Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-15668-4

Exponential entropy-based multilevel thresholding using enhanced barnacle mating optimization

Bibekananda Jena[1] · **Manoj Kumar Naik**[2] · **Rutuparna Panda**[3] · **Ajith Abraham**[4]

Received: 26 March 2021 / Revised: 21 March 2022 / Accepted: 22 April 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract
Multilevel Thresholding (MLT) is a prominent image segmentation research field that can effectively handle problems encountered while collecting meaningful information from a digital image. Most of the existing entropy-based Multilevel thresholding approaches use the logarithmic behaviour of Shannon's entropy, which does not exist for all possible points with appropriate bounded value. To evade this problem, an entropy-based on exponential information gain function is introduced as the fitness function in this paper to improve the thresholding accuracy. This research also proposes an enhanced Barnacle Mating optimization algorithm (EBMO) for obtaining appropriate threshold values by maximising the fitness function. The enhancement over basic Barnacle mating optimization is achieved by incorporating an additional Gaussian mutation strategy and a random flow towards the best solution steps with the original algorithm. The involvement of these additional steps helps the algorithm to prevent it to be stagnated at a local minimum by boosting its exploration capability. To validate the proposed optimization algorithm, it has been tested with a set of well-known benchmark functions and the CEC 2014 test suite. The results obtained in various tests are then compared with other standard and state-of-art algorithms with the help of quantitative analysis such as average, median, and standard deviation of the fitness values over several runs, qualitative analysis, such as search history, trajectory, and average fitness history and statistical analysis using Friedman Rank test and found superior to all. A more detailed analysis of the obtained results was also conducted using post hoc Bonferroni–Dunn and Holm test to observe how the proposed EBMO algorithm is significantly different from others. A comparison of the proposed exponential entropy (EE) based multilevel thresholding using EBMO (EBMO-EE) with other optimization algorithms also presented. Various performance measures such as peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), feature similarity index (FSIM), and Uniformity Measures (UM) obtained from different standard benchmark images of varying dimension are considered. It has been observed that there is an improvement of the thresholding accuracy, using EBMO, about 2% to 4% over others.

Keywords Exponential entropy · Optimal multilevel thresholding · Barnacle mating optimization

✉ Manoj Kumar Naik
naik.manoj.kumar@gmail.com

Extended author information available on the last page of the article

Published online: 18 May 2023

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

1 Introduction

Thresholding is a simple and effective approach for performing image segmentation by extracting homogenous sub-regions of the scene under consideration [23]. In a real-world application, it is used to separate the objects in the scene from the background. Based on the number of thresholds used for partitioning an image into various groups, thresholding processes are classified into bi-level and multilevel thresholding. In bi-level thresholding, a single threshold value is required to obtain from the scene which can separate the required object from the background. In the multilevel thresholding process, two or more threshold values are used to divide the gray level distribution of the image in hand into distinct groups and each group is assigned with a single intensity value. As in most of the applications, color images are used because of their more information content capability, the requirement of thresholding is highly essential in these cases to increase the processing speed by performing segmentation before any high-level processing. For the real-life segmentation applications, the best option is opting for multilevel thresholding over bi-level to extract useful information from the scene.

There are several global thresholding approaches available in the literature [5, 10, 18, 19, 32, 37, 59] to perform the segmentation. Histogram-based thresholding [68, 71] gained its popularity because of its simple and efficient way of obtaining the optimal threshold from the gray level distribution of the pixels in the image. The most common histogram-based thresholding methods use a generalization of Shannon entropy as the fitness function for threshold calculation such as Kapur's entropy [35], Renyi's entropy [66], Tsallis's entropy [3, 47], Cross Entropy [30], and Masi Entropy [73]. The segmentation process is then further improved by including the spatial correlation of the pixels into consideration with 2D variants of the above entropies calculated from a two-dimensional histogram [55, 70]. However, the use of logarithmic gain function by most of these entropies suffers from a drawback of undefined gain in information at points of highly probable or highly unlikely with appropriate bounded values [58]. For example, if the probability $'p'$ of an event within the search space is zero, the logarithmic entropy $E = log_2(1/(p=0)) = \infty$ and if the probability $'p'$ becomes one the Entropy will be $E = log_2(1/(p=1)) = 0$. In practice the gain in information value from an event must be defined within two finite limits irrespective of the chances of occurrence. For example, the gain information becomes maximum when all the pixel values are taken into consideration irrespective of the image contents. To resolve this issue, an exponential entropy (EE) [58] based multilevel thresholding method is proposed in this paper.

In the literature, generally, two types of approaches are available to perform thresholding operations for image segmentation: parametric and non-parametric approaches. The parametric approach of thresholding defines each class by estimating the parameters of the given probability density process, which is computationally expensive. Whereas in non-parametric approaches, the optimal threshold values are selected by optimizing a given fitness function such as: between class variance or entropy measures. Though the non-parametric approaches are efficient and popular, time complexity becomes high when the number of thresholds increases. Therefore Nature-inspired algorithms being used for multilevel thresholding operations over the past few decades. In the past few years, several metaheuristic algorithms have been proposed by researchers and prove their capability by solving complex engineering problems. Table 1 is a quick review of the literature on nature-inspired algorithms.

Researchers frequently employ soft computing, a subfield of Artificial Intelligence (AI), to address the computational time complexity challenge in multilevel thresholding.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 1 Brief review of nature-inspired algorithms

Nature-Inspired Algorithm	Inspired by / based on
Genetic algorithm (GA) [27]	Evolution
Particle swarm optimization (PSO) [15]	Intelligent social behaviour of flocks of birds
Differential evolution (DE) [75]	Darwin's theory of evolution
Bacteria foraging optimization (BFO) [12]	*Escherichia coli* bacteria's communal foraging behaviour
Teaching–Learning-Based Optimization (TLBO) [64]	the impact of a teacher's influence on students
Krill herd optimization (KHO) [21]	A model of individual krill herding behaviour
Gravitational Search Algorithm (GSA) [65]	The law of gravitation and mass interactions.
Cuckoo search algorithm (CS) [22]	Some cuckoo species' obligate brood parasitism, in which they deposit their eggs in the nests of other kinds of host birds found in various locations
Firefly optimization (FF) [34]	Invertebrates such as glowworms and fireflies that produce flashing illumination patterns
Grey wolf optimizer (GWO) [49]	The natural leadership structure and hunting mechanism of grey wolves
Whale optimization algorithm (WOA) [48]	Hunting strategy of humpback whales
Crow search algorithm (CSA) [6]	Crows' intelligence behaviour of storing surplus food in hidden places and retrieving it when it is needed
Grasshopper optimization algorithm (GOA) [69]	The behaviour of grasshopper swarms in both nymph and adulthood stages.
Salp Swarm Algorithm (SS) [50]	The swarming behaviour of salps in oceans when navigating and hunting
Volleyball Premier League Algorithm (VPL) [52]	Volleyball match's coaching procedure
Emperor penguin optimizer (EPO) [14]	Emperor penguins' communal huddling behaviour to survive in water
Squirrel search algorithm (SSA) [31]	Dynamic hunting behaviour of southern flying squirrels, as well as their efficient gliding mode of mobility
History-Based Adaptive Differential Evolution with Linear population size reduction algorithm (L-SHADE) [61]	An adaptive DE strategy that combines linear population size reduction with success-history-based parameter adaption.
Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [11]	A second-order approach for iteratively estimating a positive definite matrix, specifically a covariance matrix on convex-quadratic functions, is closely related to the inverse Hessian method.
Harris hawk optimization (HHO) [25]	Harris' hawks' cooperative behaviour and attacking manner
Equilibrium optimization (EO) [16]	Models of control volume mass balance used to estimate both dynamic and equilibrium phases
Sailfish optimizer (SFO) [72]	Group hunting attack-alternation tactic
Manta-ray foraging optimization (MRFO) [85]	Foraging tactics used by manta rays include chain foraging, cyclone foraging, and somersault foraging.
Tunicate swarm algorithm (TSA) [36]	Truncicates swarm activities and jer propulsion during its navigation and foraging phase
Slime mould algorithm (SMA) [41]	Imitates slime mould behaviour and morphology during foraging

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

The metaheuristic algorithms which are successfully applied to the multilevel algorithm with different fitness functions are discussed hereafter. In the article [61], the authors maximized the Otsu's between class variance [56] with the help of a Genetic algorithm (GA) for obtaining optimal thresholds at a different level of thresholding [80]. Particle swarm optimization (PSO) is used to obtain the optimal threshold values in the paper [9]. The paper shows a clear demonstration of reduction computation time with the use of the metaheuristic algorithm. A lip area extraction from the face region with the help of Bacteria foraging optimization (BFO) based on thresholding is discussed in the article [8]. The article illustrates that employing BFO for lip portion extraction has clinical implications. A maximum entropy-based multilevel thresholding approach using an Artificial bee colony algorithm is discussed in the paper [29]. The authors of the article used the method on several types of benchmark photos and claimed to get near-optimal thresholds in the majority of cases with minimal computation time. A 2D histogram and maximum Tsallis entropy-based multilevel thresholding approach using Differential evolution (DE) algorithm for optimizing the fitness function is presented in the paper [70].

To overcome the increase in computational complexity with an increase in the number of thresholds, the authors of the paper [40] used the Grey wolf optimization algorithm and found it computationally efficient over PSO and BFO based approaches. A multilevel thresholding approach using Otsu's between class variance as a fitness function is proposed in the paper [63]. The paper used the Firefly algorithm (FFA) to maximize the fitness function for generating thresholded results. The paper produced convincing thresholded results with less computation time. An application of multilevel thresholding in medical image analysis is presented in the paper [38]. The paper used real-time MR/CT images for thresholding by maximizing Otsu's between-class variance and Kapur's entropy with the help of the Crow search algorithm (CSA). The requirement of fewer parameters of CSA makes it suitable for the application. In paper [67], a multilevel thresholding strategy based on the Harris Hawks Optimization (HHO) algorithm is proposed, using minimum cross-entropy as the fitness function. The approach is put to the test on a set of benchmark images, the Berkeley segmentation database, and digital mammography images. The HHO algorithm is found superior over most of the well-known algorithms in terms of the accuracy of segmentation results. The efficacy of the Equilibrium Optimizer (EO) is explored in the paper [2] by applying it to the thresholding of images at low as well as the high level of thresholding. The results of the approach show a significant improvement of EO based methods over other state-of-art methods. The L-SHADE technique is used to find the best set of threshold values for separating clusters of pixels in the article [26]. The thresholded images were found impressive and the authors claimed that they can be used for practical applications. A basic version of the algorithm isn't always ideal for all kinds of problems. Therefore, along with these original algorithms, some of their hybrid and improved versions have also been developed by many researchers. Few of them are also applied to multilevel thresholding a problem with a significant improvement over the original version.

In Modified discrete grey wolf optimizer (MDGWO) [40], the authors discretize the grey wolf optimizer (GWO) before providing a unique attack strategy that replaces the original algorithm's search formula for an optimal solution with the weight coefficient. The algorithm performed produced better segmentation accuracy using the multilevel thresholding approach. An improved grey wolf optimizer is presented in the article (IGWO) [46] by integrating differential evolution (DE) strategy with GWO algorithm and Otsu algorithm. The experimental results show a complete dominance of the improved version over the original version of GWO. The HHO algorithm is also modified to a leader Harris hawks optimization (LHHO) [55]. The goal of this study was to improve exploration competence

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

by using an adaptive perching method throughout the exploration phase, as well as a mutation stage with the leader Harris hawk in each generation. The paper used 2D as well as 1D Masi entropy for generating thresholded images. By integrating adaptive dispersion decision-making for weaker search agents, the authors of the paper [83] introduced an adaptive equilibrium optimizer (AEO). Before applying AEO to the multilevel thresholding task, it has been tested over several benchmark functions and found superior over the basic EO algorithm. A brain image analysis using hybrid adaptive cuckoo search-squirrel search algorithm (ACS-SS) based multilevel thresholding is presented in the paper [4]. The authors hybridized the searching pattern of adaptive cuckoo search algorithm (ACS) [53] with Squirrel search algorithm (SSA) for maximizing the edge magnitude information derived from gray-level co-occurrence matrix (GLCM) [60]. The paper shows a significant improvement in the thresholding accuracy of hybridized method over basic ACS and SSA. A new color image segmentation using modified grasshopper algorithm (MGA) [43] based multilevel thresholding is proposed by the authors recently. The authors modified the exploration skill of the GOA algorithm by incorporating the Levy flight algorithm with it. The modification of GOA algorithm was found significantly different concerning segmentation performance over GOA.

In the Improved cuckoo search algorithm (ICS) [77], the authors used the fitness value of each iteration for selecting the adaptive step size without using Levy flight. The technique not only reduces the computation time but also improve the performance of the algorithm. Multilevel thresholding of breast thermal images is presented in the paper [62] using improved particle swarm optimization (IPSO). The authors of this paper used PSO algorithm with a modified updating rule for velocity which shows an improvement of thresholding accuracy over traditional PSO. Learning enthusiasm-based teaching–learning-based optimization (LebTLBO) [74] mimics the behaviours of the teaching and learning process in a classroom and calculates the likelihood of the learner (student) receiving the quantity of knowledge desired from the educator. The algorithm was also used by the authors for segmentation of the image and found a suitable method when combine with Kapur's and Tsallis entropy. A Hybrid differential evolution (HDE) for multilevel thresholding is introduced in the article [51]. The hybridization is accomplished by incorporating a Cuckoo Search-inspired reset mechanism into the differential evolution evolutionary cycle. The paper's findings demonstrate the superiority of hybridization in thresholding over basic algorithm-based approaches such as DE. The authors of the paper [78] proposed a Hybrid gravitational search algorithm with a genetic algorithm (HGSA-GA). The paper rescued the chances of early convergence of GSA algorithm by adapting the roulette selection and discrete mutation operators of GA. The improvement in population diversity achieved by this hybridization helps the algorithm to improve the performance in multilevel thresholding. The authors of the Emperor penguin and Salp swarm algorithm (ESA) [13] introduced a novel hybrid algorithm that mimicked the emperor penguin optimizer and salp swarm algorithm's huddling and swarm tendencies. The algorithm was tested over 53 benchmark functions and found superior over SS and EPO algorithms. In Improved volleyball premier league algorithm (IVPA) [1] the authors used the WOA algorithm's searching pattern for improvement of exploration skill of VPL algorithm before applying it to multilevel thresholding problem. The IVPA algorithm produced high-quality thresholded results when tested with standard as well as medical images. However, the role of optimization in the field of digital image processing is not limited to thresholding. Nowadays, optimization algorithms are being used widely in image analysis and computer vision applications. To recognize gestures in the Human-computer interaction domain, a crow

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

search-based convolution neural networks model is presented in the article [20]. In article [79], the authors applied the Antlion optimization (ALO) algorithm in the Deep neural network (DNN) model to assure optimal hyper-parameter selection for the categorization of a multimodal stroke dataset that is unbalanced in a short amount of time.

Barnacle Mating optimization [76] is one of the recently developed evolutionary algorithms based on the mating behaviour of barnacles in nature for solving numerical optimization problems. Barnacles are famous for their unique long penises, which are nearly seven to eight times their body sizes. The unique mating process using the variable penises length with its neighbours makes it different from other species on the earth. BMO's notable performance in comparison to other well-known optimization algorithms draws our attention to investigate its strength and weakness. According to the No Free Lunch (NFL) [82] theorem, which stated that performance of any optimization algorithm may not be found satisfactory for all types of optimization-related issues. Therefore, the possibility of upgrading search ability towards the optimal result of any algorithm is always available. After an in-depth investigation of the BMO, it has been observed that the existing mutation strategy makes the algorithm exploration capability limited and fails to reach an optimal solution in some of the cases during tests with different types of problems. To enhance the exploration for improving the searching efficiency two additional steps: a Gaussian mutation [7] and a random movement towards the best solution [55], are included with BMO. The supplement of these new strategies leads to the development of the proposed Enhanced Barnacle Mating Optimization (EBMO). To validate the search capability of the proposed EBMO, it has been tested with a set of classical benchmark functions and IEEE CEC 2014 test suite. The comparison of EBMO with BMO and other state-of-the-art algorithms such as MRFO, SMA, EO, HHO, SSA, L-SHADE, TSA, and CMA-ES through various analyses have been carried out and the overall performance of EBMO is found better than all. To analyse the performance of EBMO in multilevel thresholding application, it also has been tested with low dimensional standard colour images and high dimensional multispectral images collected from Landsat image gallery [39] by maximizing the exponential entropy. The performance of proposed exponential entropy (EE) based multilevel thresholding using EBMO (EBMO-EE) is compared with the above optimization algorithm and found superior to all.

The highlights of this work are as follows:

I. An Enhanced Barnacle Mating optimization (EBMO) algorithm is proposed by including an exploration boosting mechanism with the help of Gaussian mutation and random movement towards the best strategies. The EBMO's overall advantage is demonstrated by a quantitative and qualitative study of the test results across a set of classical benchmark functions and the IEEE CEC 2014 test suite. A probabilistic entropy of exponential gain function has been used to the multilevel thresholding problem
II. The concept of Exponential entropy-based thresholding is extended to multilevel thresholding by proposing an exponential entropy-based multilevel thresholding approach using EBMO(EBMO-EE).
III. using standard colour images and high dimensional images of Landsat datasets. The comparative results reveal that EBMO outperforms other state-of-the-art algorithms.

The remaining part of this article is as follows. A brief review of Exponential entropy and Barnacle Mating Optimization algorithm is presented in Section 2. An extension of exponential entropy for multilevel thresholding is discussed in Section 3. Section 4 deals

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

with the proposed Enhanced Barnacle Mating optimization algorithm concept and its performance evaluation on benchmark functions. An exponential entropy-based multilevel thresholding is presented in Section 5. The experimental results and discussions of the proposed multilevel thresholding are presented in Section 6. Finally, in Section 7, the paper came to a close with a closing remark.

2 Preliminaries

2.1 Exponential entropy (EE)

Unlike the logarithmic behaviour of Shannon's entropy, the information gain function used here is exponential. The exponential gain function makes it possible to be defined at all possible points with a bounded value. For the case of highly probable or highly unlike event having probability $'p'$, the logarithmic gain function $\log(1-p)$ produces a value that is not within the desired limit. To circumvent this issue, the gain function used here is an exponential function of $(1-p)$ and the entropy derived from this concept of a n-state system is defined as

$$H_E = \sum_{i=1}^{n} p_i e^{(1-p_i)} \tag{1}$$

where p_i is the probability of occurrence of i^{th} (event).

The desirable properties [58] which make it suitable for segmentation applications are:

i. $e^{(1-p_i)}$ can be defined for all possible points in the search space
ii. $\lim_{p_i \to 0} e^{(1-p_i)} = h_1 > 0$ and finite
iii. $\lim_{p_i \to 1} e^{(1-p_i)} = h_2 > 0$ and finite
iv. $h_1 > h_2$
v. $e^{(1-p_i)}$ decreases exponentially with an increase in the probability
vi. H_E is continuous over the range of $0 \le p_i \le 1$
vii. Maximum value of H_E will be obtained when each event has equal probability.

2.2 Barnacle mating optimization

Barnacles are micro-organisms present on the earth since the time of the Jurassic. The unique mating process of these Barnacle for their long penises of variable sizes attract the attention of the researcher. Barnacle Mating optimization (BMO) [76] is one of the results of their research. In the process of mating with neighbours within their penis lengths, the variable penis sizes play a vital role in determining the size of the mating group. BMO is an evolutionary algorithm inspired by the mating behaviour of these Barnacles. The optimization in BMO is performed through three major stages: initialization, selection, and reproduction. The role of each stage is discussed below.

In the initialization stage, a population is formed by a random set of barnacles. Each barnacle is a vector and represented by a certain number of control variables depending upon the problem at hand. The values of each control variable must lie within the defined upper and

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

lower bound. In the Selection process, BMO adopted a random selection process for offspring generation, but the selection of parents is restricted to the barnacle's penis's length '*pl*'. This mechanism makes the BMO different from other evolutionary algorithms like GA and DE. During the mating process, each barnacle can receive sperms as well as contribute sperms to only one barnacle at a time. There is also a possibility of a sperm cast process when the selection of parents is done beyond the penis's length of a barnacle. BMO also has a slightly different reproduction strategy than other evolutionary algorithms. As there is no specific mathematic formula to derive the reproduction mechanism of barnacle, BMO adopted the Hardy–Weinberg principle [9] for the generation of offspring by performing exploitation and exploration in two different conditions. The exploitation process begins when the barnacles to be meted are within the predefined range of the maximum penis length. The sperm cast mechanism in BMO [76] is treated as the exploration process, which is occurred when selections of barnacles for mating exceed the predefined maximum penis length. At the end of each iteration, the best barnacle is identified and the fittest barnacles from parents and offspring are selected for the next generation.

3 Exponential entropy for multilevel thresholding

The exponential entropy satisfies all the properties of Shannon's entropy except the additive properties which do not have any remarkable impact on the image [58] because in an image the neighbouring pixel values are normally dependent on each other. It is also successfully applied to bi-level thresholding for classifying the object in the scenes from the background [57, 58]. In this section, the study of the thresholding scheme using exponential entropy is extended to multilevel thresholding.

Let's consider an *n*-bits image $f(x,y)$ of size M×N.The range of gray level is $[0, L-1]$, where $L = 2^n - 1$. The probability of i^{th} gray level in the image can be expressed as

$$p_i = \frac{n_i}{MN}, \quad i \in [1, 2, 3, \ldots L-1] \tag{2}$$

Where, n_i represent the pixels count of i^{th} gray level.

In multilevel thresholding, more than two threshold values are used to divide an image into different homogeneous regions used for a certain application. To split the above image into $k+1$ distinct regions depending on their intensity value, k numbers of thresholds are required which may be demonstrated using a basic thresholding rule as given below.

$$g(x,y) = \begin{cases} R_1 \leftarrow l, & if\ 0 \le l < T_1 \\ R_2 \leftarrow l, & if\ T_1 \le l < T_2 \\ R_3 \leftarrow l, & if\ T_2 \le l < T_3 \\ & . \\ & . \\ & . \\ & . \\ R_{k+1} \leftarrow l, & if\ T_k \le l < L-1 \end{cases} \tag{3}$$

Where $'l'$ represents the pixel intensity, $g(x,y)$ indicates the segmented image and R_j indicates the j^{th} distinct regions of $g(x,y)$ including the foreground, background, and intermediates sections. $\{T_1, T_2 \ldots\ldots T_k\}$ are the k numbers of selected threshold values used for segmentation.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

To obtain these threshold values, it is required to maximize the information content of each distinct region. As entropy is nothing but the measure of information content in a source, in this work the optimal threshold values are selected by maximizing the exponential entropy calculated from each region. The exponential entropy of background $'H_E^1'$, foreground $'H_E^{k+1}'$ and j^{th} intermediate region H_E^j for multilevel thresholding can be expressed as:

$$H_E^1 = \sum_{i=0}^{T_1-1} \frac{p_i}{\omega_1} e^{\left(1-\frac{p_i}{\omega_1}\right)} \tag{4}$$

$$H_E^{k+1} = \sum_{i=T_k}^{L-1} \frac{p_i}{\omega_{k+1}} e^{\left(1-\frac{p_i}{\omega_{k+1}}\right)} \tag{5}$$

$$H_E^j = \sum_{i=T_{j-1}}^{T_j-1} \frac{p_i}{\omega_j} e^{\left(1-\frac{p_i}{\omega_j}\right)} \tag{6}$$

where, ω is the probability of each segmented region and expressed as

$$\omega_1 = \sum_{i=0}^{T_1-1} p_i, \omega_2 = \sum_{i=T_1}^{T_2-1} p_i, \dots \omega_j = \sum_{i=T_j}^{T_j-1} p_i, \dots .\omega_{k+1} = \sum_{i=T_k}^{L-1} p_i \tag{7}$$

The objective function for multilevel thresholding using exponential entropy obtained from all regions for getting optimal thresholds $\{T_1{}^*, T_2{}^* \dots\dots T_k{}^*\}$ can be formulated as:

$$\{T_1{}^*, T_2{}^* \dots\dots T_k{}^*\} = \underset{0 \le T_1, T_2 \dots T_k \le L-1}{\arg\max} \left(H_E^1 + H_E^2 + H_E^3 + \dots H_E^{k+1}\right) \tag{8}$$

Equation (8) is a maximization problem and can be solved with a suitable optimization algorithm to obtain the optimal threshold values.

4 Proposed enhanced barnacle mating optimization (EBMO)

After a detailed study of the Barnacle Mating Optimization (BMO) [76], it has been observed that there is a lack of diversity in the population which affects the accuracy of the algorithm in reaching the optimal solution. This leads to the development of the Enhanced Barnacle Mating Optimization (EBMO) algorithm by enhancing the exploration capability of BMO. The proposed EBMO algorithm includes two different strategies to perform this task: the first one is the use of Gaussian mutation [7]which allow the Barnacles to explore maximum regions in the search space and the second strategy is to follow a random movement approach towards the current best solution [55] which provides desired exploration towards the best candidate in each generation. The impact of these double mutation strategies in the algorithm is the fast convergence without being trapped in any local minima.

4.1 Mathematical formulation of EBMO

This section provides a detailed mathematical formation of the proposed EBMO algorithm with an additional mutational strategy. The optimization process in EBMO consists of four different stages in each generation. After the end of each generation, the newly formed

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

barnacles in the population are sorted according to their fitness value and the barnacle at top of the list is identified as the best solution obtained so far. The best solution obtained when the termination criteria are met is declared as an optimal solution. The detailed functioning of all stages is given below.

4.1.1 Initialization

Let $'N'$ be the number barnacles participated in the mating process for obtaining the optimal solution and each barnacle is a $'n'$ dimensional control vector, $X_i = \left[x_i^1, x_i^2, \ldots x_i^n\right]$, $i=1$, $2 \ldots N$. The control variables of each vector must lie between the upper $'ub'$ and lower bound $'lb'$ of the problem in hand. The best solution at each generation is represented by X_{best}.

4.1.2 Selection process

The proposed EBMO also adopted the same random parents' selection process as that of BMO by considering the barnacle's penis's length $'pl'$. The barnacles in the populations are arranged in a random manner of different groups, one is referred to as $'Dad\_barnacle'$ and another one is $'Mom\_barnacle'$. At one time, each barnacle from a group of $'Mom\_barnacle'$ can be fertilized by one from $'Dad\_barnacle'$.

The above process can be modelled mathematically by arranging the barnacles in the population based on their fitness values from best to worst. The barnacle located at the top of this list is identified as the best solution obtained so far and named barnacle#1. Similarly, barnacle#N represents the worst solution of the population and is placed at the bottom of the list. If the maximum penis length of barnacles is 'pl_{max}' times of their body, then a barnacle in the list can produce offspring by mating with another barnacle placed at a distance, not more than 'pl_{max}' from its location. For example, if 'pl_{max}'=7, barnacle#1 can mate with anyone from barnacle#2 - barnacle#7. The sperm cast process will be initiated if barnacle#1 selects any other barnacle other than above, i.e., barnacle#8 - barnacle#N. Sperm cast processes also provide the required exploration for the algorithm. Once the barnacles are ranked as per their fitness values, the random selection of parents can be expressed by the following expressions:

$$Dad\_barnacle = randperm\ (N) \tag{9}$$

$$Mom\_barnacle = randperm\ (N) \tag{10}$$

4.1.3 Reproduction

Like BMO, the proposed EBMO also adopted Hardy–Weinberg principle [24] to produce offspring in the first stage of the reproduction process. In the second stage of reproduction, the best candidates among the parents and offspring are selected to generate new offspring using Gaussian mutation and random movement towards the current best solution strategies.

Offspring generation based on hardy–Weinberg principle The reproduction process of BMO was developed by following the Hardy–Weinberg principle used here for offspring

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

generation. When the parent's barnacles to be mated are within the range of the maximum penis length 'pl_{max}' a normal mating process occurred which is referred to as exploitation and expressed as

$$x_i^{n\_new} = p\, x^n_{Dad\_barnacle} + (1-p)\, x^n_{Mom\_barnacle} \tag{11}$$

where, p is a random number normally distributed between $[0, 1]$, and x_i^n represent the n^{th} control variable of i^{th} barnacle. From Eq. (11) it can be observed that the newly formed offspring inherits the $p\%$ behaviour of the $Dad\_barnacle$ and $(1-p)\%$ behaviour of the $Mom\_barnacle$.

If the selected parents are not found suitable because of exceeding the maximum penis length 'pl_{max}', the algorithm generates offspring using the sperm cast mechanism as modelled below:

$$x_i^{n\_new} = \text{rand} \times x^n_{Mom\_barnacle} \tag{12}$$

where '$rand$' indicates a random number distributed uniformly in the range $[0, 1]$. Equation (12) shows a very simple mating process of offspring generation. This process resembles the offspring generation by a $Mom\_barnacle$ by receiving sperms from the water that has been released by any random barnacle.

Gaussian mutation and random movement towards best barnacle In EBMO, the new mutation strategies are applied to selected barnacles after the completion of a newly formed population matrix by BMO in each iteration. After sorting the barnacles in the population of 'N' barnacles according to their fitness value, the best $\frac{1}{3}rd$ of the population from the top are selected to generate two new offspring groups, each of size $\frac{N}{3}$ with the help of gaussian mutation and random movement schemes separately. These two new offspring groups are then used to replace the remaining $\frac{2N}{3}$ of the population.

The Gaussian Mutation [7]can be performed by applying a Gaussian distribution to a barnacle vector X_i in the population, as follows:

$$X_i^{gm}(itr) = X_i(itr).(\mu + \sigma.N(0, 1)\}\Big) \tag{13}$$

where $N(0, 1)$ represent a random vector of gaussian distribution, 'μ' and 'σ' represent its mean and standard deviation and X_i^{gm} is the mutated vector. In this work, the mean and standard deviation are fixed to 0 and 1 respectively for every generation 'itr'.

The random movement towards the best solution obtained so far concept in [55] also attracts our attention toward it which helps the barnacle to explore towards the best solution. This random movement is expressed mathematically as:

$$X_i^{rm}(itr) = \left(X_{best}(itr) - X_{mean}(itr)\right) - r_1\left(lb + r_2(ub - lb)\right) \tag{14}$$

where X_i^{rm} is the newly formed offspring by randomly following the best, X_{mean} indicates the mean of the population at every generation and r_1 *and* r_2 are random numbers uniformly distributed in the range $[0, 1]$.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

4.2 Pseudocode of EBMO

Input: The population matrix $N \times n$ for a n-dimensional problem, upper and lower bound $'ub'$ and $'lb'$ of each control variable, maximum penis length 'pl_{max}' and maximum iteration itr_{max}. Set $itr = 1$

Output: The best barnacle vector with its fitness value

Initialization: Randomly initialize the barnacles in the population and identify the best solution X_{best} by evaluating the fitness of each barnacle.

While $(itr < itr_{max})$

(i) Evaluate the fitness function of $f(X_i)$ for all barnacles in the population

(ii) Arrange the barnacles in the population matrix as per their fitness values from best to worst

(iii) Perform the random selection of Dad and Mom barnacles as per Eq. (9) and (10) for mating

(iv) *for* each barnacle in Dad

  (a) select the corresponding barnacle from Mom

  (b) *if* the selection of parents from Dad and Mom are within the range of pl_{max}

    Generate offspring using Eq. (11)

  else

    Generate offspring using Eq. (12)

  end if

end for

(iv) Sort the barnacles according to their fitness value from best to worst

(v) Apply Gaussian mutation and random movement schemes to the best $\frac{N}{3}$ of the population separately to generate $\frac{2N}{3}$ offspring using Eqs. (13) and (14) and replace this with remaining $\frac{2N}{3}$ of the population

(vi) Update the best solution X_{best}

(vii) $itr = itr + 1$

end while

Return X_{best}

4.3 Performance evaluation of EBMO algorithm

To examine the effectiveness of the proposed EBMO, several performance evaluations tests have been carried out on a set of 52 well-known benchmark functions including 22 classical test functions ($f_1 - f_{22}$) [55] and 30 modern test functions ($f_{23} - f_{52}$) from CEC 2014 test suit [42] as given in Appendix Tables 9, 10 and 11. From the above test functions ($f_1 - f_7$) having unimodal characteristics with a unique global minimum used to validate the ability of exploitation of optimization algorithms. Whereas test functions ($f_8 - f_{12}$) and ($f_{13} - f_{22}$) are multimodal with scalable and fixed dimensions respectively used to observe exploration ability because of the presence of many local minima in it. However, real-world problems have no defined shapes or a combination of both unimodal and multimodal functions. The composite functions ($f_{23} - f_{52}$) of IEEE CEC 2014 test suit mimic above characteristics of real-world problems by expanding, shifting, rotating, and hybridizing different types of unimodal and multimodal functions. Tests with these functions make an algorithm ready to handle real challenges in the world. The results of the tests are investigated on various qualitative, quantitative, and statistical analyses. A detailed study of search history, the trajectory of the first barnacle, average fitness history, optimization history is included in qualitative analysis. The quantitative analysis is based on inspecting the median, average, and standard deviation values over many independent runs. Whereas Non-parametric tests such as the Wilcoxon signed-rank test at a significance level of 5% and Friedman mean test along with post

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

hoc statistical analyses using Holm and Bonferroni–Dunn test also conducted to notice the substantial deference between EBMO and other algorithms. As some of the algorithms need initial parameter settings for their operation, Table 2 shows the setting of these parameters.

To validate the efficacy of the proposed EBMO, the test results obtained from various test functions have been compared with various standard and state-of-art techniques such as BMO, MRFO, SMA, EO, HHO, SSA, L-SHADE, CMA-ES, and TSA. For all 52 benchmark functions, the population size $'N'$ and maximum iteration *itrmx* of EBMO are set to 30 and 500 respectively which is equivalent to 15,000 function evaluations. To provide a fair comparison, the above parameters of BMO, MRFO, SMA, EO, HHO, TSA, and SSA also remain the same. As the population size of CMA-ES and L-SHADE are dependent upon the number of control variables, the results obtained after 15,000 evaluations have been used here for comparison.

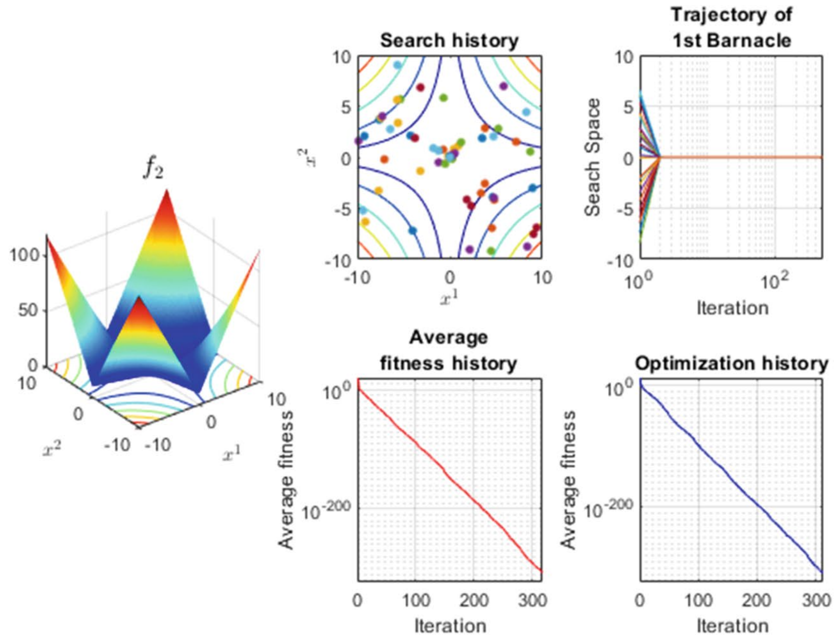
4.3.1 Qualitative analysis of EBMO

The efficiency of the proposed EBMO algorithm is demonstrated in Fig. 1 with the help of four qualitative metrics: search history, the trajectory of the first barnacle, average fitness history, and optimization history. These metrics are evaluated by solving three classical (f_2, f_{12} *and* f_{17}) and three modern complex test functions (f_{25}, f_{35} *and* f_{46}) with 15,000 evaluations to show the searching pattern of barnacles in the search space. Though these functions are defined in a high-dimensional space, the 2-dimensional view presented in Fig. 1 can provide an overview of the field topology. The search history of the algorithm is considered here as the first qualitative metric which comprises the barnacle's concentration from the beginning of the evaluation to the end. It provides a clear understanding of the searching pattern followed by the barnacles in each search space. It can be observed from the Search history plot that, the barnacles can explore every corner of the search space at the beginning and can converge at global minima for classical unimodal or multimodal functions. For the complex test functions taken from CEC 2014 test suit in Fig. 1, the barnacles try to reach the global minima by concentrating around it at end of the iteration.

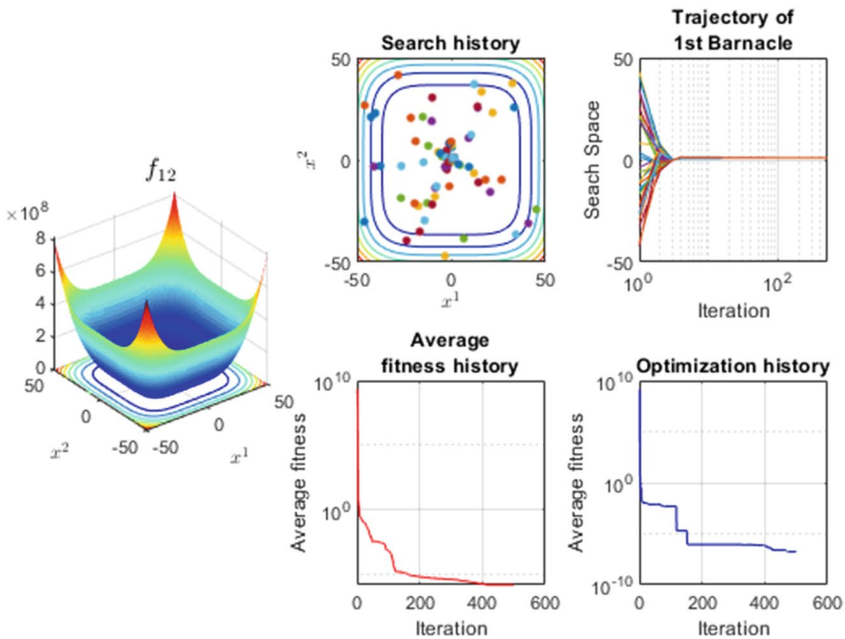
Table 2 Parameter setting of different Optimization Algorithms

Algorithm	Parameter(s)	Value
EBMO	Maximum penis length (pl_{max})	7
BMO	Maximum penis length (pl_{max})	7
MRFO	Somersault factor (S)	2
SMA	elimination-and-dispersal rate(z)	0.03
EO	Generation probability (GP)	0.5
	Exploration control parameter (a_1 *and* a_2)	2 and 1
HHO	–	–
SSA	Gliding constant and Gliding distance (G_c *and* d_g)	1.5 and 0.8
	predator probability (P_{dp})	0.1
L-SHADE	Control parameters (H *and* p)	6 and 0.11
	Arc rate (AR)	2.6
CMA-ES	Global step-size (σ)	0.25
TSA	Parameter P_{min}	1
	Parameter P_{max}	4

Content courtesy of Springer Nature, terms of use apply. Rights reserved.



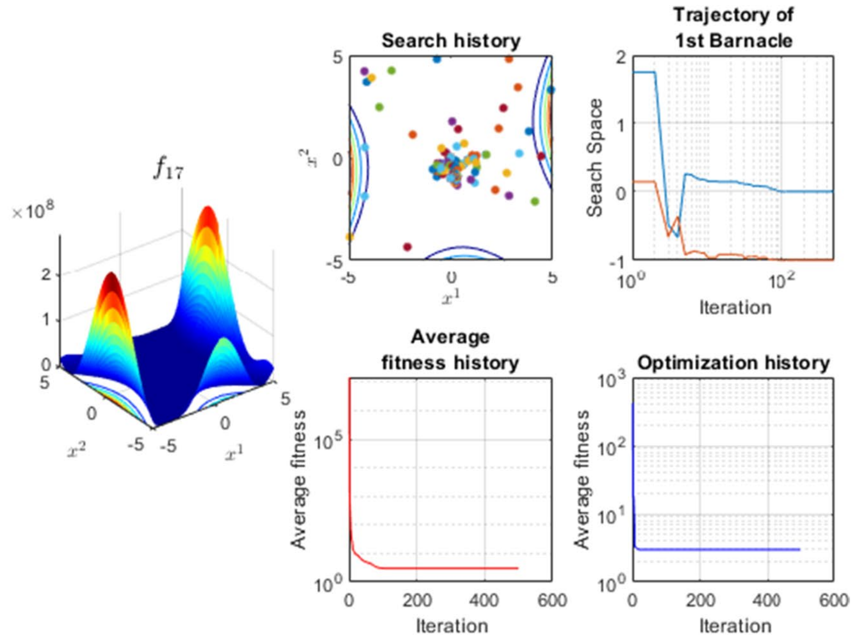
(a) f_2 qualitative result



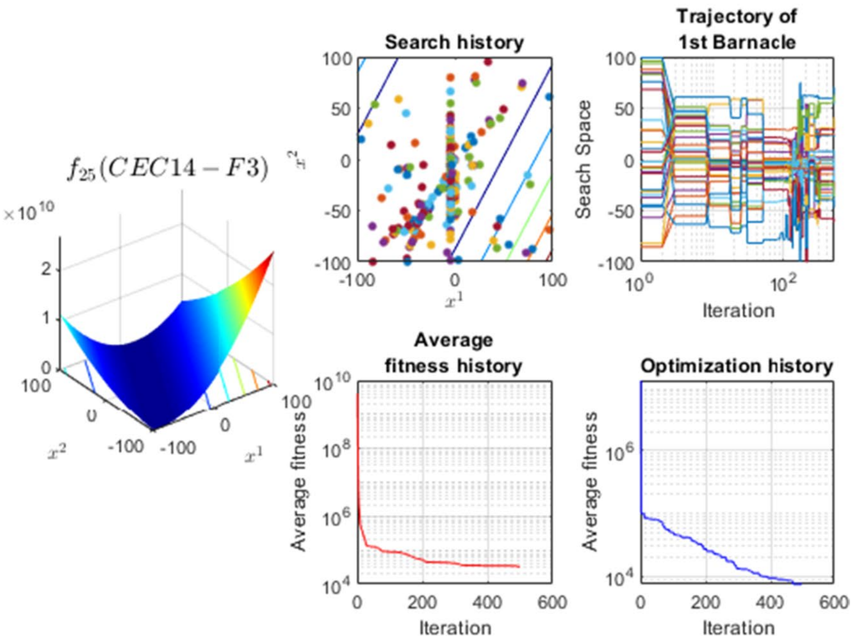
(b) f_{12} qualitative result

Fig. 1 **a** f_2 qualitative result, Fig. 1**b** f_{12} qualitative result, Fig. 1**c** f_{17} qualitative result, Fig. 1**d** $f_{25}(CEC14-F3)$ qualitative result, Fig. 1**e** $f_{35}(CEC14-F13)$ qualitative result, Fig. 1**f** $f_{46}(CEC14-F24)$ qualitative result

Content courtesy of Springer Nature, terms of use apply. Rights reserved.



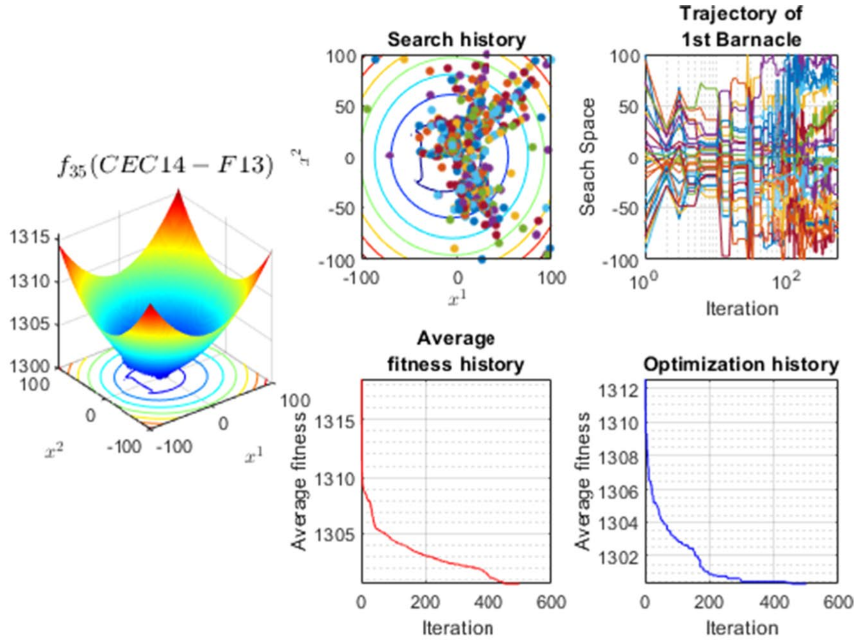
(c) f_{17} qualitative result



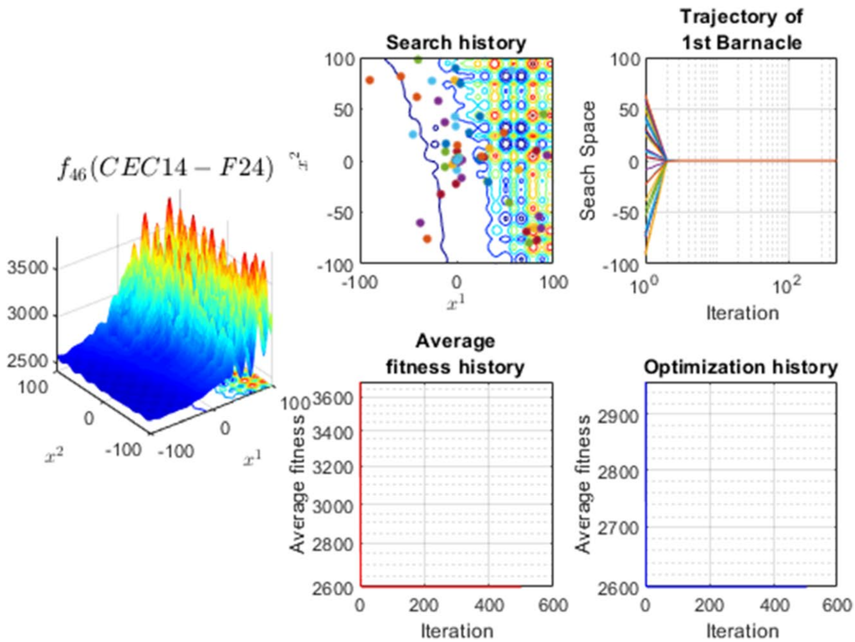
(d) $f_{25}(CEC14-F3)$ qualitative result

Fig. 1 (continued)

Content courtesy of Springer Nature, terms of use apply. Rights reserved.



(e) $f_{35}(CEC14-F13)$ qualitative result



(f) $f_{46}(CEC14-F24)$ qualitative result

Fig. 1 (continued)

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

This behaviour of the algorithms gives a clear demonstration of its exploration as well as exploitation ability.

The next qualitative metric for evaluating the performance of the EBMO is the trajectory of the first barnacle for all dimensions. It describes the locations of the first barnacle with all its dimension during the progress of the algorithm. It can be visualized from the figure that, initially the control variables of the selected classical test functions are scattered in the search space and later it converges toward the optimal solution. The oscillatory behaviour in the beginning stage shows the exploration acquired by the algorithm for searching global minima. Unlike classical test functions, the fluctuations are more in most of the cases of IEEE CEC 2014 test function due to the presence of a high level of complexity in the initial stage of the iterations.

A cumulative behaviour of all the barnacles that participated in the searching process of EBMO to reach the global solution is represented by the Average fitness history. A descending curve for average history reveals the collaborative behaviour of all barnacles to reach global minima. The last one of the qualitative metrics is the optimization history. It tracks the fitness value of the best barnacle during its journey from the beginning to the end of the iteration. The behaviour of the optimization history varies from problem to problem. For unimodal problems, the curve looks smooth. Whereas for multimodal or composite functions, the optimization curve becomes step-like at some instant indicating no improvement in fitness value during a specified period due to more complexity in the problem.

4.3.2 Performance analysis of EBMO on classical benchmark functions

The statistical results including Median value, Average value, and standard deviation over 31 independent runs for various unimodal and multimodal functions are depicted in Table 3. it can be observed that EBMO can obtain the global minima in most of the unimodal functions. For the function f_5 EBMO dominates all other algorithms in its average and standard deviation values. In the test function f_6 and f_7 the result of EBMO was found better than BMO.

The exploration ability of an algorithm can be observed by analyzing its behaviour on multimodal functions. The performance on multimodal functions of scalable dimension ($f_8 - f_{12}$) reveals that EBMO provides the optimal solution for f_8, f_9 *and* f_{10}. For functions $f_{11},$ *and* f_{12}, EBMO outperformed on BMO with a significant difference but lagged from others and occupied the 7th and 4th places respectively. It can also be observed from the statical result shown in Table 3 that, EBMO can reach close to the global minima in all most all multimodal functions with fixed dimension($f_{13} - f_{22}$) and outperformed BMO for the test functions f_{20}, f_{21} *and* f_{22}.

To observe the convergence property, convergence curves of different algorithms on six classical test functions are presented in Fig. 2. The curve shows the best fitness value attained by an algorithm versus several iterations. It is clear to observe that EBMO dominates BMO in all cases and is found superior to all for the test functions f_1 and f_{10} for its fast convergence rate. EBMO is only lagging from SSA for the test functions f_5 and f_{12}. For the rest of the functions, the performance of EBMO is very close to the leading algorithms. A boxplot is presented in Fig. 3 to realize how the optimal values of different benchmark functions are obtained on 31 independent runs of the different algorithms. EBMO has proven to be more consistent among other optimization algorithms to achieve optimal value.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 3 Statistical result for classical benchmark functions

Function	Metric	EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
f_1	Median	0	0	0	0	1.29E-41	8.31E-104	7.28E-64	3.97E-28	6.85E-11	2.07E-22
	Average	0	0	0	0	8.36E-41	4.86E-95	2.23E-08	2.00E-27	8.30E-11	8.93E-22
	Std. Dev.	0	0	0	0	2.09E-40	2.36E-94	1.22E-07	3.18E-27	4.97E-11	1.91E-21
f_2	Median	0	1.52E-298	5.82E-255	1.52E-208	4.53E-24	7.77E-54	8.00E-34	2.23E-14	1.61E-05	6.66E-14
	Average	0	1.92E-285	2.00E-249	4.00E-137	6.35E-24	8.01E-50	1.21E-06	2.56E-14	1.63E-05	1.19E-13
	Std. Dev.	0	0	0	2.23E-136	6.16E-24	4.36E-49	4.37E-06	1.45E-14	5.18E-06	1.44E-13
f_3	Median	0	0	0	0	8.19E-11	2.53E-87	3.62E-56	1.28E-13	1.41E+00	5.92E-05
	Average	0	0	0	4.87E-273	3.63E-09	1.16E-78	3.69E+03	5.47E-13	2.65E+00	2.37E-04
	Std. Dev.	0	0	0	0	7.72E-09	6.20E-78	1.48E+04	1.29E-12	3.31E+00	4.34E-04
f_4	Median	0	9.74E-295	2.57E-250	1.01E-207	1.05E-10	1.59E-52	1.36E-30	2.14E-06	1.75E-04	4.84E-09
	Average	0	1.80E-284	1.39E-241	3.96E-162	2.71E-10	6.23E-49	7.17E-09	2.96E-06	1.67E-04	9.40E-09
	Std. Dev.	0	0	0	1.74E-161	3.76E-10	3.36E-48	3.69E-08	3.12E-06	5.05E-05	1.17E-08
f_5	Median	1.53E-03	2.77E+01	2.30E+01	1.17E+00	2.53E+01	3.98E-03	0	1.39E+01	1.83E+01	5.73E+00
	Average	8.00E-03	2.78E+01	2.31E+01	5.43E+00	2.54E+01	7.17E-03	9.26E-01	1.43E+01	3.99E+01	5.68E+00
	Std. Dev.	1.60E-02	2.92E-01	4.40E-01	1.00E+01	2.61E-01	8.61E-03	5.16E+00	9.51E-01	1.20E+02	1.38E-01
f_6	Median	1.49E-04	4.37E-01	1.06E-11	5.06E-03	9.27E-06	6.28E-05	0	7.09E-28	7.74E-11	5.87E-11
	Average	1.71E-04	4.63E-01	3.26E-10	6.23E-03	1.23E-05	1.54E-04	2.32E-09	3.41E-27	8.36E-11	7.49E-11
	Std. Dev.	1.63E-04	2.83E-01	8.69E-10	4.33E-03	1.36E-05	2.33E-04	9.08E-09	8.01E-27	5.09E-11	3.90E-11
f_7	Median	1.14E-05	7.44E-04	1.19E-04	1.01E-04	1.08E-03	1.24E-04	1.18E-04	1.37E-03	4.18E-03	8.78E-03
	Average	1.39E-05	9.28E-04	1.39E-04	1.36E-04	1.27E-03	1.58E-04	2.18E-04	1.41E-03	4.93E-03	8.81E-03
	Std. Dev.	1.23E-05	7.48E-04	1.16E-04	1.03E-04	6.63E-04	1.35E-04	2.18E-04	4.92E-04	2.27E-03	3.46E-03
f_8	Median	0	0	0	0	0	0	0	5.64E+00	1.61E+02	1.93E-02
	Average	0	0	0	0	0	0	7.80E+01	5.53E+00	1.55E+02	1.82E-02
	Std. Dev.	0	0	0	0	0	0	1.27E+02	1.59E+00	2.97E+01	4.85E-02
f_9	Median	8.88E-16	8.88E-16	8.88E-16	8.88E-16	7.99E-15	8.88E-16	8.88E-16	3.29E-14	2.37E-06	6.27E-10
	Average	8.88E-16	8.88E-16	8.88E-16	8.88E-16	8.57E-15	8.88E-16	1.14E-06	4.10E-14	2.36E-06	1.49E-08
	Std. Dev.	0	0	0	0	2.26E-15	0	4.87E-06	2.54E-14	5.90E-07	2.32E-08

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 3 (continued)

Function	Metric	EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
f_{10}	Median	0	0	0	0	0	0	0	0	4.65E-10	1.06E-02
	Average	0	0	0	0	0	0	3.11E-13	0	4.91E-10	7.88E-03
	Std. Dev.	0	0	0	0	0	0	1.24E-12	0	2.47E-10	8.52E-03
f_{11}	Median	5.49E-06	1.24E-02	1.63E-12	4.12E-03	3.84E-07	4.79E-06	1.57E-32	2.37E-16	4.98E-12	6.63E-02
	Average	1.13E-05	1.37E-02	7.61E-10	7.16E-03	4.49E-07	1.23E-05	5.64E-12	2.43E-16	5.97E-12	7.39E-02
	Std. Dev.	1.46E-05	7.94E-03	2.91E-09	9.33E-03	3.70E-07	2.04E-05	3.14E-11	1.20E-16	3.02E-12	4.40E-02
f_{12}	Median	3.34E-05	2.97E+00	2.97E+00	3.56E-03	6.73E-05	9.43E-05	1.35E-32	5.47E-15	4.81E-11	2.75E-06
	Average	7.02E-05	2.97E+00	1.88E+00	6.55E-03	2.63E-02	1.55E-04	1.35E-15	8.37E-15	7.09E-11	2.84E-06
	Std. Dev.	1.09E-04	1.20E-03	1.39E+00	7.61E-03	5.25E-02	2.40E-04	7.54E-15	5.84E-15	5.40E-11	5.42E-07
f_{13}	Median	0.998004	11.718700	0.998004	0.998004	0.998004	0.998004	0.998004	9.123689	2.983996	1.298211
	Average	0.998004	9.457196	1.093818	0.998004	1.313010	1.158331	1.410945	8.900356	4.531904	1.452256
	Std. Dev.	1.22E-16	3.89E+00	5.33E-01	7.70E-13	1.75E+00	3.72E-01	1.05E+00	3.57E+00	3.41E+00	4.05E-01
f_{14}	Median	0.000323	0.000321	0.000452	0.000450	0.000315	0.000339	0.001674	0.000307	0.000307	0.000505
	Average	0.000384	0.000407	0.000659	0.000571	0.005521	0.000416	0.001674	0.000307	0.000307	0.009146
	Std. Dev.	1.09E-04	1.47E-04	3.82E-04	2.81E-04	8.90E-03	2.52E-04	1.23E-10	1.39E-19	1.90E-19	1.68E-02
f_{15}	Median	−1.031628	−1.031628	−1.031628	−1.031628	−1.031628	−1.031628	−0.995868	−1.031628	−1.031628	−1.031628
	Average	−1.031628	−1.031628	−1.031628	−1.031628	−1.031628	−1.031628	−0.980607	−1.031628	−1.031628	−1.029588
	Std. Dev.	6.52E-16	6.59E-16	6.00E-16	3.19E-10	5.93E-16	7.46E-10	4.43E-02	6.71E-16	6.77E-16	7.90E-03
f_{16}	Median	0.397887	0.397887	0.397887	0.397887	0.397887	0.397888	0.419196	0.823600	0.697072	0.397930
	Average	0.397887	0.397887	0.397887	0.397887	0.397887	0.397893	0.424973	0.840041	0.807284	0.397963
	Std. Dev.	3.57E-12	1.02E-07	0	9.15E-08	0	1.47E-05	2.35E-02	3.12E-01	4.40E-01	8.54E-05
f_{17}	Median	3	3	3	3	3	3.000000	8.280790	3	3	3.000055
	Average	3	3.870968	3	3	3	3.000000	10.320917	3	3	3.000536
	Std. Dev.	4.69E-15	4.85E+00	1.56E-15	1.45E-10	2.21E-15	3.09E-07	6.61E+00	1.93E-15	9.14E-16	1.03E-03

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 3 (continued)

Function	Metric	EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
f_{18}	Median	−3.862782	−3.862782	−3.862782	−3.862782	−3.862782	−3.861363	−3.828774	−3.862782	−3.862782	−3.862689
	Average	−3.862528	−3.862274	−3.862782	−3.862782	−3.862782	−3.859960	−3.821440	−3.862782	−3.862782	−3.862465
	Std. Dev.	1.42E-03	1.97E-03	2.64E-15	2.65E-07	2.53E-15	3.45E-03	2.66E-02	2.71E-15	2.71E-15	1.01E-03
f_{19}	Median	−3.321995	−3.203102	−3.321995	−3.203094	−3.321995	−3.008704	−2.839943	−3.321995	−3.321995	−3.318718
	Average	−3.267914	−3.244426	−3.279807	−3.248854	−3.261967	−3.038472	−2.861352	−3.298984	−3.291313	−3.220778
	Std. Dev.	5.65E-02	8.26E-02	5.78E-02	5.91E-02	6.97E-02	1.21E-01	1.91E-01	4.77E-02	5.29E-02	1.42E-01
f_{20}	Median	−10.153200	−5.055198	−10.153200	−10.152981	−10.153200	−5.053302	−10.153199	−5.055196	−10.153200	−5.048321
	Average	−10.153200	−5.055198	−7.694442	−10.152961	−8.103031	−5.050473	−8.355992	−5.055196	−9.199648	−6.235834
	Std. Dev.	5.62E-15	1.81E-15	2.81E+00	1.84E-04	2.66E+00	5.35E-03	2.46E+00	5.92E-07	2.52E+00	3.23E+00
f_{21}	Median	−10.402941	−5.087672	−10.402941	−10.402626	−10.402941	−5.085403	−10.402822	−5.087668	−10.402941	−9.488220
	Average	−10.402941	−5.087672	−8.196596	−10.402616	−9.276971	−5.395642	−8.526093	−5.087669	−10.402941	−6.814096
	Std. Dev.	5.79E-15	3.62E-15	2.88E+00	1.60E-04	2.64E+00	1.21E+00	2.57E+00	1.39E-06	3.61E-15	3.57E+00
f_{22}	Median	−10.536410	−5.128481	−5.128481	−10.536071	−10.536410	−5.125533	−10.536290	−5.128476	−10.536410	−3.812256
	Average	−10.536410	−5.651829	−6.540206	−10.535992	−8.884105	−4.949638	−8.905635	−5.128475	−10.536410	−5.955985
	Std. Dev.	3.46E-15	1.63E+00	3.15E+00	2.36E-04	3.15E+00	6.78E-01	2.62E+00	2.32E-06	3.57E-15	3.82E+00

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

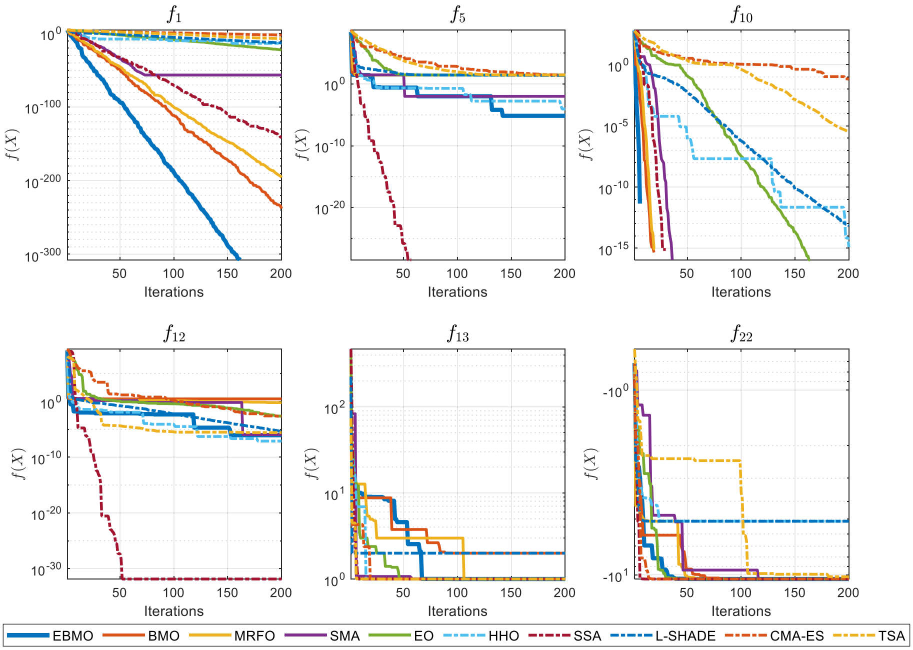


Fig. 2 The convergence curves of the six classical test functions ($f_1, f_5, f_{10}, f_{12}, f_{13}$ *and* f_{22})

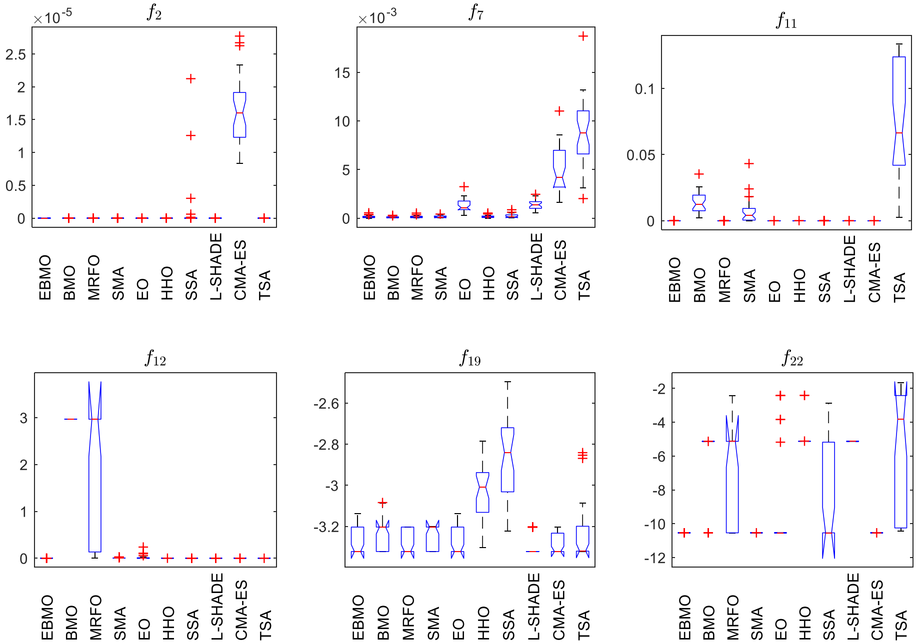


Fig. 3 Boxplot of the six classical test functions

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

4.3.3 Performance analysis of EBMO on IEEE CEC-2014 test functions

Table 4 shows a comparison of different optimization algorithms including EBMO on IEEE CEC-2014 test suit to explore its complex problem handling capabilities. Out of 30 complex test functions, EBMO can reach the optimal value for 11 test functions and outperform all for f_{27} and f_{40}. It can also be observed that EBMO surpasses BMO in all most all cases. For the test functions f_{23}, f_{24} and f_{39}, where the complexity level is very high, the result of EBMO is competitive to others and secured fifth place. For the remaining test functions, EBMO has a very close contest with leading algorithms. The convergence performance of various algorithms for 6 different test functions are given in Fig. 3. From the convergence curves, it is obvious that the enhancement introduced in BMO helps it to improve its convergence rate. Though the convergence rates of EBMO for f_{23} and f_{40} are lagging from some of the state-of-art methods, its performance is found superior or similar for other test functions (Fig. 4). To analyze EBMO consistency and overall performance, the results obtained from 31 independent runs were taken for the ANOVA test and the results are plotted in terms of the Box plot in Fig. 5. It can be observed that EBMO and BMO are consistently better than most of the optimization algorithms except in the function f_{48}.

4.3.4 Scalability analysis of EBMO

A performance assessment of the proposed EBMO algorithm for low- and high-dimensional problems with the help of scalability analysis is presented in this section. Since optimization problems in the real world often involve many variables, the algorithm is tested for 7 different dimensions: $d = \{10, 20, 40, 60, 100, 200, 400\}$ and the results are plotted in Fig. 6. The tests are performed with fixed population size and a maximum iteration count of 30 and 500 respectively for the above dimensions. As the requirements of the search agents of L-SHADE and CMA-ES algorithms depend upon the dimension size, they are not included in this test. The test reveals how effectively the algorithm can work in proportion to the increase in control variables for a given problem, while the population and maximum iterations counts are frozen to a particular value. It can be observed from the plots that EBMO can provide a consistent outcome in most of the test functions at different dimensions without affecting its performance. For the test functions $f_2, f_4, f_5, f_6 and f_{11}$, EBMO surpasses BMO and for the rest, the performance is almost similar. Overall, the steady behaviour of EBMO makes it's a suitable algorithm to face the challenges of variable dimension size problems.

4.3.5 Statistical analysis of EBMO

This section is included in the performance evaluation to demonstrate the difference between EBMO, and another algorithm based on various statistical analyses. At first, the non-parametric Friedman average rank test is conducted on the results obtained from various test functions. For a reliable comparison, the Friedman test required at least 5 different algorithms for more than 10 benchmark functions. The performance evaluation study included 10 algorithms in the comparative study, and these are applied to 52 different test functions. In this work, the Friedman tests have been conducted on three different groups. The first group consist of all classical function from f_1 to f_{22}, the CEC 2014 benchmark functions $f_{23} - f_{52}$ are included in the second

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 4 Statistical results for modern test function from CEC2014 dataset

Function	Metric	EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
$f_{23}\,(CEC14-F1)$	Median	5.87E+07	7.79E+07	8.22E+06	1.21E+07	1.29E+07	7.73E+07	1.67E+09	1.40E+09	3.07E+07	3.07E+07
	Average	6.40E+07	9.05E+07	8.44E+06	1.35E+07	1.33E+07	9.38E+07	1.68E+09	1.38E+09	3.59E+07	3.59E+07
	Std.	2.40E+07	5.54E+07	4.78E+06	8.22E+06	7.90E+06	5.19E+07	2.81E+08	2.03E+08	1.70E+07	1.70E+07
$f_{24}\,(CEC14-F2)$	Median	8.02E+08	1.45E+09	1.19E+04	1.45E+05	6.54E+04	6.17E+08	9.92E+10	6.85E+10	1.57E+05	1.57E+05
	Average	1.49E+09	1.96E+09	1.76E+04	1.71E+05	1.48E+05	8.47E+08	9.50E+10	6.92E+10	2.14E+05	2.14E+05
	Std. Dev.	1.95E+09	1.85E+09	1.49E+04	9.45E+04	2.16E+05	5.54E+08	9.76E+09	4.76E+09	2.45E+05	2.45E+05
$f_{25}\,(CEC14-F3)$	Median	1.53E+04	2.07E+04	6.54E+03	6.93E+03	2.07E+04	4.94E+04	8.92E+04	8.19E+04	2.31E+05	2.31E+05
	Average	1.63E+04	2.05E+04	1.21E+04	9.83E+03	2.17E+04	4.97E+04	1.23E+05	8.08E+04	2.33E+05	2.33E+05
	Std. Dev.	6.04E+03	7.95E+03	1.32E+04	1.01E+04	9.53E+03	7.88E+03	5.63E+04	4.83E+03	6.12E+04	6.12E+04
$f_{26}\,(CEC14-F4)$	Median	6.74E+02	6.94E+02	4.81E+02	5.50E+02	5.33E+02	7.25E+02	2.22E+04	1.42E+04	4.20E+02	4.20E+02
	Average	7.56E+02	7.18E+02	4.80E+02	5.40E+02	5.19E+02	7.66E+02	2.13E+04	1.42E+04	4.20E+02	4.20E+02
	Std. Dev.	2.70E+02	9.83E+01	4.04E+01	3.39E+01	3.02E+01	1.18E+02	2.93E+03	1.48E+03	6.29E-01	6.29E-01
$f_{27}\,(CEC14-F5)$	Median	5.20E+02	5.20E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02
	Average	5.20E+02	5.20E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02	5.21E+02
	Std. Dev.	1.92E-01	3.01E-01	6.82E-02	7.62E-02	1.00E-01	1.08E-01	6.00E-02	1.52E-01	5.43E-02	5.43E-02
$f_{28}\,(CEC14-F6)$	Median	6.25E+02	6.26E+02	6.32E+02	6.19E+02	6.11E+02	6.36E+02	6.45E+02	6.43E+02	6.00E+02	6.00E+02
	Average	6.26E+02	6.27E+02	6.32E+02	6.20E+02	6.11E+02	6.35E+02	6.44E+02	6.43E+02	6.00E+02	6.00E+02
	Std. Dev.	4.10E+00	3.99E+00	3.34E+00	3.53E+00	3.09E+00	3.30E+00	1.59E+00	9.27E-01	3.51E-03	3.51E-03
$f_{29}\,(CEC14-F7)$	Median	7.09E+02	7.16E+02	7.00E+02	7.01E+02	7.00E+02	7.06E+02	1.62E+03	1.39E+03	7.00E+02	7.00E+02
	Average	7.18E+02	7.22E+02	7.00E+02	7.01E+02	7.00E+02	7.07E+02	1.56E+03	1.39E+03	7.00E+02	7.00E+02
	Std. Dev.	2.22E+01	2.02E+01	4.38E-01	3.01E-02	1.41E-01	4.08E+00	9.10E+01	6.48E+01	9.95E-08	9.95E-08
$f_{30}\,(CEC14-F8)$	Median	9.50E+02	9.45E+02	9.49E+02	8.68E+02	8.53E+02	9.52E+02	1.20E+03	1.14E+03	9.60E+02	9.60E+02
	Average	9.47E+02	9.43E+02	9.46E+02	8.68E+02	8.62E+02	9.50E+02	1.20E+03	1.14E+03	9.58E+02	9.58E+02
	Std. Dev.	2.16E+01	2.01E+01	1.88E+01	1.49E+01	2.07E+01	2.18E+01	1.16E+01	1.66E+01	1.06E+01	1.06E+01
$f_{31}\,(CEC14-F9)$	Median	1.09E+03	1.09E+03	1.06E+03	1.04E+03	9.97E+02	1.09E+03	1.36E+03	1.27E+03	1.06E+03	1.06E+03
	Average	1.08E+03	1.09E+03	1.06E+03	1.04E+03	9.99E+02	1.10E+03	1.36E+03	1.26E+03	1.06E+03	1.06E+03
	Std. Dev.	1.68E+01	1.74E+01	2.70E+01	2.61E+01	2.61E+01	1.78E+01	4.38E+00	1.09E+01	1.00E+01	1.00E+01

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 4 (continued)

Function	Metric	EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
f_{32} $(CEC14-F10)$	Median	3.86E+03	3.80E+03	4.52E+03	2.70E+03	3.04E+03	4.75E+03	9.08E+03	8.81E+03	8.15E+03	8.15E+03
	Average	3.73E+03	3.85E+03	4.57E+03	2.60E+03	3.22E+03	4.70E+03	9.12E+03	8.76E+03	8.15E+03	8.15E+03
	Std. Dev.	6.51E+02	6.07E+02	5.99E+02	4.86E+02	7.15E+02	7.56E+02	1.64E+02	3.84E+02	3.34E+02	3.34E+02
f_{33} $(CEC14-F11)$	Median	5.20E+03	5.38E+03	5.09E+03	4.81E+03	5.34E+03	5.97E+03	9.55E+03	9.34E+03	8.45E+03	8.45E+03
	Average	5.26E+03	5.37E+03	5.21E+03	4.90E+03	5.26E+03	6.02E+03	9.50E+03	9.28E+03	8.43E+03	8.43E+03
	Std. Dev.	6.52E+02	5.76E+02	6.76E+02	7.11E+02	8.30E+02	6.12E+02	2.57E+02	3.22E+02	2.70E+02	2.70E+02
f_{34} $(CEC14-F12)$	Median	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03
	Average	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.20E+03
	Std. Dev.	6.04E-01	7.10E-01	5.52E-01	2.62E-01	5.96E-01	6.08E-01	5.74E-01	3.40E-02	3.21E-01	3.21E-01
f_{35} $(CEC14-F13)$	Median	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.31E+03	1.31E+03	1.30E+03	1.30E+03
	Average	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.31E+03	1.31E+03	1.30E+03	1.30E+03
	Std. Dev.	4.48E-01	1.55E-01	1.16E-01	1.20E-01	7.71E-02	1.30E-01	7.50E-01	5.36E-01	4.56E-02	4.56E-02
f_{36} $(CEC14-F14)$	Median	1.40E+03	1.40E+03	1.40E+03	1.40E+03	1.40E+03	1.40E+03	1.70E+03	1.67E+03	1.40E+03	1.40E+03
	Average	1.40E+03	1.41E+03	1.40E+03	1.40E+03	1.40E+03	1.40E+03	1.69E+03	1.66E+03	1.40E+03	1.40E+03
	Std. Dev.	2.97E+00	9.83E+00	2.32E-01	3.43E-01	1.47E-01	1.93E+00	2.70E+01	2.32E+01	6.35E-02	6.35E-02
f_{37} $(CEC14-F15)$	Median	1.55E+03	1.63E+03	1.65E+03	1.51E+03	1.51E+03	1.55E+03	2.76E+05	2.35E+05	1.51E+03	1.51E+03
	Average	1.57E+03	1.81E+03	1.69E+03	1.51E+03	1.51E+03	1.56E+03	2.76E+05	2.41E+05	1.51E+03	1.51E+03
	Std. Dev.	6.94E+01	3.72E+02	2.18E+02	3.88E+00	4.01E+00	1.51E+01	7.95E-11	7.14E+04	1.15E+00	1.15E+00
f_{38} $(CEC14-F16)$	Median	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03
	Average	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03
	Std. Dev.	4.35E-01	4.39E-01	4.27E-01	5.39E-01	6.60E-01	3.14E-01	2.25E-01	1.37E-01	2.30E-01	2.30E-01
f_{39} $(CEC14-F17)$	Median	2.37E+06	3.65E+06	5.62E+05	2.01E+06	1.17E+06	6.51E+06	1.02E+08	1.02E+08	1.96E+06	1.96E+06
	Average	2.81E+06	4.28E+06	9.41E+05	2.37E+06	1.32E+06	1.07E+07	1.10E+08	1.18E+08	2.30E+06	2.30E+06
	Std. Dev.	2.19E+06	3.22E+06	9.24E+05	1.43E+06	9.32E+05	1.02E+07	4.52E+07	3.95E+07	1.50E+06	1.50E+06

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 4 (continued)

Function	Metric	EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
$f_{40}\,(CEC14-F18)$	Median	2.97E+03	3.05E+03	3.35E+03	2.48E+04	3.15E+03	3.40E+05	3.97E+09	4.21E+09	1.24E+06	1.24E+06
	Average	3.97E+03	5.88E+04	5.71E+03	4.06E+04	3.68E+03	3.83E+05	4.05E+09	4.21E+09	1.66E+06	1.66E+06
	Std. Dev.	1.91E+03	3.03E+05	5.20E+03	3.94E+04	1.78E+03	3.19E+05	1.09E+09	1.18E+09	1.24E+06	1.24E+06
$f_{41}\,(CEC14-F19)$	Median	1.96E+03	1.93E+03	1.93E+03	1.91E+03	1.91E+03	2.00E+03	2.52E+03	2.36E+03	1.91E+03	1.91E+03
	Average	1.96E+03	1.95E+03	1.93E+03	1.92E+03	1.91E+03	2.00E+03	2.49E+03	2.36E+03	1.91E+03	1.91E+03
	Std. Dev.	3.40E+01	4.18E+01	1.35E+01	2.14E+01	2.19E+01	3.36E+01	1.28E+02	4.04E+01	1.02E+00	1.02E+00
$f_{42}\,(CEC14-F20)$	Median	4.53E+04	4.10E+04	2.68E+04	3.60E+04	1.95E+04	5.18E+04	1.36E+06	4.09E+05	4.23E+04	4.23E+04
	Average	4.35E+04	4.42E+04	2.97E+04	4.04E+04	2.13E+04	5.55E+04	1.53E+06	4.85E+05	5.19E+04	5.19E+04
	Std. Dev.	2.07E+04	2.24E+04	1.62E+04	2.02E+04	8.73E+03	2.61E+04	1.23E+06	3.51E+05	3.39E+04	3.39E+04
$f_{43}\,(CEC14-F21)$	Median	4.89E+05	6.38E+05	2.98E+05	1.00E+06	3.35E+05	2.04E+06	3.90E+07	4.99E+07	8.41E+05	8.41E+05
	Average	6.41E+05	8.40E+05	3.74E+05	1.11E+06	5.36E+05	3.26E+06	3.58E+07	4.94E+07	1.07E+06	1.07E+06
	Std. Dev.	3.60E+05	7.69E+05	2.61E+05	8.88E+05	5.36E+05	3.32E+06	1.29E+07	1.87E+07	7.32E+05	7.32E+05
$f_{44}\,(CEC14-F22)$	Median	2.88E+03	3.04E+03	2.95E+03	2.90E+03	2.59E+03	3.27E+03	4.74E+03	5.43E+03	2.81E+03	2.81E+03
	Average	2.90E+03	3.00E+03	2.99E+03	2.86E+03	2.59E+03	3.25E+03	4.69E+03	5.98E+03	2.81E+03	2.81E+03
	Std. Dev.	2.20E+02	2.70E+02	2.45E+02	2.11E+02	1.87E+02	2.96E+02	4.43E+02	2.16E+03	1.21E+02	1.21E+02
$f_{45}\,(CEC14-F23)$	Median	2500	2500	2500	2500	2615.284	2500	2500	2500	2631.923	2631.923
	Average	2500	2500	2500	2500	2615.294	2500	2517.412	2500	2630.732	2630.732
	Std. Dev.	0	0	0	0	4.16E-02	0	9.65E+01	1.22E-12	3.81E+00	3.81E+00
$f_{46}\,(CEC14-F24)$	Median	2600	2600	2600	2600	2600.019	2600.00002	2600	2600	2647.544	2647.544
	Average	2600	2600	2600	2600	2600.024	2600.00021	2607.304	2600.001	2645.045	2645.045
	Std. Dev.	0	0	0	0	1.08E-02	5.07E-04	4.07E+01	6.40E-04	8.20E+00	8.20E+00
$f_{47}\,(CEC14-F25)$	Median	2700	2700	2700	2700	2700	2700	2700	2700	2707.039	2707.039
	Average	2700	2700	2700	2700	2702.012	2700	2703.820	2700	2707.196	2707.196
	Std. Dev.	0	0	0	0	4.31E+00	0	2.13E+01	5.05E-13	2.52E+00	2.52E+00

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 4 (continued)

Function	Metric	EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
f_{48} $(CEC14-F26)$	Median	2800	2800	2700.620	2700.800	2700.369	2800	2716.458	2733.873	2700.267	2700.267
	Average	2751.920	2764.751	2700.633	2700.772	2722.853	2764.682	2719.303	2747.316	2706.713	2706.713
	Std. Dev.	5.05E+01	4.83E+01	1.57E-01	1.07E-01	4.24E+01	4.84E+01	9.92E+00	3.37E+01	3.59E+01	3.59E+01
f_{49} $(CEC14-F27)$	Median	2900	2900	2900	2900	3287.050	2900	3497.072	2991.935	3005.112	3005.112
	Average	2900	2900	2900	2900	3289.152	2900	3533.187	3215.813	3393.304	3393.304
	Std. Dev.	0	0	0	3.82E-11	9.19E+01	0	6.17E+02	5.03E+02	4.58E+02	4.58E+02
f_{50} $(CEC14-F28)$	Median	3000	3000	3000	3000	3732.921	3000	3000	3000	3218.704	3218.704
	Average	3000	3000	3000	3000	3794.410	3000	3928.068	3000	3221.846	3221.846
	Std. Dev.	0	0	0	0	1.64E+02	0	1.96E+03	3.31E-09	1.67E+01	1.67E+01
f_{51} $(CEC14-F29)$	Median	3100	3100	3117.795	6672.423	5491.868	3100	623,154.727	3100	3129.128	3129.128
	Average	3100	3100	3130.464	1,524,084.974	1,877,034.668	1,692,774.130	614,695.102	3100	3130.209	3130.209
	Std. Dev.	0	0	3.45E+01	3.54E+06	3.91E+06	9.40E+06	1.80E+04	2.26E-05	4.06E+00	4.06E+00
f_{52} $(CEC14-F30)$	Median	3200	3200	4384.203	14,219.585	8184.104	39,629.523	3200	3200.002	4582.895	4582.895
	Average	3200	3200	4408.460	15,225.312	8724.225	141,448.466	891,611.844	3200.002	4592.033	4592.033
	Std. Dev.	0	0	4.74E+02	1.08E+04	2.93E+03	1.83E+05	1.66E+06	1.12E-03	2.10E+02	2.10E+02

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

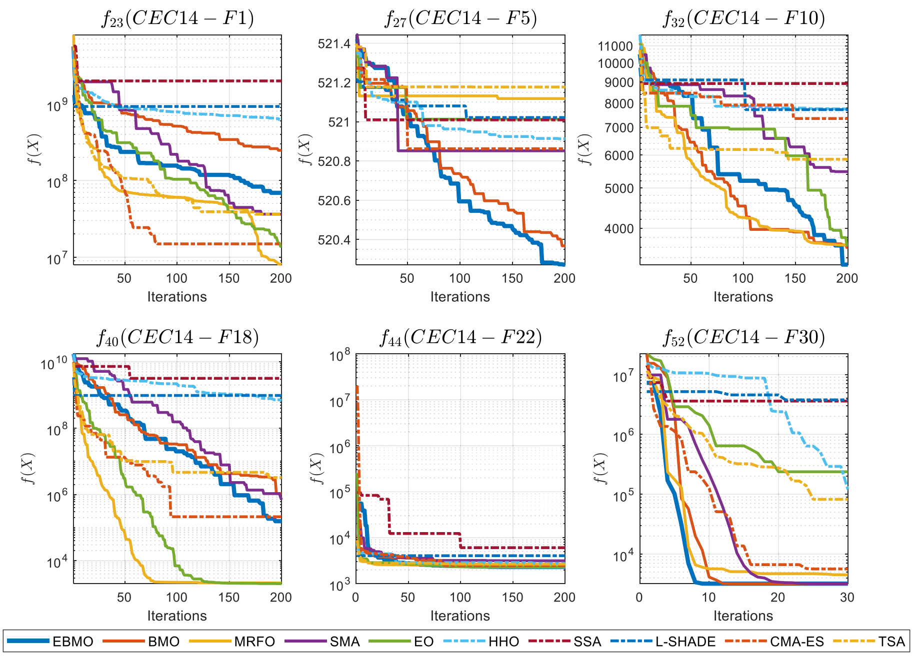


Fig. 4 The convergence curves of the six modern test functions from CEC2014 ($f_{23}, f_{27}, f_{32}, f_{40}, f_{44}$ *and* f_{52})

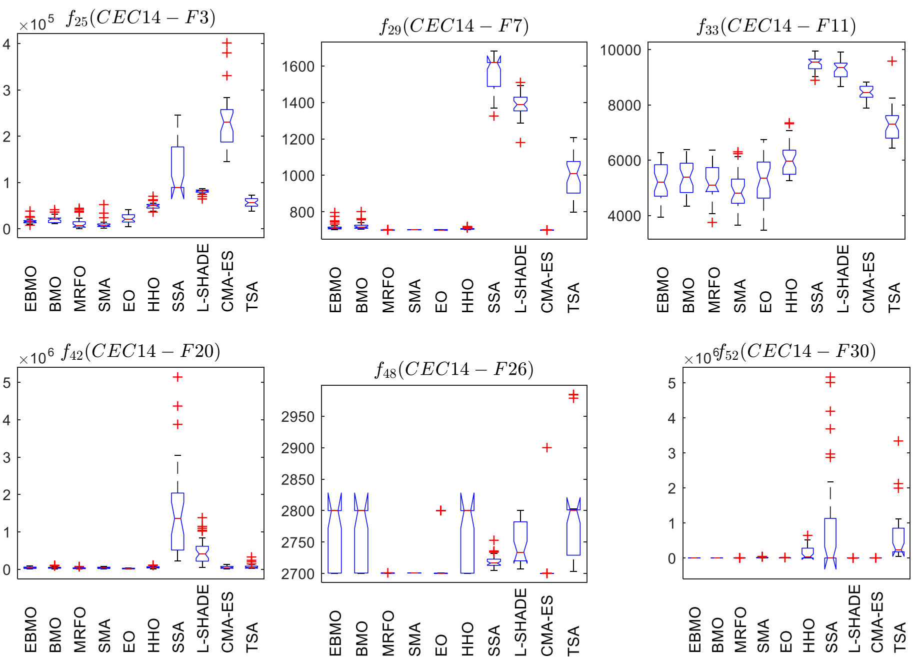


Fig. 5 Boxplot of the six modern test functions from CEC 2014

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

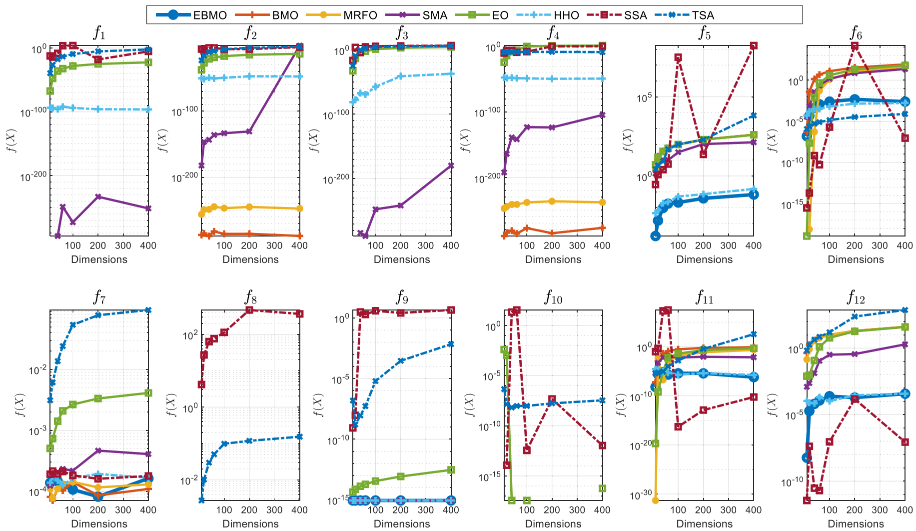


Fig. 6 Scalability analysis with $d=\{10, 20, 40, 60, 100, 200, 400\}$

group and the third group consists of all 52 test functions. The Friedman mean rank values obtained over 31 independent runs on the test functions included in each group are shown in Table 5. Based on the Friedman score, EBMO ranks first for Group 1 with a Friedman score of 3.911. For the second group, EBMO ranks fourth after lagging from SMA, MRFO, and EO. A comparison by taking all 52 test functions in the third group also has been done, which shows that the overall performance of EBMO is better than all with a Friedman mean rank of 3.864 and secured the first rank by beating all state-of-the-art methods.

In the second part, a post hoc analysis using the Bonferroni-Dunn test [84] is conducted to observe which algorithm's performance is significantly different from EBMO and which are equivalent to it. The test identifies the significant difference between the two algorithms by comparing the difference between the average ranks obtained by the method with a critical value obtained from the method given in [45] for a significance level of 95% ($\alpha=0.05$). If the difference is more than the critical value, the algorithm's performance is different significantly otherwise marked as similar. To perform the test the proposed EBMO is taken as the control algorithm and its performance is compared with the rest of the methods. Figure 7 displays the average ranks obtained from the three groups of functions defined above. The critical values calculated for each group are represented by horizontal lines. The control algorithm EBMO is superior to those algorithms for which the calculated average rank crossed these horizontal lines. For Group 1, EBMO was found significantly better than HHO, SSA, L-SHADE, and CMA-ES. Similarly, there is a significant difference in the performance of EBMO for Group 2 over SSA and L-SHADE. Finally, EBMO was found to significantly outperform HHO, SSA, L-SHADE, CMA-ES and TSA when tested on Group 3 functions.

The Bonferroni-Dunn test fails to give a fair decision about the algorithms which ranks are close to the critical values. Thus, in the final part of the statistical analysis, a Holm test [28] is conducted to identify which algorithms are better than EBMO and which are inferior. Holm's test is one of the widely used multiple test methods based on a sequential rejective process. The compared algorithms in this test are first arranged according to their p value in increasing order from top to bottom as shown in Tables 6, 7, and 8. Each algorithm is then assigned with

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 5 Friedman Score ba sed on 31 independent runs of test functions

Functions		EBMO	BMO	MRFO	SMA	EO	HHO	SSA	L-SHADE	CMA-ES	TSA
Group 1: Classical functions ($f_1 - f_{22}$)	Friedman mean rank	**3.911**	5.599	4.279	5.690	5.605	6.394	5.827	5.957	6.033	5.706
	Ranking	**1**	3	2	5	4	10	7	8	9	6
Group 2: CEC14 testbed (($f_{23} - f_{52}$)	Friedman mean rank	4.060	4.456	3.927	**3.650**	3.934	5.216	8.649	7.892	5.267	7.947
	Ranking	4	5	2	**1**	3	6	10	8	7	9
Group 3: Overall ($f_1 - f_{52}$)	Friedman mean rank	**3.864**	4.840	3.976	4.363	4.517	5.620	7.828	7.027	5.535	7.429
	Ranking	**1**	5	2	3	4	7	10	8	6	9

The best results are highlighted in bold

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

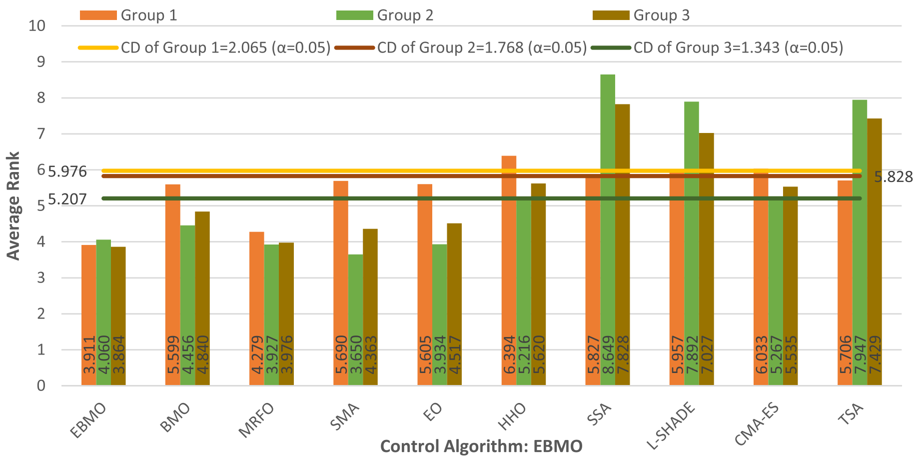


Fig. 7 Result of Bonferroni–Dunn test for various algorithms and function groups with $\alpha = 0.05$

an algorithm number $'i'$ starting from bottom to top. The test then starts from the most significant p value and rejects the defined null hypothesis sequentially if the corresponding p value is less than α/i. Once the process encounters a case where the null hypothesis is accepted, it stops the process and considers the same for the remaining cases. The results obtained from Group-1 are presented in Table 6. It can be observed that EBMO performance is significantly better than L-SHADE, BMO, MRFO, CMA-ES, EO, TSA and HHO. The Holm test conducted on the result of Group-2 functions is depicted in Table 7 and it is found that the proposed EBMO performance is statistically like HHO and BMO. Finally, a Holm test is also conducted for Group-3 function and the test results are presented in Table 8 show the superiority of EBMO result which is significantly different from most of the algorithms except BMO and HHO.

4.3.6 Discussion on results of EBMO

From the various post hoc analysis on the performance of different algorithms on benchmark functions, there is a significant improvement of EBMO over almost all algorithms including BMO for the set of classical benchmark functions. Though there is no significant

Table 6 Holm's test for group 1 functions $f_1 - f_{22}$ (EBMO as the control algorithm)

EBMO vs.	Rank	z-value	p-value	i	α/i (0.05)
L-SHADE	3.911	8.96263	1.02E-18	9	0.0055555
BMO	5.599	8.55854	3.04E-17	8	0.00625
MRFO	4.279	5.71985	1.31E-08	7	0.007143
CMA-ES	6.033	5.64047	2.06E-08	6	0.008333
EO	5.605	4.82381	1.57E-06	5	0.01
TSA	5.706	4.45409	8.43E-06	4	0.0125
HHO	6.394	4.44914	9.32E-06	3	0.016667
SSA	**5.827**	**1.40264**	**0.160953**	**2**	**0.025**
SMA	**5.690**	**1.10973**	**0.267311**	**1**	**0.05**

The best results are highlighted in bold

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 7 Holm's test for group 2 functions $f_{23}(CEC14-F1)$ *to* f_{52} $(CEC14-F30)$ (EBMO as the control algorithm)

EBMO vs.	Rank	*z*-value	*p*-value	*i*	*α*/*i* (0.05)
TSA	7.947	7.10843	2.78E-09	9	0.0055555
L-SHADE	7.892	6.00547	2.51E-09	8	0.00625
SSA	8.649	5.89629	4.57E-09	7	0.007143
MRFO	3.927	−3.56759	0.000378	6	0.008333
EO	3.934	−3.55038	0.000403	5	0.01
SMA	3.650	−3.54682	0.000409	4	0.0125
CMA-ES	5.267	−3.49433	0.000497	3	0.016667
HHO	**5.216**	**−1.27719**	**0.202371**	**2**	**0.025**
BMO	**4.456**	**0.77374**	**0.439757**	**1**	**0.05**

The best results are highlighted in bold

difference between the performance of EBMO and BMO on the CEC 2014 benchmark function, EBMO is capable of dominating BMO by some improvement on achieving the optimal result and standing one step ahead on BMO as per the Friedman test. However, the objective of enhancement of BMO has been achieved by the supplement of the additional mutation strategy from the overall Friedman test score and convergence speed. The scalability test on EBMO shows that it has also the capability to handle problems with variable dimensions.

5 The proposed exponential entropy-based multilevel thresholding using EBMO(EBMO-EE)

In this section, we proposed a multilevel thresholding method for colour images using EBMO as the optimization algorithm applied to maximize the exponential entropy discussed in Section 3. A framework of the multilevel thresholding process of the colour image is shown in Fig. 8. Initially, the colour image is divided into three colour channel planes, Red, green, and blue: $I_c(m,n) \in \{I_r(m,n), I_g(m,n), I_b(m,n)\}$. The thresholding algorithm is applied to each colour channel independently with the help of an optimization algorithm. To obtain $k+1$ distinct regions from each channel, each barnacle X_i in the EBMO algorithm is represented by a vector having k control variables, where $i=1, 2, 3\ldots\ldots N$ and 'N' represent the population size. As the objective of the thresholding is to divide the image into a distinct region without affecting the desired information present in the image, the exponential

Table 8 Holm's test for group 3 functions f_1-f_{52} (EBMO as the control algorithm)

EBMO vs.	Rank	*z*-value	*p*-value	*i*	*α*/*i* (0.05)
L-SHADE	7.027	5.94153	3.13E-09	9	0.0055555
SSA	7.828	5.84532	5.61E-09	8	0.00625
TSA	7.429	5.29713	1.18E-07	7	0.007143
MRFO	3.976	−3.55213	0.00039	6	0.008333
EO	4.517	−3.53499	0.00041	5	0.01
SMA	4.363	−3.53144	0.00042	4	0.0125
CMA-ES	5.535	−3.47917	0.00051	3	0.016667
HHO	**5.620**	**−1.27132**	**0.20375**	**2**	**0.025**
BMO	**4.840**	**0.77020**	**0.44127**	**1**	**0.05**

The best results are highlighted in bold

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

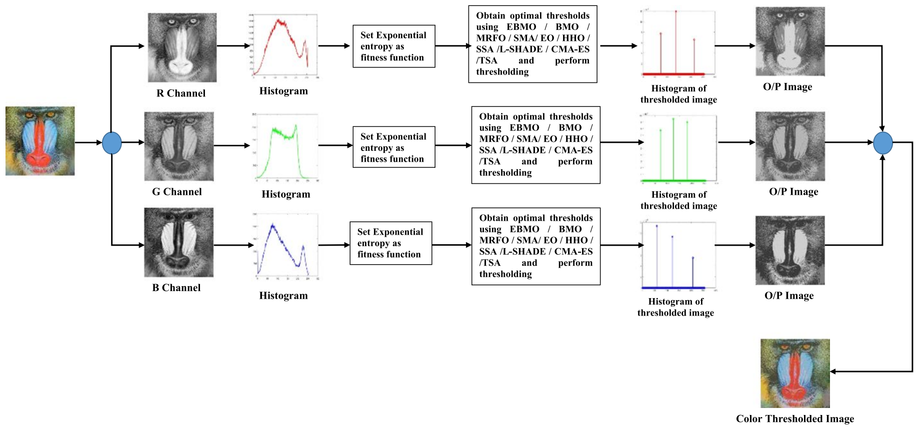


Fig. 8 Framework of proposed EBMO-EE based multilevel thresholding process

entropy which is the information measure is considered as the fitness function in our proposed method. Meta-heuristic algorithms are used to provide the desired threshold value by maximizing the fitness function. Once the thresholding operation is completed for each colour channel with the help of k thresholds, the thresholded images of all colour channels are then combined to produce the required colour RGB thresholded image:

$$T_c(m,n) = \left[T_r(m,n), T_g(m,n), T_b(m,n)\right] \tag{15}$$

Each of the colour channels is represented by only $Q = k + 1$ gray levels. So, the maximum number of the gray level required to represent the thresholded colour image $T_c(m,n)$ becomes Q^3 which is quite less than the original input image.

The different steps involved in the process of proposed EBMO-EE based multilevel thresholding are presented in Fig. 9. The thresholding process started by taking an RGB image as input. As an RGB image consists of three different channels (Red, Green, and Blue), the process is applied to each plane individually. The gray level distribution of a channel is then calculated in the form of a histogram. Once the above information is available, individual barnacle vectors in the population and other related parameters of EBMO are initialized from the program's general perspective. The number of thresholds ('k') that must be calculated for multilevel thresholding is specified as the dimension of each barnacle vector. In the population, each barnacle vector provides a collection of thresholds for segmenting the images into $k + 1$ classes. The information content of each barnacle is represented by the exponential entropy associated with the respective segmented image, which is quantified by the fitness related to each barnacle. The EBMO algorithm discussed in Section 4 is now updating the barnacle vectors in the population by its updating strategies in each iteration. At the end of the iteration, the EBMO produces the optimal threshold vector for the selected channel. The process is repeated for each colour channel.

6 Results and discussions

The performance of the proposed exponential entropy-based multilevel thresholding using EBMO (EBMO-EE) is presented in this section. All the simulations are performed in MATLAB R2015b supported by Intel Core i3-8th generation 2.3 GHz processor with

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

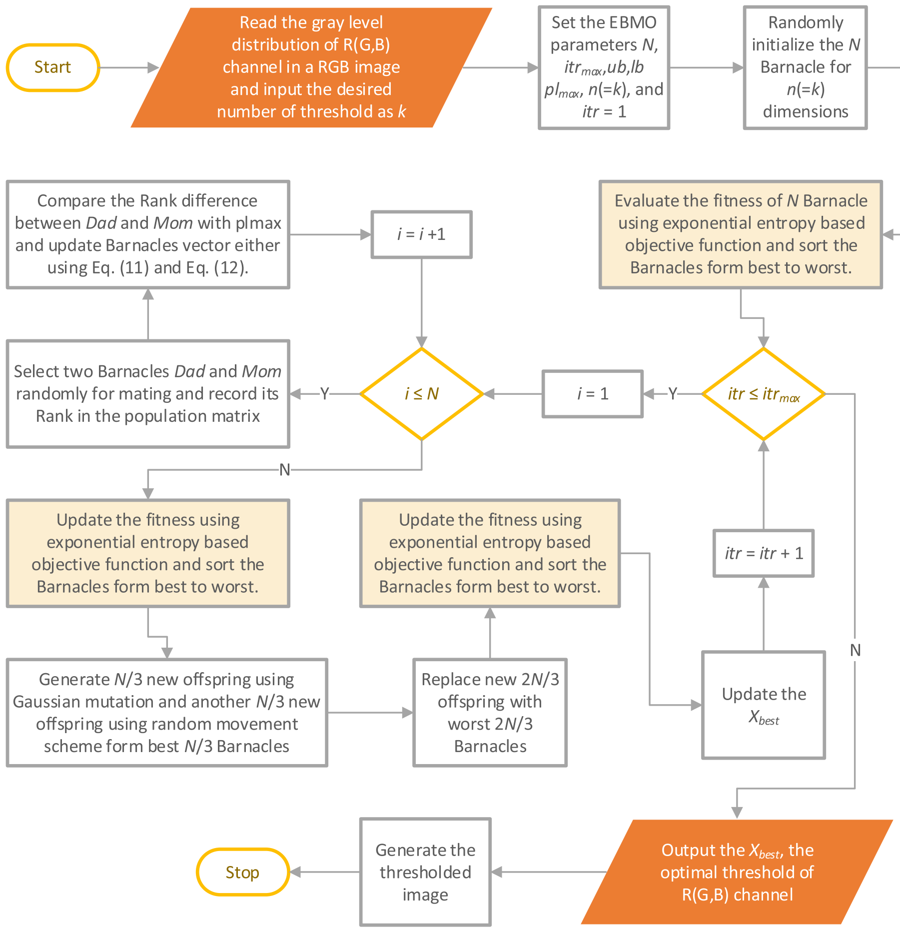


Fig. 9 Flowchart of EBMO-EE based multilevel thresholding

8 GB RAM running on Windows10 environment. Thresholding operations are carried out in two different groups of the images as shown in Figs. 10 and 11 with their corresponding histogram of RGB colour channels. The first group contains six low-dimensional standard images, while the second group contains six high-dimensional images collected using Landsat datasets [39]. The thresholding operation becomes a challenging task when the number of thresholds is more. This instigates us to use most efficient optimizers available so far such as BMO [76], MRFO [85], SMA [54], EO [16], HHO [25], SSA [31], CMA-ES [11], L-SHADE [61] and TSA [36] in comparison analysis. All the above algorithms are applied to each of the test images 31 times independently with a population size of 30 and the number of iterations fixed to 100 to maintain stability. For CMA-ES and L-SHADE equivalent number of evaluations has been taken. Low dimensional images are thresholded with a threshold dimension of 2, 3, and 4, whereas the threshold dimension of 4, 8, and 12 have been selected for high dimensional images

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

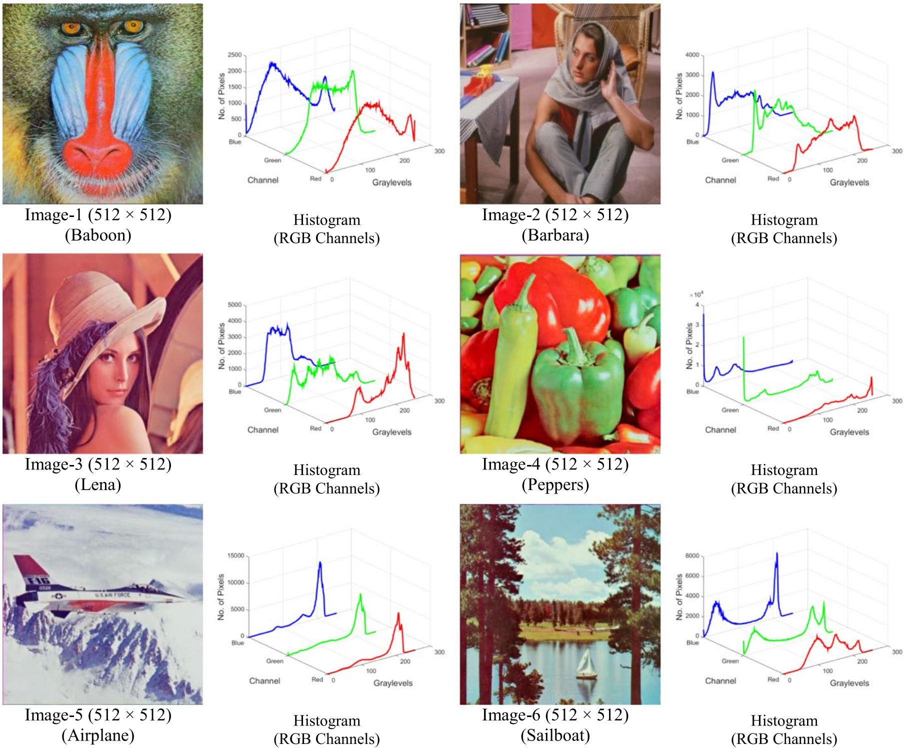


Fig. 10 The group-1 test images and their corresponding histograms

in this experimental study. To evaluate the thresholded images obtained from various algorithms at a different level of thresholding, four performance measures: peak signal to noise ratio (PSNR) [33], Feature Similarity Index Measure (FSIM) [44], Structural Similarity Index Measure (SSIM) [81]and Uniformity Measure [17]. have been calculated and used in a comparative study. A higher value, of PSNR with closer the value of FSIM, SSIM, and UM to 1 is required for ideal thresholding.

Like BMO, the performance of EBMO also affected by the penis length pl_{max} of the barnacle. The decision ongoing for the exploitation or exploration process completely depends upon the penis length. As the enhancement version of EBMO not only enhances the overall exploration but also explores the points close to the best barnacle, a balance of exploration and exploitation must be required for it to reach an optimal solution in less time. Different values of pl_{max} have been tested at a different level of thresholding on a set of low as well as high dimensional images. The effect of pl_{max} to attain the optimal solution of EBMO is shown in Fig. 12 for five different values of pl. The best result can be obtained for $pl=20$. For this comparison analysis, pl_{max} is set to 20 which is nearly 70% of the total barnacle vector in the population.

The thresholding level at low as well as higher-level thresholding of images with varied dimensions is discussed here. At lower-level thresholding, the complexity involved in the threshold's selection process is less because of the presence of well-defined separated regions in the gray level distribution of an image. Whereas, when the number of

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

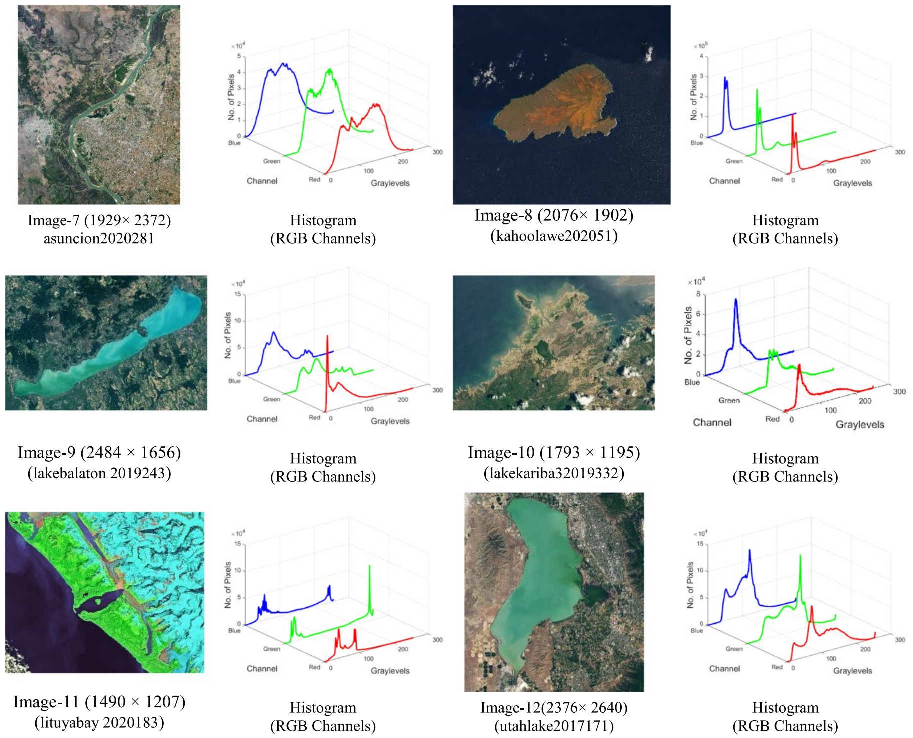


Fig. 11 The group-2 test images and their corresponding histograms

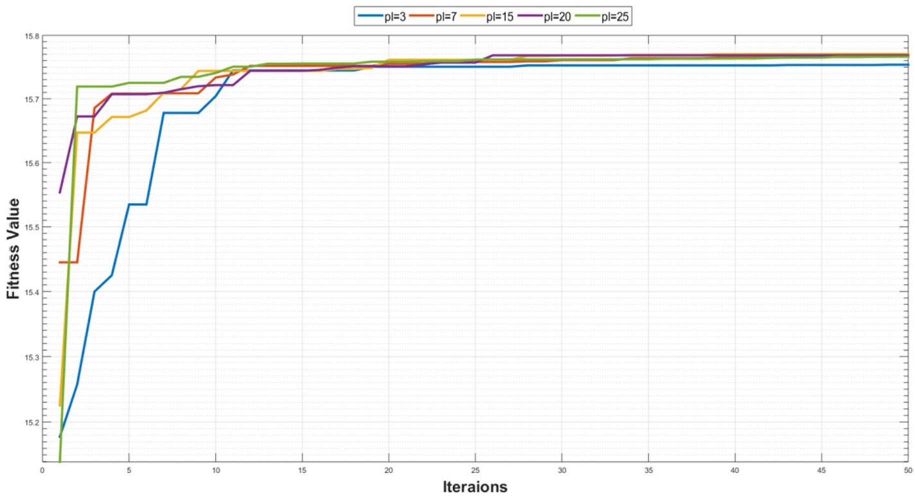


Fig. 12 Effect of penis length (*pl*) in solving multilevel thresholding tasks

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

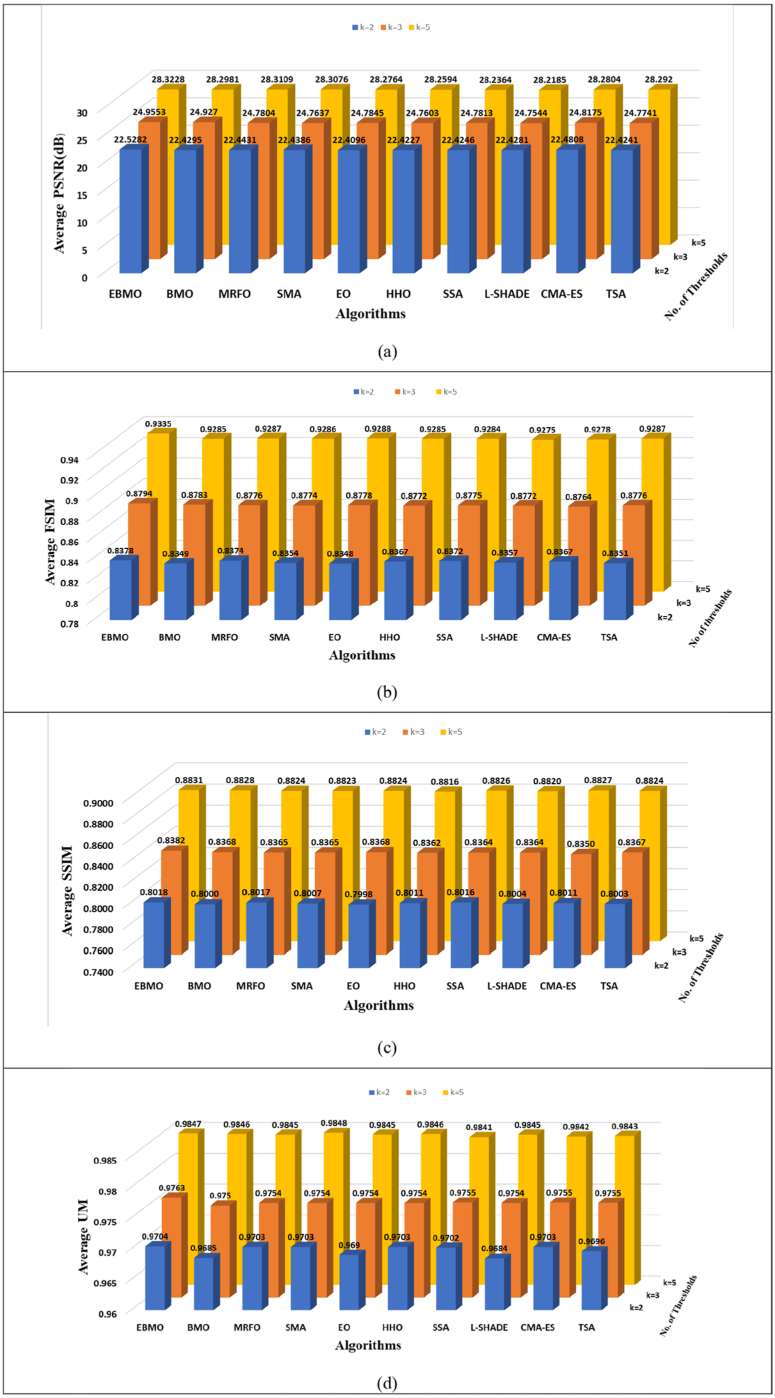


Fig. 13 Performance analysis of various algorithm on group-1 images for (**a**) Average PSNR (**b**) Average FSIM (**c**) Average SSIM and (**d**) Average UM

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

thresholds increases, the optimization algorithms encounter flat regions in the histogram which makes the thresholds selection process more complex. Such a type of complexity can only be overcome by selecting an algorithm having excellent exploration skills. The use of gaussian mutation in EBMO improves population diversity, allowing the algorithm to examine as many threshold combinations as possible. This has a coarse tuning feel to it. A random movement towards the best answer aids EBMO in further exploration around the present best, which is necessary for fine-tuning. The average PSNR, FSIM, SSIM, and UM of six thresholded images obtained from different algorithms when applied to group-1 low dimensional images are shown in Fig. 13 in the form of a bar chart. The results obtained with threshold dimensions of 2, 3, and 5 are represented by three different colours as shown in the figure. It can be observed that when the number of thresholds, $k=2$, almost all algorithms except EBMO show almost similar performance for the PSNR values. After a close fight with MRFO and SMA algorithms, EBMO produces a superior segmented image with a maximum average PSNR of 22.5282 due to exploring more threshold combinations in a short time. When the number of thresholds increases from 2 to 5, the performance of different algorithms starts varying. To assess how well the features of the images are retained in the segmented images, the paper also presented the FSIM values obtained after the application of different optimization algorithms. It can be observed that the proposed EBMO based thresholding produced a better segmented image with less feature information loss by attending higher FSIM values. Perceptual differences between original and segmented images are further measured through SSIM. For SSIM values, EBMO has a significant improvement over its counterpart BMO when $k=2$. Because of the excellent exploration skill of EBMO, the perceptual difference found in the thresholded results associated with it is less to the original image. To carefully examine segmentation results in this paper one more similarity measure UM values obtained from the thresholded images are also included in the result section. Based on the experimental results of obtained UM values, EBMO dominates other examined algorithms for $k=2$ and $k=3$ but lags marginally from the thresholding performance of SMA.

The convergence plots of different optimization algorithms taken for comparative analysis for Lena image at the threshold dimension 5 are displayed in Fig. 14. All the optimization algorithms are applied independently to the image for maximizing the exponential entropy and the number of iterations required to reach the optimized value is recorded. By

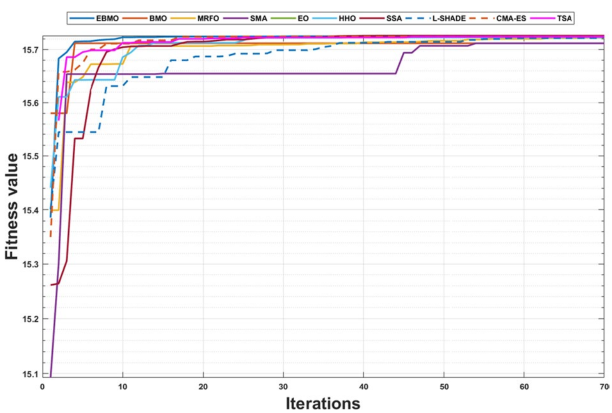


Fig. 14 Convergence curves of various optimization algorithms for Lena Image at a thresholding dimension of $k=5$

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Fig. 15 Performance analysis of various algorithms on group-2 images for (**a**) Average PSNR (**b**) Average FSIM (**c**) Average SSIM and (**d**) Average UM ▸

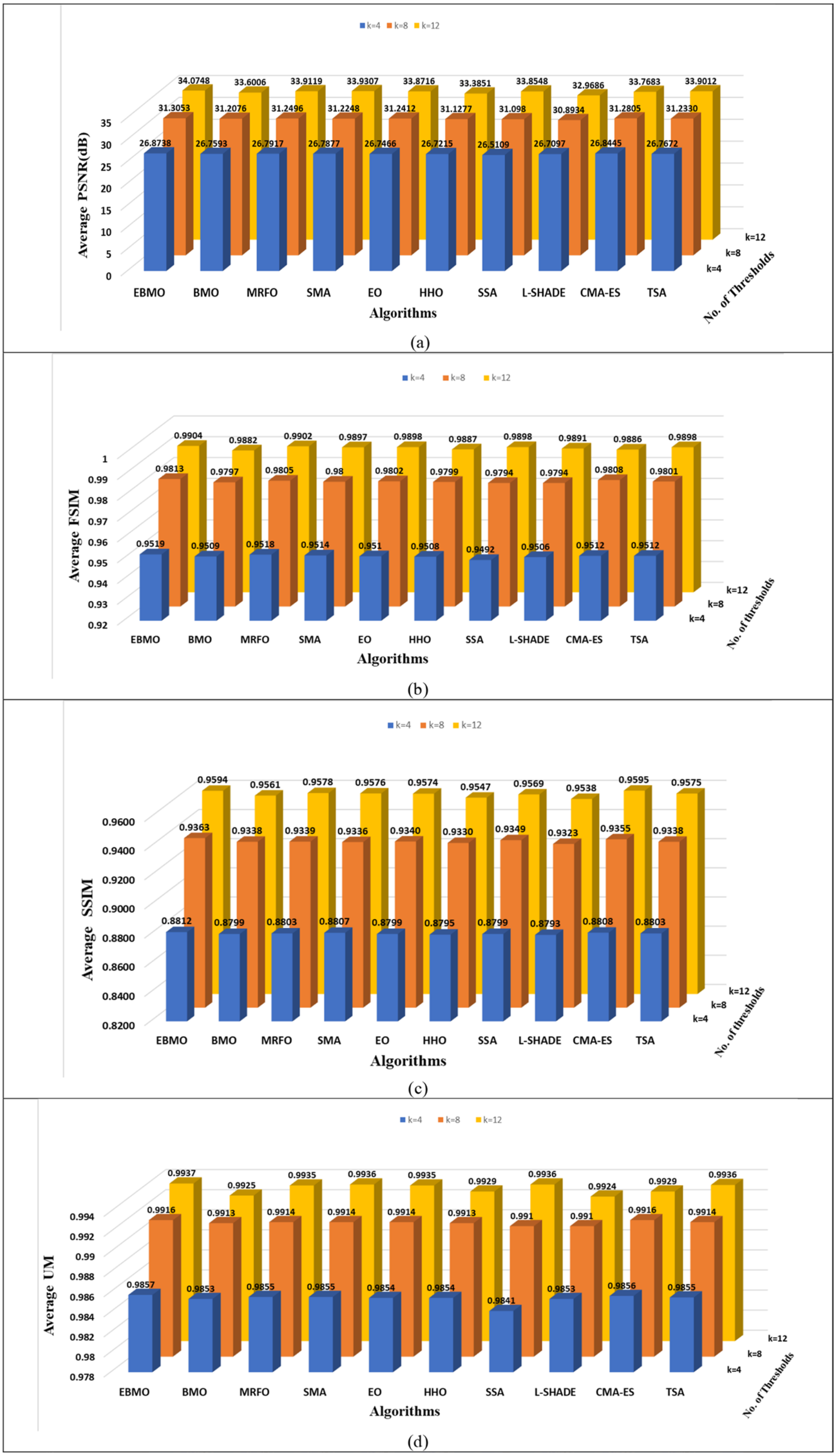
analysing the convergence plot, it can be observed that the additional mutation strategy included in EBMO helps it to converge fast with its superior exploration capability.

To observe the thresholding performance of proposed methods on high dimensional images and a higher level of thresholding, a comparative analysis with the above performance measures for remote sensing mages taken from Group-2 also has been performed with a threshold dimension of 4, 8, and 12. The proposed multilevel thresholding approach is put to the test on remote sensing photos to ensure that it can be used in practice. The results obtained are presented in Fig. 15 with the help of bar chat. The improvement over BMO due to the proposed modification can be observed from the experimental results, where it beats BMO significantly with a higher value of average PSNR, FSIM, SSIM and UM. After a close race with MRFO and SMA-based thresholding, EBMO has reclaimed first place in terms of PSNR value. A better FSIM value achieved by the proposed method shows that the desired information can be collected from the threshold image. The comparison results of SSIM values indicate that there is a close fight between EBMO and CMA-ES algorithm toward retaining the structural similarity. Finally, the largest UM values associated with suggested EBMO-based thresholding output make it a feasible method for the analysis of higher-dimensional remote sensing images.

To make a visual judgment and in-depth investigation, three images from each group are taken for further study. To encourage readers, these examples suffice for analysis and interpretation. As the best performance measures value can be obtained by retaining maximum contextual information and the inherent characteristics of the original image in the output image, the quality of thresholded images is proportional to the information preserving ability by an optimization algorithm during threshold selection. Thresholded images of one low dimensional image and their corresponding histograms obtained from various optimization algorithms are presented in Fig. 16. To identify the improvement introduced by the proposed algorithm, the best PSNR, FSIM, SSIM, and UM values associated with each output image are also given in the figure. It can be observed, as the thresholded images are taken for $k=3$, there is very close competition between different algorithms. Whereas a significant improvement can be observed for higher-level thresholding. Figure 17 shows the results of different algorithms for one high-dimensional image at $k=12$. The histogram of each colour channel is represented by the same colour. It is clear from the result that the proposed thresholding approach dominates all and can be claimed as a suitable solution for the analysis of real-time images.

At the end of the section, the thresholded images obtained from a few more low and high-dimensional test images at a different level of thresholding using different algorithms are presented in Appendix Figs. 18, 19, 20 and 21. The combined histogram associated with different thresholding levels using EBMO with exponential entropy as the fitness function is also presented for Group-1 and Group-2 images in Appendix Figs. 22 and 23. It can be seen that; the proposed exponential entropy-based multilevel thresholding using EBMO can perform multilevel thresholding tasks efficiently for low and high dimensional images at a different level of thresholding.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.



Content courtesy of Springer Nature, terms of use apply. Rights reserved.

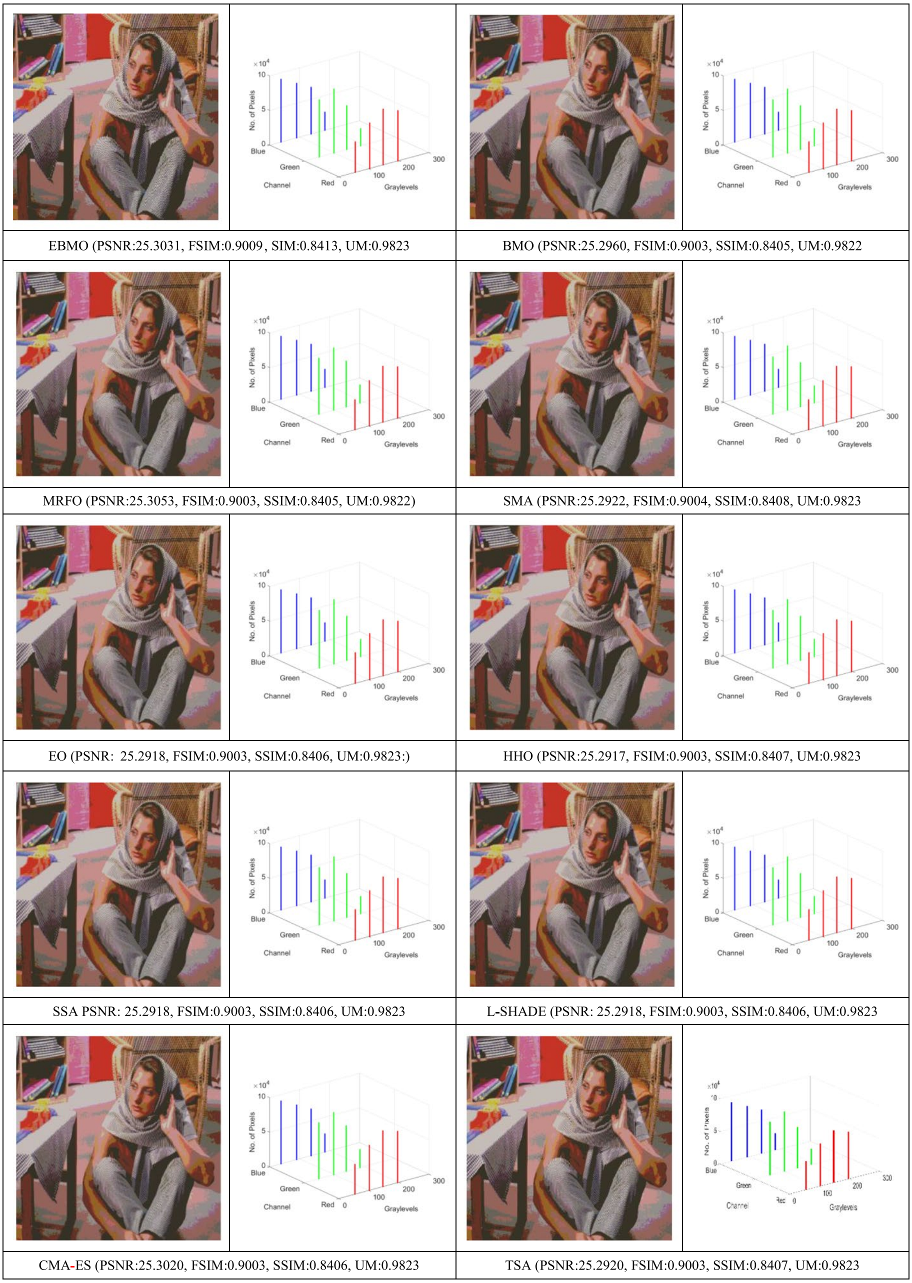


Fig. 16 4-level thresholded images ($k = 3$) and their corresponding histograms

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

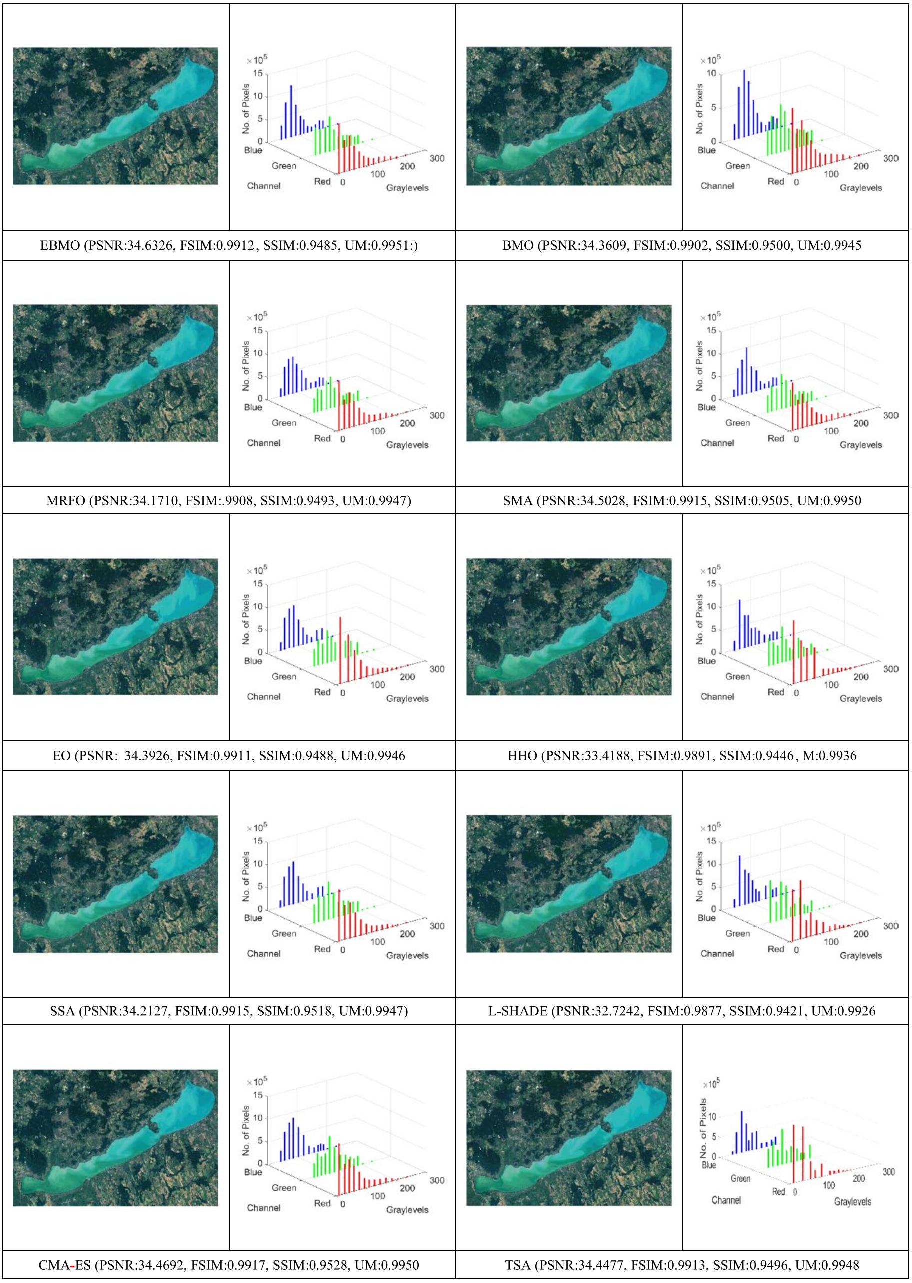


Fig. 17 13-level thresholded images ($k = 12$) and their corresponding histograms

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

7 Conclusion

This work proposes a new exponential entropy-based multilevel thresholding algorithm by introducing a first-hand search algorithm called Enhance Barnacle Mating optimization (EBMO). After an in-depth investigation of the performance of EBMO, it is found appropriate for the proposed thresholding method. The EBMO algorithm is the enhancement of the recently developed Barnacle Mating Optimization (BMO) algorithm by incorporating an additional mutation strategy and a random movement towards the best solution. The addition of these new strategies makes the Barnacles explore all possible regions which help them to avoid local minima in their search path and move fast towards the optimal solution. The qualitative and quantitative results of the EBMO are compared to well-known optimization methods BMO, MRFO, SMA, EO, HHO, SSA, L-SHADE, CMA-ES, and TSA over different benchmark test functions, which reveals that EBMO surpasses other optimization techniques. The convergence rate of the EBMO is also found quite impressive than other state-of-the-art methods.

In this work, we have investigated an exponential entropy-based technique for multilevel thresholding using EBMO (EBMO-EE), which considers the exponential gain function to estimate the information available in the thresholded image. Unlike logarithmic gain-based entropy, exponential entropy can be calculated with a finite value at all possible points. Different optimization algorithms have been applied to maximize this entropy function, including the proposed EBMO. The statistical comparison using the average PSNR, FSIM, SSIM, and UM presented in the paper claims that the exponential entropy can be taken as a suitable fitness function to perform thresholding. The EBMO is an efficient algorithm to optimize it at a different level of thresholding. The future scope of the work includes multilevel thresholding in a noisy environment. Though the performance of EBMO is superior over many state-of-art algorithms, a further improvement of the exploration and exploitation skill of the algorithm can be achieved by hybridization with an efficient optimization algorithm. The efficiency of the suggested multilevel thresholding approaches can also be improved by taking into account the spatial correlation between pixels, which is not taken into account in this study. To conclude, we believe that the proposed method for the image segmentation based on computational intelligence for intelligent healthcare applications will be found to be satisfactory.

Appendix

Table 9 Unimodal test function

Function	d	Range	f_{min}
$f_1(X) = \sum_{i=1}^{d} x_i^2$	30, 100	$[-100, 100]^d$	0
$f_2(X) = \sum_{i=1}^{d} \lvert x_i \rvert + \prod_{i=1}^{d} \lvert x_i \rvert$	30, 100	$[-10, 10]^d$	0
$f_3(X) = \sum_{i=1}^{d} \left(\sum_{j=1}^{i} x_j \right)^2$	30, 100	$[-100, 100]^d$	0
$f_4(X) = \max_i \left\{ \lvert x_i \rvert, 1 \le i \le d \right\}$	30, 100	$[-100, 100]^d$	0
$f_5(X) = \sum_{i=1}^{d-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2 \right]$	30, 100	$[-30, 30]^d$	0
$f_6(X) = \sum_{i=1}^{d} \left(\lfloor x_i + 0.5 \rfloor \right)^2$	30, 100	$[-100, 100]^d$	0
$f_7(X) = \sum_{i=1}^{d} i x_i^4 + random[0, 1)$	30, 100	$[-1.28, 1.28]^d$	0

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 10 Scalable dimension multimodal test function

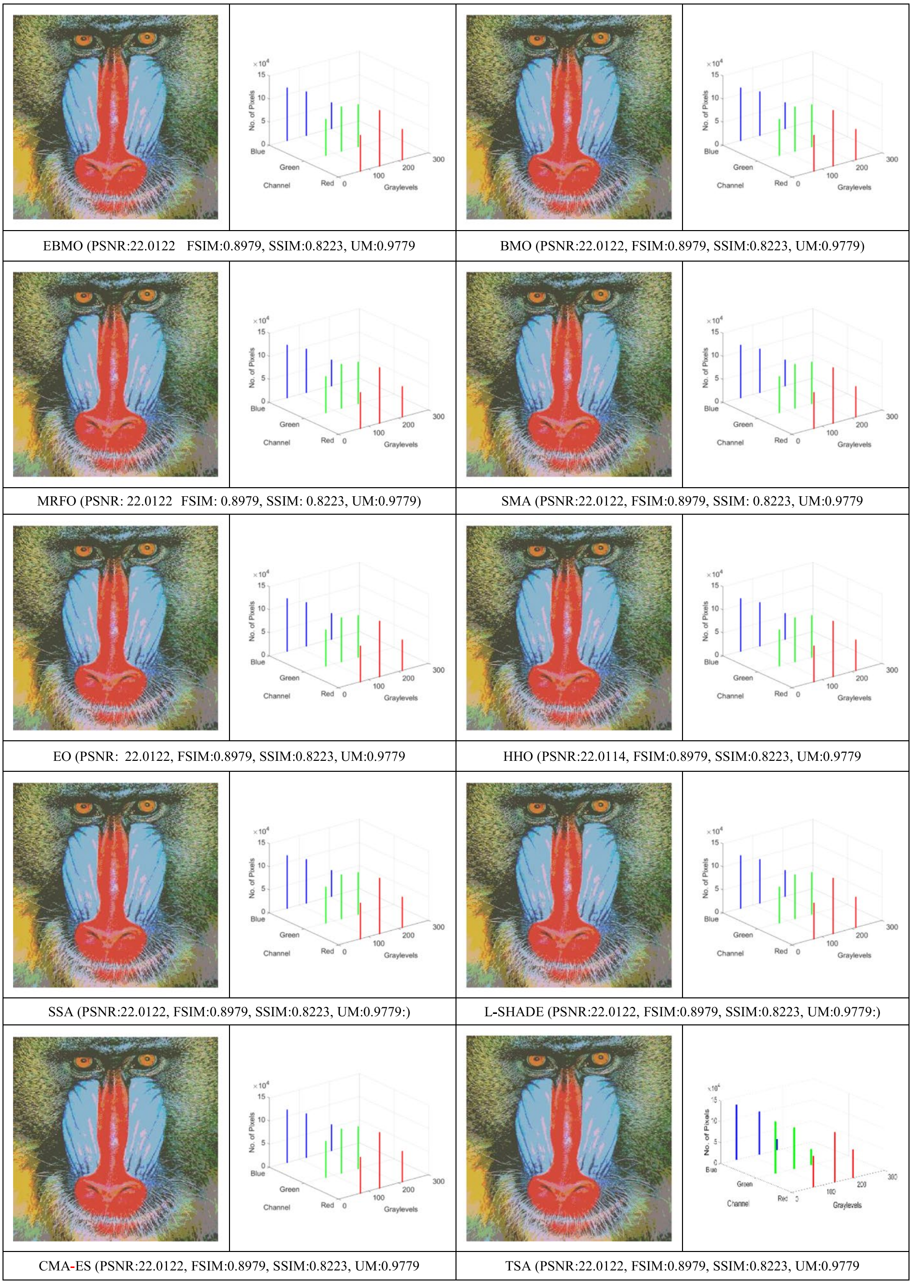
Function	d	Range	f_{min}
$f_8(X) = \sum_{i=1}^{d} \left[x_i^2 - 10\cos\left(2\pi x_i\right) + 10\right]$	30, 100	$[-5.12, 5.12]^d$	0
$f_9(X) = -20\exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos\left(2\pi x_i\right)\right) + 20 + e$	30, 100	$[-32, 32]^d$	0
$f_{10}(X) = \frac{1}{4000}\sum_{i=1}^{d} x_i^2 - \prod_{i=1}^{d}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30, 100	$[-600, 600]^d$	0
$f_{11}(X) = \frac{\pi}{d}\left\{10\sin\left(\pi y_i\right) + \sum_{i=1}^{d-1}\left(y_i - 1\right)^2\left[1 + 10\sin^2\left(\pi y_{i+1}\right)\right] + \left(y_d - 1\right)^2\right\} + \sum_{i=1}^{d} u\left(x_i, 10, 100, 4\right)$ $y_i = 1 + \frac{x_i+1}{4}$ $u\left(x_i, a, k, m\right) = \begin{cases} k\left(x_i - a\right)^m & x_i > a \\ 0 & -a < x_i < a \\ k\left(-x_i - a\right)^m & x_i < -a \end{cases}$	30, 100	$[-50, 50]^d$	0
$f_{12}(X) = 0.1\left\{\sin^2\left(3\pi x_1\right) + \sum_{i=1}^{d}\left(x_i - 1\right)^2\left[1 + \sin^2\left(3\pi x_i + 1\right)\right] + \left(x_d - 1\right)^2\left[1 + \sin^2\left(2\pi x_d\right)\right]\right\} + \sum_{i=1}^{d} u\left(x_i, 5, 100, 4\right)$ $y_i = 1 + \frac{x_i+1}{4}$ $u\left(x_i, a, k, m\right) = \begin{cases} k\left(x_i - a\right)^m & x_i > a \\ 0 & -a < x_i < a \\ k\left(-x_i - a\right)^m & x_i < -a \end{cases}$	30, 100	$[-50, 50]^d$	0

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 11 Fixed dimension multimodal test function

Function	Range	f_{min}
$f_{13}(X)=\left(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^{2}\left(x_i-a_{ij}\right)^6}\right)^{-1}$	$\begin{bmatrix}-65.536,\\ 65.536\end{bmatrix}^2$	1
$f_{14}(X)=\sum_{i=1}^{11}\left[a_i-\frac{x_1\left(b_i^2+b_ix_2\right)}{b_i^2+b_ix_3+x_4}\right]^2$	$[-5,5]^4$	0.0003075
$f_{15}(X)=4x_1^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2-4x_2^2+4x_2^4$	$[-5,5]^2$	−1.0316285
$f_{16}(X)=\left(x_2-\frac{5.1}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6\right)^2+10\left(1-\frac{1}{8\pi}\right)\cos x_1+10$	$[-5,10]\times[0,15]$	0.398
$f_{17}(X)=\left[1+\left(x_1+x_2+1\right)^2\times\left(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2\right)\right]$ $\times\left[30+\left(2x_1-3x_2\right)^2\times\left(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2\right)\right]$	$[-2,2]^2$	3
$f_{18}(X)=-\sum_{i=1}^{4}c_i\exp\left(-\sum_{j=1}^{3}a_{ij}\left(x_j-p_{ij}\right)^2\right)$	$[0,1]^3$	−3.86
$f_{19}(X)=-\sum_{i=1}^{4}c_i\exp\left(-\sum_{j=1}^{6}a_{ij}\left(x_j-p_{ij}\right)^2\right)$	$[0,1]^6$	−3.32
$f_{20}(X)=-\sum_{i=1}^{5}\left[\left(X-a_i\right)\left(X-a_i\right)^T+c_i\right]^{-1}$	$[0,10]^4$	−10.1532
$f_{21}(X)=-\sum_{i=1}^{7}\left[\left(X-a_i\right)\left(X-a_i\right)^T+c_i\right]^{-1}$	$[0,10]^4$	−10.4028
$f_{22}(X)=-\sum_{i=1}^{10}\left[\left(X-a_i\right)\left(X-a_i\right)^T+c_i\right]^{-1}$	$[0,10]^4$	−10.5363

Content courtesy of Springer Nature, terms of use apply. Rights reserved.



EBMO (PSNR:22.0122 FSIM:0.8979, SSIM:0.8223, UM:0.9779

BMO (PSNR:22.0122, FSIM:0.8979, SSIM:0.8223, UM:0.9779)

MRFO (PSNR: 22.0122 FSIM: 0.8979, SSIM: 0.8223, UM:0.9779)

SMA (PSNR:22.0122, FSIM:0.8979, SSIM: 0.8223, UM:0.9779

EO (PSNR: 22.0122, FSIM:0.8979, SSIM:0.8223, UM:0.9779

HHO (PSNR:22.0114, FSIM:0.8979, SSIM:0.8223, UM:0.9779

SSA (PSNR:22.0122, FSIM:0.8979, SSIM:0.8223, UM:0.9779:)

L-SHADE (PSNR:22.0122, FSIM:0.8979, SSIM:0.8223, UM:0.9779:)

CMA-ES (PSNR:22.0122, FSIM:0.8979, SSIM:0.8223, UM:0.9779

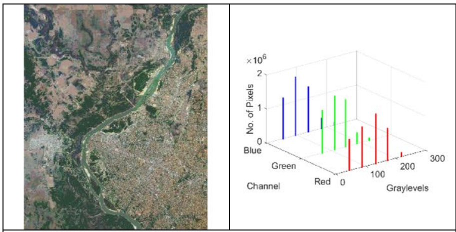
TSA (PSNR:22.0122, FSIM:0.8979, SSIM:0.8223, UM:0.9779

Fig. 18 3-level thresholded images ($k=2$) and their corresponding histograms

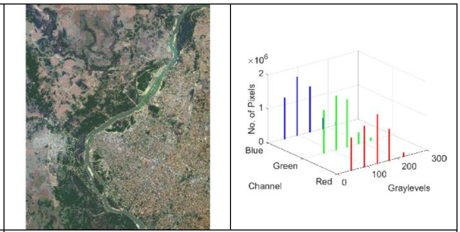
Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Fig. 19 6-level thresholded images (k = 5) and their corresponding histograms

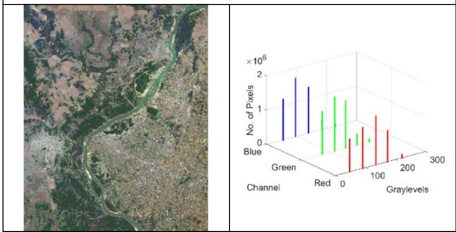
Content courtesy of Springer Nature, terms of use apply. Rights reserved.



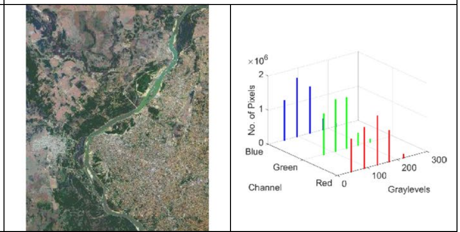
EBMO (PSNR: 27.2392. FSIM:0.9696, SSIM:0.8952, UM:0.9864



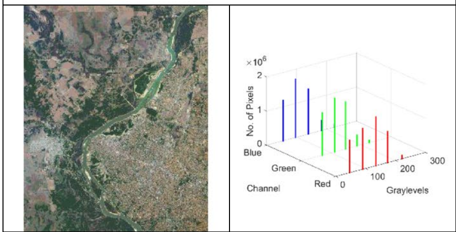
BMO (PSNR:27.0009, FSIM:0.9684, SSIM:0.8862, UM:0.9858



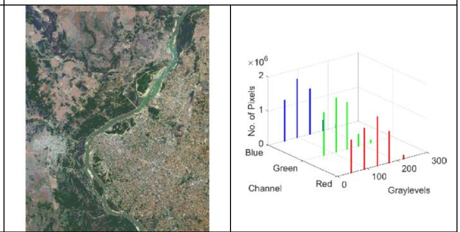
MRFO (PSNR: 27.1406, FSIM:0.9713, SSIM:0.8856, UM:0.9857



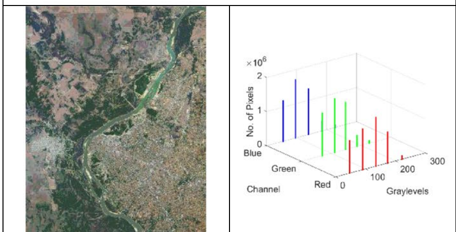
SMA (PSNR:27.1680, FSIM:0.9691, SSIM:0.8881, UM:0.9859



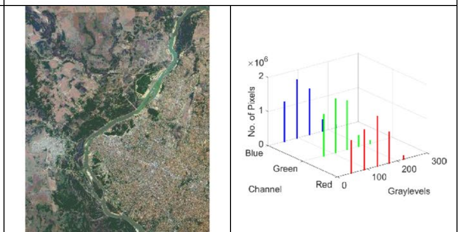
EO (PSNR: 27.0407, FSIM:0.9672, SSIM:0.8817, UM:0.9854



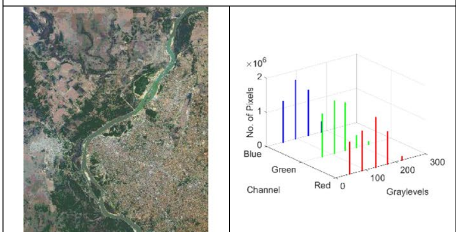
HHO (PSNR:27.0012, FSIM:0.9668, SSIM:0.8787, UM:0.9852



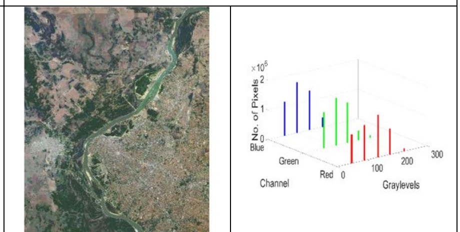
SSA (PSNR:27.0399, FSIM:0.9673, SSIM:0.8817, UM:0.9854



L-SHADE (PSNR:26.9491, FSIM:0.9659, SSIM:.8769, UM:0.9851



CMA-ES (PSNR:27.2000, FSIM:0.9686, SSIM:0.8885, UM:0.9860



TSA (PSNR:27.1044, FSIM:0.9681, SSIM:0.8849, UM:0.9857

Fig. 20 5-level thresholded images (k = 4) and their corresponding histograms

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

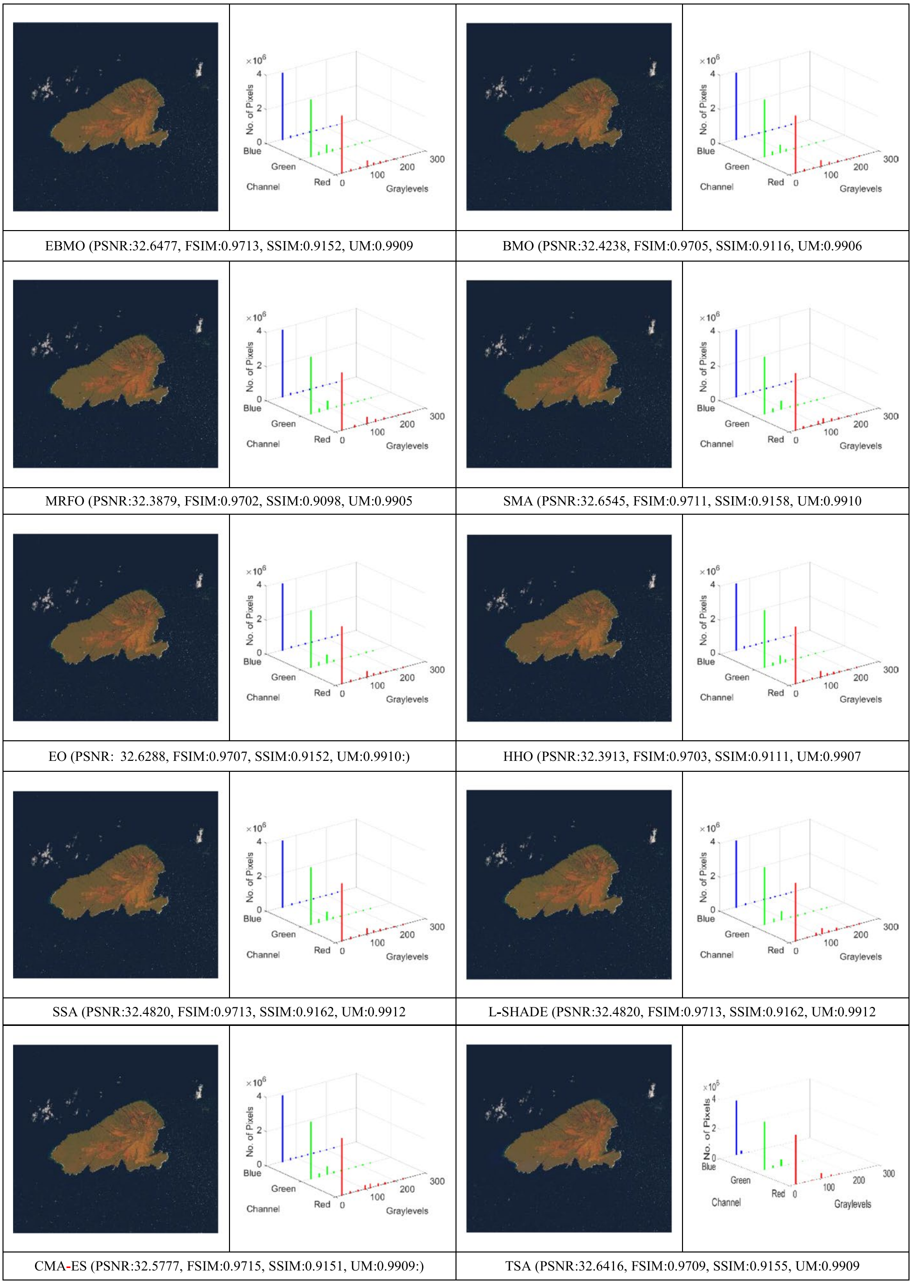


Fig. 21 9-level thresholded images (k = 8) and their corresponding histograms

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

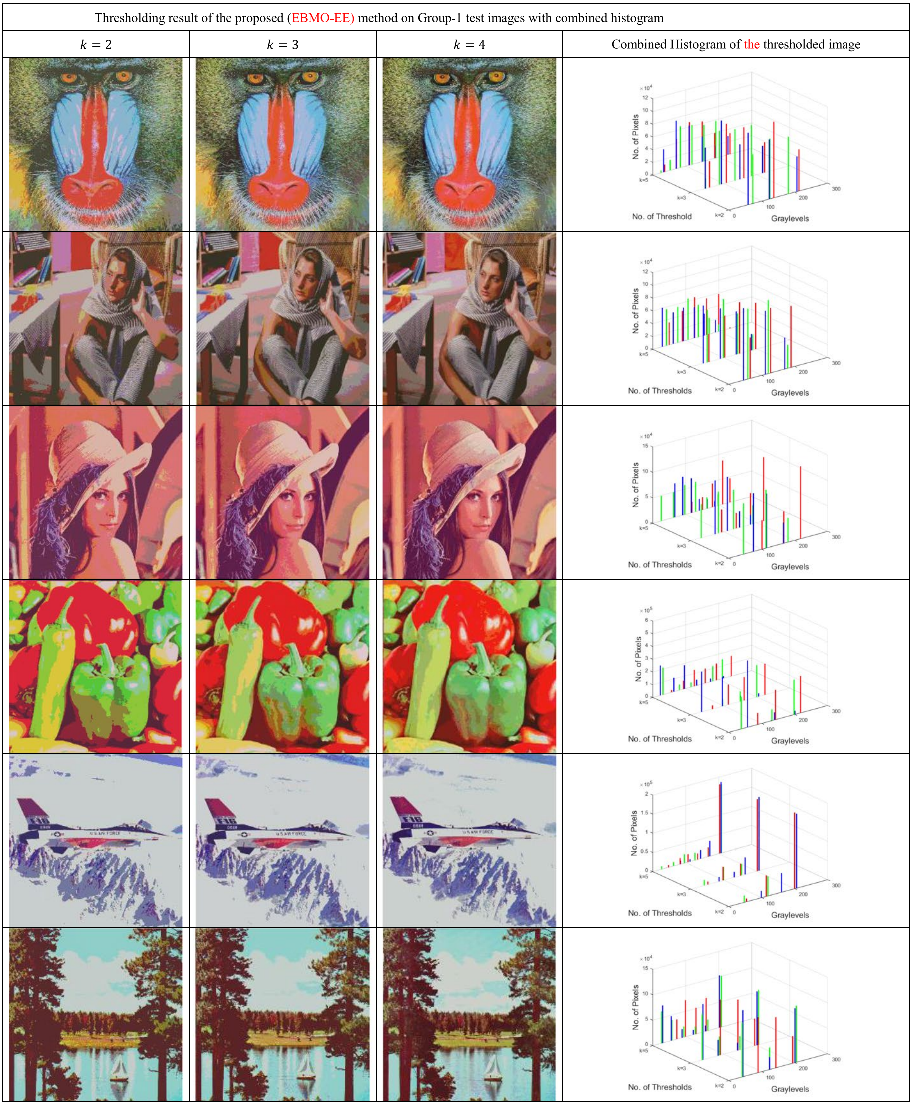


Fig. 22 Thresholding result of the proposed (EBMO-EE) method on Group-1 test images with combined histogram

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

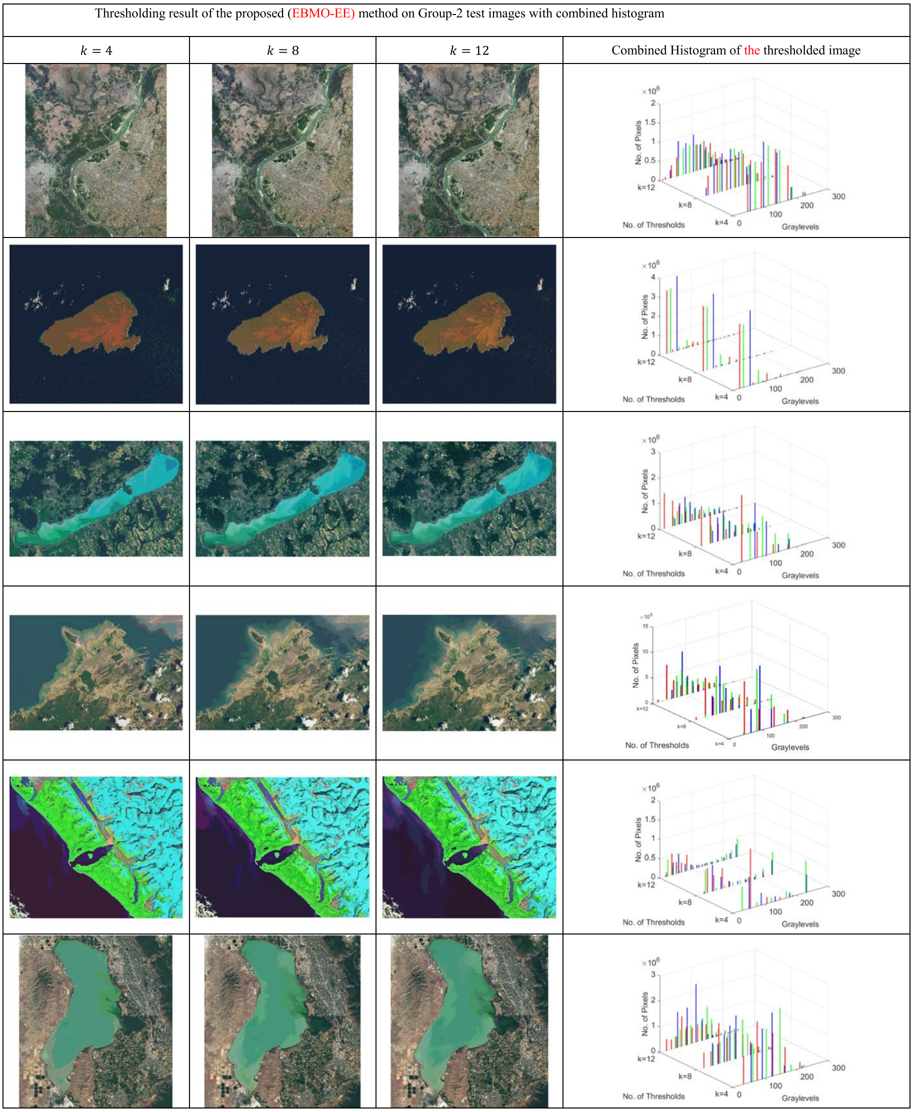


Fig. 23 Thresholding result of the proposed (EBMO-EE) method on Group-2 test images with combined histogram

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Declarations

Conflict of interest According to the authors, they have no known competing financial interests.

References

1. Abd Elaziz M, Nabil N, Moghdani R, et al (2021) Multilevel thresholding image segmentation based on improved volleyball premier league algorithm using whale optimization algorithm. Multimed Tools Appl
2. Abdel-Basset M, Chang V, Mohamed R (2021) A novel equilibrium optimization algorithm for multi-thresholding image segmentation problems. Springer, London
3. Agrawal S, Panda R, Bhuyan S, Panigrahi BK (2013) Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm. Swarm Evol Comput 11:16–30. https://doi.org/10.1016/j.swevo.2013.02.001
4. Agrawal S, Leena Samantaray RP, Dora L (2020) A new hybrid adaptive cuckoo search-squirrel search algorithm for brain MR image analysis, Hybrid Mac. Springer, Singapore
5. Ameur M, Habba M, Jabrane Y (2019) A comparative study of nature inspired optimization algorithms on multilevel thresholding image segmentation. Multimed Tools Appl 78:34353–34372. https://doi.org/10.1007/s11042-019-08133-8
6. Askarzadeh A (2016) A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput Struct 169:1–12. https://doi.org/10.1016/j.compstruc.2016.03.001
7. Bäck T, Schwefel H-P (1993) An overview of evolutionary algorithms for parameter optimization. Evol Comput 1:1–23. https://doi.org/10.1162/evco.1993.1.1.1
8. Bakhshali MA, Shamsi M (2014) Segmentation of color lip images by optimal thresholding using bacterial foraging optimization (BFO). J Comput Sci 5:251–257. https://doi.org/10.1016/j.jocs.2013.07.001
9. Chen Wei FK (2008) A multilevel thresholding algorithm for image segmentation based on particle swarm optimization. Proceedings of the 27th Chinese control conference 0:16–18. https://doi.org/10.1109/AICCSA.2016.7945752
10. Chouksey M, Jha RK, Sharma R (2020) A fast technique for image segmentation based on two Meta-heuristic algorithms. Multimed Tools Appl 79:19075–19127. https://doi.org/10.1007/s11042-019-08138-3
11. Dang VH, Vien NA, Chung TC (2019) A covariance matrix adaptation evolution strategy in reproducing kernel Hilbert space. Genet Program Evolvable Mach 20:479–501. https://doi.org/10.1007/s10710-019-09357-1
12. Das S, Biswas A, Dasgupta S, Abraham A (2009) Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications. Stud Comput Intell 203:23–55. https://doi.org/10.1007/978-3-642-01085-9_2
13. Dhiman G (2021) ESA: a hybrid bio-inspired metaheuristic optimization approach for engineering problems. Eng Comput 37:323–353. https://doi.org/10.1007/s00366-019-00826-w
14. Dhiman G, Kumar V (2018) Emperor penguin optimizer: a bio-inspired algorithm for engineering problems. Knowl-Based Syst 159:20–50. https://doi.org/10.1016/j.knosys.2018.06.001
15. Eberhart R, Kennedy J (1995) New optimizer using particle swarm theory. Proceedings of the International Symposium on Micro Machine and Human Science 39–43. https://doi.org/10.1109/mhs.1995.494215
16. Faramarzi A, Heidarinejad M, Stephens B, Mirjalili S (2020) Equilibrium optimizer: A novel optimization algorithm. Knowl-Based Syst 191. https://doi.org/10.1016/j.knosys.2019.105190
17. Feng Y, Zhao H, Li X et al (2017) A multi-scale 3D Otsu thresholding algorithm for medical image segmentation. Digit Signal Process A Rev J 60:186–199. https://doi.org/10.1016/j.dsp.2016.08.003
18. Freixenet J, Muñoz X, Raba D, et al (2002) Yet another survey on image segmentation: region and boundary information integration. In: Heyden A, Sparr G, Nielsen M, Johansen P (eds) Computer Vision --- ECCV 2002. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 408–422
19. Fu KS, Mui JK (1981) A survey on image segmentation. Pattern Recogn 13:3–16. https://doi.org/10.1016/0031-3203(81)90028-5
20. Gadekallu TR, Alazab M, Kaluri R et al (2021) Hand gesture classification using a novel CNN-crow search algorithm. Complex Intell Syst 7:1855–1868. https://doi.org/10.1007/s40747-021-00324-x
21. Gandomi AH, Alavi AH (2012) Krill herd: a new bio-inspired optimization algorithm. Commun Nonlinear Sci Numer Simul 17:4831–4845. https://doi.org/10.1016/j.cnsns.2012.05.010
22. Gandomi AH, Yang XS, Alavi AH (2013) Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng Comput 29:17–35. https://doi.org/10.1007/s00366-011-0241-y

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

23. Gonzalez RC, Woods RE (2006) Digital image processing (3rd Edition). Prentice-Hall, Inc., USA
24. Guo SW, Thompson EA (1992) Performing the exact test of hardy-Weinberg proportion for multiple alleles. Biometrics 48:361–372. https://doi.org/10.2307/2532296
25. Heidari AA, Mirjalili S, Faris H et al (2019) Harris hawks optimization: algorithm and applications. Futur Gener Comput Syst 97:849–872. https://doi.org/10.1016/j.future.2019.02.028
26. Hoang N (2020) A multilevel image thresholding approach using history-based adaptive a multilevel image thresholding approach using history-based adaptive differential evolution with linear population size reduction algorithm. DTU journal of science nad Technology
27. Holland JH (1992) Genetic algorithms. Sci Am 267:66–73
28. Holm S (1979) A simple sequentially Rejective multiple test procedure. Scand J Stat 6:65–70. https://www.jstor.org/stable/4615733
29. Horng MH (2011) Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation. Expert Syst Appl 38:13785–13791. https://doi.org/10.1016/j.eswa.2011.04.180
30. Horng MH, Liou RJ (2011) Multilevel minimum cross entropy threshold selection based on the firefly algorithm. Expert Syst Appl 38:14805–14811. https://doi.org/10.1016/j.eswa.2011.05.069
31. Jain M, Singh V, Rani A (2019) A novel nature-inspired algorithm for optimization: squirrel search algorithm. Swarm Evol Comput 44:148–175. https://doi.org/10.1016/j.swevo.2018.02.013
32. Jena B, Naik MK, Wunnava A, Panda R (2019) A comparative study on multilevel thresholding using metaheuristic algorithm. In: proceedings - 2019 international conference on applied machine learning, ICAML 2019
33. Jia H, Ma J, Song W (2019) Multilevel thresholding segmentation for color image using modified moth-flame optimization. IEEE Access 7:44097–44134. https://doi.org/10.1109/ACCESS.2019.2908718
34. Johari NF, Zain AM, Noorfa MH, Udin A (2013) Firefly algorithm for optimization problem. In: Information Technology for Manufacturing Systems IV. Trans Tech Publications Ltd, pp. 512–517
35. Kapur JN, Sahoo PK, Wong AKC (1985) A new method for gray-level picture thresholding using the entropy of the histogram. Comput Vis Graph Image Process 29:273–285. https://doi.org/10.1016/0734-189X(85)90125-2
36. Kaur S, Awasthi LK, Sangal AL, Dhiman G (2020) Tunicate swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization. Eng Appl Artif Intell 90:103541. https://doi.org/10.1016/j.engappai.2020.103541
37. Khairuzzaman AKM, Chaudhury S (2019) Masi entropy based multilevel thresholding for image segmentation. Multimed Tools Appl 78:33573–33591. https://doi.org/10.1007/s11042-019-08117-8
38. Kumar SN, Lenin Fred A, Kumar AH, et al (2020) Multilevel thresholding using crow search optimization for medical images. Comput Intell 231–258. https://doi.org/10.1515/9783110671353-014
39. Landsat Image Gallery [WWW Document] (n.d.)
40. Li L, Sun L, Guo J et al (2017) Modified discrete Grey wolf optimizer algorithm for multilevel image thresholding. Comput Intell Neurosci 2017:3295769. https://doi.org/10.1155/2017/3295769
41. Li S, Chen H, Wang M et al (2020) Slime mould algorithm: a new method for stochastic optimization. Futur Gener Comput Syst 111:300–323. https://doi.org/10.1016/j.future.2020.03.055
42. Liang JJ, Qu BY, Suganthan PN (2013) Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization
43. Liang H, Jia H, Xing Z et al (2019) Modified grasshopper algorithm-based multilevel thresholding for color image segmentation. IEEE Access 7:11258–11295. https://doi.org/10.1109/ACCESS.2019.2891673
44. Lin Zhang LZ (2011) FSIM: a feature similarity index for image quality assessment. IEEE Trans Image Process 20:2378. https://doi.org/10.1109/TIP.2011.2109730
45. Liu S-H, Mernik M, Hrnčič D, Črepinšek M (2013) A parameter control method of evolutionary algorithms using exploration and exploitation measures with a practical application for fitting Sovova's mass transfer model. Appl Soft Comput 13:3792–3805. https://doi.org/10.1016/j.asoc.2013.05.010
46. Liu Y, Sun J, Yu H, et al (2020) An improved Grey wolf optimizer based on differential evolution and OTSU algorithm. Appl Sci 10:. https://doi.org/10.3390/app10186343
47. Masi M (2005) A step beyond Tsallis and Rényi entropies. Phys Lett Section A Gen Atom Solid State Phys 338:217–224. https://doi.org/10.1016/j.physleta.2005.01.094
48. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008
49. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
50. Mirjalili S, Gandomi AH, Mirjalili SZ et al (2017) Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv Eng Softw 114:163–191. https://doi.org/10.1016/j.advengsoft.2017.07.002
51. Mlakar U, Potočnik B, Brest J (2016) A hybrid differential evolution for optimal multilevel image thresholding. Expert Syst Appl 65:221–232. https://doi.org/10.1016/j.eswa.2016.08.046

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

52. Moghdani R, Salimifard K (2018) Volleyball premier league algorithm. Appl Soft Comput J 64:161–185. https://doi.org/10.1016/j.asoc.2017.11.043
53. Naik MK, Panda R (2016) A novel adaptive cuckoo search algorithm for intrinsic discriminant analysis based face recognition. Appl Soft Comput J 38:661–675. https://doi.org/10.1016/j.asoc.2015.10.039
54. Naik MK, Panda R, Abraham A (2020) Normalized square difference based multilevel thresholding technique for multispectral images using leader slime mould algorithm. J King Saud Univ Comput Inf Sci https://doi.org/10.1016/j.jksuci.2020.10.030
55. Naik MK, Panda R, Wunnava A, et al (2021) A leader Harris hawks optimization for 2-D Masi entropy-based multilevel image thresholding. Multimed Tools Appl https://doi.org/10.1007/s11042-020-10467-7
56. Otsu (1979) Otsu_1979_otsu_method. IEEE Trans Syst Man Cybern C:62–66. https://doi.org/10.1109/TSMC.1979.4310076
57. Pal NR, Pal SK (1989) Object-background segmentation using new definitions of entropy. IEE Proceed E: Comput Digit Tech 136:284–295. https://doi.org/10.1049/ip-e.1989.0039
58. Pal NR, Pal SK (1991) Entropy: a new definition and its applications. IEEE Trans Syst Man Cybern 21:1260–1270. https://doi.org/10.1109/21.120079
59. Pal NR, Pal SK (1993) A review on image segmentation techniques. Pattern Recogn 26:1277–1294. https://doi.org/10.1016/0031-3203(93)90135-J
60. Panda R, Agrawal S, Bhuyan S (2013) Edge magnitude based multilevel thresholding using cuckoo search technique. Expert Syst Appl 40:7617–7628. https://doi.org/10.1016/j.eswa.2013.07.060
61. Piotrowski AP (2018) L-SHADE optimization algorithms with population-wide inertia. Inf Sci 468:117–141. https://doi.org/10.1016/j.ins.2018.08.030
62. Raja NSM, Sukanya SA, Nikita Y (2015) Improved PSO based multi-level thresholding for Cancer infected breast thermal images using Otsu. Proced Comput Sci 48:524–529. https://doi.org/10.1016/j.procs.2015.04.130
63. Raja NSM, Rajinikanth V, Latha K (2021) Otsu based optimal multilevel image thresholding using firefly algorithm. Stud Comput Intell 927:61–69. https://doi.org/10.1155/2014/794574Research
64. Rao RV, Savsani VJ, Vakharia DP (2011) Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. CAD Comput Aid Des 43:303–315. https://doi.org/10.1016/j.cad.2010.12.015
65. Rashedi E, Nezamabadi-pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179:2232–2248. https://doi.org/10.1016/j.ins.2009.03.004
66. REny (2004) Observability of Rényi's entropy. Phys Rev E Stat Nonlinear Soft Matter Phys 69:026128
67. Rodríguez-Esparza E, Zanella-Calzada LA, Oliva D et al (2020) An efficient Harris hawks-inspired image segmentation method. Expert Syst Appl 155. https://doi.org/10.1016/j.eswa.2020.113428
68. Sahoo PK, Soltani S, Wong AKC (1988) A survey of thresholding techniques. Comput Vis Graph Image Process 41:233–260. https://doi.org/10.1016/0734-189X(88)90022-9
69. Saremi S, Mirjalili S, Lewis A (2017) Grasshopper optimisation algorithm: theory and application. Adv Eng Softw 105:30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004
70. Sarkar S, Das S (2013) Multilevel image thresholding based on 2D histogram and maximum tsallis entropy - a differential evolution approach. IEEE Trans Image Process 22:4788–4797. https://doi.org/10.1109/TIP.2013.2277832
71. Sezgi S (2004) Survey over image thresholding techniques and quantitative performance evaluation. J Electron Imaging 13:220. https://doi.org/10.1117/1.1631316
72. Shadravan S, Naji HR, Bardsiri VK (2019) The sailfish optimizer: a novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems. Eng Appl Artif Intell 80:20–34. https://doi.org/10.1016/j.engappai.2019.01.001
73. Shubham S, Bhandari AK (2019) A generalized Masi entropy based efficient multilevel thresholding method for color image segmentation. Multimed Tools Appl 78:17197–17238. https://doi.org/10.1007/s11042-018-7034-x
74. Singh S, Mittal N, Singh H (2020) A multilevel thresholding algorithm using LebTLBO for image segmentation. Neural Comput & Applic 32:16681–16706. https://doi.org/10.1007/s00521-020-04989-2
75. Suganthan PN (2012) Differential evolution algorithm: Recent advances. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). LNCS 7505:30–46. https://doi.org/10.1007/978-3-642-33860-1_4
76. Sulaiman MH, Mustaffa Z, Saari MM, Daniyal H (2020) Barnacles mating optimizer: a new bio-inspired algorithm for solving engineering optimization problems. Eng Appl Artif Intell 87. https://doi.org/10.1016/j.engappai.2019.103330
77. Sun M, Wei H (2020) An improved cuckoo search algorithm for multi-level gray-scale image thresholding. Multimed Tools Appl https://doi.org/10.1007/s11042-020-08931-5
78. Sun G, Zhang A, Yao Y, Wang Z (2016) A novel hybrid algorithm of gravitational search algorithm with genetic algorithm for multi-level thresholding. Appl Soft Comput 46:703–730. https://doi.org/10.1016/j.asoc.2016.01.054

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

79. Thippa Reddy G, Bhattacharya S, Maddikunta PKR, et al (2020) Antlion re-sampling based deep neural network model for classification of imbalanced multimodal stroke dataset. Multimed Tools Appl https://doi.org/10.1007/s11042-020-09988-y
80. Ventura de Oliveira P, Yamanaka K (2018) Image Segmentation Using Multilevel Thresholding and Genetic Algorithm: An Approach. In: 2018 2nd International Conference on Data Science and Business Analytics (ICDSBA). pp 380–385
81. Wang Z, Bovik AC, Hamid Rahim Sheikh EPS (2017) Image quality assessment: from error visibility to structural similarity. Can J Civ Eng 44:253–263. https://doi.org/10.1139/cjce-2016-0381
82. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1:67–82. https://doi.org/10.1109/4235.585893
83. Wunnava A, Naik MK, Panda R, et al (2020) A novel interdependence based multilevel thresholding technique using adaptive equilibrium optimizer. Eng Appl Artif Intell 94:. https://doi.org/10.1016/j.engappai.2020.103836
84. Zar JH (1999) Biostatistical analysis. Prentice Hall, Englewood Cliffs
85. Zhao W, Zhang Z, Wang L (2020) Manta ray foraging optimization: an effective bio-inspired optimizer for engineering applications. Eng Appl Artif Intell 87:103300. https://doi.org/10.1016/j.engappai.2019.103300

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Bibekananda Jena[1] · Manoj Kumar Naik[2] · Rutuparna Panda[3] · Ajith Abraham[4]

Bibekananda Jena
bibekananda.jena@gmail.com

Rutuparna Panda
r_ppanda@yahoo.co.in

Ajith Abraham
ajith.abraham@ieee.org

[1] Department of Electronics and Communication Engineering, Anil Neerukonda Institute of Technology & Sciences, Sangivalasa, Visakhapatnam, Andhra Pradesh 531162, India

[2] Faculty of Engineering and Technology, Siksha O Anusandhan, Bhubaneswar, Odisha 751030, India

[3] Department of Electronics and Telecommunication Engineering, Veer Surendra Sai University of Technology, Burla, Odisha 768018, India

[4] Faculty of Computing and Data Science, FLAME University, Lavale, Pune, Maharashtra 412115, India

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH ("Springer Nature").
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users ("Users"), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use ("Terms"). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com