

Chapter 3

Hybrid BBO-DE Algorithms for Fuzzy Entropy-Based Thresholding

Ilhem Boussaïd, Amitava Chatterjee, Patrick Siarry
and Mohamed Ahmed-Nacer

Abstract This chapter shows how a recently proposed stochastic optimization algorithm, called biogeography-based optimization (BBO), can be efficiently employed for development of three-level thresholding-based image segmentation. This technique is utilized to determine suitable thresholds utilizing a fuzzy entropy-based fitness function, which the optimization procedure attempts to maximize. The chapter demonstrates how improved BBO-based strategies, employing hybridizations with differential evolution (DE) algorithms, can be employed to incorporate diversity in the basic BBO algorithm that can help the optimization algorithm avoid getting trapped at local optima and seek the global optimum in a more efficient manner. Several such hybrid BBO-DE algorithms have been utilized for this optimum thresholding-based image segmentation procedure. A detailed implementation analysis for a popular set of well-known benchmark images has been carried out to qualitatively and quantitatively demonstrate the utility of the proposed hybrid BBO-DE optimization algorithm.

I. Boussaïd (✉) · M. Ahmed-Nacer
University of Science and Technology Houari Boumediene (USTHB),
El-Alia BP 32, Bab-Ezzouar, 16111 Algiers, Algeria
e-mail: ilhem.boussaid@univ-paris12.fr

M. Ahmed-Nacer
e-mail: anacer@cerist.dz

A. Chatterjee
Electrical Engineering Department, Jadavpur University,
Kolkata, 700 032 West Bengal, India
e-mail: cha_ami@yahoo.co.in

P. Siarry
Université de Paris-Est Créteil Val de Marne, LiSSi (EA 3956)
61 avenue du Général de Gaulle, 94010 Créteil, France
e-mail: siarry@univ-paris12.fr

3.1 Introduction

Image segmentation is the process of decomposing an image into a set of regions which are visually distinct and uniform with respect to some feature. This distinguishing feature may be colour or gray-level information that is used to create histograms, or information about the pixels that indicate edges or boundaries or texture information [10]. There is a wide range of image segmentation techniques available in the literature [20, 24]. Among them, one approach of particular interest is the thresholding approach, because of its efficiency in performance and its theoretical simplicity. A comprehensive survey of image thresholding techniques is found in [26].

Thresholding techniques can be classified as bilevel and multilevel thresholding, depending on number of image segments into which an original image is decomposed. In bilevel thresholding, each image pixel is assigned to one of two brightness regions, object and background, according to whether its intensity (gray level or colour) is greater than a specified threshold T or not. In multilevel thresholding, pixels can be classified into many classes, not just foreground and background. Therefore, more than one threshold should be determined to segment the image into certain brightness regions which may correspond to one background and several objects.

Among the multitude of image thresholding techniques, entropy-based approaches have drawn a lot of attention in recent times. The principle of entropy, a well-known concept from information theory introduced by Shannon [6], is to use uncertainty as a measure to describe the information contained in a source. The maximum information is achieved when no a priori knowledge is available, in which case, it results in maximum uncertainty.

Basically, entropy thresholding considers an image histogram as a probability distribution, and determines an optimal threshold value that yields the maximum entropy. More specifically, the best entropy thresholded image is the one that preserves as much information as possible that is contained in the original unthresholded image in terms of Shannon's entropy [4].

The popular criterion for image thresholding based on maximum entropy principle was first applied by Pun [22, 23] and then improved upon in [12]. The concept was later generalized to evolve to Renyi's entropy [25] and Tsallis's entropy [33].

However, due to the possible multivalued levels of brightness in a gray-tone image, or inherent vagueness and imprecision embedded in images, the result of image thresholding is not always satisfactory. This uncertainty can be adequately analyzed through the use of fuzzy set theory [1]. This theory, proposed by Zadeh [35], is a mathematical tool to analyze vagueness and uncertainty inherent in making decisions. It has proved its efficiency and usefulness in many applications, including image processing problems. In fact, some fuzzy logic-based thresholding techniques have been proposed in the literature, where fuzzy theory is employed to select an optimal threshold by maximizing the fuzzy entropy [30, 32, 36].

Cheng et al. [5] introduced the concept of fuzzy c-partition into the maximum entropy principle to select the threshold values for gray-level images. This method was first applied for bilevel thresholding and then extended to multilevel thresholding.

Tobias and Serra [32] proposed an approach for histogram thresholding which was not based on the minimization of a threshold-dependent criterion function. The histogram threshold was determined according to the similarity between gray levels, assessed through implementation of a fuzzy measure.

In [36], Zhao et al. proposed a new technique for three-level thresholding by exploiting the relationship between the fuzzy c -partition and the probability partition. In their proposed technique, the maximum entropy principle was used to measure the compatibility between fuzzy partition and probability partition. Zhao et al. used the simplest function, which is monotonic in nature, to approximate the memberships of the bright, dark and medium fuzzy sets (defined according to pixel intensity levels) and derived a necessary condition of the entropy function arriving at a maximum. Based on the idea of Zhao et al., Tao et al. [31] designed a new three-level thresholding method for image segmentation. The authors defined a new concept of fuzzy entropy through probability analysis, fuzzy partition and entropy theory. The image is first partitioned into three parts, namely dark, gray and white, whose member functions of fuzzy region are described by a Z -function, a Π -function and a S -function, respectively. Later, Tao et al. [30] developed another system which examined the performance of their previous approach for the segmentation of infrared objects. This approach entailed the use of the ant colony optimization (ACO) method to effectively obtain the optimal combination of the free parameters of the fuzzy sets. The experimental results showed that the implementation of the proposed fuzzy entropy principle by ACO had more highly effective search performance than the genetic algorithm (GA) used in [31].

The present work aims at developing a new three-level thresholding algorithm, called DBBO-Fuzzy, based on the hybridization of biogeography-based optimization (BBO), a very recently proposed population based optimization technique, and the differential evolution (DE) algorithm, a very powerful stochastic optimizer. The approach presented in [31] is adopted as the basic support for this work.

A hybrid DE with BBO, namely DE/BBO, has been proposed in [11] where a hybrid migration operator is defined. Another combination of BBO and DE is proposed in [3], where the population is updated by applying, alternately from an iteration of the algorithm to the next, the BBO and DE updating methods. The proposed DBBO-Fuzzy algorithm incorporates the mutation procedure inherited from DE algorithm to replace the existing mutation procedure in BBO. A selection operator is also introduced in order to favor a given number of individuals for the next generation. In addition, the algorithm incorporates the features of elitism in order to prevent the best solutions from getting corrupted. The proposed algorithm is tested on a popular set of well-known benchmark images, and experimental results show that the proposed approach is reliable and efficient.

First of all, it is necessary to present some basic definitions. Section 3.2 provides the preliminaries of the fuzzy set theory and fuzzy entropy formulation, where the terminology used in [35] has been followed. The three-level thresholding problem is then formulated and the assumptions made in this paper are introduced in Sect. 3.3. Section 3.4 briefly describes the conventional BBO algorithm. The proposed algorithm is presented in Sect. 3.4, and the effectiveness of the proposed algorithm,

along with a comparison with BBO and DE-based algorithms, is demonstrated for a set of benchmark images in Sect. 3.6. Finally, Sect. 3.7 presents our conclusions.

3.2 Fuzzy Set Theory

Fuzzy set theory is a generalization of classical set theory, designed to express uncertainty and imprecision in available knowledge. The theory dates back to 1965, when Lotfi Zadeh, a professor at Berkeley, published his seminal paper on the theory of fuzzy sets and the associated logic, namely fuzzy logic [35].

Essentially, the fuzziness, a feature of imperfect information, results from the lack of crisp distinction between the elements belonging and not belonging to a set.

Definition 3.1 Let X be a universe of discourse with a generic element denoted by x_i : $X = \{x_1, x_2, \dots, x_n\}$.

A fuzzy set A in a space X is formally defined as:

$$A = \{(x_i, \mu_A(x_i)) \mid x \in X\} \quad (3.1)$$

in which $\mu_A : X \rightarrow [0, 1]$ is the membership function or characteristic function. This function assigns to each element x_i in the set a membership grade $\mu_A(x_i) \in [0, 1]$. As opposed to classical sets, where each element must have either 0 as the membership grade if the element is completely outside the set or 1 if the element is completely in the set, the theory of fuzzy sets is based on the idea that one is uncertain about whether the element is in or out of the set. Thus, the nearer the value of $\mu_A(x_i)$ to unity, the higher the grade of membership of x_i in A . The fuzzier case and the more difficult one is when $\mu_A(x_i) = 0.5$ [15].

The fuzzy set theory approach has found interesting applications in automatic control, decision making, pattern recognition, psychology, economics, medical diagnosis, image processing and other fields. A number of aspects of digital image processing have been treated by this theory, such as image quality assessment, edge detection, image segmentation, etc.

3.2.1 Fuzzy Entropy

A very frequent measure of fuzziness is referred to as the fuzzy entropy inspired by the Shannon entropy of random variables [27] and introduced for the first time by De Luca and Termini [8]. The authors established the following four axioms for fuzzy entropy:

Definition 3.2 Let E be a set-to-point map $E : F(X) \rightarrow [0, 1]$. Hence E is a fuzzy set defined on fuzzy sets and $F(X)$ is the family of all fuzzy sets in X . E is an entropy measure if it satisfies the four De Luca and Termini axioms:

1. $E(A) = 0$ iff $A \in X$ (A nonfuzzy).
2. $E(A) = 1$ (the maximum value) iff $\mu_A(x) = 0.5, \forall x \in X$.
3. $E(A) \leq E(B)$ if A is less fuzzy than B , i.e., if $\mu_A(x) \leq \mu_B(x)$ when $\mu_B(x) \leq 0.5$, and $\mu_A(x) \geq \mu_B(x)$ when $\mu_B(x) \geq 0.5$.
4. $E(A) = E(A^c)$, where A^c is the complementary set of A .

A number of studies related to the measures of fuzzy entropy and their applications have been conducted by Kaufmann [13], Bhandari and Pal [2], Kaufmann [14], Pal and Pal [19], and Fan and Xie [9].

3.3 Problem Formulation

3.3.1 Model of an Image

A digital gray-tone image refers to a two-dimensional light intensity function defined over a spatial coordinate system. Let $G = \{0, 1, \dots, L - 1\}$ be the set of intensity values, and $D = \{(x, y) : 0 \leq x \leq M - 1, 0 \leq y \leq N - 1\}$ be the spatial coordinates of the pixels for an $M \times N$ image. The digital image defines a mapping $I : D \rightarrow G$, where $0 \leq I(x, y) \leq L - 1$ gives the intensity (brightness) of the image at the spatial coordinates $(x, y) \in D$ with $L = 256$ gray levels for an 8-bit image [30, 31].

Let $D_k = \{(x, y) | I(x, y) = k, (x, y) \in D\}$, $k \in G$. The histogram of an image, defined as $H = \{h_0, h_1, \dots, h_{L-1}\}$, presents the frequency of occurrence of each gray level in the image and is obtained directly from the observation of the considered image. In view of this consideration, the k th gray level in the image is defined as follows:

$$h_k = \frac{n_k}{N * M}, \quad k = 0, 1, \dots, L - 1 \quad (3.2)$$

where n_k denotes the total number of pixels in D_k , and $N * M$ denotes the total number of pixels in the image. It is clear that

$$0 \leq h_k \leq 1 \quad \text{and} \quad \sum_{k=0}^{L-1} h_k = 1 \quad (3.3)$$

A probability partition (PP) of the image domain D is defined as

$$\Pi_L = \{D_0, D_1, \dots, D_{L-1}\}$$

which is characterized by a probabilistic distribution [30, 31]:

$$p_k \equiv P(D_k) = h_k, \quad k = 0, 1, \dots, L - 1, \quad (3.4)$$

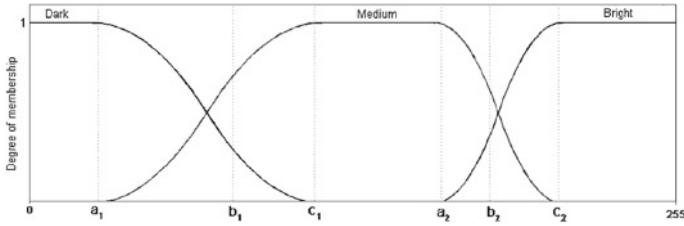


Fig. 3.1 Membership function graph

Equation (3.4) presents the relationship between the histogram H and the probability partition Π , where p_k is the probability measure of the occurrence of gray level k .

3.3.2 Three-Level Thresholding

The segmentation problem is to determine the sets $D_k \subset D$ ($k = 0, \dots, L - 1$) whose union is the entire image D . Thus, the sets that constitute the segmentation must satisfy:

$$\bigcup_{k=0}^{L-1} D_k = D \quad \text{and} \quad D_i \cap D_j = \phi \quad (i \neq j) \quad (3.5)$$

where ϕ denotes an empty set. Ideally, a segmentation method finds those sets that correspond to distinct anatomical structures or regions of interest in the image.

In the case of three-level thresholding of an image, the aim is to separate its domain D into three parts, E_d , E_m and E_b , where E_d is composed of 'dark' pixels corresponding to the smaller gray levels, E_b is composed of those 'bright' pixels corresponding to the larger gray levels, and E_m is composed of pixels with medium gray levels.

The problem is to find the unknown probabilistic fuzzy 3-partition of D , $\Pi_3 = \{E_d, E_m, E_b\}$, which is characterized by the probability distributions [30, 31]:

$$p_d = P(E_d), \quad p_m = P(E_m), \quad p_b = P(E_b) \quad (3.6)$$

The three fuzzy partitions E_d , E_m and E_b are characterized by three membership functions μ_d , μ_m and μ_b , respectively. The membership function μ_d of dark pixels corresponds to a Z-function, the membership function μ_m of medium pixels is a Π -function, and the membership function μ_b of bright pixels of the image is a S-function [31] (see Fig. 3.1). Six free parameters $\{a_1, b_1, c_1, a_2, b_2, c_2\}$ control the shapes of the three membership functions and satisfy the conditions $0 < a_1 \leq b_1 \leq c_1 \leq a_2 \leq b_2 \leq c_2 < 255$ for an image with 256 gray levels.

The three-level thresholding involves a determination of the optimal thresholds T_1 and T_2 such that the classification of a pixel $I(x, y)$ is achieved as follows:

$$\begin{aligned} D_{kd} &= \{(x, y) : I(x, y) \leq T_1, (x, y) \in D_k\} \\ D_{km} &= \{(x, y) : T_1 < I(x, y) \leq T_2, (x, y) \in D_k\} \\ D_{kb} &= \{(x, y) : I(x, y) > T_2, (x, y) \in D_k\} \end{aligned} \quad (3.7)$$

Once the parameters a_1, b_1, c_1, a_2, b_2 and c_2 are selected then $\Pi_k = \{D_{kd}, D_{km}, D_{kb}\}$ is the PP of D_k with the probabilistic distribution:

$$\begin{aligned} p_{kd} &= P(D_{kd}) = p_k \cdot p_{d|k} \\ p_{km} &= P(D_{km}) = p_k \cdot p_{m|k} \\ p_{kb} &= P(D_{kb}) = p_k \cdot p_{b|k} \end{aligned} \quad (3.8)$$

In Eq. (3.8), $p_{d|k}$ is the conditional probability of a pixel that is classified into the class “d” (dark) under the condition that the pixel belongs to D_k . Similarly, $p_{m|k}$ and $p_{b|k}$ are the conditional probabilities of a pixel belonging to classes “m” (medium) and “b” (bright), respectively.

Based on the complete probability formula, we therefore have:

$$\begin{aligned} p_d &= \sum_{k=0}^{255} p_k \cdot p_{d|k} = \sum_{k=0}^{255} p_k \cdot \mu_d(k) \\ p_m &= \sum_{k=0}^{255} p_k \cdot p_{m|k} = \sum_{k=0}^{255} p_k \cdot \mu_m(k) \\ p_b &= \sum_{k=0}^{255} p_k \cdot p_{b|k} = \sum_{k=0}^{255} p_k \cdot \mu_b(k) \end{aligned} \quad (3.9)$$

Based on Eq. (3.9), it is clear that the three-level thresholding problem is reduced to finding suitable membership functions $\mu_d(k)$, $\mu_m(k)$ and $\mu_b(k)$ of a pixel with an arbitrary intensity level k . These membership functions represent the conditional probability that a pixel is classified into the dark, medium and bright regions, respectively, with respect to the variable $k \in G$ (i.e., $p_{d|k} = \mu_d(k)$, $p_{m|k} = \mu_m(k)$ and $p_{b|k} = \mu_b(k)$). It is obvious that $\mu_d(k) + \mu_m(k) + \mu_b(k) = 1, k = 0, 1, \dots, 255$. The three membership functions are shown in Eqs. (3.10)–(3.12):

$$\mu_d(k) = \begin{cases} 1 & k \leq a_1 \\ 1 - \frac{(k-a_1)^2}{(c_1-a_1)*(b_1-a_1)} & a_1 < k \leq b_1 \\ \frac{(k-c_1)^2}{(c_1-a_1)*(c_1-b_1)} & b_1 < k \leq c_1 \\ 0 & k > c_1 \end{cases} \quad (3.10)$$

$$\mu_m(k) = \begin{cases} 0 & k \leq a_1 \\ \frac{(k-a_1)^2}{(c_1-a_1)*(b_1-a_1)} & a_1 < k \leq b_1 \\ 1 - \frac{(k-c_1)^2}{(c_1-a_1)*(c_1-b_1)} & b_1 < k \leq c_1 \\ 1 & c_1 < k \leq a_2 \\ 1 - \frac{(k-a_2)^2}{(c_2-a_2)*(b_2-a_2)} & a_2 < k \leq b_2 \\ \frac{(k-c_2)^2}{(c_2-a_2)*(c_2-b_2)} & b_2 < k \leq c_2 \\ 0 & k > c_2 \end{cases} \quad (3.11)$$

$$\mu_b(k) = \begin{cases} 0 & k \leq a_2 \\ \frac{(k-a_2)^2}{(c_2-a_2)*(b_2-a_2)} & a_2 < k \leq b_2 \\ 1 - \frac{(k-c_2)^2}{(c_2-a_2)*(c_2-b_2)} & b_2 < k \leq c_2 \\ 1 & k > c_2 \end{cases} \quad (3.12)$$

The free parameters of the three membership functions can be determined by maximizing the fuzzy partition entropy [30, 31]. The total fuzzy entropy function of partition Π_3 is defined as:

$$H(a_1, b_1, c_1, a_2, b_2, c_2) = H_d + H_m + H_b \quad (3.13)$$

where,

$$\begin{aligned} H_d &= - \sum_{k=0}^{255} \frac{p_k * \mu_d(k)}{p_d} * \ln \left(\frac{p_k * \mu_d(k)}{p_d} \right) \\ H_m &= - \sum_{k=0}^{255} \frac{p_k * \mu_m(k)}{p_m} * \ln \left(\frac{p_k * \mu_m(k)}{p_m} \right) \\ H_b &= - \sum_{k=0}^{255} \frac{p_k * \mu_b(k)}{p_b} * \ln \left(\frac{p_k * \mu_b(k)}{p_b} \right) \end{aligned} \quad (3.14)$$

The best-selected set of $\{a_1, b_1, c_1, a_2, b_2, c_2\}$ is the one that corresponds to maximum entropy H .

The optimal thresholds T_1 and T_2 that segment the image into three gray levels are obtained as the intersections of the membership function curves, i.e.,

$$\mu_d(T_1) = \mu_m(T_1) = 0.5 \quad (3.15)$$

$$\mu_m(T_2) = \mu_b(T_2) = 0.5 \quad (3.16)$$

Based on Eqs. (3.10)–(3.12), it can be written [31]:

$$T_1 = \begin{cases} a_1 + \sqrt{(c_1 - a_1) * (b_1 - a_1) / 2} & \text{if } (a_1 + c_1) / 2 \leq b_1 \leq c_1 \\ c_1 - \sqrt{(c_1 - a_1) * (c_1 - b_1) / 2} & \text{if } a_1 \leq b_1 < (a_1 + c_1) / 2 \end{cases} \quad (3.17)$$

$$T_2 = \begin{cases} a_2 + \sqrt{(c_2 - a_2) * (b_2 - a_2) / 2} & \text{if } (a_2 + c_2) / 2 \leq b_2 \leq c_2 \\ c_2 - \sqrt{(c_2 - a_2) * (c_2 - b_2) / 2} & \text{if } a_2 \leq b_2 < (a_2 + c_2) / 2 \end{cases} \quad (3.18)$$

As mentioned before, to find the optimal combination of all the fuzzy parameters, we propose to use a new optimization technique based on the hybridization of BBO and DE algorithms.

3.4 Biogeography-Based Optimization Algorithm (BBO)

The theory of biogeography grew out of the works of Wallace [34] and Darwin [7] in the past and the works of McArthur and Wilson [17] more recently. Some of the key questions that this branch of biology attempts to answer are: How do organisms reach their current habitats? Do they always occupy their current distribution patterns? Why does an ecosystem have a particular number of species? The patterns of the distribution of the species across geographical areas can usually be explained through a combination of historical factors, such as speciation, extinction and migration.

The biogeography-based optimization (BBO) algorithm, developed by Dan Simon [28], is strongly influenced by the equilibrium theory of island biogeography [17]. The basic premise of this theory is that the rate of change in the number of species on an island depends critically on the balance between the immigration of new species onto the island and the emigration of species from the island.

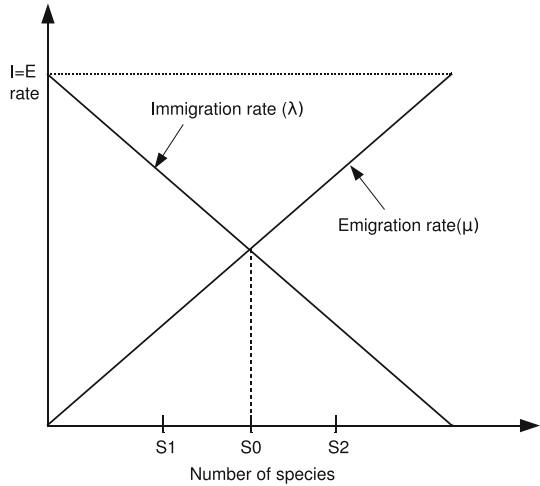
The BBO algorithm operates upon a population of individuals called islands (or habitats). Each habitat represents a possible solution for the problem at hand. The fitness of each habitat is determined by its habitat suitability index (*HSI*), a metric which determines the goodness of a candidate solution, and each habitat feature is called a suitability index variable (*SIV*). Good solutions may have large number of species, which represent habitats with a lower *HSI* than the poor solutions.

As was mentioned before, the migration pattern is determined by the immigration rate (λ) at which new species immigrate to the habitat, and the emigration rate (μ) at which populations of established species emigrate. These parameters are functions of the number of species in a habitat.

But how might immigration and emigration work on a habitat? We make two sets of assumptions regarding these processes:

Immigration: The rate of immigration (λ) declines with the number of species (S) present on the habitat. Maximum immigration rate (I) occurs when the habitat is empty and decreases as more species are added. Once all the potential colonists are on the habitat, then one can write $S = S_{max}$ (the maximum number of species the habitat can support), and the immigration rate must be equal to zero. Generally speaking,

Fig. 3.2 The relationship between the fitness of habitats (number of species), emigration rate μ and immigration rate λ



the immigration rate when there are S species in the habitat is given by:

$$\lambda_S = I \left(1 - \frac{S}{S_{max}} \right) \tag{3.19}$$

Emigration: The rate of emigration (μ) for a habitat increases with the number of species (S). Maximum emigration rate (E) occurs when all possible species are present on the habitat (i.e., when $S = S_{max}$), and must be zero when no species are present. Generally speaking, the emigration rate when there are S species in the habitat is given by:

$$\mu_S = E \left(\frac{S}{S_{max}} \right) \tag{3.20}$$

Figure 3.2 graphically represents the relationships between the number of species, emigration rate μ and immigration rate λ . Over a period of time, the counteracting forces of emigration and immigration result in an equilibrium level of species richness. The equilibrium value S^* is the point at which the rate of arrival of species λ is exactly matched by the rate of emigration μ . We have assumed here that μ and λ vary following linear relationships, but different mathematical models of biogeography that include more complex variations have also been proposed [17].

We now consider the probability P_s that the habitat contains exactly S species. The number of species changes from time t to time $(t + \Delta t)$ as follows [28]:

$$P_s(t + \Delta t) = P_s(t)(1 - \lambda_s \Delta t - \mu_s \Delta t) + P_{s-1} \lambda_{s-1} \Delta t + P_{s+1} \mu_{s+1} \Delta t \tag{3.21}$$

This states that the number of species on the habitat in one time step is based on the total number of current species on the habitat, the new immigrants and the number of species that leave the habitat during this time period. We assume here that Δt is small enough so that the probability of more than one immigration or emigration can be ignored. In order to have S species at time $(t + \Delta t)$, one of the following conditions must hold:

- There were S species at time t , and no immigration or emigration occurred between t and $(t + \Delta t)$;
- One species immigrated onto a habitat already occupied by $S - 1$ species at time t .
- One species emigrated from a habitat occupied by $S + 1$ species at time t .

The limit of Eq. (3.21) as $\Delta t \rightarrow 0$ is given by Eq. (3.22).

$$\dot{P}_S = \begin{cases} -(\lambda_S + \mu_S)P_S + \mu_{S+1}P_{S+1} & \text{if } S = 0 \\ -(\lambda_S + \mu_S)P_S + \lambda_{S-1}P_{S-1} + \mu_{S+1}P_{S+1} & \text{if } 1 \leq S \leq S_{max} - 1 \\ -(\lambda_S + \mu_S)P_S + \lambda_{S-1}P_{S-1} & \text{if } S = S_{max} \end{cases} \quad (3.22)$$

Equation (3.22) can be arranged into a single matrix form:

$$\begin{bmatrix} \dot{P}_0 \\ \dot{P}_1 \\ \vdots \\ \dot{P}_n \end{bmatrix} = \begin{bmatrix} -(\lambda_0 + \mu_0) \mu_1 & 0 & \dots & 0 \\ \lambda_0 & -(\lambda_1 + \mu_1) \mu_2 & \dots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \lambda_{n-2} - (\lambda_{n-1} + \mu_{n-1}) \mu_n & \vdots \\ 0 & \dots & 0 & \lambda_{n-1} & -(\lambda_n + \mu_n) \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_n \end{bmatrix} \quad (3.23)$$

For notational brevity, we simply write $n = S_{max}$.

The BBO algorithm can be described overall by Algorithm 3.1. The two basic operators which govern the working of BBO are the migration, described in Algorithm 3.2, and the mutation, described in Algorithm 3.3, where $rand(0, 1)$ is a uniformly distributed random number in the interval $[0, 1]$; X_{ij} is the j th SIV of the solution X_i .

The likelihood that a given solution S is expected a priori to exist as a solution for the given problem is indicated by the species count probability P_S . In this context it should be remarked that very high HSI solutions and very low HSI solutions are both equally improbable. Medium HSI solutions are relatively probable. If a given

solution has a low probability, then it is likely to mutate to some other solution. Conversely, a solution with high probability is less likely to mutate. The mutation rate $m(S)$ is inversely proportional to the solution probability:

$$m(S) = m_{max} \left(1 - \frac{P_S}{P_{max}} \right) \quad (3.24)$$

where m_{max} is a user-defined parameter, and $P_{max} = \max_S P_S$, $S = 1, \dots, S_{max}$.

Migration is used to modify existing habitats by mixing features within the population. Mutation is used to enhance diversity of the population, thereby preventing the search from stagnating. If a habitat S is selected to execute the mutation operation, then a chosen variable ($SI V$) is randomly modified based on its associated probability P_S . At the same time, the concept of elitism (i.e., copying some of the fittest individuals for the next generation) is also applied.

Algorithm 3.1 Biogeography-based optimization (BBO)

- 1: Initialize the BBO parameters: $S_{max}, E, I, m_{max}, Max_{gen}, n_{elit}, \dots$
 - 2: Initialize the generation counter : $g = 0$
 - 3: Create a random initial population $\mathbf{X}_i, i = 1, \dots, NP$
 - 4: Evaluate $f(\mathbf{X}_i), i = 1, \dots, NP$
 - 5: **for** $g = 1$ to Max_{gen} **do**
 - 6: Sort the population from best fit to least fit
 - 7: **for** $i = 1$ to NP **do**
 - 8: Map the HSI to the number of species
 - 9: Calculate the immigration rate λ_i and the emigration rate μ_i
 - 10: Modify the nonelite members of the population probabilistically with the migration operator according to *Algorithm 3.2*
 - 11: **end for**
 - 12: **for** $i = 1$ to NP **do**
 - 13: Mutate the non-elite members of the population with the mutation operator according to *Algorithm 3.3*
 - 14: **end for**
 - 15: **for** $i = 1$ to NP **do**
 - 16: Evaluate the new habitats in the population
 - 17: Replace the habitats with their new versions
 - 18: Apply elitism to preserve n_{elit} best habitats
 - 19: **end for**
 - 20: **end for**
-

3.5 Description of the Proposed DBBO-Fuzzy Algorithm

Motivated by the exploration capabilities of the differential evolution (DE) algorithm [18], a hybrid method combining the exploitation of BBO with the exploration of DE is proposed in this paper. The purpose of this hybridization is to benefit from the advantages of each algorithm and to compensate for each algorithm's weaknesses.

Algorithm 3.2 Migration

```

1: for  $i = 1$  to  $NP$  do
2:   Use  $\lambda_i$  to probabilistically decide whether to immigrate to  $X_i$ .
3:   if  $rand(0, 1) < \lambda_i$  then
4:     for  $j = 1$  to  $NP$  do
5:       Select the emigrating habitat  $X_j$  with probability  $\propto \mu_j$ 
6:       if  $rand(0, 1) < \mu_j$  then
7:         Replace a randomly selected decision variable (SIV) of  $X_i$  with its corresponding
           variable in  $X_j$ 
8:       end if
9:     end for
10:  end if
11: end for

```

Algorithm 3.3 Mutation

```

1: for  $i = 1$  to  $NP$  do
2:   Compute the probability  $P_i$  using  $\lambda_i$  and  $\mu_i$ 
3:   Use the probability  $P_i$  to compute the mutation rate  $m_i$ 
4:   for  $j = 1$  to  $D$  do
5:     Select a variable (SIV)  $X_{ij}$  with probability  $\propto P_i$ 
6:     if  $rand(0, 1) < m_i$  then
7:       Replace  $X_{ij}$  with a randomly generated variable from its range
8:     end if
9:   end for
10: end for

```

In order to find the global solution in a better manner than the BBO algorithm, the proposed algorithm, named the DBBO-Fuzzy algorithm, replaces the BBO-based mutation by a DE mutation. In addition, a selection operation is introduced in order to favor a given number of individuals for the next generation.

The proposed algorithm can be summarized as follows:

1. Initialization The algorithm starts with an initial population of NP search variable vectors (or habitats), where the problem dimension D is the number of fuzzy parameters. For the three-level thresholding problem, six parameters $\{a_1, b_1, c_1, a_2, b_2, c_2\}$ are used (i.e., $D = 6$). Since the habitats are likely to get modified over different generations, the following notation may be adopted for representing the i th habitat of the population at the current generation g as:

$$\mathbf{X}_{i,g} = (X_{i,1,g}, X_{i,2,g}, X_{i,j,g}, \dots, X_{i,D,g}) \quad (3.25)$$

where $i = 1, \dots, NP$, $j = 1, \dots, D$ and $X_{i,j}$ is the j th SIV of the habitat \mathbf{X}_i . Each decision variable, $X_{i,j,g}$, is randomly initialized within its corresponding lower bound (L_j) and upper bound (U_j), and it is intended to cover the entire search space uniformly in the form:

$$X_{i,j,0} = L_j + rand(0, 1) \times (U_j - L_j) \quad (3.26)$$

2. Evaluation of the objective function: The objective function values of the habitats are evaluated using the fuzzy entropy function given by Eq. (3.13). The objective function has six parameters (SIV) $a_1, b_1, c_1, a_2, b_2, c_2$, which satisfy the conditions $0 < a_1 \leq b_1 \leq c_1 \leq a_2 \leq b_2 \leq c_2 < 255$.

3. Migration: The migration operator reproduces a new population vector $\mathbf{M}_{i,g}$ as follows:

$$M_{i,j,g} = \begin{cases} X_{k,j,g} & \text{if } \text{rand}(0, 1) < \lambda_i \\ X_{i,j,g} & \text{otherwise} \end{cases} \quad (3.27)$$

where $i = 1, 2, \dots, NP$, $j = 1, \dots, D$ and $X_{k,j,g}$ is the j th decision variable of a randomly selected habitat $\mathbf{X}_{k,g}$. $\mathbf{X}_{k,g}$ is selected with a probability based on μ_k .

4. DE Mutation: The DBBO-Fuzzy incorporates the mutation procedure inherited from DE algorithm [21, 29] to replace the existing mutation procedure in BBO.

The mutation is performed by calculating weighted vector differences between other randomly selected habitats of the same population. A differentiation constant F is used to control the amplification of the differential variation.

Next, five different mutation schemes are outlined, inspired by the suggestions of Price et al. [21]. The general convention used to name the different DE schemes is $DE/x/y$. Here DE stands for differential evolution, x represents a string denoting the type of the vector to be perturbed (whether it is randomly selected or it is the best vector in the population with respect to fitness value) and y is the number of difference vectors considered for perturbation of x .

The mutation operation constructs, for each habitat $\mathbf{M}_{i,g}$, a mutant habitat $\mathbf{V}_{i,g}$ according to one of the following mutation schemes:

- **DE/rand/1:** This mutation scheme uses a randomly selected habitat $\mathbf{M}_{r_1,g}$, and only one weighted difference vector $F \cdot (\mathbf{M}_{r_2,g} - \mathbf{M}_{r_3,g})$ is used to perturb it.

$$\mathbf{V}_{i,g} = \mathbf{M}_{r_1,g} + F \cdot (\mathbf{M}_{r_2,g} - \mathbf{M}_{r_3,g}) \quad (3.28)$$

- **DE/current to best/1:** Here the mutant habitat is created using any two randomly selected habitats of the population as well as the best habitat in the current generation.

$$\mathbf{V}_{i,g} = \mathbf{M}_{i,g} + F \cdot (\mathbf{M}_{best,g} - \mathbf{M}_{i,g}) + F \cdot (\mathbf{M}_{r_1,g} - \mathbf{M}_{r_2,g}) \quad (3.29)$$

- **DE/best/1:** Here the habitat to be perturbed is the best habitat of the current population and the perturbation is caused by using a single difference vector.

$$\mathbf{V}_{i,g} = \mathbf{M}_{best,g} + F \cdot (\mathbf{M}_{r_1,g} - \mathbf{M}_{r_2,g}) \quad (3.30)$$

- **DE/rand/2:** to create $\mathbf{V}_{i,g}$ for each i th habitat, a total of five other distinct habitats (say the $r_1, r_2, r_3, r_4,$ and r_5 th habitats) are chosen in a random manner from the current population.

$$\mathbf{V}_{i,g} = \mathbf{M}_{r_1,g} + F \cdot (\mathbf{M}_{r_2,g} - \mathbf{M}_{r_3,g}) + F \cdot (\mathbf{M}_{r_4,g} - \mathbf{M}_{r_5,g}) \quad (3.31)$$

- **DE/best/2:** in this mutation scheme, the mutant habitat is formed by using two difference vectors, as shown below:

$$\mathbf{V}_{i,g} = \mathbf{M}_{best,g} + F \cdot (\mathbf{M}_{r_1,g} - \mathbf{M}_{r_2,g}) + F \cdot (\mathbf{M}_{r_3,g} - \mathbf{M}_{r_4,g}) \quad (3.32)$$

where the indices r_1, r_2, r_3, r_4, r_5 are randomly chosen over the interval $[1, NP]$ and should be mutually different from the running index i . F is a real constant scaling factor within the range $[0, 2]$, usually chosen to be less than 1. $\mathbf{M}_{best,g}$ is the habitat with best fitness value in the population in generation g .

5. Selection: The values of the objective function are calculated for the updated habitats. The selection operation selects either a habitat $\mathbf{X}_{i,g}$ or its newly updated habitat $\mathbf{V}_{i,g}$ to survive as a member for the next generation, according to the fitness value. For the following generation $g + 1$, new habitats $\mathbf{X}_{i,g+1}$ are selected according to the following selection rule:

$$\mathbf{X}_{i,g+1} = \begin{cases} \mathbf{V}_{i,g} & \text{if } f(\mathbf{V}_{i,g}) < f(\mathbf{X}_{i,g}) \\ \mathbf{X}_{i,g} & \text{if } f(\mathbf{V}_{i,g}) > f(\mathbf{X}_{i,g}) \end{cases} \quad (3.33)$$

The best new habitat replaces the worst corresponding one in the current population only if $\mathbf{V}_{i,g}$ is better than $\mathbf{X}_{i,g}$. This concept is similar to what happens in nature for longer-living species, where the offspring and parents are alive concurrently and have to compete.

6. Boundary constraints: If the variable value $X_{i,j,g}$ violates the boundary constraints, the corresponding violating variable value is randomly generated within the boundary constraints as follows:

$$X_{i,j,g} = L_j + rand(0, 1) \times (U_j - L_j)$$

7. Stopping criteria: If the stopping criteria are met, the vector represented by the best habitat contains the optimal combination of fuzzy parameter values that maximize the total fuzzy entropy function of partition Π_3 . Otherwise, the procedure is repeated from step 3.

Table 3.1 DBBO-fuzzy parameters

Parameters	Notation	Value
Population size (number of habitats)	NP	20
Elitism parameter	n_{elit}	2
Maximum immigration rate	I	1
Maximum emigration rate	E	1
Number of generations	Max_{gen}	20
Search domain for each parameter vector (SIV)	G	[0, 255]
Number of decision variables (fuzzy parameters)	D	6
Constant of differentiation	F	0.5
DE mutation scheme	DE/rand/1	

3.6 Experimental Settings and Results

3.6.1 Test Images

The performance of the proposed DBBO-Fuzzy algorithm is compared with those of the basic BBO algorithm [28], named here as BBO-Fuzzy, and the performance of DE-Fuzzy algorithm, which proceeds exactly as the original algorithm presented in [21].

The performance of these competing three-level thresholding algorithms are tested with a set of 12 benchmark images, each with 256 gray levels. These are commonly known as Lena, Peppers, Cameraman, Airplane, Lake, Walking bridge, Mandrill, Barbara, Boat, Elaine, GoldHill and Fingerprint. All the images are of size 512×512 pixels. The original images considered are shown in Fig. 3.3.

3.6.2 Test Design

In all experiments, the same parameter values are used for each of the three aforementioned algorithms to make a fair comparison. A population of 20 individuals is used; these are evolved during 20 generations. For each test image, ten independent runs are carried out.

For the DE-Fuzzy algorithm, $F = 0.5$ and $CR = 0.9$ are chosen, as recommended in [29]. For fair performance comparison, the same mutation scheme, DE/rand/1, is adopted for both the DE-Fuzzy and the DBBO-Fuzzy algorithms. For the BBO-Fuzzy algorithm, the same parameter settings as in [28] are used. In Table 3.1, the parameter setup used in the experiments conducted is summarized.



Fig. 3.3 Original test images: **a** Lena, **b** Peppers, **c** Cameraman, **d** Airplane, **e** Lake, **f** Walking bridge, **g** Mandrill, **h** Barbara, **i** Boat, **j** Elaine, **k** GoldHill, **l** Fingerprint

3.6.3 Results and Discussions

The three-level thresholding algorithms are employed to determine the optimal thresholds T_1 and T_2 that segment a given image into three gray levels while preserving the original information as much as possible after creation of this partition.

The objective is to maximize the total fuzzy entropy $H(b_1, c_1, a_2, b_2, c_2)$, given in Eq. (3.13), and then to find the “optimal” combination of all the fuzzy parameters $(a_1, b_1, c_1, a_2, b_2, c_2)$ that produce the maximization of fuzzy entropy. The higher value of objective function results in better segmentation.

The results are also compared based on the uniformity factor, the most commonly used measure to quantitatively judge the segmentation quality. This uniformity measure is defined as [16]:

$$u = 1 - 2 * c * \frac{\sum_{j=0}^c \sum_{i \in R_j} (f_i - \mu_j)^2}{N * (f_{max} - f_{min})^2} \quad (3.34)$$

where, c number of thresholds, R_j j th segmented region, f_i gray level of the pixel i , μ_j mean gray level of pixels in j th region, N total number of thresholds in the given image, f_{max} maximum gray level of pixels in the given image, f_{min} minimum gray level of pixels in the given image

The value of this uniformity measure should be a value within the interval $[0, 1]$. The higher the value of u , better the quality of the thresholded image.

Table 3.2 shows the average objective values (i.e., total fuzzy entropy) achieved by each algorithm for each image under test. The values in *boldface* describe the best-performing algorithm among competing algorithms. It is observed from the obtained results that the proposed DBBO-Fuzzy method obtains higher fuzzy entropy values than both BBO-Fuzzy and DE-Fuzzy in all test images.

In order to determine whether the differences between the DBBO-Fuzzy algorithm and the BBO-Fuzzy and DE-Fuzzy algorithms are statistically significant, a two-tailed t-test was conducted with $df = 10 + 10 - 2 = 18$ degrees of freedom at $\alpha = 0.05$ level of significance (i.e., at 95 % confidence level). The average objective values and the standard deviations obtained by each algorithm over ten independent runs were used to calculate the t-values. The absolute value of the computed t is found to be larger than the critical value in all test images. This suggests that, with 95 % confidence, the difference between DBBO-Fuzzy algorithm and the other two competing algorithms is statistically significant. Therefore, it is evident that the hybridization of the BBO algorithm with DE has noticeable effect on the performance of both algorithms. In these tests, 1 versus 2 means DBBO-Fuzzy algorithm versus BBO-Fuzzy algorithm, and 1 versus 3 means DBBO-Fuzzy algorithm versus DE-Fuzzy algorithm.

Table 3.3 shows the optimal thresholds obtained and the uniformity factor values attained using DBBO-Fuzzy, BBO-Fuzzy and DE-Fuzzy methods. One can observe from the results reported in Table 3.3 that the solution quality of DBBO-Fuzzy is superior to BBO-Fuzzy and DE-Fuzzy in 11 out of 12 images. The only exception

Table 3.2 Comparison of DBBO-fuzzy, BBO-fuzzy and DE-fuzzy, where *boldface* indicates the best performing algorithm

Test images	Objective values		Standard deviation				1 versus 2		1 versus 3	
	DBBO-fuzzy	BBO-fuzzy	DE-fuzzy	DBBO-fuzzy	BBO-fuzzy	DE-fuzzy	t-test	t-test	t-test	
Lena	1.3851E+01	1.3569E+01	1.3707E+01	5.3148E-02	2.8230E-02	4.1078E-02	1.4823E+01 *	6.7622E+00 *		
Peppers	1.3443E+01	1.3089E+01	1.3314E+01	3.4273E-02	5.6806E-02	5.8473E-02	1.6883E+01 *	6.0430E+00 *		
Cameraman	1.3463E+01	1.3090E+01	1.3235E+01	3.8302E-02	8.7798E-02	5.8679E-02	1.2307E+01 *	1.0301E+01 *		
Airplane	1.3543E+01	1.3380E+01	1.3451E+01	2.5830E-02	6.9561E-02	5.9266E-02	6.9381E+00 *	4.4722E+00 *		
Lake	1.3817E+01	1.3641E+01	1.3739E+01	2.2140E-02	8.0443E-02	3.2497E-02	6.6555E+00 *	6.2791E+00 *		
Walk bridge	1.4550E+01	1.4193E+01	1.4441E+01	4.3054E-02	5.7350E-02	2.9707E-02	1.5728E+01 *	6.5527E+00 *		
Mandrill	1.3815E+01	1.3593E+01	1.3725E+01	4.1507E-02	4.7428E-02	3.621E-02	1.1112E+01 *	5.1733E+00 *		
Barbara	1.4214E+01	1.3845E+01	1.4097E+01	5.7206E-02	6.9849E-02	8.0261E-02	1.2925E+01 *	3.7587E+00 *		
Boat	1.3986E+01	1.3596E+01	1.3880E+01	8.9041E-02	7.0177E-02	7.3576E-02	1.0883E+01 *	2.9037E+00 *		
Elaine	1.4089E+01	1.3771E+01	1.3999E+01	3.5250E-02	6.0926E-02	4.9027E-02	1.4276E+01 *	4.7264E+00 *		
Goldhill	1.4069E+01	1.3693E+01	1.3920E+01	6.7739E-02	3.7019E-02	4.5656E-02	1.5410E+01 *	5.7761E+00 *		
Fingerprint	1.2406E+01	1.2100E+01	1.2328E+01	4.1010E-02	5.8752E-02	4.7234E-02	1.3486E+01 *	3.9108E+00 *		

The results are mean best objective values and standard deviations over ten runs (* the difference between the two algorithms is statistically significant)

Table 3.3 Optimal threshold and uniformity factor values obtained by DBBO-fuzzy, BBO-fuzzy and DE-fuzzy, where *boldface* indicates the best performing algorithm

Test images	Optimal thresholds			Uniformity measures		
	DBBO-fuzzy	BBO-fuzzy	DE-fuzzy	DBBO-fuzzy	BBO-fuzzy	DE-fuzzy
Lena	98, 175	98, 169	83, 161	9.8391E-01	9.8246E-01	9.7213E-01
Peppers	85, 162	71, 205	63, 188	9.7720E-01	9.6821E-01	9.6464E-01
Cameraman	127, 207	132, 211	97, 198	9.6557E-01	9.5745E-01	9.6767E-01
Airplane	41, 166	46, 178	30, 153	9.8719E-01	9.8666E-01	9.8618E-01
Lake	96, 171	95, 165	64, 161	9.8451E-01	9.8420E-01	9.7778E-01
Walk Bridge	79, 154	35, 211	48, 180	9.7816E-01	9.6507E-01	9.6149E-01
Mandrill	74, 150	33, 143	51, 167	9.6755E-01	9.6617E-01	9.6448E-01
Barbara	93, 170	93, 172	90, 170	9.8233E-01	9.7595E-01	9.7342E-01
Boat	107, 185	111, 220	75, 217	9.8327E-01	9.7904E-01	9.7566E-01
Elaine	98, 175	37, 208	44, 196	9.7517E-01	9.6634E-01	9.6595E-01
GoldHill	77, 152	52, 156	67, 181	9.8172E-01	9.7873E-01	9.7072E-01
Fingerprint	94, 172	41, 152	42, 179	9.7856E-01	9.6897E-01	9.5562E-01

Table 3.4 Representative optimal parameter sets ($a_1, b_1, c_1, a_2, b_2, c_2$)

Test images	Optimal parameters					
	DBBO-fuzzy		BBO-fuzzy		DE-fuzzy	
Lena	1, 137, 140, 140, 142, 256	5, 136, 138, 141, 144, 232	1, 115, 118, 119, 124, 256			
Peppers	1, 119, 122, 122, 124, 256	5, 98, 99, 104, 246, 248	1, 83, 95, 107, 194, 256			
Cameraman	1, 177, 180, 180, 194, 256	28, 174, 175, 179, 215, 235	82, 94, 116, 119, 211, 256			
Airplane	1, 1, 136, 138, 138, 233	5, 7, 141, 147, 149, 250	1, 1, 100, 101, 132, 240			
Lake	1, 135, 136, 136, 136, 256	7, 131, 131, 132, 138, 239	20, 77, 87, 95, 143, 256			
Walk bridge	1, 111, 111, 111, 112, 256	1, 3, 116, 118, 245, 254	1, 26, 131, 141, 156, 256			
Mandrill	1, 103, 105, 105, 106, 256	1 18 90 94 102 250	1, 40, 120, 128, 131, 256			
Barbara	1, 131, 132, 133, 137, 256	2, 124, 137, 140, 146, 242	23, 115, 120, 134, 134, 256			
Boat	1, 151, 151, 151, 161, 256	24, 143, 150, 151, 245, 253	1, 100, 113, 124, 256, 256			
Elaine	1, 136, 140, 141, 143, 256	2, 5, 117, 118, 244, 246	1, 1, 149, 152, 187, 256			
Goldhill	1, 108, 108, 109, 110, 256	20, 27, 119, 120, 121, 241	1, 89, 100, 121, 173, 256			
Fingerprint	1, 132, 132, 135, 138, 256	4, 25, 105, 106, 114, 252	1, 1, 142, 145, 149, 256			

is the Cameraman image, where DBBO-Fuzzy produces a slightly worse uniformity factor than DE-Fuzzy. From this point of view also, the DBBO-Fuzzy algorithm stands out as the clear winner.

Table 3.4 presents representative optimal parameter sets $\{a_1, b_1, c_1, a_2, b_2, c_2\}$ obtained by employing DBBO-Fuzzy, BBO-Fuzzy and DE-Fuzzy algorithms.

For a visual interpretation of the three level thresholding results, the thresholded images obtained by applying DBBO-Fuzzy algorithm are presented in Fig. 3.4. After determination of the thresholds for each image, the gray levels of all pixels in a given region are changed to the average gray level of all the pixels belonging to that region.

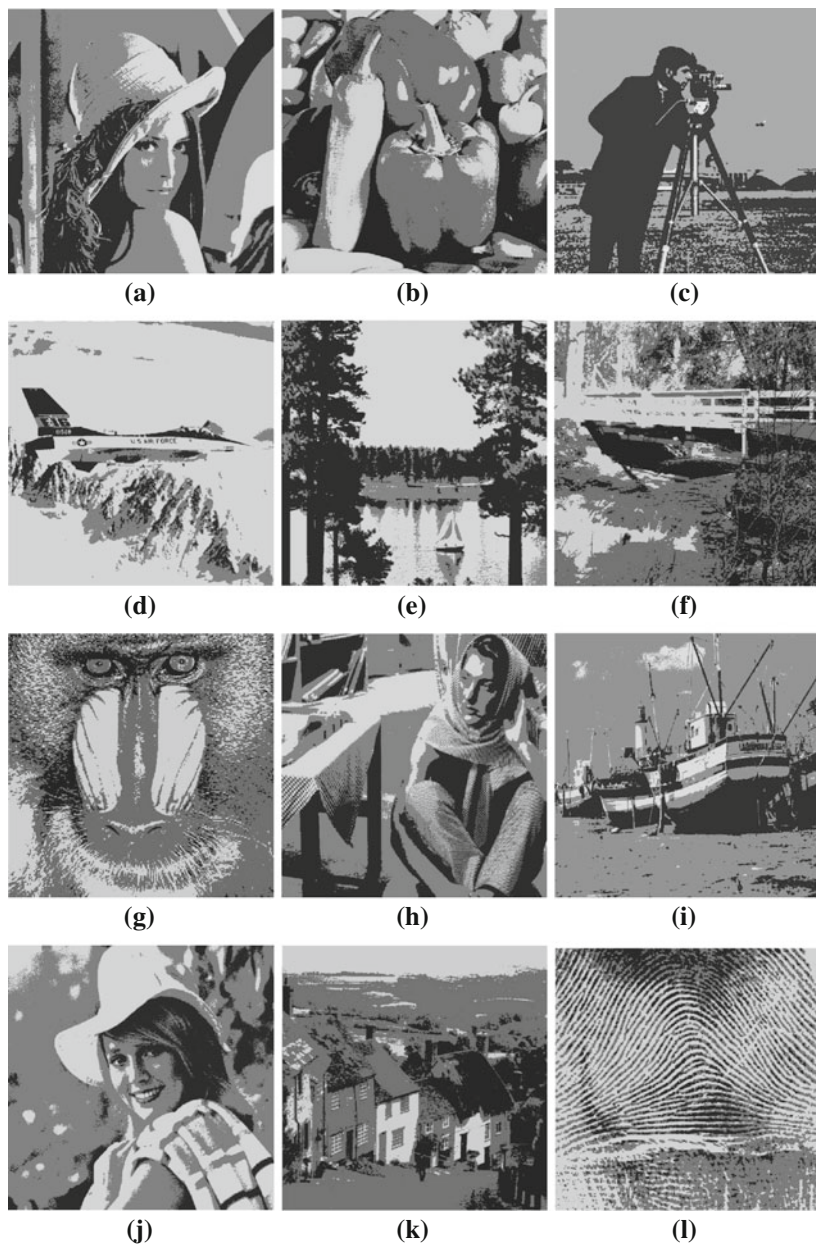


Fig. 3.4 The three-level thresholded images using DBBO-Fuzzy: **a** Lena, **b** Peppers, **c** Cameraman, **d** Airplane, **e** Lake, **f** Walking bridge, **g** Mandrill, **h** Barbara, **i** Boat, **j** Elaine, **k** GoldHill, **l** Fingerprint

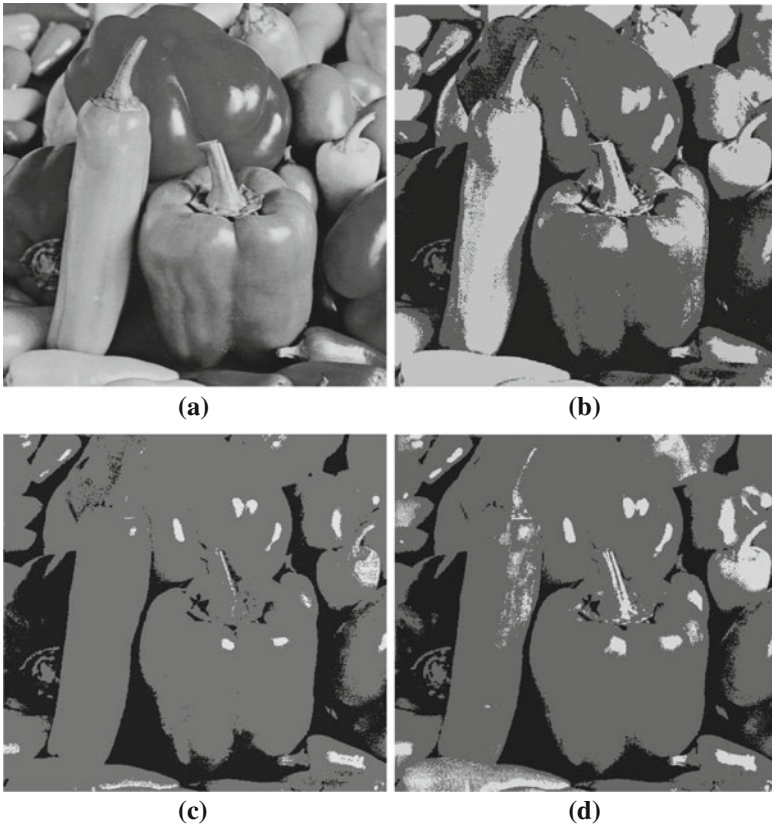


Fig. 3.5 The three-level thresholded images of peppers: **a** original image; **b** thresholded image using DBBO-Fuzzy; **c** thresholded image using BBO-Fuzzy; **d** thresholded image using DE-Fuzzy

Figures 3.5, 3.6 and 3.7 show some sample images under consideration and the resultant segmented images obtained after employing the DBBO-Fuzzy, BBO-Fuzzy and DE-Fuzzy algorithms. From the pictorial representations of the segmented images, it is clear that DBBO-Fuzzy algorithm emerges as the best performer.

3.6.3.1 Effect of the Mutation Strategy

To make a detailed, in-depth study of the DBBO-Fuzzy algorithm, it was tested employing different mutation schemes, described in Sect. 3.5, in order to investigate the effects of the mutation strategy on its performance. The results are reported in Table 3.5. To visualize the best performing scheme, the best values are given in *boldface*.

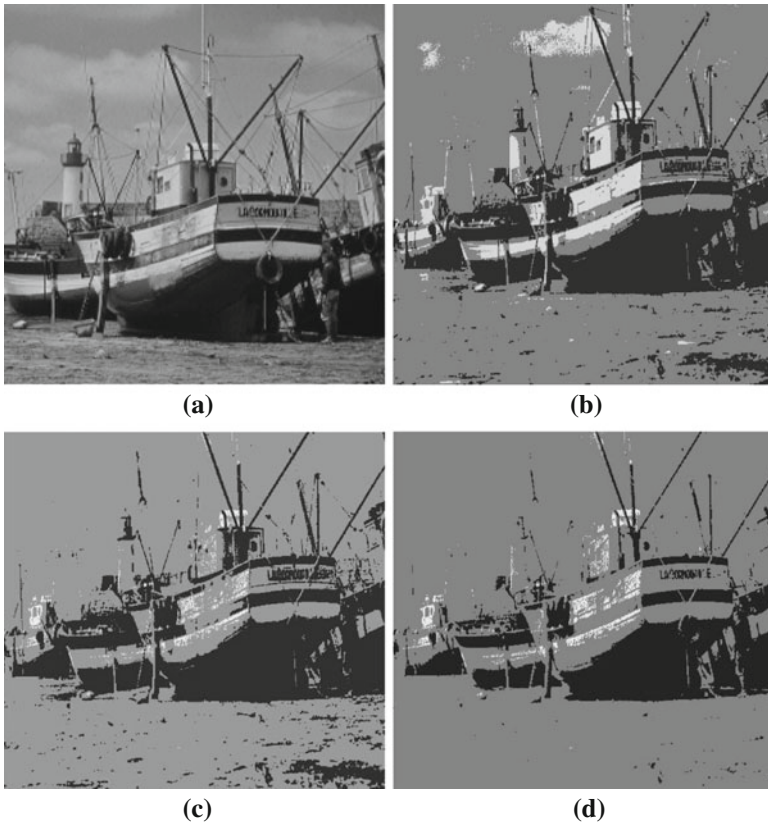


Fig. 3.6 The three-level thresholded images of boat: **a** original image; **b** thresholded image using DBBO-fuzzy; **c** thresholded image using BBO-fuzzy; **d** thresholded image using DE-fuzzy

A closer look at Table 3.5 reveals that out of the 12 test images, the DBBO-Fuzzy algorithm with DE/best/1 mutation strategy emerged as the best candidate algorithm, since it could achieve the highest values of the fuzzy entropy in eight cases (i.e., Lena, Peppers, Cameraman, Airplane, Lake, Mandrill, Elaine and Fingerprint). The DBBO-Fuzzy algorithm with DE/rand/2 proved to be the winner in only two cases (i.e., Walking bridge and Boat). The DBBO-Fuzzy algorithm with DE/rand/1 and DE/current-to-best/1 mutation schemes perform best in only one case each (i.e., GoldHill and Barbara, respectively).

In terms of the best uniformity measure value, DBBO-Fuzzy with DE/rand/1 produced the highest values in 6 test images out of 12 images (i.e., Lena, Cameraman, Walking bridge, Mandrill, Boat and Fingerprint). For the remaining six images (i.e., Peppers, Airplane, Lake, Barbara, Elaine and GoldHill), the DBBO-Fuzzy algorithm with DE/current-to-best/1 mutation scheme achieved the highest uniformity measure values.

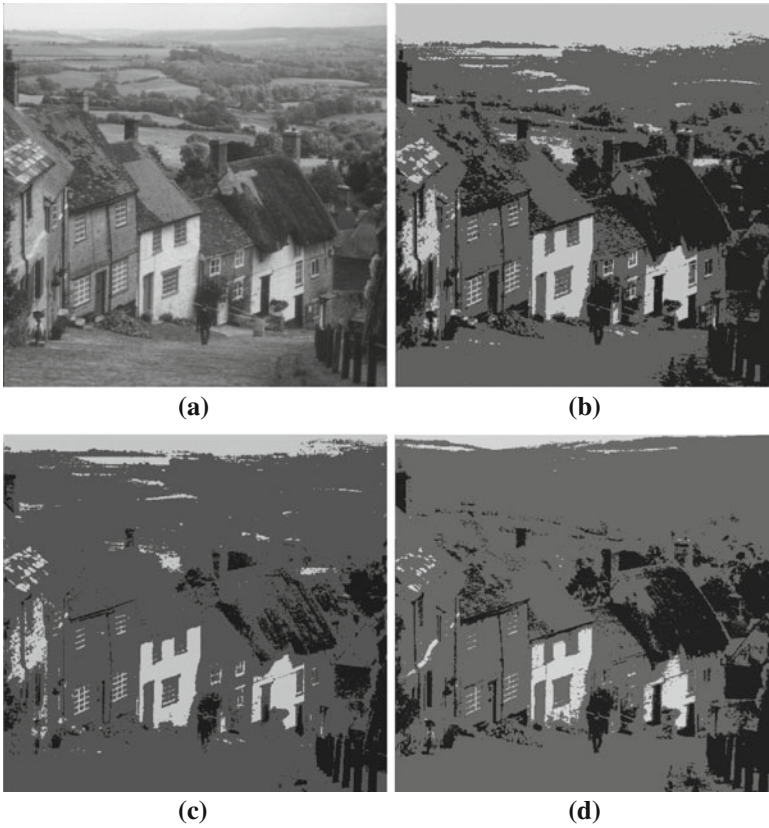


Fig. 3.7 The three-level thresholded images of goldhill: **a** original image; **b** thresholded image using DBBO-fuzzy; **c** thresholded image using BBO-fuzzy; **d** thresholded image using DE-fuzzy

All the variations of the DBBO-Fuzzy algorithm were compared to a default DBBO-Fuzzy algorithm with DE/rand/1, and differences were reported as significant if a two-tailed t-test produced a t-value larger than the critical value. The significance level α is set at 0.05.

Significant differences exist when comparing DBBO-Fuzzy with DE/rand/1 mutation strategy with the variant using DE/current-to-best/1 scheme for two test images (Airplane and Lake). The improvement in the mean objective function value obtained when using DE/best/1 mutation strategy is more significant in the case of Airplane image. For DBBO-Fuzzy with DE/rand/2 mutation, the test has provided a statistically significant difference for the Lake image. In eight out of 12 cases, DBBO-Fuzzy with DE/best/2 mutation proves to be significantly different compared to the variant using DE/rand/1 mutation.

However, in general, it can be noted that the results using different mutation schemes do not significantly affect the performance of the proposed algorithm.

Table 3.5 Comparison of DBBO-fuzzy, BBO-fuzzy and DE-fuzzy, where the best values are given in boldface

Test images	Mutation schemes	Optimal thresholds	Mean Obj \pm Std Dev	Uniformity measure	Optimal parameters
Lena	DE/rand/1	98, 175	1.3851E+01 \pm 5.3148E-02	9.8391E-01	1, 137, 140, 140, 142, 256
	DE/current-to-best/1	93, 169	1.3847E+01 \pm 5.3418E-02	9.8099E-01	1, 131, 132, 132, 133, 256
	DE/best/1	96, 171	1.3889E+01 \pm 6.8730E-02	9.8222E-01	1, 136, 136, 136, 136, 256
	DE/rand/2	99, 174	1.3815E+01 \pm 7.8345E-02	9.8302E-01	1, 139, 140, 140, 141, 256
	DE/best/2	99, 178	1.3764E+01 \pm 8.4071E-02 *	9.7977E-01	1, 138, 140, 145, 146, 256
Peppers	DE/rand/1	85, 162	1.3443E+01 \pm 3.4273E-02	9.7720E-01	1, 119, 122, 122, 124, 256
	DE/current-to-best/1	86, 162	1.3444E+01 \pm 3.9021E-02	9.7728E-01	1, 118, 118, 118, 119, 256
	DE/best/1	83, 158	1.3462E+01 \pm 4.0636E-02	9.7576E-01	1, 117, 117, 117, 117, 256
	DE/rand/2	79, 156	1.3426E+01 \pm 5.9953E-02	9.7660E-01	1, 111, 113, 114, 115, 256
	DE/best/2	88, 164	1.3392E+01 \pm 5.8353E-02 *	9.7469E-01	1, 124, 125, 126, 126, 256
Cameraman	DE/rand/1	127, 207	1.3463E+01 \pm 3.8302E-02	9.6557E-01	1, 177, 180, 180, 194, 256
	DE/current-to-best/1	131, 207	1.3431E+01 \pm 8.2592E-02	9.6261E-01	1, 184, 185, 185, 187, 256
	DE/best/1	129, 207	1.3484E+01 \pm 7.9968E-02	9.6267E-01	1, 182, 182, 182, 192, 256
	DE/rand/2	129, 210	1.3391E+01 \pm 1.2207E-01	9.6487E-01	1, 182, 183, 184, 196, 256
	DE/best/2	124, 233	1.3365E+01 \pm 1.0216E-01 *	9.6514E-01	1, 175, 176, 176, 256, 256
Airplane	DE/rand/1	41, 166	1.3543E+01 \pm 2.5830E-02	9.8719E-01	1, 1, 136, 138, 138, 233
	DE/current-to-best/1	104, 170	1.3571E+01 \pm 1.8272E-02 *	9.8772E-01	1, 146, 147, 147, 147, 227
	DE/best/1	94, 164	1.3580E+01 \pm 2.3179E-02 *	9.8733E-01	1, 133, 133, 133, 134, 239
	DE/rand/2	46, 177	1.3524E+01 \pm 3.4368E-02	9.8662E-01	1, 1, 154, 157, 157, 224
	DE/best/2	98, 172	1.3511E+01 \pm 1.5145E-02 *	9.8743E-01	1, 138, 139, 140, 153, 234
Lake	DE/rand/1	96, 171	1.3817E+01 \pm 2.2140E-02	9.8451E-01	1, 135, 136, 136, 136, 256
	DE/current-to-best/1	98, 172	1.3511E+01 \pm 1.5145E-02 *	9.8743E-01	1, 138, 139, 140, 153, 234
	DE/best/1	94, 169	1.3835E+01 \pm 2.2273E-02	9.8431E-01	1, 132, 133, 133, 133, 256
	DE/rand/2	85, 165	1.3775E+01 \pm 3.1484E-02 *	9.8342E-01	1, 119, 122, 124, 130, 256
	DE/best/2	98, 175	1.3781E+01 \pm 4.6093E-02 *	9.8339E-01	1, 137, 138, 138, 145, 256
Walk bridge	DE/rand/1	79, 154	1.4550E+01 \pm 4.3054E-02	9.7816E-01	1, 111, 111, 111, 112, 256
	DE/current-to-best/1	77, 153	1.4551E+01 \pm 5.9241E-02	9.7610E-01	1, 109, 109, 110, 111, 256

Table 3.5 continue

Test images	Mutation schemes	Optimal thresholds	Mean Obj \pm Std Dev	Uniformity measure	Optimal parameters
Mandrill	DE/best/1	90, 165	1.4555E+01 \pm 6.2526E-02	9.7631E-01	1, 127, 127, 127, 127, 256
	DE/rand/2	86, 166	1.4571E+01 \pm 4.2417E-02	9.7803E-01	1, 118, 125, 126, 130, 256
	DE/best/2	34, 157	1.4502E+01 \pm 3.9284E-02 *	9.7543E-01	1, 1, 113, 115, 116, 256
	Mutation schemes	Optimal thresholds	Mean Obj \pm Std Dev	Uniformity measure	Optimal parameters
Barbara	DE/rand/1	74, 150	1.3815E+01 \pm 4.1507E-02	9.6755E-01	1, 103, 105, 105, 106, 256
	DE/current-to-best/1	31, 147	1.3801E+01 \pm 6.0275E-02	9.6600E-01	1, 1, 102, 102, 102, 256
	DE/best/1	29, 144	1.3845E+01 \pm 1.8303E-02	9.6485E-01	1, 1, 97, 97, 97, 256
	DE/rand/2	30, 145	1.3808E+01 \pm 3.3253E-02	9.6669E-01	1, 1, 99, 99, 100, 256
	DE/best/2	36, 161	1.3759E+01 \pm 4.6532E-02 *	9.6713E-01	1, 1, 119, 119, 123, 256
	DE/rand/1	93, 170	1.4214E+01 \pm 5.7206E-02	9.8233E-01	1, 131, 132, 133, 137, 256
	DE/current-to-best/1	96, 172	1.4235E+01 \pm 3.3796E-02	9.8280E-01	1, 136, 136, 137, 137, 256
	DE/best/1	104, 178	1.4206E+01 \pm 6.4360E-02	9.7831E-01	1, 146, 146, 146, 146, 256
	DE/rand/2	94, 171	1.4228E+01 \pm 3.7467E-02	9.8236E-01	1, 131, 134, 135, 137, 256
Boat	DE/best/2	96, 172	1.4154E+01 \pm 7.4299E-02	9.7929E-01	1, 134, 137, 137, 138, 256
	DE/rand/1	107, 185	1.3986E+01 \pm 8.9041E-02	9.8327E-01	1, 151, 151, 151, 161, 256
	DE/current-to-best/1	99, 221	1.4017E+01 \pm 6.4776E-02	9.8016E-01	1, 140, 140, 140, 252, 256
	DE/best/1	109, 184	1.4019E+01 \pm 8.8207E-02	9.7937E-01	1, 154, 154, 154, 155, 256
	DE/rand/2	100, 223	1.4022E+01 \pm 6.6237E-02	9.8178E-01	1, 137, 144, 144, 256, 256
	DE/best/2	97, 205	1.3921E+01 \pm 8.5438E-02	9.8041E-01	1, 136, 137, 138, 215, 256
	DE/rand/1	98, 175	1.4089E+01 \pm 3.5250E-02	9.7517E-01	1, 136, 140, 141, 143, 256
	DE/current-to-best/1	89, 164	1.4124E+01 \pm 5.6691E-02	9.7828E-01	1, 126, 126, 126, 127, 256
	DE/best/1	91, 165	1.4127E+01 \pm 6.1915E-02	9.7573E-01	1, 128, 128, 128, 128, 256
Elaine	DE/rand/2	100, 178	1.4107E+01 \pm 4.8732E-02	9.7639E-01	1, 139, 144, 146, 146, 256
	DE/best/2	95, 171	1.4078E+01 \pm 7.1696E-02	9.7630E-01	1, 133, 134, 136, 136, 256
	DE/rand/1	77, 152	1.4069E+01 \pm 6.7739E-02	9.8172E-01	1, 108, 108, 109, 110, 256
	DE/current-to-best/1	86, 161	1.4037E+01 \pm 9.6897E-02	9.8176E-01	1, 121, 121, 121, 121, 256
	DE/best/1	94, 169	1.4056E+01 \pm 6.5751E-02	9.8152E-01	1, 133, 133, 133, 133, 256
	DE/rand/2	87, 163	1.4027E+01 \pm 9.1891E-02	9.8148E-01	1, 120, 124, 124, 125, 256
	DE/best/2	81, 159	1.3991E+01 \pm 5.8453E-02 *	9.7927E-01	1, 110, 118, 119, 120, 256

Table 3.5 continue

Fingerprint	DE/rand/1	94, 172	1.2406E+01 ± 4.1010E-02	9.7856E-01	1, 132, 132, 135, 138, 256
	DE/current-to-best/1	76, 151	1.2389E+01 ± 2.3523E-02	9.7463E-01	1, 107, 108, 108, 108, 256
	DE/best/1	85, 160	1.2416E+01 ± 3.3792E-02	9.7398E-01	1, 119, 120, 120, 120, 256
	DE/rand/2	37, 161	1.2396E+01 ± 2.5875E-02	9.7335E-01	1, 2, 121, 121, 122, 256
	DE/best/2	34, 156	1.2336E+01 ± 5.5379E-02 *	9.7322E-01	1, 1, 112, 113, 117, 256

(* the difference with the variant of DBBO-Fuzzy using DE/rand/1 mutation scheme is statistically significant)

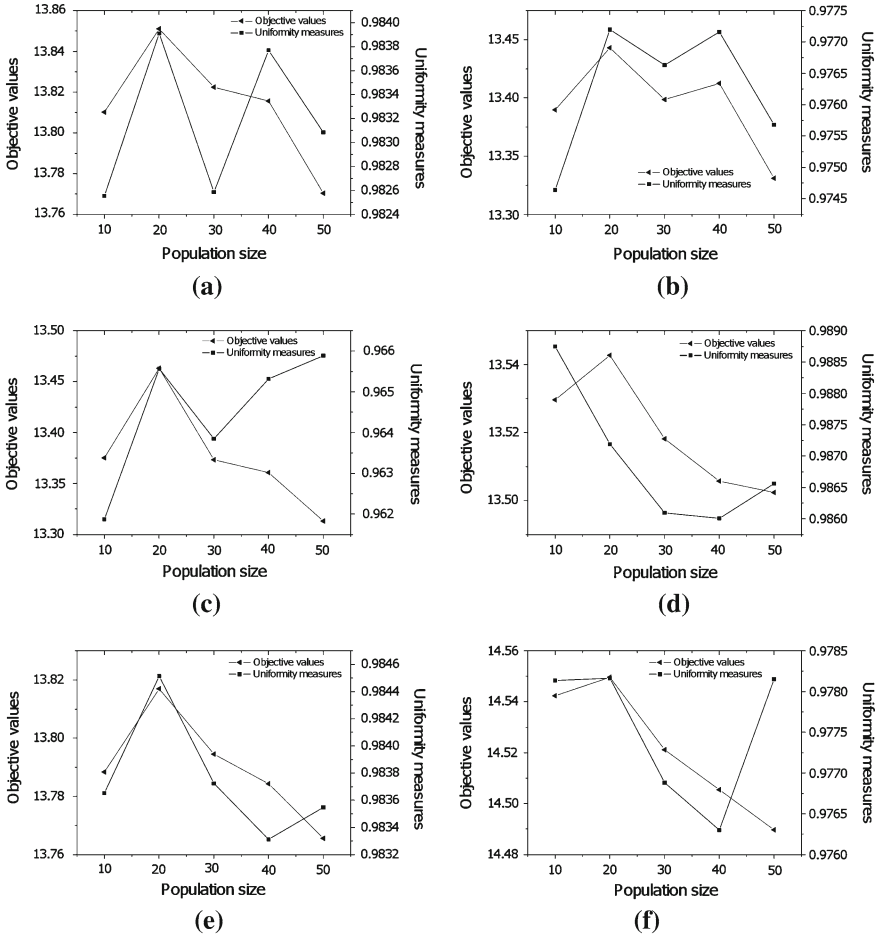


Fig. 3.8 Effect of varying the population size on the performance of DBBO-fuzzy: **a** Lena, **b** Peppers, **c** Cameraman, **d** Airplane, **e** Lake, **f** Walking bridge

3.6.3.2 Effect of the Population Size

Simulations have also been carried out for different values of population size NP , and the performance of the DBBO-Fuzzy algorithm are shown for different variations in NP in Fig. 3.8.

In general, it can be inferred that the DBBO-Fuzzy algorithm with a population size $NP = 20$ outperforms the other variants, since it could achieve the highest value of fuzzy entropy in 11 cases out of 12. Only in one case was DBBO-Fuzzy with $NP = 30$ reveals to be the best in terms of fuzzy entropy value. With $NP = 20$, the DBBO-Fuzzy variant could achieve the highest value of uniformity factor in eight cases. On the other hand, with $NP = 10$, the DBBO-Fuzzy algorithm could achieve

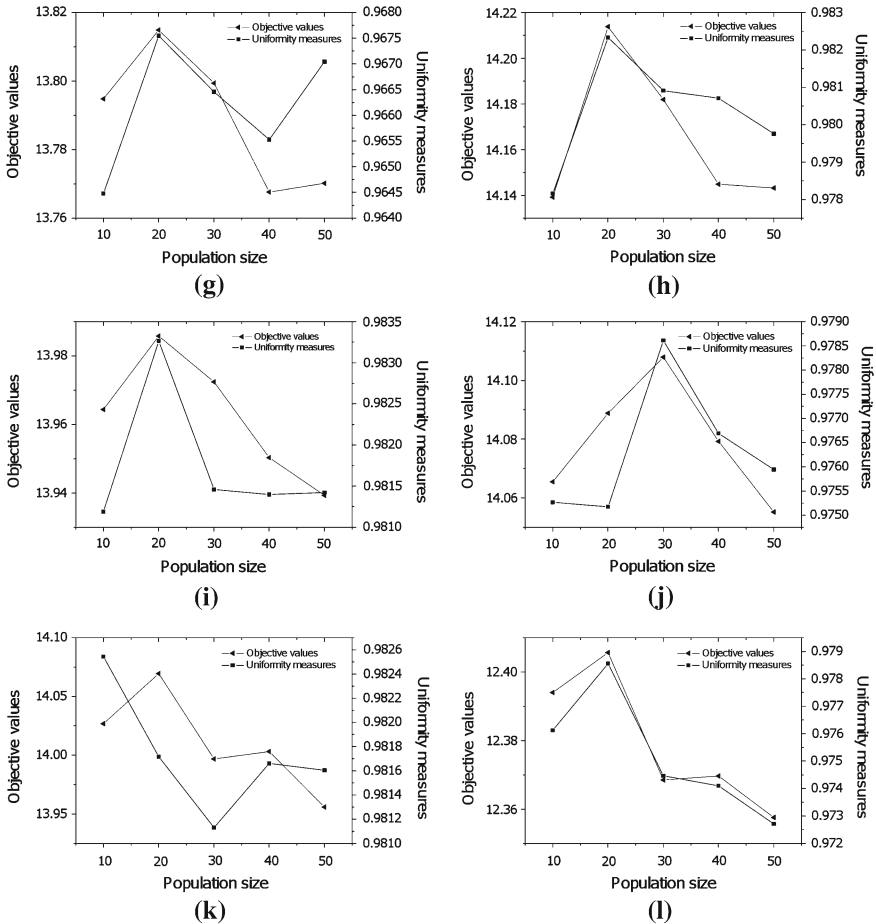


Fig. 3.8 (Cont) Effect of varying the population size on the performance of DBBO-fuzzy: **g** Mandrill, **h** Barbara, **i** Boat, **j** Elaine, **k** GoldHill, **l** Fingerprint

the highest value of uniformity factor in two cases (Airplane and GoldHill) and in one case each with $NP = 50$ (Cameraman) and $NP = 30$ (Elaine).

These results suggest that blindly increasing the population size may not have a relevant positive effect on the performance of the DBBO-Fuzzy algorithm.

3.6.3.3 Effect of the Elitism Parameter

The performance of the proposed algorithm is also evaluated in detail by varying the elitism parameter n_{elit} . In general, one can observe from Fig. 3.9 that the average objective function values tend to increase with the number of elites, although the

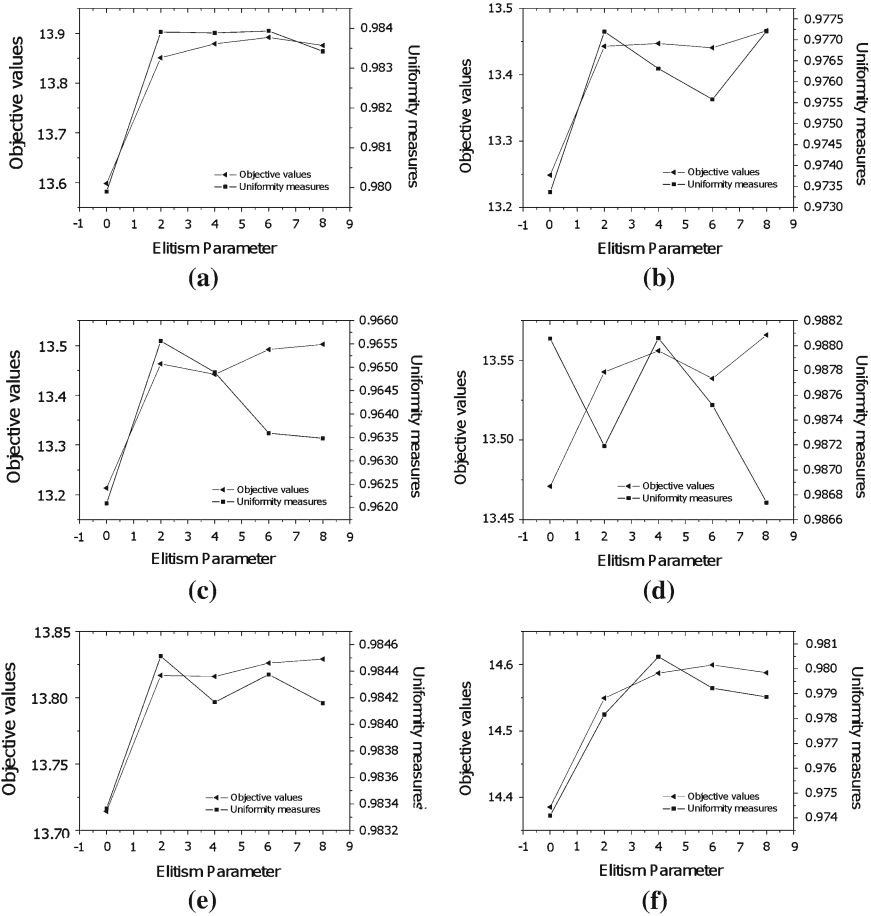


Fig. 3.9 Effect of varying the elitism parameter on the performance of DBBO-fuzzy: **a** Lena, **b** Peppers, **c** Cameraman, **d** Airplane, **e** Lake, **f** Walking bridge

uniformity measure does not follow the same trend in all test images. In nine cases out of 12, DBBO-Fuzzy with $n_{elit} = 8$ produced highest values of the fuzzy entropy. Only in two cases, the DBBO-Fuzzy variant with $n_{elit} = 6$ and in one case the DBBO-Fuzzy variant with $n_{elit} = 4$ achieved the highest values of fuzzy entropy. On the other hand, DBBO-Fuzzy without elitism ($n_{elit} = 0$) could never achieve the highest value for fuzzy entropy but produced good uniformity factor values in three cases out of 12 (i.e., Airplane, Mandrill and Boat).

Finally, an important general remark should be made here that if there is a conflicting choice between a higher fuzzy entropy value and a higher uniformity factor value, higher priority should be given to the solution having higher uniformity factor,

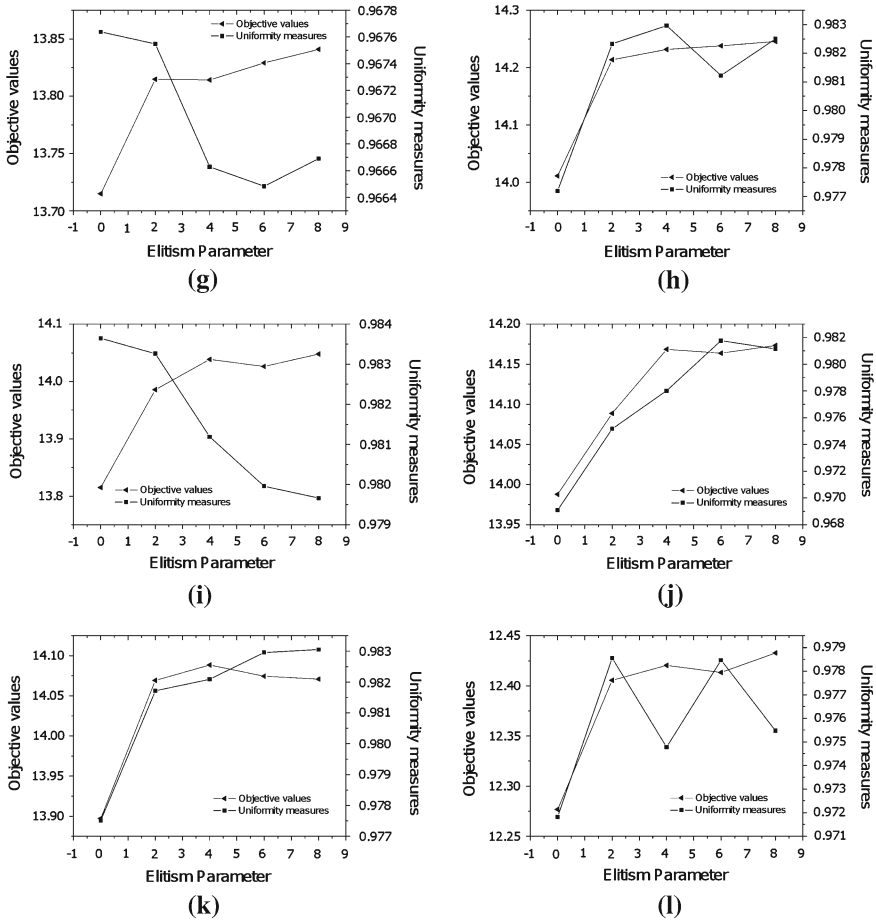


Fig. 3.9 (Cont) Effect of varying the elitism parameter on the performance of DBBO-fuzzy: **g** Mandrill, **h** Barbara, **i** Boat, **j** Elaine, **k** GoldHill, **l** Fingerprint

as it quantitatively reflects the direct impact of the quality of the output segmented image.

3.7 Conclusion

Image segmentation is a process of partitioning an image space into several homogeneous regions. Thresholding is one of the most widely used techniques in image segmentation because of its fast and easy application. However, it has been proven that thresholding often fails to produce satisfactory segmentation results due

to grayness and spatial ambiguity in images. The use of fuzzy set theory could be recognized as an adequate mathematical tool that can be used to model the inherent image vagueness. A new three-level thresholding algorithm based on the hybridization of BBO and the DE algorithms, called the DBBO-Fuzzy algorithm, has been described in this chapter. The DBBO-Fuzzy uses the DE mutation strategy to improve the global search capability and escape from local optima. The experimental results manifest that the proposed algorithm outperforms both BBO and DE algorithms and achieves a high quality of the thresholded images.

The future work will mainly focus on employing a multiobjective approach for such image segmentation problems. Instead of considering a single objective function, a biobjective model could be adopted for the three-level thresholding problem, in which one seeks to optimize simultaneously the total fuzzy entropy and the uniformity factor.

References

1. Acharya, T., Ray, A.K.: *Image Processing—Principles and Applications*. Wiley-Interscience, New Jersey (2005)
2. Bhandari, D., Pal, N.R.: Some new information measures for fuzzy sets. *Inf. Sci.* **67**, 209–228 (1993)
3. Boussaïd, I., Chatterjee, A., Siarry, P., Ahmed-Nacer, M.: Two-stage update biogeography-based optimization using differential evolution algorithm (dbbo). *Comput. Oper. Res.* **38**, 1188–1198 (2011)
4. Chang, C.I., Du, Y., Wang, J., Guo, S.M., Thouin, P.D.: Survey and comparative analysis of entropy and relative entropy thresholding techniques. *IEE Proc. Vis. Image Signal Process.* **153**(6), 837–850 (2006)
5. Cheng, H., Chen, J., Li, J.: Threshold selection based on fuzzy c-partition entropy approach. *Pattern Recogn.* **31**(7), 857–870 (1998)
6. Cover, T.M., Thomas, J.A.: *Elements of Information Theory*. Wiley-Interscience, New York (1991)
7. Darwin, C.: *Origin of Species*. Gramercy, New York (1995)
8. De Luca, A., Termini, S.: A definition of a non-probabilistic entropy in the setting of fuzzy sets theory. *Inf. Control* **20**, 301–312 (1972)
9. Fan, J., Xie, W.: Distance measure and induced fuzzy entropy. *Fuzzy Sets Syst.* **104**, 305–314 (1999)
10. Freixenet, J., Muñoz, X., Raba, D., Martí, J., Cufí, X.: Yet another survey on image segmentation: region and boundary information integration. In: *Proceedings of the 7th European Conference on Computer Vision-Part III, ECCV '02*, pp. 408–422. Springer-Verlag, London, UK (2002)
11. Gong, W., Cai, Z., Ling, C.: DE/BBO: a hybrid differential evolution with biogeography-based optimization for global numerical optimization. *Soft Comput. Fusion Found. Methodol. Appl.* (2010)
12. Kapur, J., Sahoo, P., Wong, A.: A new method for gray-level picture thresholding using the entropy of the histogram. *Comput. Vision Graph. Image Process.* **29**(3), 273–285 (1985)
13. Kaufmann, A.: *Introduction to the Theory of Fuzzy Subsets*. Academic Press, New York (1975)
14. Kaufmann, A.: *Measures of Fuzzy Information*. Mathematical Sciences Trust Society, New Delhi (1997)
15. Klir, G.J., St. Clair, U., Yuan, B.: *Fuzzy Set Theory: Foundations and Applications*. Prentice-Hall, Inc., Upper Saddle River (1997)

16. Levine, M., Nazif, A.: Dynamic measurement of computer generated image segmentations. *IEEE Trans. Pattern Anal. Mach. Intell.* **7**, 155–164 (1985)
17. MacArthur, R., Wilson, E.: *The Theory of Biogeography*. Princeton University Press, Princeton (1967)
18. Noman, N., Iba, I.: Accelerating differential evolution using an adaptive local search. *IEEE Trans. Evol. Comput.* **12**(1), 107–125 (2008)
19. Pal, N.R., Pal, S.K.: Higher order fuzzy entropy and hybrid entropy of a set. *Inf. Sci.* **61**, 211–231 (1992)
20. Pal, N.R., Pal, S.K.: A review on image segmentation techniques. *Pattern Recognit.* **26**(9), 1277–1294 (1993)
21. Price, K.V., Storn, R.M., Lampinen, J.A.: *Differential Evolution: A Practical Approach to Global Optimization*. Natural Computing Series. Springer, Berlin (2005)
22. Pun, T.: A new method for gray-level picture thresholding using the entropy of histogram. *Signal Process.* **2**(3), 223–237 (1980)
23. Pun, T.: Entropic thresholding, a new approach. *Comput. Graph. Image Process.* **16**(3), 210–239 (1981)
24. Sahoo, P.K., Soltani, S., Wong, A.K., Chen, Y.C.: A survey of thresholding techniques. *Comput. Vision Graph. Image Process.* **41**, 233–260 (1988)
25. Sahoo, P.K., Wilkins, C., Yeager, J.: Threshold selection using Renyi's entropy. *Pattern Recognit.* **30**(i1), 71–84 (1997)
26. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. *J. Electron. Imaging* **13**(1), 146–168 (2004)
27. Shannon, C.E.: *A Mathematical theory of communication*. CSLI Publications (1948)
28. Simon, D.: Biogeography-based optimization. *IEEE Trans. Evol. Comput.* **12**, 702–713 (2008)
29. Storn, R.M., Price, K.V.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **11**, 341–359 (1997)
30. Tao, W., Jin, H., Liu, L.: Object segmentation using ant colony optimization algorithm and fuzzy. entropy **28**(7), 788–796 (2007)
31. Tao, W., Tian, J., Liu, J.: Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm. *Pattern Recogn. Lett.* **24**(16), 3069–3078 (2003)
32. Tobias, O., Seara, R.: Image segmentation by histogram thresholding using fuzzy sets. *IEEE Trans. Image Process.* **11**(12), 1457–1465 (2002)
33. Tsallis, C.: Possible generalization of Boltzmann-Gibbs statistics. *J. Stat. Phys.* **52**, 479–487 (1988)
34. Wallace, A.R.: *The Geographical Distribution of Animals* (two volumes). Adamant Media Corporation, Boston (2005)
35. Zadeh, L.A.: Fuzzy sets. *Inf. Control* **8**, 338–353 (1965)
36. Zhao, M., Fu, A., Yan, H.: A technique of three-level thresholding based on probability partition and fuzzy 3-partition. *IEEE Trans. Fuzzy Syst.* **9**(3), 469–479 (2001)