IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 8, 2020, accepted May 27, 2020, date of publication June 8, 2020, date of current version June 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3000798

A CNN-RBPNN Model With Feature Knowledge Embedding and Its Application to Time-Varying Signal Classification

LU WU[1,2], YINGLONG WANG[1,2], SHAOHUA XU[1], KUN LIU[1], AND XUEGUI LI[1]

[1]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[2]Shandong Computer Science Center (National Supercomputer Center in Jinan), Jinan 250014, China

Corresponding author: Shaohua Xu (xush62@163.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC1406203, and in part by the Shandong University of Science and Technology Innovative Research Projects under Grant 2019TDJH102.

ABSTRACT A novel technique, combining the feature extraction mechanisms of a convolutional neural network (CNN) with the classification method of a radial basis probability neural network (RBPNN), is proposed for small sample set modeling and feature knowledge embedding in multi-channel time-varying signal classification. This CNN-RBPNN consists of a signal input layer, signal feature parallel extraction and integration units, and an RBPNN classifier. Each channel signal in a feature extraction unit corresponds to a 1D CNN. The extracted features are represented as feature vectors, and these vectors constitute a comprehensive feature matrix. The RBPNN classifier was designed using signal feature embedding mechanism based on radial basis kernels and the property of combining pattern subclasses into pattern classes to form complex class boundaries. A dynamic clustering algorithm was used to divide each pattern class sample into several subclasses. Typical signal samples in each pattern subclass were designated as kernel centers, in order to achieve signal categories features embedding. This process was also used to determine the number of nodes in the RBPN layer. The RBPN layer outputs were selectively summed in the pattern layer according to kernel center category, which can generate irregular class boundaries, reducing the overlap among different pattern class boundary. The proposed CNN-RBPNN replaces the full-connection layer and classifier unit of conventional CNN with RBPNN, which can extract and represent signals distribution features and structural properties, implement structural and data constraints. This can reduce the structural risks of small sample set modeling. In this study, the properties of CNN-RBPNN are analyzed and an integrated learning algorithm is proposed. An experiment was conducted using 12-lead ECG signals in a seven-classification in the case of small sample set. Results demonstrated that, the correct recognition rate is 5.7% higher than other methods in the experiment, the performance evaluation index also showed significant improvement.

INDEX TERMS Time-varying signal classification, CNN feature extraction, radial basis probability network, prior feature knowledge embedding, small sample set modelling.

I. INTRODUCTION

With the rapid development and application of intelligent sensor and internet of things (IOT) technology, multi-channel signal classification has become increasingly important in a variety of fields [1]–[3]. Nonlinear time-varying signals are a type of multi-component signal whose frequency and amplitude change with time. These signals can exhibit non-linear, non-stationary, and non-fixed probability distributions. They can also be characterized as multi-peak, stretched, drifting, or noisy and are often highly dependent on temporal changes. Combination process characteristics for multiple variable signals often exhibit a high degree of complexity, particularly in complex nonlinear dynamic systems [4], [5]. The classification and processing of such data are critical issues in the research fields of signal analysis and artificial intelligence [6]. However, the influence of uncertain factors, the unrepeatability of some process events in dynamic

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

systems, and mutual coupling in the signal system make it difficult to acquire a complete large-scale dataset. In addition, some events are highly rare, or sampling can be cost-prohibitive, often leading to a reliance on small or unbalanced sample sets for modeling and analysis [7]–[9]. Each of these issues complicates research in complex signal systems [10].

Artificial neural networks are a common and effective model used in signal processing and analysis. With the development of deep learning, various neural network architectures have been proposed for studying time-varying signals. These include deep convolutional neural networks [11], [12], automatic encoder neural networks [13], [14], recursive networks [15], recurrent networks [16], [17], and Markov chains [18], [19]. These algorithms can effectively implement the classification and analysis of time-varying signals in practical settings. However, they are also unstable when modeling small or imbalanced datasets. In these cases, classification accuracy and generalizability suffer, as the lack of constraints for domain prior knowledge affects key model properties [20].

Existing machine learning models for time-varying signal analysis can often be classified as 'end-to-end' black box systems, which require a complete dataset to support. As such, the ability to generalize these models to practical systems depends primarily on the model structure, information processing mechanism, learning strategy, and the scale and completeness of the sample set. Algorithms lacking enlighten and restriction mechanisms for domain knowledge often exhibit large exploration spaces and complex optimization processes. The classification of large-scale datasets depends primarily on the extraction, memory, and representation mechanisms for data features. However, it is difficult to analyze and explain deep level features based on distributed data representations and abstract high-level synthesis mechanisms.

Convolutional neural networks are a widely used deep learning model that offer several advantages for feature extraction and representation. The associated convolution operations, local connections, and weight sharing can effectively reduce the complexity of a network [21]. CNNs typically use a convolution kernel matrix, adopt a step traversal scanning strategy, and generate multiple feature surfaces for feature extraction and representation learning. A data structure in which the feature vector is consistent with the original signal can be maintained in the initial stage. However, with an increase in structure depth, high-level feature extraction and representation becomes abstract, which is not conducive to the analysis or interpretation of signal distributions or structural feature correspondence. This increase in depth also complicates the embedding of prior feature knowledge including category information.

As such, a novel feature extraction and representation methodology for time-varying signal classification, in which relationships are consistent between the time sequence feature matrix arrangement order and the original multi-channel signal, and the embedding and utilization for prior feature knowledge, can greatly improve the ability to feature analysis, interpretation and utilization. In addition, the embedding of typical feature knowledge can reduce the exploration process for model structure and the iterative learning space for model parameters, thereby increasing the stability, generalizability, and robustness of the model. Fewer model parameters and a stronger self-adaptive learning mechanism could further improve the modeling ability of small sample sets.

Huang proposed a radial basis probabilistic neural network (RBPNN) in 1999 [22]. This model consists of a signal input layer, a radial basis probability neuron (RBPN) hidden layer, a pattern layer, and a classifier. The RBPNN fully considers the staggered influence of multi-class patterns in recognition applications. This approach forms an effective feature interface and offers the advantages of fewer model parameters and lower computational complexity [23]. The primary information processing unit of an RBPNN is the RBPN, which adopts Gaussian radial basis kernel functions. The adjustable variables in this model are statistical parameters, such as the radial basis kernel center and variance. The radial basis kernel transform calculates the distance between input signals and kernel centers, which is essentially a measure of signal feature similarity. This process also provides an interface for the embedding of prior feature knowledge and an RBPN probability output.

In this study, CNN feature extraction and representation mechanisms are combined with the differentiation capabilities of RBPNN pattern class features and the embedding mechanisms of category feature knowledge. A multi-channel time-varying signal classification model CNN-RBPNN is established using modularization and layer-by-layer stacking. Input and feature extraction for multi-channel signals are designed as a parallel structure, with each channel input signal corresponding to a 1D CNN. Features extracted before CNN activation are used as single-channel signal features, which can more accurately represent signal shape distributions and maintain timing relationships. A regular uniform dimensional feature vector is formed after processing with the batch normalization (BN) layer. A comprehensive feature matrix can then be generated with a data structure correspondence relationship between multi-channel time-varying signals. The RBPNN was used in the design of classifiers based on a signal synthesis feature matrix, in which RBPN kernel centers directly affected RBPNN properties.

In the proposed technique, a dynamic clustering algorithm was used to divide each pattern into pattern subclasses with more similar characteristics. Typical samples with category feature marks in each pattern subclass could then be determined. After feature extraction and representation by CNN, as the kernel centers for each RBPN with a dynamic time-warping (DTW) algorithm being used to measure the similarity between time-varying signal distribution features. An exponential sigmoid function with probability membership was chosen as the RBPN action function, which was capable of determining RBPN probability outputs. The output of the radial basis layer was selectively summed in the

RBPNN pattern layer, according to RBPN kernel center category labels. That is, the membership probability of signal to pattern subclass is integrated into the membership probability to pattern class. In addition, pattern subclass boundaries were combined into irregular pattern class boundaries, which improved signal pattern feature discrimination. The Softmax classifier uses the output of the pattern layer as input, based on the maximum probability membership principle, to achieve classification of multi-channel time-varying signals.

The proposed CNN-RBPNN can extract single-channel distribution features and characterize multi-channel signal combination relationships. It can also embed typical category feature knowledge, represent and maintain the diversity of modal features, strengthen the role of signature categories in signal classification, and overcome the limitations of existing neural networks. This is primarily accomplished by forming complex irregular class boundaries, as opposed to conventional convex decision regions, thereby increasing the accuracy of classification and category feature identification. In addition, the embedding of prior pattern subclass feature knowledge imposes structural and data constraints in the RBPNN, which simplifies model structure and reduces the requirements for sample set completeness. This in turn decreases the required number of learning iterations and computational complexity for the modeling and analysis of small sample sets.

The diagnosis of cardiovascular disease using ECG signals is a typical multi-channel classification problem [24], [25]. Some diseases are not common in the clinic, which leads to the acquisition of fewer samples. ECG signals can also be characterized by multi-peak, irregular period, and drift stretching data, which complicates their structural properties. As a result, existing classification models for long-duration multi-beat ECG signals suffer from low accuracy and poor adaptability in distinguishing signal characteristics for small sample sets [26]. In this study, a CNN-RBPNN was used to conduct a classified diagnostic experiment with long 12-lead ECG signals consisting of 7 disease types, to verify the feasibility and effectiveness of the proposed model.

The remainder of this paper is organized as follows. After discussing the challenges of time-varying signal classification and the limitations of existing neural networks, a novel CNN-RBPNN classification model is developed and analyzed in Section 2. A comprehensive learning algorithm for the CNN-RBPNN is proposed in Section 3. A series of ECG signal classification experiments are then performed and the results are analyzed in Section 4. Finally, the study is summarized and the advantages and limitations of the proposed methodology are discussed in Section 5.

II. THE CNN-RBPNN MODEL

In this section, CNN-based time-varying signal feature extraction is combined with the classification mechanism of an RBPNN to develop the novel CNN-RBPNN classification model, which can embed prior diversity feature knowledge and classify multi-channel time-varying signals.

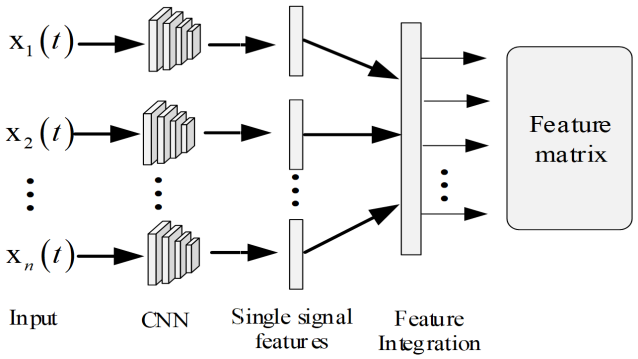

FIGURE 1. The parallel CNN model for multi-channel time-varying signal feature extraction.

A. THE PARALLEL EXTRACTION OF MULTI-CHANNEL SIGNAL FEATURES WITH A 1D CNN

A feature extraction and representation model, based on a parallel one-dimensional (1D) CNN structure, was developed for multi-channel time-varying signals. The associated model architecture and information workflow are shown in Fig. 1, where $x_1(t), x_2(t), \cdots, x_n(t)$ represent multi-channel time-varying input signals on the process interval $[0, T]$. In this expression, each $x_i(t)(i = 1, 2, \cdots, n)$ corresponds to a 1D convolutional network (CNN_i), composed of several alternating convolution and pooling layers. After the batch normalization layer processing, normalized eigenvectors were established with unified dimensions. On this basis, the signal eigenvectors in each channel were integrated into a synthesis eigenmatrix, which exhibited topological correspondence with multi-channel input signals. This matrix can simultaneously represent the distribution characteristics of single-channel signals and the structural characteristics of multi-channel signals. The full connection layer and classifier were removed from the 1D CNN and the extracted features were used prior to activation, forming a 1D convolutional neural network $\mathrm{CNN}_i(i = 1, 2, \cdots, n)$ for single-channel signal feature extraction.

This structure and the relevant parameter settings are shown in Fig. 2, where the convolution and pooling layers in each $\mathrm{CNN}_i(i = 1, 2, \cdots, n)$ are alternately stacked and connected. The convolution layer can be described as $C_j^{(i)}\left(P_j^{(i)} @ \left(l_j^{(i)} \times 1, s_j^{(i)}\right)\right), i = 1, 2, \cdots, n; j = 1, 2, \cdots, h$. The pooling layer can be represented by $S_j^{(i)}\left(P_j^{(i)} @ \left(l_j^{(i)} \times 1, s_j^{(i)}\right)\right)$. Here, $P_j^{(i)}$ is the number of feature planes in the convolution layer $C_j^{(i)}$, $l_j^{(i)}$ is the width of the convolution kernel, $w_j^{(i,p)} = \left(w_{j1}^{(i,p)}, w_{j2}^{(i,p)}, \cdots, w_{jl_j}^{(i,p)}\right)$ is a 1D convolutional kernel parameter corresponding to the p^{th} feature plane, and $s_j^{(i)}$ is the step length. The features extracted in each unit prior to activation were used as signal features. The activation functions in the convolution and pooling layers of the 1D CNN were then assigned as identity functions.

1) THE CONVOLUTION LAYER

The convolution layer is based on the original signal or the previous layer's feature graph. Convolution operations are

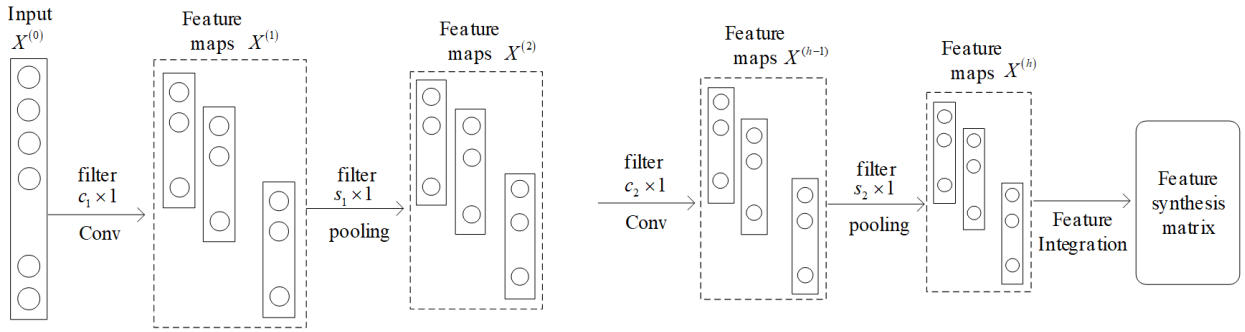

FIGURE 2. Structural and parameter settings for the one-dimensional CNN.

conducted using a learnable 1D convolution kernel and can be expressed mathematically as [27]:

$$y(t) = \sum_{i=1}^{m} x\,(t+i-1)\,w(i), \tag{1}$$

where $w(i)(i = 1, 2, \cdots, m)$ are convolution kernel parameters, m is the width of the convolution kernel, $t = 1, 2, \cdots, L-m+1$, and L is the length of the input signal $(L > m)$.

Output feature graphs for subsequent layers are obtained using 1D convolutions. These graphs can be represented by:

$$H_j^l = \sum_{i=1}^{k} H_i^{l-1} \times W_{ij}^{(l)} + b_j^l, \tag{2}$$

where H_j^l is the j^{th} feature graph in the l^{th} convolution layer, $W_{ij}^{(l)}$ represents a weighting matrix for the i^{th} graph in the $(l-1)^{th}$ layer to the j^{th} graph in the l^{th} layer, and "$\times$" indicates convolution. The term $H_i^{(l-1)}$ represents the i^{th} graph in the $(l-1)^{th}$ layer, i, j are the input and output indices, and b_j^l represents an offset corresponding to the output feature graph in the l^{th} layer.

2) THE POOLING OPERATION LAYER

The pooling layer down-samples the feature graph according to specified pooling rules. The number of input and output feature graphs remain the same, but the dimensions are reduced. The generating formula for H_j^l, the j^{th} feature graph in the l^{th} pooling layer, is given by [28]:

$$H_j^l = \beta_j^l pooling\left(H_i^{l-1}\right) + b_j^l, \tag{3}$$

where $pooling(\cdot)$ represents the pooling function. Each output feature graph corresponds to a multiplicative bias β_j^l and an additive bias b_j^l.

3) THE SYNTHESIS FEATURE MATRIX FOR MULTI-CHANNEL SIGNALS

It is evident from Fig. 2 that each single-channel signal passes through multiple alternating convolution and pooling layers. The signal is then regularized in the BN layer to form feature vectors with the same dimensions. As demonstrated by the structure of Fig. 1, feature vectors in each channel signal were integrated to form a comprehensive feature matrix of multi-channel signals.

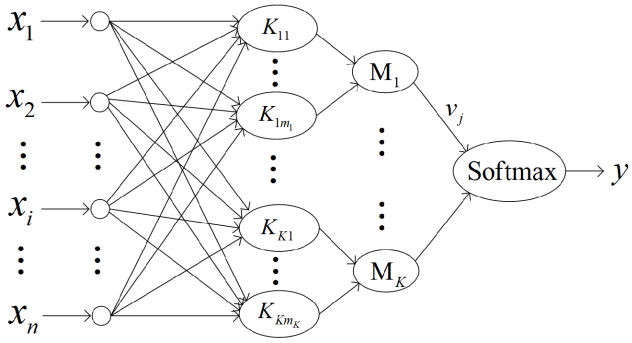

FIGURE 3. The RBPNN model.

In this study, time-varying signal features were extracted by the 1D CNN without the use of a classifier and full connection layer. A synthesis feature matrix was then generated, which is represented by convolution kernel and pooling parameters. The unique properties of these convolution and pooling operations, as well as the use of feature information prior to activation, result in a consistent topological correspondence between the comprehensive feature matrix and the original multi-channel signal. In addition, bias settings can reduce the influence of time-varying signal drift and scaling, producing more accurate distribution characteristics for single-channel signals and structural characteristics for multi-channel signals.

B. A RADIAL BASIS PROBABILISTIC NEURAL NETWORK

The RBPNN is a classification model based on Bayes decision theory, which offers high learning efficiency and distinguishing signals features ability [29], [30]. This RBPNN is composed of an input layer, a kernel transformation layer based on radial basis probability neurons, a pattern layer, and a Softmax classifier. This structure is shown in Fig. 3, where $x_1, x_2, \cdots, x_n$ are input signals in the RBPNN, $K(\cdot)$ is a radial basis probabilistic kernel function satisfying the Parzen window distribution, M_k is a pattern unit, $k = 1, 2, \cdots, K$, K is the number of signal sample set pattern classes, and m_k is the number of typical feature signals for the pattern class k. In this study, each typical feature signal was used as an RBPN kernel center.

As seen in Fig. 3, the relationship between the input and output in each RBPNN layer can be described as follows:

(1) *The input layer*: an n-dimensional signal $x = (x_1, x_2, \cdots, x_n)$ is input to the network, $x_i \in R$.

(2) *The kernel transformation layer*: the signal is received and fully connected to each node in the input layer. The output corresponding to the j^{th} kernel function unit can be expressed as:

$$u_j = \exp\left(\frac{\|x - z_j\|_F - 1}{\sigma^2}\right), \quad j = 1, 2, \cdots, m, \tag{4}$$

where $z_j (j = 1, 2, \cdots, m)$ are kernel centers, m is the number of radial basis nodes, $\|\cdot\|_F$ is the distance norm, and $K(\cdot)$ is an exponential Gaussian function.

(3) *The pattern layer*: the RBPN layer output was selectively summed according the class labels for each RBPN kernel center. The output of the pattern layer is given by:

$$q_k = \sum_{j \in \Omega_k} u_j, \tag{5}$$

where $k = 1, 2, \cdots, K$ and Ω_k is a set of sequence numbers for hidden layer nodes associated with the k^{th} pattern class.

(4) *The Softmax classifier* [31]: the output of the pattern layer was used as subsequent input. The probability of classifying signal samples into the k^{th} class is given by:

$$p\left(y = k|q^{(i)}; \theta\right) = \frac{e^{\theta_l^T \cdot q^{(i)}}}{\sum_{l=1}^{K} e^{\theta_l^T \cdot q_k}}, \tag{6}$$

where $q^{(i)} = \left(q_1^{(i)}, q_2^{(i)}, \cdots, q_K^{(i)}\right)$ are the outputs of the pattern layer corresponding to the i^{th} sample input, $\theta = (\theta_1, \theta_2, \cdots, \theta_K)$ and $\theta_l = (v_{1l}, v_{2l}, \cdots, v_{Kl})$ are classifier parameters, $l = 1, 2, \cdots, K$, and v_{kl} is the connection weight from the k^{th} node of the pattern layer to the l^{th} Softmax function node.

RBPNN classification results, based on the principle of maximum probability output, can then be expressed as:

$$y = \arg\max p\left(y = k|q^{(i)}; \theta\right). \tag{7}$$

C. THE CNN-RBPNN MODEL

The CNN-RBPNN established in this paper for multi-channel time-varying signal classification is composed of a 1D CNN, single-channel parallel extraction units, multi-channel integration units (for signal features), and an RBPNN classifier. This structure is shown in Fig. 4, from which the CNN-RBPNN information processing workflow and correspondence between input and output in each unit can be established.

1) FEATURE EXTRACTION AND PARAMETER REPRESENTATION FOR INPUT SIGNALS

(1) *Convolution layer feature extraction and representation*

The 1D convolutional kernel corresponding to the p^{th} feature plane in the i^{th} convolution layer can be represented by $w_i^{(p)} = \left(w_{i1}^{(p)}, w_{i2}^{(p)}, \cdots, w_{il_i}^{(p)}, \right)$. Here, C_i is the i^{th} convolution layer, P_i is the total number of feature planes of C_i. l_i is the width of the convolution kernel, the convolution step size is s_i. The corresponding convolution operation is defined in equation (1). Features extracted in the i^{th} convolution layer can be represented by a vector containing a convolution kernel parameter set, expressed as: $W_{C_i} = [W^1 \in R^{p_1^{(C_i)}@l_i \times 1}, \cdots, W^{P_i} \in R^{p_{P_i}^{(C_i)}@l_{p_i} \times 1}]$.

If the sampling time and density are the same for each signal sample in the training set, the length of each channel time series will also be the same. Features extracted by the 1D CNN will then exhibit the same structure and the eigenvector dimensions of each sample signal will be the same.

(2) *Pooling layer feature extraction and representation*

The pooling layer receives feature information output from the previous convolution layer, which has the same number of feature planes. In the following expressions, the pooling kernel width is represented by r_j, the length of the step is s_j', and the bias is indicated by $a_j = \left(\beta_j, b_j\right)$. After the pooling operation described by equation (3), features extracted from the j^{th} pooling layer S_j can be represented by vectors with a bias parameter set: $a = [a^1 \in R^{p_{p_1}^{S_i}}, \cdots, a^{p_j} \in R^{p_{p_i}^{S_i}}]$.

(3) *The representation of features extracted by a CNN with parallel structure*

Each single-channel input signal corresponded to a 1D CNN, from which features were extracted. The feature vector of unified dimensions was formed. Kernel parameters for convolution operators in each layer of the convolution network were uniformly recorded as $W = [W^1, W^2, \cdots, W^h]$. Similarly, bias parameters for pooling operations were uniformly recorded as $a = [a^1, a^2, \cdots, a^h]$. Feature vectors extracted by this parallel structure can then be expressed as:

$$H = \varphi\left(x(t), W, a\right), \tag{8}$$

where $x(t) \in R^L$ is the single-channel time-varying input signal, L is the length of time-varying signal samples, H is the feature vector extracted by the convolution network, φ is a composite feature extraction operation, $W = [W^1 \in R^{p_1@l_1 \times 1}, W^2 \in R^{p_2@l_2 \times 1}, \cdots, W^{h_1} \in R^{p_{h_1}@l_{h1} \times 1}]$ is a 1D convolution kernel parameter set containing convolution layers, and $a = [a^1 \in R^{s_1}, a^2 \in R^{s_2}, \cdots, a^{h_2} \in R^{s_{h_2}}]$ is the parameter set containing h_2 pooling layers.

(4) *BN normalization processing* [32]

In the next step, $x \in \mathbb{R}^{N \times C \times L}$ represents an input batch containing N signals, where C is the number of channels, L is the signal length, represents the $(tkm)^{th}$ element in the signal set, m is the temporal dimension, k is the input feature channel, and t is the index for the batch signal. The regularization process can then be expressed as:

$$y_{tkm} = \frac{x_{tkm} - u_{tk}}{\sqrt{\sigma_{tk}^2 + \varepsilon}}, \tag{9}$$

where ε is a variable parameter, $u_{tk} = \frac{1}{L}\sum_{m=1}^{L} x_{tkm}$, and $\sigma_{tk}^2 = \frac{1}{L}\sum_{m=1}^{L}(x_{tkm} - u_{tk})^2$.

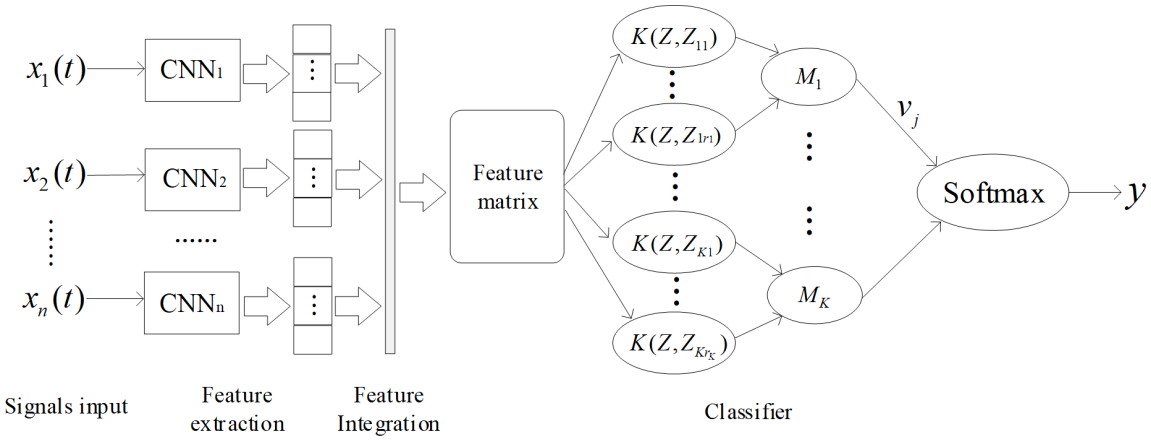

FIGURE 4. The CNN-RBPNN model.

2) INTEGRATION OF MULTI-CHANNEL SIGNAL FEATURES

The single-channel signal eigenvectors extracted by the multi-layer convolution and pooling operations were integrated and arranged according to the channel and time sequence of the input signal. The resulting integrated feature matrix is given by:

$$\begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1L} \\ h_{21} & h_{22} & \cdots & h_{2L} \\ \cdots & \cdots & \cdots & \cdots \\ h_{n1} & h_{n2} & \cdots & h_{nL} \end{bmatrix}. \tag{10}$$

This comprehensive feature matrix for multi-channel signals achieves topological correspondence with single-channel signals in each row. As a result, feature combination relationships and overall structural features can be expressed for multi-channel signal. Spatial and temporal feature fusion can then be achieved.

3) RBPNN CLASSIFICATION USING A COMPREHENSIVE FEATURE MATRIX

A DTW-DCM algorithm, based on the combination of dynamic time warping (DTW) [33], [34] and dynamic C-means clustering (DCM) [35], [36] was established for the purpose of maintaining the diversity of pattern class features. This clustering process divided the samples in each pattern class into several pattern subclasses with more similar characteristics. Typical signal samples in the pattern subclasses were used as the kernel centers for each radial basis, to embed diverse signal class features.

The primary RBPNN information-processing unit is the radial basis probability neuron (RBPN). A comprehensive feature matrix of multi-channel signals was used as input to the RBPN. The DTW algorithm was then used to measure the similarity between the input signal's comprehensive feature matrix and the RBPN kernel's central feature matrix. The output of RBPN is activated by the action function with probability significance. The outputs of RBPN layer were then selectively summed in the pattern layer, and the multi-channel time-varying signals are classified by the Softmax classifier.

The division of the pattern subclasses and the radial basis kernel center are determined as follows.

The training set consisted of K pattern classes $C_k(k = 1, 2, \cdots, K)$, each containing m_k typical sample signals. The feature matrix extracted using the CNN was then recorded as $Z_{kl}(l = 1, 2, \cdots, m_k)$ and the number of RBPN hidden layer nodes is given by $m = m_1 + m_2 + \cdots + m_K$.

The radial basis kernel centers can be determined by the following two techniques. (1) Experts select typical signal samples in each pattern class and extract its characteristic matrix using the CNN. (2) The DTW algorithm is used to measure the distance between signal samples in each pattern class subset. The dynamic C-means clustering algorithm is then used to divide the sample subset and identify the cluster center, which functions as the radial basis kernel center, extracting its feature matrix with the CNN.

The feature matrix for m typical samples selected from the training set was arranged in order as $Z_{11}, \cdots, Z_{1m_1}, Z_{21}, \cdots, Z_{2m_1}, \cdots$ The first subscript of $Z_{kl}(l = 1, 2, \cdots, m_k; k = 1, 2, \cdots, K)$ represents the signal sample category and the second subscript represents the serial number of the j^{th} typical sample feature matrix in the i^{th} pattern class. The Z_{kl} term was used as the kernel center for each node in the radial basis function layer sequentially. Selective summation of radial basis output to the pattern layer nodes only needs to be conducted using the class labels for k in Z_{kl}. The output unit of the RBPNN is a Softmax classifier, which accepts output from the pattern layer as input and is calculated using equations (6) and (7). Multi-channel signal classification results can then be determined using the principle of maximum probability output.

In this study, a CNN was used to extract multi-channel time-varying signal features, which preserve the topological correspondence between the feature matrix and original signal, reducing the impact of signal drift and scaling. The RBPNN classifier can reduce the effects of interleaving between multi-class pattern features, forming an effective

boundary between each pattern class. This facilitates the embedding of prior feature knowledge and reduces both the structural risk and requirements for sample set completeness. As such, the algorithm is suitable for small sample set modeling as it maintains the robustness and generalizability of the model.

III. THE CNN-RBPNN LEARNING ALGORITHM

CNN-RBPNN training was divided into the following stages. (1) A parallel feature extraction algorithm, based on a 1D CNN, was designed for multi-channel signals. The associated matrix representation was based on convolution kernel and pool bias parameters. (2) Typical samples and their comprehensive feature matrix were determined for each pattern class in the sample set using expert selection or a dynamic clustering algorithm. (3) A similarity measurement metric, based on DTW, was implemented for the feature matrix. (4) CNN-RBPNN information unit parameters were normalized and the cross entropy was used to construct an objective function. (5) Optimal CNN-RBPNN model parameters were identified.

A. SIGNAL FEATURE SIMILARITY MEASUREMENTS BASED ON DTW

Dynamic time warping (DTW) is a nonlinear integration algorithm, based on dynamic programming, which combines distance measurements with time warping [33]. This technique is insensitive to the delay and compression of a time series and was used to find an optimal time warping function $M = \varnothing(N)$, which non-linearly maps the temporal axis of time-varying signals to the temporal axis of a reference template. This function satisfies:

$$D = \min \sum_{n=1}^{N} d\{T(n), R[\varnothing(n)]\}, \tag{11}$$

where the test template includes an N-frame feature vector, the reference template includes an M-frame feature vector, and $d\{T(n), R[\varnothing(n)]\}$ is the distance between the n^{th} frame feature vector $T(n)$ in the test template and the m^{th} frame feature vector $R(m)$ in the reference template. The term D is a warping function representing the minimum cumulative distance for each frame of the test and reference templates under optimal time warping. Smaller values indicate higher similarity between two signal distribution features.

In this study, the DTW algorithm was applied to similarity calculations for the original signal and signal feature matrix. This similarity was calculated between synthetic feature matrices for multi-channel signals, as the rows of these matrices represent distribution characteristics for single channel signals. DTW was also used to determine the difference distance d_{ij} for channel distribution features corresponding to the two matrices. The similarity between signal features was then calculated as $r_{ij} = 1/(1 + d_{ij})$ and the similarity between characteristic matrices for multi-channel signals was determined as $r = \frac{1}{n}\sum_{l=1}^{n} r_{ij}^{(l)}$, here, l is the number of rows in the matrix.

B. DYNAMIC C-MEANS CLUSTERING

The signal feature matrix model established in this paper includes a convolution kernel parameters and pooling bias. The original time-varying signal was used to measure the distance between signals, based on the DTW algorithm. On this basis, a dynamic C-means clustering approach [35] was used to divide pattern subsets in the pattern class and determine typical samples exhibiting the signature distribution characteristics. In practical applications, it can be difficult to determine the number of clusters in advance and clustering can be sensitive to initial values. As such, a dynamic C-means (DCM) clustering algorithm is established in this study. Varying cluster numbers were used to evaluate the clustering results corresponding to specific parameter values, by calculating the degrees of coupling and separation between signal samples [36], selecting the best clustering result, and identifying the optimal clustering number. The centers of each optimal cluster were selected as typical signals for the sample set.

C. THE CONSTRUCTION OF A CROSS-ENTROPY OBJECTIVE FUNCTION

The CNN-RBPNN included parameters for the convolution kernel W, pooling layer bias b, radial basis kernel smoothing $\sigma = (\sigma_1, \sigma_2, \cdots, \sigma_m)$, and connection weights $V = (v_1, v_2, \cdots, v_K)$ between the pattern layer and the Softmax classifier, Softmax classifier parameters $\theta = (\theta_1, \theta_2, \cdots, \theta_K)$ and $\theta_l = (\theta_{1l}, \theta_{2l}, \cdots, \theta_{Kl})$, where $(l = 1, 2, \cdots, K)$.

Cross-entropy [37] was used to construct an objective function for the training sample set as follows:

$$\begin{aligned}&\min_{W,b;\theta} J\,(W, b; V; \sigma, \theta)\\ &= \frac{1}{N_{TR}} \sum_{n=1}^{N_{TR}} \sum_{k=1}^{K} \delta\left(y_n^{TR} = k\right) \cdot \log\left(Y_n^{TR}(k)\right)\\ &\quad + \lambda_1 R(W) + \lambda_2 R(b) + \lambda_3 R(\sigma) + \lambda_4 R(V) + \lambda_5 R(\theta),\end{aligned} \tag{12}$$

The last five terms in this expression are regularization terms and $\lambda_i (i = 1, 2, \cdots, 5)$ is the associated coefficient. These can be expressed as:

$$\begin{cases} Y_n^{TR}(k) = softmax\left(X_n^{TR}, \theta\right)\\ = softmax\left(\varphi_2\left(\varphi_1\left(x_n^{TR}; W, b\right), V, \sigma\right), \theta\right)\\ R(W) = \sum_{l=1}^{P} \|W^l\|_F^2\\ R(b) = \sum_{l=1}^{P} \|b^l\|_F^2\\ R(\sigma) = \sum_{l=1}^{m} \|\sigma^l\|_F^2\\ R(V) = \sum_{l=1}^{P} \|V^l\|_F^2\\ R(\theta) = \sum_{k=1}^{K} \|\theta_k\|_F^2. \end{cases} \tag{13}$$

D. THE CNN-RBPNN TRAINING ALGORITHM

The implementation process for the comprehensive CNN-RBPNN training algorithm can be described as follows.

Step 1. A Gaussian radial basis function was selected and the kernel centers were determined using the algorithm presented in Section 3.2.

Step 2. The squared sum of the mean square deviations was determined for the pattern subclass samples and the cluster centers, and an initial value of σ_{m_k} defined as:

$$\sigma_{m_k} = \left(\frac{1}{K_{m_k}} \sum_{i \in \Omega_{m_k}} \left\| x_{im_k}(t) - z_{m_k}(t) \right\|_{DTW} \right)^{\frac{1}{2}}, \quad (14)$$

where K_{m_k} is the number of signal samples in the m_k^{th} cluster of the k^{th} pattern class, $z_{m_k}(t)$ represents cluster centers, Ω_{m_k} is the sequence number set for samples contained in the pattern subclass, and $\| \cdot \|_{DTW}$ is the distance norm based on the DTW algorithm.

Step 3. According to the cross-entropy objective function, gradient descent algorithm is adopted. The terms V, σ and θ in the equation (15) belong to a classifier unit. A previous hierarchical connection structure was adopted, which is consistent with derivatives in the feedforward network. The iterative update formula for CNN-RBPNN model parameters is given by:

$$\begin{cases} W^{(k)} = W^{(k-1)} - \alpha \dfrac{\partial J(W, b, V, \sigma, \theta)}{\partial W}|_{W=W^{(k-1)}} \\ b^{(k)} = b^{(k-1)} - \alpha \dfrac{\partial J(W, b, V, \sigma, \theta)}{\partial b}|_{b=b^{(k-1)}} \\ \sigma^{(k)} = \sigma^{(k-1)} - \alpha \dfrac{\partial J(W, b, V, \sigma, \theta)}{\partial \sigma}|_{\sigma=\sigma^{(k-1)}} \\ V^{(k)} = V^{(k-1)} - \alpha \dfrac{\partial J(W, b, V, \sigma, \theta)}{\partial V}|_{V=V^{(k-1)}} \\ \theta^{(k)} = \theta^{(k-1)} - \alpha \dfrac{\partial J(W, b, V, \sigma, \theta)}{\partial \theta}|_{\theta=\theta^{(k-1)}}. \end{cases} \quad (15)$$

where α is learning efficiency. $W^{(k)}$ is the updated value of convolutional layers parameter matrix of CNN in the k^{th} iteration. $b^{(k)}$, $V^{(k)}$, $\sigma^{(k)}$, $\theta^{(k)}$ are the update values of respectively bias vector in the pool layer, the connection weight vector from RBFNN pattern layer to Softmax classifier, the RBFN kernel center smoothing parameter vector and Softmax classifier parameter vector in the k^{th} iteration.

Step 4. The error back propagation (BP) algorithm was used to fine-tune RBPPNN parameters in the training set, including $(X_1(t), y^{(1)}), (X_2(t), y^{(2)}), \cdots, (X_N(t), y^{(N)})$.

IV. APPLICATION TO THE CLASSIFICATION OF 12-LEAD ECG SIGNALS

A. EXPERIMENTAL DATA

ECG signals represent changes in the electrical potential of the human heart. These multi-peak data are periodic, non-stationary, and include significant background noise. Various cardiovascular diseases exhibit different signal distribution characteristics. The data used in this experiment were 12-lead ECG signal samples from the Chinese Cardiovascular Disease Database (CCDD) [38]. The sampling frequency was 500 Hz and each recording time was more than 10 seconds, including nine heartbeats. These samples included labeled heartbeat divisions with a diagnosis marked by medical experts. Seven disease types were included and the training set was composed of 4332 samples. A time series of 2520 sampling points in 12 channels was formed using Nyquist ECG sampling theorem [39]. This sample set was relatively small, due to the complexity of ECG signal characteristics. The disease names, sample distributions, and category codes are shown in Table 1.

TABLE 1. The experimental sample data distribution.

Dictionary Code	Disease Type	Sample Number	Category Code
$0X010101$	Normal	960	0
$0X020201$	Atrial Fibrillation	807	1
$0X020301$	Atrial Premature Beat	619	2
$0X030201$	Episodic Premature Beat	565	3
$0X050202$	Frequent Premature Beat	582	4
$0X060305$	Atrial Tachycardia	389	5
$0X080205$	Atrial Fibrillation with Rapid Ventricular Rate	410	6
Total		4332	

The data were normalized, to compensate for differing dimensions and large variability in signal magnitude, using the following formula:

$$x'(t) = x(t) - \min x(t) / \max x(t) - \min x(t), \quad (16)$$

where $\min x(t)$ and $\max x(t)$ represent the minimum and maximum values of each ECG lead on the measurement interval, respectively.

B. THE CNN-RBPNN MODEL FOR ECG SIGNAL CLASSIFICATION

The CNN-RBPNN model structure and parameters were established for 12-lead ECG signal classification as follows. The 1D CNN consisted of two convolution layers, two pooling layers, and one BN layer. The number of feature planes in the convolution layers were set to 8 and 4, respectively. The convolution kernel width and step size were set to 7 and 3, respectively. The pooling layers included a maximum pooling operation with a step size of 4. After BN layer processing, a feature vector was produced with a dimension of 568. The feature fusion layer integrated signal vectors in each channel to form a multi-channel comprehensive feature matrix with dimensions of 12×568.

The DTW-DCM clustering algorithm was used to cluster each sample subset which consisting of 7 disease types, and a total of 32 pattern subclasses are divided. The centers of each pattern subclass were used as typical characteristic signal samples. The CNN module included 12 input nodes, which

TABLE 2. Experimental results for the proposed model.

Disease Type	Accuracy (%)	Recall Rate (%)	Precision (%)	F1-Score (%)
Normal	100	98	99	99
Atrial Fibrillation	87.09	86	86	87
Atrial Premature Beat	86.03	87	87	88
Episodic Premature Beat	84.25	84	83	85
Frequent Premature Beat	70.27	71	72	75
Atrial Tachycardia	76.13	75	75	77
Atrial Fibrillation with Rapid Ventricular Rate	82.31	82	82	83

realized the input of 12-lead ECG signal and the output was the feature matrix with dimensions of 12×568. The RBPNN module included 12 input layer nodes, each of which was a single-channel signal feature vector. The number of nodes in the RBPN and pattern layers were 132 and 7, respectively. The Softmax classifier included 7 input and 7 output nodes.

C. EXPERIMENTAL RESULTS AND ANALYSIS

1) EXPERIMENTAL CONFIGURATION

Training set samples were randomly divided into two groups according to disease proportions. One group of 2888 samples constituted the training set and the other group of 1444 samples served as the test set. The learning algorithm described in Section 3 was then used to determine CNN-RBPNN parameters. The experiment was performed using an NVIDIA TITAN X GPU with a core frequency of 1418 *MHz*. Training error accuracy was set to 0.05, the maximum number of iterations was 10,000, and the learning efficiency was 0.45. Once training was complete, test set samples were classified into specific categories. Identification results for 7 disease types and various evaluation indicators are shown in Table 2.

As shown in table 2, it has achieved good classification results in normal, atrial fibrillation, atrial premature beats, and episodic premature beats. The performance evaluation indicators of frequent premature beats and atrial tachycardia are relatively low. This is due to the small sample number of these two diseases and the similar distribution characteristics of these ECG signals. Although the sample number of atrial fibrillation with rapid ventricular rate is also small, the distribution characteristics are different from other diseases, and good results have also been obtained.

Comprehensive analysis shows that the proposed technique achieved good result in the case of small sample set modeling. Due to a 1D parallel CNN feature extraction technique was used to generate a comprehensive feature matrix, which could simultaneously represent the distribution characteristics of single-lead signals and the structural characteristics of multi-lead signals. Typical signal features from each pattern class were embedded in the model, structural and data constraints were implemented, the diversity of pattern features was maintained, and the requirements for sample set completeness were reduced. As a result, the proposed algorithm has better stability and robustness for small and imbalanced sample sets.

TABLE 3. ECG signal classification results for the comparative models.

Model	Accuracy (%)	Recall Rate (%)	F1-Score (%)
CNN	75.24	74.93	75.1
LSTM+RT	76.79	75.82	75.92
GRU-PRNN	75.98	75.68	74.79
This Paper	86.25	85.37	86.01

2) COMPARATIVE ANALYSIS

A series of comparative experiments were conducted to assess conventional models in the classification of multi-channel process signals. This included a multi-channel convolutional network (MC-DCNN) [40], an LSTM + RF network combining LSTM and random forest [41], and a GRU-RNN network [3]. These three deep neural network models, combined with the methodology proposed in this study, comprised the four algorithms used for comparative disease classification. The same training and test sample sets were used in each case.

The MC-DCNN model architecture can be described as I-C1(Size)-S1-C2(Size)-S2-H-O, where 'Size' denotes the kernel size, C1 and C2 represent the numbers of filters, and S1 and S2 denote subsampling factors. The I, H, and O terms respectively indicate the number of input layers, units in the hidden layer, and MLP output layers. Comparative analysis determined an optimal architecture of 12-8(5)-2-4(5)-2-440-7. A series model of two LSTM networks was constructed with 3 hidden layers in each LSTM. A random forest classifier was also established with 160 trees. The GRU-RNN hidden layer consisted of 5 GRU stacks and included a Softmax classifier. An experimental 4-fold crossover scheme was implemented, in which the sample set was randomly divided into 4 groups according to the proportion of disease types, with 1083 samples in each group. Three of these groups were combined to form the training set and the remaining group was used as the test set in each of the 4 experiments. The average of each evaluation index across four experiments was used as a comparison metric, as shown in Table 3.

It is evident in this table that the proposed model achieved the best results among the algorithms tested. This is because the small sample set modeling used in the experiment. The complexity and feature diversity of 12-lead ECG signals makes it difficult to acquire a complete sample set. However, in the CNN feature extraction step, the included comprehensive feature matrix maintains distribution characteristics of single-lead signal and structural information for the 12-lead signals. In addition, the RBPNN classifier embeds typical signal feature knowledge for pattern classes. Complex classification boundaries are formed in the process of combining pattern subclasses into classes, which reduces the influence of incomplete sample sets. These properties improves the

TABLE 4. The 7-class confusion matrix for the proposed technique.

Type	0	1	2	3	4	5	6
0	1.0	0	0	0	0	0	0
1	0.023	0.886	0.042	0.024	0.021	0.012	0.011
2	0.032	0.028	0.864	0.034	0.014	0.013	0.013
3	0.034	0.035	0.054	0.835	0.035	0.029	0.031
4	0.024	0.031	0.008	0.010	0.703	0.199	0.031
5	0.017	0.015	0.012	0.004	0.168	0.762	0.025
6	0.023	0.061	0.038	0.025	0.026	0.018	0.824

TABLE 5. The 7-class confusion matrix for DC-CNN.

Type	0	1	2	3	4	5	6
0	0.962	0.012	0.008	0.013	0.001	0.002	0.003
1	0.020	0.823	0.072	0.024	0.024	0.013	0.024
2	0.048	0.035	0.768	0.072	0.026	0.017	0.035
3	0.037	0.042	0.074	0.743	0.048	0.023	0.034
4	0.015	0.033	0.059	0.023	0.592	0.228	0.051
5	0.018	0.033	0.092	0.057	0.213	0.505	0.083
6	0.034	0.043	0.058	0.047	0.035	0.046	0.738

classification accuracy of the proposed technique. In contrast, the three deep learning algorithms are all 'end-to-end' models that learn from a sample set and lack the inspiration or constraints of knowledge embedding. In the case of incomplete small sample sets, the corresponding model structure and parameter selection exhibit high degrees of freedom, which are prone to overfitting, resulting in poor generalizability.

The DC-CNN and the proposed technique in this study were used to calculate a confusion matrix for seven classification results in the test set [42], the results of which are shown in Tables 4 and 5. It is evident from these data that the algorithm proposed in this study has significantly improved the differentiation of samples with similar distribution characteristics. Especially for frequent premature beat and atrial tachycardia, the correct recognition rate increased by 14.12% and 25.78% respectively. This is because the proposed model is superior to DC-CNN in terms of feature knowledge embedding, feature extraction, an ability to characterize combinational relationships, memory capabilities for multi-modal signal features and the generating complex classification interfaces. As such, the methodology developed in this paper offers superior accuracy and robustness for small and imbalanced sample sets.

V. CONCLUSION

In this study, a multi-channel time-varying signal classification model combining CNN and RBPNN information processing mechanisms was developed for small sample set modeling and feature knowledge embedding. The proposed method can extract and represent the comprehensive features of multi-channel signals, has the advantages of embedding the diverse signal features knowledge of each pattern class in model, and combined multiple pattern subclass to form complex boundaries of pattern classes. It is suitable in mechanism to the case of small sample set modeling, the experiment also achieved good results. In the proposed technique, the multi-channel signal feature extraction and representation learning method based on CNN was mature. However, the feature identification and classification ability of the model mainly depends on the pattern class typical feature signals set generated by each pattern class through clustering algorithm. Therefore, for different signal types and distribution characteristics, it is an important work of constructing an appropriate clustering algorithm and selecting the typical feature signals in the pattern classes, for purpose to maintain the completeness of the signal feature set, which will be further study in the next stage.

REFERENCES

[1] H.-M. Shim and S. Lee, "Multi-channel electromyography pattern classification using deep belief networks for enhanced user experience," *J. Central South Univ.*, vol. 22, no. 5, pp. 1801–1808, May 2015.
[2] V. Chandran, "Time-varying bispectral analysis of visually evoked multichannel EEG," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 1, p. 140, Dec. 2012.
[3] D. Rajan and J. J. Thiagarajan, "A generative modeling approach to limited channel ECG classification," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 2571–2574.
[4] J. Gao, Y. L. Murphey, and H. Zhu, "Multivariate time series prediction of lane changing behavior using deep neural network," *Int. J. Speech Technol.*, vol. 48, no. 10, pp. 3523–3537, Oct. 2018.
[5] M. Lee, "Methods and apparatus for identifying audio/video content using temporal signal characteristics," U.S. Patent 7,650,616, Jan. 19, 2010.
[6] A. Gacek and W. Pedrycz, *ECG Signal Processing, Classification and Interpretation: A Comprehensive Framework of Computational Intelligence*. Springer, 2011.
[7] L.-F. Chen, H.-Y.-M. Liao, M.-T. Ko, J.-C. Lin, and G.-J. Yu, "A new LDA-based face recognition system which can solve the small sample size problem," *Pattern Recognit.*, vol. 33, no. 10, pp. 1713–1726, Oct. 2000.
[8] M. Alibeigi, S. Hashemi, and A. Hamzeh, "DBFS: An effective density based feature selection scheme for small sample size and high dimensional imbalanced data sets," *Data Knowl. Eng.*, vols. 81–82, pp. 67–103, Nov. 2012.
[9] D. Ramyachitra and P. Manikandan, "Imbalanced dataset classification and solutions: A review," *Int. J. Comput. Bus. Res.*, vol. 5, no. 4, pp. 1–29, 2014.
[10] S. Bououden, M. Chadli, and H. R. Karimi, "Control of uncertain highly nonlinear biological process based on Takagi–Sugeno fuzzy models," *Signal Process.*, vol. 108, pp. 195–205, Mar. 2015.
[11] D. Li, J. Zhang, Q. Zhang, and X. Wei, "Classification of ECG signals based on 1D convolution neural network," in *Proc. IEEE 19th Int. Conf. E-Health Netw., Appl. Services (Healthcom)*, Oct. 2017, pp. 1–6.
[12] Z. Xiong, M. Stiles, and J. Zhao, "Robust ECG signal classification for the detection of atrial fibrillation using novel neural networks," in *Proc. Comput. Cardiol. Conf. (CinC)*, Sep. 2017, pp. 1–4.
[13] S. Lange and M. Riedmiller, "Deep auto-encoder neural networks in reinforcement learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2010, pp. 1–8.
[14] W. Fei, X. Ye, Z. Sun, Y. Huang, X. Zhang, and S. Shang, "Research on speech emotion recognition based on deep auto-encoder," in *Proc. IEEE Int. Conf. Cyber Technol. Automat., Control, Intell. Syst. (CYBER)*, Jun. 2016, pp. 308–312.
[15] J. Qi, P. Shi, L. Hu, T. Zhang, and S. Xie, "ECG characteristic wave detection based on deep recursive long short-term memory," *J. Med. Imag. Health Informat.*, vol. 9, no. 9, pp. 1920–1924, Dec. 2019.
[16] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 708–712.

[17] S. Chauhan and L. Vig, "Anomaly detection in ECG time signals via deep long short-term memory networks," in *Proc. IEEE Int. Conf. Data Sci. Adv. Analytics (DSAA)*, Oct. 2015, pp. 1–7.

[18] W. J. Fitzgerald, "Markov chain Monte Carlo methods with applications to signal processing," *Signal Process.*, vol. 81, no. 1, pp. 3–18, Jan. 2001.

[19] S. Derrode and W. Pieczynski, "Signal and image segmentation using pairwise Markov chains," *IEEE Trans. Signal Process.*, vol. 52, no. 9, pp. 2477–2489, Sep. 2004.

[20] M. Wasikowski and X.-W. Chen, "Combating the small sample class imbalance problem using feature selection," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1388–1400, Oct. 2010.

[21] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 98–113, Jun. 1997.

[22] D.-S. Huang, "Radial basis probabilistic neural networks: Model and application," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 13, no. 7, pp. 1083–1101, Nov. 1999.

[23] D.-S. Huang, "Application of generalized radial basis function networks to recognition of radar targets," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 13, no. 6, pp. 945–962, Sep. 1999.

[24] S. Shadmand and B. Mashoufi, "A new personalized ECG signal classification algorithm using block-based neural network and particle swarm optimization," *Biomed. Signal Process. Control*, vol. 25, pp. 12–23, Mar. 2016.

[25] Ö. Yildirim, "A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification," *Comput. Biol. Med.*, vol. 96, pp. 189–202, May 2018.

[26] C. Saritha, V. Sukanya, and Y. N. Murthy, "ECG signal analysis using wavelet transforms," *Bulg. J. Phys.*, vol. 35, no. 1, pp. 68–77, Feb. 2008.

[27] A. M. Saxe, P. W. Koh, Z. Chen, M. Bhand, B. Suresh, and A. Y. Ng, "On random weights and unsupervised feature learning," in *Proc. ICML*, 2011, vol. 2, no. 3, p. 6.

[28] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *Proc. Int. Conf. Artif. Neural Netw.* Berlin, Germany: Springer, 2010, pp. 92–101.

[29] D.-S. Huang and J.-X. Du, "A constructive hybrid structure optimization methodology for radial basis probabilistic neural networks," *IEEE Trans. Neural Netw.*, vol. 19, no. 12, pp. 2099–2115, Dec. 2008.

[30] K. Liu, S. Xu, and N. Feng, "A radial basis probabilistic process neural network model and corresponding classification algorithm," *Int. J. Speech Technol.*, vol. 49, no. 6, pp. 2256–2265, Jun. 2019.

[31] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[32] M. Yedroudj, F. Comby, and M. Chaumont, "Yedroudj-net: An efficient CNN for spatial steganalysis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2092–2096.

[33] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2001, pp. 1–11.

[34] T. Giorgino, "Computing and visualizing dynamic time warping alignments in R: The dtw package," *J. Stat. Softw.*, vol. 31, no. 7, pp. 1–24, 2009.

[35] H. Fathabadi, "Power distribution network reconfiguration for power loss minimization using novel dynamic fuzzy C-means (dFCM) clustering based ANN approach," *Int. J. Electr. Power Energy Syst.*, vol. 78, pp. 96–107, Jun. 2016.

[36] M. Ghaffari and N. Ghadiri, "Ambiguity-driven fuzzy C-means clustering: How to detect uncertain clustered records," *Appl. Intell.*, vol. 45, no. 2, pp. 293–304, 2016.

[37] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 19–67, Feb. 2005.

[38] J.-W. Zhang, X. Liu, and J. Dong, "CCDD: An enhanced standard ecg database with its management and annotation tools," *Int. J. Artif. Intell. Tools*, vol. 21, no. 5, Oct. 2012, Art. no. 1240020.

[39] M. Mangia, J. Haboba, R. Rovatti, and G. Setti, "Rakeness-based approach to compressed sensing of ECGs," in *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Nov. 2011, pp. 424–427.

[40] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, "Time series classification using multi-channels deep convolutional neural networks," in *Proc. Int. Conf. Web-Age Inf. Manage.* Cham, Switzerland: Springer, 2014, pp. 298–310.

[41] L. D. Sharma and R. K. Sunkaria, "Inferior myocardial infarction detection using stationary wavelet transform and machine learning approach," *Signal, Image Video Process.*, vol. 12, no. 2, pp. 199–206, Feb. 2018.

[42] S. Visa, B. Ramsay, A. L. Ralescu, and E. Van Der Knaap, "Confusion matrix-based feature selection," *MAICS*, vol. 710, pp. 120–127, Apr. 2011.

LU WU was born in Shandong, China, in 1982. He received the B.Sc. and M.Sc. degrees in computer science and technology from Northwestern Polytechnic University, in 2003 and 2006, respectively. He is currently pursuing the Ph.D. degree with the College of Computer Science and Engineering, Shandong University of Science and Technology. His current research interests include machine learning, neural networks, and applications of artificial intelligence in medicine.

YINGLONG WANG received the B.Sc. degree in electronic technology and the M.S. degree in industrial automation from the Shandong University of Technology, in 1987 and 1990, respectively, and the Ph.D. degree in communication and information systems from Shandong University, in 2005. His current research interests include medical artificial intelligence, high-performance computing, and cloud computing.

SHAOHUA XU received the B.Sc. degree in mathematics from the Northeastern University of Petroleum, in 1983, the M.Sc. degree in applied mathematics from the Harbin Institute of Technology, in 1986, and the Ph.D. degree in computer software and theoretical engineering from Beihang University, in 2004. His current research interests include large data analysis, intelligent information processing technology, and 3-D visualization data modeling technology.

KUN LIU was born in Shandong, China, in 1990. He received the B.Sc. and M.Sc. degrees in mathematics from Qufu Normal University, in 2013 and 2017, respectively. He is currently pursuing the Ph.D. degree with the College of Computer Science and Engineering, Shandong University of Science and Technology. His current research interests include machine learning, fuzzy set, and neural networks.

XUEGUI LI received the B.Sc. degree in electronic information science and technology, the M.Sc. degree in computer application technology, and the Ph.D. degree in computer technology from the Northeastern University of Petroleum, in 2005, 2008, and 2017, respectively. His current research interests include artificial intelligence and large data analysis.