

Design of Solving Similarity Recognition for Cloth Products Based on Fuzzy Logic and Particle Swarm Optimization Algorithm

Bae-Muu Chang

Department of Information Management, Chienkuo Technology University
Chang-Hua, Chang-Hua 500, Taiwan, R.O.C.
[e-mail: bmchang@ctu.edu.tw]

*Corresponding author: Bae-Muu Chang

*Received November 1, 2016; revised February 16, 2017; revised April 26, 2017; accepted May 25, 2017;
published October 31, 2017*

Abstract

This paper introduces a new method to solve Similarity Recognition for Cloth Products, which is based on Fuzzy logic and Particle swarm optimization algorithm. For convenience, it is called the SRCPPF method hereafter. In this paper, the SRCPPF method combines Fuzzy Logic (FL) and Particle Swarm Optimization (PSO) algorithm to solve similarity recognition for cloth products. First, it establishes three features, length, thickness, and temperature resistance, respectively, for each cloth product. Subsequently, these three features are engaged to construct a Fuzzy Inference System (FIS) which can find out the similarity between a query cloth and each sampling cloth in the cloth database **D**. At the same time, the FIS integrated with the PSO algorithm can effectively search for near optimal parameters of membership functions in eight fuzzy rules of the FIS for the above similarities. Finally, experimental results represent that the SRCPPF method can realize a satisfying recognition performance and outperform other well-known methods for similarity recognition under considerations here.

Keywords: Similarity recognition, fuzzy inference system, particle swarm optimization, near optimal parameters, membership functions

1. Introduction

The recognition for handwritten characters or images plays an important role in the issue of research and development in pattern recognition and image processing [8, 14, 15, 17, 18, 20, 22]. Generally speaking, visual recognition process initiates with the extraction of some features from a handwritten character or an input image. In the past decade, many kinds of recognition methods have been realized via utilizing different feature representations [4, 5, 11, 12, 13, 16, 25]. For instance, the Chang's method compares an unknown input character with each of all sampling characters in the character database and calculates the similarity degrees between them according to a certain similarity measurement [5]. The above issues have strongly close relationships with similarity recognition. Accordingly, similarity recognition has become more and more popular in the research issues recently [11].

The Goltsev and Rachkovskij's method proposes a neural network which extracts orientation features that are employed to recognize hand drawn characters [9]. The network partitions input hand drawn characters into separate strokes in accordance with their orientations. There are eight neural layers in the network; each layer extracts all strokes for a certain orientation. The network adopts iterative update functions which comprise interactions of neurons in each layer by means of oriented excitatory connections and inhibitory interrelations among the corresponding neurons of different layers. Experiments display that the network can effectively classify all pixels for different hand drawn characters according to the orientations of the strokes which make up these characters and carry out a reasonable segmentation of characters. However, these hand drawn characters perhaps have randomly twisted shapes so that the proposed network cannot ensure the convergence for all pixels of input images [9].

The Chang's method utilizes a neuro-fuzzy technique to carry out handwritten character recognition [5]. The method combines a recurrent neural network (RNN) and a fuzzy inference system (FIS) to recognize different handwritten characters. It utilizes an RNN to powerfully extract oriented features for a handwritten character, and then, these features are exploited to construct an FIS which can effectively estimate the similarity ratings between a query character and each sampling character in the character database. Experimental results show that the method can achieve a satisfying recognition performance under considerations. Nevertheless, the Chang's method does not consider fuzzy logic with optimization software so as to achieve the near optimal parameters in the membership functions for fuzzy rules [5].

The Chang's method presents a character recognition technique which is called the Dempster-Shafer theory combined with neural network for handwritten character recognition [6]. The method combines an RNN and the Dempster-Shafer theory to recognize handwritten characters. First, it utilizes an RNN to powerfully extract oriented features of a handwritten character. Subsequently, the method constructs three feature variables employing extracted oriented features. Finally, three feature variables are fed into the Dempster-Shafer theory which can effectively estimate the similarity ratings between a query character and each sampling character in the character database. Experiments show that the method can realize a satisfying recognition performance. But, it does not take the advantage of characteristics of fuzzy logic [6].

The Khanale and Chitnis's method proposes a character recognition scheme which makes use of an artificial neural network (ANN) to recognize Devanagari handwritten characters [12]. The Devanagari is an alphabetic script which is utilized by a number of Indian languages. In this method, any character needs to be preprocessed and is converted into a 5×7 matrix of Boolean values. The image of character is cropped and resized to fix rows and columns. Further, it is classified into a certain class in accordance with its unique feature value by the ANN. The ANN is trained via batch propagation with adaptive learning rate. Experiments demonstrate that the method can achieve a satisfying recognition performance for certain handwritten Devanagari characters. However, the method cannot quite classify well for handwritten characters with noises and complicated writing styles [12].

The Kumar's method proposes an approach to define document image structural similarity for the applications of classification and retrieval [13]. It first builds a codebook of descriptors of speeded up robust features (SURF) which are extracted from a set of representative training images. And then, the method encodes each document and models the spatial relationships between two documents via recursively dividing the image and calculating histograms of codewords in each partition. In addition, the method trains a random forest classifier with the resulting features, which is employed for classification and retrieval. Nevertheless, it does not employ the characteristics of fuzzy logic in the method [13].

The Petrou's method proposes a rule-based classification methodology for habitat mapping via the use of a pre-existing land cover map and remote sensing data to handle uncertainty, missing information, and incorrect rule thresholds [19]. It utilizes the Dempster-Shafer theory in land cover to habitat mapping, in combination with fuzzy logic. The framework can handle lack of information by using composite classes, if necessary data for the discrimination of forming single classes are missing. Also, the method deals with uncertainty which is represented in domain expert knowledge. In addition, a number of fuzzification schemes are presented in the methodology in order to improve its performance and robustness. However, it does not consider the features of optimization in the work [19]. Accordingly, the method can be still improved.

An owner runs cloth business, who sells a variety of canvas [10]. Due to getting better business, he needs to utilize computer to handle customers' requests. There are 4000 different kinds of cloths in his company, which possess three features, length, thickness, temperature resistance, in each cloth's specification. It takes him much time to search for cloths for customers' orders. Moreover, he makes mistakes often for customers' requests due to many kinds of cloths. These motivate the author to propose an efficient method for the owner. The idea is that three features of a query cloth, which are a customer requests, are compared with those of each sampling cloth in the database one by one. That is called similarity recognition for cloth products [10].

Because the human concept of expertise knowledge can be developed by a set of fuzzy rules, we can maintain these fuzzy rules to enhance appropriate mappings to reach the nearly optimized recognition method [2, 5, 19]. Also, these rules are semantic-like, which are devised according to human prior knowledge and thinking reasoning. The FIS technique owns two

characteristics. One is that it can implement linguistic concepts. The other is that it can be easily realized to be a universal approximator, which fulfills a nonlinear mapping from an input domain to a corresponding output domain [2, 5, 19]. In addition, the PSO algorithm with the switching scheme based on soft computing is very good in nearly optimizing the parameters of membership functions for fuzzy rules in the FIS. That is, it is necessary to employ a systematic approach to find out the appropriate parameters for the recognition methods [2, 3, 7, 24].

The importance of solving similarity recognition and the features of aforementioned FIS and PSO motivate the author to design the SRCFPF method to solve recognition for cloth products. The proposed method creates three features, *length* L , *thickness* T , and *temperature resistance* R , for each cloth product. These three features are employed to build the FIS to find out the similarity between any two cloths. First, the fuzzy rule base in the FIS is composed of eight fuzzy rules [5]. Each trapezoid-shaped membership function has four parameters, which is utilized in these fuzzy rules. Subsequently, the PSO algorithm is employed to nearly optimize these parameters of membership functions in eight fuzzy rules in order to achieve better recognition results. Finally, a trained model is realized to effectively search for the similarities between a query cloth and sampling cloths in \mathbf{D} . In the testing phase, the SRCFPF method also creates three features for any query cloth, by utilizing the determined creation method in the trained model. These three features are fed into the trained model, and then the similarity results are obtained. More specifically, the above idea represents the strength and the main contribution of proposed SRCFPF method.

The remainder of this work is arranged as follows. Section 2 introduces the backgrounds which are composed of the FIS and the PSO algorithm. Section 3 describes the SRCFPF method in detail, such as conceptual design, designs of FIS, training and testing phases, etc. Section 4 indicates the near optimal parameter solutions and experimental results. Finally, the concluding remarks are proposed in Section 5.

2. The Backgrounds

2.1 The conceptual FIS

Zadeh presented the fuzzy theory in 1965 and depicted the difference between a traditional set and a fuzzy set. In the traditional set, the membership degrees of the elements are represented by crisp values. However, each element in the fuzzy set is denoted as a membership degree of an element. These membership degrees are represented by the numbers in the interval $[0, 1]$, which can better express human knowledge and logical reasoning. In recent years, the fuzzy theory has been widely exploited in the application fields, such as image processing, pattern recognition, and decision analysis, etc. [5, 15, 19]. In this paper, the FIS is employed to fulfill the similarity between the features of a query cloth C_q and the corresponding features of each sampling cloth C_l in \mathbf{D} .

More specifically, the basic structure of the FIS is shown in Fig. 1. Main functions in the structure contain the fuzzifier, the fuzzy rule base, the fuzzy inference engine, and the defuzzifier, respectively. First, the fuzzifier converts the input crisp values into the linguistic

fuzzy information, such as trapezoidal or triangular membership function. Subsequently, the fuzzy rule base is composed of various fuzzy inference rules, that is, the fuzzy inferences are rule-based [5, 19]. These fuzzy inference rules appear with IF-THEN format. In addition, the fuzzy inference engine aggregates the results of fuzzy inference rules to form a fuzzy set via utilizing, for instance, max-min of Mamdani method. Finally, the defuzzifier converts the fuzzy set into a crisp value via adopting, for example, the gravity method or the centroid method [5, 15, 19].

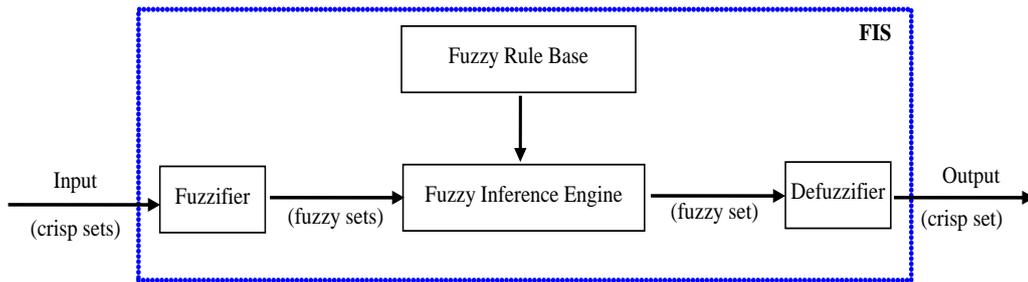


Fig. 1. The structure of a fuzzy inference system

2.2 The PSO algorithm

Kennedy proposed the PSO algorithm in 1995 which was developed according to birds' path of movement when they search for foods. It is exploited to solve optimization problems in many applications [1, 3, 7, 23, 24]. In the algorithm, birds decide the next direction and the distance of movement via referencing the past directions of movement and the current location when they look for foods. A particle in the PSO algorithm is simulated as an individual for a bird's food searching. The feature that a particle is simulated as a biological individual in the hyper-dimensional search space is employed to search for near optimal solution. That is to say, each particle will be simulated for each individual in the group of society in psychological tendencies. Apart from the individual search direction, each particle will also study the best individual in the group of society.

The current fitness value for each particle is compared with that of each individual best position (Pbest). If the particle's fitness value is greater than or equal to that of Pbest, Pbest will be kept unchanged. Or, the individual best position will be replaced with the particle's current position. Furthermore, Pbest is compared with Gbest. If the particle's fitness value is less than or equal to Gbest fitness value, the position of Gbest will be substituted with the particle's fitness value [3, 7, 24]. More clearly, Pbest and Gbest are described as follows.

$$\text{if } \text{fitness}(X_i^t) < \text{fitness}(\text{Pbest}_i) \text{ then } \text{Pbest}_i = X_i^t$$

$$\text{if } \text{fitness}(X_i^t) < \text{fitness}(\text{Gbest}) \text{ then } \text{Gbest} = X_i^t$$

where t represents the t th generation, $i = 1, 2, \dots, n$, n indicates the number of particles, $X_i = (X_{i1}, X_{i2}, \dots, X_{im}) \in \mathfrak{R}^m$ stands for the i th particle's current position, m denotes the particle's dimensionality, $\text{Pbest}_i \in \mathfrak{R}^m$ represents the best position for the i th individual particle's path of movement, $\text{Gbest} \in \mathfrak{R}^m$ indicates the position closest to the optimal solution in the group, and fitness means the fitness function.

More specifically, the training procedure for the PSO algorithm is fully depicted by the following algorithm and Fig. 2.

- Step 1: Initialize the starting parameters, such as swarm size, weight, range of movement for particles, and the number for training iterations. Moreover, the particles are randomly positioned and the movement vector is arbitrarily provided.
- Step 2: Save Gbest and all Pbest positions at the current iteration according to an evaluation process via exploiting the fitness function for all particles.
- Step 3: If the number of training iterations reaches or the accurate rate satisfies, then Gbest and Pbest positions are achieved, and the algorithm terminates. Otherwise, go to Step 4.
- Step 4: Count the movement vectors for all particles, which are depicted in Eq. (1).
- Step 5: Update the positions for all particles by means of adopting Eq. (2) and then go to Step 2.

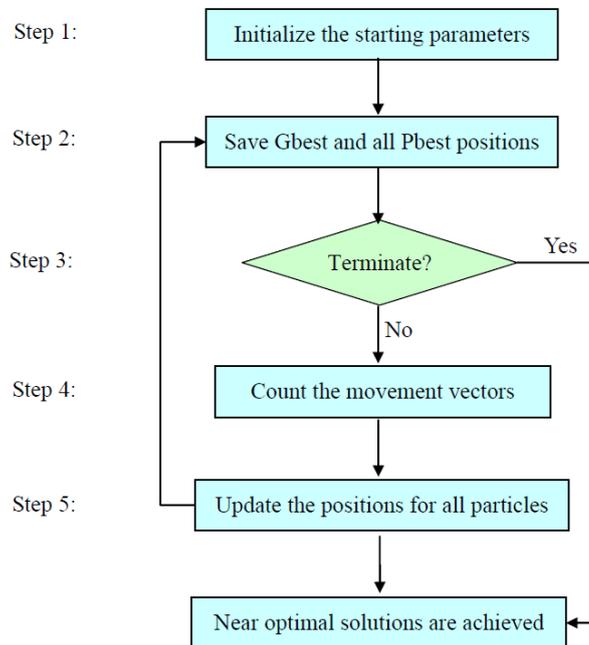


Fig. 2. The conceptual structure for the PSO algorithm

Moreover, the movement vector is represented by Eq. (1):

$$\mathbf{V}_i^{t+1} = w\mathbf{V}_i^t + c_1 \times r_1 \times (\text{Pbest}_i - \mathbf{X}_i^t) + c_2 \times r_2 \times (\text{Gbest} - \mathbf{X}_i^t), \quad (1)$$

where $\mathbf{V}_i = (V_{i1}, V_{i2}, \dots, V_{im}) \in \mathfrak{R}^m$, \mathbf{V}_i^{t+1} stands for the movement vector of particle i at the $(t+1)$ th iteration, w denotes the inertia weight, c_1 and c_2 represent the acceleration coefficients which are random numbers in $[0, 1]$, meanwhile, r_1 and r_2 are also two randomly given values in $[0, 1]$. In addition, the first term $w\mathbf{V}_i^t$ in Eq. (1) indicates the particle's inertia, the second term $c_1 \times r_1 \times (\text{Pbest}_i - \mathbf{X}_i^t)$ denotes the particle's cognition-only model, and the third term $c_2 \times r_2 \times (\text{Gbest} - \mathbf{X}_i^t)$ represents the particle's social-only model. Additionally, $\text{Pbest}_i \in \mathfrak{R}^m$ denotes the near optimal position for the i th individual particle's path of movement and $\text{Gbest} \in \mathfrak{R}^m$ indicates the position closest to the near optimal solution in the group.

Furthermore, the location of particle i is updated by Eq. (2).

$$\mathbf{X}_i^{t+1} = \mathbf{X}_i^t + \mathbf{V}_i^{t+1} \quad (2)$$

where \mathbf{X}_i^{t+1} indicates the location of particle i at the $(t+1)$ th iteration. That is to say, the next location of particle i is adding its current location vector to its next movement vector [3, 5, 24].

3. The SRCPPF method

Fig. 3 depicts the conceptual design for the SRCPPF method. During the training phase, the method first creates three kinds of features for cloth products in the cloth database \mathbf{D} , which are respectively *length* L , *thickness* T , and *temperature resistance* R . The input vectors (L, T, R) of cloths are fed into the FIS. It is imaginable that there are a large number of parameters of membership functions in eight fuzzy rules to measure the similarities between two cloths. Accordingly, it is a very important problem to select a near optimal parameter set among the huge number of parameters, and a decision threshold before employing the SRCPPF method to perform the testing phase. Here, the SRCPPF method makes use of the PSO algorithm to solve the above problems. That is, the PSO algorithm is utilized in the design of the SRCPPF method to find out a near optimal parameter set of above huge number of parameters in the membership functions of eight fuzzy rules, and a near optimal decision threshold. More clearly, \mathbf{P} contains a set of parameters for each membership function in eight fuzzy rules, and the decision threshold. Meanwhile, \mathbf{P}_0 stands for the near optimization set for \mathbf{P} .

During the testing phase, three features, *length* L , *thickness* T , and *temperature resistance* R , of a query cloth are first created by utilizing three feature creation method which is determined in the training phase. Then, in the recognition process, three features between the query cloth and each cloth in \mathbf{D} , can be employed to measure their similarities. Finally, the highest similarity is selected as the similarity degree between the query cloth and a sampling cloth.

3.1 Three features of each cloth

Let f denote a set of features including *length* L , *thickness* T , and *temperature resistance* R , respectively. It can be represented by $f = (L, T, R)$. While matching a query cloth C_q with a sampling cloth C_l in \mathbf{D} , the SRCPPF method first creates an input vector (L, T, R) for the FIS

[5, 16, 21]. Note that, $L = \frac{L_q}{L_l}$ or $\frac{L_l}{L_q}$, $T = \frac{T_q}{T_l}$ or $\frac{T_l}{T_q}$, $R = \frac{R_q}{R_l}$ or $\frac{R_l}{R_q}$.

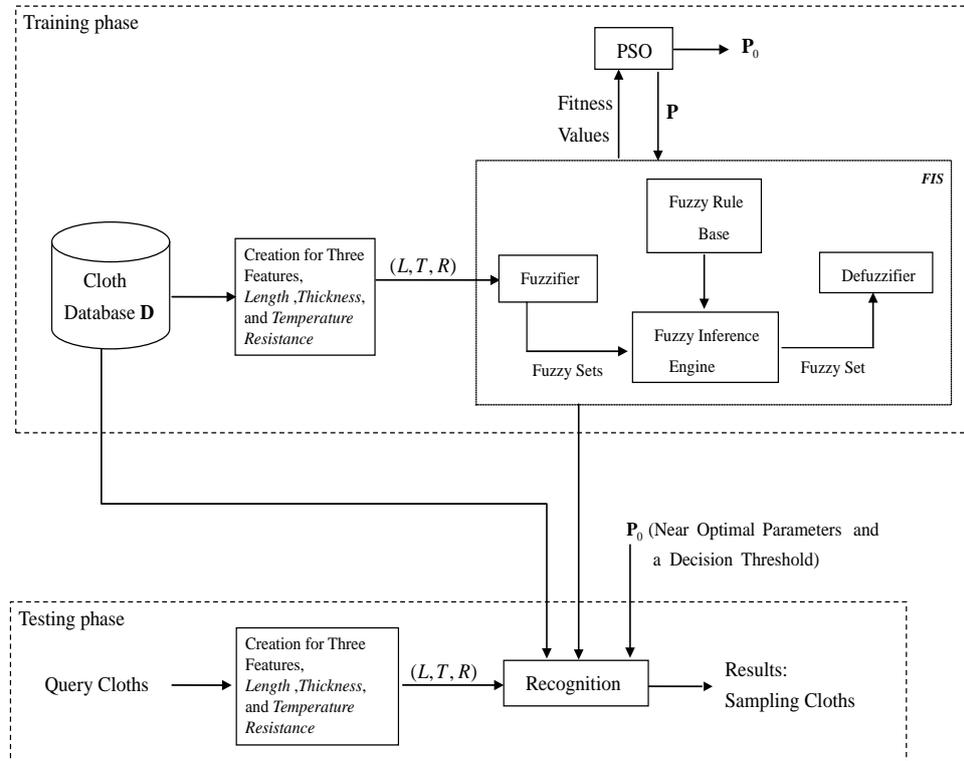


Fig. 3. The conceptual design for the SRCPPF method

3.2 The conceptual design for the FIS

Three kinds of features, the *length* L , the *thickness* T , and the *temperature resistance* R , respectively, are employed for cloth recognition in the SRCPPF method. The inputs of FIS contain three values of linguistic variables, L, T , and R . Note that the FIS plays a role of a binary classifier. That is, it maps an input vector (L, T, R) to $\{0, 1\}$. Three components in the input vector can be normalized in $[0, 1]$ without loss of generality. Thus, the FIS performs a mapping from $[0, 1]^3$ to $\{0, 1\}$. In the proposed method, this mapping can be implemented by the FL technique, shown in Fig. 1 [5, 19]. In the testing phase, the input vector (L, T, R) is fed to the FIS which utilizes the defined semantic fuzzy rules to recognize the query cloth. As a result, a constant decision function is employed to determine whether the query cloth is similar to a sampling cloth or not.

The fuzzy rule base in the FIS is composed of eight fuzzy rules [5]. The membership functions utilized in these fuzzy rules can nearly optimize recognition results. Subsequently, the FIS takes Mamdani's method as the fuzzy inference method. The fuzzy outputs of eight fuzzy rules are aggregated by utilizing the maximum operation. Also, the centroid method is employed to defuzzify the aggregation of fuzzy outputs of eight fuzzy rules in the FIS, and then the FIS produces each similarity with a crisp number [5, 19]. Finally, the SRCPPF method displays the corresponding matched sampling cloth product [5].

The fuzzy rule base consists of a set of fuzzy rules which appear with if-then format. According to the structure knowledge, the mapping behavior of FIS can be indicated by means

of eight fuzzy rules, shown in Fig. 4. Three input linguistic variable values (L, T, R) and two individual membership functions (L, H) in the premise part, form $2^3 = 8$ fuzzy rules [5]. In the consequence part, the fuzzy linguistic variable, S , respectively associates with five linguistic terms, very low (VL), low (L), medium (M), high (H), and very high (VH). Note that, each fuzzy rule contains four fuzzy linguistic variables, (L, T, R) and S (similarity degree). For L, T , and R , each linguistic variable has two linguistic terms, high (H) and low (L), while S has five linguistic terms, VH, H, M, L, and VL. The above seven linguistic terms can be represented by fuzzy sets with 1-D trapezoid-shaped membership functions [5]. These membership functions of linguistic terms have the same support of $[0, 1]$. In general, these membership functions are 1-D nonlinear functions [5]. For example, Table 1 shows seven kinds of membership functions which are defined in a general form by Eq. (3),

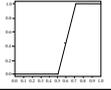
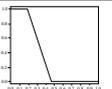
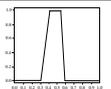
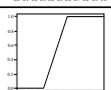
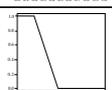
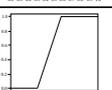
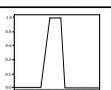
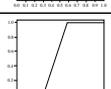
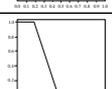
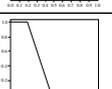
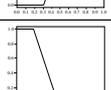
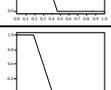
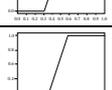
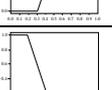
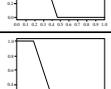
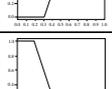
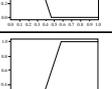
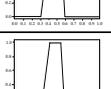
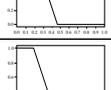
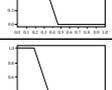
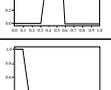
$$\mu(x; p, q, r, s) = \begin{cases} 0, & x \leq p \\ \frac{x-p}{q-p}, & p \leq x \leq q \\ 1, & q \leq x \leq r \\ \frac{s-x}{s-r}, & r \leq x \leq s \\ 0 & s \leq x \end{cases}, \quad x \in [0, 1] \quad (3)$$

where $\mu(x; p, q, r, s)$ indicates the membership function for seven linguistic terms H, L, VH, H, M, L, and VL. Note that corresponding parameters p, q, r , and s of seven kinds of membership functions are different. For instance, parameters r and s for linguistic term H are more than one. Also, p and q for L are less than zero [5].

- Rule 1: If L is H and T is H and R is H then S is VH
- Rule 2: If L is H and T is H and R is L then S is M
- Rule 3: If L is H and T is L and R is H then S is M
- Rule 4: If L is H and T is L and R is L then S is L
- Rule 5: If L is L and T is H and R is H then S is H
- Rule 6: If L is L and T is H and R is L then S is M
- Rule 7: If L is L and T is L and R is H then S is M
- Rule 8: If L is L and T is L and R is L then S is VL

Fig. 4. Eight fuzzy rules in the FIS

Table 1. Membership functions for eight fuzzy rules

Rule	Premise Part			Consequence Part
	Length	Thickness	Temperature Resistance	Similarity
	<i>L</i>	<i>T</i>	<i>R</i>	<i>S</i>
1	 High	 High	 High	 Very High
2	 High	 High	 Low	 Medium
3	 High	 Low	 High	 Medium
4	 High	 Low	 Low	 Low
5	 Low	 High	 High	 High
6	 Low	 High	 Low	 Medium
7	 Low	 Low	 High	 Medium
8	 Low	 Low	 Low	 Very Low

Let Ψ indicate the set of parameters which decide the shapes of these membership functions:

$$\Psi = \bigcup_{j=1}^8 \{ \bigcup_{i=1}^4 \{ (p_{ji}, q_{ji}, r_{ji}, s_{ji}) \} \} \tag{4}$$

where p_{ji}, q_{ji}, r_{ji} , and $s_{ji}, i = 1, 2, 3$ denote four parameters of the i th membership function in the premise part of Rule j , respectively. Moreover, p_{j4}, q_{j4}, r_{j4} , and s_{j4} respectively represent four parameters of membership function in the consequence part of Rule j [5].

In the FIS, the fuzzy inference engine utilizes the max-min integration of Mamdani method. The fuzzy outputs of eight fuzzy rules are aggregated via employing the maximum operation. In addition, the centroid method is utilized to defuzzify the aggregation of eight fuzzy outputs [5]. Finally, in the process of defuzzifier, a threshold constant δ is employed to determine whether the query cloth C_q is similar to the sampling cloth C_l or not. In a word, the FIS serves as a binary classifier which assigns the query cloth C_q into one of two classes. If the output crisp value is more than δ , it indicates C_q is similar to C_l . Otherwise, C_q is dissimilar to C_l .

As my research, the length L_q of a query cloth C_q similar to L_l of a sampling cloth C_l in

\mathbf{D} indicates high similarity degree between C_q and C_l in *length*. Also, T_q similar to T_l represents high similarity degree between the query cloth and the sampling cloth in *thickness*. Meanwhile, R_q similar to R_l denotes high similarity degree between the query cloth and the sampling cloth in *temperature resistance*. In this case, it has a very high possibility of the similarity degree between the query cloth and the sampling cloth. Conversely, low similarity degree in *length*, little similarity degree in *thickness*, and low similarity degree in *temperature resistance*, convey a very low possibility of the similarity degree between the query cloth and the sampling cloth [5].

As the author's implementations, an observation is found out that the *thickness* T and the *temperature resistance* R are more important than the *length* L . All values of three fuzzy linguistic variables are high, the similarity between the query cloth C_q and the sampling cloth C_l in \mathbf{D} is definitely very high, as Rule 1. Conversely, all values of three fuzzy linguistic variables are low, the similarity is absolutely very low, as Rule 8. Similarly, Rule 4 is low and Rule 5 is high. In addition, Rules 6 and 7 are medium. Remainders, two of three are high, the similarity is medium, as Rules 2 and 3.

3.3 The construction of training and testing sets

The training and the testing sets for the SRCPPF method are built via dividing \mathbf{D} with $(100 - r)\%$ and $r\%$, respectively. Let \mathbf{D}^{tr} and \mathbf{D}^{te} respectively stand for the training and the testing sets, where $\mathbf{D} = \mathbf{D}^{tr} \cup \mathbf{D}^{te}$, $\mathbf{D}^{tr} \cap \mathbf{D}^{te} = \phi$. Furthermore, let m , m_{tr} , and m_{te} represent the cardinalities of \mathbf{D} , \mathbf{D}^{tr} , and \mathbf{D}^{te} , respectively. Thus, $m = m_{tr} + m_{te}$, $\frac{m_{tr}}{m} \times 100 = 1 - r$, and $\frac{m_{te}}{m} \times 100 = r$. Also, C_l^{tr} stands for the l th cloth of

\mathbf{D}^{tr} and its corresponding feature vector is denoted by f_l^{tr} of \mathbf{D}^{tr} in \mathbf{D} . In addition, the PSO algorithm is utilized to select near optimal parameters Ψ and a decision threshold δ for the training set of cloths in \mathbf{D} . More clearly, the training procedure of SRCPPF method is described as follows:

Step 1: Input \mathbf{D} .

Step 2: For each cloth C_l^{tr} in \mathbf{D}^{tr} of \mathbf{D}

create the length feature L , the thickness feature T , the temperature resistance feature R , respectively.

Step 3: For \mathbf{D}^{tr} in \mathbf{D} ,

// PSO procedure
For each generation // for each epoch of PSO algorithm

For each particle k // for each parameter set, (Ψ_k, δ)

For each $C_l^{tr} \in \mathbf{D}^{tr}$ // for all training cloths

adjust the parameters of membership functions in eight fuzzy rules
for C_l^{tr} itself.

Loop

calculate the fitness for (Ψ_k, δ) which is the average of accurate rates for each C_l^{tr} .

save parameters Pbest and Gbest which have the highest fitness during

each generation.

Loop

save Gbest in (Ψ_0, δ_0) .

Loop

Loop

Step 4: Output (Ψ_0, δ_0) , where $\Psi_0 = \bigcup_{j=1}^8 \{ \bigcup_{i=1}^4 \{ (p_{ji}^0, q_{ji}^0, r_{ji}^0, s_{ji}^0) \} \}$ represent the near optimal parameters and δ_0 denotes the final decision threshold.

Note that the accurate rate AR which is defined in Eq. (5) is utilized in the design of the fitness function for selecting Gbest and Pbest positions of particles. More specifically, for a candidate parameter Ψ_k , its fitness function is the average of accurate rates of all training cloths, C_l^{tr} , in \mathbf{D}^{tr} of \mathbf{D} . Subsequently, the subprogram ‘‘Cloth Recognition’’ can be described in the following.

Cloth Recognition:

Step 1: Set the parameters of the SRCFPF method with Ψ_0, δ_0 .

Step 2: For a query image $C_q \in \mathbf{D}^{te}$,

three determined creation methods in the training phase are employed to create three kinds of features, L, T , and R , respectively.

Step 3: For each C_q^{te} in \mathbf{D}^{te} of \mathbf{D}

three features are fed into the FIS to obtain a similar sampling cloth.
calculate the accurate rate AR using Eq. (5).

Step 4: Output the AR .

On the completion of the training phase of SRCFPF method, it outputs \mathbf{P}_0 for the whole set of cloths \mathbf{D}^{tr} in \mathbf{D} . Then, the cloth recognition can be realized by calling ‘‘Cloth Recognition’’. In this paper, the optimization of the SRCFPF method is to search for the near optimal parameters, $\Psi_0 = \bigcup_{j=1}^8 \{ \bigcup_{i=1}^4 \{ (p_{ji}^0, q_{ji}^0, r_{ji}^0, s_{ji}^0) \} \}$, by selecting from the parameter combinations, and also its corresponding near optimal final decision, δ_0 . Accordingly, the near optimal solution can be represented by $\mathbf{P}_0 = \Psi_0 \cup \{ \delta_0 \}$.

4. The experimental results

4.1 The cloth database \mathbf{D}

Many cloth data from Jie-Tian Company in the Internet are collected and built as the cloth database \mathbf{D} which is utilized for performance evaluation in this paper [10]. Here, \mathbf{D} comprises 4000 cloth data while each one is represented as $f = (L, T, R)$. L, T , and R respectively represent the length, the thickness, and the temperature resistance.

4.2 The quantitative index for the recognition performance

The accurate rate of similarity for each generation is calculated as the fitness of PSO algorithm, where the accurate rate indicates the rate between the number of accurate evaluation samples and the number of total evaluation samples, shown in Eq. (5) [26].

$$\text{fitness} = \text{accurate rate } AR = \frac{TP + TN}{TP + FN + FP + TN} \times 100\% \quad (5)$$

where TP (True Positive) and TN (True Negative) denote correct evaluation samples, FP (False Positive) and FN (False Negative) represent false evaluation samples. FP indicates that the number of incorrect cloths which are mistakenly considered as correct. FN denotes that the number of correct cloths which are mistakenly regarded as incorrect.

The training and the testing sets are constructed via dividing the construction of \mathbf{D} with $(100 - r)\%$ and $r\%$, respectively, where r is set to 25 in the experiments. Note that the AR is employed for the design of fitness function for deciding Gbest and Pbest positions of particles. Moreover, the fitness function is the average of AR for all training cloths in \mathbf{D}_t^r .

4.3 The near optimization of membership functions in the FIS

When the SRCFPF method nearly optimizes the membership functions employed in eight fuzzy rules in the FIS, there are very large amount of calculations for the parameters of membership functions in these eight fuzzy rules. Therefore, it is extremely complicated to search for near optimal values of these parameters in the membership functions. In the past decade, the PSO algorithm has been popularly employed to solve this type of optimization problems [3, 24]. Consequently, in the SRCFPF method, the PSO algorithm is exploited to find out an approximate optimal solution for the optimization problem. That is to say, it is indispensable to utilize a systematic approach to receive these appropriate parameters and the near optimal solution.

The FIS makes use of the PSO algorithm to achieve four approximate optimal parameters for each trapezoid-shaped membership function in the FL technique and the threshold constant, δ , for final decision. $\mathbf{P} = \Psi \cup \{\delta\}$ is composed of a set of parameters for each membership function in eight fuzzy rules and the decision threshold, δ . In the SRCFPF method, the number of particles is set to 50, in the PSO algorithm. The algorithm exploits the fitness function shown in Eq. (5) so as to evaluate the fitnesses of particles. It expires when the number of largest training generation reaches 50 or the fitness value satisfies the problem's requirements. Assume $\mathbf{P}_0 = \Psi_0 \cup \{\delta_0\}$ be a near optimal solution for the optimization problem in the FIS. It can be achieved via selecting the best one with the highest average of AR among those of candidate particles. In addition, δ_0 represents the near optimal decision threshold which judges whether the query cloth is similar to the sampling cloth or not. Ψ_0 is a near optimal parameter set selected from Ψ in Eq. (4), which can be searched for by the PSO algorithm and be indicated by Eq. (6),

$$\Psi_0 = \bigcup_{j=1}^8 \left\{ \bigcup_{i=1}^4 \{ (p_{ji}^0, q_{ji}^0, r_{ji}^0, s_{ji}^0) \} \right\} \quad (6)$$

where $p_{ji}^0, q_{ji}^0, r_{ji}^0$, and s_{ji}^0 denote the near optimal parameters for the i th membership function in Rule j , respectively.

Table 1 exhibits that each linguistic term has four parameters and each fuzzy rule has four linguistic terms. That is, there are totally 128 parameters in eight fuzzy rules, which are displayed in Eq. (4). Hence, the cardinality of the near optimal parameter set Ψ_0 , $|\Psi_0|$, is 128 and the cardinality of \mathbf{P}_0 , $|\mathbf{P}_0|$, is 129, exhibited in **Tables 2** and **3**.

Table 2. The near optimal parameters for membership functions in eight fuzzy rules

Rule 1	$(p_{11}, q_{11}, r_{11}, s_{11}) = (0.431, 0.603, 1, 1)$	$(p_{12}, q_{12}, r_{12}, s_{12}) = (0.385, 0.562, 1, 1)$	$(p_{13}, q_{13}, r_{13}, s_{13}) = (0.395, 0.570, 1, 1)$	$(p_{14}, q_{14}, r_{14}, s_{14}) = (0.542, 0.735, 1, 1)$
Rule 2	$(p_{21}, q_{21}, r_{21}, s_{21}) = (0.455, 0.596, 1, 1)$	$(p_{22}, q_{22}, r_{22}, s_{22}) = (0.376, 0.543, 1, 1)$	$(p_{23}, q_{23}, r_{23}, s_{23}) = (0, 0, 0.178, 0.4300)$	$(p_{24}, q_{24}, r_{24}, s_{24}) = (0.326, 0.407, 0.520, 0.581)$
Rule 3	$(p_{31}, q_{31}, r_{31}, s_{31}) = (0.457, 0.600, 1, 1)$	$(p_{32}, q_{32}, r_{32}, s_{32}) = (0, 0, 0.170, 0.429)$	$(p_{33}, q_{33}, r_{33}, s_{33}) = (0.376, 0.559, 1, 1)$	$(p_{34}, q_{34}, r_{34}, s_{34}) = (0.330, 0.415, 0.523, 0.578)$
Rule 4	$(p_{41}, q_{41}, r_{41}, s_{41}) = (0.467, 0.616, 1, 1)$	$(p_{42}, q_{42}, r_{42}, s_{42}) = (0, 0, 0.165, 0.438)$	$(p_{43}, q_{43}, r_{43}, s_{43}) = (0, 0, 0.165, 0.410)$	$(p_{44}, q_{44}, r_{44}, s_{44}) = (0, 0, 0.137, 0.369)$
Rule 5	$(p_{51}, q_{51}, r_{51}, s_{51}) = (0, 0, 0.161, 0.423)$	$(p_{52}, q_{52}, r_{52}, s_{52}) = (0.386, 0.576, 1, 1)$	$(p_{53}, q_{53}, r_{53}, s_{53}) = (0.395, 0.561, 1, 1)$	$(p_{54}, q_{54}, r_{54}, s_{54}) = (0.502, 0.643, 1, 1)$
Rule 6	$(p_{61}, q_{61}, r_{61}, s_{61}) = (0, 0, 0.176, 0.430)$	$(p_{62}, q_{62}, r_{62}, s_{62}) = (0.400, 0.592, 1, 1)$	$(p_{63}, q_{63}, r_{63}, s_{63}) = (0, 0, 0.206, 0.447)$	$(p_{64}, q_{64}, r_{64}, s_{64}) = (0.327, 0.406, 0.510, 0.593)$
Rule 7	$(p_{71}, q_{71}, r_{71}, s_{71}) = (0, 0, 0.176, 0.408)$	$(p_{72}, q_{72}, r_{72}, s_{72}) = (0, 0, 0.193, 0.436)$	$(p_{73}, q_{73}, r_{73}, s_{73}) = (0.383, 0.576, 1, 1)$	$(p_{74}, q_{74}, r_{74}, s_{74}) = (0.328, 0.411, 0.513, 0.605)$
Rule 8	$(p_{81}, q_{81}, r_{81}, s_{81}) = (0, 0, 0.168, 0.396)$	$(p_{82}, q_{82}, r_{82}, s_{82}) = (0, 0, 0.186, 0.437)$	$(p_{83}, q_{83}, r_{83}, s_{83}) = (0, 0, 0.196, 0.448)$	$(p_{84}, q_{84}, r_{84}, s_{84}) = (0, 0, 0.093, 0.196)$

Table 3. The training phase of the PSO algorithm in the experiments

the number of particles	50
the dimension of a particle	$128 + 1 (\delta)$
the range of movement for particles	$[0, 1]$
the number of generations	60
the acceleration coefficients c_1 and c_2	random number in $[0, 1]$
two randomly generated values r_1 and r_2	random number in $[0, 1]$
the number of parameters of the FIS	129
the final decision threshold	0.617

4.4 Experimental results

Here, the trapezoid-shaped functions are employed as the membership functions in eight fuzzy

rules [5]. The calculation of FIS in the SRCPFP method is performed with MATLAB tool box. The FIS takes Mamdani's minimum inference method as the fuzzy inference engine [5]. In order to obtain better recognition results, the PSO algorithm is utilized to train the parameters in these membership functions for near optimization. A training set is constructed with 3000 different cloths. Through the PSO algorithm training, the parameters of membership functions in eight fuzzy rules will be nearly optimized.

Table 4 depicts that the SRCPFP method is effective, flexible, near optimal, and easily implemented by readers. The FIS can easily reflect the human concept of expertise knowledge while developing a recognition method [5]. In addition, the PSO algorithm is good for solving optimization problems [3]. It can nearly optimize the parameters of membership functions in the FIS and sufficiently obtain the better results in recognition or similarity. Thus, the SRCPFP method can more accurately recognize cloth products.

Table 4. The SRCPFP method compares with four existing methods

Items Compared	Methods	Pal and Singh 2010	Singh et al. 2009	Khanale and Chitnis 2011	Chang et al. 2008	SPCPFP
Recognition		Yes	Yes	Yes	Yes	Yes
Linguistic Variables		No	No	No	Yes	Yes
Fuzzy Inference System		No	No	No	Yes	Yes
Design using Human Knowledge Concept		No	No	No	Yes	Yes
Near Optimization		No	No	No	No	Yes

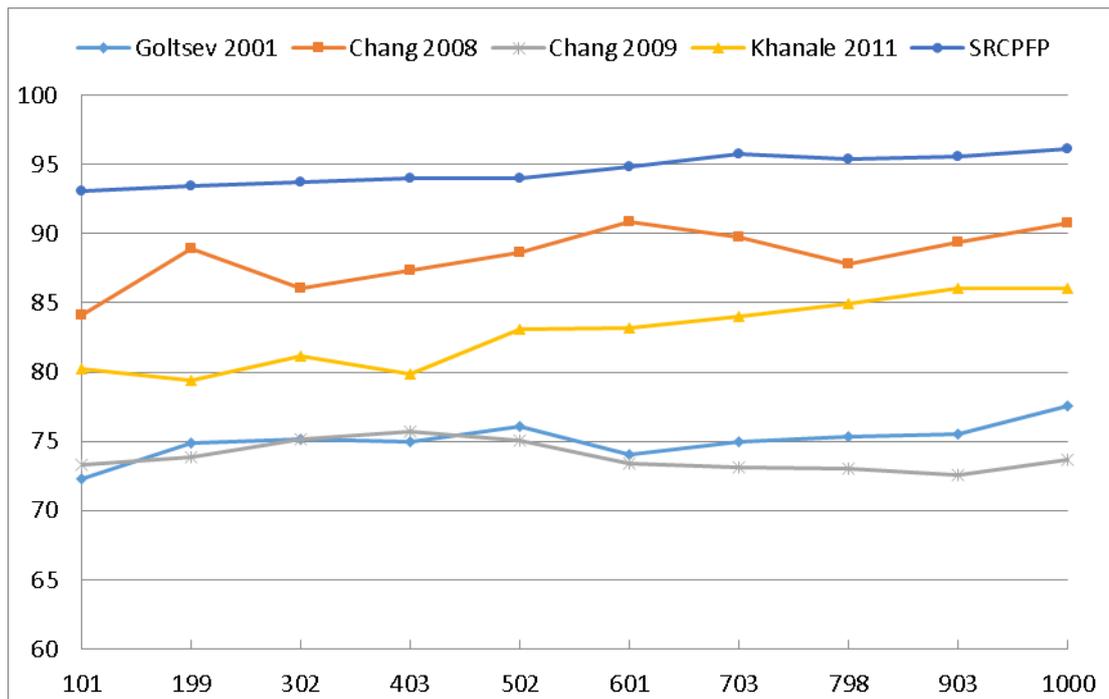
In the experiments, a query cloth set \diamond is constructed with 1000 cloths, which is from **D**. The author randomly builds ten subsets \diamond_i of \diamond , where $\diamond = \cup \diamond_i, i=1,2,\dots,10$. Each cardinality of $\diamond_i, |\diamond_i|$, is given in the first and the third columns of **Table 5**. These subsets are separately utilized to evaluate the SRCPFP method.

In order to evaluate the SRCPFP method, the author performs the accurate rate AR over \mathbf{D}^{te} (*i.e.* \diamond). Observing **Table 5**, the more the number of query cloths chosen, the more accurate the similarity for the SRCPFP method. Employing the FIS integrated with the PSO algorithm is easier and more effective ways in designing a recognition method, the SRCPFP method can more precisely evaluate fuzzy similarity for cloth recognition. **Table 5** demonstrates that the SRCPFP method definitely achieves satisfying results in ten given cases.

Also, the SRCPFP method is further evaluated by comparing four existing well-known methods over ten subsets \diamond_i of \diamond . **Fig. 5** depicts that the SRCPFP method also outperforms four existing methods in ten given cases.

Table 5. The accurate rate AR (%) for the SRCPFP method

# of query cloths randomly selected	Rate	method	SRCPFP method	# of query cloths randomly selected	Rate	method	SRCPFP method
$ s_1 = 101$			93.07	$ s_6 = 601$			94.85
$ s_2 = 199$			93.47	$ s_7 = 703$			95.73
$ s_3 = 302$			93.71	$ s_8 = 798$			95.36
$ s_4 = 403$			94.05	$ s_9 = 903$			95.57
$ s_5 = 502$			94.02	$ s_{10} = 1000$			96.10

**Fig. 5.** The comparison in terms of accurate rate AR (%) for various methods

In order to make a comparison, the SRCPFP method is also evaluated over another related dataset which is arbitrarily collected and built by the author. In the experiments, the query cloth dataset contains 1000 different cloths. Ten subsets are randomly built from the cloth dataset as mentioned above. Further, the SRCPFP method is assessed by comparing four existing well-known methods over ten subsets, respectively. **Fig. 6** draws that the SRCPFP method still outperforms four existing methods in ten given subsets because of its performing

optimization.

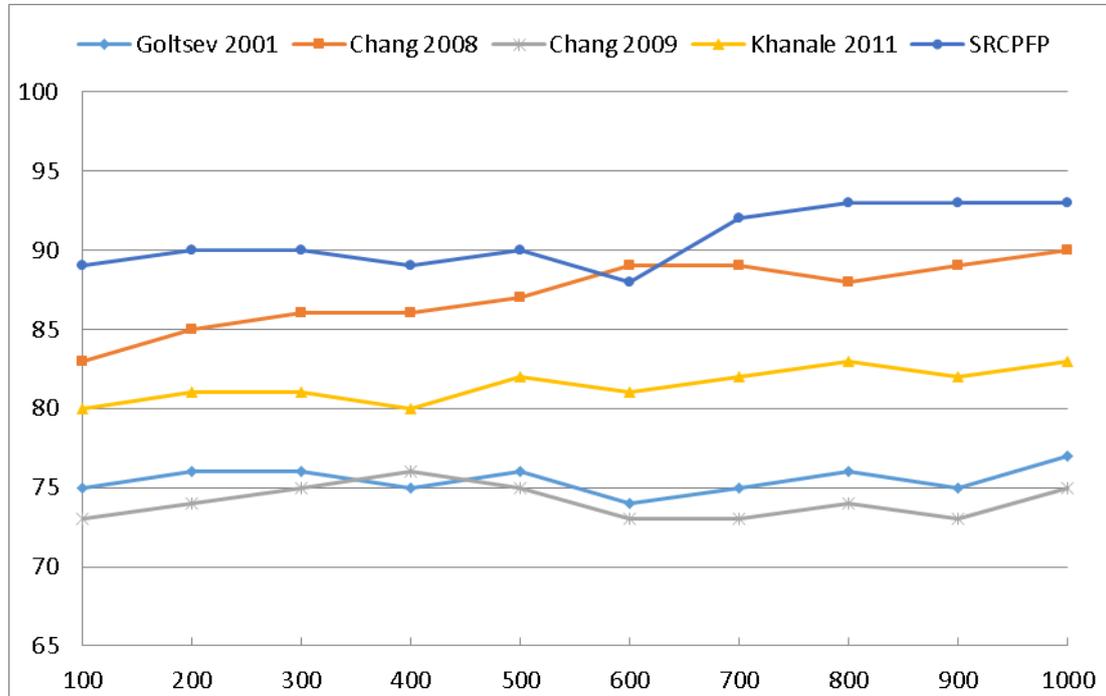


Fig. 6. The comparison in terms of AR (%) for various methods over another related dataset

5. Conclusion

This paper has presented a new method for solving similarity recognition, called the SRCPPF method. The method combines the FL and the PSO algorithm to solve the similarity recognition for cloth products. It creates three features, *length*, *thickness*, and *temperature resistance*, for each cloth product, and then, these three features are employed to build the FIS to effectively find out similarity for each query cloth. Meanwhile, the FIS integrated with the PSO algorithm can powerfully search for the near optimal parameters of membership functions in the fuzzy rules, for the similarities between each query cloth and sampling cloths in the cloth database **D**. Nowadays, many similarity problems can be solved by an FIS optimized by the PSO algorithm. In this paper, the PSO algorithm is utilized to nearly optimize 128 parameters and the decision threshold δ . Experimental results demonstrate that the SRCPPF method achieves the satisfying recognition performance and outperforms other well-known methods for similarity recognition under considerations here, and applies in most of similarity applications.

Acknowledgments

The author gratefully acknowledges the valuable comments and suggestions of the anonymous reviewers, which have improved the presentation.

References

- [1] K.Y. Bao, X.Y. Ma and J.M. Wei, "Prolong life-span of WSN using clustering method via swarm intelligence and dynamical threshold control scheme," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 6, pp. 2504-2526, June 2016. [Article \(CrossRef Link\)](#)
- [2] H.S. Behera, P.K. Dash and B. Biswal, "Power quality time series data mining using S-transform and fuzzy expert system," *Applied Soft Computing*, vol. 10, pp. 945-955, 2010. [Article \(CrossRef Link\)](#)
- [3] F.V.D. Bergh and A.P. Engelbrecht, "A convergence proof for the particle swarm optimiser," *Fundamenta Informaticae*, vol. 105, no. 4, pp. 341-374, 2010. [Article \(CrossRef Link\)](#)
- [4] A. Biem, "Minimum classification error training for online handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1041-1051, 2006. [Article \(CrossRef Link\)](#)
- [5] B.-M. Chang, H.-H. Tsai and P.-T. Yu, "Handwritten character recognition using a neuro-fuzzy system," *International Journal of Innovative Computing, Information and Control*, vol. 4, no. 9, pp. 2345-2362, September 2008. [Article \(CrossRef Link\)](#)
- [6] B.-M. Chang, H.-H. Tsai and P.-T. Yu, "The Dempster-Shafer theory combined with neural network in handwritten character recognition," *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 9, pp. 2561-2573, September 2009. [Article \(CrossRef Link\)](#)
- [7] C.-H. Chen, M.-T. Su, C.-J. Lin and C.-T. Lin, "A hybrid of bacterial foraging optimization and particle swarm optimization for evolutionary neural fuzzy classifier," *International Journal of Fuzzy Systems*, vol. 16, no. 3, pp. 422-433, September 2014. [Article \(CrossRef Link\)](#)
- [8] B. Cyganek, "Hybrid ensemble of classifiers for logo and trademark symbols recognition," *Soft Computing*, vol. 19, issue 12, pp. 3413-3430, Dec. 2015. [Article \(CrossRef Link\)](#)
- [9] A. Goltsev and D. Rachkovskij, "A recurrent neural network for partitioning of hand drawn characters into strokes of different orientations," *International Journal of Neural Systems*, vol.11, no.5, pp.463-475, 2001. [Article \(CrossRef Link\)](#)
- [10] G. Katiyar and S. Mehfuz, "A hybrid recognition system for off-line handwritten characters," *SpringerPlus*, 5:357, 2016. [Article \(CrossRef Link\)](#)
- [11] P.B. Khanale and S.D. Chitnis, "Handwritten Devanagari character recognition using artificial neural network," *Journal of Artificial Intelligence*, vol. 4, no. 1, pp. 55-62, 2011. [Article \(CrossRef Link\)](#)
- [12] J. Kumar, P. Ye and D. Doermann, "Structural similarity for document image classification and retrieval," *Pattern Recognition Letters*, vol. 43, pp. 119-126, 2014. [Article \(CrossRef Link\)](#)
- [13] C.-H. Lee, F.-Y. Chang and C.-M. Lin, "DSP-based optical character recognition system using interval type-2 neural fuzzy system," *International Journal of Fuzzy Systems*, vol. 16, no. 1, pp. 86-96, March 2014. [Article \(CrossRef Link\)](#)
- [14] C.-T. Lin, C.-M. Yeh, S.-F. Liang, J.-F. Chung and N. Kumar, "Support-vector-based fuzzy neural network for pattern classification," *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 1, pp. 31-41, 2006. [Article \(CrossRef Link\)](#)
- [15] L.M. Lorigo and V. Govindaraju, "Offline Arabic handwriting recognition: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 5, pp. 712-724, 2006. [Article \(CrossRef Link\)](#)
- [16] S. Marinai, E. Marino and G. Soda, "Font adaptive word indexing of modern printed documents," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1187-1199, 2006. [Article \(CrossRef Link\)](#)
- [17] A. Pal and D. Singh, "Handwritten English character recognition using neural network," *International Journal of Computer Science & Communication*, vol. 1, no. 2, pp. 141-144, 2010. [Article \(CrossRef Link\)](#)
- [18] Z.I. Petrou, V. Kosmidou, I. Manakos, T. Stathaki, M. Adamo, C. Tarantino, V. Tomaselli, P. Blonda and M. Petrou, "A rule-based classification methodology to handle uncertainty in habitat mapping employing evidential reasoning and fuzzy logic," *Pattern Recognition Letters*, vol. 48, pp. 24-33, October 2014. [Article \(CrossRef Link\)](#)

- [19] D. Singh, M. Dutta and S. H. Singh, "Neural network based handwritten Hindi character recognition system," in *Proc. of the 2nd Bangalore Annual Compute Conference*, Article 15, 2009. [Article \(CrossRef Link\)](#)
- [20] R. Sivanesan, A. Anwar, A. Talwar, R. Menaka and R. Karthik "A novel scheme for detection of Parkinson's disorder from hand-eye coordination behavior and DaTscan images," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 9, pp. 4367-4385, September 2016. [Article \(CrossRef Link\)](#)
- [21] T. Steinherz, E. Rivlin, N. Intrator and P. Neskovic, "An integration of online and pseudo-online information for cursive word recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 669-683, 2005. [Article \(CrossRef Link\)](#)
- [22] Y. Wang, J. Tang, Z. Pan and C. Yan, "Particle swarm optimization-based planning and scheduling for a laminar-flow operating room with downstream resources," *Soft Computing*, vol. 19, issue 10, pp. 2913-2926, October 2015. [Article \(CrossRef Link\)](#)
- [23] P.-S. You, Y.-C. Hsieh and C.-M. Huang, "A particle swarm optimization based algorithm to the internet subscription problem," *Expert Systems with Applications*, vol. 36, issue 3, pp. 7093-7098, April 2009. [Article \(CrossRef Link\)](#)
- [24] Q. Zou, Y. Cao, Q. Li, C. Huang and S. Wang, "Chronological classification of ancient paintings using appearance and shape features," *Pattern Recognition Letters*, vol. 49, pp. 146-154, November 2014. [Article \(CrossRef Link\)](#)
- [25] X.Y. Zhang, S.Q. Hu, H.L. Zhang and X. Hu, "A real-time multiple vehicle tracking method for traffic congestion identification," *KSII Transactions on Internet and Information Systems*, vol. 10, no.6, pp. 2483-2503, June 2016. [Article \(CrossRef Link\)](#)



Bae-Muu Chang received the BS degree in computer science from Tamkang University, Taipei, Taiwan, in 1977, the MS degree in computer science from New York Institute of Technology, New York, USA, in 1989, and the PhD degree in computer science and information engineering from National Chung Cheng University, Chiayi, Taiwan, in 2009. Since 1990, he has been with the Department of Information Management at Chienkuo Technology University, Changhua, Taiwan, where he is currently an associate professor. His research interests include neural networks, fuzzy systems, character recognition, intelligent filter design, intelligent networks, image processing, and e-Learning.