Chain Code Extraction of Handwritten Recognition using Particle Swarm Optimization

Dewi Nasien, Habibollah Haron, Haswadi Hasan, Siti S. Yuhaniz
Soft Computing Research Group
Faculty of Computer Science and Information System
Universiti Teknologi Malaysia
81310 UTM Skudai
Johor Bahru, Malaysia
dwien82@gmail.com,{habib,haswadi,sophia}@utm.my

ABSTRACT*: As one of soft computing optimization tools, Particle Swarm Optimization (PSO) has been applied in many fields such as in handwritten recognition and classification. Associated with the development of PSO algorithm is the data representation to be used as input to the algorithm. One of data representation scheme is Freeman chain code (FCC). As one of traditional scheme in representing data, FCC is still relevant as a scheme in data compression and representation. The scheme is proposed in this paper to represent Latin handwritten as isolated character and as input to the PSO algorithm. The main problem related to FCC and handwritten recognition to be solved in this paper is in representing and recognizing character because the length of the FCC depends on the starting points. Therefore, every pixel (or node) must be coded with a specific number depending on its direction. In addition to the isolated characters especially the upper-case characters, the traversing process of each pixel (or node) of this type is more difficult because of the problem in finding several branches and revisiting the same nodes. As a solution, one continuous route is proposed to solve the problems which cover all of the nodes in the character image. The proposed PSO algorithms extract the FCC from the thinned binary image of the character and recognize difficult characters based on the series of FCC. Two performance parameters used to measure the performance of the PSO algorithm are the computational time and route length. Result shows that the PSO algorithms successfully extract FCC and recognize character at a relatively shorter computational time and route length.*

Keywords: Freeman chain code, Particle swarm optimization, Latin handwritten recognition

Received: 14 Septembr 2010, Revised 18 October 2010, Accepted 27 October 2010

1. Introduction

In general, handwritten recognition system consists of three phases that is pre-processing, feature extraction, and classification. The success rate of the system does not only depend on its classification system but also includes the entire phases. Therefore, it is obvious to take careful design steps for all of the three phases, including feature extraction. In feature extraction, input of the phase plays important role in determining the successful of extraction.

An automated character recognition system is a solution that can interpret characters automatically. The automatic recognition of characters can be extremely useful where it is necessary to process large volume of handwritten characters. It has been a subject of research for more than 40 years [1]. The research in Off-line Character Recognition (OCR) has started in the early 1960s. Handwriting character recognition (HCR) is a challenging problem since there is a variation of same characters due to its fonts and sizes. They make recognition task more difficult and recognition of character is not accurate. Moreover, the achievements by the researchers are differ for different special case for every different databases used. Different databases make the handling differ in the solution of the recognition characters because of variation and complexity of data in the database.

PSO is a population based stochastic optimization technique developed in 1995, inspired by social behavior of bird flocking or fish schooling [2]. The system is initialized with a population of random solutions and searches for optima by updating generations. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. It has been widely used in optimizing many fields including recognition of handwritten characters. It also accepts many type of data representation as input the techniques such as chain code and mathematical representation.

Chain code has been widely used to encode the boundary line for its simplicity and low storage requirement [3-4]. Chain code representation gives boundary of character image where the codes represent the direction of where is the location of the next pixel, and have a connection to a starting point. Isolated character, especially upper-case on Latin characters usually has branches on every nodes character, and this give difficulty to decide where it should go. Moreover, a revisit to the previous visited node is often needed to visit all of the nodes. These difficulties motivate the use of population-based meta-heuristics. Freeman chain code (FCC) is selected as representation and identification of image characters because it is easy for isolated character based on outer boundary representation. The term of isolated characters is an individual (single) character. In other word, isolating a character is important before recognizing it.

In chain code representation, one continuous route that covers all of the nodes of the image is recommended to solve such problems. Thus, a character recognition method via population-based meta-heuristics methods namely Particle Swarm Optimization (PSO) is proposed to generate FCC which executes extraction and recognition simultaneously. Knowing that the use of population-based meta-heuristics methods to construct FCC is not widely explored and the existence of length problem in representing and recognition characters of FCC, they motivate this research. This method enables us to extract and recognize such difficult characters. The performance parameters to evaluate the performance of the algorithm are computational time and route length.

This paper is divided into six sections. Introduction to important terms related to the contents and brief explanation on the contents of this paper is given in this chapter. Related work and problems in HCR, relation of PSO and FCC with HCR, and the importance of this work to HCR are given in Section 2. Section 3 gives the methodology of the work that consists of five steps. Based on the methodology, detail explanation on development of the PSO algorithm is explained in Section 4. Result and analysis are given in Section 5 and followed by conclusion and future work in Section 6.

2. Related Work

Chain code is an object related data structure for representing the boundary of a binary object on a discrete grid [5]. The first approach of chain code was introduced by Freeman in 1961 that is known as Freeman chain code [6]. Chain code representation gives boundary of character image where the codes represent the direction of where is the location of the next pixel and the connections to the starting point. There are two kinds of directions FCC, namely 4-neighborhood and 8- neighborhood. This paper utilized 8-neighbourhood in extraction of characters which starts from 1 until 8. The code is different from common FCC that uses 0 until 7 because it is easy to distinguish direction or non direction.

Chain code representation has been used in several HCR problems. A fast handwritten word recognition system for real time applications is presented where the pre-processing, segmentation and feature extraction are implemented using chain code representation [7]. A new type of generalized chain code (GCC) [8] is proposed for lossless encoding of handwriting data captured by a tablet and real-time recognition of isolated on-line handwritten characters. Discussion on the issues of determination of upper and lower contours of the word, determination of significant local extrema on the contour and determination of reference lanes from contour representations of handwritten words can be referred in [9]. In [10], it proposes a recognition method based on handwriting acceleration, line crossing point segmentation, macrostructures (isolated traces), and chain coding and time-frequency analysis. Meanwhile, FCC is used to discriminate sets of feature vectors in a multi-font Bengali characters recognition system [11]. A novel fast method for line segment extraction based on chain code representation of thinned sketches (or edge maps) is presented and exploited for Persian signature recognition [12]. A robust and effective of an HCR system based on modified chain code of Persian and Arabic alphabets is proposed in [13]. In [14], a HCR system is proposed by finding the contour of a handwriting image and a set of chain code from global and local histogram. Soft computing methods have been shown to be effective in solving HCR problem using chain code approaches as in [15]. A recognition model for English handwritten (lowercase, uppercase and letter) character recognition that uses FCC as the representation technique of an image character is proposed in [16]. A new scale invariant optimized chain code for Nastaliq character representation is presented in [17]. This previous works shows that chain code representation is useful in HCR.

The PSO methods discussed will be used to generate the continuous FCC from thinned binary image (TBI) which will act as the image feature. The encouragement of them is motivated by following considerations:

1. The starting node for the FCC construction influences its length. In addition, a handwritten character often has several branches and this make difficult to decide where it should go on direction. Moreover, a revisit to the previous visited nodes is often needed to visit all of the nodes. These difficulties in handling several branches have motivated the use of soft computing methods.

2. The performance of the recognition phase depends on the data input provided from the previous phases. The input data for the recognition phase (in this case is the FCC) must correctly represent and distinguish each character. Since a handwritten character can be represented with several FCCs, the methods must have ability to provide such results. In order to achieve this, population-based meta-heuristics are used to generate the FCC.

PSO is one of famous swarm inspired method in computational intelligence based on simulation of social system. PSO simulates the behavior of bird flocking and can be used as an optimizer. The PSO provides a population-based search procedure in which individuals are called particles. All of particles have fitness values which are evaluated by the fitness function to be optimized and velocities which direct the flying of the particles. Particles velocities are adjusted according to historical behavior of each particle and its neighbors while they fly through search space. Therefore, particles have a tendency to fly towards the better search area over the course of search process [18].

PSO approaches also have been used in several HCR and FCC application. The Chinese word segmentation based on the improved Particle Swarm Optimization (PSO) neural networks is implemented as in [19]. A solution is obtained by searching globally using FPSO (Fuzzy cluster Particle Swarm Optimization) algorithm, which has strong parallel searching ability, encoding real number, and optimizing the training weights, thresholds, and structure of neural networks. Then based on the optimization results obtained from FPSO algorithm, the optimization solution is continuously searched by following BP algorithm, which has strong local searching ability, until it is discovered finally. In [20], PSO-based method is exploited to recognize unconstrained handwritten digits. Each class is encoded as a centroid in multidimensional feature space and PSO is employed to probe the optimal position for each centroid. PSO also has been used successfully in image watermarking, data clustering, character recognition, image thresholding and recognition [21-23].

Although PSO has been used widely in image processing, unfortunately the study about FCC extraction using these methods has not been widely proposed. Motivated from current situation, this paper proposed the PSO in generating such FCCs from a thinned binary image of handwritten character as a continuous path with the shortest length.

3. Research Methodology

This section explains the methodology of all stages in the framework applied in this paper. A proper methodology must be carried out before the implementation of this paper in order to achieve the objectives of this research that have been defined. There are five stages in the framework that are the thinning process to produce thinned binary image (TBI) of the character, problem formulation, character transformation into graph, solution representation, and calculating the objective function.

3.1 Thinning Process in Pre-Processing Stage

In pre-processing stage, thinning is an important step in HCR. The purpose of thinning is to delete the redundant information and at the same time retain the characteristic features of the character. Thinning is applied to find a skeleton of a character. The example of skeleton and its original image of character are as shown in Figure 1.

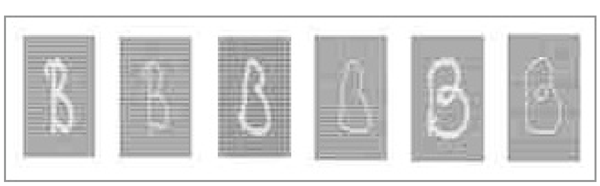

Figure 1. Original Image and Skeleton

The development of thinning algorithms proposed by [24] that apply neural network approach has been used in this research to produce the TBI. The output of the pre-processing stage is then used in the feature extraction to extract the character.

Figure 2 shows the example of the TBI obtained using the proposed thinning algorithm [20]. The TBI that is presented as a binary image is used to generate the FCC by the proposed PSO algorithm. Binary image is a representation with only two possible gray values for each pixel, such as “0” and “1”. Background is represented by “0” and foreground is represented by “1”.

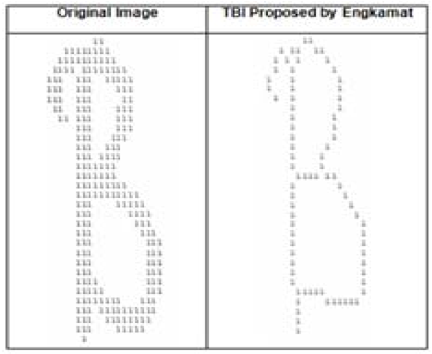

Figure 2. “B1” character by Engkamat [20]

3.2 Problem Formulation

The problem of formulation approaches used in this paper consists of the following steps to generate the FCC which is character transformation into graph a solution representation and calculating the objective function.

3.3 Character Transformation into Graph

A generation of FCC from a binary character can be modeled as a route of a graph problem. Therefore, the image is transformed into a set of vertices and connecting edges. The vertices of the graph is taken from the node that has only one neighbor and the node that has more than two neighbors and these vertices identification is represented as red circle and blue circle respectively as shown in Figure 3. In addition, the edges of the graph are indicated by the edge that comes from the nodes which have two neighbors connecting the vertices from previous neighbor. The complete graph can be seen in Figure 4.

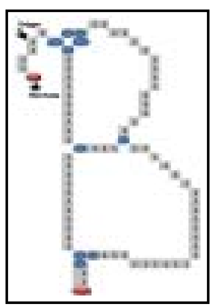

Figure 3. Vertices identification

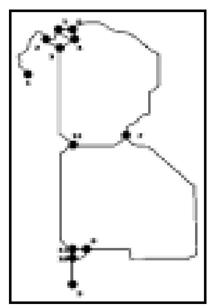

Figure 4. Character transformation

The solution of representation is presented by directed graph (digraph) because specifying the starting of FCC vertex is necessary. Therefore, Table 1 provides all the edges in directed graph with theirs lengths. The lengths are obtained from the total number of nodes between two vertices. It should be noted that the edge length of 0 means there is no node between that particular vertices.

3.4 Solution Representation

The proposed PSO use a sequence of edges to represent the FCC solutions. The edge is used (not vertex) as the solution representation to avoid misidentification in two conditions. First condition is an edge that is derived and ended from the same node. Second condition is two different edges can be derived from the same node and also ended in another same node. Thus, one edge can be visited twice and as a result the solution representation can have a complete tour since a revisit to the previous visited nodes is often needed. Figure 5 shows an example of the solution representation while Table 2 shows the edge and their length based on Figure 5.

Based on Table 2 every edge uses a sequence of edge to represent the FCC. The sequence of edges is checked whether it is continuously connected to the previous edge in the sequence. For instance: for edge 26, start vertex is 9 and end vertex is 10 and it continue to edge 22 where the start vertex is 10 and end vertex is 8

26 22 20 18 14 11 6 2 1 3

9 30 24 20 32 10 4 2 1 5

12 13 17 32 30 27 25 26 22 20

18 14 11 6 2 1 3 9 30

Figure 5. An example of solution representation

Edge	Start Vertex	End Vertex	Length	Edge	Start Vertex	End Vertex	Length
1	1	2	5	17	6	7	0
2	2	1	5	18	7	6	15
3	2	3	0	19	7	8	15
4	3	2	0	20	8	7	23
5	2	5	0	21	8	10	23
6	5	2	0	22	10	8	0
7	3	4	0	23	8	11	0
8	4	3	0	24	11	8	0
9	3	12	10	25	10	9	2
10	12	3	10	26	9	10	2
11	4	5	0	27	11	10	0
12	5	4	0	28	10	11	0
13	4	6	0	29	11	12	11
14	6	4	0	30	12	11	11
15	5	6	9	31	12	7	3
16	6	5	0	32	7	12	3

Table 1. Edges and their length generated

3.5 Objective Function

The role of population based meta-heuristic is to minimize the objective function of the solution representation. The objective function is to express the quality of FCC solution. The objective function is defined as the number of nodes which the FCC must visit from the starting node until all of the nodes are visited. The visit is (revisit is counted too) starting from the first edge until an edge where all of the nodes are visited.

While calculating the objective function of a solution representation, it is repaired to meet its feasibility issue. The feasibility issue of the represented solution representation is whether the sequence of the edges is continuously connected until all of the nodes are visited. In order to achieve the continuous connection, the edge which is not connected to the previous edge is swap by another closest feasible edge in the sequence. It should be noted that while traversing from one edge to next edge, a repair process is forced so as it meet the feasibility issue. If the repair process is unsuccessful, objective function is assigned as equal to 2 multiply with the number of the nodes.

For the solution representation in Figure 4, the walk only uses the following sequence: (26 22 20 18 14 11 6 2 1 3 9 30 24) since all of the nodes are already visited with only that sequence. The value of objective function for this sequence is (2 + 0 + 23 + 15 + 0 + 0 + 0 + 5 + 5 + 0 + 10 + 11 + 0) + 13 = 84. Equation 1 shows the formula of objective function.

$$ObjectiveFunction = Length + NumberOfSequenceOccurence \qquad (1)$$

Edges	Start Vertex	End Vertex	Length		Edges	Start Vertex	End Vertex	Length
26	9	10	2		12	5	4	0
22	10	8	0		13	4	6	0
20	8	7	23		17	6	7	0
18	7	6	15		32	7	12	3
14	6	4	0		30	12	11	11
11	4	5	0		27	11	10	0
6	5	2	0		25	10	9	2
2	2	1	5		26	9	10	2
1	1	2	5		22	10	8	0
3	3	3	0		20	8	7	23
9	3	12	10		18	7	6	15
30	12	11	11		14	6	4	0
24	11	8	0		11	4	5	0
20	8	7	23		6	5	2	0
32	7	12	3		2	2	1	5
10	12	3	10		1	1	2	5
4	3	2	0		3	2	3	0
2	2	1	5		9	3	12	10
1	1	2	5		30	12	11	11
5	2	5	0					

Table 2. Edges and their length generated

4. Implementation

4.1 Parameter Values

Setting the parameter values and input data is determined based on literature review and trial and error. The input character is isolated handwritten on upper-case Latin characters (A-Z). The database used is Center of Excellence for Document Analysis and Recognition (CEDAR) which consists of 126 characters as shown in Table 3. The input of original CEDAR image is 50x50 pixels and output of TBI is same size.

Character	The total number for every character
Q	3 x 1
I, P	4 x 2
A - H J - O R - Z	5 x 23
Total	126 characters

Table 3. The detail of image properties for dataset

Parameter values used in the proposed PSO are as shown in the Table 4. PSO was developed to optimize real (float) parameters of a real valued function. The number of iterations in the proposed PSO is associated with local search procedures. In addition, it is important to select the best solution which will survive for the next iteration and to select the basis of number parameter selection of the proposed algorithm using pilot study.

Parameter	Value
Maximum number of iterations	200
Number of population (n)	100
Inertia weight (I)	0.2
Relative weight of local best (C_1)	0.4
Relative weight of global best (C_2)	1.6
Number of local search	10

Table 4. The PSO Parameter Value

4.2 The Particle Swarm Optimization Algorithm

Particle Swarm Optimization (PSO) uses collaboration among particles to explore the solution space. It updates its particle position using information from previous particle position, local best position, and global best position. The implementation of PSO to generate the FCC is depicted in Table 5.

```
Input data and setting parameter values
Generate random initial population
Repeat
    Generate trial population
        Calculate next velocity
        Calculate next particle position
        Apply SPV rule
        Apply local search
    Update local best and global best
Until stopping criterion is achieved
```

Table 5. The pseudocode of PSO

Since the input for the PSO algorithm is a binary image as shown in Figure 4, several parameter values must be determined. The algorithm starts with initial population generated randomly. Then new position particle is calculated using information from previous particle position, local best position, and global best position. The proposed PSO algorithm is to keep record about the best solutions found and they are used as the local best position. The algorithm stops with predetermined number of iterations.

First of all, the initialization of the proposed PSO uses the following parameters: maximum number of iterations, number of population (n), inertia weight (I), relative weight of local best (C_1), relative weight of global best (C_2), and number of local search. In addition, the input binary image is processed to generate more efficient data structure to be used in the main PSO iterations. Initial population is randomly generated. The population consists of n solution which takes real-value representation which will be converted to discrete representation using smallest possible value (SPV) rule. After that, the objective function for every solution in the population is calculated based on procedure explained in Section 3. The methods for calculating the next velocity vector and next particle position are shown in Equation 2 and 3. Then the discrete solution is constructed from the real-value representation using SPV rule.

$$Velocity(k+1) = Velocity(k)+I*P(k)+(C_1*rand*P_{lbest}(k))+(C_2*rand*P_{gbest}(k)) \quad (2)$$

$$P(k+1)=Velocity(k+1)+P(k) \quad (3)$$

where: *Velocity* (k) is velocity at iteration k, I is inertia weight, $P(k)$ is a particle position at iteration k, *rand* is a random number between 0 and 1, (C_1) is relative weight of local best, (C_2) is relative weight of global best, $P_{lbest}(k)$ is local best solution at iteration k and $P_{gbest}(k)$ is a global best position at iteration k. For the offspring Next, the offspring solution is improved in the local search process. Next, the offspring solution is improved in the local search process. The best particle position found is recorded as the global best particle. In addition, a collection of best n particles is recorded as the local best particles.

5. Result and Analysis

The experiment consists of 10 replications where each replication contains 100 FCC solution for every TBI. The performance measures of the algorithms are the route length and computation time. Table 6 shows the route length and computation time for B characters ("B1-B5") of four algorithms with 10 replications.

Characters	Route Length (chain code length)									
	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
B1	91.06	90.93	90.59	90.55	90.47	92.47	91.27	94.22	90.49	90.88
B2	72.00	72.00	72.00	72.00	72.00	72.00	72.00	72.00	72.00	72.00
B3	104.24	104.24	104.24	104.2	104.2	104.24	104.24	104.24	104.24	103.62
B4	70.52	70.52	70.52	70.52	70.52	70.52	70.52	70.52	70.52	70.52
B5	174.00	173.95	173.57	174.00	174.00	174.00	174.00	173.93	173.39	173.86
Characters	**Computation Time (seconds)**									
	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
B1	92.09	94.13	96.03	97.28	92.27	100.40	99.07	94.83	99.26	95.70
B2	1.00	1.01	1.02	1.00	1.00	1.02	1.00	1.00	1.01	1.02
B3	49.13	48.45	50.19	50.48	47.89	45.75	45.17	46.55	47.87	45.86
B4	36.41	35.97	36.19	35.93	35.69	35.57	36.01	38.37	37.66	37.08
B5	159.90	150.19	140.64	122.87	110.34	110.74	113.32	106.22	124.91	115.84

Table 6. The route length and computation time for B characters ("B1-B5")

After all B characters ("B1-B5) are finished in term of route length and computation time, so the next step is to classify into 3 categories: best, average and worst. Best is the minimum value group of number of iterations in a set of values. Average is calculated by adding a group of number iterations and then divided by the count numbers of every character. Worst is the maximum value a group of number of iteration in a set of values. On the other hand, the computation time categories are average and total. Average is calculated by adding a group of number iterations and then divided by the count numbers of every character. Total is the sum of group of number iterations in a set of values.

6. Conclusion and Future Work

The proposed PSO algorithm in generating FCC of handwritten character image is used to produce one continuous route and to minimize the length of FCC. It is implemented by using their particular characteristics to find a collection of good FCC solutions which minimize the FCC length and is recorded as the global best particle. Meanwhile, a collection of best n particles is recorded as the local best particles

In FCC, the route length and computation time are selected in this experiment because they depend on the starting point that automatically affected on the route length and number of times needed to solve the chain code. This method enables us to extract and recognize such difficult characters in relatively shorter computational time and route length. For future work, the proposed algorithm can be compared with other meta-heuristic algorithm so that its performances can be evaluated.

7. Acknowledgment

The authors honorably appreciate Ministry of Science, Technology and Innovation (MOSTI) for the eScience Fund and Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM) for the support in making this projects success, also to Komarudin in assisting the development and testing of the algorithm.

References

[1] Huang, Y, S., Suen, C, Y., (1995). A method of combining multiple experts for the recognition of unconstrained handwritten numerals. *Journal of Pattern Analysis and Machine Intelligence* 17 (1) 90-94.

[2] Kennedy, J., Eberhart, R (1995). Particle swarm optimization. *In:* Proc. of the 4th International Conference on Neural Network (ICNN'95), p. 1942-1948. IEEE.

[3] Shaojie, Z., Kai-Kuang, M. (2000). A novel shape matching method using biological sequence dynamic alignment. *In:* Proc. of International Conference on Multimedia and Expo I (ICME 2000), p. 343-346. IEEE.

[4] Neuhoff, D., Castor, K.(1985). A rate and distortion analysis of chain codes for line drawings. *Journal of Information Theory* 31 (1) 53-68.

[5] Jahne, B (1991). Digital image processing: concepts, algorithms and scientific applications. Berlin : Springer-Verlag.

[6] Freeman, H. (1961). Techniques for the digital computer analysis of chain-encoded arbitrary plane curves. *In: Proc. of the 17th National Election*, p. 312-324.

[7] Gyeonghwan, K., Govindaraju, V.(1995). Handwritten word recognition for real-time applications. *In:* Proc. of the 1st International Conference Document Analysis and Recognition (ICDAR 95), p. 24-27. IEEE.

[8] Hung, Y. (1996). A chain coding approach for real-time recognition of on-line handwritten characters. *In: P*roc. of the 6th International Conference Acoustics, Speech, and Signal Processing (ICASSP'96), p. 3426-3429. IEEE.

[9] Madhvanath, S., Govindaraju, V., (1997). Contour-based image preprocessing for holistic handwritten word recognition. *In: Proc. of the 2nd International Conference Document Analysis and Recognition (ICDAR'97)*, p.536-539. IEEE.

[10] Martinez, J. C., de J. Lopez, J., Luna Rosas, F, J., (1999). A low-cost system for signature recognition. *In:* Proc of the 42nd Midwest Symposium (MWSCAS'99), p.101-104. IEEE.

[11] Mahmud, J. U., Raihan, M. F., Rahman, C, M (2003). A complete OCR system for continuous Bengali characters. *In:* Proc. of Conference Convergent Technologies for Asia-Pacific Region (TENCON'03), p. 1372-1376.

[12] Chalechale, A., Mertins, A (2003). Line segment distribution of sketches for Persian signature recognition. *In: Pr*oc. of Conference Convergent Technologies for Asia-Pacific Region (TENCON'03), p. 11-15. IEEE.

[13] Alaei, A., Pal, U., Nagabhushan, P.(2009). Using Modified Contour Features and SVM Based Classifier for the Recognition of Persian/Arabic Handwritten Numerals. *In*: Proc. of International Conference Advances in Pattern Recognition (ICAPR'09), p. 391-394. IEEE.

[14] Siddiqi, I., Vincent, N., (2009). A Set of Chain Code Based Features for Writer Recognition. *In:* Proc. of International Confernece Document Analysis and Recognition (ICDAR'09), p.981-985. IEEE.

[15] Hasan, H., Haron, H., Hashim, S, Z. (2009). Freeman Chain Code Extraction Using Differential Evolution (DE) and Particle Swarm Optimization (PSO). *In:* Proc. of International Conference Soft Computing and Pattern Recognition (SOCPAR'09), p.77-81. IEEE.

[16] Nasien, D., Haron, H., Yuhaniz, S, S (2010). Support Vector Machine (SVM) for English Handwritten Character Recognition. *In:* Proc of International Conference Computer Engineering and Applications (ICCEA'10), p. 249-252. IEEE.

[17] Rehman, M, A, U (2010) A New Scale Invariant Optimized Chain Code for Nastaliq Character Representation. *In:* Proc of International Conference Computer Modeling and Simulation (ICCMS'10), p. 400-403. IEEE.

[18] Grosan, C., Abraham, A., Han, S., Gelburh, A (2005). *Hybrid particle swarm – evolutionary algorithm for search and optimization*. Lecture notes in computer science, Springer Berlin/Heidelberg, p. 623-632.

[19] Bao, F., Li, A., Qin, Z., (2004). Photo Time-Stamp Recognition Based on Particle Swarm Optimization. *In:* Proc of International Conference Web Intelligence (WIC'04), p. 529-532. IEEE.

[20] Sahel Ba-Karait, N, O., Shamsuddin, S, M. (2008). Handwritten Digits Recognition Using Particle Swarm Optimization. *In:* Proc. of the 2nd Asia International Conference Modeling and Simulation (AICMS'08), p. 615-619. IEEE.

[21] Kao, I, W., Tsai, C, Y., Wang, Y,C. (2007). An effective particle swarm optimization method for data clustering, *In:* Proc of International Conference Industrial Engineering and Engineering Management (IEEM'07), p.548 – 552. IEEE.

[22] Sarfraz, M., Al-Awami, A,T. (2007). Arabic character recognition using particle swarm optimization with selected and weighted moment invariants, *In*: Proc. of International Symposium Signal Processing and Its Applications, (SSPA'07), p.1 – 4. IEEE.

[23] Zhengchun Lin., Zhiyan W., Yanqing, Z. (2008). Image Thresholding Using Particle Swarm Optimization. *In:* Proc. of International Conference MultiMedia and Information Technology (MMIT'08), p. 245 – 248. IEEE.

[24] Engkamat, A, A, (2005) *Enhancement of Parallel Thinning Algorithm for Handwritten Characters Using Neural Network*. MSc Thesis. Universiti Teknologi Malaysia.

Author biographies

Dewi Nasien was born in Pekanbaru, Indonesia in 1982. She received her degree in the field of Information and Technology at Universitas Islam Negeri SUSQA Pekanbaru, Indonesia in 2005. She obtained an M.Sc in Computer Science at Universiti Teknologi Malaysia, Johor Bahru, Malaysia in 2008. Currently, she is concluding a PhD at the Universiti Teknologi Malaysia, Johor Bahru, Malaysia. Her research interests are in the field of pattern recognition, handwritten character recognition and soft computing.

Habibollah Haron was born in Malacca, Malaysia in 1966. He received his Bachelor in Computer Science from Universiti Teknologi Malaysia in 1989, then Master degree in 1995 from University of Sussex, UK. He obtained his PhD in the area of Computer Aided Geometric Design (CAGD) in the year of 2004 at Universiti Teknologi Malaysia. He has been with Universiti Teknologi Malaysia (UTM) since 1991 as a faculty member, and currently work as Associate Professor since 2008 with UTM. He also has served as Associate Professor at College of Computer and Information Sciences, Imam Mohammad Ibn Saud University, Riyadh from September 2009 till February 2011. His research areas are robotic, computer aided design and soft computing.

Haswadi Hasan was born in Kedah, Malaysia in 1972. He received his Bachelor in Computer Science from Universiti Teknologi Malaysia in 1996, then Master degree in 1999 from University of Salford, UK. He is currently a PhD student in the area of Feature Extraction and Recognition in Universiti Teknologi Malaysia. He has been with Universiti Teknologi Malaysia (UTM) since 1996 as a faculty member, and currently on study leave for PhD study.

Siti Sophiayati Yuhaniz was born in Selangor, Malaysia, 1980. She received her Bachelor degree in the field of Computer Science at Universiti Teknologi Malaysia in 2001, then obtained her PhD in the field of on-board satellite image processing at University of Surrey, UK in 2008. She is one of the faculty members of Faculty of Computer Science and Information System, UTM, under the Modeling and Industrial Computing Department. She began lecturing in 2001, and a member of Soft Computing Research Group (SCRG) which is under the K-Economy Research Alliance, Universiti Teknologi Malaysia. Her research interests are in the field of soft computing, pattern recognition, and remote sensing.