

APPLYING CHAOTIC IMPERIALIST COMPETITIVE ALGORITHM FOR MULTI-LEVEL IMAGE THRESHOLDING BASED ON KAPUR'S ENTROPY

Maryam Rouhani Nejad¹, Mehdi Fartash^{1,*}

¹ Department of Electrical and Computer Engineering, Arak Branch, Islamic Azad University, Arak, Iran, e-mail: ma.rohaninejad@yahoo.com

* Corresponding author e-mail: m-fartash@iau-arak.ac.ir

Received: 2015.12.15

Accepted: 2016.02.01

Published: 2016.03.01

ABSTRACT

Segmentation is one of the most important operations in image processing and computer vision. Normally, all image processing and computer vision applications are related to segmentation as a pre-processing phase. Image thresholding is one of the most useful methods for image segmentation. Various methods have been represented for image thresholding. One method is Kapur thresholding, which is based on maximizing entropy criterion. In this study, a new meta-heuristic algorithm based on imperialist competition algorithm was proposed for multi-level thresholding based on Kapur's entropy. Also, imperialist competitive algorithm is combined with chaotic functions to enhance search potency in problem space. The results of the proposed method have been compared with particle optimization algorithm and genetic algorithm. The findings revealed that the proposed method was superior to other methods.

Keywords: segmentation, thresholding, histogram, imperialist competitive algorithm, chaotic.

INTRODUCTION

Image segmentation as a pre-processing phase plays an important role in image processing and computer vision applications. In fact, segmentation quality has great effects on other processing steps. Without proper segmentation, an efficient algorithm cannot operate efficiently. Hence, segmentation is constantly considered as an important phase for computer vision. There are various methods for segmentation. One method is applying histogram. Histogram of an image represents the way of intensity distribution of images. One segmentation method based on histogram is thresholding. Thresholding is an important technique for performing image segmentation and is computationally efficient. The main purpose is to determine a threshold for bi-level thresholding or several thresholds for multi-level thresholding. Bi-level thresholding determines only one threshold which separates pixels into 2 groups while multi-level thresholding determines several thresholds which separate pixels into multiple groups.

In thresholding, maximizing entropy criterion is a common criterion for selecting optimal thresholds [1, 2, 3].

Thresholding based on entropy was first introduced by Pun. Pun proposed a method to select image optimal thresholds through representing new criterion of maximizing entropy between gray-levels of objects and gray-levels of image background. Another entropy-based method is Kapur's method. Kapur modified Pun's thresholding method by adding 2 gray-level probability distribution: one for background and another one for objects [1].

Almost all primary thresholding methods are able to select optimal thresholding. Using these methods for multi-level thresholding, most of them are improper in terms of performance time and increased thresholds results in exponential increase of time. One recent thresholding method is using meta-heuristic algorithms. These algorithms seek to minimize or maximize a function. Maitra and Chatterjee [4] first introduced a particle optimization algorithm for image multi-level threshold-

ing. In their method, a multidimensional particle is changed into several one dimensional particles. The method prevents early convergence of particle optimization algorithm. This combined algorithm has increased efficiency of particle optimization algorithm in thresholding. The cost function used by them was based on Kapur's function. Musrrat Ali et al. [4] applied synergetic differential evolution, which is an improved version of differential evolution for thresholding, and applied fitness function based on Kapur's function. Also, algorithms such as honey bee mating and tabu search [5] have been used for the purpose.

IMPERIALIST COMPETITIVE ALGORITHM

Imperialist competitive algorithm is an optimization algorithm introduced by Atashpaz and Lucas in 2007 [6]. This algorithm is inspired from a socio-political process started by some initial populations. In this algorithm, every element of population is called a country. The countries are divided into 2 classes: colonies and imperialists. Each imperialist colonizes and controls some countries. The core of this algorithm consists of a policy of attraction and colonial competition. According to policy of attraction historically applied by imperialists, such as France and England in their colonies, imperialists made efforts to destroy culture, tradition, and language of their colonies (e.g. through building schools to teach their own languages). In this algorithm, the policy is applied through moving colonies of an empire based on a specific relation. When a colony reaches a better position than the current imperialist, it replaces the current imperialist state of the empire. Also, the power of an empire consists of imperialist's power and a percentage of average of its colonies' power. Imperialist competition is another important part of this algorithm. During this process, weak empires lose their power and vanish. Finally, there is one empire which controls the world. In this situation, imperialist competitive algorithm reaches optimal point of objective function and stops.

PROPOSED APPROACH

Thresholding through Kapur's method

Let there be L gray-level in the image I and these gray-levels range {1, 2, ..., L}. When the

pixels of an image are in L gray-level ([0, ..., L]) and the number of pixels in i level is n_i , normal histogram is as follows:

$$p_i = n_i/N, p_i \geq 0, \sum_{i=1}^L p_i = 1 \quad (1)$$

Kapur's method maximizes criterion of entropy of histogram so that separated areas have maximum central distribution [7]. The criterion is represented as follows for bi-level thresholding:

$$\text{Maximize } J(t) = H_0 + H_1 \quad (2)$$

where:

$$H_0 = - \sum_{i=0}^{t-1} \frac{p_i}{w_0} \ln \frac{p_i}{w_0}, w_0 = \sum_{i=0}^{t-1} p_i$$

$$H_1 = - \sum_{i=t}^{L-1} \frac{p_i}{w_1} \ln \frac{p_i}{w_1}, w_1 = \sum_{i=t}^{L-1} p_i \quad (3)$$

Maximum threshold is a threshold that maximizes the previous equation. In fact, H_0 and H_1 are entropies of each part. The objective of Kapur's method is to maximize this sum (maximizing the entropy).

Kapur's criterion of optimization for multi-level thresholding has developed. This criterion is defined as follows. Multi-level thresholding is supposed as an m dimensional optimization problem. We want to determine m for thresholding $[t_1, t_2, \dots, t_m]$. The objective is to determine the maximum of the following function:

$$J([t_1, t_2, \dots, t_m]) = H_0 + H_1 + H_2 + \dots + H_m \quad (4)$$

$$H_0 = - \sum_{i=0}^{t_1-1} \frac{p_i}{w_0} \ln \frac{p_i}{w_0}, w_0 = \sum_{i=0}^{t_1-1} p_i$$

$$H_1 = - \sum_{i=t_1}^{t_2-1} \frac{p_i}{w_1} \ln \frac{p_i}{w_1}, w_1 = \sum_{i=t_1}^{t_2-1} p_i$$

$$H_2 = - \sum_{i=t_2}^{t_3-1} \frac{p_i}{w_2} \ln \frac{p_i}{w_2}, w_2 = \sum_{i=t_2}^{t_3-1} p_i$$

$$H_m = - \sum_{i=t_m}^{L-1} \frac{p_i}{w_m} \ln \frac{p_i}{w_m}, w_m = \sum_{i=t_m}^{L-1} p_i \quad (5)$$

Chaotic imperialist competitive algorithm

In the proposed method, a sequence generated by chaotic maps is used in the parameters of imperialist competitive algorithm instead of a random sequence of numbers. In this section, initial countries are generated as a repetition of chaotic maps. In this algorithm, N and i denote the num-

ber of dimensions of the problem and a country in the population, respectively. Varmin and varmax are lower and upper bounds, respectively. Also, $x_{i,j}$ denotes j^{th} dimension of i^{th} country in the population. Pseudo-code of initial population generation algorithm using chaotic maps is as follows:

```

MCI= the maximum number of chaotic iteration
i =0
while (i ≤ population size )
    Randomly initialize the first chaotic variables
    j =0
    while (j < N)
        generate chaotic variable  $cm_{i,j}$  according to the selected map
         $x_{i,j} = \text{var min} + cm_{i,j} \times (\text{var max} - \text{var min})$ 
         $j = j + 1$ 
    end while
     $i = i + 1$ 
end while
    
```

Overall structure of the proposed algorithm for image segmentation

The proposed algorithm based on chaotic imperialistic competition for multi-level thresholding is described in this section.

- Step 1: Give initial values to ICA including Num Of Countries, Num Of Imp, and Num Of Col. The relationship between these parameters is as follows:

$$Num\ of\ Countries = Num\ of\ Imp + Num\ of\ Col \quad (6)$$

- Step 2: Produce initial countries by continuous repetition of chaotic map.
- Step3: Compute cost function for each country based on Kapur’s method.
- Step 4: Select Num Of Imp of the strongest countries as imperialist, and the rest are Num Of Col. To establish an initial empire, the normalized cost of the empire can be defined as follows:

$$C_k = \max_i \{c_i\} - c_k \quad (7)$$

Where c_k is the cost of the k^{th} empire and C_k is its normalized cost. Finally, the normalized power of each empire is defined as follows:

$$p_k = \left| \frac{C_k}{\sum_{i=1}^{Num\ of\ Imp} C_i} \right| \quad (8)$$

The normalized power of an empire represents the number of initial colonies that should be conquered by an imperialist and is defined as follows. $N.C_n$ denotes the initial number of colonies of the n^{th} empire.

$$N.C_n = round(p_n \cdot Num\ Of\ Col) \quad (9)$$

- Step 5: Colonies turn into imperialists chaotically.
- Step 6: Apply revolution operator on each colony. In each repetition, produce a random number between 0 and 1 for each imperialist. Next, the result is compared with probability of revolution rate. When the random number is less than the revolution rate, the revolution is produced.
- Step 7: Calculate the cost of all colonies and imperialists of the empire. In previous steps, if there is a colony in the empire which has lower cost than the imperialist that colony takes the control of the empire.
- Step 8: Obtain the total power of all empires.
- Step 9: Perform imperialistic competition.
- Step 10: Remove the weakest empire. When an empire is removed, all its colonies are removed too.
- Step 11: Add one unit to Dec value.
- Step 12: When Dec value is more than \max_{Dec} ($Dec > \max_{Dec}$), you can stop; otherwise return to step 6.

EXPERIMENTAL RESULTS AND COMPARATIVE PERFORMANCES

Five pictures: “Lena”, “Pepper”, “Bird”, “Camera”, and “Goldhill”, were applied to evaluate the performance of the proposed algorithm. These test images and corresponding histograms are shown in Figure 1.

We implemented the proposed algorithm in language of Matlab (2014) under a computer with 2GHz CPU, 1 G RAM with Windows 8.1 system. The size of “Camera” and “Pepper” image is 256×256 and the size of other images is 512×512. The most appropriate parameters of the algorithm to conduct experiments are shown in Table 1.

Computation time and thresholds

In this section, computation time and thresholds are obtained using the particle optimization algorithm and genetic algorithm. In obtaining thresholds using meta-heuristic algorithm, time

Table 1. Parameters used for thresholding images

Parameter	Value of parameter
Population	50
Number of initial empires	5
Revolution rate	0.1
Beta value	2

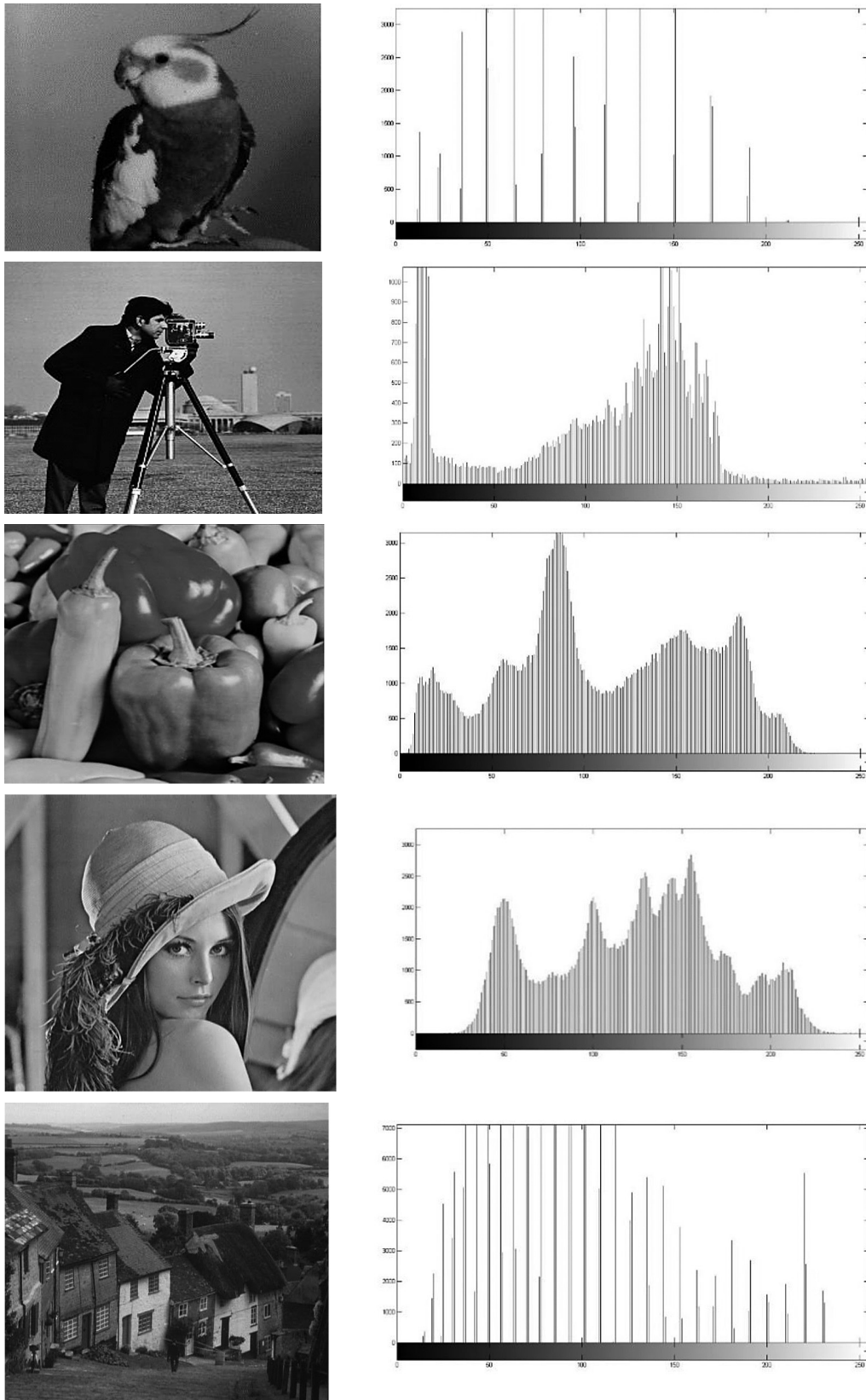


Fig. 1. Test images and corresponding histograms

is one of the major aspects. By increasing the number of thresholds, computation time increases. Table 2 demonstrates optimal thresholds and computation times based on entropy criterion.

As shown in Table 2, selected thresholds of the proposed algorithm are close to the optimal thresholds of the particle optimization algorithm and genetic algorithm. Also, the computation

Table 2. Optimal thresholds and computation times based on entropy criterion

Image (Size)	K	Proposed approach		PSO		GA	
		Thresholds	Computation time(s)	Thresholds	Computation time(s)	Thresholds	Computation time(s)
Lena (512×512)	2	94,152	3.32	94,152	3.15	91,149	3.45
	3	80,128,171	3.34	79,127,170	3.25	80,124,173	3.65
	4	75,114,145,180	4.26	78,112,134,175	3.95	80,126,159,185	4.16
	5	73,110,136,157,187	4.58	79,110,140,167,188	4.23	80,116,146,179,213	4.52
Pepper (256×256)	2	71,139	3.12	76,144	3.02	84,144	3.65
	3	64,122,170	3.65	72,124,171	3.96	65,116,175	3.58
	4	81,89,128,173	4.17	57,92,130,172	4.16	62,108,142,177	4.19
	5	51,86,118,150,179	4.89	56,84,115,150,179	4.56	52,90,128,166,191	4.44
Bird (512×512)	2	73,141	2.97	72,143	2.96	71,142	2.65
	3	62,118,159	3.15	64,117,160	3.25	68,119,159	2.98
	4	60,96,129,165	3.68	61,97,130,172	3.95	60,99,140,180	3.65
	5	56,92,130,156,176	4.15	57,93,135,160,170	4.20	59,95,130,150,176	4.42
Camera (256×256)	2	71,144	3.15	71,143	2.65	72,145	3.65
	3	57,116,154	3.69	71,134,166	2.96	71,143,196	3.96
	4	41,94,139,169	4.16	65,121,147,172	3.15	59,119,155,203	4.20
	5	36,82,121,148,172	4.65	45,78,121,146,172	3.96	51,106,141,167,194	4.69
Goldhill (512×512)	2	93,160	3.65	89,156	3.15	89,156	3.19
	3	82,126,179	3.95	77,130,176	3.53	77,128	3.65
	4	69,100,137,185	4.15	64,106,147,189	3.88	64,105,146,188	3.98
	5	62,90,117,148,190	4.95	58,93,130,164,198	4.23	56,92,129,163,196	4.65

times of the proposed algorithm are often a little more than that of PSO and genetic algorithm due to using chaotic functions in producing random numbers.

Analysis of signal to noise ratio

The popular performance indicator, peak signal to noise ratio (PSNR) is used to compare segmentation results by threshold techniques of multilevel image [8–10]. PSNR criterion has been investigated as an evaluation criterion of quality of thresholding. Increased value of PSNR results in higher quality of thresholding. We define the criterion, measured in decibel (dB) as:

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

where: RMSE (root mean-squared error) is defined as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^M \sum_{j=1}^N (I(i, j) - \hat{I}(i, j))^2}{MN}} \quad (11)$$

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

Table 3 shows that by increasing the number of thresholding, signal to noise ratio increases. The results show that the proposed method often presents higher PSNR. Also, most of the time, the genetic algorithm has lower accuracy compared with other methods.

Cost values

The purpose of all meta-heuristic algorithms is to maximize or minimize a function which is called fitness function or cost function. In this study, an entropy based cost function is used, and the purpose is to maximize this function. Table 4 shows the value of cost using the entropy based method.

As shown in Table 4, the cost function increases by increased number of thresholds. Also, the values of the cost function of the proposed method were higher compared with other methods except for “Goldhill” image and the threshold of 4.

Stability analysis

In general, meta-heuristic methods are random algorithms; each implementation may reveal different results. The following formula is used to analyze stability of meta-heuristic algorithms:

Table 3. Signal to noise ratio for thresholding 5 images using entropy criterion

Image (Size)	K	Proposed approach	PSO	GA
Lena (512×512)	2	12.39	12.34	12.33
	3	15.65	15.33	14.99
	4	17.96	17.83	17.08
	5	20.46	20.44	19.54
Pepper (256×256)	2	12.58	12.51	12.51
	3	15.96	15.09	14.71
	4	18.65	18.09	17.69
	5	21.09	20.73	20.06
Bird (512×512)	2	12.32	12.20	12.13
	3	15.25	15.30	14.90
	4	18.11	17.91	16.69
	5	20.65	20.01	19.65
Camera (256×256)	2	12.65	12.25	11.94
	3	15.29	15.21	14.82
	4	18.69	18.00	17.16
	5	21.62	20.96	19.79
Goldhill (512×512)	2	12.44	12.34	12.11
	3	14.96	14.90	14.75
	4	17.95	17.90	16.96
	5	20.93	19.90	19.96

Table 4. Value of cost using the entropy based method

Image (Size)	K	Proposed approach	PSO	GA
Lena (512×512)	2	12.3545	12.3544	12.3545
	3	15.7963	15.7625	15.7862
	4	18.5796	18.5523	18.5421
	5	21.2560	21.2469	21.2693
Pepper (256×256)	2	12.6354	12.6296	12.6235
	3	15.8209	15.8109	15.8061
	4	18.3546	18.3261	18.3465
	5	21.0192	21.0012	21.0125
Bird (512×512)	2	5.3085	5.2904	5.2864
	3	6.2241	6.2142	6.2032
	4	6.8666	6.8514	6.5412
	5	7.6554	7.6232	7.6214
Camera (256×256)	2	12.2465	12.2365	12.1542
	3	15.2325	15.2261	15.2156
	4	18.2931	18.2874	18.2765
	5	20.9564	20.9123	20.9235
Goldhill (512×512)	2	6.3962	6.3541	6.3921
	3	7.4105	7.4021	7.3965
	4	8.1445	8.1536	8.1521
	5	8.7974	8.3564	8.6935

$$std = \sqrt{\frac{\sum_{i=1}^k (\delta_i - \mu)^2}{k}} \tag{12}$$

This formula is used to obtain standard deviation. Standard deviation shows the distribution of data. K denotes the number of implementations of algorithm (10 implementations), and δ_i indicates the optimal value for objective function in each implementation. Also, the average of δ values is denoted by μ . Lower standard deviation results in higher stability of algorithm. Table 5 shows the standard deviations using the entropy based method.

As shown in table 5, all standard deviations are low, which shows higher stability of the proposed method compared with other methods. Higher stability means that the results of different implementations are closer while lower stability means that the results are not close.

CONCLUSIONS

In this study, we proposed a maximizing entropy criterion based method to select multi-level thresholds using chaotic imperialist competitive algorithm. Imperialist competitive algorithm is

Table 5. Standard deviation for entropy criterion based thresholding

Image (Size)	K	Proposed approach	PSO	GA
Lena (512×512)	2	0.0064	0.1423	0.2070
	3	0.048	0.4155	0.5555
	4	0.096	2.3601	3.0654
	5	0.058	4.5341	5.9532
Pepper (256×256)	2	0.065	0.0956	0.1455
	3	0.045	0.1629	0.2891
	4	0.0025	2.1102	3.9732
	5	0.0012	3.2057	4.6592
Bird (512×512)	2	0.056	2.7001	3.6523
	3	0.089	0.0090	2.0125
	4	0.008	3.2551	2.0365
	5	0.012	6.2654	4.0625
Camera (256×256)	2	0.0096	0.0908	0.3815
	3	0.0054	6.3502	9.6256
	4	0.0018	2.4498	4.2618
	5	0.009	8.9650	11.0258
Goldhill (512×512)	2	0.022	0.9123	1.2565
	3	0.016	2.2325	3.2592
	4	0.092	6.6935	5.6249
	5	0.012	9.0958	8.3562

a new meta-heuristic algorithm impressed from imperialistic competition between countries. In this algorithm, each solution is called a country. Imperialists attempt to attract colonies toward themselves, and this process continues until there is only one imperialist. This algorithm has been proved efficient for different optimization problems. The purpose of this study was to segment images using chaotic imperialist competitive algorithm. Thereby, using chaotic functions, the efficiency of the imperialist competitive algorithm was improved. Using chaotic functions for population production the variety of the population increased. Next, the new algorithm was used for image segmentation through thresholding. Kapur-based cost function which applies the entropy criterion was used as fitness function. The results were applied on some images. Several criteria including time, signal to noise ratio, fitness function and standard deviation were applied to evaluate the efficacy. The results revealed that the algorithm is superior to other algorithms in efficacy. Also, applying chaos results in a variety of initial results and increased efficacy of the algorithm.

REFERENCES

1. 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* 3, 1985, 273–285.
2. Portes de Albuquerque M., Esquef I.A. Gesualdi Mello A.R. Image thresholding using Tsallis entropy. *Pattern Recognition Letters* 2004, 1059–1065.
3. Pal N.R., Pal S.K. Entropic thresholding. *Signal Processing* 1989, 97–108.
4. Maitra M, Chatterjee A. A hybrid cooperative–comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding. *Expert Systems with Applications* 2008, 2, 1341–1350.
5. Ali M., Ahn C.W., Pant M. Multi-level image thresholding by synergetic differential evolution. *Applied Soft Computing* 2014, 1–11.
6. Jiang Y., Tsai P., Hao Z., Cao L. Automatic multilevel thresholding for image segmentation using stratified sampling and Tabu Search. *Soft Computing* 2014, 1–13.
7. Atashpaz-Gargari E., Lucas C. Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: *Proceedings Evolutionary Computation, CEC 2007 IEEE Congress on*, 25-28 sept 2007, Singapore 2007, 4661–4667.
8. Horng M.H. A multilevel image thresholding using the honey bee mating optimization. *Applied Mathematics and Computation* 9, 2010, 3302–3310.
9. Dirami A., Hammouche K., Diaf M., Siarry P. Fast multilevel thresholding for image segmentation through a multiphase level set method. *Signal Processing* 1, 2013, 139–153.
10. Raja N., Rajinikanth V., Latha K. Otsu based optimal multilevel image thresholding using firefly algorithm. *Modelling and Simulation in Engineering* 2014, 37.