Research Article

Evolutionary Game Algorithm for Image Segmentation

Jin Zhong and Hao Wu

College of Computer Science, Hefei Normal University, Hefei 230601, China

Correspondence should be addressed to Jin Zhong; 952639879@qq.com

The traditional two-dimensional Otsu algorithm only considers the limitations of the maximum variance of between-cluster variance of the target class and background class; this paper proposes evolutionary game improved algorithm. Algorithm takes full consideration of own pixel cohesion of target and background. It can meet the same of maximum variance of between-cluster variance. To ensure minimum threshold discriminant function within the variance, this kind of evolutionary game algorithm searching space for optimal solution is applied. Experimental results show that the method proposed in this paper makes the detail of segmentation image syllabify and has better antijamming capability; the improved genetic algorithm which used searching optimal solution has faster convergence speed and better global search capability.

1. Introduction

Image segmentation is the first step in image analysis, understanding, and pattern recognition, which is also one of the most important steps. Image segmentation is often used in medical image processing, such as nuclear magnetic resonance image, but it is also widely used in geographical space, environmental meteorology, and other fields. At present, there are many methods of image segmentation. Since each image has its own target and background, there is no general segmentation method for all images. Threshold segmentation is one of the most commonly used methods of image segmentation. Its essence is through the image histogram information to determine the threshold of image segmentation. Threshold segmentation method has the minimum error method, the maximum entropy method, and the maximum between-class variance methods. The traditional Otsu method is based on the statistical properties of the first-order histogram. It is widely used because of its simple algorithm and high real-time performance. Therefore, many literatures have improved the Otsu method and proposed a series of two-dimensional thresholding segmentation methods. In the literature [1], the statistical properties based on two-dimensional histogram are proposed to determine the

threshold value. It includes not only the gray information of the image pixels. But also the spatial information of the pixels. The effect is better than the one-dimensional Otsu method and has stronger antinoise performance. However, the traditional two-dimensional Otsu algorithm is complex and has long running time. And the effect of segmentation is vague on the contour. This paper is based on the evolutionary game algorithm of protein secretion mechanism. The two-dimensional thresholds are continuously optimized to obtain the two-dimensional thresholds that need to be image segmented. The algorithm has a good segmentation result and relatively good real-time performance and to a great extent solves the problem of local convergence.

2. The Principle of Two-Dimensional Maximum Between-Class Variance (Otsu) and Its Improvement

Otsu method Proposed by Japanese scholars in 1979 is the classic threshold segmentation methods. The basic idea is that it obtained the largest class difference of targets and background between the parties by image histogram statistics to determine the optimal threshold for image segmentation dynamically.
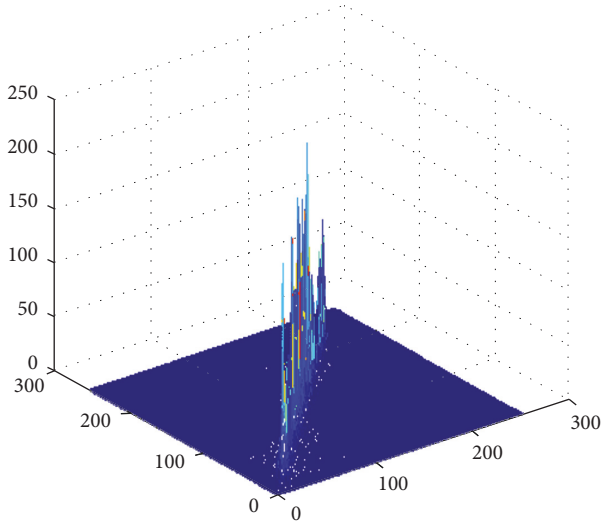
Figure 1: Two-dimensional histogram of an image histogram.

Figure 2: Two-dimensional projection plane.

2.1. Two-Dimensional Maximum Between-Class Variance Methods (Two-Dimensional Otsu Algorithm Method).

Two-dimensional Otsu method is a two-dimensional correlation threshold method. The principle is as follows. Assume that the gray level of the original image f is L, size is $M * N$, and the value of each pixel in the image corresponds to a gray level. Scan the whole image. Calculating the neighborhood gray value of each pixel in the image, the size of the window neighborhood is usually odd. Get a smooth image g. Obviously g gray level is also L. Let m_{ij} be pixel value i of that image f. The number of pixels in the neighborhood gray value of j is the gray value and its binary group (i, j) of neighborhood pixels average. The result is two-dimensional histogram of point gray—neighborhood pixels gray. Then the two-dimensional joint probability density is

$$p_{ij} = \frac{m_{ij}}{M \times N}. \tag{1}$$

p_{ij} Meet $\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} p_{ij} = 1$. It is $L * L$ matrix with size of $L * L$.

Figure 1 shows two-dimensional histogram of the schematic image. Figure 2 is the corresponding projection plane, f-axis represents the gray value of each pixel, and g-axis means neighborhood pixels' average of each pixel gray. Figures 1 and 2 show the peak of pixels intensity in the vicinity of the diagonal, because of the largest proportion of the target region and background region of the image. Distribution of the pixels gray level is uniform relatively.

Difference of point gray value and its neighborhood pixels gray average is not large, so they are concentrated near the diagonal.

Assume that value of the background and objectives in the image as two classes is C_b and C_o, respectively. (s, t) is two-dimensional threshold vector for images segmentation, and occurrence probability of the background is

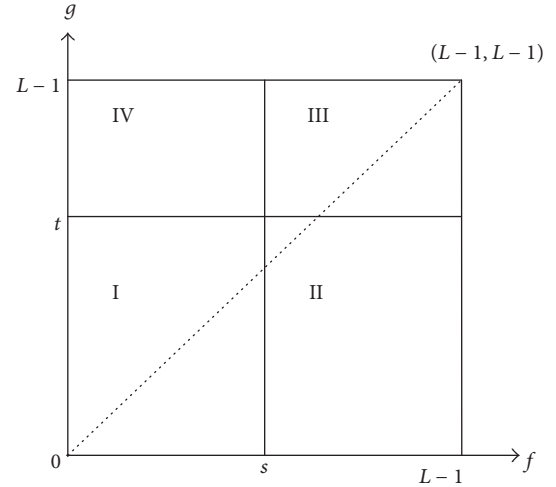$$\omega_b = p(C_b) = \sum_{i=0}^{s} \sum_{j=0}^{t} p_{ij} = \omega_b(s, t). \tag{2}$$

Occurrence probability of the target is

$$\omega_o = p(C_o) = \sum_{i=s+1}^{L-1} \sum_{j=t+1}^{L-1} p_{ij} = \omega_o(s, t). \tag{3}$$

In most cases, the occurrence probability of the noise and edge is very small. According to Figure 2, the probability of the regions II and IV can be regarded as 0 approximately.

According to Figure 2, the probability of the regions II and IV can be regarded as 0 approximately.

This can be considered as $\omega_b + \omega_o = 1$. The two classes of background and target correspond to the mean vector to be

$$\mu_b(s, t) = \left(\mu_{bi}, \mu_{bj}\right)^T$$

$$= \left[\frac{\sum_{i=0}^{s} \sum_{j=0}^{t} i p_{ij}}{\omega_b(s, t)}, \frac{\sum_{i=0}^{s} \sum_{j=0}^{t} j p_{ij}}{\omega_b(s, t)}\right]^T,$$

$$\mu_o(s, t) = \left(\mu_{oi}, \mu_{oj}\right)^T \tag{4}$$

$$= \left[\frac{\sum_{i=s+1}^{L-1} \sum_{j=t+1}^{L-1} i p_{ij}}{\omega_o(s, t)}, \frac{\sum_{i=s+1}^{L-1} \sum_{j=t+1}^{L-1} j p_{ij}}{\omega_o(s, t)}\right]^T.$$

Ensemble average is

$$\mu_{\text{all}}(s, t) = \left(\mu_{\text{all}i}, \mu_{\text{all}j}\right)^T = \left[\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} i p_{ij}, \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} j p_{ij}\right]^T. \tag{5}$$

Define discrete matrix

$$\sigma_B = \omega_b \left[\left(\mu_b - \mu_{\text{all}}\right)\left(\mu_b - \mu_{\text{all}}\right)^T\right]$$

$$+ \omega_o \left[\left(\mu_o - \mu_{\text{all}}\right)\left(\mu_o - \mu_{\text{all}}\right)^T\right]. \tag{6}$$

The trace of the discrete matrix class as the measure of dispersion of the background and objectives is

$$\mathrm{tr}\left(\sigma_B\right) = \omega_b\left[\left(\mu_{bi} - \mu_{\mathrm{all}i}\right)^2 + \left(\mu_{bj} - \mu_{\mathrm{all}j}\right)^2\right]$$
$$+ \omega_o\left[\left(\mu_{oi} - \mu_{\mathrm{all}i}\right)^2 + \left(\mu_{oj} - \mu_{\mathrm{all}j}\right)^2\right]. \quad (7)$$

When the trace of the dispersion matrix is the maximum value, we obtain the optimal segmentation threshold

$$\mathrm{tr}\left(s_{(s^*,t^*)}\right) = \max\left(\mathrm{tr}\left(s_{(s,t)}\right)\right), \quad 0 \le s, t \le L-1. \quad (8)$$

2.2. Two-Dimensional Maximum of Between-Class Variance Method. In the traditional two-dimensional Otsu method, the threshold discriminant function (trace of the dispersion matrix) takes into account only the variance of the target class and the background class; that is, the larger the variance between classes, the better the segmentation effect. However, the traditional two-dimensional Otsu algorithm does not take into account its own classified information of each type of pixels of target class and background class, that is, considering the cohesion within the classes. So, in this paper, the measure of the dispersion within the class is also introduced to the recognition function of threshold. This can reflect the efficiency of the overall classification, specifically, the following: two classes c_o and c_b existing in two-dimensional histogram and calculated variance of the center of the target class μ_o. The center of the background class μ_o with (i, j) is from each gray value and its neighborhood pixels average separately:

$$d_o = \frac{\sum_{i=0}^{s}\sum_{j=0}^{t}\left[\left(i - \mu_{oi}\right)^2 + \left(j - \mu_{oj}\right)^2\right]p_{ij}}{\omega_o},$$

$$d_b = \frac{\sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}\left[\left(i - \mu_{bi}\right)^2 + \left(j - \mu_{bj}\right)^2\right]p_{ij}}{\omega_b}. \quad (9)$$

Obviously, the smaller the value of the variance of the target class and background class, the better its cohesion and the better the segmentation effect.

The definition of the measure of dispersion within class is

$$\rho_s = \omega_o d_o + \omega_b d_b. \quad (10)$$

Obviously, ρ_s is required smaller value, and cohesion within the classes gets the better effect.

We are considering the between-class variance and within the class variance. The threshold makes between-class variance the largest, and, at the same time, within the class variance meets the minimum value. A new discriminant function of the threshold is used in this article as follows:

$$\phi_s\left(s, t\right) = \frac{\omega_o \times \omega_b \times \mathrm{tr}\left(\sigma_B\right)}{\rho_s}. \quad (11)$$

Molecular $\omega_o \times \omega_b \times \mathrm{tr}(\sigma_B)$ shows the property of between-class variance and denominator ρ_s performance within the class cohesion; when $\phi_s(s, t)$ obtains the maximum value, effect of the image segment is best.

That is, when value of the discriminant function is the maximum, (s', t') is the best threshold for the segmentation threshold.

Now, the target class and background class are separated greatly, and the cohesion of target class and background is the best kind.

The optimal threshold vector to achieve the gray image banalization is got as follows:

$$f\left(i, j\right) = \begin{cases} 0, & i < t, j < s, \\ 1, & i \ge t \text{ or } j \ge s. \end{cases} \quad (12)$$

3. Threshold Vector Evolutionary Game Algorithm

In this paper, the optimal solution of the threshold vector can be obtained by the evolutionary game algorithm. The new generation of each generation of the algorithm is randomly paired and repeated. Each individual is to optimize their objective function to the optimal value [2–8]. The target value is determined by the evolutionary game matrix. Set the objective function as $f_1(x, y), f_2(x, y)$. The objective function has two mixed variables x and y. The game structure of evolutionary algorithm is designed as follows:

$$\begin{bmatrix} \text{Agent} & \text{Fitness} \\ V_1 = \left(x_1, y_1\right) & F_1 \\ V_2 = \left(x_2, y_2\right) & F_2 \\ \vdots & \vdots \\ V_k = \left(x_k, y_k\right) & F_k \end{bmatrix} \begin{matrix} \text{Plays} \\ \leftrightarrow \\ \leftrightarrow \\ \vdots \\ \leftrightarrow \end{matrix} \begin{bmatrix} \text{Agent} & \text{Fitness} \\ G_1 = \left(x_1', y_1'\right) & F_1' \\ G_2 = \left(x_2', y_2'\right) & F_2' \\ \vdots & \vdots \\ G_k = \left(x_k', y_k'\right) & F_k' \end{bmatrix}. \quad (13)$$
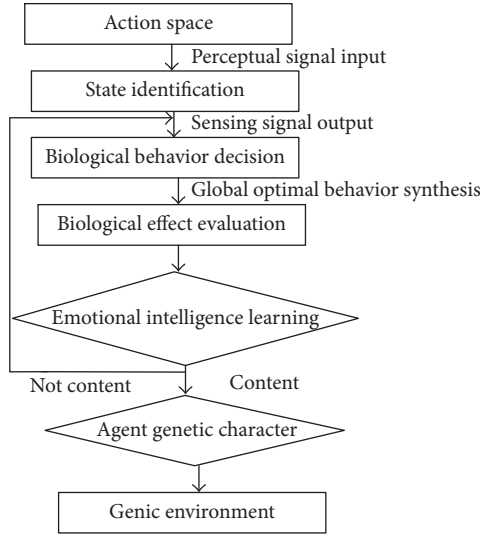
FIGURE 3: Architecture of population for the proliferation of evolution PEP.

The fitness of F_i is determined by the game matrix, $i = 1, 2, \ldots, k$. For the participants of the two random pairs of the game through the evolution game, the G_i of each participant's game payoff is the difference between the two objective functions:

$$
\begin{aligned}
G_1\left(V_i, V_i'\right) &= G_1\left((x_i, y_i), (x_i', y_i')\right) \\
&= f_1(x_i, y_i) - f_2(x_i', y_i'), \\
G_2\left(V_i, V_i'\right) &= G_2\left((x_i, y_i), (x_i', y_i')\right) \\
&= f_1(x_i', y_i') - f_2(x_i, y_i).
\end{aligned}
\tag{14}
$$

According to this value, the fitness of each objective function is calculated by

$$
\begin{aligned}
F_i &= 100 \times \frac{G_1\left((x_i, y_i), (x_i', y_i')\right)}{\omega}, \\
F_i' &= 100 \times \frac{G_2\left((x_i, y_i), (x_i', y_i')\right)}{\omega}.
\end{aligned}
\tag{15}
$$

In the formula, a is B or C fitness scalar. ω is the maximum value of $G_k((x_i, y_i), (x_i', y_i'))$. Each generation of the subgame of the evolutionary algorithm is based on the fitness value.

3.1. Evolutionary Game Algorithm Based on Protein Secretion Mechanism.

Drawing on the main signal hypothesis in recent years, that is, the cooperative translation of protein secretion process [9], in this paper, based on the self-organization evolutionary game algorithm of protein secretion mechanism behavior, we use the method of signal peptide in the cell matrix to guide the synthesis of secreted protein synthesis as an algorithm to realize [9], as shown in Figure 3.

It is assumed that the state of the protein secretion system is discrete and the state, behavior, and emotional intelligence are represented by a collection of three-dimensional

sequences, represented by the I, I_{at}, I_t. The game matrix is described as

$$
\begin{bmatrix} I_{at}(t+1) \\ I_t \\ 1 \end{bmatrix} = \begin{bmatrix} I_{11} & -\omega & I_{13} \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} I_t \\ I_{at}(t-1) \\ 1 \end{bmatrix}.
\tag{16}
$$

The line number of the matrix represents the behavior sequence. The value of the element of the matrix represents the corresponding behavioral states corresponding to the emotional intelligence value. I_{at} is expressed sentiment value a of behavior execution under state k. I_t represents the state of t intelligence. The high level regulation of protein secretion is achieved by evaluating the state of the system and by feedback learning based on the results of the evaluation. Here is the selection behavior of the algorithm steps in state recognition. Hypothetical selection behavior steps are as follows:

(1) After the behavior evolution of the state recognition, the state action selection rule is described by the formula, and all possible behaviors in the state can be represented by the "$*$":

$$
A = A\text{function}\left(I_{*t}\right) = \max\left(I_{*t}\right).
\tag{17}
$$

(2) The rules of evaluation rules of emotional intelligence of the state of emotional intelligence factor are

$$
I_t = I\text{function}\left(I_{*t}\right) = \max\left(I_{*t}\right).
\tag{18}
$$

(3) The learning rules of emotional intelligence are described in formula, in which ω represents learning rate:

$$
\begin{aligned}
\left(I_{*t}(t+1)\right) &= U\text{function} = \left(I_*, I_{*t}\right) \\
&= (1-\omega) I_{*t}(t) + \omega I_{*t}.
\end{aligned}
\tag{19}
$$

Define the end condition of an emotional learning after the end of emotional learning. The self-adaptation of the subject to the behavioral environment is preserved in the protein secretion environment. These adaptive results are passed to the individual through the output of the protein secretion mechanism. Compared to the next generation, the next generation has a more adaptive behavior environment. Based on the mechanism of protein secretion, this paper designs the self-organization evolutionary game algorithm, as shown in Figure 4.

Here is the solving process for specific threshold vector.

(1) Coding. Gray level of an image is from 0 to 255, encoding binary. For one-dimensional Otsu method, the threshold is only one which can be represented by 8-bit binary.

The two-dimensional Otsu method has two threshold segmentations, so we use 16-bit (bit string length) method to achieve a binary encoding, that is $(x_0, x_1, \ldots, x_7, x_8, \ldots, x_{15})$, where top eight represent segmentation threshold s, and after eight they represent segmentation threshold t.

According to the improved two-dimensional histogram strategy proposed in Section 2.2 a strategy is that, in this test, N is 30, so $0 + 30 \leq s, t \leq 255 - 30$; that is, $0 + 30 \leq s, t \leq 255 - 30$.

(2) Initialization. Let the initial population be of popsize. Because the optimization parameters are two-dimensional
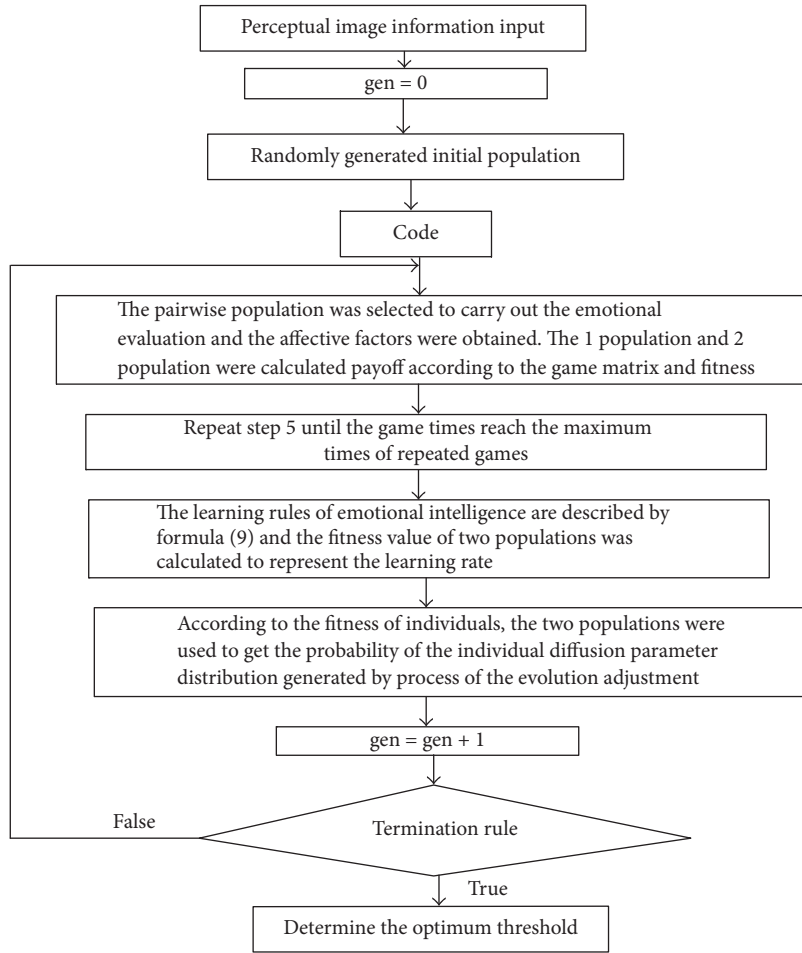
FIGURE 4: Evolutionary algorithm based on the mechanism of protein secretion.

threshold, so Initialize can randomly generate a pop size line, 16-column matrix by random number generator.

The following operations are for the targeted population initialized. Here process of decoding needs to take into account the range of s and t.

(3) Fitness Function. According to the analysis of Section 2.2, this new the threshold discriminant function $\phi_s(s, t)$ is the fitness function, and when $\phi_s(s, t)$ obtains the maximum, value of the parameters s and t is the optimal threshold for image segmentation.

(4) Select. The traditional genetic algorithm used in the roulette selection method usually leads to the loss of diversity of the population, and genetic algorithm will prematurely lose its evolutionary capacity.

This article takes the choice explained here. First of all, to retain some of the individuals with the highest fitness value of the parent directly into the offspring, for some suboptimal individuals after a certain mutation the excellent individuals can also enter the progeny; that is, through the evolution of different populations of the parent population, we get offspring.

This way not only ensures the outstanding individuals of the population structure into the next generation, but also ensures the diversity of the population. This makes the probability of having similar individual in the population reduced, increases the following crossover and mutation operation efficiency, and improves the convergence of the algorithm.

(5) Crossover. Crossover operation is the unique original characteristics [10] of genetic algorithm of evolutionary algorithm.

There is a problem about the traditional crossover operator; when Stocks evolve to the local optimum, many individuals are very similar, which is "close."

Cross-string of individuals is the same largely; the role of crossover operation does not work.

In order to avoid "close" cross, in this article, we selected individuals randomly for crossover Hamming distance judgments on the mating pool.

If we select the individual on the Hamming distance $H(s_i, s_j) \geq \lambda$ $(0 \leq \lambda \leq l)$, according to the original the probability, we operate cross-cross.

TABLE 1: The threshold obtained from several algorithms.

Image	One-dimensional Otsu method is based on the traditional standards of GA	Threshold Method of literature 3	Traditional two-dimensional Otsu standards-based GA	This method
Horse	96	(83, 145)	(83, 146)	(66, 143)
Airplane	174	(130, 209)	(131, 204)	(51, 218)
Tank	116	(175, 87)	(173, 85)	(34, 200)

Conversely, if $H(s_i, s_j) < \lambda$, then we determine that the individuals are "close" and replace one of the individuals, until the conditions are met or all individuals in the group are tried to be replaced.

(6) Variation. The mutation operation in traditional genetic algorithm is based on a certain mutation probability for values of each individual bit string, that is, 1 to 0 or 0 to 1. Typically, the probability of mutation is relatively small. In the implementation process, some individuals could not have taken place to mutate, wasting a lot of computing resources. This article uses a dynamic mutation probability in order to increase the diversity of the population. The basic idea is to determine whether the current population is or not "mature"; if it is, then it mutates on a larger probability; otherwise, according to conventional mutation probability, f_{max} is defined as the best current population fitness; f_{avi} is the current population average fitness; if $k f_{max} \geq f_{avi}$, where $(0.5 \leq k \leq 1)$, as intensive factor, then our algorithm determines that the population does not "premature" and mutate on conventional probability. On the contrary, if the population is "premature" and mutates on much bigger than conventional probability to all individuals, it results in more individuals into the next generation, in order to ensure population out of local convergence.

4. Simulations and Analysis of Experiments

Image segmentation tests were carried out by the proposed image segmentation, that is, two-dimensional Otsu method based on improved genetic algorithm. We compared between segmentation of horse picture, airplane image, and tank image by the one-dimensional Otsu method, the traditional two-dimensional Otsu method, improved two-dimensional Otsu method based on the traditional GA, and the proposed algorithm.

Experiment hardware platform is 2.6 GHz Pentium 4, Capacity of memory is 512 MB, and operating environment is Matlab 7.0.

Various experimental parameters are set as follows

(1) one-dimensional Otsu method using the standard threshold of GA optimization, size of population: 10; number of iterations: 15; length of code: 8; probability of crossover: 0.7; probability of mutation: 0.05;

(2) traditional neighborhood of two-dimensional Otsu method, size of window neighborhood: 3;

(3) this method adopting an improved two-dimensional Otsu method and the improved GA, size of window neighborhood: 3; size of population: 10; number of iterations: 150; length of code: 16; probability of crossover: 0.7; Hamming discriminant distance: 2; larger mutation probability: 0.2; small mutation probability: 0.01; intensive factor: 0.8.

Various types of experimentally derived physical threshold are in Table 1, and result of one-dimensional Otsu method, traditional Otsu method based on the standards of GA, and this method is the average obtained for 30 independent experiments.

In the actual image acquisition process, images often will be affected by the external environment; the image will contain all kinds of noise; in the following example, horse pictures added a variety of noise; then this method deals with the results of noise.

Running time of one-dimensional Otsu method is shorter; however, from Figures 5, 6, and 7, the segmentation results show that traditional two-dimensional Otsu algorithm and out method are better; many details are not separated out, especially in Figure 6; the segmentation of the target aircraft is more accurate.

Figures 8, 9, and 10 show that various types of noise have a great impact on results of all one-dimensional Otsu algorithm. Its effects and antijamming capability are poorer than this method.

The traditional two-dimensional Otsu method is better than the one-dimensional Otsu method; the segmentation of images containing noise removal can be seen very clearly.

But its calculation is based on exponential growth because the algorithm has a double loop, a total of calculation times needed are $M \times N$, that is, $\text{tr}(s_{(s,t)})$, and $M \times N$ is the total pixel number of times.

Every time it cumulates $s \times t + (L - s) \times (L - t)$ points; therefore, the algorithm complexity is $o(L^4)$.

This calculation is very large, and in practice such a long running time is unacceptable.

Relative to the one-dimensional Otsu method, the traditional two-dimensional Otsu method is the sacrifice of time in exchange for consideration of segmentation accuracy.

In this paper, an improved GA optimizes the threshold parameters; it is clear that the convergence for the calculation of $\phi_s(s, t)$ is product of the number of generations and size of population; Table 2 shows that the average convergence times of generation algebra of our algorithm are 15; therefore, it is also necessary to calculate $15 \times 10 = 150$ times. At the same
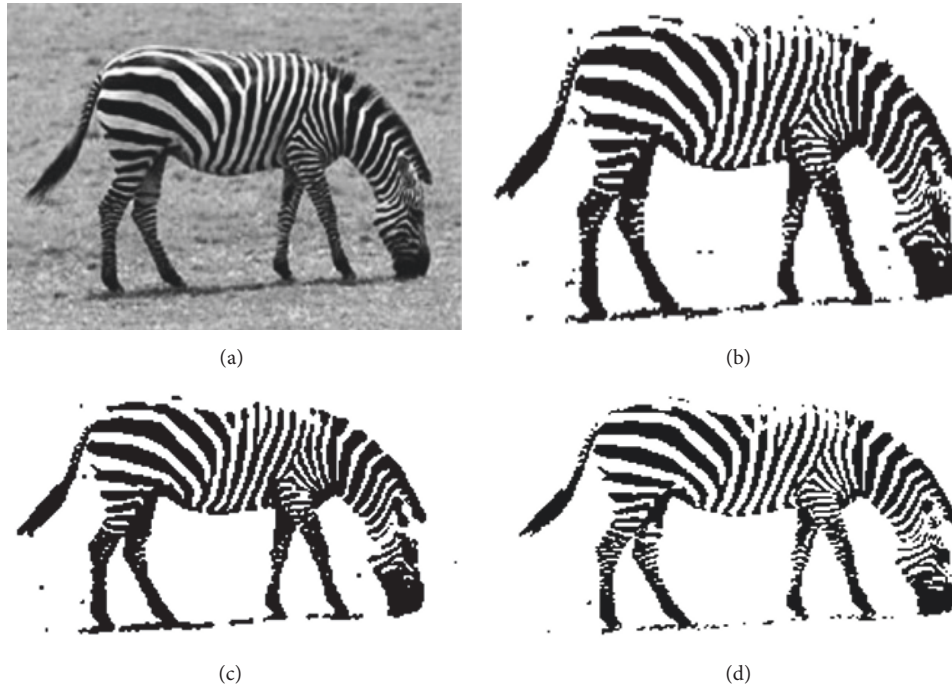
(a)

(b)

(c)

(d)

Figure 5: Segmentation of horse image. (a) Horse original. (b) Segmentation of one-dimensional Otsu. (c) The traditional two-dimensional Otsu method. (d) Our algorithm.
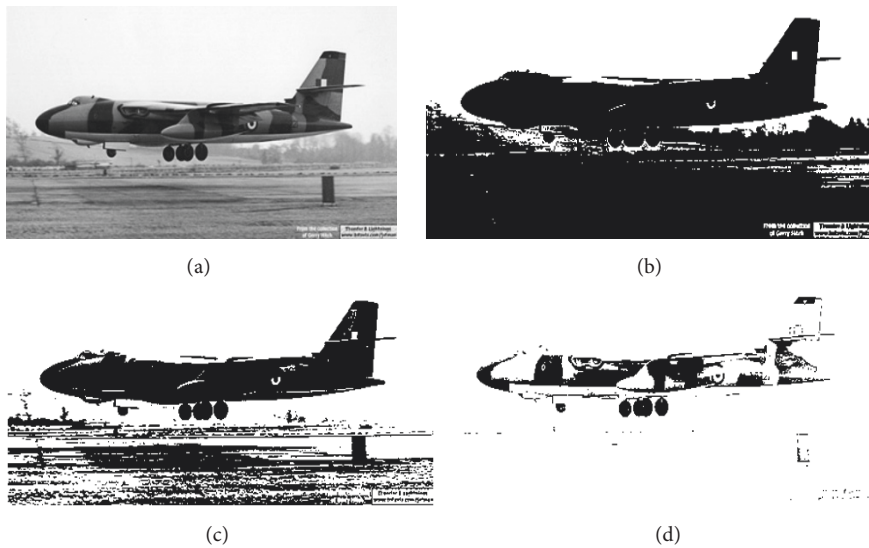

(a)

(b)

(c)

(d)

Figure 6: Segmentation of airplane image. (a) Horse original. (b) Segmentation of one-dimensional Otsu. (c) The traditional two-dimensional Otsu method. (d) Our algorithm.

time, it also needs to cumulate $s \times t + (L - s) \times (L - t)$ points. Thus, time complexity of the algorithm is $o(L^2)$.

For an image, the traditional Otsu method requires 256×256 calculating of $\mathrm{tr}(s_{(s,t)})$. Out algorithm only needs to calculate 150 times of $\phi_s(s, t)$, 437 times faster.

In fact, the improved GA algorithm in this paper accelerates the convergence process; it is basically the average convergence times to twenty, saving the running time of this algorithm greatly and avoiding local convergence problems from the traditional GA algorithm.

In order to verify that the improved GA has better convergence, the improved two-dimensional Otsu method also used standard GA parameters to do experiment and compared with our method.

Figures 11, 12, and 13 are the evolution curve of two algorithms about horse image, dragon image, and tank image, respectively.
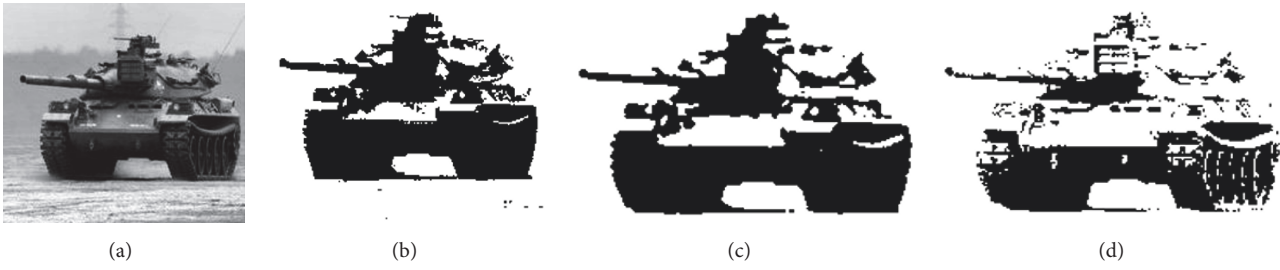
(a)   (b)   (c)   (d)

Figure 7: Segmentation of tank image. (a) Horse original. (b) Segmentation of one-dimensional Otsu. (c) The traditional two-dimensional Otsu method. (d) Our algorithm.
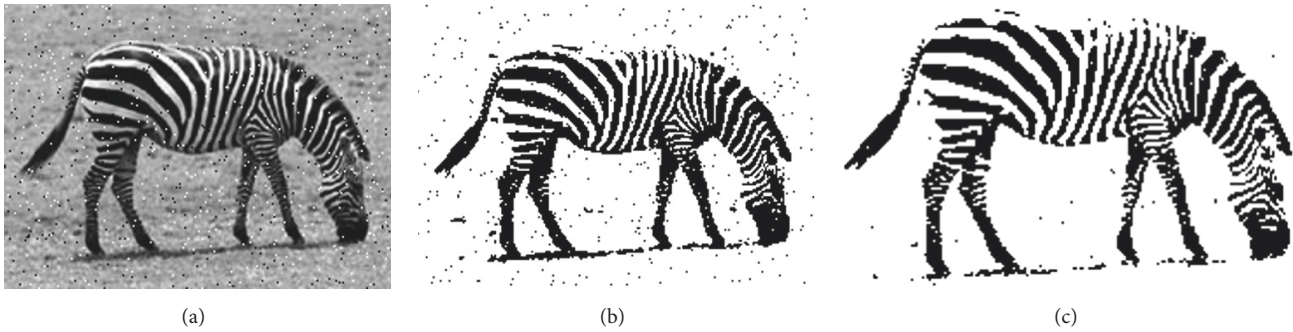

(a)   (b)   (c)

Figure 8: Horse image (0.02-degree encryption salt and pepper noise). (a) Original. (b) The results of one-dimensional Otsu method. (c) The results of our algorithm.

(a)   (b)   (c)
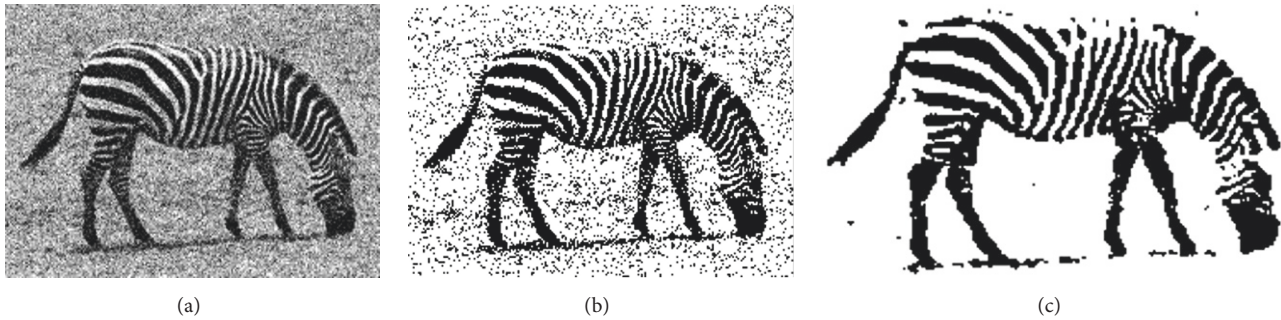
Figure 9: Horse image (mean of Gaussian noise is 0 and variance is 0.02). (a) Original. (b) The results of one-dimensional Otsu method. (c) The results of our algorithm.

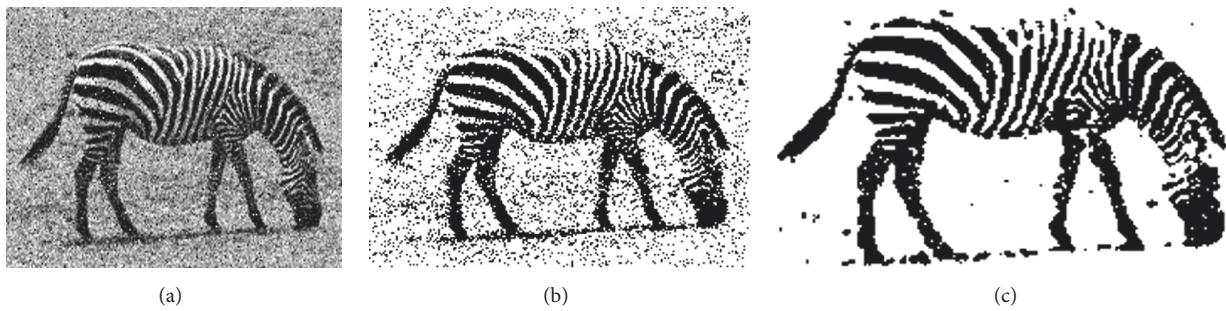(a)   (b)   (c)

Figure 10: Horse image (mixed mean of Gaussian noise is 0 and variance is 0.02 and 0.02-degree encryption salt and pepper noise). (a) Original. (b) The results of one-dimensional Otsu method. (c) The results of our algorithm.

TABLE 2: Comparison of the running time of several two-dimensional Otsu algorithms.

| Photo (specification)	Method of literature 3		The traditional method based on the traditional GA Otsu			This method (improved Otsu + improved GA)		
	Time complexity	Running time (/s)	Running time	The average convergence times of 30 times	The average convergence times of 30 times (/s)	Time multiplexing complex degree	The average convergence times of 30 times	The average convergence times of 30 times (/s)
Horse (158 * 228)	$o(L^4)$	6193	$o(L^2)$	137	14.3424	$o(L^2)$	15	4.1222
Airplane (230 * 500)	$o(L^4)$	8537	$o(L^2)$	113	17.4219	$o(L^2)$	14	5.9717
Tank (212 * 178)	$o(L^4)$	6723	$o(L^2)$	129	14.5323	$o(L^2)$	11	4.3312
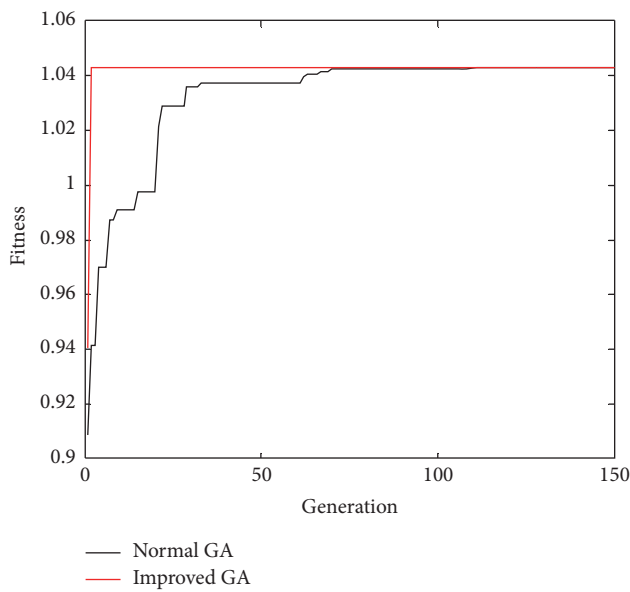
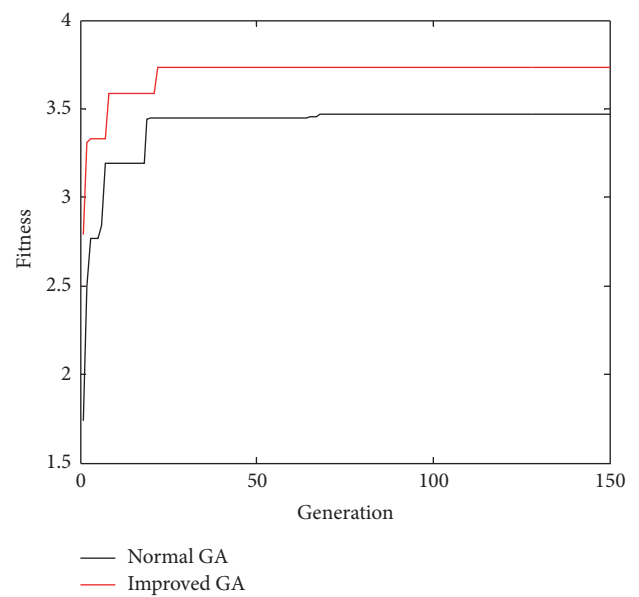FIGURE 11: Two evolutionary curves' algorithm for horse image.

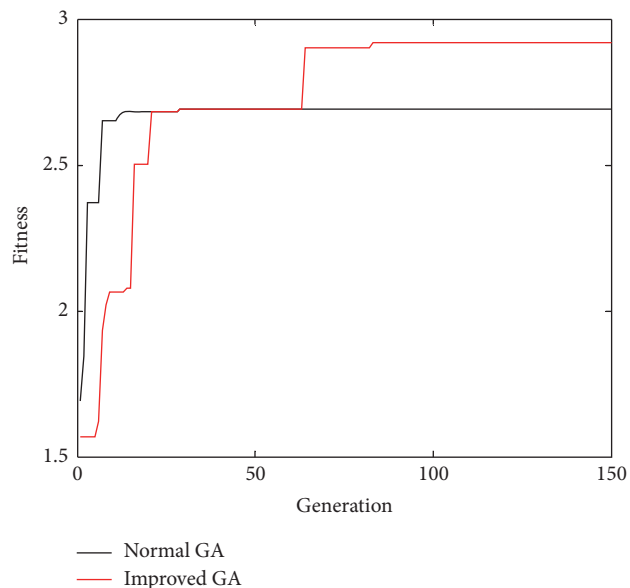FIGURE 13: Two evolutionary curves' algorithm for tank image.

FIGURE 12: Two evolutionary curves' algorithm for airplane image.

According to Figures 11 and 12, we can get result of the improved GA algorithm and the traditional GA algorithm. In Figure 11, convergence times of the improved GA are 4, whereas convergence times of the traditional GA algorithm are 87; they have been the largest fitness value.

In Figure 12, convergence times of the improved GA is 22, convergence times of the traditional GA is 78, and the traditional GA get into a premature local convergence and did not reach maximum fitness value.

In Figure 13, convergence times of the improved GA is 13, convergence times of the traditional GA is 61, and there is no maximum fitness value.

Therefore, the improved genetic algorithm GA obtained the optimal threshold that is clearly better than the optimal solution traditional genetic algorithm obtained, but also it is closer to the optimal solution and overcomes "premature" problem from the traditional GA algorithm largely.

The horse figure, dragon figure, and tank image show the comparative results of running time of several algorithms

from 30 figures of independent experiments, shown in Table 2.

Results of experimental show that the proposed improved genetic algorithm based optimization method to improve two-dimensional Otsu threshold vector is effective; it can both search the global optimal vector and get the cost of a very small amount of time, showing the higher robustness and real-time of algorithm.

5. Conclusion

The paper points out that the traditional two-dimensional Otsu method only considered deficiency of the largest variance between classes and proposes the threshold discriminant function which not only can reflect variance between classes, but also can reflect the variance within classes, and it obtained better segmentation than the traditional two-dimensional Otsu, while the genetic algorithm improved by this method has better population diversity and global search capacity.

Results of simulation show that the proposed method not only has better and faster segmentation effect, but also has some practical value.

Through a lot of pictures of other experiments, the results are generally better than the traditional two-dimensional Otsu method.

However, the relative algorithm GA based on the traditional standard two-dimensional Otsu method already has a very good time performance, but there are still some obstacles, because, in real-time applications, its running time is more than 4 seconds. So how to reduce running time and do not affect the segmentation result is an issue worthy of consideration.
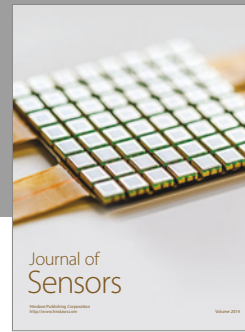
Conflicts of Interest

There are no conflicts of interest in the publication of this article.

Acknowledgments

References

[1] R. C. Gonzalez, *Digital Image Processing (2nd) [M]*, Publishing House of Electronics Industry, Beijing, China, 2003.

[2] Y. Min, *Digital Image Processing [M]*, China Machine Press, Beijing, China, 2006.

[3] J. Z. Liu and W. Q. Li, "Automatic thresholding using the otsu algorithm based on the two-dimensional gray image," *Journal of Electrical and Computer Engineering*, vol. 19, no. 1, pp. 101–105, 1993.

[4] K. Wei, T. Zhang, X. Shen, and J. Liu, "An improved threshold selection algorithm based on particle swarm optimization for image segmentation," in *Proceedings of the 3rd International Conference on Natural Computation, ICNC 2007*, pp. 591–594, August 2007.

[5] T. L. Huang and X. Bai, "An improved algorithm for medical image segmentation," in *Proceedings of the 2nd International Conference on Genetic and Evolutionary Computing, WGEC 2008*, pp. 289–292, September 2008.

[6] P. K. Sahoo and G. Arora, "A thresholding method based on two-dimensional Renyi's entropy," *Pattern Recognition*, vol. 37, no. 6, pp. 1149–1161, 2004.

[7] Q. Chang, L. Wang, C. Xing, and A. Liy, "Image threshold selection based on genetic algorithm," *Computer Engineering and Application*, vol. 38, no. 22, pp. 35–37, 2002.

[8] S. M. Bhandarkar and H. Zhang, "Image segmentation using evolutionary computation," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 1, pp. 1–21, 1999.

[9] W. Peizhen, D. Peiming, and C. Weinan, "A new mixed genetic algorithm for multilevel thresholding," *Chinese Journal of Image And Graphics*, vol. 5, no. 1, pp. 44–47, 2000.

[10] M. Q. Li, J. S. Kou, D. Lin, and S. Li, *Basic Theory and Application of Genetic Algorithm*, Science Press, Beijing, China, 2002.