

Research Article

A SA-ANN-Based Modeling Method for Human Cognition Mechanism and the PSACO Cognition Algorithm

Shuting Chen ¹ and Dapeng Tan ²

¹Hangzhou Medical College, Hangzhou 310053, China

²Zhejiang University of Technology, Hangzhou 310032, China

Correspondence should be addressed to Dapeng Tan; 78913258@qq.com

Received 12 July 2017; Revised 2 November 2017; Accepted 26 November 2017; Published 4 January 2018

Academic Editor: Michele Scarpiniti

Copyright © 2018 Shuting Chen and Dapeng Tan. 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.

Artificial neural networks (ANNs) are the important approaches for researching human cognition process. However, current ANNs-based cognition methods cannot address the problems of complex information understanding and fault-tolerant learning. Here we present a modeling method for cognition mechanism based on a simulated annealing-artificial neural network (SA-ANN). Firstly, the relationship between SA processing procedure and cognition knowledge evolution is analyzed, and a SA-ANN-based inference model is set up. Then, based on the inference model, a Powell SA with combinatorial optimization (PSACO) algorithm is proposed to improve the clustering efficiency and recognition accuracy for the cognition process. Finally, three groups of numerical instances for knowledge clustering are provided, and three comparative experiments are performed by self-developed cognition software. The simulated results show that the proposed method can increase the convergence rate by more than 20%, compared with the back-propagation (BP), SA, and restricted Boltzmann machines based extreme learning machine (RBM-ELM) algorithms. The comparative cognition experiments prove that the method can obtain better performances of information understanding and fault-tolerant learning, and the cognition accuracies for original sample, damaged sample, and transformed sample can reach 99.6%, 99.2%, and 97.1%, respectively.

1. Introduction

Human cognition is an information processing procedure of acquiring and understanding the knowledge by thoughts, experiences, and senses. Newell and Simon find that there are some similarities in the functional structures and processing procedures between human beings and computers. They are special systems comprising systematic modules of input, output, processing, and storage, which process the input information and perform the tasks of feature extraction and pattern recognition for different objects [1]. Accordingly, the essence of information processing theory regards the computer system as a cognition model and reveals the inherent relationships between cognition process and digital information processing.

Related research works [2–5] suggested that the human and the computer are the information processing systems composed of manipulating symbols that are the general

designation of the concept patterns such as languages, tags, or marks. The function of the symbols is to represent, sign, or indicate the entities of the external world. Traditional models of human cognition are based on the qualitative analysis or empirical rules, yet providing accurate digital representations of the process of cognition is difficult. Therefore, it is necessary to introduce the advanced information technologies into human recognition area and propose a cognition mechanism modeling method.

Artificial neural networks (ANNs), a form of connectionism, are computing systems inspired by the biological neural networks that constitute animal or human brains. They can perform abstract simulation for entity neuron networks, with the features of fault-tolerance, efficiency, and self-adaptability, and have better cognitive functions, such as learning and memory. Since the information processing mechanism of ANNs is similar to the cognition process of human beings, ANNs can provide suitable approaches

for the numerical simulation of human cognition [6–9]. Moreover, ANNs can be used not only to verify the relevant experimental findings but to research perception, associative learning, memory, and other cognitive issues. Sturt et al. developed a hybrid structure recursive neural network (HS-RNN) to study human language processing, where the semantic structure was described by grammar symbols and classified by the HS-RNN. The experimental results showed that this method can be used to effectively process nonrestrictive text [10]. Vokey and Higham conducted a simulation study on the process of learning artificial grammar by using an ANN and contested the conclusions of Tunney et al. based on the numerical results of two simulated experiments [11–13]. Adopting an ANN to classify the characteristics of lower back pain (LBP), Hallner and Hasenbring introduced psychological risk factors as ANN input neurons and used the error back-propagation (BP) algorithm to train the order values of the network. The results showed that the accuracy rate of LBP classification could reach more than 83% [14]. Kaplan et al. proposed a hybrid ANN model that adopted the Hamming network to generate question assumptions, used the Hopfield network to implement associative memory and calculate optimization, and completed a numerical study on Wisconsin card sorting test (WCST) performance scores for palinphasia and distractibility [15]. Hélie et al. used unsupervised Bayesian networks to simulate the environment propensity for knowledge learning. The experimental results showed that it was difficult to study by using environmental propensity for stiff competition learner, and periodic associative memory was more important in stiff competition [16]. Levine presented a neural network modeling method for affective computing and systematically discussed the neural mechanisms of emotion and cognition [17]. Chartier et al. introduced nonlinear dynamical systems into research on ANNs and proposed so-called nonlinear dynamical artificial neural networks (NDANNs), which can compensate for the deficiencies of traditional ANNs with respect to nonlinear reasoning. The numerical results showed that NDANNs can effectively reveal the internal regulators of multivalued, located, limited region objects and nonperiodic or chaos behaviors [18]. Tsagkaris et al. proposed a neural network learning program for cognitive radio systems to explore the prediction and intelligent management of radio communication systems [19]. Schmidt-Atzert et al. compared predictions from two different statistical methods, ANNs and multiple regression analysis, and concluded that ANNs should be used only as a selection algorithm if their validity in different cohorts has been confirmed [20]. Alexandridis and Maru used an empirical case study to demonstrate the process of extracting linguistic concepts from data and training a Hopfield ANN for semantic network classification [21]. Grossi et al. proposed a new data mining tool based on an ANN acting on the small data set formed by questionnaire items applied serially over time. This technology was promising in the remote monitoring of patients' psychological conditions in different settings, providing the possibility of implementing personalized psychological interventions [22]. Neftci et al. demonstrated an event-driven random BP (eRBP) rule that used an error-modulated synaptic plasticity for learning

deep representations in neuromorphic computing hardware [23]. Oyedotun and Khashman proposed a neural network algorithm based on modifying the emotional neural network (EmNN) model to unify the prototype and adaptive-learning theories, which could be applied to two real-life challenging tasks: static hand-gesture recognition and face recognition [24].

From the above references, we can conclude that ANNs and their varieties of algorithms have been widely applied in the area of human cognition mechanism, with adequate results. However, such research works commonly focus on solving specific matters related to psychological cognition and do not involve the systematic studies of the cognition process and inherent regularities. Moreover, the cognition accuracies of above methods rely on large quantity of input data samples and are apt to be influenced by the data samples with high nonlinearities.

Although the BP-ANNs, NDANNs, HS-RNN, and Hopfield ANN can support nonlinear data mapping, the ability of information understanding, that is, complex knowledge inference, needs to be improved. Moreover, owing to the randomness, transformation, or damage of input data, the fault-tolerant learning is hard to be performed by the above methods, and the clustering efficiency and cognition accuracy need to be improved.

The simulated annealing (SA) method [25–27] can accept the fault-tolerant solutions by the Boltzmann distribution law and support jump-style searching for knowledge learning process, which is close to knowledge evolution rules of human cognition. Therefore, this paper introduces SA method into human cognition area and proposes a SA-ANN-based modeling method for human cognition mechanism. Concerning the above research target, the corresponding research works are described as follows. (1) On the basis of information processing theory, human cognition and its fundamental characteristics are analyzed, and a SA-ANN-based inference model is set up. (2) Based on the inference model, a Powell SA with combinatorial optimization (PSACO) algorithm is proposed to define and simulate the key factors of human cognition, such as feature extraction, knowledge clustering, pattern template reconstruction, and pattern recognition. (3) The numerical instances with nonlinear data samples are provided, and a demo software system is developed to verify the conclusions from the simulation.

As its main scientific contributions, this paper introduces SA-ANN method into human cognition area and presents a mathematical pathway to acquire accurate digital representations of cognition process. This research not only offers direct suggestions for research in complex system modeling, inference analysis, and the optimization of cognitive psychology, but also provides a universal reference for research in the areas of dynamic pattern recognition, nonlinear signal processing, and monitoring of objects with uncertainties.

This paper is organized as follows. In Section 2, the relationships between ANNs and human cognition are presented, a SA-ANN-based modeling method for human cognition is proposed, and the relevant theory on dynamical pattern recognition is discussed. In Section 3, a BP-ANN instance for human cognition is presented, and the PSACO algorithm

is proposed. In Section 4, the numerical instances are performed, and the clustering results for cognition process are discussed. In Section 5, cognition software is developed, and the experimental results are discussed. In Section 6, the conclusions are presented.

2. A SA-ANN-Based Method for Modeling Human Cognition

2.1. SA-ANN-Based Cognition Mechanism Model. As mentioned above, ANNs have the capabilities of improved learning, associative memory, and parallel processing, and they are hence widely applied in the areas of automatic control [28–30], signal processing [31–33], and pattern recognition [34–36]. The neurons of ANNs are connected by weights and thresholds, and adaptive computing is used to preprocess external input information through the ANNs' nonlinear mapping ability until the output meets the expected values. Moreover, an ANN can use one or several groups of standard sample data to obtain the corresponding standard output data and analyze the relationships between the input and output. These relationships can be regarded as the criteria for the inputted real-time data, and the above processing procedure is a typical machine learning process. ANNs store the knowledge that is learned from data samples in the form of weights and thresholds to construct a memory database in order to facilitate judgment and recognition [37]. Commonly, the input of an ANN includes the extracted features or signature sequence, and the output is the feature data of target objects.

Considering the above characteristics of ANNs, we can identify the following similarities between the information processing of an ANN and the process of human cognition: (1) The input information for learning or clustering is limited, and the external information can be accepted only by the extraction of certain information to obtain features or a signature sequence. (2) The memory rooted in continuous learning determines the judgment and identification of the input information, so learning is the key component of information processing. (3) The executive approach of learning and recognition is associative thinking or nonlinear identification, which includes self-adaptive performance. (4) Learning and recognition can work concurrently, so they have real-time characteristics. (5) Extracting feature data from information preprocessing is not necessarily effective, and the final recognition results depend only on partial feature data. Therefore, the information processing system is sensitive to the so-called effective features. Accordingly, the ability to obtain the effective features is an important precondition that influences the effectiveness of the information processing system.

SA is a heuristic random search method and has special search strategies, compared with traditional random search algorithms. It not only involves the adequate random factors, but also integrates with the natural mechanism of physical annealing process. During the course of SA iteration, the better solutions, that is, the searched data elements with smaller errors, can be accepted by the target function, and the worse solutions with higher errors are also accepted to jump out the trap of local optimal solution, according to the Boltzmann distribution law [26]. Moreover, SA can provide

jump-style searching for knowledge learning process, which is close to knowledge evolution rules of human cognition.

For the cognition process, finding an approximate global optimum is more important than finding a precise local optimum in a fixed amount of time [38], so the SA may be preferable to alternatives such as ANNs with gradient descent algorithms. Since the search strategies of SA are apt to avoid falling into local optimal solutions, if it is used to train the ANNs, it might make the search algorithms jump out of local optimum and increase the search stability for global optimum.

Referring to the above contents, we can find that SA has the capability of fault-tolerant processing and can obtain global optimal solution. Therefore, this paper introduces SA into human cognition area, in combination with the information processing theory and ANNs, and set up an inference model for cognition mechanism, as shown in Figure 1. The model is divided into three logic layers: the environmental layer, the sensory layer, and the feature extraction-recognition layer. (1) The environmental layer refers to the entities of the objective environment that need to be determined or identified. (2) The sensory layer contains a sensory perception module (SPM) and a sensory effect module (SEM), which acquires information from the environmental layer. Human beings obtain external information through the approaches of vision, hearing, touch, and so forth. Computer systems' external information collectors include input devices, sensors, or data acquisition subsystems. (3) The feature extraction-recognition layer comprises a characteristic extraction module (CEM), a knowledge study and recognition module (KSRM), and an output processing module (OPM). They are the core components of the inference model and perform the tasks of information extraction, knowledge learning, and pattern recognition based on SA-ANN.

Referring to the model in Figure 1, we can conduct the following data dependencies of the functional modules. Firstly, SPM obtains information from the environmental layer, transfers it to the CEM for preliminary processing, and extracts the feature vectors of the signature sequence as the input data of the KSRM. Then, the KSRM receives the feature vectors as the SA schemes and executes nonlinear distribution computation to generate SA results. Subsequently, OPM performs target state identification and obtains state data for the expected state space. Finally, the final processed results are shown or described by the SEM.

The CEM is a massive set of algorithms, each of which corresponds to an information acquisition algorithm. When a specific object is processed, one or several algorithms might be selected to extract the feature data. Apparently, the initial selected algorithms are not necessarily optimal or valid. For example, something may not be easily distinguished by vision but may be relatively easily identified by audio. With respect to this matter, the selected algorithms are required to be optimized or replaced, which means that methods must be adjusted or converted to address the problem until the expected results are matched. As a core component of the information processing system, the KSRM needs to maintain the parallel processing capabilities of knowledge learning and pattern recognition.

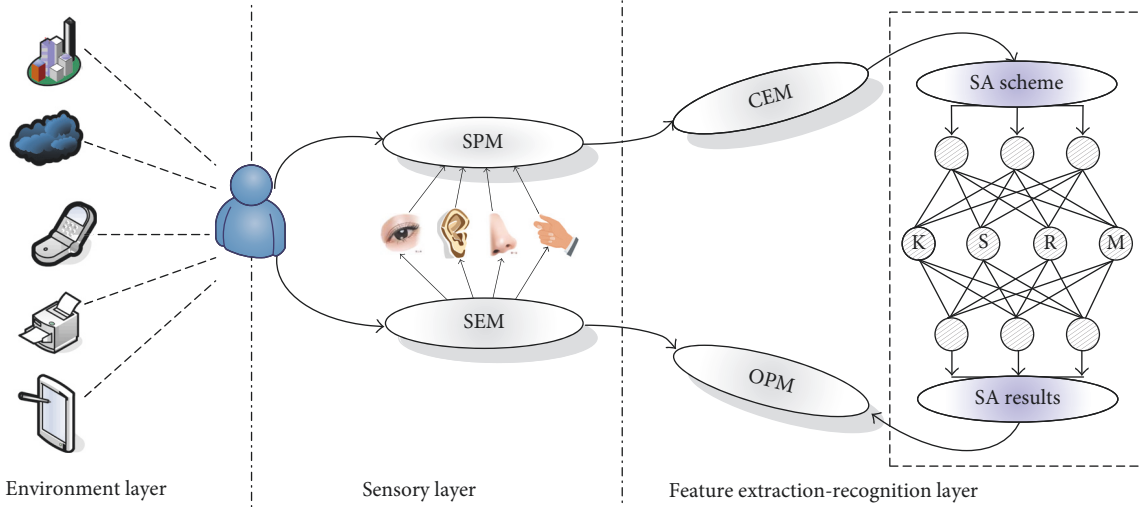


FIGURE 1: Abridged general view of SA-ANN based cognition mechanism model.

2.2. Pattern Recognition in SA-ANN Cognition Framework.

In the hypothesis of SA-ANN-based cognition mechanism model, human cognition can be regarded as a data-driven dynamical pattern recognition issue, essentially subjected to external input information, in which the process of cognition is the dynamical mapping from feature information to the system state patterns [39–41]. As a commonality, pattern recognition and human cognition both identify the characteristics of the target object and learn based on standard target states or a template for recognition. If the effectiveness satisfies the expected design requirements, the recognition system would achieve the real-time judgment or on-line recognition of the target objects.

Human environments constantly change, and the received information is uncertain. Therefore, human cognition involves typical dynamical pattern recognition. Dynamical pattern recognition can be defined as follows: given a time point t , identify the target pattern Ω_t by the input pattern X_t . The input pattern is an observation instance described by the signature sequence of the target object at time t , and it is defined as $X_t = \{x_0, x_1, \dots, x_{m-1}\}$, $X_t \subset R^m$, in which R^m is an m -dimensional vector space and m is the number of signature sequences. Ω_t denotes the target states of the objects needed to be identified at time t , all of which constitute a target pattern state space $\Omega_t = \{\omega_0, \omega_1, \dots, \omega_{n-1}\}$, and n is the number of target patterns [42–45]. If it is assumed that there are q obtained observation sequences $y_q = \{y_{t-q+1}, y_{t-q+2}, \dots, y_t\}$ by a certain time t , the matter concerning the information processing involved in psychological cognition concerns the identification of a mapping from the input sequence y_q to target state space:

$$\phi : y_q \longrightarrow \Omega_t. \quad (1)$$

During the course of pattern recognition, through the use of the observation sequence y_q , the conditional probability $P(\omega_i | y_q)$ under every state must be calculated. Therefore, revealing the mapping from y_q to Ω_t is equal to determining

an optimum pattern state $\hat{\omega}$ that can obtain the maximum conditional probability $P_{\max}(\hat{\omega} | y_q)$ [46–48]. For the above process, two issues must be addressed: the observation value sequence y_q from the collected physical signal, whose owned feature information can satisfy the basic demands of state recognition, must be determined; the optimum pattern state $\hat{\omega}$, that is, the ground truth of every target state, must be obtained to act as the benchmark or threshold for on-line target state recognition.

Regarding the above operations, the information pre-processing and feature extraction of the target object are important and indispensable steps, which are related to the validity and reasonableness of the system recognition results. The extracted m feature parameters can construct a feature vector $X(m)$, and the number of feature parameters is the dimension of the feature vector. For a specific target object, we construct a data set that contains the feature vectors, related environments, existing conditions, and transactions to define a vector scope $E(X)$ for the target object. If the value of $E(X)$ is large, the vector scope is more extensive, and the computational load of feature extraction will be higher. Under the condition of a certain $E(X)$, the goal of feature extraction is to realize the coverage of information for the pattern or state of the target object. The range or extent of information coverage can be defined as the coverage ratio $Q(X)$ of the feature vector for the target object. For example, by using simple profile features, one can recognize daily necessities, people contacted one more than twice, and other features with a recognition accuracy up to 100%. Therefore, we believe that the $Q(X)$ of these profile features is close to 100%.

For the human information processing, the recognition of different people can be achieved by using the feature vector $\mathbf{X} = \{\text{head, shape, voice}\}$. Apparently, the head, shape, and voice are relatively rough or high-level features, where the vector element *head* can be divided into finer or lower-level features, including the mouth, eyes, ears, nose, forehead, and hair. Similarly, the other two elements can also be refined. Therefore, if an element x_i of feature vector is at higher level,

it can be refined into a lower-level vector. The above method of processing is defined as the refinement operation for this vector element, and it is denoted as

$$x'_i = \text{Ref}(x_i, r_i), \quad (2)$$

where x'_i is the refined vector element and r_i is the number of dimensions of the new vector after refinement. Accordingly, the feature vectors for which the $\text{Ref}()$ operation was conducted have nested structures; that is, the vectors should have more dimensions, and the number of dimensions for the refined vector is

$$m' = \sum_{i=0}^{m-1} r_i. \quad (3)$$

Considering the $\text{Ref}()$ operation, when the different data elements of a high-level vector must be refined with the same dimension, the defective dimensions of the data elements that are not easy to refine can be fulfilled by the *Null*. Apparently, the $\text{Ref}()$ operation makes the vector have more quantities of information than can be valued by information entropy, and the coverage ratio $Q(X)$ is increased, so the recognition accuracy for a specific target object will be improved. Of course, if the refinement dimensions arrive at a threshold value, whose $Q(X)$ is close to 100%, the $\text{Ref}()$ operation should be ceased to save computation resources. For example, the high-level vector element *head* can be refined as the low-level vector $x'_1 = \{\text{mouth, eyes, ears, nose, forehead, hair}\}$ to reduce the search range of the recognition process and increase the recognition efficiency.

However, even if a feature vector is refined, it cannot distinguish different target objects effectively and the appending operations for vector dimension should be executed, yielding the following expression:

$$X' = \text{App}(X, u), \quad (4)$$

where X' is the feature vector after dimensions are appended and u is the number of appended dimensions. Obviously, the feature vector by $\text{App}()$ operation also has more dimensions, and the number of dimensions should be $m + u$.

For instance, with respect to the recognition of target objects with high similarities, such as monozygotic twins, because most of their detailed characteristics are very close, it is difficult to achieve effective identification by using the $\text{Ref}()$ operation for original vector elements with a high number of dimensions. To resolve this matter, the vector should be processed by the $\text{App}()$ operation and transformed into a new vector, guaranteeing the recognition accuracy.

Many objects may share the same or similar characteristics; for example, aquatic mammals have many of the same features as fish, but they are not fish. Therefore, when the recognition requirements are satisfied, the feature vector should be effective in a special scope that can define the versatility of the feature vector. Apparently, versatility is one of the most important indexes to evaluate the quality of feature extraction. Therefore, the ultimate goal of feature extraction is to obtain feature vectors with a minimal number of dimensions that can meet the coverage of practical information for target objects and possess sufficient versatility.

3. ANNs-Based Cognition Computation and the PSACO Algorithm

3.1. BP-ANN-Based Learning/Recognition. For the practical applications of ANNs, most execution models adopt the BP network and its various algorithms. The BP network is a nonfeedback system comprising input layer, hidden layer (the number ≥ 1), and output layer. The input layer, hidden layer, and output layer are connected by a forward mode, and the number of neural neurons in each layer may be different. The connection weights of neural neurons can be trained by the error BP algorithm between the input layer and the hidden layer as well as between the hidden layer and output layer, and the thresholds among the three layers can also be changed in this way [49]. BP networks generally use the steepest descent gradient method (SDGM) to revise the weights and thresholds [50].

The BP network implements the mapping of input to output by using an error function minimization process. The error function, that is, the so-called cost function, is usually defined as the error sum of squares of the desired output and the actual output of all output layer neurons:

$$\varepsilon = \frac{1}{2} \sum_p (\widehat{O}_p - O_p)^2, \quad (5)$$

where p is the number of samples, \widehat{O}_p is the desired network output, and O_p is the actual output of the network [51].

The learning course of the BP network is realized by training the given data set, and it can address a large number of pattern mappings without any known mathematical knowledge to describe the mapping between the input and output. Therefore, the input patterns are mapped to the desired output patterns, with known patterns to train the network, and the network will exhibit recognition capability. Learning is a process in which humans continuously acquire external information, construct a memory database, and possess the ability to identify similar information. Therefore, it can be inferred that great similarities exist in the learning process between the neural networks and human beings, which is consistent with the conclusions in Section 2.

Using BP-ANN to recognize a specific target entity, we should define feature data, which builds up the input of the BP network from the information of the target object. As indicated in Section 2, all of the information elements of the target object are partially effective. Consequently, they must be preprocessed to transform them into m static or dynamic parameters as the input of the BP network, which are represented by the symbol $X = \{x_0, x_1, \dots, x_{m-1}\}$. Accordingly, the symbol $\Omega = \{\omega_0, \omega_1, \dots, \omega_{n-1}\}$ represents the ergodic state information of the target object.

According to the above hypothesis, a BP-ANN-based human cognition algorithm is described as follows.

(1) The m feature data elements (network input) and n expected patterns (network output) are normalized, allowing them to be processed and recognized by the computer.

(2) According to the fundamental characteristics of target objects, the appropriate action function is selected. Considering the continuities, extensibilities, and associations of

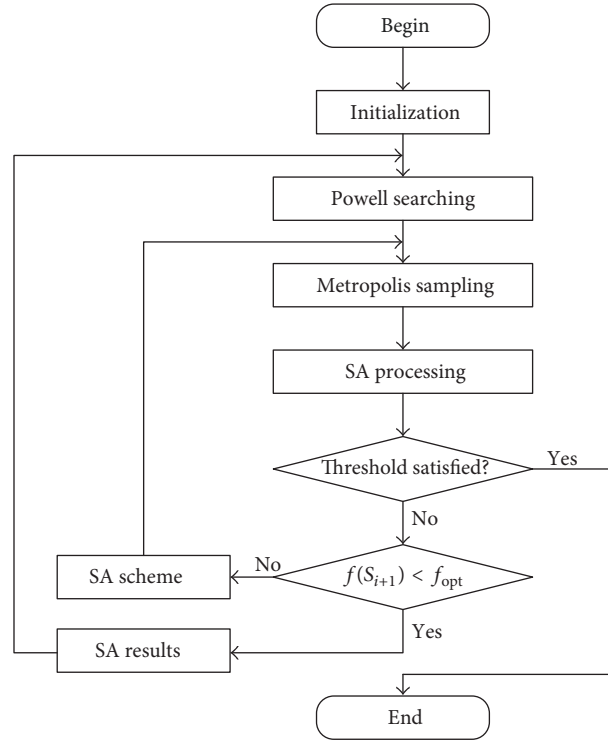


FIGURE 2: Flow chart of the PSACO algorithm.

psychological cognition, we can select the Sigmoid function [52, 53].

(3) In an empirical data set, the initial values of weights and thresholds are selected, and the neural network is initialized.

(4) The feature data of the target object is input by the input layer neurons, and after the transition actions of weights, thresholds, and the action function, the network output is obtained.

(5) The square sum τ of the error, that is, the difference between the current output and expected output, is calculated. If τ is less than the error threshold ε , then the algorithm is ended; otherwise, it proceeds to the next step.

(6) The connection weights from the output layer to the hidden layer and the output layer neuron thresholds are modified.

(7) The connection weights from the input layer to the hidden layer and the hidden layer neuron thresholds are updated.

(8) Does the algorithm arrive at the maximum number of training procedures? If so, the iteration ends; otherwise, it returns to step (3).

3.2. PSACO Algorithm. BP algorithm has the merits of simplicity and availability, but it is apt to fall into local optimal solutions, especially for the objects with high complexities or uncertainties. Since the Powell algorithm has efficient local searching ability, it is adopted to optimize the local searching process of SA iteration for ANN-based cognition, and a hybrid processing algorithm, the so-called Powell simulated annealing with combinatorial optimization (PSACO)

algorithm, is proposed. Accordingly, the PSACO algorithm has strong global and local searching ability, and the real-time performance and recognition accuracy of ANN will be improved. The procedures of the PSACO algorithm are shown in Figure 2, and the steps are described as follows.

(1) According to an original SA scheme, all the node weights of ANN construct a data set.

(2) The data set is regarded as a solution vector, and the target function is

$$E = \frac{1}{2p} \sum_p \sum_k (t_{pk} - o_{pk})^2, \quad (6)$$

where the minimum of target function is the optimal solution, p is the samples number, k is the output layer neuron number, t_{pk} is the expected output of p samples on k layer neurons, and o_{pk} is the factual output of p samples on k layer neurons.

(3) Based on the target function, a group of initializing network weights S_0 are created randomly, and the following initial conditions are configured: annealing temperature $T_0 > 0$, iteration times $i = 0$, and verification precision threshold is ε .

(4) Let $f_{opt} = f(S_0)$, $S_p = S_0$, where $f(x)$ is neuron action function.

(5) Let S_p be the initial point $S(0)$; the Powell algorithm is adopted to optimize the searching process.

(6) The first local minimal point is searched, and a new group of weights S'_p is obtained; then let $S_i = S'_p$, $f_{opt} = f(S_i)$.

(7) Let S_i be the iteration value x , $T = T_i$, and the current solution $S(i) = x$. The optimal conservation simulated

annealing operation is performed to obtain a new group of weights S_{i+1} .

(8) Let $i = i + 1$; the temperature descending method is

$$T_i = \frac{T_0}{1 + i}. \quad (7)$$

(9) If S_{i+1} matches the threshold, the PSACO algorithm will be ended. Otherwise, if $f(S_{i+1}) < f_{\text{opt}}$, then let $S_p = S_{i+1}$, and turn into step (4); if $f(S_{i+1}) \geq f_{\text{opt}}$, then let $S_i = S_{i+1}$, and turn into step (7).

Compared with the traditional SA algorithms, the improvements of the proposed PSACO algorithm are described as follows.

(1) As shown in Figure 2, the PSACO algorithm divides the searching process into three stages: Powell searching, Metropolis sampling, and SA processing. For the first stage, the Powell algorithm is to obtain an initial optimal point, which can reduce the search range and computation load of SA. Powell algorithm is an effective direct searching method, which divides the whole iteration into several stages, and every stage consists in $n + 1$ times of one-dimensional searching. Firstly, it searches along the n directions in turn and obtains a better point. Then, it searches along the direction of the line between the better point and output point and gains the optimal point in the stage. Finally, the last search direction replaces a smaller moving direction among the former n directions, and the next stage iteration is started. The steps of Powell algorithm are described as follows.

(a) For an initial point $x^{(0)}$, corresponding to n linearly independent directions: $d^{(1,1)}, d^{(1,2)}, \dots, d^{(1,n)}$, set an error threshold $0 < \varepsilon < 1$ and $k = 1$.

(b) Set $x^{(k,0)} = x^{(k-1)}$, start from the $x^{(k,0)}$, search along the directions of $d^{(k,1)}, d^{(k,2)}, \dots, d^{(k,n)}$ in turn, and obtain the points of $x^{(k,1)}, x^{(k,2)}, \dots, x^{(k,n)}$; start from the $x^{(k,n)}$, search along the direction of $d^{(k,n+1)} = x^{(k,n)} - x^{(k,0)}$, and obtain the $x^{(k)}$.

(c) If $\|x^{(k)} - x^{(k-1)}\| < \varepsilon$, stop iteration, and gain the $x^{(k)}$; otherwise, let $d^{(k+1,j)} = d^{(k,j)}$, where j is the smaller moving vector in the stage.

(d) Set $k = k + 1$, and return to step (b).

Apparently, for the lower data sampling rate of this stage, the Powell algorithm can perform the advantage of searching efficiency, reduce the search range and computation load of SA, and improve the convergence performance of the whole optimizing process.

(2) The annealing temperature iteration is performed by the method in (7). Because the temperature computation method of traditional SA algorithms is

$$T_i = \frac{T_0}{1 + \ln(i)}. \quad (8)$$

The proposed algorithm has higher cooling rate, that is, the approaching speed for the convergence threshold.

(3) For the Metropolis sampling algorithm, the disturbance variable ΔS_i is created by a random variable creator oriented to the Cauchy cost function:

$$\begin{aligned} S'_i &= S_i + \Delta S_i, \\ \Delta S_i &= \lambda \xi + \mu, \end{aligned}$$

$$\lambda = \frac{(S_{\text{max}} - S_{\text{min}})}{2} * 12.706,$$

$$\xi = \tan[(\eta - 0.5) * \pi], \quad (9)$$

where η is a random variable distributed uniformly in the interval (0, 1). Since the Cauchy probability density distribution has steeper peak and flatter flanks compared with the Gaussian distribution, it can perform more precise local searching, with the ability to jump out the local optimal solution. Therefore, the cooling process can adopt higher temperature descending rate in (7) to improve the convergence efficiency.

In order to verify the effectiveness and advantages of the proposed PSACO algorithm, the numerical instances and cognition experiments are performed, and the related contents are presented in Sections 4 and 5.

4. Numerical Instances and Clustering Results

4.1. Numerical Instances. To check the effectiveness of the proposed algorithm, this paper provides numerical instances for the process of human cognition. The implementation scheme involves the information collection on the images of target objects by machine vision methods. The image segmentation technologies are adopted to obtain the feature regions and identify the target objects by the recognition algorithms.

As shown in Figure 3, the target objects are four common animals: horse, donkey, mule, and goat. Since the target objects have many similar characteristics (quadruped, with hooves, a tail, long ears, and a quadruped shape), the process of recognizing and distinguishing the objects can reflect the general process of human cognition. Considering the above recognition targets, the numerical stimulation for cognition process includes three schemes: CNIS-I, CNIS-II, and CNIS-III.

(1) *CNIS-I Involves the Recognition of the Horse and Donkey.* The two animals share common features: quadruped, four hooves, one tail, and two long ears. According to general knowledge, we know that the donkey's ears are much longer than those of the horse, so they can be identified by the length of the ear. The feature vector can be defined as

$$\begin{aligned} X_1 &= \{x_{11}, x_{12}, x_{13}, x_{14}\} \\ &= \{\text{quadruped, hooves, tail, long ears}\}. \end{aligned} \quad (10)$$

Since the data elements in (10) are the existing qualitative features, the feature vectors corresponding to the horse and donkey are denoted as $X_{1h} = \{1, 1, 1, 0\}$, $X_{1d} = \{1, 1, 1, 1\}$. The above two feature vectors are the required training data (input patterns) for this recognition. Because the scope of the target object to be recognized involves only two animals, the relevant expected states (output patterns) are $\Omega_{1h} = \{1, 0\}$, $\Omega_{1d} = \{0, 1\}$, as shown in Table 1.

(2) *CNIS-II Requires the Recognition of the Horse, Donkey, and Mule.* The three kinds of animals share common features:

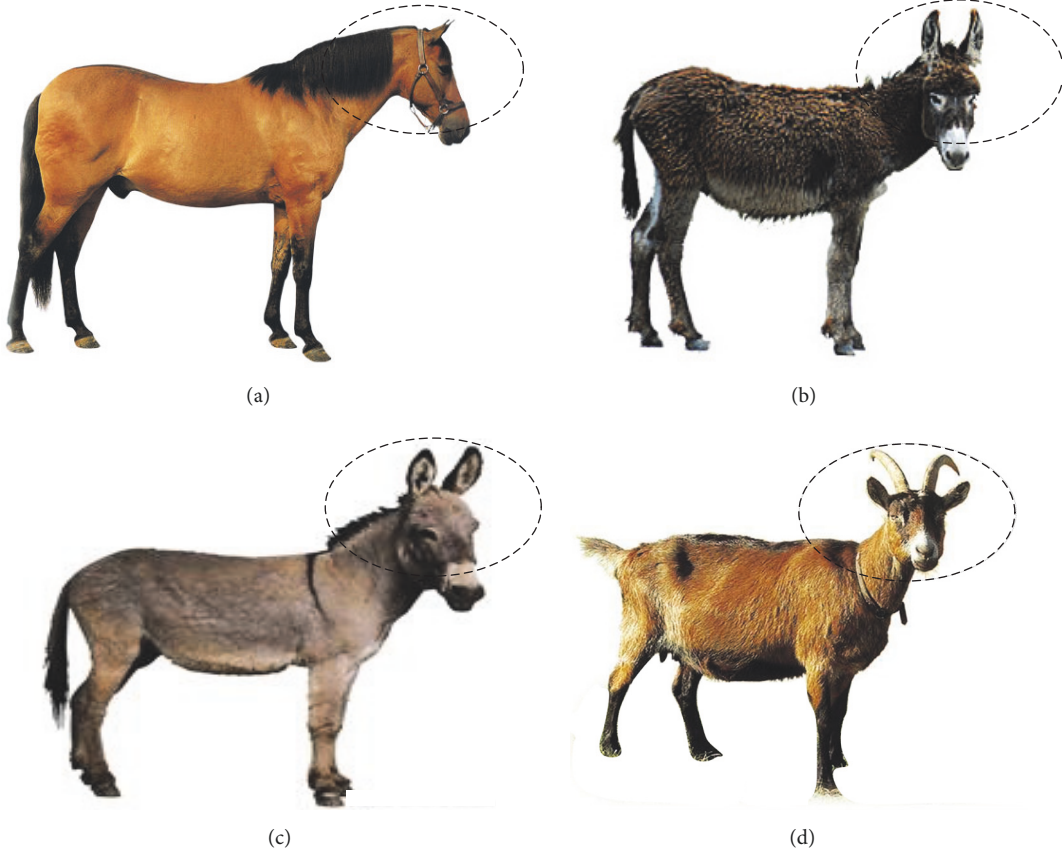


FIGURE 3: Target objects for cognition simulation: (a) horse, (b) donkey, (c) mule, (d) goat.

TABLE 1: Relationships of input/output patterns in CNIS-I.

| Objects | Input patterns | Output patterns |
|---------|---------------------------|--------------------------|
| Horse | $X_{1h} = \{1, 1, 1, 0\}$ | $\Omega_{1h} = \{1, 0\}$ |
| Donkey | $X_{1d} = \{1, 1, 1, 1\}$ | $\Omega_{1d} = \{0, 1\}$ |

quadruped, hooves, tail, and long ears. According to general knowledge, we know that the horse's ears are relatively short (accounting for 10% to 20% of the head length), the donkey has the longest ears (accounting for 60% to 80% of the head length), and the mule's ear length is somewhere in between (accounting for 30% to 40% of the head length). Therefore, the three animals can be identified by the ear length. However, the feature vector described in (10) cannot satisfy the recognition requirements. To solve this problem, the data element x_{14} in (10) must be processed with the Ref() operation, and the feature vector is transformed as

$$\begin{aligned} X_2 &= \{x_{21}, x_{22}, x_{23}, \text{Ref}(x_{14}, 1)\} \\ &= \{\text{quadruped, hooves, tail, long ears (ear length)}\}. \end{aligned} \quad (11)$$

Similarly, the data elements x_{21} , x_{22} , and x_{23} are the existing qualitative features. x_{14} is refined into the quantitative element x_{24} , so the feature vectors corresponding to the horse, donkey, and mule are denoted as $X_{2h} = \{1, 1, 1, 0.1 \sim 0.2\}$, $X_{2d} = \{1, 1, 1, 0.6 \sim 0.8\}$, $X_{2m} = \{1, 1, 1, 0.3 \sim 0.4\}$,

respectively. The above three feature vectors are the required training data (input patterns) for the second recognition. Because the target object scope contains three animals, the relevant expected states (output patterns) are $\Omega_{2h} = \{1, 0, 0\}$, $\Omega_{2d} = \{0, 1, 0\}$, $\Omega_{2m} = \{0, 0, 1\}$, as shown in Table 2.

(3) *CNIS-III Performs the Recognition of the Horse, Donkey, Mule, and Goat (Male)*. The four animals share the following common features: quadruped, four hooves, one tail, and two long ears. The goat's ears are relatively small, but the proportion of the head length is close to that of the mule, so the feature vectors in (10) and (11) cannot satisfy the requirements of the third recognition. To solve this problem, the vector in (10) must be processed with the App() operation; that is, a dimension, absence/presence of horns, must be added, and the vector can be transformed as

$$\begin{aligned} X &= \text{App}(X_1, 1) = \{x_{31}, x_{32}, x_{33}, x_{34}, x_{35}\} \\ &= \{\text{quadruped, hooves, tail, long ears, horns}\}, \end{aligned} \quad (12)$$

where the data elements also contain qualitative features. Therefore, the feature vectors of the horse, donkey, mule, and goat are recorded as $X_{3h} = \{1, 1, 1, 0.1 \sim 0.2, 0\}$, $X_{3d} = \{1, 1, 1, 0.6 \sim 0.8, 0\}$, $X_{3m} = \{1, 1, 1, 0.3 \sim 0.4, 0\}$, $X_{3g} = \{1, 1, 1, 0.3 \sim 0.4, 1\}$. The above described four feature vectors capture the recognition of the required training data (input mode). In this case, the four feature vectors are

TABLE 2: Relationships of input/output patterns in CNIS-II.

| Objects | Input patterns | Output patterns |
|---------|--------------------------------------|-----------------------------|
| Horse | $X_{2h} = \{1, 1, 1, 0.1 \sim 0.2\}$ | $\Omega_{2h} = \{1, 0, 0\}$ |
| Donkey | $X_{2d} = \{1, 1, 1, 0.6 \sim 0.8\}$ | $\Omega_{2d} = \{0, 1, 0\}$ |
| Mule | $X_{2m} = \{1, 1, 1, 0.3 \sim 0.4\}$ | $\Omega_{2m} = \{0, 0, 1\}$ |

TABLE 3: Relationships of input/output patterns in CNIS-III.

| Objects | Input patterns | Output patterns |
|---------|---|--------------------------------|
| Horse | $X_{3h} = \{1, 1, 1, 0.1 \sim 0.2, 0\}$ | $\Omega_{3h} = \{1, 0, 0, 0\}$ |
| Donkey | $X_{3d} = \{1, 1, 1, 0.6 \sim 0.8, 0\}$ | $\Omega_{3d} = \{0, 1, 0, 0\}$ |
| Mule | $X_{3m} = \{1, 1, 1, 0.3 \sim 0.4, 0\}$ | $\Omega_{3m} = \{0, 0, 1, 0\}$ |
| Sheep | $X_{3g} = \{1, 1, 1, 0.3 \sim 0.4, 1\}$ | $\Omega_{3s} = \{0, 0, 0, 1\}$ |

the required training data (input patterns) for the third recognition. Because the target object scope contains all four animals, the relevant expected states (output patterns) are as follows: $\Omega_{3h} = \{1, 0, 0, 0\}$, $\Omega_{3d} = \{0, 1, 0, 0\}$, $\Omega_{3m} = \{0, 0, 1, 0\}$, $\Omega_{3s} = \{0, 0, 0, 1\}$, as shown in Table 3.

4.2. Clustering Results Discussion. According to the numerical instance schemes in Section 4.1, three groups of data samples are provided. Each sample consists of four columns corresponding to the four animals (horse, mule, donkey, and goat), and a column contains 30 images. The first one is the original sample, as shown in Figure s1 of the supporting information section, which contains 120 images of the four animals, with regular orthogonal view and higher resolution. The second is the damaged sample, as shown in Figure s2, which has 120 damaged images by the methods of increasing transparency, varying RGB balance, changing contrast ratio, adding error points, and decreasing resolution, and aims to characterize the process of fault-tolerant learning. The last one is the transformed sample, as shown in Figure s3, in which the 120 images are shown by the pattern of different view angles, local amplifications, front profiles, and cartoons, and aims to verify the understanding ability of the algorithm. For once clustering or recognition, the required animal images are selected with random order and construct a training sample.

Taking the above data samples as the cognition target objects, three groups of clustering comparative experiments with the BP, SA, and restricted Boltzmann machine based extreme learning machine (RBM-ELM) algorithms [54, 55] are performed, in which each group of experiment is oriented to one numerical instance scheme (CNIS-I, CNIS-II, CNIS-III), and the results are shown in Figures 4, 5, and 6, respectively.

For the clustering results of CNIS-I about original sample (shown in Figure 4(a)), we can find that the clustering curves of the BP, SA, and PSACO algorithms converge quickly, and the iteration error can reach less than 10^{-8} after 80 iteration steps. Owing to the global search strategy of SA, the clustering performance of SA-ANN is worse than BP-ANN and PSACO algorithms. For the higher hidden layer neuron number and algorithm complexity, the RBM-ELM requires

more than 200 iteration steps to performance convergence. Due to the effects of Powell algorithm and combinatorial optimization, the proposed PSACO algorithm has the best convergence rate. The cognition objects of CNIS-I are very simple, except for the RBM-ELM algorithm, so the clustering differences of the other three algorithms are not apparent. In Figure 4(b), with the increment of complexities of cognition objects, the convergence iteration steps of four algorithms increase by 160 times, especially for the RBM-ELM. Similarly, the proposed algorithm has the best clustering performance, and the advantage tends to be clear. For the results about transformed sample (shown in Figure 4(c)), the convergence iteration steps increase successfully because of the higher complexities of target objects; the BP, SA, and PSACO algorithms require more 200 times, and the iteration steps of RBM-ELM algorithm exceed 500. Obviously, the proposed algorithm has apparent clustering advantage, and the average convergence rate can increase to 22.5%, 29.0%, 177%, respectively, compared with the BP, SA, and RBM-ELM algorithms.

From Figure 5, it can be inferred that the convergence iteration steps increase apparently because of the higher complexities of target objects, in which the dimension number of target vector increases to 3. For the clustering results of CNIS-II about original sample (shown in Figure 5(a)), we can find that the BP, SA, and PSACO algorithms take on good convergence trends, and the iteration error can reach less than 10^{-8} after 600 iteration steps. Similarly, the RBM-ELM algorithm has the lowest convergence rate, and the proposed PSACO algorithms have the best clustering performance. Different from Figure 4(a), owing to the enough complexity of target objects, the propose algorithm can obtain adequate superiority, and the convergence rate can increase to 19.2% and 42.0%, compared with the BP and SA algorithms. For the RBM-ELM, the proposed algorithm can perform more than three times of clustering efficiency. In Figure 5(b), the complexities of cognition objects increase continuously, the iteration steps of convergence exceed 1600 times, and the clustering advantage of the proposed algorithm tends to be more remarkable. For the results about transformed sample (shown in Figure 5(c)), the convergence iteration steps increase successively, and the BP, SA, and RBM-ELM algorithms require more 1000 times. However, the iteration steps

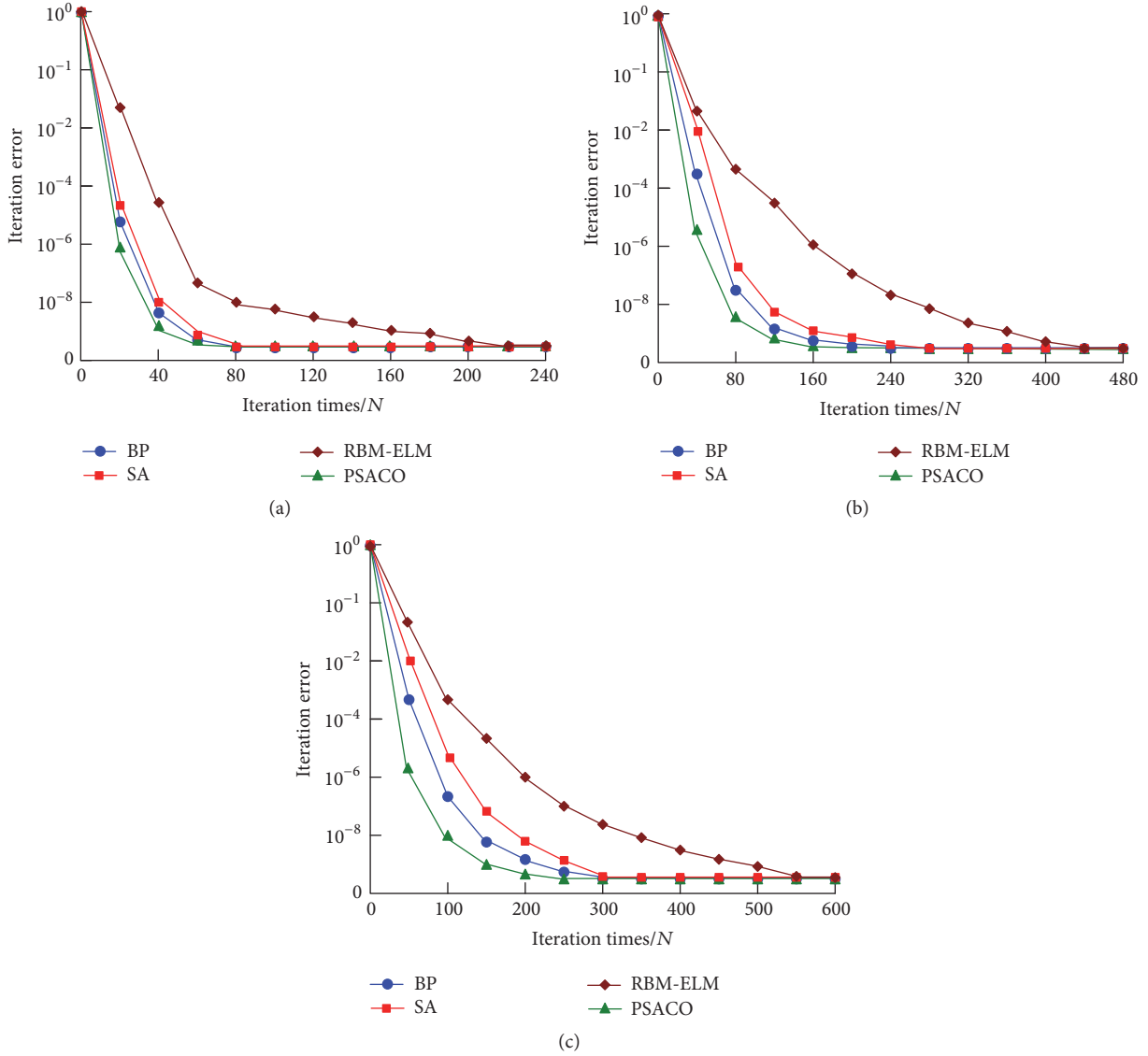


FIGURE 4: Clustering comparative results with the BP, SA, and RBM-ELM algorithms for CNIS-I: (a) original sample, convergence iteration times: 60, 71, 79, 223; (b) damaged sample, convergence iteration times: 163, 185, 206, 442; (c) transformed sample, convergence iteration times: 213, 275, 300, 543.

of the proposed algorithm only increase to less than 800 times, which proves that the method can be subjected to lower influences with the complexity increment of target objects.

From Figure 6, it is found that the convergence iteration steps increase continuously by the increment of complexities of target objects, in which the dimension number of target vector increases to 4. For the clustering results of CNIS-III about original sample (shown in Figure 6(a)), we can find that the four algorithms can converge after 1700 iteration steps. The RBM-ELM algorithm has the lowest convergence rate, and there are no apparent differences in the error curve of BP and SA algorithms. The proposed PSACO algorithm has the best clustering performance, with obvious advantage. In Figure 6(b), with the increment of complexities of cognition objects, except for the proposed algorithm, the convergence

iteration times of other three algorithms increase by more 1000. Similarly, the proposed algorithm can perform the best clustering performance, and reach more than two times of convergence rate compared with RBM-ELM algorithm. The convergence rates of the BP and SA algorithms are relatively close. For the results about transformed sample (shown in Figure 6(c)), the convergence iteration steps increase successively, in which the BP and SA algorithms require more 1600 times, and the iteration steps of RBM-ELM algorithm exceed 3000. Obviously, the proposed algorithm has apparent clustering advantage, and the convergence rate can increase to 25.4%, 25.8%, 160%, respectively, compared with the BP, SA, and RBM-ELM algorithms. Moreover, the above results prove that clustering performance of BP and SA algorithms tends to be limited for the target objects with higher complexities.

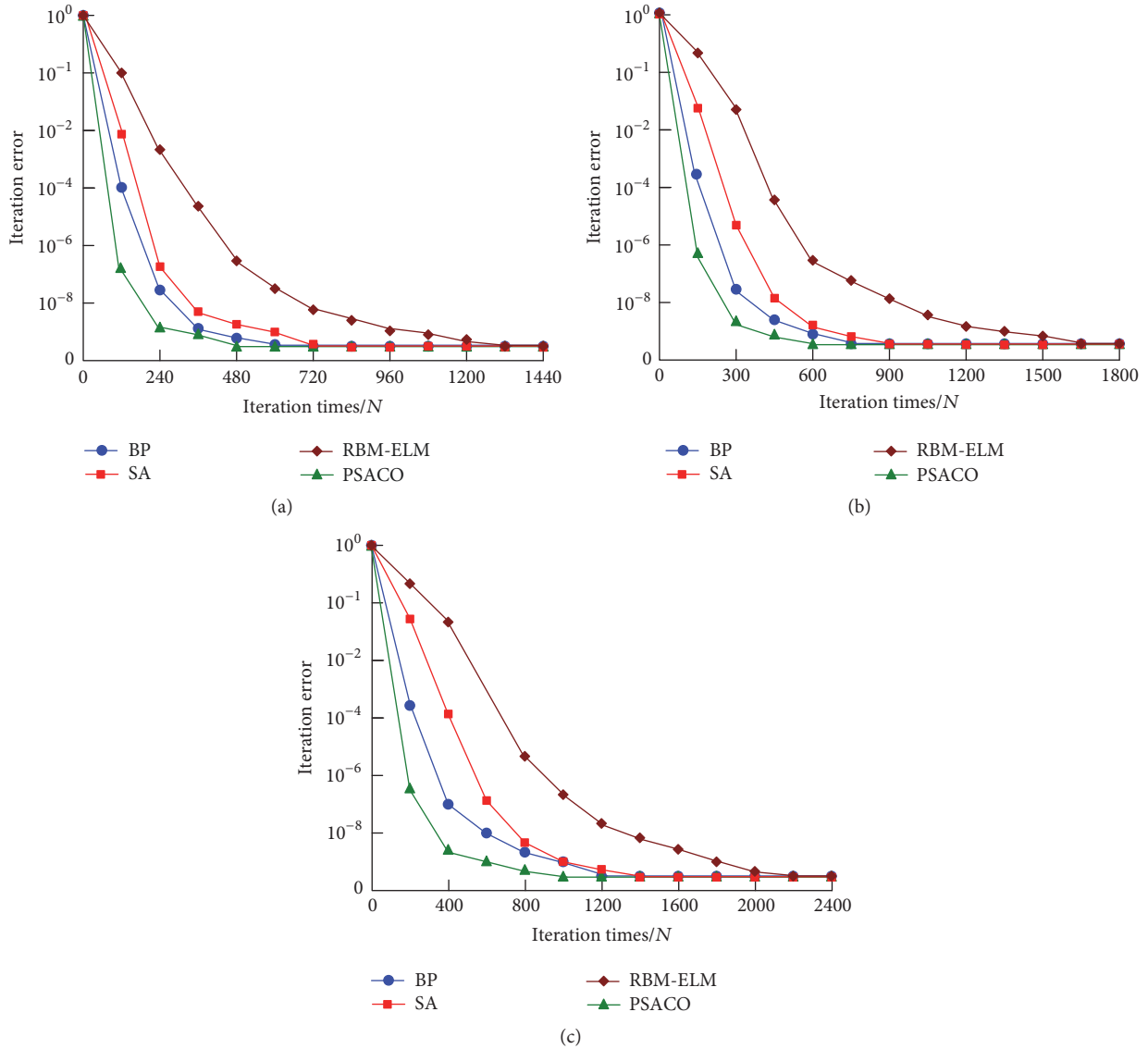


FIGURE 5: Clustering comparative results with the BP, SA, and RBM-ELM algorithms for CNIS-II: (a) original sample, convergence iteration times: 405, 501, 698, 1323; (b) damaged sample, convergence iteration times: 575, 622, 733, 1655; (c) transformed sample, convergence iteration times: 790, 1031, 1195, 2198.

From the three groups of clustering comparative experiments, the following laws can be obtained. (1) By the increment of object complexities, the clustering process requires more iteration steps, in which the original sample has the lowest cognition difficulties, and the transformed sample has the highest complexities. (2) The RBM-ELM algorithm has the worst convergence rate, due to its higher hidden layer neuron number and algorithm complexity. (3) For the global search pattern, the SA requires higher iteration steps than BP and PSACO. (4) Since the Powell algorithm and combinatorial optimization method are performed, the proposed PSACO algorithm can obtain the best clustering performance compared with the other three algorithms, especially for more complex target objects.

5. Cognition Experiments and Results Discussion

5.1. Instance Design of the Cognition System. Based on the above research works, we develop an application software system to verify the effectiveness of the proposed algorithm. As shown in Figure 7, the software system comprises a real-time data receiving and management component (DRMC), an image partition interface component (IPIC), a shape extraction component (SEC), an object characteristics library component (OCLC), and a system function interface (SFI).

The original data of the target objects is acquired by the self-designed embedded dual-CPU real-time image data acquisition card [56]. After the preprocessing operations of

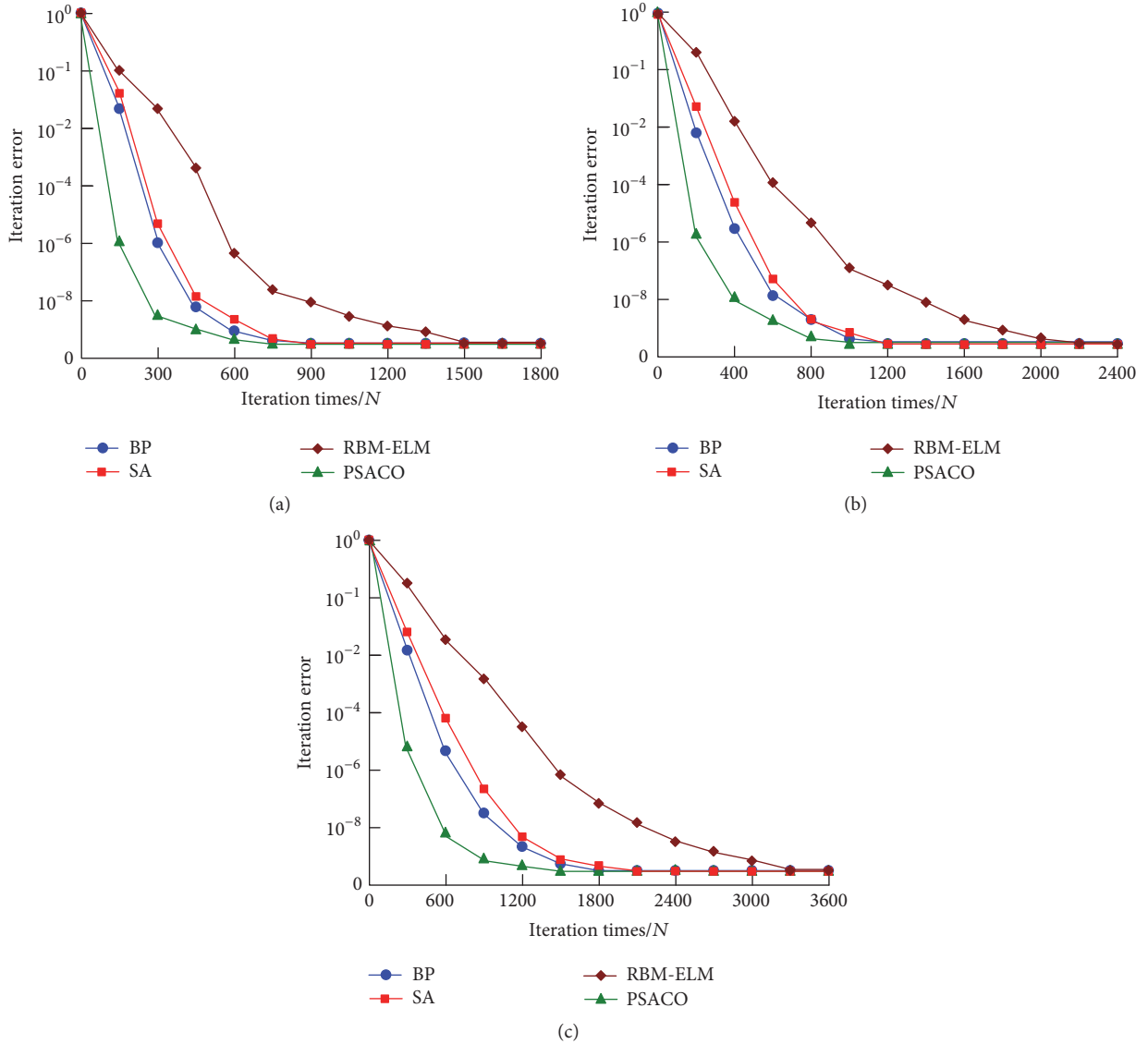


FIGURE 6: Clustering comparative results with the BP, SA, and RBM-ELM algorithms for CNIS-III: (a) original sample, convergence iteration times: 655, 748, 864, 1755; (b) damaged sample, convergence iteration times: 916, 1060, 1068, 2188; (c) transformed sample, convergence iteration times: 1217, 1631, 1640, 3312.

filtering and denoising, the real-time image data is packed according to the data transfer protocols and then uploaded to the DRMC. The DRMC is the entity of the SPM in the SA-ANN-based inference model that checks, unpacks, and distributes the data packets uploaded from the data acquisition card and that then provides an appropriate and reliable data source for the processing and computing of the following data. The IPIC and SEC are the entities of the CEM in a theoretical model. The IPIC is the real-time image segmentation oriented to the selected characteristic graphic region. The SEC adopts image processing algorithms to extract the feature data from the segmented image region, and the feature data are then stored in the OCLC by DSC as the knowledge database of the recognition system. The KSRC is the core component of the instance system, which performs knowledge learning and pattern recognition for target objects

through an ANN and constitutes the physical entity of the KSRM. The SFI corresponds to the modules of the OPM and SEM and takes charge of data postprocessing and output, which mainly include outputting results, displaying graphics, configuring data acquisition parameters, printing report sheets, monitoring intermediate results, and adjusting the target object description and characteristic parameters.

According to the scheme shown in Figure 7, in the Linux operating system, in combination with GNU-GCC/G++ and Qt technology, we develop application software for the ANNs-based simulation of human cognition, and the main control interface is shown in Figure 8. The software adopts the Tab-widget mode for the embedded system environment, and it can provide the configuration of system parameters, learning/training, cognition/recognition, and process

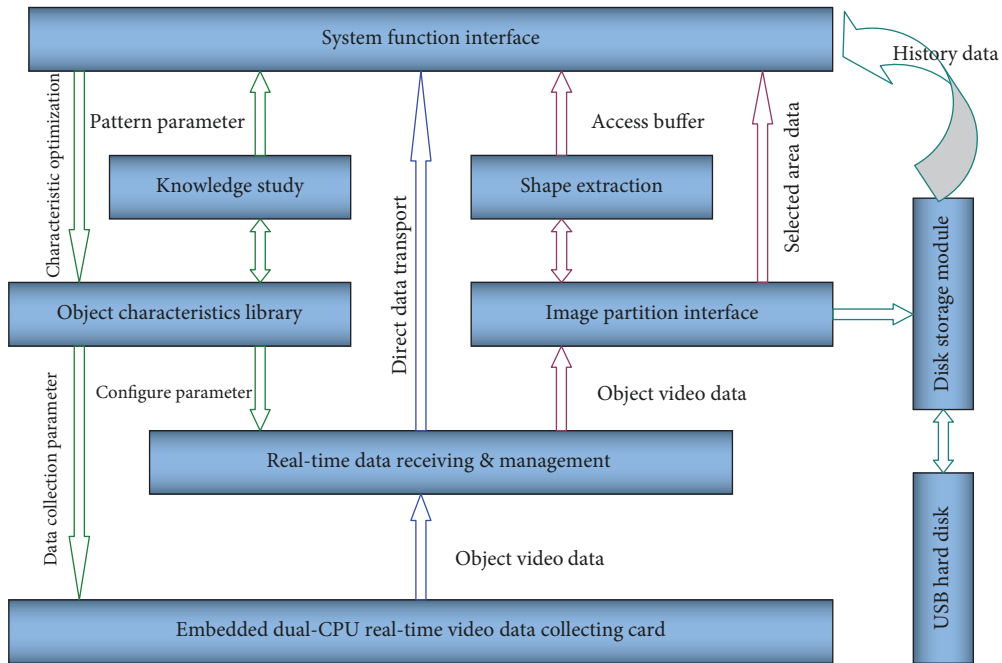


FIGURE 7: Module composition and data interaction of the cognition software.

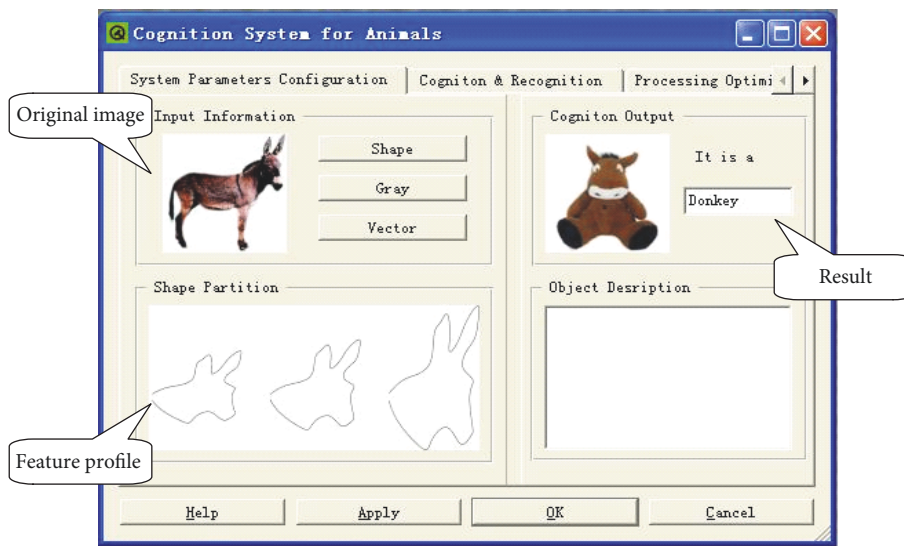


FIGURE 8: Main control interface of the cognition software.

optimization. With respect to the target objects (horse, donkey, mule, and goat) described in Section 4.1, the target image data are acquired by computer vision technology, and the original images are preprocessed by graying and vectorization. Then, the region segmentation for preprocessing images is performed on the selected feature image region [57–61]. In this case, the selected region of the feature image is the head, and the feature region profiles are shown in Figure 8. Based on the above operations, the feature vectors are constructed by the feature elements described in Section 4.1, and information on the process of cognition and the final recognition results is provided by the three algorithms.

5.2. *Experimental Results Discussion.* To check the effectiveness and advantages of the proposed PSACO algorithm, three groups of recognition comparative experiments with the BP, SA, and RBM-ELM algorithms are performed. The experimental procedures are described as follows. (1) The image data of the target objects are selected from the three groups of samples (original, damaged, and transformed) by the random order and construct a training sample. (2) The region segmentation for the preprocessed image is performed, and the image profile is obtained. (3) The feature data are extracted as the input of ANNs, and the recognition results are provided by the four algorithms.

TABLE 4: Recognition results of CNIS-I for original sample.

| Targets | Recognition results | | | | | | | | Recognition rate/% | | | |
|------------|---------------------|-----|-----|-------|------------|-----|-----|-------|--------------------|-----|-----|-------|
| | ω_1 | | | | ω_2 | | | | BP | SA | RBM | PSACO |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO |
| ω_1 | 200 | 200 | 200 | 200 | 0 | 0 | 0 | 0 | 100 | 100 | 100 | 100 |
| ω_2 | 0 | 0 | 0 | 0 | 200 | 200 | 200 | 200 | 100 | 100 | 100 | 100 |

TABLE 5: Recognition results of CNIS-II for original sample.

| Targets | Recognition results | | | | | | | | Recognition rate/% | | | | | | | |
|------------|---------------------|-----|-----|-------|------------|-----|-----|-------|--------------------|-----|-----|-------|------|------|-----|-------|
| | ω_1 | | | | ω_2 | | | | ω_3 | | | | | | | |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO |
| ω_1 | 296 | 300 | 300 | 300 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 98.7 | 100 | 100 | 100 |
| ω_2 | 5 | 0 | 0 | 0 | 292 | 298 | 300 | 300 | 3 | 2 | 0 | 0 | 97.3 | 99.3 | 100 | 100 |
| ω_3 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 298 | 300 | 300 | 300 | 99.3 | 100 | 100 | 100 |

TABLE 6: Recognition results of CNIS-III for original sample.

| Targets | Recognition results | | | | | | | | | | | | | | | | Recognition rate/% | | | |
|------------|---------------------|-----|-----|-------|------------|-----|-----|-------|------------|-----|-----|-------|------------|-----|-----|-------|--------------------|------|------|-------|
| | ω_1 | | | | ω_2 | | | | ω_3 | | | | ω_4 | | | | | | | |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO |
| ω_1 | 480 | 499 | 499 | 499 | 2 | 1 | 0 | 0 | 10 | 0 | 1 | 0 | 8 | 0 | 0 | 1 | 96.0 | 99.8 | 99.8 | 99.8 |
| ω_2 | 12 | 7 | 1 | 2 | 443 | 473 | 497 | 496 | 35 | 11 | 1 | 2 | 10 | 9 | 1 | 0 | 88.6 | 94.6 | 99.4 | 99.2 |
| ω_3 | 0 | 2 | 1 | 0 | 9 | 4 | 0 | 1 | 488 | 490 | 498 | 499 | 3 | 4 | 1 | 0 | 97.6 | 98.0 | 99.6 | 99.8 |
| ω_4 | 10 | 2 | 1 | 0 | 40 | 10 | 0 | 0 | 2 | 4 | 1 | 2 | 448 | 486 | 498 | 498 | 89.6 | 97.2 | 99.6 | 99.6 |

(1) *Cognition Results for Original Sample.* Firstly, taking the original sample as the target objects, the recognition comparative results for the four animals, corresponding to the three schemes (CNIS-I, CNIS-II, CNIS-III), are shown in Tables 4, 5, and 6, respectively.

For the CNIS-I, the horse and donkey are the target objects, represented by the following symbols: ω_1 and ω_2 . Two hundred experiments are completed, and the results are shown in Table 4. From the table, we can see that the four algorithms can recognize the target objects correctly because of lower target complexity.

Considering the second scheme, the horse, donkey, and mule are the target objects, represented by the symbols ω_1 , ω_2 , and ω_3 . To increase the computation difficulties, three hundred experiments are performed, and the results are shown in Table 5. From the table, the following results can be obtained: BP algorithm has 14 faulty decisions, and the average recognition rate decreases to 98.4%; SA algorithm begins to produce faulty decisions by the increment of complexities of target objects; the donkey is easy to identify because of its long ears, and the cognition difficulty of mule is relatively larger; RBM-ELM can perform higher classification accuracy and has no faulty decisions; since the feature vector is processed by refinement operations, the proposed PSACO algorithm can identify the three target objects accurately.

In the third stage, the horse, donkey, mule, and goat are the target objects, as represented by the symbols ω_1 , ω_2 , ω_3 , and ω_4 . To further increase the computation difficulties, five hundred times of experiments are performed, and the results

are shown in Table 6. From the table, we can obtain the following regularities: with the increase of recognition complexity, the recognition rates of BP and SA algorithm decrease apparently, especially for the mule and goat; the RBM-ELM and PSACO algorithms begin to produce faulty decisions caused by differences in the sample data of target objects, with recognition accuracies of more than 99%; since the refinement is applied and dimensions are added, the PSACO algorithm can reliably recognize the goat, demonstrating that the recognition performance of this method is close to that of humans.

(2) *Cognition Results for Damaged Sample.* As indicated in Section 4.1, the damaged sample has higher cognition difficulties and can characterize the fault-tolerant abilities of the algorithms. Concerning the damaged sample, the recognition results corresponding to the three schemes are shown in Tables 7, 8, and 9, respectively.

For the first experiment, the horse and donkey are the target objects. Two hundred times of recognition experiments are completed, and the results are shown in Table 7, where the SA, RBM-ELM, and PSACO algorithm can recognize the target objects correctly; the BP-ANN has one faulty decision that is caused by the damaged features.

In the second stage, the horse, donkey, and mule are the target objects, and the results of three hundred times of recognition experiments are shown in Table 8. From the table, we can obtain the following laws: by the increment of complexities of target objects, the proposed PSACO algorithm can keep recognition rate more than 99%; since the RBM-ELM

TABLE 7: Recognition results of CNIS-I for damaged sample.

| Targets | Recognition results | | | | | | | | Recognition rate/% | | | |
|------------|---------------------|-----|-----|-------|------------|-----|-----|-------|--------------------|-----|-----|-------|
| | ω_1 | | | | ω_2 | | | | BP | SA | RBM | PSACO |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO |
| ω_1 | 200 | 200 | 200 | 200 | 0 | 0 | 0 | 0 | 100 | 100 | 100 | 100 |
| ω_2 | 1 | 0 | 0 | 0 | 199 | 200 | 200 | 200 | 99.5 | 100 | 100 | 100 |

TABLE 8: Recognition results of CNIS-II for damaged sample.

| Targets | Recognition results | | | | | | | | | | | | Recognition rate/% | | | |
|------------|---------------------|-----|-----|-------|------------|-----|-----|-------|------------|-----|-----|-------|--------------------|------|------|-------|
| | ω_1 | | | | ω_2 | | | | ω_3 | | | | BP | SA | RBM | PSACO |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO |
| ω_1 | 263 | 290 | 295 | 299 | 18 | 6 | 2 | 1 | 19 | 4 | 3 | 0 | 87.7 | 96.7 | 98.3 | 99.7 |
| ω_2 | 15 | 2 | 4 | 0 | 254 | 284 | 293 | 297 | 31 | 14 | 3 | 3 | 84.7 | 94.7 | 97.7 | 99.0 |
| ω_3 | 16 | 5 | 3 | 0 | 18 | 6 | 3 | 1 | 266 | 289 | 294 | 299 | 88.7 | 96.3 | 98.0 | 99.7 |

algorithm requires larger sample number to guarantee the recognition accuracy, its recognition rate decreases to 98.0%; owing to more complex damaged features, the recognition rate of SA algorithm decreases obviously, especially for the mule; the recognition rate of BP algorithm decreases to less than 90% because of weaker fault-tolerant processing ability.

In the third stage, the horse, donkey, mule, and goat are the target objects, and the results of five hundred times of comparative experiments are shown in Table 9. From the table, the following regularities can be inferred: owing to the vector optimization operations, that is, element refining and dimension extending, the PSACO algorithm is subjected to lower influences by the increment of recognition complexity, with average recognition rate of 99.2%; the recognition rates of SA and RBM-ELM algorithms tend to decrease continuously, with no apparent differences; the BP algorithm has the worst recognition performance, and the average accuracy is less than 85%. Obviously, the proposed algorithm can cope with the feature differences generated by the damaged processing operations and can describe the fault-tolerant learning process of human cognition.

(3) *Cognition Results for Transformed Sample.* As indicated in Section 4.1, the transformed sample has different views and abstracted profiles, with the highest cognition difficulties, and can characterize the understanding abilities of the algorithms. With respect to the transformed sample, the recognition results corresponding to the three schemes are shown in Tables 10, 11, and 12, respectively.

For the first experiment, the horse and donkey are the target objects. Two hundred times of experiments are completed, and the results are shown in Table 10. Although this experiment has the lowest cognition complexity, it is hard for the BP algorithm to understand the transformed features of target objects, the SA algorithm has 9 faulty decisions, and the RBM-ELM algorithm makes 6 faulty decisions. Because of different transformed features, the proposed PSACO algorithm produces the first faulty decision for mule in this experiment.

In the second stage, the horse, donkey, and mule are the target objects, and the results of three hundred times of comparative experiments are shown in Table 11. From the table, we can obtain the following results: by the increment of complexities of target objects, the recognition accuracies of BP, SA, and RBM-ELM algorithms decrease apparently, with the average rates 80.2%, 91.5%, 91.9%, respectively; because of the combinatorial optimization for SA, the proposed PSACO algorithm can hold the transformed features and obtain a recognition rate of more than 97%.

In the third stage, the horse, donkey, mule, and goat are the target objects, and the results of five hundred times of comparative experiments are shown in Table 12. From the table, the following laws can be inferred: the recognition rate of BP algorithm has decreased to less than 80%; SA and RBM-ELM can produce close recognition effects, and the recognition rates have decreased to 91.0% for more complex transformed features. The recognition rate of the PSACO algorithm also decreases, yet it is more than 95%. The above results prove that the proposed algorithm can resolve the feature variations of the transformed images and can simulate the understanding process of human cognition.

6. Conclusions

Human cognition is an important topic of artificial intelligence, and many attempts suggest that ANNs can provide adequate research approaches. To address the problems of complex information understanding and fault-tolerant learning, a SA-ANN-based modeling method for human cognition is proposed. Concerning the above research target, the corresponding studies have been performed, and the conclusions are as follows.

(1) According to information processing theories, the physical process of human cognition and its fundamental characteristics are analyzed, and a SA-ANN-based inference model oriented to cognition is set up. The hierarchical structure, functional modules, and data dependencies of the model are presented, in which the KSRM is the core component

TABLE 9: Recognition results of CNIS-III for damaged sample.

| Targets | Recognition results | | | | | | | | | | | | Recognition rate/% | | | | | | | |
|------------|---------------------|-----|-----|------------|-----|-----|------------|-------|-----|------------|-----|-------|--------------------|-----|-------|-------|------|------|------|------|
| | ω_1 | | | ω_2 | | | ω_3 | | | ω_4 | | | SA | RBM | PSACO | | | | | |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | | | | |
| ω_1 | 428 | 477 | 490 | 498 | 32 | 10 | 3 | 1 | 15 | 10 | 4 | 0 | 25 | 13 | 3 | 1 | 85.6 | 95.4 | 98.0 | 99.6 |
| ω_2 | 22 | 7 | 6 | 2 | 413 | 463 | 475 | 492 | 53 | 21 | 9 | 2 | 12 | 9 | 10 | 4 | 82.6 | 92.6 | 95.0 | 98.4 |
| ω_3 | 3 | 6 | 6 | 1 | 29 | 10 | 4 | 1 | 435 | 480 | 488 | 498 | 33 | 4 | 2 | 0 | 87.0 | 96.0 | 97.6 | 99.6 |
| ω_4 | 12 | 6 | 2 | 0 | 40 | 10 | 4 | 2 | 23 | 4 | 4 | 2 | 425 | 478 | 490 | 496 | 85.0 | 95.6 | 98.0 | 99.2 |

TABLE 10: Recognition results of CNIS-I for transformed sample.

| Targets | Recognition results | | | | | | | | Recognition rate/% | | | |
|------------|---------------------|-----|-----|-------|------------|-----|-----|-------|--------------------|------|------|-------|
| | ω_1 | | | | ω_2 | | | | BP | SA | RBM | PSACO |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO |
| ω_1 | 190 | 199 | 199 | 200 | 10 | 1 | 1 | 0 | 95.0 | 99.5 | 99.5 | 100 |
| ω_2 | 22 | 8 | 5 | 1 | 178 | 192 | 195 | 199 | 89.0 | 96.0 | 97.5 | 99.5 |

TABLE 11: Recognition results of CNIS-II for transformed sample.

| Targets | Recognition results | | | | | | | | | | | | Recognition rate/% | | | |
|------------|---------------------|-----|-----|-------|------------|-----|-----|-------|------------|-----|-----|-------|--------------------|------|------|-------|
| | ω_1 | | | | ω_2 | | | | ω_3 | | | | BP | SA | RBM | PSACO |
| | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO | BP | SA | RBM | PSACO |
| ω_1 | 241 | 277 | 280 | 293 | 39 | 10 | 9 | 2 | 20 | 13 | 11 | 5 | 80.3 | 92.3 | 93.3 | 97.7 |
| ω_2 | 25 | 16 | 14 | 6 | 232 | 268 | 271 | 291 | 43 | 16 | 15 | 3 | 77.3 | 89.3 | 90.3 | 97.0 |
| ω_3 | 19 | 6 | 9 | 4 | 32 | 15 | 14 | 1 | 249 | 279 | 277 | 295 | 83.0 | 93.0 | 92.3 | 98.3 |

and should present parallel processing abilities for knowledge learning and pattern recognition. The rationalities and efficiencies of the algorithms in KSRM are the key factor in determining the effectiveness of the model and the relevant information processing system.

(2) Based on the inference model, the dynamical pattern recognition for human cognition is discussed systematically, and the corresponding information processing and feature extraction methods are proposed. The theoretical analysis shows that the ultimate goal of feature extraction is to obtain feature vectors with a minimal number of dimensions that can obtain practical information coverage for target objects and possess sufficient versatility.

(3) A PSACO algorithm for human cognition is proposed, and the corresponding computation procedures are presented. To obtain specific target objects, the key factors of cognition, including feature extraction, knowledge clustering, and pattern recognition, are defined and simulated, and the instances of numerical iteration are provided. The comparative numerical results show that the proposed algorithm has higher clustering efficiency, compared with the BP, SA, and RBM-ELM algorithms.

(4) Based on the proposed PSACO algorithm, a cognition software system is developed to determine the simulation conclusions. The comparative experimental results show that the proposed method can correctly recognize the testing objects, with adequate capabilities of information understanding and fault-tolerant learning, and can simulate the basic process of human cognition.

In summary, as its main scientific contribution, this paper introduces SA-ANN technologies into the area of human cognition and provides a mathematical path to accurate digital representation of human cognition. This work can not only offer direct suggestions for research on modeling methods, inference analysis, and the optimization of the process of artificial intelligence, but also provide a universal reference for research in the areas of dynamic pattern recognition, non-linear signal processing, monitoring of objects with uncertainty, computer-aided instruction, and modern rehabilitation treatment. Subsequent studies should focus on the facets of deep-learning and creative learning.

Nomenclature

| | |
|--------------------|--|
| $\text{App}()$: | Vector appending operation |
| d : | Searching direction |
| $E(X)$: | Vector scope |
| $f(x)$: | Neuron action function |
| f_{opt} : | Optimum solution |
| k : | Output layer neuron number |
| m : | Number of signature sequences |
| n : | Number of target patterns |
| o_{pk} : | Factual output of SA |
| O_p : | Actual network output |
| \bar{O}_p : | Desired network output |
| p : | Number of samples |
| $P(\omega y)$: | Conditional probability |
| q : | Number of observation sequences |
| $Q(X)$: | Coverage ratio |
| r_i : | Number of dimensions of the refined vector |
| $\text{Ref}()$: | Refinement operation |
| R^m : | m -dimensional vector space |
| S : | Network weight |
| t : | Time variable |
| t_{pk} : | Expected output of SA |
| T : | Annealing temperature |
| u : | Number of appended dimension |
| x_i : | Input data element |
| x'_i : | Refined vector element |
| X' : | Dimension-appended feature vector |
| $X(m)$: | Feature vector |
| X_t : | Input pattern at t |
| y_q : | Observation sequences |

Greek Letters

| | |
|------------------|-----------------------|
| ϕ : | General mapping |
| ε : | Error threshold |
| η : | Random variable |
| τ : | Error square sum |
| ω : | Pattern state element |
| $\hat{\omega}$: | Optimum pattern state |
| Ω_t : | Target pattern |

TABLE 12: Recognition results of CNIS-III for transformed sample.

| Targets | Recognition results | | | | | | | | | | | | Recognition rate/% | | | |
|------------|---------------------|-----|-----|------------|-----|-----|------------|-----|-----|------------|-----|-----|--------------------|------|------|-------|
| | ω_1 | | | ω_2 | | | ω_3 | | | ω_4 | | | BP | SA | RBM | PSACO |
| ω_1 | BP | SA | RBM | BP | SA | RBM | BP | SA | RBM | BP | SA | RBM | BP | SA | RBM | PSACO |
| | 382 | 460 | 455 | 44 | 15 | 5 | 4 | 25 | 15 | 6 | 49 | 10 | 2 | 76.4 | 92.0 | 97.6 |
| ω_2 | 42 | 17 | 12 | 366 | 442 | 453 | 477 | 55 | 31 | 8 | 37 | 10 | 7 | 73.2 | 88.4 | 95.4 |
| ω_3 | 35 | 6 | 19 | 58 | 10 | 11 | 4 | 388 | 454 | 490 | 33 | 19 | 2 | 77.6 | 90.8 | 98.0 |
| ω_4 | 22 | 4 | 18 | 50 | 17 | 15 | 8 | 43 | 15 | 4 | 385 | 464 | 486 | 77.0 | 92.8 | 97.2 |

Abbreviations

| | |
|---------|--|
| ANNs: | Artificial neural networks |
| BP: | Back-propagation |
| CEM: | Characteristic extraction module |
| DRMC: | Data receiving and management component |
| ELM: | Extreme learning machine |
| EmNN: | Emotional neural network |
| eRBP: | Event-driven random BP |
| HS-RNN: | Hybrid structure recursive neural network |
| IPIC: | Image partition interface component |
| KSRM: | Knowledge study and recognition module |
| LBP: | Lower back pain |
| NDANNs: | Nonlinear dynamical artificial neural networks |
| OCLC: | Object characteristics library component |
| OPM: | Output processing module |
| PSACO: | Powell SA with combinatorial optimization |
| RBM: | Restricted Boltzmann machine |
| SA-ANN: | Simulated annealing-artificial neural network |
| SDGM: | Steepest descent gradient method |
| SEC: | Shape extraction component |
| SEM: | Sensory effect module |
| SFI: | System function interface |
| SPM: | Sensory perception module |
| SRN: | Simple recurrent network |
| WCST: | Wisconsin card sorting test. |

Disclosure

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors wish to acknowledge the financial support of the Natural Science Foundation of China under Grants nos. 51775501, 51375446; the Zhejiang Provincial Science Foundation under Grant no. LR16E050001; the Zhejiang Provincial Health Department Program under Grant no. 2015KYA067; the Zhejiang Provincial Education Science Plan under Grants nos. 2013SCG386, 2017SCG028.

Supplementary Materials

Supplementary 1. Figure s1: Original sample of target objects.

Supplementary 2. Figure s2: Damaged sample of target objects.

Supplementary 3. Figure s3: Transformed sample of target objects.

References

- [1] A. Newell and H. A. Simon, "Computer simulation of human thinking," *Science*, vol. 134, no. 3495, pp. 2011–2017, 1961.
- [2] M. Kvassay, P. Krammer, L. Hluchý, and B. Schneider, "Causal analysis of an agent-based model of human behaviour," *Complexity*, vol. 2017, Article ID 8381954, 18 pages, 2017.
- [3] A. Newell, "The knowledge level," *Artificial Intelligence*, vol. 18, no. 1, pp. 87–127, 1982.
- [4] D.-P. Tan, S.-M. Ji, and M.-S. Jin, "Intelligent computer-aided instruction modeling and a method to optimize study strategies for parallel robot instruction," *IEEE Transactions on Education*, vol. 56, no. 3, pp. 268–273, 2013.
- [5] D. Tan, L. Zhang, and Q. Ai, "An embedded self-adapting network service framework for networked manufacturing system," *Journal of Intelligent Manufacturing*, pp. 1–18, 2016.
- [6] S. Rómoli, M. Serrano, F. Rossomando, J. Vega, O. Ortiz, and G. Scaglia, "Neural network-based state estimation for a closed-loop control strategy applied to a fed-batch bioreactor," *Complexity*, vol. 2017, Article ID 9391879, 16 pages, 2017.
- [7] J. Chen, J. Yang, J. Zhao, F. Xu, Z. Shen, and L. Zhang, "Energy demand forecasting of the greenhouses using nonlinear models based on model optimized prediction method," *Neurocomputing*, 2015.
- [8] D. P. Tan, S. T. Chen, G. J. Bao, and L. B. Zhang, "An embedded lightweight GUI component library and the ergonomics optimization method for industry process monitoring," in *Frontiers of Information Technology and Electronic Engineering*, 2017.
- [9] M. Diykh, Y. Li, and P. Wen, "Classify epileptic EEG signals using weighted complex networks based community structure detection," *Expert Systems with Applications*, vol. 90, pp. 87–100, 2017.
- [10] P. Sturt, F. Costa, V. Lombardo, and P. Frasconi, "Learning first-pass structural attachment preferences with dynamic grammars and recursive neural networks," *Cognition*, vol. 88, no. 2, pp. 133–169, 2003.
- [11] J. R. Vokey and P. A. Higham, "Opposition logic and neural network models in artificial grammar learning," *Consciousness and Cognition*, vol. 13, no. 3, pp. 565–578, 2004.
- [12] R. J. Tunney and D. R. Shanks, "Does opposition logic provide evidence for conscious and unconscious processes in artificial grammar learning?" *Consciousness and Cognition*, vol. 12, no. 2, pp. 201–218, 2003.
- [13] P. A. Higham and J. R. Vokey, "The controlled application of a strategy can still produce automatic effects: reply to redington," *Journal of Experimental Psychology: General*, vol. 129, no. 4, pp. 476–480, 2000.
- [14] D. Hallner and M. Hasenbring, "Classification of psychosocial risk factors (yellow flags) for the development of chronic low back and leg pain using artificial neural network," *Neuroscience Letters*, vol. 361, no. 1-3, pp. 151–154, 2004.
- [15] G. B. Kaplan, N. S. Şengör, H. Gürvit, I. Genç, and C. Güzeliş, "A composite neural network model for perseveration and distractibility in the Wisconsin card sorting test," *Neural Networks*, vol. 19, no. 4, pp. 375–387, 2006.
- [16] S. Hélie, S. Chartier, and R. Proulx, "Are unsupervised neural networks ignorant? Sizing the effect of environmental distributions on unsupervised learning," *Cognitive Systems Research*, vol. 7, no. 4, pp. 357–371, 2006.
- [17] D. S. Levine, "Neural network modeling of emotion," *Physics of Life Reviews*, vol. 4, no. 1, pp. 37–63, 2007.
- [18] S. Chartier, P. Renaud, and M. Boukadoum, "A nonlinear dynamic artificial neural network model of memory," *New Ideas in Psychology*, vol. 26, no. 2, pp. 252–277, 2008.

- [19] K. Tsagkaris, A. Katidiotis, and P. Demestichas, "Neural network-based learning schemes for cognitive radio systems," *Computer Communications*, vol. 31, no. 14, pp. 3394–3404, 2008.
- [20] L. Schmidt-Atzert, S. Krumm, and D. Lubbe, "Toward Stable Predictions of Apprentices' Training Success: Can Artificial Neural Networks Outperform Linear Predictions?" *Journal of Personnel Psychology*, vol. 10, no. 1, pp. 34–42, 2011.
- [21] K. Alexandridis and Y. Maru, "Collapse and reorganization patterns of social knowledge representation in evolving semantic networks," *Information Sciences*, vol. 200, pp. 1–21, 2012.
- [22] E. Grossi, A. Compare, and M. Buscema, "The concept of individual semantic maps in clinical psychology: A feasibility study on a new paradigm," *Quality & Quantity*, vol. 48, no. 1, pp. 15–35, 2014.
- [23] E. O. Neftci, C. Augustine, S. Paul, and G. Detorakis, "Event-driven random back-propagation: enabling neuromorphic deep learning machines," *Frontiers in Neuroscience*, vol. 11, Article ID 324, 2017.
- [24] O. K. Oyedotun and A. Khashman, "Prototype-incorporated emotional neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, no. 99, pp. 1–13.
- [25] D.-P. Tan, P.-Y. Li, Y.-X. Ji, D.-H. Wen, and C. Li, "SA-ANN-based slag carry-over detection method and the embedded WME platform," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 10, pp. 4702–4713, 2013.
- [26] V. Lukovac, D. Pamučar, M. Popović, and B. Đorović, "Portfolio model for analyzing human resources: An approach based on neuro-fuzzy modeling and the simulated annealing algorithm," *Expert Systems with Applications*, vol. 90, pp. 318–331, 2017.
- [27] T. Yang, A. A. Asanjan, M. Farizad, N. Hayatbini, X. Gao, and S. Sorooshian, "An enhanced artificial neural network with a shuffled complex evolutionary global optimization with principal component analysis," *Information Sciences*, vol. 418–419, pp. 302–316, 2017.
- [28] A. Demirenen, H. L. Zeynelgil, and N. S. Sengor, "Automatic generation control using ANN technique for multi-area power system with SMES units," *Electric Power Components and Systems*, vol. 32, no. 2, pp. 193–213, 2004.
- [29] C. Li, S.-M. Ji, and D.-P. Tan, "Multiple-loop digital control method for a 400-hz inverter system based on phase feedback," *IEEE Transactions on Power Electronics*, vol. 28, no. 1, pp. 408–417, 2013.
- [30] D.-P. Tan, S.-M. Ji, and Y.-Z. Fu, "An improved soft abrasive flow finishing method based on fluid collision theory," *The International Journal of Advanced Manufacturing Technology*, vol. 85, no. 5–8, pp. 1261–1274, 2016.
- [31] A. C. Zimmermann, C. L. N. Veiga, and L. S. Enemas, "Unambiguous signal processing and measuring range extension for fiber Bragg gratings sensors using artificial neural networks—a temperature case," *IEEE Sensors Journal*, vol. 8, no. 7, pp. 1229–1235, 2008.
- [32] S. Hussain and A. Alalili, "A hybrid solar radiation modeling approach using wavelet multiresolution analysis and artificial neural networks," *Applied Energy*, vol. 208, pp. 540–550, 2017.
- [33] D. P. Tan, T. Yang, J. Zhao, and S. M. Ji, "Free sink vortex Ekman suction-extraction evolution mechanism," *Acta Physica Sinica*, vol. 65, no. 5, Article ID 054701, 2016.
- [34] J. García, C. Pope, and F. Altiramas, "A distributed K -means segmentation algorithm applied to *lobesia botrana* recognition," *Complexity*, vol. 2017, Article ID 5137317, 14 pages, 2017.
- [35] D. Tan, S. Ji, P. Li, and X. Pan, "Development of vibration style ladle slag detection methods and the key technologies," *Science China Technological Sciences*, vol. 53, no. 9, pp. 2378–2387, 2010.
- [36] N. Zheng, L. Su, D. Zhang, L. Gao, M. Yao, and Z. Wu, "A computational model for ratbot locomotion based on cyborg intelligence," *Neurocomputing*, vol. 170, pp. 92–97, 2015.
- [37] S. Bahrami and F. Doulati Ardejani, "Prediction of pyrite oxidation in a coal washing waste pile using a hybrid method, coupling artificial neural networks and simulated annealing (ANN/SA)," *Journal of Cleaner Production*, vol. 137, pp. 1129–1137, 2016.
- [38] M. L. Dantas Dias and A. R. Rocha Neto, "Training soft margin support vector machines by simulated annealing: A dual approach," *Expert Systems with Applications*, vol. 87, pp. 157–169, 2017.
- [39] Z. Wu, N. Zheng, S. Zhang, X. Zheng, L. Gao, and L. Su, "Maze learning by a hybrid brain-computer system," *Scientific Reports*, vol. 6, no. 1, Article ID 31746, 2016.
- [40] N. Sengupta and N. Kasabov, "Spike-time encoding as a data compression technique for pattern recognition of temporal data," *Information Sciences*, vol. 406–407, pp. 133–145, 2017.
- [41] D. Tan, Y. Ni, and L. Zhang, "Two-phase sink vortex suction mechanism and penetration dynamic characteristics in ladle teeming process," *Journal of Iron and Steel Research, International*, vol. 24, no. 7, pp. 669–677, 2017.
- [42] L. Chen, W. Chew, and M. Garland, "Spectral pattern recognition of in situ FT-IR spectroscopic reaction data using minimization of entropy and spectral similarity (MESS): Application to the homogeneous rhodium catalyzed hydroformylation of isoprene," *Applied Spectroscopy*, vol. 57, no. 5, pp. 491–498, 2003.
- [43] J. Jurek, "Recent developments of the syntactic pattern recognition model based on quasi-context sensitive languages," *Pattern Recognition Letters*, vol. 26, no. 7, pp. 1011–1018, 2005.
- [44] B. X. Du, Y. Wu, Y. H. Lu, and X. Zhang, "PD pattern recognition of acoustic signals from polarity characteristics in generators," in *Proceedings of the ICPADM 2006 - 8th International Conference on Properties and Applications of Dielectric Materials*, pp. 710–713, Indonesia, June 2006.
- [45] D. P. Tan and L. B. Zhang, "A WP-based nonlinear vibration sensing method for invisible liquid steel slag detection," *Sensors and Actuators B: Chemical*, vol. 202, pp. 1257–1269, 2014.
- [46] I. Trendafilova, "Pattern recognition methods for damage diagnosis in structures from vibration measurements," *Key Engineering Materials*, no. 204–205, pp. 85–94, 2001.
- [47] I. Trendafilova and W. Heylen, "Categorisation and pattern recognition methods for damage localisation from vibration measurements," *Mechanical Systems and Signal Processing*, vol. 17, no. 4, pp. 825–836, 2003.
- [48] N. Zheng, M. Jin, H. Hong, L. Huang, Z. Gu, and H. Li, "Real-time and precise insect flight control system based on virtual reality," *IEEE Electronics Letters*, vol. 53, no. 6, pp. 387–389, 2017.
- [49] M. R. Widyanto, H. Nobuhara, K. Kawamoto, K. Hirota, and B. Kusumoputro, "Improving recognition and generalization capability of back-propagation NN using a self-organized network inspired by immune algorithm (SONIA)," *Applied Soft Computing*, vol. 6, no. 1, pp. 72–84, 2005.
- [50] R. G. Brereton, "Steepest ascent, steepest descent, and gradient methods," *Comprehensive Chemometrics*, vol. 1, pp. 577–590, 2010.
- [51] W. Wang, P. H. A. J. M. V. Gelder, J. K. Vrijling, and J. Ma, "Forecasting daily streamflow using hybrid ANN models," *Journal of Hydrology*, vol. 324, no. 1–4, pp. 383–399, 2006.

- [52] E. Soria-Olivas, J. D. Martin-Guerrero, A. J. Serrano-López, J. Calpe-Maravilla, J. Vila-Francés, and G. Camps-Valls, "Efficient pruning of multilayer perceptrons using a fuzzy sigmoid activation function," *Neurocomputing*, vol. 69, no. 7-9, pp. 909–912, 2006.
- [53] Z. J. Rong, B. B. Dan, and J. G. Yi, "A BP neural network predictor model for desulfurizing molten iron," *Lecture Notes in Computer Science*, vol. 3584, pp. 728–735, 2005.
- [54] P. Y. Jin, B. Y. Gao, H. J. Lu, and L. N. Chen, "A method of extreme learning machine based on restricted Boltzmann machine," *Mathematics in Practice and Theory*, vol. 46, no. 11, pp. 157–161, 2016.
- [55] X.-H. He, D. Wang, Y.-F. Li, and C.-H. Zhou, "A novel bearing fault diagnosis method based on gaussian restricted Boltzmann machine," *Mathematical Problems in Engineering*, vol. 2016, Article ID 2957083, 8 pages, 2016.
- [56] J. Kim, J. Kim, K. Shin, H. Lee, and S. Park, "ANN-based tensile force estimation for pre-stressed tendons of PSC girders using FBG/EM hybrid sensing," *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 59, no. 10, pp. 544–552, 2017.
- [57] R. Song and S. Chen, "A self-tuning proportional-integral-derivative-based temperature control method for draw-texturing-yarn machine," *Mathematical Problems in Engineering*, vol. 2017, Article ID 1864321, 17 pages, 2017.
- [58] S. Kobashi, N. Kamiura, Y. Hata, and F. Miyawaki, "Volume-quantization-based neural network approach to 3D MR angiography image segmentation," *Image and Vision Computing*, vol. 19, no. 4, pp. 185–193, 2001.
- [59] R. Yahiaoui, F. Alilat, and S. Loumi, "Parallelization of Fuzzy ARTMAP architecture on FPGA: multispectral classification of ALSAT-2A images," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 12, pp. 9487–9495, 2017.
- [60] S. Zhang, T. Wang, J. Dong, and H. Yu, "Underwater image enhancement via extended multi-scale Retinex," *Neurocomputing*, vol. 245, pp. 1–9, 2017.
- [61] D. Tan, L. Li, Y. Zhu, S. Zheng, and X. Jiang, "An embedded cloud database service method for distributed industry monitoring," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1–1, 2017.




Hindawi

Submit your manuscripts at
www.hindawi.com

