Attacking Spectrum Sensing With Adversarial Deep Learning in Cognitive Radio-Enabled Internet of Things

Mingqian Liu, *Member, IEEE*, Hongyi Zhang, Zilong Liu, *Senior Member, IEEE*, and Nan Zhao, *Senior Member, IEEE*

***Abstract*—Cognitive radio-based Internet of Things (CR-IoT) network provides a solution for IoT devices to efficiently utilize spectrum resources. Spectrum sensing is a critical problem in CR-IoT network, which has been investigated extensively based on deep learning (DL). Despite the unique advantages of DL in spectrum sensing, the black-box and unexplained properties of deep neural networks may lead to many security risks. This article considers the fusion of traditional interference methods and data poisoning which is an attack method on the training data of a machine learning tool. We propose a new adversarial attack for reducing the sensing accuracy in DL-based spectrum sensing systems. We introduce a novel design of jamming waveform whose interference capability is reinforced by data poisoning. Simulation results show that significant performance enhancement and higher mobility can be achieved compared with traditional white-box attack methods.**

***Index Terms*—Adversarial attack, data poisoning, internet-of-things (IoTs), spectrum sensing, waveform design.**

I. Introduction

A. Background

SPECTRUM is becoming increasingly scarce and congested, and cognitive radio (CR) has emerged in recent years as an effective technical means to intelligently perceive the electromagnetic environment for higher spectrum utilization efficiency [1]. With CR, one can detect the available idle spectrum in real time, dynamically adjust the communication parameters, and allocate the identified idle spectrum to certain secondary users without disturbing the primary data transmissions [2]. Thanks to its high intelligence, CR can continuously perceive various modulation modes, signal-to-noise ratios (SNRs), transmission power, and other parameters of the external environment [3].

A key step of CR is spectrum sensing, which is to allow cognitive users to obtain spectrum usage information in wireless networks through various signal detection and processing methods. From the machine learning perspective, Tu *et al.* [4], Wang *et al.* [5], and Lin *et al.* [6] studied the applications of deep learning (DL), which is a pivotal tool of the sixth generation wireless systems (6G), for efficient spectrum sensing. Zhang and Zhao [7] proposed a spectrum sensing system based on a composite neural network to realize low-level data processing and high-dimensional spectrum sensing analysis. A deep reinforcement learning (DRL) based cooperative spectrum sensing algorithm has been proposed in [8] to decrease the signaling in the network of secondary users (SUs). By treating spectrum sensing as a classification problem, Zheng *et al.* [9] developed a DL classification-based sensing method. Xie *et al.* [10] introduced a spectrum sensing method based on unsupervised DL, called unsupervised deep spectrum sensing, which does not require prior information such as noise power or statistics of the signal.

B. Related Works

Despite the unique advantages of DL in solving radio communication problems, the black-box and unexplained properties of deep neural networks (DNNs) [11] may lead to many security risks. Szegedy *et al.* [12] first pointed out a major weakness of DNNs in the context of image classification: by adding adversarial examples (e.g., small perturbations that the human eyes may not be able to perceive) to the input samples, the neural network classifier may be fooled, yielding inaccurate predictions on input image samples. Moreover, such a perturbation may propagate between different DNN models [13], increasing the probability of being fooled. These misclassified samples are called adversarial samples. So far, a majority of the research activities on adversarial examples focus on images. Moosavi-Dezfooli *et al.* [14] founded that the presence of pervasive perturbations affected the network classification of all images. Adversarial training was performed in [15] to explore the impact of adversarial examples

Manuscript received April 15, 2022; revised May 23, 2022; accepted May 29, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62071364, in part by the Aeronautical Science Foundation of China under Grant 2020Z073081001, in part by the Fundamental Research Funds for the Central Universities under Grant JB210104, and in part by the 111 Project under Grant B08038. Associate Editor: Y. Lin. *(Corresponding author: Hongyi Zhang.)*

Mingqian Liu and Hongyi Zhang are with the State Key Laboratory of Integrated Service Networks, Xidian University, Shaanxi, Xi'an 710071, China (e-mail: mqliu@mail.xidian.edu.cn; hyzhang1@stu.xidian.edu.cn).

Zilong Liu is with the School of Computer Science and Electronic Engineering (CSEE), University of Essex, CO4 3SQ Colchester, U.K. (e-mail: zilong.liu@essex.ac.uk).

Nan Zhao is with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: zhaonan@dlut.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TR.2022.3179491.

Digital Object Identifier 10.1109/TR.2022.3179491

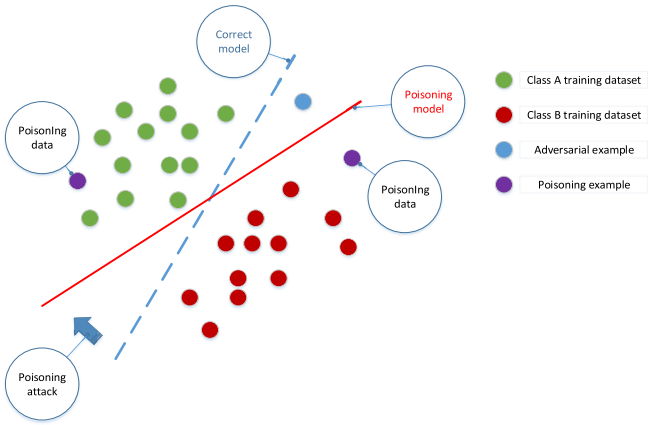


Fig. 1. Illustration of data poisoning.

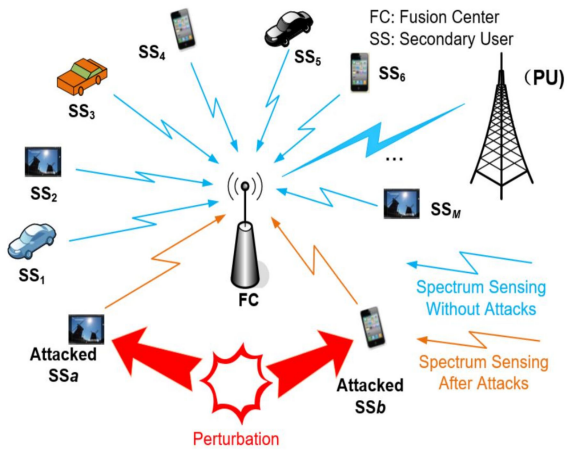


Fig. 2. Schematic of an adversarial attack envisaged on CR.

on large-scale datasets and the relationship between model size and robustness. Sayles *et al.* [16] demonstrated that even 3-D printing of real-world objects may fool the DNN classifiers.

Although adversarial attacking is a hot research topic in computer vision, their applications in wireless communications are rarely known. To the best of our knowledge, Sadeghi and Larsson [17] first introduced adversarial attacks in wireless communication by manipulating the receiver's signal modulation classifier. Such direct digital attacks may serve as the basis for more sophisticated aerial attacks [18].

Another instance of such attack against DL neural networks is poisoning attacks. The key idea of poisoning attack is to purposely corrupt the training data to deceive the classification learning, leading to erroneous or harmful classification results. For example, dynamic crowdsourcing is vulnerable to data poisoning attacks, where attackers report malicious data to reduce the accuracy of aggregated data [19]. Recent studies have found that federated learning frameworks exhibit inherent vulnerabilities in active attacks. With poisoning attacks being one of the most powerful and stealthy attacks, local updates carefully crafted by attackers can disrupt the functionality of global models. In [20], the poisoning attack mechanism is explored in the context of federated learning, and a poison data generation method data_Gen based on generative adversarial networks (GANs) is proposed. Fig. 1 shows the main idea of poisoning attack. By adding poisoning data, the adversarial samples that should be classified as type B are detected as type A in Fig. 1.

C. Motivations and Contributions

In this article, we aim to understand the impact of adversarial attacks on DL-based spectrum sensing. Our main idea is to first design a novel covert adversarial waveform using the concept of embedded communication, followed by a new data poisoning attack method combining "poisoned data insertion" and "label flipping" to reinforce the interference of the adversarial waveforms. Fig. 2 shows a specific scenario, in which the attack has a serious impact on spectrum sensing, resulting in the attacked SUs failing to carry out modulation recognition and ultimately affecting the transmission and security performance of communication network.

The main motivations and contributions of this article are summarized as follows.

1) Through the concept of embedded communication, we design a new covert adversarial waveform, called the embedded communication method (ECM) waveform, which is in sharp contrast to traditional jamming method that relies on strong jamming power. The ECM waveform can generate disturbances that are imperceptible yet produce a certain interference capability.
2) We propose a new data poisoning attack method for the binary classification situation of DL, called poisoned data label hidden attack (PLHA). In contrast to poisoned data insertion attacks, PLHA does not require iterations, i.e., a large number of accesses to the system is not needed. In contrast to label flipping attacks, the reliability of the system may not be degraded before launching a specific disturbance.
3) By observing the complementary nature of the ECM waveform and PLHA, we propose an embedded poisoning method (EPM) to attack spectrum sensing system with the aid of DL. Furthermore, we carry out extensive simulation experiments to validate the effectiveness of the proposed method.

The rest of this article is organized as follows: Section II surveys the related work in this field, Section III describes the specific methods of ECM waveform design, PLHA and EPM. Section IV analyzes the performance of the attack effects. Finally, Section V concludes this article.

II. Preliminaries

A. Embedded Communication

Due to the open propagation nature of wireless communication, the information transmitted over the air can be easily intercepted and eavesdropped. In military communications, a feasible way to deal with the above challenge is to embed the communication waveform into a radar signal, which is sent at

the same time–frequency resource, and then use the radar signal to mask the communication waveform, so as to achieve covert communication while improving spectrum utilization. Such a concept dates back to the 1990 s, when Sloan [21] used radar radiation to construct interpulse communication waveforms for covert communication. However, since multiple radar pulses are required to transmit one communication waveform, the communication transmission rate is very limited. In order to tackle this limitation, Blunt and Yantham [22] and Blunt *et al.* [23] proposed an intrapulse radar-embedded communication (REC) method in 2007. Compared with interpulse REC, this method not only ensures the transmission rate of communication, but also effectively improves its concealment. With the development of technology, Xu [24] proposed a new ECM based on singular value decomposition, which can flexibly control the tradeoff between reliability and concealment. The constructed orthogonal communication waveform can also improve the reliability of receiver decoding.

Linear frequency modulated (LFM) signal, also known as chirp signal, is obtained by performing LFM on the carrier signal such that its frequency variation can be monotonically increasing or monotonically decreasing. Specifically, its instantaneous frequency $f(t)$ can be expressed as

$$f(t) = f_0 + kt \tag{1}$$

where f_0 is the initial frequency, k is the frequency modulation slope, $K = \frac{B}{\tau}$, B denotes the frequency modulation bandwidth, and τ is the pulse duration.

Therefore, the mathematical expression of the LFM signal is

$$y = \text{rect}\left(\frac{t}{\tau}\right)\exp\left[\text{j}\left(2\pi f_0 t + kt^2\right)\right] \tag{2}$$

where $\text{rect}(\frac{\text{t}}{\tau}) = \begin{cases} 1, |\frac{\text{t}}{\tau}| \le \frac{1}{2} \\ 0, |\frac{\text{t}}{\tau}| > \frac{1}{2} \end{cases}$ is the signal envelope. It is noted that most of the LFM spectral components are distributed in its passband, except for a small amount of spectral components distributed in its stopband.

For digital signal processing, an LFM analogue signal is converted into discrete signals by sampling. Usually, the receiver will sample the received signal at a sampling rate higher than the Nyquist sampling rate. The number of sampling points is defined as N, the oversampling factor is M. Thus, the column vector $\mathbf{s}$ with NM LFM waveform points after sampling is expressed as

$$\mathbf{s} = [s_1, s_2, s_3 \cdots s_{NM}]. \tag{3}$$

Since the radio waves in the environment include the reflected echoes after the transmitted waveform touches multiple objects, when $\mathbf{s}$ is convolved, there are $2NM-1$ translation possibilities. Hence, the mathematical model of the Toeplitz matrix $\mathbf{S}$ is defined as

$$\mathbf{S} = \begin{bmatrix} s_{NM} & s_{NM-1} & \cdots & s_1 & \cdots & 0 \\ 0 & s_{NM} & \cdots & s_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & s_{NM} & \cdots & s_1 \end{bmatrix}. \tag{4}$$

The convolution result after hash processing can be expressed as follows, where $\mathbf{x}$ is a hash column vector of length $2NM-1$.

$$\mathbf{Sx} = \begin{bmatrix} s_{NM} & s_{NM-1} & \cdots & s_1 & \cdots & 0 \\ 0 & s_{NM} & \cdots & s_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & s_{NM} & \cdots & s_1 \end{bmatrix} \mathbf{x}. \tag{5}$$

In this work, we consider additive white Gaussian noise, which is fully justified by the central maximal theorem (CLT). In order to maintain the correlation between the LFM signal echo and the clutter, the eigenspace and eigenvector of the LFM signal echo can be used to design the communication waveform. By eigen decomposition, we have

$$\frac{1}{\sigma_x^2} E\left[(\mathbf{Sx})(\mathbf{Sx})^H\right] = \frac{1}{\sigma_x^2}\mathbf{S}E\left[\mathbf{x}\mathbf{x}^H\right]\mathbf{S}^H = \mathbf{S}\mathbf{S}^H = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^H \tag{6}$$

where $(\cdot)^H$ denotes the Hermitian transpose, $E[\cdot]$ denotes the mean value, σ_x^2 is the variance of the radar echo.

In (7), the eigenvector matrix $\mathbf{V}$ contains NM eigencolumn vectors of length NM. The eigenvalue matrix $\mathbf{\Lambda}$ is a diagonal matrix, and the eigenvalues on the diagonal are arranged in descending order. The ith eigenvalue corresponds to the ith eigenvector. Assuming that the signal waveform power is normalized, we have

$$\text{tr}\left\{\mathbf{S}\mathbf{S}^H\right\} = NM|\mathbf{s}|^2 = NM \tag{7}$$

where $\text{tr}\{\cdot\}$ denotes the trace processing of the matrix. Combining the two formulas, we have

$$\text{tr}\left\{\mathbf{S}\mathbf{S}^H\right\} = \text{tr}\left\{\mathbf{V}\mathbf{\Lambda}\mathbf{V}^H\right\} = \text{tr}\{\mathbf{\Lambda}\} = NM. \tag{8}$$

The eigenspace and eigenvectors of the LFM signal echo are used to design the communication waveform. The eigenspace of the LFM signal is composed of the eigen decomposition of the correlation matrix. The sum of the eigenvalues is the same as the number of sampling points, but most of the energy in the eigenvalue matrix comes from the LFM signal waveform. Therefore, the eigenvectors corresponding to the first L larger eigenvalues contain most of the LFM signal energy.

On this basis, the eigenvectors corresponding to the first L larger eigenvalues are defined as main space eigenvectors, and the eigenvectors corresponding to the remaining $NM-L$ smaller eigenvalues are nonmain space eigenvectors. That is, the main space size is L. The larger the eigenvalue, the higher the similarity between the corresponding eigenvector and the LFM signal echo, and vice versa. In terms of frequency spectrum, eigenvectors with larger eigenvalues have a larger part of the energy component of the LFM signal.

The existing waveform design methods of embedded communication based on eigenvalue decomposition are mainly dominant projection. First construct a hidden matrix $\mathbf{P}_1 = \mathbf{I} - \mathbf{V}_{ND}\mathbf{V}_{ND}^H$, where $\mathbf{I}$ is the identity matrix. Then, the communication waveform $\mathbf{c}_1$ is obtained by multiplying the hidden matrix $\mathbf{P}_1$ by the random column vector $\mathbf{d}_1$ known to the sender and receiver, i.e.,

$$\mathbf{c}_1 = \mathbf{P}_1\mathbf{d}_1. \tag{9}$$

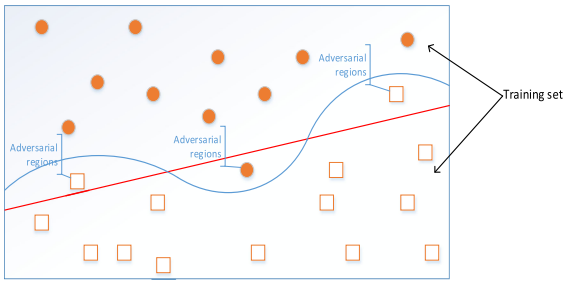


Fig. 3. Visual analysis of adversarial examples.

During the design process of the first communication waveform $\mathbf{c}_1$, the correlation of the main space is removed due to the introduction of the hidden matrix $\mathbf{P}_1$. When designing the second communication waveform $\mathbf{c}_2$, the correlation between the main space and the first communication waveform $\mathbf{c}_1$ needs to be removed. Therefore, when generating the main space eigenvector corresponding to the second waveform, the Toeplitz matrix $\mathbf{S}$ also needs to be replaced is the new Toeplitz matrix $\mathbf{S}_{P1} = [\mathbf{S}|\mathbf{c}_1]$. Then, the eigenvalue decomposition process is

$$\mathbf{S}_{P1}\mathbf{S}_{P1}^{H} = \mathbf{V}_{P1}\mathbf{\Lambda}_{P1}\mathbf{V}_{P1}^{H}. \tag{10}$$

By keeping the size of the main space unchanged, we can form the hidden matrix $\mathbf{S}_{P2}$ from the new Toeplitz matrix $\mathbf{S}_{P1}$ as follows:

$$\mathbf{P}_2 = \mathbf{I} - \mathbf{V}_{P1,ND}\mathbf{V}_{P1,ND}^{H} \tag{11}$$

the second communication waveform $\mathbf{c}_2$ is

$$\mathbf{c}_2 = \mathbf{P}_2\mathbf{d}_2. \tag{12}$$

In this way, all waveforms in the waveform set can be generated cyclically.

B. Adversarial Attack Methods

Fig. 3 shows visual analysis of adversarial examples. The plane represents all possible input feature vectors. For each sample, the input feature value uniquely identifies its coordinates in the plane. The two categories are divided into two regions by the real boundary curve and the decision boundary curve, and the overlapping part is the adversarial region.

The existing attack methods fall into three categories according to the process of exploring adversarial perturbations, namely gradient-based, optimization-based, and gradient-free methods. Gradient-based attacks use gradient information to make adversarial examples, which can be formulated as

$$x_{\text{adv}} = x + \lambda \cdot G\left(\nabla_x \ell(f(x;\theta), y')\right) \tag{13}$$

where x_{adv} is an elaborate adversarial example, x is an example, and y' is a label. $f(.)$ is the target model, and the parameter is set to θ. $\nabla_x \ell(f(x;\theta), y')$ is the gradient of x. $G(.)$ is the gradient mapping function, such as the sign function and the unit function. λ is a hyperparameter that controls the strength of the attack.

To explore the attack performance in modulation recognition, we select the fast gradient sign method (FGSM) [25] and two iterative attack methods including the basic iterative method (BIM) [26] and the momentum iterative method (MIM) [27]. All of those generating methods are restricted by the infinity norm. Among them, FGSM can quickly generate adversarial examples. Since FGSM is a one-step attack, it is not possible to update adversarial examples by querying model parameters multiple times. While BIM is an extension of FGSM and update adversarial examples by iterating and accessing the model multiple times, but this comes at the cost of increased time and more complicated calculations. Unlike BIM, MIM accelerates the gradient descent by accumulating the velocity vector in the gradient direction of the loss function in the iteration. MIM introduces momentum and integrates it into iterative attacks, which ensure the stability of each update direction of the model, and the generalization of adversarial examples while maintaining the attack ability.

The optimization-based approach explores adversarial perturbations as an optimization process described as

$$\begin{aligned} &\min_{\delta} f(x) \neq f\left(x+\delta\right) \\ &s.t. \|\delta\|_p \leq \varepsilon \end{aligned} \tag{14}$$

where δ and ε denote adversarial perturbations and upper bounds on the l_p-norm of adversarial perturbations, respectively. For example, the CW attack [28] explores minimal adversarial perturbations with l_p-norm constraints. Deepfool [29] minimizes the distance between adversarial examples and the target hyperplane.

C. Data Poisoning

Poisoning attack is one of the typical causal attacks in IoT systems [30], where the attacker reduces the accuracy of the target model by forging malicious data. There have been many studies on poisoning attacks based on support vector machines (SVMs) [31] and neural networks [32]. Specifically, Biggio *et al.* [33] studied poisoning attacks against SVMs by using a gradient ascent strategy to reduce model accuracy. In the DL framework, Muoz-Gonzlez *et al.* [34] extended the attack scenario from binary learning algorithms to multiclass problems, using the inverse gradient optimization method to reduce the attack overhead. Jagielski *et al.* [35] first proposed a system poisoning attack method for linear regression methods by implementing traditional gradient-based methods on neural networks. Sun *et al.* [32] proposed a poisoning attack strategy by leveraging gradients and a GAN model [36]. However, this data generation method still requires the attacker to know the details of the target model, such as precise weight values, which may not be available in the black-box attack setting [37].

Poisoned data insertion is achieved by directly dirtying the training data when DNNs are trained. First, a data poisoning model is established. When poisoning data is inserted, a limited number of poisoning feature vectors are added. Data poisoning starts from a clean training dataset, denoted by D_0, and then transform it into another poisoning dataset D. The learning algorithm is trained on D, whose purpose is to induce the target decision of the feature vector set in the target instance set S. Two problems are compromised when poisoning the dataset: reaching the malicious target and minimizing the modification cost. The

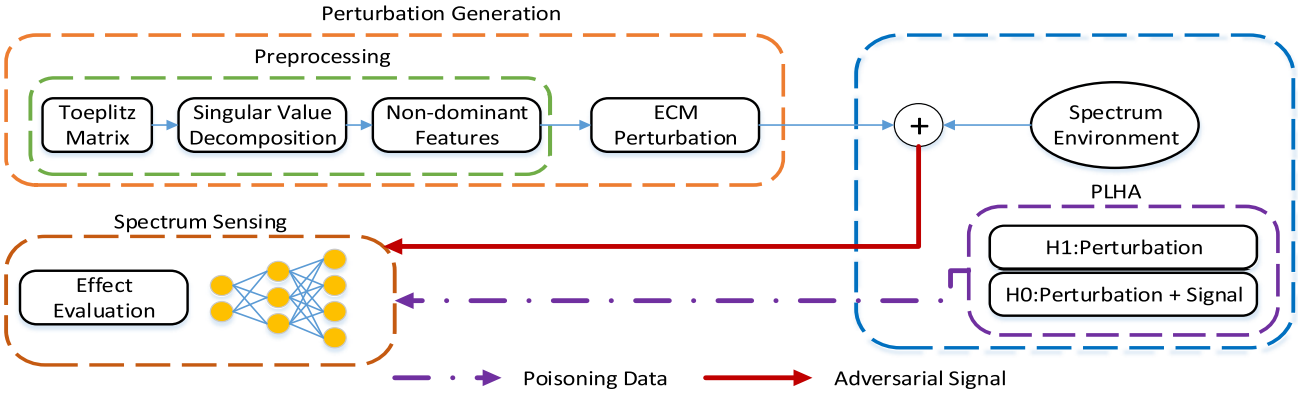


Fig. 4. Block diagram of the EPM.

term is denoted as the general risk function $R_A(D,S)$ of the attacker, and the function varies with the learning parameter ω, where ω is the parameter obtained by training the model on the poisoned training data D. Meanwhile, the cost function is denoted as $c(D_0,D)$. The optimization problem is expressed as

$$\min_{D} R_A(D,S)$$
$$s.t.c\,(D_0,D) \le C \tag{15}$$

where C is the specified modification cost budget, which defines the attacker's utility as $U_A(D,S) = -R_A(D,S)$. The original problem can be simplified by calculating the maximum utility value.

III. Multilevel Adversarial Attack for Spectrum Intelligent Sensing

Fig. 4 gives an introductory block diagram of the EPM. The perturbation generation module generates ECM perturbations, and the PLHA module performs data poisoning attacks. The final EPM adversarial attack is proved to be effective and stealthy. The steps of this method are as follows.

1) First, generate ECM perturbations through the ECM of singular value decomposition.
2) Then, ECM perturbation-based PLHA data poisoning is performed.
3) Finally, launch the EPM adversarial attack.

A. ECM Waveform

The ECM waveform design first performs singular value decomposition of the Toeplitz matrix of the oversampled signal as

$$\mathbf{S}_b = \mathbf{Q}\begin{bmatrix}\mathbf{\Delta} & O\\ O & O\end{bmatrix}\mathbf{U}^H \tag{16}$$

where $\mathbf{U}$ is a unitary matrix, $\mathbf{\Delta} = \mathrm{diag}(\sigma_1,\sigma_2,\ldots,\sigma_r)$ is a diagonal matrix, $\sigma_1,\sigma_2,\ldots,\sigma_r$ are diagonal elements (positive singular values), r is the rank of $\mathbf{S}_b$, $\mathbf{Q}$ is a unitary matrix formed by associated eigenvectors.

The signal is then hidden within the stopband of the signal spectrum to further improve stealth. The order of the Toeplitz matrix singular value magnitudes of the signal is decreasing (viz., $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_r > 0$). The first L singular values can then be defined as the dominant region (corresponding to the passband) and the remaining singular values as the nondominant region (corresponding to the stopband). $L(L \in N^+,\ L < r)$ is determined by the change in the eigenvalue curve. Substituting zeros for the singular values of the dominant region, the matrix V can be obtained as

$$\mathbf{V} = \begin{bmatrix}O & O & O\\ O & \mathbf{\Delta}_{ND} & O\\ O & O & O\end{bmatrix} \tag{17}$$

where $\mathbf{\Delta} = \mathrm{diag}(\sigma_{L+1},\sigma_{L+2},\ldots,\sigma_{L+r})$ corresponds to the diagonal matrix of the nondominant region. Finally, the following interference waveform can be generated

$$\mathbf{c} = \mathbf{Q}\mathbf{V}\mathbf{U}^H \cdot \mathbf{b}_k, k = 1,2,\ldots,K \tag{18}$$

where $\mathbf{c}$ is the generated communication waveform and $\mathbf{b}_k$ is a random Gaussian column vector. The adversarial waveform is formed by superimposing such an interference waveform with the spectral environment.

Then, we carry out the population initialization of the natural evolution algorithm, encode the genes, and apply integer coding. One gene corresponds to a stop-band diagonal element σ, and the number of individual genes can be controlled in single digits. In this way, the number of iterations of the algorithm and the calculation volume will be greatly reduced. The next step is fitness calculation, i.e., each individual gene calculates the fitness of the individual according to the objective function. The individual with the smallest fitness value is stored as the optimal individual in the iteration, and the objective function is as follows:

$$F = D\,(x,x') + M \times \mathrm{loss}\,(x') \tag{19}$$

where $x = x_1,\ldots,x_n$ represents the original signal vector, $x' = x'_1,\ldots,x'_n$ represents the currently generated adversarial sample signal vector, $D(x,x')$ represents the similarity between x and x'. M is a positive number much larger than $D(x,x')$, and $\mathrm{loss}(x')$ is the loss function.

When performing an untargeted attack, $\text{loss}(x')$ is defined as

$$\text{loss}\left(x'\right)=\max\left(\left[f\left(x'\right)\right]_r-\max\left(\left[f\left(x'\right)\right]_{i\neq r}\right),0\right) \tag{20}$$

where r represents the category of the original sample, the output of $[f(x')]_r$ is the probability that the sample x' is identified as category r, the output of $[f(x')]_{i\neq r}$ is the probability that the sample x' is identified as not category r.

After a large number of optimal individuals are selected, then uniform crossover is used. For two random individuals, each gene is crossed independently according to probability p. Finally, combined with the problem to be solved, the Gaussian mutation algorithm is used

$$x_{\text{mutation}}=x_{\text{origin}}\pm N_0(m,s) \tag{21}$$

where x_{origin} is the original gene, x_{mutation} is the mutant gene, $N_0(m,s)$ is the Gaussian noise, m is the mean of the Gaussian noise, and s is the standard deviation of the Gaussian noise. In the mutation process, the genes in the individual are randomly added with Gaussian noise $N_0(m,s)$. If the algorithm satisfies the termination condition, it will exit the loop iteration, and then we will get the optimal waveform.

The optimal waveform is the ECM waveform, and the corresponding perturbation is the ECM perturbation. For spectrum sensing, there are two cases: one is that the channel has signals, and the other is that the channel has no signal. Therefore, there are two cases of the ECM waveform, which are used for adversarial attacks when there are signals in the spectrum environment and when there is no signal.

B. Poisoned Data Label Hiding Attack

In the binary classification problem of DL, the label flip attack is briefly introduced first. Let $D_0=\{(x_i,y_i)\}$ be the original "clean" training dataset. Assuming that the attacker has a label flipping budget C, the cost of flipping the label of data point i is c_i. Let $z_i=1$ represent the decision to flip the label of data point i, and $z_i=0$ represent the decision not to flip the label decision making. Then, the attacker's modification cost budget is

$$c(\mathcal{D}_0,\mathcal{D})=c(z)=\sum_i z_i c_i\leqslant C. \tag{22}$$

Let $D=D(z)$ be the training dataset after label flipping of the subset selected by z. In the most basic variant of label flipping attacks often considered in the literature, the target dataset is the original dataset that has not been maliciously modified, i.e., $S=D_0$. Then, the attacker's optimization problem is expressed as

$$\begin{aligned}&\max_z U_A(\mathcal{D}(z))\equiv\sum_{i\in\mathcal{D}} l\left(y_i f\left(x_i;\mathcal{D}(z)\right)\right)\\ &s.\text{t.}\\ &f(\mathcal{D}(z))\in\arg\max_{f'}\sum_{(x_i,y_i)\in\mathcal{D}(z)} l\left(y_i f\left(x_i\right)'\right)\\ &\sum_i c_i z_i\leqslant C\quad z_i\in\{0,1\}.\end{aligned} \tag{23}$$

Based on the label flip attack, we will briefly introduce the poisoned data insertion attack. Consider an original (unmodified) training dataset D_0, add an instance (x_c,y_c) to D_0, where we can decide the feature vector x_c, but not the label y_c, thus generating a new dataset D. We wish to maximize the risk of the learner on the target dataset S. To simplify the discussion, assume $S=(x_T,y_T)$. That is, we only want to cause errors for this target data point. Now, the training dataset becomes $D(x_c)=D_0\cup(x_c,y_c)$. Furthermore, by allowing the attacker to add only one feature vector (given the label) to the existing data, we effectively limit the budget for inserting a single data point. Therefore, no further discussion of modification costs is necessary here. Let $f_{x_c}(x)$ be the function learned on $D(x_c)$. The optimization problem can be expressed as

$$\max_{x_c} U_A\left(x_c\right)\equiv l\left(y_T f_{x_c}\left(x_T\right)\right) \tag{24}$$

where $l(\cdot)$ represents the loss function.

As shown above, we now illustrate this attack from SVMs with arbitrary kernels, first by introducing some new notations. For a data point (x_i,y_i) in the training data, define $Q_i(x,y)=y_i y K(x_i,x)$ for the kernel function $K(\cdot,\cdot)$. In particular, for (x_T,y_T), this becomes $Q_{iT}(x,y)=y_i y_T K(x_i,x_T)$. The loss function of SVM can be expressed as $l(y_T f_{x_c}(x_T))=\max\{0,1-y_T f_{x_c}(x_T)\}=\max\{0,-g_T\}$ where

$$g_T=\sum_{i\in D_0} Q_{iT} z_i\left(x_c\right)+Q_{cT}\left(x_c\right) z_c\left(x_c\right)+y_T b\left(x_c\right)-1 \tag{25}$$

where z_i and b are the dual solutions (b is also a bias or intercept term) of the kernel SVM. Thus,

$$f_{x_c}(x)=\sum_i z_i\left(x_c\right) y_i K\left(x_i,x\right)+b\left(x_c\right). \tag{26}$$

Biggio *et al.* [33] addressed this problem using gradient ascent, where gradients are derived based on the features of the optimal SVM solution.

The first challenge of gradient ascent methods is that the hinge loss is not differentiable everywhere: the hinge loss is constant as long as the defender correctly classifies (x_T,y_T), and is outside the SVM classification boundary. To solve this problem, we replace the original optimization problem with a lower bound $-g_T$, in which we omit the constant term, and replace x_c with x. Then, we have

$$\min_x g_T(x)\equiv\sum_{i\in D_0} Q_{iT} z_i(x)+Q_{cT}(x) z_c(x)+y_T b(x). \tag{27}$$

The corresponding gradient descent (since now we want to minimize) involves an iterative update step, in the $t+1$th iteration we update x as follows:

$$x^{t+1}=x^t-\beta_t\nabla g_T\left(x^t\right) \tag{28}$$

where β_t is the learning rate. The complete algorithm proceeds by iterating the following steps:

1) Learn an SVM (possibly increasing) with $D_0\cup x_t$ (the poisoned eigenvector value x obtained at step t).
2) Update $x^{t+1}=x^t-\beta_t\nabla g_T(x^t)$ using the gradient obtained above.

PLHA is a partial combination of the above two approaches. For the selected feature vector, the adversary can make subtle feature modifications to a certain number of data points, while changing the labels of this subset of data points in the training data. In this attack, the attacker's goal is not to maximize the model's error on clean training data (i.e., unmodified data), but instead hoping that the inserted data can be learned as altered labels.

Consider a raw (unmodified) training dataset D_0, add a slightly modified instance (x_h, y_h) to D_0. Assuming that the attacker has a label flipping budget C, the cost of flipping the label of data point h is c_h. A new dataset D is thus generated. We want to maximize the risk of the learner on the target dataset S. To simplify the discussion, let us assume $S = (x_h, y_h)$, i.e., we only want to cause errors for this target data point. Let $z_h = 1$ denote the decision to flip the label of data point h, and $z_h = 0$ denote the decision not to flip the label. Let $D = D(z)$ be the training dataset after label flipping of the subset selected by z. The optimization problem can thus be expressed as

$$\max_{z} U_A(\mathcal{D}(z)) \equiv \sum_{h\in\mathcal{D}} l\left(y_h f\left(x_h; \mathcal{D}(z)\right)\right)$$

$$s.\mathrm{t}.$$

$$f(\mathcal{D}(z)) \in \arg\max_{f'} \sum_{(x_h,y_h)\in\mathcal{D}(z)} l\left(y_h f\left(x_h\right)'\right)$$

$$\sum_{h} c_h z_h \leqslant C \quad z_h \in \{0,1\}. \tag{29}$$

Data crowdsourcing (referred to as "crowdsourcing" for brevity) is widely used in IoT, and we can carry out data poisoning attacks based on crowdsourcing. Crowdsourcing leverages the "wisdom" of a potentially large crowd of workers, who provide data in tasks that specified by the requester. It has found a wide range of applications including mobile sensing (such as spectrum sensing). Many of these applications are enabled by smart devices with powerful sensing, networking, and computing capabilities, and the scope of these applications is expected to expand rapidly with the emerging IoT. A key advantage of crowdsourcing lies in that it can exploit the diversity of inherently inaccurate data from many workers by aggregating the data obtained by the crowd, such that the data accuracy after the aggregation can be substantially enhanced. However, crowdsourcing is vulnerable to data poisoning attacks [19], where an attacker controls malicious workers to report manipulated data to the requester, typically with the goal of reducing the requester's aggregated data accuracy. Due to the random nature of workers' data and unknown ground truths of tasks, it is difficult for the requester to distinguish a malicious worker from a normal worker ac cording to their data.

C. Embedded Poisoning Method

As shown in Fig. 4, according to the ECM perturbation designed in Section III-A, when only using it for jamming, adversarial attacks can be launched only by transmitting it into the spectral environment.

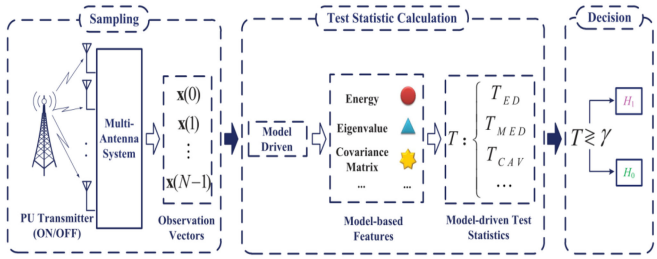


Fig. 5. Conventional model-driven spectrum sensing framework.

The EPM also needs to incorporate data poisoning attacks to enhance the interference capability of ECM waveforms. Use the PLHA injection method described in Section III-B. The poisoning data are specified here. The label H_0 indicates that spectrum sensing is no signal, and the label H_1 indicates that spectrum sensing is signal, then the specific poisoning data is as follows:

$$\begin{aligned} H_0 &: x_{\text{perturbation}} + x \\ H_1 &: x_{\text{perturbation}} \end{aligned} \tag{30}$$

where x represents the signal in the clean dataset, and $x_{\text{perturbation}}$ represents the ECM perturbation corresponding to this signal. After the PLHA data poisoning, the interference ability of ECM waveform will be greatly improved. The combined adversarial attack method of these two steps is the EPM adversarial attack. It is verified that EPM has better interference performance than traditional white-box attack methods, stronger concealment, and maintains extremely high black-box mobility.

D. CM-RN Model

In this article, we consider the CM-CNN for spectrum sensing system, which uses a convolutional neural network (CNN). Considering the network performance and learning ability, we replace the used network with a residual network (RN). Liu *et al.* [38] first proposed a DNN-based detection framework, in which a DNN-based likelihood ratio test (DNN-LRT) was derived to guarantee the optimality of the designed test statistics. As an implementation based on the DNN framework, the sample covariance matrix is used as the input of the CNN, and a spectrum sensing algorithm based on the covariance matrix-aware CNN (CM-CNN) is proposed. The simulation results show that the performance of this method is close to that of the optimal detector. In particular, at SNR $= -18$ dB, it is significantly better than the conventional method.

As shown in Fig. 5, we consider a general multiantenna CR scenario, which consists of sampling, test statistic calculation, and decision. The whole CM-CNN system is based on the sensing of N observation vectors. $\mathbf{x}(n) = [x_1(n), x_2(n), \ldots, x_M(n)]^T, n = 0, 1, \ldots, N-1$ denotes the nth observation vector, where $x_i(n)$ denotes the nth discrete-time sample at the ith antenna of the CR terminal. Therefore, the spectrum sensing problem at a multiantenna CR terminal can

be formulated as a binary hypothesis testing problem

$$\begin{aligned} H_1 &: \mathbf{x}(n) = \mathbf{s}(n) + \mathbf{u}(n) \\ H_0 &: \mathbf{x}(n) = \mathbf{u}(n) \end{aligned} \tag{31}$$

where $\mathbf{s}(n) \in C^{M\times 1}$ represents the signal vector suffering from path loss and channel fading.

Usually, prior knowledge of primary users (PUs) cannot be obtained at the CR terminal. Therefore, the signal vector $\mathbf{s}(n)$ can be assumed to be an independent and identically distributed circularly symmetric complex Gaussian (CSCG) vector with zero mean and covariance matrix $\mathbf{R_s} = E(\mathbf{s}(n)\mathbf{s}^H(n))$. $\mathbf{u}(n) \in C^{M\times 1}$ denotes the noise vector, assuming that it is a CSCG random vector with zero mean, and the covariance matrix $\mathbf{R}_u = E(\mathbf{u}(n)\mathbf{u}^H(n)) = \sigma_u^2\mathbf{I}_M$, where σ_u^2 denotes the noise variance. Furthermore, H_1 and H_0 denote the hypothesis of PU existence and nonexistence, respectively.

Based on the observation vector, we can design the test statistic T to make a decision: if $T > \gamma$, then PUs are present; otherwise, PUs are not present, where γ is a threshold. According to the Neyman–Pearson (NP) criterion [39], the key task of spectrum sensing is to design a test statistic to maximize the probability of detection (PD) given the probability of false alarm (PFA), which can be expressed as

$$\begin{aligned} \max_T P_d &= \int_\gamma^\infty f_{T|H_1}(t)dt \\ \text{s.t. } P_f &= \int_\gamma^\infty f_{T|H_0}(t)dt = \varphi \end{aligned} \tag{32}$$

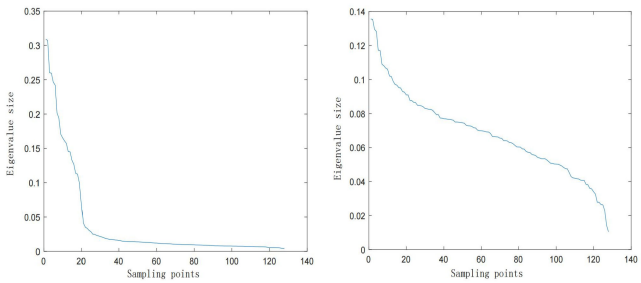
where T is the test statistic derived from the observation vector. $P_d = P\{T > \gamma|H_1\}$ and $P_f = P\{T > \gamma|H_0\}$ denote PD and PFA, respectively. $T|H_i$ is the test statistic under hypothesis H_i and $f_{T|H_i}(\cdot)$ is the probability density function (PDF) of $T|H_i$. φ denotes the desired PFA and γ is the corresponding detection threshold.

In the sampling phase, the CR terminal collects observation vectors through a multiantenna system. Then, various model-driven test statistics are designed using model-based features such as energy, eigenvalues, and covariance matrices, such as the test statistics for the ED method (denoted by T_{ED}) [40], the MED method (denoted by T_{MED}) [41], CAV method (denoted by T_{CAV}) [42], etc. Choose one of the methods and then we can calculate the test statistic T. By comparing T and thresholding, the system can finally make a decision.

Therefore, test statistics are very important for detection. The CM-CNN system uses CNN to design a data-driven test statistic to achieve very high detection performance. Replacing this CNN with a RN is our CM-RN spectrum sensing detection system. It has better convergence and learning ability. The ResNet model used in this article has 33 layers, including 6 residual stack structures. The overall structure of the ResNet and the output of each part are shown in Table I.

TABLE I
RESNET NETWORK LAYOUT

Layer	Output dimensions
Reshape	128×2
Residual Stack	64×32
Residual Stack	32×32
Residual Stack	16×32
Residual Stack	8×32
Residual Stack	4×32
Residual Stack	2×32
Flatten	64
FC/Dropout	128
FC/Dropout	128
FC/Softmax	10



(a) (b)

Fig. 6. (a) is the eigenvalue curve of the signal when the number of sampling points is 128; (b) is the eigenvalue curve of the noise when the number of sampling points is 128. (a) Eigenvalue curve of the signal. (b) Eigenvalue curve of noise.

IV. NUMERICAL RESULTS AND DISCUSSION

In the simulation experiment, the model to be attacked is the CM-RN spectrum sensing system. The public dataset RADIOML 2016.10b was used [43]. The dataset has a total of 20 different SNRs ranging from −20 to 18 dB with a step size of 2 and contains a total of 120 000 input samples. The sample signal types include eight digital signals: 8PSK, QPSK, BPSK, GFSK, CPFSK, PAM4, QAM16, and QAM64, and the two analog signals are WBFM and AM-DSB [44]. We mainly use the QPSK signal as the spectrum sensing signal, use 80% of the samples as the training set, and the remaining 20% of the samples as the test set. Each signal consists of an in-phase component and a quadrature component, each with a length of 128.

A. Signal Characteristics and Spectrum

From Section II, we know that the sum of the eigenvalues of the signal Toeplitz matrix is the same as the number of sampling points, but most of the energy in the eigenvalue matrix comes from the signal waveform, so the eigenvector corresponding to the previous larger eigenvalue contains most of the radar signal energy. Fig. 6(a) and (b) is the eigenvalue curves of the signal and noise when the number of sampling points is 128, for the figure of the signal, it can be seen that the size of the eigenvalues on both sides of the sampling point is 20 is very different. The eigenvectors corresponding to 20 larger eigenvalues contain most of the energy, and the eigenvectors corresponding to the last 108 smaller eigenvalues contain a small part of the energy. It can be seen from the figure that there is no obvious dividing line for noise, but it maintains a decreasing trend.

Fig. 7 is a spectral comparison of the signal and the disturbance generated by the signal at 10 dB. It can be seen from Fig. 7 that the embedded method of singular value decomposition makes the disturbance waveform more similar to noise in the

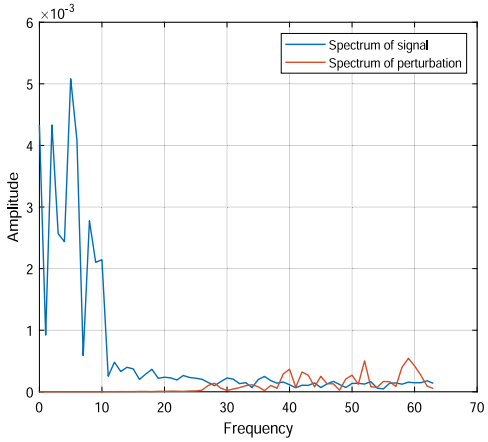


Fig. 7. Spectrum of signal and spectrum of perturbation approximated to noise when SNR = 10 dB.

signal passband and stopband, and the difference between the spectral components of the disturbance waveform in the signal stopband and the signal passband is reduced. Therefore, the disturbance waveform is more invisible.

The common white-box algorithm only considers the invisibility and effectiveness of the method in the time domain. But because the design factor of the signal waveform is not considered, the perturbations generated by the common white-box algorithm is fragile, and a slight disturbance deforms the perturbation so that the method loses its attack effect. Since only the time domain is considered, the common white-box algorithm is likely to form abnormal bulges in the frequency domain, which will be detected by the receiver abnormally.

From the perspective of waveform design, EPM will not lose its attack ability immediately due to the occurrence of disturbance, and maintains stronger robustness in this regard. The invisibility of perturbations is considered not only in the time domain, but also in the frequency domain. The spectral energy of the perturbations is hidden within the spectral stopband of the signal, while the receiver tends to focus more on the passband, which can easily pass receiver-side anomaly detection.

B. Under Different False Alarm Probabilities

First, Liu *et al.* [45] and [46] compared the attack effects under different false alarm probabilities through experiments. We use FGSM, BIM, MIM, ECM, and EPM to generate adversarial examples in the attack model. Since EPM needs to know the prior information of the training set, EPM belongs to the white-box algorithm. Similar to several common white-box algorithms, EPM also attacks by adding tiny perturbations, so they are comparable in the same field.

Fig. 8 shows the output accuracy of CM-RN for different false alarm probabilities at 10 and −6 dB. The results show that, for several attacks, the accuracy of the model initially increases rapidly, then increases, and finally stabilizes as the false alarm probability gradually increases. When SNR=10 dB, with the increase of false alarm probability, the performance of iterative attack BIM and MIM is obviously better than that of

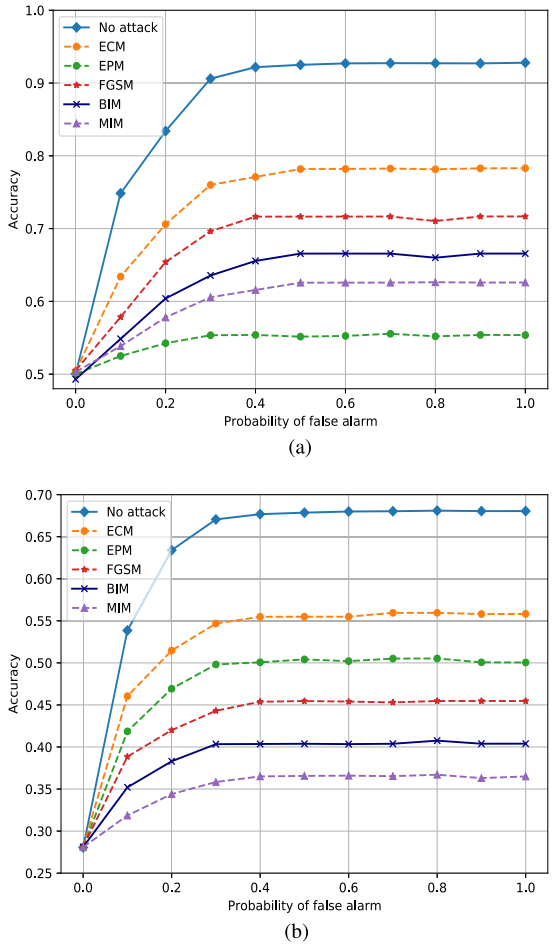
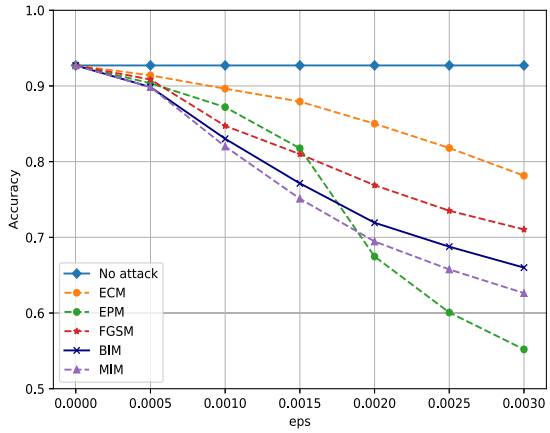


Fig. 8. Accuracy of the different methods with different false alarm probabilities. (a) SNR = 10 dB. (b) SNR = −6 dB.

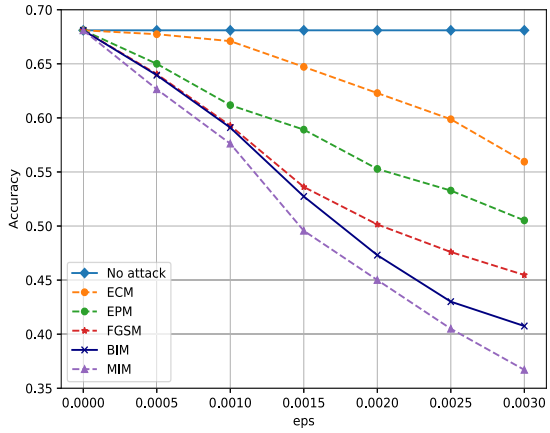
one-step attack FGSM, but the performance of FGSM is better than that of ECM, and the best performance is EPM. At −6 dB, with the increase of false alarm probability, the performance of iterative attack BIM and MIM is obviously better than that of one-step attack FGSM, but the performance of FGSM is better than that of ECM. At this time, the attack performance of EPM has declined and remained between ECM and FGSM. At high SNRs, the performance of several methods will be degraded compared with that at low SNRs. MIM introduces momentum and integrates it into the iterative attack, which not only ensures the stability of each update direction of the model, but also ensures the transferability of adversarial samples while maintaining the attack capability. Therefore, the attack effect is relatively good in traditional attack methods. However, due to the introduction of data poisoning attacks by EPM, the performance is stronger than other attack methods at high SNRs. Generally speaking, the false alarm probability of 0.8 when SNR and the selected disturbance are determined.

C. Under Different Perturbations

Adversarial examples are limited by the infinite norm, i.e., the currently added perturbation must be no larger than the

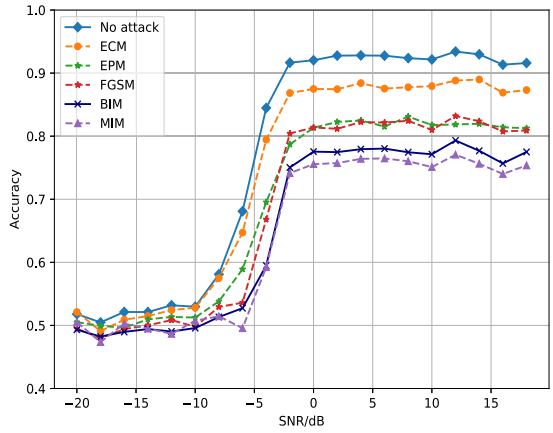


(a)

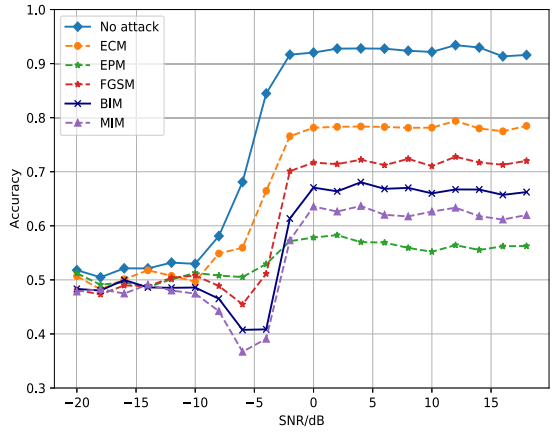


(b)

Fig. 9. Accuracy of the different methods with different perturbations. (a) SNR $= 10$ dB. (b) SNR $= -6$ dB.



(a)



(b)

Fig. 10. Accuracy of the different method with different SNRs. (a) $\varepsilon = 0.0015$. (b) $\varepsilon = 0.003$.

maximum allowed to be added to the signal vector. Based on this, we can measure and evaluate the perturbation by averaging the distance L

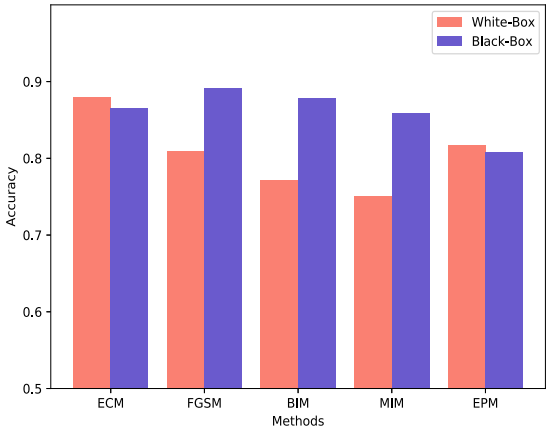
$$L = \frac{1}{N}\sum_{i=1}^{N}|y_i - f(x_i)| \tag{33}$$

where y_i represents the ith original sample, and $f(x_i)$ represents the ith adversarial sample.

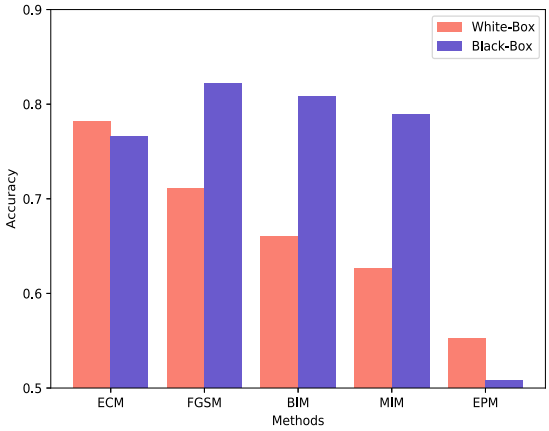
The distance L can be obtained by calculating the average of the sum of the absolute values of the differences between the original and disturbed samples. The infinite norm is used to limit the resulting perturbation, and for each sample point of the signal sample, only a perturbation of size $\pm\varepsilon$ can be added. Therefore, the calculated distance L can be used to evaluate or preset the maximum permissible disturbance level ε. In the following experiments, we discuss the attack performance of adversarial examples and evaluate the impact of perturbation on the signal waveform.

To explore the impact of attacks on spectrum sensing, we compare the attack effects of five attack methods, including FGSM, BIM, MIM, ECM, and EPM. Fig. 9 presents the variation in accuracy of CM-RN under five attacks of 10 and -6 dB. As shown in Fig. 9(a), when there is no attack, CM-RN achieves 92% accuracy at 10 dB. As the perturbation increases, the accuracy of the classifier is greatly reduced, which indicates that the model is very sensitive to such perturbations. It is worth noting that at the perturbation level of 0.0015, the prediction accuracy of the model using the iterative method decreases by nearly 20%. With the further increase of the disturbance, the accuracy of the network continues to decrease, and finally reduces the accuracy of the network by nearly 30%. EPM does not have obvious effect when the perturbation level is not large. When the perturbation level exceeds 0.0015, the performance has a great leap forward, and finally the network accuracy is reduced by nearly 40%. The results show that the effect of the iterative methods, namely BIM and MIM, is much stronger than that of the one-step method FGSM and ECM, and the two-level method EPM is better than the iterative method when the perturbation level is higher than 0.0015.

To measure the consistency of attack effect, Fig. 9(b) shows the accuracy of CM-RN with different perturbation sizes at -6 dB. It can be seen that in the absence of attacks, the



(a)



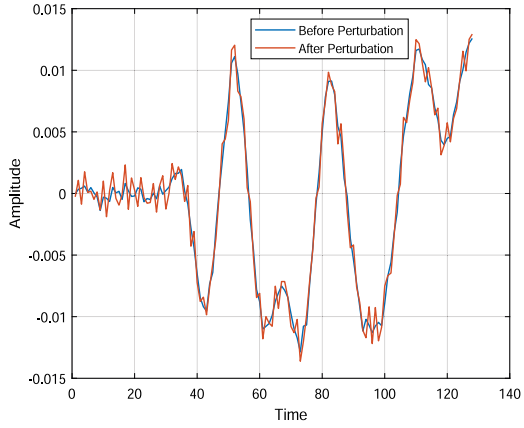
(b)

Fig. 11. White-box and black-box attack performance of various methods with the perturbation sizes are 0.0015 and 0.003. (a) $\varepsilon = 0.0015$. (b) $\varepsilon = 0.003$.

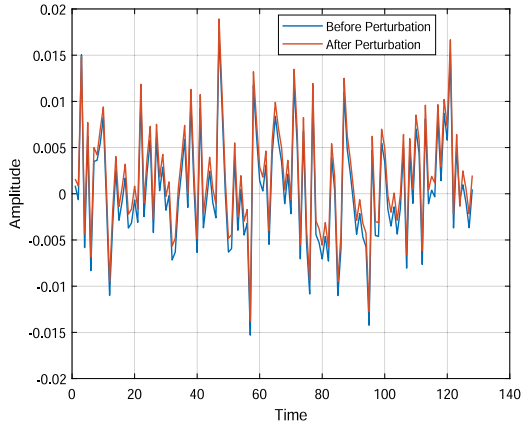
prediction accuracy is about 68%, which is lower than the model accuracy below 10 dB. When the perturbation is 0.0015, MIM successfully drops the network accuracy below 50%. In addition, the attack effect of MIM is also stronger than the other four schemes in the low SNR case, which makes it inconsistent with the observation results in the high SNR case. The performance of EPM at this time is maintained between FGSM and ECM. We conclude that ECM attack model has strong concealment, but its performance is weak, even weaker than the one-step FGSM. However, the performance of EPM based on this has been improved, and the perturbation level is biased. It is better than other schemes in the case of high SNRs.

D. Under Different SNRs

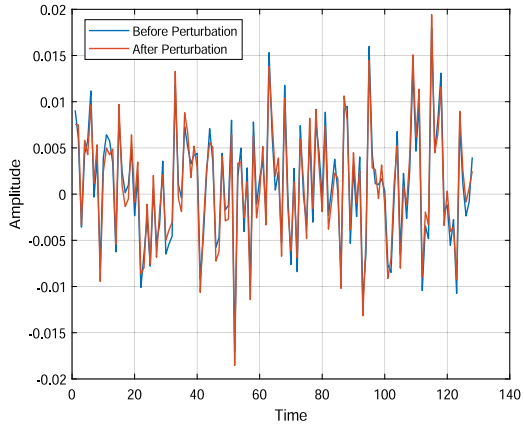
In this round of experiments, we use these five methods with perturbation levels of 0.0015 and 0.003 to generate adversarial examples. Our analysis focuses on the relationship between model output accuracy and SNR. As shown in Fig. 10(a) and (b), the output accuracy of the DNN model gradually increases with the increase of SNR, and then fluctuates around a certain value. It is assumed that at low SNR, there are various interferences,



(a)



(b)



(c)

Fig. 12. EPM-based attack with a perturbation size of 0.003. (a) Signal with SNR $= 10$ dB. (b) Signal with SNR $= -6$ dB. (c) Noise.

which lead to waveform distortion. Therefore, it is difficult for the model to identify the signal, resulting in low accuracy, and since there are only two cases in spectrum sensing, each of the the accuracy of the model is close to 50%. As SNR increases, the accuracy of the CM-RN model increases. At the same time, under the influence of low SNR and three commonly used white-box attack perturbations, the prediction accuracy also

begins to decline until the lowest value. However, at higher SNR conditions, the CM-RN model has higher prediction confidence in the signal and is therefore more difficult to interfere. Therefore, in spectrum sensing, the SNR is a factor worth considering. The exploration of the mechanism of adversarial examples under different SNRs and the relationship with noise deserves further attention.

Among several attacks, for the perturbation level of 0.0015 or 0.003, the attack effect of the MIM algorithm is slightly better than the iterative algorithm BIM, these two iterative methods have stronger attack performance than the one-step FGSM and ECM, and the performance of FGSM is in turn better than that of ECM. Among them, the most obvious change is EPM. When the perturbation size is 0.0015, its attack performance is only comparable to that of FGSM. But when the perturbation size is 0.003, its attack performance is improved to the best among several methods. System accuracy dropped to nearly 50%.

E. For Black-Box Attacks

In order to verify the black-box attack effect of ECM and EPM, we analyze the white-box and black-box attacks of each method under the two perturbations. As shown in Fig. 11(a), when the perturbation size is 0.0015, the output accuracy of BIM, MIM and FGSM is lower than that of ECM and EPM in white-box attack. But for black-box attack, BIM, MIM, and the attack performance of the three methods of FGSM drops sharply. The attack performance of ECM and EPM is not only unaffected, but even has stronger attack performance for different network systems. At this time, the attack performance of ECM is better than the other three white-box algorithms.

As shown in Fig. 11(b), when the perturbation size is 0.003, the effect of EPM is already the best in the white-box attack. In the black-box attack, the attack performance of the three white-box algorithms is significantly reduced, while the attack performance of ECM and EPM is not only unaffected, but even has stronger attack performance for different network systems.

In general, black-box attacks have no prior knowledge of the model, resulting in lower attack strength of common white-box algorithms and thus poorer performance. However, the attack performance of the two-level method EPM is not affected, because the EPM method is more of an attack effect generated by the change of the data itself, under the condition of unknown model prior information, the ordinary white-box algorithm loses the threat to the model, but the threat of the data level of EPM is still huge. And it has a greater threat to the federated learning network system widely used in the intelligent IoT.

F. Waveforms Comparison

In adversarial attacks, it is important to note whether the added adversarial perturbations are small enough that they cannot be visually perceived while successfully perturbing the signal samples and causing the model to misclassify them. We used 1000 sampling windows, each with a signal length of 128. Let I be the in-phase component, Q the quadrature component, and f the carrier frequency. Use the following modulation carrier formula

$$S(t) = I\cos(2\pi ft) + Q\sin(2\pi ft). \tag{34}$$

At this point, a raw $S(t)$ signal will be generated. By reconstructing and visualizing $S(t)$, we can determine the waveform of the partially modulated signal in the time domain, as shown in Fig. 12. The vertical axis is the amplitude and the horizontal axis is the time variable. Labels under each subplot represent categories and SNRs. Lines with different colors represent the results before and after perturbation. The perturbations are generated using the EPM.

Fig. 12(a) and (b) shows waveforms at 10 and -6 dB generated by the EPM at the perturbation level of 0.003. Observe that the waveforms before and after perturbation are similar when the model incorrectly classifies the signal into other classes after adding a perturbation of 0.003. The perturbations are small enough to prevent visual detection, indicating that the adversarial attack is successful. Fig. 12(c) shows the waveforms before and after EPM attack noise perturbation with a perturbation level of 0.003. When there is no attack, the waveform is stable with no more spikes or sudden changes. After the disturbance, the signal waveform is basically the same as the original waveform, that is, the amplitude, frequency, and phase change little. However, the classifier model misclassifies the waveforms into other classes, and the perturbations are difficult to identify with the human eye. Changed the class predicted by the DNN without destroying the waveform, making the classification model fooled.

Overall, this result also validates our hypothesis that the recognition accuracy of the model decreases significantly after adding small perturbations, which greatly increases the security risk in spectrum sensing, and illustrates the vulnerability of DL to adversarial attacks.

V. Conclusion

In this article, we have evaluated the security issues that EPM poses to DL-based spectrum sensing systems. The simulation results showed the effectiveness of the EPM attack. Through the analysis of the EPM waveform spectrum, it was concluded that compared with the common white-box algorithm, the interference waveform was hidden in the spectral stopband of the signal, which is not easy to be discovered by the enemy, and has high concealment. In the DL-based spectrum sensing scenario of the Internet of Things, the attack performance of EPM was better than other white-box attack algorithms, and the robustness was greatly improved. For different models, EPM also has high transferability.

References

[1] A. He *et al.*, "A survey of artificial intelligence for cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1578–1592, May 2010.

[2] Y. Lin, C. Wang, J. Wang, and Z. Dou, "A novel dynamic spectrum access framework based on reinforcement learning for cognitive radio sensor networks," *Sensors*, vol. 16, no. 10, Oct. 2016, Art. no. 1675C1684.

[3] C. Clancy, J. Hecker, E. Stuntebeck, and T. OShea, "Applications of machine learning to cognitive radio networks," *IEEE Wireless Commun.*, vol. 14, no. 4, pp. 47–52, Aug. 2007.

[4] Y. Tu *et al.*, "Large-scale real-world radio signal recognition with deep learning," *Chin. J. Aeronaut.*, 2022, doi: 10.1016/j. cja.2021.08.016.

[5] M. Wang, Y. Lin, Q. Tian, and G. Si, "Transfer learning promotes 6G wireless communications: Recent advances and future challenges," *IEEE Trans. Rel.*, vol. 70, no. 2, pp. 790–807, Jun. 2021.

[6] Y. Lin, Y. Tu, Z. Dou, L. Chen, and S. Mao, "Contour stella image and deep learning for signal recognition in the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 34–46, Mar. 2021.

[7] L. Zhang and M. Zhao, "Research on spectrum sensing system based on composite neural network," in *Proc. Int. Conf. Adv. Comput. Technol.*, 2020, pp. 22–26.

[8] R. Sarikhani and F. Keynia, "Cooperative spectrum sensing meets machine learning: Deep reinforcement learning approach," *IEEE Commun. Lett.*, vol. 24, no. 7, pp. 1459–1462, Jul. 2020.

[9] S. Zheng, S. Chen, P. Qi, H. Zhou, and X. Yang, "Spectrum sensing based on deep learning classification for cognitive radios," *China Commun.*, vol. 17, no. 2, pp. 138–148, Feb. 2020.

[10] J. Xie, J. Fang, C. Liu, and L. Yang, "Unsupervised deep spectrum sensing: A variational auto-encoder based approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5307–5319, May 2020.

[11] D. Castelvecchi, "Can we open the black box of AI?," *Nature News.*, vol. 538, no. 7623, Oct. 2016, Art. no. 20.

[12] C. Szegedy *et al.*, "Intriguing properties of neural networks," in *Proc. Int. Conf. Learn. Representations*, 2015, paper no. 6199.

[13] C. Song, C. Xu, S. Yang, Z. Zhou, and C. Gong, "A black-box approach to generate adversarial examples against deep neural networks for high dimensional input," in *Proc. IEEE 4th Int. Conf. Data Sci. Cyberspace*, 2019, pp. 473–479.

[14] S. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 86–94.

[15] K. Han, Y. Li, and B. Xia, "A cascade model-aware generative adversarial example detection method," *Tsinghua Sci. Technol.*, vol. 26, no. 6, pp. 800–812, Dec. 2021.

[16] A. Sayles, A. Hooda, M. Gupta, R. Chatterjee, and E. Fernandes, "Invisible perturbations: Physical adversarial examples exploiting the rolling shutter effect," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14661–14670.

[17] M. Sadeghi and E. G. Larsson, "Adversarial attacks on deep-learning based radio signal classification," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 213–216, Feb. 2019.

[18] T. J. OShea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, Feb. 2018, Art. no. 168C179.

[19] Y. Zhao, X. Gong, F. Lin, and X. Chen, "Data poisoning attacks and defenses in dynamic crowdsourcing with online data quality learning," *IEEE Trans. Mobile Comput.*, vol. 1, no. 1, Dec. 2021, Art. no. 1.

[20] J. Zhang, B. Chen, X. Cheng, H. T. T. Binh, and S. Yu, "PoisonGAN: Generative poisoning attacks against federated learning in edge computing systems," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3310–3322, Mar. 2021.

[21] M. Axline Robert Jr., G. R. Sloan, and R. E. Spalding, "Radar transponder apparatus and signal processing technique," Sandia National Labs. Albuquerque, NM, USA,) USA, 1996.

[22] S. D. Blunt and P. Yantham, "Waveform design for radar-embedded communications," in *Proc. Int. Waveform Diversity Des. Conf.*, 2007, pp. 214–218.

[23] S. D. Blunt, J. Stiles, C. Allen, D. Deavours, and E. Perrins, "Diversity aspects of radar-embedded communications," in *Proc. Int. Conf. Electromagnetics Adv. Appl.*, 2007, pp. 439–442.

[24] J. Xu and B. Li, "A new radar-embedded communication waveform based on singular value decomposition," in *Proc. IEEE 2nd Int. Conf. Comput. Commun. Eng. Technol.*, 2019, pp. 234–238.

[25] I. Goodfellow, J. Shlens, and C. Szegedyn, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Representations*, 2015, Paper no. 189C199.

[26] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. Int. Conf. Learn. Representations*, 2016, Paper no. 128C141.

[27] Y. Dong and F. Liao, "Boosting adversarial attacks with momentum," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, Paper no. 9185C9903.

[28] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Secur. Privacy*, 2017, pp. 39–57.

[29] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A. simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2574–2582.

[30] Y. Shi and Y. E. Sagduyu, "Evasion and causative attacks with adversarial deep learning," in *Proc. IEEE Mil. Commun. Conf.*, 2017, pp. 243–248.

[31] C. Burkard and B. Lagesse, "Analysis of causative attacks against svms learning from data streams," *Proc. 3rd ACM Int. Workshop Secur. Privacy Analytics.*, vol. 1, no. 1, 2017, Paper no. 31C36.

[32] G. Sun, Y. Cong, J. Dong, Q. Wang, L. Lyu, and J. Liu, "Data poisoning attacks on federated machine learning," *IEEE Internet Things J.*, vol. 1, no. 1, Nov. 2021, Art. no. 1.

[33] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," in *Proc. 29th Int. Conf. Mech. Learn.*, 2013, Paper no. 1807C1814.

[34] L. Muoz-Gonzlez *et al.*, "Towards poisoning of deep learning algorithms with back-gradient optimization," in *Proc. 10th ACM Workshop Artif. Intell. Secur.*, 2017, Paper no. 27C38.

[35] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li, "Manipulating machine learning: Poisoning attacks and countermeasures for regression learning," in *Proc. IEEE Symp. Secur. Privacy*, 2018, pp. 19–35.

[36] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Conf. Neural Inf. Process. Syst.*, 2014, Paper no. 2672C2680.

[37] M. Mozaffari-Kermani, S. Sur-Kolay, A. Raghunathan, and N. K. Jha, "Systematic poisoning attacks on and defenses for machine learning in healthcare," *IEEE J. Biomed. Health Inform.*, vol. 19, no. 6, Nov. 2015, Art. no. 1893C1905.

[38] C. Liu, J. Wang, X. Liu,, and Y. -C. Liang, "Deep CM-CNN for spectrum sensing in cognitive radio," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2306–2321, Oct. 2019.

[39] S. M. Kay, "Fundamentals of statistical signal processing: Detection theory," *Detection Theory.*, vol. 2, no. 1, Nov. 1998, Art. no. 1.

[40] F. F. Digham, M. Alouini, and M. K. Simon, "On the energy detection of unknown signals over fading channels," *IEEE Trans. Commun.*, vol. 55, no. 1, pp. 21–24, Jan. 2007.

[41] Y. Zeng, C. L. Koh, and Y.-C. Liang, "Maximum eigenvalue detection: Theory and application," in *Proc. IEEE Int. Conf. Commun.*, 2008, pp. 4160–4164.

[42] Y. Zeng and Y. Liang, "Spectrum-sensing algorithms for cognitive radio based on statistical covariances," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1804–1815, May 2009.

[43] DeepSig, "Deepsig dataset: Radioml2016. 10b," 2016. [Online]. Available: https://www.deepsig.io/datasets

[44] M. Liu, Z. Liu, W. Lu, Y.i Chen, X. Gao, and N. Zhao, "Distributed few-shot learning for intelligent recognition of communication jamming," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 3, pp. 395–405, Apr. 2022.

[45] M. Liu *et al.*, "Location parameter estimation of moving aerial target in space-air-ground integrated networks-based IoV," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5696–5707, Apr. 2022.

[46] M. Liu, C. Liu, M. Li, Y. Chen, S. Zheng, and N. Zhao, "Intelligent passive detection of aerial target in space-air-ground integrated networks," *China Commun.*, vol. 19, no. 1, pp. 52–63, Jan. 2022.

Mingqian Liu (Member IEEE) received the B.E. degree in electrical engineering from Information Engineering University, Zhengzhou, China, in 2006, and the Ph.D. degree in communication and information system from Xidian University, Xi'an, China, in 2013.

He is currently an Associate Professor with the State Key Laboratory of Integrated Services Networks, Xidian University, where he was a Postdoctoral Researcher from 2014 to 2016. From November 2018 to November 2019, he was a Visiting Scholar with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA. His research interests include communication signal processing, statistical signal processing, and artificial intelligence

Hongyi Zhang received the B.S. degree in June 2020 in communication engineering from Xidian University, Xi'an, China, where he is currently working toward the Ph.D. degree in communication and information system.

His research interests include communication signal processing and artificial intelligence.

Zilong Liu (Senior Member, IEEE) received the bachelor's degree from the School of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan, China, in 2004, the master's degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2007, and the Ph.D. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, in 2014.

He is currently a Lecturer (Tenured Assistant Professor) with the School of Computer Science and Electronics Engineering, University of Essex, Essex, U.K. From January 2018 to November 2019, he was a Senior Research Fellow with the Institute for Communication Systems, Home of the 5G Innovation Centre (5GIC), University of Surrey, Guildford, U.K., during which he studied the air-interface design of 5G communication networks (e.g., machine-type communications, V2X communications, and 5G New Radio). Prior to his career in U.K., he spent nine and half years with NTU, first as a Research Associate from July 2008 to October 2014 and then as a Research Fellow from November 2014 to December 2017. The Ph.D. Thesis "Perfect- and Quasi-Complementary Sequences," focusing on fundamental limits, algebraic constructions, and applications of complementary sequences in wireless communications, has settled a few long-standing open problems in the field. He has authored or coauthored more than 80 peer-reviewed journal papers, including more than 40 IEEE transactions papers. His research interests include the interplay of coding, signal processing, and communications, with a major objective of bridging theory and practice as much as possible, and applying various machine learning and AI tools for enhanced communication and networking.

Dr. Liu is an Associate Editor for IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE WIRELESS COMMUNICATIONS LETTERS, IEEE ACCESS, *Frontiers in Communications and Networks, and Frontiers in Signal Processing*. He is or was currently the General Co-Chair of the 10th International Workshop on Signal Design and its Applications in Communications (IWSDA'2022) and a the TPC Co-Chair of the 2020 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS'2020). He was a Tutorial Speaker of VTC-Fall'2021 and APCC'2021. Besides, he was or is currently a TPC Member of a number of IEEE conferences/workshops (e.g., ICC, GLOBECOM, WCSP, ICCS, and SETA). He is a Senior Member of IEEE.

Nan Zhao (Senior Member, IEEE) received the B.S. degree in electronics and information engineering, the M.E. degree in signal and information processing, and the Ph.D. degree in information and communication engineering from the Harbin Institute of Technology, Harbin, China, in 2005, 2007, and 2011, respectively.

He is currently a Professor with the School of Information and Communication Engineering, Dalian University of Technology, Dalian, China, where he did postdoctoral research from 2011 to 2013. He has authored or coauthored more than 100 papers in refereed journals and international conferences. His research interests include interference alignment, cognitive radio, wireless power transfer, optical communications, and indoor localization.

Dr. Zhao is a Senior Member of the Chinese Institute of Electronics. He is currently the Editor of the *Wireless Networks, Physical Communication, AEU-International Journal of Electronics and Communications, Ad Hoc & Sensor Wireless Networks,* and *KSII Transactions on Internet and Information Systems*. He was a Technical Program Committee (TPC) Member for many interferences, e.g., Globecom, VTC, and WCSP