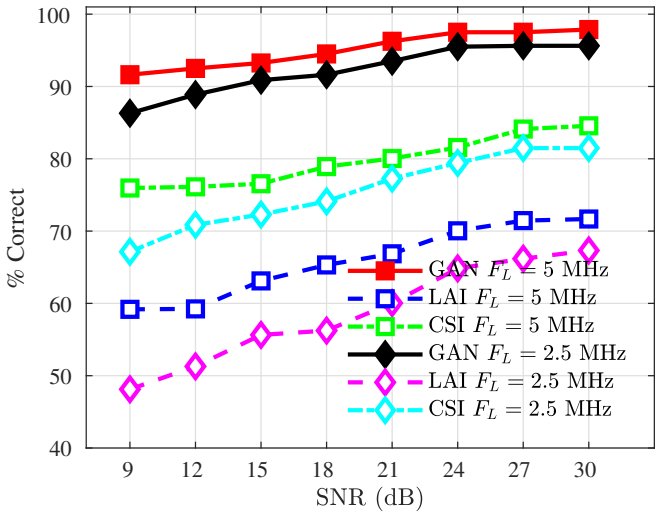
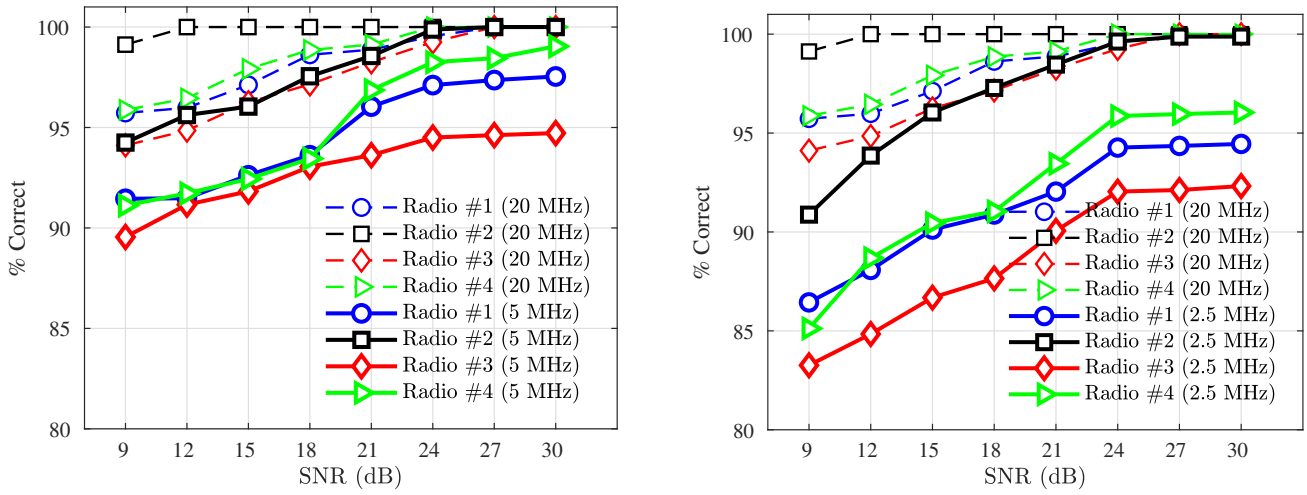


Fig. 4: Average percent correct classification performance results generated from IEEE 802.lla Wi-Fi preambles that are transmitted by four radios, collected at sampling rates of F_L equal to 2.5 MHz (◊) and 5 MHz (□) and upsampled to a frequency of 20 MHz using a cGAN (solid lines and closed markers). Comparative assessment is facilitated by including the CNN-based SEI results corresponding with the LAI (dashed lines and unfilled markers) and CSI (dashed-dot lines and unfilled markers) upsampling approaches.



(a) Collected sampling frequency of F_L=5 MHz.

(b) Collected sampling frequency of F_L=2.5 MHz.

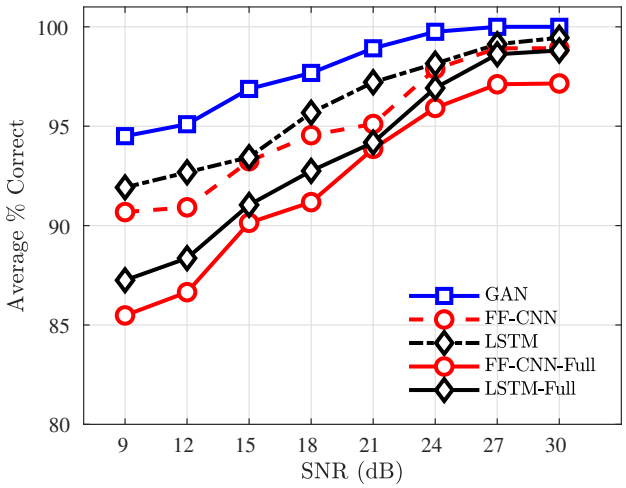
Fig. 5: Percent correct classification performance results (solid lines) generated from IEEE 802.lla Wi-Fi preambles that are transmitted by four radios, collected at sampling frequencies of F_L equal to 2.5 MHz or 5 MHz, and upsampled to a frequency of F_H=20 MHz using a cGAN prior to CNN classification. The percent correct classification results associated with the CNN classification of these same preambles–but collected using a sampling frequency of 20 MHz–are included for completeness and illustrated using dashed lines.

performance improvement occurs when the radios' preambles are collected uing a sampling frequency of F_L=5 MHz and upsampled to F_H=20 MHz using the G from the corresponding trained cGAN. For this case, the largest improvement in SEI performance is roughly 7% (i.e., 84% to 91% when comparing "CNN only" to our approach) at an SNR of 9 dB. While the smallest improvement is 3% at an SNR value equal to 27 dB. This is surprising as it seems reasonable to expect that the largest percentage improvement would occur when the F_L=10 MHz sampled preambles are upsampled to 20 MHz prior to CNN classification. This expectation is based upon the fact that higher sampling frequencies capture more of the radio-specific features exploited by the SEI process, so there are fewer of these features that need to be learned and "filled in" by the G for the 10 MHz upsampling case versus the other two. The SEI performance improvement observed for preambles that are upsampled to F_H=20 MHz from F_L=5 MHz is *not* achieved when using signals sampled at a frequency of F_L=2.5 MHz during collection. This indicates that there is a lower limit to the presented approach.

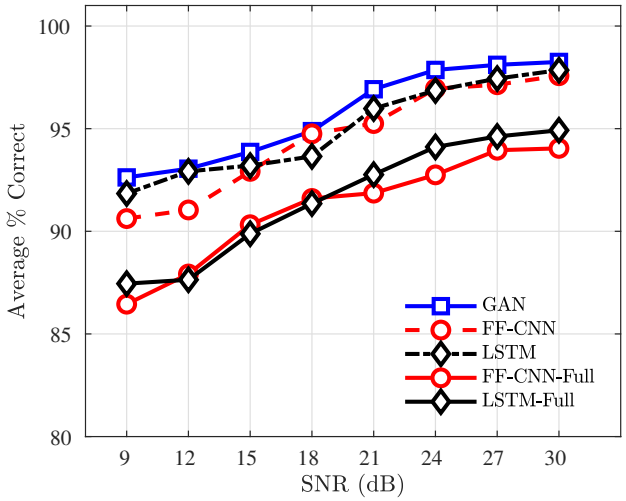
The presented cGAN-driven upsampling approach (solid lines and filled markers) is compared with the LAI (dashed lines and unfilled markers) and CSI (dash-dot lines and unfilled markers) upsampling approaches–as described in Sect. 5.3–using average percent correct classification performance for lower sampling frequencies of 2.5 MHz (◊) and 5 MHz (□) for SNR values of 9 dB to 30 dB in steps of 3 dB between consecutive values, Fig. 4. Results associated with the F_L=10 MHz lower sampling frequency case are omitted for the following reasons: (i) only a marginal improvement (i.e., maximum of 3%) is observed between the cGAN upsampled and "CNN only" results shown in Fig. 3, and (ii) their omission improves visual clarity of the 2.5 MHz and 5 MHz results without diminishing the value of the comparative assessment. For the average percent correct classification results presented in Fig. 4, the cGAN-driven upsampling approach results in superior performance over the results corresponding to the LAI and CSI approaches for all investigated SNR values. As a matter of fact the results associated with the LAI and CSI approaches never achieve the 90% average percent correct performance threshold for any of the investigated SNR values. These results show that the cGAN's ability to learn the radios' SEI distributions is required to upsample signals prior to CNN-based discrimination. The CSI approach provides superior performance over LAI. When using F_L=2.5 MHz sampled preambles, the CSI approach's average percent correct classification performance outperforms the results associated with the LAI approach using preambles sampled using a frequency of F_L=5 MHz for all SNR values. The largest margin–between the CSI's F_L=2.5 MHz results and the LAI's F_L=5 MHz results–is 11% (i.e., 70% versus 59%) at an SNR value of 12 dB and the smallest is 8% at an SNR of 9 dB. The CSI performance improvement–over the LAI approach–is attributed to its use of a *cubic* interpolating function instead of the linear function used by LAI. This allows the CSI approach to better model the behaviors present in signals collected at a sampling frequency of F_H=20 MHz. Fig. 5 presents percent correct classification performance results for each radio whose collected IEEE 802.11a Wi-Fi preambles are upsampled to a sampling frequency of F_H=20 Mhz using a cGAN. For completeness, Fig. 5 also includes the individual radio percent correct classification performance for the case when CNN-based SEI is performed using preambles collected at a sampling frequency of 20 MHz (i.e., no upsampling) and are indicated using thinner lines than those associated with the cGAN upsampled results. The results shown in Fig. 5(a) and Fig. 5(b) correspond to IEEE 802.11a Wi-Fi preambles that are collected at sampling frequencies of F_L=5 MHz and F_L=2.5 MHz, respectively. The F_L=5 MHz SEI results, Fig. 5(a), outperform that of the F_L=2.5 MHz case, Fig. 5(b), and experience less spread across the four radios. These results are expected as the cGAN is filling in 87.5% of the IQ samples–along with their associated SEI features–for the F_L=2.5 MHz case versus 75% when the original sampling frequency is F_L=5 MHz. When considering this information, the presented results are quite impressive considering that individual radio discrimination performance for the F_L=2.5 MHz and F_L=5 MHz cases remain above 83% and 89% for all radios and SNR values. The relative performance of the four radios remains consistent with Radio #2 being the most easily distinguished and Radio #3 being the hardest to distinguish. For both upsampling cases–going from 5 MHZ to 20 MHz and 2.5 MHZ to 20 MHz–the percent correct classification performance is poorer than the case of SEI performed using preambles collected at a sampling frequency of 20 MHz. This suggests that the cGAN is not completely capturing the distributions of the SEI features present in the four radios' waveforms. Some possible solutions could be to use more waveforms in the training of the cGAN, use of an alternate waveform representation (e.g., Fourier or Gabor transform), or a combination of the two.

7.1 Comparison with State-of-Art SEI

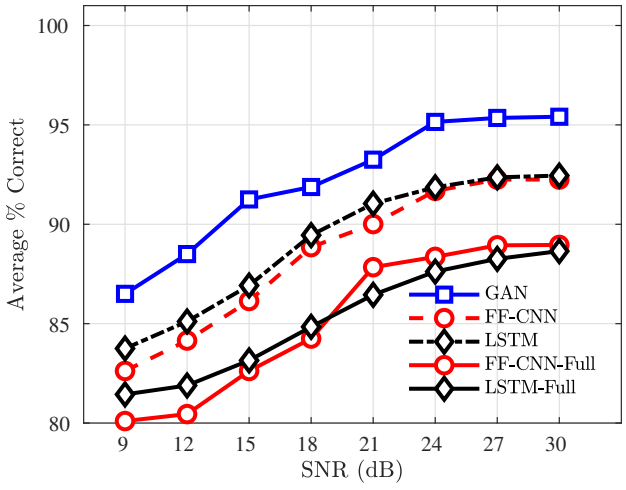
The performance of the SEI approach presented in this paper is compared against the one in [57]. The work in [57] presents a DL-based SEI protocol to identify wireless devices with the ability to process variable-length NN inputs and use online augmentation to improve the SEI performance at low SNRs. The proposed SEI protocol is



(a) Collected sampling frequency of F_L=10 MHz.



(b) Collected sampling frequency of F_L=5 MHz.



(c) Collected sampling frequency of F_L=2.5 MHz.

Fig. 6: Average percent correct classification performance results generated from IEEE 802.lla Wi-Fi preambles that are transmitted by four radios, collected at sampling rates of F_L equal to 2.5 MHz, 5MHz, or 10 MHz, and (i) upsampled to a frequency of 20 MHz using a cGAN prior to CNN classification (□), (ii) Classified using variable-length Flatten-Free CNN (FF-CNN) (◯) and LSTM (◊) in [57]. The Average percent correct classification performance results designated with "Full" are the ones that include $F = 20$ MHz preambles in the training set.

tested using LoRa/LoRaWAN technology devices which is a popular low power wide area network (LPWAN) protocol. LoRaWAN adopts an Adaptive Data Rate (ADR) mechanism that enables wireless devices to adjust transmission configuration in real time [57]. The ADR capability enables the LoRa devices to adjust the length of the preamble and payload to tackle continuously changing wireless channel conditions.

LoRa modulates data at the physical layer using Chirp Spread Spectrum (CSS). Three pieces of information are needed to generate a LoRa preamble using CSS modulation: (i) signal amplitude, (ii) bandwidth, and (iii) Spreading Factor (SF). The SF parameter controls the length of the signal and results in increasing bandwidths from 125 kHz to 500 kHz for SF values of 7 to 12 as detailed in [57]. Our selection of the approach in [57] is due to the fact that both our approach and the one in [57] are designed to process variable length inputs (a.k.a., preambles). In [57], the length of the preamble is controlled by the CSS modulation's SF parameter, while in our approach the preamble length is controlled by the sampling frequency.

The work in [57] represents LoRa preambles using a channel independent spectrogram to mitigate the wireless channel effects. Each preamble's spectrogram S^i is a matrix of $N \times (M-1)$ elements with each element designated by $S^i_{k,m}$ in which $k = 1, 2, \ldots, N$ and $m = 1, \ldots, M-1$. N represents the length of the spectrogram window, which is set to sixty-four in [57]. $(M-1)$ is the width of S^i where M is related to N by,

$$M = \frac{8 \times \left(\frac{2^{SF}}{B}\right) \times F_L - N}{R} + 1, \tag{9}$$

where F_L is the sampling frequency (a.k.a., the reciprocal of the sampling interval), and R is the hop size which is always set to thirty-two by the authors of [57]. In [57], the spectrogram S^i has a fixed length of $N = 64$, $F_s = 250$ KHz, and a variable width of $M-1$ that depends on the selected spreading factor SF. In order to compare our approach with that in [57], we generate spectrograms in which the width, $M-1$, is determined by the sampling frequency, F_L, instead of the spreading factor SF. So, in our case the spreading factor is fixed and the sampling frequency changes, which is the opposite of [57]. For the results presented in this section, spectrograms S^i are generated for all IEEE 802.11a preambles–described in Sect. 5.1–for $F_s = [2.5, 5, 10]$ MHz and $SF = 7$ for all sampling frequencies.

The authors of [57] introduce online augmentation to increase the SEI system's robustness to noise. In online augmentation, AWGN is added to each minibatch–selected from the training set–directly before being passed to the NN. In online augmentation, the NN model is trained using (steps × minibatch size × noise realizations) noisy signals. The results in this section are generated using online augmentation with 13 steps, a minibatch size of 128, and ten noise realizations. The authors of [57] use four NN architectures to process the variable-width, channel independent spectrograms generated from the variable length, LoRa preambles. Based on the results presented in [57], all NN architectures exhibit similar performance. Therefore, only two NNs are selected for the comparative assessment, which are a: (i) Flatten-Free CNN (FF-CNN) where the flatten layer of a ResNet network is replaced with a global average pooling 2D layer that always provides a fixed-length vector to the dense layers regardless of the spectrogram's width at the input of the FF-CNN, and (ii) LSTM, where a global average pooling 1D layer is used in a similar way as in the FF-CNN to produce a fixed-length vector for the following dense layers. Detailed descriptions of the FF-CNN and LSTM architectures are presented in [57].

Comparative assessment is enabled by regenerating the results shown in Fig. 3, but with online augmentation integrated into the GAN training process. Average percent correct classification performance is shown for the: regenerated GAN-based upsampling as well as the FF-CNN and LSTM results using IEEE 802.11a Wi-Fi spectrograms for all three sampling frequencies when the classification model is trained with and without the original 20 MHz sampled

preambles included. It can be seen that GAN-based SEI average percent correct classification performance exceeds that of the FF-CNN and LSTM spectrogram-based approaches for all SNR values for sampling frequencies of 10 MHz, Fig. 6(a), and 2.5 MHz, Fig. 6(c). For $F_L = 5$ MHz, Fig. 6(b), all three SEI approaches exhibit the similar performance for SNR values of 15 dB and above; however, GAN- and LSTM-based SEI average percent correct classification performance surpasses that of the FF-CNN for SNR values below 15 dB.

The SEI approach in [57] allows identification of emitters using signals of variable-length by learning length independent features. Our observation is based on the fact that all of the signals–regardless of length–are mapped to the same length dense layer for both NNs (a.k.a., FF-CNN and LSTM); thus, the sampling frequency of the signal to be classified does not change. However, this is problematic when length-variability is due to changing sample frequencies, because the signal points measured during the continuous to discrete conversion change. Hence, the SEI features present in the discrete signals change as well, which leads to feature inconsistencies across different length signals that subsequently inhibits learning of emitter discriminating features. Our cGAN approach does not suffer from these limitations, because it is trained using the 20 MHz sampled signals such that its G uses the learned, SEI feature distributions to map the lower sampling rate signals' SEI features into the 20 MHz sampled signals' space. In other words, the G is constructing 20 MHz sampled versions of the F_L sampled signals (a.k.a., upsampling). Thus, in our approach the trained CNN performs SEI using only 20 MHz sampled signals and not lower sampling frequency signals that have been "flattened" to meet the same, fixed length. cGAN approach performance does degrade as F_L decreases. However, this is attributed to the limited size of the 20 MHz sampled data set (i.e., only 2,000 preambles per emitter), because the limited data set size impedes learning of emitter accurate SEI feature distributions.

8 Conclusion

This work presents a novel DL-based approach that uses a trained cGAN's generative network to upsample the collected preambles of four IEEE 802.11a Wi-Fi radios from sampling frequencies of [2.5, 5, and 10] MHz to 20 MHz in an effort to improve SEI performance while reducing IoT device hardware requirements that collect the signals of the radio(s) being identified. The cGAN-based approach is compared with two traditional interpolation approaches as well as an SEI approach in which the collected signals are *not* upsampled prior to CNN-based classification. The results show that the cGAN-based approach is superior for all three lower sampling frequencies. The largest improvement in average percent correct classification performance occurs when 5 MHz sampled signals are upsampled to 20 MHz. Our investigation shows that a trained DL architecture can be used to upsample signals while learning a given radio's SEI features so that the SEI process' discrimination performance increases. Future research will focus on improving the SEI performance of the upsampled signals as well as increasing the number of radios used.

9 References

1 Statista, "Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions)," https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/, 2019.

2 Ray, I., D. Kar, J. Peterson, and S. Goeringer, "Device Identity and Trust in IoT-sphere Forsaking Cryptography," in *International Conference on Collaboration and Internet Computing (CIC)*, 2019.

3 Wright, J., "KillerBee: Practical ZigBee Exploitation Framework or 'Wireless Hacking and the Kinetic World'," pp. 1–39. [Online]. Available: https://www.inguardians.com/works/

4 Kandah, F., J. Cancelleri, D. Reising, A. Altarawneh, and A. Skjellum, "A Hardware-Software Co-design Approach to Identity, Trust, and Resilience for IoT/CPS at Scale," in *International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, July 2019, pp. 1125–1134.

5 Reising, D., J. Cancelleri, T. Loveless, F. Kandah, and A. Skjellum, "Radio identity verification-based iot security using rf-dna fingerprints and svm," *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 8356–8371, 2021.

6 Restuccia, F., S. D'Oro, Salvatore, A. Al-Shawabka, M. Belgiovine, L. Angioloni, S. Ioannidis, Stratis and K. Chowdhury, and T. Melodia, "DeepRadioID: Real-Time Channel-Resilient Optimization of Deep Learning-based Radio Fingerprinting Algorithms," in *ACM Int'l Symposium on Mobile Ad Hoc Networking & Computing*, ser. Mobihoc, 2019.

7 Mirza, M. and S. Osindero, "Conditional generative adversarial nets," *arXiv*, 2014. [Online]. Available: https://www.inguardians.com/works/

8 Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

9 de Boor, C., *A Practical Guide to Spline*. Springer-Verlag, New York, 01 1978, vol. 27.

10 Baldini, G. and R. Giuliani, "An Assessment of the Impact of Wireless Interferences on IoT Emitter Identification using Time Frequency Representations and CNN," in *Global IoT Summit (GIoTS)*, June 2019, pp. 1–6.

11 Baldini, G., C. Gentile, R. Giuliani and G. Steri, "Comparison of Techniques for Radiometric Identification based on Deep Convolutional Neural Networks," *IET Electronics Letters*, vol. 55, no. 2, pp. 90–92, 2019.

12 O'Shea, T., T. Roy, and T. Clancy, "Over-the-Air Deep Learning Based Radio Signal Classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, Feb 2018.

13 Wong, L., W. Headley, S. Andrews, R. Gerdes and A. Michaels, "Clustering Learned CNN Features from Raw I/Q Data for Emitter Identification," in *IEEE Military Communications Conference (MILCOM)*, Oct 2018, pp. 26–33.

14 Mendis, G., J. Wei, and A. Madanayake, "Deep Learning Based Radio-Signal Identification With Hardware Design," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, no. 5, pp. 2516–2531, Oct 2019.

15 Riyaz, S., K. Sankhe, S. Ioannidis, and K. Chowdhury, "Deep Learning Convolutional Neural Networks for Radio Identification," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 146–152, Sep. 2018.

16 Merchant, K., S. Revay, G. Stantchev and B. Nousain, "Deep Learning for RF Device Fingerprinting in Cognitive Communication Networks," *IEEE J. of Selected Topics in Sig Proc*, vol. 12, no. 1, pp. 160–167, Feb 2018.

17 Jafari, H., O. Omotere, D. Adesina, H. Wu, and L. Qian, "IoT Devices Fingerprinting Using Deep Learning," in *IEEE Military Communications Conference (MILCOM)*, Oct 2018, pp. 1–9.

18 Guyue, L., Y. Jiabao, Y. Xing, and A. Hu, "Location-Invariant Physical Layer Identification Approach for WiFi Devices," *IEEE Access*, vol. 7, pp. 06 974–106 986, Aug 2019.

19 Youssef, K., L. Bouchard, K. Haigh, J. Silovsky, B. Thapa, and C. Valk, "Machine Learning Approach to RF Transmitter Identification," *IEEE J. of Radio Frequency Identification*, vol. 2, no. 4, pp. 197–205, Dec 2018.

20 Pan, Y., S. Yang, H. Peng, T. Li and W. Wang, "Specific Emitter Identification Based on Deep Residual Networks," *IEEE Access*, 2019.

21 Yu, J., A. Hu, F. Zhou, Y. Xing, Y. Yu, G. Li, and L. Peng, "Radio frequency fingerprint identification based on denoising autoencoders," in *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2019, pp. 1–6.

22 Jian, T., B. Rendon, E. Ojuba, N. Soltani, Z. Wang, K. Sankhe, A. Gritsenko, J. Dy, K. Chowdhury, and S. Ioannidis, "Deep Learning for RF Fingerprinting: A Massive Experimental Study," *IEEE Internet of Things Magazine*, vol. 3, no. 1, pp. 50–57, 2020.

23 Robinson, J., S. Kuzdeba, J. Stankowicz, and J. Carmack, "Dilated Causal Convolutional Model For RF Fingerprinting," in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, 2020, pp. 0157–0162.

24 L. Ding, S. Wang, F. Wang, and W. Zhang, "Specific emitter identification via convolutional neural networks," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2591–2594, 2018.

25 P. Chen, Y. Guo, G. Li, L. Wang, and J. Wan, "Discriminative adversarial networks for specific emitter identification," *Electronics Letters*, vol. 56, 02 2020.

26 L. Peng, J. Zhang, M. Liu, and A. Hu, "Deep learning based rf fingerprint identification using differential constellation trace figure," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 1091–1095, 2020.

27 N. Yang, B. Zhang, G. Ding, Y. Wei, G. Wei, J. Wang, and D. Guo, "Specific emitter identification with limited samples: A model-agnostic meta-learning approach," *IEEE Communications Letters*, pp. 1–1, 2021.

28 C. Morin, L. S. Cardoso, J. Hoydis, J.-M. Gorce, and T. Vial, "Transmitter classification with supervised deep learning," in *Cognitive Radio-Oriented Wireless Networks*, A. Kliks, P. Kryszkiewicz, F. Bader, D. Triantafyllopoulou, C. E. Caicedo, A. Sezgin, N. Dimitriou, and M. Sybis, Eds. Cham: Springer International Publishing, 2019, pp. 73–86.

29 S. Behura, S. Kedia, S. M. Hiremath, and S. K. Patra, "Wist id—deep learning-based large scale wireless standard technology identification," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 4, pp. 1365–1377, 2020.

30 J. M. McGinthy, L. J. Wong, and A. J. Michaels, "Groundwork for neural network-based specific emitter identification authentication for iot," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6429–6440, 2019.

31 P. Tang, Y. Xu, G. Wei, Y. Yang, and C. Yue, "Specific emitter identification for iot devices based on deep residual shrinkage networks," *China Communications*, vol. 18, no. 12, pp. 81–93, 2021.

32 Y. Liu, J. Wang, J. Li, H. Song, T. Yang, S. Niu, and Z. Ming, "Zero-bias deep learning for accurate identification of internet-of-things (iot) devices," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2627–2634, 2021.

33 X. Zha, H. Chen, T. Li, Z. Qiu, and Y. Feng, "Specific emitter identification based on complex fourier neural network," *IEEE Communications Letters*, pp. 1–1, 2021.

34 Gong, J., X. Xu, Y. Qin, and W. Dong, "A generative adversarial network based framework for specific emitter characterization and identification," in *2019 11th*

International Conference on Wireless Communications and Signal Processing (WCSP), 2019, pp. 1–6.
35 H. Ji, T. Wan, W. Xiong, and J. Liao, "A method for specific emitter identification based on surrounding-line bispectrum and convolutional neural network," in *2020 IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE)*, 2020, pp. 328–332.
36 Gong, J., X. Xu, and Y. Lei, "Unsupervised specific emitter identification method using radio-frequency fingerprint embedded infogan," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2898–2913, 2020.
37 R. Li, J. Hu, S. Li, and W. Ai, "Specific emitter identification based on multi-domain features learning," in *2021 IEEE International Conference on Artificial Intelligence and Industrial Design (AIID)*, 2021, pp. 178–183.
38 Wang, Y., G. Gui, H. Gacanin, T. Ohtsuki, O. Dobre, and V. Poor, "An efficient specific emitter identification method based on complex-valued neural networks and network compression," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2305–2317, 2021.
39 L.-Z. Qu, H. Liu, K.-J. Huang, and J.-A. Yang, "Specific emitter identification based on multi-domain feature fusion and integrated learning," *Symmetry*, vol. 13, no. 8, 2021. [Online]. Available: https://www.mdpi.com/2073-8994/13/8/1481
40 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 *2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)*, 2019, pp. 98–104.
41 J. Wang, B. Zhang, J. Zhang, N. Yang, G. Wei, and D. Guo, "Specific emitter identification based on deep adversarial domain adaptation," in *2021 4th International Conference on Information Communication and Signal Processing (ICICSP)*, 2021, pp. 104–109.
42 C. Cun, T. Li, and J. Zhu, "Specific emitter identification based on eye diagram," in *2021 IEEE 21st International Conference on Communication Technology (ICCT)*, 2021, pp. 1261–1265.
43 Y. Peng and Y. Zhou, "Specific emitter identification via squeeze-and-excitation neural network in frequency domain," in *2021 40th Chinese Control Conference (CCC)*, 2021, pp. 8310–8314.
44 W. Shen and W. Wang, "Node identification in wireless network based on convolutional neural network," in *2018 14th International Conference on Computational Intelligence and Security (CIS)*, 2018, pp. 238–241.
45 Tyler, J., M. Fadul, D. Reising, and F. Kandah, "An analysis of signal energy impacts and threats to deep learning based sei," in *IEEE Int'l Conf on Communications (ICC)–Accepted*, 2022.
46 S. Krug and M. O'nils, "Iot communication introduced limitations for high sampling rate applications," 2018.
47 Fadul, M., D. Reising, and M. Sartipi, "Identification of ofdm-based radios under rayleigh fading using rf-dna and deep learning," *IEEE Access*, vol. 9, pp. 17 100–17 113, 2021.
48 Porkodi, R. and V. Bhuvaneswari, "The Internet of Things (IoT) Applications and Communication Enabling Technology Standards: An Overview," in *International Conference on Intelligent Computing Applications*, March 2014.
49 Tyler, J., M. Fadul, D. Reising, and E. Kaplanoglu, "Simplified Denoising for Robust Specific Emitter Identification of Preamble-based Waveforms," in *IEEE GLOBECOM*, Dec 2021.
50 Lajos, H., A. Yosef, W. Li, and J. Ming, *MIMO-OFDM for LTE, Wi-Fi, and WiMAX*. John Wiley and Sons, Ltd., 2011.
51 Goodfellow, I., Y. Bengio and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.
52 Seyfioğlu, M., A. Özbayoğlu, and S. Gürbüz, "Deep Convolutional Autoencoder for RADAR-based Classification of Similar Aided and Unaided Human Activities," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 4, pp. 1709–1723, Aug 2018.
53 Patterson, J. and A. Gibson, *Deep Learning A Practitioners Approach*. O'Reilly Media, 2017.
54 Masci, J., U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction," in *Artificial Neural Networks and Machine Learning – ICANN 2011*, T. Honkela, W. Duch, M. Girolami, and S. Kaski, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 52–59.
55 Creswell, A., T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.
56 Kignma, D. and J. Ba, "Adam: A method for stochastic optimization," *arXiv*, 2015. [Online]. Available: https://arxiv.org/pdf/1412.6980.pdf
57 G. Shen, J. Zhang, A. Marshall, M. Valkama, and J. Cavallaro, "Towards length-versatile and noise-robust radio frequency fingerprint identification," *arXiv preprint arXiv:2207.03001*, 2022.