Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 1445 – 1460

doi:10.3906/elk-1703-240

An improved form of the ant lion optimization algorithm for image clustering problems

Metin TOZ*
Department of Computer Engineering, Faculty of Technology, Düzce University, Düzce, Turkey

Received: 20.03.2017 • **Accepted/Published Online:** 22.01.2019 • **Final Version:** 22.03.2019

Abstract: This paper proposes an improved form of the ant lion optimization algorithm (IALO) to solve image clustering problem. The improvement of the algorithm was made using a new boundary decreasing procedure. Moreover, a recently proposed objective function for image clustering in the literature was also improved to obtain well-separated clusters while minimizing the intracluster distances. In order to accurately demonstrate the performances of the proposed methods, firstly, twenty-three benchmark functions were solved with IALO and the results were compared with the ALO and a chaos-based ALO algorithm from the literature. Secondly, four benchmark images were clustered by IALO and the obtained results were compared with the results of particle swarm optimization, artificial bee colony, genetic, and K-means algorithms. Lastly, IALO, ALO, and the chaos-based ALO algorithm were compared in terms of image clustering by using the proposed objective function for three benchmark images. The comparison was made for the objective function values, the separateness and compactness properties of the clusters and also for two clustering indexes Davies–Bouldin and Xie–Beni. The results showed that the proposed boundary decreasing procedure increased the performance of the IALO algorithm, and also the IALO algorithm with the proposed objective function obtained very competitive results in terms of image clustering.

Key words: Image clustering, improved ant lion optimization, Davies–Bouldin, Xie–Beni

1. Introduction

Clustering is an unsupervised data grouping technique that has been widely applied in many fields such as machine learning, pattern recognition, data mining, and image processing [1]. It aims to reveal the hidden structures in an unlabeled dataset and thus to provide the possibility of making a preliminary assessment about the organization of the dataset [2]. By utilizing a clustering algorithm, a dataset can be divided into several disjoint groups of data points according to some similarity measures. The algorithm tries to maximize the similarity within each group while minimizing the similarity between the groups. The clustering algorithms can be classified into two basic categories, hierarchical and partitional clustering [3]. The data points are being grouped into a tree-like structure with the hierarchical clustering while the dataset is divided into several clusters that provide some predefined criteria by using the partitional clustering [3]. In the literature, there are a number of subtypes of these two clustering approaches; agglomerative and divisive clustering are two hierarchical clustering techniques, the agglomerative clustering starts by assigning each data member to a distinct cluster and continues by combining the successive clusters while divisive clustering begins with one cluster and continues by dividing this cluster into different numbers of clusters [1]. Both of these techniques

*Correspondence: metintoz@duzce.edu.tr

continue until a predefined stopping criterion is met. A partitional approach starts with a predefined number of clusters and tries to divide the dataset into that number of disjoint clusters by evaluating the data points according to some optimization criteria [4]. K-means clustering algorithm that was proposed by McQueen [5] and its fuzzy-based version, the fuzzy C-means (FCM) proposed by Dunn [6] and improved by Bezdek [7], are the two most widely known partitional algorithms. These two algorithms have very simple formulations to be applied to most kinds of the clustering applications and also very computational efficient. However, they have some disadvantages such as being trapped in the local minima and being very sensitive to the selection of the initial cluster centers [4]. The partitional clustering can also be regarded as an optimization process because of optimizing a certain criterion such as minimizing the distance between the cluster centers[8]. Therefore, in order to tackle the drawbacks of K-means and FCM algorithms, many researchers proposed to use evolutionary and/or metaheuristic optimization algorithms. Some of the studies about hybridization FCM or K-means with the optimization algorithms can be given as follows; Nikham and Amiri [9] proposed a hybrid clustering algorithm based on a fuzzy adaptive form of particle swarm optimization (PSO), ant colony optimization (ACO), and K-means algorithms and presented that their algorithm showed better results than some other metaheuristic algorithms such as PSO, genetic algorithms (GA), and ACO. Krishnasamy et al. [3] combined K-means and cohort intelligence algorithm and proposed a new hybrid data clustering algorithm named K-MCI. In [10], Biniaz and Abbasi combined FCM with an unsupervised ACO algorithm in medical image segmentation applications. Kumar and Sahoo [11] proposed a two-step artificial bee colony (ABC) algorithm, where they produced the initial population by using K-means, and they also proposed an improved solution search equation based on PSO social behavior. The authors showed that their algorithm outperforms the classical form of ABC in solving the clustering problem. Wang et al. [12] composed the supervised learning normal mixture model and the FCM. They conducted some experiments on real datasets and concluded that the supervised learning normal mixture model can improve the performance of FCM. Toz and Toz [13] proposed a hybrid clustering algorithm based on differential search optimization algorithm and FCM in order to use in image clustering applications. Different from the hybrid algorithms, metaheuristic and evolutionary optimization algorithms have also been successfully used to solve clustering problems. In [14], the authors proposed to use GA as a clustering technique and showed the superiority of GA-clustering algorithm over K-means by using some artificial and real-life datasets. Shelokar et al. [15] proposed to use ACO for clustering purposes and concluded that the ACO is superior to simulated annealing, GA, and tabu search techniques. Omran et al. [16] developed a PSO-based approach to solve image clustering problem and showed that their method is better than K-means, FCM, K-Harmonic means, and GA. An improved form of gravitational search algorithm by a special encoding scheme, called grouping encoding has been used in [17] in order to solve data clustering problem and it has been shown that the proposed method can be efficiently used for multivariate data clustering problem. Tang et al. [18] proposed a new algorithm, intrusive tumor growth inspired optimization algorithm, and solved the data clustering problem by using their algorithm. Karaboga and Ozturk [19] proposed to use the ABC algorithm for clustering applications and showed that the ABC algorithm outperforms PSO and nine classification techniques from the literature in solving data clustering problem. In another study, Ozturk et al. [4] proposed to use the ABC algorithm for solving image clustering problem by using a new objective function. They tested the proposed objective function by several benchmark images with different optimization algorithms and found that their objective function gives the best results with the ABC algorithm in terms of separateness and compactness of the clusters.

Ant lion optimization (ALO) algorithm is a new population-based optimization algorithm proposed by Mirjalili [20] in 2015 that is based on the hunting behaviors of the antlions. It was shown that this algorithm outperforms several optimization algorithms such as PSO, GA, cuckoo search, and firefly algorithm for solving some benchmark functions [20]. The ALO algorithm has been used to solve different kinds of optimization problems since it has been introduced. In [21], Zawbaa et al. proposed to use an improved form of ALO by chaos (CALO) to solve feature selection problem in data mining. They formulated the problem as a multiobjective optimization problem and tested the proposed algorithm on different datasets. Authors concluded that the CALO outperforms ALO, PSO, and GA in terms of the quality of the selected features. Babers et al. [22] used ALO to solve a multiobjective optimization problem defined for social networks, namely, detection of the optimum number of communities in online social networks. They showed that the ALO algorithm can be efficiently used to find an optimized community structure. Chopra and Mehta [23] solved the optimum generation scheduling problem for the thermal generators in power systems and used three test systems for performance evaluation. Their results showed that ALO obtained competitive results according to the compared algorithms such as PSO and GA. In this study, we firstly proposed to use the radius decrement process of the vortex search algorithm[27] for the boundary decreasing procedure of the ALO algorithm in order to improve its performance in image clustering and named the new form of the algorithm as the improved ALO (IALO) algorithm. Secondly, we propose a new version of the objective function proposed by Ozturk et al.[4] by deducing the maximum value of the database from its equation to make the objective function to be less sensitive to the outliers. Finally, we performed experimental studies both for testing the performance of the IALO algorithm on solving twenty-three benchmark functions from the literature [28] and for testing the performances of the proposed methods in image clustering. The obtained results are presented in a comparative manner and, in order to show the effect of the quality of the clustering operations, the results of the image clustering applications are also evaluated in terms of two clustering indexes, Davies–Bouldin (DBI) [24] and Xie–Beni (XBI) [25]. The rest of the paper is organized as follows; the ALO algorithm and the proposed IALO algorithm are presented in details in Section 2, the solution method of the image clustering problems by using the IALO algorithm and the proposed objective function is given in Section 3, the experiments and the comparisons are presented in Section 4, and finally the paper concluded with the last section.

2. Ant lion optimization algorithm

ALO is a new population-based optimization algorithm proposed by Mirjalili [20] in 2015. It is based on the hunting behavior of the antlions when they are in the larval form. Antlion larvae use cone-shaped pits as traps for hunting their preys that are mainly ants. Once an antlion realizes a prey is in the pit it throws sands toward the edge of the pit to cause the prey to slip down to the bottom. At the end of the hunt, the antlion consumes its prey and prepares the pit for the next hunt [20]. The mathematical model of the ALO algorithm is based on the relationships between the antlion, the pit, and the ants as preys. The ALO algorithm uses two populations to solve an optimization problem. The first one is for the ants that are the candidate solutions to the problem and move stochastically over the search space while the second for the antlions that are hidden in random locations in the search space. Both of the populations are defined in the same manner as follows [20];

$$P_A = \begin{bmatrix} P_{A_{11}} & \dots & P_{A_{1d}} \\ \vdots & \ddots & \vdots \\ P_{A_{n1}} & \dots & P_{A_{nd}} \end{bmatrix} \quad P_{AL} = \begin{bmatrix} P_{AL_{11}} & \dots & P_{AL_{1d}} \\ \vdots & \ddots & \vdots \\ P_{AL_{n1}} & \dots & P_{AL_{nd}} \end{bmatrix} \tag{1}$$

where P_A and P_{AL} are the populations of the ants and the antlions, respectively. Both of them are the matrices of the same size and d and n are the dimension of the solution and the number of the populations, respectively.

The ants move randomly over the search space by a random walk vector as given in Eq. (2) [20].

$$W(t) = \begin{bmatrix} 0 & U(2v(t_1) - 1) & \dots & U(2v(t_T) - 1) \end{bmatrix} \tag{2}$$

where $W(t)$ is the random walk matrix, t is the step of the random walk which is determined as the iteration number and U and T are the cumulative sum function and the maximum number of iterations, respectively. Finally, $v(t)$ is a stochastic function defined as follows [20];

$$v(t) = \{\ 1\ if\ rand > 0.5 \quad 0\ if\ rand \leq 0.5 \tag{3}$$

where $rand$ indicates a random number with uniform distribution in the interval [0,1]. In order to keep the ant's position in the boundaries of the search space, the following min-max normalization function is used for the random walk matrix of the ants [20].

$$W_i^t = \frac{(W_i^t - a_i)(q_i^t - l_i^t)}{\beta_i - \alpha_i} + l_i^t \tag{4}$$

In the equation, W_i^t, l_i^t, and q_i^t are the random walk vector and the minimum and maximum values of the lower and upper boundaries for i'th variable at t'th iteration, respectively. Lastly, α_i and β_i are the minimum and maximum values of random walk of i'th variable. In one step of the iteration process, each of the ants is assumed to be in the trap of only one antlion which is selected by a roulette wheel mechanism according to its fitness value. In order to model an ant being in the trap, the minimum and maximum boundaries of the random walks of the ant is being effected by the position of the selected antlion [20].

$$l_i^t = l^t + P_{AL_j}^t \quad q_i^t = q^t + P_{AL_j}^t \tag{5}$$

where $P_{AL_j}^t$ $(j = 1, 2, \dots, n)$ is the position of the selected (j'th) antlion at t'th iteration and l^t and q^t are the minimum and maximum values of all the variables of the i'th ant. In order to simulate sliding of an ant towards the bottom of the pit, l^t and q^t are adaptively decreased as follows;

$$l_i^t = \frac{l^t}{\tau}\ ;\ q_i^t = \frac{q^t}{\tau}\ ;\ \tau = 10^w \frac{t}{T} \tag{6}$$

where $w = \{2\ if\ t > 0.1T; 3\ if\ t > 0.5T; 4\ if\ t > 0.75T; 5\ if\ t > 0.9T; 9\ if\ t > 0.95T\}$, τ is an adaptively increased number, and w is the parameter used to adjust the level of the exploitation [20]. In the optimization process, the evaluations of both matrices are obtained by a fitness function [20];

$$F_A = \begin{bmatrix} f(P_{A_{11}}, & P_{A_{12}} \dots & , P_{A_{1d}}) \\ \vdots & \ddots & \vdots \\ f(P_{A_{n1}}, & P_{A_{n2}} \dots & , P_{A_{nd}}) \end{bmatrix} \quad F_{AL} = \begin{bmatrix} f(P_{AL_{11}}, & P_{AL_{12}} \dots & , P_{AL_{1d}}) \\ \vdots & \ddots & \vdots \\ f(P_{AL_{n1}}, & P_{AL_{n2}} \dots & , P_{AL_{nd}}) \end{bmatrix} \tag{7}$$

where F_A and F_{AL} are the fitness vector of theP_A and P_{AL} matrices, respectively, while f is the problem-depended objective function. The catching and consuming of the ant by the antlion is simulated by comparison of their fitness values. If the fitness value of the ant is greater than the antlion's fitness value, then it is assumed that the ant is caught and consumed by the antlion and the antlion moves the position of the ant for the next hunt [20].

$$P_{AL_j}^t = P_{A_i}^t\ if\ f(P_{A_i}^t) > f(P_{AL_j}^t), \tag{8}$$

where $P^t_{AL_j}$ and $P^t_{A_i}$ show the positions of the selected antlion and the i'th ant at t'th iteration, respectively. The last stage of the ALO algorithm is elitism which is provided by assuming that all the ants move towards the best antlion that has the greatest fitness value while moving towards the selected antlions by the roulette wheel simultaneously.

$$P^t_{A_i} = \frac{V^t_A + V^t_E}{2}, \tag{9}$$

where V^t_A and V^t_E are the random walks of the selected ant towards the selected antlion by roulette wheel and towards the elitist antlion at t'th iteration, respectively. The details of the algorithm can be found in [20].

2.1. Improved form of the ALO algorithm

The effectiveness of the ALO algorithm in terms of its searching capabilities such as avoiding the local optima and converging to the best solution was shown by comparing it with some of the state-of-the-art optimizations algorithms [20]. However, the ALO algorithm has a drawback. It uses a step-by-step decreasing procedure for boundary shrinking as seen in Eq. (6). In this equation, it can be seen that the w value is increased at some stationary points of the iterations to provide the boundary shrinking around the candidate solutions. Although this procedure provides an absolute reduction of the boundary around the solutions, it also restricts the random search capabilities of the algorithm because of the stationary points. In order to solve this issue, in this study, we propose a new boundary decreasing procedure for the ALO algorithm. The proposed procedure is inspired by the radius decrement process of the vortex search algorithm [27] and is based on the inverse of the incomplete gamma function for decreasing the l^t and q^t values. The incomplete gamma function is defined as follows [27];

$$\gamma(x, a) = \int_0^x e^{-t} t^{a-1} dt \tag{10}$$

Here, $a > 0$ and is known as the shape parameter while x is a random number. In this study, the inverse of the incomplete gamma function is calculated by using the $gammaincinv(x, a)$ function of the $MATLAB^{®}$. Based on this function, the proposed decreasing procedure can be defined as;

$$l^t = \frac{1}{x} gammaincinv(x, 1 - \frac{t}{T}) l^t, \tag{11}$$

$$q^t = \frac{1}{x} gammaincinv(x, 1 - \frac{t}{T}) q^t. \tag{12}$$

The difference between the boundary decreasing procedure and the proposed one can be shown by drawing the graph of the two procedures. Let the upper bound of one dimension of a candidate solution be 5 and the maximum number of the iterations for the algorithm be 200. By using these parameters, the decreasing of the upper bound of the solution would be as given in Figure 1 for both of the methods. It should be noted that the x value of the proposed method is selected as x = 0.05 in the graphics.

As can be seen from the figure, the decreasing of the upper bound of the solution changed only at stationary points when the decreasing procedure of the ALO is used. On the other hand, when the proposed method is used, the decreasing of the upper bound occurs in a curve-shaped structure. Therefore, it would be possible for the algorithm to search more points around the solution than the other method. Thus, the proposed procedure offers more search capabilities to the algorithm. In order to test the effect of the proposed

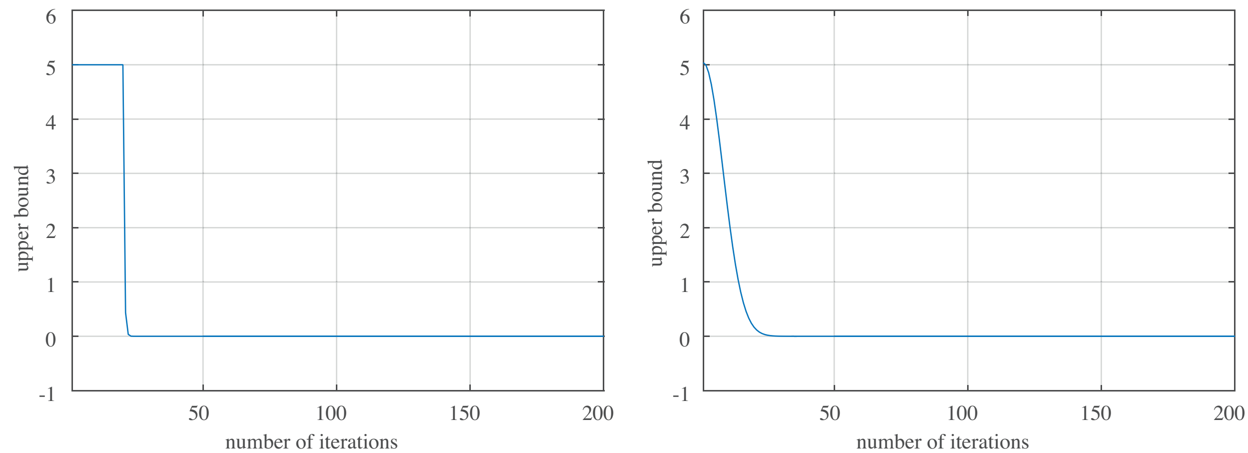

Figure 1. Boundary decreasing procedures for the ALO algorithm (left) and the proposed procedure (right).

method, we used twenty-three benchmark functions that have been used for comparison of the performances of the optimization algorithms in the literature [28]. These functions are combined from unimodal (F1-F7), multimodal (F8-F13), and fixed-dimension multimodal (F14-F23) functions and given in Table 1. In order to test the performance of the IALO algorithm, we compared its performance with the ALO algorithm [20] and with the chaos-based ALO (CALO) algorithm proposed by Zawbaa et al. [21]. It should be noted that the $MATLAB^{®}$ code for the standard ALO algorithm was obtained from its publicly available package that is referred in [20] and for the CALO was obtained from its implementation that performed by Emary (the second author of the CALO [21]). The benchmark functions given in Table 1 are solved with the three algorithms by using the same parameters (population number = 40, the maximum number of the iterations = 500) under the same conditions. In addition, since the authors of the CALO algorithm stated that the best chaotic map is the tent map for their algorithm, CALO was used with the tent map and the x parameter of the IALO algorithm was selected as $x = 0.01$. In the test, the algorithms were run 30 times for each of the functions and the mean and the standard deviations of the obtained minimum objective function values were recorded. These results are given in Table 2. As can be seen from Table 2, the IALO algorithm outperformed the ALO and CALO algorithms on solving nine of $(f1, f2, f6, f9, f11, f12, f15, f18,$ and $f20)$ twenty-three benchmark functions while the CALO got the best results on solving eight functions $(f7, f14, f16, f17, f19, f21 - f23)$ and the ALO performed better than the others only for six functions $(f3, f4, f5, f8, f10,$ and $f13)$. According to these results, it can be said that while the proposed IALO algorithm can solve problems from all the three function groups, unimodal, multimodal, and fixed-dimension multimodal, CALO is better at solving only fixed-dimension multimodal functions except f7 and the ALO is perform well on the unimodal and multimodal functions. On the other hand, in terms of the number of the functions that the algorithms get better results, it can be said that the IALO outperformed the other two algorithms. Therefore, it is shown that the proposed method improved the ALO algorithm in terms of avoiding the local optima and convergence to the best value.

3. Solving the image clustering problem with the IALO algorithm

Clustering problem is to divide a dataset into several groups by gathering similar items in the same group while increasing the dissimilarity between the groups. Formally, the clustering problem can be defined as follows. Let

Table 1. Benchmark functions [28]

F	fmin	Dim	[min max]	Formulations
f1	0	30	[-100 100]	$\sum_{i=1}^{n} x_i^2$
f2	0	30	[-10 10]	$\sum_{i=1}^{n} \|x_i\| + \prod_{i=1}^{n} \|x_i\|$
f3	0	30	[-100 100]	$\sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j\right)^2$
f4	0	30	[-100 100]	$max_i \{\|x_i\|,\ 1 \leq i \leq n\}$
f5	0	30	[-30 30]	$\sum_{i=1}^{n-1} [100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2]$
f6	0	30	[-100 100]	$\sum_{i=1}^{n} (\|x_i + 0.5\|)^2$
f7	0	30	[-1.28 1.28]	$\sum_{i=1}^{n} ix_i^4 + random\,(0,1)$
f8	-418.929×5	30	[-500 500]	$\sum_{i=1}^{n} -x_i \sin\left(\sqrt{\|x_i\|}\right)$
f9	0	30	[-5.12 5.12]	$\sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$
f10	0	30	[-32 32]	$-20exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$
f11	0	30	[-600 600]	$\frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
f12	0	30	[-50 50]	$\frac{\pi}{n}\left\{10\sin(\pi y_1) + \sum_{i=1}^{n}(y_i - 1)^2\left[1 + 10sin^2(\pi y_{i+1})\right] + (y_n - 1)^2\right\}$ $+\sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i+1}{4} u(x_i, a, k, m) = \{k(x_i - a)^m\ if\ x_i > a$ $0\ if a < x_i < a\ ;\ k(-x_i - a)^m\ if\ x_i < -a$
f13	0	30	[-50 50]	$0.1\left\{sin^2(\beta\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2\left[1 + sin^2(3\pi x_i + 1)\right]\right\}$ $+0.1(x_n - 1)^2\left[1 + sin^2(2\pi x_n)\right] + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$
f14	1	2	[-65 65]	$\left(\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^{2}(x_i - a_{ij})^6}\right)^{-1}$
f15	0.0003	4	[-5 5]	$\sum_{i=1}^{11}[a_i - \frac{x_1\left(b_i^2 + b_i x_2\right)}{b_i^2 + b_i x_3 + x_4}]^2$
f16	-1.0316	2	[-5 5]	$4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^4 + 4x_2^4$
f17	0.398	2	[-5 5]	$\left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$
f18	3	2	[-2 2]	$\left[1 + (x_1 + x_2 + 1)^2\left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right]$ $\times[30 + (2x_1 - 3x_2)^2 \times \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)]$
f19	-3.86	3	[1 3]	$-\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$
f20	-3.32	6	[0 1]	$-\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$
f21	-10.1532	4	[0 10]	$\sum_{i=1}^{5}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$
f22	-10.4028	4	[0 10]	$\sum_{i=1}^{7}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$
f23	-10.5363	4	[0 10]	$\sum_{i=1}^{10}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$

S be a dataset that has m number of data points;

$$S = \{s_1, s_2, \ldots, s_p, \ldots, s_N\}, \tag{13}$$

where s_p is a data point of S with F features. The data points are mainly represented in a vectoral form where each element of the vector depicts one feature of the data. The main objective of the clustering problem

Table 2. The results of the solving benchmark functions with ALO, IALO(proposed) and CALO [21]

	ALO			IALO			CALO[21]		
f	mean	std	min	mean	std	min	mean	std	min
f1	4,38E-09	1,81E-09	1,84E-09	**3,68E-11**	1,14E-10	2,78E-15	5,191353	2,733238	1,176234
f2	0,553539	1,324468	1,23E-05	**0,000346**	0,000774	2,42E-07	0,777602	0,474966	0,216296
f3	**0,000659**	0,000835	2,64E-06	0,576631	0,628072	0,011881	14,87261	9,451631	2,012524
f4	**0,000856**	0,001198	8,00E-05	0,027898	0,092275	0,000161	2,156635	0,834552	0,977903
f5	**27,84226**	62,20052	0,000426	349,8356	744,8857	0,068356	195,6839	306,3529	19,08243
f6	4,62E-09	2,22E-09	8,71E-10	**4,54E-11**	1,76E-10	1,99E-14	4,156796	1,998273	1,587905
f7	0,015767	0,009823	0,001654	0,01374	0,009379	0,002192	**0,009526**	0,005312	0,004004
f8	**-2439,06**	449,8478	-3557,65	-2819,06	313,4645	-3617,37	-2759,21	364,5259	-4142,76
f9	19,40166	11,24677	2,984877	**14,72538**	5,069263	6,964708	19,00927	7,718614	8,799327
f10	**0,292401**	0,613405	1,41E-05	0,784114	1,006101	1,79E-07	2,824831	0,517123	1,895919
f11	0,221252	0,107537	0,059104	**0,20489**	0,100166	0,041876	0,97251	0,112632	0,632719
f12	1,485045	1,788868	8,20E-11	**0,119317**	0,233767	4,72E-09	0,840488	0,803796	0,045096
f13	**0,000701**	0,003838	5,33E-10	0,002199	0,004472	4,26E-10	0,206559	0,122454	0,052298
f14	2,707552	2,359855	0,998004	1,229549	0,620521	0,998004	**0,998004**	2,00E-07	0,998004
f15	0,002843	0,005945	0,000468	**0,002191**	0,004945	0,0005	0,003489	0,006737	0,00059
f16	-1,03163	9,86E-14	-1,03163	-1,03163	5,76E-16	-1,03163	**-1,03158**	6,84E-05	-1,03163
f17	0,397887	5,59E-14	0,397887	0,397887	0	0,397887	**0,397944**	6,49E-05	0,397888
f18	3	3,31E-13	3	**3**	4,93E-15	3	3,000139	0,0003	3
f19	-3,86278	2,30E-13	-3,86278	-3,86278	2,89E-12	-3,86278	**-3,86275**	3,34E-05	-3,86278
f20	-3,26236	0,060657	-3,322	**-3,27746**	0,059564	-3,322	-3,26677	0,063849	-3,32194
f21	-6,37655	3,27959	-10,1532	-7,28477	2,7947	-10,1532	**-8,31094**	2,615133	-10,151
f22	-7,10152	3,442793	-10,4029	-8,33325	3,2587	-10,4029	**-9,47831**	2,25927	-10,3992
f23	-8,24708	3,360059	-10,5364	-8,25432	3,363594	-10,5364	**-9,50591**	2,59076	-10,5352

is to divide S into L number of clusters;

$$C = \{C_1, C_2, \ldots, C_L\}, \tag{14}$$

where C is an optimum-partitioned form of the dataset and $C_i (i = 1, 2, ...L)$ represents i'th cluster. Therefore, S can be rewritten as follows.

$$S = \bigcup_{i=1}^{L} C_i. \tag{15}$$

The clustering problem can be solved in two manners; in hard clustering, a data point belongs to only one cluster while in fuzzy clustering, the data point belongs to all the clusters by different membership values. In this study, the hard clustering was performed. Therefore, in order to divide the S into L number of clusters through a hard clustering process the following conditions needed to be met [4]. Each data point of the data set needed to be attended to a cluster, a data point needed to be only a member of one cluster and each cluster should have at least one data point. The similarity measure is one of the most important parts of a clustering algorithm since it is used to determine the similarity between a data point and a cluster center in order to decide whether the data point belongs to that cluster or not. Representation of a group (or cluster) is generally made by a single element, the cluster center, that can or cannot be a real element of that group. In order to perform the clustering process by using an optimization algorithm, the selected objective function of the algorithm should compute and evaluate the similarities between the members and/or clusters in each iteration step. One of the last proposed objective functions was proposed by Ozturk et al. [4] for image clustering and tested on the ABC, PSO, and GA algorithms for several benchmark images. It was shown that the objective function outperforms three well-known objective functions in the literature with ABC optimization algorithm [4].

$$F(P_{A_i}, S) = J_e \frac{d_{max}(S, P_{A_i})}{d_{min}(S, P_{A_i})}(d_{max}(S, P_{A_i}) + z_{max} - d_{min}(S, P_{A_i}) + MSE), \tag{16}$$

where S is the dataset of the image to be clustered and P_{A_i} is a candidate solution to the image clustering problem. $F(P_{A_i}, S)$ is the fitness value calculated for P_{A_i}, $d_{max}(S, P_{A_i})$ is the maximum value of the average distances calculated for all the clusters according to their centers, $d_{min}(S, P_{A_i})$ is the minimum average distance between any pairs of clusters, z_{max} is the maximum value of the data set, J_e is quantization error which defined to calculate the quality of the clustering process, and MSE is the mean of the square of the all the distances calculated for all the patterns according to their related centers [4]. The variables in Eq. (16) are defined as follows [4];

$$d_{max}(S, P_{A_i}) = max(\sum_{\forall s_p \in L_{i,k}} \frac{d(s_p, m_{i,k})}{r_{i,k}}), \tag{17}$$

$$d_{min}(S, P_{A_i}) = min(d(m_{i,j}, m_{i,k}))\ j \neq k, \tag{18}$$

$$J_e = \frac{\sum_{k=1}^{L} \sum_{\forall s_p \in L_k} d(s_p, m_k)/r_k}{L}, \tag{19}$$

$$MSE = \frac{1}{N} \sum_{k=1}^{L} \sum_{\forall s_p \in L_k} d(s_p, m_k)^2, \tag{20}$$

where $L_{i,k}$ is the k'th cluster determined by P_{A_i}, $r_{i,k}$ is the number of patterns of the cluster $L_{i,k}$, $m_{i,k}$ is the center of $L_{i,k}$, and finally, $d(s_p, m_{i,k})$ is the Euclidian distance between s_p and $m_{i,k}$. The objective function given in Eq. (16) should be minimized for well-separated and well-compacted clusters [4]. Therefore, the proposed algorithm should minimize d_{max}, J_e, and MSE while maximizing d_{min}. The objective function given in Eq. (16) includes the maximum value of the database (z_{max}). According to the equation, it is obvious that this value will dramatically change the objective function value when the dataset includes outlier element(s). Therefore, z_{max} makes the objective function more sensitive to the outliers. Moreover, a very high z_{max} value will reduce the effects of d_{max}, d_{min}, and MSE values on the objective function and thus the quality of the clustering. Therefore, we propose to deduce z_{max} from the equation and take the absolute value of the difference between d_{max} and d_{min} in order to make the objective function less sensitive to the outliers and increase the effects of the d_{max}, d_{min}, and MSE values on the objective function value and improve the clustering quality.

$$F(P_{A_i}, S) = J_e \frac{d_{max}(S, P_{A_i})}{d_{min}(S, P_{A_i})} (abs(d_{max}(S, P_{A_i}) - d_{min}(S, P_{A_i})) + MSE), \tag{21}$$

where abs is the absolute value function. In this study, the proposed objective function given in Eq. (21) was used with the proposed IALO algorithm to solve the image clustering problem. The structure of the populations of the ant and antlions for the IALO algorithm are defined as follows;

$$P_A = \begin{bmatrix} A_{1,1}, & A_{1,2} \ldots, & A_{1,L} \\ \vdots & \ddots & \vdots \\ A_{n,1}, & A_{n,2} \ldots, & A_{n,L} \end{bmatrix} \quad P_{AL} = \begin{bmatrix} P_{A_{1,1}}, & P_{A_{1,2}} \ldots, & P_{A_{1,L}} \\ \vdots & \ddots & \vdots \\ P_{A_{n,1}}, & P_{A_{n,2}} \ldots, & P_{A_{n,L}} \end{bmatrix} \tag{22}$$

Here, P_A and P_{AL} are the populations of the ant and antlions defined for solving the image clustering problem, respectively, while L is the number of the clusters. And, the pseudo code of the clustering algorithm based on IALO is given in Figure 2.

```
[row col]←size(test image)
S←reshaped test image by 1×(row×col) in grayscale
[lb ub]←maximum and minimum values of S
L←5 (number of clusters)
P_AL←Randomly generated n×L sized population of antlions between lb and ub values
P_A←Randomly generated n×L sized population of ants between lb and ub values
; Calculate the fitness values of the antlions
    For each antlion of P_AL
        For each pattern (s_p ) of S
            assign s_p  to the nearest cluster of the antlion
            calculate the fitness value (F) for the antlion by Equation 21
        End
    End
; Main loop of the IALO algorithm for determining the best solution
    While stopping criterion is not satisfied
        ; Create random walks
        For each ant of P_A
            create a random walk for the ant by using the proposed ub and lb decreasing mechanism
            by Equations 11 and 12
            update the position of the ant according to the created random walk
        End
        ; Calculate the fitness values of the ants
        For each ant of P_A
            For each pattern (s_p ) of S
                assign s_p  to the nearest cluster of the ant
                calculate the fitness value (F)  for the ant by Equation 21
            End
        End
        ; Update the positions of the antlions by comparing their fitness values by the ants' fitness
        values
        ; Update the elitist antlion in the population
    End
Best solution (found cluster centers)←The elitist antlion
```

Figure 2. Pseudo code of the IALO algorithm for image clustering by the proposed objective function.

4. Experiments and comparison

In order to present the performance of the IALO algorithm for solving image clustering problem, we selected two groups of benchmark images. The first group includes Lena, Airplane, 48025, and 42049 images (Figure 3). The last two images of this group are from the image segmentation dataset by Berkeley [26]. The reason for selection of these images is that the authors of the paper given in [4] also used them with their proposed objective function and they presented the results of the clustering of these images by PSO, ABC, GA, and K-means algorithms. Therefore, by selecting these images, we are able to present an accurate comparison of the performance of the IALO algorithm according to the algorithms tested in [4]. The second group is composed of 35008, 43051, and 35010 benchmark images (Figure 4) from the image segmentation dataset by Berkeley [26]. These three images are used to test the performance of the standard ALO, IALO, and the CALO algorithms on solving image clustering problem by the proposed objective function in this study. During the optimization procedure, the obtained minimum objective function values and the properties that are related to the separateness and compactness of the clustering process (d_{max}, d_{min} and J_e) were saved in each trial. These properties are used for the performance evaluation of the algorithms in addition to two clustering validity indexes, DBI [24] and XBI [25]. Davies–Bouldin index (DBI) is a well-known general cluster separation measure proposed by Davies and Bouldin [24] to measure the performance of the clustering algorithms. It is based on the ratio of the sum of within cluster-scatter to between-cluster separation [24].

$$P_i = \frac{1}{n_i} \sum_{s_j \in L_i} d(s_j, m_i)^2, \tag{23}$$

$$R_{i,j} = \frac{P_i + P_j}{d(m_j, m_i)^2} \;; i \neq j \;\; i = 1, 2, \ldots, L, \tag{24}$$

$$DBI = \frac{1}{L} \sum_{k=1}^{L} R_k \;;\; R_k = max(R_{i,j}). \tag{25}$$

XB Index (XBI) was proposed in [25] and based on the ratio of the sum squares within clusters (SSW), and the sum squares between clusters (SSB). SSW is a measure of the compactness while SSB is a criterion for the separateness [4]. The formulations of XBI are as follows [4].

$$SSW = \sum_{k=1}^{L} \sum_{\forall s_p \in L_k} d(s_p, m_k)^2, \tag{26}$$

$$SSB = \sum_{k=1}^{L} n_k d(m_k, M)^2, \tag{27}$$

$$XBI = L \frac{SSW}{SSB}, \tag{28}$$

where M is the mean value of the dataset. In this study, DBI and XBI were used to evaluate the results of the image clustering and it should be noted that in order to get more accurate clustering solutions the DBI and XBI should be minimized. Image clustering was performed for the first group of the benchmark images by using the same optimization parameters with the study given in [4]. Therefore, the number of clusters was determined as 5, the number of populations and the algorithm trials were defined as 30, and the maximum number of the iterations was limited to 250 in all the trials. Since the parameters for the PSO, GA, ABC, and K-means were given in [4], they are not also given here. The obtained results of the image clustering for the first group of images are presented in Table 3. The table includes the results of d_{max}, d_{min}, J_e, DBI, and XBI that obtained by using the proposed IALO algorithm with the proposed objective function in this study and also the results of the ABC, PSO, and GA from the study by Ozturk et al. [4] for 30 runs. In the table, the best mean values related to the each image are written in bold. In Table 3, it can be clearly seen that the




42049

48025

Lena

Airplane

Figure 3. The first group of benchmark images.


35008

35010

43051

Figure 4. The second group of benchmark images.

IALO algorithm with the proposed objective function outperforms all the algorithms in terms of minimizing clustering validity index XBI and gets the best results for the Lena and 42049 images for minimizing DBI and J_e values. For the Lena and 48025 images, it also gets the best result in terms of minimizing the d_{max} value. The only parameter that IALO shows the worst results is d_{min} but it can be ignored because of the results for the cluster validity indexes DBI and XBI. According to these results, it can be said that IALO outperforms the other algorithms in terms of minimizing the cluster validity indexes and also gets very competitive results for minimizing the d_{max} and J_e. The second group of images was used to evaluate the clustering performance of the IALO algorithm with the proposed objective function against the ALO, and the CALO algorithms. Therefore, we performed the image clustering for the second group of the images by the ALO, IALO, and CALO algorithms by using the proposed objective function given in Eq. (21). In these experiments, the number of clusters was determined as 5, the number of populations as 30, and the maximum number of the iterations was limited to 100. And also, the CALO algorithm was used with the tent map and the x value was determined as 0.001 for the IALO algorithm. Finally, the algorithms were run 30 times and the statistical results of objective function values, d_{max}, d_{min}, J_e, DBI, and XBI were recorded. These results are given in Table 4. In the table, the best mean values related to each image are written in bold.

Table 3. The results obtained by IALO and the proposed objective function and those taken from [4].

Images	Alg.	J_e		d_{max}		d_{min}		DBI		XBI	
		mean	std	mean	std	mean	std	mean	std	mean	std
Lena	IALO	**8.5974**	0.0315	**8.8565**	0.0476	34.8374	0.3347	**0.153**	0.0011	**2.18E-5**	1.33E-7
	ABC[4]	9.736	0.108	10.676	0.133	39.955	0.967	0.1548	0.0044	0.2297	0.0047
	PSO[4]	9.793	0.198	10.691	0.224	**40.564**	0.503	0.1554	0.0045	0.2361	0.0083
	GA[4]	10.211	0.489	11.046	0.583	40.223	1.945	0.1629	0.0073	0.2585	0.0287
	K-means[4]	9.430	0.036	11.712	0.261	34.833	0.461	0.154	0.001	0.2266	5.9E-5
Airplane	IALO	9.737	0.0321	10.7497	0.1618	40.4416	0.6103	0.158	0.002	**1.14E-5**	2.91E-8
	ABC[4]	9.710	0.046	**10.734**	0.322	40.531	1.234	**0.1569**	0.0008	0.2533	0.0111
	PSO[4]	9.723	0.169	10.835	0.361	**41.329**	1.202	0.1571	0.0012	0.2566	0.0153
	GA[4]	10.083	0.502	11.435	0.659	39.697	2.302	0.166	0.0126	0.2535	0.0236
	K-means[4]	9.854	0.121	15.434	0.577	21.167	3.573	0.233	0.001	0.2048	0.0003
48025	IALO	10.7139	0.089	**11.6726**	0.2061	39.6405	1.6537	0.1561	0.0039	**4.33E-5**	6.40E-7
	ABC[4]	10.895	0.103	12.276	0.302	44.716	1.914	**0.1504**	0.0023	0.2781	0.0100
	PSO[4]	10.872	0.098	12.079	0.272	41.917	2.750	0.1552	0.0052	0.2699	0.0141
	GA[4]	11.277	0.701	13.082	1.274	**45.798**	4.553	0.1594	0.0089	0.2871	0.0294
	K-means[4]	**10.526**	0.021	12.512	0.232	31.350	0.249	0.1683	0.0012	0.2440	0.0002
42049	IALO	**8.6817**	0.022	9.938	0.098	38.5154	0.3026	**0.1542**	0.0018	**1.32E-5**	5.04E-8
	ABC[4]	8.699	0.053	**9.900**	0.158	38.024	0.766	0.1558	0.0025	0.1281	0.0025
	PSO[4]	8.744	0.139	10.026	0.254	**38.930**	0.863	0.1558	0.0022	0.1286	0.0036
	GA[4]	9.158	0.718	10.521	0.990	37.543	2.878	0.1655	0.0173	0.1486	0.0314
	K-means[4]	8.989	0.244	12.759	1.470	19.838	7.271	0.2088	0.2088	0.1298	0.0089

The obtained results given in Table 4 indicate that the proposed IALO algorithm outperforms the other two forms of the ALO algorithm in terms of minimizing the clustering validity indexes, DBI for all the three images and XBI for the 35008 and 35010 images. IALO was also superior in terms of minimizing the objective function value for the 35008 and 43051 images. In terms of the other parameters, it can be said that the IALO algorithm obtained very competitive results. The standard form of the ALO algorithm gets the second place by obtaining the best results for the d_{max} value for all the images and objective function value and d_{min} for the 35010 image and finally J_e and XBI for 43051 image. On the other hand, the CALO algorithm cannot show the best results for any comparison parameters except the d_{min} parameter for the 43051 image. As another performance evaluation, the clustering forms of the images of the second group are drawn according to the best results of the three algorithms and given in Figure 5.

In order to show the differences between the clustering results of the algorithms, we depicted the same regions on the clustered forms of the second group images with a blue circle. These circles show the same region on the figures and include some details according to the original image. It can be seen that the details of the original images can be seen in the resultant images of the ALO and IALO algorithms while they cannot be seen in the clustered images by the CALO algorithm. Thus, it can be concluded that the superiority of the ALO and the IALO algorithms on the CALO algorithm in image clustering can also be seen visually in addition to the results given in Table 4. On the other hand, it is very hard to see visually the differences between the performances of the ALO and IALO algorithms. However, the results of Table 4 indicates that the IALO

Table 4. The image clustering results of the ALO, IALO, and CALO[21] by the proposed objective function.

Images	Alg.	$Obj.val$	J_e	d_{max}	d_{min}	DBI	XBI
35008	IALO	**728,1738**	**11.3975**	14.9358	**43.3313**	**0.1515**	**4.357E-5**
		23.0159	*0.1639*	*0.3776*	*0.9187*	*0.0082*	*6.17E-7*
	CALO[21]	753.3415	11.5223	15.1945	43.0898	0.1531	4.393E-5
		12.3611	*0.1347*	*0.2935*	*1.586*	0.0071	*5.52E-7*
	ALO	736.2994	11.4493	**14.7926**	43.1965	0.1559	4.359E-5
		38.3905	*0.2374*	*0.5961*	*0.9604*	*0.0125*	*6.13E-7*
43051	IALO	**395.1102**	10.8623	15.0607	37.0075	**0.1405**	3.677E-5
		11.357	*0.1642*	*0.6034*	*2.0069*	*0.0061*	*1.30E-6*
	CALO[21]	410.1765	10.91244	15.0289	**38.0184**	0.1439	3.654E-5
		22.6691	*0.29674*	*1.12092*	*2.556*	*0.0061*	*2.233E-6*
	ALO	400.6007	**10.7991**	**14.6963**	37.7084	0.14216	**3.613E-5**
		21.3415	*0.2792*	*1.113*	*3.2374*	*0.0082*	*2.36E-6*
35010	IALO	561.3453	**10.7414**	12.4293	42.5042	**0.1664**	**2.079E-5**
		23.0585	*0.1476*	*0.2381*	*3.3127*	*0.0216*	*7.941E-7*
	CALO[21]	581.8477	10.8047	12.45159	42.42049	0.1735	2.114E-5
		28.6938	*0.2555*	*0.2691*	*4.4382*	*0.0416*	*1.62E-6*
	ALO	**559.1882**	10.7657	**12.2865**	**43.6025**	0.1734	2.116E-5
		39.72	*0.2249*	*0.1599*	*4.7838*	*0.0483*	*1.73E-6*

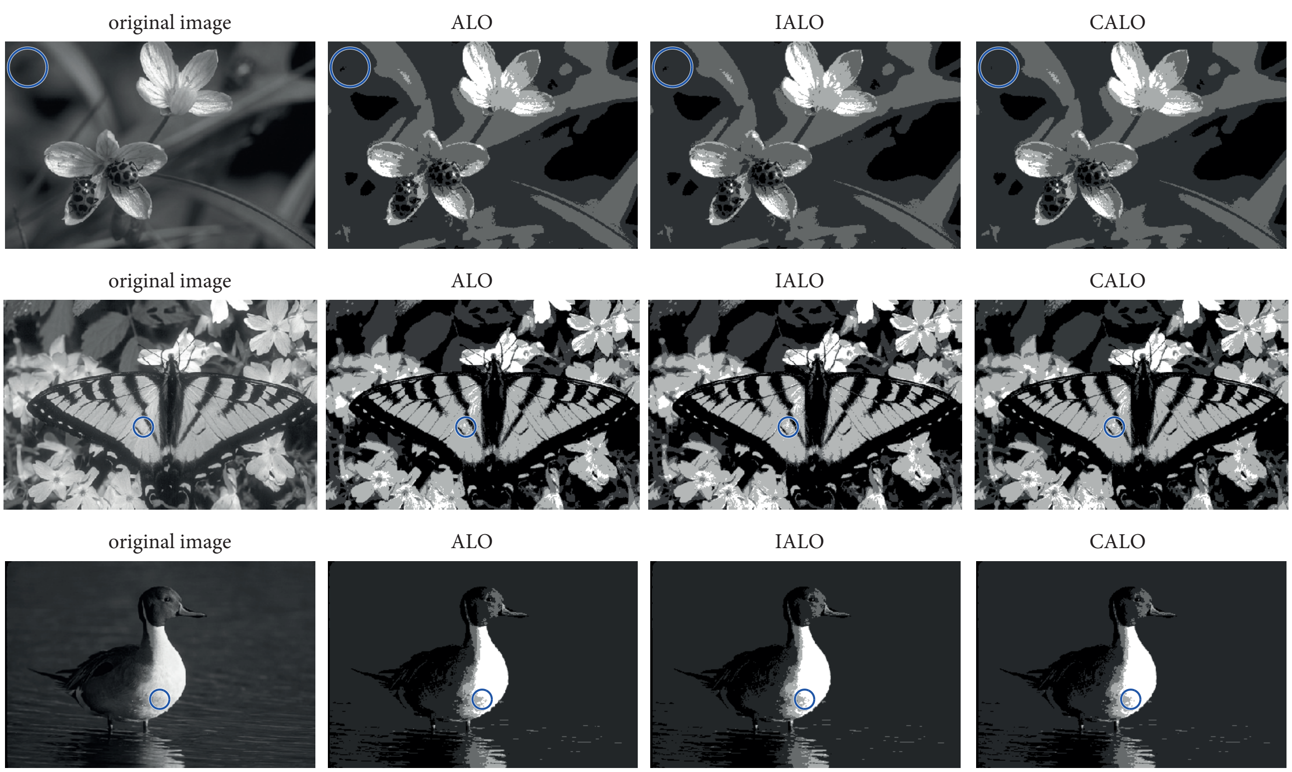

Figure 5. Clustered forms of the images of the second group by the ALO, IALO, and CALO [21] algorithms.

algorithm shown better performance than the ALO algorithm. Finally, it can be clearly said that the proposed IALO algorithm and the proposed objective function can be efficiently used for solving the image clustering problems.

5. Conclusion

The ALO algorithm was improved by using the inverse of the incomplete gamma function for its boundary decreasing procedure. Moreover, a recently proposed objective function for the image clustering was also improved and used with the proposed algorithm. Two groups of benchmark images were used to test the performances of the proposed methods. The performance of the improved ALO algorithm was compared with the results of the ABC, PSO, GA, and K-means from the literature and also compared with the ALO and a chaos-based ALO algorithm by using the proposed objective function for image clustering. The results showed that the proposed ALO algorithm with the proposed objective function outperforms the other algorithms in terms of minimizing the objective function and the cluster validity indexes DBI and XBI and also gets very competitive results in minimizing the quantization error and the intracluster distance while maximizing the intercluster distance.

References

[1] Jain AK, Murty MN, Flynn PJ. Data clustering: a review. ACM Computing Surveys 1999; 31 (3): 264-323.

[2] Azar AT, El-Said SA, Hassanien AE. Fuzzy and hard clustering analysis for thyroid disease. Computer Methods Programs in Biomedicine 2013; 111 (1): 1-16.

[3] Krishnasamy G, Kulkarni AJ, Paramesran R. A hybrid approach for data clustering based on modified cohort intelligence and K-means. Expert Systems with Applications 2014; 41 (13): 6009-6016.

[4] Ozturk C, Hancer E, Karaboga D. Improved clustering criterion for image clustering with artificial bee colony algorithm. Pattern Analysis Applications 2015; 18 (3): 587-599.

[5] MacQueen J. Some methods for classification and analysis of multivariate observations. In: 5th Berkeley Symposium on Mathematical Statistics and Probability; Berkeley, USA; 1967. pp. 281-297.

[6] Dunn JC. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. Journal of Cybernetics 1973; 3 (3): 32–57.

[7] Bezdek JC. Pattern Recognition with Fuzzy Objective Function Algorithms. New York, NY, USA: Plenum Pres, 1981.

[8] Omran M. Particle swarm optimization methods for pattern recognition and image processing. PhD, University of Pretoria, Environment and Information Technology, Hatfield, South Africa, 2004.

[9] Niknam T, Amiri B. An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. Applied Soft Computing 2010; 10 (1): 183-197.

[10] Biniaz A, Abbasi A. Unsupervised ACO: Applying FCM as a supervisor for ACO in medical image segmentation. Journal of Intelligent & Fuzzy Systems 2014; 27 (1): 407-417. doi:10.3233/IFS-131008

[11] Kumar Y, Sahoo G. A two-step artificial bee colony algorithm for clustering. Neural Computing and Applications 2017; 28 (3): 537-551.

[12] Wang W, Wang C, Cui X, Wang A. A Clustering algorithm combine the FCM algorithm with supervised learning normal mixture model. In: 19th International Conference on Pattern Recognition; Tampa, FL, USA; 2008. pp. 1-4.

[13] Toz M, Toz G. A novel image clustering algorithm based on DS and FCM. In: Medical Technologies National Conference (TIPTEKNO); Bodrum, Turkey; 2015. pp. 1-4.

[14] Maulik U, Bandyopadhyay S. Genetic algorithm-based clustering technique. Pattern Recognition 2000; 33 (9): 1455-1465. doi:10.1016/S0031-3203(99)00137-5

[15] Shelokar PS, Jayaraman VK, Kulkarni BD. An ant colony approach for clustering. Analytica Chimica Acta 2004; 509 (2) :187-195. doi:10.1016/j.aca.2003.12.032

[16] Omran M, Engelbrecht AP Salman, A. Particle swarm optimization method for image clustering. International Journal of Pattern Recognition and Artificial Intelligence 2005; 19 (3): 297-321. doi:10.1142/S0218001405004083

[17] Dowlatshahi MB, Nezamabadi-pour H. GGSA: A grouping gravitational search algorithm for data clustering. Engineering Applications of Artificial Intelligence 2014; 36: 114-121. doi:10.1016/j.engappai.2014.07.016

[18] Tang D, Dong S, He L, Jiang Y. Intrusive tumor growth inspired optimization algorithm for data clustering. Neural Computing and Applications 2016; 27(2): 349-374.

[19] Karaboga D, Ozturk C. A novel clustering approach: artificial bee colony (ABC) algorithm. Applied Soft Computing 2011; 11 (1): 652-657. doi:10.1016/j.asoc.2009.12.025

[20] Mirjalili S. The ant lion optimizer. Advances in Engineering Software 2015; 83: 80-98. doi:10.1016/j.advengsoft.2015.01.010

[21] Zawbaa HM, Emary E, Grosan C. Feature selection via chaotic Antlion optimization. PLoS ONE. 2016; 11 (3):e0150652, doi:10.1371/journal.pone.0150652

[22] Babers R, Ghali NI, Hassanien AE, Madbouly NM. Optimal community detection approach based on ant lion optimization. In: 2015 11th International Computer Engineering Conference (ICENCO); Cairo, Egypt; 2015. pp. 284-289.

[23] Chopra N, Mehta S. Multi-objective optimum generation scheduling using Ant Lion Optimization. In: Annual IEEE India Conference (INDICON); New Delhi, India; 2015. pp. 1-6.

[24] Davies DL, Bouldin DW. A cluster separation measure. IEEE Transactions on Pattern Analysis and Machine Intelligence 1979; PAMI-1 (2): 224-227.

[25] Zhao Q, Xu M, Fränti. Sum-of-squares based cluster validity index and significance analysis adaptive and natural computing algorithms. In: Kolehmainen M, Toivanen P, Beliczynski B (editors). Adaptive and Natural Computing Algorithms. Berlin, Germany: Springer, 2009, pp. 313-322.

[26] Martin D, Fowlkes C, Tal D, Malik J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Eighth IEEE International Conference on Computer Vision; Vancouver, BC, Canada; 2001. pp. 416-423.

[27] Dogan B., Ölmez T. A new metaheuristic for numerical function optimization: Vortex Search Algorithm, Information Sciences 2015; 293: 125-145. doi:10.1016/j.ins.2014.08.053

[28] Mirjalili S, Lewis A. The whale optimization algorithm. Advances in Engineering Software 2016; 95: 51-67.