[Egyptian Informatics Journal 20 (2019) 11–25](https://doi.org/10.1016/j.eij.2018.03.006)

Contents lists available at [ScienceDirect](http://www.sciencedirect.com/science/journal/11108665)

Egyptian Informatics Journal

journal homepage: [www.sciencedirect.com](http://www.sciencedirect.com/)

Full length article

[](http://crossmark.crossref.org/dialog/?doi=10.1016/j.eij.2018.03.006&domain=pdf)Teaching-learning-based optimization algorithm to minimize cross entropy for Selecting multilevel threshold values

Harmandeep Singh Gill [a](#_bookmark0),[1](#_bookmark3),[⇑](#_bookmark7), Baljit Singh Khehra [b](#_bookmark1),[2](#_bookmark4), Arjan Singh [c](#_bookmark2),[3](#_bookmark5), Lovepreet Kaur [c](#_bookmark2),[4](#_bookmark6)

a *Department of Computer Applications, I. K. Gujral Punjab Technical University, Jalandhar-144603, Punjab, India*

b *Department of Computer Science & Engineering, Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib-140407, Punjab, India*

c *Department of Mathematics, Punjabi University, Patiala-147002, Punjab, India*

a r t i c l e i n f o

*Article history:*

Received 19 August 2017

Revised 22 January 2018

Accepted 29 March 2018

Available online 25 April 2018

*Keywords:*

Cross entropy

Teacher-Learning-based Optimization (TLBO)

Thresholding PSNR

Uniformity

a b s t r a c t

Image thresholding is one of the most important approaches for image segmentation. Among multilevel thresholding techniques, cross entropy has been widely used by researchers to find multilevel threshold values. In multilevel cross entropy thresholding techniques, main target is to find an optimal combination of threshold values at different levels for minimizing the cross entropy. In this paper, Teaching-Learning- based Optimization (TLBO) algorithm is used to find an optimal combination of threshold values at dif- ferent levels for minimizing the cross entropy. TLBO algorithm is inspired by passing on knowledge within a classroom environment where students first gain knowledge from a teacher and then through mutual interaction. This new proposed approach is called the TLBO-based minimum cross entropy thresholding (TLBO-based MCET) algorithm. The performance of the proposed algorithm is tested on a number of standard test images. For comparative analysis, the results of TLBO-based MCET algorithm are compared with the results of Firefly-based minimum cross entropy thresholding (FF-based MCET), Honey Bee Mating Optimization-based minimum cross entropy thresholding (HBMO-based MCET) and Quantum Particle Swarm Optimization-based minimum cross entropy thresholding (Quantam PSO- based MCET). Comparative analysis is done based on two most popular objective performance measures, Peak Signal to Noise Ratio (PSNR) and Uniformity. From experimental results, it is observed that the pro- posed method is an efficient and feasible method to search an optimal combination of threshold values at 2nd, 3rd, 4th and 5th levels.

© 2019 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information, Cairo

University. This is an open access article under the CC BY-NC-ND license ([http://creativecommons.org/](http://creativecommons.org/licenses/by-nc-nd/4.0/)

[licenses/by-nc-nd/4.0/](http://creativecommons.org/licenses/by-nc-nd/4.0/)).

1. Introduction

Image segmentation is an active field in medical imaging, machine vision and satellite imagery. Main goal of image segmen-

\* Corresponding author.

*E-mail addresses:* [gill.gurseerat@gmail.com](mailto:gill.gurseerat@gmail.com) (H. Singh Gill), [baljitkhehra@](mailto:baljitkhehra@rediffmail.com) [rediffmail.com](mailto:baljitkhehra@rediffmail.com) (B. Singh Khehra), [arjanpu@gmail.com](mailto:arjanpu@gmail.com) (A. Singh), [lovepreetkhehra13](mailto:lovepreetkhehra13@gmail.com) [@gmail.com](mailto:lovepreetkhehra13@gmail.com) (L. Kaur).

1 +91-8146088188

2 +91-9463446505

3 +91-9814668883

4 +91-9464601473

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

**Production and hosting by Elsevier**

tation is to partitioning an image into regions which are meaning- ful for a particular task. Segmentation is generally the essential component of pattern recognition systems in which objects of interest are found and isolated from the rest of the scene. After image segmentation, some features are extracted from objects and at the end, objects are classified in particular groups or classes based on extracted features. For medical applications, segmenta- tion is used for the detection of organs such as the brain, heart, lungs or liver in CT or MR images [[1]](#_bookmark32). It is also used to distinguish pathological tissue such as a tumor from normal tissue. Depending on the particular application, different techniques for image seg- mentation have been used, such as image thresholding, edge detec- tion, region growing, stochastic models, ANN and clustering techniques [[2]](#_bookmark33). Thresholding is one of the most frequently used methods in image segmentation because it is computationally sim- ple and never fails to define disjoint regions with closed, connected boundaries [[3]](#_bookmark34). Its basic idea is to divide the image into target and background regions by the threshold value. Intensity value of each

<https://doi.org/10.1016/j.eij.2018.03.006>

1110-8665/© 2019 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information, Cairo University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

pixel of the image is compared to the threshold value. If its value is greater than the threshold value, then the pixel is considered as target region pixel and set to white; otherwise the pixel is consid- ered as background region pixel and set to black. The success of thresholding depends upon the selection of an optimal threshold value. Over the years, various thresholding techniques have been proposed by researchers to select a suitable threshold value [[4]](#_bookmark21). In the past years, many thresholding techniques using entropy measures have been proposed by researchers to select an appropri- ate threshold value for segmenting images [[5–9]](#_bookmark21). In these tech- niques, first, two probability distributions, one for the target and the other for the background are obtained from gray level values of input image. After this, the procedure is adopted for maximizing the total entropy of two probability distributions to obtain the threshold. Different definitions of entropies differentiated these techniques from each other. Liu et al. [[10]](#_bookmark21) proposed a fuzzy classi- fication approach for image thresholding. Cheng et al. [[11]](#_bookmark21) used fuzzy c-partition entropy approach to select threshold. In this approach, parameterized fuzzy membership functions are used to classify pixels of image into target and background. This approach is based on the selection of an optimal combination of parameters of fuzzy membership functions for maximizing the entropy of fuzzy 2-partition. After this, the selected optimal parameters are used to find optimal threshold. Benabdelkader & Boulemden [[12]](#_bookmark21) proposed a recursive approach to search a suitable combination of parameters of fuzzy membership functions. In this approach, trapezium fuzzy membership function with two parameters is used. In 2011, Tang et al. [[13]](#_bookmark21) also proposed a recursive program- ming approach to find a suitable combination of fuzzy membership function parameters. This recursive approach is applied on two fuzzy membership functions (s-function and z-function) with three parameters. Tao et al. [[14]](#_bookmark21) applied ant colony optimization to search a combination of parameters of fuzzy membership func- tions for maximizing the entropy of fuzzy 2-partition. Li and Lee

[[15]](#_bookmark21) proposed minimum cross entropy approach for image seg- mentation. In 2011, Nie et al. [[16]](#_bookmark22) proposed a thresholding method that is based on two-dimensional cross entropy. In this method, gray level co-occurrence matrix is used to obtain two- dimensional cross entropy. Horng [[17]](#_bookmark23) proposed honey bee mating optimization algorithm for calculating the minimum cross entropy objective function. Tang et al. [[18]](#_bookmark24) used genetic algorithm to reduce computation burden for computing the minimum cross entropy objective function. Horng and Liou [[19]](#_bookmark25) proposed firefly algorithm to search multilevel thresholds by using cross entropy principal. Recently, fuzzy entropy based techniques have been pro- posed by numerous researchers to find optimal threshold. These techniques are frequently used for thresholding because it is the general belief that fuzziness and uncertainty exist in images. Although the cross entropy has been applied by many researchers to search multilevel thresholds for image segmentation, it is worth noting that selection of an optimal combination of thresholds for minimizing the cross entropy of fuzzy 2-partition in reasonable amount of time is a challenging task. Thus, selection of an optimal combination of thresholds for minimizing the cross entropy can be formulated as a combinatorial optimization problem. Over the last decade, various metaheuristic algorithms have been used by researchers to solve combinatorial optimization problems. Such algorithms are Genetic Algorithm (GA) [[18,20–22]](#_bookmark24), Ant Colony Optimization (ACO) algorithm [[14,23]](#_bookmark21), Biogeography based Opti- mization (BBO) approach [[24,25]](#_bookmark27), bacterial foraging optimization algorithm [[26,27]](#_bookmark28), gravitational search algorithm [[28]](#_bookmark29), cuckoo opti- mization algorithm [[29]](#_bookmark30), hybrid approaches [[30,31]](#_bookmark31), etc. Teaching- Learning-Based Optimization (TLBO) algorithm is a newly intro- duced member in the optimal algorithm family [[32,33]](#_bookmark35). It is inspired by the philosophy of teaching and learning. The search mechanism of TLBO algorithm is a population-based. Initially, a

set of some feasible solution candidates of the given problem is randomly generated called the population. After this, feasible solu- tions are modified to achieve optimal solution by the simulation of a classical school learning process. This process consists of two phases: teaching phase and student phase [[34]](#_bookmark36). Teacher phase sim- ulates the learning of the students through the teacher. During this phase, the best feasible solution acts as teacher. Other feasible solutions are improved by moving their positions towards the position of the teacher by taking into account the current mean value of the feasible solutions. Student phase simulates the learn- ing of the students through their mutual interaction. During this phase, two feasible solutions are randomly selected. If the first one is better than second one, then the first one is moved towards the second one. Otherwise, the first one is moved away from the second one. Main advantage of TLBO algorithm over other opti- mization algorithms is that it used only common controlling parameters while it is free from algorithm-specific parameters [[35]](#_bookmark37). Common controlling parameters are common in running any population based optimization algorithms like population size and number of generations while algorithm-specific parameters are specific to that algorithm and different algorithms have differ- ent algorithm-specific parameters to control. For example, GA’s algorithm-specific parameters are mutation rate and crossover rate. Similarly, BBO’s algorithm-specific parameters are maximum immigration rate, maximum emigration rate and mutation rate. The optimal selection of algorithm-specific parameters is also a problem. The improper selection of algorithm-specific parameters decreases the performance of optimization algorithms. Due to the improper selection of algorithm-specific parameters, either the computational cost of the algorithm will increase or yield the local optimal solution [[36]](#_bookmark38). Recently, TLBO algorithm has been widely applied to obtain global optimal solutions for a variety of optimization problems [[37–41]](#_bookmark39) with less computational cost and high consistency. From this motivation, the feasibility of TLBO algorithm is investigated to search an optimal combination of thresholds for minimizing the cross entropy.

The objective of this paper is to search an optimal combination of thresholds for minimizing the cross entropy. The selected opti- mal combination of thresholds is used to segment the input image. The proposed thresholding approach is called the TLBO-based min- imum cross entropy thresholding (TLBO-based MCET) algorithm. Here, the fitness function for TLBO algorithm is cross entropy of the input image and segmented image.

In this study, a set of five standard test images is used to eval- uate the performance of the proposed algorithm. The proposed approach is compared with three different approaches included Firefly-based minimum cross entropy thresholding (FF-based MCET) [[19]](#_bookmark25), Honey Bee Mating Optimization-based minimum cross entropy thresholding (HBMO-based MCET) [[17]](#_bookmark23) and Quan- tum Particle Swarm Optimization-based minimum cross entropy thresholding (Quantam PSO-based MCET) [[42]](#_bookmark40).

The rest of the paper is organized as follows. Section [2](#_bookmark8) intro- duces the TLBO algorithm. In Section [3](#_bookmark9), the proposed TLBO-based minimum cross entropy thresholding (TLBO-based MCET) algo- rithm is described in details. Performance evaluation is discussed in detail in Section [4](#_bookmark12). Finally, Section [5](#_bookmark20) concludes the paper and suggests future research directions.

1. Teacher-Learning-based Optimization algorithm

Teacher-Learning-based Optimization (TLBO) algorithm is a new kind of metaheuristic algorithm that is based on a teaching– learning process. It is firstly introduced by Rao et al. [[32]](#_bookmark35) to solve constrained mechanical design optimization problems. It is inspired by passing on knowledge within a classroom environment

where students first gain knowledge from a teacher and then through mutual interaction [[34]](#_bookmark36). TLBO algorithm is a population-

based optimization algorithm in which a group or class of students

*If F*(*Xk* ) > *F*(*Xk* )

*Xk* = *Xk*

*u* *v*

+ *r*(*Xk*

— *Xk* )

considered as population. Thus, a student of class represents a fea- sible solution of the problem. Different subjects offered to the class are considered as different design variables of the optimization problem and student’s result is treated as the fitness value of fea- sible solution of the optimization problem.

TLBO algorithm consists of two phases: Teacher phase and stu-

*new SP*;*u*;*j*

*Else*

*k*

*X*

*new SP*;*u*;*j*

*Endif*

*u*;*j*

*k u*;*j*

= *X*

*u*;*j*

+ *r*(*Xk*

*v*;*j*

*v*;*j*

*k u*;*j*

— *X* )

dent phase. Working of these phases is described below [[32–34,36]](#_bookmark35).

where *F*(*X*) is a fitness function that is used to find the fitness value

of feasible solution, *Xk*

*new SP*;*u*;*j*

denotes the *j*th design variable of the

* 1. *Teacher phase*

*new SP*;*u*

modified feasible solution in student phase at *k*th teaching- learning cycle.

The teacher phase means learning of the students from the tea- cher. On the basis of teaching-learning philosophy, the most expe-

*k*

After this, the fitness value of *Xk*

is evaluated

rienced, knowledgeable and highly learned person in the society is considered as teacher. The teacher tries to improve the knowledge

*k*

*new SP*;*u*

*If F*(*X*

) > *F*(*X*

*new*;*u*)

level of students and helps students to obtain good marks. But, stu- dents gain knowledge and obtain marks according to the quality of teaching delivered by the teacher and the quality of students pre- sent in the class. For simulation, suppose there are *‘n’* number of subjects (design variables, *j* = 1, 2,.. .,*n*) offered to *‘Np’* number of students (population size, *i* = 1, 2,.. .,*Np*). At any teaching-learning

cycle (iteration, *k* = 0, 1, 2,.. .*In*), *Mk* is the mean result of students

*j*

*k*

*new*;*u*

*X*

*Else*

*k new*;*u*

*X*

*Endif*

*k*

*new SP*;*u*

= *X*

*k new*;*u*

= *X*

in a particular subject *’j’*. Teacher is the most experienced, knowl- edgeable and highly learned person in the society. To simulate this concept, the best student (feasible solution) in the entire popula-

tion is considered as teacher. Let *Xk* be the most feasible solution of the population at *k*th teaching-learning cycle and *Xk* denotes the *j*th design variable in the best feasible solution of the popula-

*T*

*T*;*j*

tion at *k*th teaching-learning cycle i.e. the result of the teacher in

subject *‘j’*. The difference between the result of the teacher and the mean result of the students in subject *‘j’* is given by [[32]](#_bookmark35)

*Dk* = *r*(*Xk* — *TF Mk*) (1)

*j T*;*j j*

where *TF* is a teaching factor that decides the value of the mean to

be changed and *r* is a random number in the range [0 1]. *TF* is not a parameter of the TLBO algorithm and its value can either be 1 or 2 [[36]](#_bookmark38).

Feasible solutions (students) are improved by moving their positions towards the position of the best feasible solution (tea- cher) by taking into account the current mean value of the feasible solutions. To simulate this fact, the *i*th feasible solution in the pop- ulation at *k*th teaching-learning cycle is updated according to the following expression:

* 1. *Basic TLBO algorithm*

Based upon the above discussion, TLBO algorithm can be re- written in the following steps [[32,43]](#_bookmark35):

Step1: [Initialization] Initialize the optimization parameters

* + - Population size (the number of students or learners): *Np*
    - Number of iterations: *In*
    - Number of design variables or parameters (the subjects or courses offered)
    - Limits of design variables

Step2: [Initialize the population]

Generate random population according to the population size and the number of design variables.

Step3: [Fitness Evaluation]

Evaluate fitness of feasible solutions in the population and arrange these solutions according to their fitness values

Step4: [Teacher Phase]

Modify solution by simulating the concept: the learning of the students through the teacher

Step5: [Student Phase]

Modify solution by simulating the concept: the learning of the students through their mutual interaction

*j*

*Xk i j* = *Xk*

*new*; ;

*old*; ;

*i j* + *Dk*

(2)

Step6: [Repeat] Go to Step 3 until the stopping criteria (maxi-

*k new*;*i*

If *X*

is better than *Xk*

, then *Xk*

is accepted; Otherwise it is

mum iteration: In) is not met

Step7: Stop

rejected. All the accepted feasible solutions are maintained and these become the input to the student phase.

*old*;*i*

*new*;*i*

* 1. *Student phase*

In this phase, students gain knowledge through mutual interac- tion. A student interacts randomly with other students of the class to improve knowledge. A student (*u*) learns something new from

another student (*v*) of the class if student (*v*) has more knowledge than student (*u*). Thus, if student (*v*) is better than student (*u*), then student (*u*) is moved towards student (*v*). Otherwise, student (*u*) is moved away from student (*v*). The learning philosophy of this phase is simulated as below:

Two students (feasible solutions, *Xk* , *Xk* ) are randomly selected

1. Proposed TLBO-based MCET algorithm

In this section, first the objective function based on cross entropy is generated. After this, the generated objective function is minimized using Teaching-Learning-based Optimization (TLBO) algorithm to obtain multilevel threshold values for segmenting images.

* 1. *Cross entropy based objective function*

The cross entropy was developed by Kullback [[44]](#_bookmark40) in 1968. The cross entropy is a measure of closeness between two sampling dis-

tributions. Let *P* = {*p* ; *p* ; ... ; *p* } and *Q* = {*q* ; *q* ; ... ; *q* } be two

*u v* 1 2 *n*

1 2 *n*

numbers belong to [1; *Np*] and *u*–*v*. from the class (population), where, *u*, *v* are two integer random

probability distributions defined on the same set of values. The cross entropy between *P* and *Q* is defined as

*E P*; *Q*

*n*

*C* (

X*p* log *pi*

*i*=1

*i*

3 *E T*

*T*1

*rp* log *r*

X

*C* ( 1)=

*r*

*r*=0

*L*max

*rp* log *r*

X

*r*

*r*=*T*1+1

*T*1

*rp* log *S* 0 *T*

X

—

*r*

[ ( ;

1)]

*r*=0

having *L*max *+ 1* gray levels ranging from 0 to *L*max and *M* × *N* be Let *f*(*x, y*) be a mathematical function that defines a digital image the size of the image, then *f* (*x*; *y*)∈ {0; 1; 2; ... ; *r*; ... ; *L*max} is a gray

level value of the pixel that has coordinate position (*x*, *y*),

*L*max

— *rpr* log[*S*(*T*1 + 1; *L*max) (12)

X

)=

*i*

*q*

( )

+

*r*=*T*1+1

*L*max *T*1 *L*max

*x* ∈ {1; 2; .. . ; *M*} and *y* ∈ {1; 2; .. . ; *N*}. Let *hr* be the frequency of

*EC* (*T*1)= X*rpr* log *r* — X*rpr* log[*S*(0; *T*1)] — X *rpr* log[*S*(*T*1 + 1; *L*max)]

then *pr* = *hr*/(*M* × *N*) is the probability of occurrence of gray level occurrence of a particular gray level value *r* in the digital image *f*, value *r* in the image.

*r*=0

*r*=0

*r*=*T*1 +1

(13)

For bi-level image segmentation, a threshold value *T*1 ∈ [0; *L*max]

is selected that segments the image into two regions: one is Back-

ground (*BG*) and other is Target (*TG*). Gray level values of all pixels

pixels of *TG* region is greater than *T1*. In segmented image *gT*1 (*x*; *y*), of *BG* region is less than or equal to *T1* while gray level values of all belongingness of pixel (*x*; *y*) to *BG* and *TG* regions is described as

(*x*; *y*)=

In segmentation, the main goal is to minimize the variance between input image and segmented image. So here, the main objective is to find an optimal value of threshold value (*T1*) such that the total cross entropy (*EC*) of the input image and segmented image is minimal.

*L*max

## *T*\* = arg min[*EC* (*T*1)] (14)

*S*(0; *T* ) *f* (*x*; *y*) 6 *T* 1 *T*

*g*

1

1

(4)

1

*T*1

where

X

X

*S*(*T*1 + 1; *L*max) *f* (*x*; *y*) > *T*1

First term in Eq. [(12)](#_bookmark10) is a constant term. So the objective function can be rewritten as

P*b rp*

P

*S*(*a*; *b*)= *r*=*a r*

*b*

*r*=*a*

*pr*

*T*1 *L*max

*r*

*r*

(5)

*O*(*T*1)=—

*rp* log[*S*(0; *T*1)] —

*rp* log[*S*(*T*1 + 1; *L*max)] (15)

*r*=0

*r*=*T*1+1

Now, probability distribution of the gray level values of the input image region that belongs to *BG* region is described as

*rp* log

*r*=*T*1+

(16)

*T*1 "P*T*1 *rp* #

*L*max

X

"P*L*max

1*rpr*#

1 *r* P*T*1 *p*

X

*O*(*T* )=—

*rp* log

*r*=0 *r*

—

*rpr* where *r* = 0; 1; .. . ; *T*1

*r*=0

*r*=0

*r*

*r* P*L*max *p*

Probability distribution of the gray level values of *BG* region is

*r*=*T*1+1

*r*=*T*+11

*r*

Let *m*0(*a*; *b*)= P*b apr* and *m*1(*a*; *b*)= P*b arpr* , then

described as

*S*(0; *T*1)*pr* where *r* = 0; 1; 2; ... ; *T*1

*r*=

*O T m*1 0 *T*

## log *m*1(0; *T*1)

*r*=

*m*1 *T* 1 *L*

( 1)=—

( ;

1)

*m*0 (0; *T*1)

—

( 1 +

;

max)

Then, the cross entropy of the input image region that belongs to *BG* region and *BG* region is defined as

*T*1

## × log

*m*1(*T*1 + 1; *L*max)

*m*0(*T*1 + 1; *L*max)

## (17)

*EC BG*(*T*1)= X*rp* log *rpr*

;

*r*=0

*r*

*S*(0; *T*1)*pr*

(6)

For three-level image segmentation, two threshold values

*T*1; *T*2 ∈ [0; *L*max] are selected that segments the image into three

Now, probability distribution of the gray level values of the input image region that belongs to *TG* region is described as *rpr* , where

*r* = *T1 + 1, T1 + 2,* .. .*, L*max

regions.

### *O T T*

( 1;

2) =—

*m*1 0 *T*

( ;

## log *m*1(0; *T*1)

*m*1 *T* 1 *T*

( 1 +

;

2)

Probability distribution of the gray level values of *TG* region is described as

*S*(*T*1 + 1; *L*max)*p* , where *r* = *T1 + 1, T1 + 2,* .. .*, L*max

## × log

*m*1(*T*1 + 1; *T*2)

*m*0(*T*1 + 1; *T*2)

1)

*m*0(0; *T*1)

—

— *m*1(*T*2 + 1; *L*max)

*r*  *m*1(*T* + 1; *L* )

Then, the cross entropy of the input image region that belongs to *TG* region and *TG* region is defined as

× log

2 max

*m*0(*T*2 + 1; *L*max)

(18)

*EC TG*(*T*1)= X *rp* log *rpr*

;

*r*

*S*(*T*1 + 1; *L*max)*pr*

*r*=*T*1+1

*L*max

(7)

For *k*-level image segmentation, *k* — *1* threshold values

*T* ; *T* ; .. . ; *T* ; *T* ∈ [0; *L* ] are selected that segments the image

1

2

*k*—2

*k*—1

max

into *k* regions.

The total cross entropy of the input image and segmented image is

*O T T T*

*m*1 0 *T*

log *m*1(0; *T*1)

defined as

*E* (*T* )= *E* (*T* )+ *E* (*T* ) (8)

( 1; 2; ... *k*—1)= —

( ; 1)

*m*0 (0; *T*1)

*C* 1 *C*;*BG* 1

*T*

*C*;*TG* 1

*L*

— *m*1(*T*1 + 1; *T*2) log

*m*1(*T*1 + 1; *T*2)

*m*0 (*T*1 + 1; *T*2)

*E T* X1 *rp* log *rpr* Xmax *rp* log *rpr*  9

*r*

( )

*C* ( 1)=

*r*

*r*=0

*S*(0; *T*1)*p* +

*r*

*r*=*T*1+1

*S*(*T*1 + 1; *L*max)*pr*

— *m*1(*T*2 + 1; *T*3) log

*m*1(*T*2 + 1; *T*3)

*m*0

*T* X*rp* log *r*

*T*1

*EC* ( 1)=

*r*=0

*r*

*S*(0; *T*1)

*L*max

*rp* log 10

X *r*

+

*r T*

1

*r*

*S*(*T*1 + 1; *L*max)

(

)

(*T*2 + 1; *T*3)

*m*1(*T*

+ 1; *L*max) log

*k*—1 max

*m*0 (*Tk* 1 + 1; *L*max)

+ 1; *L* )

*T*1 *T*1

*Ec*(*T*1)= X*rpr* log *r* — X*rpr* log[*S*(0; *T*1)] + X *rpr* log *r*

= 1 +

*L*max

—

## (19)

— ... — *m*1(*Tk*

—1

*O*(*T*1; *T*2; .. . *Tk* 1)= —X*m*1(*Ti* 1; *Ti*) log *m* (*Ti*—1; *Ti*)

X

*r*=0

*r*=0

*r*=*T*1+1

*k*

1

*L*max

— *rpr* log[*S*(*T*1 + 1; *L*max)] (11)

—

*i*=1

—

*m*0 (*Ti*—1; *Ti*)

## (20)

*r*=*T*1+1

where *T0* = 0 and *Tk* = *L*max

* 1. *TLBO algorithm to minimize cross entropy for Selecting Multiple*

*m*0 (*Ti*

; *Ti*)

*threshold values*

1. 1 *m*1(*Ti*—1; *Ti*)

The formulation of optimization problem is as follows:

*k*

*O*(*T*1; *T*2; .. . *Tk*—1)=—

*i*=1

*m* (*Ti*—1; *Ti*) log

—1

(21)

Minimize *O*(*T*)

Where *O(T)* is the objectives function and *T* is a vector for design

variables

(T1, T2, .. .,*Tk*—1)

Subject to constraints

* 1. 0 6 *T*1; *T*2;...; *Tk*—1 6 *L*max
  2. 0 6 *T*1 6 *T*2 6 .. . 6 *Tk*—1 6 *L*max

In case of gray level images whose pixel values are 8 bits,

*L*max = 255

tion of *T*1; *T*2; ... ; *Tk*—1 such that *O*(*T*1; *T*2; .. . *Tk*—1) is minimized. In the other words, the objective is to find an optimal combina- Here, objective function is

In case of gray level images that have 256 gray levels, the search space of parameters is as follow:

## (*T*1; *T*2; .. . *Tk*—1)∈ [0; 255]

In this paper, Teaching-Learning-based Optimization (TLBO)

algorithm is used to find an optimal combination of threshold val- ues for minimizing the cross entropy. The block diagram of the pro- posed algorithm is shown in [Fig. 1](#_bookmark11) and the detail is introduced as follows:

Initialization

Step 1: Initialize the parameters

* Population size (the number of students or learners): *Np*
* Number of iterations: *I*max
* Number of design variables or parameters (the subjects or courses offered: *k* — *1*): *T*1; *T*2; ... ; *Tk*—1
* Limits of design variables: 0 6 *T*1; *T*2; ... *Tk*—1 6 255

**Fitness Evaluation**

Select the Best Solution who acts as a Teacher

Select Two Feasible Solutions Randomly

Yes

*Termination Criterion Satisfied*

No

Modify Feasible Solutions by Simulating the Concept: *the Learning of the Students through their Mutual Interaction*

Modify Feasible Solutions by Simulating the Concept: *the Learning of the Students through the Teacher*

Arrange Feasible Solutions according to their Fitness Values

Population of Feasible Solutions

**Generate New Population**

Select Best Feasible Solution: *Optimum Solution*

**Student Phase**

**Teacher Phase**

**Initialize**

Parameters

Fig. 1. Block Diagram of Proposed Algorithm.

Step 2: Initialize the population

Generate random population according to the population size and the number of design variables.

function for a feasible solution is minimum, then it is called the most fit feasible solution)

*FI* = *O*(*XI*), *I* = 1, 2, .. . , *I*max*and i* = 1, 2, 3, .. . , *Np*

*i* *i*

*Pop* = *X*[*Np*, *k* — 1]

Population is represented by a matrix of size *Np*× *k* — *1*

Arrange feasible solution candidates of the population in ascending order according to their respective objective function numeric val-

ues*XI* and the corresponding fitness function value *FI*

{*Xi*1}∈ *T*1, {*Xi*2}∈ *T*2, .. . {*Xik*—1}∈ *Tk*—1, *i* = 1, 2, 3, ... , *Np*

Population is a random set of individuals. It is not necessary

that randomly generated population satisfies the constrain: 0 6 *T* 6 *T* 6 ... 6 *T* 6 *L* . To solve this problem, the follow-

6

*I*

*i*

2 *XI* 3

1

6 *X*2 7

*I*

7

64

75

*i*

2 *FI* 3

1

6 *F*2 7

*I*

=

·

6 7

64

75

*I*

1 2 *k*—1

max

i.e.*X* = ·

and *F*

ing mathematical processing is done: For bi-level segmentation

· ·

1. *FI*

' = *Xi*1

*X*

*i*1

{*X*' }∈ *T*1

*i*1

For three-level segmentation

' = *Xi*1

*X*

*i*1

*X*'2 = *X*'1 + (255 — *X*'1)\* (*Xi*2/255)

*i i* *i*

# {*X*' }∈ *T*1, {*X*' }∈ *T*2

*Np Np*

{*XI* }∈ *T*1, {*XI* }∈ *T*2, {*XI* }∈ *T*3, ... , {*XI* }∈ *Tk* 1

*i*1 *i*2 *i*3 *ik*—1 —

Teacher Phase

Step 4: Modify solution by simulating the concept: the learn- ing of the students through the teacher

This phase simulates the learning of the students through the teacher. During the teacher phase, the best feasible solution candi- date acts as teacher. Other feasible solution candidates are

improved by moving their positions towards the position of the

*i*1 *i*2

For four-level segmentation

teacher by taking into account the current mean value of the feasi- ble solution candidates.

' = *Xi*2

*X*

*i*2

For this, select the best solution (*XI I*

*i*

*O*(*X* )=min

) who acts as a teacher

## *X*' = *X*' + (255 — *X*' )\* (*Xi*3/255)

for that iteration

*i*3 *i*2 *i*2

*XI I* *I*

' '

*X*

= *X*

*i*

*i*1 *i*12

\* (*Xi*1

## /255)

*Teacher* = *X*1 = *XO*(*XI* )=min

Calculate the mean of the population column-wise, which will

# {*X*' }∈ *T*1, {*X*' }∈ *T*2, {*X*' }∈ *T*3

*i*1 *i*2 *i*3

For more than four level segmentation (fifth-level segmentation

to *k*-level segmentation)

give the mean for the particular course (design variable or parameter) as

*MI* = [*mI* , *mI* , *mI* , .. . , *mI* ]

*X*' = *Xi*2

1 2 3

*k*—1

*i*2

' '

*X*

= *X*

*i*3 *i*2

' '

*X*

= *X*

*i*1 *i*2

' '

*X*

= *X*

*i*4 *i*3

+ (255 — *X*' )\* (*Xi*3/255)

\* (*Xi*1/255)

*i*2

+ (255 — *X*' )\* (*Xi*4/255)

*i*3

*ik*—2

*I*

*mI* = 11

1

*X*

*X*

*I*

*mI* = 12

2

*XI*

*I*

21

+ *X*

*I*

+ *X*

22

+ *XI*

+· · · . + *XI*

*Np*

*Np* 1

# +· · · + *XI*

*Np* 2

*Np*

# +· · · + *XI*

'

*X*

*ik*—1

'

*ik*—2

= *X*

# + (255 — *X*'

)\* (*Xik*—1/255)

*mI* = 13 23

#### *Np*

3

*Np* 3

## {*X*' }∈ *T*1, {*X*' }∈ *T*2, {*X*' }∈ *T*3, .. . , {*X*' }∈ *Tk*—1

*i*1 *i*2 *i*3

*ik*—1

*XI* + *XI*

+· · · + *XI*

Thus, after the above said modification, population is described as follows

*I*

*k*—1

*m*

= 1*k*—1 2*k*—1 *Npk*—1

#### *Np*

' The teacher will try to shift the mean from *MI* towards *XI*

*Pop* = *X* [*Np*, *k* — 1]

Hence, initial generated population is as follows:

which will act as a new mean for the iteration. So,

*Teacher*

*I*

*M*

*new*

*I*

*Teacher*

= *X*

*XI* = *X*'

The difference between two means is expressed as

where *I* = 0, 1, 2, ... , *I*

*new*

max

*DI* = *r*(*MI*

— *TF MI*)

Initially, *I* = 0

Fitness Evaluation

Step 3: Evaluate fitness of feasible solutions in the popula- tion and arrange these solutions according to their fitness

where *TF* is a teaching factor that decides the value of the mean to be changed, and *r* is a random number in the range [0 1]

Now, the above difference between two means modifies the existing solution according to the following expression

values

Evaluate the value of the objective function for each feasible

*old*,*i*

*i*

*I*

*new TP*,*i*

*X*

*I*

*old*,*i*

= *X*

+ *DI*

solution candidate as its fitness (if the numeric value of objective

where *XI*

= *XI*

Find the fitness function value *FI*

*XI* , *i* = 1, 2, .. . , *Np*

*new TP*,*i*

of each

(*continued*)

*Input:*

*new TP*,*i*

*I*

*IfF*

*new TP*,*i*

*I*

*old*,*i*

< *F*

&& 0 6 *X*

*new TP*,*i*1 6

{*Xi*1}∈ *T*1, {*Xi*2}∈ *T*2, ... {*Xik*—1}∈ *Tk*—1*, i* = 1, 2, 3, .. .,*Np*

/⁄modify population to satisfy the constraint:

*I*

*X*

*new TP*,*i*2

*I*

*new TP*,*ik*—1

6 ... 6 *X*

= *X*

*I*

6 255

0 6 *T*1 6 *T*2 6 ... 6 *Tk*—1 6 *L*max⁄/

/⁄for bi-level segmentation⁄/

*I*

*X*

*new*,*i*

*Else*

*I*

*X*

*new*,*i*

*I*

*new TP*,*i*

*I*

= *X*

*old*,*i*

' = *Xi*1

{*X*' }∈ *T*1

*X*

*i*1

*i*1

/⁄for three-level segmentation⁄/

*Endif*

' = *Xi*1

' '

*X*

*X*

= *X*

*i*1

*i*2 *i*1

+ (255 — *X*' )\* (*Xi*2/255)

{*X*' }∈ *T*1, {*X*' }∈ *T*2

*i*1

where *FI* = *FIi* = 1, 2, .. . *N*

*i*1 *i*2

*old*,*i i p*

/⁄for four-level segmentation⁄/

Student Phase

Step 5: Modify solution by simulating the concept: the learn- ing of the students through their mutual interaction

' = *Xi*2

' '

*X*

*X*

= *X*

*i*2

*i*3 *i*2

*X*' = *X*'

+ (255 — *X*' )\* (*Xi*3/255)

\* (*Xi*1/255)

*i*2

*i*1 *i*12

In this phase, students gain knowledge through mutual interac-

{*X*' }∈ *T*1, {*X*' }∈ *T*2, {*X*' }∈ *T*3

*i*1 *i*2 *i*3

*I*

tion. A student (*X* ) tries to improve knowledge by peer learning /

*u*

from an arbitrary student (*XI* ), where, *u*, *v* are two integer random

numbers belong to [1, *Np*] and *u*–*v*. If *XI* is better than *XI* , then *XI*

*v*

⁄for more than four level segmentation

(fifth-level segmentation to *k*-level segmentation)⁄/

*X*'2 = *Xi*2

*v u u* *i*

is moved towards *XI* . Otherwise, *XI* is moved away from *XI* . This

*X*'3 = *X*'2 + (255 — *X*'2)\* (*Xi*3/255)

*v u* *v*

*i i* *i*

concept is simulated as follows:

*i*

*i*

*X*'1 = *X*'2 \* (*Xi*1/255)

Select two feasible solution candidates *XI* and *XI* from *XI*

*X*'4 = *X*'3 + (255 — *X*'3)\* (*Xi*4/255)

*u v new*

*i i* *i*

.. .... ........⁄⁄........................................

*I I X*'

= *X*'

+ (255 — *X*'

)\* (*Xik*—1/255)

*IfFu* < *Fv*

*ik*—1

*ik*—2

*ik*—2

*I I I* *I*

{*X*'1}∈ *T*1, {*X*' }∈ *T*2, {*X*' }∈ *T*3, ... , {*X*'

}∈ *Tk*—1

*Xnew SP*

= *X* + *r*(*X*

— *X* )

*i i*2 *i*3

*ik*—1

*Else*

,*u u u* *v*

/⁄representation of *i*th feasible solution candidate of the population⁄/

*XI* = *XI* + *r*(*XI* — *XI* )

' ' ' ' '

*new SP*,*u u v u*

*Endif*

where *FI* , *FI* are fitness values of *XI* , *XI* respectively.

*Xi* ← [*Xi*1 *Xi*2 *Xi*3 .. . *Xik*—1]

*X*' = [*Ti*1 *Ti*2 *Ti*3 .. . *Tik*—1]

*i*

/⁄modified population⁄/

*Pop* = *X*'

/⁄initially generated population⁄/

*u v u v*

After this, evaluate the fitness value (*FI*

*new SP*,*u*

*I*

*new SP*,*u*

) of *X*

*XI* = *X*'

*I* = 0, 1, 2, ... , *I*max

*I*

*If F*

*new SP*,*u*

*XI*

*I*

*new*,*u*

< *F*

6 *XI*

&& 0 6

6 ... 6 *XI*

6 255

/⁄initial⁄/

*I* = 0

*new SP*,*u*1 *I*

*X*

= *X*

*new*,*u*

*new SP*,*u*2

*I*

*new SP*,*u*

*new SP*,*uk*—1

/⁄representation of *i*th feasible solution candidate of the initially generated population⁄/

*Else*

*XI* ← [*XI XI XI* ... *XI* ]

*i i*1 *i*2 *i*3 *ik*—1

*i*1

*i*2

*i*3

*ik*—1

*I*

*X*

= *X*

*new*,*u*

*I*

*new*,*u*

{*XI* }∈ *Ti*1, {*XI* }∈ *Ti*2, {*XI* }∈ *T*

*i*3, ... , {*XI*

}∈ *T*

*ik*—1

*Else*

*XI* = [*TI TI TI* ... *TI*

], *i* = 1, 2, 3, .. ., *Np*

*i i*1 *i*2 *i*3

*Output:*

*ik*—1

Thus, new population is generated as follows:

/⁄representation of the best feasible solution candidate with the largest fitness value⁄/

*I*+1 *I*

*XI*max = *XI*max = *XI*max

=[ *XI*max

*XI*max

*XI*max ... *XI*max ]

*X* = *Xnew*

*Best* 1

*O*(*XI*max )=min

11 12 13

*ik*—1

= [*T*\* *T*\* *T*\* ... *T*\* ]

*i*

Step 6: Go to Step 3 until the stopping criteria (maximum iter- ation: *I*max) is not met.

Step 7: Stop

1 2 3

*Begin*

*I* = 0

*k*—1

Pseudo- Code of TLBO Algorithm to Minimize Cross Entropy for Selecting Multiple Threshold Values

*Input:*

Population Size: *Np*

Number of Iterations: *I*max

Number of Variables (*k* — 1) : *T*1, *T*2, ... , *Tk*—1

*for i* = 1 *to Np do XI* = *XI*[*i*, *k* — 1]

*FI* = *O*(*XI*)

*i*

*i* *i*

*endfor*

/⁄arrange feasible solution candidates of the population in ascending order according to their respective fitness values and find best fit feasible solution candidate along with

its fitness value⁄/

that satisfy the constraint: 0 6 *T*1, *T*2, ... *Tk*—1 6 255⁄/ *Pop* = *X* [*Np*, *k* — *1*] /⁄generate a random set of *Np* individuals (population)

*I*

*Best I*

*X*

*F*

*Best*

= *XI*

= *FI*

1

1

(*continued on next page*)

(*continued*)

*Input:*

*While*(*I* < *I*max)*do Teacher* = *XI*

*Best*

*for j* = 1*tok* — 1*do sum mj* = 0

*endfor*

*for i* = 1*toNpdo*

*sum m*1 = *sum m*1 + *XI*[*i*, 1]

*sum m*2 = *sum m*2 + *XI*[*i*, 2]

.

.

*sum mk*—1 = *sum mk*—1 + *XI*[*i*, *k* — 1]

*endfor*

*m*1 = *sum m*1/*Np m*2 = *sum m*2/*Np*

.

.

*mk*—1 = *sum mk*—1/*Np*

*M* = [*m*1 *m*2 *m*3. .*mk*—1]

*r* = *rand*(0, 1) *TF* = *rand*(0, 1)

*D* = *r*(*Teacher* — *TF* \* *M*)

*for i* = 1*toNpdo*

(*continued*)

*Input:*

respective fitness values and find best fit feasible solution candidate⁄/

*XI*+1 *I*+1

*Best* = *X*1 *I* = *I* + 1

*endwhile*

*End*

Pseudo-Code of Fitness Value

*Input:*

/⁄input image⁄/

*f* = imread (‘Input Image’)

/⁄feasible solution candidates of the population in an arbitrary iteration⁄/

*Xii* = 1, 2, ... , *Np*

/⁄number of segments required: *k*⁄/

2 6 *k* < *L*max

*Output:*

/⁄fitness values of feasible solutions of the population in an arbitrary iteration⁄/

*Fi*, *i* = 1, 2, ... , *Np*

*I*

*X*

*new TP*,*i*

*I*

*F*

*new TP*,*i*

*If FI*

= *XI* + *D*

*new TP*,*i* )

*i*

= *O*(*X*

*I*

< *FI*&& 0 6 *XI*

6 *XI*

*Begin*

/⁄find size of input image⁄/ [*M*, *N*] = size (*f*)

/⁄find the number of pixels in the image whose gray level *r,*

*new TP*,*i*

6 ... 6 *XI*

*i*

6 255

*new TP*,*i*1

*new TP*,*i*2

*r = 0, 1, 2,* .. .*,L*max⁄/

*I*

*X*

*new*,*i*

*else*

*I*

*X*

*new*,*i*

*endif endfor do*

*new TP*,*ik*—1

*I*

= *X*

*new TP*,*i*

= *XI*

*i*

*hr* , *r* = 0, 1, 2, ... , *L*max

/⁄find the probability distribution of the gray level values of

the image⁄/

*p*(*r*)= *hr*/(*M* × *N*)

*for i* = 1 to *Np*

*T1* = *X* [*i*, 1]

*T2* = *X* [*i*, 2]

*u* = *rand*(1, *Np*)

*v* = *rand*(1, *Np*)

}*while*(*u*–*v*)

*Xu* = *X*

*I*

*new*,*u*

*I*

*Xv* = *X*

*new*,*v*

*Xu* = *O*(*Xu*) *Xv* = *O*(*Xv* )

*If Fu* < *Fv*

*XSP*,*u* = *Xu* + *r*(*Xu* — *Xv* )

*else*

*XSP*,*u* = *Xu* + *r*(*Xv* — *Xu*)

*endif*

*FSP*,*u* = *O*(*XSP*,*u* )

*If FSP*,*u* < *FI* &&0 6 *XSP*,*u*1 6 *XSP*,*u*2 6 ... 6 *XSP*,*uk*—1 6 255

*u*

.

.

*Tk*—*1* = *X* [*i*, *k* — 1]

/⁄find cross entropy⁄/

*S* = *0*

*for j* = *0* to *k S1j* = *0*

*S2j* = *0*

*S3j* = *0*

*if j* = *1*

*a* = *0*

*b* = *Tj else*

*if j* = *k*

*a* = *Tj*—*1* + 1

*I*

*X*

*new*,*u*

*else*

*I*

*X*

*new*,*u*

*endif*

= *XSP*,*u*

*I*

= *X*

*new*,*u*

*b* = *L*max

*else*

*a* = *Tj*—*1* + 1

*b* = *Tj*

*endif*

/⁄generation of new population⁄/

*for i* = 1*toNpdo*

*I*+1 *I*

*X* = *X i*

*i new*,

*endfor*

/⁄arrange feasible solution candidates of the new population in ascending order according to their

*endif*

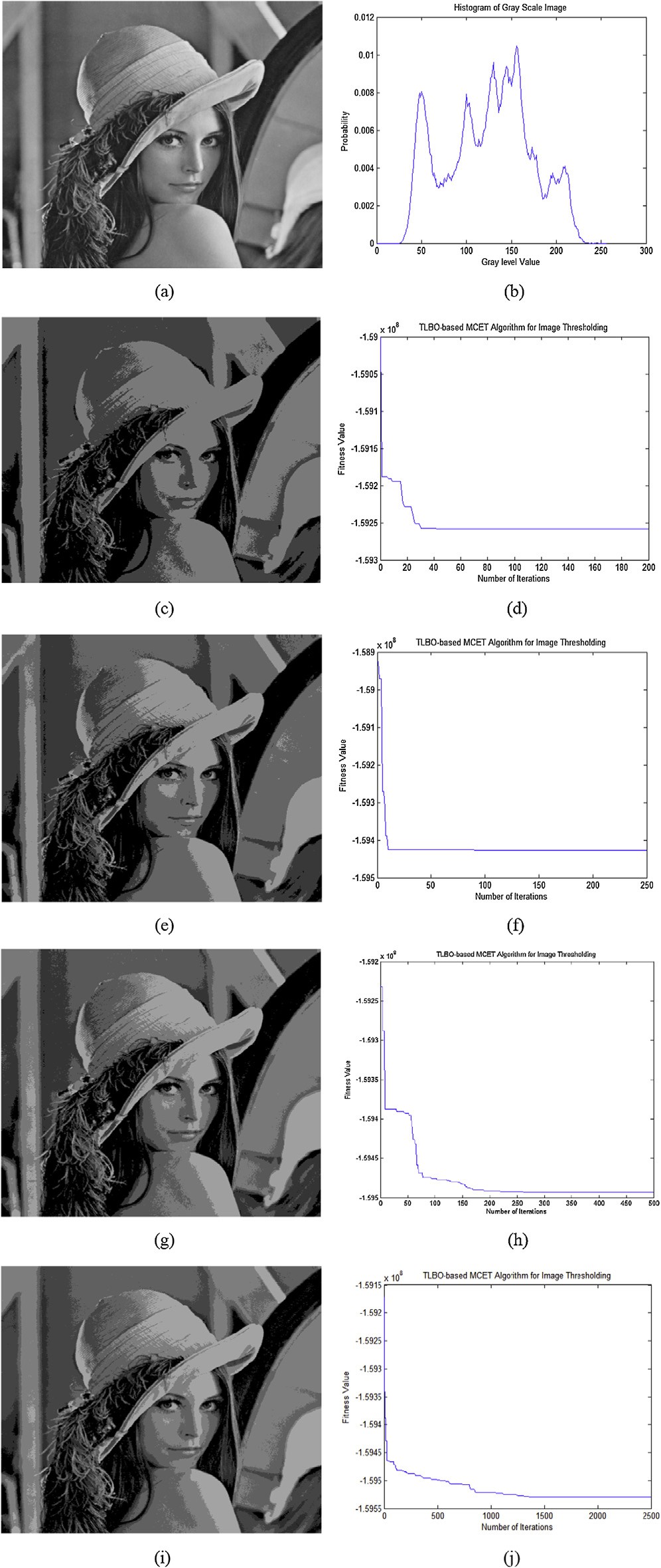
*for r1* = *a* to *b*

*S1j* = *S1j* + *p(r1) S2j* = *S2j* + *r1*⁄*p(r1)*

*endfor*

*Lj* = log(*S*2*j* )

*S*1*j*

(*continued*)

*Input:*

*for r2* = *a* to *b*

*S3j* = *S3j* + *r2*⁄*p(r2)*⁄*Lj endfor*

*S* = *S* + *S3j*

*Fi =* —*S endfor*

*endfor*

*End*

1. Experimental results &Analysis

In this section, the proposed TLBO-based minimum cross entropy thresholding (TLBO-based MCET) algorithm is imple- mented in MATLAB 7.7.0 (R2008b) with 2.2 GHz Intel(R) Core (TM) 2 Duo CPU T7500 machine of 1.99 GB RAM. The fitness func- tion for the proposed work is cross entropy of the input image and segmented image as shown in the following equation:

X 1 *m*1(*Ti*—1, *Ti*)

*k*

*O*(*T*1, *T*2, .. . *Tk*—1)=—

*i*=1

*m* (*Ti*—1, *Ti*) log

*m*0(*Ti*

—1

, *Ti*)

(22)

Here, Teaching-Learning-based Optimization (TLBO) algorithm is used to find an optimal combination of threshold values

(*T*1, *T*2, .. . *Tk*—1) for minimizing the cross entropy.

‘‘Cameraman” and ‘‘Goldhill” with size 512 × 512, 512 × 512, Five standard test images, named ‘‘Lena”, ‘‘Pepper”, ‘‘Bird”,

204 × 204, 256 × 256 and 512 × 512, respectively, are used for

conducting experiments. These original standard images are

shown in [Fig. 2](#_bookmark13)(a) - [6](#_bookmark15)(a) and their histograms are shown in [Fig. 2](#_bookmark13)

(b) - [6](#_bookmark15)(b). The segmented images with different thresholds using the proposed approach are illustrated in [Fig. 2](#_bookmark13)(c, e, g, i) – [Fig. 6](#_bookmark15)(c, e, g, i) and the performance characteristics of the proposed approach to segment five standard test images with different thresholds are displayed in [Fig. 2](#_bookmark13)(d, f, h, j) – [Fig. 6](#_bookmark15)(d, f, h, j). As we increase the number of thresholds, the segmented image rapidly tends the original image form the visual point of view. [Fig. 2](#_bookmark13)(c, e, g, i) – [Fig. 6](#_bookmark15)(c, e, g, i) show that segmented images obtained from the proposed approach at different levels are visu- ally acceptable. Thus, from subjective evaluation point of view, it is observed that the proposed approach (TLBO-based MCET algo- rithm) has ability to segment images. For comparative analysis, three other algorithms namely Quantum PSO-based MCET, FF- based MCET and HBMO-based MCET are considered. (See [Figs. 3–5](#_bookmark14)) In order to obtain the objective evaluation of the proposed algo- rithm and the consistent comparison analysis with other methods, two most popular objective elevation parameters, peak signal to noise ratio (PSNR) [[17,19]](#_bookmark23) and uniformity [[16,18]](#_bookmark22), are used. PSNR

is defined as

*PSNR* = 20log10

## 255 23

### *RMSE*

( )

Fig. 2. (a) The test image ‘‘Lena”, (b) its histogram, (c) 2-level threshoding image by

where RMSE is the root mean-squared error that is defined as

## sﬃPﬃﬃﬃﬃﬃ*M*ﬃﬃﬃﬃﬃPﬃﬃﬃﬃﬃ*N*ﬃﬃﬃﬃﬃ[ﬃﬃ*f*ﬃﬃ(ﬃﬃ*i*ﬃ,ﬃﬃ*j*ﬃﬃ)ﬃﬃﬃ—ﬃﬃﬃﬃ*g*ﬃﬃﬃ(ﬃﬃ*i*ﬃ,ﬃﬃ*j*ﬃﬃﬃﬃﬃ2ﬃﬃ

TLBO-based MCET algorithm, (d) the performance characteristics of TLBO-based MCET algorithm at 2-level thresholding, (e) 3-level threshoding image by TLBO-

based MCET algorithm, (f) the performance characteristics of TLBO-based MCET

### *RMSE* =

*i*=1 *j*=1 )]

*M* \* *N*

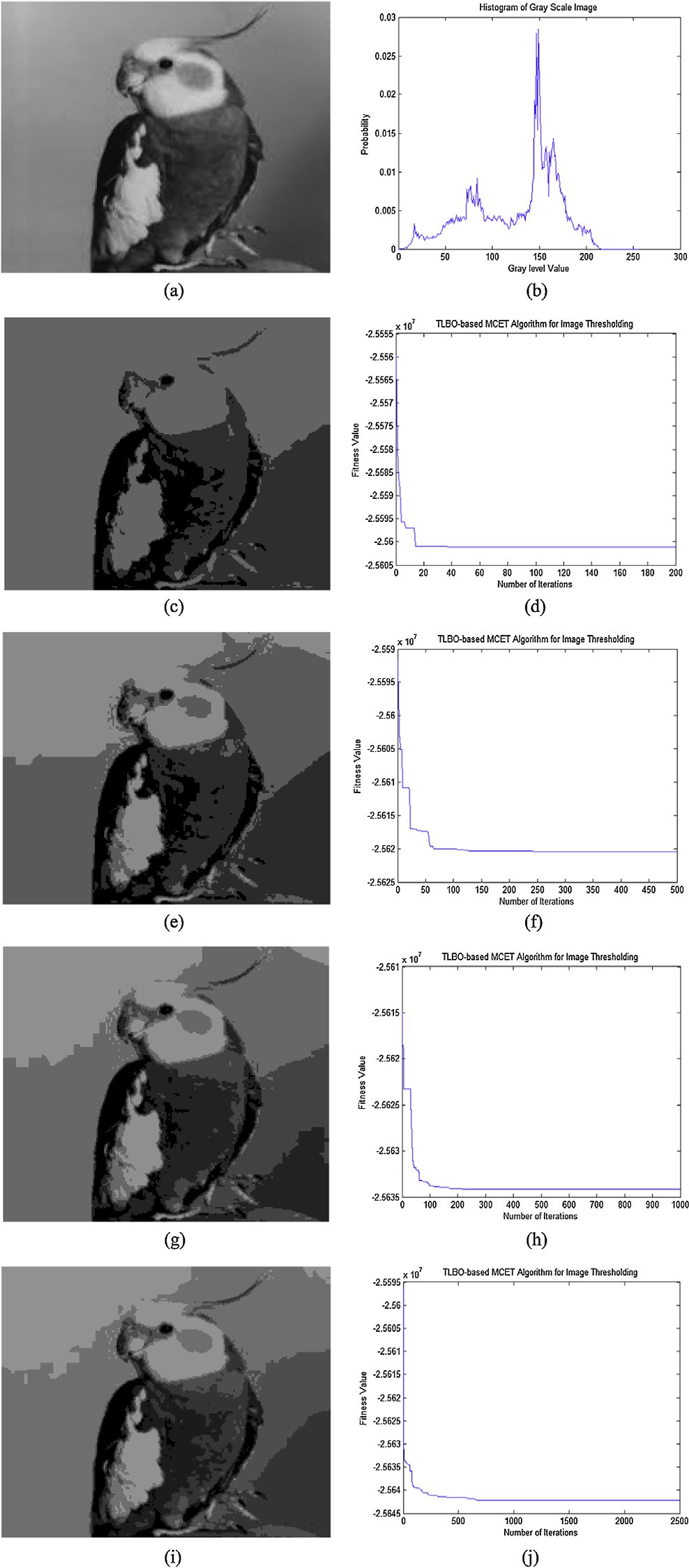
## (24)

algorithm at 3-level thresholding, (g) 4-level threshoding image by TLBO-based MCET algorithm, (h) the performance characteristics of TLBO - based MCET algorithm at 4-level thresholding, (i) 5-level threshoding image by TLBO-based

Here *f* and *g* are input and segmented images of size *M* × *N*,

respectively.

MCET algorithm and (j) the performance characteristics of TLBO - based MCET algorithm at 5-level thresholding.



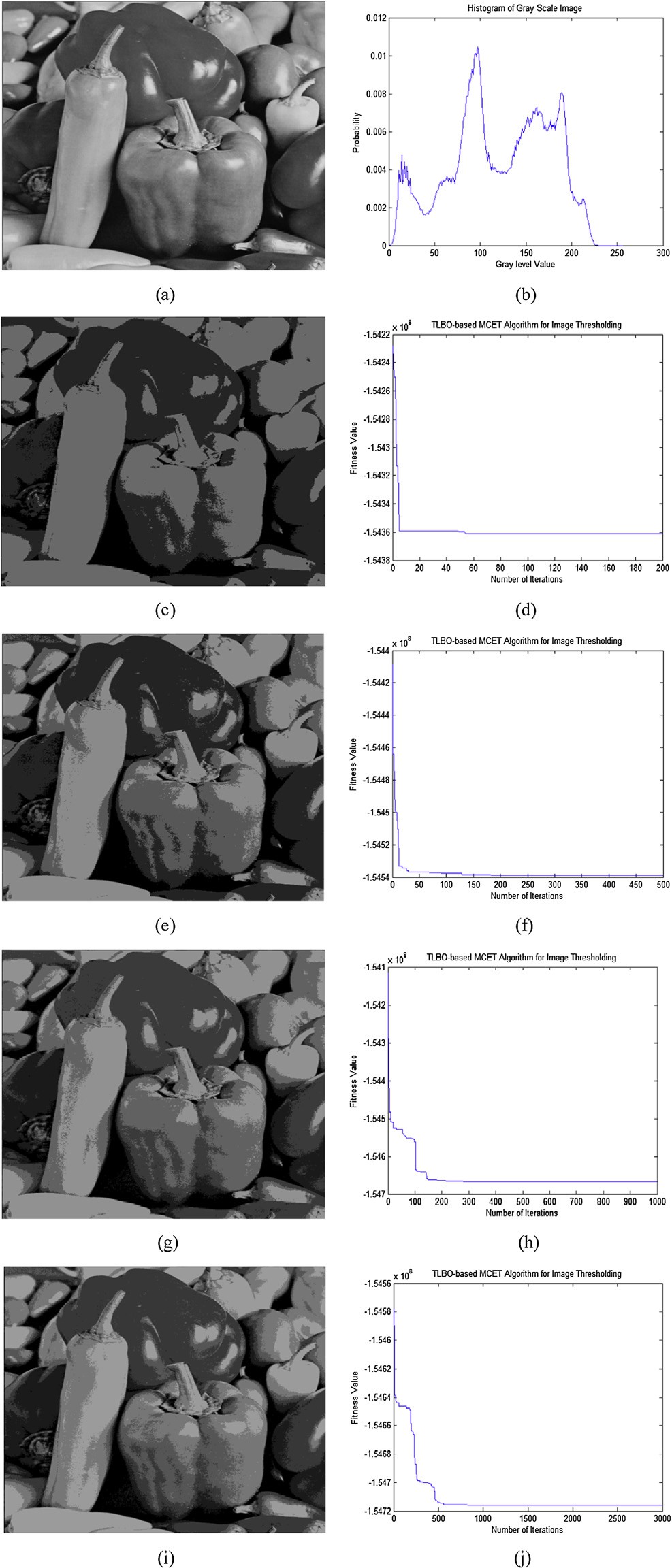
Fig. 3. (a) The test image ‘‘Pepper”, (b) its histogram, (c) 2-level threshoding image by TLBO-based MCET algorithm, (d) the performance characteristics of TLBO-based MCET algorithm at 2-level thresholding, (e) 3-level threshoding image by TLBO- based MCET algorithm, (f) the performance characteristics of TLBO-based MCET algorithm at 3-level thresholding, (g) 4-level threshoding image by TLBO-based MCET algorithm, (h) the performance characteristics of TLBO - based MCET algorithm at 4-level thresholding, (i) 5-level threshoding image by TLBO-based MCET algorithm and (j) the performance characteristics of TLBO - based MCET algorithm at 5-level thresholding.

Fig. 4. (a) The test image ‘‘Bird”, (b) its histogram, (c) 2-level threshoding image by TLBO-based MCET algorithm, (d) the performance characteristics of TLBO-based MCET algorithm at 2-level thresholding, (e) 3-level threshoding image by TLBO- based MCET algorithm, (f) the performance characteristics of TLBO-based MCET algorithm at 3-level thresholding, (g) 4-level threshoding image by TLBO-based MCET algorithm, (h) the performance characteristics of TLBO - based MCET algorithm at 4-level thresholding, (i) 5-level threshoding image by TLBO-based MCET algorithm and (j) the performance characteristics of TLBO - based MCET algorithm at 5-level thresholding.

Uniformity parameter measures region homogeneity in image and it is defined as

*k*

PSNR is measured in decibel (dB) and is used to determine the quality of the segmented images. Higher PSNR indicates that the

*U* = 1 —

1

*k*=0

h2P

P(*i*,*j*)∈*Rk*

{*f* (*i*, *j*)— l }2i

## (25)

quality of segmented image quality is better.

*M* \* *N* \* (*f* max — *f* min)2

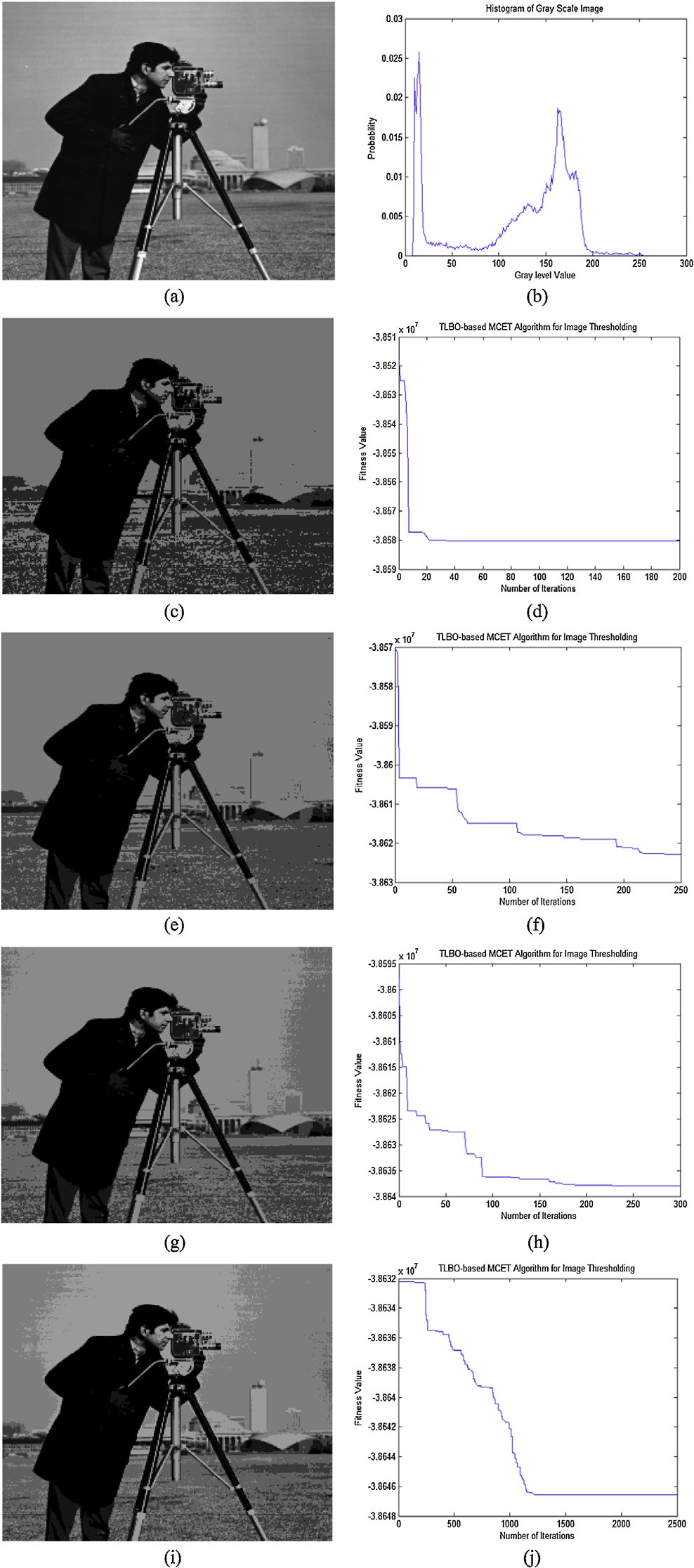
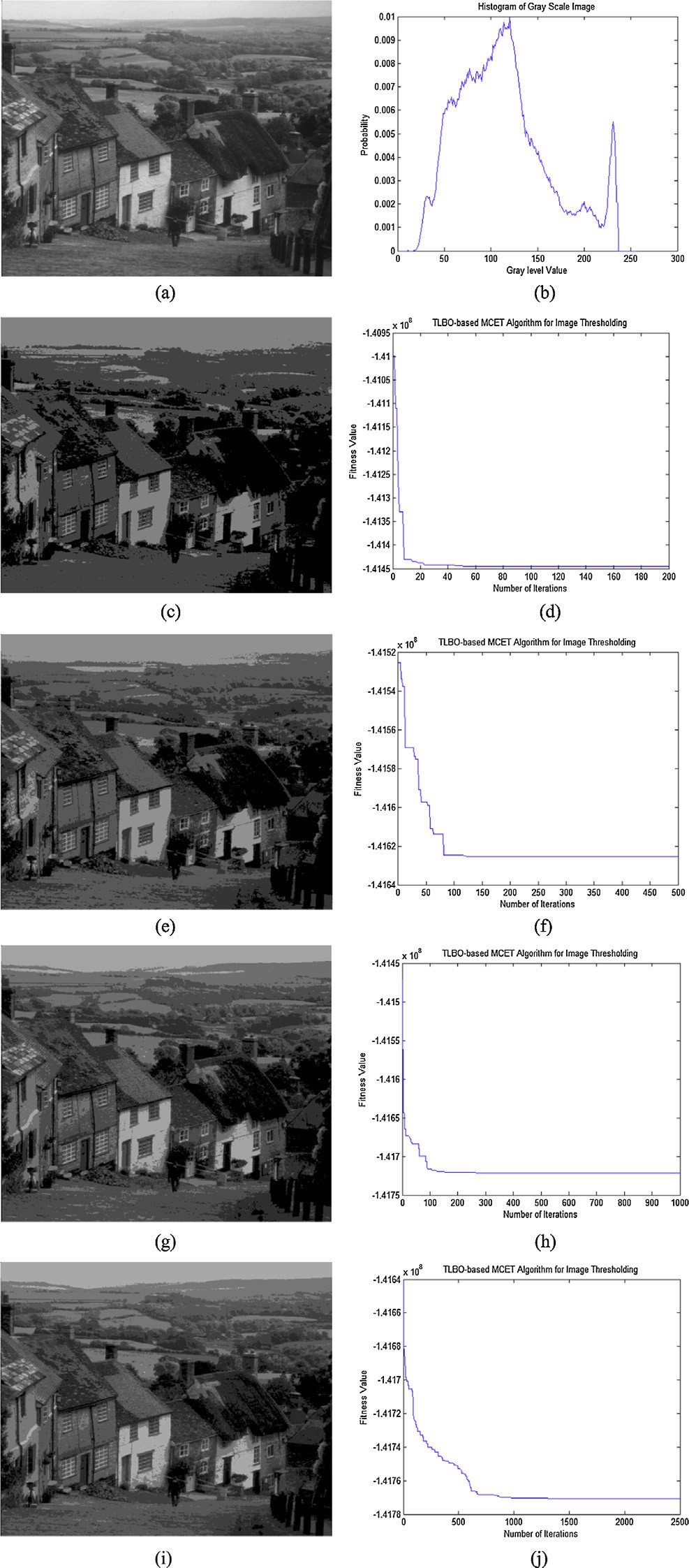
 

Fig. 5. (a) The test image ‘‘Cameraman”, (b) its histogram, (c) 2-level threshoding image by TLBO-based MCET algorithm, (d) the performance characteristics of TLBO- based MCET algorithm at 2-level thresholding, (e) 3-level threshoding image by TLBO-based MCET algorithm, (f) the performance characteristics of TLBO-based MCET algorithm at 3-level thresholding, (g) 4-level threshoding image by TLBO- based MCET algorithm, (h) the performance characteristics of TLBO - based MCET algorithm at 4-level thresholding, (i) 5-level threshoding image by TLBO-based MCET algorithm and (j) the performance characteristics of TLBO - based MCET algorithm at 5-level thresholding.

Fig. 6. (a) The test image ‘‘Goldhill”, (b) its histogram, (c) 2-level threshoding image by TLBO-based MCET algorithm, (d) the performance characteristics of TLBO-based MCET algorithm at 2-level thresholding, (e) 3-level threshoding image by TLBO- based MCET algorithm, (f) the performance characteristics of TLBO-based MCET algorithm at 3-level thresholding, (g) 4-level threshoding image by TLBO-based MCET algorithm, (h) the performance characteristics of TLBO - based MCET algorithm at 4-level thresholding, (i) 5-level threshoding image by TLBO-based MCET algorithm and (j) the performance characteristics of TLBO - based MCET algorithm at 5-level thresholding.

l*k*

where *Rk* is the *k*th segmented region; *f* (*i*, *j*) is the gray level value of

(*i*,*j*)∈*Rk f* (*i*, *j*)

*nk*

= P

## (26)

pixel (*i*, *j*); *f*max and *fmin* are the maximum and minimum gray level in the input image, respectively and l*k* is the mean gray level of pix- els in *k*th region that is defined as

Here *nk* is the total number of pixels in the segmented region *Rk*.The value of the uniformity lies between 0 and 1. For the better seg- mented image quality, the value of uniformity should be higher.

For evaluating the performance of the proposed TLBO-based MCET algorithm, five standard test images are segmented by this algorithm at 2-level, 3-level, 4-level and 5-level. The performance metrics for checking the effectiveness of the proposed approach are PSNR and uniformity which are used to determine the quality of the segmented images. For comparison, the results of Quantum PSO-based MCET, FF-based MCET and HBMO-based MCET algo- rithms are also considered for the same five standard test images. [Table 1](#_bookmark16) shows the selected thresholds (2, 3, 4 and 5 threshold val- ues) of the five test images using TLBO-based MCET algorithm, Quantum PSO-based MCET, FF-based MCET and HBMO-based MCET algorithms. PSNR and uniformity values of five segmented images obtained by TLBO-based MCET, Quantum PSO-based MCET, FF-based MCET and HBMO-based MCET algorithms are tabulated

in [Table 2](#_bookmark17) and [Table 3](#_bookmark18) respectively. [Table 4](#_bookmark19) shows the fitness val- ues of objective function obtained by TLBO-based MCET, Quantum PSO-based MCET, FF-based MCET and HBMO-based MCET algo- rithms to segment five standard test images at various levels. [Table 2](#_bookmark17), [Table3](#_bookmark18) and [Table 4](#_bookmark19) provide quantitative standard for eval- uating. Such tables show that the number of thresholds increase, the PSNR, the uniformity and the fitness value are enlarged. From [Table 1](#_bookmark16), it is found there is no significance difference in the selected thresholds at various levels for four test images, named ‘‘Pepper”, ‘‘Bird”, ‘‘Cameraman” and ‘‘Goldhill” by TLBO-based MCET algo- rithm, Quantum PSO-based MCET, FF-based MCET and HBMO- based MCET algorithms. However, there is significance difference in the selected thresholds at various levels for ‘‘Lena” standard test image by the proposed approach as compared to other algorithms.

Table 1

Multilevel threshold values obtained through TLBO-based MCET algorithm, GA-based MCET algorithm, FF-based MCET algorithm and HBMO-based MCET algorithm for five standard test images.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Image *k*  (Size) | TLBO-based MCET Algorithm | Quantum PSO-based MCET Algorithm | FF-based MCET Algorithm | HBMO-based MCET Algorithm |
| Lena 2 | 81, 140 | 53, 117 | 53, 117 | 53, 117 |
| (512 × 512) 3 | 72, 119, 165 | 45, 91, 143 | 46, 95, 150 | 46, 95, 150 |
| 4 | 68, 107, 139, 175 | 43, 76, 121, 157 | 40, 77, 114, 160 | 40, 77, 114, 160 |
| 5 | 59, 86, 115, 144, 178 | 30, 55, 92, 107, 157 | 29, 53, 84, 117, 161 | 28, 52, 83, 117, 161 |
| Pepper 2 | 52, 125 | 52, 126 | 52, 125 | 52, 125 |
| (512 × 512) 3 | 47, 107, 157 | 45, 106, 158 | 48, 107, 157 | 48, 107, 157 |
| 4 | 35, 74, 116, 162 | 30, 78, 117, 158 | 35, 75, 117, 163 | 35, 75, 117, 163 |
| 5 | 34, 71, 104, 137, 171 | 34, 72, 111, 140, 173 | 34, 72, 104, 137, 172 | 34, 71, 104, 136, 171 |
| Bird 2 | 60, 118 | 61, 119 | 61, 118 | 61, 118 |
| (204 × 204) 3 | 56, 109, 155 | 59, 114, 165 | 59, 111, 157 | 59, 111, 157 |
| 4 | 42, 79, 118, 158 | 53, 82, 114, 159 | 45, 83, 122, 160 | 45, 83, 122, 160 |
| 5 | 36, 66, 95, 129, 162 | 35, 73, 103, 135, 168 | 37, 66, 97, 132, 164 | 37, 66, 98, 132, 164 |
| Cameraman 2 | 50, 135 | 50, 138 | 50, 136 | 50, 136 |
| (256 × 256) 3 | 31, 86, 142 | 30, 86, 142 | 29, 82, 143 | 29, 82, 143 |
| 4 | 27, 75, 123, 156 | 32, 80, 117, 155 | 28, 75, 124, 157 | 28, 75, 124, 157 |
| 5 | 28, 73, 114, 144, 170 | 26, 63, 112, 141, 171 | 27, 70, 112, 144, 171 | 27, 70, 114, 144, 171 |
| Goldhill 2 | 84, 148 | 85, 149 | 85, 149 | 85, 149 |
| (512 × 512) 3 | 68, 107, 162 | 68, 106, 163 | 70, 109, 163 | 70, 109, 163 |
| 4 | 61, 93, 129, 178 | 62, 93, 138, 184 | 62, 94, 130, 179 | 62, 94, 130, 179 |
| 5 | 54, 80, 106, 137, 183 | 53, 82, 105, 133, 187 | 55, 81, 107, 138, 184 | 55, 81, 107, 138, 184 |

Bold values indicate optimal threshold values, searched by proposed algorithm, through which either PSNR or Uniformity is obtained maximum.

Table 2

PSNR values of TLBO-based MCET algorithm, GA-based MCET algorithm, FF-based MCET algorithm and HBMO-based MCET algorithm for five standard test images.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Image *k*  (Size) | TLBO-based MCET Algorithm | Quantum PSO-based MCET Algorithm | FF-based MCET Algorithm | HBMO-based MCET Algorithm |
| Lena 2 | 15.5306 | 14.8638 | 14.8638 | 14.8638 |
| (512 × 512) 3 | 17.3866 | 17.7146 | 17.6864 | 17.6864 |
| 4 | 18.7021 | 19.6546 | 19.5150 | 19.5150 |
| 5 | 20.0644 | 19.2888 | 20.1972 | 20.1558 |
| Pepper 2 | 15.1896 | 15.2337 | 15.1896 | 15.1896 |
| (512 × 512) 3 | 17.6338 | 17.4696 | 17.7093 | 17.7093 |
| 4 | 19.9153 | 19.8414 | 19.9644 | 19.9644 |
| 5 | 21.6366 | 21.6358 | 21.6751 | 21.6136 |
| Bird 2 | 16.0486 | 16.1355 | 16.0431 | 16.0431 |
| (204 × 204) 3 | 18.5123 | 17.9528 | 18.4755 | 18.4755 |
| 4 | 20.3177 | 19.7368 | 20.5288 | 20.5288 |
| 5 | 22.1644 | 22.2094 | 22.2883 | 22.2964 |
| Cameraman 2 | 15.9935 | 15.8946 | 15.9713 | 15.9713 |
| (256 × 256) 3 | 18.7556 | 18.7461 | 18.4850 | 18.4850 |
| 4 | 21.1499 | 21.3946 | 21.1308 | 21.1308 |
| 5 | 22.7206 | 22.1507 | 22.6106 | 22.5393 |
| Goldhill 2 | 14.4587 | 14.4082 | 14.4082 | 14.4082 |
| (512 × 512) 3 | 17.0286 | 17.0313 | 16.8969 | 16.8969 |
| 4 | 18.8143 | 18.6798 | 18.7570 | 18.7570 |
| 5 | 20.5004 | 20.5888 | 20.4292 | 20.4292 |

Bold value indicates maximum PNSR value.

Table 3

Uniformity of TLBO-based MCET algorithm, GA-based MCET algorithm, FF-based MCET algorithm and HBMO-based MCET algorithm for five standard test images.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Image *k*  (Size) | TLBO-based MCET Algorithm | Quantum PSO-based MCET Algorithm | FF-based MCET Algorithm | HBMO-based MCET Algorithm |
| Lena 2 | 0.9714 | 0.9555 | 0.9555 | 0.9555 |
| (512 × 512) 3 | 0.9792 | 0.9613 | 0.9632 | 0.9632 |
| 4 | 0.9831 | 0.9714 | 0.9715 | 0.9715 |
| 5 | 0.9843 | 0.9609 | 0.9686 | 0.9684 |
| Pepper 2 | 0.9692 | 0.9693 | 0.9692 | 0.9692 |
| 512 × 512) 3 | 0.9744 | 0.9740 | 0.9746 | 0.9746 |
| 4 | 0.9776 | 0.9762 | 0.9778 | 0.9778 |
| 5 | 0.9813 | 0.9814 | 0.9814 | 0.9812 |
| Bird 2 | 0.9733 | 0.9736 | 0.9734 | 0.9734 |
| (204 × 204) 3 | 0.9774 | 0.9774 | 0.9781 | 0.9781 |
| 4 | 0.9788 | 0.9784 | 0.9797 | 0.9797 |
| 5 | 0.9812 | 0.9812 | 0.9815 | 0.9816 |
| Cameraman 2 | 0.9824 | 0.9826 | 0.9825 | 0.9825 |
| (256 × 256) 3 | 0.9827 | 0.9826 | 0.9825 | 0.9825 |
| 4 | 0.9844 | 0.9840 | 0.9846 | 0.9846 |
| 5 | 0.9872 | 0.9862 | 0.9870 | 0.9871 |
| Goldhill 2 | 0.9703 | 0.9706 | 0.9706 | 0.9706 |
| (512 × 512) 3 | 0.9731 | 0.9731 | 0.9737 | 0.9737 |
| 4 | 0.9787 | 0.9784 | 0.9789 | 0.9789 |
| 5 | 0.9806 | 0.9797 | 0.9809 | 0.9809 |

Bold value indicates maximum uniformity.

Table 4

Fitness values of TLBO-based MCET algorithm, GA-based MCET algorithm, FF-based MCET algorithm and HBMO-based MCET algorithm for five standard test images.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Image *k*  (Size) | TLBO-based MCET Algorithm | Quantum PSO-based MCET Algorithm | FF-based MCET Algorithm | HBMO-based MCET Algorithm |
| Lena 2 | —1.5926e+008 | —1.5909e+008 | —1.5909e+008 | —1.5909e+008 |
| (512 × 512) 3 | —1.5943e+008 | —1.5930e+008 | —1.5930e+008 | —1.5930e+008 |
| 4 | —1.5949e+008 | —1.5944e+008 | —1.5944e+008 | —1.5944e+008 |
| 5 | —1.5953e+008 | —1.5943e+008 | —1.5947e+008 | —1.5947e+008 |
| Pepper 2 | —1.5436e+008 | —1.5436e+008 | —1.5436e+008 | —1.5436e+008 |
| (512 × 512) 3 | —1.5454e+008 | —1.5454e+008 | —1.5454e+008 | —1.5454e+008 |
| 4 | —1.5467e+008 | —1.5466e+008 | —1.5467e+008 | —1.5467e+008 |
| 5 | —1.5472e+008 | —1.5471e+008 | —1.5472e+008 | —1.5472e+008 |
| Bird 2 | —2.5601e+007 | —2.5601e+007 | —2.5601e+007 | —2.5601e+007 |
| (204 × 204) 3 | —2.5620e+007 | —2.5619e+007 | —2.5620e+007 | —2.5620e+007 |
| 4 | —2.5634e+007 | —2.5632e+007 | —2.5634e+007 | —2.5634e+007 |
| 5 | —2.5642e+007 | —2.5641e+007 | —2.5642e+007 | —2.5642e+007 |
| Cameraman 2 | —3.8580e+007 | —3.8580e+007 | —3.8580e+007 | —3.8580e+007 |
| (256 × 256) 3 | —3.8623e+007 | —3.8623e+007 | —3.8623e+007 | —3.8623e+007 |
| 4 | —3.8638e+007 | —3.8636e+007 | —3.8638e+007 | —3.8638e+007 |
| 5 | —3.8647e+007 | —3.8646e+007 | —3.8647e+007 | —3.8647e+007 |
| Goldhill 2 | —1.4144e+008 | —1.4144e+008 | —1.4144e+008 | —1.4144e+008 |
| (512 × 512) 3 | —1.4163e+008 | —1.4163e+008 | —1.4163e+008 | —1.4163e+008 |
| 4 | —1.4172e+008 | —1.4171e+008 | —1.4172e+008 | —1.4172e+008 |
| 5 | —1.4177e+008 | —1.4177e+008 | —1.4177e+008 | —1.4177e+008 |

Bold value indicates minimum cross entropy value.

In this case, the proposed approach is superior to the other meth- ods in terms of uniformity and fitness value. Form experimental results, it is possible to appear the fact that the selected thresholds of the TLBO-based MCET algorithm can effectively find the ade- quate solutions based on the minimum cross entropy criterion. Thus, we can say that the proposed method is an efficient and fea- sible method to search an optimal combination of threshold values at 2nd, 3rd, 4th and 5th levels.

[Table 5](#_bookmark26) shows the computation time and the number of itera- tions required to find an optimal combination of threshold values at 2nd, 3rd, 4th and 5th levels for the segmentation of five standard test images. From computational complexity point of view, table values indicate that the proposed algorithm has an ability to find an optimal combination of threshold values at 2nd, 3rd, 4th and

5th levels for the segmentation of standard test images in the rea- sonable amount of time and iterations.

1. Conclusion and future scope

In this paper, a new multilevel image thresholding approach based on Teacher-Learning-based Optimization, which is named TLBO-based minimum cross entropy thresholding (TLBO-based MCET) algorithm, has been presented to find multilevel optimal threshold values for segmenting gray-scale digital images. Optimal combinations of threshold values at 2nd, 3rd, 4th and 5th levels are searched by TLBO for five standard test images, named ‘‘Lena”, ‘‘Pepper”, ‘‘Bird”, ‘‘Cameraman” and ‘‘Goldhill”.

Table 5

Number of iterations and computation time of TLBO-based MCET algorithm for test images.

1. [Otsu N. A threshold selection method from gray-level histograms. IEEE Transa](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0020) [Syst Man Cybernet 1979;9(1):108–13](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0020).
2. [Pun T. Entropies thresholding, a new method. Computer Vision, Graphics](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0025) [Image Process 1981;16(2):210–39](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0025).

Image

(Size)

Lena

(512 × 512)

*k* Number of Iterations Computation Time (s)

2 47 0.9237

3 92 1.8032

1. [Kapur JN, Sahoo PK, Wong AKC. A new method for gray-level picture](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0030) [thresholding using the entropy of the histogram. Computer Vision, Graphics](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0030) [Image Process 1985;29(3):273–85](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0030).
2. [Sahoo P, Wilkins C, Yeager J. Threshold selection using Renyi’s entropy. Pattern](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0035) [Recognit 1997;30(1):71–84](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0035).

|  |  |  |  |
| --- | --- | --- | --- |
| 4 | 261 | 5.1156 | [8] [de Albuquerque MP, Esquef IA, Mello ARG. Image thresholding using tsallis-](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0040) |
| 5 | 1391 | 27.2636 | [entropy. Pattern Recognit Lett 2004;25(9):1059–65](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0040). |

1. [Shitong W, Chung FL. Note on the equivalence relationship between renyi-](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0045) [entropy and tsallis-entropy based thresholding. Pattern Recognit Lett 2005;26](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0045) [(14):2309–12](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0045).

|  |  |  |  |
| --- | --- | --- | --- |
| Pepper | 2 | 57 | 1.2027 |
| (512 × 512) | 3 | 184 | 3.8824 |
|  | 4 | 292 | 6.1612 |
|  | 5 | 859 | 18.1249 |

1. [Liu D, Jiang Z, Feng H. A novel Fuzzy classification entropy approach to image](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0050) [thresholding. Pattern Recognit Lett 2006;27(16):1968–75](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0050).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bird | 2 | 38 | 0.3717 | [11] [Cheng HD, Chen JR, Li J. Threshold Selection based on Fuzzy c-partition](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0055) |
| (204 × 204) | 3 | 244 | 2.3668. | [Entropy Approach. Pattern Recognit 1998;31(7):857–70](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0055). |
|  | 4 | 222 | 2.1534 | [12] [Benabdelkader S, Boulemden M. Recursive Algorithm based on Fuzzy 2-](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0060) |

5 668 6.4796

[partition entropy for 2-level Image thresholding. Pattern Recognit 2005;38](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0060) [(8):1289–94](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0060).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cameraman | 2 | 37 | 0.1624 | [13] Tang Y, Mu W, Zhang Y, Zhang X. A fast recursive Algorithm based on Fuzzy 2- |
| (256 × 256) | 3 | 237 | 1.0402 | partition entropy approach for threshold selection. Neurocomputing2011;74 |
|  | 4 | 232 | 0.9976 | (8):3072–78. |
|  | 5 | 1205 | 5.1815 | [14] [Tao W, Jin H, Liu L. Object segmentation using ant colony algorithm and Fuzzy](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0070) |
| Goldhill | 2 | 52 | 0.8892 | [entropy. Pattern Recognit Lett 2007;28(7):788–96](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0070).  [15] [Li CH, Lee CK. Minimum cross entropy thresholding. Pattern Recognit 1993;26](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0075) |
| (512 × 512) | 3 | 123 | 2.1033 | [(4):617–25](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0075). |
|  | 4 | 263 | 4.4973 | [16] [Nie F, Gao C, Guo Y, Gan M. Two-dimensional minimum local cross-entropy](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0080) |
|  | 5 | 1363 | 23.3073 | [thresholding based on co-occurrence matrix. Comput Electr Eng 2011;37](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0080) |
|  |  |  |  | [(5):757–67](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0080). |
|  |  |  |  | [17] [Horng MH. Multilevel minimum cross entropy threshold selection based on](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0085) |

The major contribution of the proposed approach is the applica- tion of the cross entropy based TLBO for gray-scale digital image segmentation. The proposed approach is a new variant of multi- level thresholding algorithm to segment gray-scale digital images by employing the concept of cross entropy. The proposed algo- rithm selects multilevel optimal threshold values to segment gray-scale digital images. This is done by the concept of cross entropy and framing the problem of threshold selection as an opti- mization problem. Here, the optimization problem is to minimize the cross entropy between the segmented image and the original image. One of recent optimization techniques named Teaching- Learning-based Optimization (TLBO) has been used to solve the optimization problem. The proposed approach is novel, as the con- cept of multilevel threshold selection has not been explored using cross entropy and TLBO. To examine the performance of the pro- posed approach, five different standard digital test images have been segmented through selected threshold values at 2nd, 3rd, 4th and 5th levels by the proposed approach. The simulation and experimental results show the proposed algorithm has an ability to search multilevel optimal threshold values to segment digital images. For evaluating the effectiveness of the proposed algorithm, two measures, namely PSNR, uniformity, are used. PSNR and uni- formity are used to measure the quality of the thresholded images. From the experimental results on various types of images, it is observed that the proposed method produces the better quality thresholded images than the compared methods. Thus, the pro- posed method is an efficient method to search multilevel optimal threshold values for segmenting digital images.

The proposed approach has great potential future in the field of image segmentation. The work is under further progress to seg- ment medical images like mammograms, CT or MR images.

References

1. [Dougherty G. Digital image processing for medical applications. New Delhi,](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0005) [India: Cambridge University Press; 2010](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0005).
2. [Quintanilla-Dominguez J, Ojeda-Magana B, Cortina-Januchs MG, Ruelas R,](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0010) [Vega-Corona A, Andina D. Image segmentation by Fuzzy and possibilistic](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0010) [clustering algorithms for the identification of microcalcifications. Scientia](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0010) [Iranica 2011;18(3):580–9](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0010).
3. [Castleman KR. Digital image processing. New Delhi, India: Pearson Education](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0015) [Inc.; 2012](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0015).

[the honey bee mating optimization. Expert Syst Appl 2010;37(6):4580–92](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0085).

1. [Tang K, Yuan X, Sun T, Yang J, Gao S. An improved scheme for minimum cross](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0090) [entropy threshold selection based on genetic algorithm. Knowledge Syst](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0090) [2011;24(8):1131–8](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0090).
2. [Horng MH, Liou RJ. Multilevel minimum cross entropy threshold selection](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0095) [based on the firefly algorithm. Expert Syst Appl 2011;38(12):14805–11](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0095).
3. [Tao WB, Tian JW, Liu J. Image segmentation by three-level thresholding based](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0100) [on maximum fuzzy entropy and genetic algorithm. Pattern Recognit Lett](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0100) [2003;24(16):3069–78](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0100).
4. Singh Y, Khadelwal S. Genetic algorithm based optimization of threshold parameters in fingerprint quality estimation. In: 3rd IEEE international conference on emerging trends in engineering and technology (ICETET). New York: IEEE Press; 2010. p. 11–4.
5. [Wei JH, Kai Y. Study of improved immune genetic algorithm for threshold](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0110) [image segmentation based on Fuzzy maximum entropy. In: 10th IEEE](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0110) [international conference on computer, mechatronics, control and electronic](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0110) [engineering (CMCE), vol. 4. New York: IEEE Press; 2010. p. 52–5](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0110).
6. Ye Z, Zheng Z, Yu X, Ning X. Automatic threshold selection based on ant colony optimization algorithm. In: IEEE international conference on neural networks and brain (ICNN&B’05). New York: IEEE Press, vol. 2; 2005. p. 728–32
7. [Kim SS, Byeon JH, Yu H, Liu H. Biogeography-based optimization for optimal](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0120) [job scheduling in cloud computing. Appl Math Comput 2014;247(15):266–80](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0120).
8. Gupta S, Bhuchar K, Sandhu PS. Implementing color image segmentation using biogeography-based optimization. In: International conference on software and computer applications, vol. 9. Singapore: IACSIT Press; 2011. p. 79–86.
9. Sathya PD, Kayalvizhi R. Image segmentation using minimum cross entropy and bacterial foraging optimization algorithm. In: IEEE international conference on emerging trends in electrical and computer technology (ICETECT). New York: IEEE Press; 2011. p. 500–06.
10. [Sathya PD, Kayalvizhi R. Optimal multilevel thresholding using bacterial](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0135) [foraging algorithm. Expert Syst Appl 2011;38(12):15549–64](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0135).
11. [Rashedi E, Hezamabadi-pour H, Saryazdi S. GSA: a gravitational search](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0140) [algorithm. Informat Sci 2009;179(13):2232–48](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0140).
12. [Rajabioun R. Cuckoo optimization algorithm. Appl Soft Comput 2011;11](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0145) [(8):5508–18](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0145).
13. [Agarwal C, Mishra A, Sharma A. A novel gray-scale image watermarking using](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0150) [hybrid Fuzzy – BPN architecture. Egyptian Inform J 2015;16(1):83–102](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0150).
14. [Agarwal C, Mishra A, Sharma A. Gray-scale image watermarking using GA-BPN](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0155) [hybrid network. J Visual Commun Image Represent 2013;24(7):1135–46](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0155).
15. [Rao RV, Savsani VJ, Vakharia DP. Teaching-learning-based optimization: a](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0160) [novel method for constrained mechanical design optimization problems.](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0160) [Computer-Aided Des 2011;43(3):303–15](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0160).
16. [Rao RV, Savsani VJ, Vakharia DP. Teaching-learning-based optimization: a](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0165) [optimization method for continuous non-linear large scale problems. Informat](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0165) [Sci 2012;183(1):1–15](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0165).
17. [Crepinsek M, Liu SH, Mernik L. A note on teaching-learning-based](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0170) [optimization algorithm. Informat Sci 2012;212(1):79–93](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0170).
18. [Waghmare G. Comments on a note on teaching-learning-based optimization](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0175) [algorithm. Informat Sci 2013;229(20):159–69](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0175).
19. [Rao RV, Patel V. An improved teaching-learning-based optimization algorithm](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0180) [for unconstrained optimization problems. Scientia Iranica 2013;20(3):710–20](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0180).
20. Singh R, Verma HK. Teaching-learning-based optimization algorithm for parameter identification in the design of IIR filters. J Inst Eng India Ser B 2014;94(4):285–94.
21. [Rao RV, More KC. Optimal design of the heat pipe using TLBO (teaching-](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0190) [learning-based optimization) algorithm. Energy 2015;80:535–44](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0190).
22. [Sahu BK, Pati S, Mohanty PK, Panda S. Teaching-learning based optimization](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0195) [algorithm based Fuzzy-PID controller for automatic generation control of](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0195) [multi-area power system. Appl Soft Comput 2015;27:240–9](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0195).
23. [Xu Y, Wang L, Wang SY, Liu M. An effective teaching–learning-based](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0200) [optimization algorithm for the flexible job-shop scheduling problem with](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0200) [Fuzzy processing time. Neurocomputing 2015;148:260–8](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0200).
24. Priyambada S, Mohanty PK, Sahu BK. Automatic voltage regulator using TLBO algorithm optimized PID controller. In: IEEE 9th international conference on industrial and information systems (ICIIS). New York: IEEE Press; 2014. p. 1–6.
25. [Zhao Y, Fang Z, Wang K, Pang H. Multilevel minimum cross entropy threshold](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0210) [selection based on quantum particle swarm optimization. In: IEEE 8th ACIS](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0210) [international conference on software engineering, artificial intelligence,](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0210) [networking, and parallel/distributed computing (SNPD 2007). New](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0210) [York: IEEE Press; 2007. p. 65–9](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0210).
26. Wang KL, Wang HB, Yu LX, Ma XY, Xue YS. Towards teaching-learning- based optimization algorithm for dealing with real-parameter optimization problems. In: 2nd international conference on computer science and electronics engineering (ICCSEE 2013). Paris: Atlantis Press; 2013. p. 607–09.
27. [Kullback S. Information theory and statistics. New York: Dover; 1968](http://refhub.elsevier.com/S1110-8665(17)30273-6/h0220).