Pulses Classification Based on Sparse Auto-Encoders Neural Networks

KAN REN [ID], HONGLIANG YE, GUOHUA GU, AND QIAN CHEN

Jiangsu Key Laboratory of Spectral Imaging and Intelligent Sense, Nanjing University of Science and Technology, Nanjing 210094, China

Corresponding author: Kan Ren (k.ren@njust.edu.cn)

ABSTRACT Unsupervised learning is applicable to classification that does not know the number of specific categories in advance, and sparse auto-encoders (SAE) are widely used for feature extraction of unsupervised learning. Therefore, this paper proposes an electromagnetic signal classification system based on SAE which is combined with the machine learning clustering algorithm. In particular, we propose to perform feature preprocessing on signals using STFT. Then, the features extracted by SAE training are clustered by t-SNE and DBSCAN to obtain clustering results. Finally, we prove the feasibility of this method classification by comparing with traditional clustering methods. Because of the feature extraction, SAE not only learns the key feature information but also effectively compresses the data content, which greatly reduces the data dimension that the clustering algorithm needs to deal with and improves the clustering accuracy. As the experimental results show, the evaluation indicators of the result obtained by our method are significantly improved compared with the traditional clustering algorithms, the compactness (CP) index decreases by 73.76%; the Davies-Bouldin Index (DB) decreases by 18.50%; the Dunn Validity Index (DVI) increases by 6.24%; and the Rand Index (RI) increases by 43.14%.

INDEX TERMS Machine learning sparse auto-encoders STFT, t-SNE, DBSCAN.

I. INTRODUCTION

In modern life, especially military intelligence, the use of electromagnetic targets to transmit information is becoming more and more frequent. As the number of transmitted signals increases and the content is different, there are many techniques to classify the received electromagnetic signals. In the existing classification algorithms, the most convenient and faster way is to use machine learning methods [1]–[3], such as support vector machines (SVM) [4]–[6], k-means [7]–[9], Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [10], [11], artificial neural networks [12]–[15], etc.

Most previous methods for pulses classification are more inclined to know the total number of categories in advance, and then set the number of output categories for classification. Zhang et al. [16] proposed a novel edit distance with real penalty (ERP)-based k-nearest neighbors (KNN) classifier. Dai et al. [17] proposed a modulation classification method using stacked sparse auto-encoders combined with softmax

The associate editor coordinating the review of this manuscript and approving it for publication was Shihao Yan.

regression classifier. Jeong et al. [34] proposed a new deep belief network model to generate a more efficient threat library for radar signal classification. However, in actual situations, there are many kinds of pulses received, and it is difficult to know the exact number of categories. The method proposed in this paper can be better applied to these actual situations.

An important step before clustering is feature extraction, a better feature extraction algorithm can make clustering more accurate. Among the many methods of feature extraction, neural networks are the most popular algorithms at present, and SAE (Sparse Auto-Encoders) is a simple and efficient neural network that can automatically learn features from unlabeled data [18]. SAE is based on AE (Auto-Encoders), introducing sparse regularization in vector of hidden layer or output layer [19]. SAE only needs one middle hidden layer, and it can get the deep features of input data after continuously training. Another commonly used neural network for classification is CNN [20]. However, CNN has a large amount of computation due to the use of a large number of convolution operations. Compared with CNN, SAE only uses the fully connected layer. It only needs

to calculate the weight and gradient, and the calculation amount and training time are less than CNN. CNN not only needs a suitable pre-training model, but is also commonly used for image classification [21]. Our proposed method transforms the original image into a time-frequency map using STFT, and then transforms the time-frequency map into a one-dimensional matrix for SAE training, reducing the amount of computation.

Although k-means [22] is a commonly used clustering algorithm, it needs to set the k value in advance, and the optimal k value is difficult to select. But DBSCAN does not need to set the number of classifications in advance like k-means, the classification results can be adjusted according to our own needs. Therefore, this article combines DBSCAN and neural networks used to extract features for classification. In summary, we have two contributions in this article: (1) We combine SAE neural networks with traditional clustering algorithms and improve the accuracy of unsupervised clustering, (2) We conduct comparing experiments on other method [33] to prove the advantages of our method.
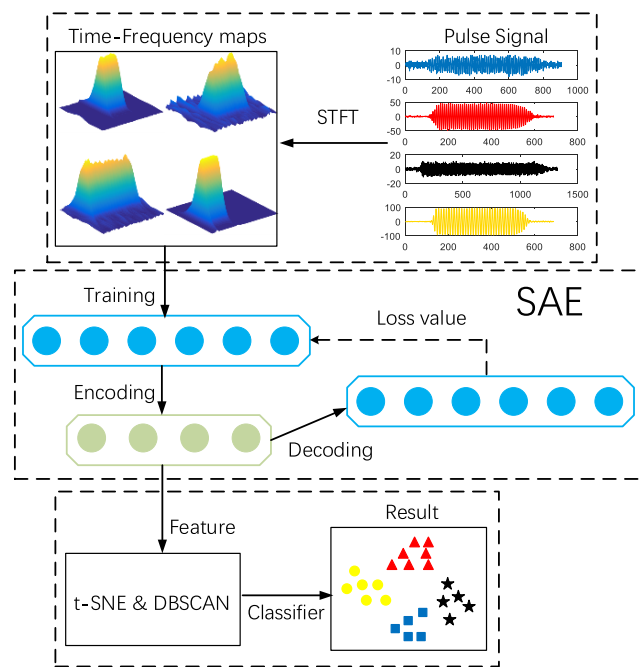

FIGURE 1. The framework of the method for clustering.

The paper proposes a pulses classification method based on machine learning. As shown in Fig.1, first we propose a fuzzy features preprocessing method using STFT (short-time Fourier transform), and the original signals are converted into time-frequency signals. Then we send the time-frequency signals to SAE to learn the deep features. After training, the feature data of the hidden layer are extracted. Finally, we reduce the dimensionality of the features and visualize them by using t-SNE, and DBSCAN is used for clustering. Since the specific number of pulse classes is not known in advance, the proposed method is an unsupervised training process.

The effectiveness of the method is confirmed by observing the pulses in the same category. This method can greatly reduce the workload of manual classification and improve work efficiency.

The rest of the paper is organized as follows. Section II introduces the algorithms used in the paper, including SAE, t-SNE and DBSCAN. In section III, we introduce the experimental steps and evaluation indicators. In section IV, we show the results of the classification experiment and comparison with the existing methods. Section V concludes the study.

II. PROBLEM FORMULATION
A. SHORT-TIME FOURIER TRANSFORM
In fact, the most important difference between different types of radar pulses is their frequency variation with time [21]. So we use STFT to convert the raw pulses into a time-frequency signals before SAE training. Through the feature preprocessing of STFT, we can get better deep features in SAE training.

The Windowed Fourier Transform of a function is defined in [23] as:

$$S_x(t,f) = \int_{-\infty}^{+\infty} x(\tau)w(\tau - t, f)\exp(-j2\pi f\tau)d\tau \quad (1)$$

Which satisfies $\int_{-\infty}^{+\infty} w(\tau - t, f)\,d\tau = 1$ for $\forall$ frequency f. Fig.2 shows the time-frequency transform results using different window functions with the same length of window function. We can see that different window functions have little effect on our experimental results. However, the first side-lobe relative to main-lobe attenuation, the main lobe width and the minimum attenuation of the stop band of the Kaiser window are adjustable. So in order to better adjust the results, we select the Kaiser window as a window-function, and $w(t,f)$ is defined as

$$w_K(t,f) = \frac{I_0\left(\alpha(f)\sqrt{1-t^2}\right)}{I_0(\alpha(f))} \quad (2)$$

In this formula, the $I_0(\cdot)$ is the zero-order Bessel function of the first type, and $\alpha(f)$ is a frequency dependent parameter [24], where $\alpha(f)$ is proportional to the frequency f:

$$\alpha(f) \sim \beta f \quad (3)$$

and β is a constant.

In this paper, the length of window function is 60 and the β that affects the side-lobe of the window function is 5. The number of overlapping points when calculating the spectrum is 41, and the number of FFT points is 256. The sampling frequency defaults to 1Hz.

B. SPARSE AUTO-ENCODER
In this section, we briefly introduce the structure of SAE neural network and its internal mathematical operations. As shown in Fig. 3, the basic structure of SAE is divided into three parts: encoder, middle hidden layer and decoder.
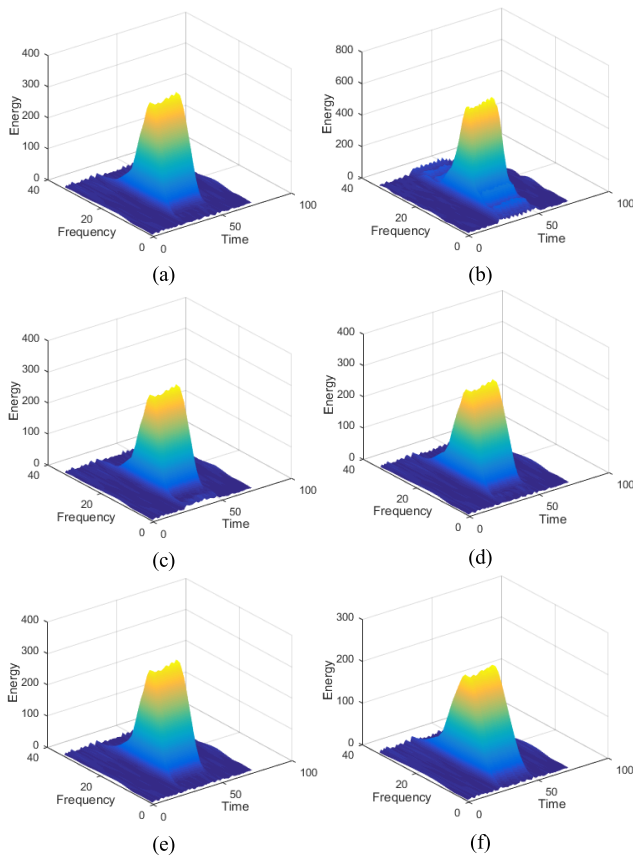
FIGURE 2. Time-frequency images by using Kaiser window (a); Rectangular window (b); Triangular window (c); Hanning window (d); Hamming window (e); Chebyshev window (f).
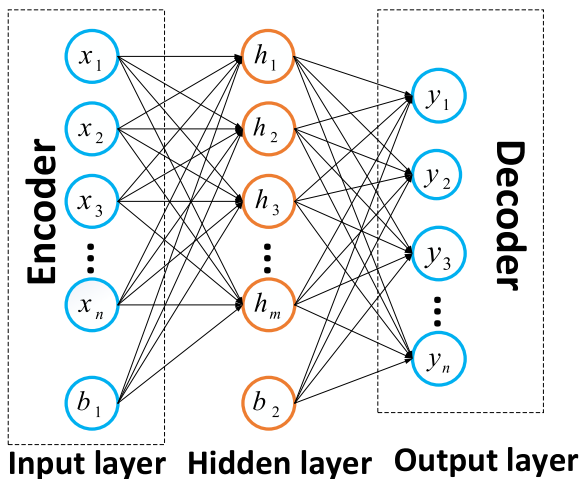

FIGURE 3. SAE neural network structure.

The training goal is to achieve the reconstruction of the input signal, then the hidden middle layers can be regarded as the learned feature.

SAE allows the network to convert a set of quantities into a set of additional quantities through learning. This set of quantities can in turn be restored to the original data by decoding.

It indicates that the outputs of the middle layer are another expression of the original input.

From the structure chart we can see that the neural network is an arithmetic unit with input values of $x_1, x_2, x_3 \ldots\ldots x_n$ and bias units. The encoder function, $h = f(Wx + b)$, maps the input layer x to the hidden layer h, and the decoder function, $y = f(Wh + b)$, is used to recover x from h [18], where $f: R \rightarrow R$ is called the activation function. In this paper, we choose $f(\cdot)$ to be the sigmoid function:

$$f(z) = \frac{1}{1 + \exp(-z)} \quad (4)$$

And W is the connection parameter called weight matrix between the two different layers.

For a single sample (x, y), the cost function can be defined as:

$$J(W, b; x, y) = \frac{1}{2} \left\| h_{W,b}(x) - y \right\|^2 \quad (5)$$

Given a data set containing m samples, we can define the overall cost function as

$$J(W, b) = \left[\frac{1}{m} \sum_{i=1}^{m} J\left(W, b; x^{(i)}, y^{(i)}\right) \right]$$
$$+ \frac{\lambda}{2} \sum_{l=1}^{n_l-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} \left(W_{ji}^{(l)} \right)^2$$
$$= \left[\frac{1}{m} \sum_{i=1}^{m} \left(\frac{1}{2} \left\| h_{W,b}\left(x^{(i)}\right) - y^{(i)} \right\|^2 \right) \right]$$
$$+ \frac{\lambda}{2} \sum_{l=1}^{n_l-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} \left(W_{ji}^{(l)} \right)^2 \quad (6)$$

The first item in the above definition of $J(W, b)$ is a mean squared term. The second item is a regularization item (also called weight decay term) that tends to reduce the magnitude of weight and prevent overfitting. And we define the loss value as:

$$L = \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{1}{2} \left(y_j^{(i)} - x_j^{(i)} \right)^2 \quad (7)$$

Suppose that $a_i(x^{(j)})$ represents the activation of hidden unit when the network is given a special input $x^{(j)}$. Further, let $\hat{\rho}_i = \frac{1}{m} \sum_{j=1}^{m} a_i(x^{(j)})$ be the average activation of hidden unit i [17]. We would like to enforce the constraint $\hat{\rho}_i = \rho$, ρ is a sparsity parameter. To achieve this constraint, we will add an additional penalty factor to our optimization objective function, and the penalty factor is as follows [25]:

$$\sum_{i=1}^{m} KL(\rho \| \hat{\rho}_i) = \sum_{i=1}^{m} \rho \log \frac{\rho}{\hat{\rho}_i} + (1 - \rho) \log \frac{1 - \rho}{1 - \hat{\rho}_i} \quad (8)$$

Here, m is the number of hidden neurons in the hidden layer, and index i represents each neuron in the hidden layer.

So the cost function of SAE is defined as:

$$
\begin{aligned}
J_{sparse}(W,b) &= \left[\frac{1}{m} \sum_{i=1}^{m} \left(\frac{1}{2} \left\| h_{W,b}\left(x^{(i)}\right) - y^{(i)} \right\|^2 \right) \right] \\
&+ \frac{\lambda}{2} \sum_{l=1}^{n_l-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} \left(W_{ji}^{(l)} \right)^2 + \beta \sum_{i=1}^{m} KL\left(\rho \parallel \hat{\rho}_i \right)
\end{aligned} \quad (9)
$$

C. VISUALIZATION ALGORITHM

1) DIMENSIONALITY REDUCTION MAPPING

The feature data obtained by SAE training is high-dimensional data, and the amount of clustering calculation is large, so it is necessary to reduce the dimension. The t-distributed Stochastic Neighbor Embedding (t-SNE) is a very popular algorithm for dimensionality reduction of high-dimensional data, which was proposed by Laurens van der Maaten and Geoffrey Hinton in 2008 [26]. This algorithm has been widely used in the field of machine learning because it can effectively convert high-dimensional data into two-dimensional images [27], [28].

t-SNE (t-distributed stochastic neighbor embedding) is a machine learning algorithm for dimensionality reduction. It is a non-linear dimensionality reduction algorithm, which is very suitable for the reduction of high-dimensional data to 2D or 3D for visualization. t-SNE adds two improvements to SNE: one is to transform SNE into a symmetric SNE, the other is to use t distribution instead of the original Gaussian distribution in the low-dimensional space, and the high-dimensional space does not change.

t-SNE converts the distance relationship between high-dimensional data to a conditional probability to represent similarity. If there are two points x_i and x_j in the high-dimensional space, then we use Gaussian distributions to define their conditional probabilities as [26]:

$$
p_{ij} = \frac{\exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma_i^2} \right)}{\sum_{k \neq i} \exp\left(-\frac{\|x_i - x_k\|^2}{2\sigma_i^2} \right)} \quad (10)
$$

And The conditional probabilities of the low-dimensional space are defined by the t-probability distribution as follow [26]:

$$
q_{ij} = \frac{\left(1 + \|y_i - y_j\|^2 \right)^{-1}}{\sum_{k \neq l} \left(1 + \|y_k - y_l\|^2 \right)^{-1}} \quad (11)
$$

Then the cost function is determined by minimizing the Kullback-Leibler divergence between the joint distributions p and q [29]:

$$
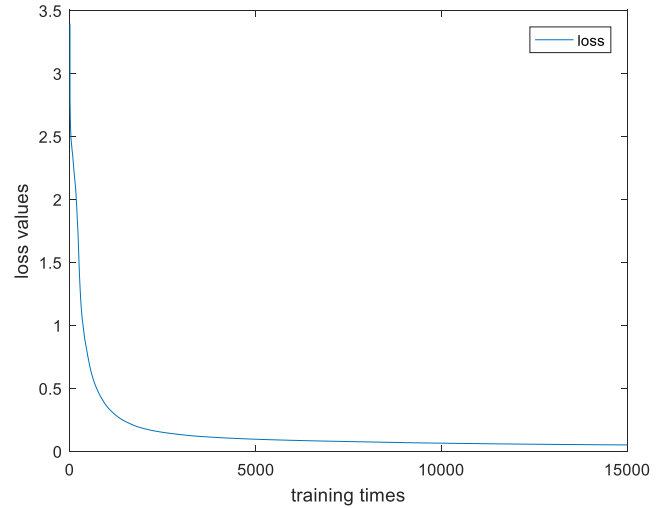C(\varepsilon) = KL(P \parallel Q) = \sum_{i=j} p_{ij} \, log \, \frac{p_{ij}}{q_{ij}} \quad (12)
$$

FIGURE 4. Loss values curve.

And the result of cost function for the gradient of y_i is as follows:

$$
\frac{\delta C}{\delta y_i} = 4 \sum_{j} \left(p_{ij} - q_{ij} \right) \left(y_i - y_j \right) \left(1 + \|y_i - y_j\|^2 \right)^{-1} \quad (13)
$$

2) CLUSTER ANALYSIS

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based spatial clustering algorithm. The algorithm divides regions of sufficient density into clusters and can find clusters of arbitrary shape in a spatial database with noise, it defines the cluster as the largest set of points connected by density [30].

DBSCAN has the following advantages:

(1) Fast clustering, efficient processing of noise points and finding spatial clustering of arbitrary shapes;

(2) Compared with k-means, there is no need to input the number of clusters to be divided;

(3) The shape of the cluster is not biased;

(4) The parameters of filter noise can be entered when needed.

The DBSCAN algorithm has two very important parameters: Eps and MinPts. Eps represents the radius, and the neighborhood within a given point radius Eps is called the Eps-neighborhood of the point. MinPts represents the minimum number of neighbor points that a given point becomes the core object in the neighborhood.

DBSCAN searches for clusters by checking the Eps-neighborhood of each point in the data set. If the Eps-neighborhood of point p contains more points than MinPts, create a cluster with p as the core object; then, DBSCAN iteratively aggregates objects that are directly reachable from these core objects, this process may involve the consolidation of some density-reachable clusters. The process ends when no new points are added to any of the clusters.
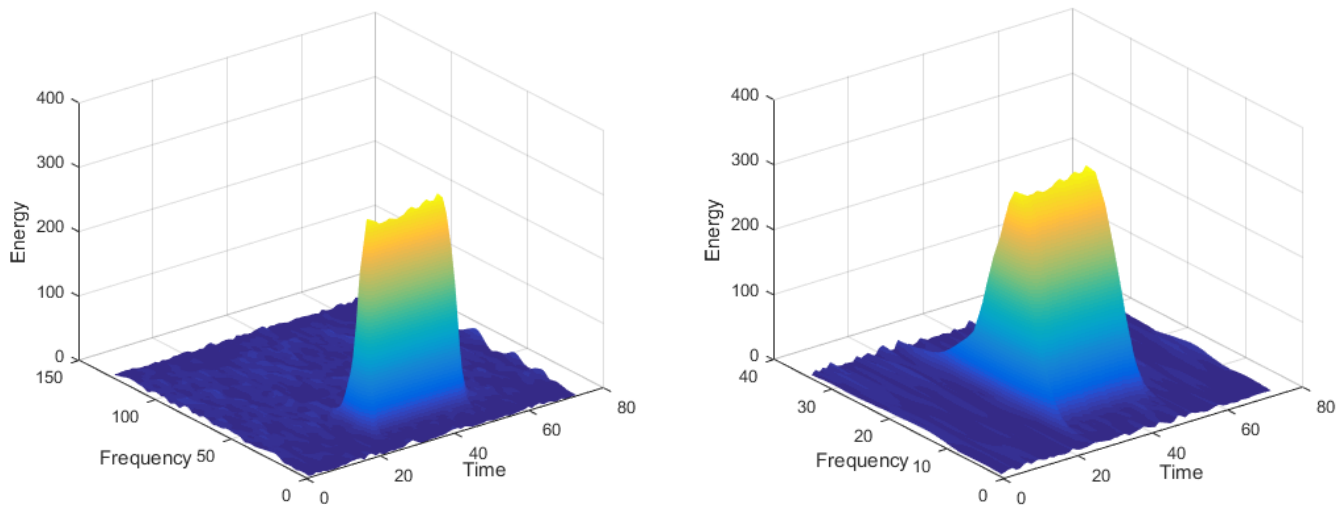
FIGURE 5. Before processing (a) and after processing (b).

III. EXPERIMENTS

In this section, we introduce the experimental parameter settings and evaluation indicators. We have separated 3155 pulse signals from the existing IF (Intermediate Frequency) radar signals. Our goal is to classify these pulse signals by unsupervised training. First, we blur the time-frequency maps of the initial signal to obtain more accurate and useful feature information by sacrificing the image resolution. We design a SAE model in the second part, and we show some representative results of unsupervised training. Finally, we introduce five indexes to evaluate our clustering method.

All experiments were carried out on MATLAB 2015b of window 10 operating system using a 64-bit computer with 4.2GHz Intel core i7-7700K CPU and 8 GB RAM.

A. FUZZY PROCESSION

Since the resolution of the time-frequency maps generated by the original pulse signal is too high, the amount of data is very large, and the training time of SAE will increase accordingly. So we reduce the resolution by intercepting useful feature information parts. After processing, important signal features are preserved while reducing the resolution.

B. PARAMETER SETTINGS

In order to reduce the computational complexity of SAE, we reshape the two-dimensional time-frequency data of a single pulse into one-dimensional matrix data, and the length of each one-dimensional matrix is 2482. Then we process the values of the matrix elements using the normalization method. We select several hidden layer values for comparison, by considering the loss value and the training time, we finally select 350 hidden layer neurons, and we only use one hidden layer. So the architecture of the neural network model is [2482 350 2482]. The epochs of training are 15000 to minimize the loss value as much as possible. Regarding batch

TABLE 1. Different batch size comparison results.

Batch size	100	500	1000	3155
Training time of each epoch (s)	18.611	3.483	1.734	**0.587**
Loss value	0.0406	0.0461	0.0475	0.0484

size, we have selected several different values to compare the training time of each epoch and the final loss value, as shown in Table 1. As we can see from Table 1, the larger the batch size is, the shorter the time it takes to train each epoch, but the final loss values are not much different. So we decided to use full batch size which is 3155 as the final batch size. The sparsity target is 0.05. The momentum is 0.5 and the initial learning rate is 1.

We process the values of the input matrix elements using the normalization method. This processed matrix is trained as an input signal to a neural network. After 15000 trainings, we can export the matrix data of the middle hidden layer which represents the feature of the signal after training. At the same time, an output matrix is obtained. Then we reconstruct the time-frequency map of the output matrix.

Fig.4 shows the change about loss values during training, and the final loss value is 0.0484, which means that the difference between the output and input data is very small.

C. EVALUATION INDICATORS

Since what we do is unsupervised classification, clustering results cannot be evaluated by accuracy without knowing the exact number of categories. So we introduce Compactness (CP), Separation (SP), Davies-Bouldin Index (DB), Dunn Validity Index (DVI) [31] and Rand Index(RI) [32] five indexes to evaluate our clustering method. We evaluate
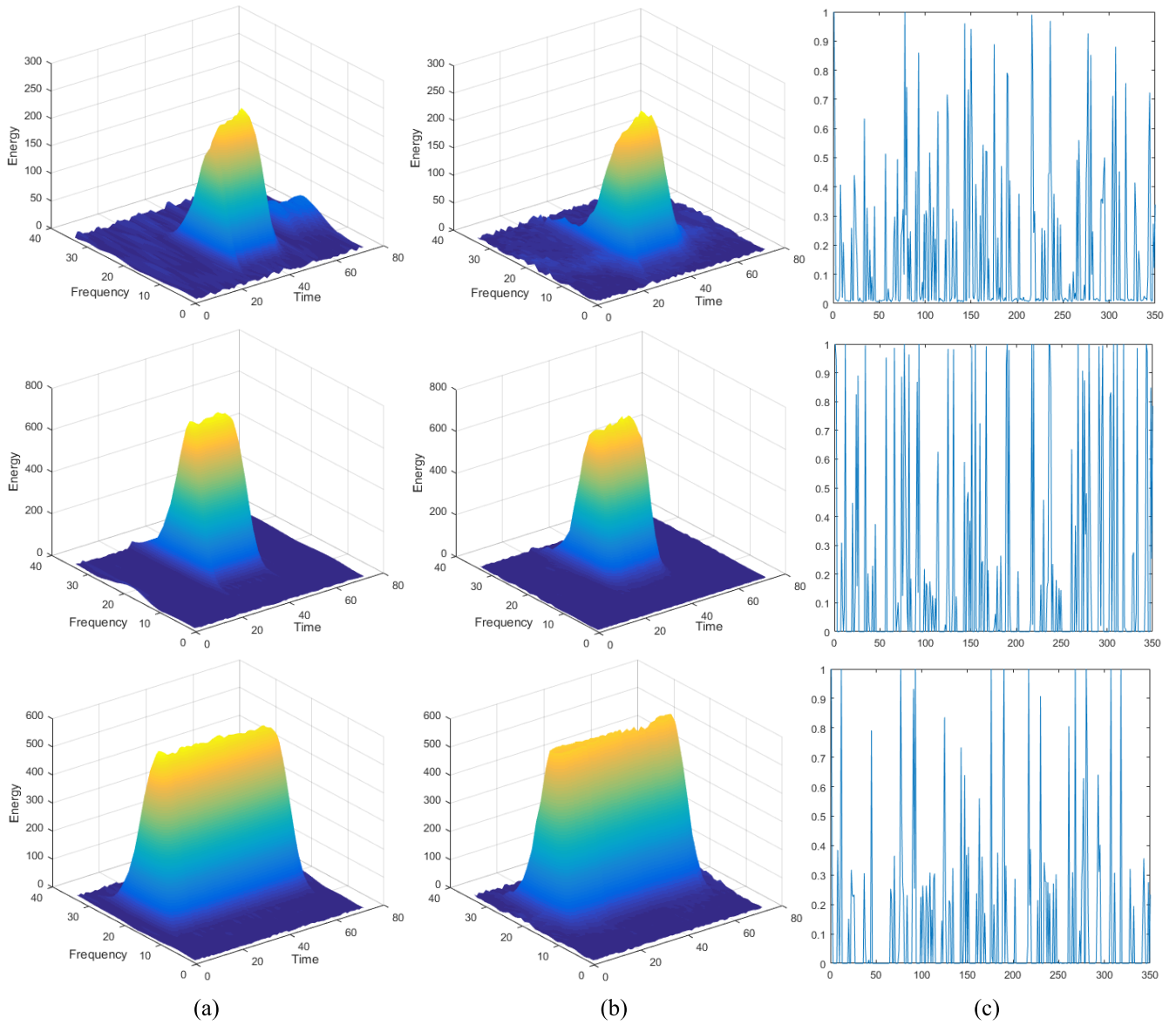
FIGURE 6. Input data of encoder (a), output data of decoder (b) and feature of hidden layer(c).

our experimental results by comparing these five clustering indexes of our method with other methods.

1) COMPACTNESS (CP)

CP represents the average distance from each point to the cluster center in each class, the smaller the CP is, the closer the cluster distance within the class is, the better the clustering result is. The value of CP is computed by:

$$\overline{CP}_i = \frac{1}{|\Omega_i|} \sum_{x_i \in \Omega_i} \|x_i - w_i\| \quad \overline{CP} = \frac{1}{K} \sum_{k=1}^{K} \overline{CP}_k \quad (14)$$

2) SEPARATION (SP)

SP represents the average distance between two cluster centers. The higher the SP is, the farther the cluster distance

between classes is, and the better the clustering result is. The SP is calculated as follow:

$$\overline{SP} = \frac{2}{k^2 - k} \sum_{i=1}^{k} \sum_{j=i+1}^{k} \|w_i - w_j\|_2 \quad (15)$$

3) DAVIES-BOULDIN INDEX (DB)

DB represents the maximum value of the quotient obtained by dividing the sum of the intra-class mean distances (CP) of any two classes and the distance between the centers of the two clusters. The smaller the DB is, the smaller the intra-class distance is, and the distance between classes is father, the clustering result is better. It calculated by:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left(\frac{\overline{CP}_i + \overline{CP}_j}{\|w_i - w_j\|_2} \right) \quad (16)$$

TABLE 2. Comparisons of evaluation indicators.

Cluster validity indices	CP	SP	DB	DVI	RI(480 test pulses)
Our method	**6.8925**	0.8043	**0.6394**	**0.0647**	**0.8541**
IFL and DBSCAN	26.2693	**0.9938**	0.7845	0.0609	0.5967

4) DUNN VALIDITY INDEX (DVI)

DVI represents the quotient of the minimum distance between the elements of any two different classes and the maximum distance between the elements in any same class. The larger the DVI is, the farther the distance between the classes is, the distance within the class is smaller, and the clustering result is better. The value of DVI is computed by:

$$DVI = \frac{\min\limits_{0<m\neq n<K}\left\{\min\limits_{\substack{\forall x_i\in\Omega_m \\ \forall x_j\in\Omega_n}}\left\{\|x_i-x_j\|\right\}\right\}}{\max\limits_{0<m\leq K}\max\limits_{\forall x_j,x_i\in\Omega_m}\left\{\|x_i-x_j\|\right\}} \quad (17)$$

5) RAND INDEX (RI)

Because it is unsupervised clustering, we can't know the final clustering result of our method in advance, so we introduce the Rand Index(RI) to evaluate our test results. If X is used to represent the actual category information, and Y is used to represent the sample class label obtained by the clustering result, then the RI index can be defined as:

$$RI = \frac{a+b}{a+b+c+d} = \frac{a+b}{C_n^2} \quad (18)$$

where *a* denotes the number of sample pairs belonging to the same class in X and in Y; *b* denotes the number of sample pairs belonging to different classes in X and also different classes in Y; *c* denotes the number of sample pairs belonging to the same class in X but different classes in Y; *d* denotes the number of sample pairs belonging to different classes in X but the same class in Y. The larger the value is, the closer the result of clustering is to the real situation of the data, and the better the clustering effect.

IV. RESULTS
A. FUZZY PROCESSION
Here is a comparison between a pre- and post-processing example. From the Fig.5, we can see that although the resolution of the time-frequency map is reduced after processing, the important amplitude feature part is preserved. This step reduces the amount of training data and improves the training speed.

B. CLUSTERING RESULTS
After training, we select some sets of data to show the time-frequency maps before and training and the feature data of hidden layer. From Fig.6, we can see that there are significant differences in the feature information (part c) by different types of signals. So based on this conclusion, we can classify
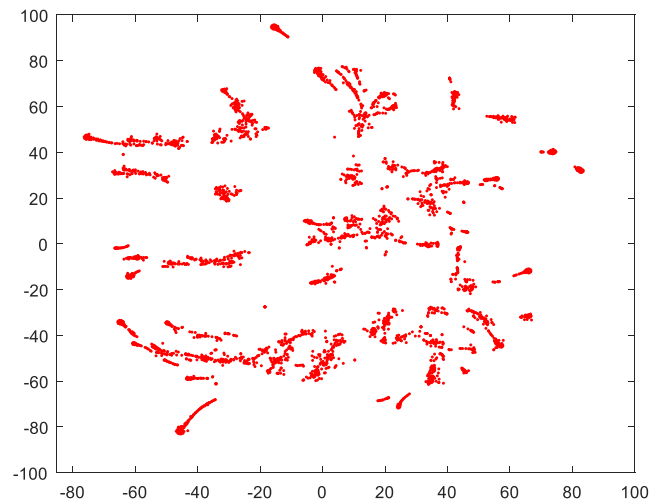
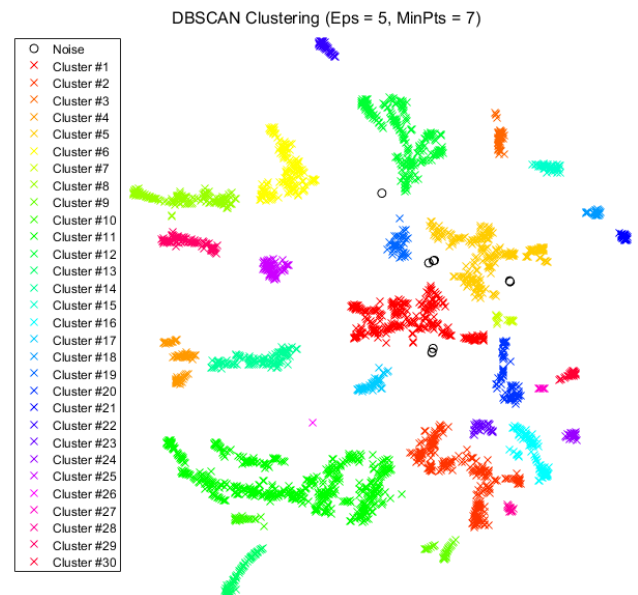

FIGURE 7. Visual dimension reduction by t-SNE.

FIGURE 8. Result of DBSCAN.

the feature signal instead of the original signal. Then these features are fed into the t-SNE model for dimensionality reduction, the result is shown in Fig. 7.

Next, we send the results of the dimensionality reduction processing to DBSCAN for visual classification processing, as shown in Fig.8. In terms of DBSCAN parameter selection, we have analyzed and processed several times, and finally set Eps to 5 and MinPts to 7. From the figure, we can see that the
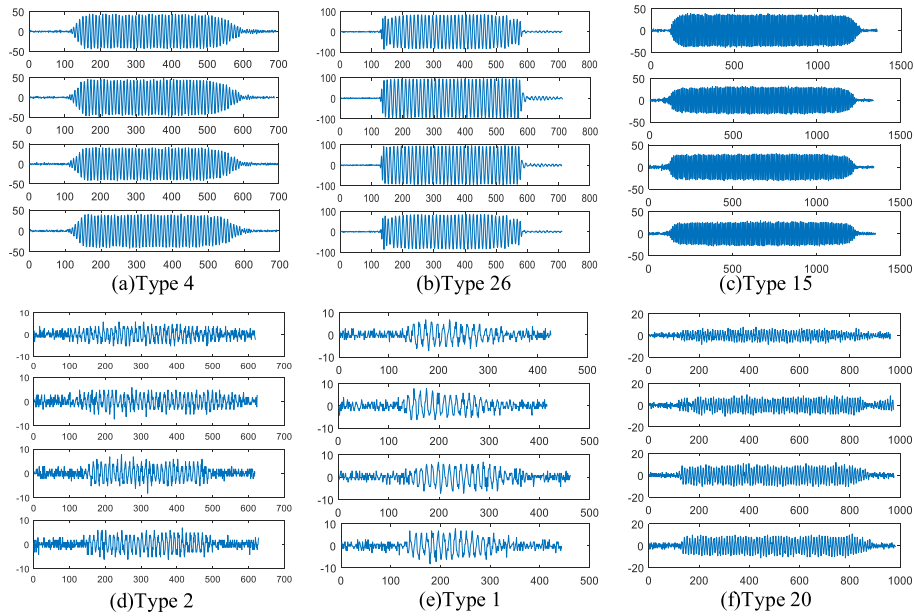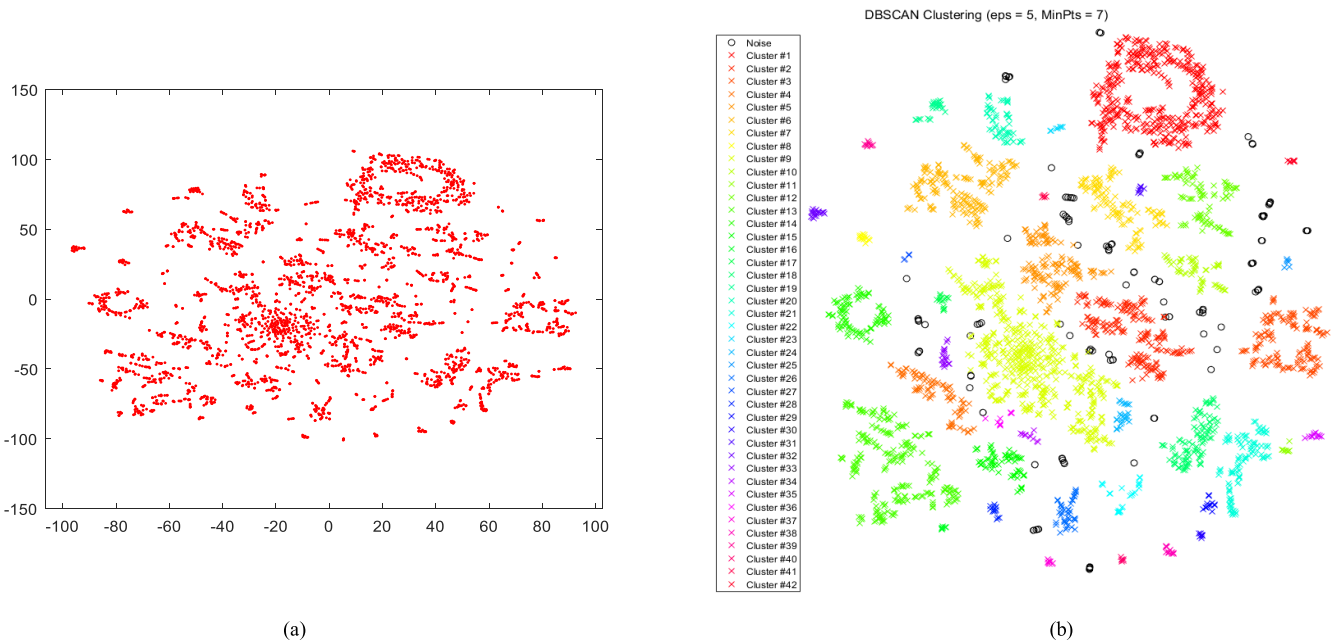
FIGURE 9. Clustering results.

FIGURE 10. Bézier modeling of the IFL (a) and DBSCAN results (b).

features of 3155 pulse signals are divided into 30 categories. We pick out the same cluster of data, and randomly select several data from the data distributed to the same cluster for comparison. The comparison results are shown in Fig.9. As can be seen from (a) and (b) of Fig. 9, the classification method proposed in this paper can accurately distinguish subtle differences in pulse shape such as rising and falling edges of pulses. It can be seen from (d) and (e) that the

classification effect of the pulse length is also obvious when the pulse shapes are similar.

C. COMPARED WITH OTHER METHODS

In this section, in order to evaluate the accuracy and effectiveness of our method classification, we compare this method with the combination of t-SNE and other clustering algorithms [33]. The method in [33] uses the estimated
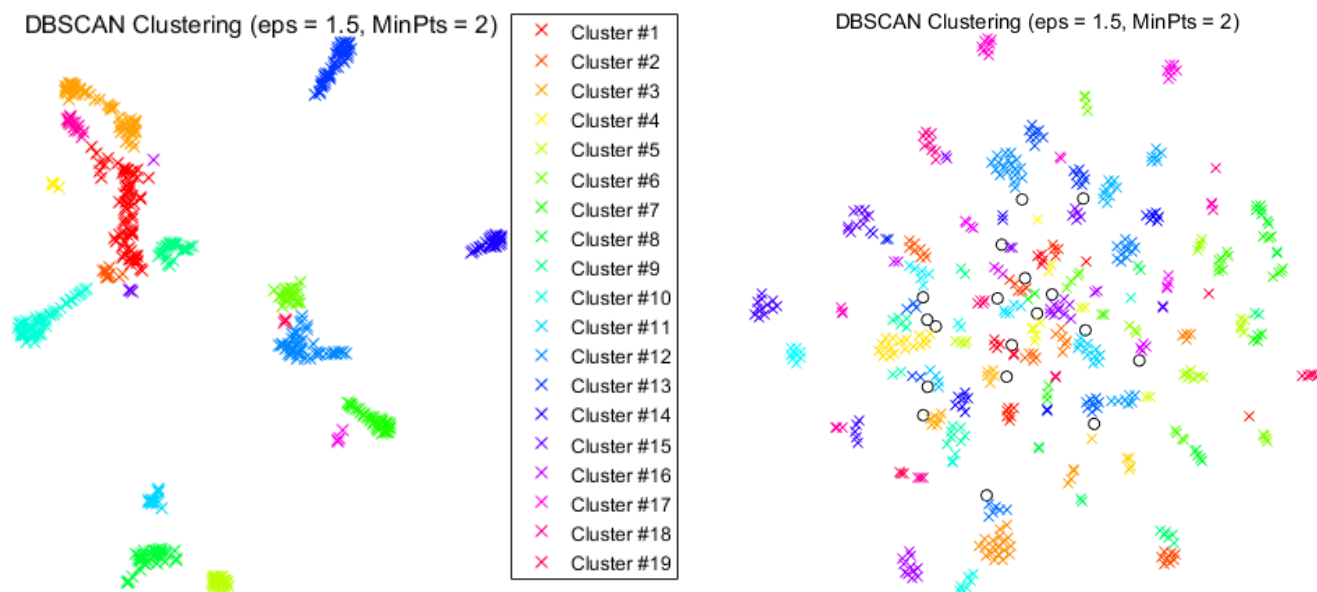
FIGURE 11. The comparison results of 480 testing pulses.

instantaneous frequency law (IFL) to extract a set of features and DBCSAN for radar pulses classification, the results are shown in Fig.10, and the comparison of four evaluation indicators is shown in Table.2.

From the comparison of the two methods, it can be seen that the proposed method is more suitable for the unsupervised training classification. In the t-SNE results obtained by the method in [33], the data dimensionality reduction effect is not ideal: (1) In the t-SNE clustering results, different types of data clusters are not completely separated, which makes the clustering work of DBSCAN difficult to distinguish. (2) After DBSCAN clustering, many small clusters and the noise data which are difficult to cluster are obviously more. (3) In the clustering results, the classification of errors is significantly increased after extracting data and observing. It can also be seen from Table.2 that the three indicators of CP, DB and DVI in the proposed method are better than those in [33]. The compactness(CP) index decreases by 73.76%; the Davies-Bouldin Index(DB) decreases by 18.50%; the Dunn Validity Index(DVI) increases by 6.24%. We think that the reason why the separation(SP) index is reduced is that the number of clustering categories are too large and too scattered, so the average distance between classes becomes larger, which is also a disadvantage that the SP index does not consider the effect within the class.

In order to test the correctness of our unsupervised classification method, we use artificially classified 480 pulse signals for testing, these pulses have been divided into 22 categories by manual classification. The clustering results are shown in Figure.11, our method divides them into 19 categories. The value of RI is 0.8541. And the comparison of the index RI is shown in Table.2, the Rand Index(RI) increases by 43.14%. From the results we can still see that the clustering accuracy is also higher than method in [33].

V. CONCLUSION

In this paper, a classification model of radar pulses based on SAE and DBSCAN is proposed. By combining the feature extraction function of SAE and the clustering function of DBCSAN, we have successfully achieved unsupervised classification with high accuracy. Because SAE can better explore the deep features of data, the clustering results are more accurate after SAE training. The comparison of the evaluation indicators after clustering has shown the practicability of the proposed method. However, in this paper, we only use one layer of hidden layer. So in later research, we can increase the number of hidden layer of SAE to obtain deeper features and achieve more accurate clustering results. Neural network can greatly reduce human manual labor and improve work efficiency, it will play a very important role in many fields in the future.

REFERENCES

[1] A. Mathur and G. M. Foody, "Multiclass and binary SVM classification: Implications for training and classification users," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 241–245, Apr. 2008.

[2] M. Seetha, K. V. N. Sunitha, D. V. L. Parameswari, and G. Ravi, "Accuracy assessment of object oriented and knowledge base image classification using P-trees," in *Proc. 2nd Int. Conf. Comput. Automat. Eng. (ICCAE)*, Singapore, Feb. 2010, pp. 760–763.

[3] M. Ross, C. A. Graves, J. W. Campbell, and J. H. Kim, "Using support vector machines to classify student attentiveness for the development of personalized learning systems," in *Proc. 12th Int. Conf. Mach. Learn. Appl.*, Miami, FL, USA, Dec. 2013, pp. 325–328.

[4] O. Chapelle, P. Haffner, and V. N. Vapnik, "Support vector machines for histogram-based image classification," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1055–1064, Sep. 1999.

[5] F. Bovolo, L. Bruzzone, and L. Carlin, "A novel technique for subpixel image classification based on support vector machine," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2983–2999, Nov. 2010.

[6] E. Pasolli, F. Melgani, and Y. Bazi, "Support Vector Machine Active Learning Through Significance Space Construction," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011.

[7] K. Krishna and M. N. Murty, "Genetic K-means algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 3, pp. 433–439, Jun. 1999.

[8] L. Jia, M. Li, P. Zhang, Y. Wu, and H. Zhu, "SAR image change detection based on multiple kernel K-means clustering with local-neighborhood information," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 6, pp. 856–860, Jun. 2016.

[9] M. Laszlo and S. Mukherjee, "A genetic algorithm using hyper-quadtrees for low-dimensional k-means clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 533–543, Apr. 2006.

[10] L. Yang, X. Guangqiang, L. Xiaomei, and L. Hua, "Dependent function interval parameters training algorithm based on DBSCAN clustering," in *Proc. 31st Chin. Control Conf.*, Hefei, China, Jul. 2012, pp. 7709–7712.

[11] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-time superpixel segmentation by DBSCAN clustering algorithm," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5933–5942, Dec. 2016.

[12] W. Xue, X. Wenxia, and L. Guodong, "Image edge detection algorithm research based on the CNN's neighborhood radius equals 2,"in *Proc. Int. Conf. Smart Grid Elect. Automat. (ICSGEA)*, Zhangjiajie, China, Aug. 2016, pp. 115–119.

[13] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 1096–1103.

[14] H. Zheng, M. Chen, W. Liu, Z. Yang, and S. Liang, "Improving deep neural networks by using sparse dropout strategy," in *Proc. IEEE China SummitInt. Conf. Signal Inf. Process. (ChinaSIP)*, Xi'an, China, Jul. 2014, pp. 21–26.

[15] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, *Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction*. Berlin, Germany: Springer, 2011, pp. 52–59. doi: 10.1007/978-3-642-21735-7_7.

[16] D. Zhang, W. Zuo, D. Zhang, Y. Li, and N. Li, "Gaussian ERP kernel classifier for pulse waveforms classification," in *Proc. 20th Int. Conf. Pattern Recognit.*, Istanbul, Turkey, Aug. 2010, pp. 2736–2739.

[17] A. Dai, H. Zhang, and H. Sun, "Automatic modulation classification using stacked sparse auto-encoders," in *Proc. IEEE 13th Int. Conf. Signal Process. (ICSP)*, Chengdu, China, Nov. 2016, pp. 248–252.

[18] S.-Z. Su, Z.-H. Liu, S.-P. Xu, S.-Z. Li, and R. Ji, "Sparse auto-encoder based feature learning for human body detection in depth image," *Signal Process.*, vol. 112, pp. 43–52, Jul. 2015. doi: 10.1016/j.sigpro.2014.11.003.

[19] W. Liu, T. Ma, Q. Xie, D. Tao, and J. Cheng, "LMAE: A large margin auto-encoders for classification," *Signal Process.*, vol. 141, pp. 137–143, Dec. 2017.

[20] L. Cain, J. Clark, E. Pauls, B. Ausdenmoore, R. Clouse, and T. Josue, "Convolutional neural networks for radar emitter classification," in *Proc. IEEE 8th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Las Vegas, NV, USA, Jan. 2018, pp. 79–83.

[21] C. Wang, J. Wang, and X. Zhang, "Automatic radar waveform recognition based on time-frequency analysis and convolutional neural network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, New Orleans, LA, USA, Mar. 2017, pp. 2437–2441.

[22] L. Zhang, S. Deng, and S. Li, "Analysis of power consumer behavior based on the complementation of K-means and DBSCAN," in *Proc. IEEE Conf. Energy Internet Energy Syst. Integr. (EI2)*, Beijing, China, Nov. 2017, pp. 1–5.

[23] R. G. Stockwell, L. Mansinha, and R. P. Lowe, "Localization of the complex spectrum: The S transform," *IEEE Trans. Signal Process.*, vol. 44, no. 4, pp. 998–1001, Apr. 1996.

[24] E. Sejdic, I. Djurovic, and J. Jiang, "S-transform with frequency dependent kaiser window," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Honolulu, HI, USA, Apr. 2007, pp. III-1165–III-1168.

[25] T. Junbo, L. Weining, A. Juneng, and W. Xueqian, "Fault diagnosis method study in roller bearing based on wavelet transform and stacked auto-encoder," in *Proc. CCDC*, May 2015, pp. 4608–4613.

[26] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[27] S. Mukherjee, "t-SNE based feature extraction technique for multi-layer perceptron neural network classifier," in *Proc. Int. Conf. Intell. Comput., Instrum. Control Technol. (ICICICT)*, Kannur, India, Jul. 2017, pp. 660–664.

[28] M. Pan, J. Jiang, Q. Kong, J. Shi, Q. Sheng, and T. Zhou, "Radar HRRP target recognition based on t-SNE segmentation and discriminant deep belief network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 9, pp. 1609–1613, Sep. 2017.

[29] L. Van Der Maaten, "Accelerating t-sne using tree-based algorithms.," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3221–3245, 2014.

[30] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining (KDD)*, 1996, pp. 226–231.

[31] Y. Shim, J. Chung, and I.-C. Choi, "A comparison study of cluster validity indices using a nonhierarchical clustering algorithm," in *Proc. Int. Conf. Comput. Intell. Modelling, Control Automat. Int. Conf. Intell. Agents, Web Technol. Internet Commerce (CIMCA-IAWTIC)*, Vienna, Austria, Nov. 2005, pp. 199–204.

[32] J. Handl and J. Knowles, "An evolutionary approach to multiobjective clustering," *IEEE Trans. Evol. Comput.*, vol. 11, no. 1, pp. 56–76, Feb. 2007.

[33] F. Digne, A. Baussard, A. Khenchaf, C. Cornu, and D. Jahan, "Classification of radar pulses in a naval warfare context using Bézier curve modeling of the instantaneous frequency law," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 3, pp. 1469–1480, Jun. 2017.

[34] C. M. Jeong, Y. G. Jung, and S. J. Lee, "Deep belief networks based radar signal classification system," *J. Ambient Intell. Hum. Comput.*, pp. 1–13, Apr. 2018. doi: 10.1007/s12652-018-0774-7.

KAN REN received the B.S. degree in optical information science and technology from the Nanjing University of Science and Technology, Nanjing, China, in 2006 and the Ph.D. degree in electronic engineering from the University of Surrey, Guildford, U.K., in 2011. Since 2011, he has been an Associate Professor with the School of Electronic Engineering and Optoelectronic Technology, Nanjing University of Science and Technology, Nanjing, China. His current research interests include computer version and machine learning.
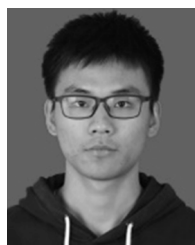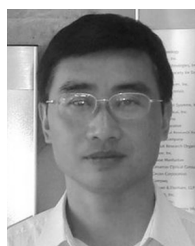
HONGLIANG YE received the B.S. degree in electronic science and technology from the Nanjing University of Science and Technology, Nanjing, China, in 2017. He is currently pursuing the master's degree with the Nanjing University of Science and Technology. His research interests include deep learning, optimization, and its application in image processing.

GUOHUA GU received the B.S. and M.S. degrees in optical instrument from the Nanjing University of Science and Technology, in 1989 and 1996, respectively, and the Ph.D. degree in optical engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2001. Since 2007, he has been a Professor with the School of Electronic Engineering and Optoelectronic Technology, Nanjing University of Science and Technology, Nanjing, China. His current research interests include optical design, computer version, and machine learning.

QIAN CHEN received the B.S. and M.S. degrees in optoelectronic technology from the Nanjing University of Science and Technology, in 1987 and 1991, respectively, and the Ph.D. degree in optical engineering from the Nanjing University of Science and Technology, Nanjing, China, in 1996. Since 1996, he has been a Professor with the School of Electronic Engineering and Optoelectronic Technology, Nanjing University of Science and Technology, Nanjing, China. His current research interests include optical design and computer version.

• • •