# Active Attack Detection Based on Interpretable Channel Fingerprint and Adversarial Autoencoder

Zijie Ji*, Binbing Yang†, Phee Lep Yeoh†, Yan Zhang*, Zunwen He*, and Yonghui Li†

* School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, P.R.China
† School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia
Emails:{jizijie, zhangy *(Corresponding author)*, hezunwen}@bit.edu.cn, {byan0078, phee.yeoh, yonghui.li}@sydney.edu.au.

ICC 2022 - IEEE International Conference on Communications | 978-1-5386-8347-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICC45855.2022.9838483

***Abstract*—This paper investigates how to build an active attack detection framework that is driven by fundamental channel modeling and practical wireless datasets. Firstly, we propose the concept of interpretable channel fingerprints (ICFs), which correspond to the spatial-temporal parameters in real physical wireless signal propagation channels. Based on this, we design an adversarial autoencoder (AAE) with a semi-supervised learning network, which takes as inputs the power spectrum of quantized ICFs and enables small sample learning multiclassification tasks for different types of wireless channel active attacks. We have experimentally verified the performance of our AAE network using the Wireless InSite ray tracing software. Our results show that the proposed semi-supervised network outperforms the fully-supervised network especially in small sample conditions. We highlight the need for careful selection of the hyperparameters for learning rate and mini-batch size, and the system parameters for the ICF power spectrum resolution. We show that the detection accuracy of the proposed AAE model can reach more than 98% with only a small number of input samples.**

***Index Terms*—Active attack detection, adversarial autoencoder, deep learning, physical layer security.**

## I. Introduction

Channel probing is an indispensable phase in wireless communication systems, where estimates of the channel state information (CSI) are used for various applications such as beamforming [1], localization [2], and physical-layer key generation [3]. These applications assume that channel estimates are sufficiently accurate, which could be disrupted by active attacks from malicious wireless nodes. Therefore, it is essential to identify active attacks while transmitting and receiving data in wireless communication systems.

There are many well-known techniques for identifying active wireless attacks. Automatic modulation recognition can be used to detect modulation modes of unknown signals [4], but they are ineffective against more proactive attacks such as pilot contamination attacks (PCA). To this end, random pilots-based attack detection methods [5], [6] are proposed to address the problem of PCA. However, such approaches require randomization of pilots and modification of the protocol, which is not always practical. Active attack detection schemes based on radio frequency fingerprints (RFF) [7], [8] have also been considered. In these schemes, hardware imperfections at the wireless nodes can result in a unique RFF, which is used to distinguish different devices. Therefore, a PCA can be detected based on the unknown RFF from an illegitimate node. A limitation of RFF identification is that it relies heavily on accurate channel compensation, and the detection accuracy is also compromised when the number of devices increases, especially when they are from the same manufacturer [7].

The aforementioned approaches neglect the random characteristics of wireless channels, which can be used to provide additional security dimensions. In [9], a two dimensional channel quantization was proposed for physical-layer authentication enhancement, which was formulated as a threshold-based binary hypothesis testing problem. Recently, a threshold-free physical-layer authentication was proposed based on the strong prediction capability of machine learning algorithms [10]. In [11], the authors employed a deep learning structure termed single-hidden-layer multiple measurement (SHMM) Siamese network to detect PCA with a high accuracy. However, most machine learning schemes use the channel matrices as the input of the network for training, such that they are purely data-driven which is effective but inexplicable. Meanwhile, fully-supervised learning requires a huge amount of labeled data for training to achieve preferable performance, while data acquisition is usually time-consuming and laborious.

To tackle the above problems, we propose a joint model- and data- driven approach termed interpretable channel fingerprint and adversarial autoencoder (ICF-AAE) based active attack detection. The contributions of this work are summarized as follows.

- We propose to use the power spectrum composed of a set of channel parameters including quantized delay, azimuth angle of arrival (AAoA), and elevation angle of arrival (EAoA) as the channel fingerprint, an interpretable indicator corresponding to real signal propagation, instead of the more common channel impulse response (CIR).
- We design an AAE based semi-supervised network to implement attack detection, which supports small sample and multi-classification learning. It outperforms the fully-supervised methods when the labeled sample is limited, and it is able to distinguish multiple attack types to prevent large-scale attackers.
- Extensive experimental results are conducted using wireless signal propagation data generated from the Wireless Insite ray tracing based software for a typical industrial wireless scenario. Moreover, the impact of system and network parameters including spacial and temporal res-

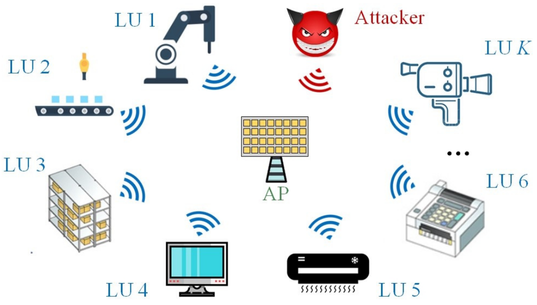

Fig. 1. Industrial wireless IoT system model with an active attacker.

olutions, learning rate, and the number of epochs are analyzed, which provides useful insights for its practical implementation.

The remainder of this paper is organized as follows. Section II introduces the system and attack model. Section III presents the proposed ICF-AAE based attack detection scheme. Experimental setups, results, and discussions are provided in Section IV. Conclusions are drawn in Section V.

## II. System and Attack Model

### A. System Model

As illustrated in Fig. 1, we consider a typical in-building industrial wireless scenario, where a central controller/access point (AP) equipped with an $N_{\mathrm{r}}$-antenna uniform rectangular array (URA) serves a large number of low-cost single-antenna Internet of Things (IoT) devices. The locations of all $K$ legitimate users (LUs) are fixed, such as sensors and actuators configured on robots and operational facilities. To satisfy massive connectivity, the transceivers adopt a random access scheme, where random but unique orthogonal pilots are allocated for different LUs at each time slot. The assigned pilot can be used for both channel estimation and identification.

### B. Attack Model

We assume there is an active attacker who can disrupt the channel probing process by either transmitting legitimate pilots to mimic a LU or sending jamming signals to interfere the AP. Specifically, in this paper we consider four types of attacks:

- Camouflage attack: The attacker located close to one LU listens to the global channels and transmits a selected pilot to the AP to disguise as the victim LU assigned to this pilot when the victim is not transmitting.
- Collocated attack: The attacker is close to one LU so that their spatial locations can be considered overlapped and sends a selected pilot in regardless of LUs' states.
- PCA: The non-collocated attacker transmits the same pilot when the LU allocated this pilot is transmitting so that the AP misestimates the legitimate channel.
- Jamming attack: Instead of transmitting pilot signals, the attacker sends interference signals to the AP, which can randomize the channel estimations of LUs.

The difficulty of detecting these four attacks in the signal domain is arranged from hardest for the camouflage attack to easiest for the jamming attack, and accurately identifying all of them requires a combination of multiple existing methods, which increases the complexity of the system. However, since the location or selected pilot/jamming signal of the attacker is different from LUs', their physical propagation paths, i.e., ICFs, will be distinct which can be utilized by the AP to detect different active attacks.

## III. Proposed ICF-AAE Based Attack Detection

In this section, we first introduce how to construct the ICF from the received signals, and then present the structure of our AAE network. By feeding the obtained ICFs into the designed AAE and training the network in a semi-supervised manner, it can distinguish whether an attack exists and the specific attack type with a high accuracy even when the sample size is small.

### A. Interpretable Channel Fingerprint

Although CSI is also a location-dependent characteristic, it is the vector superposition of multipath, where distinct features that can be used for attack detection may be cancelled out. To this end, inspired by channel modeling, we consider to exploit the propagation parameters of multiple paths as a unique ICF, which increases the dimension of variables and thus makes "fingerprints" more legible.

To collect ICFs, the high-resolution space-alternating generalized expectation maximization (SAGE) algorithm [12] is employed to jointly estimate channel parameters (including the complex amplitude, delay, AAoA, and EAoA) of the multipath components (MPCs). Given the pilot sequence allocated for LU $k \in \{1, 2, ..., K\}$ as $x_k(t)$, the signal propagated through $M$ paths and received at the AP can be expressed by

$$\mathbf{y}(t; \mathbf{\Theta}_{k,m}) = \sum_{m=1}^{M} \alpha_{k,m} \mathbf{a}(\mathbf{\Omega}_{k,m}) x_k(t - \tau_{k,m}) + \sqrt{\frac{N_0}{2}} \mathbf{n}(t), \tag{1}$$

$$\mathbf{a}(\mathbf{\Omega}_{k,m}) = [a_1(\mathbf{\Omega}_{k,m}), a_2(\mathbf{\Omega}_{k,m}), ..., a_{N_{\mathrm{r}}}(\mathbf{\Omega}_{k,m})]^T, \tag{2}$$

$$a_n(\mathbf{\Omega}_{k,m}) = \exp(j 2\pi \lambda^{-1} \langle e(\mathbf{\Omega}_{k,m}), r_n \rangle), n = 1, 2, ..., N_{\mathrm{r}}, \tag{3}$$

$$\begin{aligned} &\langle e(\mathbf{\Omega}_{k,m}), r_n \rangle = [x_n, y_n, z_n] \\ &\times [\cos(\phi_{k,m}) \sin(\theta_{k,m}), \sin(\phi_{k,m}) \sin(\theta_{k,m}), \cos(\theta_{k,m})]^T, \end{aligned} \tag{4}$$

where $\mathbf{a}(\Omega_{k,m})$ is the steering vector of the URA with the $n$th component $a_n(\Omega_{k,m})$, $r_n$ is the location of the $n$th antenna element with coordinates of $(x_n, y_n, z_n)$ with respect to a reference point, and $\mathbf{n}(t)$ is the $N_{\mathrm{r}} \times 1$-dimensional complex white Gaussian noise with the power of $N_0$. $\mathbf{\Theta}_{k,m} = [\alpha_{k,m}, \tau_{k,m}, \phi_{k,m}, \theta_{k,m}]$, $m = 1, 2, ..., M$, are the channel parameter sets to be estimated, where $\alpha_{k,m}$, $\tau_{k,m}$, $\phi_{k,m}$, and $\theta_{k,m}$ denote the complex amplitude, delay, AAoA, and EAoA for the $m$th MPC, respectively. Given the cost function

$$z(\tau, \mathbf{\Omega}; \hat{\mathbf{x}}(t; \hat{\mathbf{\Theta}}'_{k,m})) = \mathbf{a}^H(\mathbf{\Omega}) \int x_k(t' - \tau) \hat{\mathbf{x}}(t'; \hat{\mathbf{\Theta}}'_{k,m}) dt', \tag{5}$$

$$\hat{\mathbf{x}}(t;\hat{\mathbf{\Theta}}'_{k,m}) = \mathbf{y}(t;\mathbf{\Theta}_{k,m}) - \sum_{m'\neq m}^{M} \mathbf{s}(t;\hat{\mathbf{\Theta}}'_{k,m'}), \qquad (6)$$

where $\mathbf{s}(t;\mathbf{\Theta}_{k,m}) = \alpha_{k,m}\mathbf{a}(\mathbf{\Omega}_{k,m})\delta(t-\tau_{k,m})$ denotes the $m$th MPC for LU $k$ and $(\cdot)^H$ is the conjugate transpose operator, $\hat{\mathbf{\Theta}}''_{k,m}$ can be iteratively estimated based on the previous $\hat{\mathbf{\Theta}}'_{k,m}$ as

$$\hat{\tau}''_{k,m} = \arg\max_{\tau}\left\{\left|z(\tau,\hat{\mathbf{\Omega}}'_{k,m};\hat{\mathbf{x}}(t;\hat{\mathbf{\Theta}}'_{k,m}))\right|\right\}, \qquad (7)$$

$$\hat{\mathbf{\Omega}}''_{k,m} = [\hat{\phi}''_{k,m},\hat{\theta}''_{k,m}] = \arg\max_{\mathbf{\Omega}}\left\{\left|z(\hat{\tau}''_{k,m},\mathbf{\Omega};\hat{\mathbf{x}}(t;\hat{\mathbf{\Theta}}'_{k,m}))\right|\right\}, \qquad (8)$$

$$\hat{\alpha}''_{k,m} = \frac{1}{N_{\mathrm{r}}} z(\hat{\tau}''_{k,m},\hat{\mathbf{\Omega}}''_{k,m};\hat{\mathbf{x}}(t;\hat{\mathbf{\Theta}}'_{k,m})), \qquad (9)$$

where $|\cdot|$ denotes the modulus of a complex number, until the estimated parameter sets $\hat{\mathbf{\Theta}}_{k,m}$ converge to constants.

Thereafter, we transform these obtained channel parameters into images whose features can be extracted and learnt by deep neural networks. The wireless channel reconstructed based on $\hat{\mathbf{\Theta}}_{k,m}$ is expressed by

$$\hat{\mathbf{h}}(t;\hat{\mathbf{\Theta}}_{k,m}) = \sum_{m=1}^{M} \hat{\alpha}_{k,m}\mathbf{a}(\hat{\mathbf{\Omega}}_{k,m})\delta(t-\hat{\tau}_{k,m}), \qquad (10)$$

where $\delta(\cdot)$ is the Dirac delta function. As such, the power delay profile (PDP) and power angular profiles (PAPs) (including azimuth PAP (APAP) and elevation PAP (EPAP)) can be given as

$$P_{\mathrm{PDP},k}(\tau) = \frac{1}{N_{\mathrm{r}}}\left\|\hat{\mathbf{h}}(\hat{\tau}_{k,m})\right\|^2 = \sum_{m=1}^{M} |\hat{\alpha}_{k,m}|^2\delta(\tau-\hat{\tau}_{k,m}), \qquad (11)$$

$$P_{\mathrm{APAP},k}(\phi) = \sum_{m=1}^{M} |\hat{\alpha}_{k,m}|^2\delta(\phi-\hat{\phi}_{k,m}), \qquad (12)$$

$$P_{\mathrm{EPAP},k}(\theta) = \sum_{m=1}^{M} |\hat{\alpha}_{k,m}|^2\delta(\theta-\hat{\theta}_{k,m}), \qquad (13)$$

respectively, where $\|\cdot\|$ is the Euclidean norm. Equations (11)–(13) indicate the statistical properties of the received power in the delay and angular domain, and they are proportional to the distribution of the delay, AAoA, and EAoA of the received signals. Due to the limited number of antennas and bandwidth, the corresponding resolutions in the delay, AAoA, and EAoA domains are $1/B$, $360^\circ/N_{\mathrm{r}}^{\mathrm{h}}$, and $180^\circ/N_{\mathrm{r}}^{\mathrm{v}}$, where $B$, $N_{\mathrm{r}}^{\mathrm{h}}$, and $N_{\mathrm{r}}^{\mathrm{v}}$ denote the bandwidth, the numbers of antennas in horizontal and vertical directions, respectively. Therefore, in the power spectrum of the ICFs, we set delay as the $x$-axis, and the concatenated AAoA and EAoA as the $y$-axis, quantizing and forming images of size $(\tau_{\max}/B)\times(N_{\mathrm{r}}^{\mathrm{h}}+N_{\mathrm{r}}^{\mathrm{v}})$ pixels as the ICFs, where $\tau_{\max}$ is the maximum delay.

Since ICFs show the scattering conditions experienced by wireless signals, active attacks can be identified after obtaining legitimate node locations and their corresponding ICFs. For different attack types, their characteristics on ICFs are also different, so that they can be further distinguished. For example, camouflage attack reflects the location of the attacker itself, PCA corresponds to the superposition of the attacker's location and the location of a legitimate node, while jamming attack makes ICFs scrambled as pilots are drowned in jamming signals.

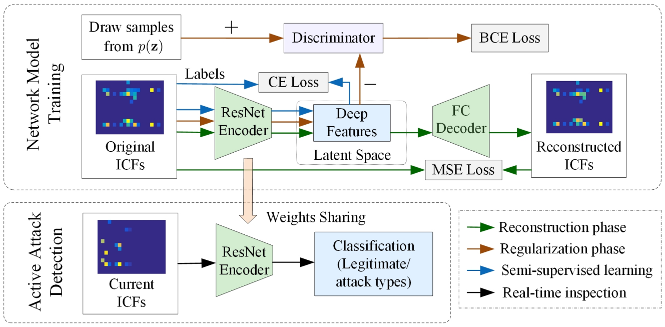

Fig. 2. The architecture of the proposed semi-supervised AAE network and its training and application process for active attack detection.

## B. Adversarial Autoencoder Network

AAE [13] is a promising deep neural network architecture for computer vision-related tasks, which enables small sample semi-supervised learning. As shown in Fig. 2, the proposed model mainly includes three modules (encoder, decoder, and discriminator), and requires three training phases (reconstruction, regularization, and semi-supervised learning) and one application phase (real-time inspection). Specifically, by compressing the sparse ICFs, the encoder ($E$) extracts the deep features that can be used to reconstruct the original images, which as a whole is also called the latent space. The decoder ($D$) restores the images from the latent space and minimizes the gap with the inputs to ensure that the encoder extracts the most informative features. The above process from encoder inputs to decoder outputs is called reconstruction phase, where the reconstruction loss is provided by minimum squared error (MSE). Let the original and reconstructed images be denoted by $\mathbf{x}$ and $\hat{\mathbf{x}}$, the objective function is expressed as

$$\min_{E,D}\left\{\|\mathbf{x}-\hat{\mathbf{x}}\|^2\right\}, \qquad (14)$$

Then, in the regularization phase, the discriminator ($D_{\mathrm{s}}$) is trained to evaluate whether the input samples are from a prior distribution $p(\mathbf{z})$ or from the output of the encoder. In contrast, the encoder is trained to confuse the discriminator. By constructing a min-max adversarial game which can be written as

$$\begin{aligned}\min_{E}\max_{D_{\mathrm{s}}}\{&\mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\left[\log(D_{\mathrm{s}}(\mathbf{z}))\right]\\ &+\mathbb{E}_{\mathbf{x}'\sim p(\mathbf{x}')}\left[\log(1-D_{\mathrm{s}}(E(\mathbf{x}')))\right]\},\end{aligned} \qquad (15)$$

we can impose $p(\mathbf{z})$ on the aggregate posterior of features in the latent space $p(\mathbf{x}')$ so as to make the pre-trained network capable to detect unknown data inputs. One-hot vectors and binary cross entropy (BCE) are employed as the target and loss function. The above two phases are both unsupervised learning and only need unlabeled ICF images.

Next, a small number of labeled samples are fed in to train the encoder to recognize which class the input samples belongs to. The features suitable for reconstruction with a specified distribution are further refined to show the differences between various categories, which is beneficial for multi-classification. The predicted class is determined by using the cross entropy (CE) loss function and softmax out which is defined as

$$\mathrm{Pre} = \arg\min_{i} \left\{ -\sum_{i}^{N_{\mathrm{c}}} t_i \log \left( \frac{e^{x'_i}}{\sum_{j}^{N_{\mathrm{c}}} e^{x'_j}} \right) \right\}, \tag{16}$$

where $x'_i, x'_j \in \mathbf{x}'$, $t_i$ is the $i$th target class, and $N_{\mathrm{c}}$ is the total number of classes. So far, the network training is completed in a semi-supervised manner.

Finally, the weights of well-trained encoder can be transferred for online active attack detection. Meanwhile, the data accumulated during real-time inspection with different accuracies can be used as unlabelled/labelled feedback for dataset expansion and performance improvement.

## IV. Experimental Results and Analysis

### A. Dataset and network setups

As depicted in Fig. 3, the static wireless propagation channels in an industrial factory warehouse scenario is constructed using a commercial ray-tracing software, Wireless InSite [14]. The size of the considered area is approximately 40.2 m × 78.5 m × 20.8 m. There are $K = 40$ LUs fixed on robots and shelves, which are denoted by green blocks located 2 m above the ground and first floor. They communicate with the AP in the middle of the entire space indicated by the yellow block. During the channel probing, an attacker located at the same position as any LU or another position (randomly selected from 302 red blocks located 1.2 m above the ground and first floor) may send attack signals according to the four attack types discussed in Section II-B. These red blocks traverse almost all reachable locations not too far from LUs. The carrier frequency is 2.4 GHz and all antennas are omnidirectional with vertical polarization. According to the generated model, we can obtain the corresponding channel parameter set $\mathbf{\Theta}_{k,m}$ that accurately describes the propagation environment, in which the number of effective scattering paths $M = 25$. The transmitting power is set to 25 dBm and Gaussian noise is added to simulate the estimation errors [12]. Based on Eq. (11)–(13), the estimated channel parameters can be quantized to the closest interval to form the PDP and PAPs. The total number of images used for training and validation is about 12,000, including 2,400 for each of the four attack types and the legitimate case, i.e., the total number of classes is $N_{\mathrm{c}} = 5$. The ratio of samples in the training and validation (finding suitable hyper-parameters) or test (verifying the effectiveness of the trained model) set is 8:2. All samples are resized to 87 × 65 pixels and normalized before being fed into the encoder.

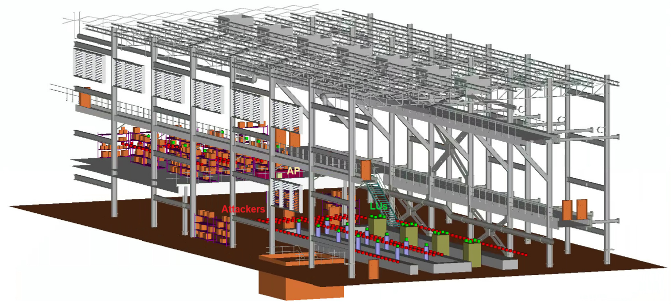

Fig. 3. Ray-tracing based factory model, where the yellow, green, and red blocks denote locations for AP, LUs, and potential attacker, respectively.

As for the network architecture, the classical ResNet50 [15] followed by a fully-connected (FC) layer with a output latent space size of 5 is selected as the encoder, while the decoder and discriminator are both modeled by FC networks. ReLU and Sigmoid function are employed as the activation functions between each two of three linear hidden layers and before the output layers, respectively. Adam optimizer is used in all networks and $p(\mathbf{z}) = Cat(\mathbf{z}) = \mathcal{U}(0,4)$. We note that $p(\mathbf{z})$ can be modified based on the probability of various attack types. The unsupervised reconstruction and regularization phases are pre-trained for 50 epochs and the semi-supervised learning is trained for 150 epochs to observe its convergence. We consider the learning rate $\eta = 1 \times 10^{-4}$, the mini-batch size $\varpi = 32$, the estimation error ratio $\gamma = -50$ dB, $N_{\mathrm{r}}^{\mathrm{h}} = 4$, $N_{\mathrm{r}}^{\mathrm{v}} = 2$, and $B = 100$ MHz, if not specified otherwise.

### B. Experimental results

*Fully vs Semi-supervised learning:* From Fig. 4, we can see that under the condition of the same proportion of labeled data, the detection accuracy achieved by our semi-supervised network is always better than that of the fully-supervised network [15]. This highlights the performance benefits of our semi-supervised network from both having more informative features extracted by the AE and the generalization brought by the adversarial network. Meanwhile, the proposed AAE model converges faster since the weights have undergone pre-training in the reconstruction and regularization phases. Besides, as the proportion of labeled data increases, the gain by using semi-supervised learning decreases, i.e., the gap between the two lines of the same color becomes smaller (from 19% to 6%). This indicates that AAE is more appropriate for the small sample condition, and its performance trained with 10% labeled data even outperforms that of fully-supervised network trained with 30% labeled data. Further increasing the proportion of labeled data has limited performance improvement, because under the case of 50% labeled data, the detection accuracy after convergence is higher than 96%.

*Impact of hyper-parameters:* Fig. 5 provides guidance on how to select a suitable learning rate $\eta$ and mini-batch size $\varpi$. We observe that a smaller learning rate generally generates a better performance. The learning rate determines the step size of the weight update, so the model will not converge if it is set too large, which can be seen in cases with $\eta = 1 \times 10^{-2}$. In comparison, the impact of mini-batch size on performance is relatively limited, especially when the curves converge. To find the best $(\eta, \varpi)$ combination, we average the detection

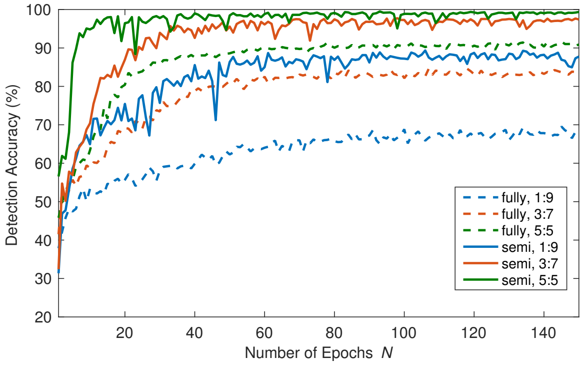

Fig. 4. Detection accuracy versus number of epochs for the fully and semi-supervised learning schemes with various proportion of labeled data.

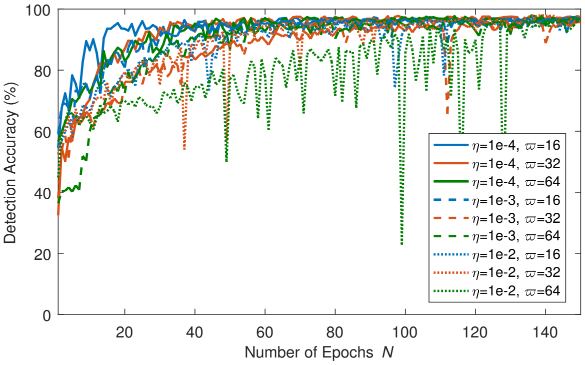

Fig. 5. Detection accuracy versus number of epochs for different learning rate $\eta$ and mini-batch size $\varpi$.

accuracy during the last 50 epochs and select the combination with the largest mean as fixed hyper-parameters for subsequent network training. In this regard, $(1 \times 10^{-4}, 32)$ is selected for $(\eta, \varpi)$, which realizes the highest average accuracy of 96.9% compared to 96.4% achieved by $(1 \times 10^{-4}, 16)$ and 96.5% achieved by $(1 \times 10^{-4}, 64)$.

*Impact of system parameters:* Fig. 6 shows the ICF images of various categories (including one legitimate case and four active attack cases) for the system configured with different numbers of antennas and bandwidths. Note that more antennas and larger bandwidth correspond to higher resolutions in the spatial and delay domain, respectively. Interestingly, we observe that there is an optimal point for the detection performance in the selection of system parameters. We see that setting the resolution too large or too small will deteriorate the detection performance. Specifically, the accuracies accomplished by the first ($N_\mathrm{r}^\mathrm{h} = 4, N_\mathrm{r}^\mathrm{v} = 2, B = 25$ MHz) and the third set of parameters ($N_\mathrm{r}^\mathrm{h} = 64, N_\mathrm{r}^\mathrm{v} = 32, B = 400$ MHz) are only 94.1% and 83.6%, respectively, whereas the second set of parameters ($N_\mathrm{r}^\mathrm{h} = 16, N_\mathrm{r}^\mathrm{v} = 8, B = 100$ MHz) results in an accuracy of 98.2%. This is because too low resolution renders the network unable to fully distinguish multipath, while too high resolution makes images sparse, thus effective features are difficult to be extracted. In Fig. 7, we see that the estimation error ratio $\gamma$ has a significant impact on the detection accuracy. When the estimation error is large, the channel parameters estimated from the received signals from the same location may also be obviously different, so maintaining an accurate estimation is important. When the estimation error ratio is low, e.g., $\gamma = -50$ dB, increasing either the number of antennas or bandwidth can increase the performance. In contrast, when the estimation error ratio is large, increasing the antennas or bandwidth may not result in improved performance because the low resolution ICFs alleviates the diversity caused by estimation errors.

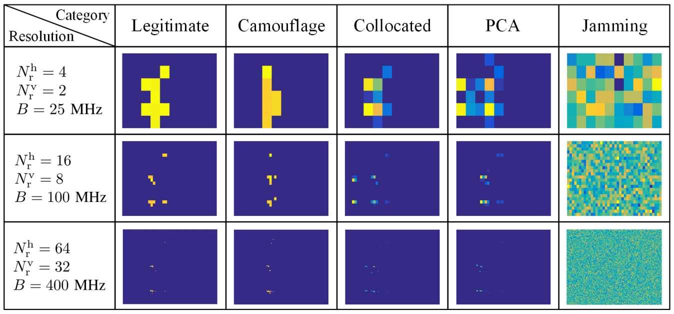
Category
Resolution
Legitimate
Camouflage
Collocated
PCA
Jamming
$N_\mathrm{r}^\mathrm{h} = 4$
$N_\mathrm{r}^\mathrm{v} = 2$
$B = 25$ MHz
$N_\mathrm{r}^\mathrm{h} = 16$
$N_\mathrm{r}^\mathrm{v} = 8$
$B = 100$ MHz
$N_\mathrm{r}^\mathrm{h} = 64$
$N_\mathrm{r}^\mathrm{v} = 32$
$B = 400$ MHz

Fig. 6. Comparisons between categories of ICFs under different resolutions caused by different system parameter settings.

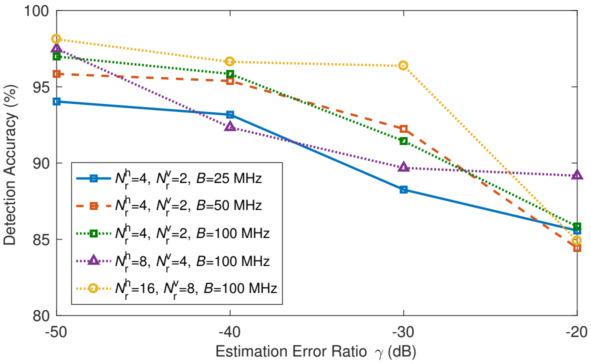

Fig. 7. Detection accuracy as a function of estimation error ratio for different numbers of antennas and bandwidths.

## V. Conclusion

In this paper, we proposed a joint model- and data- driven active attack detection scheme, in which interpretable channel fingerprints and adversarial autoencoder were exploited. By training a semi-supervised network with power spectrum images mapped from real propagation parameters, the existence of active attack and the attack type could be determined. Experimental results verified the advantages of our proposed scheme over the fully-supervised baseline. The impacts of hyper- and system parameters on the security performance were analyzed via multiple comparative experiments, where the guidance on how to select a suitable combination was also provided.

## Acknowledgment

This work was supported by the National Key R&D Program of China under Grant 2020YFB1804901, the National Natural Science Foundation of China under Grant 61871035, the China Scholarship Council scholarship, and Ericsson company.

## References

[1] H. Fu, S. Feng, W. Tang, and D. W. K. Ng, "Robust secure beamforming design for two-user downlink MISO rate-splitting systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8351–8365, Dec. 2020.

[2] L. Ning, B. Li, C. Zhao, Y. Tao, and X. Wang, "Detection and localization of the eavesdropper in MIMO systems," *IEEE Access*, vol. 8, pp. 94984–94993, 2020.

[3] Z. Ji *et al.*, "Wireless secret key generation for distributed antenna systems: A joint space-time-frequency perspective," *IEEE Internet Things J.*, Early access, May 2021.

[4] F. Zhang, C. Luo, J. Xu, and Y. Luo, "An efficient deep learning model for automatic modulation recognition based on parameter estimation and transformation," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3287–3290, Oct. 2021.

[5] J. K. Tugnait, "Pilot spoofing attack detection and countermeasure," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2093–2106, May 2018.

[6] X. Wang, M. Liu, D. Wang, and C. Zhong, "Pilot contamination attack detection using random symbols for massive MIMO systems," in *Proc. IEEE Veh. Technol. Conf. (VTC Spring)*, Sydney, NSW, Australia, Jun. 2017, pp. 1–7.

[7] J. Zhang, R. Woods, M. Sandell, M. Valkama, A. Marshall, and J. Cavallaro, "Radio frequency fingerprint identification for narrowband systems, modelling and classification," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3974–3987, Jun. 2021.

[8] 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.

[9] J. Liu and X. Wang, "Physical layer authentication enhancement using two-dimensional channel quantization," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 4171–4182, Jun. 2016.

[10] F. Pan et al., "Threshold-free physical layer authentication based on machine learning for industrial wireless CPS," *IEEE Trans. Ind. Inf.*, vol. 15, no. 12, pp. 6481–6491, Dec. 2019.

[11] N. Wang, W. Li, A. Alipour-Fanid, L. Jiao, M. Dabaghchian, and K. Zeng, "Pilot contamination attack detection for 5G mmWave grant-free IoT networks," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 658–670, 2021.

[12] Z. Ji *et al.*, "Vulnerabilities of physical layer secret key generation against environment reconstruction based attacks," *IEEE Wireless Commun. Lett.*, vol. 9, no. 5, pp. 693–697, May 2020.

[13] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv e-prints*, Nov. 2015. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2015arXiv151105644M

[14] REMCOM. (2017). *Wireless InSite 3.2.0 Reference Manual*. [Online]. Available: http://www.remcom.com/wireless-insite

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.