remote sensing

MDPI

Article

Intelligent Radar Jamming Recognition in Open Set Environment Based on Deep Learning Networks

Yu Zhou *, Song Shang, Xing Song †, Shiyu Zhang, Tianqi You and Linrang Zhang

National Laboratory of Radar Signal Processing, Xidian University, Xi'an 710071, China
* Correspondence: zhouyu@mail.xidian.edu.cn; Tel.: +86-18192338726
† Current address: No. 207, Jiufeng 3rd Road, Jiangxia District, Wuhan 430206, China.

Abstract: Jamming recognition is an essential step in radar detection and anti-jamming in the complex electromagnetic environment. When radars detect an unknown type of jamming that does not occur in the training set, the existing radar jamming recognition algorithms fail to correctly recognize it. However, these algorithms can only recognize this type of jamming as one that already exists in our jamming library. To address this issue, we present two models for radar jamming open set recognition (OSR) that can accurately classify known jamming and distinguish unknown jamming in the case of small samples. The OSR model based on the confidence score can distinguish known jamming from unknown jamming by assessing the reliability of the sample output probability distribution and setting thresholds. Meanwhile, the OSR model based on OpenMax can output the probability of jamming belonging to not only all known classes but also unknown classes. Experimental results show that the two OSR models exhibit high recognition accuracy for known and unknown jamming and play a vital role in sensing complex jamming environments.

Keywords: jamming recognition of radar; deep learning; neural networks; open set recognition

check for updates

Citation: Zhou, Y.; Shang, S.; Song, X.; Zhang, S.; You, T.; Zhang, L. Intelligent Radar Jamming Recognition in Open Set Environment Based on Deep Learning Networks. *Remote Sens.* **2022**, *14*, 6220. https://doi.org/10.3390/rs14246220

Academic Editor: Gwanggil Jeon

Received: 31 October 2022
Accepted: 6 December 2022
Published: 8 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The electromagnetic environment of radar is becoming increasingly complex with the advancement of modern electronic jamming technology, and new types of jamming are occurring more regularly [1–5]. Many intelligent technologies are applied in the field of radar detection and anti-jamming. Jamming recognition technologies are important for modern radars. It is an important prerequisite to ensure that we can successfully counter enemy jamming measures in electronic warfare. Only by correctly identifying the jamming signal emitted by the jammer can the corresponding measures for specific types of jamming be implemented. Hence, the radar jamming recognition module plays a primary role in the intelligent radar system. The high performance of the radar jamming recognition module is the prerequisite for the intelligent anti-jamming system to obtain an excellent effect.

The applications of radar in military and civil areas have rapidly progressed due to the extensive development of machine learning and deep learning technologies. Recently, some researchers have combined technologies of machine learning and radars to implement identity authentication. Among these researches, radar has shown great promise in healthcare applications as it requires neither contact nor line-of-sight and does not give rise to privacy concerns associated with video imaging [6–8]. Technologies of deep learning have also been used in the field of smart grids. A smart detection method based on secure federated deep learning is proposed to prevent false data injection attacks (FDIA) [9]. A deep learning-based short-term voltage stability (STVSA) model has been proposed which achieves higher accuracy in the case of small samples compared with the previous methods [10]. The two categories of radar jamming recognition methods are feature extraction-based [11] and deep learning-based methods [12]. A convolutional neural network (CNN)-based radar jamming recognition method was proposed in 2019, first using the time-frequency images

of the various jamming signals as the CNN inputs [13]. A weighted integrated CNN radar jamming recognition algorithm based on transfer learning was proposed in 2021 for the low recognition accuracy of deep learning jamming recognition methods in the case of small samples [14]. A jamming recognition method based on Residual Neural Network (ResNet) was proposed in 2021, using the bispectrum diagonal slice extracted from the jamming signals as the input of ResNet to realize the recognition of active deception jamming [15].

However, some problems in the current radar jamming recognition methods remain to be solved: Existing radar jamming recognition methods assume that the types of jamming signals in the testing environment are the same as those in the training jamming library. With this assumption, jamming recognition is referred to as close set recognition. When radars detect an unknown type of jamming in the actual open set jamming scenario, the existing radar jamming recognition models will incorrectly identify it as a known type of jamming in the training jamming library. Nevertheless, jammers and radars are encountered in the actual situation. Moreover, new types of jamming signals are constantly developed. Consequently, collecting all types of jamming data would be difficult for the training jamming library of our jamming recognition model. Hence, the actual radar jamming environment is the open set environment. In particular, the type of jamming that does not exist in the training jamming library is likely to occur in the testing environment. The recognition in the open set environment is called open set recognition (OSR), and it has already been utilized in the fields of image recognition [16].

Most of the original OSR models are implemented by using support vector machines (SVM) to constrain some parts of the open space [16–19]. Some of them can only distinguish whether the test samples belong to known or unknown sample, but cannot classify which class the known samples belong to [20]. With the gradual deepening of research in this field, recognition models that can distinguish unknown types and correctly identify known types begin to appear [21], and OSR is successfully implemented using deep learning networks [21–23]. Scheierer [16] introduced the concept of OSR and proposed the 1-vs-set model from the perspective of constraining open space, which is the first OSR model. The 1-vs-set model distinguishes known and unknown targets by using two parallel hyperplanes that do not effectively utilize the open space. Quasi-linear discriminants are suggested to overcome this problem, which outperform the 1-vs-set model proposed by Scheierer. These discriminants can be trained from either binary or positive-only samples by using constrained quadratic programs related to SVMs [24]. Bendale and Boult [13] proposed a deep neural network model for OSR by introducing the OpenMax layer to estimate the probability of an unknown class, which pioneered the use of deep neural networks in OSR. OpenMax has been used in text classification to solve the open set text classification problem, and has achieved good performance [21].

In the face of increasingly complex electromagnetic environments, radar jamming recognition needs the ability to operate normally in open set scenarios. This requires that the radar not only classifies the known jamming accurately, but also distinguishes the unknown jamming correctly so that specific anti-jamming measures can be implemented promptly. We urgently need to find an intelligent recognition model that enables radar to automatically process unknown jamming signals. In view of the great success of the deep learning method in the classification and recognition of radar jamming signals, we hope to use the deep learning model to achieve the OSR for radar jamming signals. For these reasons, we propose two OSR models to address the open set problem in the radar anti-jamming environment so that radars can recognize the known jamming and distinguish the unknown: a model based on the confidence score and a model based on OpenMax. The OSR model based on the confidence score quantifies the difference between known and unknown classes based on the probability distribution of their output and sets a threshold to realize the discrimination of known and unknown types. The probability distribution of known jamming is concentrated on one class. By contrast, the probability distribution of unknown jamming is more uniform. The OSR model based on OpenMax replaces the SoftMax layer with the OpenMax layer by altering the full connection layer

of the second-last layer of the original classifier (the previous layer of the SoftMax layer). The modified OpenMax layer outputs the probability that the jamming samples belong to each known class and the probability that they belong to the unknown class. OSR is achieved by using the category to which the highest probability of the output probability distribution belongs as the classification result. The main contributions of this work are as follows: First, we introduce the concept of OSR in the radar jamming recognition. Second, we propose the confidence score-based model to distinguish unknown types of jamming signals. Third, we combine the OSR models with the deep networks in the intelligent jamming recognition module.

In this paper, Section 2 introduces the common jamming signals and their time-frequency images. Section 3 introduces the open set jamming recognition models proposed in this work. Section 4 introduces several experiments to verify the performance of the OSR model. Section 5 summarizes the main work of this paper and provides a conclusion.

2. Jamming Signal Analysis

Radar jamming can be divided into intentional and unintentional jamming according to the purpose. Meanwhile, intentional jamming can be divided into active and passive jamming. Radar active jamming can be divided into three categories: suppression, deceptive, and composite jamming [25].

This section introduces the jamming signal model and time-frequency domain images of active radar jamming. The types of jamming signals vary in the actual scenarios. Among the three types of jamming signals, suppression jamming interferes with radar in specific frequency bands through high-energy jamming signals, rendering radar ineffective in these bands. Deceptive jamming interferes with radar by sending many false targets to it, making it impossible for the radar to detect the real target. Composite jamming combines the advantages of suppression jamming and deceptive jamming. This mechanism prevents the radar from normally operating in a specific frequency band and detects a significant number of false targets.

Eleven types of common jamming signals are simulated to verify the validity of jamming recognition models proposed in this work. As shown in Figure 1, the three types of suppression jamming include noise amplitude modulation (AM) jamming, noise frequency modulation (FM) jamming, and radio frequency (RF) noise jamming. Meanwhile, deceptive jamming includes range gate pull-off (RGPO) jamming, velocity gate pull-off (VGPO) jamming, smeared spectrum (SMSP) jamming, interrupted sampling repeater jamming (ISRJ), and chopping and interleaving (C&I) jamming. Among these types of deceptive jamming, RGPO and VGPO jamming can produce a single false target with the incorrect distance or velocity. SMSP jamming, ISRJ, and C&I jamming can produce multiple false targets by modulating and retransmitting the signal saved in the digital radio frequency memory system. Composite jamming includes RGPO and noise AM jamming, RGPO and noise FM jamming, and RGPO and RF noise jamming, which are additive jamming. These 11 types of jamming signals are simulated according to the generation mechanism [1–5], and the simulation results are shown in Section 4.

After the radar active jamming signal is obtained, we convert it into a time–frequency image by short-time Fourier transform (STFT) and take the time–frequency image of active radar jamming as a dataset to provide a data basis for the subsequent jamming classification [26]. The operation process of STFT is shown in Figure 2.

STFT operates as follows:

1. Divide the signal into several sub-signals.
2. Add windows and perform Fourier transform (FT) for each sub-signal.
3. Combine the Fourier transform results of each sub-signal.

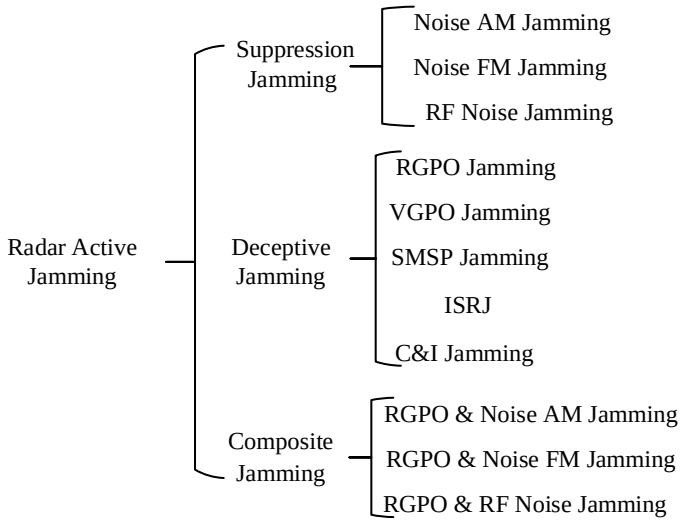


Figure 1. 11 types of jamming signals.

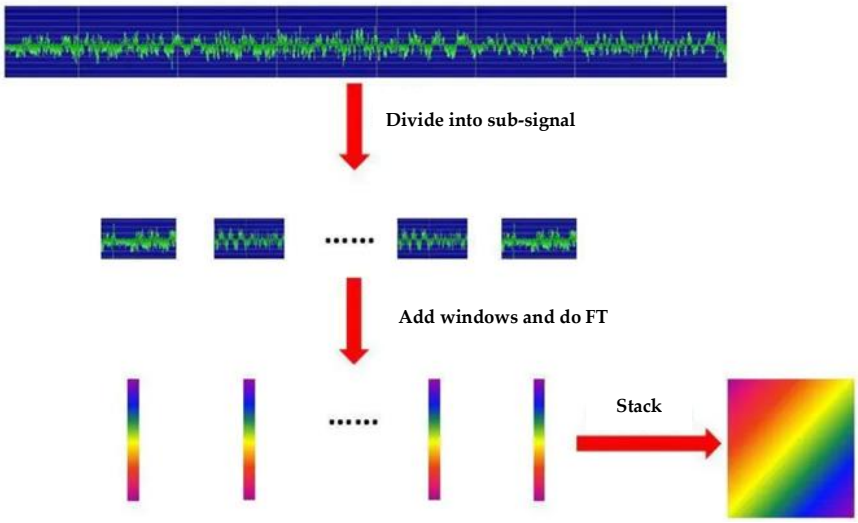


Figure 2. The operation process of STFT.

Thereafter, the time–frequency image of jamming signal can be obtained. The mathematical expression of STFT is shown as follows:

$$X(f,t) = \int_{-\infty}^{+\infty} \omega(t-\tau)s(t)e^{-j2\pi ft}d\tau \quad (1)$$

where $s(t)$ is the jamming signal to be analyzed, $\omega(t)$ is the window function, and $X(f,t)$ is the Fourier transform of sub-signal $\omega(t-\tau)s(t)$.

We convert 1D time–domain signals into 2D time–frequency images through STFT to ensure that the data can provide information in time–domain and frequency–domain dimensions and can be directly used as the CNN inputs.

3. Two Models of Open Set Recognition

Most current jamming recognition techniques assume that the radar is in a closed environment. In particular, the type of jamming radar encountered during training is the same as that encountered during testing, and there is no unknown type of jamming in the testing set that is not present in the training set.

By contrast, with the constant appearance of new jamming types, there is no guarantee that the jamming detected in the actual situation would be known in advance during training, and the previously assumed close set scenarios are increasingly unsuitable for radar jamming recognition. In this work, the concept of open set is introduced in radar jamming detection. Specifically, an unknown jamming is detected that does not arise during training. Figure 3 shows the open set environment. The recognition ability of radar in a complex electromagnetic environment is improved by studying the recognition method of jamming signals in open set scenes.

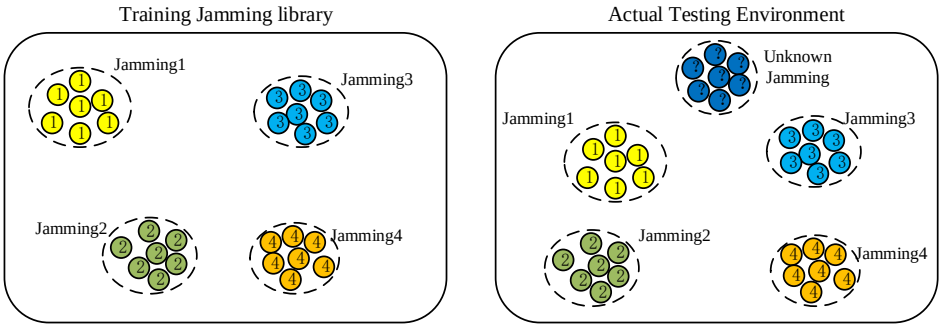


Figure 3. The open set jamming environment.

The following describes the OSR definition [27]:

Definition 1 (OSR). *The training and test sets consisting of sample-label pairs are represented as follows:*

$$D_{tr} = \{(x_i \in X_{tr}, y_i \in Y_{tr} \equiv 1, \ldots, C_{TR}) | i = 1, 2, \ldots, m\},$$

$$D_{te} = \left\{\left(x_j \in X_{te}, y_j \in (Y_{tr} \cup \ Y_e) \equiv Y_{te} \equiv 1, \ldots, C_{TR}, \ldots, C_{TE}\right) \middle| j = 1, 2, \ldots, n\right\},$$

where D_{tr} and D_{te} are the training and testing sets, respectively; x_i and x_j are the training and testing samples; y_i and y_j are the labels of x_i and x_j; m and n are the numbers of training and testing samples; X_{tr} and X_{te} are the sets of training and testing samples; Y_{tr} and Y_{te} are the label sets of the training and testing samples; and Y_e is the label set in D_{te} but not in D_{tr}.

We define that $Y_e = C_{TR} + 1, \ldots, C_{TE}$. Accordingly, more labels are present in the testing set that do not appear in the training set. OSR uses training set D_{tr} to learn a recognition model that can correctly classify the known categories Y_{tr} and accurately distinguish the unknown categories Y_e.

The following part of this section will introduce the pre-training networks used for the OSR models.

3.1. Pre-Training Networks

Before the OSR, a pre-training neural network should be chosen as the radar jamming classifier. Herein, we select the CNN model based on transfer learning as a classifier for jamming recognition to ensure that the model has better generalization performance and is not easy to overfit. The original domain dataset for transfer learning is the ImageNet [28], which includes more than 14 million labeled images and covers more than 2000 categories. The target area is the time–frequency image of the jamming signal. The ImageNet dataset fully meets the requirements of the original domain dataset for transfer learning. Accordingly, the pre-training model used for transfer learning can obtain sufficient prior knowledge from ImageNet. The pre-training model based on transfer learning causes the jamming recognition model to converge faster with less overfit during training [29].

The common modes for transfer learning include feature, instance, and model transfers. Among these modes, model transfer is one of the most commonly used transfer learning methods in deep learning [30]. Model transfer refers to training a deep learning model (also

known as a pre-training model) in the original domain, then transferring the parameters of the pre-training model to the recognition model in the target domain, and finally training the recognition model on the small sample dataset to obtain the small sample recognition model. The training process of model-based transfer learning is shown in Figure 4.

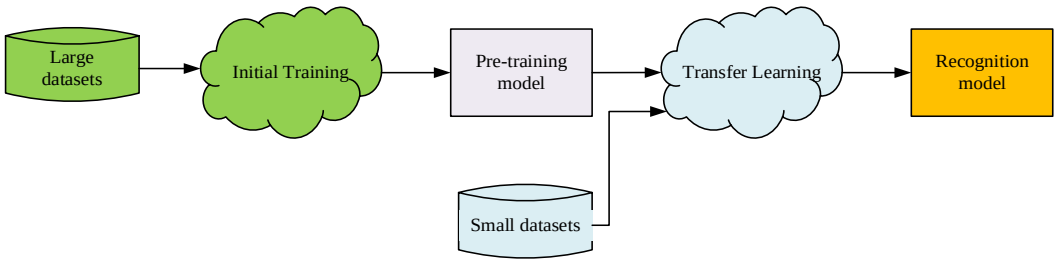


Figure 4. The training process of transfer learning.

The network structure is presented in Figure 5. The model's input data is the time–frequency image of the active jamming signal. The structure of the model consists of two parts: the classification model based on ImageNet and the shallow CNN classification model. First, we construct a network with the same parameters as the network trained by the ImageNet dataset and use it as the feature extraction network for jamming time–frequency images. Then, we train a shallow network to obtain the recognition results through the jamming feature output by the deep network.

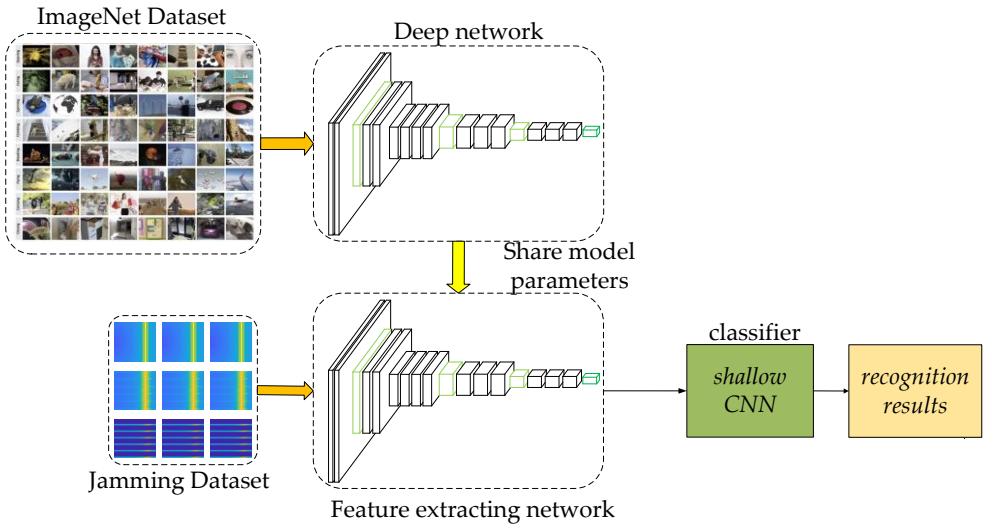


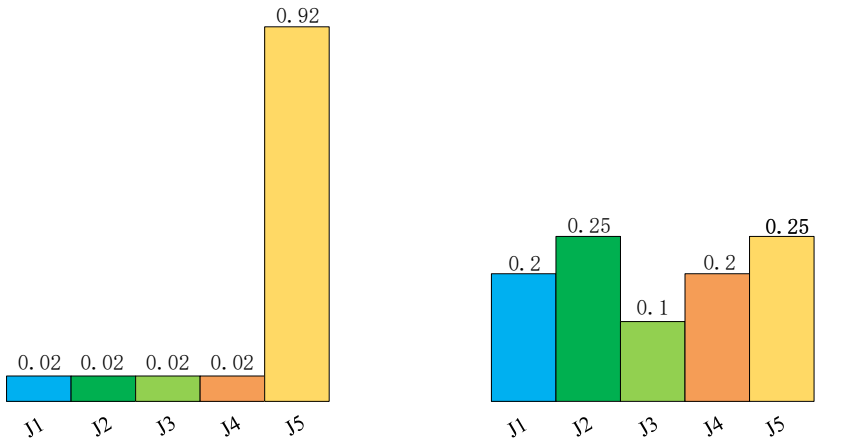
Figure 5. The structure of pre-training networks.

3.2. OSR Model Based on Confidence score

3.2.1. Definition of Confidence Score

The traditional recognition models in close set scenes are not applicable to the jamming recognition in open set scenes because they recognize unknown jamming as known jamming based on the probability that they belong to each known class in an open set environment.

Although unknown jamming outputs the probability of belonging to each known class at the SoftMax layer through a traditional deep network classifier, the probability distribution is significantly lower than that of the known jamming. This difference is depicted in Figure 6.



(a) Probability Distribution of Known Jamming (b) Probability Distribution of Unknown Jamming

Figure 6. The difference of probability distribution between known and unknown jamming.

Figure 6 shows that the output probability distribution of unknown and known jamming is significantly different after passing through the classifier. The probability distribution of the known jamming concentrates on a specific class, while that of the unknown jamming scatters on all the classes.

Based on the above-mentioned difference between known and unknown jamming, we propose a model to distinguish both of them: A coefficient is designed to describe the difference of probability distribution between known and unknown jamming, and a threshold is determined to define how large the coefficient is to classify it as known or unknown jamming.

Our chosen coefficient is called the confidence score, which borrows from the concept of information entropy H [31]. The equation of H is shown in Equation (2):

$$H = -\sum_{k=1}^{N} p_k log_2 p_k \tag{2}$$

where H is information entropy, p_k is the probability of the kth information, and N is the number of information class. The more scattered the probability distribution $p_k(k = 1, \ldots, N)$ is, the larger H is. H is range from $[0, log_2 N]$. Therefore, H can be used as an indicator to measure the amount of information that a sample belongs to each class in the classification mission.

We use H to define a new coefficient called confidence score C, which is shown in Equation (3), where $p_k(k = 1, \ldots, N)$ is the probability distribution that the jamming belongs to each class. N is the number of known jamming classes in the training set. When the posterior probability distribution p_k is more uniform, the confidence score C is smaller, and when the probability distribution p_k is more concentrated, the confidence score C is larger. Confidence score C ranges from $[0, 1]$.

$$C = 1 - \frac{H}{log_2 N} = 1 + \frac{\sum_{k=1}^{N} p_k log_2 p_k}{log_2 N} \tag{3}$$

3.2.2. Structure of the OSR Model Based on Confidence Score

The OSR model based on the confidence score is shown in Figure 7. This model consists of two parts. The first part is a jamming classifier. This part aims to obtain the probability distribution that the input jamming signal belongs to different known classes, which can be obtained by the SoftMax layer. The second part is the reliability detection of the classification result. This part judges whether the recognition result of the classifier is reliable by calculating the confidence score of the output probability distribution. If the

finding is reliable, then the result that the jamming belongs to a known class is outputted; otherwise, input jamming is regarded as unknown.

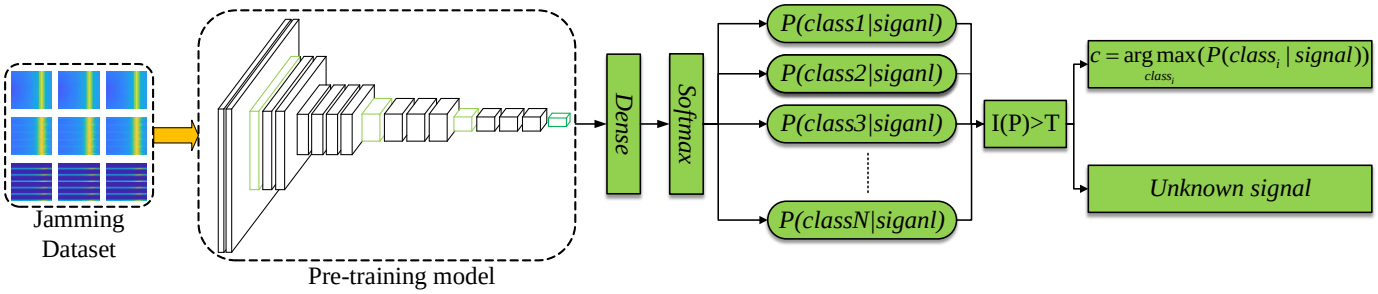


Figure 7. The structure of OSR model based on confidence score.

We need to find a threshold T to judge whether the confidence score C is reliable or not (Figure 7). When C is smaller than T, the input signal does not belong to any of the known classes; hence, the output result is that this jamming signal is an unknown signal. When C is bigger than T, the input signal belongs to one of the known classes, and the specific result will be outputted as follows:

$$c = argmax_{class_i}(P(class_i|signal)) \tag{4}$$

If the probability distribution of the sample is reliable, then the class with the highest probabilistic value in the probability distribution of the input sample is selected as the output classification result, which is shown in Equation (4).

3.2.3. Calculation of Confidence Score Threshold

Given that we have no information about unknown jamming in the actual open set environment, the threshold T can only be determined using the training set. The training procedure of threshold T is described as follows:

First, we need to simulate the open set environment in the training set, so we assume a part of the known jamming categories in the training set as unknown. Here we assume four types of jamming as known and the other four types as unknown.

Second, we subdivide the training set into fitting and validation sets, where all the samples in the fitting set are assumed known jamming, and the validation set includes the assumed known jamming and the assumed unknown jamming. The result of the division is shown in Figure 8.

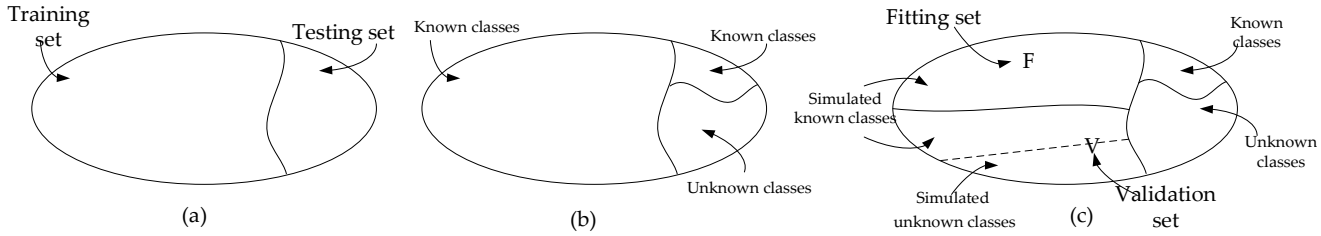


Figure 8. The division of jamming dataset.

Third, we use samples in the fitting set F to train an initial classifier, which will be used to train the optimal threshold $\hat{T}$ in step 4.

Fourth, we use the initial classifier obtained in step 3 to train the optimal threshold $\hat{T}$ in the validation set. The objective function of training the optimal threshold is shown in Equation (5).

$$\hat{T} = argmin_T\{\sum_{i=1}^{M_1} sign(T - c_i) \times \omega_1 + \sum_{j=1}^{M_2} sign(c_j - T) \times \omega_2\} \tag{5}$$

where M_1 is the number of assumed known jamming in validation set V, M_2 is the number of assumed unknown jamming in validation set V, c_i is the confidence score of an assumed known jamming sample, c_j is the confidence score of an assumed unknown jamming sample, ω_1 is the misjudgment penalty weight of the assumed known jamming, ω_2 is the misjudgment penalty weight of the assumed unknown jamming, $sign()$ is the sign function, and $sign(x)$ is positive for $x > 0$ and negative for $x < 0$.

The explanation of Equation (5) is as follows: If a jamming sample is an assumed known one, then we hope that its confidence score c_i is bigger than threshold T, so we let $sign(T - c_i)$ be positive as the misjudgment penalty for c_i is smaller than T. In the same way, if a jamming sample is an assumed unknown one, then we hope that its confidence score c_j is smaller than threshold T, so we let $sign(c_j - T)$ be positive as the misjudgment penalty for c_j is bigger than T. Lastly, we obtain the penalty function to be optimized and then minimize it to obtain the optimal threshold $\hat{T}$ by doing this for all samples in the validation set (including those assumed to be known and those assumed to be unknown).

In the actual training process, we choose genetic algorithm (GA) [32] to optimize the Equation (5).

3.2.4. Training and Testing Process

The training process is shown in Figure 9. This process consists of two parts. The first part is to obtain confidence score threshold T. We need to subdivide the training set into fitting set F and validation set V. Then, an initial jamming classifier is trained on the fitting set by using the pre-training model, and an optimal confidence score threshold $\hat{T}$ is found by minimizing the misjudgment penalty function on validation set V. The second part is to train a jamming classifier, which uses the pre-training model to train one jamming classifier on all the jamming training datasets.

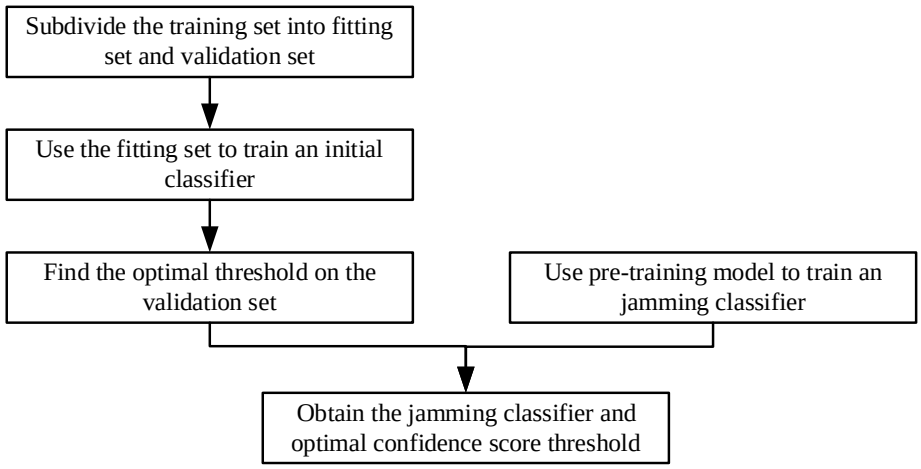


Figure 9. The training process of OSR model based on confidence score.

The testing process of the OSR model based on the confidence score is shown in Figure 10. First, the jamming samples to be classified are inputted into the trained jamming classifier to obtain the probability distribution that the jamming samples belong to different known jamming types. Then, the confidence score C of the probability distribution is calculated by using Equation (3). Finally, we judge whether the confidence score C exceeds the optimal threshold $\hat{T}$ or not. We choose the jamming type with the highest probability value as the output recognition result for $C > \hat{T}$; otherwise, we output it as an unknown jamming type. In the testing process, the time complexity of processing one jamming sample is O(N+2), where N is the number of jamming types in the training set.

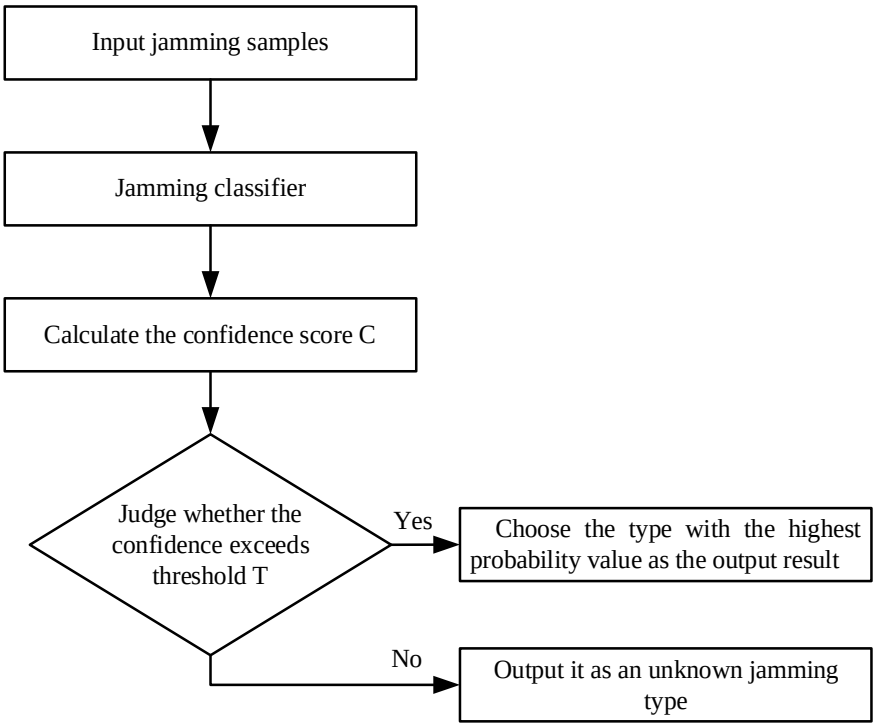


Figure 10. The testing process of OSR model based on confidence score.

3.3. OSR Model Based on OpenMax

The OSR model based on the confidence score needs to simulate the open set scenario and determine the threshold of the confidence score. This approach causes two problems: First, the open set environment simulated by the training set might have a massive deviation from the actual open set environment. Second, because certain unknown jamming might be highly similar to a known jamming class, the probability distribution would be similar to that of some known jamming, causing the confidence score threshold to fail to distinguish between the known and unknown classes. An OpenMax-based OSR model is proposed to solve these problems, which does not require setting any threshold.

3.3.1. Structure of the OSR Model Based on OpenMax

Most jamming classifiers working in a close set environment generate a probability distribution over the known N types of jamming (where N represents the number of jamming types in the training set). When an unknown jamming sample is inputted into the jamming classifier, it is forcibly recognized as one of the N known types of jamming. The OpenMax layer can estimate the probability that the input jamming sample belongs to the unknown jamming by extending the SoftMax layer. This layer changes the output dimension of the jamming classifier from N to $N+1$ (where 1 represents other unknown jamming types in the testing environment). When an unknown jamming sample is inputted into an open set jamming recognition model based on OpenMax, the output probability distribution of the OpenMax will have a higher probability on the unknown jamming type, thus detecting the unknown jamming type. The structure of the OSR model based on OpenMax is shown in Figure 11.

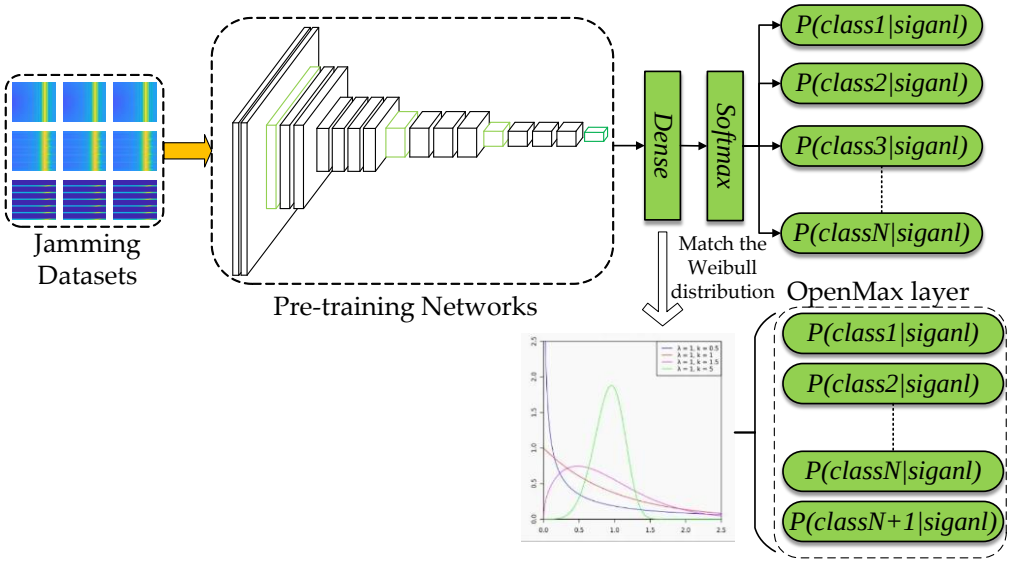


Figure 11. The structure of the OSR model based on OpenMax.

The model consists of three parts. The first part is the jamming classifier, the function of which is to train a jamming classifier on the training set by using the pre-training model, extracting the output of the second-last layer of the jamming classifier as the activation vector (the full connection layer before SoftMax). Given that CNN can efficiently extract image features and reflect them on the activation vectors, we can measure the similarity between the input jamming samples by calculating the distance between the activation vectors of the input jamming samples. The second part matches the boundary of known jamming and the Weibull distribution of the known jamming boundary. The function of this part is to match the Weibull distribution of the distance between the input jamming sample and the activation vector boundary of each known jamming. The probability that the input jamming samples do not belong to known jamming can be determined through the Weibull distribution. The third part is to build the OpenMax layer. The function of this part is to create the probability distribution of $N+1$ types based on the Weibull distribution of each jamming type to ensure that the probability that the input jamming sample belongs to an unknown type can be included in the output probability distribution.

3.3.2. Match the Boundary of Known Jamming

The activation vectors extracted by the classifier network from the input jamming samples of the same type will gather together because of the similarity of the jamming samples from the same type. The samples of the different types will form distinct communities in the space of the activation vector because of the variations among the different types of jamming. Moreover, a difference exists between the unknown and the known jamming samples. The unknown jamming type is far from the known jamming type community in the activation vector space. In this work, the center of the activation vector of each known jamming type is the average activation vector. Each known jamming type's average activation vector is determined by averaging the activation vectors of all samples. The distribution of the jamming samples in the activation vector space is illustrated in Figure 12. The dimension of the activation vector is N (where N is the number of known jamming types). Here, only two dimensions are taken to intuitively reflect the activation vector's characteristics.

In Figure 12, every jamming type has its boundary point, which is far from the average activation vector. The activation vector communities of the known jamming types can be obtained in the activation vector space after the training process. Consequently, if an unknown jamming type is inputted into the classifier during the testing process, an

activation vector that is far from all of the jamming activation vector communities can be obtained.

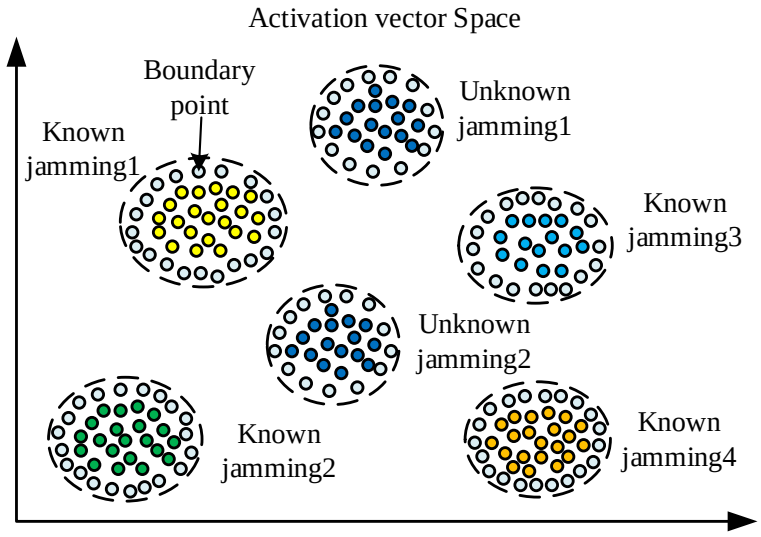


Figure 12. The activation vector space.

Based on the above principle, when a new jamming sample is inputted, its activation vector can be determined whether it belongs to the jamming type by determining whether it is within the boundary of the known jamming communities.

We use the extreme value theorem [33] to define the boundary for each type of jamming, which is as follows: Given n samples of identical and independent samples $Z_1, Z_2, \ldots, Z_n$ from an unknown distribution $F(z)$, the maximum of n samples $Z_m = max(Z_i), i \in [1, n]$ follows the Weibull distribution. The cumulative distribution function (CDF) $F(x)$ of the Weibull distribution is as follows:

$$P(Z_m \leq x) = F(x) = 1 - exp\left(-\frac{\|x - \tau\|}{\lambda}\right)^{\kappa} \tag{6}$$

where τ, λ, and κ are the three parameters of the Weibull distribution; Z_m is the maximum value; x is a variable of CDF; and $P(Z_m \leq x)$ means the probability that the maximum value Z_m is less than x.

The jamming samples of the same type can be considered from the same distribution and independent of each other. The boundary points of each type of jamming can be considered to be the samples with the farthest distance between the activation vector and the average activation vector. Based on the extreme value theorem, the distance between the activation vector and the average activation vector of each known jamming boundary point follows the Weibull distribution. The boundary of each known jamming can be obtained by matching the Weibull distribution of the boundary point of each known type. This work uses 10 samples, the activation vector of which is farthest from the average activation vector to match the Weibull distribution of the boundary points of every known jamming type. When a new jamming sample is inputted, the distance between the jamming sample activation vector and the average activation vector of each known jamming type is first calculated. Then, the distance is inputted into the Weibull distribution of the boundary points of each type of jamming, and the probability that the boundary distance of each type of jamming is less than that distance is obtained, that is, the probability that the jamming sample is outside the boundary of the corresponding known jamming type (the input jamming sample does not belong to the known jamming type).

3.3.3. Build the OpenMax Layer

The OpenMax layer constructs a probability distribution of class N+1 from the Weibull distribution at the boundary of known jamming types. Accordingly, the probability distribution of the output contains the probability that the input jamming samples are of unknown jamming types.

The building process of the OpenMax layer is shown in Algorithm 1. First, we extract the activation vector $v(x) = v_1(x), \ldots, v_N(x)$ of the testing jamming sample x. Next, the probability $\omega_i(x)$ that the testing jamming samples belong to each type of known jamming is calculated using the boundaries of the known jamming matched by Weibull distribution (where ω_i denotes the probability that test jamming sample x belongs to class i of known jamming and is derived from the probability that 1-Test sample does not belong to class i of known jamming). Then, the new activation vector is redefined. The length of the new activation vector changes from N to $N+1$. The new activation vector contains two parts. The first part is the activation value of the test samples for known jamming of class N, which includes N activation values $\hat{v}(x) = v(x) \bigodot \omega(x)$. This value is obtained by performing a dot product between the original activation vector and the probability that a testing sample belongs to each of the known jamming types. The second part is the activation value of the testing jamming sample on the unknown jamming class, which is $\hat{v}_{N+1}(x) = \sum_i v_i(x)(1 - \omega_i(x))$. This value is obtained by performing a dot product first between the original activation value and the probability that a testing sample does not belong to each of the known jamming types and then summing up the results. Finally, the probability distribution of the OpenMax output is obtained based on the new activation vector. The probability distribution of the output includes the probability that the test jamming samples belong to all types of known jamming and the probability that the test jamming samples belong to unknown jamming. The final output chooses the type with the highest probability as the recognition result. OpenMax changes the output N dimensions of the open set jamming recognition model from $N+1$ dimensions by estimating the probability that the testing jamming samples belong to the unknown jamming. When the unknown jamming samples are inputted into the open set jamming recognition model based on OpenMax, the probability distribution of the OpenMax output has a higher probability on the unknown jamming type, thus distinguishing the unknown jamming type. Furthermore, the OpenMax-based OSR model does not need to simulate the open set environment to set thresholds compared with a confidence score-based OSR model. The algorithm of OpenMax has the time complexity of O(2N+5). The time complexity of OpenMax-based model is in the same order of magnitude as that of confidence score-based model.

Algorithm 1. OpenMax layer building algorithm

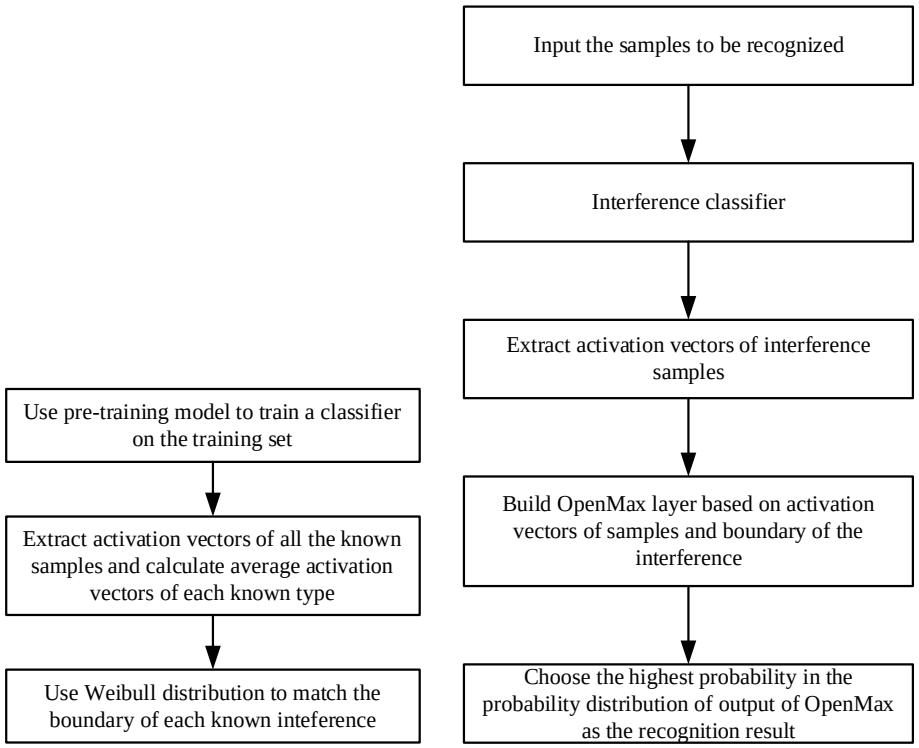
OpenMax: Estimate the probability that the input jamming is an unknown class
Input: Activation vectors of testing samples $v(x) = v_1(x), \ldots, v_N(x)$**;**
Average activation vector for known jamming types μ_j **and Weibull distribution parameters of boundary points** $\rho_j = \left(\tau_j, \lambda_j, \kappa_j\right)$**;**
1 for $i = 1, \ldots, N$ *do*
2 $\quad \boldsymbol{\omega}_i(x) = 1 - exp\left(-\frac{x-\tau}{\lambda}\right)^{\kappa}$
3 end
4 redefinition activation vector $\hat{v}(x) = v(x) \bigodot \boldsymbol{\omega}(x)$
5 define the unknown part of the activation vector $\hat{v}_{N+1}(x) = \sum_i v_i(x)(1 - \omega_i(x))$
6. Probability distribution generated by OpenMax:

$$p(y = j|x) = \frac{exp\left(\hat{v}_j(x)\right)}{\sum_{i=1}^{N+1} exp(\hat{v}_i(x))}$$

7. recognition result: $y^* = argmaxP(y = j \mid x)$

3.3.4. Training and Testing Process

The training and testing processes of the model based on OpenMax is shown in Figure 13. The training process is described as follows:



(a) Training process of OSR model based on OpenMax

(b) Testing process of OSR model based on OpenMax

Figure 13. Training and testing process of OSR model based on OpenMax.

First, a pre-training model is used to train a jamming classifier on the jamming training set. Then, the activation vectors of all known jamming samples are extracted, and the average activation vectors of every type of known jamming are calculated. Finally, the Weibull distribution is used to match the boundary of each type of known jamming. The testing process is described as follows: First, the jamming samples to be recognized are inputted into the jamming classifier, and the activation vectors of the jamming samples are extracted. Then, an OpenMax layer is constructed based on the activation vectors of the jamming samples and the Weibull distribution of each known jamming boundary. Finally, the highest probability type in the probability distribution of the OpenMax output is selected as the output of the recognition results.

4. Simulation and Analysis

This section describes in detail the experimental parameters and results, which includes the simulation results of the jamming signal, the parameters of pre-training networks, and the performance of the OSR proposed in this work. Several experiments are implemented to verify the performance and robustness of the two OSR models we proposed. These experiments include the improvement of recognition performance after L1 and L2 regularization, the analysis of average accuracy in open set jamming recognition with and without the number of unknown types of jamming fixed, and the robustness analysis of the two OSR models. Experiments show that both models perform effectively in complex open set environments (lower JNR and more unknown types) and can be used in the jamming recognition part of radars in practice.

4.1. Jamming Signal Datasets

Eleven types of jamming signal have been simulated in this work, and their time–frequency images have been obtained by STFT. The parameters of the time–frequency images are as follows: the sampling frequency of the jamming signal is 400 MHz, and the sampling time is 600 μs. The window function is the Hamming window, the length of which is 512. The jamming to noise ratio (INR) is −5–10 dB. The size of time–frequency images is 224 × 224 × 3. The time–frequency images of the 11 types of jamming signal are shown in Figure 14, and the specific parameters of each jamming signal are illustrated in Table 1.

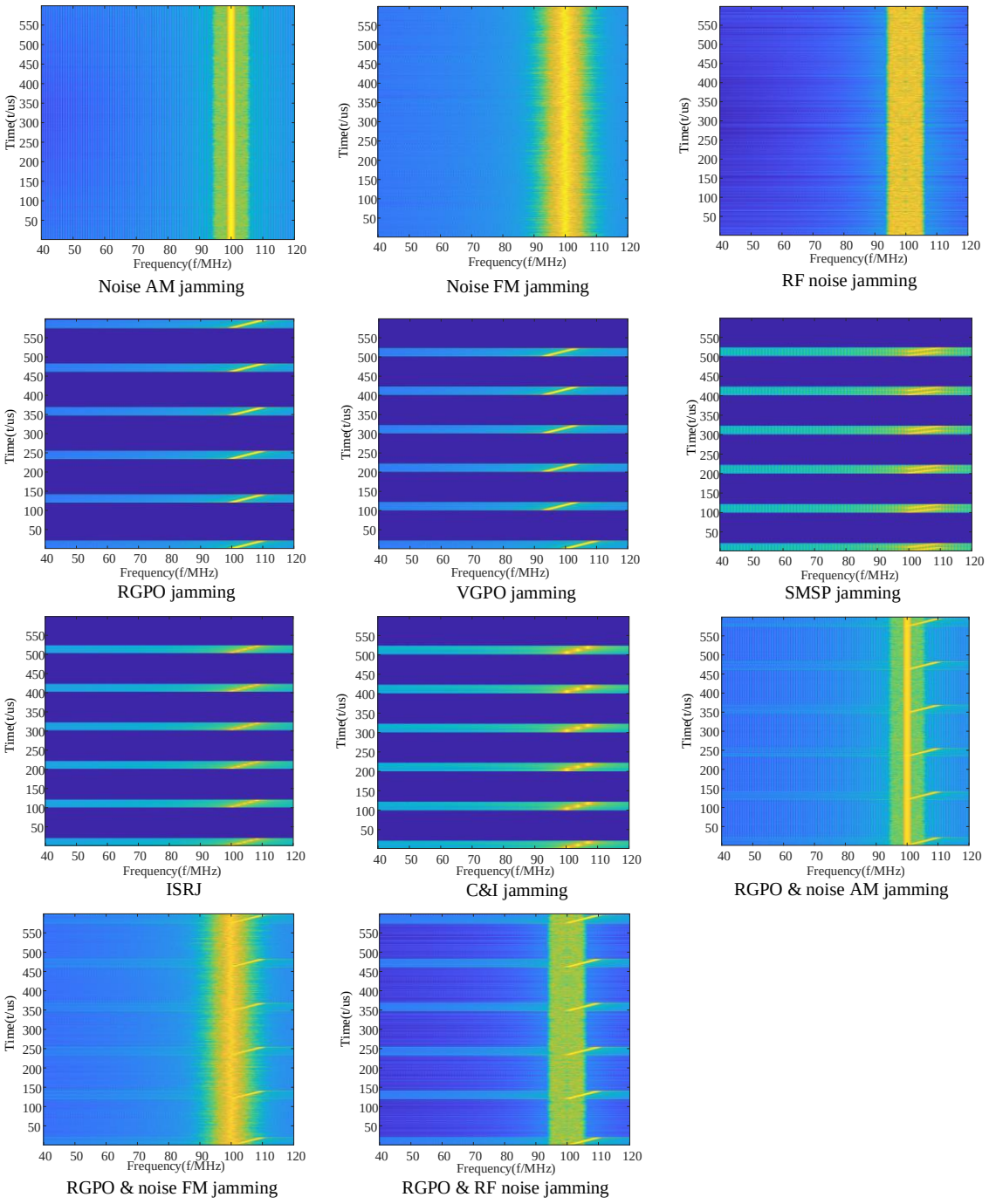


Figure 14. Time-frequency images of jamming signal.

Table 1. Radar active jamming dataset parameters settings.

Type of Jamming	Parameter Name	Parameter Range
Noise AM jamming	Carrier Frequency	80~120 MHz
	Bandwidth	2~10 MHz
Noise FM jamming	Carrier Frequency	80~120 MHz
	Bandwidth	2~10 MHz
	Chirp Rate	10 MHz/V
RF noise jamming	Carrier Frequency	80~120 MHz
	Bandwidth	2~10 MHz
RGPO jamming	Jamming to echo amplitude ratio	1.2~1.8
	Pull-off time	100~500 μs
VGPO jamming	Jamming to echo amplitude ratio	1.2~1.8
	Pull-off time	100~500 μs
SMSP jamming	Sampling Multiple	2~6
C&I jamming	Sub-pulse Number	2~10
	Retransmission Number	4~8
ISRJ	Sampling Period	4~10 μs
	Sampling Time	1~5 μs

4.2. Parameters of Pre-Training Networks

Due to the gradient elimination problem of deep network, if the number of layers of the network continues to increase, then the recognition performance of the model will degrade after increasing to a certain number of layers. The pre-training networks in this work will use the residual network structure [34] to address this problem. A batch normalization processing layer [35] is added to improve the training speed of the residual module and prevent it from overfitting after the convolution layer of the classical residual module.

We designed a pre-training network based on the improved residual module. The pre-training network consists of four parts. The first part is a convolution layer that has 64 convolution kernels with the size of 7×7. The second part includes four residual module groups [35], each of which has a convolution kernel size of 5×5; the number of residual modules is 3, 4, 6, 3, and the number of convolution kernels is 32, 64, 64, 128, respectively. The third part is a full connection layer. The fourth part is a SoftMax classification layer. The activation functions in the convolution layer and the residual module are the Rectified Linear Units [36]. The parameters of the pre-training networks are shown in Table 2.

Table 2. Parameters of pre-training networks.

Layer	Type	Kernel Size	Activation Function	Batch Normalization	Pooling Layer
1	Convolution layer	7×7,64	ReLU	Yes	Max Polling
2	Residual module	[5×5, 32]×3	ReLU	Yes	Null
3	Residual module	[5×5, 64]×4	ReLU	Yes	Null
4	Residual module	[5×5, 64]×6	ReLU	Yes	Null
5	Residual module	[5×5,128]×3	ReLU	Yes	Null
6	Full connection layer	Null	Null	Yes	Null
7	SoftMax layer	Null	Null	No	Null

4.3. Regularization Analysis of Network Parameter

The recognition performance of the two OSR models with and without regularization will be analyzed below. Regularization [37] is a method of preventing network models from overfitting by restricting the objective function. The two commonly used regularization methods are L1 and L2 regularization. Three groups of experiments are performed on two open set jamming recognition models. The first group is that two open set jamming recognition models do not use regularization, and the second group is that two open set jamming recognition models use L1 regularization. The third group is that two open set jamming recognition models use L2 regularization. The jamming data used in this experiment are 11 active radar jamming signals introduced in Section 2. Every jamming in the training set of each experiment consists of 100 samples, and that in the test set involves 100 samples. Both models use the same training and test sets for training and testing.

The accuracy, average precision, average recall, and average F-score [38] of the two open set jamming recognition models are calculated during the testing process. The recognition performance results of the two open set jamming recognition models before and after regularization are shown in Tables 3 and 4, respectively. In Table 3, the regularization improves the accuracy of the open set jamming recognition model based on the confidence score by about 2%, and the recognition performance of the L2 regularization is better. In Table 4, the regularization improves the accuracy of the OpenMax-based open set jamming recognition model by about 3%, of which L2 improves the recognition performance better. It can be proved that both the confidence score-based and OpenMax-based OSR models can obtain excellent performance in open set environment after L2 regularization, of which the accuracy is as high as 90%.

Table 3. The performance of the OSR model based on confidence score.

Regularization	Accuracy	Average Precision	Average Recall	Average F-Measure
Null	88.65%	87.36%	89.71%	88.43%
L1	89.72%	89.32%	89.77%	89.68%
L2	90.46%	90.22%	90.41%	90.33%

Table 4. The performance of the OSR model based on OpenMax.

Regularization	Accuracy	Average Precision	Average Recall	Average F-Measure
Null	88.36%	87.53%	89.91%	88.92%
L1	90.33%	89.34%	90.48%	90.12%
L2	91.10%	93.35%	91.10%	91.38%

4.4. Recognition Performance Analysis of OSR

Two open set validation experiments are designed to verify the recognition effect of the OSR model on unknown jamming, which are the performance of two OSR models with the number of unknown jamming types fixed and with the increase of unknown jamming types. In these experiments, we can find the models' ability to distinguish unknown jamming signals.

4.4.1. Experiment with the Number of Unknown Jamming Fixed

This experiment set up eight known jamming types, namely, noise AM jamming, noise FM jamming, RF noise jamming, RGPO jamming, VGPO jamming, SMSP jamming, C&I jamming, and ISRJ, and three unknown jamming types, including RGPO & noise AM jamming, RGPO & noise FM jamming, and RGPO & RF noise jamming. The jamming training set contains eight known jamming, and the jamming testing set contains eight

known and three unknown jamming. Existing closed-set jamming recognition models (CNN jamming recognition model), the open set jamming recognition model based on confidence score, and the open set jamming recognition model based on OpenMax are trained on the jamming training set and tested on the testing set. The recognition result of the first experiment is shown in Figures 15–17.

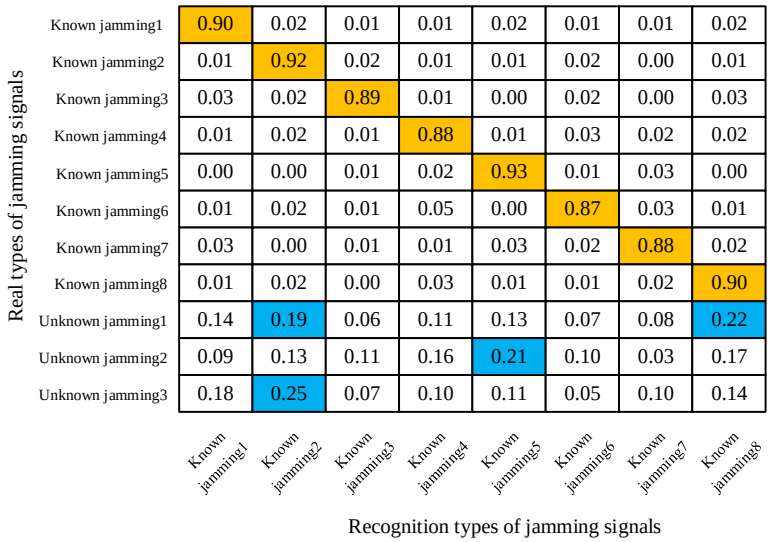


Figure 15. Recognition results of model based on close set CNN model.

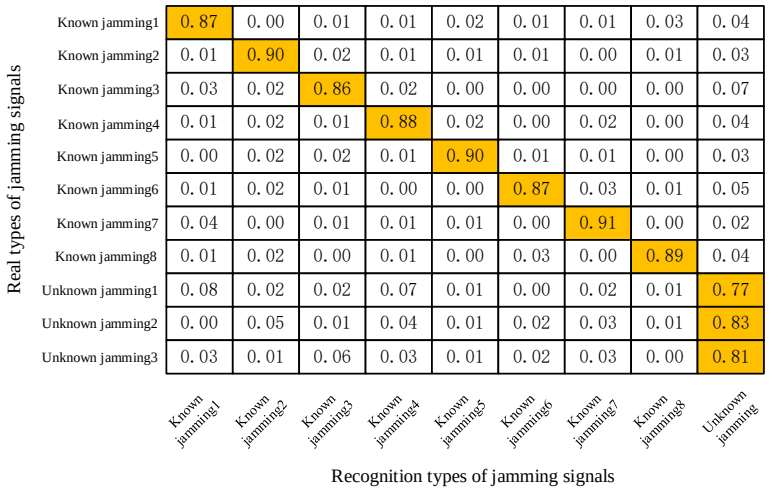


Figure 16. Recognition results of model based on confidence score.

Figure 15 shows the confusion matrix of the recognition result of the CNN model based on the close set. Figure 16 shows the confusion matrix of the recognition result of the model based on the confidence score. Figure 17 shows the confusion matrix of the recognition result of the model based on OpenMax.

In Figure 15, the CNN model based on a close set obtains good results on the recognition of known jamming types, and the average accuracy rate of the known types is 89.63%. However, this model does not efficiently perform if the samples belong to an unknown type and are recognized as one of the known jamming types. This notion indicates that the existing close set recognition model is not competent for jamming recognition in open set scenarios. Figure 16 shows that the OSR model based on the confidence score has a high average accuracy rate not only on the known jamming samples (88.5%) but also on the unknown jamming samples (80.33%). In Figure 17, the OSR model based on OpenMax

has better recognition results, whose average accuracy rate of known jamming samples is 92.13%; that of unknown jamming samples is 89.67%. It is proven in this experiment that both the confidence score-based and OpenMax-based OSR models have the ability to distinguish the unknown type of jamming signals.

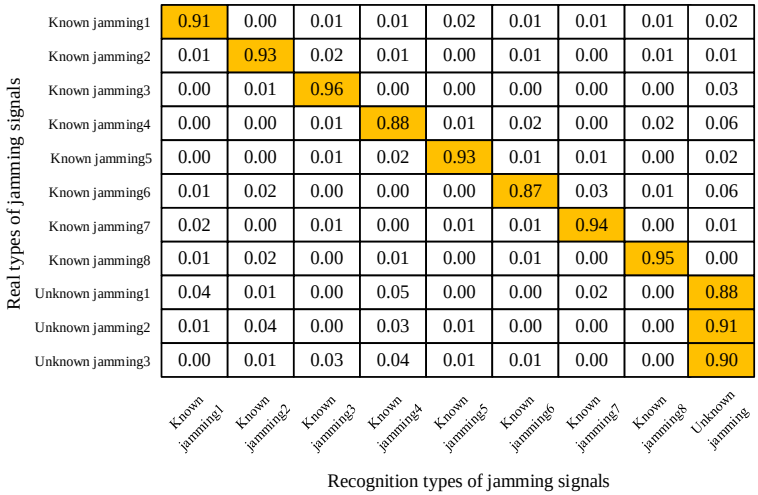


Figure 17. Recognition results of model based on OpenMax.

4.4.2. Experiment with the Unknown Jamming Type Increasing

The recognition performance of the two models will be tested in different open set environments to further verify the two jamming OSR models. Five open set jamming testing environments are set up. The numbers of unknown jamming types in open set jamming testing environments are one class, two classes, three classes, four classes, and five classes. The recognition results of two OSR models in different open set testing environments are shown in Figure 18. The recognition average accuracy rate of the OSR model based on the confidence score and the OSR model based on OpenMax gradually decreases with the number of unknown jamming types increasing in the open set jamming testing environment. Moreover, the average accuracy of these two models remains as high as nearly 85%, even though the number of unknown types is five. The OSR model based on OpenMax has better recognition performance than the open set jamming recognition model based on confidence scores under different open set jamming testing environments. It can be proved in this experiment that the two OSR models have good generalization performance in the open set environment with many unknown jamming types and they can maintain basic recognition ability (The average accuracy is above 85%) even though there are nearly 50% unknown jamming types (5 of the 11 jamming types are unknown) in the open set environment.

4.5. Robustness Analysis of OSR

To verify the stability of the jamming OSR model proposed in this work under different jamming-to-noise ratios (JNRs), we also perform an experiment to test the recognition performance of the models under different JNRs. In this experiment, the number of jamming samples in the training set is 100. The JNRs of the testing set are -4, -2, 0, 2, 4, 6, 8, 10, 12, and 14 dB.

The accuracy of the two OSR models under different JNRs is shown in Figure 19. The accuracy of the OSR model based on the confidence score and OpenMax is about 60% when the JNR is 0 dB. The recognition accuracy of the two OSR models increases with the increase in the JNR. The accuracy of the two models is basically the same when the JNR is lower than 0 dB. When the JNR rises to 4 dB, the accuracy of the model based on OpenMax is 10% higher than that of the model based on the confidence score. The accuracy of the two models is about 90% when the JNR reaches 8 dB. Moreover, the accuracy of these two

models remains approximately 90% with the increase in the JNR. Based on the results of Figure 19, the performance of the two OSR models differs slightly under different JNRs, with that of the open set jamming recognition model based on OpenMax slightly higher. Although the recognition performance will inevitably decline with the reduction of JNR, we can see from Figure 19 that when the JNR is more than 8dB, both models have higher average accuracy, which is as high as 90%.

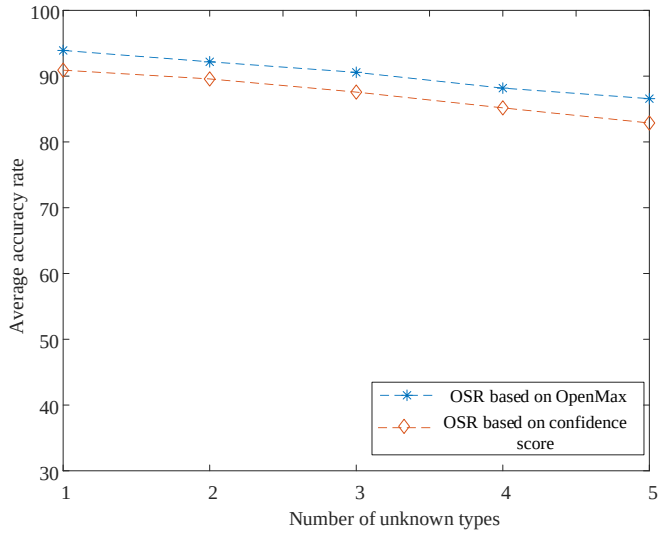


Figure 18. Recognition results of different open set environments.

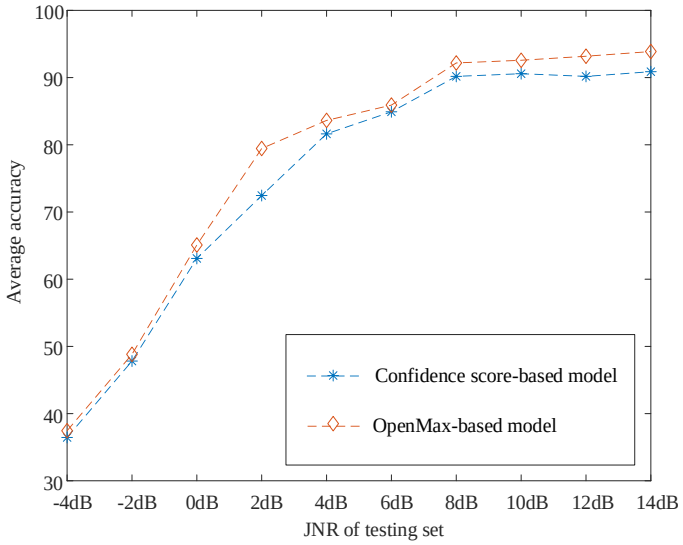


Figure 19. Average accuracy under different JNRs.

5. Conclusions

When the existing radar jamming recognition models are tested in an open test environment with unknown jamming, the existing jamming recognition models will recognize the unknown jamming as one known type in the training jamming library. This work discusses the jamming recognition models in open set scenarios to solve the problem existing in the jamming recognition methods of radar.

In this work, the jamming recognition of radar in open set scenarios is studied, and two solutions for open set jamming scenarios are presented, which are based on the confidence score and OpenMax OSR model. The OSR model based on the confidence score generates the probability distribution of the input jamming signals over N known

jamming types through the jamming classifier. Then, the confidence score of the probability distribution is calculated to determine whether the result of the jamming recognition is reliable. If the finding is reliable, then the result of the jamming classifier is outputted; otherwise, the output jamming is unknown. The OSR model based on OpenMax first extracts the activation vector from the input jamming signal to the jamming classifier and then constructs an OpenMax layer containing the probability that the input jamming sample is an unknown jamming type based on the activation vector of the jamming sample and the Weibull distribution of each known jamming boundary. Finally, the model selects the type with the highest probability in the probability distribution of the OpenMax output as the output of the recognition result.

After verification, the two OSR models can perform well in the open set jamming environment. The average accuracy is as high as 90% after regularization, and the OSR performance remains good with the increase of the unknown types in the testing set. Meanwhile, the two models also have good robustness. Thus, the unknown jamming types can be detected in a low JNR. The two models proposed in this paper need to be trained before the actual test. Among them, the training set is used to obtain the threshold in the confidence score-based model during the training process, while the boundary of Weibull distribution should be matched in the OpenMax-based model in advance. In the test, the algorithms in the two models have the same magnitude of time complexity.

Through the two open set jamming recognition models proposed in this paper, the radar can be competent for the task of correctly identifying the type of jamming signal in the open electromagnetic environment. This makes it easier for the radar anti-jamming system to implement anti-jamming measures for the specific type of jamming. The concept and technology of OSR can not only be used in radar jamming recognition, but also can be used in the recognition and positioning of civil radars and satellites, such as remote sensing intelligent interpretation, identity authentication, and pattern classification. However, current studies on OSR are not able to properly classify samples of unknown types into specific categories. Further studies can focus on how to solve this problem.

Author Contributions: S.S. and Y.Z. were involved in the computational framework, conceptualization, methodology, data analysis, results interpretation, and paper writing; S.Z. and X.S. simulated the experiments and processed the measured data; T.Y. was involved in methodology; L.Z. had the general supervision of the study and provided solutions to any problems that arose. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded in part by the National Natural Science Foundation of China under Grant 61871305, Grant 62171336, and Grant 61731023.

Acknowledgments: The authors would like to thank the anonymous Reviewers for their valuable comments, which improved the paper's quality.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhao, G. *Principle of Radar Countermeasure*, 2nd ed.; Xidian University Press: Xi'an, China, 2012.
2. Qu, Q.; Wei, S.; Liu, S.; Liang, J.; Shi, J. JRNet: Jamming Recognition Networks for Radar Compound Suppression Jamming Signals. *IEEE Trans. Veh. Technol.* **2020**, *69*, 15035–15045. [CrossRef]
3. Xun, Y.; Cui, Z.; Hao, X. Optimal Parameters Selection of Noise FM Jamming and Its Simulation. In Proceedings of the 2011 International Conference on Consumer Electronics, Communications and Networks (CECNet), Xianning, China, 16–18 April 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 2385–2389. [CrossRef]
4. Sparrow, M.J.; Cikalo, J. ECM Techniques to Counter Pulse Compression Radar. US, Issued 2006. Available online: https://wenku.baidu.com/view/fa1ec13f40323968011ca300a6c30c225901f0a6?fr=xueshu_top (accessed on 25 July 2006).
5. Lu, L.; Gao, M. An Improved Sliding Matched Filter Method for Interrupted Sampling Repeater Jamming Suppression Based on Jamming Reconstruction. *IEEE Sens. J.* **2022**, *22*, 9675–9684. [CrossRef]
6. Islam, S.M.M.; Lubecke, V.M. BreathID: Radar's New Role in Biometrics. *IEEE Aerosp. Electron. Syst. Mag.* **2021**, *36*, 16–23. [CrossRef]
7. Islam, S.M.M.; Boric-Lubecke, O.; Lubecke, V.M. Identity Authentication in Two-Subject Environments Using Microwave Doppler Radar and Machine Learning Classifiers. *IEEE Trans. Microw. Theory Tech.* **2022**, *70*, 5063–5076. [CrossRef]

8. Islam, S.; Borić-Lubecke, O.; Zheng, Y.; Lubecke, V. Radar-Based Non-Contact Continuous Identity Authentication. *Remote Sens.* **2020**, *12*, 2279. [CrossRef]
9. Li, Y.; Wei, X.; Li, Y.; Dong, Z.; Shahidehpour, M. Detection of False Data Injection Attacks in Smart Grid: A Secure Federated Deep Learning Approach. *IEEE Trans. Smart Grid* **2022**, *13*, 4862–4872. [CrossRef]
10. Li, Y.; Zhang, M.; Chen, C. A Deep-Learning intelligent system incorporating data augmentation for Short-Term voltage stability assessment of power systems. *Appl. Energy* **2022**, *308*, 118347. [CrossRef]
11. Xingyu, Y.; Huailin, R.; Haoran, F. A recognition algorithm of deception jamming based on image of time-frequency distribution. In Proceedings of the 2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC), Macau, China, 21–23 July 2017.
12. Lin, J.; Fan, X. Radar Active Jamming Recognition Based on Recurrence Plot and Convolutional Neural Network. In Proceedings of the 2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, 18–20 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1511–1515. [CrossRef]
13. Liu, Q.; Zhang, W. Deep Learning and Recognition of Radar Jamming Based on CNN. In Proceedings of the 2019 12th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 14–15 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 208–212. [CrossRef]
14. Lv, Q.; Quan, Y.; Feng, W.; Sha, M.; Dong, S.; Xing, M. Radar Deception Jamming Recognition Based on Weighted Ensemble CNN With Transfer Learning. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–11. [CrossRef]
15. Wang, K.; Dong, Z.; Wan, T.; Jiang, K.; Xiong, W.; Fang, X. Research on Radar Active Deception Jamming Identification Method Based on RESNET and Bispectrum Features. In Proceedings of the 2021 International Conference on Computer Engineering and Application (ICCEA), Kunming, China, 25–27 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 491–495. [CrossRef]
16. Scheirer, W.J.; Rocha, A.D.R.; Sapkota, A.; Boult, T.E. Toward Open Set Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 1757–1772. [CrossRef] [PubMed]
17. Scheirer, W.J.; Jain, L.P.; Boult, T.E. Probability Models for Open Set Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 2317–2324. [CrossRef] [PubMed]
18. Jain, L.P.; Scheirer, W.J.; Boult, T.E. Multi-Class Open Set Recognition Using Probability of Inclusion. In *Computer Vision—ECCV 2014*; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2014; pp. 393–409. [CrossRef]
19. Scherreik, M.D.; Rigling, B.D. Open set recognition for automatic target classification with rejection. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 632–642. [CrossRef]
20. Fei, G.; Liu, B. Social Media Text Classification under Negative Covariate Shift. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; Association for Computational Linguistics. pp. 2347–2356. [CrossRef]
21. Bendale, A.; Boult, T.E. Towards Open Set Deep Networks. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 1563–1572. [CrossRef]
22. Yang, H.-M.; Zhang, X.-Y.; Yin, F.; Yang, Q.; Liu, C.-L. Convolutional Prototype Network for Open Set Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 2358–2370. [CrossRef]
23. Prakhya, S.; Venkataram, V.; Kalita, J. Open Set Text Classification Using CNNs. In Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017), Kolkata, India, 18–21 December 2017; NLP Association of India. pp. 466–475. Available online: https://aclanthology.org/W17-7557 (accessed on 30 October 2022).
24. Cevikalp, H.; Triggs, B. Polyhedral Conic Classifiers for Visual Object Detection and Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 261–269. Available online: https://openaccess.thecvf.com/content_cvpr_2017/html/Cevikalp_Polyhedral_Conic_Classifiers_CVPR_2017_paper.html (accessed on 30 October 2022).
25. Chen, B. *Modern Radar System Analysis and Design*, 1st ed.; Xidian University Press: Xi'an, China, 2012.
26. Zhao, L.; Li, N.; Li, H.; Wang, R.; Li, M. BDS Satellite Clock Prediction Considering Periodic Variations. *Remote Sens.* **2021**, *13*, 4058. [CrossRef]
27. Gao, F.; Liu, Y.; Li, H. A Survey on Open Set Recognition. *J. Nanjing Univ. (Nat. Sci.)* **2022**, *58*, 115–134. [CrossRef]
28. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255. [CrossRef]
29. Xiao, Y.; Liu, W.; Gao, L. Radar Signal Recognition Based on Transfer Learning and Feature Fusion. *Mob. Netw. Appl.* **2019**, *25*, 1563–1571. [CrossRef]
30. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
31. Wen, L.-C.; Zhang, X.-F.; Zhu, L.-M. Method of Ameliorative Multi-Objective Synthetic Evaluation Based on Entropy Weight and Its Application. In Proceedings of the 2009 Chinese Control and Decision Conference, Guilin, China, 17–19 June 2009; pp. 1538–1541. [CrossRef]
32. Kun, L.; Liwei, J.; Xiaoming, S. An Efficient Hybridized Genetic Algorithm. In Proceedings of the 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), Chongqing, China, 10–12 December 2018; pp. 118–121. [CrossRef]
33. Kotz, S.; Nadarajah, S. *Extreme Value Distributions: Theory and Applications*, 1st ed.; River Edge: London, UK, 2000; ICP.

34. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. 2016, pp. 770–778. Available online: https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html (accessed on 27 June 2016).
35. Loffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456, PMLR. Available online: https://proceedings.mlr.press/v37/ioffe15.html (accessed on 11 February 2015).
36. Agarap, A.F. Deep Learning Using Rectified Linear Units (ReLU). *arXiv* **2019**, arXiv:1803.08375. [CrossRef]
37. Ni, J.-C.; Zhang, Q.; Su, L.-H.; Liang, J.; Huo, W.-J. L1/2 Regularization Sar Imaging Via Complex Image Data: Regularization Parameter Selection for Target Detection Task. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 2298–2301. [CrossRef]
38. Sarker, G.; Besra, M.; Dhua, S. A Malsburg Learning Back Propagation Combination for Handwritten Alpha Numeral Recognition. In Proceedings of the 2015 International Conference on Advances in Computer Engineering and Applications, Ghaziabad, India, 19–20 March 2015; pp. 493–4498. [CrossRef]