Zero-Bias Deep Learning for Accurate Identification of Internet-of-Things (IoT) Devices

AQ1 Yongxin Liu, Jian Wang, Jianqiang Li, *Member, IEEE*, Houbing Song, *Senior Member, IEEE*, Thomas Yang, *Senior Member, IEEE*, Shuteng Niu, and Zhong Ming, *Member, IEEE*

***Abstract*—The Internet of Things (IoT) provides applications and services that would otherwise not be possible. However, the open nature of IoT makes it vulnerable to cybersecurity threats. Especially, identity spoofing attacks, where an adversary passively listens to the existing radio communications and then mimic the identity of legitimate devices to conduct malicious activities. Existing solutions employ cryptographic signatures to verify the trustworthiness of received information. In prevalent IoT, secret keys for cryptography can potentially be disclosed and disable the verification mechanism. Noncryptographic device verification is needed to ensure trustworthy IoT. In this article, we propose an enhanced deep learning framework for IoT device identification using physical-layer signals. Specifically, we enable our framework to report unseen IoT devices and introduce the zero-bias layer to deep neural networks to increase robustness and interpretability. We have evaluated the effectiveness of the proposed framework using real data from automatic dependent surveillance-broadcast (ADS-B), an application of IoT in aviation. The proposed framework has the potential to be applied to the accurate identification of IoT devices in a variety of IoT applications and services.**

***Index Terms*—Big data analytics, cybersecurity, deep learning, Internet of Things (IoT), noncryptographic identification, zero-bias neural network.**

I. Introduction

THE Internet of Things (IoT) is characterized by the interconnection and interaction of smart objects (objects or devices with embedded sensors, onboard data processing capability, and a means of communication) to provide applications and services that would otherwise not be possible [1]. The convergence of sensor, actuator, information, and communication technologies in IoT produces massive amounts of data that need to be sifted through to facilitate reasonably accurate decision-making and control [2]. Big data analytics has the potential to enable the move from IoT to real-time control [3]. However, due to the open nature of IoT, IoT is subject to cybersecurity threats [4], [5]. One typical cybersecurity threat is identity spoofing attacks where an adversary passively collects information and then mimic the identity of legitimate devices to send fake information or conduct other malicious activities. Such attacks can be extremely dangerous when appear in critical infrastructures [6].

Conventional approaches to prevent identity spoofing attacks employ cryptographic algorithms to verify that a trusted source generates a message. However, the cryptographic approaches depend on the secrecy of encryption keys and encounter challenges from the open and heterogeneous ecosystems of IoT. For example, a number of commercially successful IoT systems, which do not operate with cryptographic keys, require a huge investment to become cryptographically secure [7]. Therefore, there is a need for noncryptographic solutions to verify the identity of IoT devices, thus ensuring trustworthy IoT.

Noncryptographic IoT device identification is inspired by the signal identification technology in speech and acoustic signal processing [8]. The assumption is that each signal source modulate its unique features into the propagated signals. Comparably, in noncryptographic IoT device identification, we assume that each wireless transmitter randomly picks up certain types of imperfectness (also known as, radiometric fingerprint) during their manufacture [9] and could be reflected in the demodulated signals. Existing works on noncryptographic device identification can be classified into two categories: 1) specific feature recognition and 2) deep learning. Specific feature-based approaches focus on deriving distinctive features (also known as, transmitter fingerprints) from received signals [10], [11] to recognize known devices. Deep-learning-based approaches do not require knowing devices' radiometric characteristics and show even higher accuracy [12], [13]. However, the challenge of applying deep learning approaches for IoT device identification lies in two aspects: 1) unseen device recognition and 2) model interpretability. The first challenge requires deep neural networks (DNNs) to report unseen devices rather than erroneously associating them with known ones. The second challenge requires that the behaviors of neural networks to be interpretable.

In this article, we propose an enhanced deep learning framework for accurate and interpretable identification of IoT devices with mathematically assured performance. We propose a zero-bias dense layer for DNNs to jointly verify known devices and identify unknown ones. The effectiveness of the proposed framework in handling massive signal recognition

AQ2 Manuscript received May 3, 2020; revised July 19, 2020; accepted August 19, 2020. This work was supported in part by the Embry-Riddle Aeronautical University's Faculty Innovative Research in Science and Technology Program and in part by the National Science Foundation under Grant 1956193. *(Corresponding authors: Jianqiang Li; Houbing Song.)*

Yongxin Liu, Jian Wang, Houbing Song, Thomas Yang, and Shuteng Niu are with the Department of Electrical Engineering and Computer Science, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114 USA (e-mail: h.song@ieee.org).

Jianqiang Li and Zhong Ming are with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: lijq@szu.edu.cn).

Digital Object Identifier 10.1109/JIOT.2020.3018677

and improving the performance of traditional neural networks has been demonstrated. The contributions of this article are as follows.

1) We provide a novel enhancement, the zero-bias layer, to replace the last dense layer in conventional neural networks to increase its interpretability without losing accuracy.
2) We provide a novel technique to characterize how well a neural network can distinguish from different classes.
3) We enable our framework to automatically report unknown devices rather than erroneously associating them with known ones.

Our research offers not only a solution to accurate identification of IoT devices, thus useful in promoting trustworthy IoT but also a deep learning framework for intrusion detection. In addition, the introduction of the zero-bias layer in DNNs represents an advance in deep learning, thus leveraging deep learning to enable the move from IoT to real-time control.

The remainder of this article is organized as follows. A literature review of noncryptographic device identification is presented in Section II. We formulate our problem in Section III with methodology presented in Section IV. Performance evaluation is presented in Section V with conclusions in Section VI.

II. Related Works

Noncryptographic device identification is emerging as a solution to physical-layer security of IoT. Coresponding methods can be classified into two categories: 1) specific feature based and 2) deep learning based.

A. Specific Feature-Based Approaches

The specific feature-based approaches require human efforts to discover distinctive features for device identification. The methods rely on the fact that there are various manufacturing imperfectnesses in wireless devices' RF frontends. These imperfectnesses do not degrade the communication quality but can be exploited to identify each transmitter uniquely. Those features are named physical unclonable features (PUFs) [14], [15]. There are two categories of PUFs: 1) error pattern and 2) transient patterns.

In error pattern approaches, it is assumed that the statistical properties of received symbols' noise could uniquely profile wireless devices. Azarmehr *et al.* [16] showed that phrase error of phase lock loop (PLL) in transmitters can provide promising results even with the low signal-to-noise ratio (SNR). Zhuang *et al.* [17] used the difference between received signals and theoretical templates to construct error vectors. Error vectors' statistics and time-frequency features are combined as fingerprints for transmitter identification. Peng *et al.* [18] employed differential constellation trace figure (DCTF) to capture the time-varying modulation error of ZigBee devices. They then develop their low-overhead classifier to identify 54 ZigBee devices.

In transient pattern approaches, it is assumed that a malicious entity cannot forge the transient response characteristic of wireless transmitters [19]. Transient patterns are commonly seen at the beginning and end of wireless packet transmission. In [20], nonlinear in-band distortion and spectral regrowth of the signals are utilized to distinguish the masquerade emitter. Köse *et al.* [21] employed the transient energy spectrum on transmitters' turn-on amplitude envelops to identify, and they showed that frequency-domain features outperform time-domain features.

Feature-based approaches require efforts to manually extract features or high-order statistics for different scenario. Therefore, more effortless and versatile methods are required.

B. Deep Neural Network-Based Approaches

DNNs are frequently used as a general-purpose BlackBox for pattern recognition. Naturally, they are applied to perform device-specific identification.

A typical DNN-enabled wireless device identification system employs convolutional layers to extract latent features. Convolutional layers apply filters (also known as, kernels) to obtain helpful information automatically. Such benefit reduces the hardship of manual feature discovery. Yu *et al.* [22] provided a novel method that perform the signal denoising and emitter identification simultaneously using an autoencoder and a convolution neural network (CNN). Their solution shows promising results even with low SNR. Similar work in [23] employs stacked denoising autoencoder and show similar results. DNNs perform well even on raw signals. Riyaz *et al.* [24] provided an optimized deep convolutional neural network to classify software-defined radio (SDR)-based emitters in 802.11AC channels, they show that, even by using raw signals without feature engineering, CNN surpasses the best performance of conventional statistical learning methods. In [25], neural networks were trained on raw IQ samples using the open data set[1] from CorteXlab. Their work also show similar results. Compared with the specific feature-based approach, DNNs dramatically reduce the requirement of domain knowledge and the quality of fingerprints.

In general, DNNs are becoming a promising building block in noncryptographic wireless device identification. DNNs encounter a challenge in terms of anomaly detection, which requires that deep learning-enabled identification systems not only to perform well on trained objects but also can report unknown objects that it would make a wrong decision. Furthermore, for dependable machine learning in practical scenarios, we need to understand how a neural network associates an input with a corresponding label. These two aspects are rarely covered in signal identification, thus motivating our research.

III. Problem Definition

In this research, we focus on deriving the protocol-agnostic solution to identify of IoT devices from physical-layer signals. The reason is that signal features directly correspond to hardware components and reveals the identities of IoT devices.

We define that an IoT device i transmits specific message with corresponding baseband signal $m_i(t)$. $m_i(t)$ is

[1] https://wiki.cortexlab.fr/doku.php?id=tx-id

modulated into

$$M_i(t) = C_i[m_i(t)] \tag{1}$$

where $C_i(x)$ denotes the frequency band processing chains. At receiver j, the received signal becomes

$$R_{ij}(t) = S_{ij}[M_i(t)] \tag{2}$$

where S_{ij} denotes the effect of wireless channel between i and j. This function can incorporate the effect of attenuation or additive noise. The demodulated signal is

$$\begin{aligned}\hat{m}_i(t) &= S_j^{-1}\left\{C_j^{-1}\left[R_{ij}(t)\right]\right\} \\ &= S_j^{-1}\left\{C_j^{-1}\left[S_{ij}[C_i[m_i(t)]]\right]\right\}\end{aligned} \tag{3}$$

where $C_j^{-1}(x)$ and $S_j^{-1}(x)$ are j's estimated reverse function of $C_i(x)$ and S_{ij}, respectively. The estimation can hardly be idealistic. Therefore, at the receiver side, j, the effect of such discrepancies are reflected in $\hat{m}_i(t)$ as

$$\hat{m}_j(t) = r_i(t) + \delta_j(t) \tag{4}$$

where $r_i(t)$ is directly correlated with $m_i(t)$ while the residual, $\delta_j(t)$, is utilized to recognize a wireless device. As long as $\delta_j(t)$ is uncorrelated with messages $m_i(t)$, the recognition algorithm is protocol agnostic. Apparently, this is a classification problem, to avoid the hardship of feature engineering, we use DNN and convert IoT device recognition problem into three subproblems.

1) Given message-related baseband signals from various wireless transmitters, how to extract message-independent components to develop a classifier using DNNs?
2) How to enable our classifier to properly respond to unseen signals?
3) How can we evaluate the distinguisability between different devices?

IV. Proposed Framework

In this section, we first present the feature extraction methods and then introduce the zero-bias deep learning framework for accurate and interpretable identification of IoT devices.

A. Baseband Demodulation

In this research, we use an independent SDR receivers, denoted as j', to collect baseband signals from wireless transmitters, denoted as $\hat{m}_{j'}(t)$. Given input signal x, the quadrature demodulation function is defined as

$$\begin{aligned}C_{j'}^{-1}(x) &= I(t) + \boldsymbol{i}\cdot Q(t) \\ &= \mathrm{LPF}[x\cdot\cos(\omega_c t + \phi_0) + \boldsymbol{i}\cdot x\cdot\sin(\omega_c t + \phi_0)]\end{aligned} \tag{5}$$

where $I(t)$ and $Q(t)$ are inphase and quadrature components, respectively. ω_c and ϕ_0 are the center frequency and the phase offset of the receiver (j'), respectively. $\boldsymbol{i}$ denotes the imaginary part of the complex function. With PLL, ω_c and ϕ_0 are supposed to be sufficiently close to RF characteristics of device i.

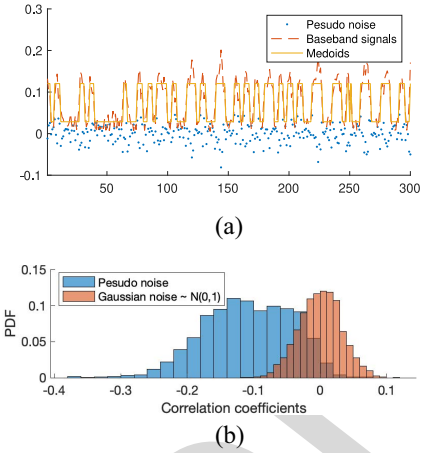


Fig. 1. Property of pseudonoise extraction. (a) Noise extraction on typical signals. (b) Correlation coefficients of pseudonoise.

LPF denotes a low-pass filter. Therefore, at j', demodulated baseband is

$$\hat{m}_{j'}(t) = C_{j'}^{-1}\left[R_{ij'}(t)\right] \tag{6}$$

$\hat{m}_{j'}(t)$ is complex valued, and its instantaneous amplitude, phase and frequency are $\|\hat{m}_{j'}(t)\| = \sqrt{I^2(t)+Q^2(t)}$, $\angle\hat{m}_{j'}(t) = \tan^{-1}(Q(t)/I(t))$ and $\hat{\Omega}_{j'}(t) = ([d\angle\hat{m}_{j'}(t)]/dt)$, respectively.

Note that discrepancies exist between $\hat{m}_j(t)$ and $\hat{m}_{j'}(t)$. Even if the wireless channel effect at receiver j and j' are different, we assume that an SDR receiver could still capture the effect of each wireless device's frequency band processing chain, $C_i(x)$, to recognize them.

B. Feature Extraction

For protocol-agnostic device recognition, we need to remove message-correlated part $r_i(t)$ from $\hat{m}_{j'}(t)$. In this way, we ensure that our device recognition mechanism is protocol agnostic. In addition, we only use the first 1 024 samples of $\hat{m}_{j'}(t)$.

1) Pseudonoise Extraction: Suppose we have derived the numerical sequence of instantaneous metrics (amplitude, phase, or frequency), corresponding procedures are as follows.

Step 1: We separate the sequence [denoted as $s_{j'}(n)$] into several nonoverlap segments, with each segment's duration less than one symbol duration.

Step 2: For each segment, we perform *k-medoids* algorithm on signals instantaneous phase or amplitudes with $k = 2$. In essence, we use a clustering algorithm to associate numeric values to their closest medoids (representative values). Notably, we could only expect one or two possible choices of amplitudes or phases.

Step 3: In each segment, we generate the pseudonoise as

$$n_{j'}(n) = s_{j'}(n) - m_k\left[s_{j'}(n)\right] \tag{7}$$

where m_k denotes the medoid of $s_{j'}(n)$, We subtract rationale signals from the demodulated baseband signals directly.

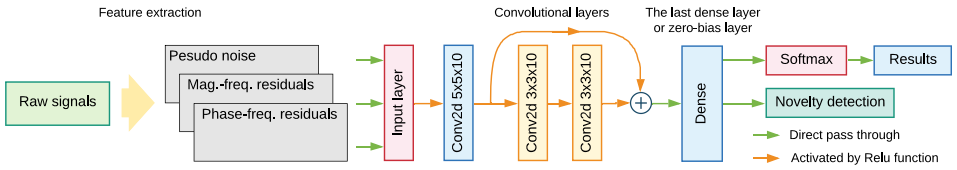


Fig. 2. Deep neural architecture for wireless transmitter identification.

A brief comparison of related signals is in Fig. 1(a). Medoids could be regarded as a less noisy version of demodulated baseband signals $\hat{m}_{j'}(t)$.

The distribution of correlation coefficients (derive from 10 000 samples) of pseudonoise against corresponding baseband signals is depicted in Fig. 1(b). The pseudo noise signals are weakly correlated with original messages.

2) Frequency Domain Features: We subtract the Fourier transforms of both complex-valued baseband signals $\hat{m}_{j'}(t)$ and the reconstructed rationale baseband signals to extract message uncorrelated residual components in the frequency domain, formulated as

$$\delta_j(\omega) = \mathrm{FFT}\big[\hat{m}_{j'}(t)\big] - \mathrm{FFT}\big[r_{j'}(t)\big] \tag{8}$$

where $r_{j'}(t)$ is the reconstructed rational baseband signal. Note that $\hat{m}_{j'}(t)$ is complex valued (QPSK) while $r_{j'}(t)$ can be real valued (2FSK, 2PSK, etc.). We convert residual components into a magnitude sequence ($\|\delta_j(\omega)\|$), namely, Mag.-Freq. residuals, and a phase sequence ($\angle\delta_j(\omega)$), namely, Phase-Freq. residuals, respectively.

C. Zero-Bias Deep Learning Framework for Accurate Identification of IoT Devices

In this section, we present our enhancement to conventional neural networks, which is generalizable to other neural-classification problems.

The architecture of deep-learning-enabled classifier for device identification is given in Fig. 2. Convolutional layers with skip connections are employed to extract latent features, we also use a dense layer followed by a softmax layer for final classification. However, in the last dense layer, we propose a modified approach.

Suppose we have m-dimension input vectors with batch size k, we need to convert them into k n-dimension outputs. A conventional dense layer would perform a linear calculation as

$$\boldsymbol{Y_1} = \boldsymbol{W_1 X} + \boldsymbol{b_1} \tag{9}$$

where $\boldsymbol{X}$, $\boldsymbol{b_1}$, and $\boldsymbol{W_1}$ denote the m by k input matrix, bias neurons, and an n by m weights matrix, respectively. If we break the regular dense layer into two consecutive parts, depicted in Fig. 3, a regular dense layer denoted by L_1 and a dense layer L_2 without bias, respectively. Then, (9) becomes

$$\boldsymbol{Y_2} = \boldsymbol{W_2 Y_1} = \boldsymbol{W_2 W_1 X} + \boldsymbol{W_2 b_1} \tag{10}$$

where $\boldsymbol{W_1}$ and $\boldsymbol{b_1}$ belong to L_1 and $\boldsymbol{W_2}$ belongs to L_2, respectively. Note that (10) and (9) are performing equivalent transforms to $\boldsymbol{X}$ and should not degrade the network

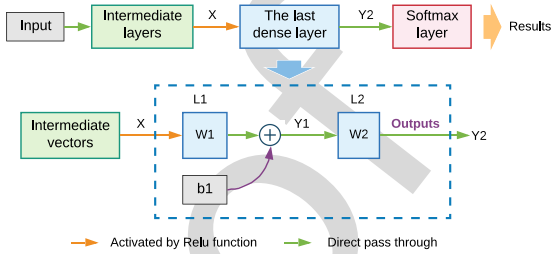


Fig. 3. Data flow of the zero-bias dense layer.

performance. Moreover, in L_2, we can rewrite the matrix calculation into vectors

$$\boldsymbol{Y_2}\big[\boldsymbol{y_{1k}}\big] = \big[\boldsymbol{w_{21}} \cdot \boldsymbol{y_{1k}}, \boldsymbol{w_{22}} \cdot \boldsymbol{y_{1k}}, \ldots, \boldsymbol{w_{2n}} \cdot \boldsymbol{y_{1k}}\big] \tag{11}$$

where $\boldsymbol{w_{21}}, \ldots, \boldsymbol{w_{2n}}$ are row vectors corresponding to n output classes, $\boldsymbol{y_{1k}}$ is one of the k column vectors in batch, and $\boldsymbol{Y_2}[\boldsymbol{y_{1k}}]$ is the output vector. The process in (11) can be rewritten using *cosine similarity*

$$\boldsymbol{w_{2n}} \cdot \boldsymbol{y_{1k}} = \|\boldsymbol{w_{2n}}\| \cdot \|\boldsymbol{y_{1k}}\| \cdot \cos(\boldsymbol{w_{2n}}, \boldsymbol{y_{1k}}). \tag{12}$$

If L_2 is followed by a softmax layer and we take $\boldsymbol{w_{21}}, \ldots, \boldsymbol{w_{2n}}$ as fingerprints of classes 1 to n, we conclude that L_2 actually calculates a scaled version of cosine similarities among input against fingerprints of target classes.

Moreover, we can safely generalize this discovery to understand the behavior of last dense layers in neural networks.

Remark 1 (Property of Dense Layers): If an output vector of a dense layer represent the degrees of confidence of corresponding class/position against an input, then each confidence degree is jointly controlled by the magnitude of the class/position-related fingerprint, the fingerprint's cosine similarity to the input, and the bias neuron of this class.

Although the magnitude of an input feature vector $\|\boldsymbol{y_{1k}}\|$ seems to take effect as in (12), but in the consecutive softmax layer, the magnitude $\|\boldsymbol{y_{1k}}\|$ only contributes to a common base number as

$$\text{class} = \frac{\exp\big[\|\boldsymbol{y_{1k}}\| \cdot \|\boldsymbol{w_{2n}}\| \cdot \cos(\boldsymbol{w_{2n}}, \boldsymbol{y_{1k}})\big]}{\sum_n \exp\big[\|\boldsymbol{y_{1k}}\| \cdot \|\boldsymbol{w_{2n}}\| \cdot \cos(\boldsymbol{w_{2n}}, \boldsymbol{y_{1k}})\big]} \tag{13}$$

where the base number, $\exp\|\boldsymbol{y_{1k}}\|$ only controls the steepness of the monotonic mapping curve. According to Remark 1, we can derive another important remark.

Remark 2 (Neural Networks' Partiality): As long as prior layers do not converge to constant functions, A neural network's partiality to specific classes is encoded in its last dense layer before softmax, and the bias is jointly controlled by the magnitude of class-related fingerprint vector and the bias neuron of the corresponding class.

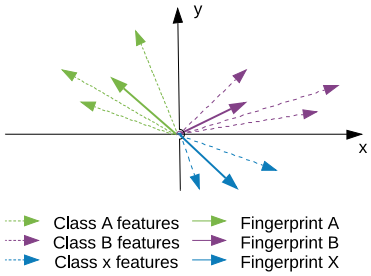


Fig. 4. Relation of fingerprint vectors and feature vectors.

In our proposed paradigm of dense layer without bias neurons, we can derive more specific corollaries.

Corollary 1 (Fingerprints' Magnitude): If the variance of the magnitude of fingerprints vectors is small, the layer L_2 has less bias to specific classes.

Currently, we have two approaches to remove the unwanted effects of fingerprint vectors' magnitudes.

1) We can use regularization to eliminate the variance of fingerprints, we make their values relative close.
2) We can replace (11) with the following equation:

$$\boldsymbol{Y_2} = \left[\frac{\boldsymbol{w_{21}}}{\sqrt{\boldsymbol{w_{21}^2}}}, \ldots, \frac{\boldsymbol{w_{2n}}}{\sqrt{\boldsymbol{w_{2n}^2}}}\right]^T \left[\boldsymbol{y_{11}}, \ldots, \boldsymbol{y_{1k}}\right]. \quad (14)$$

Moreover, we can eliminate the side effects of feature vectors' magnitude at the same time

$$\boldsymbol{Y_2} = \lambda \left[\frac{\boldsymbol{w_{21}}}{\sqrt{\boldsymbol{w_{21}^2}}}, \ldots, \frac{\boldsymbol{w_{2n}}}{\sqrt{\boldsymbol{w_{2n}^2}}}\right]^T \left[\frac{\boldsymbol{y_{11}}}{\sqrt{\boldsymbol{y_{11}^2}}}, \ldots, \frac{\boldsymbol{y_{1k}}}{\sqrt{\boldsymbol{y_{1k}^2}}}\right] \quad (15)$$

where λ is a trainable value to provide the freedom of controlling the steepness of the mapping curve in the softmax layer. Please be noted that $\boldsymbol{Y_2}$s are differentiable in these two scenarios and (14) is still equivalent to linear operations.

We eliminate the classifiers' partiality or bias. We treat the possibility of each class equally and its the essence of "zero-bias" dense layer. With the zero-bias enhancement, we have Corollary 2.

Corollary 2 (Fingerprints' Mutual Distances): Fingerprints in the zero bias dense layer (L_2) act as angular representatives of corresponding classes and should have sufficiently small mutual cosine similarities.

A simplified example of Corollary 2 is given in Fig. 4, suppose we have three classes (A, B, and X) for a DNN to distinguish from, the fingerprint vector of each class only captures a representative direction. With this property, we only need to insert or remove fingerprints in L_2, to register or remove corresponding classes.

Another benefit is to evaluate how well different classes are mutually distinguishable from each other. We can construct a fingerprint distance (FD) matrix as

$$FD = \begin{bmatrix} \cos(\boldsymbol{w_1}, \boldsymbol{w_1}) & \cdots & \cos(\boldsymbol{w_1}, \boldsymbol{w_n}) \\ \vdots & \ddots & \vdots \\ \cos(\boldsymbol{w_n}, \boldsymbol{w_1}) & \cdots & \cos(\boldsymbol{w_n}, \boldsymbol{w_n}) \end{bmatrix}. \quad (16)$$

This matrix can directly reflect how well different classes are separated in the latent space. We replace the last dense layer with the zero-bias dense layer (contains both L_1 and L_2) in the MNIST example [26] and plot the FD matrices when training accuracy reaches 60.2% and 95.8%, respectively. As in Fig. 5, fingerprints are distantly separated with higher accuracy.

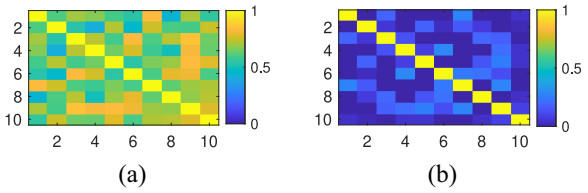


Fig. 5. FD matrix of Minst example. (a) After 1 epoch. (b) After 10 epochs.

In this section, we propose a new scheme of creating zero-bias neural networks and a thorough analysis of the mechanism of dense layers. A summary of our enhancement is given as follows.

Remark 3 (Zero-Bias Layer Enhancement): We replace the last dense layer of a neural network with a consecutive structure consisting of a regular dense layer (L_1) and a zero-bias similarity comparing layer (L_2).

We note that some researches directly employ (15) as cosine similarity [27], [28] in deep learning, we differentiate from them as: 1) we provided a mathematically equivalent transform, by using another regular fully connected layer L_1 and 2) our experiments show that directly applying cosine similarity without L_1 dramatically increases the difficulty of training.

D. Novel Device Identification

A wireless device identification system needs to identify anomalous signals from novel devices. In a conventional neural network, the softmax layer associates labels to the largest activation. Such behavior would result in wrong answers given falsified signals from unknown devices. Suppose that the zero-bias layer enhancement in (15) is applied, the output of the layer directly represent cosine similarities. We define the concept *Similarity Response* as follows.

Definition 1 (Similarity Response): For input, the maximum value in the output vector after zero-bias or regular dense layer is defined as its similarity response.

An unknown device with the false identity can be detected if its signals' similarity responses are below a reasonable threshold. For example, if the similarity response of known devices follows a Gaussian distribution, $N(m_k, \sigma_k)$, an input with the highest similarity less than $m_k - \sigma_k$ can be subject to novel or even spoofing device.

V. Performance Evaluation

Automatic dependent surveillance-broadcast (ADS-B) [29], which accurately observe and track air traffic, is a fundamental

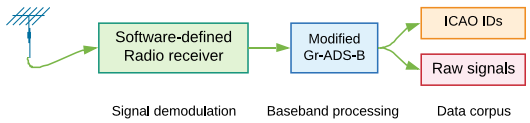


Fig. 6. Collection of ADS-B signals.

Fig. 7. Geographic distribution of aircraft transponders.

safety infrastructure modern aviation. This system is designed to be simple and widely adaptable but its extremely vulnerable to identity spoofing attacks. In this section, we present our performance evaluation results using real ADS-B data and demonstrate how our proposal could be elegantly applied in practical systems.

A. Evaluation Data Set

Nowadays, Commercial aircraft are equipped with dedicate 1090-MHz transponders to broadcast its geocoordinates, velocities, altitudes, headings, as well as their unique identifiers, also known as International Civil Aviation Organization (ICAO) IDs. Such signals provide a great variety of signals from known wireless devices. In our data collection pipeline depicted in Fig. 6, we used a modified *gr-adsb* library to decode ADS-B messages and store raw baseband digital signals. We collected the ADS-B signal from more than 140 aircraft at Daytona Beach international airport (ICAO: DAB) for 24 h (January 4, 2020) using an SDR receiver (USRP B210). The receiver is configured with a sample rate of 8 MHz. During this period, more than 30 000 ADS-B messages are collected with coordinates and SNR (in colors) depicted in Fig. 7.

B. Known Device Verification

We first conduct a general performance test of the system (depicted in Fig. 2). As depicted, the deep learning model can associate received signals with accuracy greater than 94.3%. Furthermore, a brief comparison of DNN with the proposed zero-bias layer, regular dense layer, and only cosine similarity before softmax[2] on the same data set is given in Fig. 8. As depicted, DNNs with the zero-bias layer or regular dense layer reach almost identical performance. However, the zero-bias layer requires more training iterations, and its rising rate of accuracy is lower at the beginning. Interestingly, if we only use cosine similarity directly after convolutions, the deep learning system cannot converge.

[2]Similar network architecture with cosine similarity and softmax directly after convolution filters.

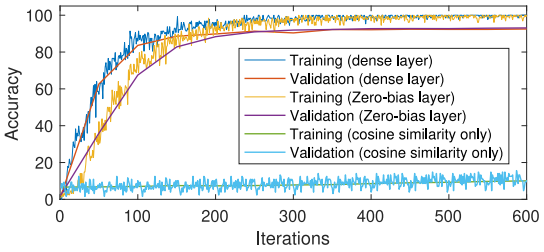


Fig. 8. Comparison of training performance.

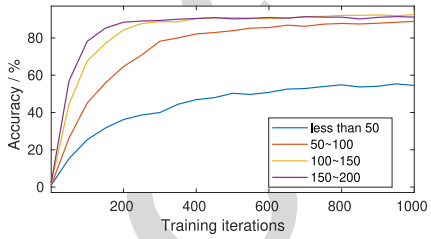


Fig. 9. Validation accuracy in terms of training data size for each transmitter.

To evaluate the deep learning model in terms of training data quantity, we manually limit the number of samples of each transmitter in the training set and use this specially "reduced" training set to train the zero-bias DNN model. As depicted in Fig. 9, the model converges after 800 iterations (40 epochs) and show that we only need 200 samples to recognize each transmitter.

C. Novel Device Identification

We randomly pick ADS-B signals from 30 aircraft to train the neural network and use signals from the remaining 120 aircraft as unseen novel devices' signals. We compare the performance of our zero-bias layer, regular dense layer, and one-class support vector machine (SVM), respectively. In this section, we define the optimal decision boundary as

$$\max_{\tau} \|\text{cdf}(P_u(\tau)) - \text{cdf}(P_k(\tau))\| \tag{17}$$

where $P_u(\tau)$ and $P_k(\tau)$ are probability distribution functions of similarity response of unknown and known devices. $\text{cdf}(\cdot)$ denotes the cumulative density function.

1) Zero-Bias and Regular Dense Layer: We employ the zero-bias layer [use (15)] for final output. The probability distribution and decision thresholds are given in Fig. 10(a) and (d), respectively. Fig. 10(a) demonstrates that the similarities response of unknown signals are higher than unknown signals in most cases. Fig. 10(d) shows that we can easily select an optimum separation threshold to maximize the decision boundary of the anomaly detection algorithm. In our application, we choose the median value of similarity responses on known signals minus its standard deviation as a decision threshold.

We train the identical neural network but with the zero-bias layer replaced by the regular dense layer. But the anomaly detection performances are much worse, as depicted

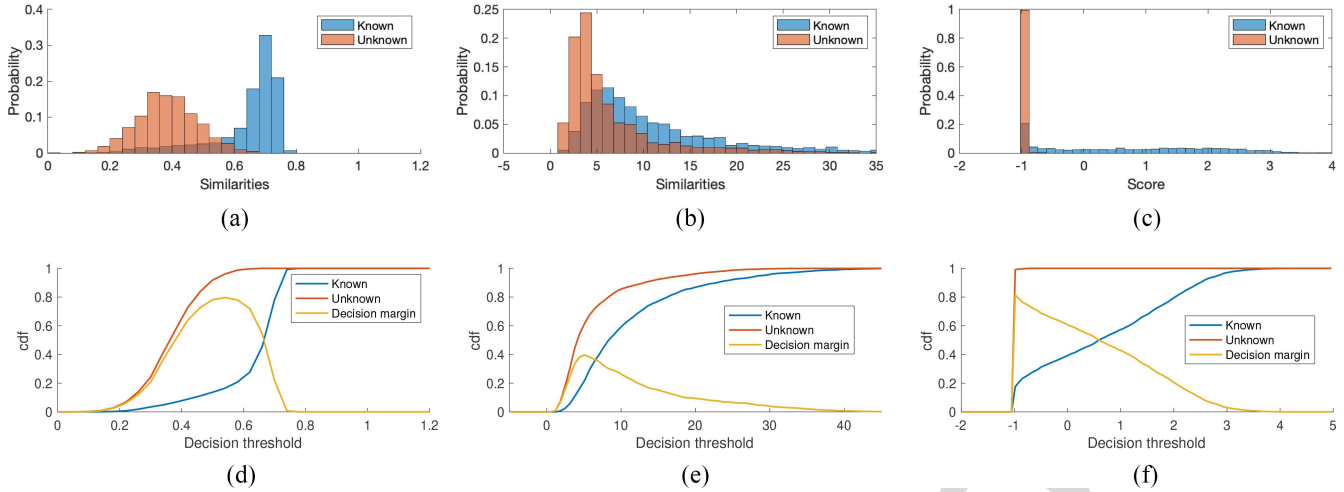


Fig. 10. Performance of Threshold-based anomaly detections. (a) Zero-bias DNN. (b) Regular DNN. (c) One-class SVM. (d) Zero-bias DNN. (e) Regular DNN. (f) One-class SVM.

in Fig. 10(b) and (e), the similarity response of regular dense layer on known and unknown data are severely overlapped. The optimal decision boundary in this scenario is small.

2) One-Class SVM: We use the feature vectors in training signals (directly produce by convolutional layers) of zero-bias DNN to train a one-class SVM model, we then use feature vectors from the validation set as unseen signals to test the performance of one-class SVM. We collect the prediction scores on both known signals and unknown signals with statistic results presented in Fig. 10(c) and (f), respectively. The result indicates that the prediction scores of known devices' signal occupy a much wider area (larger variance), which may cause difficulty for choosing the right threshold. The fact indicates that the performance of the zero-bias layer-enabled DNN in anomaly detection is comparable with one-class SVM. However, in our experiment, the one-class SVM model ultimately stores more than 5000 support vectors, while the zero-bias layer only stores directional fingerprints of known aircraft transponders (less than 200). Therefore, we believe our solution is more adaptable for real-time machine learning.

VI. Conclusion

In this article, we propose a novel deep learning framework for IoT device identification. Different from existing works, we focus on how to enable deep learning to be practically usable and dependable. Our contributions are as follows. First, we analyze the mathematical essence of IoT device identification and use residual signals to identify real-world ADS-B transmitters. We got a promising recognition rate of 94% among more than 130 airborne transponders. Second, we thoroughly analyze the behavior of the last fully connected layer in DNNs and propose our improvement, the zero-bias layer, for interpretable and dependable machine learning in IoT. Experiments show that we obtain equivalent accuracy compared to the regular DNN, but obtain much better performances in terms of anomaly detection. Therefore, we believe the zero-bias layer can be generalized to other domains, such as virus detection or unsupervised intrusion detection. In the future, we will focus on how to efficiently discover reusable function blocks in pretrained networks and apply them to new domains.

References

[1] S. Jeschke, C. Brecher, H. Song, and D. Rawat, *Industrial Internet of Things: Cybermanufacturing Systems*. Cham, Switzerland: Springer, 2017. AQ3

[2] G. Dartmann, H. Song, and A. Schmeink, *Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things*. Amsterdam, The Netherlands: Elsevier, 2019.

[3] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of Things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, 2016.

[4] I. Butun, P. Österberg, and H. Song, "Security of the Internet of Things: Vulnerabilities, attacks, and countermeasures," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 616–644, 1st Quart., 2020.

[5] Y. Liu, J. Li, Z. Ming, H. Song, X. Weng, and J. Wang, "Domain-specific data mining for residents' transit pattern retrieval from incomplete information," *J. Netw. Comput. Appl.*, vol. 134, pp. 62–71, May 2019.

[6] A. F. Skarmeta, J. L. Hernandez-Ramos, and M. V. Moreno, "A decentralized approach for security and privacy challenges in the Internet of Things," in *Proc. IEEE World Forum Internet Things (WF-IoT)*, Seoul, South Korea, 2014, pp. 67–72.

[7] J. Wang *et al.*, "Fountain code enabled ADS-B for aviation security and safety enhancement," in *Proc. IEEE 37th Int. Perform. Comput. Commun. Conf. (IPCCC)*, Orlando, FL, USA, 2018, pp. 1–7.

[8] X. Yue, Y. Liu, J. Wang, H. Song, and H. Cao, "Software defined radio and wireless acoustic networking for amateur drone surveillance," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 90–97, Apr. 2018.

[9] W. Wang, Z. Sun, S. Piao, B. Zhu, and K. Ren, "Wireless physical-layer identification: Modeling and validation," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 9, pp. 2091–2106, Sep. 2016.

[10] J. Wang, N. Juarez, E. Kohm, Y. Liu, J. Yuan, and H. Song, "Integration of SDR and UAS for malicious Wi-Fi hotspots detection," in *Proc. IEEE Integr. Commun. Navig. Surveillance Conf. (ICNS)*, Herndon, VA, USA, 2019, pp. 1–8.

[11] Y. Zou, Y. Wang, S. Ye, K. Wu, and L. M. Ni, "TagFree: Passive object differentiation via physical layer radiometric signatures," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Kona, HI, USA, 2017, pp. 237–246.

[12] S. Chen, S. Zheng, L. Yang, and X. Yang, "Deep learning for large-scale real-world ACARS and ADS-B radio signal classification," 2019. [Online]. Available: arXiv:1904.09425.

[13] F. Restuccia *et al.*, "DeepRadioid: Real-time channel-resilient optimization of deep learning-based radio fingerprinting algorithms," 2019. [Online]. Available: arXiv:1904.07623.

[14] B. Chatterjee, D. Das, S. Maity, and S. Sen, "RF-PUF: Enhancing iot security through authentication of wireless nodes using in-situ machine learning," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 388–398, Feb. 2019.

[15] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," *Proc. IEEE*, vol. 102, no. 8, pp. 1126–1141, Aug. 2014.
[16] M. Azarmehr, A. Mehta, and R. Rashidzadeh, "Wireless device identification using oscillator control voltage as RF fingerprint," in *Proc. IEEE 30th Can. Conf. Elect. Comput. Eng. (CCECE)*, Windsor, ON, Canada, 2017, pp. 1–4.
[17] Z. Zhuang *et al.*, "FBSleuth: Fake base station forensics via radio frequency fingerprinting," in *Proc. Asia Conf. Comput. Commun. Security*, 2018, pp. 261–272.
[18] 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.
[19] B. Danev, H. Luecken, S. Capkun, and K. El Defrawy, "Attacks on physical-layer identification," in *Proc. 3rd ACM Conf. Wireless Netw. Security*, 2010, pp. 89–98.
[20] A. C. Polak and D. L. Goeckel, "Identification of wireless devices of users who actively fake their RF fingerprints with artificial data distortion," *IEEE Trans. Wireless Commun.*, vol. 14, no. 11, pp. 5889–5899, Nov. 2015.
[21] M. Köse, S. Taşcioğlu, and Z. Telatar, "RF fingerprinting of IoT devices based on transient energy spectrum," *IEEE Access*, vol. 7, pp. 18715–18726, 2019.
[22] J. Yu *et al.*, "Radio frequency fingerprint identification based on denoising autoencoders," 2019. [Online]. Available: arXiv:1907.08809.
[23] J. Huang, Y. Lei, and X. Liao, "Communication transmitter individual feature extraction method based on stacked denoising autoencoders under small sample prerequisite," in *Proc. 7th IEEE Int. Conf. Electron. Inf. Emerg. Commun. (ICEIEC)*, Macau, China, 2017, pp. 132–135.
[24] S. Riyaz, K. Sankhe, S. Ioannidis, and K. Chowdhury, "Deep learning convolutional neural networks for radio identification," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 146–152, Sep. 2018.
[25] C. Morin, L. Cardoso, J. Hoydis, J.-M. Gorce, and T. Vial, "Transmitter classification with supervised deep learning," 2019. [Online]. Available: arXiv:1905.07923.
[26] *Create Simple Deep Learning Network for Classification*, MathWorks, Natick, MA, USA, May 2018. https://www.mathworks.com/help/deeplearning/ug/create-simple-deep-learning-network-for-classification.html
[27] S. Gidaris and N. Komodakis, "Dynamic few-shot visual learning without forgetting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 4367–4375.
[28] C. Luo, J. Zhan, X. Xue, L. Wang, R. Ren, and Q. Yang, "Cosine normalization: Using cosine similarity instead of dot product in neural networks," in *Proc. Int. Conf. Artif. Neural Netw.*, 2018, pp. 382–391.
[29] J. Sun. (May 2017). *An Open-Access Book About Decoding Mode-S and ADS-B Data*. [Online]. Available: https://mode-s.org/

Yongxin Liu received the first Ph.D. degree from the South China University of Technology, Guangzhou, China. He is currently pursuing the second Ph.D. degree with the Department of Electrical Engineering and Computer Science, Embry-Riddle Aeronautical University, Daytona Beach, FL, USA.

His major research interests include data mining, wireless networks, the Internet of Things, and unmanned aerial vehicles.

Jian Wang received the M.S. degree from South China Agricultural University, Guangzhou, China, in 2017. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Computer Science, Embry-Riddle Aeronautical University, Daytona Beach, FL, USA.

He is a Graduate Research Assistant with the Security and Optimization for Networked Globe Laboratory (www.SONGLab.us), Embry-Riddle Aeronautical University. His research interests include wireless networks, unmanned aerial systems, and machine learning.

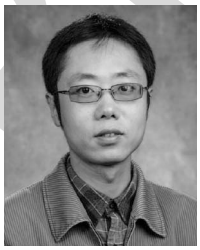
Jianqiang Li (Member, IEEE) received the B.S. and Ph.D. degrees from the South China University of Technology, Guangzhou, China, in 2003 and 2008, respectively.

He is a Professor with the College of Computer and Software Engineering, Shenzhen University, Shenzhen, China. His major research interests include Internet of Things, robotic, hybrid systems, and embedded systems.

Houbing Song (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in 2012.

In August 2017, he joined the Department of Electrical Engineering and Computer Science, Embry-Riddle Aeronautical University, Daytona Beach, FL, USA, where he is currently an Assistant Professor and the Director of the Security and Optimization for Networked Globe Laboratory (www.SONGLab.us).

Dr. Song serves as an Associate Technical Editor for *IEEE Communications Magazine* and an Associate Editor for the IEEE INTERNET OF THINGS JOURNAL.



Thomas Yang (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Central Florida, Orlando, FL, USA, in 2004.

He is currently a Full Professor of electrical and computer engineering with Embry-Riddle Aeronautical University, Daytona Beach, FL, USA. His research interests include signal processing for wireless communication, autonomous multiagent systems, and machine learning.

Shuteng Niu received the M.S. degree from Embry-Riddle Aeronautical University, Daytona Beach, FL, USA, in 2018, where he is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Computer Science.

He is a Graduate Research Assistant with the Security and Optimization for Networked Globe Laboratory (www.SONGLab.us), Embry-Riddle Aeronautical University. His research interests include machine learning, data mining, and signal processing.

Zhong Ming (Member, IEEE) is a Professor with the College of Computer and Software Engineering, Shenzhen University, Shenzhen, China. He led three projects of the National Natural Science Foundation, and two projects of the Natural Science Foundation of Guangdong Province, China. His major research interests include home networks, Internet of Things, and cloud computing.

Prof. Ming is a Senior Member of the Chinese Computer Federation.