Journal of

Software
Engineering

Image Segmentation Using Automatic Selected Threshold Method Based on Improved Genetic Algorithm

Gong Kun Luo

College of Electrical and Information Engineering, Guangxi University of Science and Technology, Liuzhou, China

ABSTRACT

In this study, an image segmentation using automatic selected threshold method based on improved genetic algorithm is presented. It can overcome the shortcomings of the existing image segmentation methods which only consider pixel gray value without considering spatial features and computational complexity of these algorithms is too large. Encoding, crossover, mutation operator and other parameters of genetic algorithm are improved moderately in this method. Optimal threshold for image segmentation is converted into an optimization problem in this new method. The selection algorithm is optimized by using simulated annealing temperature parameters to achieve selective pressures. In order to achieve image segmentation, the optimal threshold is solved by using optimizing efficiency of improved genetic algorithm. Simulation results show that the new algorithm greatly reduces the optimization time and enhances the anti-noise performance of image segmentation and improves the efficiency of image segmentation. Thus, the new method can facilitate subsequent processing for computer vision and can be applied to real-time image segmentation.

Key words: Image segmentation, Otsu, improved genetic algorithm, automatic selected threshold

INTRODUCTION

Image threshold segmentation is a very important and common method for area segmentation which is used for image classification and pre-preliminary. Image segmentation is a kind of technology or processes which can segment image into characteristics regions for extracting interested object (Wang et al., 2010a; Felzenszwalb and Huttenlocher, 2004). Image segmentation has been the focus and difficulty in the field of image processing. Image segmentation is considered a bottleneck of image processing because subsequent processing for image depends on the quality of image segmentation such as feature extraction, object recognition and so on. Threshold segmentation method is one of the typical image segmentation methods. In practice, the threshold segmentation method usually used as a basic segmentation technique can achieve good effect for image segmentation. For threshold segmentation method, the hardest and most critical problem is how to set the threshold. However, threshold segmentation method is sensitive to noise as in the process of image segmentation only considering the gradation value of the pixel without considering the spatial characteristics. As involving logarithmic operation, the computing speed of threshold segmentation method is slow, so it is not suitable for real-time image processing. Otsu method is sensitive to the size of target and noise, only acquiring better image segmentation results for image with unimodal variance between classes. The 2D threshold algorithm which has complex calculation and long running time is susceptible to noise interference. Mathematical operation of

simple statistical method is simple but it only has better segmentation effect for image whose size ratio of background and target is 1:1, when deviation exists, segmentation effect is poor and vulnerable to noise. Moment-kept method which need not iterative or search has fast computing speed and can be extended to multi-threshold segmentation but it is vulnerable target size and segmentation accuracy is not ideal. Calculation of minimum error method is simple but it is not conducive to real-time processing for large amount of computation. In order to address these shortcomings, this study proposes an automatic threshold image segmentation algorithm based on improved genetic algorithm. Encoding, crossover, mutation operator and other parameters of genetic algorithm are improved appropriately in this new method. Improved genetic algorithm is introduced to automatically threshold selection to obtain ideal segmentation using the probability of global optimization using the quickly search optimization characteristic of improved genetic algorithm for adaptive global optimization probabilistic search. It can greatly reduce the computation time, improve anti-noise performance and segmentation efficiency.

OTSU IMAGE THRESHOLD SEGMENTATION

Otsu is deduced on the basis of the principle of least squares. It is a classification category function. When value of this classification function is maximum, we can obtain the optimal threshold K* for image segmentation. This threshold will be used to divide the whole image pixel into two parts, foreground and background. Selection of the optimal threshold would make the biggest difference between the foreground and background, the probability of misclassification is minimized. Selection of the optimal threshold can be achieved by following process (Wang and Li, 2011; Vese and Chan, 2002).

Let an image gray value 1, 2,..., m, total m level, where the pixels number of gray value i is n_i, the total pixels number of an image is as shown in Eq. 1:

$$N = \sum_{i=1}^{m} n_i \tag{1}$$

The probability of each gray scale value can be calculated by Eq. 2:

$$p_i = n_i/N \tag{2}$$

where, i = 1, 2, 3,..., m.

Then, all gray values of image is divided into two groups by using gray level k, i.e., $G_0 = \{1, 2, 3,... k\}$ and $G_1 = \{k+1, k+2,..., m\}$. The k is called image segmentation threshold, G_0 is foreground class which contains all of the gray value less than pixels k and the total pixels is:

$$S_0 = \sum_{i=1}^{k} n_i$$

The G_1 called background class contains all the pixel gray value greater than k and the sum of the pixels is:

$$S_1 = \sum_{i=k+1}^{m} n_i$$

The probability of appearing of G_0 and G_1 is respectively ψ_0 and ψ_1 calculated by Eq. 3:

$$\begin{cases} \psi_0 = \dfrac{s_0}{N} = \dfrac{\sum\limits_{i=1}^{k} n_i}{N} = \sum\limits_{i=1}^{k} p_i = \psi(k) \\[4mm] \psi_1 = \dfrac{s_0}{N} = \dfrac{\sum\limits_{i=k+1}^{m} n_i}{N} = \sum\limits_{i=k+1}^{m} p_i = 1 - \psi(k) \end{cases} \tag{3}$$

Let μ_0 and μ_1 are the average gray value of G_0 and G_1, they describe the cluster center of two classes and can be calculated through Eq. 4:

$$\begin{cases} \mu_0 = \sum\limits_{i=1}^{k} \dfrac{n_i p_i}{\psi_0} = \sum\limits_{i=1}^{k} \dfrac{n_i p_i}{\psi(k)} \\[4mm] \mu_1 = \sum\limits_{i=k+1}^{m} \dfrac{n_i p_i}{\psi_1} = \sum\limits_{i=k+1}^{m} \dfrac{n_i p_i}{(1 - \psi(k))} \end{cases} \tag{4}$$

Average gray value μ of the whole image is calculated by Eq. 5:

$$\mu = \psi_0 \mu_0 + \psi_1 \mu_1 \tag{5}$$

Between-class variance describing class distance can be calculated by Eq. 6:

$$\sigma^2(k) = \psi_0(\mu_0 - \mu)^2 + \psi_1(\mu_1 - \mu)^2 \tag{6}$$

Equation 7 is the solution of Eq. 5 and 6:

$$\sigma^2(k) = \psi_0 \psi_1(\mu_0 - \mu_1)^2 \tag{7}$$

Obviously, the greater the distance between classes is, the more open two classes is and the better segmentation is. That the optimal threshold value is shown in Eq. 8:

$$K^* = \underset{0 < k \leq m}{\text{Arg max}}\, \sigma^2(k) \tag{8}$$

Best segmentation can be get when maximum threshold value K^* is used to segment an image.

AUTOMATIC THRESHOLD IMAGE SEGMENTATION BASED ON IMPROVED GENETIC ALGORITHM

In this study, an image segmentation using automatic selected threshold method based on improved genetic algorithm is presented. It can use the characteristics of rapid global optimization of improved genetic algorithm.

Brief introduction of GA: The GA works using analogies to the patterns of natural selection and reproduction that are displayed in biological populations. This conception presented by Holland for

computer science applications. It also has been extended to engineering optimization by Goldberg and others and has been accepted as a global optimization technique. The basic idea of GA and the present real-coded GA is described in this study.

One of the analogies to biological populations is the use of genes and chromosomes to represent each individual in a population of designs. The variables that describe a design solution are mapped or coded into binary strings of ones and zeros, these strings are the genes of a design solution. The continuous variables are represented with binary strings; they are actually represented as discrete values. These variables are assigned a maximum and minimum value and the resolution between each discrete value depends on the number of bits used to represent the variables. This causes the discrepancy between the binary representation space and the actual real representation space.

The real-coded method is adopted for the vanes diffuser optimum design in the present study. The use of the real-coded method for design parameters is thought to be an efficient approach to this discrepancy. The floating number representation is robust, accurate and efficient, since it is conceptually closest to the real design space. In the present real-coded GA, a chromosome is coded as a finite-length string of the floating numbers corresponding to the design variables (Carson *et al.*, 2002).

Improvement of GA: In order to use GA to apply for image segmentation automatically selecting optimal space and quickly finding the optimal threshold, in this study, GA is improved as following; firstly, the threshold value of each space are decomposed the minimum size threshold such as each pixel of a frame image. And their codes are used as chromosome of GA. Then fitness function is defined by Eq. 9:

$$f(x_{(i,j)}) = (x_{(i,j-1)} - x_{(i,j)})$$
(9)

where, $f(x_{(i,j)})$ is threshold fitness value of the second i minimum granularity at the moment of j. The $x_{(i,j-1)}$ and $x_{(i,j)}$ is the second i minimum granularity at the moment of j-1 and j severally.

- **Improved selection operation:** Selection operation is used to determine which of the individuals in a generation will survive to become parents for the next generation. The Boltzmann selection method (Boykov and Funka-Lea, 2006) is adopted in the present real coded GA

The Boltzmann selection uses the temperature parameter of simulated annealing to implement a changeable selective pressure. The Boltzmann selection is applied to select parents and its procedure is as following:

- Transform the fitness of individual into the Boltzmann fitness by Eq. 10:

$$\phi(x_{(i,j)}) = e^{f(x_{(i,j)})T}$$
(10)

where, $f(x_{(i,j)})$ is the fitness of individual, $\varphi(x_{(i,j)})$ is the Boltzmann fitness of individual, T is temperature parameter which descend during generation increase. The value of T can be defined as in Eq. 11:

$$T = \max\{[1 - e^{(1+k-kK)^{\alpha}}]T_0, \beta\}$$
(11)

where, k and K are the local and maximum generation, α is one control parameter which can control velocity of constringency of GA, T_0 is the original temperature and it equals to ten times of the maximum fitness of the first generation, β is a small positive constant parameter and it is initialized 2.5 or 0.01 at present study.

- Select the individuals using the Roulette Wheel selection according to the Boltzmann fitness of individuals

Forepart in the evolution of group of temporal relation datum, the Boltzmann selection has a high tolerance and the search is carried out over a large portion of the design space. The diversity of population is maintained in the early generations, then in the procedure. When the best optimization values have been located and partially refined, the Boltzmann selection has a lower tolerance and it eliminates the smaller optimization values and concentrates on refining the better ones.

- **Improved crossover and mutation operation:** Once the parents have been selected, the next generation is formed via., crossover and mutation operations. In this process, two children are created from their parents. The approach used in this effort is the simulated binary crossover (Wang *et al.*, 2010b) and non-uniform mutation (Grady, 2006)

A probability distribution around two parents to create two children solutions is used in the simulated binary crossover. There are two aspects that give the self-adaptive power of a real-coded GA:

- Children's solutions closer to parent's solutions are more likely to be created
- The span of children's solutions is proportional to the span of parent's solutions

The procedure for creating two children solutions $\psi_{(i,j)}^{k}$ and $\psi_{(i,j+1)}^{k}$ from parent's solutions $\psi_{(i,j)}^{k+1}$ and $\psi_{(i,j+1)}^{k+1}$ is described as followings:

Step 1: Choose a random number ran$\in$[0, 1]
Step 2: Calculate λ by Eq. 12:

$$\lambda = \begin{cases} (2\ \text{ran})^{(\sigma+1)} & \text{ran} \in [0,\ 0.5] \\ [1/(2-2\ \text{ran})]^{(\sigma+1)} & \text{ran} \in (0.5,\ 1] \end{cases} \tag{12}$$

where, σ is the distribution index and any nonnegative real number and it is set 1.0 in this study. A larger value of σ makes it higher probability to create near parent's solutions and a small value of distant solutions which are allowed to be selected as children's solutions.

Step 3: Compute the solutions of children by Eq. 13:

$$\begin{cases} \psi_{(i,j)}^{k+1} = 0.5[(1+\lambda)\psi_{(i,j)}^{k} + (1-\lambda)\psi_{(i,j+1)}^{k}] \\ \psi_{(i,j+1)}^{k+1} = 0.5[(1-\lambda)\psi_{(i,j)}^{k} + (1+\lambda)\psi_{(i,j+1)}^{K}] \end{cases} \tag{13}$$

Non-uniform mutation is used to examine the design space. If a child $\psi_{(i,j)}^k$ is to be mutated, the new value $\psi_{(i,j)}^k{}'$ is randomly generated within the limitations of variables by Eq. 14:

$$\psi_{(i,j)}^k{}' = \begin{cases} \psi_{(i,j)}^k + (Max - \psi_{(i,j)}^k)ran(1-k/K)^\eta & ran \le 0.5 \\ \psi_{(i,j)}^k + (\psi_{(i,j)}^k - Min)ran(1-k/K)^\eta / & ran \le 0.5 \end{cases} \tag{14}$$

where $ran \epsilon [0,1]$, η is the refinement parameter and it is set 6.0 in this study. Min and Max are the lower and upper bounds of $\psi_{(i,j)}^k$. This adaptive mutation offers a clear balance between exploration and accuracy. At the first stages of algorithm, we set $k/K \approx 0$ which leads to a wide exploration whereas the last stages are devoted to refinement.

Each resulted child's chromosome is the combination of information inherited from its parents. Because the parents are selected as good designs, combining chromosomes from their parents usually (but not always) results in better child designs. When a good child is created, it will win in the selection process and then pass on parts of its chromosome to future generations. Poor child designs will not survive and chromosome patterns corresponding to poor performance will be removed from the population. Through this process of combination, good chromosome is adapted to form children. The better of these children are selected as parents for the next generation and then the GA evolves patterns of information in the chromosomes corresponding to designs with good fitness value. This process provides most of the GA's optimization power (Yezzi *et al.*, 2002).

The crossover operation does much of the work to find optimal designs but the mutation operation assists in the design space search. After the child's strings have been formed, a non-uniform mutation which will change the value of design variables may occur in the design space. This mutation can allow a new information pattern to be introduced in a chromosome that was not present in a child's parents. As in nature, the mutation process in the GA occurs with low probability. The probability of mutation is set 0.02 in this study.

- **Implementation process of improved genetic algorithm:** To begin a GA run, an initial generation of designs is created at random in the design space. Deviating slightly from natural populations, this population size remains constant from one generation to the next. Maintaining a constant population size is not a strict requirement but it makes the coding of GA simpler

To assign fitness values, the Navier-Stokes solver is used to obtain the fitness of each individual in our study. The selection operation is used to pick up the individuals from the population and make them to become parents for the next generations according to the individual's fitness. Then the process continues through crossover and mutation to form new generations. With the selection operation favoring good individuals and the crossover operation combining the features of good individuals, the GA population moves toward the global optimal design.

Because no derivatives are used and the search is conducted with a population of design points rather than a single point-to-point search. The GA is unlikely to stop its search at local optimization values. The best fitness individual encountered during the run of GA provides a near globally optimal design. For computational simplicity in this implementation, the assigned maximum generation's number is used in the application of GA in the present study.

Many variants of GA have been developed to accelerate the convergence speed and minimize computational expense. The two variant strategies which are used are the elitism and generation

gap in our study. The strategy of elitism forces GA to keep intact the best individual of the current population into the next generations. Thus, the best individual can not be destroyed through crossover and mutation and can only be replaced by a better individual when one is created.

The generation gap means that the assigned percentage of parent chromosomes directly turns into the child chromosomes without undergoing the genetic operation of the crossover and mutation. These child chromosomes compete with the other new generated child chromosomes in the successive generation. The better individual information may be kept in the use of the strategy of generation gap. The convergence of GA can be accelerated and the computational time for the evaluation of individual fitness can be decreased by this strategy. This strategy is very fit for the real datum optimization problems, because the measurement of individual is time-consuming.

Specific operations for image segmentation using automatic selected threshold method based on improved genetic algorithm: In the process of image segmentation using automatic selected threshold method based on improved genetic algorithm, following calculation process can be performed to achieve automatic threshold selection and segmentation purposes:

Step 1: Initialize populations

Step 2: Calculate value of individual fitness and the total value of population's fitness and their proportion by making use of Eq. 9 and 10 and be ready for the Roulette Wheel selection

Step 3: Choose two individuals by using the Roulette Wheel selection and regard them as parents of genetic algorithm

Step 4: Select intersection randomly and carry out operation for intercross and work out the filial generations of operation for crossover by using Eq. 13

Step 5: Execute operation for variation according to the probability and figure out the filial generations of operation for mutation by using Eq. 14

Step 6: Put the filial generations which are gained in the course of intercross and variation into the next populations

Step 7: If the scale of the next populations does not satisfy request, go to step 3

Step 8: Come into being new generations

Step 9: If the feasible solutions do not find or the times of circulation are not maximal, go to step 2

Step 10: Finding the optimal solution, after this solution is stored, the purposes of automatic threshold selection is achieved

Step 11: Image will be segmented by using threshold selected out to get ideal image segmentation

SIMULATION RESULTS AND ANALYSIS

Otsu method and the algorithm proposed in this study are used to segment image. Figure 1 and 2 show the effect of segmentation of rice and cameraman image.

Images added Gaussian noise and segmentation renderings of Otsu and algorithm proposed in this study are shown in Fig. 3. Images added salt and pepper noise and segmentation renderings of Otsu and algorithm proposed in this study are shown in Fig. 4. As can be seen from Fig. 3 and 4, the proposed algorithm has better noise immunity in the process of image segmentation.
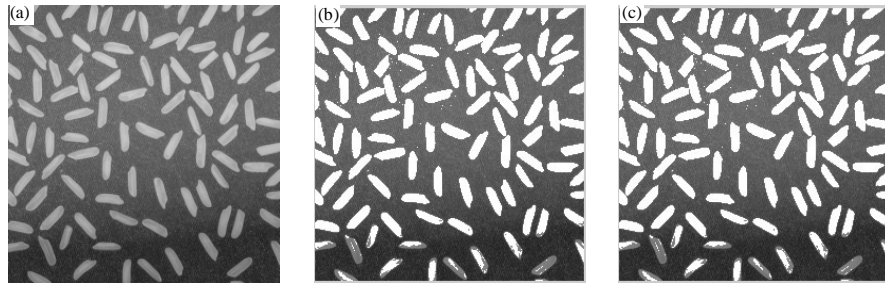
Fig. 1(a-c): Segmentation renderings of rice, (a) Original image, (b) Otsu algorithm and (c) Algorithm proposed in this study
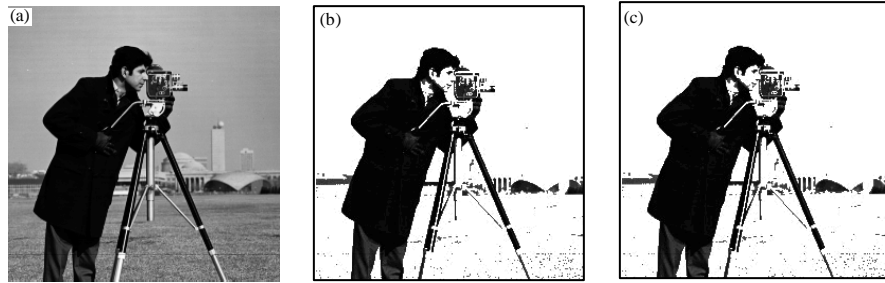

Fig. 2(a-c): Segmentation renderings of cameraman, (a) Original image, (b) Otsu algorithm and (c) Algorithm proposed in this study
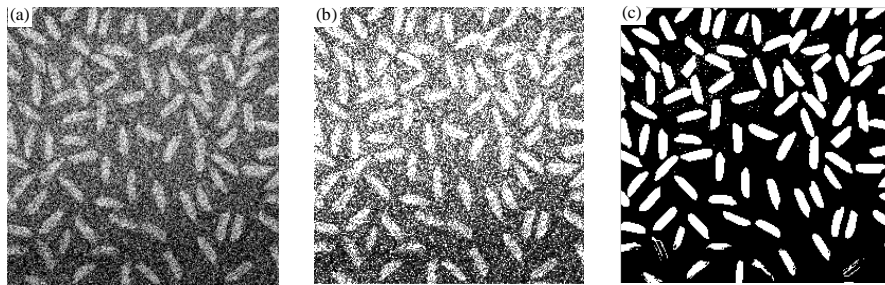

Fig. 3(a-c): Segmentation renderings of rice added Gaussian noise, (a) Noise image, (b) Otsu algorithm and (c) Algorithm proposed in this study
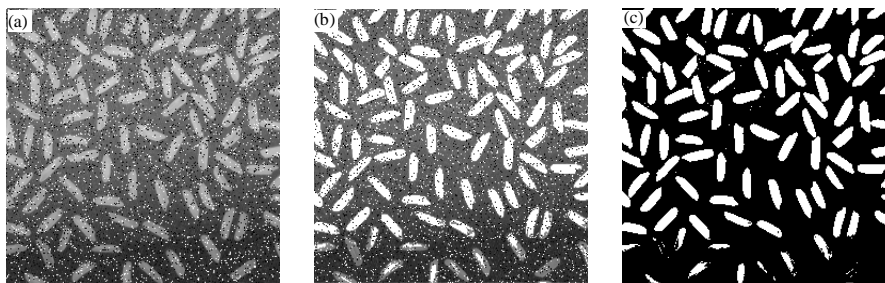

Fig. 4(a-c): Segmentation renderings of rice added salt and pepper noise, (a) Noise image, (b) Otsu algorithm and (c) Algorithm proposed in this study

Table 1: Comparison of computing time and threshold of Otsu method and algorithm proposed in this study

Image	Iterations	Algorithm		Otsu	
		Computing time (sec)	Threshold	Computing time (sec)	Threshold
Rice	28	0.252	112	0.422	132
Cameraman	30	0.238	85	0.344	89
Rice (added salt and pepper noise)	28	0.229	113	0.271	138
Rice (added Gaussian noise)	28	0.214	112	0.263	117

Comparison of computing time and threshold of Otsu method and threshold image segmentation algorithm based on improved genetic algorithm is shown in Table 1.

The results show that the computing time of segmentation algorithm based on improved genetic algorithm is less than Otsu method and it owns stronger anti-noise ability than Otsu method.

CONCLUSION

In this study, improved genetic algorithm is introduced into automatically threshold selection using the characteristics of rapid global optimization of improved genetic algorithm based on researching segmentation algorithm of Otsu. In the concrete realization of improved genetic operators, relevant policies are introduced to ensure the smooth of solution and global convergence. The experiment results show that when the improved genetic algorithm is introduced into image segmentation, the computing time of image segmentation can be greatly shorten and noise immunity of image segmentation algorithm can be improved. Thereby image segmentation using automatic selected threshold method based on improved genetic algorithm can facilitate the subsequent processing in computer vision and it can be applied to real-time image segmentation.

ACKNOWLEDGMENT

REFERENCES

Boykov, Y. and G. Funka-Lea, 2006. Graph cuts and efficient N-D image segmentation. Int. J. Comput. Vision, 70: 109-131.

Carson, C., S. Belongie, H. Greenspan and J. Malik, 2002. Blobworld: Image segmentation using expectation-maximization and its application to image querying. IEEE Trans. Pattern Anal. Mach. Intel., 24: 1026-1038.

Felzenszwalb, P.F. and D.P. Huttenlocher, 2004. Efficient graph-based image segmentation. Int. J. Comput. Vision, 59: 167-181.

Grady, L., 2006. Random walks for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell., 28: 1768-1783.

Vese, L.A. and T.F. Chan, 2002. A multiphase level set framework for image segmentation using the Mumford and Shah model. Int. J. Comput. Vision, 50: 271-293.

Wang, Z., S. Li, Y. Lv and K. Yang, 2010a. Remote sensing image enhancement based on orthogonal wavelet transformation analysis and pseudo-color processing. Int. J. Comput. Intell. Syst., 3: 745-753.

Wang, Z.W., S.Z. Li, S.Z. Su, M.Z. Liu and Q.X. Cai, 2010b. A quickly-speed running watermarking algorithm based on quantum evolutionary algorithm. J. Optoelectron. Laser, 21: 737-742.

Wang, Z. and S. Li, 2011. Face recognition using skin color segmentation and template matching algorithms. Inform. Technol. J., 10: 2308-2314.

Yezzi, Jr. A., A. Tsai and A. Willsky, 2002. A fully global approach to image segmentation via., coupled curve evolution equations. J. Visual Commun. Image Representation, 13: 195-216.