



*Research article*

## Open set recognition algorithm based on Conditional Gaussian Encoder

Yan Tang<sup>1,\*</sup>, Zhijin Zhao<sup>1,2</sup>, Chun Li<sup>1</sup> and Xueyi Ye<sup>1</sup>

<sup>1</sup> School of Communication Engineering, Hangzhou Dianzi University, Hangzhou, China

<sup>2</sup> State Key Lab of Information Control Technology in Communication System, The 36th Research Institute of China Electronics Technology Group Corporation, Jiaxing, China

\* **Correspondence:** Email: tangyan@hud.edu.cn; Tel: +86-187-8477-4994.

**Abstract:** For the existing Closed Set Recognition (CSR) methods mistakenly identify unknown jamming signals as a known class, a Conditional Gaussian Encoder (CG-Encoder) for 1-dimensional signal Open Set Recognition (OSR) is designed. The network retains the original form of the signal as much as possible and deep neural network is used to extract useful information. CG-Encoder adopts residual network structure and a new Kullback-Leibler (KL) divergence is defined. In the training phase, the known classes are approximated to different Gaussian distributions in the latent space and the discrimination between classes is increased to improve the recognition performance of the known classes. In the testing phase, a specific and effective OSR algorithm flow is designed. Simulation experiments are carried out on 9 jamming types. The results show that the CSR and OSR performance of CG-Encoder is better than that of the other three kinds of network structures. When the openness is the maximum, the open set average accuracy of CG-Encoder is more than 70%, which is about 30% higher than the worst algorithm, and about 20% higher than the better one. When the openness is the minimum, the average accuracy of OSR is more than 95%.

**Keywords:** Jamming recognition; Open Set Recognition (OSR); Conditional Gaussian Encoder (CG-Encoder); residual network; Kullback-Leibler (KL) divergence

---

## 1. Introduction

### 1.1. Background and motivation

In order to ensure the communication quality, corresponding anti-jamming measures should be adopted according to different jamming [1]. The anti-jamming effect depends on the accurate

identification of communication interference types, so the research of communication interference identification has important application value, which has attracted many researchers' attention. The current interference recognition methods include traditional and deep learning-based interference recognition algorithms.

The traditional interference recognition algorithm consists of feature extraction and pattern recognition. The algorithm process is generally as follows: first extract the signal features, such as signal high-order cumulant [2,3], signal space [4,5] and time-frequency domain analysis [6], and then use the pattern recognition algorithm to classify, including decision tree, support vector machine, back propagation neural network and other methods. The rationality of feature selection determines the recognition effect, and the feature selection is related to the cognition of professionals to interference, so human factors have great influence.

The key of interference recognition algorithm based on deep learning lies in data set, which avoids manual feature selection. The data set can be amplitude spectrum [7], IQ dual channel signal [7] and signal time-frequency diagram [8–11]. Literature [12] proved that convolution neural network (CNN) can extract features with good separability. This algorithm is better than the traditional interference recognition algorithm.

However, the electromagnetic environment is more and more complex, and new types of jamming are emerging. However, the existing algorithms can only recognize the known jamming types, that is, Closed Set Recognition (CSR). When a new type appears, the existing methods identify it as one of the known mistakenly, but cannot identify it as an unknown jamming accurately. Therefore, the Open Set Recognition (OSR) problem of communication jamming signal needs to be solved urgently.

In reference [13], OSR is defined as: the knowledge in training is incomplete, but the algorithm can detect and reject unknown classes samples in the process of testing. OSR methods include traditional machine learning-based methods and deep learning-based methods. The models of the former methods [14–17] cannot be applied to large-scale raw data without feature extraction, and the latter methods can not only overcome this problem, but also get better results. Open set recognition methods based on deep learning can be divided into two branches: discriminative model and generative model. At present, the proposed discriminative models are almost all for computer vision, text classification and other classification tasks, but cannot be directly applied to noisy signal recognition. The proposed generative models cannot get ideal effect when applied to OSR of noisy signals, because the generative models are reconstructed based on the training samples, when the training samples are noisy signals, the noise greatly affects the reconstruction effect. In order to cope with the OSR of noisy 1-dimensional jamming signals, an OSR network structure with Conditional Gaussian Encoder (CG-Encoder) is proposed, in this paper. CG-Encoder adopts the ResNet network structure [18] as a whole. After the convolution layer, it connects two parallel fully connected layers, which are used to learn the mean and variance of input to get their corresponding latent vectors. At the end, it is a SoftMax classifier. During training, the conditional posterior distributions approximate multiple multivariate Gaussian models, to enhance the discrimination of latent features between the classes and achieve better classification. Then, we obtain the Gaussian probability density thresholds by ensuring 98% training data to be recognized as known and the rest as unknown. When testing, the probability density function thresholds are used to judge whether test samples, which include known and unknown, are unknown. The results show that CG-Encoder algorithm achieves better effect in OSR and CSR recognition.

## 1.2. Related work

### 1.2.1. Discriminative model

Bendale et al. [19] proposed the OpenMax model and replaced the SoftMax activation layer in the neural network with the OpenMax layer, which is used to estimate the probability that input images come from unknown classes. This is the first solution for an open set deep network. Prakhya et al. [20] explored open set text categorization along the OpenMax model. Shu et al. [21] replaced SoftMax layer with 1-vs-rest layer, and proposed deep open classifier (DOC) model for text classification. Kardan et al. [22] proposed a COOL (Competitive Overcomplete Output Layer) neural network, and demonstrated the effectiveness of COOL by applying it to high-dimensional images. Dhamija et al. [23] solves the OSR problem by combining SoftMax with novel entropy open set and target ball loss. Shu et al. [24] proposed a joint open classification model to determine whether a pair of samples belong to the same class, where the sub-model can be used as a distance function of clustering to discover hidden classes in rejected samples. But these models cannot be directly applied to noisy signal recognition, which are suitable for computer vision, text classification and etc.

### 1.2.2. Generative model

Different from the discriminative model, the generative method uses GAN [25], auto-encoder [26] and flow-based model [27] to generate unknown or known samples to help the classifier learn the decision boundary between known and unknown samples. Ge et al. [28] proposed the G-OpenMax algorithm, which is a direct extension of OpenMax, using conditional generative to synthesize unknown classes. This algorithm provides explicit probability estimates of the generative unknown classes, enabling the classifier to locate decision margins based on the knowledge of known classes and generative unknown classes. Unlike G-OpenMax, Neal et al. [29] introduced a new data set enhancement technique called OSRCI, which uses the VAEGAN architecture to generate synthetic open set examples that are close to but not part of any known class. Similar to [29], Jo et al. [30] used GAN technology to generate pseudo data as unknown class data to further enhance the robustness of unknown class classifier. Yoshihashi et al. [31] proposed a Classification-Reconstruction Open Set Recognition (CROSR), which uses latent representations to reconstruct, enabling unknown class detection robustly without compromising the classification accuracy of known classes. Oza and Patel [32] proposed a C2AE model using a class conditional auto-encoder with novel training and testing methods, which uses class conditional auto-encoder to derive the decision boundary from EVT reconstruction errors. Variational Auto-Encoder (VAE) [33] is combined with clustering [34], one class [35] or Gaussian mixed model (GMM) [36] algorithm for OSR. The posterior distribution  $q_{\phi}(\mathbf{z}|\mathbf{x})$  in latent space is trained to approximate a prior distribution  $p_{\theta}(\mathbf{z})$ , which enables VAE to correctly describe the known data, and the deviated samples will be identified as unknown. Xin et al. [26] provided VAE with a kind of conditional Gaussian distribution learning, which can detect unknown and classify known samples by forcing different latent features to approach different Gaussian models. Zhang et al. [27] proposed a joint embedded space consisting of a classifier and a flow-based density estimator. But these generative models cannot get ideal effect of OSR of noisy signals.

However, the CG-Encoder we propose in this paper not only classifies known jamming types, but also detects an unknown jamming accurately.

### 1.3. Contributions and structure of the paper

The contributions of this paper are mainly as follows:

- To our knowledge, we are the first to study open set recognition of communication interference signals.
- We propose a new classification model called CG-Encoder. Compared with previous methods based on convolution neural network, the proposed method not only achieves better classification results, but also can be used for unknown detection.
- We found a novel unknown detection method based on probability density function. The proposed algorithm is superior to other detection methods for unknown signals.
- We conduct experiments on nine common classes of communication jamming, and the results show that our method outperform existing methods and achieve new state-of-the-art performance.

The rest of the paper is organized as follows. Section 2 introduces briefly Variational Auto-Encoder (VAE) and Deep Residual Structure. Section 3 discusses Open Set Recognition Algorithm Based on CG-Encoder in detail. Finally, Section 4 gives the algorithm simulations and performance analysis in detail. Finally, Section 5 concludes the paper.

## 2. Basic principles

### 2.1. Variational auto-encoder (VAE)

VAE [33] is generally composed of two neural networks: encoder and decoder. The parameters, input and output of the encoder are  $\phi$ , sample  $\mathbf{x}$  and latent representation  $\mathbf{z}$ , respectively. The parameters, input and output of the decoder are  $\theta$ ,  $\mathbf{z}$  and the probability distribution of samples. The loss function of VAE is as follows:

$$L(\theta, \phi, \mathbf{x}) = -D_{KL}(q_{\phi}(\mathbf{z} | \mathbf{x}) || p_{\theta}(\mathbf{z})) + E_{q_{\phi}(\mathbf{z} | \mathbf{x})}[\log p_{\theta}(\mathbf{x} | \mathbf{z})] \quad (1)$$

where  $D_{KL}(q_{\phi}(\mathbf{z} | \mathbf{x}) || p_{\theta}(\mathbf{z}))$  is the KL-divergence between the approximate posterior distribution  $q_{\phi}(\mathbf{z} | \mathbf{x})$  and the prior distribution  $p_{\theta}(\mathbf{z})$  and  $E_{q_{\phi}(\mathbf{z} | \mathbf{x})}[\log p_{\theta}(\mathbf{x} | \mathbf{z})]$  represents the reconstruction error.

In general,  $p_{\theta}(\mathbf{z})$  is multivariate standard normal Gaussian, so  $q_{\phi}(\mathbf{z} | \mathbf{x})$  is a multivariate Gaussian distribution with diagonal covariance matrix:

$$q_{\phi}(\mathbf{z} | \mathbf{x}) = N(\mathbf{z}; \boldsymbol{\mu}, \sigma^2 \mathbf{I}) \quad (2)$$

where the mean  $\boldsymbol{\mu}$  and the standard deviation  $\boldsymbol{\sigma}$  are the encoding multilayered perceptrons' (MLPs) outputs.  $\mathbf{z}$  is defined as:

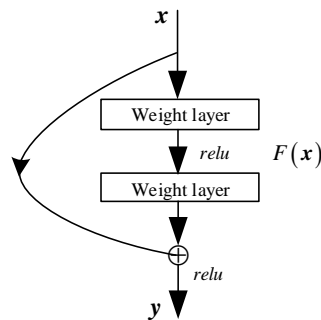
$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \bullet \boldsymbol{\xi} \quad (3)$$

where  $\boldsymbol{\xi} \sim N(\mathbf{0}, \mathbf{I})$ ,  $\bullet$  is the element-wise product. The KL-divergence [18] can be calculated:

$$\begin{aligned} L_{KL} &= -D_{KL}(q_{\phi}(\mathbf{z} | \mathbf{x}) \| p_{\theta}(\mathbf{z})) \\ &= \frac{1}{2} \sum_{j=1}^J (1 + \log(\sigma_j^2) - \mu_j^2 - \sigma_j^2) \end{aligned} \quad (4)$$

where  $J$  is the dimensionality of  $\mathbf{z}$ . By minimizing  $L(\boldsymbol{\theta}, \phi, \mathbf{x})$ , the VAE is trained not only to reconstruct the input accurately, but also to force  $q_{\phi}(\mathbf{z} | \mathbf{x})$  in latent space to approximate  $p_{\theta}(\mathbf{z})$ .

## 2.2. Deep residual structure



**Figure 1.** Residual basic block.

ResNet [18] is a deep residual structure, which is constructed from the basic block shown in Figure 1, it is defined as:

$$\mathbf{y} = \text{relu}(F(\mathbf{W}, \mathbf{x}) + \mathbf{x}) \quad (5)$$

where  $\mathbf{x}$  and  $\mathbf{y}$  are the input and output vectors, the function  $F(\mathbf{W}, \mathbf{x})$  represents the residual map to be learned, and  $\text{relu}$  is one of the nonlinear operations. The structure in Figure 1 has two layers, then  $F(\mathbf{W}, \mathbf{x}) = \mathbf{W}_2 * \text{relu}(\mathbf{W}_1 * \mathbf{x})$ , where the bias term is ignored to simplify the representation.

The dimensions of  $\mathbf{x}$  and  $F(\mathbf{W}, \mathbf{x})$  must be equal in Eq. (5), which can be matched by linear projection  $\mathbf{W}_s$ .

$$\mathbf{y} = \text{relu}(F(\mathbf{W}, \mathbf{x}) + \mathbf{W}_s \mathbf{x}) \quad (6)$$

The form of residual function  $F$  is variable, and the trunk of basic block can stack more layers. For the sake of simplicity, the above symbol is about the fully connected layer. In fact, the function  $F(\mathbf{W}, \mathbf{x})$  can represent multiple convolution layers, and the elements are added channel by channel on two feature maps.

### 3. Open set recognition algorithm based on CG-Encoder

In the communication jamming recognition identification, the input sample is 1-dimensional jamming signal  $\mathbf{x} = \mathbf{x}_0 + \text{noise}$ ,  $\mathbf{x}_0$  is the jamming signal with a sampling length of  $l$ , and noise is Gaussian white noise with the same length. If the inputs are reconstructed by the usual VAE model, this will be affected by the noise, and the reconstruction loss cannot be used as the condition of unknown detection. Therefore, this paper only uses the encoder network to learn the latent feature distribution of the classes, and judges whether the test samples are known or unknown by their probability density value, so as to realize the unknown detection.

#### 3.1. Design of the CG-Encoder structure

As shown in Figure 2, the structural block diagram of the jamming signal OSR method (CG-Encoder) consists of three modules. Encoder, Classifier, and Detector.

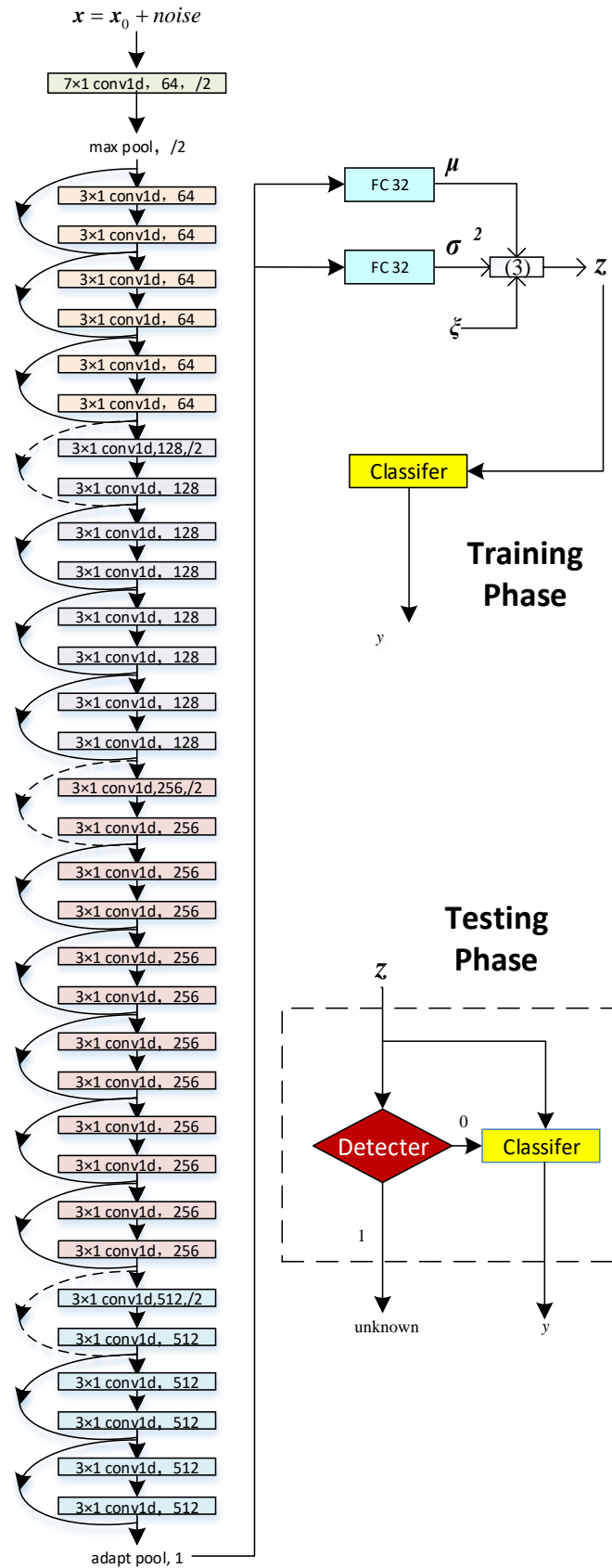
Encoder is a 1-dimensional residual network, which consists of 33 1-dimensional convolution layers (including 16 basic residual blocks), two 1-dimensional pooling layers and two fully connected layers. Its input is  $\mathbf{x}$ ; The outputs are the mean  $\boldsymbol{\mu}$  and variance  $\boldsymbol{\sigma}^2$  obtained by the two parallel fully connected layers respectively. The nonlinear function softplus is used to ensure that all components of variance are greater than 0.

The input and output dimensions of the residual blocks of the solid shortcut in the Figure 2 are the same, and Eq. (5) is used to calculate output. And the dimensions are not same in the dotted shortcut, and the input and output should map linearly using Eq. (6).

The convolution layer parameters meanings are convolution kernel size, type of convolution layer, number of convolution kernel, and change of the sequence length through that layer. For example, the first layer parameters are  $\{7 \times 1 \text{ conv1d}, 64, /2\}$ , which mean that the layer uses 1-dimensional convolution (conv1d) with convolution kernel size of  $7 \times 1$ , the number of convolution kernel is 64, and the jamming sequence length after the layer is shortened by half. The pooling layers are max pool and adapt pool respectively. The former reduces the signal sequence length by half after, and the latter can accept inputs of any length sequence and make the output length fixed, here set a fixed value as 1.

Classifier is a fully connected layer with SoftMax as the activation function. Its input is  $\mathbf{z}$  obtained by Eq. (3) and its output is a known class label.

Detector is modeled by information hidden in the latent representation  $\mathbf{z}$ . During testing, the detector is viewed as a binary classifier. When output is 1,  $\mathbf{x}$  is recognized as unknown jamming, and when output is 0,  $\mathbf{x}$  is recognized as class  $y$ .



**Figure 2.** CG-Encoder structure.

### 3.2. Closed set training phase

In the training phase,  $q_\phi(\mathbf{z} | \mathbf{x}, k)$  are forced to approximate multiple multivariate Gaussian distributions  $p_{\theta,k}(\mathbf{z}) = N(\mathbf{z}; \boldsymbol{\mu}_k, \mathbf{I})$  for the above proposed model, where  $k$  is the index of a known class.  $\boldsymbol{\mu}_k$  is output of a fully connected layer and represents the mean vector of the  $k$ -th class Gaussian distribution. The KL-divergence (Eq. (4)) is modified as follows:

$$\begin{aligned} L_{KL} &= -D_{KL}(q_\phi(\mathbf{z} | \mathbf{x}, k) \| p_{\theta,k}(\mathbf{z})) \\ &= \int q_\phi(\mathbf{z} | \mathbf{x}, k) (\log p_{\theta,k}(\mathbf{z}) - \log q_\phi(\mathbf{z} | \mathbf{x}, k)) d\mathbf{z} \\ &= \int N(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2) (\log N(\mathbf{z}; \boldsymbol{\mu}_k, \mathbf{I}) - \log N(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2)) d\mathbf{z} \\ &= \frac{1}{2} \sum_{j=1}^J \left( 1 + \log(\boldsymbol{\sigma}_j^2) - (\boldsymbol{\mu}_j - \boldsymbol{\mu}_{j,k})^2 - \boldsymbol{\sigma}_j^2 \right) \end{aligned} \quad (7)$$

CG-Encoder has no decoder compared with VAE, so the loss function discards the reconstruction error in Eq. (1) and adds the classification loss

$$L_c = -\frac{1}{num} \sum_{i=1}^{num} \log \frac{e^{\mathbf{W}_{y_i}^T \mathbf{z}_i + b_{y_i}}}{\sum_{j=1}^K e^{\mathbf{W}_j^T \mathbf{z}_i + b_j}} \quad (8)$$

where  $num$  is the batch size,  $K$  is the number of known classes,  $\mathbf{z}_i$  is the feature of the  $i$ -th sample,  $y_i$  is the class label corresponding to  $\mathbf{x}_i$ , and  $\mathbf{W}_j$  and  $b_j$  are the weight and bias of class  $j$ .

The loss function of CG-Encoder is

$$L = L_{KL} + \lambda L_c \quad (9)$$

where  $\lambda$  is a constant. The parameters of CG-Encoder are optimized by minimizing the loss function  $L$ , and the training method is consistent with the common closed set training method. During training, the latent vector  $\mathbf{z}$  of correctly classified training set samples is saved for later open set testing to use.

### 3.3. Open set testing phase

#### 3.3.1. Establishment of multivariate gaussian model

According to the class labels of training samples, the latent vector  $\mathbf{z}$  is divided into  $K$  sets, namely  $\{\mathbf{z}_1\}$ ,  $\{\mathbf{z}_2\}$ , ...,  $\{\mathbf{z}_K\}$ , each set contains only one class latent representation. The mean vector and covariance matrix of  $K$  kinds of multivariate Gaussian distribution models can be obtained from

$$\boldsymbol{\mu}_k = \frac{1}{m_k} \sum_{i=1}^{m_k} \mathbf{z}_k^{(i)}, k = 1, \dots, K \quad (10)$$

$$\boldsymbol{\Sigma}_k = \frac{1}{m_k} \sum_{i=1}^{m_k} (\mathbf{z}_k^{(i)} - \boldsymbol{\mu}_k) (\mathbf{z}_k^{(i)} - \boldsymbol{\mu}_k)^T, k = 1, \dots, K \quad (11)$$



where  $m_k$  is the number of samples in  $\{z_k\}$ . Furthermore, the probability density function of each kind of jamming signal multivariate Gaussian distribution model can be obtained as follows

$$f_k(z_k) = N(z_k; \mu_k, \Sigma_k), k = 1, \dots, K$$

$$= \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_k|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} (z_k^{(i)} - \mu_k)^T \Sigma_k^{-1} (z_k^{(i)} - \mu_k)\right) \quad (12)$$

where  $n$  is the dimension of latent space.

### 3.3.2. Threshold setting

Because the signal distribution can provide effective information for unknown detection, according to Eq. (12), the probability density values of all latent vectors in  $K$  sets  $\{z_1\}, \{z_2\}, \dots, \{z_K\}$ , namely  $\{p_1\}, \{p_2\}, \dots, \{p_K\}$  are calculated and arranged in descent in each set. In a manner similar to Reference [26], the threshold  $\varepsilon_k$  is set to less than the probability density of the first 98% and greater the probability density of the last 2%.

### 3.3.3. Open set test algorithm

The specific steps of the algorithm are as follows:

- Calculate the latent space distribution model  $f_k(z_k)$  of each known class according to Eqs. (10) – (12).
- According to section 3.3.2, set the threshold  $\varepsilon_k$  of each known class.
- Input the test jamming sample  $x_t$  to the trained encoder for obtaining its latent vector  $z_t$ .
- The probability density values of the latent vector  $z_t$  in various Gaussian models  $f_k(z_t) = N(z_t; \mu_k, \Sigma_k), k = 1, \dots, K$  calculated by Eq. (12).
- If  $f_k(z_t) < \varepsilon_k$ , the Detector detects  $x_t$  as unknown, otherwise, it gets its class  $y$  through Classifier.

## 4. Algorithm simulation and performance analysis

The Adam optimizer with initial learning rate of 0.001 is used, and the batch size is fixed to 256; the dimension  $n$  of latent representation  $z$  is 32, parameter  $\lambda$  is set to 100.

### 4.1. Datasets

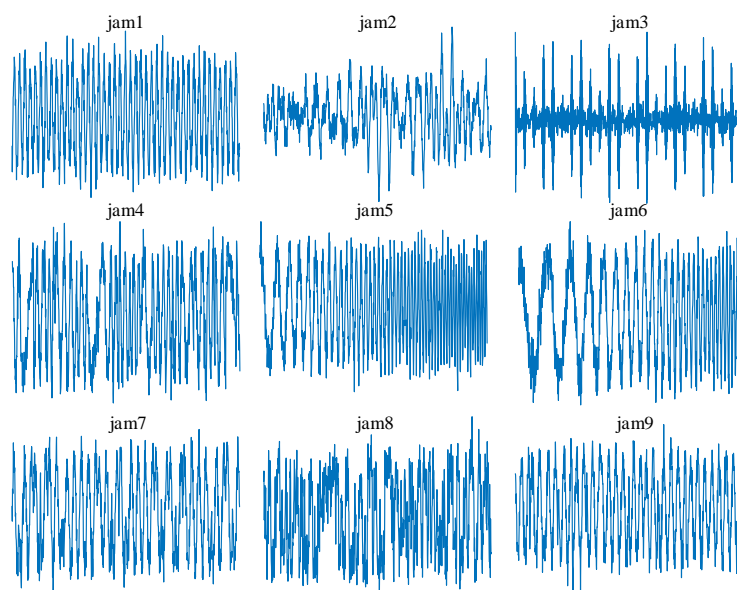
In order to test the performance of the proposed OSR method, simulation experiments are carried out on 9 kinds of jamming signals, including single-tone jamming, multi-tone jamming, periodic Gaussian pulse jamming, frequency hopping jamming, linear sweeping frequency jamming, second sweeping frequency jamming, BPSK modulation jamming, noise frequency modulation jamming and QPSK modulation jamming. The range of jamming-to-noise rate (JNR) is -10~18dB,

with a value taken every 2dB. The additive noise is Gaussian white noise in the signal band. The sampling frequency is 10MHz, the number of sampling points is  $l$ , the size of jamming sample is expressed as  $1 \times l$ , and the parameters of each jamming type are shown in Table 1.

Figure 3 shows the time domain waveforms of the above 9 jamming signals randomly generated when JNR = 10dB and  $l = 1024$ .

**Table 1.** Parameters setting of jamming.

Jamming types	Corresponding label	Parameters Setting
single-tone	jam1	The center frequency $f_c$ is between [100,400] kHz, and the phase $\varphi$ is between $[0, 2\pi]$ .
Multi-tone	jam2	The number N of audio is [2,10], and $f_c$ and $\varphi$ are the same as jam1.
periodic Gaussian pulse	jam3	The pulse period T is $2.5 \sim 10 \mu s$ , and the duty cycle is $1/8 \sim 1/2$
frequency hopping	jam4	$N = 20$ , $\{f_c\}$ is between [100,400] kHz, the frequency hopping period $T_H$ is between [3.2, 6.4] $\mu s$ , and the phase is between $[0, 2\pi]$ .
linear sweeping frequency	jam5	The starting frequency $f_{c1}$ is [50,100] kHz, and the ending frequency $f_{c2}$ is [300,1000] kHz.
second sweeping frequency	jam6	The frequency is quadratic, and other parameters are the same as jam5.
BPSK modulation	jam7	The information symbol is a 32-bits 0,1 random sequence, the symbol period is $3.2 \mu s$ , and the modulation signal is sinusoidal signal.
noise frequency modulation	jam8	The frequency modulation coefficient is between 0.125 and 0.933, and the carrier signal parameters are the same as jam1.
QPSK modulation	jam9	The information symbol is a 32-bit 0,1 random sequence, the symbol period is $3.2 \mu s$ , I-channel modulation signal is sinusoidal signal, Q-channel modulation signal is cosine signal.



**Figure 3.** Time domain waveforms of 9 jamming signals.

#### 4.2. Performance analysis of CSR and OSR

The performance of CSR and OSR of CG-Encoder algorithm and the following three algorithms with JNR of -10 ~ 18dB is simulated and analyzed.

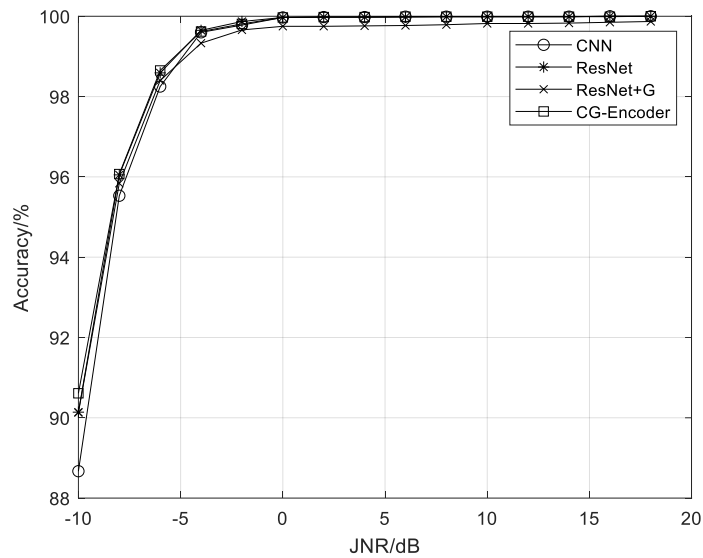
**(1) CNN [12].** The network structure of this algorithm is similar to CG-Encoder, the difference is that there is no shortcut, and only one fully connected layer is connected after convolution layers to get the latent vector  $z$ . The threshold of unknown detection is the confidence that makes 98% of the correctly classified training samples known. If the confidence of test sample is greater than the threshold, it is known. If the confidence of test sample is less than the threshold, it is unknown. The model can be regarded as a traditional CNN.

**(2) ResNet [18].** The network structure of this algorithm is similar to CG-Encoder, but only one fully connected layer is connected after convolution layers to get the latent vector  $z$ . Unknown detection algorithm is the same as CNN. This model can be regarded as a common ResNet structure.

**(3) ResNet+G [26].** The network structure of this algorithm is similar to CG-Encoder, the difference is that the posterior distribution of all classes approximates a single multivariate Gaussian distribution. The open set testing phase is the same as section 3.3.3. This model can be regarded as that ResNet learning a multivariate Gaussian model, named ResNet+G.

##### 4.2.1. CSR performance analysis

CSR is usually recognition for known classes, without using unknown detector. Set the number of sampling points to 1024. The training set classes include jam1 ~ jam8, each class has 2000 samples under each JNR, a total of 240000. The testing set classes are also jam1 ~ jam8, with 2000 samples for each category under each JNR. In this paper, the accuracy is used to measure the performance of the algorithm. The experimental results of the four algorithms are shown in Figure 4.



**Figure 4.** Closed set recognition accuracy of four algorithms.

It can be seen from Figure 4 that the closed set recognition accuracy of the four algorithms increases with the increase of JNR. When  $JNR > -10\text{dB}$ , the accuracy is higher than 88%, and the accuracy is close to 1 when  $JNR > 0\text{dB}$ , so the recognition performance of four networks for known classes is better. Under the low JNR, CNN performance is slightly inferior to the other three networks. Under the high JNR, ResNet+G performance is slightly inferior to the other three networks. ResNet performance is better than CNN, which shows that shortcut in residual structure can improve recognition performance of known classes. ResNet performance is better than ResNet+G, which indicates that the difference between classes will be reduced by the approximation of posterior distribution to a single Gaussian model. CG-Encoder performance is equivalent to ResNet, which illustrates that the latent distribution of different classes approximates different Gaussian models, which can improve the performance of CSR.

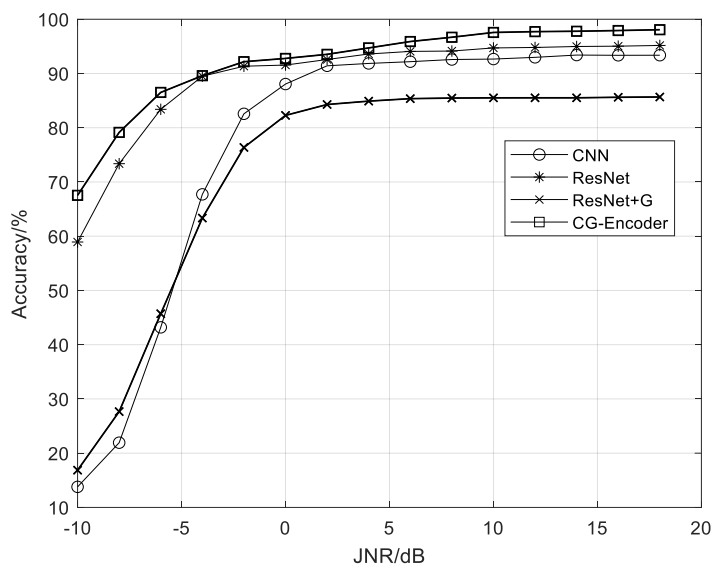
#### 4.2.2. OSR performance analysis

The training set of OSR is consistent with CSR, and jam9 is added to the testing set as the unknown class to verify the unknown detection performance of the four algorithms. The experimental results of OSR are shown in Figure 5.

In the case of OSR, the accuracy of open set recognition increases with the increase of JNR. The CG-Encoder algorithm has the best performance, which proves the OSR effectiveness of the algorithm for noisy jamming signals. When  $JNR = -10 \sim 0\text{dB}$ , the accuracy is low, which indicates that noise has a great influence on OSR performance. When  $JNR > 5\text{dB}$ , the change of accuracy is small, and the performance of each algorithm is stable.

When  $JNR > 0\text{dB}$ , the performance of OSR of network structure is  $\text{CG-Encoder} > \text{ResNet} > \text{CNN} > \text{ResNet+G}$ , and CG-Encoder is about 2%, 4% and 10% higher than the average accuracy of other three algorithms respectively.  $\text{CG-Encoder} > \text{ResNet+G}$  shows that the latent distribution of different classes approximates different Gaussian models, which not only makes the known classes more separable, but also improves the division between known and unknown classes.  $\text{ResNet} > \text{CNN}$

shows that the shortcut method improves the accuracy of known classes, and also indirectly benefits the performance of OSR.  $\text{CNN} > \text{ResNet+G}$  indicates that when all the latent distributions of all classes belong to one distribution, the unknown class will approach the distribution. Even if residual structure is adopted, the performance of ResNet+G will not be better than that of ordinary CNN network.

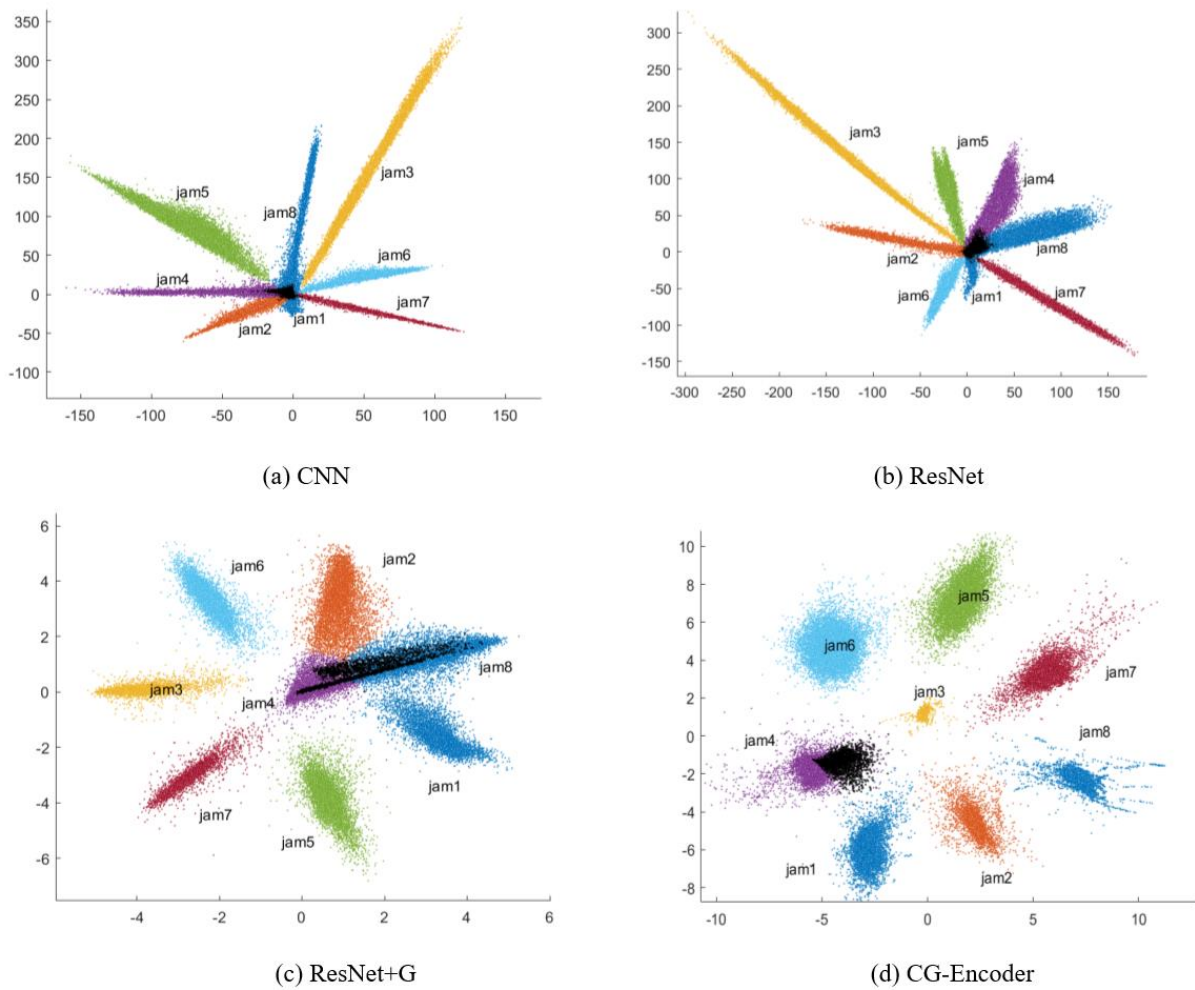


**Figure 5.** Open set recognition accuracy of four algorithms.

#### 4.3. Visual analysis of latent space features

In order to better observe the latent space features of the samples, the dimension of latent representation  $z$  is set to 2, and four kinds of algorithm network models are retrained to visualize the latent space that each network learned on the 2-dimensional plane, as shown in Figure 6.

Figure 6 (a), (b), (c) and (d) is the 2-dimensional distribution of latent feature space learned by CNN, ResNet, ResNet+G and CG-Encoder algorithms respectively. Each point represents a sample, in which clusters of known classes are labeled at their corresponding positions, and the unknown class jam9 is represented by black cluster. In fact, the blank area is other unknown classes. It can be seen from Figure (a) and (b) that CNN and ResNet map the features of unknown class to the overlapping place of all known class, which are relatively far away from the center of each known class. As can be seen from Figure (c), ResNet+G network makes the unknown class almost coincide with jam4 and jam8, which is difficult to distinguish. As can be seen from Figure (d), CG-Encoder algorithm completely separates the known classes, whose effect the former three algorithms cannot achieve. Although the unknown class are close to jam4, they only overlap a little, and the Detector can effectively detect the unknown jamming.



**Figure 6.** Visualization of 2D latent space.

#### 4.4. Openness of OSR

Openness is related to the number of training classes  $N_{train}$  and the number of test class  $N_{test}$ .

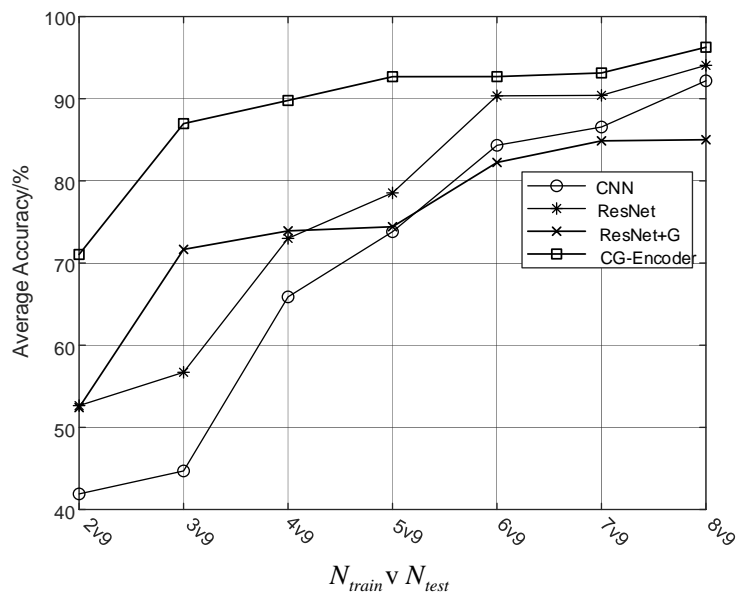
The formula is given in Reference [13].  $O = 1 - \sqrt{\frac{2 \times N_{train}}{N_{train} + N_{test}}}$ . In the experiment of this section,

$N_{train} = 2 \sim 8$ ,  $N_{test} = 9$ . And it means that the unknown class contains  $1 \sim 7$  different classes.

According to the formula of openness, the larger  $N_{train}$ , the smaller  $O$ , the less unknown information.

Figure 5 shows that the OSR performance of the four algorithms is relatively stable when JNR > 0dB, so the average accuracy between JNR = 0 ~ 18dB is used to analyze the openness of the four algorithms. As shown in Figure 7, the horizontal axis represents the degree of openness. In order to

be more intuitive,  $N_{train} \vee N_{test}$  is used instead of its corresponding  $O$  value. On the whole, the OSR performance of the four algorithms increases with the increase of the number of known class, indicating that the less unknown information, the better the OSR performance. The CG-Encoder algorithm proposed in this paper has the best recognition effect under different openness. When  $N_{train} = 2$ 、 $N_{test} = 9$ , the OSR average accuracy of CG-Encoder algorithm is more than 70%, which is about 30% and 20% higher than CNN and ResNet+G, respectively. When the openness is the minimum, the OSR average accuracy of CG-Encoder algorithm can reach more than 95%.



**Figure 7.** Openness of OSR.

It can also be concluded from Figure 7, when  $N_{train} \leq 4$ , the recognition performance of ResNet+G is better than that of CNN and ResNet algorithm, while when  $N_{train} \geq 5$ , the recognition performance of ResNet+G is worse than that of ordinary CNN algorithm, which indicates that the more the number of known classes is, the more confusion between classes caused by using posterior distribution to approximate a single distribution will be, and the more features of each class need to be learned.

## 5. Conclusion

In order to solve the problem that the existing jamming signal recognition algorithms mistakenly recognize the unknown class as a known class with a certain probability, a CG-Encoder network structure suitable for 1-dimensional signal OSR is constructed based on ResNet and multivariate Gaussian model. This paper not only defines a reasonable loss function for the training of the network, but also designs a specific OSR process. For nine types of jamming, simulation experiments are carried out under JNR= -10 ~ 18dB. The CSR and OSR performance of CNN,

ResNet, ResNet+G and CG-Encoder network algorithms are compared. The feature learning ability of each algorithm is further compared by visualizing the latent space, and the algorithm openness on OSR is analyzed. The results show that CG-Encoder can achieve more than 98% CSR effect when JNR is -6dB, its OSR performance is better than the other three networks, and the OSR accuracy can reach more than 95% when the openness is the smallest.

## Acknowledgment

This work was supported by the National Natural Science Foundations of China under grant nos. U19B2016, and Zhejiang Provincial Key Lab of Data Storage and Transmission Technology, Hangzhou Dianzi University.

## Conflict of Interest

The authors declare there is no conflict of interest.

## References

1. F. Q. Yao, Communication anti-jamming engineering and practice, *Beijing Publishing House Electron. Industry*, (2008), 1–8.
2. Y. Y. Wen, J. Y. Wei, H. Chen, A new algorithm of interferences signals recognition, *Space Electron. Technol.*, **1** (2015), 85–88.
3. J. X. Wang, Q. Chang, Y. Tian, J. Huang, Research on GNSS interference signal detection method, *Navig. Position. Tim.*, **4** (2020), 117–122.
4. G. S. Wang, Q. H. Ren, Z. G. Jang, Y. Liu, B. Z. Xu, Jamming classification and recognition in transform domain communication system based on signal feature space, *Syst. Eng. Electron.*, **39** (2017), 1950–1958.
5. G. C. Huang, G. S. Wang, Q. H. Ren, S. F. Dong, W. T. Gao, S. Wei, Adaptive recognition method for unknown interference based on Hilbert signal space, *J. Electron. Inform. Technol.*, **41** (2017), 1916–1923.
6. J. Y. Liu, Research on electronic jamming identification method based on time frequency domain analysis, *University Electron. Sci. Technol. China*, 2018.
7. G. J. Xun, Research on identification of typical communication jamming signals, *University Electron. Sci. Technol. China*, 2018.
8. Q. Liu, W. Zhang, Deep learning and recognition of radar jamming based on CNN, *2019 12th International Symposium on Computational Intelligence and Design (ISCID), IEEE*, **1** (2019), 208–212.
9. T. F. Chi, Recognition algorithm for the four kinds of interference signals, *Huazhong University Sci. Technol.*, 2019.
10. Z. B. Zhang, Y. X. Fan, X. Meng, Pattern recognition method of communication interference based on power spectrum density and neural network, *J. Terahertz Sci. Electron. Inform. Technol.*, **17** (2019), 959–963.



11. Y. Cai, K. Shi, F. Song, Y. F. Xu, X. M. Wang, H. Y. Luan, Jamming pattern recognition using spectrum waterfall. a deep learning method, *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, IEEE, (2019), 2113–2117.
12. Z. L. Wu, Y. L. Zhao, Z. D. Yin, H. C. Luo, Jamming signals classification using convolutional neural network, *2017 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, IEEE, (2017), 062–067.
13. W. J. Scheirer, A. R. Rocha, A. Sapkota, T. E. Boult, Towards open set recognition, *IEEE Transact. Pattern Anal. Mach. Intell.*, **35** (2013), 1757–1772.
14. M. D. Scherreik, B. D. Rigling, Open set recognition for automatic target classification with rejection, *IEEE Transact. Aerosp. Electron. Systems*, **52** (2016), 632–642.
15. P. R. M. Jnior, R. M. D. Souza, R. D. O. Werneck, B. V. Stein, D.V. Pazinato, W. R. Almeida, et al, Nearest neighbors distance ratio open-set classifier, *Mach. Learn.*, **106** (2017), 359–386.
16. E. M. Rudd, L. P. Jain, W. J. Scheirer, T. E. Boult, The extreme value machine, *IEEE Transact. Pattern Anal. Mach. Intell.*, **40** (2018) ,762–768.
17. E. Vignotto, S. Engelke, Extreme value theory for open set classification GPD and GEV classifiers, *arXiv preprint*, *arXiv:1808.09902*, 2018.
18. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2016), 770–778.
19. A Bendale, T. E. Boult, Towards open set deep networks, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2016), 1563–1572.
20. S. Prakhya, V. Venkataram, J. Kalita, Open set text classification using convolutional neural networks, *International Conference on Natural Language Processing*, 2017.
21. L. Shu, H. Xu, B. Liu, DOC: Deep open classification of text documents, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, (2017), 2911–2916.
22. N. Kardan, K. O. Stanley, Mitigating fooling with competitive overcomplete output layer neural networks, *International Joint Conference on Neural Networks (IJCNN)*, (2017), 518–525.
23. A. R. Dhamija, M. Günther, T. Boult, Reducing network agnostophobia, *Advances in Neural Information Processing Systems*, (2018), 9157–9168.
24. L. Shu, H. Xu, B. Liu, Unseen class discovery in open-world classification, *arXiv preprint*, *arXiv:1801.05609*, 2018.
25. I. Goodfellow, J. P. Abadie, M. Mirza, B. Xu, D. W. Farley, S. Ozair, et al., Generative adversarial nets, *Adv. Neural Inform. Process. Systems*, (2014), 2672–2680.
26. X. Sun, Z. N. Yang, C. Zhang, Xin Sun, K. V. Ling, G. H. Peng, Conditional gaussian distribution learning for open set recognition, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2020),13480–13489.
27. H. J. Zhang, A. Li, J. Guo, Y. W. Guo, Hybrid models for open set recognition, *Proceedings of European Conference on Computer Vision*, (2020),102–117.
28. Z. Y. Ge, S. Demyanov, Z. Chen, R. Garnavi, Generative OpenMax for multi-class open set classification. *British Machine Vision Conference 2017*, British Machine Vision Association and Society for Pattern Recognition, 2017.
29. L. Neal, M. Olson, X. Fern, W. K. Wong, F. X. Li, Open set learning with counterfactual images, *Proceedings of the European Conference on Computer Vision (ECCV)*, (2018), 613–628.

30. I. Jo, J. Kim, H. Kang, Y. D. Kim, S. Choi, Open set recognition by regularising classifier with fake data generated by generative adversarial networks, *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (2018), 2686–2690.
31. R. Yoshihashi, W. Shao, R. Kawakami, S. D. You, M. Iida, T. Naemura, Classification-reconstruction learning for open-set recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. (2019), 4016–4025.
32. P. Oza, V. M. Patel, C2ae: Class conditioned auto-encoder for open-set recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2019), 2307–2316.
33. D. P. Kingma, M. Welling, Auto-encoding variational bayes, *arXiv: Machine Learning*, 2013.
34. C. Aytekin, X. Ni, F. Cricri, E. Aks, Clustering and unsupervised anomaly detection with l2 normalized deep auto-encoder representations, *2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil*, (2018), 1–6.
35. L. Ruff, R. Vandermeulen, N. Goernitz, P. Liznerski, M. Kloft, K. R. Müller, Deep one-class classification, *International Conference on Machine Learning, PMLR*, (2018), 4393–4402.
36. B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, et al, Deep autoencoding Gaussian mixture model for unsupervised anomaly detection, *International Conference on Learning Representations*, 2018.



AIMS Press

©2021 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>).