

Image Segmentation Using Thresholding and Swarm Intelligence

Zhiwei Ye

Hubei University of Technology / School of Computer Science, Wuhan, China
Email: weizhiye121@163.com

Zhengbing Hu

Huazhong Normal University / Department of Information Technology Wuhan, China

Xudong Lai

Wuhan University / School of Remote Sensing & Information Engineering, Wuhan, China
Email: laixudong@whu.edu.cn

Hongwei Chen

Hubei University of Technology / School of Computer Science, Wuhan, China

Abstract—Image segmentation is a significant technology for image process. Many segmentation methods have been brought forward to deal with image segmentation, among these methods thresholding is the simple and important one. To overcome shortcoming without using space information many thresholding methods based on 2-D histogram are often used in practical work. These methods segment images by using the gray value of the pixel and the local average gray value of it, and thus provide better results than the methods based on 1-D histogram. However, its time-consuming computation is often an obstacle in real time application systems. In this paper, fast image segmentation methods based on swarm intelligence and 2-D Fisher criteria thresholding are presented. The proposed approaches have been implemented and tested on several real images. Experiments results indicate that the proposed methods provides improved search performance which are efficient methods to help select optimum 2D thresholds with much less computation cost and suitable for real time applications.

Index Terms— Image Segmentation, Thresholding, Swarm Intelligence; 2-D Fisher Criteria

I. INTRODUCTION

Image segmentation denotes a process by which a raw input image is partitioned into non-overlapping regions such that each region is homogeneous and the union of two adjacent regions is heterogeneous, which is one of the most important and most difficult low-level image analysis tasks and has received a great deal of attention. Several general-purpose algorithms and techniques have been developed for image segmentation, such as clustering methods, histogram-based methods, edge detection, region growing methods, graph partitioning methods etc. [1-10].

In many applications of image processing, the gray levels of pixels belonging to the object are substantially different from the gray levels of the pixels belonging to the background. Thresholding becomes a simple but effective tool for image segmentation for its simplicity, especially in the fields where real time processing is needed. The key of this method is to select the threshold value (or values when multiple-levels are selected). Several popular methods are used in industry including the maximum entropy method, Otsu's method (maximum variance), and et al. k-means clustering can also be used[1-7].

Usually, threshold selection is performed by using histogram. In an ideal case, the histogram has a deep and sharp valley between two peaks representing objects and background respectively. The gray level at the bottom of the valley is selected as appropriate threshold. Researches show that the segmentation performance of most thresholding methods are influenced by the shape of histogram of images to be segmented, especially if the histogram of images in reality have no distinct sharp valleys or the valley is flat and broad. One way to overcome this shortcoming is to take spatial relationship of pixel as well as gray level of pixel's into account. To this end, Abutaleb proposed a 2-D entropic thresholding method based on 2-D histogram [3]. But one primary problem associated with 2-D histogram based thresholding method is that the computation burden to find 2-D threshold vector is very large.

In common, most of all existing automatic selection schemes are to construct different target function to measure segmentation performance, then search target function and take gray value which make the function extreme as optimal thresholds[1-12]. To solve this problem, fast recursive algorithm and evolutionary algorithm had been proposed in [11-12]. However, the automatic selection of a robust, optimum threshold that separates different objects or separates an object from background has remained a challenge. Swarm

Manuscript received January 1, 2011; revised June 1, 2011; accepted July 1, 2011

Corresponding author, Xudong Lai

intelligence (SI) is the collective behavior of decentralized, self-organized systems, natural or artificial. Nowadays, Swarm Intelligence-based techniques are used in a number of applications and have got good performance [13-19]. In this paper, we proposed to use two typical swam intelligence algorithms to find two dimensional threshold pair for 2-D Fisher criterion function thresholding method.

The reminder of this paper is organized as follows. In section 2, for the integrity of this paper, we simply describe the object segmentation method based on 2-D Fisher linear optimal discriminant analysis. How to use particle swarm optimization algorithm and artificial bee colony algorithm to find the optimal combination of threshold pair is presented in section 3. In section 4, we evaluate the performance of the proposed thresholding approach using some real images and compare it with existing techniques. Experimental results and conclusion remarks are given in section 5.

II. TWO-DIMENSIONAL FISHER IMAGE SEGMENTATION ALGORITHM

Fisher's linear discriminant is a classification method that projects high-dimensional data onto a line and performs classification in this one-dimensional space. It aims to search for a linear transform that reduces the dimension of a given n -dimension statistical model, consisting of K classes, to d ($d \leq n$) dimensions.

The transform should be such that a maximum amount of discrimination information is preserved in the lower dimensional model and it tries to preserve distance of already well separated classes, which may result in a large overlap or even occlusion of neighboring classes, see Fig.1. The class information is condensed in two scatter matrices, i.e., the between-class scatter and within-class scatter. The fisher criterion is defined as

$$J(w) = \frac{|m_1 - m_2|^2}{s_1^2 + s_2^2} \quad (1)$$

where m represents a mean, s^2 represents a variance, and the subscripts denote the two classes. The transformation L is obtained by maximizing (1), which implies that the between-class scatter is maximum while the within-class scatter is minimum and the discrimination is optimal.

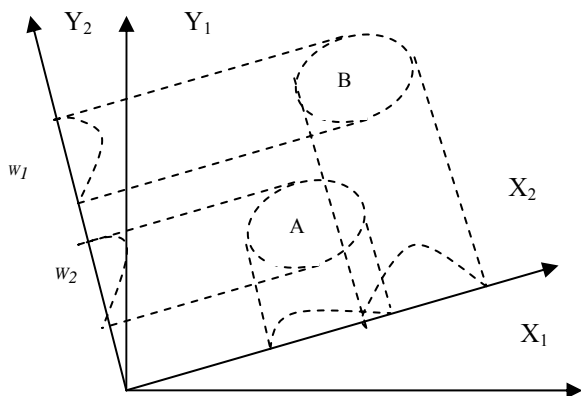


Figure 1. The projection of 2-D feature vector

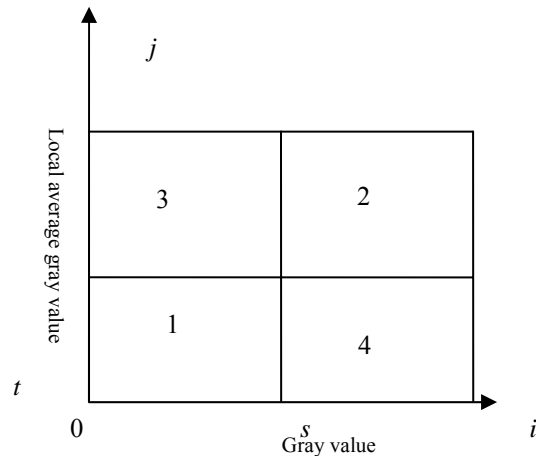


Figure 2. Quadrants in 2-D histogram

This method has strong parallels to linear perceptrons. In the paper, Fisher linear optimal discriminant analysis function is employed to select 2-D threshold [12], details are as follow.

Let F be an image of size $M \times N$ and $f(x, y)$ be the gray value of the pixel located at (x, y) . Define the value of 2-D histogram $P(i, j)$ as the pixel gray value $f(x, y) = i$ and the average neighborhood pixel gray value as $g(x, y) = j, (i, j = 0, 2, \dots, L)$. Supposed the gray-scale of the image is L ($L = 0, 1, \dots, 255$). In a $k \times k$ neighborhood window, calculate the local average gray value $g(x, y)$ of the pixel located at (x, y) for the image, $f(x+m, y+n)$ denotes gray value of around point (x, y) , which can be expressed by(2).

$$g(x, y) = \frac{1}{k^2} \sum_{m=-k/2}^{k/2} \sum_{n=-k/2}^{k/2} f(x+m, y+n) \quad (2)$$

where $1 \leq x \leq M, 1 \leq y \leq N$, M and N are the width and height of the image, generally $k=3$. Let r denote the integer part of the number r . The gray value $f(x, y)$ and local average gray value $g(x, y)$ are used to form a pair. Let r_{ij} be the frequency of pair (i, j) , where $f(x, y) = i$ and $g(x, y) = j$ then

$$P_{ij} = \frac{r_{ij}}{M \times N}, 1 \leq i \leq L, 1 \leq j \leq L \quad (3)$$

is 2-D histogram of image F , 2-D histogram is a matrix of size $L \times L$ which can be depicted as in Fig. 2.

By means of threshold pair (s, t) , where s is a threshold for gray value and t is a threshold for local average gray value, 2-D histogram can be divided into 4 quadrants. Quadrants 1 and 2 represent the distributions of background and object class while quadrants 3 and 4 represent the distributions of pixels near edges and noise. Because of homogeneity, pixels interior to the objects or the background contribute mainly to the near-diagonal elements while the pixels of edges and noise should

contribute to the off-diagonal elements. In common, the contribution of quadrants 3 and 4 can be negligible. Hence, it is reasonable to assume that $p_{ij} \approx 0$ for $i = s+1 \dots L$; $j = 1, \dots, t$ and $i = 1, \dots, s$; $j = t+1, \dots, L$. Here, quadrants 1 and 2 are considered, which correspond to object and background.

Project on two coordinates of the 2-D histogram, expressed as $H(i)$ and $W(j)$ respectively, the mean and variance of the 2-D histogram of Fisher criterion can be given by following equations.

$$\begin{aligned} \mu_0 &= (\mu_0^i, \mu_0^j) & \mu_1 &= (\mu_1^i, \mu_1^j) & (4) \\ \sigma_0^2 &= (\sigma_{0i}^2, \sigma_{0j}^2) & \sigma_1^2 &= (\sigma_{1i}^2, \sigma_{1j}^2) & (5) \end{aligned}$$

Where

$$\mu_0^i = \frac{\sum_{i=0}^s i * H(i)}{\sum_{i=0}^s H(i)} \quad (6)$$

$$\mu_0^j = \frac{\sum_{j=0}^t j * W(j)}{\sum_{j=0}^t W(j)} \quad (7)$$

$$\mu_1^i = \frac{\sum_{i=s+1}^L i * H(i)}{\sum_{i=s+1}^L H(i)} \quad (8)$$

$$\mu_1^j = \frac{\sum_{j=t+1}^L j * W(j)}{\sum_{j=t+1}^L W(j)} \quad (9)$$

$$\sigma_{0i}^2 = \sum_{i=0}^s (i - \mu_0^i)^2 * H(i) \quad (10)$$

$$\sigma_{0j}^2 = \sum_{j=0}^t (j - \mu_0^j)^2 * W(j) \quad (11)$$

$$\sigma_{1i}^2 = \sum_{i=s+1}^L (i - \mu_1^i)^2 * H(i) \quad (12)$$

$$\sigma_{1j}^2 = \sum_{j=t+1}^L (j - \mu_1^j)^2 * W(j) \quad (13)$$

$$H(i) = \sum_{j=0}^{L-1} N(i, j), i = 0, 1, \dots, L-1 \quad (14)$$

$$W(j) = \sum_{i=0}^{L-1} N(i, j), j = 0, 1, \dots, L-1 \quad (15)$$

Thus 2-D Fisher criterion function is defined below:

$$J_F(s, t) = \frac{([\mu_0^i, \mu_0^j] - [\mu_1^i, \mu_1^j]) * ([\mu_0^i, \mu_0^j] - [\mu_1^i, \mu_1^j])^T}{\sigma_{0i}^2 + \sigma_{0j}^2 + \sigma_{1i}^2 + \sigma_{1j}^2} \quad (16)$$

Optimal threshold pair (s^*, t^*) can be obtained by maximizing (17).

$$(s, t) = \text{ArgMax}(J_F(s, t)) \quad (17)$$

No matter what level of the image's signal-to-noise ratio, 2-D histogram often has two distinct peaks. When select a suitable 2-D threshold (s, t) pair which separates the object from the background, we can get better segmentation results, which has a strong anti-noise capability of 2-D histogram.

III. SWARM INTELLIGENCE APPROACH FOR IMAGE THRESHOLDING

As a matter of fact, the 2-D Fisher segmentation model presented before is essentially a function optimization problem; the inward of it is a process to pursue the maximum 2-D Fisher criterion function, which can be viewed as an optimization problem. Therefore, we can use swarm intelligence algorithm to optimize the 2-D Fisher principle. In this section, artificial bee colony approach and particle swarm optimization approach are employed to deal with it.

A. Artificial bee colony approach

Artificial Bee Colony Algorithm (ABC) is an optimization algorithm based on the intelligent foraging behaviors of honey bee swarm, proposed by Karaboga in 2005[13]. It is an optimization algorithm based on particular intelligent behavior of honey bee swarms. Since 2005, D. Karaboga and other research group have been studying the ABC algorithm and its applications to real world problems [13-18].

In ABC model, the colony consists of three groups of bees: employed bees, onlookers and scouts. It is assumed that there is only one artificial employed bee for each food source [13-15]. In other words, the number of employed bees in the colony is equal to the number of food sources around the hive. Employed bees go to their food source and come back to hive and dance on this area. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source. Onlookers watch the dances of employed bees and choose food sources depending on dances. The main ideas and steps of the proposed method are as following [17-18].

Each food source is a possible solution for the 2D-Fisher function problem and the nectar amount of a food source represents the quality of the solution represented by the fitness value. At the first step, the ABC generates a randomly distributed initial population of SN solutions, where SN is the number of food sources and it is equal to the number of employed bees. Each solution x_i ($i = 1, 2, \dots, SN$) is a n -dimensional vector and n is the number of optimization parameters, here n is equal to 2.

In ABC for 2D-Fisher thresholding, based on (16), the fitness function is defined as below:

$$fit_i = J_F(s, t) \quad (18)$$

where $J_F(s, t)$ is the objective function value of solution i , the objective function can be calculated by mapping the position of food source into threshold vector, fit_i is the fitness value of solution i . After initialization, the

population of the positions (solutions) is subjected to repeated cycles of the search processes of the employed bees, the onlooker bees and the scout bees. Maximum cycle number MCN is one of the four control parameters in the ABC algorithm.

In each iteration, every employed bee determines a food source in the neighborhood of its current food source and evaluates its nectar amount (fitness). The i -th food source position is represented as $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$. $F(X_i)$ refers to the nectar amount of the food source located at X_i . After watching the dancing of employed bees, an onlooker bee goes to the region of food source at X_i with the probability:

$$p_i = \frac{fit_i}{\sum_{j=1}^{NS} fit_j} \quad (19)$$

In order to produce a candidate food position (threshold pair) from the old one in memory, the ABC uses the following expression:

$$v_{ij} = x_{ij} + \phi_{ij} (x_{ij} - x_{kj}) \quad (20)$$

Where $k \in \{1, 2, \dots, SN\}$ and $j \in \{1, 2, \dots, n\}$ are randomly chosen indexes; k has to be different from i and ϕ_{ij} is a random number in the range $[-1, 1]$.

After each candidate source position v_{ij} is produced and then evaluated by the artificial bee, its performance is compared with that of its old one and a greedy selection mechanism is employed as the selection operation between the old and the new candidate. Otherwise, if the new food source has equal or better quality than the old source, the old one in memory is replaced by the new one.

In ABC algorithm, providing that a position cannot be improved further through a predetermined number of cycles, the related food source is assumed to be abandoned. The value of predetermined number of cycles is an important control parameter of the ABC algorithm, which is called "limit for abandonment". Assuming that the abandoned source is $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$, the scout discovers a new food source to replace X_i . This operation can be defined as:

$$x_{ij} = l_i + \delta(u_i - l_i) \quad (21)$$

where $j \in \{1, 2, \dots, n\}$, l_i and u_i are the lower and upper bound of the parameter x_{ij} and δ is a random number in the range $[0, 1)$. Where l_i and u_i are lower and upper bounds of the variable X_i , in this paper, l_i is 0 and u_i is 255.

It can be concluded from the above explanation that there are four control parameters used in the ABC: the number of food sources which is equal to the number of employed or onlooker bees (SN), the value of limit, the maximum cycle number (MCN).

The pseudo code of the ABC algorithm is showed in Fig.3.

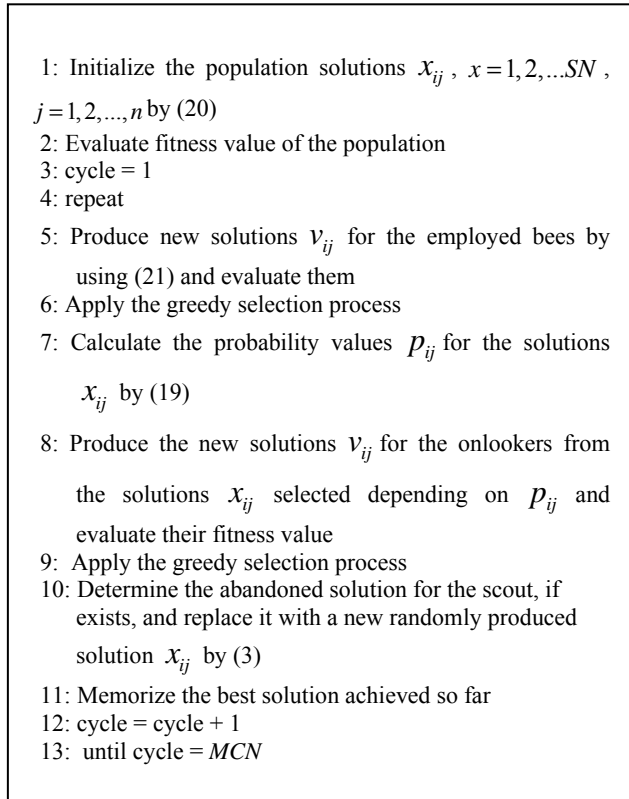


Figure 3. The pseudo code of ABC

B. Particle Swarm Optimization approach

Particle swarm optimization algorithm (PSO) is an evolutionary computation technique developed by Kennedy and Eberhart in 1995 [19], motivated by social behavior of organisms such as bird flocking and fish schooling. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. Compared to GA, the advantages of PSO are that PSO is easy to implement and there are few parameters to adjust. PSO has been successfully applied in many fields: function optimization, artificial neural network training, fuzzy system control, and other areas where GA can be applied [19-22].

As stated before, PSO simulates the behaviors of bird flocking. Suppose the following scenario: a group of birds are randomly searching food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they do know how far the food is on iteration. So what's the best strategy to acquire the food? The effective one is to follow the bird that is nearest to the food.

PSO learned from the scenario and used it to solve the optimization problems. In PSO, each single solution is a "bird" in the search space. We call it "particle". All of

particles have fitness values which are evaluated by the fitness function to be optimized, and have velocities which direct the flying of the particles. The particles fly through the problem space by following the current optimum particles.

Supposed that D is the dimension of the search space, a search group is constituted by m particles. The i -th particle is a vector $X = (x_{i1}, x_{i2}, \dots, x_{id})$, $i = 1, 2, \dots, m$ in D dimension search space. PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. Each particle's movement is influenced by its local best known position and is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions. That is, at each iteration, each particle is updated by following two "best" values. The first one is the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called $p_{best[]} = (p_{i1}, p_{i2}, \dots, p_{id})$. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the population. This best value is a global best and called $G_{best[]} = (p_{g1}, p_{g2}, \dots, p_{gd})$. After finding the two best values, the particle updates its velocity and positions with following (22) and (23).

$$v_{id}(t+1) = wv_{id}(t) + c_1r_1(p_{id}(t) - x_{id}(t)) + c_2r_2(p_{gd}(t) - x_{id}(t)) \quad (22)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (23)$$

Where t is the number of the current time step, w is the inertia weight, c_1 and c_2 are the learning factors,

$x_{id}(t)$ is the position of the particle i at time t . It has D coordinates.

$v_{id}(t)$ is the velocity at time t . It has D components.

$p_{id}(t)$ is the previous best position, at time t . It has D coordinates.

$p_{gd}(t)$ is the best previous best position found in the neighbourhood. It has D coordinates.

r_1 and r_2 are random numbers between (0,1). c_1 and c_2 are positive constant parameters, usually $c_1 = c_2 = 2$.

Particles' velocities on each dimension are clamped to a maximum velocity V_{max} . If the sum of accelerations would cause the velocity on that dimension to exceed V_{max} , which is a parameter specified by the user, then the velocity on that dimension is limited to V_{max} .

The main idea of using PSO to search the best thresholding pair of 2-D Fisher function is as below.

The initial population is generated with N number of solutions and each solution is a D -dimension vector, here D is set to 2 that each solution represents 2-D candidate threshold. X_i represents the i -th particle position in the population which denotes a candidate threshold pair and

its fitness can be measure by 2-D Fisher function. The processes of standard particle swarm optimization algorithm for 2D-Fisher threshold method are as follow:

- 1: Initialize population of particles with random positions from $[0,255]$ and velocities between $[-V_{max}, V_{max}]$, set the maximum iterations N and let $t = 0$;
- 2: Evaluate each particle's fitness value. Calculate the objective function of each particle;
- 3: Update optimal value of individual particle according to its current fitness;
- 4: Update global best among all particles according the fitness of the swarms;
- 5: Change the velocity of each particle using (22);
Move each particle to its new position using (23);
- 6: Repeat until reach the condition of end, otherwise go to step2.
- 7: Output the best position, that is, the optimal threshold value pair (s, t) .

The whole process can be summarized in Fig.4.

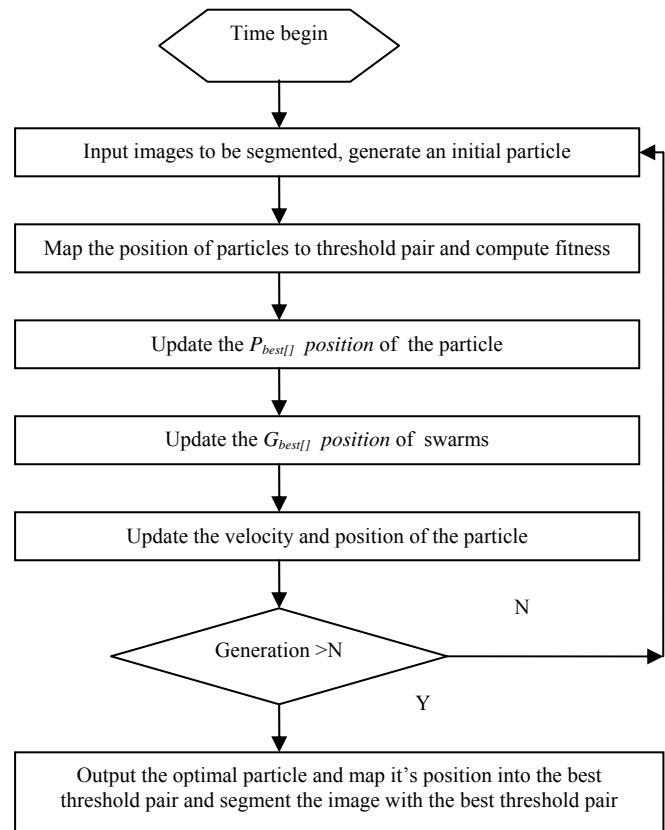


Figure.4 The flowchart of PSO

However, standard particle swarm algorithm has its own limitations, the quality of initial solution can not be guaranteed, the convergence velocity is slow in the later search, and itself is easy to fall into local optimum. For these shortcomings, the researchers have made a series of improved algorithm, in which chaos particle swarm algorithm can well make up for these shortages of classical particle swarm optimization[23-24]. Among

these improved algorithms, chaos particle swarm optimization (CPSO) is a relatively simple and effective algorithm [24]. Therefore, to enhance search efficiency chaos particle swarm optimization algorithm is employed to seek best threshold pair of 2D-Fisher function too. The main idea of CPSO is as follow.

Generally the randomness motion state obtained by a deterministic equation is called chaos, a variable which shows the chaotic state is called a chaotic variable. Logistic equation is a typical chaotic system.

$$z_{i+1} = \mu z_i(1 - z_i) \quad i = 0, 1, 2, \dots \quad \mu \in (2, 4) \quad (24)$$

where μ is the control parameter and (24) can be regarded as a dynamic system. Once the μ value is determined, an arbitrary initial value $z \in [0, 1]$, can iterate a determined time series. A chaotic variable within a certain range has the following characteristics: Randomness, that is, it performs as well as a random variable; ergodicity, that is, it may go through all the state space without repeat; Regularity, that is, the variable is decided by a determined equation. Chaos optimization method is a novel optimization method. It uses the features of chaotic systems to achieve the global optimum, and it does not require the objective function has the nature of continuity and differentiability. Introducing the chaos ideas to particle swarm optimization can well overcome the shortcomings of classical particle swarm optimization, and find the solution in a faster and better way.

Steps of Chaos Particle Swarm Optimization algorithm is as Fig.5.

IV. EXPERIMENT AND RESULT

To show effectiveness of the proposed method, some real images (Each image includes distinct object and the background, and the object can be exactly distinguished from the background by some suitable threshold) are used to evaluate the performance of the proposed method; moreover, the segmentation performance is compared with some commonly used algorithms presented in the literature, that is Otsu method, 1-D Fisher criterion function method, 2-D entropy method.

Three images for segmentation are illustrated in Fig. 6, 7 and 8. In this paper, Fig.6 (a), Fig.7 (a) and Fig. 8 (a) are three original images. Owing to the segmentation performance sensitive to the images' brightness and contrast, in order to repeat the experiments to testify the method in this paper in other occasions the three selected image's histograms are shown in Fig.6 (b), Fig.7 (b) and Fig. 8 (b).

As mentioned in [25], in a general way, as for a gray level image, the quasi threshold often varies a small value around the optimal threshold. Therefore, to avoid falling into local optimal threshold a fluctuant threshold A is imposed in our test. That is, after main circle ends we get a quasi pair of threshold (s, t) . Then local search is performed, that we change (s, t) in the rang of $(([s - A], [s + A]), ([t - A], [t + A]))$ and compute 2-D Fisher function to get the optimal pair of threshold.

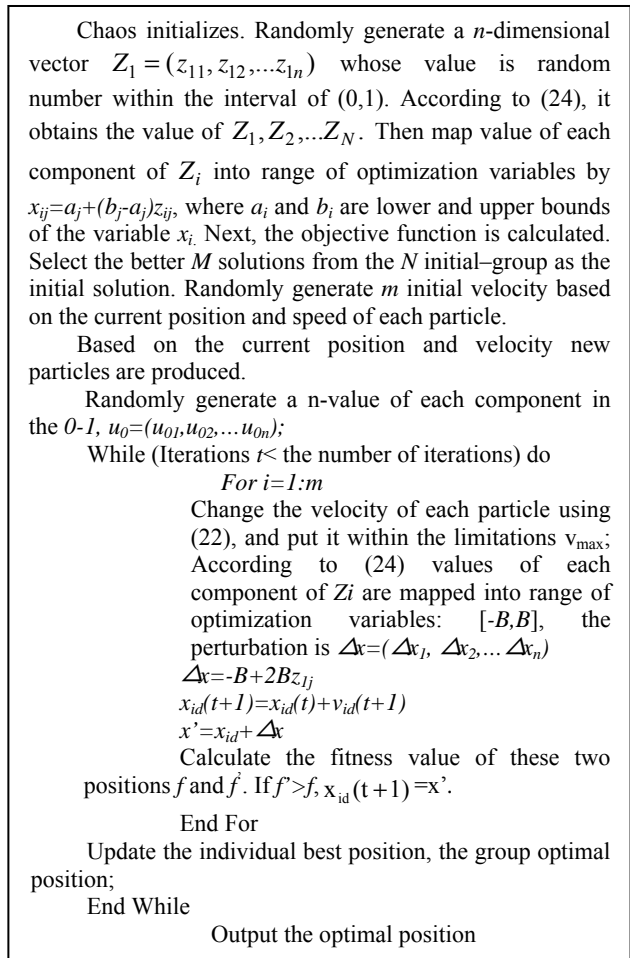


Figure.5 procedure of CPSO

In the implementation of the proposed methods, the following parameters are used. In PSO and CPSO, the parameters of the population size of PSO is $M = 20$, $c_1 = c_2 = 2$ the inertia weight w is set to from 1 to 0.4 in the way of linear decline, the V_{\max} is set to 10, the maximum number of iteration is 80, A is set 10. In CPSO, the number of initialization particles is 40, after the first iteration, the number of particles is 20, the maximum number of iteration is 20, chaos control parameter $\mu = 4$, $B = 5$. ABC algorithm has a few control parameters: Maximum number of cycles (MCN) equals to the maximum number of generation and the colony size equals to the population size, for the test problems carried out in this work, the colony size of 30 can provide an acceptable convergence speed for search, for the proposed problem, the MCN is set to 50. The percentages of onlooker and employed bees were 50% of the colony, the value of "limit" is set to 10. So, the total objective function evaluation number of PSO is $80 * 20 + 20 * 20 = 2000$, CPSO is $20 * 40 + 400 + 40 = 1240$; and ABC is $30 * 50 + 30 / 2 + 400 = 1915$.

The experimental results of three examples are shown in Figs.6-8. Each figure shows the original gray-level image, the thresholding images using the proposed as well as other methods used in the comparison. It can be

seen that Fig. 6-8(c) are the thresholded images with 1-D Fisher criterion method, Fig.6-8(d) are the thresholded images with 1-D Otsu method; Fig.6-8(e) are the thresholded images with 2-D entropy method. Fig.6-8(f) are the segmentation images using the proposed method, these segmented images are better than those of other three thresholding methods in visual effect. Table1 lists the optimal threshold values that are found for these images in four different methods. It can conclude that the 2-D Fisher criterion method has better performance than the other three approaches.

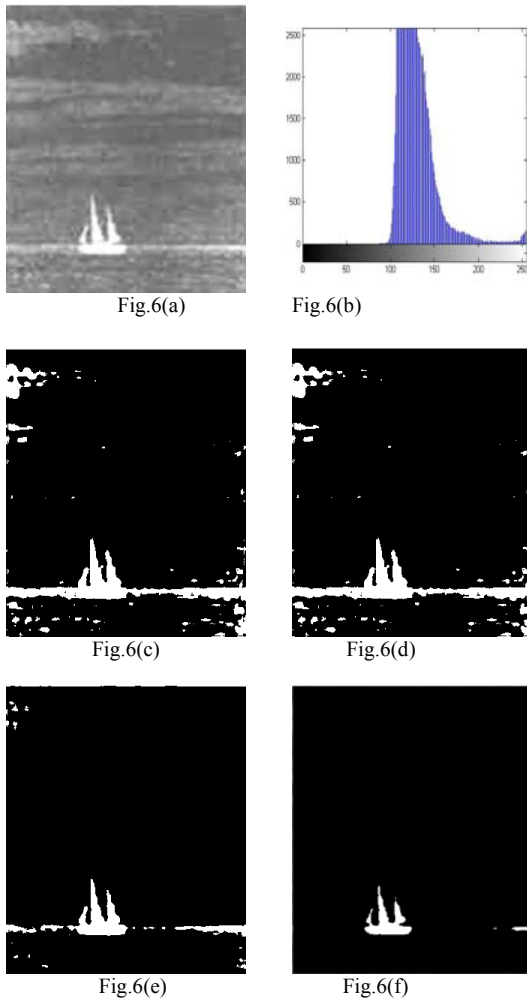


Figure. 6 Top row, from left to right: original image1 (320x386), 1-D histogram of the image. Middle row, from left to right: thresholding results by 1-D Fisher method (T = 162), results by Otsu method (T = 160). Bottom row, from left to right: thresholding results by 2-D entropy method (s = 183,t =157), by the proposed method(s=225,t = 224).

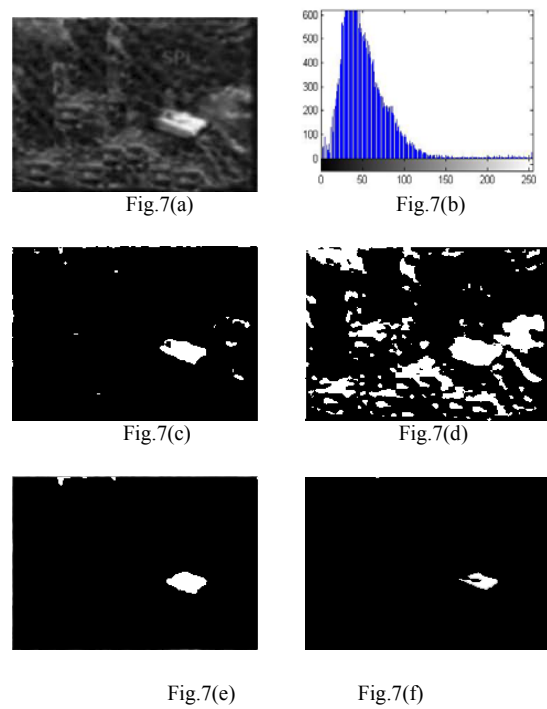


Figure. 7 Top row, from left to right: original image3 (224x156), 1-D histogram of the image. Middle row, from left to right: thresholding results by 1-D Fisher method (T = 121), results by Otsu method (T = 72). Bottom row, from left to right: thresholding results by 2-D entropy method (s = 127,t =105), by the proposed method(s=200,t = 199).

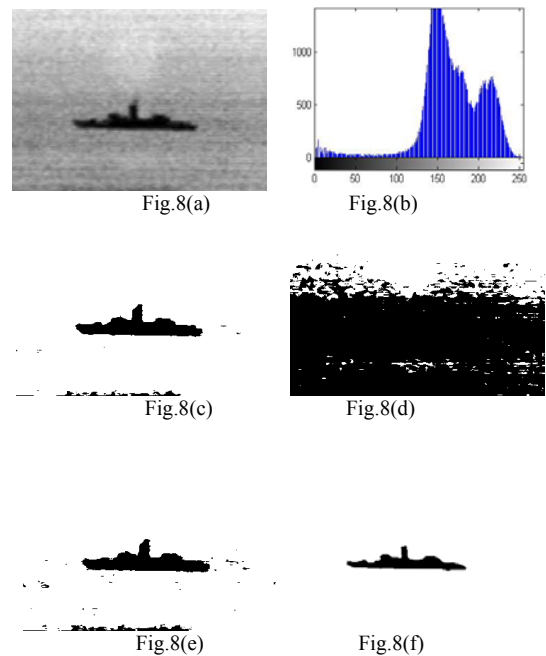


Figure. 8 Top row, from left to right: original image2 (320x256), 1-D histogram of the image. Middle row, from left to right: thresholding results by 1-D Fisher method (T = 117), results by Otsu method (T = 173). Bottom row, from left to right: thresholding results by 2-D entropy method (s = 125,t =122), by the proposed method(s=56,t = 53).

TABLE I. COMPARISONS OF THRESHOLD FOR THE IMAGES IN FIGS

Figure	Thresholding algorithms			
	proposed	1-D Fisher	1-D Otsu	2-Dentropy
1	225 224	162	160	183 157
2	200 199	121	72	127 105
3	56 53	117	173	125 122

TABLE II. COMPARISON OF THE SI-BASED AND EXHAUSTIVE SEARCH ALGORITHM

Method image	Fisher function computation times			
	PSO-based	CPSO-based	ABC-based	Exhaustive
1	2000	1200	1915	256*256
2	2000	1200	1915	256*256
3	2000	1200	1915	256*256

Although 2-D Fisher criterion method has better performance than 1-D Fisher method, it needs much more computing time than that of 1-D Fisher method. In this paper, PSO, CPSO and ABC are used to find the best 2-D threshold pair. Each method proposed in this paper is carried out 40 times. The exhaustive search method is also conducted for deriving the optimal solutions for comparison. Table 2 shows the 2-D Fisher criterion threshold derived by the SI-based algorithm and the optimal threshold by the exhaustive search method. As is shown in table2, the optimal pair of thresholds obtained by the proposed methods has no difference to corresponding optimal thresholds acquired by the exhaustive search method. In contrast, all maximum times of 2-D Fisher function computation by ABC and PSO is below 2000, while the enumeration algorithm need 256*256 times of computation. It is known that the most time consumed in 2-D Fisher thresholding method is to calculate 2-D Fisher function, hence, it is obvious that our method can reduce the computation time and enhance the efficiency of old 2-D Fisher thresholding method; moreover, CPSO algorithms has better performance than that of PSO and standard ABC. Anyhow, the computation time used for the SI-based algorithm is much less than that of exhaustive search algorithm. Hence, the proposed SI-based method is more effective in finding 2-D histogram threshold for image analysis.

V. CONCLUSION

The automatic selection of a robust, optimum threshold to separate different objects, or to separate an object from the background, has remained a challenge in image segmentation. The 2-D histogram based thresholding such as 2D-Fisher method is a good method to do image segmentation, but it gives rise to the exponential increment of computation time in image segmentation. In this paper, we have investigated the object segmentation performance of the 2D-Fisher method. Furthermore, we compare the proposed 2-D Fisher segmentation method with the present typical object segmentation approaches

based on 2-D entropy and Otsu. The results of the performance evaluation using a number of examples show that the proposed 2-D Fisher segmentation method provides very good segmentation performance among all compared thresholding method. Additionally, we have designed PSO, CPSO and ABC strategy to find the optimal threshold pair of 2-D histogram segmentation method. The experiment results show that the implementation of the proposed 2-D Fisher by proposed ABC, PSO has more highly effective search performance than exhaustive search method especially the CPSO has best performance in this three approaches and therefore, is suitable for real-time image analysis and vision applications.

ACKNOWLEDGMENT

This work is supported by Natural Science Foundation of Hubei Province of China (2009CDB100), (2011CDB075),(2011CDB155) and doctor fund of Hubei University of Technology and

REFERENCES

- [1] Sahoo, P.K., Solutani, S., Wong, A.K.C., 1988. A survey of thresholding techniques. *Comput. Vision Graphics Image Process.* 41, pp.233–260.
- [2] Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, *Digital Image Processing Using*. Prentice Hall, 2002, pp.567 -634
- [3] S. Abutaleb. 1989. Automatic Thresholding of Gray-Level Pictures Using Two-Dimensional Entropy[J]. *Comput. Vision Graphics Image Process*, 47, pp.22-32
- [4] Pal, N.R., Pal, S.K., 1993. A review on image segmentation techniques. *Pattern Recognition*, 26, pp.1277–1294.
- [5] Yen, J.C., Chang, F.J., Chang, S., 1995. A new criterion for automatic multilevel thresholding. *IEEE Trans. Image Process.* IP-4, 370–378.
- [6] Sahoo, P., Wilkins, C., Yeager, J., 1997. Threshold selection using Renyi’s entropy. *Pattern Recognition*, 30, pp.71–84.
- [7] Mehmet Sezgin Bu` lent Sankur. 2004. Survey over image thresholding techniques and quantitative performance evaluation *Journal of Electronic Imaging* 13(1), pp.146–165
- [8] F. Du, W. K. Shi, L. Z. Chen, et al, 2005. Infrared image segmentation with 2-D maximum entropy method based on PSO. *Pattern Recognition Letters*, 26, pp. 597-603
- [9] T. Srikanthan and K. V. Asari, 2001. Automatic segmentation algorithm for the extraction of lumen region and boundary from endoscopic images, *Med. Biol. Eng. Comput.* 39(1), pp.8-14
- [10] L. Li, J. Gong, W. Chen, 1997. Gray-Level Image Thresholding Based on Fisher Linear Projection of Two-Dimensional Histogram, *Pattern Recognition*, 30, pp. 743-749.
- [11] Wenbing Tao, Hai Jin, Liman Liu. 2007. Object segmentation using ant colony optimization algorithm and fuzzy entropy. *Pattern Recognition Letters* 28, pp. 788–796
- [12] TANG Yinggan HUANG Na, GUAN Xiping, 2009. Infrared Image Segmentation Using Two-Dimensional Fisher Linear Optimal Discriminant Analysis and Particle Swarm Optimization. *Chinese Journal Of Electron Devices*,32(1), pp.12-16

- [13] D.Karaboga, "An idea based on bee swarm for numerical optimization," Tech. Rep. TR-06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
- [14] D. Karaboga and B. Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," J. Global Optim, Vol. 39, pp. 459–471, 2007
- [15] D. Karaboga and B. Basturk. "On the performance of artificial bee colony (ABC) algorithm," Appl. Soft Comput., Vol. 8, pp. 687–697, 2008.
- [16] N. Karaboga, A new design method based on artificial bee colony algorithm for digital IIR filters, Journal of the Franklin Institute, 346 (4), 328-348, 2009.
- [17] Fei Kang; Junjie Li; Haojin Li; Zhenyue Ma; Qing Xu An Improved Artificial Bee Colony Algorithm Intelligent Systems and Applications (ISA), 2010 2nd International Workshop on Issue Date: 22-23 May 2010: 1-4
- [18] Ivona Brajevic, Milan Tuba, and Milos Subotic. Performance of the improved artificial bee colony algorithm on standard engineering constrained problems. International journal of mathematics and computers in simulation. 2011, Vol.5 (2):135-143
- [19] Kennedy J , Eberhart R. Particle swarm optimization[A]. Proc IEEE Int Conf on Neral Networks [C]. Perth, 1995. 1942-1948
- [20] Shi, Y.; Eberhart, R.C. . "Parameter selection in particle swarm optimization". Proceedings of Evolutionary Programming VII (EP98). 1998.pp. 591–600
- [21] Shi, Y.; Eberhart, R.C. (1998). "A modified particle swarm optimizer". Proceedings of IEEE International Conference on Evolutionary Computation. pp. 69–73
- [22] van den Bergh F,Engelbrecht A P.Cooperative learning in neural networks using particle swarm optimization].South African Computer Journal,2000,26(6):84-90.
- [23] Suganthan P N. Particle swarm optimizer with neighborhood operator[A].In:Proc. on evolutionary computation[C].Piscataway,NJ: IEEE Service Center.1999, pp. 1958-1962.
- [24] Gao Shang ,Yang Jing-Yu ,“Research on Chaos Particle Swarm Optimization Algorithm”, pattern recognition and artificial intelligence, April 2006,Pages 267-270 (in Chinese)
- [25] Zheng hong, Pan Li. The automatic selection of image threshold on the basis of genetic algorithm. journal of image and graphics (in Chinese) vol 4(A), No .4 1999 327~330



Zhiwe Ye, born in Hubei China, 5.17, 1978, received the B.S. degree in land resource management and Ph.D. degree in Photogrammetry from Wuhan University, Wuhan, China in 2001, and 2006.

He is an associate Professor in school of computer science, Hubei university of technology, Wuhan China. He has published more than twenty papers in the area of image processing and swarm intelligence. His research interests include image processing, pattern recognition and artificial intelligence intelligence. His current research interests include image processing, swarm intelligence. Dr. Ye is a member of IEEE and ACM.



Xudong LAI, born in 1972, received the B.S. degree, M.S. degree and Ph.D. degree in Photogrammetry from Wuhan University, Wuhan, China in 1995, 2003 and 2006. He is associate Professor in School of Remote Sensing and Information Engineering, Wuhan University, China. He majors in LIDAR, GIS and Photogrammetry. He has

published about 20 papers in related research fields. Dr. Lai is a member of IEEE.



Hongwei Chen, born in 1975, he received PHD in computer science, Nanjing University of Posts and Telecommunications, CS, China, 2006MSc, China University of Geosciences, CS, China, 2003. Now he is an Associate Professor in the School of Computer Science at Hubei University of Technology, Wuhan, China. He is a member of IEEE and ACM; currently, his major field of

study is Peer-to-Peer and intelligence computation.