Chapter 2
Multilevel Segmentation in Digital Images

Abstract Segmentation is used to divide an image into separate regions, which in fact correspond to different real-world objects. One interesting functional criterion for segmentation is the Tsallis entropy (TE), which gives excellent results in bi-level thresholding. However, when it is applied to multilevel thresholding (MT), its evaluation becomes computationally expensive, since each threshold point adds restrictions, multimodality and complexity to its functional formulation. In this chapter, a new algorithm for multilevel segmentation based on the Electromagnetism-Like algorithm (EMO) is presented. In the approach, the EMO algorithm is used to find the optimal threshold values by maximizing the Tsallis entropy. Experimental results over several images demonstrate that the proposed approach is able to improve the convergence velocity, compared with similar methods such as Cuckoo search, and Particle Swarm Optimization.

2.1 Introduction

Segmentation is one of the basic steps of an image analysis system, and consists in separating objects from each other, by considering characteristics contained in a digital image [1]. It has been applied to feature extraction [2], object identification and classification [3], surveillance [4], among other areas. In order to obtain homogeneous regions of pixels, a common method is using the histogram´s information with a thresholding approach [5]. This method is considered the easiest one in segmentation, and it works taking threshold values which separate adequately the distinct regions of pixels in the image being processed. In general, there are two thresholding approaches, namely bi-level and multilevel. In bi-level thresholding (BT), it is only needed a threshold value to separate the two objects of an image (e.g. foreground and background). For real life images, BT doesn't provide appropriate results. On the other hand, multilevel thresholding (MT) divides the pixels in more than two homogeneous classes and it needs several

E. Cuevas et al., *Evolutionary Computation Techniques: A Comparative Perspective*, Studies in Computational Intelligence 686,
DOI 10.1007/978-3-319-51109-2_2

threshold values [5, 6]. Threshold methods are divided in parametric and non-parametric [6, 7]. In parametric approaches, it is necessary estimating the parameters of a probability density function capable of modelling each class. Such an approach is time consuming and computationally expensive. A nonparametric technique employs a given criteria (between-class variance, entropy and error rate [9–8]) which must be optimized to determine the optimal threshold values. These approaches result an attractive option due their robustness and accuracy [9].

For bi-level thresholding there exist two classical methods: the first one, proposed by Otsu [10], maximizes the variance between classes, whereas the second one, proposed by Kapur in [11], uses the entropy maximization to measure the homogeneity among classes. Their efficiency and accuracy have been already proved by segmenting pixels into two classes [12]. Both methods, Otsu and Kapur, can be expanded for multilevel thresholding; however, their computational complexity is increased, and also its accuracy decreases with each new threshold added into the searching process [12, 13].

The Tsallis entropy (TE), proposed in [14], is known as the non-extensive entropy, and can be considered as an extension of Shannon's entropy. Recently, there exist several studies that report similarities among the Tsallis, the Shannon and the Boltzmann-Gibbs entropies [13, 16–15]. Different to the Otsu and Kapur methods, the Tsallis entropy produces a functional formulation whose accuracy does not depend on the number of threshold points [16]. In the process of image segmentation, under the TE perspective, it is selected a set of threshold values that maximize the TE functional formulation, so that each pixel is assigned to a determined class according to its corresponding threshold points. TE gives excellent results in bi-level thresholding. However, when it is applied to multilevel thresholding (MT), its evaluation becomes computationally expensive, since each threshold point adds restrictions, multimodality and complexity to its functional formulation. Therefore, in the process of finding the appropriate threshold values, it is desired to limit the number of evaluations of the TE objective function. Under such circumstances, most of the optimization algorithms do not seem to be suited to face such problems as they usually require many evaluations before delivering an acceptable result.

As an alternative to traditional thresholding techniques, the problem of MT has also been handled through evolutionary methods. In general, they have demonstrated, under several circumstances, to deliver better results than those based on deterministic approaches in terms of accuracy and robustness [17, 18]. Under such conditions, recently, an extensive amount of evolutionary optimization approaches have been reported in the literature to find the appropriate threshold values by maximizing the complex objective function produced by Tsallis entropy. Such approaches have produced several interesting segmentation algorithms using different optimization methods such as Differential evolution (DE) [5], Particle Swarm Optimization algorithm (PSO) [19], Artificial Bee Colony (ABC) [16], Cuckoo Search algorithm (CSA) [13], Bacterial Foraging Optimization (BFOA) [12] and Harmony Search (HS) [20]. All these approaches permit with different results to optimize the TE fitness function in despite of its high multimodality characteristics.

However, one particular difficulty in their performance is the demand for a large number of fitness evaluations before delivering a satisfying result.

This chapter presents a multilevel thresholding method that uses the Electromagnetism-Like Algorithm (EMO) to find the best threshold values. EMO is a population-based evolutionary method which was firstly introduced by Birbil and Fang [21] to solve unconstrained optimization problems. The algorithm emulates the attraction–repulsion mechanism between charged particles within an electromagnetism field. Each particle represents a solution and carries a certain amount of charge which is proportional to its fitness value. In turn, solutions are defined by position vectors which give real positions for particles within a multi-dimensional space. Moreover, objective function values of particles are calculated considering such position vectors. Each particle exerts repulsion or attraction forces over other members in the population; the resultant force acting over a particle is used to update its position. Clearly, the idea behind the EMO methodology is to move particles towards the optimum solution by exerting attraction or repulsion forces among them. Different to other evolutionary methods, EMO exhibits interesting search capabilities such as fast convergence still keeping its ability to avoid local minima in high modality environments [27–22]. Recent studies [30–23] demonstrate that the EMO algorithm presents the best balance between optimization results and demand of function evaluations. Such characteristics have attracted the attention of the evolutionary computation community, so that it has been effectively applied to solve a wide range of engineering problems such as flow-shop scheduling [24], communications [25], vehicle routing [26], array pattern optimization in circuits [27], neural network training [28], image processing [29] and control systems [30].

In this chapter, a new algorithm for multilevel segmentation based on the Electromagnetism-Like algorithm (EMO) is presented. In the approach, the EMO algorithm is used to find the optimal threshold values by maximizing the Tsallis entropy. As a result, the proposed algorithm can substantially reduce the number of function evaluations preserving the good search capabilities of an evolutionary method. In our approach, the algorithm uses as particles the encoding of a set of candidate threshold points. The TE objective function evaluates the segmentation quality of the candidate threshold points. Guided by the values of this objective function, the set of encoded candidate solutions are modified by using the EMO operators so that they can improve their segmentation quality as the optimization process evolves. In comparison to other similar algorithms, the proposed method deploys better segmentation results yet consuming less TE function evaluations.

The rest of the chapter is organized as follows. In Sect. 2.2, the standard EMO algorithm is introduced. Section 2.3 gives a simple description of the Tsallis entropy method. Section 2.4 explains the implementation of the proposed algorithm. Section 2.5 discusses experimental results and comparisons after testing the proposal over a set of benchmark images. Finally, in Sect. 2.6 the conclusions are discussed.

2.2 Electromagnetism—Like Optimization Algorithm (EMO)

EMO is a population-based evolutionary method which was firstly introduced by Birbil and Fang [31] to solve unconstrained optimization problems. Different to other evolutionary methods, EMO exhibits interesting search capabilities such as fast convergence still keeping its ability to avoid local minima in high modality environments [27–22]. Recent studies [30–23] demonstrate that the EMO algorithm presents the best balance between optimization results and demand of function evaluations. From the implementation point of view, EMO utilizes N different n-dimensional points $x_{i,t}, \quad i = 1, 2, \ldots, n$, as a population for searching the feasible set $\mathbf{X} = \{x \in \mathbb{R}^n | l_i \leq x \leq u_i\}$, where t denotes the number of iteration (or generation) of the algorithm. The initial population $\mathbf{Sp}_t = \{x_{1,t}, x_{2,t}, \ldots, x_{N,t}\}$ (being $t = 1$), is taken of uniformly distributed samples of the search region, $\mathbf{X}$. We denote the population set at the t-th iteration by $\mathbf{Sp}_t$, and the members of $\mathbf{Sp}_t$ changes with t. After the initialization of $\mathbf{Sp}_t$, EMO continues its iterative process until a stopping condition (e.g. the maximum number of iterations) is met. An iteration of EMO consists of two main steps: in the first step, each point in $\mathbf{Sp}_t$ moves to a different location by using the attraction-repulsion mechanism of the electromagnetism theory [29]. In the second step, points moved by the electromagnetism principle are further perturbed locally by a local search and then become members of $\mathbf{Sp}_{t+1}$ in the $(t + 1)$-th iteration. Both the attraction-repulsion mechanism and the local search in EMO are responsible for driving the members, $x_{i,t}$, of $\mathbf{Sp}_t$ to the close proximity of the global optimum.

As with the electromagnetism theory for charged particles, each point $x_{i,t} \in \mathbf{Sp}_t$ in the search space $\mathbf{X}$ is assumed as a charged particle where the charge of a point is computed based on its objective function value. Points with better objective function value have higher charges than other points. The attraction-repulsion mechanism is a process in EMO by which points with more charge attract other points from $\mathbf{Sp}_t$, and points with less charge repel other points. Finally, a total force vector F_i^t, exerted on a point (e.g. the i-th point $x_{i,t}$) is calculated by adding these attraction—repulsion forces, and each $x_{i,t} \in \mathbf{Sp}_t$ is moved in the direction of its total force to the location $y_{i,t}$. A local search is used to explore the vicinity of the each particle according to its fitness. The members, $x_{i,t+1} \in \mathbf{Sp}_{t+1}$, of the $(t + 1)$-th iteration are then found by using:

$$x_{i,t+1} = \begin{cases} y_{i,t} & \text{if} \quad f(y_{i,t}) \leq f(z_{i,t}) \\ z_{i,t} & \text{otherwise} \end{cases} \tag{2.1}$$

Algorithm 2.1 shows the general scheme of EMO. We also provided the description of each step following the algorithm.

Algorithm 2.1 [EMO $(N, Iter_{\max}, Iter_{local}, \delta)$]

1. Input parameters: Maximum number of iteration $Iter_{\max}$, values for the local search parameter such $Iter_{local}$ and δ, and the size N of the population.
2. Initialize: set the iteration counter $t = 1$, initialize the number of $\mathbf{Sp}_t$ uniformly in $\mathbf{X}$ and identify the best point in $\mathbf{Sp}_t$.
3. while $t < Iter_{\max}$ do
4. $F_i^t \leftarrow \text{CalcF}(\mathbf{Sp}_t)$
5. $y_{i,t} \leftarrow \text{Move}(x_{i,t}, F_i^t)$
6. $z_{i,t} \leftarrow \text{Local}(Iter_{local}, \delta, y_{i,t})$
7. $x_{i,t+1} \leftarrow \text{Select}(\mathbf{Sp}_{t+1}, y_{i,t}, z_{i,t})$
8. end while

Input parameters (Line 1): EMO algorithm runs for $Iter_{\max}$ iterations. In the local search phase, $n \times Iter_{local}$ is the maximum number of locations $z_{i,t}$, within a δ distance of $y_{i,t}$, for each i dimension.

Initialize (Line 2): The points $x_{i,t}$, $t = 1$, are selected uniformly in $\mathbf{X}$, i.e. $x_{i,1} \sim Unif(\mathbf{X}), \quad i = 1, 2, \ldots, N$, where *Unif* represents the uniform distribution. The objective function values $f(x_{i,t})$ are computed, and the best point is identified for minimization:

$$x_t^B = \arg \min_{x_{i,t} \in \mathbf{S}_t} \{f(x_{i,t})\} \tag{2.2}$$

and for maximization:

$$x_t^B = \arg \max_{x_{i,t} \in \mathbf{S}_t} \{f(x_{i,t})\} \tag{2.3}$$

Calculate force (Line 4): In this step, a charged-like value $(q_{i,t})$ is assigned to each point $(x_{i,t})$. The charge $q_{i,t}$ of $x_{i,t}$ depends on $f(x_{i,t})$ and points with better objective function have more charge than others. The charges are computed as follows:

$$q_{i,t} = \exp\left(-n \frac{f(x_{i,t}) - f(x_t^B)}{\sum_{j=1}^{N} f(x_{i,t}) - f(x_t^B)}\right) \tag{2.4}$$

Then the force, $F_{i,j}^t$, between two points $x_{i,t}$ and $x_{j,t}$ is calculated using:

$$F_{i,j}^t = \begin{cases} (x_{j,t} - x_{i,t}) \frac{q_{i,t} \cdot q_{j,t}}{\left\| x_{j,t} - x_{i,t} \right\|^2} & \text{if} \quad f(x_{i,t}) > f(x_{j,t}) \\ (x_{i,t} - x_{j,t}) \frac{q_{i,t} \cdot q_{j,t}}{\left\| x_{j,t} - x_{i,t} \right\|^2} & \text{if} \quad f(x_{i,t}) \leq f(x_{j,t}) \end{cases} \tag{2.5}$$

The total force, F_i^t, corresponding to $x_{i,t}$ is now calculated as:

$$F_i^t = \sum_{j=1, j\neq i}^{N} F_{i,j}^t \tag{2.6}$$

Move the point $x_{i,t}$ along F_i^t (Line 5): In this step, each point $x_{i,t}$ except for x_t^B is moved along the total force F_i^t using:

$$x_{i,t} = x_{i,t} + \lambda \frac{F_i^t}{\|F_i^t\|}(RNG), \quad i = 1, 2, \ldots, N;\ i \neq B \tag{2.7}$$

where $\lambda \sim Unif(0, 1)$ for each coordinate of $x_{i,t}$, and RNG denotes the allowed range of movement toward the lower or upper bound for the corresponding dimension.

Local search (Line 6): For each $y_{i,t}$ a maximum of $iter_{local}$ points are generated in each coordinate direction in the δ neighbourhood of $y_{i,t}$. This means that the process of generating local point is continued for each $y_{i,t}$ until either a better $z_{i,t}$ is found or the $n \times Iter_{local}$ trial is reached.

Selection for the next iteration (Line 7): In this step, $x_{i,t+1} \in \mathbf{Sp}_{t+1}$ are selected from $y_{i,t}$ and $z_{i,t}$ using Eq. (2.1), and the best point is identified using Eq. (2.2) for minimization or Eq. (2.3) for maximization.

As it can be seen from Eqs. 2.1–2.8, the process to compute the elements of the new population $\mathbf{Sp}_{t+1}$ involves several operations that consider local and global information. Such process is more laborious than most of the evolutionary approaches which use only one equation to modify the individual position. This fact could be considered as an implementation disadvantage of the EMO method.

2.3 Tsallis Entropy (TE)

The entropy is defined in thermodynamic to measure the order of irreversibility in the universe. The concept of entropy physically expresses the amount of disorder of a system [13, 16]. In information theory, Shannon redefines the theory proposed by Boltzmann-Gibbs, and employ the entropy to measure the uncertainty regarding information of a system [16]. In other words, is possible the quantitatively measurement of the amount of information produced by a process.

The entropy in a discrete system takes a probability distribution defined as $p = \{p_i\}$, which represents the probability of find the system in a possible state i. Notice that $0 \leq p_i \leq 1, \sum_{i=1}^{k} p_i = 1$, and k is the total number of states. In addition, a physical or information system can be decomposed in two statistical independent subsystems A and B with probabilities p^A and p^B, where the probability of the composed system is given by $p^{A+B} = p^A \cdot p^B$. Such definition has been verified using the extensive property (additive) Eq. (2.8) proposed by Shannon [13, 16]:

$$S(A+B) = S(A) + S(B) \tag{2.8}$$

Tsallis proposed a generalized form of statistics based on the related concepts and the multi-fractal theory. The Tsallis entropic form is an important tool used to describe the thermo statistical properties of non-extensive systems and is defined as:

$$S_q = \frac{1 - \sum_{i=1}^{k} (p_i)^q}{q-1} \tag{2.9}$$

where S is the Tsallis entropy, q is the Tsallis entropic index that represents the degree of non-extensivity and k is the total number of possibilities of the system. Since Tsallis entropy is non-extensive, it is necessary to redefine the additive entropic rule of Eq. (2.8).

$$S_q(A+B) = S_q(A) + S_q(B) + (1-q) \cdot S_q(A) \cdot S_q(B) \tag{2.10}$$

Since image segmentation has non-additive information content, it is possible to use the Tsallis entropy to find the best thresholds [16]. A digital gray scale image has k gray levels that are defined by the histogram. The easiest thresholding considers to classes divided by a one threshold (bi-level), to solve this problem is considered the probability distribution of the gray levels ($p_i = p_1, p_2, \ldots p_k$). For each class A and B two probability distributions are created Eq. (2.11)

$$p_A = \frac{p_1}{P^A}, \frac{p_2}{P^A}, \ldots \frac{p_{th}}{P^A} \quad \text{and} \quad p_B = \frac{p_1}{P^B}, \frac{p_2}{P^B}, \ldots \frac{p_k}{P^B} \tag{2.11}$$

where

$$P^A = \sum_{i=1}^{th} p_i \quad \text{and} \quad P^B = \sum_{i=th+1}^{k} p_i \tag{2.12}$$

The TE for class A and Class B is defined as follows:

$$S_q^A(th) = \frac{1 - \sum_{i=1}^{th} \left(\frac{p_i}{P^A}\right)^q}{q-1}, \quad S_q^B(th) = \frac{1 - \sum_{i=th+1}^{k} \left(\frac{p_i}{P^B}\right)^q}{q-1} \tag{2.13}$$

TE value depends directly on the selected threshold value, and it maximizes the information measured between two classes. If the value of $S_q(th)$ is maximized it means that th is the optimal value. In order to verify the efficiency of the selected th, in Eq. (2.14) is proposed an objective function using Eq. (2.10):

$$TH_{opt}(th) = \arg\max \left(S_q^A(th) + S_q^B(th) + (1-q) \cdot S_q^A(th) \cdot S_q^B(th) \right) \tag{2.14}$$

The previous description of this bi-level method can be extended for the identification of multiple thresholds. Considering *nt* thresholds, it is possible separate the original image into (*nt-1*) classes. Under the segmentation approach, the optimization problem turns into a multidimensional situation. The candidate solutions are conformed as $\mathbf{th}^j = [th_1, th_2, \ldots th_{nt}]$. For each class is computed the entropy using the Tsallis methodology and the objective function is redefined as follows:

$$\begin{aligned} &TH_{opt}(\mathbf{th}) = \arg\max(L) \\ &L = S_q^1(th_1) + S_q^2(th_2) + \cdots + S_q^{nt}(th_{nt}) + (1-q) \cdot S_q^1(th_1) \cdot S_q^2(th_2) \ldots S_q^{nt}(th_{nt}) \end{aligned} \tag{2.15}$$

where

$$\begin{aligned} S_q^1(th_1) = \frac{1 - \sum_{i=1}^{th_1} \left(\frac{p_i}{P^1}\right)^q}{q-1}, \quad S_q^2(th) = \frac{1 - \sum_{i=th_1+1}^{th_2} \left(\frac{p_i}{P^2}\right)^q}{q-1}, \ldots, \\ S_q^{nt}(th) = \frac{1 - \sum_{i=th_{nt-1}+1}^{th_{nt}} \left(\frac{p_i}{P^{nt}}\right)^q}{q-1} \end{aligned} \tag{2.16}$$

Notice that for each threshold the entropy is computed and corresponds to a specific class. However there exist an extra class it means that exist $nt+1$ classes. The extra class is considered *default* class because it is computed from *nt* to *k* Eq. (2.17).

$$S_q^{def}(th_k) = \frac{1 - \sum_{i=th_{nt}+1}^{k} \left(\frac{p_i}{P^k}\right)^q}{q-1} \tag{2.17}$$

From Eq. 2.14, it is evident that TE presents a simple functional formulation for bi-level thresholding. However, as it is shown by Eqs. 2.15 and 2.16, when it is considered multilevel thresholding (MT), its evaluation becomes computationally expensive, since each threshold point adds restrictions, multimodality and complexity to its functional formulation. Therefore, in the process of finding the appropriate threshold values, it is desired to limit the number of evaluations of the TE objective function. Under such circumstances, most of the optimization algorithms do not seem to be suited to face such problems as they usually require many evaluations before delivering an acceptable result.

2.4 Multilevel Thresholding Using EMO and Tsallis Entropy (TSEMO)

In this chapter, a new algorithm for multilevel segmentation based on the Electromagnetism-Like algorithm (EMO) is presented. In the approach, the EMO algorithm is used to find the optimal threshold values by maximizing the complex Tsallis entropy. Different to other evolutionary methods, EMO exhibits interesting search capabilities such as fast convergence still keeping its ability to avoid local minima in high modality environments [27–22]. Recent studies [30–23] demonstrate that the EMO algorithm presents the best balance between optimization results and demand of function evaluations. As a result, the proposed segmentation algorithm can substantially reduce the number of function evaluations preserving the good search capabilities of an evolutionary method. However, as it can be seen from Eqs. 2.1–2.8, the process of EMO, to compute the elements of the new population, involves several operations that consider local and global information. Such process is more laborious than most of the evolutionary approaches which use only one equation to modify the individual position. This fact could be considered as an implementation disadvantage of the EMO method. In this section, the proposed approach is discussed.

2.4.1 Particle Representation

Each particle uses *nt* decision variables in the optimization algorithm. Such elements represent a different threshold point used for the segmentation. Therefore, the complete population is represented as:

$$\mathbf{Sp}_t = [\mathbf{th}_1, \mathbf{th}_2, \ldots, \mathbf{th}_N], \quad \mathbf{th}_i = [th_1, th_2, \ldots, th_{nt}]^T \tag{2.18}$$

where t represents the iteration number, T refers to the transpose operator, N is the size of the population.

2.4.2 EMO Implementation

The proposed segmentation algorithm has been implemented considered the Tsallis pseudo-additive entropic rule as objective function Eq. (2.15). The implementation of the EMO algorithm can be summarized into the following steps:

Step 1 Read the image I and store it into I_{Gr}.
Step 2 Obtain histogram h^{Gr} of I_{Gr}.
Step 3 Initialize the EMO parameters: $Iter_{\max}$, $Iter_{local}$, δ, k and N.
Step 4 Initialize a population $\mathbf{Sp}_t$ of N random particles with nt dimensions.
Step 5 Compute the Tsallis entropy $S_q^i(\mathbf{Sp}_t)$ for each element of $\mathbf{Sp}_t$, Eqs. (2.16) and (2.17). Evaluate $\mathbf{Sp}_t$ in the objective function $TH_{opt}(\mathbf{Sp}_t)$ Eq. (2.15).
Step 6 Compute the charge of each particle using Eq. (2.4), and with Eqs. (2.5) and (2.6) compute the total force vector.
Step 7 Move the entire population $\mathbf{Sp}_t$ along the total force vector using Eq. (2.7).
Step 8 Apply the local search to the moved population and select the best elements of this search based on their objective function values.
Step 9 The t index is increased in 1, If $t \geq Iter_{\max}$ or if the stop criteria is satisfied the algorithm finishes the iteration process and jump to step 11. Otherwise jump to step 7.
Step 10 Select the particle that has the best x_t^{Bc} objective function value Eqs. (2.3) and (2.15).
Step 11 Apply the thresholds values contained in x_t^{Bc} to the image I_{Gr}.

2.4.3 Multilevel Thresholding

Ones the EMO algorithm finds the best threshold values that maximize the objective function. These are used to segment the image pixels. There exist several ways to apply the thresholds, in this chapter we use the following rule for two levels:

$$I_s(r,c) = \begin{cases} I_{Gr}(r,c) & \text{if} \quad I_{Gr}(r,c) \leq th_1 \\ th_1 & \text{if} \quad th_1 < I_{Gr}(r,c) \leq th_2 \\ I_{Gr}(r,c) & \text{if} \quad I_{Gr}(r,c) > th_2 \end{cases} \tag{2.19}$$

where $I_s(r,c)$ is the gray value of the segmented image, $I_{Gr}(r,c)$ is the gray value of the original image both in the pixel position r,c. th_1 and th_2 are the threshold values obtained by the EMO approach. Equation (2.19) can be easily extended for more than two levels Eq. (2.20).

$$I_s(r,c) = \begin{cases} I_{Gr}(r,c) & \text{if} \quad I_{Gr}(r,c) \leq th_1 \\ th_{i-1} & \text{if} \quad th_{i-1} < I_{Gr}(r,c) \leq th_i\,, \quad i = 2,3,\ldots nt-1 \\ I_{Gr}(r,c) & \text{if} \quad I_{Gr}(r,c) > th_{nt} \end{cases} \tag{2.20}$$

2.5 Experimental Results

The proposed algorithm has been tested under a set of 11 benchmark images. Some of these images are widely used in the image processing literature to test different methods (Lena, Cameraman, Hunter, Baboon, etc.) [13]. All the images have the same size (512 × 512 pixels) and they are in JPGE format.

In order to carry out the algorithm analysis the proposed TSEMO is compared to state-of-the-art thresholding methods, such Cuckoo Search algorithm (CSA) [13] and Particle Swarm Optimization (PSO) [17]. Since all the methods are stochastic, it is necessary to employ statistical metrics to compare the efficiency of the algorithms. Hence, all algorithms are executed 35 times per image, according to the related literature the number the thresholds for test are $th = 2, 3, 4, 5$ [13, 19]. In each experiment the stop criteria is set to 50 iterations. In order to verify the stability at the end of each test the standard deviation (STD) is obtained Eq. (2.21). If the STD value increases the algorithms becomes more instable [29].

$$STD = \sqrt{\sum_{i=1}^{Iter_{\max}} \frac{(\sigma_i - \mu)}{Ru}} \tag{2.21}$$

On the other hand the peak-to-signal ratio (*PSNR*) is used to compare the similarity of an image (image segmented) against a reference image (original image) based on the mean square error (*MSE*) of each pixel [5, 13, 30, 32]. Both *PSNR* and *MSE* are defined as:

$$\begin{aligned} PSNR &= 20\log_{10}\left(\frac{255}{RMSE}\right), \quad (\text{dB}) \\ RMSE &= \sqrt{\frac{\sum_{i=1}^{ro}\sum_{j=1}^{co}\left(I_{Gr}(i,j) - I_{th}(i,j)\right)}{ro \times co}} \end{aligned} \tag{2.22}$$

where I_{Gr} is the original image, I_{th} is the segmented image and *ro*, *co* are the total number of rows and columns of the image, respectively. The Structure Similarity Index (*SSIM*) is used to compare the structures of the original umbralized image [33] it is defined in Eq. (2.23). A higher *SSIM* value means that the performance of the segmentation is better.

$$\begin{aligned} SSIM(I_{Gr}, I_{th}) &= \frac{\left(2\mu_{I_{Gr}}\mu_{I_{th}} + C1\right)\left(2\sigma_{I_{Gr}I_{th}} + C2\right)}{\left(\mu_{I_{Gr}}^2 + \mu_{I_{th}}^2 + C1\right)\left(\sigma_{I_{Gr}}^2 + \sigma_{I_{th}}^2 + C2\right)} \\ \sigma_{I_oI_{Gr}} &= \frac{1}{N-1}\sum_{i=1}^{N}\left(I_{Gr_i} + \mu_{I_{Gr}}\right)\left(I_{th_i} + \mu_{I_{th}}\right) \end{aligned} \tag{2.23}$$

where $\mu_{I_{Gr}}$ and $\mu_{I_{th}}$ are the mean value of the original and the umbralized image respectively, for each image the values of $\sigma_{I_{Gr}}$ and $\sigma_{I_{th}}$ corresponds to the standard deviation. $C1$ and $C2$ are constants used to avoid the instability when

$\mu_{I_{Gr}}^2 + \mu_{I_{th}}^2 \approx 0$, experimentally in [12] both values are $C1 = C2 = 0.065$. Another method used to measure the quality of the segmented image is the Feature Similarity Index (*FSIM*) [34]. *FSIM* calculates the similarity between two images, in this cases the original gray scale image and the segmented image Eq. (2.24). As *PSNR* and *SSIM* the higher value is interpreted as better performance of the thresholding method.

$$FSIM = \frac{\sum_{w\in\Omega} S_L(w)PC_m(w)}{\sum_{w\in\Omega} PC_m(w)} \tag{2.24}$$

where Ω represents the entire domain of the image:

$$\begin{aligned} S_L(w) &= S_{PC}(w)S_G(W) \\ S_{PC}(w) &= \frac{2PC_1(w)PC_2(w) + T_1}{PC_1^2(w) + PC_2^2(w) + T_1} \\ S_G(W) &= \frac{2G_1(w)G_2(w) + T_2}{G_1^2(w) + G_2^2(w) + T_2} \end{aligned} \tag{2.25}$$

G is the gradient magnitude (GM) of an image and is defined as:

$$G = \sqrt{G_x^2 + G_y^2} \tag{2.26}$$

PC is the phase congruence:

$$PC(w) = \frac{E(w)}{\left(\varepsilon + \sum_n A_n(w)\right)} \tag{2.27}$$

$A_n(w)$ is the local amplitude on scale n and $E(w)$ is the magnitude of the response vector in w on n. ε is an small positive number and $PC_m(w) = \max(PC_1(w), PC_2(w))$. On the other hand, Table 2.1 presents the parameters for the EMO algorithm. They have been obtained using the criterion proposed in [31] and kept for all test images.

2.5.1 Tsallis Entropy Results

In this section, the results of the TSEMO algorithm are analyzed, considering as objective function Eq. (2.15) the Tsallis entropy [35]. The approach is applied over

Table 2.1 EMO parameters

$Iter_{\max}$	$Iter_{local}$	δ	N
200	10	0.25	50

the complete set of benchmark images whereas the results are registered in Table 2.2. Such results present the best threshold values obtained after testing the TSEMO algorithm, considering four different threshold points $th = 2, 3, 4, 5$. In Table 2.2, it is also shown the *PSNR*, *STD*, *SSIM* and *FSIM* values.

There have been selected five images of the set to show (graphically) the segmentation results. Figure 2.1 presents the images selected from the benchmark set and their respective histograms which possess irregular distributions (particularly Fig. 2.1j). Under such circumstances, classical methods face great difficulties to find the best threshold values.

Table 2.3 shows the images obtained after processing 5 original images selected from the entire benchmark set, applying the proposed algorithm. The results present the segmented images considering four different threshold levels $th = 2, 3, 4, 5$. In Table 2.3, it is also shown the evolution of the objective function during one execution. From the results, it is possible to appreciate that the TSEMO converges (stabilizes) around the first 100 iterations. The segmented images provide evidence that the outcome is better with $th = 4$ and $th = 5$; however, if the segmentation task does not requires to be extremely accurate then it is possible to select $th = 3$.

2.5.2 Comparisons

In order to demonstrate that the TSEMO is an interesting alternative for MT, the proposed algorithm is compared with two state-of-the-art implementations. The methods used for comparison are: the Cuckoo Search Algorithm (CSA) [13] and the Particle Swarm Optimization (PSO) [19], both methods uses the Tsallis entropy.

All the algorithms run 35 times over each selected image. The images used for this test are the same of the selected in Sect. 2.5.1 (Camera man, Lena, Baboon, Hunter and Butterfly). For each image is computed the *PSNR*, *STD*, *SSIM*, *FSIM* values and the mean of the objective function.

The comparison results between the three methods are divided in two tables, Table 2.4 shows the *STD* and mean values of the fitness function. Table 2.5 presents the values of the quality metrics obtained after apply the thresholds over the test images.

The fitness values of four methods are statistically compared using a non-parametric significance proof known as the Wilcoxon's rank test [36] that is conducted with 35 independent samples. Such proof allows assessing result differences among two related methods. The analysis is performed considering a 5% significance level over the best fitness (Tsallis entropy) value data corresponding to the five threshold points. Table 2.6 reports the p-values produced by Wilcoxon's test for a pair-wise comparison of the fitness function between two groups formed as TSEMO versus CSA, TSEMO versus PSO. As a null hypothesis, it is assumed

Table 2.2 Result after applying the MTEMO to the set of benchmark images

Image	k	Thresholds x_i^B	*PSNR*	*STD*	*SSIM*	*FSIM*
Camera man	2	71, 130	23.1227	31.00 E–04	0.9174	0.8901
	3	71, 130, 193	18.0122	72.01 E–04	0.8875	0.8456
	4	44, 84, 120, 156	24.9589	86.01 E–03	0.9363	0.9149
	5	44, 84, 120, 156, 196	23.0283	7.90 E–01	0.9289	0.8960
Lena	2	79, 127	23.9756	7.21 E–05	0.9083	0.8961
	3	79, 127, 177	21.0043	14.37 E–04	0.8660	0.8197
	4	62, 94, 127, 161	24.0020	18.69 E–03	0.9057	0.8851
	5	62, 94, 127, 161, 194	23.3736	39.82 E–02	0.8956	0.8684
Baboon	2	15, 105	23.5906	18.51 E–06	0.9480	0.9437
	3	51, 105, 158	19.9394	28.78 E–02	0.9011	0.9059
	4	33, 70, 107, 143	23.5022	22.65 E–02	0.9530	0.9594
	5	33, 70, 107, 143, 179	21.9540	37.13 E–01	0.9401	0.9417
Hunter	2	60, 119	22.8774	17.89 E–04	0.9192	0.8916
	3	60, 119, 179	20.2426	54.12 E–04	0.9031	0.8652
	4	46, 90, 134, 178	22.4723	1.94 E–02	0.9347	0.9159
	5	46, 90, 134, 178, 219	22.4025	1.23 E–01	0.9349	0.9173
Airplane	2	69, 125	25.4874	17.31 E–04	0.9685	0.9239
	3	69, 125, 180	22.9974	17.89 E–04	0.9433	0.8909
	4	55, 88, 122, 155	28.5400	19.21 E–03	0.9848	0.9677
	5	55, 88, 122, 155, 188	26.4997	35.08 E–03	0.9663	0.9417
Peppers	2	70, 145	19.6654	54.83 E–02	0.8697	0.8378
	3	70, 145, 223	17.2736	1.31 E–01	0.8437	0.7534
	4	46, 88, 132, 175	21.8275	3.02 E–04	0.8976	0.8552
	5	46, 88, 132, 175, 223	21.1207	6.34 E–03	0.8976	0.8304
Living room	2	55, 111	22.6665	47.11 E–03	0.9116	0.8966
	3	55, 111, 179	18.0379	15.27 E–04	0.8482	0.8132
	4	42, 85, 124, 162	21.7235	93.35 E–03	0.9170	0.9090
	5	42, 85, 124, 162, 201	21.3118	94.32 E–03	0.9183	0.9029
Blonde	2	62, 110	25.8389	31.91 E–04	0.9645	0.9503
	3	62, 110, 155	21.5001	37.05 E–04	0.9012	0.8759
	4	36, 65, 100, 134	25.9787	17.45 E–03	0.9606	0.9491
	5	36, 65, 100, 134, 168	23.1835	48.20 E–03	0.9328	0.9077
Bridge	2	65, 131	20.1408	22.71 E–04	0.8619	0.8749
	3	65, 131, 191	18.7016	40.49 E–04	0.8410	0.8479
	4	45, 88, 131, 171	21.4247	38.48 E–03	0.9168	0.9279
	5	45, 88, 131, 171, 211	21.0157	66.16 E–03	0.9153	0.9217
Butterfly	2	83, 120	26.7319	96.11 E–03	0.9493	0.9195
	3	83, 120, 156	24.4582	39.04 E–03	0.9386	0.8934
	4	70, 94, 119, 144	27.0221	14.59 E–02	0.9653	0.9417
	5	70, 94, 119, 144, 172	25.7809	98.61 E–02	0.9610	0.9283
Lake	2	71, 121	27.8565	10.69 E–04	0.9729	0.9638
	3	71, 121, 173	23.7695	12.87 E–04	0.9399	0.9288
	4	41, 80, 119, 159	24.7454	11.97 E–03	0.9587	0.9422
	5	41, 80, 119, 159, 197	22.4347	11.80 E–03	0.9439	0.9213

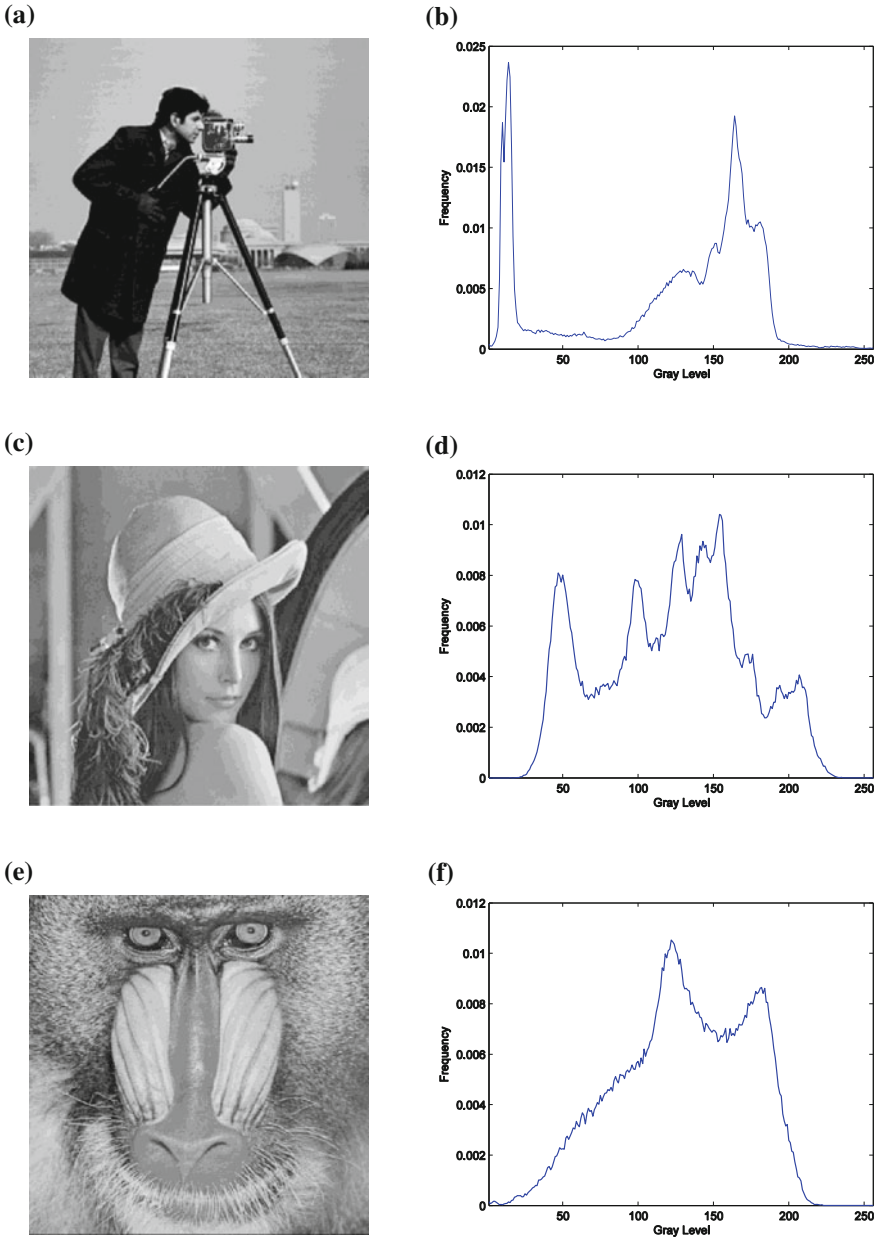


Fig. 2.1 **a** Camera man, **c** Lena, **e** Baboon, **g** Hunter and **i** Butterfly, the selected benchmak images. **b**, **d**, **f**, **h**, **j** histograms of the images

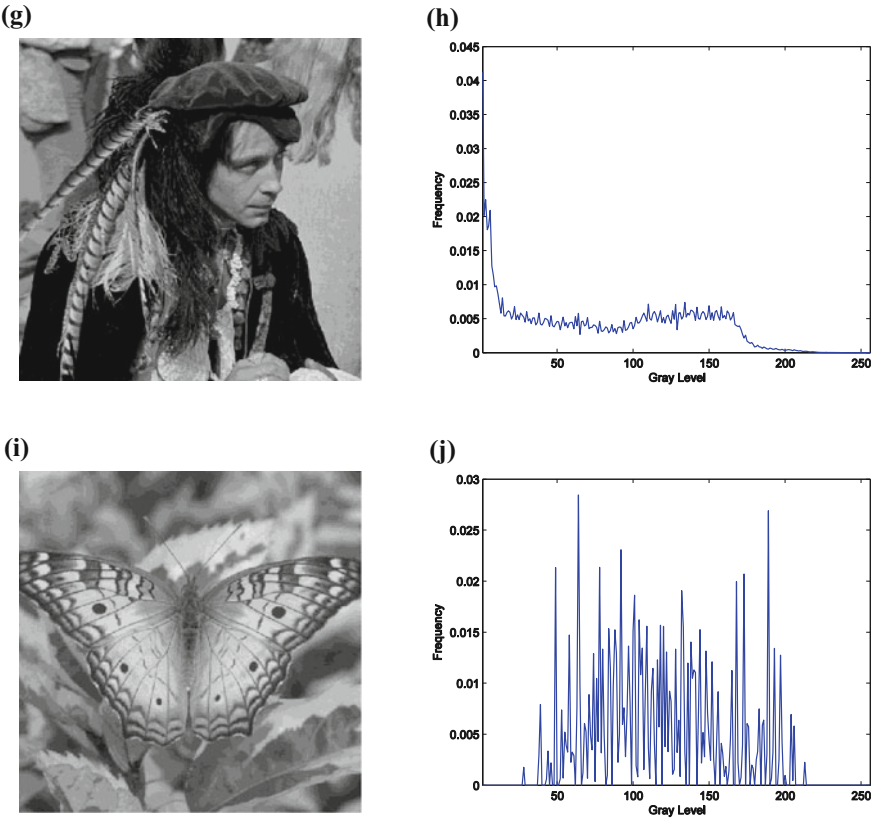
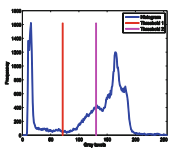
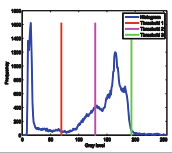
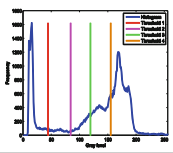
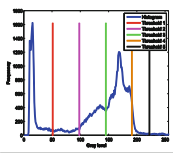
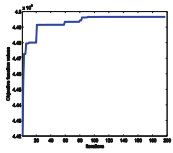
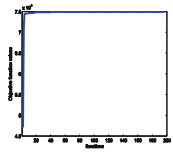
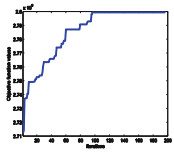
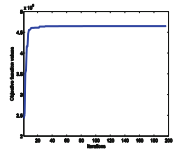
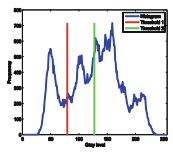
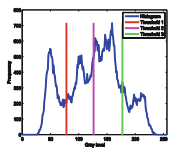
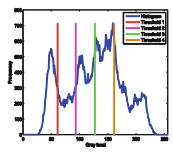
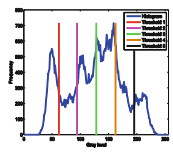
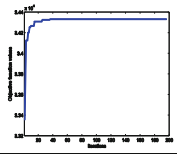
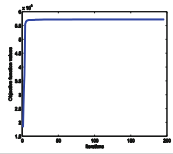
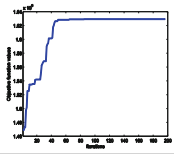
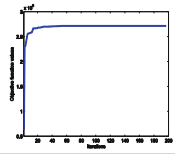


Fig. 2.1 (continued)

that there is no difference between the values of the two algorithms tested. The alternative hypothesis considers an existent difference between the values of both approaches. All p-values reported in Table 2.6 are less than 0.05 (5% significance level) which is a strong evidence against the null hypothesis, indicating that the TSEMO fitness values for the performance are statistically better and it has not occurred by chance.

On the other hand, to compare the fitness of the three methods Table 2.7 shows the fitness values obtained for the reduced set of image (5 images). Each algorithm runs 1000 times and the best value of each run is stored, at the end of the evolution process the best stored values are plotted. From Table 2.6 it is possible to analyze that TSEMO and CSA reach the maximum entropy values in less iterations than the PSO method.

Table 2.3 Results after appliying the MT-EMO using Tsallis entropy over the selected benchamark images

	$th = 2$	$th = 3$	$th = 4$	$th = 5$
Camera man				
				
				
Lena				
				
				

(continued)

Table 2.3 (continued)

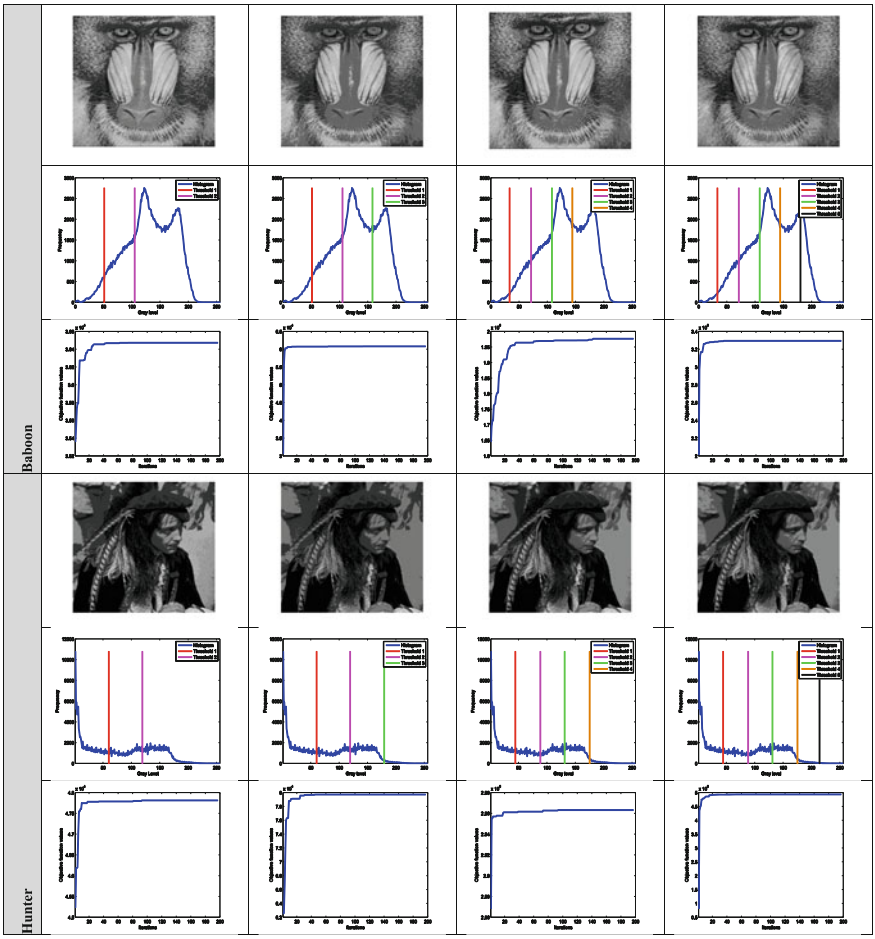


Table 2.3 (continued)

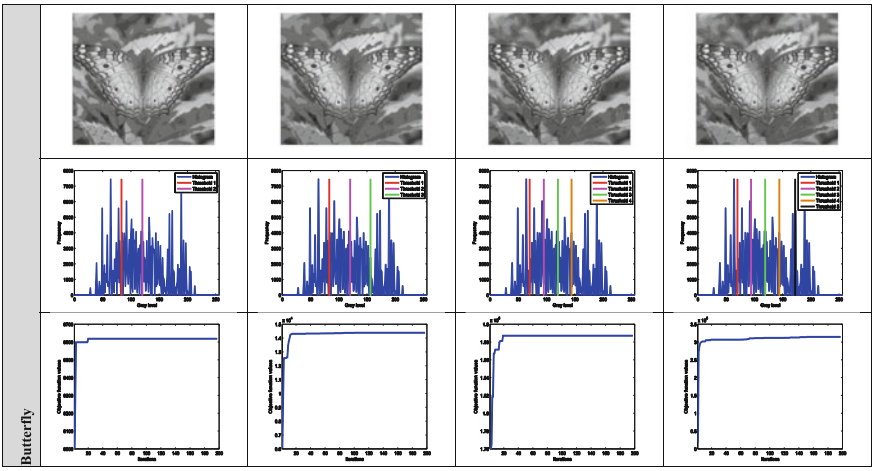


Table 2.4 Comparison of the *STD* and mean values of the TSEMO, CSA and PSO applied over the selected test images using Tsallis method

Image	*k*	TSEMO		CSA		PSO	
		STD	Mean	*STD*	Mean	*STD*	Mean
Camera man	2	31.00 E–04	4.49 E+04	89.56 E–04	4.02 E+04	83.00 E02	4.19 E+04
	3	72.01 E–04	7.49 E+04	98.32 E–04	6.99 E+04	89.00 E+00	7.27 E+04
	4	86.01 E–03	2.79 E+06	18.68 E–03	2.18 E+06	12.35 E+02	2.37 E+06
	5	7.90 E–01	4.65 E+06	69.98 E–01	4.56 E+06	5.38 E+03	4.28 E+06
Lena	2	7.21 E–05	3.43 E+04	2.61 E+00	3.33 E+04	15.27 E+00	3.30 E+04
	3	14.37 E–04	5.72 E+04	3.39 E+00	5.67 E+04	3.31 E+00	5.62 E+04
	4	18.69 E–03	1.62 E+06	5.52 E+00	1.45 E+06	7.35 E+00	1.45 E+06
	5	39.82 E–02	2.71 E+06	8.50 E+01	2.55 E+06	2.92 E+00	2.59 E+06
Baboon	2	18.51 E–06	3.64 E+04	15.11 E–02	3.47 E+04	2.64 E+00	3.40 E+04
	3	28.78 E–02	6.08 E+04	40.80 E–02	6.05 E+04	1.44 E+00	6.03 E+04
	4	22.65 E–02	1.97 E+06	62.02 E–02	1.90 E+06	8.11 E+00	1.86 E+06
	5	37.13 E–01	3.29 E+06	52.74 E–02	3.20 E+06	2.68 E+00	3.20 E+06
Hunter	2	17.89 E–04	4.78 E+04	7.38 E–04	4.70 E+04	4.38 E+00	4.72 E+04
	3	54.12 E–04	7.97 E+04	2.95 E–04	7.89 E+04	9.47 E+00	7.85 E+04
	4	1.94 E–02	2.96 E+06	1.62 E–01	2.93 E+06	1.04 E+01	2.92 E+04
	5	1.23 E–01	4.94 E+06	2.46 E–01	4.89 E+06	3.23 E+02	4.75 E+04
Butterfly	2	96.11 E–03	8.61 E+03	12.78 E–02	8.56 E+03	6.36 E–01	8.55 E+03
	3	39.04 E–03	1.43 E+04	19.00 E–02	1.38 E+04	11.56 E–01	1.35 E+04
	4	14.59 E–02	1.88 E+05	11.04 E–01	1.80 E+05	1.04 E+00	1.81 E+05
	5	98.61 E–02	3.14 E+05	1.58 E+00	3.07 E+05	3.58 E+00	2.96 E+05

Table 2.5 Comparison of the *PSNR*, *SSIM* and *FSIM* values of the TSEMO, CSA and PSO applied over the selected test images using Tsallis method

Image	*k*	TSEMO			CSA			PSO		
		PSNR	*SSIM*	*FSIM*	*PSNR*	*SSIM*	*FSIM*	*PSNR*	*SSIM*	*FSIM*
Camera man	2	23.1227	0.9174	0.8901	23.1194	0.9173	0.8901	22.9737	0.9160	0.8871
	3	18.0998	0.8875	0.8509	18.7480	0.8918	0.8456	18.0122	0.8874	0.8441
	4	25.0021	0.9369	0.9151	24.5479	0.9349	0.9097	23.3230	0.9280	0.8976
	5	22.9136	0.9286	0.8950	22.5284	0.9243	0.8891	21.9598	0.9222	0.8839
Lena	2	23.9982	0.9088	0.8966	23.9756	0.9083	0.8961	23.9594	0.9085	0.8953
	3	21.2592	0.8699	0.8255	20.9669	0.8655	0.8192	20.9989	0.8659	0.8196
	4	23.9783	0.9056	0.8849	23.9493	0.9056	0.8846	23.8175	0.9032	0.8815
	5	23.4275	0.8954	0.8691	23.3099	0.8960	0.8689	23.3777	0.8949	0.8674
Baboon	2	23.7510	0.9496	0.9452	23.5906	0.9480	0.9410	23.5048	0.9475	0.9323
	3	19.9386	0.9007	0.9057	19.9031	0.8810	0.8759	19.8021	0.8729	0.8729
	4	23.5165	0.9532	0.9593	23.5106	0.9270	0.9295	23.5163	0.9125	0.9159
	5	22.0538	0.9410	0.9408	21.9071	0.9399	0.9112	21.7165	0.9350	0.9377
Hunter	2	22.8783	0.9192	0.8916	22.8074	0.9089	0.8826	22.7910	0.9093	0.8818
	3	20.2581	0.9034	0.8654	20.0026	0.8931	0.8552	20.0858	0.8921	0.8521
	4	22.4221	0.9341	0.9159	21.3972	0.9237	0.9055	21.5061	0.9244	0.9024
	5	22.5014	0.9355	0.9199	21.3171	0.9236	0.9063	21.3754	0.9254	0.9005
Butterfly	2	26.8352	0.9504	0.9212	25.7319	0.9493	0.9195	25.1635	0.9431	0.9150
	3	24.4144	0.9383	0.8926	23.4545	0.9300	0.8834	23.5251	0.9315	0.8846
	4	27.1226	0.9653	0.9420	26.0314	0.9653	0.9317	26.0810	0.9653	0.9321
	5	25.8838	0.9609	0.9285	24.0086	0.9516	0.9201	24.4870	0.9533	0.9142

Table 2.6 Wilcoxon *p*-values of the compared algorithm TSEMO versus CSA and TSEMO verssus PSO

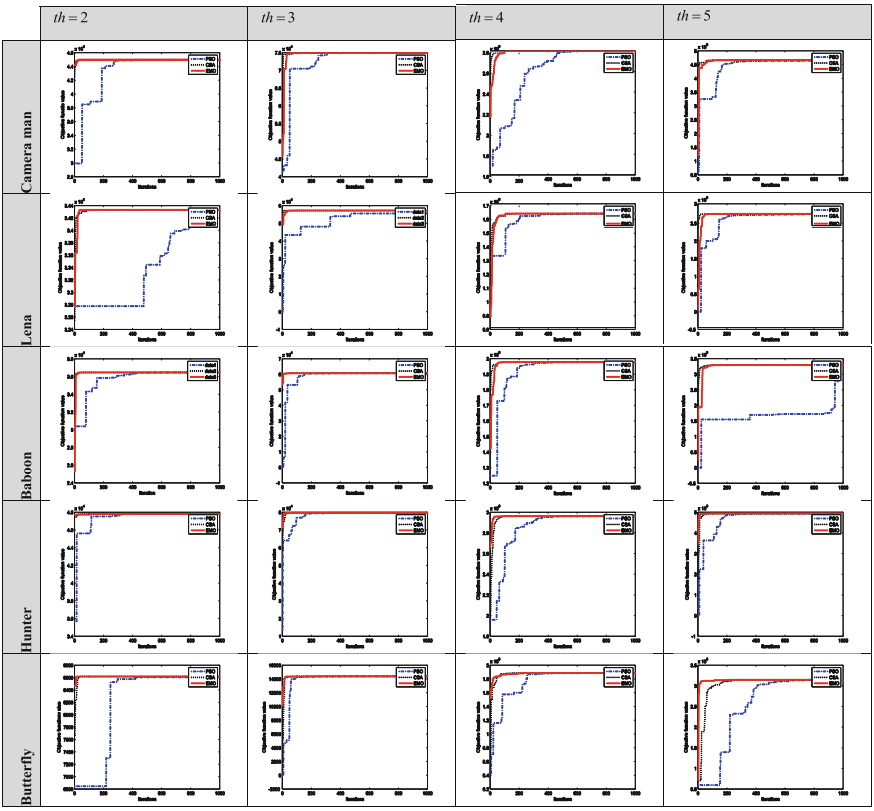
Image	*k*	*p*-values	
		TSEMO versus CS	TSEMO versus PSO
Camera man	2	6.2137 E–07	8.3280 E–06
	3	1.0162 E–07	2.0000 E–03
	4	8.8834 E–08	13.710 E–03
	5	16.600 E–03	50.600 E–03
Lena	2	3.7419 E–08	1.6604 E–04
	3	1.4606 E–06	1.3600 E–02
	4	1.2832 E–07	2.9000 E–03
	5	3.9866 E–05	8.9000 E–03
Baboon	2	1.5047 E–06	2.5500 E–02
	3	6.2792 E–05	5.1000 E–03
	4	2.1444 E–12	3.3134 E–05
	5	2.1693 E–11	1.8000 E–03

(continued)

Table 2.6 (continued)

Image	k	p-values	
		TSEMO versus CS	TSEMO versus PSO
Hunter	2	2.2100 E–02	2.2740 E–02
	3	3.6961 E–04	1.1500 E–02
	4	6.8180 E–02	9.9410 E–09
	5	5.8200 E–02	2.4939 E–04
Airplane	2	3.0000 E–03	6.6300 E–03
	3	7.6000 E–03	3.5940 E–02
	4	4.8092 E–12	1.1446 E–06
	5	1.0023 E–09	2.7440 E–02
Peppers	2	2.7419 E–04	1.3194 E–04
	3	2.6975 E–08	3.5380 E–02
	4	1.5260 E–08	6.0360 E–02
	5	7.2818 E–08	7.6730 E–02
Living room	2	1.4000 E–03	2.6340 E–02
	3	6.8066 E–08	2.8000 E–03
	4	8.7456 E–07	5.8730 E–03
	5	1.7000 E–03	5.1580 E–03
Blonde	2	3.0000 E–03	4.1320 E–02
	3	5.9000 E–03	8.9300 E–02
	4	1.3800 E–02	2.7700 E–02
	5	2.3440 E–02	5.6000 E–03
Bridge	2	1.5000 E–03	1.5700 E–02
	3	1.4300 E–02	1.5350 E–02
	4	1.7871 E–06	7.0400 E–03
	5	8.7000 E–03	1.2400 E–02
Butterfly	2	1.5000 E–03	1.1150 E–02
	3	3.1800 E–02	1.3760 E–02
	4	4.8445 E–07	8.1800 E–03
	5	1.6000 E–02	1.0630 E–02
Lake	2	7.6118 E–06	2.9500 E–02
	3	1.2514 E–06	6.5644 E–06
	4	2.2366 E–10	6.6000 E–03
	5	5.3980 E–06	9.4790 E–03

Table 2.7 Fitness comparsion of PSO (blue line), CSA (Black line) and EMO (red line) applied for multilevel tresholding using TE



2.6 Conclusions

In this chapter, a new algorithm for multilevel segmentation based on the Electromagnetism-Like algorithm (EMO) has been presented. The proposed approach considers the segmentation process as an optimization problem, where EMO is employed to find the optimal threshold points that maximize the Tsallis entropy (TE). As a result, the proposed algorithm can substantially reduce the number of function evaluations preserving the good search capabilities of an evolutionary method. In our approach, the algorithm uses as particles the encoding of a set of candidate threshold points. The TE objective function evaluates the segmentation quality of the candidate threshold points. Guided by the values of this objective function, the set of encoded candidate solutions are modified by using the EMO operators so that they can improve their segmentation quality as the optimization process evolves. In order to evaluate the quality of the segmented images,

the use of the *PSNR*, *STD*, *SSIM* and *FSIM* is proposed. Such metrics considers the coincidences between the original and the segmented image.

The study compares the proposed approach with other two similar approaches the Cuckoo Search algorithm (CSA) and Particle Swarm Optimization algorithm (PSO). The efficiency of the algorithms is evaluated in terms of *PSNR*, *STD*, *SSIM*, *FSIM* and fitness values. Such comparisons provide evidence of the accuracy, convergence and robustness of the proposed approach. The fitness of TSEMO is compared with the CSA and PSO where is possible to see that both EMO and CSA need a reduced number of iterations to converge. However the speed of convergence of EMO is higher than de CSA in the same way PSO is the slower and it has lack of accuracy. Although the results offer evidence to demonstrate that the TSEMO method can yield good results on complicated images, the aim of our chapter is not to devise a multilevel thresholding algorithm that could beat all currently available methods, but to show that electro-magnetism systems can be effectively considered as an attractive alternative for this purpose.

References

1. Cuevas, E., Zaldivar, D., Pérez-Cisneros, M., Seeking multi-thresholds for image segmentation with Learning Automata, Machine Vision and Applications, 22 (5), (2011), 805–818.
2. Y. Kong, Y. Deng, Q. Dai, and S. Member, "Discriminative Clustering and Feature Selection for Brain MRI Segmentation," IEEE Signal Process. Lett., vol. 22, no. 5, pp. 573–577, 2015.
3. X. Cao, Q. Li, X. Du, M. Zhang, and X. Zheng, "Exploring effect of segmentation scale on orient-based crop identification using HJ CCD data in Northeast China," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 17, p. 012047, 2014.
4. A. K. Bhandari, V. K. Singh, A. Kumar, and G. K. Singh, "Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy," *Expert Syst. Appl.*, vol. 41, no. 7, pp. 3538–3560, 2014.
5. S. Sarkar and S. Das, "Multilevel Image Thresholding Based on 2D Histogram and Maximum Tsallis Entropy—A Differential Evolution Approach," *Lect. Notes Comput. Sci.*, vol. 22, no. 12, pp. 4788–4797, 2013.
6. B. Akay, "A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding," *Appl. Soft Comput.*, vol. 13, no. 6, pp. 3066–3091, 2012.
7. H. Xia, S. Song, and L. He, "A modified Gaussian mixture background model via spatiotemporal distribution with shadow detection," *Signal, Image Video Process.*, 2015.
8. G. Moser, S. B. Serpico, and S. Member, "Generalized Minimum-Error Thresholding for Unsupervised Change Detection From SAR Amplitude Imagery.pdf," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2972–2982, 2006.
9. Sezgin M, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electron. Imaging*, vol. 13, no. January, pp. 146–168, 2004.
10. N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans. Syst. Man. Cybern.*, vol. 9, no. 1, pp. 62–66, 1979.
11. A. K. C. J. N. Kapur, P. K. Sahoo, A. K. C. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram." Computer Vision Graphics Image Processing, pp. 273–285, 1985.
12. P. D. Sathya and R. Kayalvizhi, "Optimal multilevel thresholding using bacterial foraging algorithm," *Expert Syst. Appl.*, vol. 38, no. 12, pp. 15549–15564, 2011.

13. S. Agrawal, R. Panda, S. Bhuyan, and B. K. Panigrahi, "Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm," *Swarm Evol. Comput.*, vol. 11, pp. 16–30, 2013.
14. C. Tsallis, "Possible generalization of Boltzmann-Gibbs statistics," *J. Stat. Phys.*, vol. 52, pp. 479–487, 1988.
15. E. K. Tang, P. N. Suganthan, and X. Yao, "An analysis of diversity measures," *Mach. Learn.*, vol. 65, no. April, pp. 247–271, 2006.
16. Y. Zhang and L. Wu, "Optimal multi-level thresholding based on maximum Tsallis entropy via an artificial bee colony approach," *Entropy*, vol. 13, pp. 841–859, 2011.
17. C. Tsallis, "Computational applications of nonextensive statistical mechanics," *J. Comput. Appl. Math.*, vol. 227, no. 1, pp. 51–58, 2009.
18. Cuevas, E., Ortega-Sánchez, N., Zaldivar, D., Pérez-Cisneros, M., Circle detection by Harmony Search Optimization, Journal of Intelligent and Robotic Systems: Theory and Applications, 66(3), (2012), 359–376.
19. N. Sri, M. Raja, G. Kavitha, and S. Ramakrishnan, "Analysis of Vasculature in Human Retinal Images Using Particle Swarm Optimization Based Tsallis Multi-level Thresholding and Similarity Measures," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7677, no. 1, pp. 380–387, 2012.
20. Oliva, D., Cuevas, E., Pajares, G., Zaldivar, D., Perez-Cisneros, M., Multilevel thresholding segmentation based on harmony search optimization, Journal of Applied Mathematics, 2013, 575414.
21. Ş. I. Birbil and S. C. Fang, "An electromagnetism-like mechanism for global optimization," *J. Glob. Optim.*, vol. 25, no. 1, pp. 263–282, 2003.
22. A. M. A. C. Rocha and E. M. G. P. Fernandes, "Modified movement force vector in an electromagnetism-like mechanism for global optimization," *Optim. Methods Softw.*, vol. 24, no. 2, pp. 253–270, 2009.
23. H. L. Hung and Y. F. Huang, "Peak to average power ratio reduction of multicarrier transmission systems using electromagnetism-like method," *Int. J. Innov. Comput. Inf. Control*, vol. 7, no. 5, pp. 2037–2050, 2011.
24. A. Yurtkuran and E. Emel, "A new Hybrid Electromagnetism-like Algorithm for capacitated vehicle routing problems," *Expert Syst. Appl.*, vol. 37, no. 4, pp. 3427–3433, 2010.
25. J. Y. Jhang and K. C. Lee, "Array pattern optimization using electromagnetism-like algorithm," *AEU - Int. J. Electron. Commun.*, vol. 63, pp. 491–496, 2009.
26. C. H. Lee and F. K. Chang, "Fractional-order PID controller optimization via improved electromagnetism-like algorithm," *Expert Syst. Appl.*, vol. 37, no. 12, pp. 8871–8878, 2010.
27. L. N. De Castro and F. J. Von Zuben, "Learning and optimization using the clonal selection principle," *IEEE Trans. Evol. Comput.*, vol. 6, no. 3, pp. 239–251, 2002.
28. A. M. A. C. Rocha and E. M. G. P. Fernandes, "Hybridizing the Electromagnetism-like algorithm with Descent Search for Solving Engineering Design Problems," *Int. J. Comput. Math.*, vol. 86, no. 10–11, pp. 1932–1946, 2009.
29. P. Ghamisi, M. S. Couceiro, J. A. Benediktsson, and N. M. F. Ferreira, "An efficient method for segmentation of images based on fractional calculus and natural selection," *Expert Syst. Appl.*, vol. 39, no. 16, pp. 12407–12417, 2012.
30. P. Wu, W.-H. Yang, and N.-C. Wei, "An Electromagnetism Algorithm of Neural Network Analysis—an Application To Textile Retail Operation," *J. Chinese Inst. Ind. Eng.*, vol. 21, no. 1, pp. 59–67, 2004.
31. K. De Jong, "Learning with genetic algorithms: An overview," *Mach. Learn.*, vol. 3, pp. 121–138, 1988.
32. B. Naderi, R. Tavakkoli-Moghaddam, and M. Khalili, "Electromagnetism-like mechanism and simulated annealing algorithms for flowshop scheduling problems minimizing the total weighted tardiness and makespan," *Knowledge-Based Syst.*, vol. 23, no. 2, pp. 77–85, 2010.
33. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

34. D. Z. Lin Zhang, Lei Zhang, XuanqinMou, "FSIM : A Feature Similarity Index for Image," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, 2011.
35. C. Tsallis, "Entropic nonextensivity: A possible measure of complexity," *Chaos, Solitons and Fractals*, vol. 13, pp. 371–391, 2002.
36. S. García, D. Molina, M. Lozano, and F. Herrera, "A Study on the Use of Non-Parametric Tests for Analyzing the Evolutionary Algorithms' Behaviour: A Case Study on the CEC 2005 Special Session on Real Parameter Optimization," *J. Heuristics*, vol. 15, pp. 617–644, 2009.

http://www.springer.com/978-3-319-51108-5

Evolutionary Computation Techniques: A Comparative Perspective

Cuevas, E.; Osuna, V.; Oliva, D.A.

2017, XV, 222 p. 74 illus., 33 illus. in color., Hardcover
ISBN: 978-3-319-51108-5