

Image Feature Extraction using Quantum-PSO and Chaotic Map

Seba Aziz Sahy

Foundation of Technical Education, Institute of Medical Technology AL-Mansur, Baghdad, Iraq

Abstract: One of the population-based heuristic global search algorithms is the Particle Swarm Optimization (PSO) algorithm that is motivated through patterns of social behavior of organisms which live and interact within large groups. The PSO is depended on researches on swarms such as fish schooling and bird flocking Quantum Particle Swarm (Q-PSO) algorithm based on quantum individual, the theory of quantum used the change the adapting mode of the particles. In this study, Q-PSO algorithm was used in order to enhance the speed of search and the convergence precision and guarantee the effectiveness and simplification. It is simpler and more powerful than the algorithms available. The simulation and its application in the feature extraction prove its high efficiency.

Key words: Particle swarm optimization, quantum particle swarm optimization, chaotic number generator, logistic map, simplification, efficiency

INTRODUCTION

Image feature extraction consists of decreasing the amount of assets necessary to define a large set of image data. The analysis of image contains complex data sacrificed from the number of variables involved which a large memory size and computation need speed. A classification algorithm almost required training samples where feature extraction is used. The feature of an image can be extracted in a different method and used in many applications the important thing is the unity of this feature that will be useful in any application (Sun *et al.*, 2007). A swarm intelligent optimization used for extract features or enhances it. A most common term for methods that are used for construction is a group of the variables to get around these problems and maintain an accurate description of data (Hashemi and Nowrouzian, 2012).

The best outcomes are accomplished when experts construct a group of application-dependent features (citation needed) through a process referred to as feature engineering. Nevertheless, if these experts hold a small knowledge, general dimensionality reduction techniques may be helpful. In 2004, the quantum behaved particle swarm algorithm was presented for the first time by Sun *et al.* (2012). The researches on the quantum particle swarm optimization mostly focused on the upcoming three viewpoints: the first is proof theoretic research, the contraction expansion factor is improved by the second one and the last one is combined with other algorithms.

In 2005, a proposal of particle swarm algorithm was presented by Van den Bergh and Engelbrecht (2004) for the combinational logic circuit. The algorithm of quantum mechanical particle swarm was proposed by Mikki and

Kishk (Neshat *et al.*, 2012) in 2006, it was used for the optimization of electromagnetic issues. It is important to mention that Quantum Particle Swarm Optimization (QPSO) is considered to be a part of image watermarking in the medical field where it is used for copyright protection and authentication (Sun, 2009). One of the most serious challenges in the digital watermarking system is the trade-off between the robustness and imperceptibility. It is right to consider image watermarking as optimization matter through the use of QPSO algorithm and human visual system attribute in adaptive quantization index modulation and singular value decomposition in conjunction with discrete cosine transform and discrete wavelet transform. In the literature (Omkar *et al.*, 2009) a more efficient and modified version of the chaotic sequences (CQPSO) along with the QPSO is proposed and evaluated. An artificial fish swarm algorithm depends on a filtering methodology for trial solutions admission is analyzed for general restricted global optimization problems (Niknam *et al.*, 2009). The adequacy of the proposed method is shown through the preliminary numerical experiments with a familiar benchmark group of problems of engineering design. Literature (Behnamian and Ghomi, 2010) introduced an enhanced dynamic clustering algorithm, it consolidates the QPSA with the k-means algorithm by the introduction of the latest distance metric rules and improving the encoding of quantum particles. The algorithm possesses a quantum behaved particle swarm global search capability. With respect to increase the convergence speed, the k-means algorithm is applied to optimize each particle. Through modifying the value of the fitness function, the proposed algorithm is able to inspect

the optimal clustering number of clusters, so, this number and centers are not subject to subjective factors. Extensive experiments confirmed the efficiency of the algorithm. The proposal of the quantum particle swarm evolutionary algorithm was made depending on two things particle swarm optimization and quantum evolutionary algorithm (Dong *et al.*, 2010).

Particle Swarm Optimization (PSO): Eberhart and Kennedy (Dong *et al.*, 2010) proposed the Particle Swarm Optimization (PSO) algorithm. The algorithm of PSO based on population search algorithm, moreover, it depends on the simulating the Bee's social behavior, birds a school of fishes. The algorithm of PSO mainly proposed to graphically simulate the unpredictable choreography and graceful of a bird folk. Every individual within the swarm is represented by a vector in multidimensional search space. This vector possesses as well one assigned vector to determine the following movement of the particle and is called the velocity vector. The PSO algorithm in addition define the manner to update the velocity of each particle (Abdel-Kader, 2011). The velocity of every particle is updated depending on the best position that has been explored, so far and the actual velocity; moreover, it also depend on the best position explored by the swarm. The process of PSO then iterated for a fixed number of times or until achieved the minimum error that based on the desired performance index. It is shown that this simple model can efficiently handle the difficult problems of optimization. The PSO was originally developed for real-valued spaces but many problems are defined for discrete-valued spaces where the domain of the variables is finite.

The initialization process of the particles is performed by assigning random positions in the search space. Velocities are randomly initialized with the range (v_{min} , v_{max}). For each particle, a new velocity is calculated with respect to each iteration and the new position is specified as the sum of the new velocity and the previous position Eq. 1 (Wang *et al.*, 2012):

$$x(t+1) = x(t) + v(t+1) \quad (1)$$

However, updating the new velocity, the best position achieved so far by all particles ($g(t)$), the best position of current particle achieved so far ($p(t)$, particle best) and global best) are employing the following Eq. 2 (Wang *et al.*, 2012):

$$v(t+1) = \omega v(t) + \phi_1 \text{rand}(0, 1) (p(t) - x(t)) + \phi_2 \text{rand}(0, 1) (g(t) - x(t)) \quad (2)$$

where, ω represents the inertia weight that is used to controls the magnitude of the old velocity $v(t)$ while the

significance of $p(t)$ and $g(t)$ are determined by ϕ_1 and ϕ_2 , respectively. Moreover, the parameter v_{max} is constrain v_i at any time step of the algorithm (Tajbakhsh *et al.*, 2012).

Quantum Particle Swarm Algorithm (QPSA): The quantum computing into the Particle Swarm Algorithm (PSA) is introduced by the algorithm of quantum behaved particle swarm optimization, starting from a mechanical perspective of the fact that the particle exists in space has quantum behavior. The algorithm surmounting the drawbacks while keeps presenting the benefits of PSA which enhances the performances of algorithms used for optimization in an effective manner (Thanushkodi and Deeba, 2012).

The principle of the QPSA: The algorithm of quantum particle swarm is proposed by Sun *et al.* with respect to the principle of quantum mechanics. The QPSA is used with the quantum space as a consequence of DELTA potential well. The quantum space particle algorithm utilizes wave function to describe (Eq. 3) (Xu *et al.*, 2013):

$$|\Psi|^2 dx dy dz = Q dx dy dz \quad (3)$$

where, $|\Psi|^2$ donates the square module of wave function which represents the probability density of particles in a position to appear. The variable Q defines the probability density function which also satisfies the normalization condition (Eq. 4) (Xu *et al.*, 2013):

$$\int_{-\infty}^{+\infty} |\Psi|^2 dx dy dz = \int_{-\infty}^{+\infty} Q dx dy dz \quad (4)$$

Suppose that there is a population for the D dimension of quantum space and contains n particles. The i th particle has the location $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ as well the particle during the history of best location is $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$; after all particles of best historical position is $p_g = (p_{g1}, p_{g2}, p_{gD})$. However, in quantum space, the locations of particles after the particles can be obtained according to the stochastic simulation of Monte Carlo measurement Eq. 5 (Arani *et al.*, 2013):

$$x_{id} = p_{id} \pm \frac{L}{2} \ln\left(\frac{1}{u}\right) (i = 1, 2, \dots, n) (d = 1, 2, \dots, D) \quad (5)$$

where, u represents a randomly chosen number for the range of $(0, 1)$. L is acquired by the current position of the particle and the historical best position is $L = 2 \cdot \beta \cdot |p_{id} - x_{id}|$. Therefore, the update of formula of the QPSO is Eq. 6 (Arani *et al.*, 2013):

$$x_{id}(t+1) = p_{id} \pm \beta \cdot |p_{id} - x_{id}(t)| \cdot \ln\left(\frac{1}{u}\right) \quad (6)$$

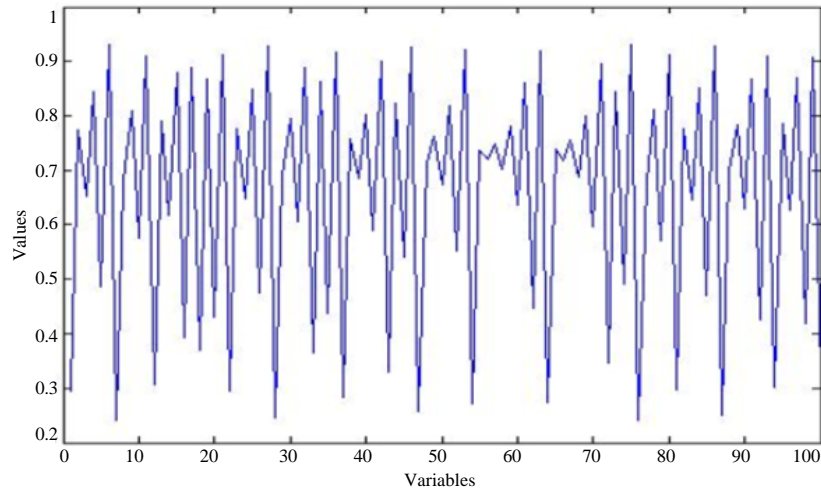


Fig. 1: Logistic map sequences point

where t is denote the iteration number of the algorithm. The factor of contraction expansion is represented by β which is the unique parameter of QPSA. However, Sun *et al.* are attempted to enhance the QPSA in order to avoid the convergence of premature by introducing m best in the algorithm as following (Arani *et al.*, 2013) Eq. 7:

$$mbest(t) = \frac{1}{n} \sum_{i=1}^n p_i t$$

$$= \left[\frac{1}{n} \sum_{i=1}^n p_{i1}(t), \frac{1}{n} \sum_{i=1}^n p_{i2}(t), \dots, \frac{1}{n} \sum_{i=1}^n p_{iD}(t) \right] \quad (7)$$

Among them, p_i is represent the best position of the i th particles while n is denote the how many particles are there. However, “ $mbest$ ” used to find the average of best location of number n of particles for finding solutions for problems depend on the variable’s dimension. After introducing the $mbest$, the updated formula is defined as following Eq. 8 (Li *et al.*, 2013):

$$L = 2 \cdot \beta \cdot |mbest - x_{id}|,$$

$$x_{id} = p_{id} \pm \beta \cdot |nbest_d - x_{id}| \cdot \ln\left(\frac{1}{u}\right) \quad (8)$$

As a result, the new-updating formula of the quantum particle swarm optimization particle become like the following (Li *et al.*, 2013):

$$p_{id} = \phi \cdot p_{id} + (1 - \phi) \cdot p_{gd} \cdot mbest(t) = \frac{1}{n} \sum_{i=1}^n p_i(t)$$

$$= \left[\frac{1}{n} \sum_{i=1}^n p_{i1}(t), \frac{1}{n} \sum_{i=1}^n p_{i2}(t), \dots, \frac{1}{n} \sum_{i=1}^n p_{iD}(t) \right] \quad (9)$$

$$X_i(t+1) = p_i \pm \beta \cdot |mbest - X_i(t)| \cdot \ln\left(\frac{1}{u}\right)$$

where, the random number in $(0, 1)$ is represented by ϕ ; and the rest parameters have the same definition as the ones referred to above (Elloumi *et al.*, 2014).

Dynamic chaotic systems: Chaos in general defined as clear fact which seem to be in nonlinear known systems reactive to the primary states in addition to having pseudo random activity. In case of that, the chaotic dynamical systems are faced with the Lyapunov exponential function they will together continue steadily in chaos mode. However, the pseudo random conduct draws the consideration of many cryptographic systems to this noticeable fact (Ji *et al.*, 2014). The pseudo random character attempt to produce clear data of the system in a new form to appear as a random to the attacker sight. However, it seems recognizable to the destined recipient and probably be decrypted until now many of chaos algorithms that build on cryptography are proposed. Indeed, many of these algorithms are used in one-way or another in order to encrypt the text and image as well. For an encryption system, it is essential to have the appropriate speed to cipher an image with enormous data (Chen *et al.*, 2014).

The Logistic mapping: The logistic mapping is said to be as the paradigmatic exemplification for the chaotic mapping. Despite the fact of that, the logistic mapping has one-dimensional, nevertheless, the control reaction is extremely perfect. The logistic formula is represented by the following Eq. 10 (Wu *et al.*, 2015) (Fig. 1):

$$a_{n+1} = \lambda a_n (1 - a_n) \quad (n = 0, 1, 2, \dots) \quad (10)$$

In this equation, the variable is denoted by a_n in addition, λ is a refer to system parameter while $\lambda \in (0, 4)$,

$a_n \in (0, 1)$. In the condition that $1 \leq \lambda < 3$, the system will take the action of 'fixed point'. In the case of $\lambda = 3$, the system will start with the transmission phase. Whereas, $\lambda = 3.5699456$, the system will assume the chaotic condition as shown in Fig. 1.

MATERIALS AND METHODS

The proposed method is consisting of stages start with pre-process fingerprint image and extract a feature from it as shown in Fig. 2. There are two main stages in the proposed method PSO and Q-PSO. Each stage has specific functions, the two stages are explained in Fig. 3. Extract unique feature is a goal of this research based on starting point and select neighbourhood point.

Image per-processing: The pre-processing stage contains image noise removal using a median filter, image contrast adjustment and histogram equalization. These steps are useful for getting a grayscale image with more accurate results.

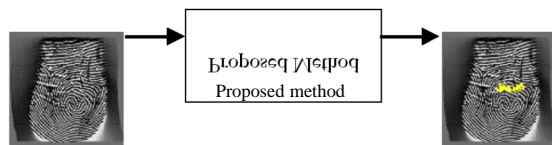


Fig. 2: General view of the proposed method

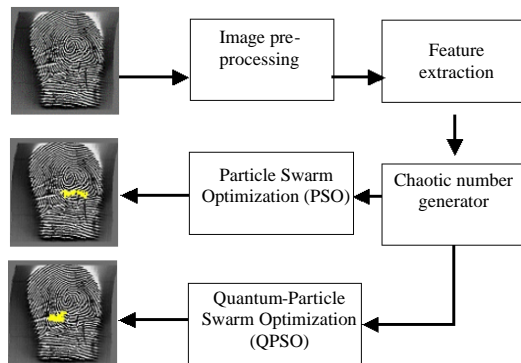


Fig. 3: A framework of the proposed method

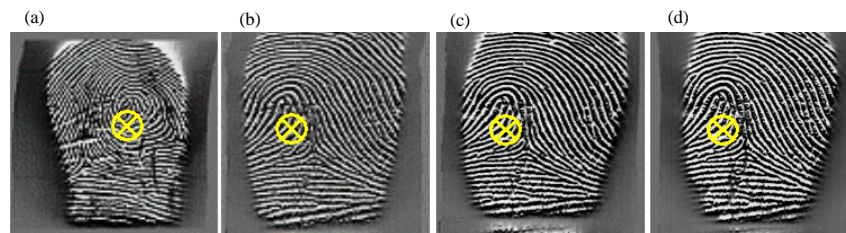


Fig. 4: a-d) Fingerprint initialization starting point

Feature extraction: The feature extract by selecting an initial point in the image, these coordinate of these point will be input to the next step for finding the bet neighboring point. The total point should be without duplicating. The duplicating point will ignore. The total point will depend on a number of iteration. The selecting of the starting point shown in Fig. 4. The selecting of the initial point may random or centre point of the image but it be useful to select the circular point of the fingerprint.

Chaotic number generator: Random number used in next step for control the PSO and QPSO. These numbers are used as a random number but it can have controlled by a logistic map that will always give the same sequences when used the same initial condition as shown in Table 1. The number of chaotic numbers generated depend on the number of iteration used in PSO and QPSO. The form of the number is a random number but it can be changed depending on changing initial conditions.

Particle Swarm Optimization (PSO): In this steps depend on starting point and the number of iteration, the PSO steps applied to find pbest and gbest that in each iteration after select the initial population from neighboring points. The pbest get it after evaluating the population and update pbest history. From multiple pbest select the gbest and update gbest history. In each iteration select the gbest location as a point in features. After all, iteration finished the total points of features are getting it. The total steps of PSO show in Fig. 5.

Quantum Particle Swarm Optimization (PSO): In this steps depend on starting point and the number of iteration, the QPSO steps applied by Initializing the algorithm parameters (population size n , particle dimension d , the maximum number of iterations MAXGEN), initializing population, initialization particles history pbest and global history optimal value gbest. Evaluate individual fitness value. Update the optimal population in history. Update the history global optimal particle in a population. The particles are updated through

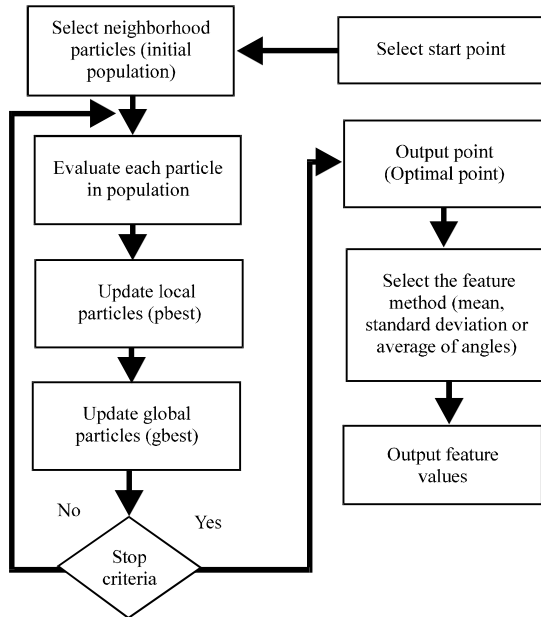


Fig. 5: Total steps of PSO

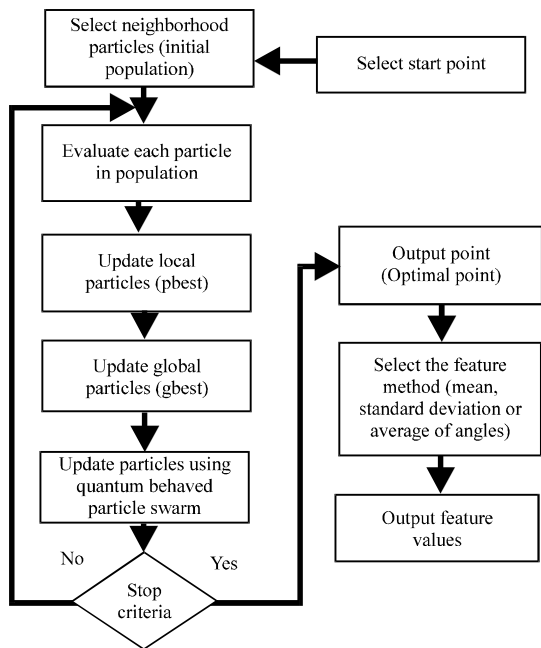


Fig. 6: Total steps of QPSO

the use of quantum behaved particle swarm optimization algorithm formula, all available particles in space. When the algorithm reaches the highest number of iterations, then points will be the optimal solution. After all iterations are finished, the total points of features are got it. The total steps of QPSO show in Fig. 6.

Table 1: Chaotic number generator

#	CNG
1	0.294531
2	0.909820
3	0.913023
4	0.605473
5	0.589069
6	0.818679
7	0.860903
8	0.895980
9	0.296486
10	0.868168
11	0.773888
12	0.305586
13	0.295772
14	0.889694
15	0.901580
16	0.552879
17	0.446006
18	0.347125
19	0.776866
20	0.426279
21	0.651735
22	0.790353
23	0.775781
24	0.365519
25	0.330489
26	0.920713
27	0.920269
28	0.844083
29	0.645626
30	0.910886
31	0.845376
32	0.617133
33	0.647858
34	0.863769
35	0.824108
36	0.271891
37	0.273281
38	0.490172
39	0.852143
40	0.302329
41	0.486850
42	0.880027
43	0.849702
44	0.438271
45	0.539883
46	0.737328
47	0.739683
48	0.930768
49	0.469272
50	0.785597

RESULTS AND DISCUSSION

The proposed method applied on several fingerprint images, to test two proposed methods (PSO and Q-PSO) the main characteristics is to specify unique feature for each fingerprint that will be helpful in the identification algorithm. It was taking a dataset of 20 fingerprint grayscale images for the experiment result as shown in Fig. 7 with size 100*100 pixel. In proposed method is chosen as a starting point and then find the best points that are connected by using similar methods and excluding the points, the results showed that the second

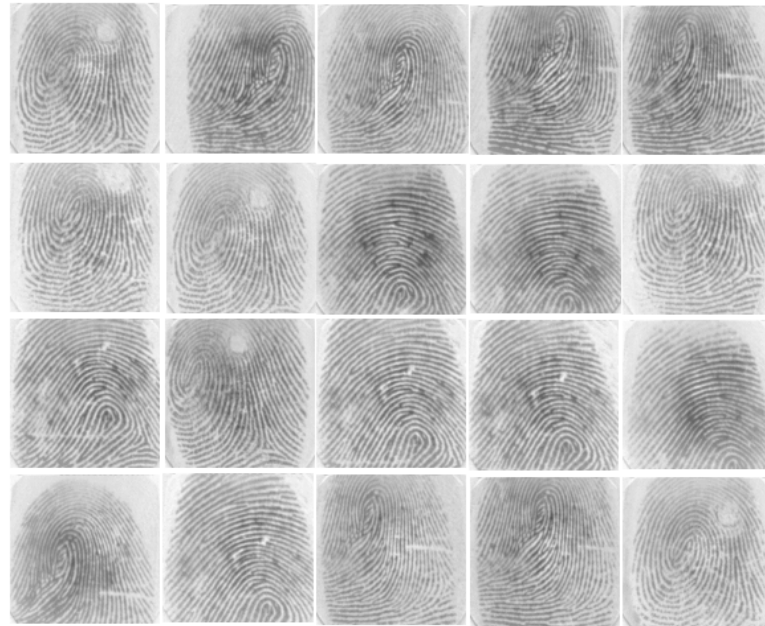


Fig. 7: Fingerprint image training set

Table 2: PSO feature 1 of starting point (50, 50) mean

Mean	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	142.8000	135.0750	137.9167	139.0500	140.4100
Image 2	180.9500	178.4750	181.7167	182.5500	175.2500
Image 3	170.5500	179.2500	180.2500	177.6750	180.0500
Image 4	138.6000	141.8250	162.1500	161.9500	162.3100
Image 5	192.4000	199.8250	200.3667	201.4750	200.7500

Table 3: Q-PSO feature 1 of starting point (50, 50) mean

Mean	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	131.1500	133.6250	144.4833	147.5000	148.8700
Image 2	163.6500	165.3750	159.6167	157.3250	159.1100
Image 3	148.1000	158.9000	165.3667	168.5375	171.0200
Image 4	146.0500	147.2500	160.4833	158.4625	158.4100
Image 5	195.0500	200.0000	199.1333	201.7625	203.3000

Table 4: PSO feature 1 of starting point (55, 55) mean

Mean	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	142.8000	135.0750	137.9167	139.0500	140.4100
Image 2	180.9500	178.4750	181.7167	182.5500	175.2500
Image 3	170.5500	179.2500	180.2500	177.6750	180.0500
Image 4	138.6000	141.8250	162.1500	161.9500	162.3100
Image 5	192.4000	199.8250	200.3667	201.4750	200.7500

Table 5: Q-PSO feature 1 of starting point (55, 55) mean

Mean	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	131.1500	133.6250	144.4833	147.5000	148.8700
Image 2	163.6500	165.3750	159.6167	157.3250	159.1100
Image 3	148.1000	158.9000	165.3667	168.5375	171.0200
Image 4	146.0500	147.2500	160.4833	158.4625	158.4100
Image 5	195.0500	200.0000	199.1333	201.7625	203.3000

algorithm (Q-PSO) produces similar points less than the first-method (PSO). The results points will have used to extract features such as mean, standard deviation or average angles with a starting point.

The mean of extracted point was find in different iteration (20, 40, 60, 80 or 100) for first starting point (50,

50) as shown in Table 2 and 3 for PSO and Q-PSO, respectively while Table 4 and 5 for another starting point (55, 55) also for PSO and Q-PSO, respectively.

The second feature extracted from resulted point is average angles that calculate for each point with respect to starting point all point makes different angle and the

Table 6: PSO feature 2 of starting point (50, 50) average angles

Average angle	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	0.52710137	0.20843239	0.03445625	-0.19694983	-0.22096907
Image 2	-0.54487645	-0.10376318	-0.06015946	-0.07471701	-0.00089074
Image 3	0.87472957	0.09482523	-0.24856845	-0.19141513	-0.37546464
Image 4	0.14038972	-0.25425806	0.09479174	0.35095143	0.45672198
Image 5	-0.61286160	-0.23617695	-0.28143229	-0.30368115	-0.15274606

Table 7: Q-PSO feature 2 of starting point (50, 50) average angles

Average angle	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	-1.05694298	-1.04692824	-1.02748108	-0.95303703	-0.85959683
Image 2	0.25383432	-0.16956243	-0.17439951	-0.27869806	-0.37375058
Image 3	-0.01869384	-0.12877244	-0.12432262	-0.04539367	0.04251374
Image 4	-0.83227055	-0.02578560	0.24624179	0.34578336	0.40787584
Image 5	0.09159371	0.00348499	0.24458722	0.16342714	0.13730375

Table 8: PSO feature 2 of starting point (55, 55) average angles

Average angle	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	-0.26745867	-0.34837458	0.16041742	0.31074727	0.37548345
Image 2	0.19831132	0.01512806	0.01813390	0.07122231	0.12862925
Image 3	-0.34662344	-0.76078064	-0.68865443	-0.31802553	-0.04468614
Image 4	-0.34827509	-0.75642668	-0.88170533	-0.84762607	-0.91650922
Image 5	0.50919123	0.26187188	0.06474584	-0.01890081	-0.16497301

Table 9: Q-PSO feature 2 of starting point (55, 55) average angles

Average angle	Iteration = 20	Iteration = 40	Iteration = 60	Iteration = 80	Iteration = 100
Image 1	0.46490929	-0.00213446	-0.23676694	-0.35045772	-0.49429184
Image 2	-0.46758687	0.27212863	0.43779903	0.40692685	0.39469099
Image 3	0.12344514	0.61300433	0.67102950	0.63927369	0.64633823
Image 4	-0.30820329	0.07144496	0.15083970	-0.19837580	-0.24251734
Image 5	1.11319511	0.76554826	0.56123204	0.41054982	0.27524466

average will be unique for each fingerprint image as shown in Table 6 and 7 for 20, 40, 60, 80 and 100 iterations with starting point (50, 50) with implementation PSO and Q-PSO, respectively. By the same way applied to another starting point (55, 55) in Table 8 and 9, respectively.

CONCLUSION

The PSO used for selection related point to the starting point in fingerprint this point will be used for feature extraction. The Q-PSO algorithms depend on a dynamic parameter adjustment which is conducive to jump out of local optima and move to the global extreme the results show that the improved algorithm to solve ability and can better solve the global optimum to asset the accuracy requirement. The result explains that all features are unique to the corresponding fingerprint image. The number of iteration is used for controlling the complexity of the algorithm.

REFERENCES

- Abdel-Kader, R.F., 2011. An improved discrete PSO with GA operators for efficient QoS-multicast routing. *Intl. J. Hyb. Inf. Technol.*, 4: 23-38.
- Arani, B.O., P. Mirzabeygi and M.S. Panahi, 2013. An improved PSO algorithm with a territorial diversity-preserving scheme and enhanced exploration-exploitation balance. *Swarm Evol. Comput.*, 11: 1-15.
- Behnamian, J. and S.F. Ghomi, 2010. Development of a PSO-SA hybrid metaheuristic for a new comprehensive regression model to time-series forecasting. *Exp. Syst. Appl.*, 37: 974-984.
- Chen, G., L. Liu, P. Song and Y. Du, 2014. Chaotic improved PSO-based multi-objective optimization for minimization of power losses and L index in power systems. *Energy Convers. Manage.*, 86: 548-560.
- Dong, S.F., Z.C. Dong, J.J. Ma and K.N. Chen, 2010. Improved PSO algorithm based on chaos theory and its application to design flood hydrograph. *Water Sci. Eng.*, 3: 156-165.
- Elloumi, W., H. El Abed, A. Abraham and A.M. Alimi, 2014. A comparative study of the improvement of performance using a PSO modified by ACO applied to TSP. *Appl. Soft Comput.*, 25: 234-241.
- Hashemi, S.A. and B. Nowrouzian, 2012. A novel discrete particle swarm optimization for FRM FIR digital filters. *J. Comput.*, 7: 1289-1296.
- Ji, W., J. Wang and J. Zhang, 2014. Improved PSO based on update strategy of double extreme value. *Intl. J. Contr. Autom.*, 7: 231-240.

- Li, Y.L., W. Shao, L. You and B.Z. Wang, 2013. An improved PSO algorithm and its application to UWB antenna design. *IEEE. Antenn. Wirel. Propag. Lett.*, 12: 1236-1239.
- Neshat, M., S.F. Yazdi, D. Yazdani and M. Sargolzaei, 2012. A new cooperative algorithm based on PSO and K-means for data clustering. *J. Comput. Sci.*, 8: 188-194.
- Niknam, T., B. Amiri, J. Olamaei and A. Arefi, 2009. An efficient hybrid evolutionary optimization algorithm based on PSO and SA for clustering. *J. Zhejiang Univ. Sci.*, 10: 512-519.
- Omkar, S.N., R. Khandelwal, T.V.S. Ananth, G.N. Naik and S. Gopalakrishnan, 2009. Quantum behaved Particle Swarm Optimization (QPSO) for multi-objective design optimization of composite structures. *Exp. Syst. Appl.*, 36: 11312-1322.
- Sun, J., 2009. Particle swarm optimization with particles having quantum. Ph.D Thesis, Jiangnan University, Wuxi, China.
- Sun, J., W. Xu, W. Fang and Z. Chai, 2007. Quantum-Behaved Particle Swarm Optimization with Binary Encoding. In: *Adaptive and Natural Computing Algorithms*, Beliczynski, B. *et al.* (Eds.). LNCS. 4431, Springer-Verlag, Berlin, Heidelberg, ISBN: 978-3-540-71589-4, pp: 376-385.
- Sun, J., X. Wu, V. Palade, W. Fang and C.H. Lai *et al.*, 2012. Convergence analysis and improvements of quantum-behaved particle swarm optimization. *Inf. Sci.*, 193: 81-103.
- Tajbakhsh, A., K. Eshghi and A. Shamsi, 2012. A hybrid PSO-SA algorithm for the travelling tournament problem. *Eur. J. Ind. Eng.*, 6: 2-25.
- Thanushkodi, K. and K. Deeba, 2012. On performance analysis of hybrid intelligent algorithms (Improved PSO with SA and Improved PSO with AIS) with GA, PSO for multiprocessor job scheduling. *WSEAS. Trans. Comput.*, 11: 131-147.
- Van den Bergh, F. and A.P. Engelbrecht, 2004. A cooperative approach to particle swarm optimization. *IEEE Trans. Evol. Comput.*, 8: 225-239.
- Wang, Y., J. Zhou, C. Zhou, Y. Wang and H. Qin *et al.*, 2012. An improved self-adaptive PSO technique for short-term hydrothermal scheduling. *Exp. Syst. Appl.*, 39: 2288-2295.
- Wu, T., Y.S. Yan and X. Chen, 2015. Improved dual-group interaction QPSO algorithm based on random evaluation. *Contr. Decis.*, 30: 526-530.
- Xu, G., W. Qiuhua, Y. Shouwang, C. Wei and Z. Changhai, 2013. Improved PSO algorithms for improving the subdivision accuracy of photoelectric rotary encoder. *Inf. Laser Eng.*, 42: 1508-1513.