Arabian Journal for Science and Engineering (2019) 44:3005–3020
https://doi.org/10.1007/s13369-018-3400-2

RESEARCH ARTICLE - COMPUTER ENGINEERING AND COMPUTER SCIENCE

An Improved PSO-Based Multilevel Image Segmentation Technique Using Minimum Cross-Entropy Thresholding

Rupak Chakraborty[1] · Rama Sushil[1] · M. L. Garg[1]

Received: 27 October 2017 / Accepted: 7 June 2018 / Published online: 18 June 2018

Abstract

Entropy-based thresholding techniques are quite popular and effective for image segmentation. Among different entropy-based techniques, minimum cross-entropy thresholding (MCET) has received wide attention in the field of image segmentation. Considering the high time complexity of MCET technique for multilevel thresholding, recursive approach to reducing its computational cost is highly desired. To reduce the complexity, further optimization techniques can be applied to find optimal multilevel threshold values. In this paper, a novel improved particle swarm optimization (IPSO)-based multilevel thresholding algorithm is proposed to search the near-optimal MCET thresholds. The general PSO algorithm often suffers from premature convergence problem which has been addressed in the IPSO by decomposing a high-dimensional swarm into several one-dimensional swarms, and then premature convergence is removed from each one-dimensional swarm. The proposed technique is applied to the set of grayscale images, and the experimental results infer that it produces better MCET optimal threshold values at a higher and faster convergence rate. The qualitative and quantitative results are compared with existing optimization techniques like modified artificial bee colony, Cuckoo search, Firefly, particle swarm optimization, and genetic algorithm. It has been observed that the proposed technique performs better in terms of producing better fitness value, less CPU time as quantitative measurements, and effective misclassification error, peak signal-to-noise ratio, feature similarity index measurement, complex wavelet structural similarity index measurement values as qualitative measurements compared to other considered state-of-the-art methods.

Keywords Multilevel thresholding · MCET · IPSO · ME · PSNR · FSIM · CW-SSIM

1 Introduction

Image segmentation is a fundamental step for meaningful analyzing and interpretation of an image. It is considered as a mandatory preprocessing step for extracting objects from its backgrounds in many computer vision oriented applications. The goal of segmentation carries the concept that the image regions should contain similar types of homogeneous pixels with respect to some common features like color, intensity, and texture so that it can be analyzed properly. Thresholding is one of the popular methods to achieve this goal, which can easily discriminate the objects from its background pixels. The main aim is to find the efficient threshold value for binary thresholding and multiple threshold values for multiple thresholding so that separation is carried out properly.

In bilevel thresholding, the total image region is subdivided into two homogeneous regions based on the histogram and edge detection. This technique generates a binary image where all pixels carry higher gray values than a threshold level, are put into a single class whereas pixels having less values than a threshold level are put into another class. Some useful histogram, edge detection, minimum variance, interactive pixel classification, and entropy-based bilevel thresholding techniques are surveyed in the literature [1–5]. Further global entropy-based techniques got attention for bilevel image segmentation in the form of Kapur et al. [6], Tsalli [7] and Wong and Sahoo [8] entropy which was applied for successful separation of objects from its backgrounds. Minimum cross-entropy-based segmentation was proposed by Li et al. [9,10] where thresholding value was obtained by

✉ Rupak Chakraborty
rupak.jis@gmail.com

Rama Sushil
hod.it@dituniversity.edu.in

M. L. Garg
dr.ml.garg@dituniversity.edu.in

[1] DIT University, Dehradun, India

minimizing the cross-entropy for better segmentation. Pal [11] proposed the segmentation technique by also minimizing the cross-entropy with the mixture of poison distributions. The effectiveness of MCET over standard entropy can be surveyed [12,13]. MCET-based threshold selection algorithm works fast and provides more accurate segmentation results. However, all these mentioned techniques are quite effective for bilevel thresholding, but not effective enough for complex segmentation problems which need multiple threshold values for better separation between objects and backgrounds.

To continue with the literature review process, Perez and Gonzalez [14]; Tao et al. [15]; proposed multilevel thresholding techniques which were found very useful for dividing the images into multiple levels to produce effective segmentation results. To remove the high-level complexity of multilevel thresholding technique, Arora et al. [16] proposed a famous mean and standard deviation-based multilevel thresholding approach for better separation of objects and backgrounds. Multilevel thresholding techniques with entropy maximization or minimization applied with any metaheuristic technique holds recent research interest.

Section 2 presents a literature survey related to metaheuristic approaches, used to find multiple threshold values for multilevel image segmentation. Section 3 describes the recursive minimum cross-entropy thresholding (MCET) approach. Section 4 presents the proposed algorithm, called improved PSO (IPSO). The simulated results of the proposed algorithm along with other considered optimized methods are presented in Sect. 5. The limitations of the proposed method with future work are discussed in Sect. 6. Finally, Sect. 7 draws a conclusion of the paper.

2 Related Works with Metaheuristic Approaches

In computer science, a metaheuristic is a high-level procedure to generate a search algorithm which provides a good solution to an optimization problem with limited computation capacity. A sample set of large solution sets are generated by these techniques. As these approaches can provide better and easy solutions for a wide variety of complex problems, so it has been flourishing since almost last decade. These techniques are helpful to find optimal solutions of NP-hard problems because users do not need to have deep knowledge regarding initial population selections of problems and as well as derivatives. Some popular derivative-free optimizing techniques like genetic algorithm (GA) [17,18] and firefly (FF) algorithm with fuzzy entropy [19] are surveyed in the literature. Artificial bee colony (ABC) [20–22], cuckoo search [23], a novel gravitational search algorithm [24] based approaches to find optimal solutions of metaheuristic problems are found in the literature as well.

Particle swarm optimization (PSO) is another stochastic global optimization algorithm of relatively recent category [25–28], inspired by real-life swarm behaviors and is well used for multilevel optimization. To choose a better optimization technique for using in our research, comparative literature survey continues between PSO with GA [29], PSO with ABC [30], PSO with cuckoo search and DE [31], and PSO with firefly [32], to reach to the conclusion that which one to opt among these evolutionary algorithms. It is concluded from the literature that PSO arguably works as better metaheuristic approach compared to others to find optimal solutions.

In standard PSO algorithm, several particles (candidate solutions) fly in a multidimensional search space, and fitness of each of them is evaluated in each iteration. Here the new velocity at the start of a new iteration of a particle is calculated based on its previous velocity, that is the distance between its present position and the best position found so far known as **Pbest**. Similarly, the distance between particle's present position and its best distance found for the entire swarm is known as **Gbest**. However, the main drawback of this basic PSO is that it may get stuck in suboptimal solution regions and the problem may increase in high-dimensional problems, as its success depends on the combination of global exploration and local exploitation features during the optimization process. So this stochastic approach was also needed improvement. Literature reveals that multi-swarm-based PSO was proposed by Mukhopadhyay and Banerjee [33] and Zheng et al. [34] for finding better optimized values. So considering the improvement of standard PSO, our research is focused to propose an improved particle swarm optimization technique. Now to finalize minimum cross-entropy (MCET) as objective function in our research, our survey continued and found that Sarkar et al. [35]; Yin [36]; Olivia et al. [37]; Pare et al. [38]; Horng and Liou [39] proposed recursive minimum cross-entropy-based multilevel segmentation with differential evolution (DE), PSO, and cuckoo search (CS), and firefly (FF) algorithm, respectively. Encouraged by the effective results from above literatures, MCET has been chosen as objective function among different entropies. Next subsection gists proposed innovative improved PSO (IPSO)-based approach with recursive minimum image cross-entropy.

2.1 Author's Contribution

The proposed improved particle swarm optimization (IPSO) approach will be applied to remove the "curse of dimensionality," and to overcome the problem of "premature convergence," for finding the best solution from the populations.

- Composite high-dimensional swarm is broken into several one-dimensional swarms in search space.
- Each swarm communicates with each other by exchanging information to find composite fitness of an entire system.
- Next, each particle's new velocity is updated based on the Pbest information of any particle within a swarm to avoid premature convergence by the logic discussed later.
- Recursive MCET is applied as the objective function to find multiple thresholds.
- The outcomes of the proposed algorithm (IPSO) are then compared with existing PSO, GA [18], modified artificial bee colony (MABC) [40], cuckoo search (CS) [38], firefly (FF) [39] algorithms.
- Quantitative and qualitative comparisons are carried out with above-mentioned optimization techniques in terms of misclassification error (ME), peak signal-to-noise ratio (PSNR) [19], feature similarity index measurement (FSIM) [41], and complex wavelet structural similarity index measurement (CW-SSIM) [42] to show the better working of our algorithm.

3 Minimum Cross-Entropy Thresholding

3.1 Minimum Cross-Entropy

Let $F = f_1, f_2, \ldots, f_n$ and $G = g_1, g_2, \ldots, g_n$ are two probability distributions on the same set where n represents the size of test set. The cross-entropy C between F and G is then calculated by Kullback [43] like this

$$C(F, G) = \sum_{i=1}^{n} f_i \log \frac{f_i}{g_i} \tag{1}$$

The minimum cross-entropy thresholding (MCET) algorithm is used to select the threshold by minimizing the cross-entropy between original and thresholded images. Let I is the original image and $h(i)$, where $i = 1, 2, \ldots, L$ be the corresponding histogram and L is the number of gray levels. Then thresholded image, I_t where t specifies threshold value, can be constructed by

$$I_t(x, y) = \begin{cases} \mu(1, t), & I(x, y) < t \\ \mu(t, L+1), & I(x, y) \geq t \end{cases} \tag{2}$$

where $\mu(x, y) = \sum_{i=x}^{y-1} ih(i) / \sum_{i=x}^{y-1} h(i)$. The cross-entropy is then calculated by according to Li and Lee [9] in the following manner:

$$\begin{aligned} C(t) = & -\sum_{i=1}^{t-1} ih(i) \log\left(\frac{i}{\mu(1, t)}\right) \\ & + \sum_{i=t}^{L} ih(i) \log\left(\frac{i}{\mu(t, L+1)}\right) \end{aligned} \tag{3}$$

Then above calculated cross-entropy is minimized for finding the optimal threshold value t^* like that

$$t^* = \arg \min_t C(t) \tag{4}$$

The computational complexity for finding t^* is $O(nL^2)$. However, for finding multilevel n thresholding values, the computational complexity may jump to $O(nL^{n+1})$ which will be very high [44].

3.2 Recursive MCET

According to [36,45], the MCET objective function of Eq. (3) can be redefined as:

$$\begin{aligned} C(t) = & -\sum_{i=1}^{L} ih(i) \log(i) - \sum_{i=1}^{t-1} ih(i) \log(\mu(1, t)) \\ & -\sum_{i=t}^{L} ih(i) \log(\mu(t, L+1)) \end{aligned} \tag{5}$$

As the first term is constant for a given image, so the objective function can be redefined as:

$$\begin{aligned} \sigma(t) = & -\sum_{i=1}^{t-1} ih(i) \log(\mu(1, t)) \\ & -\sum_{i=t}^{L} ih(i) \log(\mu(t, L+1)) \\ = & -\sum_{i=1}^{t-1} ih(i) \log\left(\frac{\sum_{i=1}^{t-1} ih(i)}{\sum_{i=1}^{t-1} h(i)}\right) \\ & -\sum_{i=t}^{L} ih(i) \log\left(\frac{\sum_{i=t}^{L} ih(i)}{\sum_{i=t}^{L} h(i)}\right) \\ = & -p^1(1, t) \log\left(\frac{p^1(1, t)}{p^0(1, t)}\right) \\ & -p^1(t, L+1) \log\left(\frac{p^1(t, L+1)}{p^0(t, L+1)}\right) \end{aligned} \tag{6}$$

where $p^0(x, y) = \sum_{i=x}^{y-1} h(i)$ and $p^1(x, y) = \sum_{i=x}^{y-1} ih(i)$ are zero and first moment on partial range of the image histogram, respectively. This recursive technique can easily be applied to find multilevel threshold values. Let n is the number of selected thresholds denoted by $t_1, t_2, \ldots, t_n$.

```
BEGIN
Create and initialize N particles of D-dimensional swarm with their X, V and Pbest and the Gbest of the
population
      REPEAT
            FOR i=1 to N
                  IF $f(X_i) < f(\mathbf{Pbest_i})$
                        $\mathbf{Pbest_i} = X_i$
                  IF $f(X_i) < f(\mathbf{Gbest})$
                        $\mathbf{Gbest} = X_i$
                  ENDIF
                  ENDIF
            FOR d=1 to D
                  Calculate new velocity of particle:
                  $V_i^d \longleftarrow wV_i^d + c_f * rand1_i^d * (Pbest_i^d - X_i^d) + c_g * rand2_i^d * (Gbest^d - X_i^d)$
                  $V_i^d \longleftarrow sign(V_i^d) * min\ (\ abs(V_i^d), V_{max}^d)$
                  Calculate new position of the particle:
                  $X_i^d \longleftarrow X_i^d + V_i^d$
            ENDFOR
          ENDFOR
        UNTIL termination condition meets
      END
```

Fig. 1 Basic PSO algorithm

Two dummy thresholds $t_0 \equiv 0$, $t_{n+1} \equiv L + 1$ along with $t_0 < t_1 < \cdots < t_n < t_{n+1}$ can be added for the convenience of illustration of the problem. The modified objective function for multilevel thresholding will look like now:

$$\sigma(t_1, t_2, \ldots, t_n) = -\sum_{i=1}^{n+1} p^1(t_{i-1}, t_i)\ \log\left(\frac{p^1(t_{i-1}, t_i)}{p^0(t_{i-1}), t_i}\right) \tag{7}$$

From Eq. (7), it is found that complexity is $O(nL^n)$, which is slightly lesser than the previously calculated complexity $O(nL^{n+1})$, but still, it is computationally expensive if $n \geq 3$. So to reduce the complexity further, we have proposed a novel improved particle swarm optimization (IPSO)-based approach for minimizing $\sigma(t)$. Here user input of n will be chosen in that manner so that undersegmentation or oversegmentation problem can be avoided to produce effective results.

4 Overview of Particle Swarm Optimization (PSO) Algorithm

In PSO algorithm, particles (multiple solutions) are put into a multidimensional spaces, and the fitness of each particle is evaluated in each iteration. Here the new velocity (subsequently new position of a particle), at the start of a new iteration, is calculated on the basis of its present velocity. The distance between particle's present position and its best position found so far called as Pbest, whereas the distance between particle's present position and the position of the best particle among the entire swarm is called the Gbest. Say, it is required to solve a D-dimensional optimization problem by minimizing the objective function $f(x)$ given as

$$\min\ \ f(x), x = [x_1, x_2, \ldots, x_D] \tag{8}$$

where D specifies the number of parameters to be optimized. Here in basic PSO algorithm, a swarm of N "particles" is flown in a D-dimensional search space randomly in the quest for the optimum solution of a fitness function. Figure 1 shows the working mechanism of PSO for optimum fitness. Let $X_i = (X_i^1, X_i^2, \ldots, X_i^D)$ represents the position, $V_i = (V_i^1, V_i^2, \ldots, V_i^D)$ the velocity and $Pbest_i = (Pbest_i^1, Pbest_i^2, \ldots, Pbest_i^D)$ the best previous position with the lowest fitness value found so far, for each ith particle. So far the best position found by the swarm in the search space is denoted by $Gbest = (Gbest1, Gbest^2, \ldots, Gbest^D)$. The velocity V_i^d of each particle in dth dimension for each iteration is determined by the additive influence of V_i^d found in the previous iteration (called "momentum" component), and individual weighting of its distance from $Pbest_i^d$ (known as a cognitive component) and its distance from $Gbest^d$ (known as social component). c_f and c_g are used as the acceleration coefficients to the extent of the stochastic weighting for the cognitive and social components individually. The "momentum" component carries the inertia weight (w) for global exploitation during initial stages and local exploitation during later stages of the optimization process. Initially, w contains high value and gradually decreases according to the following equation:

$$w = w_{\text{initial}} - \theta_{\text{IW}} \times it \tag{9}$$

where w is the inertia weight in the current iteration, w_{initial} is the inertia weight at the starting condition, it is the present iteration count, θ_{IW} is the slope of inertia weight variation.

4.1 Drawback of PSO

However, this basic PSO suffers from two basic problems like "curse of dimensionality" and a tendency of premature convergence which is quite apparent in several optimization algorithms. In this problem, particle's motion may get stuck because of the limited computing resources which may lead to PSO to get prone to local optima convergence. As a result of that, some particles will gain low fitness values although their dimension values lie very close to the global optimal solution. Cooperation operation can remove this problem by acquiring information from "best" dimensions and preventing useful information from being unnecessarily discarded. In the proposed improved technique, "curse of dimensionality" problem is resolved by dividing a composite high-dimensional swarm into several one-dimensional swarms, which cooperate with each other dynamically by exchanging the information to form composite fitness of an entire system. As high-dimensional space is divided into several one-dimensional spaces, so cooperation between each one dimension helps to save the most useful information to accelerate convergence. Further to overcome the problem of premature convergence, caused by traditional PSO algorithm, the new velocity of each particle can be modified based on the Pbest information of any particle within the swarm, chosen according to a specific algorithm. In our next subsection, we propose our improved particle swarm optimization algorithm (called IPSO) to overcome both the problems, "curse of dimensionality" and the tendency of premature convergence at a time. Figure 2 describes the proposed algorithm.

4.2 Proposed Improved Particle Swarm Optimization (IPSO)

In this proposed approach, a D-dimensional problem is decomposed into D-one-dimensional swarms where each swarm consists of N particles. A final global solution is evaluated by aggregating all the Gbest solutions achieved from each individual swarm. The fitness evaluation of particles is done based on the introduction of a new parameter called context vector (denoted by **CV**) (defined in Fig. 2) that will be used to exchange information among all the individual swarms. The size of context vector (CV) for a D-dimensional problem should be equal to D-dimensional itself. Here, when a jth swarm is active, then the "context vector" is formed by remaining $(D - 1)$ swarm's Gbest particles whose values were constant during the working of the jth swarm. Then the jth row of the "context vector" is filled by each particle of jth swarm one by one. Each such "context vector" is calculated for finding its composite fitness. So the Pbest value $(P_j.Pbest_i)$ for the ith particle and the Gbest solution $(P_j.Gbest)$ for jth swarm are determined in that manner so that they not only depend on the performance of the jth swarm alone. Now the velocity and position of each particle in the jth swarm (denoted by $P_j.X_i$ and $P_j.V_i$) are updated based on these Pbest and Gbest solutions. Lastly the jth entry of the "context vector" is filled by the newly calculated $P_j.Gbest$ and this process is continued for each jth swarm, till all the relevant context vectors are filled one by one. So, in brief, it can be concluded that the total search space is divided into D subspaces for D individual swarms and these swarms communicate with each other through their corresponding "context vectors" to determine their individual $P_j.Pbest_i$ and $P_j.Gbest$. The final context vector is calculated by concatenating all the evaluated $P_j.Gbest$ determined across all the swarms.

However, this above-mentioned technique may still be get trapped in suboptimal locations within search space, where all individual solutions may fail to produce a better solution every time. To overcome this problem, we propose some modifications to the above-mentioned approach to determine new velocities and positions of each particle in each individual swarm in subspaces. So, the created problem of stagnation because of premature convergence is seriously taken care off, by permitting each particle to adjust its velocity (so, the position also), based on the Pbest information of any par-

```
BEGIN
DEFINE CV (j,y) ≡ ($P_1$.Gbest, $P_2$.Gbest,….. $P_{j-1}$.Gbest, y, $P_{j+1}$.Gbest, …….. $P_D$.Gbest)
Create and initialize N particles for each of the D one-dimensional swarms with their corresponding X.V
and Pbest and the Gbest of the population
    REPEAT:
        FOR j=1 to D
            IF ($j^{th}$ swarm got stagnated for last few iterations)
                Find $N_{replace}$ "best" particles and $N_{replace}$ worst particles in the $j^{th}$ swarm on
                the basis of their fitness values
                Destroy those $N_{replace}$ worst particles and replace them by clones of $N_{replace}$
                "best" particles
            ENDIF
            FOR i=1 to N
                IF f(CV(j,$P_j.X_i$)) < f(CV(j,$P_j.Pbest_i$))
                $P_j.Pbest_i$ = $P_j.X_i$
                IF f(CV(j,$P_j.X_i$)) < f(CV (j, $P_j$.Gbest ) )
                $P_j$.Gbest = $P_j.X_i$
                ENDIF
              ENDIF
              Calculate new velocity of particle:
            $P_j.V_i$ ⟵ w * $P_j.V_i$ + $c_i^{'}$ * $rand_i$ * ( $P_j$ . $Pbest_{f_i}$ - $P_j$. $X_i$ )
                Where $f_i$ will be selected according to Step 1- Step 4
            $P_j.V_i$ ⟵ sign ($P_j.V_i$) * min (abs ($P_j.V_i$), $P_j$ .$V_{max}$ )
            Calculate final position of particle:
            $P_j.X_i$ ⟵ $P_j.X_i$ + $P_j$ .$V_i$
            END FOR
        END FOR
      UNTIL termination condition meets
    END
```

Fig. 2 Proposed improved particle swarm optimization (IPSO) algorithm

ticle within the swarm. This helps to discourage premature convergence within the swarm strongly.

Now to select the particle, whose Pbest can be used to find the new velocity of any given particle within a given one-dimensional swarm is shown in Fig. 3. Hence, the new velocity update relation for each particle in a given swarm is demonstrated as

$$P_j.V_i \leftarrow w * P_j.V_i + c_i^{'} * rand_i * (P_j.Pbest_{f_i} - P_j.X_i) \quad (10)$$

where f_i decides which particle's Pbest should be followed by this ith particle. It is defined in Steps 1–4 in Fig. 3.

4.3 "Best" Particles Cloning and "Worst" Particle Destruction

This module is employed to each one-dimensional local swarm for further improvement of this optimization strategy. As the local swarms may get be stagnated for the last few iterations because of almost insignificant improvements in the fitness values of the Gbest particle for the given swarm, then particles are needed to sort according to their Pbest values. Assume N_{replace} are the number of particles identified based on their lowest fitness values, corresponding to their Pbest positions (called "Worst" particles), to be replaced. Similarly, N_{replace} number of particles are also being identified in

```
Step 1 : Generate a random number in the range [0,1]. If it is higher than one design parameter (denoted
         as "selection probability"), then particle will learn from its own Pbest, otherwise it will learn from
         another particle's Pbest and will follow the selection procedure, shown in steps 2-4.
Step-2 : Putting the current particle aside, choose two particles randomly from the population.
Step-3 : Now compare these two particles' Pbests and choose the particle with higher fitness.
Step-4 : The particle found with better fitness is chosen and its Pbest is used to evaluate the velocity of
         the current particle under consideration.
```

Fig. 3 Algorithm for "Best" particle selection

```
Step 1 : Compute $p^0(x,y)$ and $p^1(x,y)$ for $1 < x < y < L$ using the recursive programming technique.
Step-2 : Generate N particles according to the particle formulation using Eq.11.
Step-3 : Generate random number in the range [0,1].
Step-4 : Repeat for a given maximum number of iterations.
Step-5 : Evaluate the fitness of each particle using the objective function mentioned in Eq.7.
Step-6:  Now follw the steps 2-4 (mentioned in the algorithm in Fig.3) to determine the particle's velocity
         using Eq.10.
Step-7 : New position of the particle can be calculated as shown in the last step in IPSO algorithm in Fig.2.
```

Fig. 4 IPSO-based algorithm for MCET problem

this swarm, based on their highest fitness values, corresponding to their Pbest positions (called as "Best" particles). These particles will be used to replace the worst particles identified so far. Now a superior swarm will be formed to obtain better optimization performance in the search space. At the end of this approach, it can be found that each swarm contains N_{replace} number of positions where two particles are present simultaneously. So these particles are "cloned" to each other and this process is repeated for each one-dimensional swarm. We have considered 50% of the particles ($N_{\text{replace}} = N/2$) are declared as "Worst" whereas remaining 50% are designated as "Best" particles.

4.4 MCET-Based IPSO Working

It is formulated that each particle can work as a candidate solution to the multilevel MCET problem. For an n-threshold MCET problem, particle formulation can be like that

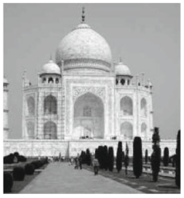
$$P = (t_1, t_2, \ldots, t_n),\quad \text{subject to } 1 < t_1 < t_2 < \cdots < t_n < L \tag{11}$$

Here, n parameters in the particle representation can work as a candidate solution corresponding to the multiple thresholds for the MCET problem. N-particles are generated randomly according to Eq. 11 are considered as an initial swarm. The values of zero and first moments ($p^0(x, y)$ and $p^1(x, y)$) for $1 < x < y < L$ are precomputed using the recursive programming technique for n thresholds by the objective function as described in Sect. 3.2. In the MCET approach with IPSO, the particle deriving the minimal value is considered to be the best, so the objective function in Eq. 7 is minimized. The IPSO-based algorithm for MCET problem is further illustrated in Fig. 4.

5 Result Analysis

Simulations of the proposed scheme is evaluated in MATLAB R2015a in a workstation with Intel $core^{TM}$ i3 3.2 GHz processor. The results are statistically compared with some

Table 1 Original grayscale images

Bridge	Dog	Dyno
Flower	Tajmahal	Building
		

popular recently developed techniques like modified artificial bee colony algorithm proposed by Bhandari et al. [40]; an efficient cuckoo search algorithm proposed by Pare et al. [38]; MCET-based firefly algorithm proposed by Horng and Liou [39]; genetic algorithm proposed by Pare et al. [18]; and basic PSO algorithm surveyed in the literature as well. Our approach strictly follows the steps mentioned in the Fig. 4. All the tested algorithms run for 100 independent times where each run was carried out for the $D \times 1000$ (D denoted the dimension of search space) number of fitness evaluations. Segmentation levels are set from 4 to 6 to avoid undersegmentation problem. As quantitative measurement, fitness evaluations (FEs) and computation time (t) are compared with the above-mentioned stochastic metaheuristic approaches and peak signal-to-noise ratio (PSNR), uniformity measure (ME), feature similarity index measurement (FSIM), and complex wavelet structural similarity index measurement (CW-SSIM) are evaluated as qualitative comparison.

5.1 Quantitative Measurement

The proposed technique is tested on a set of grayscale and medical images. Images are resized to 225 × 225 for effective segmentation. Original grayscale images are shown in Table 1. It is observed from Table 2 that the proposed algorithm produces better values in terms of computation time (t), objective fitness values (f_{obj}), and standard deviation (f_{std}) for levels 4 and 6. As cross-entropy is minimized in our approach, so the value of the objective function with the proposed optimizer produces fewer values compared to other stochastic optimization algorithms for all the levels of segmented images. Further to check the stability of the proposed algorithm, standard deviation values are calculated for 100 iterations. As less standard (f_{std}) deviation values imply the more stability of the algorithm, so it is calculated for different algorithms and results infer that the proposed approach overcomes challenges of others. As high-level computation time of recursive algorithm throws a challenge to the researchers, so attention is given to reduce time complexities. Further, it can be noted in Table 2, that computation time (t) which is

Table 2 Comparison of computational time (t), objective value (f_{obj}) and standard deviation (f_{std}) between IPSO, MABC, CS, FF, GA, and PSO using MCET

Im	4-level						6-level					
	IPSO	MABC	CS	FF	GA	PSO	IPSO	MABC	CS	FF	GA	PSO
1												
t	3.391	3.498	3.647	3.782	3.891	3.992	5.063	5.128	5.198	5.222	5.275	5.315
f_{obj}	0.168	0.177	0.183	0.189	0.198	0.201	0.066	0.073	0.076	0.078	0.080	0.082
f_{std}	0.021	0.027	0.037	0.042	0.051	0.064	0.062	0.068	0.082	0.088	0.095	0.101
2												
t	3.271	3.397	3.542	3.682	3.857	3.971	5.271	5.332	5.402	5.452	5.534	5.624
f_{obj}	0.156	0.168	0.176	0.185	0.192	0.198	0.065	0.072	0.075	0.079	0.089	0.097
f_{std}	0.055	0.067	0.072	0.081	0.087	0.093	0.072	0.078	0.087	0.096	0.104	0.109
3												
t	3.412	3.557	3.689	3.831	3.923	3.998	5.372	5.457	5.479	5.531	5.612	5.708
f_{obj}	0.162	0.169	0.176	0.186	0.192	0.199	0.068	0.082	0.085	0.091	0.096	0.099
f_{std}	0.043	0.049	0.066	0.077	0.086	0.092	0.071	0.084	0.093	0.097	0.104	0.107
4												
t	3.181	3.321	3.478	3.588	3.699	3.802	5.141	5.203	5.288	5.358	5.447	5.534
f_{obj}	0.169	0.178	0.185	0.191	0.196	0.202	0.060	0.072	0.075	0.083	0.089	0.093
f_{std}	0.044	0.051	0.061	0.070	0.075	0.082	0.072	0.080	0.087	0.091	0.098	0.107
5												
t	3.211	3.326	3.482	3.567	3.702	3.810	5.411	5.476	5.512	5.547	5.601	5.678
f_{obj}	0.172	0.189	0.194	0.199	0.206	0.213	0.064	0.078	0.085	0.089	0.095	0.099
f_{std}	0.045	0.058	0.065	0.074	0.081	0.088	0.074	0.085	0.091	0.096	0.105	0.110
6												
t	3.331	3.451	3.556	3.678	3.811	3.926	5.411	5.476	5.512	5.547	5.617	5.721
f_{obj}	0.170	0.179	0.186	0.191	0.196	0.199	0.065	0.073	0.078	0.086	0.089	0.095
f_{std}	0.055	0.063	0.072	0.084	0.089	0.097	0.094	0.099	0.105	0.110	0.116	0.120

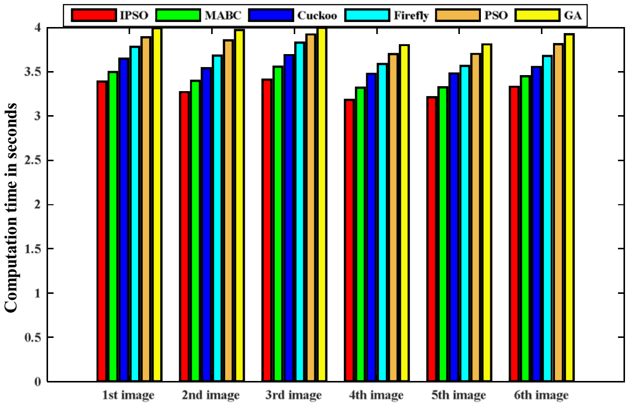


Fig. 5 Comparison of computation time of 4th level segmented images using different evolutionary algorithms

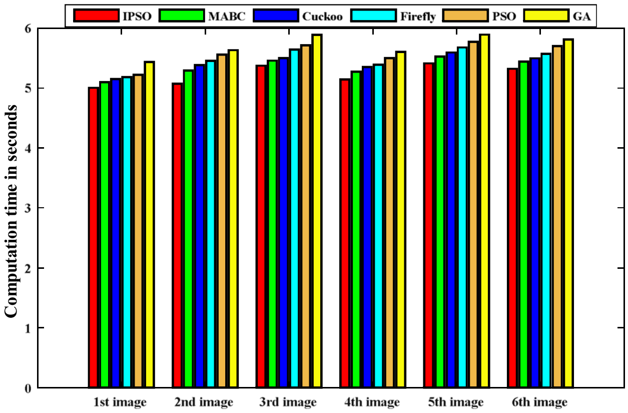


Fig. 6 Comparison of computation time of 6th level segmented images using different evolutionary algorithms

obtained for our algorithm sweeps other techniques comprehensively. The results of our technique are followed by the MABC, CS, FF, GA, and PSO in terms of objective value, computation time and standard deviation in the mentioned order. Further Figs. 5 and 6 clearly suggest the fastness of our approach for fourth and sixth levels of above-mentioned six experimental images. The value of modified artificial bee colony (MABC) approach almost touches our curve, whereas CS, FF, GA, and PSO maintain this order.

Table 3 represents the sixth-level segmented images and their corresponding convergence plots. It is quite clearly visible that the convergence values are almost touching to 0 in our proposed scheme for 100 iterations for all the considered images where values of other techniques struggling to reach zero value. The stability of the convergence values is also considered when it is found that, for last 50 iterations generated convergence curves are almost straight for our technique and remaining are taking more time than this. Some medical images (shown in Table 4) are also tested for better visualization and effectiveness of the algorithm. Table 5 generates the sixth-level comparative segmented images of evolutionary algorithms. The results of our technique are quite better and effective in terms of visual quality as well as convergence values. Colors are applied to these images for better visual comparison of segmented images. Figures 7 and 8 draw the comparison of convergence curves for fourth and sixth levels of segmented medical images respectively. It can be observed that the starting value of the fourth level plot of the proposed approach has been started from 0.511 and ended with 0.13, whereas for the sixth level, it is 0.921 and 0.052 which suggest the better working of the technique for the increment of levels.

5.2 Qualitative Results

In this subsection, researchers have shown the segmented results of level 6 by applying the proposed IPSO technique with other techniques and segmented images generated by the proposed algorithm are found visually more clear and effective. Tables 3 and 5 display the 6th level segmented results of gray and medical images for all the stochastic approaches namely IPSO, MABC, CS, FF, GA, and PSO. Table 6 lists threshold values for level 4 and 6 of grayscale images for all the algorithms. Results suggest that the threshold values obtained by the proposed technique are quite better compared to others for Bridge, Dog, Dyno, Flower, Taj and Building images. Further parametric measurements of segmented images obtained by different algorithms are given in next subsection in the form of ME for medical images, PSNR, FSIM, and CW-SSIM for grayscale images to show the better and effective segmentation results of the proposed algorithm.

5.3 Misclassification Error

Misclassification error is used to measure the uniformity in segmented images to compare the performances of optimization techniques [19]. It is measured by using the following formula:

$$\mathrm{ME} = 1 - 2 * T * \frac{\sum_{i=0}^{T} \sum_{j \in R_i} (\mathrm{In}_j - \mu_i)^2}{n * (\mathrm{In}_{\max} - \mathrm{In}_{\min})^2} \tag{12}$$

where T is the number of thresholds which are used to segment the image, R_i is the ith segmented region, In_j is the intensity level of pixels in that particular segmented area, μ_i is the mean of an ith thresholded region of an image, n denotes the total number of pixels in the image and $\mathrm{In}_{\min}$ and $\mathrm{In}_{\max}$ are minimum and maximum intensities of image respectively. Misclassification error values lie between 0 to 1 and fewer values indicate better performance in terms of visibility factor. It is found from Table 7 that the proposed approach generates fewer values for all the levels (4 to 6) of medical images compared to other optimization techniques.

Table 3 Comparative results of segmented images and convergence plot between different algorithms of sixth level

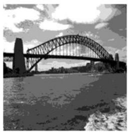
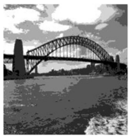
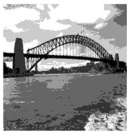
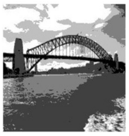
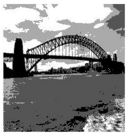
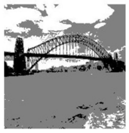
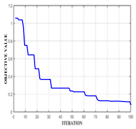
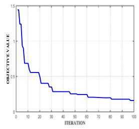
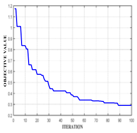
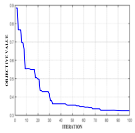
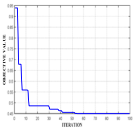
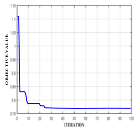
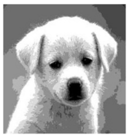
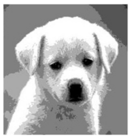
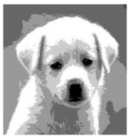
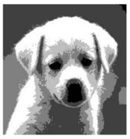
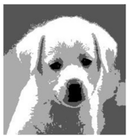
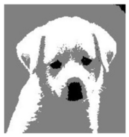
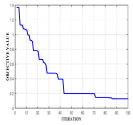
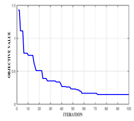
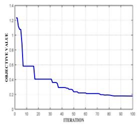
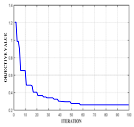
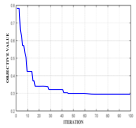
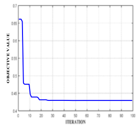
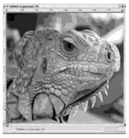
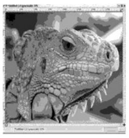
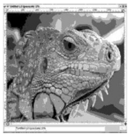
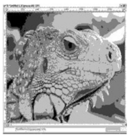
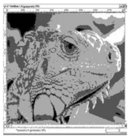
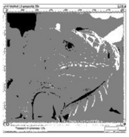
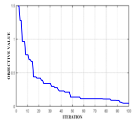
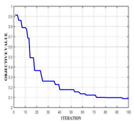
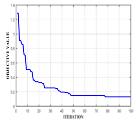
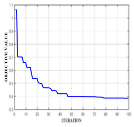
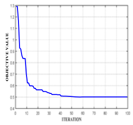
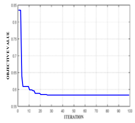
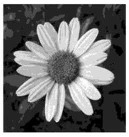
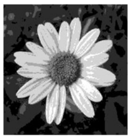
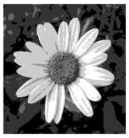
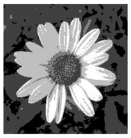
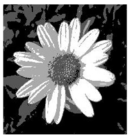
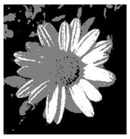
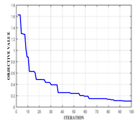
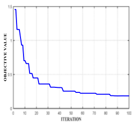
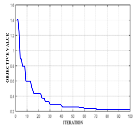
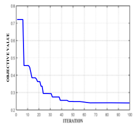
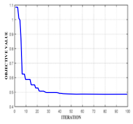
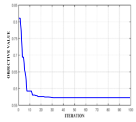
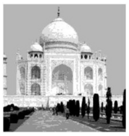
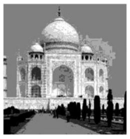
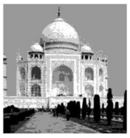
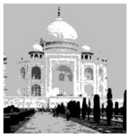
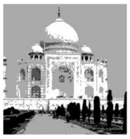
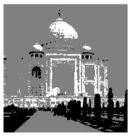
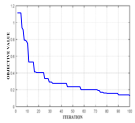
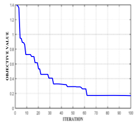
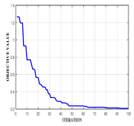
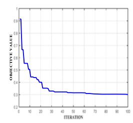
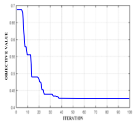
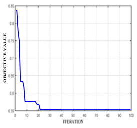
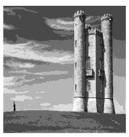
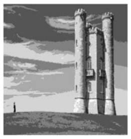
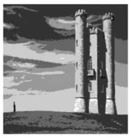
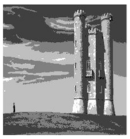
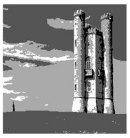
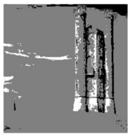
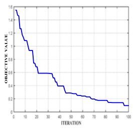
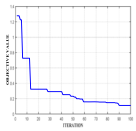
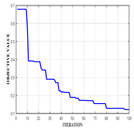
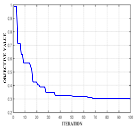
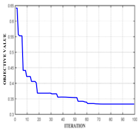
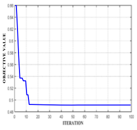
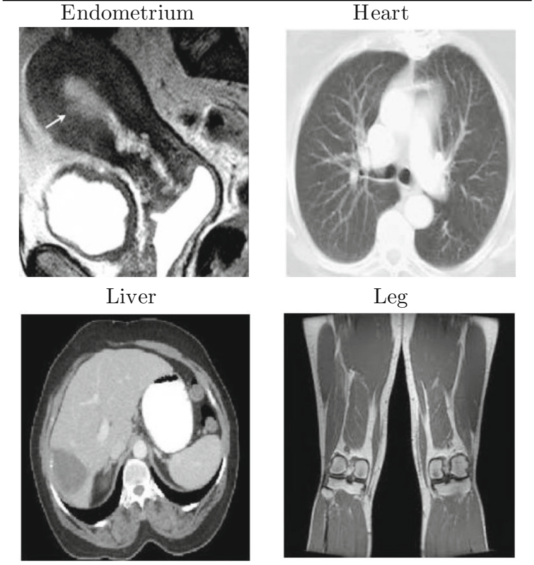
IPSO	MABC	CS	FF	GA	PSO
					
					
					
					
					
					
					
					
					
					
					
					

Table 4 Original Medical Images



Values are inversely proportional to the increment of threshold levels which suggest better working of the algorithm.

5.4 Peak Signal-to-Noise Ratio (PSNR)

To measure the dissimilarity or visual difference between original and segmented images, PSNR values are calculated in decibel (dB) unit. Images with high PSNR values suggest the better quality of segmentation. It can be calculated by the following equation:

$$\mathrm{PSNR} = 10 \times \log_{10}\left(\frac{255^2}{\mathrm{RMSE}}\right)(\mathrm{dB}) \tag{13}$$

where 255 is the maximum gray value and RMSE stands for root-mean-square error, defined in the below equation:

$$\mathrm{RMSE} = \frac{1}{R \times C}\sum_{i}^{R}\sum_{j}^{C}\{I(i,j) - I'(i,j)\}^2 \tag{14}$$

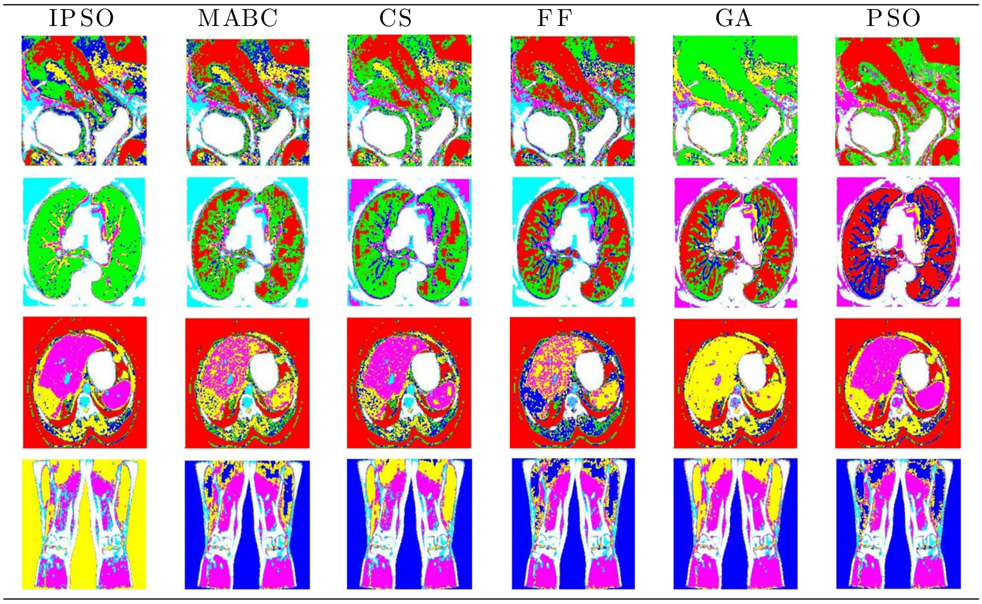
where I and I' are the original and thresholded image, respectively. R and C are the rows and column size whereas i, j represent the pixel values with size 225 × 225 of original and segmented images, respectively. It is observed from the Table 8 that PSNR values of the proposed IPSO-MCET-based approach are quite high compared to other optimization techniques for level 4 to 6. Further, it is noticed that MABC, CS, FF, GA, and PSO closely follow our technique.

5.5 Feature Similarity Index Measurement (FSIM)

It is used to evaluate the visual similarity between original and segmented images [41]. Higher FSIM values indicate the better quality of segmented images. Say, f_1 and f_2 are original and segmented images, then phase congruency PC_1 and PC_2 as the PC map values and G_1 and G_2 as gradient magnitude (GM) values can be extracted from f_1 and f_2. Further FSIM measurement can be done based on these PC_1, PC_2, G_1, G_2 values. Firstly, local similarity map is computed, next similarity map is pooled into a single similarity score. The similarity measure is calculated as follows:

$$S_{\mathrm{PC}}(x) = \frac{2\mathrm{PC}_1(x)\cdot\mathrm{PC}_2(x) + M_1}{\mathrm{PC}_1^2(x) + \mathrm{PC}_2^2(x) + M_1} \tag{15}$$

Table 5 Comparison of six-level segmented different medical images by different optimizers



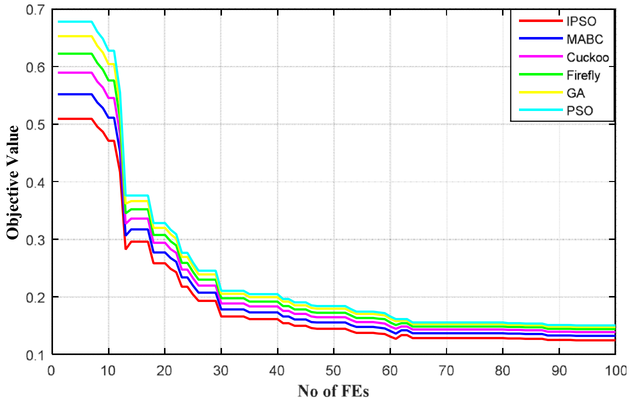


Fig. 7 Comparison of convergence plot of IPSO, MABC, CS, FF, GA, and PSO for Lv = 4 of medical image data set

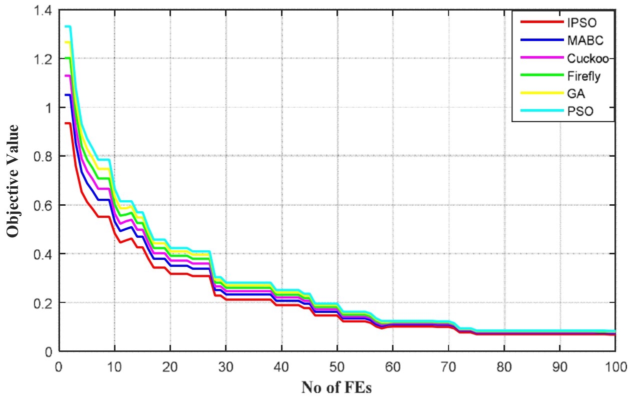


Fig. 8 Comparison of convergence plot of IPSO, MABC, CS, FF, GA, and PSO for Lv = 6 of medical image data set

where M_1 is a positive constant used to increase the stability of S_{PC}. Then GM values $G_1(x)$ and $G_2(x)$ are compared and similarity measure is defined as follows:

$$S_G(x) = \frac{2G_1(x)\cdot G_2(x) + M_2}{G_1^2(x) + G_2^2(x) + M_2} \tag{16}$$

where M_2 value depends on the dynamic range of GM values. Now $S_{\mathrm{PC}}(x)$ and $S_G(x)$ are combined to find the similarity $S_L(x)$ for $f_1(x)$ and $f_2(x)$.

$$S_L(x) = [S_{\mathrm{PC}}(x)]^{\alpha}\cdot[S_G(x)]^{\beta} \tag{17}$$

where α, β are used for adjusting the relative importance of PC, GM features. Then the maximum of $\mathrm{PC}(x)$ (denoted by $\mathrm{PC}_{\max}(x)$) is found which gives the weighted importance of $S_L(x)$ in the overall similarity between f_1 and f_2. Finally, FSIM between f_1 and f_2 is defined in the following equation:

Table 6 Optimal threshold values obtained by different evolutionary algorithms with MCET approach

Images	Optimal threshold values		
	Algo	$Th = 4$	$Th = 6$
Bridge	IPSO	44, 82, 187, 220	2, 11, 68, 159, 191, 233
	MABC	43, 91, 184, 237	5, 18, 77, 173, 191, 239
	CS	47, 97, 195, 251	10, 29, 82, 172, 195, 240
	FF	52, 96, 197, 250	20, 28, 92, 170, 199, 242
	GA	55, 99, 190, 251	28, 32, 96, 167, 197, 247
	PSO	50, 99, 203, 252	21, 38, 97, 172, 201, 248
Dog	IPSO	35, 71, 146, 245	4, 41, 78, 123, 148, 223
	MABC	35, 77, 155, 252	5, 48, 89, 130, 153, 231
	CS	40, 82, 162, 253	8, 58, 97, 132, 154, 233
	FF	36, 91, 170, 251	7, 65, 95, 133, 155, 236
	GA	42, 97, 185, 250	10, 68, 98, 140, 159, 243
	PSO	43, 99, 187, 248	21, 65, 99, 150, 171, 245
Dyno	IPSO	41, 71, 84, 181	11, 46, 124, 137, 174, 242
	MABC	49, 72, 84, 191	15, 47, 129, 146, 176, 244
	CS	54, 80, 96, 205	17, 59, 135, 155, 177, 247
	FF	51, 86, 105, 207	20, 67, 133, 164, 178, 249
	GA	53, 88, 107, 209	24, 69, 136, 166, 179, 250
	PSO	52, 80, 109, 208	19, 72, 138, 169, 175, 253
Flower	IPSO	11, 45, 81, 137	21, 56, 114, 161, 209, 227
	MABC	17, 47, 81, 139	22, 59, 118, 160, 215, 231
	CS	20, 48, 84, 149	29, 62, 125, 164, 216, 232
	FF	21, 51, 90, 155	32, 63, 126, 165, 217, 233
	GA	23, 54, 92, 156	33, 65, 128, 167, 219, 232
	PSO	25, 57, 95, 157	30, 67, 132, 169, 221, 235
Taj	IPSO	47, 85, 111, 158	15, 34, 57, 83, 118, 181
	MABC	40, 94, 119, 170	15, 37, 58, 94, 125, 181
	CS	42, 102, 123, 171	17, 47, 59, 95, 124, 184
	FF	38, 107, 131, 180	19, 48, 61, 95, 125, 185
	GA	42, 109, 135, 182	20, 52, 63, 98, 127, 188
	PSO	43, 110, 137, 184	23, 54, 65, 99, 129, 190
Building	IPSO	45, 83, 110, 157	12, 32, 55, 81, 116, 183
	MABC	41, 93, 118, 172	15, 36, 56, 93, 123, 180
	CS	47, 105, 127, 170	19, 45, 55, 92, 122, 182
	FF	35, 106, 130, 183	29, 49, 63, 85, 124, 187
	GA	37, 108, 132, 184	32, 51, 66, 88, 127, 189
	PSO	39, 105, 137, 187	41, 53, 67, 89, 121, 196

$$\mathrm{FSIM} = \frac{\sum_{x\in\sigma} S_L(x)\cdot \mathrm{PC}_{\max}(x)}{\sum_{x\in\sigma} \mathrm{PC}_{\max}(x)} \tag{18}$$

where σ represents the whole image spatial domain. Our proposed technique produces better FSIM values for all the levels of segmented images tested noted in Table 9. It is observed that sixth-level values of our technique are almost closer to 1, whereas values generated by remaining approaches are quite behind from 1.

Table 7 Comparison of ME values of segmented medical images using different optimization algorithms with MCET

Images	Algorithms	ME		
		$Th = 4$	$Th = 5$	$Th = 6$
Med-1	IPSO	0.815	0.726	0.641
	MABC	0.832	0.744	0.655
	CS	0.871	0.765	0.669
	FF	0.899	0.771	0.682
	GA	0.907	0.786	0.689
	PSO	0.923	0.797	0.694
Med-2	IPSO	0.818	0.726	0.635
	MABC	0.832	0.734	0.653
	CS	0.841	0.748	0.671
	FF	0.859	0.757	0.682
	GA	0.867	0.765	0.693
	PSO	0.879	0.777	0.699
Med-3	IPSO	0.828	0.735	0.626
	MABC	0.842	0.754	0.643
	CS	0.851	0.768	0.655
	FF	0.859	0.776	0.662
	GA	0.869	0.783	0.672
	PSO	0.877	0.791	0.682
Med-4	IPSO	0.824	0.753	0.625
	MABC	0.832	0.764	0.633
	CS	0.841	0.779	0.649
	FF	0.859	0.785	0.661
	GA	0.869	0.792	0.668
	PSO	0.878	0.799	0.682

5.6 Complex Wavelet Structural Similarity Index Measurement (CW-SSIM)

Structural similarity index measurement (SSIM) is extended to CW-SSIM to the complex wavelet domain [42]. CW-SSIM is preferred for its insensitivity to "non-structured" geometric image distortions. In the complex wavelet transform domain, let us assume that $c_x = \{c_{x,i} \mid i = 1, \ldots, N\}$ $c_y = \{c_{y,i} \mid i = 1, \ldots, N\}$ are two sets of coefficients extracted from the same spatial location in the same wavelet subbands of two images I, I'. Then the CW-SSIM can be defined as

$$S(c_x, c_y) = \frac{2 \mid \sum_{i=1}^{N} c_{x,i} c_{y,i}^{*} \mid + K}{\sum_{i=1}^{N} \mid c_{x,i} \mid^2 + \sum_{i=1}^{N} \mid c_{y,i} \mid^2 + K} \tag{19}$$

where c^* and K are complex conjugates of c and positive constant, respectively. Higher CW-SSIM values indicate better segmentation results. Table 10 displays the high values of the proposed approach which outperform other techniques.

Table 8 Comparison of PSNR values of different optimization algorithms with MCET

Images	Algorithms	PSNR		
		$Th = 4$	$Th = 5$	$Th = 6$
1	IPSO	35.908	36.876	37.765
	MABC	34.312	34.814	35.203
	CS	33.071	33.908	34.439
	FF	32.809	33.556	34.012
	GA	32.527	33.126	33.716
	PSO	32.118	32.607	33.014
2	IPSO	35.918	35.876	36.965
	MABC	34.342	34.914	35.433
	CS	33.431	33.908	34.481
	FF	33.109	33.877	34.162
	GA	32.921	33.527	33.983
	PSO	32.809	33.267	33.772
3	IPSO	34.718	34.876	35.721
	MABC	33.942	34.614	34.743
	CS	33.031	33.308	33.985
	FF	32.709	32.976	33.512
	GA	32.429	32.683	32.832
	PSO	32.209	32.517	32.792
4	IPSO	35.824	36.673	37.305
	MABC	34.442	34.614	34.943
	CS	33.731	33.909	34.389
	FF	33.139	33.675	34.001
	GA	32.809	33.136	33.768
	PSO	32.729	33.015	33.242
5	IPSO	33.913	34.356	34.985
	MABC	33.040	33.424	33.947
	CS	32.531	32.808	33.008
	FF	32.158	32.775	32.912
	GA	32.007	32.635	32.813
	PSO	31.918	32.325	32.704
6	IPSO	36.118	36.746	37.504
	MABC	35.342	35.614	35.843
	CS	34.731	34.978	35.289
	FF	33.889	34.276	34.912
	GA	33.639	33.906	34.305
	PSO	33.405	33.778	34.112

6 Future Work and Discussion

To search the limitation of the proposed work, it is further compared with two different latest image segmentation techniques [46,47]. Gao et al. [46] considered ground truth image sets in their approach first. They claimed their segmented results as best, based on the F-score values which were the combination of the true-positive rate (TPR) and false-positive rate (FPR) to identify defect areas in segmented images. One

Table 9 Comparison of FSIM values of different optimization algorithms with MCET

Images	Algorithms	FSIM		
		$Th = 4$	$Th = 5$	$Th = 6$
1	IPSO	0.728	0.876	0.965
	MABC	0.689	0.714	0.883
	CS	0.631	0.708	0.839
	FF	0.609	0.696	0.812
	GA	0.588	0.633	0.792
	PSO	0.574	0.601	0.756
2	IPSO	0.698	0.746	0.835
	MABC	0.589	0.624	0.783
	CS	0.531	0.609	0.719
	FF	0.509	0.596	0.711
	GA	0.482	0.537	0.688
	PSO	0.467	0.521	0.654
3	IPSO	0.718	0.838	0.911
	MABC	0.677	0.710	0.866
	CS	0.631	0.708	0.839
	FF	0.599	0.677	0.792
	GA	0.577	0.640	0.752
	PSO	0.546	0.621	0.735
4	IPSO	0.804	0.903	0.999
	MABC	0.688	0.774	0.903
	CS	0.671	0.768	0.889
	FF	0.659	0.766	0.822
	GA	0.634	0.736	0.801
	PSO	0.612	0.724	0.798
5	IPSO	0.737	0.856	0.985
	MABC	0.625	0.684	0.783
	CS	0.611	0.678	0.779
	FF	0.609	0.666	0.762
	GA	0.577	0.646	0.752
	PSO	0.545	0.635	0.740
6	IPSO	0.778	0.846	0.904
	MABC	0.699	0.772	0.893
	CS	0.653	0.770	0.879
	FF	0.650	0.696	0.812
	GA	0.632	0.677	0.801
	PSO	0.613	0.656	0.787

Table 10 Comparison of CW-SSIM values of different optimization algorithms with MCET

Images	Algorithms	CW-SSIM		
		$Th = 4$	$Th = 5$	$Th = 6$
1	IPSO	0.838	0.977	1.205
	MABC	0.779	0.884	1.083
	CS	0.731	0.808	0.939
	FF	0.708	0.796	0.915
	GA	0.678	0.774	0.890
	PSO	0.654	0.765	0.870
2	IPSO	0.797	0.848	0.955
	MABC	0.699	0.724	0.883
	CS	0.641	0.709	0.839
	FF	0.609	0.696	0.819
	GA	0.587	0.667	0.796
	PSO	0.576	0.654	0.785
3	IPSO	0.908	0.998	1.211
	MABC	0.787	0.810	0.967
	CS	0.731	0.800	0.929
	FF	0.699	0.777	0.882
	GA	0.678	0.744	0.866
	PSO	0.656	0.724	0.852
4	IPSO	0.907	0.993	1.199
	MABC	0.778	0.874	0.993
	CS	0.771	0.868	0.988
	FF	0.759	0.856	0.912
	GA	0.735	0.847	0.888
	PSO	0.712	0.829	0.867
5	IPSO	0.887	0.926	0.995
	MABC	0.715	0.784	0.893
	CS	0.711	0.768	0.879
	FF	0.709	0.766	0.867
	GA	0.672	0.702	0.839
	PSO	0.667	0.689	0.801
6	IPSO	0.878	0.946	1.004
	MABC	0.799	0.871	0.983
	CS	0.753	0.875	0.949
	FF	0.755	0.796	0.917
	GA	0.732	0.784	0.887
	PSO	0.725	0.776	0.870

major drawback of the proposed algorithm is that it does not consider defective areas in ground truth images for segmentation which can be given attention later. In terms of quality measurement of segmented images, standard metrics like PSNR, FSIM are used in the proposed method rather using F-measure values which may be taken care of in the future. But the proposed approach outperformed this algorithm in terms of computational cost as improved PSO (IPSO) works faster than GA, and MCET produces outcome quickly as compared to traditional entropy-based approach. Rafiee et al. [47] focused to segment the low depth-of-field images (DOF) based on the k-cluster blocks to extract region of interest (ROI) from the segmented blocks of images. The calculated F-measure values achieved by this algorithm for camera captured low depth images suggested the best extraction of ROI. But applied optimizer to this approach increased the computational complexities a lot. So IPSO-based optimization approach may further be applied for ROI extraction from DOF images as the future extension of our work. Sulistyo et al. [48,49] proposed feature extraction based image segmentation technique to analyze nitrogen status from colored plant images. Genetic algorithm (GA) has been applied there as global optimization technique to normalize plant images. The proposed IPSO can be tested instead of GA there to compare the convergence rate of the algorithm. Alkassar et al. [50] proposed an enhanced feature extraction approach for eye images obtainable from UBIRIS and UTIRIS databases. Proposed skin-based sclera segmentation algorithm reduces computational complexities for low-resolution images. Like this, we can also apply the proposed optimization algorithm to extract features from RGB, HSV, CIE color spaces, and further segment the images from known datasets as a future work.

7 Conclusion

A minimum cross-entropy-based approach with improved particle swarm optimization (IPSO-MCET) is proposed here for multilevel thresholding of grayscale images with the desired output. To find the effective multiple thresholds, cross-entropy is minimized with this improved technique. In that improved PSO, high-dimensional swarms are decomposed into several one-dimensional swarms which exchange information among themselves to generate the overall fitness value. This technique is well organized to remove “curse of dimensionality” problem as well as to discourage premature convergence. This optimization strategy is further enhanced by replacing the worst particles with the best particles. Outcomes are compared in terms of objective values, computation time, and standard deviation with other recent popular stochastic approaches like MABC, CS, FF, GA, and PSO. It is found that segmented images generate better parametric measurements as well as effective threshold values. In short, it can be concluded that the proposed approach outperformed other optimization algorithms for multilevel segmentation with respect to all the quantitative and qualitative fields.

References

1. Arifin, A.Z.; Asano, A.: Image segmentation by histogram thresholding using hierarchical cluster analysis. Pattern Recogn. Lett. **27**(13), 1515–1521 (2006)
2. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. **9**(1), 62–66 (1979)
3. Revol, C.; Jourlin, M.: A new minimum variance region growing algorithm for image segmentation. Pattern Recogn. Lett. **18**(3), 249–258 (1997)
4. Sezgin, M.; et al.: Survey over image thresholding techniques and quantitative performance evaluation. J. Electron. Imaging **13**(1), 146–168 (2004)
5. Weszka, J.S.: A survey of threshold selection techniques. Comput. Graph. Image Process. **7**(2), 259–265 (1978)
6. Kapur, J.N.; Sahoo, P.K.; Wong, A.K.: A new method for gray-level picture thresholding using the entropy of the histogram. Comput. Vis. Graph. Image Process. **29**(3), 273–285 (1985)
7. Du, J.: Property of Tsallis entropy and principle of entropy increase. ArXiv preprint arXiv:0802.3424 (2008)
8. Wong, A.K.; Sahoo, P.K.: A gray-level threshold selection method based on maximum entropy principle. IEEE Trans. Syst. Man Cybern. **19**(4), 866–871 (1989)
9. Li, C.H.; Lee, C.: Minimum cross entropy thresholding. Pattern Recogn. **26**(4), 617–625 (1993)
10. Li, C.; Tam, P.K.S.: An iterative algorithm for minimum cross entropy thresholding. Pattern Recogn. Lett. **19**(8), 771–776 (1998)
11. Pal, N.R.: On minimum cross-entropy thresholding. Pattern Recogn. **29**(4), 575–580 (1996)
12. Al-Ajlan, A.; El-Zaart, A.: Image segmentation using minimum cross-entropy thresholding. In: IEEE International Conference on Systems, Man and Cybernetics, 2009. SMC 2009, pp. 1776–1781. IEEE (2009)
13. Sathya, P.; Kayalvizhi, R.: Image segmentation using minimum cross entropy and bacterial foraging optimization algorithm. In: 2011 International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT), pp. 500–506. IEEE (2011)
14. Perez, A.; Gonzalez, R.C.: An iterative thresholding algorithm for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **6**, 742–751 (1987)
15. Tao, W.; Jin, H.; Liu, L.: Object segmentation using ant colony optimization algorithm and fuzzy entropy. Pattern Recogn. Lett. **28**(7), 788–796 (2007)
16. Arora, S.; Acharya, J.; Verma, A.; Panigrahi, P.K.: Multilevel thresholding for image segmentation through a fast statistical recursive algorithm. Pattern Recogn. Lett. **29**(2), 119–125 (2008)
17. Cao, L.; Bao, P.; Shi, Z.: The strongest schema learning GA and its application to multilevel thresholding. Image Vis. Comput. **26**(5), 716–724 (2008)
18. Pare, S.; Bhandari, A.K.; Kumar, A.; Singh, G.K.; Khare, S.: Satellite image segmentation based on different objective functions using genetic algorithm: a comparative study. In: 2015 IEEE International Conference on Digital Signal Processing (DSP), pp. 730–734. IEEE (2015)
19. Naidu, M.; Kumar, P.R.; Chiranjeevi, K.: Shannon and fuzzy entropy based evolutionary image thresholding for image segmentation. Alex. Eng. J. (2017)
20. Horng, M.H.: Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation. Expert Syst. Appl. **38**(11), 13785–13791 (2011)
21. Karaboga, D.; Gorkemli, B.; Ozturk, C.; Karaboga, N.: A comprehensive survey: artificial bee colony (ABC) algorithm and applications. Artif. Intell. Rev. **42**(1), 21–57 (2014)
22. Ma, M.; Liang, J.; Guo, M.; Fan, Y.; Yin, Y.: Sar image segmentation based on artificial bee colony algorithm. Appl. Soft Comput. **11**(8), 5205–5214 (2011)
23. Suresh, S.; Lal, S.: An efficient cuckoo search algorithm based multilevel thresholding for segmentation of satellite images using different objective functions. Expert Syst. Appl. **58**, 184–209 (2016)
24. Chao, Y.; Dai, M.; Chen, K.; Chen, P.; Zhang, Z.: A novel gravitational search algorithm for multilevel image segmentation and its application on semiconductor packages vision inspection. Optik Int. J. Light Electron Opt. **127**(14), 5770–5782 (2016)

25. Chander, A.; Chatterjee, A.; Siarry, P.: A new social and momentum component adaptive pso algorithm for image segmentation. Expert Syst. Appl. **38**(5), 4998–5004 (2011)
26. Gao, H.; Xu, W.; Sun, J.; Tang, Y.: Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm. IEEE Trans. Instrum. Meas. **59**(4), 934–946 (2010)
27. Önüt, S.; Tuzkaya, U.R.; Doğaç, B.: A particle swarm optimization algorithm for the multiple-level warehouse layout design problem. Comput. Ind. Eng. **54**(4), 783–799 (2008)
28. Sathya, P.; Kayalvizhi, R.: Pso-based tsallis thresholding selection procedure for image segmentation. Int. J. Comput. Appl. **5**(4), 39–46 (2010)
29. Ye, Z.; Ye, Y.; Yin, H.: Qualitative and quantitative study of gas and PSO based evolutionary intelligence for multilevel thresholding. In: 2017 10th International Symposium on Advanced Topics in Electrical Engineering (ATEE), pp. 812–817. IEEE (2017)
30. Akay, B.: A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding. Appl. Soft Comput. **13**(6), 3066–3091 (2013)
31. Civicioglu, P.; Besdok, E.: A conceptual comparison of the cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. Artif. Intell. Rev. **39**(4), 315–346 (2013)
32. Pal, S.K.; Rai, C.; Singh, A.P.: Comparative study of firefly algorithm and particle swarm optimization for noisy non-linear optimization problems. Int. J. Intell. Syst. Appl. **4**(10), 50 (2012)
33. Mukhopadhyay, S.; Banerjee, S.: Global optimization of an optical chaotic system by chaotic multi swarm particle swarm optimization. Expert Syst. Appl. **39**(1), 917–924 (2012)
34. Zheng, H.; Jie, J.; Hou, B.; Fei, Z.: A multi-swarm particle swarm optimization algorithm for tracking multiple targets. In: 2014 IEEE 9th Conference on Industrial Electronics and Applications (ICIEA), pp. 1662–1665. IEEE (2014)
35. Sarkar, S.; Das, S.; Chaudhuri, S.S.: A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution. Pattern Recogn. Lett. **54**, 27–35 (2015)
36. Yin, P.Y.: Multilevel minimum cross entropy threshold selection based on particle swarm optimization. Appl. Math. Comput. **184**(2), 503–513 (2007)
37. Oliva, D.; Hinojosa, S.; Osuna-Enciso, V.; Cuevas, E.; Pérez-Cisneros, M.; Sanchez-Ante, G.: Image segmentation by minimum cross entropy using evolutionary methods. Soft Comput. 1–20 (2017)
38. Pare, S.; Kumar, A.; Bajaj, V.; Singh, G.: An efficient method for multilevel color image thresholding using cuckoo search algorithm based on minimum cross entropy. Appl. Soft Comput. **61**, 570–592 (2017)
39. Horng, M.H.; Liou, R.J.: Multilevel minimum cross entropy threshold selection based on the firefly algorithm. Expert Syst. Appl. **38**(12), 14805–14811 (2011)
40. Bhandari, A.K.; Kumar, A.; Singh, G.K.: Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using kapurs, otsu and tsallis functions. Expert Syst. Appl. **42**(3), 1573–1601 (2015)
41. Zhang, L.; Zhang, L.; Mou, X.; Zhang, D.: FSIM: a feature similarity index for image quality assessment. IEEE Trans. Image Process. **20**(8), 2378–2386 (2011)
42. Sampat, M.P.; Wang, Z.; Gupta, S.; Bovik, A.C.; Markey, M.K.: Complex wavelet structural similarity: a new image similarity index. IEEE Trans. Image Process. **18**(11), 2385–2401 (2009)
43. Kullback, S.: Information Theory and Statistics. Courier Corporation, Chelmsford (1997)
44. Tang, K.; Yuan, X.; Sun, T.; Yang, J.; Gao, S.: An improved scheme for minimum cross entropy threshold selection based on genetic algorithm. Knowl. Based Syst. **24**(8), 1131–1138 (2011)
45. Hammouche, K.; Diaf, M.; Siarry, P.: A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation. Comput. Vis. Image Underst. **109**(2), 163–175 (2008)
46. Gao, B.; Li, X.; Woo, W.L.; yun Tian, G.: Physics-based image segmentation using first order statistical properties and genetic algorithm for inductive thermography imaging. IEEE Trans. Image Process. **27**(5), 2160–2175 (2018)
47. Rafiee, G.; Dlay, S.S.; Woo, W.L.: Region-of-interest extraction in low depth of field images using ensemble clustering and difference of Gaussian approaches. Pattern Recogn. **46**(10), 2685–2699 (2013)
48. Sulistyo, S.B.; Woo, W.; Dlay, S.: Ensemble neural networks and image analysis for on-site estimation of nitrogen content in plants. In: Proceedings of SAI Intelligent Systems Conference, pp. 103–118. Springer (2016)
49. Sulistyo, S.; Woo, W.L.; Dlay, S.; Gao, B.: Building a globally optimized computational intelligent image processing algorithm for on-site nitrogen status analysis in plants. IEEE Intell. Syst. (2018)
50. Alkassar, S.; Woo, W.L.; Dlay, S.S.; Chambers, J.A.: Enhanced segmentation and complex-sclera features for human recognition with unconstrained visible-wavelength imaging. In: 2016 International Conference on Biometrics (ICB), pp. 1–8. IEEE (2016)