

## Research Article

# A Novel Classification and Identification Scheme of Emitter Signals Based on Ward's Clustering and Probabilistic Neural Networks with Correlation Analysis

Xiaofeng Liao,<sup>1</sup> Bo Li ,<sup>1,2</sup> and Bo Yang<sup>1</sup>

<sup>1</sup>College of Electronic and Information Engineering, Southwest University, 400715 Chongqing, China

<sup>2</sup>Department of Mathematics, Army Logistics University of PLA, 401331 Chongqing, China

Correspondence should be addressed to Bo Li; 56050009@qq.com

Received 9 June 2018; Revised 22 August 2018; Accepted 13 September 2018; Published 5 November 2018

Academic Editor: Reinoud Maex

Copyright © 2018 Xiaofeng Liao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The rapid development of modern communication technology makes the identification of emitter signals more complicated. Based on Ward's clustering and probabilistic neural networks method with correlation analysis, an ensemble identification algorithm for mixed emitter signals is proposed in this paper. The algorithm mainly consists of two parts, one is the classification of signals and the other is the identification of signals. First, self-adaptive filtering and Fourier transform are used to obtain the frequency spectrum of the signals. Then, the Ward clustering method and some clustering validity indexes are used to determine the range of the optimal number of clusters. In order to narrow this scope and find the optimal number of classifications, a sufficient number of samples are selected in the vicinity of each class center to train probabilistic neural networks, which correspond to different number of classifications. Then, the classifier of the optimal probabilistic neural network is obtained by calculating the maximum value of classification validity index. Finally, the identification accuracy of the classifier is improved effectively by using the method of Bivariable correlation analysis. Simulation results also illustrate that the proposed algorithms can accurately identify the pulse emitter signals.

## 1. Introduction

Under the conditions of the rapid development of modern information technology, various types of communication equipment, such as radar and radio navigation equipment, radio and television equipment, and electronic computer and peripheral equipment, have been used by military and large technology companies. Its applications have been extended from the ground, air, and sea to outer space. In order to obtain important information promptly, accurately, and effectively, the signal characteristics derived from the same types of communication sources need to be extracted and analyzed. The realization of identification from general communication signals to individual signals has a significance to win the initiative. Therefore, many researchers have devoted to the study of the identification of emitter signals. Their purpose is to further improve the accuracy of signal classification and identification.

The earliest research on the identification of emitter signals began in the 1970s, and it was one of the key technologies in Electronic Warfare systems. At present, the identification of emitter is mainly based on two modeling techniques: syntactic pattern-based methods and parametric pattern-based methods. In [1], Visnevski author had proposed a syntactic model to identify multifunction radar (MFR) signals. In the process of modeling, the MFRs were considered as stochastic discrete event systems that communicated information by use of radar word level modeling, radar phrase level modeling, and radar sentence level modeling. The radar word was a fixed arrangement of finite number of pulses, the radar phrase was a series of limited number of radar words, and the radar sentence was a combination of limited number of radar phrases. His simulation experiments had shown that the designed principle was effective for identifying MFRs. Based on the syntactic model of MFRs, Alex Wang and Vikram Krishnamurthy had used the

stochastic context-free grammar to describe the behaviors of the MFR system, and some good results were obtained. Although the stochastic context-free grammar was a model for capturing the essential features of the MFR dynamics [2], it had some defects in estimating the parameters of stochastic context-free grammar. Therefore, the expectation maximization algorithm had been proposed by LP Dai *et al.* to estimate these parameters, which can be used to further estimate the characteristic parameters of MFR [3]. From this point of view, the ultimate goal of the modeling technique based on syntax was to find the feature parameters of emitters. It highlighted the importance of identification technology based on the parametric pattern. The feature parameter matching technique was a basic method of this pattern. The main characteristic of this method was to identify the emitter signals by matching the measured signal characteristic parameter vector with the corresponding characteristic parameters in the known database (the libraries of radar types). This method depended on the feature parameter database, and it can only be applied to emitter identification problem with invariant characteristic parameters. Moreover, the libraries of types had also an inherent uncertainty resulting inevitably from data collection methods. Therefore, Jan Matuszewski *et al.* had proposed the knowledge-based techniques to identify emitters [4, 5]. They think that the information of known radar platforms (knowledge), including position, intent, and recent operational history, plays an important role in the identification of emitters. Their approaches had been successfully used to identify some specific emitters. In order to fully utilize this knowledge to identify emitters, Janusz Dudczyk had proposed an idea of constructing Emitter DataBase based on the entity-relationship modeling. The entity-relationship diagram was introduced to realize this idea, which had pointed out a new direction for the construction of a complete and accurate Electronic Intelligence systems [6]. Meanwhile, artificial intelligence techniques and some optimized feature selection methods were used to improve the identification accuracy of emitter signals. In [7], the authors had proposed a vector neural network with a supervised learning algorithm which worked for signal classification and emitter identification. This network took carrier frequency, pulse width, and pulse repetition interval as inputs to complete the identification. In [8], the authors had proposed an identification method of radar signals based on immune radial-basis function neural network, which can improve the convergence speed and performance of the algorithm. In [9], a multichannel recognition system with an independent distance which was defined on impartiality condition had been proposed to identify the specific emitters. By modifying the distance in each special recognition channel, radio frequency, pulse width, and pulse repetition interval of radar signals can be extracted and classified into an appropriate class. Beyond that, some methods had been proposed to reduce the identification error rate. In [10], wavelet features were used as inputs of neural networks to identify emitters. In [11–13], the support vector machine was introduced to identify emitter signals. In [14, 15], the fuzzy  $c$ -means and probabilistic neural networks (PNN) were used to identify emitter signals. However, the

determination of the optimal number of clusters was a major challenge for these methods, and they cannot effectively improve the identification accuracy. Therefore, Jawad *et al.* designed a clustering validity function in the hidden-layer output space of PNN to find the optimal number of clusters. Their methods were successfully applied to the classification of land use [16]. But, the method of determining the range of clustering number was subjective, which may reduce the identification accuracy of the algorithm. To overcome this deficiency and further improve accuracy, a classification and identification scheme of emitter signals based on the Ward clustering method (WCM) and PNN with correlation analysis was proposed in this paper. Its advantages are presented in three aspects:

- (1) The self-adaptive filtering, Ward's clustering, and clustering validity indexes are skillfully used to determine the scope of the optimal number of clusters.
- (2) The classification validity index  $D$  is flexibly used to find the optimal PNN classifier.
- (3) The probabilistic neural networks with Bivariable correlation analysis approach are proposed to improve the identification accuracy of emitter signals.

The rest of this paper is organized as follows. In Section 2, the classification and identification schemes including adaptive filtering, frequency spectrum, evaluation indexes, WCM and PNN classifier are introduced. The flowchart and the pseudocodes of the classification algorithms are designed in Section 3. The flowchart and the pseudocodes of the identification algorithm are given in Section 4. The identification experiments are also carried out in this section. The comparison discussions of different schemes are proposed in Section 5. Some innovations and applicable conditions of the proposed method are summarized in Section 6.

## 2. Classification Model of Emitter Signals

**2.1. Self-Adaptive Filtering.** In the field of engineering technology, the signal  $x(n)$  received at time  $n$  usually contains two parts. One is the useful signal  $s(n)$ , which is what we need, and it enables us to understand the properties of the object to be studied. The other is the interference signal  $x_1(n)$ , which is what we do not need, and it prevents us from understanding the properties of the object to be studied. The actual signal will be obtained once the two parts are combined together. That is,  $x(n) = s(n) + x_1(n)$ .

Weakening the interference signal  $x_1(n)$  and maintaining or enhancing the useful signal  $s(n)$  are an important purpose of signal processing. The usual method is to use a frequency function  $H(f)$  to multiply the frequency spectrum  $X(f)$  of the signal  $x(n)$ . This process is called filtering. Its essence is to weaken the interference signal and highlight the useful signal. Currently, the most widely used filters include Kalman filtering, Wiener filtering, median filtering, sequential statistical filtering, wavelet transform, self-adaptive filtering, etc. In terms of adaptability and filtering performance, one of the best filtering methods is the self-adaptive filtering, which is developed on the basis of

Kalman filtering, Wiener filtering, and linear filtering. The most important feature of the self-adaptive filtering is that it can track the time-varying characteristics of input signals and eliminate the unknown interference contained in the signals. Self-adaptive filtering based on the least mean square algorithm is proposed by Widrow and Hoff, and it has been widely used in many fields because of its simplicity, robustness, and easy implementation. The principle diagram of self-adaptive filtering technology is shown in Figure 1.

Figure 1 presents the schematic diagram of noise elimination for self-adaptive filter. The actual signal  $x(n)$  contains interference signal  $x_1(n)$  generated from signal channel 1. In order to eliminate it, the noise signal  $x_0(n)$  which is independent of  $s(n)$  but related to  $x_1(n)$  must be sampled from noise source through the signal channel 2. The main function of the self-adaptive filter is to process  $x_0(n)$  so that the output  $y(n)$  approximates to  $x_1(n)$ . Under the condition of convergence of filtering algorithm, the output  $e(n)$  of the system approximates to  $s(n)$  when  $y(n)$  approaches  $x_1(n)$ . The iterative formulas of self-adaptive filtering algorithm based on the least mean square are defined as follows [17]:

$$\begin{aligned} e(n) &= d(n) - \omega^T(n)X(n), \\ \omega(n+1) &= \omega(n) + 2\mu e(n)X(n), \end{aligned} \quad (1)$$

where  $d(n)$  is the desired signal;  $X(n) = [x(n), x(n-1), \dots, x(n-M+1)]^T$  is the input signal vector at time  $n$ ;  $M$  is the length of filter; and  $\mu$  is the fixed step size and satisfies  $0 < \mu < 1/MP_{in}$ , where  $P_{in}$  is the input power of filter.  $\omega(n) = [w_0(n), w_1(n), \dots, w_{M-1}(n)]^T$  is the weight vector of  $M$  order adaptive filter at time  $n$ .  $\omega(n) = 0$  at initial time.  $y(n) = \omega^T(n)X(n)$  represents the actual output signal of filter. In noise elimination applications,  $x(n)$  is usually used as the desired signal  $d(n)$  and  $x_0(n)$  is used as the input signal of the filter to eliminate  $x_1(n)$ . After many iterations, the difference between  $d(n)$  and  $y(n)$  is the estimate of signal  $s(n)$ . This algorithm has the advantages of small amount of computation, easy implementation, and stable performance, but its convergence speed is relatively slow. Therefore, the authors proposed a variable step adaptive filtering algorithm based on bell-shaped function [18]. The variable step size is given as follows:

$$\mu(n) = \mu_{\max} \left( 1 - e^{-a|e(n)|^b} \right), \quad (2)$$

where  $\mu_{\max} \in (0, 1/MP_{in})$  is the maximum step size that can maintain the convergence of the adaptive filtering algorithm. Experiments show that this algorithm can effectively improve the convergence speed and reduce the steady-state error when  $a = 0.08$  and  $b = 4$ . In subsection 3, the method is used to eliminate the interference signal of emitters.

**2.2. The Spectrum of Signals.** In order to classify and identify signals, it is necessary to analyze the frequency spectrum and energy spectrum of the signals. From the electrical knowledge,  $P = V^2/R$ , where  $V$  represents voltage and  $R$  represents resistance. If the resistance  $R = 1$  and  $V$  is replaced by signal  $x(t)$ , the instantaneous energy is  $x^2(t)$ . Thus, the total

energy of the signals can be expressed as  $\int_{-\infty}^{+\infty} x^2(t) dt$ . According to Parseval's theorem, it can be obtained by the following equation:

$$\int_{-\infty}^{+\infty} |x(t)|^2 dt = \int_{-\infty}^{+\infty} |X(f)|^2 df, \quad (3)$$

where

$$X(f) = \int_{-\infty}^{+\infty} x(t)e^{-2\pi ift} dt, \quad (4)$$

is the Fourier transform for signal  $x(t)$  and  $f$  represents the frequency of signal  $x(t)$ .  $|X(f)|$  is called the amplitude spectrum and  $\arg X(f)$  is called the phase spectrum of  $x(t)$ . The frequency spectrum of  $x(t)$  consists of amplitude spectrum and phase spectrum.  $|X(f)|^2$  is called energy spectrum density. Equation (3) indicates that the energy of  $x(t)$  is closely related to  $|X(f)|^2$ , and it can be obtained by calculating its integral on  $(-\infty, +\infty)$ . Therefore, we can obtain the frequency distribution and energy distribution of each signal by analyzing the spectrum of the signal. Then, the key amplitude and the frequency of energy distribution can be further obtained, which provide conditions for directly observing the similarities and differences of paired signals.

**2.3. The Ward Clustering Method.** As a hierarchical agglomerate cluster algorithm, the WCM has a wide range of applications [19–21]. First, it is started by accepting each node as a separate cluster. Then, the clusters with minimum distance between themselves are combined in pairs at each stage of the algorithm. This smallest distance is called the Ward distance and defined as follows:

$$d_{rs} = \frac{n_r \cdot n_s}{n_r + n_s} \|\bar{x}_r - \bar{x}_s\|^2, \quad (5)$$

where  $r$  and  $s$  represent the two distinct clusters,  $n_r$  and  $n_s$  represent the number of data points of two clusters, respectively,  $\bar{x}_r$  and  $\bar{x}_s$  represent the center of the corresponding cluster, and  $\|\cdot\|$  is Euclidean norm. The centers and cardinal numbers of the new cluster are updated according to the following equations:

$$\bar{x}_r' = \frac{n_r \cdot \bar{x}_r + n_s \cdot \bar{x}_s}{n_r + n_s}, \quad (6)$$

$$n_r' = n_r + n_s, \quad (7)$$

Ward's clustering algorithm has the following steps:

*Step 1.* Each sample point is treated as a cluster. At this time, the sum of squares of deviations for each cluster is equal to 0.

*Step 2.* Two arbitrary clusters are merged, and the sum of squares of deviations is calculated from Equations (5)–(7). If we assume that there are  $N$  clusters in total, it must be calculated  $N(N-1)/2$  times.

*Step 3.* The two clusters with the smallest squared sum of deviations are combined into one class. The method eventually aggregates all sample points into one class when the number of clusters is unknown.

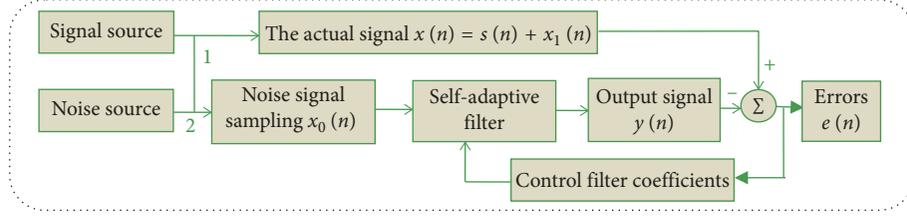


FIGURE 1: Flowchart of the self-adaptive filtering technology.

If the number of clusters is known, the WCM can be directly used to classify the signal data after removing the noise. Otherwise, it can be obtained by analyzing the dendrogram of clustering. This is a rather subjective approach, which is difficult to help us finding the true number of clusters for a given data set. In recent research, the clustering validity indexes, such as Calinski-Harabasz (CH) index, Gap index, Silhouette (Silh) index, and Davies-Bouldin (DB) index, have been demonstrated to be the best validation tools for determining the optimal number of clusters [22–27].

**2.3.1. Calinski-Harabasz Index.** For a given set  $Y = \{y_1, y_2, \dots, y_N\}$ , assume that the dimension of each entity  $y_i$  is  $v$ ,  $i = 1, 2, \dots, N$ .  $K$  nonempty disjoint cluster sets  $S = \{S_1, S_2, \dots, S_K\}$  around the centroid set  $C = \{c_1, c_2, \dots, c_K\}$  can be obtained by minimizing the within-cluster distance  $W_K$ :

$$W_K = \sum_{k=1}^K \sum_{y_i \in S_k} d(y_i, c_k), \quad (8)$$

where  $d(y_i, c_k)$ ,  $k = 1, 2, \dots, K$  represents the squared Euclidean distance between the entity  $y_i$  and the centroid  $c_k$ , that is,

$$d(y_i, c_k) = \sum_{j \in v} (y_{ij} - c_{kj})^2. \quad (9)$$

Then, the CH index is defined as follows [24]:

$$\text{CH}(K) = \frac{(T_1 - W_K)/(K - 1)}{W_K/(N - K)}, \quad (10)$$

where  $W_K$  is defined as in (8), and  $T_1$  can be calculated by  $T_1 = \sum_{i=1}^N \sum_{j \in v} (y_{ij} - \bar{y}_j)^2$ .

The CH index can reflect the compactness of the cluster by means of the overall within-cluster variance. The separation degree of the clusters can be reflected by the overall between-cluster variance. Therefore, a good clustering scheme corresponds to a higher value of CH index.

**2.3.2. Silhouette Index.** For each entity  $y_i$ , its silhouette value measures the similarity between the entity  $y_i$  and the points in its own cluster, when compared to the points in other clusters. This similarity is reflected by measuring the distance between the entity  $y_i$  and the points derived from different clusters. The silhouette value of the entity  $y_i$  is defined as follows [24]:

$$s(y_i) = \frac{b(y_i) - a(y_i)}{\max\{a(y_i), b(y_i)\}}, \quad (11)$$

where  $a(y_i)$  is the average distance from the entity  $y_i \in S_k$  to all other points  $y_j \in S_k$ ,  $b(y_i)$  is the minimum distance from the entity  $y_i$  to all other points  $y_j$ , which satisfies  $y_j \in S_l, y_i \notin S_l, l \neq k$ . Therefore,  $-1 \leq s(y_i) \leq 1$ . If  $s(y_i)$  is close to zero, the entity  $y_i$  could be assigned to another cluster. A negative value of  $s(y_i)$  indicates that the corresponding assignment seriously damages cluster cohesion, and the clustering result of  $y_i$  is not advisable.  $y_i$  is well matched to its own cluster when  $s(y_i)$  is close to 1. Finally, the validity of the whole clustering can be quantified by Silh index, and it is defined as follows:

$$\text{Silh}(K) = \sum_{i=1}^N \frac{s(y_i)}{N}. \quad (12)$$

The Silh index can be used with any distance metric, including the Manhattan distances and Euclidean distances.

**2.3.3. Davies-Bouldin Index.** A good partition should have a larger intercluster separation degree and stronger within-cluster homogeneity and compactness. The DB index is proposed based on this idea [26]. More concretely, it is constructed by a ratio of within-cluster and between-cluster distances. The DB index is defined as follows:

$$\text{DB}(K) = \left( \sum_{k=1}^K \max_{k \neq j} \left\{ \frac{(\bar{d}_k + \bar{d}_j)}{d_{kj}} \right\} \right) / K, \quad (13)$$

where

$$\bar{d}_k = \left( \frac{1}{|S_k|} \sum_{i=1}^{|S_k|} |y_i - c_k|^q \right)^{1/q}, \quad (14)$$

represents the average distance between each point  $y_i$  in cluster  $k$  and the centroid of cluster  $k$ .  $|S_k|$  is the number of points in cluster  $k$ . If  $q = 1$ ,  $\bar{d}_k$  is the average Euclidean distance between the points in cluster  $k$  to the centroid of cluster  $k$ . If  $q = 2$ ,  $\bar{d}_k$  is the standard deviation of the distance of points in cluster  $k$  to the center of cluster  $k$ . When  $k$  in  $\bar{d}_k$  is replaced by  $j$ ,  $\bar{d}_j$  can be obtained. In addition,  $d_{kj}$  can be calculated according to the following equation:

$$d_{kj} = \left( \sum_{h=1}^v |\bar{c}_{kh} - \bar{c}_{jh}|^p \right)^{1/p}. \quad (15)$$

It represents the distances between the centroids of the  $k$ th and the  $j$ th clusters.  $\bar{c}_{kh}$  is the  $h$ th component of the centroid of cluster  $k$  and  $d_{kj}$  is the Minkowski metric of the centroids which characterizes clusters  $k$  and  $j$ . Specifically,  $p = 1$ ,  $d_{kj}$  is the Manhattan distance between centroids  $p = 2$ ,  $d_{kj}$  is the Euclidean distance between centroids.

The DB index can reflect the degree of within-cluster dispersion and between-cluster separation. So, the true number of clusters may be determined according to the minimum value of the DB index.

**2.3.4. Gap Index.** Robert Tibshirani et al. proposed the gap statistic method for estimating the number of clusters in a set of data [27]. A graph of the within-cluster dispersion versus the number of clusters  $k$  for a clustering procedure shows that the within-cluster dispersion decreases monotonically as  $k$  increases, but from some  $k$  onwards, the decrease becomes flatter obviously. Such position is called ‘elbow’, it often implies the appropriate number of clusters. The gap criterion gives an approach to estimate the number of clusters by locating this ‘elbow’. Therefore, under this criterion, the optimal number of clusters occurs at the largest gap value. The Gap index is defined as follows:

$$\text{Gap}_N(K) = E_N\{\log(U_K)\} - \log(U_K), \quad (16)$$

where  $N$  represents the number of points,  $K$  represents the number of clusters that are evaluated,  $U_K$  defined in (17) represents the within-cluster dispersion degree.

$$U_K = \sum_{k=1}^K \frac{1}{2N_k} H_k, \quad (17)$$

where  $N_k$  is the number of points in cluster  $k$ ,  $H_k$  is the sum of the distances of any two points in the  $k$ th cluster. The expected value  $E_N\{\log(U_K)\}$  is determined by Monte Carlo sampling from a reference distribution. The Gap index can also be used for any distance metric.

The WCM belongs to unsupervised categorization technique, which can help us find the centroid of each cluster. However, the classification accuracy of this method is limited, which makes the method not able to be used directly for signal recognition. Comparatively, PNN can effectively improve the accuracy of classification and identification [28, 29].

**2.4. Probabilistic Neural Network Classifier.** As a method of nonparametric Parzen windows estimation, PNN is first proposed by Specht. It is a nonlinear classification technique and essentially a parallel algorithm based on Bayesian minimum risk criterion [30]. Given a sample to be identified  $x$ , its posterior probability  $P(S_k|x)$  can be obtained by PNN classifier. However, if the probability densities of the classes to be separated are unknown, the training samples with known identity need to be used to estimate them. Finally, the trained PNN is used to determine the identity of  $x$ . A typical PNN classifier consists of an input layer, a pattern layer (hidden layer), a summation layer and an output layer. The flowchart of the PNN is shown in Figure 2.

The input layer neurons are used to receive values from training samples and send data to the neurons in the pattern layer, which is fully connected to the input layer. The number of neurons in the input layer is equal to the length of the input vector. The number of neurons in the pattern layer is the same as the number of training samples. Here, all neurons are collected into different groups, and the  $i$ th neuron in group  $k$  corresponds to a Gaussian function  $f_i^{(k)}(x, \sigma)$ ,  $i = 1, 2, \dots, m_k$ , where  $m_k$  represents the number of neurons in group  $k$ ,  $k = 1, 2, \dots, K$ . Gaussian function which is also called probability density function is defined as follows:

$$f_i^{(k)}(x, \sigma) = \frac{1}{(2\pi)^{v/2} \sigma^v} \exp\left[-\sum_{j=1}^v \frac{(x_{ij}^{(k)} - x_j)^2}{2\sigma^2}\right], \quad (18)$$

where  $v$  is the dimension of the input vector  $x = (x_1, x_2, \dots, x_v)$ ,  $x_j$  is the  $j$ th component of the input vector  $x$ ,  $x_{ij}^{(k)}$  is the  $j$ th component of the  $i$ th neuron in class  $k$ . The so-called smoothing parameter  $\sigma \in (0, 1)$  determined experimentally by comparing their corresponding classification accuracy plays an important role in estimation error of the PNN classifier. The outputs of pattern layer are connected to the summation units depending on the class of patterns. There is one neuron for each group, and each neuron in summation layer sums the outputs derived from the pattern layer neurons as follows:

$$p_k(x) = \frac{1}{(2\pi)^{v/2} \sigma^v m_k} \sum_{i=1}^{m_k} \exp\left[-\sum_{j=1}^v \frac{(x_{ij}^{(k)} - x_j)^2}{2\sigma^2}\right]. \quad (19)$$

Finally, the output layer neuron output a number 1 and multiple numbers 0. The value of 1 corresponds to the classifier’s decision result for input vectors. More specifically, the input vector  $x$  belongs to class  $k$  if  $p_k(x) > p_{k'}(x)$  for all  $k' = 1, 2, \dots, K$  and  $k \neq k'$ .

Hence, the main purpose of training PNN is to find the optimal estimate of probability density function according to the training samples and their labels, to ensure that the classifier works at the condition of minimum error rate and risk. When the samples to be identified are sent to the pattern layer, the output of each neuron is calculated according to the trained density function. Finally, the identified results are obtained through computations in the summation layer and output layer. Due to the following advantages, it is a wise choice to use PNN as a further classifier to classify signals [31]:

- (1) It has a simple structure, and it is easy to train. In the PNN based on probability density function estimation, the weight of the neuron in pattern is directly taken from the input sample value.
- (2) The training process of the network is simple, and there is no need to retrain for a long time when adding or reducing the number of groups.
- (3) It is not easy to produce local optimal solution, and its precision is higher than that of other classification approaches. No matter how complex the classification problem is, as long as there are enough training

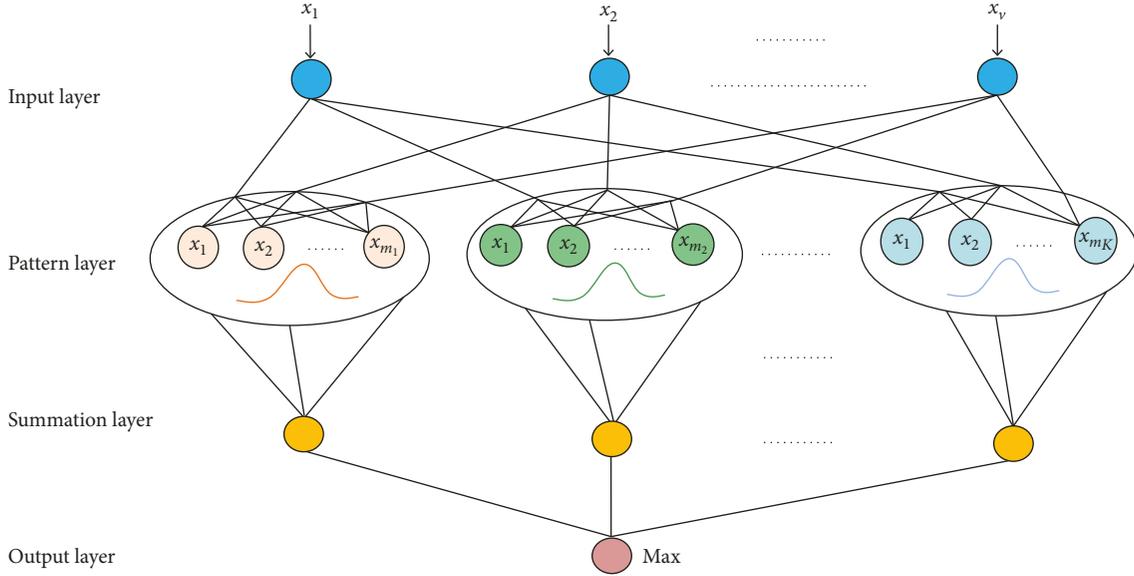


FIGURE 2: Flowchart of the PNN.

samples, the optimal solution under the Bayes criterion can be obtained.

### 3. Classification Scheme and Experiments of Emitter Signals

*3.1. Flowchart and Algorithms of Classification Scheme.* The flowchart of the proposed classification algorithms is shown in Figure 3.

Figure 3 indicates that the proposed scheme is composed of four modules, that is, data processing module, preclassification module, evaluation module, and accurate classification module. In the evaluation module, the clustering validity indexes are used to determine the range of  $K$  if it is unknown. For each  $K \in [K_{\text{Min}}, K_{\text{Max}}]$ , the classification validity index  $D$  is calculated as follows [16]:

$$D(K) = \frac{K \sum_{j=1}^{N_1} \max_{1 \leq k \leq K} \{p_{kj}\} - N_1}{N_1 (K - 1)}, \quad (20)$$

where  $N_1$  is the number of input vectors,  $p_{kj}$  is the element of the matrix  $Q$  of size  $K \times N_1$  in the output of PNN's pattern layer representing the membership of the  $j$ th input vector to the cluster  $k$ . When  $N_1 = 1$ ,  $p_{k1}$  is equal to  $p_k(x)$  presented in (19). When  $N_1 > 1$ , matrix  $Q = (p_{kj})_{K \times N_1}$  can be obtained by PNN.  $\max_{1 \leq k \leq K} \{p_{kj}\}$  is the largest element of the  $j$ th column in the matrix  $Q$ . Equation (20) indicates that  $D(K)$  is a nonlinear function related to  $K$ , when  $N_1 \neq \sum_{j=1}^{N_1} \max_{1 \leq k \leq K} \{p_{kj}\}$ . Therefore,  $K$  corresponding to the maximum value of  $D(K)$  is the optimal number of clusters  $K^*$ .

The pseudocodes are listed in Algorithm 1 if  $K$  is known.

If the classification number  $K$  is unknown, WCM and clustering validity indexes are used to determine the range of  $K$ . The corresponding pseudocodes are listed in Algorithm 2.

The algorithms show that the supervised learning PNN is used to classify samples. Therefore, the training (teaching) samples must be selected first. By Ward clustering method, we have obtained a preliminary classification of all samples. That is, the identities of some samples have been determined, except for some boundary points, which need to be further determined by the trained PNN. Therefore, some labeled proximity points  $x_{kj}$  ( $k = 1, 2, \dots, K, j = 1, 2, \dots, J_k$ ) around the center  $c_k$  can be selected to train PNN, where  $J_k$  represents the number of samples selected in class  $k$  and it should be preset, such as  $J_k = [a \cdot |S_k|]$ ,  $a \in [0.6, 0.8]$ .

*3.2. Classification Experiments.* A signal set  $Rs$  sampled from some pulse emitters is used to test the effectiveness of the proposed algorithms. Each emitter emits continuous signals in the pulse state. After a period of time, the receiver will receive multiple signals from all emitters. These signals are converted into digital signals by the analog-digital converter. The sampling frequency is 1.01 MHz. Signal samples  $y_i$  ( $i = 1, 2, \dots, 500$ ) are randomly extracted from these digital signals. The signal set  $Rs = [y_1, y_2, \dots, y_{500}]^T$  and the dimension of  $y_i$  is 1024. Considering that each  $y_i$  is disturbed by signals from other emitters, the mean value of all signals is used as the noise signal  $Is = [Is(1) Is(2) \dots Is(1024)]$ , where  $Is(n) = \sum_{i=1}^{500} y_i(n)/500$ ,  $n = 1, 2, \dots, 1024$ . First, the self-adaptive filtering is applied to process these signals. In this algorithm, the actual signal  $x(n)$  corresponds to  $y_i(n)$  and the noise signal  $x_0(n)$  corresponds to  $Is(n)$ . Then, Fourier transform is used to obtain the amplitude spectrum of all processed signals. The amplitude spectrum of the thirteenth signal is shown in Figure 4.

Figure 4(a) shows the sampling signal contains obvious white noise, which makes the feature of the useful signal unclear. However, most of the noises are removed after using the self-adaptive filtering. Moreover, the characteristics of the signals are highlighted so that the amplitude spectrum of

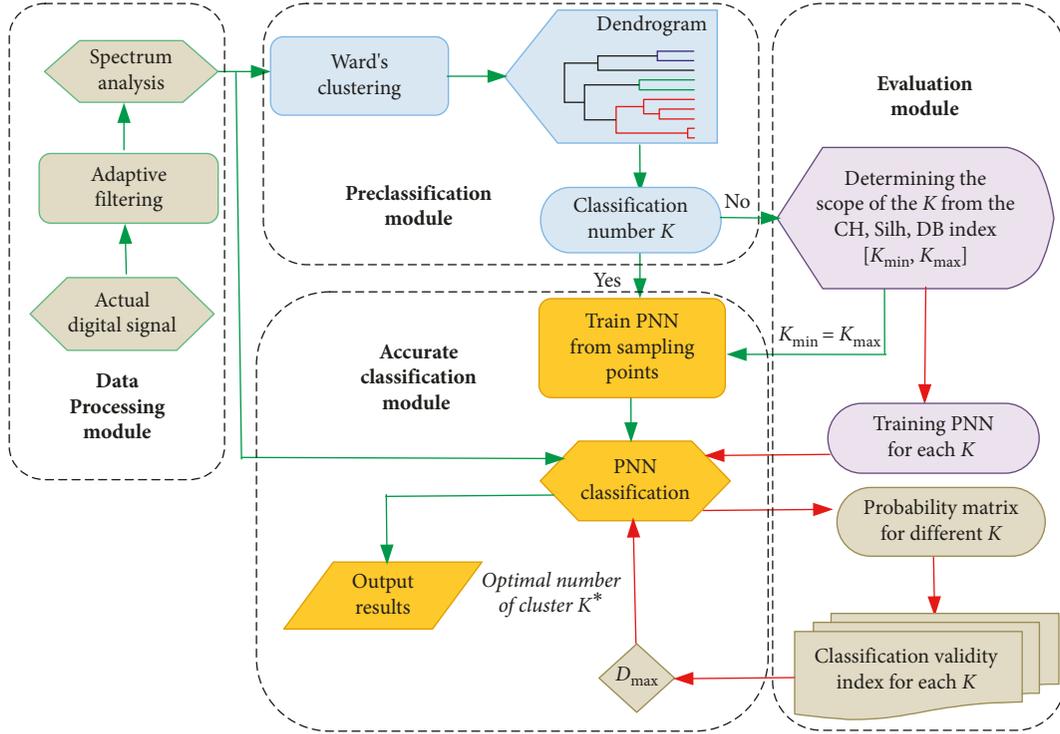


FIGURE 3: Flowchart of signal classification.

**Input:** The original signal vectors  $y_i$  that need to be classified.  
The classification number  $K$ .

**Output:** The classified label vector  $Cy$ .  
Compute  $s_i$  by using self-adaptive filtering for  $y_i$ ;  
Compute frequency spectrum  $G_i$  of  $s_i$ ;  
Compute center  $c_k$  of each class  $k$  by using WCM;  
Select training samples  $x_{kj}$  around  $c_k$  and record their labels  $Lx_{kj}$ ;  
Create the PNN classifier by using  $x_{kj}$  and  $Lx_{kj}$ ;  
Determine the class of the remaining samples.

**End:** Classification Algorithm 1.

ALGORITHM 1: Classification Algorithm 1.

them can be analyzed correctly. For these transformed signals, the clustering dendrogram can be obtained by using the WCM, and it is shown in Figure 5.

When the signals are divided into 3, 4, 5, and 6 classes, the intercluster distances are 75.5, 57.5, 27, and 14, respectively. The increments of the distance between the clusters are 13, 30.5, and 18 in turn. If the number of elements in each class is required relatively close, the ideal number of classifications is 3, 4, and 5. However, these numbers need to be further determined by the clustering validity indexes. The number of clusters  $K$  corresponding to different evaluation indexes is shown in Figure 6.

Figure 6 shows that the optimal number of clusters is 3 when the DB index and the Silh index are used, and it is 2 when the CH index is used. However, it becomes to 5 when the Gap index is used. Therefore, the optimal number of classifications should belong to the interval [2, 5]. In this case, the PNN needs to be used to obtain more accurate results.

For each  $K$  in this interval, seventy ( $a = 0.7$ ) sample points nearby each  $c_k$  are selected to train PNN classifiers. Then, the optimal classifier is obtained by calculating the maximum value of  $D(K)$ . The results have shown that  $D(5) = 1$ , while the other values are less than  $D(5)$ . That is to say, the optimal number of classifications is 5. Since the size of matrix  $R_s$  is  $500 \times 1024$ , three columns are randomly selected as the  $X$ -axis, the  $Y$ -axis, and the  $Z$ -axis to plot the classification results diagram. Let

$$Ac_k = \begin{pmatrix} y_{1k}(1) & y_{1k}(2) & \cdots & y_{1k}(1024) \\ y_{2k}(1) & y_{2k}(2) & \cdots & y_{2k}(1024) \\ \vdots & \vdots & & \vdots \\ y_{M_k k}(1) & y_{M_k k}(2) & \cdots & y_{M_k k}(1024) \end{pmatrix}, \quad (21)$$

represents a matrix consisting of all signals in class  $k$  after classifying, where  $M_k$  is the number of samples in class  $k$ ,  $y_{jk}$

**Input:** The original signal vectors  $y_i$  that need to be classified.  
**Output:** The classified label vector  $Cy$ .  
 Compute  $s_i$  by using self-adaptive filtering for  $y_i$ ;  
 Compute frequency spectrum  $G_i$  of  $s_i$ ;  
 Compute Ward's clustering dendrogram;  
 Compute  $CH(K)$ ,  $Silh(K)$ ,  $DB(K)$  and  $Gap(K)$ ; Compute  $K_{\min}, K_{\max}$ ;  
 if  $K_{\min} = K_{\max}$ , then  
   Compute center  $c_k$  of each class  $k$ ;  
   Select training samples  $x_{kj}$  around  $c_k$  and record their labels  $Lx_{kj}$ ;  
   Create the PNN classifier by using  $x_{kj}$  and  $Lx_{kj}$ ;  
   Determine the class of the remaining samples;  
   Output the classified label of  $y_i$ ;  
 else for  $K = K_{\min} : K_{\max}$   
   for  $k = 1 : K$   
     Compute center  $c_k$  of each class  $k$ ;  
     Select samples  $x_{kj}$  around  $c_k$  and record their labels  $Lx_{kj}$ ;  
     Create the PNN classifier by using  $x_{kj}$  and  $Lx_{kj}$ ;  
     Determine the class of the remaining samples;  
     Compute matrix  $Q = (p_{kj})$  and  $D(K)$  for each  $K$ ;  
     Compute  $D_{\max}$  and the optimal number of classifications  $K^*$ ;  
     Output the classified label of  $y_i$ ;  
 end  
 end  
**End:** Classification Algorithm 2.

ALGORITHM 2: Classification Algorithm 2.

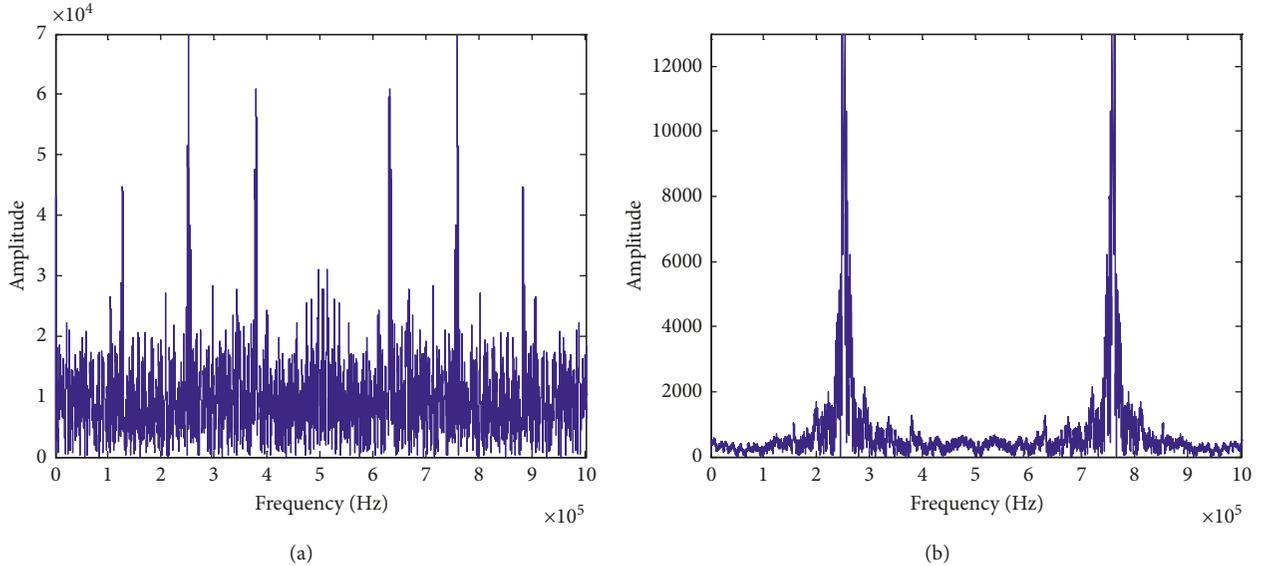


FIGURE 4: The double-sided amplitude spectrum of a sampling signal. (a) Before filtering; (b) after filtering.

is the  $j$ th sample in class  $k$ , and  $j = 1, 2, \dots, M_k$ . Thus,  $Ac = [Ac_1 \ Ac_2 \ \dots \ Ac_5]^T$  is a matrix of size  $500 \times 1024$ . If the data on columns  $a_1, a_2, a_3$  of  $Ac$  are selected to form a matrix  $\overline{Ac}$  of size  $500 \times 3$ , each row of  $\overline{Ac}$  is a three-dimensional array, that is, a point in the coordinate system. When  $a = 0.5, \sigma = 1$ , scatter plots of these data are shown in Figure 7. The first column of  $\overline{Ac}$  corresponds to the X-axis, the second column of  $\overline{Ac}$  corresponds to the Y-axis, and the third column corresponds to the Z-axis.

The experimental results show that all signal samples are divided into five classes, and each class contains 100 signals. Therefore, all signals can be thought to come from five emitters, and each emitter emits 100 signals. Although only three distribution figures of all classified signals are presented in Figure 7, in fact, in our experiments, we have obtained more than 1000 scatter plots which are drawn by randomly selecting three columns from matrix  $Ac$ . In these classification results, the distributions of sample sets are

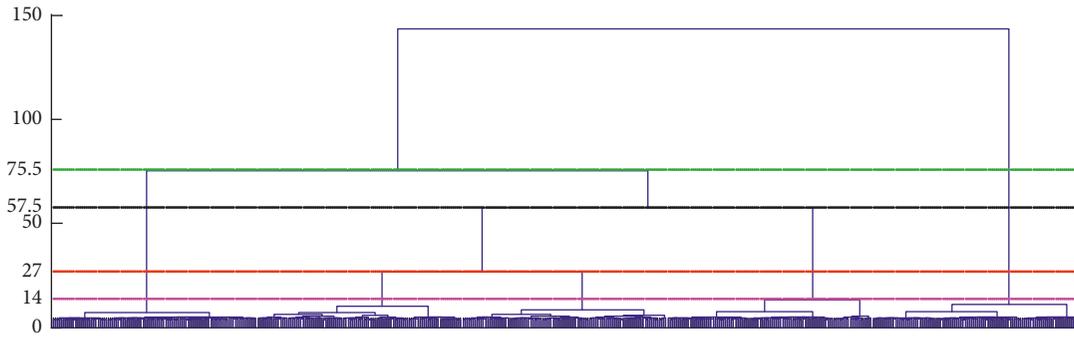


FIGURE 5: The dendrogram of the signals after using the WCM.

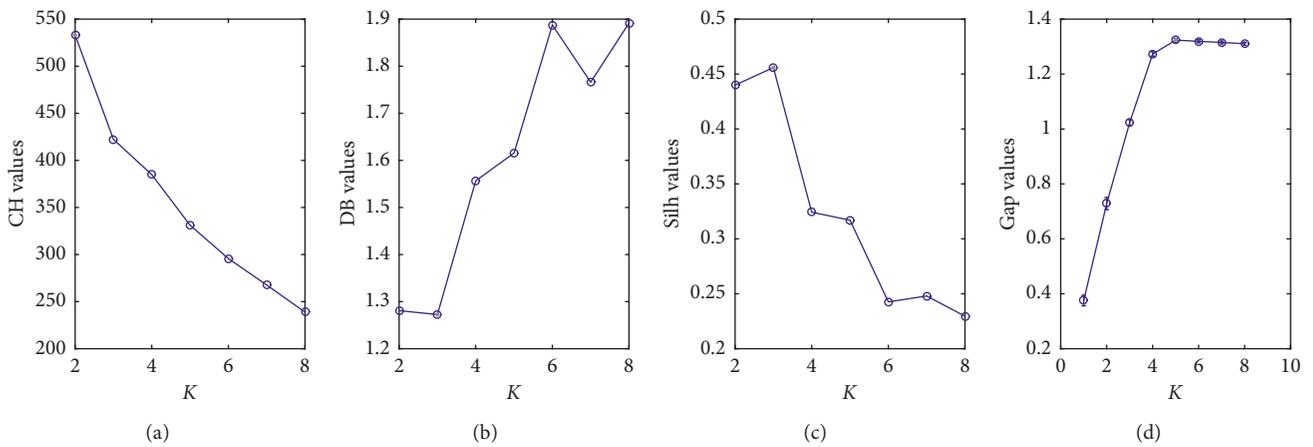


FIGURE 6: The number of clusters  $K$  versus the different evaluation indexes. (a) The number of clusters  $K$  versus CH values. (b) The number of clusters  $K$  versus DB values. (c) The number of clusters  $K$  versus Silh values. (d) The number of clusters  $K$  versus Gap values.

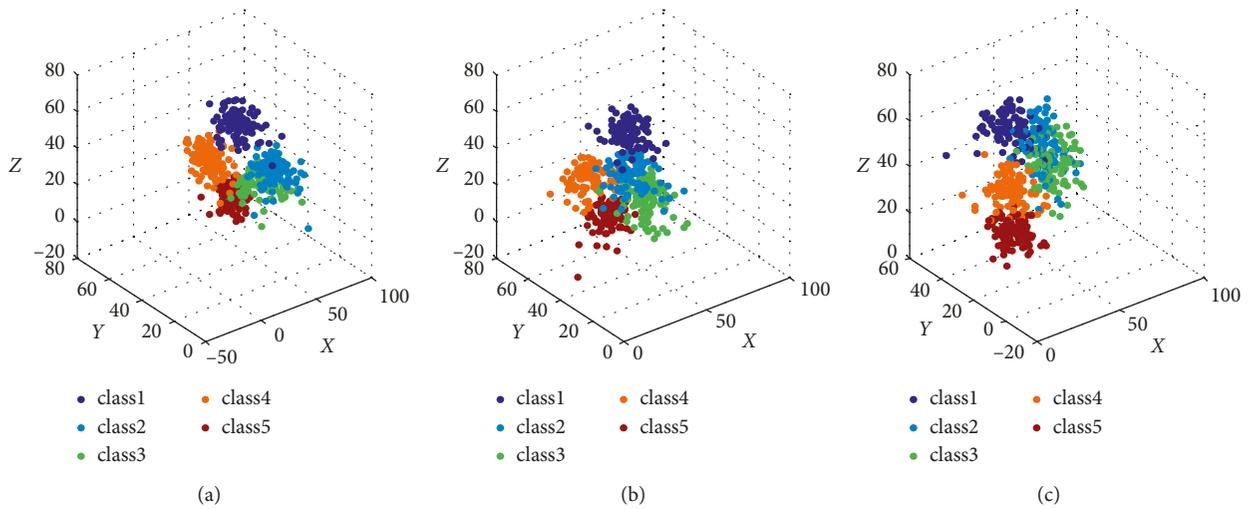


FIGURE 7: The scatter plots of classification results. (a) The classified results when  $a_1 = 245, a_2 = 562, a_3 = 720$ . (b) The classified results when  $a_1 = 268, a_2 = 369, a_3 = 547$ . (c) The classified results when  $a_1 = 139, a_2 = 367, a_3 = 741$ .

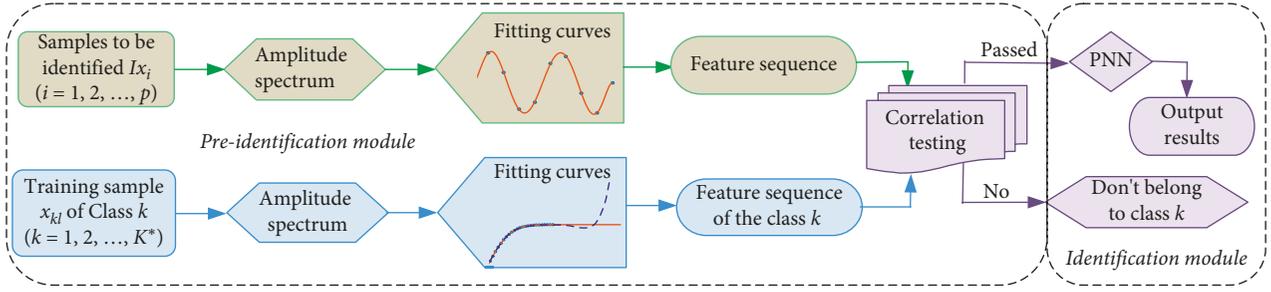


FIGURE 8: Flowchart of signal identification.

similar and the separations of them are obvious. Therefore, the proposed methods can effectively distinguish signals from different emitters.

#### 4. Identification Scheme and Experiments of Emitter Signals

**4.1. Flowchart and Algorithms of Identification Scheme.** Besides classification, data identification is an important function of PNN. Since PNN is based on the maximum posterior probability, it will give the optimal solution under the Bayesian criterion, whether or not the samples to be identified  $I x_i$  belong to the five determined classes that have been obtained in Section 3. So, if a sample belongs to one of them, it will accurately identify it. However, it may lose its function when the sample does not belong to them. At this time, the amplitude spectrum of all samples can be analyzed before, so that they can be identified whether they belong to the determined classes in advance. Considering that it is difficult to comparatively analyze the amplitude spectrum of the signals to be identified and each signal in every class, we adopt the curve fitting method to find the feature sequences of each class and the signals to be identified. Finally, the correlation degree of these sequences is calculated to get the preliminary identity information of the samples to be identified. This method is called Bivariable correlation analysis, and it is introduced as follows.

**Step 1. Simplifying amplitude spectrum.** Let  $L_i$  represent the length of the signal  $I x_i$ ,  $F s_i$  represent the Fourier transform of  $I x_i$ ,  $|F s_i|$  is the amplitude spectrum of  $I x_i$ ,  $|F s_i| = [g_i(1) g_i(2) \cdots g_i(L_i)]$ . The sequence  $F_i$  of length  $N_2$  is extracted from  $|F s_i|$  which takes  $\Delta T_i$  as the step size, where  $\Delta T_i = \lfloor L_i / N_2 \rfloor$ . For each training sample  $x_{kl}$  in class  $k$ , the corresponding sequence  $F_{kl}$  can be obtained according to the same method, where  $k = 1, 2, \dots, K^*$ ,  $l = 1, 2, \dots, L$ ,  $i = 1, 2, \dots, p$ , and  $L$  represents the number of training samples in class  $k$ , and  $p$  represents the number of samples to be identified.

**Step 2. Fitting curve.** The fitting curve  $\hat{z}_i$  can be obtained by fitting the sequence  $F_i$  and the fitting curve  $\hat{y}_{kl}$  in class  $k$  can be obtained by fitting the sequence  $F_{kl}$ .

**Step 3. Constructing the feature sequence.** First, for different  $k$ ,  $\hat{y}_{kl}(t)$ ,  $t = 1, 2, \dots, T$  can be calculated after giving the

upper bound  $T$ . Then, the  $t$ th signal feature of the class  $k$  can be obtained by the following equation:

$$\hat{Y}_k(t) = \frac{1}{L} \sum_{l=1}^L \hat{y}_{kl}(t). \quad (22)$$

Finally,  $C_s(k) = \{\hat{Y}_k(1), \hat{Y}_k(2), \dots, \hat{Y}_k(T)\}$  is the feature sequence of the class  $k$ . Similarly, the feature sequences  $Z_i = \{\hat{Z}_i(1), \hat{Z}_i(2), \dots, \hat{Z}_i(T)\}$ , ( $i = 1, 2, \dots, p$ ) of the samples to be identified can be obtained.

**Step 4. Correlation analysis.** The correlation coefficient between the  $Z_i$  and the  $C_s(k)$  can be calculated as the following equation:

$$r_{ik} = \frac{T(\sum_{t=1}^T \hat{Y}_k(t) \cdot \hat{Z}_i(t)) - (\sum_{t=1}^T \hat{Y}_k(t)) \cdot (\sum_{t=1}^T \hat{Z}_i(t))}{\sqrt{[T(\sum_{t=1}^T \hat{Y}_k^2(t)) - (\sum_{t=1}^T \hat{Y}_k(t))^2] \cdot [T(\sum_{t=1}^T \hat{Z}_i^2(t)) - (\sum_{t=1}^T \hat{Z}_i(t))^2]}} \quad (23)$$

where  $T(\sum_{t=1}^T \hat{Y}_k(t) \cdot \hat{Z}_i(t)) - (\sum_{t=1}^T \hat{Y}_k(t)) \cdot (\sum_{t=1}^T \hat{Z}_i(t))$  represents the sample covariance of  $C_s(k)$  and  $Z_i$ ,  $[T(\sum_{t=1}^T \hat{Y}_k^2(t)) - (\sum_{t=1}^T \hat{Y}_k(t))^2]$  and  $[T(\sum_{t=1}^T \hat{Z}_i^2(t)) - (\sum_{t=1}^T \hat{Z}_i(t))^2]$  are the sample standard variation of  $C_s(k)$  and  $Z_i$ , respectively. The performance of the correlation test indicates that the sample to be identified belongs to the corresponding class when  $r \geq 0.95$ . All samples that do not satisfy this condition should be removed. Finally, the remaining samples can be effectively identified by the trained PNN. The flowchart of the proposed identification scheme is shown in Figure 8.

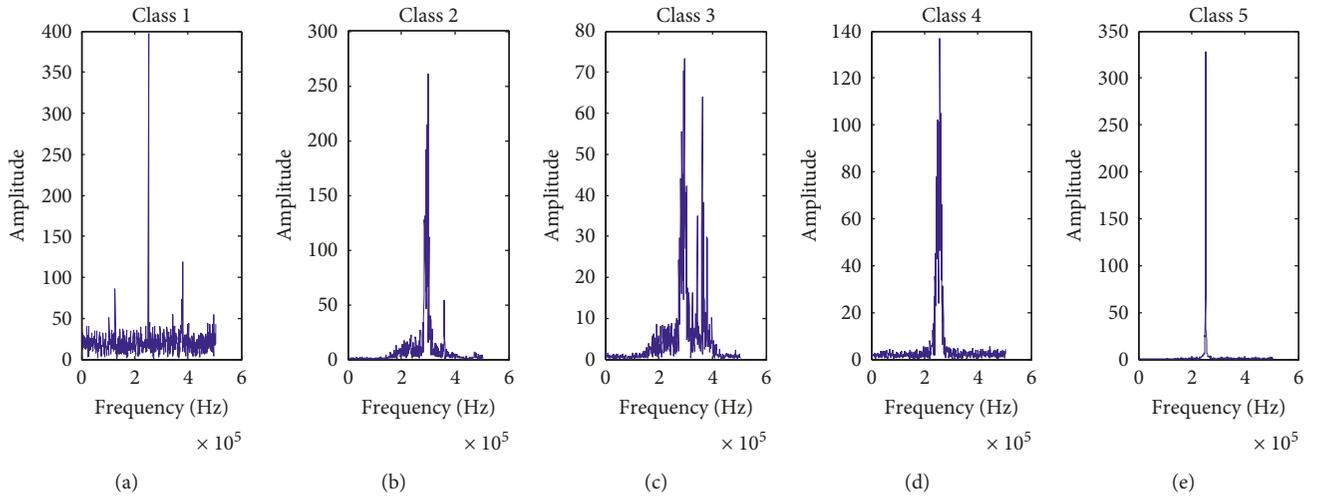
Figure 8 shows that the flowchart mainly includes two blocks. One is the preidentification module, and the other is the identification module. The role of the previous module is to eliminate the sample with a small correlation to the determined classes. When the correlations between the samples to be identified and several different classes are high, PNN will accurately identify it based on Bayesian criteria. The pseudocodes are given in Algorithm 3, which is called identification algorithm.

Through the methods proposed in Section 3, all signal samples have been classified to five classes. That is to say, the identity of each sample has been determined. Thus, the identification algorithm consists of two parts. On the one hand, some proximity points  $x_{kl}$  around the center  $c_k$  of class  $k$  are selected to train PNN,  $k = 1, 2, \dots, K^*$ ,  $l = 1, 2, \dots, L$ .  $x_{kl}$  is the input vector of the network, and its label is

**Input:** The training samples  $x_{kl}$ , label matrix  $Bx_{kl}$ .  
Samples to be identified  $Ix_i$ .

**Output:**  $Ix_i$  is the element of the class  $k$ .  
Normalize  $x_{kl}$ ,  $Ix_i$ ;  
Create PNN model by using  $x_{kl}$  and  $Bx_{kl}$ ;  
Adjust the dimension of  $Ix_i$  to make it consistent with the  $x_{kl}$ ;  
Determining sequences  $F_i$  and  $F_{kl}$ ;  
Compute fitting curves  $\hat{z}_i$ ,  $\hat{y}_{kl}$ ;  
Create feature sequences  $Cs(k), Z_i$ ;  
Compute  $r_{ik}$ , and if  $r_{ik} \geq 0.95$ , then  
    Put the  $Ix_i$  into the trained PNN;  
    else  
         $Ix_i$  does not belong to any class;  
**End:** Identification algorithm.

ALGORITHM 3: Identification algorithm.

FIGURE 9: The single-sided amplitude spectrum of class  $k, k = 1, 2, \dots, 5$ . (a)  $k = 1$ ; (b)  $k = 2$ ; (c)  $k = 3$ ; (d)  $k = 4$ ; (e)  $k = 5$ .

the output of the system. On the other hand, the trained PNN classifier is used to determine the identity of  $Ix_i$ ,  $i = 1, 2, \dots, p$ .

If the dimension of  $Ix_i$  is greater than  $x_{kl}$ , the method of adding time window can be used to adjust their dimensions. Let  $D_x$  represent the dimension of  $x_{kl}$  and  $D_{Ix}$  represent the dimension of  $Ix_i$ . The time window  $Ts = \lfloor D_{Ix}/D_x \rfloor$  is used to reduce the value of  $D_{Ix}$ . Thus, the adjusted samples to be identified  $\bar{I}x_i = [Ix_i(1), Ix_i(1) + Ts, \dots, Ix_i(1) + (D_x - 1)Ts]$ . When  $D_{Ix} < D_x$ , the same method can be used to reduce the dimension of  $x_{kl}$  and make it consistent with  $D_{Ix}$ . At this moment,  $D_{Ix}$  and  $D_x$  need to be exchanged.

**4.2. Identification Experiments.** By adjusting some parameters of emitters, such as pulse width and output power, signals that are different from  $y_i (i = 1, 2, \dots, 500)$  can be obtained. Suppose that sample  $Ix_i (i = 1, 2, 3, 4)$  is randomly selected from these signals for identification and the dimension of  $Ix_i$  is 10240.  $x_{kl}$  represents the  $l$ th training

sample in class  $k$ ,  $Bx_{kl}$  is the label matrix. First,  $\bar{I}x_i = [Ix_i(1), Ix_i(1) + 10, \dots, Ix_i(1) + 10230]$  can be obtained according to the method of adding time window. Then, the single-sided amplitude spectrum of center  $c_k$  for class  $k$  is shown in Figure 9;  $k = 1, 2, \dots, 5$ . The double-sided amplitude spectrum of  $\bar{I}x_i (i = 1, 2, 3, 4)$  is shown in Figure 10. Finally, the identity of  $\bar{I}x_i (i = 1, 2, 3, 4)$  is preliminary determined by observing the similarity between these amplitude spectrum.

It is observed that the amplitude spectrum of  $\bar{I}x_1$  is similar to that of the signals in class 2 or class 3, and  $\bar{I}x_2$  is likely to belong to class 5. But, the class of  $\bar{I}x_3$  and  $\bar{I}x_4$  is difficult to determine only by observation. At this time, it is necessary to carry out correlation test for these samples. The test results are shown in Table 1 when  $N_2 = 300$  and  $T = 100$ .

Table 1 shows that  $r_{12} = 0.9704$  and  $r_{13} = 0.9945$ , which are all greater than 0.95. So,  $\bar{I}x_1$  might be the element of class 2 or class 3.  $\bar{I}x_2$  should be classified into class 5. These results are consistent with the observations.  $\bar{I}x_3$  and  $\bar{I}x_4$  should be removed before formal classification. Therefore, it is

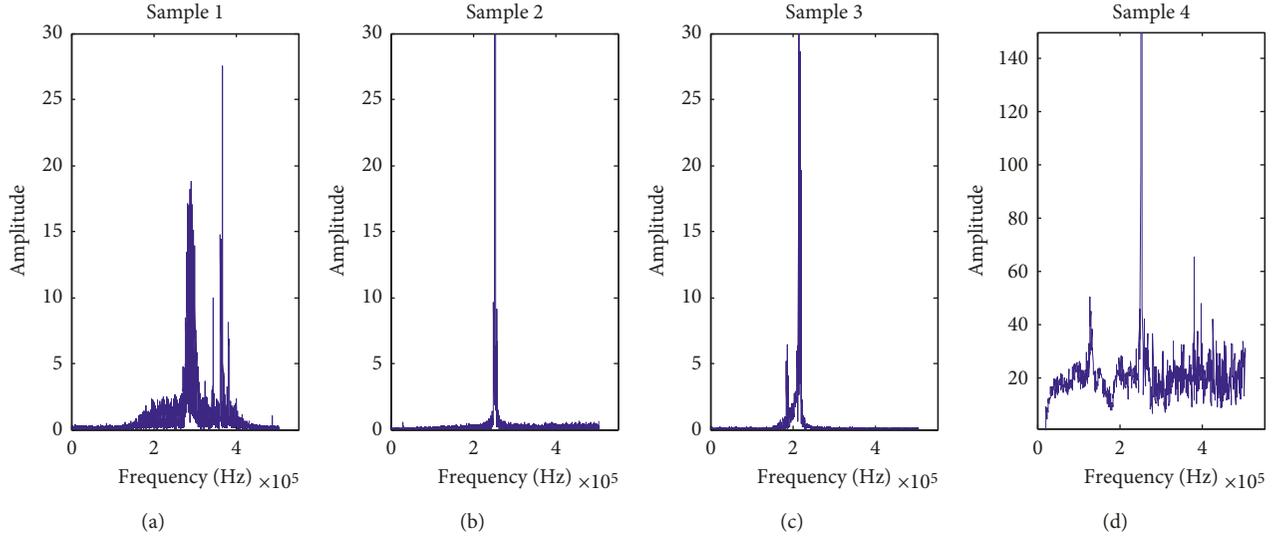


FIGURE 10: The double-sided amplitude spectrum of  $\bar{I}x_i$  ( $i = 1, 2, 3, 4$ ). (a)  $i = 1$ ; (b)  $i = 2$ ; (c)  $i = 3$ ; (d)  $i = 4$ .

TABLE 1: Correlation test of the samples to be identified.

| $r_{ik}$     | Class 1 | Class 2 | Class 3 | Class 4 | Class 5 |
|--------------|---------|---------|---------|---------|---------|
| $\bar{I}x_1$ | 0.3687  | 0.9704  | 0.9945  | 0.5473  | 0.6910  |
| $\bar{I}x_2$ | 0.7991  | 0.7447  | 0.6674  | 0.7882  | 0.9941  |
| $\bar{I}x_3$ | 0.2364  | 0.8907  | 0.7982  | 0.5202  | 0.5583  |
| $\bar{I}x_4$ | 0.7864  | 0.3572  | 0.3083  | 0.6887  | 0.6841  |

necessary to identify  $\bar{I}x_1$  and  $\bar{I}x_2$  by using PNN.  $L$  points are selected as training samples in each class to train PNN,  $L = 70$ . Finally, The trained PNN is used to judge the identity of  $\bar{I}x_1$  and  $\bar{I}x_2$ . Since  $Ix_1$  corresponds to  $\bar{I}x_1$ ,  $Ix_2$  corresponds to  $\bar{I}x_2$ , we can obtained the identity of  $Ix_1$  and  $Ix_2$  by identifying  $\bar{I}x_1$  and  $\bar{I}x_2$ . The identified results are shown in Figure 11 when  $\sigma = 1$ .

In order to obtain intuitive results, the data on columns 41, 256, 682 of  $Ix_1$  are selected to form a three-dimensional array  $[Ix_1(41) Ix_1(256) Ix_1(682)]$ . Data on the same columns of  $Ac$  are selected to form a matrix of size  $500 \times 3$ . All data are drawn in Figure 11(a). Obviously,  $Ix_1$  should be the element in class 3 (black solid ball) and  $Ix_2$  should be the element in class 5 (red solid ball). The same results can also be obtained when the data on columns 35, 709, and 929 of  $Ix_1$  and  $Ac$  are selected. The corresponding results are drawn in Figure 11(b).

## 5. Comparative Experiments

The performance of the proposed algorithm can be reflected through comparative experiments. Therefore, it is necessary to compare it with some usual identification methods, such as support vector machine based schemes [32], particle swarm optimization and support vector machine based schemes [12], artificial neural networks and intelligent filter based schemes [33], PNN and simplified fuzzy adaptive resonance theory map neural networks based schemes [15], and fuzzy c-means based schemes [34]. When the class label

of each signal is known, the performance of the above methods can be compared by calculating the classification accuracy and the identification accuracy. The comparison results are shown in Table 2. The samples that cannot be correctly judged are shown in Figure 12.

Table 2 shows that the classification accuracy obtained from Liu's and our methods is 100%. The value of identification accuracy obtained from the proposed method also reaches 100%. Figure 12(a) shows that some algorithms fail to give the correct classification results for some signals. In terms of signal classification, the number of samples without correct classification for each method is 9, 0, 18, 11, 27, and 0. By comparison, the identification accuracy derived from Zarei's and Cannon's scheme is poor, and they fail to identify the sample 3 and 4 (Figure 12(b)). This is mainly because artificial neural networks and fuzzy c-means algorithm can give a judgment result for each signal, whether or not the signal belongs to the determined classes. Although the PNN has some similar problems, they can be skillfully solved by the method of Bivariable correlation analysis. It makes the proposed method have unique advantages in signal identification.

## 6. Concluding Remarks

It is an indisputable fact that the probabilistic neural networks can be used to classify and identify patterns. It has a wide range of application, including the identification of emitter signals. In this paper, a novel classification and

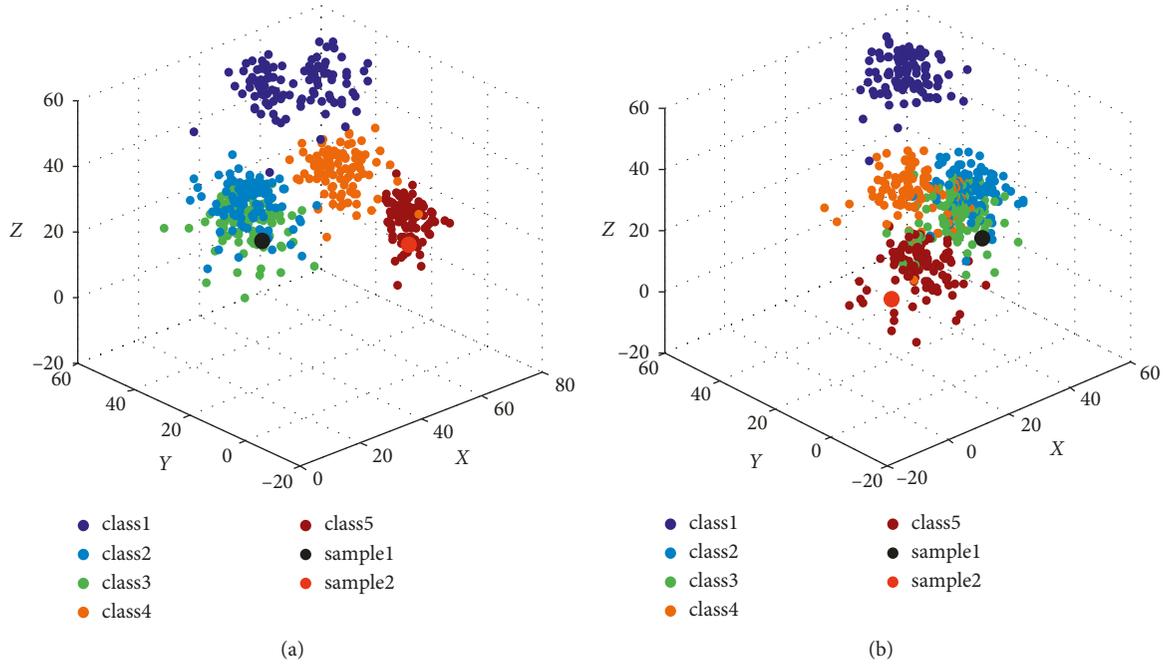


FIGURE 11: The identified results of  $[Ix_i(a_1)Ix_i(a_2)Ix_i(a_3)]$ ,  $i = 1, 2$ . (a)  $a_1 = 41, a_2 = 256, a_3 = 682$  and  $i = 1$ ; (b)  $a_1 = 35, a_2 = 709, a_3 = 929$  and  $i = 2$ .

TABLE 2: Accuracy comparison of different methods ( $K = 5$ ).

| Literature                  | Gryllias's [32] | Liu's [12] | Zarei's [33] | Jaouher's [15] | Cannon's [34] | This work |
|-----------------------------|-----------------|------------|--------------|----------------|---------------|-----------|
| Classification accuracy (%) | 98.20           | 100.00     | 96.40        | 97.80          | 94.60         | 100.00    |
| Identification accuracy (%) | 75.00           | 75.00      | 50.00        | 75.00          | 50.00         | 100.00    |

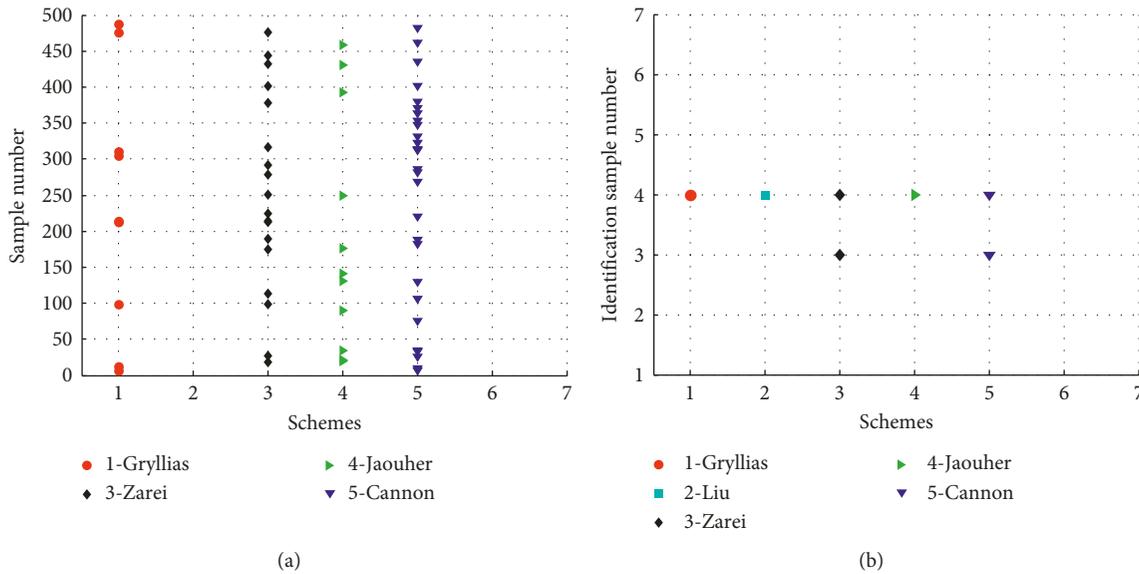


FIGURE 12: The samples that cannot be judged correctly by different schemes. (a) The samples that cannot be correctly classified. (b) The samples that cannot be correctly identified.

identification scheme, which is designed by the WCM and the PNN with correlation analysis, has been proposed for emitter signals. The scheme starts with self-adaptive filtering processing and spectrum analysis, then the WCM and clustering validity indexes including CH, Silh, DB, and Gap are utilized to determine the range of the optimal number of clusters. For different classification number  $K$ , 70 samples nearby each center are selected as training samples to establish PNN classifiers. Finally, the optimal PNN classifier which is used to identify signals is determined by the maximum of the classification validity index  $D(K)$ . At this stage, the method of Bivariable correlation analysis is cleverly used to improve the identification accuracy of PNN classifier. Experiments show that the proposed method can obtain higher accuracy, and it is more stable than other schemes in identification problems.

Finally, it should be pointed out that the scheme presented above is mainly used to classify the obtained signals, which are derived from some pulse emitters. The classification and identification of signals derived from some continuous wave emitters or the mixed emitters of these two types are the next topic to be studied. In addition, the proposed method can also be used to identify other signals, such as, biomedical signals and monitoring signals of digital virtual assets. When the data set to be classified are not numeric, they can first be converted to binary string and then converted to the required format.

## Data Availability

Data involve secrets and need to be kept confidential.

## Conflicts of Interest

The authors declare that there are no conflicts of interests regarding the publication of this paper.

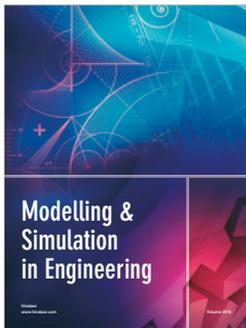
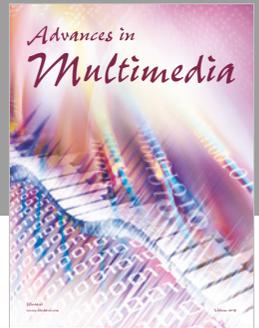
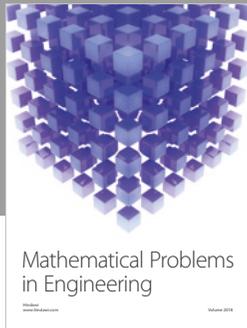
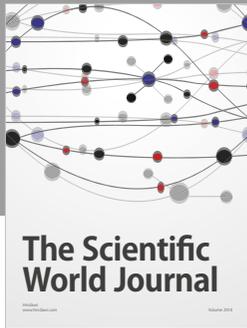
## Acknowledgments

This study was funded in part by the National Key Research and Development Program of China under Grant 2016YFB0800601 and in part by the National Natural Science Foundation of China under Grants 61472331 and 61772434.

## References

- [1] N. Visnevski, *Syntactic Modeling of Multi-Function Radars*, PhD's thesis, McMaster University, 2005.
- [2] A. Wang and V. Krishnamurthy, "Threat estimation of multi-function radars: modeling and statistical signal processing of stochastic context free grammars," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 793–796, Honolulu, HI, USA, 2007.
- [3] L. Dai, B. Wang, and B. Cai, "A method for states estimation of multi-function radar based on stochastic context free grammar," *Journal of Air Force Engineering University*, vol. 15, no. 3, pp. 31–39, 2014.
- [4] M. Jan and L. Paradowski, "A knowledge based approach for emitter identification," in *Proceedings of International Conference on Microwaves and Radar (MIKON'98)*, vol. 3, pp. 810–814, Krakow, Poland, 1998.
- [5] M. Jan, *Specific Emitter Identification*, in *Proceedings of International Radar Symposium*, pp. 1–4, Krakow, Poland, 2008.
- [6] J. Dudczyk, "The concept of ELINT DataBase based on ERD modelling," *Elektronika*, vol. 2018, pp. 34–37, 2018.
- [7] C. S. Shieh and C. T. Lin, "A vector network for emitter identification," *IEEE Transactions on Antennas and Propagation*, vol. 50, no. 8, pp. 1120–1127, 2002.
- [8] B. Tang and H. Guang rui, "Recognition of radar signal using immune RBF network," *Journal of Data Acquisition and Processing*, vol. 17, no. 4, pp. 371–375, 2002.
- [9] J. Dudczyk, "A method of feature selection in the aspect of specific identification of radar signals," *Bulletin of the Polish Academy of Sciences, Technical Sciences*, vol. 65, no. 1, pp. 113–119, 2017.
- [10] G. Zhang, Y. Zhou, and W. Jiang, "A method of Radiator identification based on Bayes theory," *Signal Processing*, vol. 20, no. 4, pp. 350–352, 2004.
- [11] G. X. Zhang, H. N. Rong, and W. D. Jin, "Application of support vector machine to radar emitter signal recognition," *Journal of Southwest Jiaotong University*, vol. 41, no. 1, pp. 25–30, 2006.
- [12] Z. Liu, H. Cao, X. Chen, and Z. He, "Multi-fault classification based on wavelet SVM with PSO algorithm to analyze vibration signals from rolling element bearings," *Neurocomputing*, vol. 99, no. 1, pp. 399–410, 2013.
- [13] Q. Liu, "A novel radar emitters scheme recognition algorithm using support vector machine," *Information Technology Journal*, vol. 13, no. 4, pp. 725–729, 2014.
- [14] R. Hussain, "Synthetic Aperture Radar (SAR) images features clustering using Fuzzy c-means (FCM) clustering algorithm," *Computational Ecology and Software*, vol. 2, no. 4, pp. 220–225, 2012.
- [15] J. Ben Ali, L. Saidi, and A. Mouelhi, "Linear feature selection and classification using PNN and SFAM neural networks for a nearly online diagnosis of bearing naturally progressing degradations," *Engineering Applications of Artificial Intelligence*, vol. 42, pp. 67–81, 2015.
- [16] J. Iounousse, S. Er-Raki, and A. E. Motassadeq, "Using an unsupervised approach of Probabilistic Neural Network (PNN) for land use classification from multitemporal satellite images," *Applied Soft Computing*, vol. 30, pp. 1–13, 2015.
- [17] Z. Wu, *Advanced digital signal processing*, pp. 133–135, China Machine Press, Beijing, China, 2009.
- [18] C. Gou and L. Yi, "A variable step size LMS algorithm based on bell-shaped function," *Marine Electric and Electronic Engineering*, vol. 35, no. 11, pp. 31–39, 2015.
- [19] N. Yorek, I. Ugulu, and H. Aydin, "Using self-organizing neural network map combined with Ward's clustering algorithm for visualization of students' cognitive structural models about aliveness concept," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 2476256, 14 pages, 2016.
- [20] B. E. Husic and V. S. Pande, "Ward clustering improves cross-validated markov state models of protein folding," *Journal of Chemical Theory and Computation*, vol. 13, no. 3, pp. 963–967, 2017.
- [21] A. J. Gómez-Núñez, B. Vargas-Quesada, and F. de Moya-Anegón, "Updating the SCImago journal and country rank classification: a new approach using Ward's clustering and alternative combination of citation measures," *Journal of the Association for Information Science and Technology*, vol. 67, no. 1, pp. 178–190, 2015.
- [22] O. Arbelaitz, I. Gurrutxaga, and J. Muguerza, "An extensive comparative study of cluster validity indices," *Pattern identification*, vol. 46, no. 1, pp. 243–256, 2013.

- [23] S. Lukasik and P. A. Kowalski, "Clustering using flower Pollination algorithm and calinski-harabasz index," *IEEE Congress on Evolutionary Computation (CEC)*, pp. 2724–2728, 2016.
- [24] R. Cordeiro de Amorim and C. Hennig, "Recovering the number of clusters in data sets with noise features using feature rescaling factors," *Information Sciences*, vol. 324, pp. 126–145, 2015.
- [25] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, no. 20, pp. 53–65, 1987.
- [26] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1*, no. 2, pp. 224–227, 1979.
- [27] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society*, vol. 63, no. 2, pp. 411–423, 2001.
- [28] S. R. Mohanty, P. K. Ray, N. Kishor, and B. K. Panigrahi, "Classification of disturbances in hybrid DG system using modular PNN and SVM," *International Journal of Electrical Power and Energy Systems*, vol. 44, no. 1, pp. 764–777, 2013.
- [29] S. Wang, X. Li, S. Zhang, J. Gui, and D. Huang, "Tumor classification by combining PNN classifier ensemble with neighborhood rough set based gene reduction," *Computers in Biology and Medicine*, vol. 40, no. 2, pp. 179–189, 2010.
- [30] J. Iounousse, S. Er-Raki, and A. E. Motassadeq, "Using an unsupervised approach of Probabilistic Neural Network (PNN) for land use classification from multitemporal satellite images," *Applied Soft Computing*, vol. 30, pp. 1–13, 2015.
- [31] A. T. C. Goh, "Probabilistic neural network for evaluating seismic liquefaction potential," *Canadian Geotechnical Journal*, vol. 39, no. 1, pp. 219–232, 2002.
- [32] K. C. Gryllias and I. A. Antoniadis, "A Support Vector Machine approach based on physical model training for rolling element bearing fault detection in industrial environments," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 2, pp. 326–344, 2012.
- [33] J. Zarei, M. Amin Tajeddini, and H. Reza Karimi, "Vibration analysis for bearing fault detection and classification using an intelligent filter," *Mechatronics*, vol. 24, no. 2, pp. 151–157, 2014.
- [34] L. Cannon Robert, V. Dave Jitendra, J. C. Bezdek, and M. M. Trivedi, "Segmentation of a thematic Mapper image using fuzzy c-means clustering algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. GE-24, no. 3, pp. 400–408, 1986.



Hindawi

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

