Multilevel Image Thresholding Selection Using the Artificial Bee Colony Algorithm

Ming-Huwi Horng and Ting-Wei Jiang

Department of Computer Science and Information Engineering,
National Pingtung Institute of Commerce, PingTung, Taiwan
{horng,mh.horng}@npic.edu.tw

Abstract. Image thresholding is an important technique for image processing and pattern recognition. The maximum entropy thresholding (MET) has been widely applied. A new multilevel MET algorithm based on the technology of the artificial bee colony (ABC) algorithm is proposed in this paper called the maximum entropy based artificial bee colony thresholding (MEABCT) method. Three different methods, such as the methods of particle swarm optimization, HCOCLPSO and honey bee mating optimization are also implemented for comparison with the results of the proposed method. The experimental results manifest that the proposed MEABCT algorithm can search for multiple thresholds which are very close to the optimal ones examined by the exhaustive search method. Meanwhile, the results using the MEABCT algorithm is the best and its computation time is relatively low compared with other four methods.

Keywords: Maximum entropy thresholding, artificial bee colony algorithm, particle swarm optimization, honey bee mating optimization.

1 Introduction

Thresholding is one of the most important techniques for performing image segmentation.. The nonparametric approaches are widely used methods to select for multilevel thresholds. These approaches find the thresholds that separate the gray-level regions of an image in an optimal manner based on some discriminating criteria such as the between class variance, entropy and cross entropy. The popular method, Otsu's method [1], selected optimal thresholds by maximizing the between class variance. However, inefficient formulation of between class variance makes the methods very time consuming in multilevel threshold selection. To overcome this problem, Liao *et al.* [2] proposed a fast recursive algorithm, Fast Otsu's method, along with a look-up-table to implement in the application of multilevel thresholding. Kapur *et al.* [3] proposed a method for gray-level picture thresholding using the entropy of the histogram. Zhang *et al.* [4] adopted the particle swarm optimization algorithm to maximize the entropy for underwater image segmentation. Madhubanti *et al.* [5] proposed a hybrid cooperative-comprehensive learning based PSO algorithm (HCOCLPSO) based on maximum entropy criterion. Yin [6] developed a recursive programming techniques to reduce the order of magnitude of computing the multilevel thresholds and further used the particle swarm optimization (PSO) algorithm to minimize the

F.L. Wang et al. (Eds.): AICI 2010, Part II, LNAI 6320, pp. 318–325, 2010.

cross entropy. Horng [7] and Jiang [10] applied the honey bee mating optimization (HBMO) to search for the thresholds of histogram of image.

The artificial bee colony (ABC) algorithm is a new swarm-based approach for optimization, in which the search algorithm is inspired by the foraging behavior of bee colony. Recently, Karaboga et al. [8, 9] had proposed a developed model of artificial bee colony (ABC) algorithm that simulated these social behaviors of honey bees for searching for the numerical optimization problems. In this paper, we applied the ABC algorithm to search for the multilevel thresholds using the maximum entropy (MET) criterion. This proposed method is called the maximum entropy based artificial bee colony thresholding (MEABCT) algorithm. In the experiments of this paper, the exhaustive search method is conducted for deriving the optimal solutions for comparison with the results generated from MEABCT algorithm. Furthermore, the three different methods that are PSO, HCOCLPSO algorithm and HBMO methods are also implemented for comparison.

2 Maximum Entropy Artificial Bee Colony Thresholding Algorithm

The entropy criterion, proposed by Kapur et al [3], had been widely used in determining the optimal thresholding in image segmentation. The original algorithm had been developed for bi-level thresholding. The method can also extend to solve multilevel thresholding problems and can be described as follows.

Let there be L gray levels in a given image **I** and these gray levels are in the range {0, 1, 2,…, L-1}. Then one can define as $P_i = h(i)/N$, where

$h(i)$ denotes the number of pixels with gray-level i.

N denotes total number of pixels in the image.

Here, given a problem to select D thresholds, $[t_1, t_2, \ldots, t_D]$ for a given image **I**, the objective function f is to maximize:

$$
\begin{aligned}
& f([t_1, t_2, \ldots, t_D]) = H_0 + H_1 + H_2 + \ldots + H_D \\
& \omega_0 = \sum_{i=0}^{t_1-1} P_i, \quad H_0 = -\sum_{i=0}^{t_1-1} \frac{P_i}{\omega_0} \ln \frac{P_i}{\omega_0} \\
& \omega_1 = \sum_{i=t_1}^{t_2-1} P_i, \quad H_1 = -\sum_{i=t_1}^{t_2-1} \frac{P_i}{\omega_1} \ln \frac{P_i}{\omega_1} \\
& \omega_2 = \sum_{i=t_2}^{t_3-1} P_i, \quad H_2 = -\sum_{i=t_2}^{t_3-1} \frac{P_i}{\omega_2} \ln \frac{P_i}{\omega_2}, \ldots \\
& \omega_D = \sum_{i=t_D}^{t_D-1} P_i, \quad H_D = -\sum_{i=t_D}^{t_D-1} \frac{P_i}{\omega_D} \ln \frac{P_i}{\omega_D}
\end{aligned}
\tag{1}
$$

In this paper, a maximum entropy based artificial bee colony thresholding (MEABCT) algorithm is developed based on the meta-heuristic approach proposed by Karaboga [8]. This proposed algorithm tries to obtain this optimum D-dimensional vector $[t_1, t_2, \ldots, t_D]$, which can maximize (1). The objective function is also used as the fitness function for HEABCT algorithm. The details of MEABCT algorithm is introduced as follows.

Step 1. (Generate the initial population of solutions)

Generate the SN solutions z_i ($i = 1,2,...,SN$) with D dimensions denoted by matrix Z.

$$Z = [z_1, z_2, ..., z_{SN}] \text{, and } z_i = (z_{i,1}, z_{i,2}, ..., z_{i,D}) \tag{2}$$

where $z_{i,j}$ is the jth component value that is restricted into [0,…,L] and the $z_{i,j} < z_{i,j+1}$ for all j. The fitness of all solutions z_i are evaluated and set cycle=1, meanwhile, the trail number of each solution z_i , $trail_i$, is assigned to 0.

Step 2. (Place the employed bees on their food sources)

In the step 2, each employed bee produces a new solution v_i by using (3) and tests the fitness value of the new solution.

$$v_{ij} = z_{i,j} + \phi_{ij}(z_{i,j} - z_{k,j}) \tag{3}$$

where $k \in \{1,2,.....,SN\}$ but $k \neq i$ and $j \in \{1,2,....,D\}$ are randomly selected indexes. ϕ_{ij} is a random number between [-1, 1]. If the fitness of the new one is higher than that of the previous one, the employed memorizes the new position and forgets the old one. Otherwise it keeps the old solution.

Step 3. (Send the onlooker bees on the food sources depending on their nectar amounts)

In this step 3, we first calculate the probability value p_i of the solution z_i by means of their fitness values using (4).

$$p_i = \frac{fit(z_i)}{\sum_{i=1}^{SN} fit(z_i)} \tag{4}$$

An onlooker bee selects a solution to update its solution depending on the probabilities and determines a neighbor solution around the chosen one. The selection procedure for the first onlooker, a random number is produced between [0, 1] and if this number is less than p_1, its solution is selected for updating its solution. Otherwise, the random number is compared with p_2 and if less than that, the second solution is chosen. Otherwise, the third probability of third solution is checked. This process is repeated until all onlookers are distributed onto solutions. The distributed onlooker bee updates its own solution like the employed bees does.

Step 4. (Send the scouts to the search area for discovering new food source)

If the solution z_i does not be improved through the Step 2 and Step 3, the $trail_i$ value of solution z_i will be added by 1. If the $trail_i$ of solution is more than the predetermined "*limit*", the solution z_i is regarded to be an abandoned solution, meanwhile, the employed bee will be changed into a scout. The scout randomly produces the new solution by (5) and then compares the fitness of new solution with that of its old one. If the new solution is better than the old solution, it is replaced with the old one and set its own $trail_i$ into 0. This scout will be changed into employed bee. Otherwise, the old one is retained in the memory.

$$z_{ij} = z_{\min, j} + rand(0,1)(z_{\max, j} - z_{\min, j}) \tag{5}$$

where the $z_{\min, j}$ and $z_{\max, j}$ are the minimum and maximum thresholds of the jth component of all solutions, the $rand(0,1)$ is a random number generating function that produces the random number between [0, 1].

Step 5. (Record the best solution)
In this step, we memory the best solution so far and add the cycle by one.

Step 6. (Check the termination criterion)
If the cycle is equal to the maximum cycle number (MCN) then finish the algorithm, else go to Step 2.

3 Expermetal Results and Discussion

We implement the proposed MEABCT algorithm in language of Visual C++ under a personal computer with 2.4GHz CPU, 2G RAM with window XP system. All designed programs are designed by revising the original programs given in the homepage of artificial bee colony algorithm [11]. It is clear from the MEABCT algorithm that there are three control parameters: the number of food sources which is equal to the number of employed bees or onlooker bees *(SN*), the value of "*limit*" and the maximum cycle number (MCN). In all experiments, we select the colony size (*SN*) 100, MCN 200, and *limit* value 30. Three images of Fig. 1 named "LENA", "PEPPER", and "BIRD" are used for conducting our experiments.

In order to obtain the consistent comparisons, a popular performance indicator, peak signal to noise ratio (PSNR), is used to compare the segmentation results by using the multilevel image threshold techniques [12]. For the sake of completeness we define PSNR, measured in decibel (dB) as

$$PSNR = 20\log_{10}\left(\frac{255}{RMSE}\right) \quad \textbf{(dB)} \tag{6}$$

where the RMSE is the root mean-squared error, defined as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(I(i,j) - \hat{I}(i,j))^2}{MN}} \tag{7}$$

Here I and $\hat{I}$ are original and segmented images of size $M \times N$, respectively.

Firstly, we execute the MEABCT algorithm on partitioning the three images. For evaluating the performance of the proposed MEHBMOT algorithm, we have implemented this method on the three test images. Table 4 shows the selected thresholds

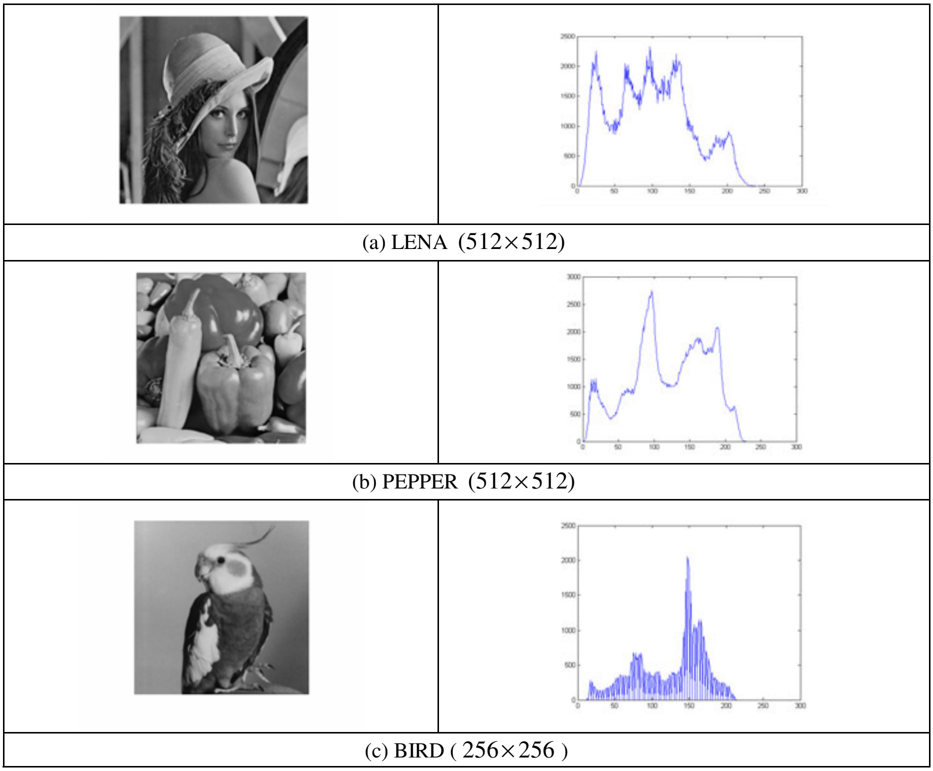

Fig. 1. The test images and corresponding histograms: (a) LENA, (b) PEPPER and (c) BIRD

Table 1. The selection thresholds for three test images by using the MEFFT and the exhaustive search method (k: number of thresholds)

Image (size)	k	Exhaustive		MEABCT	
		Thresholds	Computation time (ms)	Thresholds	Computation time (ms)
LENA	2	80,150	4.89	80,150	1.39
	3	60,109,160	158.49	60,109,160	5.94
	4	56,100,144,182	8290	56,100,144,182	24.39
	5	44,79,114,150,185	451304	44,79,115,148,186	189.35
PEPPER	2	74,146	3.73	74,146	1.45
	3	61,112,164	145.58	61,112,164	5.98
	4	57,104,148,194	7965	57,104,148,194	29.78
	5	42,77,113,153,194	439784	42,77,113,153,194	187.35
BIRD	2	71,138	4.13.	71,138	1.12
	3	70,129,177	132.67	70,129,177	2.89
	4	51,96,139,177	6564	51,94,138,178	17.85
	5	46,74,104,141,178	414789	46,74,105,142,177	109.35

derived by the MEABCT algorithm and the optimal thresholds generated from the exhaustive search method. We find that the selected thresholds of MEABCT algorithm are equivalent or very close to optimal thresholds derived by the exhaustive search methods. Furthermore, we find that the computation times of exhaustive search method grows exponentially with the number of required thresholds. Obviously, the computation needs for the exhaustive search are absolutely unacceptable for $k \geq 4$ (k: number of thresholds). The computation times of the MEABCT algorithm is significantly faster compared to the exhaustive search algorithm. The performance metrics for checking the effectiveness of the method are chosen as the computation time so as to get an idea of complexity, and the PSNR which is used to determine the quality of the thresholded images. Table 2 shows the selected thresholds, computation time, PSNR value and the corresponding fitness value of five test images with different thresholds. This table provides quantitative standard for evaluating. This table shows that the number of thresholds increase, the PSNand the fitness value are enlarged.

The MEABCT and other three multilevel thresholding methods that are MEHBMOT, PSO and HCOCLPSO algorithm are implemented for the purpose of comparisons. Table 3 shows the selected thresholds of the three test images. It is interesting that the selected thresholds by the MEABCT algorithm are equivalent (for 2- or 3-threshold problems) or very close (4- or 5-threshold problem) to the ones MEHBMOT algorithm. The thresholds obtained by PSO algorithms in the segmentation of BIRD image are also distinct from the one of the MEABCT algorithm in 5-level thresholding. It is possible to reveal that the PSO algorithm is unsuitable to search for thresholds. Table 4 shows the computation time and the corresponding PSNR values of the four different multilevel thresholding methods. Several aspects are found in the two tables. The computation time of the MEABCT algorithm is between the MEHBMOT and PSO in the segmentation of LENA, PEPPER and BIRD images. An aspect is found that the HCOCLPSO algorithm is not acceptable because of the heavy need of computation times. Finally, from the corresponding the fitness values of selected thresholds using MEABCT algorithm it appears the fact that the selected thresholds of the MEABCT algorithm can effectively find the adequate solutions based on the maximum entropy criterion.

Table 2. Thresholds, computation times, PSNR values and Fitness values for test images by using MEABCT algorithm.

Image	k	Thresholds	Computation time (ms)	PSNR (dB)	Fitness Value
LENA	2	*80,150*	*1.39*	*15.46*	*12.6990*
	3	*60,109,160*	*5.94*	*18.55*	*15.7658*
	4	*56,100,144,182*	*24.39*	*19.71*	*18.5875*
	5	*44,79,115,148,186*	*189.35*	*21.68*	*21.2468*
PEPPER	2	*74,146*	*1.45*	*16.47*	*12.6348*
	3	*61,112,164*	*5.98*	*18.42*	*15.6892*
	4	*57,104,148,194*	*29.78*	*19.21*	*18.5397*
	5	*42,77,113,153,194*	*187.35*	*21.81*	*21.2830*
BIRD	2	*71,138*	*1.12*	*17.44*	*11.1647*
	3	*70,129,177*	*2.89*	*18.53*	*13.8659*
	4	*51,94,138,178*	*17.85*	*20.84*	*16.4558*
	5	*46,74,105,142,177*	*109.35*	*22.72*	*18.6961*

Table 3. The selected thresholds used the four different image thresholding algorithms

Image	k	Selected thresholds			
		MEABCT	MEHBMOT	PSO	HCOCLPSO
LENA	2	80,150	80,150	80,150	80,150
	3	60,109,160	60,109,160	60,109,160	60,109,160
	4	56,100,144,182	56,100,144,182	56,100,144,182	56,100,144,182
	5	44,79,115,148,186	44,80,115,150,185	43,79,114,150,185	46,83,118,153,187
PEPPER	2	74,146	74,146	74,146	74,146
	3	61,112,164	61,112,164	72,135,193	61,112,164
	4	57,104,148,194	57,104,148,194	58,105,148,194	57,104,148,194
	5	42,77,113,153,194	42,77,113,153,194	43,77,113,153,194	42,77,114,154,194
BIRD	2	71,138	71,138	71,138	71,138
	3	70,129,177	70,129,177	70,129,177	70,130,177
	4	51,96,139,177	51,96,139,177	51,94,138,177	51,96,140,177
	5	46,74,104,141,177	46,74,104,141,177	51,96,139,177,248	44,71,97,139,177

Table 4. The computation times and the corresponding PSNR of the four different multilevel thresholding methods

Image	k	Computation times (ms)/PSNR(dB) (k: number of thresholds)							
		MEABCT		MEHBMOT		PSO		HCOCLPSO	
LENA	2	1.39	15.46	1.45	15.46	1.36	15.46	1.69	15.46
	3	5.94	18.55	6.95	18.55	4.89	18.55	13.58	18.55
	4	24.39	19.71	23.65	19.71	25.69	19.71	169.5	19.71
	5	189.35	21.68	432.6	21.63	137.56	21.61	1158	21.56
PEPPER	2	1.45	16.47	1.87	16.47	1.56	16.47	2.26	16.47
	3	5.98	18.42	6.78	18.42	5.23	17.40	18.43	18.42
	4	29.78	19.21	36.76	19.21	28.43	19.23	219.6	19.21
	5	187.35	21.81	234.9	21.81	154.26	21.39	1086	21.41
BIRD	2	1.12	17.44	1.09	17.44	1.15	17.44	2.10	17.44
	3	2.89	18.53	3.94	18.53	3.17	18.23	15.28	18.34
	4	17.85	20.84	18.65	20.77	19.94	20.73	132.5	20.89
	5	109.35	22.72	106.1	22.65	113.97	20.77	1153	22.20

4 Conclusion

In this paper, we have proposed a method, called MEABCT algorithm, for multilevel thresholds selection using the maximum entropy criterion. The MEABCT algorithm simulates the behavior of honey bee mating to develop the algorithm to search for the adequate thresholds for image segmentation. The MEABCT algorithm is demonstrated that it can rapidly converge. The segmentation results are promising and it encourage further researches for applying the MEABCT algorithm to complex and real-time image analysis problem such as the automatic target recognition and the complex document analysis.

Acknowledgments. Authors would like to thank the National Science council, ROC, under Grant No. NSC 98-2221-E-251-004 for support of this work.

References

1. Otsu: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, Cybernetics 9, 62–66 (1979)
2. Liao, P.S., Chen, T.S., Chung, P.C.: A fast algorithm for multilevel thresholding. Journal of Information Science and Engineering 17, 713–727 (2001)
3. Kapur, J.N., Sahoo, P.K., Wong, A.K.C.: A new method for gray-level picture thresholding using the entropy of the histogram. Computer Vision Graphics Image Processing 29, 273–285 (1985)
4. Zhang, R., Liu, L.: Underwater image segmentation with maximum entropy based on Particle Swarm Optimization (PSO). In: Proceedings of IMSCCS 2006, pp. 360–363 (2006)
5. Madhubanti, M., Amitava, A.: A hybrid cooperative-comprehensive learning based algorithm for image segmentation using multilevel thresholding. Expert Systems with Application 34, 1341–1350 (2008)
6. Yin, P.Y.: Multilevel minimum cross entropy threshold selection based on particle swarm optimization. Applied Mathematics and Computation 184, 503–513 (2007)
7. Horng, M.H.: A multilevel image thresholding using the honey bee mating optimization. Applied Mathematics and Computation 215, 3302–3310 (2010)
8. Karaboga, D., Basturk, D.: On the performance of artificial bee colony algorithm. Applied Soft Computing 8, 687–697 (2008)
9. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization; artificial bee colony (ABC) algorithm. J. Glob. Opyim. 39, 459–471 (2007)
10. Jiang, T.W.: The application of image thresholding and vector quantization using the honey bee mating optimization. Master thresis, National Ping Rung Institute of Commerce (2009)
11. Karaboga, D.: Artificial bee colony algorithm homepage, `http://mf.erciyers.edu.tw/abc/`
12. Yin, P.I.: Multilevel minimum cross entropy threshold selection based on particle swarm optimization. Applied Mathematics and Computation 184, 503–513 (2007)