TELKOMNIKA, Vol. 11, No. 9, September 2013, pp. 5229~5236
ISSN: 2302-4046

Multilevel Minimum Cross Entropy Thresholding using Artificial Bee Colony Algorithm

Ming-Huwi Horng
Department of Computer Science and Information Engineering, National PingTung Institute of Commerce
PingTung, TAIWAN
e-mail: horng@npic.edu.tw

Abstract

The minimum cross entropy thresholding (MCET) has been widely applied in image processing. In this paper, a new multilevel MCET algorithm based on the artificial bee colony (ABC) algorithm is proposed. The proposed thresholding algorithm is called ABC-based MCET algorithm. Four different methods including the exhaustive search, the honey bee mating optimization (HBMO), the particle swarm optimization (PSO) and the quantum particle swarm optimization (QPSO) methods are also implemented for comparison with the results of the proposed method. The experimental results demonstrate that the proposed ABC-based MCET algorithm can efficiently search for multiple thresholds that are very close to the optimal ones selected by using the exhaustive search method. Compared with the other three thresholding methods, the segmentation results using the ABC-based MCET algorithm is the best. It is promising to encourage further research for applying the HBMO algorithm to complex problems of image processing and pattern recognition.

Keywords*: image thresholding, artificial bee colony algorithm, particle swarm optimization, honey bee mating optimization, quantum particle swarm optimization*

1. Introduction

Image thresholding is a basic component of many computer vision applications [1-2]. While humans can easily differentiate an object from a complex background, image thresholding is a difficult task to separate them. The gray-level histogram of an image is usually considered as an efficient tool for developing the image thresholding algorithms. Nonparametric 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 selected the optimal threshold by maximizing the between class variance of gray levels of the object and the background portions [3]. However, the Otsu's method is one of the better threshold selection methods for real world images with regard to uniformity and shape measures [4]. The inefficient formulation of between class variance makes the methods very time consuming for the multilevel threshold selection. The hybrid cooperative-comprehensive learning PSO algorithm based on maximum entropy criterion is proposed to select thresholds [5].

The minimum cross entropy thresholding (MCET) algorithm has been widely adopted to search for the thresholds for image thresholding [6-7]. Yin [8] proposed a recursive programming technique which effectively reduces the magnitude of computing the minimum cross entropy objective function in the multilevel thresholding applications. Then, a particle swarm optimization (PSO) algorithm was adopted to search for the near-optimal thresholds. The honey bee mating optimization (HBMO) was applied to search for the thresholds of histogram of image. The experimental results demonstrated that the result of using HBMO algorithm was superior to other algorithms such as the PSO, HCOLPSO and Fast Otsu's methods [9].

Over the last decade, modeling the behavior of social insects, such as ants and bees, for the purpose of search and problems solving has been the context of the emerging area of swarm intelligence. The artificial bee colony (ABC) algorithm may also be considered as a typical swarm-based approach for optimization, in which the search algorithm is inspired by the foraging behavior of bee colonies. Karaboga and Basturk [10] (2008) have recently proposed a developed model of the artificial bee colony (ABC) algorithm that simulated these social

Received March 7, 2013; Revised June 8, 2013; Accepted June 20, 2013

behaviors of honey bees for searching for the numerical optimization problems. This paper applies the ABC algorithm to search for the multilevel thresholds using the minimum cross entropy (MCET) criterion. This proposed method is called the artificial bee colony-based MCET (ABC-based MCET) algorithm. In the experiments presented in this paper, the exhaustive search method is conducted to derive the optimal solutions for comparison with the results generated from ABC-based MCET algorithm. The three different methods including the honey bee mating optimization (HBMO), the PSO and the QPSO algorithms are implemented in the several real images for purposes of comparison.

2. Research Method

The proposed algorithm has two main phases. The first phase involves generating the objective function based on cross entropy for later developing the ABC algorithm. The objective function is revised to act as the fitness function of this ABC algorithm. The second phase introduces the ABC algorithm for multi-level image thresholding based on the minimum cross entropy.

2.1. Cross Entropy Measure Criterion

The cross entropy was first developed for measuring information context. Let $F=\{f_1, f_2,..., f_N\}$ and $G=\{g_1, g_2,..., g_N\}$ be two probability distributions on the same set. The cross entropy between F and G is defined by:

$$D(F,G)=\sum_{i=1}^{N} f_i \log\frac{f_i}{g_i} \tag{1}$$

The minimum cross entropy thresholding algorithm selects several thresholds by minimizing the cross entropy between the original image and the resulting image. Let **I** be the original image and $h(i)$, $i=1,2,...,L$, be the corresponding histogram with L being the number of gray levels. Then the resulting image, denoted by I_t, using t as the threshold value is constructed by:

$$I_t(x,y)=\begin{cases} \mu(1,t) & I(x,y)<t, \\ \mu(t,L+1) & I(x,y)\ge t, \end{cases} \tag{2}$$

Where:

$$\mu(a,b)=\sum_{i=a}^{b-1} ih(i) \Big/ \sum_{i=a}^{b-1} h(i)$$

The cross entropy is then calculated by:

$$D(t)=\sum_{i=1}^{L} ih(i)\log(i)-\sum_{i=1}^{t-1} ih(i)\log(\mu(1,t))- \sum_{i=t}^{L} ih(i)\log(\mu(t,L+1)). \tag{3}$$

The minimum cross entropy thresholding algorithm determines the optimal threshold t^* by minimizing the cross entropy based on Equation (4).

$$t^*=\arg\min_t\{D(t)\}. \tag{4}$$

Since the first term is constant for a given image, the objective function can be re-written as:

$$\eta(t) = -\sum_{i=1}^{t-1} ih(i)\log(\mu(1,t)) - \sum_{i=t}^{L} ih(i)\log(\mu(t,L+1))$$

$$= -(\sum_{i=1}^{t-1} ih(i))\log(\frac{\sum_{i=1}^{t-1} ih(i)}{\sum_{i=1}^{t-1} h(i)}) - (\sum_{i=t}^{L} ih(i))\log(\frac{\sum_{i=t}^{L} ih(i)}{\sum_{i=t}^{L} h(i)}) \quad (5)$$

Where $m^0(a,b) = \sum_{i=a}^{b-1} h(i)$ and $m^1(a,b) = \sum_{i=a}^{b-1} ih(i)$ are the zero-moment and first-moment on partial range of the image histogram.

Assume that it is required to select c thresholds denoted by $t_1, t_2, t_3, ..., t_c$. For the convenience of computation, the two dummy thresholds $t_0 = 1,\ t_{c+1} = L+1$ are used. The objective function then becomes:

$$\eta(t_1, t_2, ..., t_c) = -\sum_{i=1}^{c+1} m^1(t_{i-1}, t_i)\log(\frac{m^1(t_{i-1}, t_i)}{m^0(t_{i-1}, t_i)}). \quad (6)$$

In our proposed algorithm, we try to obtain this optimum c-dimensional vector $[t_1, t_2, t_3, ..., t_c]$ which can minimized the Equation (6). The ABC algorithm is usually designed to solve maximization problems, thus, we take the reciprocal of $\eta(t_1, t_2, ..., t_c)$ as the fitness function and try to maximize it.

2.2. ABC-based MCET Algorithm

In this paper, a minimum cross entropy based artificial bee colony thresholding algorithm is developed based on the meta-heuristic approach (Karaboga and Basturk, 2008). The details of ABC-based MCET algorithm are 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.

$$Z = [z_1, z_2, ..., z_{SN}],\ z_i = (z_{i,1}, z_{i,2}, ..., z_{i,D}) \quad (7)$$

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 is evaluated and then set cycle=1 and the trail number of each solution z_i and $trail_i$ are equal to 0.

Step 2. Place the employed bees on their food sources

In Step 2, each employed bee produces a new solution v_i by using (8) and computes the fitness value of the new solution. If the fitness of the new one is higher than that of the previous one, the employed bee memorizes the new position and forgets the old one; otherwise the employed bee keeps the old solution.

$$v_{ij} = z_{ij} + \phi_{ij}(z_{ij} - z_{kj}) \quad (8)$$

Step 3. Send the onlooker bees to the food sources depending on their amount of nectar

In Step 3, we first calculate the probability value p_i of the solution z_i by means of their fitness values using (9). An onlooker bee selects a solution to update its solution depending on the probabilities and determines a neighbor solution around the chosen one. In the selection procedure for the first onlooker, a random number is produced between [0, 1] and if this number is less than p_1, the solution is updated. 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 have been distributed to solutions. The distributed onlooker bee updates its own solution just as the employed bees do.

$$p_i = \frac{fit(z_i)}{\sum_{i=1}^{SN} fit(z_i)} \tag{9}$$

Step 4. Send the scouts to the search area to discover new food sources

If the solution z_i is not improved through the Step 2 and Step 3, the $trail_i$ value of solution z_i will be increased by 1. If the $trail_i$ of solution is more than the predetermined "limit" the solution z_i is considered to be an abandoned solution, meanwhile, the employed bee will be changed into a scout. The scout randomly produces the new solution by (10) 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 an employed bee. Otherwise, the old one is retained in the memory.

$$z_{ij} = z_{\min,j} + rand(0,1)(z_{\max,j} - z_{\min,j}),\ j = 1,2,...,D. \tag{10}$$

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

Step 5. Record the best solution

In this step, the best solution so far is recorded and increases the cycle by 1.

Step 6. Check the termination criterion

If the cycle is equal to the maximum cycle number (MCN) then the algorithm is finished; otherwise go to Step 2.

3. Results and Discussion

We implement the all of the algorithms in Visual C++ 6.0 on a personal computer with 2.4GHz CPU, 1G RAM running window XP system. The designed programs are revised from ones given by the homepage of artificial bee colony algorithm [11]. Three control parameters that are 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) are set in the colony size (*SN*) 50, MCN 100, and *limit* value 100. The initial solutions of the four thresholding methods are assigned to be 50. Five images named "LENA", "PEPPER", "BIRD", "CAMERA", and "GOLDHILL" are used for conducting our experiments. These original test images and their histograms are shown in Figure 1.

The popular performance indicator, peak signal to noise ratio (PSNR), is used to compare the segmentation results by using the multilevel image threshold techniques [6]. For the sake of completeness we define PSNR, measured in decibel (dB) as:

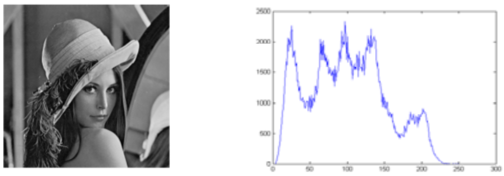
$$PSNR = 20\log_{10}\left(\frac{255}{RMSE}\right) \quad \textbf{(dB)} \tag{11}$$

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

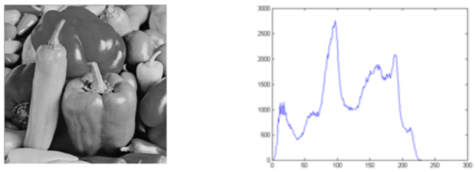
$$RMSE = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(I(i,j) - \hat{I}(i,j))^2}{MN}} \tag{12}$$

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

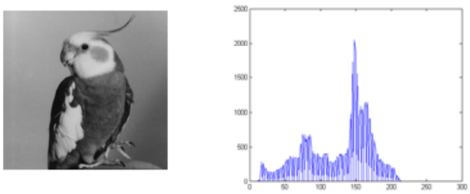
First, we execute the ABC-based MCET algorithm on partitioning the five test images. The exhaustive search is also conducted for deriving the optimal solution for comparison. Table 1 shows the selected thresholds derived by the ABC-based MCET algorithm and the optimal thresholds generated from the exhaustive search method. We find that the selected thresholds of ABC-based MCET 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 $T \geq 4$ (T: number of thresholds). The computation times of the ABC-based MCET algorithm is significantly faster compared to the exhaustive search algorithm.



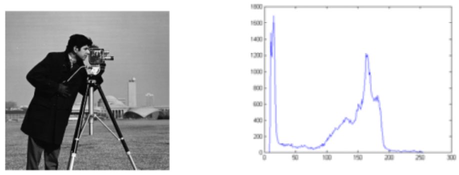
(a) LENA (512×512)



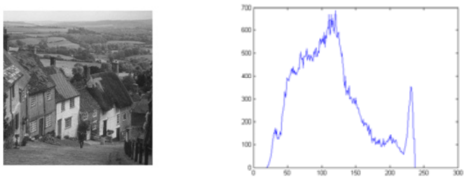
(b) PEPPER (512×512)



(c) BIRD (256×256)



(d) CAMERA (256×256)



(e) GOLDHILL (256×256)

Figure 1. The Test Images and Corresponding Histograms: (a) LENA, (b) PEPPER, (c) BIRD, (d) CAMERA and (e) GOLDHILL

Table 1. The Selection Thresholds for Five Test Images using Exhaustive Search and ABC-based MCET Algorithm

Image	k-thresholds	Exhaustive Search		ABC-based MCET	
		Optimal thresholds	Execution time(ms)	Optimal thresholds	Execution time(ms)
Lena	2	53,117	4.89	53,117	1.34
	3	46,95,150	138.36	46,95,150	5.43
	4	40,77,114, 160	7243	40,77,114,160	26.39
	5	29,53,83,117,161	431084	29,53,83,117,161	186.46
Pepper	2	52,125	4.98	52,125	1.48
	3	48,107,157	135.58	48,107,157	5.67
	4	35,75,117,163	7365	35,75,117,163	25.78
	5	34,71,104,136,171	439784	34,71,104,136,171	189.85
Bird	2	61,118	4.63.	61,118	1.45
	3	59,111,157	132.67	59,111,157	5.85
	4	45,83,122,160	7464	45,83,122,160	24.65
	5	37,66,97,132,164	414789	37,66,97,133,164	179.54
Camera	2	50,136	4.54	50,136	1.48
	3	29,82,143	139.87	29,82,143	5.12
	4	28,75,124,157	7456	28,75,124,157	26.28
	5	27,70,114,144,171	439757	27,70,114,144,171	184.26
Goldhill	2	85,149	4.57	85,149	1.39

3	70,109,163	143.45	70,109,163	5.45
4	62,94,130,179	7390	62,94,130,179	27.57
5	55,81,107,138,184	444569	55,81,107,138,184	187.63

Table 2. Selected Thresholds, Computing Time, PSNR and the Fitness Value for Test Images by using the ABC-based MCET Algorithm

Image (size)	K	thresholds	Execution time(sec)	PSNR(dB)	Fitness value(× 10^{-9})
Lena	2	53,117	1.34	16.05	-8.05339670116485
	3	46,95,150	5.43	18.32	-8.03604752777857
	4	40,77,114,160	26.39	20.39	-8.02813000289651
	5	29,53,83,117,161	186.46	21.78	-8.02452714885615
Pepper	2	52,125	1.48	15.31	-6.45796303585292
	3	48,107,157	5.67	17.90	-6.45058964705751
	4	35,75,117,163	25.78	20.22	-6.44529415933625
	5	34,71,104,136,171	189.85	21.99	-6.44328356330378
Bird	2	61,118	1.45	16.09	-2.46107827285174
	3	59,111,157	5.85	18.57	-2.45925032132233
	4	45,83,122,160	24.65	21.37	-2.45799402578107
	5	37,66,97,133,164	179.54	23.21	-2.45722342106554
Camera	2	50,136	1.48	16.09	-2.57228692895071
	3	29,82,143	5.12	18.94	-2.56952423601636
	4	28,75,124,157	26.28	21.37	-2.56855166716966
	5	27,70,114,144,171	184.26	22.89	-2.56796758423638
Goldhill	2	85,149	1.39	14.55	-2.80385427364911
	3	70,109,163	5.45	17.19	-2.80035448214976
	4	62,94,130,179	27.57	19.00	-2.79850184527408
	5	55,81,107,138,184	187.63	20.75	-2.79755275761192

Table 3. The Selected Thresholds for the Four Multilevel Threshold Selection Methods

Image (size)	K	Selected thresholds (k: number of thresholds)			
		ABC-based MCET algorithm	HBMO-based MCET algorithm	PSO-based algorithm	Quantum PSO-based algorithm
Lena	2	53,117	53,117	53,117	53,117
	3	46,95,150	46,95,150	47,97,154	45,91,143
	4	40,77,114,160	40,77,114,160	42,79,116,158	43,76,121,157
	5	28,52,83,117,161	28,52,83,117,161	24,54,79,117,170	30,55,92,107,157
Pepper	2	52,125	52,125	52,125	52,126
	3	48,107,157	48,107,157	48,108,158	45,106,158
	4	35,75,117,163	35,75,117,163	33,74,119,167	30,78,117,158
	5	34,71,104,136,171	34,71,104,136,171	34,71,106,138,172	34,72,111,140,173
Bird	2	61,118	61,118	61,118	61,119
	3	59,111,157	59,111,157	58,111,161	59,114,165
	4	45,83,122,160	45,83,122,160	45,87,127,161	53,82,114,159
	5	37,66,97,132,164	37,66,98,132,164	36,64,99,135,165	35,73,103,135,168
Gamera	2	50,136	50,136	50,136	50,138
	3	29,82,143	29,82,143	30,84,143	30,86,142
	4	28,75,124,157	28,75,124,157	28,75,124,157	32,80,117,155
	5	27,70,114,144,171	27,70,114,144,171	27,68,111,144,171	26,63,112,141,171
Goldhill	2	85,149	85,149	85,149	85,149
	3	70,109,163	70,109,163	71,111,166	68,106,163
	4	62,94,130,179	62,94,130,179	62,95,131,179	62,93,138,184
	5	55,81,107,138,184	55,81,107,138,184	56,82,108,145,190	53,82,105,133,187

For evaluating the performance of the proposed ABC-based MCET algorithm, we have implemented this method on the five test images. 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 PSNR and the fitness value are enlarged. The ABC-based MCET and other three multilevel thresholding methods that are HBMO-based MCET, PSO-based MCET, and QPSO-based MCET algorithms are implemented for the purpose of comparisons. Table 3 shows the selected thresholds of the five test images. It is interesting that the selected thresholds by the ABC-based MCET algorithm are approximately equivalent to the ones HBMO-based MCET algorithm; nevertheless, there are significant differences of selected thresholds with regard to the PSO-based MCET and the QPSO- based MCET algorithms. Only a threshold obtained by ABC-based MCET algorithm in the segmentation of BIRD image in the 5-level thresholding is distinct to the one using HMBO-based MECT algorithm. The results reveal that the capability of threshold selection by using the ABC-based and HBMO-based MECT algorithm is not significantly difference. Table 4 shows the computation time and the corresponding PSNR values of the four different multi-level thresholding methods. The fitness values of corresponding thresholds are shown in Table 3. Several aspects are found in the two tables. The computation time of the ABC-based MCET algorithm is faster than the ones of other three algorithms as the number of thresholds exceed 3. The corresponding PSNRs of the five images using the ABC-based MCET algorithm are superior to the ones of the PSO-based MCET and QPSO-based MCET algorithms.

Table 4. The Computation Times and the Corresponding PSNR of the Four Different Multilevel Thresholding Methods

Image	k	ABC-based MCET algorithm		HBMO-based MCET algorithm		PSO-based MCET algorithm		Quantum PSO-based MCET algorithm	
		Computation time (sec)	PSNR	Computation time (sec)	PSNR	Computation time (sec)	PSNR	Computation time (sec)	PSNR
Lena	2	1.34	16.05	1.49	16.05	1.39	16.05	1.98	16.05
	3	5.43	18.32	6.58	18.32	5.68	18.32	6.58	18.27
	4	26.39	20.39	34.35	20.39	28.69	20.27	35.98	20.29
	5	186.46	21.78	256.25	21.78	213.65	21.70	278.65	20.73
Pepper	2	1.48	15.31	1.54	15.31	1.52	15.31	1.85	15.35
	3	5.67	17.90	6.67	17.90	5.98	17.88	6.47	17.66
	4	25.78	20.22	33.95	20.22	29.71	20.10	38.59	20.10
	5	189.85	21.99	268.65	21.99	221.89	21.95	298.65	21.94
Bird	2	1.45	16.09	1.32	16.09	1.53	16.09	1.95	16.22
	3	5.85	18.57	7.01	18.57	6.14	18.41	6.24	18.20
	4	24.65	21.37	35.43	21.37	28.96	21.07	34.26	19.96
	5	179.54	23.21	271.87	22.92	231.85	22.82	284.36	22.58
Gamera	2	1.48	16.09	1.67	16.09	1.46	16.09	1.75	16.02
	3	5.12	18.94	6.92	18.94	5.68	18.76	6.89	18.86
	4	26.28	21.37	35.01	21.37	29.25	21.37	38.95	21.62
	5	184.26	22.89	261.39	22.89	241.58	22.87	286.58	22.46
Goldhill	2	1.39	14.55	1.43	14.55	1.54	14.55	1.85	14.55
	3	5.45	17.19	6.54	17.19	6.16	17.03	6.47	17.14
	4	27.57	19.00	32.86	19.00	29.01	19.00	34.95	18.28
	5	187.63	20.75	258.54	20.75	259.65	20.73	292.84	20.61

The results are probable to demonstrate that the PSO and QPSO algorithm can not effectively search for the good thresholds for image segmentation. However, there are not significantly differences between ABC-based MCET and HBMO-based MCET algorithm. It reveals that the two different evolutionary algorithms of simulating the behaviors of bee colony are evenly matched in the selection of thresholds. The only difference between ABC-based and HBMO-based MCET algorithms from the Table 4 is the threshold selection of BIRD image as the 5-level thresholding, however, the PSNR values of 5-level BIRD segmented image of ABC-based MECT algorithm is superior to the ones of the HBMO-based MCET algorithm. The experimental results demonstrate that the ABC-based MCET algorithm can effectively find the adequate thresholds than the other three popular evolutionary algorithms.

4. Conclusion

This paper presents a new multilevel image thresholding scheme based on the artificial bee colony algorithm. From the experimental results we find the two important contributions. One is that the proposed ABC-based MCET algorithm can more efficient to search the near optimal solutions compared to the exhaustive search method. The other is that the quality of segmentation images using ABC-based MECT method is superior to HBMO-based and PSO-based algorithms.

Acknowledgement

The authors would like to thank the National Science Council, ROC, under Grant No. NSC 101-2221-E-251-008 for support of this work.

References

[1] M Usman Skram, Ibaa Jamal, Anam Tariq, *Blood Vessel Enhancement and Segmentation for Screening of Diabetic Retinopathy. TELKOMNIKA Indonsian Journal of Electrical Engineering.* 2012; 10(2): 327-334.
[2] Jun Lai, Mei Xie. Automatic Segmentation for Pulmonary Vessels in Plain Thoracic CT scans, *TELKOMNIKA Indonsian Journal of Electrical Engineering*. 2012; 10(4), 743-751.
[3] N Otsu. *A Threshold Selection Method from Gray-level Histograms*. *IEEE Transactions on Systems, Man, Cybernetics.* 1979; 9: 62-66.
[4] PK Shoo, Soltani, SAKC Wong, YC Chen. *A Survey of Thresholding Techniques*. Computer Vision Graphics and Image Processing. 1998; 41: 233-236.
[5] Maitra M, Chatterja A. *A Hybrid Cooperative-comprehensive Learning based Algorithm for Image Segmentation using Multilevel Thresholding*. Expert systems with Application. 2008; 34: 1341-1350
[6] CH Li, CK Lee. Minimum Cross Entropy Thresholding. *Pattern Recognition.* 1993; 26: 617-625.
[7] CH Li, PKS Tam. An Iterative Algorithm for Minimum Cross Entropy Thresholding. *Pattern Recognition Letter.* 1998; 19: 771-776.
[8] PY Yin. *Multilevel Minimum Cross Entropy Threshold Selection based on Particle Swarm Optimization*. Applied Mathematics and Computation. 2007; 184: 503-513.
[9] MH Horng. *A Multilevel Image Thresholding using the Honey Bee Mating Optimization*. Applied Mathematics and Computation. 2010; 215: 3302-3310.
[10] D Karaboga, B Basturk. *On the Performance of Artificial Bee Colony Algorithm.* Applied Soft Computing. 2008; 8: 687-697.
[11] D Karaboga. *Artificial Bee Colony Algorithm homepage: http://mf.erciyers.edu.tr/abc/,* (Accessed on; July, 2012).