Computer Science Journal of Moldova, vol.22, no.3(66), 2014

Multilevel image thresholding by nature-inspired algorithms: A short review*

Milan Tuba

Abstract

Nondeterministic metaheuristic optimization and digital image processing are two very different research fields, both extremely active and applicable. They touch in a very limited area, but that narrow interaction opens new very promising applications for digital image processing and new and different deployment of metaheuristic optimization. Multilevel image thresholding is very important for image segmentation, which in turn is crucial for higher level image analysis. The problem includes exponential combinatorial optimization with complex objective functions which are solvable only by nondeterministic methods. This short review presents successful applications of the nature-inspired metaheuristics to multilevel image thresholding.

Keywords: Digital image processing, multilevel thresholding, swarm intelligence, metaheuristic optimization.

1 Introduction

Digital image processing is one of the most applicable research areas used in many practical applications in different fields like medicine, security, quality control, astronomy etc. It consists of very different techniques belonging to low level signal processing, medium level morphological processing and segmentation for feature detection and high level artificial intelligence algorithms for object recognition, information extraction, representation and understanding. At different stages

*This work was supported by Ministry of Science of Republic of Serbia project Ref. Nr. III-44006

of image processing some hard optimization problems occur. For example, multilevel image thresholding is a step in segmentation, but even though this problem at first sight seems to be simple, to determine optimal n numbers in the range [0-255] is NP-hard combinatorial problem. JPEG quantization matrix optimization is also an exponential combinatorial problem since the number of coefficients, after DCT, is 64 and each can be an integer between 0 and 255.

Such problems cannot be solved in reasonable time by standard mathematical deterministic methods. Nature inspired metaheuristic algorithms have recently been successfully used for this type of hard optimization problems. They try to guide random Monte-Carlo search by simulating some successful systems from the nature. Swarm intelligence is an important branch of nature inspired algorithms where collective intelligence of different species like ants, bees, cuckoos, fireflies, bats, fish, birds, krill etc. is simulated.

This short review presents the problem, history and state-of-the-art for multilevel image thresholding by nature-inspired nondeterministic metaheuristics, particularly swarm intelligence algorithms.

2 Multilevel thresholding

The goal of image segmentation is to divide an image into homogeneous and disjoint sets of pixels sharing similar properties such as intensity, color or contours. Image segmentation usually represents the first step in image understanding. The results obtained by segmentation are used for further higher-level methods such as feature extraction, semantic interpretation, image recognition, and classification of objects.

Image thresholding is one of the most widely used segmentation techniques that performs image segmentation based on the information contained in the image histogram. Selection of the multiple thresholds is crucial in image segmentation since proper segmentation depends on adequately computed thresholds. Figure 1 shows two standard benchmark images with corresponding histograms. We can notice that the first image has four prominent peaks in the histogram, while the second one has three. Reasonable segmentation of the first image thus would require three thresholds between four peaks. More careful examination can show that the first image has six peaks, two of them not so emphasized. It shows that the problem of determining how many thresholding levels are required, as well as where they should be placed, is not trivial and does not have unique best answer.

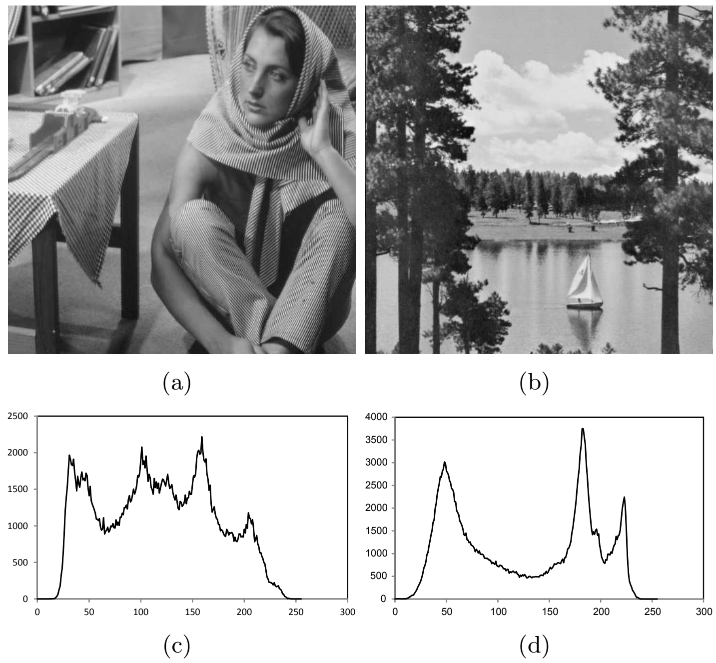

Figure 1. Two often used benchmark images with corresponding histograms: (a) and (c) Barbara, (b) and (d) Lake.

3 Criteria for selecting thresholds

The criteria for selecting thresholds can be different. In the previous example we tried to use our insight for that purpose, however for automatic processing of large number of images that is impossible. Most often some objective criteria are used. Two most successful criteria or objective functions are Kapur's entropy and between-class variance Otsu's criterion.

3.1 Kapur's thresholding method

The Kapur's method [1] based on the entropy is used to perform multilevel thresholding. For this method the threshold criteria is formulated as follows. Let an image I contain n pixels with gray levels belonging to the set $\{0, 1, \ldots, L-1\}$. Let $h(i)$ present the number of pixels at gray level i, and $p_i = h(i)/n$ be the probability of occurrences of gray level i in the image I. The subdivision of an image into $k+1$ classes can be considered as a k-dimensional optimization problem for the calculation of k optimal thresholds $(t_0, t_1, \ldots, t_{k-1})$. The optimal thresholds are obtained by maximizing the objective function

$$f(t_0, t_1, \ldots, t_{k-1}) = \sum_{i=0}^{k} H_i, \tag{1}$$

where the entropies H_i are defined by

$$\begin{aligned}
H_0 &= -\sum_{i=0}^{t_0-1} \frac{p_i}{w_0} ln \frac{p_i}{w_0}, \quad & w_0 &= \sum_{i=0}^{t_0-1} p_i \\
H_1 &= -\sum_{i=t_0}^{t_1-1} \frac{p_i}{w_1} ln \frac{p_i}{w_1}, \quad & w_1 &= \sum_{i=t_0}^{t_1-1} p_i \\
& & &\vdots \\
H_k &= -\sum_{i=t_{k-1}}^{L-1} \frac{p_i}{w_k} ln \frac{p_i}{w_k}, \quad & w_k &= \sum_{i=t_{k-1}}^{L-1} p_i.
\end{aligned} \tag{2}$$

3.2 Otsu's thresholding method

Otsu's method [2] is based on the maximization of the between-class variance and represents another very popular method proposed for image thresholding. The algorithm for this method can be described as follows. Assume that an image I can be represented by L gray levels. The probabilities of pixels at level i are denoted by p_i, so $p_i \geq 0$ and $p_0 + p_1 + \cdots + p_{L-1} = 1$. Cumulative probabilities for classes $A_i,\ i = 0, 1,\ ..., k$ can be defined as

$$w_0 = \sum_{i=0}^{t_0-1} p_i,\ w_1 = \sum_{i=t_0}^{t_1-1} p_i,\ \ ...,\ w_k = \sum_{i=t_{k-1}}^{L-1} p_i, \tag{3}$$

where t_j are the thresholds separating these classes. For $k+1$ classes A_i, $(i = 0, 1, \ldots, k)$ the goal is to maximize the objective function

$$f(t_0, t_1, \ldots, t_{k-1}) = \sum_{i=0}^{k} \sigma_i, \tag{4}$$

where the sigma functions are defined by:

$$\begin{aligned}
\sigma_0 &= w_0 \left(\sum_{i=0}^{t_0-1} \frac{ip_i}{w_0} - \sum_{i=0}^{L-1} ip_i \right)^2 \\
\sigma_1 &= w_1 \left(\sum_{i=t_0}^{t_1-1} \frac{ip_i}{w_1} - \sum_{i=0}^{L-1} ip_i \right)^2 \\
&\vdots \\
\sigma_k &= w_k \left(\sum_{i=t_{k-1}}^{L-1} \frac{ip_i}{w_k} - \sum_{i=0}^{L-1} ip_i \right)^2 .
\end{aligned} \tag{5}$$

Kapur's and Otsu's objective functions may not be appropriate for all applications, but they are the starting point and for any additional particular requirements further refinement is possible. Four more standard bencmark images often used for testing thresholding algorithms are in Figure 2.

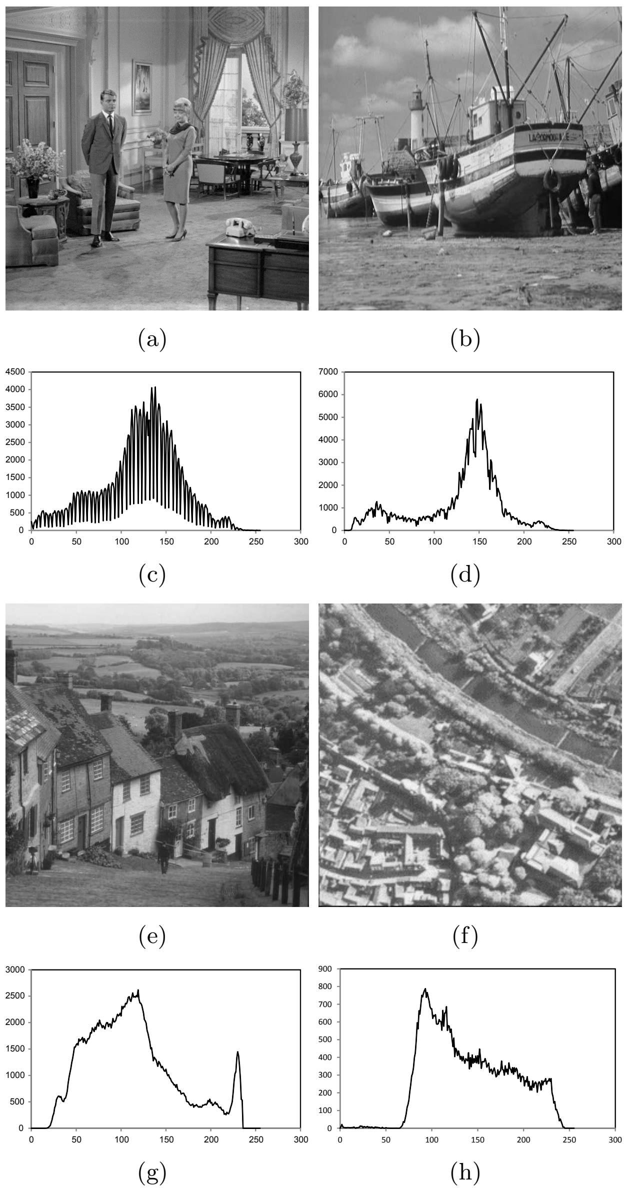

Figure 2. Four other often used benchmark images with corresponding histograms: (a) and (c) Living room, (b) and (d) Boats, (e) and (g) GoldHill, (f) and (h) Aerial.

4 Exact results

Best possible thresholds (global optimum) can always be found by exhaustive search. All possibilities can be examined and the best one selected. However, the number of possible values for one threshold is 255 (the number of gray levels) and for k thresholds is 255^k. This exponential growth prevents exhaustive search to be used for larger number of desired thresholds. Table 1 shows for the six mentioned bencmark

Table 1. Thresholds, Objective function values and processing time for the exhaustive search for Kapur's method

Images	K	Threshold values	Objective function	Time (ms)
Barbara	2	96, 168	12.668336540	25
	3	76, 127, 178	15.747087798	341
	4	60, 99, 141, 185	18.556786861	11103
	5	58, 95, 133, 172, 210	21.245645310	666869
Living room	2	94,175	12.405985592	31
	3	47, 103, 175	15.552622213	339
	4	47, 98, 149, 197	18.471055578	12612
	5	42, 85, 124, 162, 197	21.150302316	478114
Boats	2	107, 176	12.574798244	25
	3	64, 119, 176	15.820902860	342
	4	48, 88, 128, 181	18.655733570	11461
	5	48, 88, 128, 174, 202	21.401608305	469862
Goldhill	2	90, 157	12.546393623	24
	3	78, 131, 177	15.607747002	329
	4	65, 105, 147, 189	18.414213765	11958
	5	59, 95, 131, 165, 199	21.099138996	399458
Lake	2	91, 163	12.520359742	24
	3	72, 119, 169	15.566286745	336
	4	70, 111, 155, 194	18.365636309	12658
	5	64, 99, 133, 167, 199	21.024982760	410753
Aerial	2	68, 159	12.538208248	29
	3	68, 130, 186	15.751881495	347
	4	68, 117, 159, 200	18.615899102	11390
	5	68, 108, 141, 174, 207	21.210455499	599570

images optimal threshold values, optimal objective function value for Kapur's method and running time in milliseconds. The same data for Otsu's objective function are presented in Table 2.

Table 2. Thresholds, objective function values and processing time for the exhaustive search for Otsu's method

Images	K	Threshold values	Objective function	Time (ms)
Barbara	2	82, 147	2608.610778507	39
	3	75, 127, 176	2785.163280467	89
	4	66, 106, 142, 182	2856.262131671	3014
	5	57, 88, 118, 148, 184	2890.976609405	100079
Living room	2	87, 145	1627.909172752	39
	3	76, 123, 163	1760.103018395	88
	4	56, 97, 132, 168	1828.864376614	2945
	5	49, 88, 120, 146, 178	1871.990616316	130397
Boats	2	93, 155	1863.346730649	38
	3	73, 126, 167	1994.536306242	89
	4	65, 114, 147, 179	2059.866280428	2931
	5	51, 90, 126, 152, 183	2092.775965336	75879
Goldhill	2	94, 161	2069.510202452	38
	3	83, 126, 179	2220.372641501	88
	4	69, 102, 138, 186	2295.380469158	2775
	5	63, 91, 117, 147, 191	2331.156597921	74674
Lake	2	85, 154	3974.738214185	39
	3	78,140,194	4112.631097687	89
	4	67, 110, 158, 198	4180.886161109	2613
	5	57, 88, 127, 166, 200	4216.943583790	73019
Aerial	2	125, 178	1808.171050536	46
	3	109, 147, 190	1905.410606582	103
	4	104, 134, 167, 202	1957.017965982	2670
	5	99, 123, 148, 175, 205	1980.656737348	99880

Since processing time for 5 thresholds is for Otsu's method almost 2 minutes (on Intel i7-4770K CPU) and it increases 255 times for each additional threshold, expected processing time for 6 thresholds would be around 8 hours, for 7 thresholds around 2 months, for 8 thresholds around 40 years, for 9 thresholds around 10,000 years etc.

5 Algorithms for thresholding

Since computational time for finding multiple thresholds grows exponentially with the number of desired thresholds, the exhaustive search is not an option, as it was shown in the previous tables. Before swarm intelligence nondeterministic metaheuristics became popular, other algorithms were attempted in effort to tackle that hard optimization problem. Papamarkos and Gatos in 1994 proposed three-stage algorithm that included hill-clustering technique applied to the image histogram in order to approximately determine the peak locations of the histogram, approximation by rational functions of the histogram segments between the peaks and finally the application of the one-dimensional Golden search minimization algorithm [3]. Different approach was proposed by Yin and Chen in 1997. They proposed an iterative scheme that starts with a bi-level thresholding and uses the initial results to obtain higher-order thresholds [4].

5.1 Nature-inspired algorithms

These early attempts were soon overtaken by number of nature inspired algorithms well suited for such hard problems. Swarm intelligence was still in its infancy, so other older nature inspired algorithm dominated at first. Genetic algorithms are among the oldest nature inspired algorithms. Yang et al. in 2003 developed a relative entropy multilevel thresholding method based on genetic algorithm where the relative entropy was treated as the fitness function for the genetic algorithm [5]. The use of genetic algorithms for the multilevel thresholding continues, so Hammouche et al. proposed the method that combines a genetic algorithm with a wavelet transform [6]. It uses lower resolution version of the histogram and corresponding thresholds are later projected onto the original space. Manikandan et al. in 2014 used real coded genetic algorithm with simulated binary crossover based multilevel thresholding for the segmentation of medical brain images [7].

Other interesting nature inspired algorithms were used for the digital image multilevel thresholding. Oliva et al. introduced multilevel thresholding algorithm based on the harmony search evolutionary method which is inspired by musicians improvising new harmonies while playing [8]. Sun and Zhang used the improved variant of the gravitational search algorithm strengthened with genetic algorithm for ability to achieve generation jumping when getting stuck at local optima [9]. Oliva et al. proposed a version of electromagnetism-like evolutionary method which mimics the attraction-repulsion mechanism among charges to evolve the members of a population. This approach obtained good results on both objective funtions, Kapur's and Otsu's [10].

Differential evolution is a very successful, also rather old, nondeterministic mataheuristic widely applied to many different optimization problems. It was also successfully applied to multilevel image thresholding. Sarkar and Das used differential evolution algorithm with 2D histogram based approach and obtained superior results compared to genetic algorithm, particle swarm optimization, artificial bee colony, and simulated annealing. Berkley segmentation data set (BSDS300) with 300 distinct images was used for testing [11]. Charansiriphaisan et al. showed that ordinary differential evolution fails when the number of thresholds is greater than 12. They introduced an improved differential evolution using a new mutation strategy to overcome this problem. Their tests were conducted on 20 real images and the number of thresholds varied from 2 to 16 [12]. Ouadfel and Meshoul conducted a comprehensive comparative study by investigating the potential of the differential evolution algorithm compared to two other bio-inspired algorithms: artificial bees colony and particle swarm optimisation [13].

5.2 Older swarm intelligence algorithms

Ant colony optimization is the oldest swarm intelligence algorithm. It was used for multilevel thresholding in a number of papers by Liang, Yin et al. They proposed a hybrid optimization scheme based on an ant colony optimization algorithm with the Otsu and Kittler's methods, where Otsu's method was more successful [14]. In another article they combined parametric and non-parametric approaches. An ant colony optimization algorithm considered the non-parametric objective between-class variance while the expectation-maximization (EM) algorithm focused on the parametric objective overall fitting error of probability distributions [15].

Particle swarm optimization is old and most used swarm intelligence algorithm. Yin used recursive programming technique which reduces computing time for minimum cross entropy thresholding objective function and then applied particle swarm optimization algorithm for searching the near-optimal thresholds [16]. Convergence of the proposed method was analyzed mathematically. Liu et al. proposed the modified adaptive particle swarm optimization algorithm with dynamic population strategy that enables the population size to vary at run time. It can increase when the swarm converges and decrease when the swarm disperses [17]. Kurban et al. presents a comparison of evolutionary and swarm-based optimization algorithms for multilevel color image thresholding problem. Evolution strategy, genetic algorithm, differential evolution, adaptive differential evolution and swarm-based algorithms such as particle swarm optimization, artificial bee colony, cuckoo search and differential search algorithm have been tested and compared using Kapur's entropy as the fitness function to be maximized. Swarm algorithms gave better results but were slower [18]. Liu et al. presented another version of modified particle swarm optimization algorithm that employs two new strategies to improve the performance of original particle swarm optimization: adaptive inertia and adaptive population. Adaptive inertia allows inertia weight to change with the searching state, which helps the algorithm to increase search efficiency and convergence speed. Adaptive population strategy keeps the population size also variable which mainly helps the algorithm to jump out of local optima. The searching state is estimated as exploration or exploitation according to whether the best so far solution been updated in number of consecutive generations or not [19].

Artificial bee colony is one of the recent and very successful swarm intelligence metaheuristics. Horng applied it to the maximum entropy thresholding and compared the results to the particle swarm optimization, the hybrid cooperative-comprehensive learning based PSO algorithm, the Fast Otsu's method and the honey bee mating optimization. Quality of results was similar, but speed of convergence was better for the artificial bee colony algorithm [20]. Zhang and Wu used artificial bee colony to prove that Tsallis entropy as a general information theory entropy formalism may be better for thresholding than Shannon entropy [21]. Akay compared two successful swarm-intelligence-based global optimisation algorithms, particle swarm optimisation and artificial bee colony for finding the optimal multilevel thresholds. Kapur's entropy and between-class variance have been used as fitness functions [22]. Charansiriphaisan et al. analyse and discuss a family of artificial bee colony algorithms: the standard ABC, ABC/best/1, ABC/best/2, IABC/best/1, IABC/rand/1, and CABC, and some particle swarm optimization-based algorithms for searching multilevel thresholding. The experimental results showed that IABC/best/1 outperformed the other techniques [23]. Osuna-Enciso et al. used the method based on the mixture of Gaussian functions to approximate the 1D histogram of a gray level image and whose parameters are calculated using three nature inspired algorithms: particle swarm optimization, artificial bee colony optimization and differential evolution [24].

Honey bee mating is another swarm intelligence algorithm based on bees. Horng applied it to the minimum cross entropy thresholding and compared it to the exhaustive search, the particle swarm optimization and the quantum particle swarm optimization [25]. Horng also applied honey bee mating algorithm to the maximum entropy thresholding [26].

Bacterial foraging algorithm, glowworm swarm optimization, seeker optimization alorithm and shuffled frog-leaping algorithm are examples of not so widely used metaheuristics, nevertheless successful for some applications. Sathya and Kayalvizhi used bacterial foraging algorithm for multilevel thresholding. They applied Kapur's and Otsu's objective functions and the feasibility of the proposed technique has been tested on ten standard test images and compared with particle swarm optimization algorithm and genetic algorithm [28]. Same authors improved the global searching ability and convergence speed of the bacterial foraging algorithm by allowing the best bacteria among all the chemotactic steps to be passed to the subsequent generations [29]. Luo et al. tested glowworm swarm optimization for the Otsu based thresholding [30], while Tuba and Brajevic implemented the modified human seeker optimization algorithm [27]. Horng applied the shuffled frog-leaping algorithm to the minimum cross entropy thresholding [31], as well as to the maximum entropy thresholding [32]. Compared to the honey bee mating optimization, the firefly algorithm, the particle swarm optimization, the hybrid cooperative-comprehensive learning based PSO algorithm and artificial bee colony algorithm they showed great potential.

5.3 Recent swarm intelligence algorithms

Cuckoo search is one of the latest swarm intelligence algorithms, created by Yang. Agrawal et al. implemented cuckoo search for multilevel thresholding and the results were compared with that of bacteria foraging optimization, artificial bee colony algorithm, particle swarm optimization and genetic algorithm [33], while Panda et al. introduced new theoretical formulation for objective functions based on edge magnitude of an image. The gray level co-occurrence matrix (second order statistics) of the image was used for obtaining multilevel thresholds by optimizing the edge magnitude using cuckoo search technique [34]. Bhandari et al. used cuckoo search algorithm and wind driven optimization for multilevel thresholding using Kapur's entropy. Experimental results have been examined on standard set of satellite images using various numbers of thresholds [35]. Brajevic and Tuba [36] provided comprehensive analysis of the cuckoo search and firefly algorithms for Kapur and Otsu objective functions and compared results with a number of other optimization algorithms that included differential evaluation and particle swarm optimization. Roy et al. concentrated on the minimum cross-entropy criterion for image segmentation and cuckoo search algorithm has been compared against old genetic algorithms and PSO [37].

Firefly is also one of the latest swarm intelligence algorithms created by Yang. Horng and Liou tested firefly algorithm for minimum cross entropy thresholding and compared it to the exhaustive search, the particle swarm optimization, the quantum particle swarm optimization and honey bee mating optimization [38]. Brajevic and Tuba, as mentioned before, also improved results for this algortihm [36]. Raja Et al. implemented histogram based multilevel thresholding using Brownian distribution guided firefly algorithm. A bounded search technique was also included to improve the optimization accuracy with fewer iterations. Otsu's between-class variance objective function was maximized [39].

Bat algorithm is the latest swarm intelligence algorithm created by Yang. Alihodzic and Tuba were the first to implement bat algorithm for multilevel thresholding. Results were very good and they modified bat algorithm to further improve its performance [40]. Both, Kapur's and Otsu's criteria were used and exhaustive search was performed for 2, 3, 4, and 5 thresholds. The testing was done on the same 6 benchmark images used in [36]. All these swarm intelligence algorithms required less that 0.1 sec of computational time for 5 thresholds and could easily be applied to any size problem, with result almost always equal to results of exhaustive search i.e. global optimum. The proposed improved bat algorithm, by taking some features of the DE and ABC algorithms, obtained the best results compared to the rest of algorithms. It actually achieved the best result for both, mean value and variance, for all tested cases and can be considered the state-of-the-art for this area.

Table 3 shows computational times and average required iterations for 6 tested images and 6 tested algorithms for Kapur's method. Most significant conclusions concerning the convergence speed of the tested algorithms for Kapur's method are shown in Table 4.

Table 3. Mean of the CPU times (in milliseconds) and mean of the iteration numbers for the PSO, DE, CS, FA, BA and IBA based on Kapur's entropy criterion for six test images over 50 runs

Alg.	K	Barbara		Living room		Boats	
		M time (ms)	M iter	M time (ms)	M iter	M time (ms)	M iter
PSO	2	2.18	9.22	102.82	1165.14	3.08	11.84
	3	3.30	14.28	21.96	218.56	16.23	136.58
	4	49.00	495.36	77.23	853.62	123.62	1367.86
	5	88.16	1050.8	153.53	1725.22	74.46	814.22
DE	2	3.56	14.59	4.34	18.92	3.38	16.92
	3	6.3	30.0	8.02	70.60	8.84	43.32
	4	24.46	240.68	33.62	322.91	46.92	477.16
	5	47.8	527.96	77.14	801.21	65.48	683.29
CS	2	71.04	194.54	53.30	129.84	53.84	135.58
	3	150.71	420.42	125.48	322.42	128.92	330.22
	4	189.31	518.58	222.48	570.6	170.28	436.02
	5	301.65	786.64	499.42	1303.8	247.15	631.36
FA	2	15.01	11.96	17.02	11.9	16.39	12.32
	3	34.07	29.82	37.24	29.7	36.24	29.24
	4	43.30	38.6	50.25	77.9	54.68	117.8
	5	50.15	44.04	104.74	515.06	76.22	241.94
BA	2	1.78	2.04	96.82	722.92	24.17	142.96
	3	2.52	6.74	89.78	734.8	64.96	421.48
	4	40.2	146.46	124.16	1098.72	115.34	969.66
	5	60.9	412.92	175.32	1718.82	98.42	785.04
IBA	2	2.12	9.14	6.34	25.14	5.36	10.02
	3	3.84	16.8	5.3	22.8	5.5	22.4
	4	7.16	26.26	9.92	35.48	10.3	43.3
	5	8.66	40.06	24.7	134.38	14.4	50.9
Alg.	K	Goldhill		Lake		Aerial	
		M time (ms)	M iter	M time (ms)	M iter	M time (ms)	M iter
PSO	2	2.54	8.84	2.41	8.86	2.62	10.7
	3	3.18	13.54	3.65	14.58	3.24	13.9
	4	10.22	97.86	26.99	295.76	4.03	19.76
	5	23.56	258.5	77.64	893.98	58.54	695.92
DE	2	2.15	16.44	5.0	15.74	3.81	17.01
	3	7.94	69.30	6.24	30.04	6.51	30.96
	4	48.36	125.48	22.12	165.08	14.12	124.56
	5	22.2	116.28	55.9	603.01	56.38	624.16
CS	2	82.29	209.48	70.48	183.36	56.32	150.88
	3	149.68	441.64	138.23	375.66	104.75	292.22
	4	180.31	500.11	220.44	604.58	174.84	482.26
	5	280.55	720.34	336.29	915.92	193.21	532.76
FA	2	13.66	10.08	15.92	12.32	13.86	11.0
	3	32.08	28.66	34.56	29.42	32.83	29.18
	4	41.11	36.6	43.84	37.66	43.18	38.96
	5	48.95	42.01	50.31	43.6	50.46	45.46
BA	2	2	1.8	1.6	1.74	1.22	5.06
	3	2.84	6.94	2.32	8.82	3.6	14.4
	4	6.76	37.18	56.88	397.8	32.6	134.52
	5	30.92	157.92	39.24	238.12	64.48	463.78
IBA	2	2.88	8.92	2.28	9.18	2.7	11.18
	3	3.84	16.62	5	17.5	5.96	19.88
	4	6.28	28.82	7.14	26.98	7.48	30.98
	5	7.88	38.62	9.5	42.7	9.5	44.98

Table 4. The number of evaluations for all test images and all threshold values for Kapur's method

Alg.	Trsh. 2	Trsh. 3	Trsh. 4	Trsh. 5	Total
PSO	1214	411	3130	5439	10194
DE	96	186	1456	3356	5094
CS	1004	2183	3112	4891	11189
FA	70	176	347	932	1525
BA	876	1193	2784	3777	8631
IBA	74	116	192	352	734

In Table 4 (for Kapur's criterion) in each column labeled by $Thrs.\ k$ $(k = 2, 3, 4, 5)$ for each of the tested algorithms the sum of mean number of required iterations for each test image is reported. In the case of the FA and especially the IBA method, the number of iterations does not grow rapidly with the increase of the number of thresholds as is the case with the rest of algorithms. From Table 4 it can also be observed that the IBA converges in considerably less iterations compared to the rest of algorithms.

6 Conclusion

In this short review of swarm intelligence algorithms application to the multilevel image thresholding this hard optimization problem was defined, together with appropriate objective functions, namely Kapur's entropy method and Otsu's between-class variance method. The literature review of nature-inspired and specifically swarm intelligence algorithms with the current state is given. Testing is usually done on standard benchmark images and some synthetic images. There are very few results for larger number of thresholds for which the exact solutions from exhaustive search are not known.

Acknowledgments. Ministry of Science of Republic of Serbia has supported part of the research for this paper, Grant No. III-44006.

References

[1] J. N. Kapur, "A new method for gray-level picture thresholding using the entropy of the histogram," *Computer Vision, Graphics,*

and Image Processing, vol. 29, pp. 273–285, March 1985.

[2] N. Otsu, "A threshold selection method for grey level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[3] N. Papamarkos and B. Gatos, "A new approach for multilevel threshold selection," *CVGIP: Graphical Models and Image Processing*, vol. 56, no. 5, pp. 357–370, 1994.

[4] P.-Y. Yin and L.-H. Chen, "A fast iterative scheme for multilevel thresholding methods," *Signal Processing*, vol. 60, no. 3, pp. 305–313, 1997.

[5] Z.-H. Yang, Z.-B. Pu, and Z.-Q. Qi, "Relative entropy multilevel thresholding method based on genetic optimization," in *Proceedings of 2003 International Conference on Neural Networks and Signal Processing, ICNNSP'03*, vol. 1, pp. 583–586, 2003.

[6] K. Hammouche, M. Diaf, and P. Siarry, "A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation," *Computer Vision and Image Understanding*, vol. 109, no. 2, pp. 163–175, 2008.

[7] S. Manikandan, K. Ramar, M. Willjuice Iruthayarajan, and K. Srinivasagan, "Multilevel thresholding for segmentation of medical brain images using real coded genetic algorithm," *Measurement: Journal of the International Measurement Confederation*, vol. 47, no. 1, pp. 558–568, 2014.

[8] D. Oliva, E. Cuevas, G. Pajares, D. Zaldivar, and M. Perez-Cisneros, "Multilevel thresholding segmentation based on harmony search optimization," *Journal of Applied Mathematics*, vol. Article no. 575414, 2013.

[9] G. Sun and A. Zhang, "A hybrid genetic algorithm and gravitational search algorithm for image segmentation using multilevel

thresholding," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7887 LNCS, pp. 707–714, 2013.

[10] D. Oliva, E. Cuevas, G. Pajares, D. Zaldivar, and V. Osuna, "A multilevel thresholding algorithm using electromagnetism optimization," *Neurocomputing*, vol. 139, pp. 357–381, 2014.

[11] S. Sarkar and S. Das, "Multilevel image thresholding based on 2d histogram and maximum tsallis entropy - a differential evolution approach," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4788–4797, 2013.

[12] K. Charansiriphaisan, S. Chiewchanwattana, and K. Sunat, "A global multilevel thresholding using differential evolution approach," *Mathematical Problems in Engineering*, vol. Article no. 974024, 2014.

[13] S. Ouadfel and S. Meshoul, "Bio-inspired algorithms for multilevel image thresholding," *International Journal of Computer Applications in Technology*, vol. 49, no. 3-4, pp. 207–226, 2014.

[14] Y.-C. Liang, Y.-C. Yin, and A.-L. Chen, "Multilevel thresholding using ant colony optimization," in *Lecture Notes in Engineering and Computer Science*, vol. 2, pp. 1848–1853, 2007.

[15] Y.-C. Liang and Y.-C. Yin, "A new multilevel thresholding approach based on the ant colony system and the em algorithm," *International Journal of Innovative Computing, Information and Control*, vol. 9, no. 1, pp. 319–337, 2013.

[16] P.-Y. Yin, "Multilevel minimum cross entropy threshold selection based on particle swarm optimization," *Applied Mathematics and Computation*, vol. 184, no. 2, pp. 503–513, 2007.

[17] Y. Liu, C. Mu, and W. Kou, "Optimal multilevel thresholding using the modified adaptive particle swarm optimization," *International Journal of Digital Content Technology and its Applications*, vol. 6, no. 15, pp. 208–219, 2012.

[18] T. Kurban, P. Civicioglu, R. Kurban, and E. Besdok, "Comparison of evolutionary and swarm based computational techniques for multilevel color image thresholding," *Applied Soft Computing Journal*, vol. 23, pp. 128–143, 2014.

[19] Y. Liu, C. Mu, W. Kou, and J. Liu, "Modified particle swarm optimization-based multilevel thresholding for image segmentation," *Soft Computing*, vol. Article in Press, 2014.

[20] M.-H. Horng, "Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13785–13791, 2011.

[21] Y. Zhang and L. Wu, "Optimal multi-level thresholding based on maximum tsallis entropy via an artificial bee colony approach," *Entropy*, vol. 13, no. 4, pp. 841–859, 2011.

[22] B. Akay, "A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding," *Applied Soft Computing Journal*, vol. 13, no. 6, pp. 3066–3091, 2013.

[23] K. Charansiriphaisan, S. Chiewchanwattana, and K. Sunat, "A comparative study of improved artificial bee colony algorithms applied to multilevel image thresholding," *Mathematical Problems in Engineering*, vol. 2013, 2013.

[24] V. Osuna-Enciso, E. Cuevas, and H. Sossa, "A comparison of nature inspired algorithms for multi-threshold image segmentation," *Expert Systems with Applications*, vol. 40, no. 4, pp. 1213 – 1219, 2013.

[25] M.-H. Horng, "Multilevel minimum cross entropy threshold selection based on the honey bee mating optimization," *Expert Systems with Applications*, vol. 37, no. 6, pp. 4580–4592, 2010.

[26] M.-H. Horng, "A multilevel image thresholding using the honey bee mating optimization," *Applied Mathematics and Computation*, vol. 215, no. 9, pp. 3302–3310, 2010.

[27] M. Tuba and I. Brajevic, "Modified seeker optimization algorithm for image segmentation by multilevel thresholding," *International Journal of Mathematical Models and Methods in Applied Sciences*, vol. 7, no. 4, pp. 370–378, 2013.

[28] P. Sathya and R. Kayalvizhi, "Optimal multilevel thresholding using bacterial foraging algorithm," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15549–15564, 2011.

[29] P. Sathya and R. Kayalvizhi, "Modified bacterial foraging algorithm based multilevel thresholding for image segmentation," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 4, pp. 595–615, 2011.

[30] Q. Luo, Z. Ouyang, X. Chen, and Y. Zhou, "A multilevel threshold image segmentation algorithm based on glowworm swarm optimization," *Journal of Computational Information Systems*, vol. 10, no. 4, pp. 1621–1628, 2014.

[31] M.-H. Horng, "Multilevel minimum cross entropy threshold selection with shuffled frog-leaping algorithm," *Advances in Information Sciences and Service Sciences*, vol. 4, no. 11, pp. 168–176, 2012.

[32] M. Horng, "Multilevel image threshold selection based on the shuffled frog-leaping algorithm," *Journal of Chemical and Pharmaceutical Research*, vol. 5, no. 9, pp. 599–605, 2013.

[33] S. Agrawal, R. Panda, S. Bhuyan, and B. Panigrahi, "Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm," *Swarm and Evolutionary Computation*, vol. 11, pp. 16–30, 2013.

[34] R. Panda, S. Agrawal, and S. Bhuyan, "Edge magnitude based multilevel thresholding using cuckoo search technique," *Expert Systems with Applications*, vol. 40, no. 18, pp. 7617–7628, 2013.

[35] A. Bhandari, V. Singh, A. Kumar, and G. Singh, "Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using kapur's entropy," *Expert Systems with Applications*, vol. 41, no. 7, pp. 3538–3560, 2014.

[36] I. Brajevic and M. Tuba, "Cuckoo search and firefly algorithm applied to multilevel image thresholding," in *Cuckoo Search and Firefly Algorithm: Theory and Applications* (X.-S. Yang, ed.), vol. 516 of *Studies in Computational Intelligence*, pp. 115–139, Springer International Publishing, 2014.

[37] S. Roy, U. Kumar, D. Chakraborty, S. Nag, A. Mallick, and S. Dutta, "Comparative analysis of cuckoo search optimization-based multilevel image thresholding," *Advances in Intelligent Systems and Computing*, vol. 2, no. 309 AISC, pp. 327–342, 2015.

[38] M.-H. Horng and R.-J. Liou, "Multilevel minimum cross entropy threshold selection based on the firefly algorithm," *Expert Systems with Applications*, vol. 38, no. 12, pp. 14805–14811, 2011.

[39] N. Sri Madhava Raja, V. Rajinikanth, and K. Latha, "Otsu based optimal multilevel image thresholding using firefly algorithm," *Modelling and Simulation in Engineering*, vol. Article no. 794574, 2014.

[40] A. Alihodzic and M. Tuba, "Improved bat algorithm applied to multilevel image thresholding," *The Scientific World Journal*, vol. 2014, Article ID 176718, p. 16, July 2014.

Milan Tuba[1,2]

Received September 16, 2014

[1]Faculty of Mathematics, University of Belgrade
Studentski trg 16, 11000 Belgrade, Serbia
[2]Faculty of Computer Science, Megatrend University
Bulevar umetnosti 29, 11070 Belgrade, Serbia
E–mail: *tuba@ieee.org*