Research Article

An Improved Quantum-Inspired Genetic Algorithm for Image Multilevel Thresholding Segmentation

Jian Zhang,¹ Huanzhou Li,¹ Zhangguo Tang,¹ Qiuping Lu,¹
Xiuqing Zheng,² and Jiliu Zhou³

¹ School of Physics and Electronics Engineering, Sichuan Normal University, Chengdu 610066, China

² College of Computer Science, Sichuan Normal University, Chengdu 610066, China

³ College of Computer Science, Sichuan University, Chengdu 610065, China

Correspondence should be addressed to Jian Zhang; dctscu07@163.com

Received 4 June 2013; Revised 29 December 2013; Accepted 24 February 2014; Published 2 April 2014

Academic Editor: Jianming Shi

Copyright © 2014 Jian Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A multilevel thresholding algorithm for histogram-based image segmentation is presented in this paper. The proposed algorithm introduces an adaptive adjustment strategy of the rotation angle and a cooperative learning strategy into quantum genetic algorithm (called IQGA). An adaptive adjustment strategy of the quantum rotation which is introduced in this study helps improving the convergence speed, search ability, and stability. Cooperative learning enhances the search ability in the high-dimensional solution space by splitting a high-dimensional vector into several one-dimensional vectors. The experimental results demonstrate good performance of the IQGA in solving multilevel thresholding segmentation problem by compared with QGA, GA and PSO.

1. Introduction

Image segmentation which is considered as an important basic operation for computer vision is a low-level image processing task, and its result could be presented as input to higher-level processing tasks such as pattern recognition, object tracking, and scene analysis. Image segmentation is also a classical problem of image processing. Various techniques for image segmentation have been proposed and improved so far, but most of the existing segmentation methods are designed for specific applications. There's neither a unified theoretical system for image segmentation nor a kind of common and effective approach to all types of images. For the advantages of simplicity of implementation, small amount of calculation, and stable performance, image thresholding technique becomes an effective and widely used tool in image segmentation [1, 2]. It is useful to separate objects from background or discriminate objects from objects that have distinct grey levels. Thresholding can be divided into bilevel thresholding and multilevel thresholding. Bilevel thresholding classifies the pixels into two groups (object and background), one including those pixels with gray levels above a certain threshold and the other including the rest.

Multilevel thresholding divides the pixels into several classes. The pixels belonging to the same class have gray levels within a specific range defined by several thresholds [3]. Especially for the images of which target and background occupy different range of gray, the target and background which are with different gray levels can be separated by thresholding techniques, such as extraction of the logos, printed characters, or images in the document images [4]; segmentation of blood cell image and CT image in medical applications [5, 6]; and segmentation of infrared thermal image in infrared nondestructive detection [7]. The use of thresholding techniques has been a very crucial step for fast and accurate segmentation.

Thresholding methods are distinguished as six categories, based on the exploitation of histogram shape information, measurement space clustering, histogram entropy information, image attribute information, spatial information, and local characteristics, respectively, and then spread out dozens of objective functions [7]. Generally, existing image thresholding segmentation problem is transformed into an optimization problem: to propose an objective function which is based on the different information considered or different theory and then to obtain the optimal thresholds by getting

the maximum or minimum value of the objective function. Among all the objective functions existing, the thresholding technique, the objective functions based on the entropy (Kapur method) [8], and the between-class variance (Otsu method) [9] are the most popular ways [10]. Otsu method can obtain a good segmentation result to the images with obvious bimodal histogram; Kapur method can also achieve a desired result to the images with no obvious bimodal or no bimodal histogram. However, due to the exhaustive searching mechanism, the increasing number of the thresholds leads computation time to grow exponentially which makes the above two methods become time-consuming issues, and it will greatly limit the application of the multilevel thresholding segmentation algorithm.

Image thresholding segmentation is also a task to find the optimal parameters in the complex parameter space, and the global optimal parameters of the space can be found through optimization algorithms in an acceptable period of time for their fast computational ability. Many algorithms are performed to multilevel thresholding segmentation to improve the computational efficiency [11–13].

Genetic algorithm (GA) provides a common system framework to solve complex optimization problems, and independent on the specific problem areas. GA has been applied to select the parameters in segmentation problem successfully and plays a very important role in image segmentation [14–18]. However, it falls into local optimum easily for the decrease of the diversity of the population in the later period of evolution [19].

Quantum-inspired genetic algorithm (QGA) is a new optimization algorithm which combines the concept of quantum computing and classical GA. QGA which was firstly introduced by Narayanan and Moore [20] is a new and promising branch of evolutionary algorithms. The experiment in [20] has demonstrated that QGA has a faster convergence speed than classical GA. In QGA, quantum-inspired bit (Q-bit) and quantum-inspired gate (Q-gate) are applied to represent genotype individuals and update Q-bits to generate offspring, and the genotypes and phenotypes are linked by a probabilistic observation process. The Q-bit encoding of chromosome of QGA has a better diversity of population, and the Q-gate which is defined as an update operator of QGA to drive the individuals toward better solutions can treat the balance between exploration and exploitation and avoid premature convergence and escape local optimal efficiently. Nowadays, the research on QGA has attracted the interest of many researchers of various research areas. QGA has been introduced to combinatorial optimization problems [21–23], numerical optimization problems [24], face detection [25], disk allocation [26], image segmentation [27, 28], feature selection [29], multiobjective optimization problems [30, 31], digital filter design [32], power dispatch [33], and so forth. The research on QGA has become a rapidly expanding field. QGA was introduced to image segmentation and achieved a faster speed of convergence than GA [34]. QGA was applied in [35] to determine the parameters and got better segmentation results than traditional fuzzy entropy-based method.

The convergence speed of QGA is dependent on the rotation angle greatly. An appropriate angle can help to accelerate the convergence speed and improve the searching ability of the algorithm. Many studies are interested in the research of the rotation angles and the update strategies [36]. Fixed rotation angles in lookup table are proposed in [21]; a H_e gate is employed in [22] to avoid premature convergence; the adjustments of the rotation angles which decrease gradually with the iterations are introduced in [37, 38]. Additionally, for high-dimensional optimization problem, the individual is an n -dimensional vector. In the processing of evolution, the update operation of Q-gate is only guided by the current best individual which has the maximum fitness but not always has a reasonable structure of the solution, and each update step is performed on a full n -dimensional vector. This leads to the possibility of some components of the n -dimensional vector having been moved closer to the best solution, while others have actually been moved away from the best solution [39]. This kind of holistic information exchange may cause the sharp decline of the search ability with the increase of the dimension. So it still can potentially get trapped in suboptimal locations in search space. In order to solve the above problems and improve the performance of the QGA-based multilevel thresholding segmentation, an improved QGA (IQGA) with adaptive rotation angle and cooperative learning for maximum entropy based multilevel thresholding segmentation is proposed in this paper. The experimental results show that the convergence speed, accuracy, and stability of the proposed algorithm are superior to traditional GA and QGA.

This paper is arranged as follows. Section 2 describes basic concept of the QGA and the improved QGA which introduced the adaptive adjustment strategy and cooperative learning strategy. Section 3 presents the maximum entropy criterion to multilevel thresholding segmentation. The detailed algorithm for IQGA-based image multilevel thresholding segmentation is presented in Section 4. The experimental results are given in Section 5. The conclusions are drawn in Section 6.

2. Improved Quantum Genetic Algorithm

2.1. QGA. QGA is an attractive tool to provide efficient solutions for most complex optimization problems. It is a probabilistic optimization algorithm which introduces the concept and theory of quantum computing into classical GA. Like classical GA, QGA is characterized by the representation of individuals, population diversity, and the use of a fitness evaluation mechanism. However, a quantum-inspired bit (Q-bit) representation is used in QGA instead of binary, numeric, or symbolic representation which is usually used in GA. Since the Q-bit representation can achieve a linear superposition of states given its probabilistic approach, it is conducive to population diversity. A Q-bit is the basic computing unit in a QGA and is defined as a column vector: $(\alpha\beta)^T$ (a Q-bit is often represented as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ in quantum mechanical ket-notation), where the numbers of α and β satisfy the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. The values of $|\alpha|^2$ and

$|\beta|^2$ denote the probabilities that the Q-bit will be found in the states of “0” and “1,” respectively. By a process of probabilistic observation, each Q-bit can be rendered into one binary bit. A multi-Q-bit system can be extended naturally. For example, a m -Q-bits system can be described as follows:

$$q_i = \begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots & \beta_m \end{pmatrix}. \quad (1)$$

This representation has the advantage that it is able to represent any superposition of states. The combination of the m -Q-bits is called quantum chromosome. GA uses the operations such as selection, crossover, and mutation, to maintain the diversity of the population and to obtain the global optimal solution. However, QGA applies quantum-inspired rotation gate (Q-gate) $U(\Delta\theta_i)$ to update the state of Q-bits only. The Q-gate is represented as follows:

$$U(\Delta\theta_i) = \begin{pmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{pmatrix}. \quad (2)$$

The Q-gate $U(\Delta\theta_i)$ is a unitary operation which is used to change the phase of the Q-bit and does not change the length of the Q-bit. The Q-gate is employed to update the state of a Q-bit as follows:

$$\begin{pmatrix} \alpha'_i \\ \beta'_i \end{pmatrix} = U(\Delta\theta_i) \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = \begin{pmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{pmatrix} \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix}, \quad (3)$$

where $(\alpha_i, \beta_i)^T$ and $(\alpha'_i, \beta'_i)^T$ are the i th Q-bits of the chromosome before and after the updating, respectively. $\Delta\theta_i$ represents the rotation angle of each Q-bit whose value and direction can be adjusted by some strategies.

2.2. Adaptive Adjustment Strategy of the Q-Gate. In QGA, the evolution of the individual is guided by the current best individual and adjusted by the Q-gate, so that the individual approximates and converges to the global optimal solution ultimately. A Q-gate is used to change the state of the quantum chromosome to evolve the individual. In the processing of evolution, the choice of the quantum rotation angles is very important, and a suitable choice of the rotation angles can help improving the search ability of the algorithm. Different design of the quantum rotation angle is suitable for different problem. Although the designs of the quantum rotation angle are different, the core idea of the designs is the same—to make the current individual evolve to a higher fitness solution.

Considered the feature of the problem and the relation between the current individual and the current best individual, an adaptive rotation angle is proposed in this study, and the rotation angle is defined as follows:

$$\Delta\theta_i = g \cdot \Delta\theta, \quad (4)$$

where the value of $\Delta\theta_i$ can be looked up in Table 1.

g is an adjustment function which is defined as

$$g = \exp\left(\frac{f_{\text{best}} - f_i}{f_{\text{best}} - f_{\text{worst}}}\right), \quad (5)$$

TABLE 1: Lookup table of θ_i .

x_i	b_i	$f(x_i) \geq f(b_i)$	$\Delta\theta$
0	0	False	0
0	0	True	0
0	1	False	0.01π
0	1	True	0
1	0	False	-0.01π
1	0	True	0
1	1	False	0
1	1	True	0

where f_i , f_{best} , and f_{worst} represent the fitness of the current individual, the current best individual, and the current worst individual, respectively. The value of f reduces when f_i approach to f_{best} in order to decrease the search step length and vice versa. The rotation angle is updated by

$$\theta'_i = \theta_i + \Delta\theta_i. \quad (6)$$

2.3. Cooperative Learning Strategy. The classical analysis on GA considers that the population represents a set of competing structures that explore various parts of the search space in parallel. Instead of emphasizing the notion of competition, Cobb regarded population as a set of cooperative agents, and the members are cooperative in that they share information; that is, they communicate partial solutions to one another [40]. Without sharing information, the time required for an agent to find a solution may be very long. It is believed that a group of cooperating agents engaged in problem solving can solve a task faster than either a single agent or the same group of agents working in isolation from each other [41]. A general model was presented for coevolution of cooperating species to improve the performance of GA [42]. A cooperative method was introduced to particle swarm optimization (PSO) to help the particle contribute its merits to the population in each dimension and to promote the cooperation between particles in each dimension [39].

As represented by the quantum probability model in QGA, each individual of the population, regardless of its fitness value being high or low, represents a potential solution which evolves towards the global optimum guided by the current best individual. The relationship among individuals is not only competition but also cooperation. That is to say, the information sharing and exchange exist in QGA also, but the information sharing and exchange are between the individual and the current best individual. The current best individual has the maximum fitness value, but, for high-dimensional optimization problem, the structure of the solution of the current best individual is not always reasonable for each individual is usually represented as an n -dimensional vector. This means that the same update operation performed on each dimensional of an individual may not always evolve the individual. Therefore, it is important to take into account not only the whole adjustment of the individual, but also the rationality of the structure of the solution. And it is an effective way to improve the global and local optimization ability of QGA to reinforce the holistic and local information

```

Choose  $m$   $n$ -dimensional individuals
Repeat:
  For each individual  $q_i$  with fitness  $f_i$   $i \in [1, m]$ 
    For each dimension  $j \in [1, n]$ 
       $q_{ij} = (q_{i1}, \dots, q_{bj}, \dots, q_{im})$ 
      Evaluate  $q_{ij}$  and store its fitness  $f_{ij}$ 
    Endfor
     $f'_{\text{best}} = \max(f_{ij})$ 
     $q'_{\text{ibest}} = \arg \max(f_{ij})$ 
    If  $f'_{\text{best}} > f_i$ 
       $q_i = q'_{\text{ibest}}$ 
    Endif
  Endfor
Until termination criterion is satisfied

```

PSEUDOCODE 1: Pseudocode for cooperative learning strategy.

exchange and promote the information sharing among individuals, especially for the high-dimensional optimization. In other words, updating a part of an individual instead of updating it as a whole item sometimes may help accelerate the evolution processing and improve the search capability of QGA.

Image multilevel thresholding segmentation is a high-dimensional optimization problem, and it is time-consuming. Applying QGA to multilevel thresholding can reduce computational time greatly. The individual in QGA-based method is an n -dimensional vectors, and the performance of QGA declines with the increase in the dimensions. In order to improve the performance of QGA, a cooperative learning strategy is proposed in this study. The n -dimensional vector is split into n one-dimensional vectors, and then each one-dimensional vector is updated in the direction of the current best individual. This mean is called cooperative learning in this study. The cooperative learning strategy is presented as follows.

Assuming an n -dimensional unconstrained optimization which needs to maximize the objective function $f(x)$,

$$X^* = (x_1^*, x_2^*, \dots, x_n^*) = \arg \max (X), \quad (7)$$

where n represents the dimension of the parameters, $X^* = (x_1^*, x_2^*, \dots, x_n^*)$ is the global optimal solution, and X is the collection of the candidate solutions.

The QGA which introduced cooperative learning strategy is performed to this optimization problem. Assuming $q_{\text{best}} = (q_{b1}, \dots, q_{bj}, \dots, q_{bn})$ is the current optimal n -dimensional individual with fitness f_{best} , some n -dimensional individuals $Q = \{q_1, \dots, q_i, \dots, q_m\}$ are selected based on the fitness proportional model. For every individual $q_i = (q_{i1}, \dots, q_{ij}, \dots, q_{in})$ with fitness f_i , each dimensional component of q_i is replaced by the corresponding dimensional component of q_{best} , and its fitness value f_{ij} is computed. The best individual with the maximum fitness $f'_{\text{ibest}} = \max\{f_{i1}, \dots, f_{ij}, \dots, f_{in}\}$ is selected, and then its fitness f'_{ibest} is selected to be compared with f_i , if $f'_{\text{ibest}} > f_i$, q_i is replaced by q'_{ibest} ; otherwise, q_i is retained. By means of this way,

the structures of the individuals are readjusted. This method accelerates the global convergence. The pseudocode of the cooperative learning strategy is shown in Pseudocode 1.

3. Entropy Criterion Based Measure

An image which contains N pixels and G gray levels is divided into M categories ($C_1, C_2, \dots, C_k, \dots, C_M$) by T ($T = \{t_1, t_2, \dots, t_k, \dots, t_{M-1}\}$, $t_1 < t_2 < \dots < t_k < \dots < t_{M-1}$) thresholds. For convenience, the lowest gray $g_{\min}$ and the highest gray $g_{\max}$ are set as the lowest threshold t_0 and the highest threshold t_M , respectively. If a pixel with gray g meets the condition $t_{k-1} < g \leq t_k$, the pixel is classified under the category C_k . Searching for the optimal threshold which met the given classification requirements in the histogram, we can maximize or minimize an objective function which chooses the threshold as its parameter. That is to say, thresholding problem can be seen as a global optimization problem to choose a collection of thresholds T^* by optimization of an objective function $f(T)$. This can be described as follows:

$$T^* = \arg \max_{0 \leq T \leq G-1} f(T). \quad (8)$$

Based on the entropy criteria, Kapur method looks the object and background of the image as two different sources, and when the total entropy of the two categories is maximum, the best segmentation and its corresponding optimal threshold are considered to have been obtained. Kapur method was originally used to solve the bilevel thresholding segmentation and later extended to the multithresholding segmentation. Kapur method works as follows.

For an image with G gray levels, the number of the pixels with gray g is $h(g)$, and the total number of the pixels of the image is $N = \sum_{g=0}^{G-1} h(g)$; the normalized probability of each gray level is defined as $p(g) = h(g)/N$. Determining $M-1$ thresholds for a given image, the Kapur entropy can be defined as:

$$f(t_1, \dots, t_k, \dots, t_{M-1}) = H_0 + H_1 + \dots + H_{M-1}, \quad (9)$$

where

$$\begin{aligned} \omega_0 &= \sum_{g=0}^{t_1-1} p_g, & H_0 &= - \sum_{g=0}^{t_1-1} \frac{p_g}{\omega_0} \ln \frac{p_g}{\omega_0} \\ \omega_1 &= \sum_{g=0}^{t_2-1} p_g, & H_1 &= - \sum_{g=t_1}^{t_2-1} \frac{p_g}{\omega_1} \ln \frac{p_g}{\omega_1} \\ &\vdots & & \\ \omega_{M-1} &= \sum_{g=t_{M-1}}^{G-1} p_g, & H_{M-1} &= - \sum_{g=t_{M-1}}^{G-1} \frac{p_g}{\omega_{M-1}} \ln \frac{p_g}{\omega_{M-1}}, \end{aligned} \quad (10)$$

where $H_0, H_1, \dots, H_{M-1}$ represent the histogram entropy of each category. The best thresholds $T^* = \{t_1^*, \dots, t_k^*, \dots, t_{M-1}^*\}$ based on maximum entropy are satisfied with

$$T^* = \underset{T}{\operatorname{argmax}} f(t_1, \dots, t_k, \dots, t_{M-1}). \quad (11)$$

Kapur method can obtain the best thresholds T with the maximum entropy by computing and comparing the entropy values of all categories. Its time complexity is $O(G^{M-1})$. Obviously, the computation complexity and calculation time will increase greatly with the increase in the threshold numbers that limits the implementation and generalization of the Kapur method. IQGA proposed in this study can solve this problem in an acceptable time.

4. IQGA for Image Multilevel Thresholding Segmentation

The proposed algorithm uses IQGA to solve image multilevel thresholding segmentation problem, and the details are described in this section.

4.1. Quantum Angle Encoding. A Q-bit can be represented as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where the numbers of α and β satisfy the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. Make $\alpha = \cos \theta$ and $\beta = \sin \theta$, so a Q-bit can be represented by a phase angle θ in a two-dimensional space. The phase angle θ which is called quantum angle is represented by (θ) . (θ) is equivalent to $|\psi\rangle = \cos(\theta)|0\rangle + \sin(\theta)|1\rangle$ or $(\cos(\theta), \sin(\theta))^T$. A quantum chromosome with length of m can be represented as $q = (\theta_1 | \theta_2 | \dots | \theta_m)$. The update of the quantum chromosome can be simplified to the following expression: $\theta' = \theta_i + \Delta\theta$.

A gray image includes $256 = 2^8$ gray levels in the range of $[0, 255]$, and each gray level can be coded by an 8-Q-bits quantum angle $q_i = (\theta_{i1}, \theta_{i2}, \dots, \theta_{ij}, \dots, \theta_{i8})$, $\theta_{ij} \in [0, 2\pi]$. If an image is divided into M classes by $M - 1$ thresholds, the

corresponding chromosome Q_i is composed by $L = 8 \times (M - 1)$ bits quantum genes:

$$\begin{aligned} Q_i &= (q_1, q_2, \dots, q_i, \dots, q_{(M-1)}) \\ &= (\theta_{11}, \theta_{12}, \dots, \theta_{1j}, \dots, \theta_{18}, \dots, \theta_{i1}, \theta_{i2}, \dots, \theta_{ij}, \dots, \\ &\quad \theta_{i8}, \dots, \theta_{(M-1)1}, \theta_{(M-1)2}, \dots, \theta_{(M-1)j}, \dots, \theta_{(M-1)8}), \end{aligned} \quad (12)$$

where $\cos \theta_{ij}$, $\sin \theta_{ij}$ represent the probability amplitude of 0 and 1. The binary number of the gray value can be obtained by quantum measure.

4.2. Crossover. It is necessary to introduce crossover in the algorithm to increase the diversity of the population and avoid premature convergence. Two-point crossover with the crossover probability P_c is adopted as follows: choose two chromosomes with length L and select $l (l \in [1, L])$ cross-bits. For example, select a pair of chromosomes q_i, q_j randomly:

$$\begin{aligned} q_i &= (\theta_{i1}, \theta_{i2}, \dots, \theta_{i(m-1)}, \theta_{im}, \dots, \theta_{i(n-1)}, \theta_{in}, \dots, \\ &\quad \theta_{8 \times (M-1)-1}, \theta_{8 \times (M-1)}), \\ q_j &= (\theta_{j1}, \theta_{j2}, \dots, \theta_{j(m-1)}, \theta_{jm}, \dots, \theta_{j(n-1)}, \theta_{jn}, \dots, \\ &\quad \theta_{8 \times (M-1)-1}, \theta_{8 \times (M-1)}), \end{aligned} \quad (13)$$

and generate cross-bits $m, n (0 < m, n \leq 8 \times (M - 1))$; the crossed chromosomes are presented as

$$\begin{aligned} q'_i &= (\theta_{i1}, \theta_{i2}, \dots, \theta_{i(m-1)}, | \theta_{jm}, \dots, \theta_{j(n-1)}, | \theta_{in}, \dots, \\ &\quad \theta_{i[8 \times (M-1)-1]}, \theta_{i[8 \times (M-1)]}), \\ q'_j &= (\theta_{j1}, \theta_{j2}, \dots, \theta_{j(m-1)}, | \theta_{im}, \dots, \theta_{i(n-1)}, | \theta_{jn}, \dots, \\ &\quad \theta_{j[8 \times (M-1)-1]}, \theta_{j[8 \times (M-1)]}). \end{aligned} \quad (14)$$

4.3. Mutation. Mutation imitates the phenomenon of a certain gene mutation on chromosome in the processing of evolution. Mutation changes the structure and the physical properties of the chromosome. The operation of mutation in QGA can increase the diversity of the population effectively. This study introduces the NOR-gate and Hadamard-gate to the mutation operation. The proposed mutation is performed as follows: select one or several gene bits randomly on a chromosome with mutation probability P_m , and update the gene bits by NOR-gate or Hadamard-gate randomly. The operation is performed by

$$\theta'_i = \begin{cases} \frac{\pi}{2} - \theta_i, & \text{NOR-gate} \\ \frac{\pi}{4} - \theta_i, & \text{Hadamard-gate.} \end{cases} \quad (15)$$

5. Experimental Results

In this section, the performance of the proposed algorithm is evaluated by comparing its results with QGA which using

```

Begin
 $g \leftarrow 0$ 
Initialize quantum chromosome population  $Q(g)$ 
Make  $P(g)$  by observing the states of  $Q(g)$ 
Evaluate  $P(g)$ 
Store the best solution among  $P(g)$  into  $B(g)$ 
While termination criteria are not satisfied do
     $g \leftarrow g + 1$ 
    Make  $P(g)$  by observing the states of  $Q(g)$ 
    Evaluate  $P(g)$ 
    Store the best solution among  $P(g)$  into  $B(g)$ 
    Update  $Q(g)$  using Q-gate
    Choose  $m$  individuals and perform cooperative learning strategy
    Crossover
    Mutation
Endwhile
End

```

PSEUDOCODE 2: Pseudocode for the IQGA.

TABLE 2: Optimal segmentation thresholds, objective function values, and the average computation times of the standard ES-Kapur method with $C = 2, 3, 4$.

Image	Segmentation results	C		
		2	3	4
Cameraman	T	128-193	44-104-193	44-97-146-197
	f	12.1687	15.2274	18.3955
Hunter	T	91-172	62-117-174	62-117-174-230
	f	12.6357	15.8115	18.7441
Baboon	T	79-143	44-98-152	33-74-114-159
	f	12.2168	15.2801	18.1304
Scene	T	92-163	73-120-170	71-113-157-196
	f	12.5254	15.5654	18.3681
Tree	T	86-160	79-129-178	76-118-157-196
	f	12.1677	15.3271	18.0325
Fl6	T	71-173	69-127-183	67-106-145-185
	f	12.2115	15.5038	18.3120
Lena	T	97-164	82-126-175	44-98-152
	f	12.3471	15.3183	18.0126
Peppers	T	93-179	73-126-178	46-84-130-179
	f	12.5532	15.8258	18.7338
Average computation times (s)		9.5236	354.3866	19428.6

a fixed look-up table and full cross-interference scheme [43], PSO [13] and GA [15] which using a fixed lookup table and full cross-interference scheme [43]. In order to compare the performances of the algorithms, all algorithms (PSO, GA, QGA, and IQGA) adopted the same objective function which is based on maximum entropy criterion. The parameters are set as follows: *population* = 20, crossover probability $P_c = 0.2$, and mutation probability $P_m = 0.1$. The pseudocode for the proposed algorithm is shown in Pseudocode 2.

Some popular images are selected in the experiment. The original test images along with their corresponding gray level histogram are shown in Figures 1 and 2.

Table 2 shows the optimal segmentation thresholds, the maximum values of the objective function, and the average computation times obtained by standard Kapur method. Since the standard Kapur method is an exhaustive algorithm, the results of the method can be used to be the standard evaluation criteria, and the results obtained by other algorithms can be evaluated by this criterion. The letters “ T ” and “ f ” represent the optimal objective function values and the corresponding segmentation thresholds, respectively. As can be seen from Table 2, the average computational time increases from 9.5236 seconds to 19428 seconds exponentially when C changes from 2 to 4. The exhaustive method is a time-consuming task, and it is almost difficult to apply the method

FIGURE 1: Test images. (a) Cameraman; (b) Hunter; (c) Baboon; (d) Scene; (e) Tree; (f) F16; (g) Lena; (h) Peppers.

FIGURE 2: Gray level histogram of the test images. (a) Cameraman; (b) Hunter; (c) Baboon; (d) Scene; (e) Tree; (f) F16; (g) Lena; (h) Peppers.

TABLE 3: Comparison of the mean computational time and the average convergence generation.

ε	C	GA		QGA		IQGA		PSO	
		t (s)	g	t (s)	g	t (s)	g	t (s)	g
1%	2	0.1248	2	0.2405	3.3	0.1831	1.6	0.1955	3.1
	3	0.1456	5.3	0.5722	10.2	0.2627	2.2	0.3588	8.7
	4	2.2828	31	1.074	16.6	0.5541	6.4	2.7550	19.1
0.5%	2	0.2092	5.0	0.3291	5.8	0.2524	2.6	0.2241	3.4
	3	0.8652	18.5	0.6489	12.3	0.3335	4.8	0.5647	19.7
	4	4.2961	37.2	1.4944	24.6	0.9776	11.4	3.4690	33.4

to obtain the optimal thresholds while increasing the number of the thresholds. Otherwise, the QGA-based multilevel thresholding method can reduce the computational time to several seconds.

5.1. Comparison of the Computation Speed. An experiment is performed to contrast the convergence speed and the

average computational time of the GA, QGA, particle swarm optimization (PSO), and the proposed algorithms. Computation ceased when the value of the objective f_i meets the termination condition: $\text{Dis} = |f_i - f_{\max}|/f_{\max} \leq \varepsilon$. Every algorithm is executed 100 times and yielded the following results presented in Table 3. The letters “ t ” and “ g ” in Table 3 represent the mean computational time and the average

TABLE 4: Comparison of the objective function values with $C = 2, 3, 4$.

Image	C	Objective function values			
		GA	QGA	IQGA	PSO
Cameraman	2	12.1687	12.1687	12.1687	12.1687
	3	15.2174	15.2273	15.2273	15.1733
	4	18.3864	18.3949	18.3955	18.3902
Hunter	2	12.6333	12.6356	12.6356	12.6333
	3	15.8115	15.8020	15.8115	15.8020
	4	18.7101	18.7421	18.7441	18.7374
Baboon	2	12.2167	12.2167	12.2167	12.2167
	3	15.2772	15.2789	15.2801	15.2782
	4	18.1263	18.1278	18.1304	18.1294
Scene	2	12.5244	12.5253	12.5253	12.5248
	3	15.5618	15.5654	15.5654	15.5253
	4	18.3606	18.3631	18.3681	18.3647
Tree	2	12.1671	12.1676	12.1676	12.1676
	3	15.3270	15.3270	15.3270	15.3270
	4	18.0272	18.0300	18.0325	18.0246
F16	2	12.2114	12.2114	12.2114	12.2114
	3	15.5022	15.5038	15.5038	15.5028
	4	18.3082	18.3104	18.3120	18.2972
Lena	2	12.3469	12.3471	12.3471	12.3471
	3	15.3174	15.3174	15.3183	15.3168
	4	17.9775	18.0079	18.0126	18.0061
Peppers	2	12.5311	12.5532	12.5532	12.5532
	3	15.7133	15.8116	15.8258	15.8205
	4	18.7338	18.7338	18.7338	18.7334

TABLE 5: Comparison of the optimal thresholds with $C = 2, 3, 4$.

Image	C	Optimal thresholds			
		GA	QGA	IQGA	PSO
Cameraman	2	128-193	128-193	128-193	128-193
	3	65-129-193	44-104-193	44-104-193	48-115-193
	4	42-96-143-197	44-96-146-197	44-97-146-197	43-96-146-197
Hunter	2	91-174	92-172	91-172	91-174
	3	62-117-174	62-118-173	62-117-174	62-118-174
	4	68-126-177-230	64-120-174-230	62-117-174-230	67-127-173-230
Baboon	2	79-143	79-143	79-143	79-143
	3	44-100-152	44-95-150	44-98-152	44-96-150
	4	33-71-113-160	33-74-115-162	33-74-114-159	33-74-113-159
Scene	2	94-164	92-163	92-163	94-163
	3	76-124-172	73-120-170	73-120-170	76-120-172
	4	72-115-161-198	69-111-160-197	71-113-157-196	69-113-159-197
Tree	2	86-159	86-160	86-160	86-160
	3	79-129-178	79-129-178	79-129-178	79-129-178
	4	75-115-153-191	76-120-160-197	76-118-157-196	76-120-157-193
F16	2	71-173	71-173	71-173	71-173
	3	69-127-184	69-127-183	69-127-183	69-125-183
	4	67-108-148-186	64-103-143-186	67-106-145-185	64-106-147-186
Lena	2	98-164	97-164	97-164	97-164
	3	78-123-165	83-128-176	82-126-175	82-128-176
	4	65-97-137-185	63-96-135-179	64-97-138-179	64-96-135-180
Peppers	2	92-177	93-177	93-179	92-177
	3	73-126-178	74-125-181	73-126-178	73-125-177
	4	46-84-129-179	46-84-130-179	46-84-130-179	45-83-130-179

TABLE 6: Comparison of the value of mean and variance with $C = 2, 3, 4$.

Image	C	GA		QGA		IQGA		PSO	
		Mean	Var.	Mean	Var.	Mean	Var.	Mean	Var.
Cameraman	2	12.1141	$1.1630e-02$	12.1513	$2.0682e-03$	12.1684	$1.5723e-7$	12.1214	$1.1307e-02$
	3	15.0477	$3.8718e-02$	15.1642	$1.1989e-03$	15.2254	$1.0641e-5$	15.0725	$2.1288e-02$
	4	18.1732	$5.3940e-02$	18.2751	$1.1967e-02$	18.3881	$2.9388e-5$	18.2372	$3.9943e-02$
Hunter	2	12.5949	$5.4531e-03$	12.6115	$5.8455e-04$	12.6355	$1.5758e-7$	12.6019	$4.3691e-03$
	3	15.6759	$1.6188e-02$	15.7445	$1.2617e-03$	15.8088	$9.0123e-6$	15.7206	$1.8004e-02$
	4	18.5733	$1.4035e-02$	18.5996	$4.7036e-03$	18.7372	$7.3949e-5$	18.5823	$9.9633e-03$
Baboon	2	12.1810	$7.7419e-04$	12.2109	$4.2185e-05$	12.2165	$2.9464e-8$	12.1907	$5.9346e-04$
	3	15.2305	$2.8331e-03$	15.2550	$5.4095e-04$	15.2785	$2.5557e-6$	15.2394	$2.8012e-03$
	4	17.9545	$2.6872e-02$	18.0479	$4.8961e-03$	18.1219	$2.8246e-5$	17.8392	$2.9964e-02$
Scene	2	12.5163	$1.7481e-05$	12.5214	$1.7481e-05$	12.5252	$2.8823e-8$	12.5170	$1.6997e-05$
	3	15.4991	$5.0432e-03$	15.4989	$1.8712e-03$	15.5629	$5.0827e-5$	15.4889	$4.3875e-03$
	4	18.2521	$1.0527e-02$	18.2742	$3.3150e-03$	18.3587	$1.6565e-4$	18.2604	$9.4887e-03$
Tree	2	12.1499	$1.1541e-03$	12.1644	$6.4341e-06$	12.1674	$3.7896e-8$	12.1532	$7.8016e-04$
	3	15.2076	$1.8796e-02$	15.2626	$2.3912e-03$	15.3252	$2.3026e-6$	15.2360	$1.3356e-02$
	4	17.8916	$1.0394e-02$	17.9567	$2.9546e-03$	18.0293	$4.7901e-6$	17.9001	$1.0041e-02$
F16	2	12.1925	$5.9577e-04$	12.2027	$5.0603e-05$	12.2113	$1.3833e-7$	12.1977	$3.6732e-04$
	3	15.3711	$1.9190e-02$	15.4379	$2.1738e-03$	15.5019	$3.0649e-6$	15.4136	$2.0084e-03$
	4	18.2093	$5.3186e-03$	18.2459	$2.1708e-03$	18.3039	$6.6394e-5$	18.2209	$4.3377e-03$
Lena	2	12.3411	$1.1129e-04$	12.3352	$1.0222e-04$	12.3467	$1.1837e-06$	12.3397	$1.0966e-04$
	3	15.2064	$1.5275e-03$	15.2950	$3.5576e-04$	15.3166	$3.4162e-06$	15.2399	$1.0054e-03$
	4	17.9582	$3.6579e-02$	17.8461	$1.4690e-02$	17.9949	$2.7414e-04$	17.8327	$2.1767e-02$
Peppers	2	12.5311	$1.2613e-03$	12.5410	$5.4765e-04$	12.5521	$3.4546e-06$	12.5395	$9.0061e-04$
	3	15.7133	$1.2829e-02$	15.7728	$2.3938e-03$	15.8259	$5.8941e-05$	15.7420	$1.0766e-02$
	4	18.5535	$1.3629e-02$	18.5664	$1.2948e-02$	18.7338	$1.0004e-04$	18.5591	$1.3310e-02$

convergence generation, respectively. Data in this table is the average of 100 times repeated experiments of the test images. The results show that the convergence generation of the proposed algorithm is much smaller than other methods, and the computational speed of the proposed algorithm is faster than other methods.

5.2. Comparison of the Search Ability and Stability. In order to test the search ability and stability of the proposed algorithm, some experiments have been implemented as follows. The terminal condition of the algorithms is changed to compare the stability and the search ability of the algorithm. Iteration will be terminated when $|\bar{f}^t - \bar{f}^{t-1}| < \varepsilon$ ($\bar{f}^t$ presents the average of the fitness at generation t , $\varepsilon = 0.0001$) or reach to the maximum number of iterations ($\text{max\_gen} = 500$). Since all the compared methods except ES-Kupar in this study are based on stochastic search mechanism, every experiment except ES-Kupar runs 100 times and the performance will be evaluated by the results of the 100-times experiments.

The maximum objective function values and the optimal thresholds with $C = 2, 3, 4$ are shown in Tables 4 and 5, respectively. As can be seen from the tables, QGA and IQGA have obtained all optimal solutions when $C = 2$, while PSO and GA have obtained most optimal solutions; with the increasing numbers of the segmentation threshold, the ability of the PSO and GA decreased sharply, QGA can search most

of the global optimal solutions, and IQGA can obtain all the global optimal solutions. It can be seen that the search ability outperforms the PSO, GA, and QGA. The optimal thresholded images which are obtained by the proposed algorithm are shown in Figure 3.

Table 6 represents the variance and the mean values of the objective functions. It can be seen that the variance values increase with the increasing numbers of the threshold; that is to say, the stability of the algorithms declines with the increasing number of the threshold. In the cases of the variance, IQGA is smaller than other algorithms by more than two orders of magnitude. Utilizing the quantum superposition and coherence, QGA maintains the diversity of the population better than classical GA and PSO, so QGA has a stronger performance than GA. The strategy which introduces cooperative learning method to QGA promotes the local information exchanges among the individuals and enhances the ergodicity of the solutions. Therefore, IQGA has better search performance and stronger stability than QGA.

6. Conclusion

A multilevel thresholding image segmentation method based on IQGA has been proposed in this paper. Combining the merits of QGA and the cooperative learning strategy and adjusting the quantum rotation angle adaptively, the

FIGURE 3: Thresholded images. Images from top to bottom rows are thresholded images with $C = 2, 3, 4$, respectively. (a) Cameraman; (b) Hunter; (c) Baboon; (d) Scene; (e) Tree; (f) F16; (g) Lena; (h) Peppers.

proposed algorithm is employed for several test images. The significant improvement performance of the IQGA has been demonstrated by comparing it with QGA, GA, and PSO for several test images.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work is supported by the project of Sichuan Provincial Education Department under Grant 13za0151.

References

- [1] E. Cuevas, D. Zaldivar, and M. Pérez-Cisneros, "Seeking multi-thresholds for image segmentation with learning automata," *Machine Vision and Applications*, vol. 22, no. 5, pp. 805–818, 2011.
- [2] S. S. Al-Amri, N. V. Kalyankar, and S. Khamitkar, "Image segmentation by using threshold techniques," *Journal of Computing*, vol. 2, no. 5, pp. 83–86, 2010.
- [3] K. Hammouche, M. Diaf, and P. Siarry, "A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation," *Computer Vision and Image Understanding*, vol. 109, no. 2, pp. 163–175, 2008.
- [4] M. Kamel and A. Zhao, "Extraction of binary character/graphics images from grayscale document images," *CVGIP: Graphical Models and Image Processing*, vol. 55, no. 3, pp. 203–217, 1993.
- [5] Q. Liao and Y. Deng, "An accurate segmentation method for white blood cell images," in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 245–248, 2002.
- [6] D. Almeida, J. Folgado, P. Fernandes, and R. Ruben, "Histogram based threshold segmentation of the human femur body for patient specific acquired ct scan images," in *Computational*

Modelling of Objects Represented in Images: Fundamentals, Methods and Applications III, p. 281, 2012.

- [7] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 146–168, 2004.
- [8] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram," *Computer Vision, Graphics, & Image Processing*, vol. 29, no. 3, pp. 273–285, 1985.
- [9] N. Otsu, "Threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [10] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, no. 9, pp. 1277–1294, 1993.
- [11] E. Cuevas, F. Sención, D. Zaldivar, M. Pérez-Cisneros, and H. Sossa, "A multi-threshold segmentation approach based on artificial bee colony optimization," *Applied Intelligence*, vol. 37, no. 3, pp. 321–336, 2012.
- [12] P.-Y. Yin and L.-H. Chen, "A fast iterative scheme for multilevel thresholding methods," *Signal Processing*, vol. 60, no. 3, pp. 305–313, 1997.
- [13] P.-Y. Yin, "Multilevel minimum cross entropy threshold selection based on particle swarm optimization," *Applied Mathematics and Computation*, vol. 184, no. 2, pp. 503–513, 2007.
- [14] W.-B. Tao, J.-W. Tian, and J. Liu, "Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm," *Pattern Recognition Letters*, vol. 24, no. 16, pp. 3069–3078, 2003.
- [15] P.-Y. Yin, "A fast scheme for optimal thresholding using genetic algorithms," *Signal Processing*, vol. 72, no. 2, pp. 85–95, 1999.
- [16] C. C. Lai and D. C. Tseng, "A hybrid approach using Gaussian smoothing and genetic algorithm for multilevel thresholding," *International Journal of Hybrid Intelligent Systems*, vol. 1, no. 3–4, pp. 143–152, 2004.
- [17] L. Cao, P. Bao, and Z. Shi, "The strongest schema learning GA and its application to multilevel thresholding," *Image and Vision Computing*, vol. 26, no. 5, pp. 716–724, 2008.
- [18] Z.-H. Yang, Z.-B. Pu, and Z.-Q. Qi, "Relative entropy multilevel thresholding method based on genetic optimization," in *Proceedings of the International Conference on Neural Networks and Signal Processing (ICNNSP '03)*, vol. 1, pp. 583–586, December 2003.
- [19] P. G. Espejo, S. Ventura, and F. Herrera, "A survey on the application of genetic programming to classification," *IEEE Transactions on Systems, Man and Cybernetics C: Applications and Reviews*, vol. 40, no. 2, pp. 121–144, 2010.
- [20] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '96)*, pp. 61–66, May 1996.
- [21] K.-H. Han and J.-H. Kim, "Genetic quantum algorithm and its application to combinatorial optimization problem," in *Proceedings of the Congress on Evolutionary Computation*, vol. 2, pp. 1354–1360, July 2000.
- [22] K.-H. Han and J.-H. Kim, "Quantum-inspired evolutionary algorithms with a new termination criterion, He gate, and two-phase scheme," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, pp. 156–169, 2004.
- [23] K.-H. Han and J.-H. Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 6, pp. 580–593, 2002.
- [24] G. X. Zhang and H. Rong, "Real-observation quantum-inspired evolutionary algorithm for a class of numerical optimization problems," in *Computational Science—ICCS 2007*, pp. 989–996, Springer, 2007.
- [25] J.-S. Jang, K.-H. Han, and J.-H. Kim, "Face detection using quantum-inspired evolutionary algorithm," in *Proceedings of the Congress on Evolutionary Computation (CEC '04)*, vol. 2, pp. 2100–2106, June 2004.
- [26] K.-H. Kim, J.-Y. Hwang, K.-H. Han, J.-H. Kim, and K.-H. Park, "A quantum-inspired evolutionary computing algorithm for disk allocation method," *IEICE Transactions on Information and Systems*, vol. 86, no. 3, pp. 645–649, 2003.
- [27] H. Talbi, M. Batouche, and A. Draao, "A quantum-inspired evolutionary algorithm for multiobjective image segmentation," *International Journal of Mathematical, Physical and Engineering Sciences*, vol. 1, pp. 109–114, 2007.
- [28] K. Benatchba, M. Koudil, Y. Boukir, and N. Benkhelat, "Image segmentation using quantum genetic algorithms," in *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics (IECON '06)*, pp. 3556–3562, November 2006.
- [29] G. X. Zhang, L. Z. Hu, and W. D. Jin, "Resemblance coefficient and a quantum genetic algorithm for feature selection," in *Discovery Science*, pp. 155–168, 2004.
- [30] G. Zhang, W. Jin, and L. Hu, "Quantum evolutionary algorithm for multi-objective optimization problems," in *Proceedings of the IEEE International Symposium on Intelligent Control*, pp. 703–708, October 2003.
- [31] Y. Kim, J.-H. Kim, and K.-H. Han, "Quantum-inspired multiobjective evolutionary algorithm for multiobjective 0/1 knapsack problems," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 2601–2606, July 2006.
- [32] G. Zhang, W. Jin, and L. Hu, "A novel parallel quantum genetic algorithm," in *Proceedings of the 4th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT '03)*, pp. 693–697, August 2003.
- [33] J. G. Vlachogiannis and K. Y. Lee, "Quantum-inspired evolutionary algorithm for real and reactive power dispatch," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1627–1636, 2008.
- [34] L. M. Huang, "Image segmentation based on quantum genetic algorithm," *Computer Applications and Software*, vol. 26, no. 9, pp. 247–249, 2007.
- [35] H. Yu and J. Fan, "Parameter optimization based on quantum genetic algorithm for generalized fuzzy entropy thresholding segmentation method," in *Proceedings of the 5th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '08)*, vol. 1, pp. 530–534, October 2008.
- [36] G. Zhang, "Quantum-inspired evolutionary algorithms: a survey and empirical study," *Journal of Heuristics*, vol. 17, no. 3, pp. 303–351, 2011.
- [37] G.-X. Zhang, N. Li, W.-D. Jin, and L.-Z. Hu, "Novel quantum genetic algorithm and its applications," *Frontiers of Electrical and Electronic Engineering in China*, vol. 1, no. 1, pp. 31–36, 2006.
- [38] G. X. Zhang and H. Rong, "Improved quantum-inspired genetic algorithm based time-frequency analysis of radar emitter signals," in *Rough Sets and Knowledge Technology*, pp. 484–491, Springer, 2007.
- [39] F. van den Bergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, 2004.
- [40] H. G. Cobb, "Is the genetic algorithm a cooperative learner?" DTIC Document, 1995.

- [41] S. H. Clearwater, T. Hogg, and B. A. Huberman, "Cooperative problem solving," *Computation: The Micro and the Macro View*, pp. 33–70, 1992.
- [42] M. Potter and K. de Jong, "A cooperative coevolutionary approach to function optimization," in *Parallel Problem Solving from Nature—PPSN III*, pp. 249–257, 1994.
- [43] L. C. Jiao, *Advances in Natural Computation, Machine Learning and Image Understanding*, Xidian Press, 2008.