International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.203-213
ISSN 2356-5608

An efficient multithresholding method for image segmentation based on PSO

Fayçal HAMDAOUI
Electrical Department
CSR Research Unit of EµE Laboratory
Faculty of Sciences of Monastir (FSM) – University of Monastir, Tunisia
faycel_hamdaoui@yahoo.fr

Anis SAKLY
Electrical Department
Industrial Systems Study and Renewable Energy (ESIER), National Engineering School of Monastir (ENIM), University of Monastir, Tunisia
sakly_anis@yahoo.fr

Abdellatif MTIBAA
Electrical Department
CSR Research Unit of EµE Laboratory
National School of Engineering of Monastir – Av Ibn ElJazzar 5019, University of Monastir, Tunisia
abdellatif.mtibaa@enim.rnu.tn

***Abstract*—In the area of image processing, segmentation of an image into multiple regions is very important for classification and recognition steps. It has been widely used in many application fields such as medical image analysis to characterize and detect anatomical structures, robotics features extraction for mobile robot localization and detection and map procession for lines and legends finding. In this paper, we describe a novel method for segmentation of images based on Particle Swarm optimization for determining multilevel threshold for a given image. The proposed method is compared with other known multilevel segmentation methods to demonstrate its efficiency. Experimental results show that this method can reliably segment and give threshold values than other methods considering different measures.**

***Keywords*—Multilevel segmentation; Particle Swarm Optimization; fitness function; Image processing**

I. INTRODUCTION

Segmentation is to divide an image into region corresponding to objects. These objects include more information than pixels. Indeed, interpretation of images based on objects gives more illustrative and meaningful information's than information gives based on individual pixel interpretation. The principle aim of image segmentation is to apply a specific treatment or to interpret the image content.

If human knows naturally separate objects in an image due to high-level knowledge that consist on understanding of the objects and the scene. Developing segmentation algorithms still one of the most topics research common in the field of image processing. So far, there are many segmentation methods that can be classified into four main types including region based segmentation like region-growing [1, 2] and region based split and merging [3, 4], edge-based segmentation [5, 6], histogram thresholding based method [7, 8] and Segmentation based on hybridization between two of the first three segmentations.

Thresholding method is one of the most common methods for the segmentation of images into two or more clusters [9, 10]. It is a simple and popular method for digital image processing that can be divided into three different types: global thresholding methods [11, 12], local thresholding methods [13, 14] and optimal thresholding methods [15, 16]. In the former, global thresholding methods are used to determinate a threshold for the entire image, it concern only the binarization and the result after segmentation is a binary image. Local thresholding methods are fast and for this reason they are suitable for the case of multilevel thresholding. However the major drawback of this method is the determination of the number of thresholds. The advantage of the optimal thresholding methods is the objective function. Indeed, the determining of the best threshold values amounts to optimize the objective function.

Several algorithms have been widely proposed in the literature for the bi-level [17, 18, 19, 20, 21] and also for the multi-level thresholding problem. For two level thresholding, solving the problem is same as finding the threshold value called T which satisfies this condition: pixels which are lower than T represent the object and the other pixels the background. This problem could be extended to n-level thresholding when distinct objects are depicted within a given scene, multiple threshold values called $T_1, T_2, ..., T_i$ with $i \geq 2$ have to be determinate. A variety of multilevel thresholding approaches have been proposed for image segmentation including the Otsu criterion [22, 23, 24]. This method is simple but it has a disadvantage that it is computationally expensive. To overcome this problem, many papers have been published in the literature designed especially for computation acceleration of the specific objective function [25, 26]. Among this category, we find methods that utilize the meta-heuristic optimization methods namely GA [27, 28, 29], ABC [30, 31, 32], PSO [33, 34, 35]. Meta-heuristic algorithms have been inspired from nature and they are used to solve difficult optimization problems. Those optimization algorithms could be used to solve many complex optimization problems, which are non-linear, non-differentiable and multi-model.

In this paper, a proposed algorithm for image segmentation based on PSO is used to automatically determinate the threshold values in multilevel thresholding problem. The PSO algorithm is based on swarm behavior of birds where particles (in our case: pixels), fly through the search space using two simples equations for velocity and position. This algorithm has been widely used to solve global and local optimization problems. This algorithm has undergone several evolutions, we cite the Darwinian Particle Swarm Optimization (DPSO) [36] and the fractional calculus based PSO called FODPSO [37]. This work mainly focuses on using a new variant of PSO

International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.203-213
ISSN 2356-5608

denoted as MMPSO (Multithresholding based on Modified PSO) and it is the first time to verify and apply this method to multilevel segmentation. We start by presenting a brief review on the most known meta-heuristic algorithms. Next in section 2, we introduce the n-level thresholding problem formulation. Section 3 presents a brief review of the traditional PSO and the MMPSO. In section 4, benchmarks images used to demonstrate the advantages given by the proposed method in comparison with other commonly used algorithm such as genetic algorithm (GA) and traditional PSO. Finally, section 5 outlines conclusions.

II. RELATED WORK BASED META-HEURESTIC ALGORITHM

A. *Genetic Algorithm (GA)*

Genetic algorithms (GAs) are meta-heuristics search methods belonging to the class of evolutionary algorithms (EAs). They are inspired by the analogy between the optimization process and the evolution of organisms. A GA is used to search for global or optimal solutions when no deterministic method exists or when the deterministic method is computationally complex. GA is a population based algorithm that was proposed by John Holland in 1975 [38]. Then by Goldberg in 1989 [39], Holland in 1992 [40], Man et al. in 1996 [41], Schmitt and Petrowski in 2001 [42, 43]. Later, this technique has been used in many fields such as image segmentation, which has been transformed into an optimization problem like in [44, 45, 46, 47].

Multithresholding amounts to finding more than a threshold, this means several solutions. Each solution is represented as chromosome, each chromosome is constructed from genes and solutions generated per iteration are called population. The size of the population is the number of solutions per iteration. Let n the population size of randomly generated individuals. The genetic algorithm starts with n random solution. Then the best member solutions are selected to generate new solutions, so the best generated solutions will be added to the next iteration and all bad solutions will be rejected. The selection of the best solution in each generation is based best fitness evaluation values of every individual in the population to form a new population. The stopping criterion is determinate by the number of generations that has been produced, or on a satisfactory fitness value that has been reached for the population. Generally, the genetic algorithm is based on four steps: population initialization, evaluation of fitness, reproduction and termination criterion.

TABLE I. PSEUDO-CODE OF THE GA ALGORITHM

1. **Initialize** a population of random individual solutions
2. **While stopping criterion not met**, do:
   2-1. Create a new population
   2-2. Until creation of the entire population, do:
      i. **Select** a pair of individuals that has the lowest value of fitness.
      ii. **Cross-over** the two individuals to produce two new individuals.
   2-3. Randomize each individual of the new population to mutate.
   2-4. Replace the population with the new population.
3. Display the best solution found over the search.

B. *Ant Colony Optimization (ACO)*

The ant colony algorithms are a family of meta-heuristics inspired from nature using swarm intelligence. The behavior of ants in their search for food has been studied and applied to solve complex optimization problems. Simply, ants initially start by moving randomly. Once the food has been found, again they join their colony by filing in the way a chemical substance called pheromone [48]. Other ants that are experiencing the same path have a high probability to stop their random movements and follow the path marked by the substance (pheromone): this was called the phenomenon of stigmergic optimization [49]. After some research, there will be several paths that lead to food. The shortest path will be covered without necessarily having a global vision of the path [50, 51]: this phenomenon is based positive feedback. Therefore, the long paths ultimately disappear. Finally, all the ants will follow the shortest path.

Dorigo [52] and Coloni [53] are the first who have trying to implement an inspired ACO analogy to solve the problem of searching for an optimal path in a graph. Then, several problems have emerged and drawing on various aspects based behavior of ants. Multithresholding is among the areas in which the ACO algorithm was implemented to obtain the optimal thresholds in the field of image segmentation [54, 55, 56, 57].

Table 2 below gives the overall description of the ACO algorithm:

TABLE II. PSEUDO-CODE OF THE ACO ALGORITHM

1. **Set** parameters and **initialize** pheromone trails
2. for *i* from *1* to number_of_iterations do
   for *j* from *1* to population_size do
      Building solutions based on the probability of state transition
      Stop when all ants have been generated
3. **Evaluate** all solutions and **select** the best one even iteration
4. **Apply** the pheromone update rule
5. Continue until reaching the stopping criterion

C. *Artificial Bee Colony (ABC)*

Artificial Bee Colony algorithm was firstly proposed by Karaboga [58] in 2005 for searching numerical optimization problems based on intelligent foraging behavior of honey bee swarm. Later, further improvements have been carried out for the ABC algorithm by Karaboga and Basturk in 2006 and 2008 [59, 60]. Colony of ABC model consists of three groups of bees: employed bees, onlookers and scouts [61].

The bee which discovered and a source of food to exploit belong to the employed bees group. The second groups of bees

International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.203-213
ISSN 2356-5608

called onlookers are those waiting in the hive for information about the sources of food from the employed bees. The third group of scouts' bees is the set of bees which will randomly search for the food sources around the hive. After exploiting a source of food, a bee belonging to the employed bees group returns to the hive and shares information about the nectar amount produced in the food source with other bees. The employed bee starts dancing in the dance area of the hive. Communication among bees related to the quality of food sources takes place in the dancing area. This dance is called a waggle dance and it is made to share information with a probability proportional to the profitability of the food source. More profitable the source is, more the dancing duration is so longer. An onlooker on the dance floor watches numerous dances and selects to employ herself at the most profitable source. After watching several dances, an onlooker bee chooses a source of food and becomes employed bee. In a similar way, a scout is called employed when it finds a source of food. After completely exploiting a source of food, all employed bees abandoned it and change into onlookers or scouts [62, 63]. Typically algorithm for the ABC algorithm is given in table 3 below:

TABLE III. PSEUDO-CODE OF THE ABC ALGORITHM

1: **Initialize** the population of solution and generate food sources
2: **Assign the employed** bees on their food sources
3: **Assign onlooker** bees to the food sources depending on their amount of nectar
4: The scout bees **randomly research** the neighbor area to discover new food sources.
5: The best solution is **recorded and increases** the cycle by 1.
6: **The algorithm is end** if the cycle is equal to the maximum cycle number (max cycle), otherwise go to Step 2

The ABC algorithm finds optimal solutions for the optimization problems. Many researchers use this algorithm to determine the threshold values for the multilevel thresholding problem [64, 65, 66 67, 68].

D. Shuffled Frog Leaping Algorithm (SFLA)

SFLA is a recent meta-heuristic algorithm proposed by Eusuff and Lansey in 2003 [69] that mimics the principle of a group of frogs evolution that searches discrete locations containing as much food as available. SFLA combines the advantages of the local search tool of the PSO algorithm and the idea of mixing information from parallel local searches to move toward a global solution [70]. The SFLA algorithm has been tested on several combinatorial problems and has demonstrated effectiveness [71] in various global solutions [72, 73].

In general case, when we apply the SFLA algorithm to found an optimum solution, each frog has a different solution from others frogs. This solution is determined according to the fitness function and its adaptability. SFLA algorithm involves a population defined by a set of frogs (solutions). The entire population is partitioned into a predefined number of subsets referred to as memplexes. Those mememplexes are considered as different crops of frogs to performing a local search. Frogs of each memplex have their own strategy to explore the environment in different directions. After a predefined number of memetic evolution, the exchange of information between memplexes takes place in a procedure of shuffling [74]. This procedure must ensure that the evolution toward a particular interval is free from all prejudices. Memetic evolution and shuffling are performed alternatively until reaching the convergence criterion or otherwise until a stopping criterion. Steps of SFLA are given below [12].

Step 1: Initial population of F frogs, in which individual frogs are equivalent to the GA chromosomes, is created randomly.

Step 2: All frogs are sorted in descending order based on their fitness values and divided into m memplexes, each memplex containing p frogs; the frog that is placed first moves to the first memplex, the second one moves to the second memplex, the pth one to the pth memplex, and the (p + 1)th returns to the first memplex, etc.

Step 3: Within each memplex, the frogs having the best and the worst fitness are identified. The frog with the best fitness in the whole population is identified. During the evolution of memplexes, worst frogs jump to reach the best ones.

Step 4: After a defined number of memplex evolution stages, all frogs of memplexes are collected and sorted in descending order again based on their fitness. Step 2 divides frogs into different memplexes again, and then step 3 is achieved.

Step 5: If a predefined solution or a fixed iteration number is reached, the algorithm stops.

Table 4 shows the proposed algorithm based SFLA technique:

TABLE IV. PSEUDO-CODE OF THE SFLA ALGORITHM

```
Begin;
  Generate random population of P solutions (individuals);
  For each individual frog in the population: calculate fitness (i);
  Sort the whole population P in descending order of their fitness;
    Divide the population P into m memeplexes;
    For each memeplex;
      Determine the best and worst individuals;
      Improve the worst individual position
      Repeat for a specific number of iterations;
    End;
    Combine the evolved memeplexes;
    Sort the population P in descending order of their fitness;
  Check if termination = true;
End;
```

The SFLA algorithm had been recently used in determining the optimal thresholding in the field of image segmentation exactly in the identification of the bi-level [75, 76] and multi-level thresholding [77, 78, 79, 80, 81, 82].

International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.203-213
ISSN 2356-5608

III. Proposed Approach

A. Image Multilevel Thresholding: Optimization problem

Multilevel thresholding segments images into several distinct regions. Using this process, it is possible to determinate more than one threshold value for a given gray-level image and segments it into certain brightness regions, which correspond to one background and several objects. Let consider a gray-level image that contains N pixels distributed as objects and background. The multilevel threshold selection can be considered as the problem of finding a set $T(l), l = 1,2,..., L$ of threshold values with L is defined as the intensity level of the image. As a result of thresholding, the original image will be transformed to an image with *L+1* levels. If *T(l), l=1,2 , …, L* are the threshold values with *T(1) < T(2) < T(3), …, < T(L)* and $f(x, y)$ is the image function which gives the gray-level value of the pixel with coordinates *(x, y)*. The resultant image $F(x, y)$ as explain before, is defined as:

$$F(x,y)\begin{cases} 0, & if \quad f(x,y) \le T(1) \\ 1, & if \quad T(1) \le f(x,y) \le T(2) \\ . & . \quad . \\ . & . \quad . \\ L & if \quad f(x,y) \ge T(L) \end{cases} \tag{1}$$

The problem of multilevel thresholding can be reduced to an optimization problem. The goal becomes to search and found the threshold values that maximize the fitness function ϕ of the gray-level component, defined as:

$$\phi = \max_{T(1) < T(2) < T(3), \ldots, < T(L)} \sigma^2(T) \tag{2}$$

With σ^2 is the between-class variance generally defined by:

$$\sigma^2 = P_1\sigma_1^2 + P_2\sigma_2^2 \tag{3}$$

And:

$$\sigma_1^{\,2} = \frac{1}{T}\sum_{i=0}^{T-1}(h(i) - \mu_1)^2 \; ; \; \sigma_2^{\,2} = \frac{1}{256-T}\sum_{i=T}^{255}(h(i) - \mu_2)^2 \tag{4}$$

$$\mu_1 = \frac{1}{T}\sum_{i=0}^{T-1} h(i) \; ; \; \mu_2 = \frac{1}{256-T}\sum_{i=T}^{255} h(i) \tag{5}$$

$$P_1 = \frac{1}{W \times H}\sum_{i=0}^{T-1} h(i) \; ; \; P_2 = \frac{1}{W \times H}\sum_{i=T}^{255} h(i) \tag{6}$$

With h is the histogram of this image and N and M are respectively width and height of the image.

The major drawback of this problem is the computational effort that is much larger as the number of threshold levels increase. In the last decade, biologically inspired methods have been used as computationally efficient alternatives to analytical methods to solve optimization problems [83, 84].

B. The MMPSO Algorithm

Particle Swarm Optimization (PSO) is a population-based optimization algorithm belonging to the evolutionary computation paradigm [85]. It is proposed by [86] to solve problems with continuous variables. It is very suitable to solve complex problem with multiple decision at low cost of computational time. As compared with other evolutionary computation algorithms, PSO has many advantages such as non-use of genetic operation; like crossover and mutation with Genetic Algorithm, PSO has a memory so it can learn from others neighbor or itself; after moving to the new group it has more information from its parents and can find the best threshold value in short time and it requires only mathematical operations which make its implementation very easy and not cost in terms of execution time.

PSO is an efficient algorithm that is based on a population initialized with a random solution called particle. Each particle represents an approximate solution to a complex problem in the search space. This solution is determined based on the collective experiences of the same swarm. In PSO, each particle is characterized by its own position vector and velocity vector. The movement of these vectors in the search space is controlled by the following recursive equations:

$$v_{im} = w * v_{im} + c_1 * rand1() * (p_{im} - x_{im}) + c_2 * rand2() * (p_{gm} - x_{im}) \tag{7}$$

$$x_{im} = x_{im} + v_{im} \tag{8}$$

Where x_{im} is the i[th] position of the particle of the swarm; v_{im} the velocity of this particle; p_{im} the best previous position of the i[th] particle; p_{gm} is the best position of particle in the swarm; $1 \le m \le M$ with *M* is the search space; *rand1()* and *rand2()* are the two independents random number with uniform distribution in the range [0, 1]; c_1 and c_2 are two positives constants of accelerations coefficients called cognitive and social parameter respectively; w is called inertia weight and it is used to control the balance between exploration and search space exploitation. The PSO algorithm is briefly detailed in this table 5 below:

TABLE V. Pseudo-code of the PSO Algorithm

PSO algorithm
Initialization :
Initialize the position x_i and the velocity v_i of each particle as follows :

$$x_{im} = x_{\min} + (x_{\max} - x_{\min}) * rand() \quad (9)$$

$$v_{im} = v_{\min} + (v_{\max} - v_{\min}) * rand() \quad (10)$$

do
 for i from 1 to N
Update :
 if $fitness(x_i) \prec fitness(p_i)$ then
  up to date of p_{best} ; $p_i = x_i$
  up to date of g_{best}
  for i from 1 to M
   up to date of v_i and x_i
  end
 end
 end
end

PSO approach is based on the memory and the social interaction among individuals. In the general case, the fitness function allows determining the best position for a particle i to make moves from its current (x_i, t) to the next $(x_i, t+1)$. Moving process is depends on three stages:

1-The current velocity (v_i, t)

2-The best performance (evaluated by the objective function: fitness (f_i, t))

3-The best position of its neighbors (g, t)

Fitness function is used to assess pixels' (particles') for selecting the best individual. It is the most important step that directly affects results of the best position for each individual. So, the overall contribution of this work is to introduce a new fitness function that will give advantages for fast determination of threshold values so that the solutions for the optimal problem. For this, the new introduced algorithm based on the new fitness is called MMPSO and it will be well explained.

Firstly, let calculated a weight P_j for each particle i according to its location using Eq.(11) and then sum SP_i for all particles using Eq (12):

$$P_j = 2^{p-j} x(i, j) \quad (11)$$

$$SP_i = \sum_{j=0}^{j=p} P_j \quad (12)$$

With p is the number of iterations.

Then, the sum is normalized using Eq. (13)

$$N(i) = \frac{255 \times SP_i}{2^{p-1}} \quad (13)$$

After that, let be introduced four parameters $lowsum$ $lownum$ $highsum$ $highnum$ to zero. Then, each particle of the original image with intensity $L(i)$ is compared with $N(i)$ as explained in the pseudo code given in Table 6 below:

TABLE VI. PSEUDO-CODE OF THE COMPARISON

for i from1 to M
 if L(i) < N(i) then
  $lowsum = lowsum + L(i)$
  $lownum = lownum + 1$
 else
  $highsum = highsum + L(i)$
  $highnum = highnum + 1$
 end
end

The computation of the new fitness function depends to the final image after segmentation. For each particle i, after comparing all pixels with *N(i)*, two coefficients u_1 and u_2 are calculated according to Eq. (14) and (15). Finally, the fitness function of the particle i is calculated using Eq. (16).

$$u_1 = \frac{lowsum}{lownum} \quad (14)$$

$$u_2 = \frac{highsum}{highnum} \quad (15)$$

$$fitness(i) = lownum \times highnum \times (u_1 - u_2)^2 \quad (16)$$

This new fitness function increases the probability to use more positions. So, it guarantees and allows the best speed of the convergence to the sought threshold value. This is demonstrated by the following experimental results.

The proposed MMPSO in this work is shown in the table 7 below:

TABLE VII. PSEUDO-CODE OF THE MMPSO ALGORITHM

Final MMPSO code
1. Parameter initialization; including swarm size M, inertia weight w , cognitive and social parameter the position c_1 and c_2 , maximum and minimum velocity values $V_{\max}$ and $V_{\min}$, number of iterations N_{iter}
2. Initialize population of particles with random positions and velocities using the two equations below:
 $x_{im} = x_{\min} + (x_{\max} - x_{\min}) * rand()$
 $v_{im} = v_{\min} + (v_{\max} - v_{\min}) * rand()$
3. Evaluate the fitness function of each particle:
While the stopping criterion is not met
 for each particle i in the swarm
  $Pi = 2^{p-i} x(i, j)$

International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.203-213
ISSN 2356-5608

$$SP_i = \sum_{j=0}^{j=p} P_j$$

$$N(i) = \frac{255 \times SP_i}{2^{p-1}}$$

end
lowsum=0; lownum=0; highnum=0; highsum=0;
for *i* from *1* to *M*
  if *L(i)* < *N(i)* then
    $lowsum = lowsum + L(i)$
    $lownum = lownum + 1$
  else
    $highsum = highsum + L(i)$
    $highnum = highnum + 1$
  end
end
if *lownum=0* then
  $u_1=0$
else

$$u_1 = \frac{lowsum}{lownum}$$

end
if *highnum=0* then
  $u_2=0$
else

$$u_2 = \frac{highsum}{highnum}$$

end
4. if $fitness(x_i) \prec fitness(p_i)$ then
  up to date of p_{best} ; $p_i = x_i$
  up to date of g_{best}
  for i from 1 to N
5. up to date of v_i and x_i
  end
 end
end
end

IV. EXPERIMENTAL RESULTS

MMPSO-based image segmentation which is proposed in this paper was implemented in MATLAB (2011b) on a computer having Intel Core 2 Duo T5800 processor (2.00 GHz) and 3 GB of memory. The proposed methods are tested on a few benchmark images (256 x 256 pixels in size) including: Airplane, Hunter and Map. Fig. 1 blow illustrates all images with their histogram.

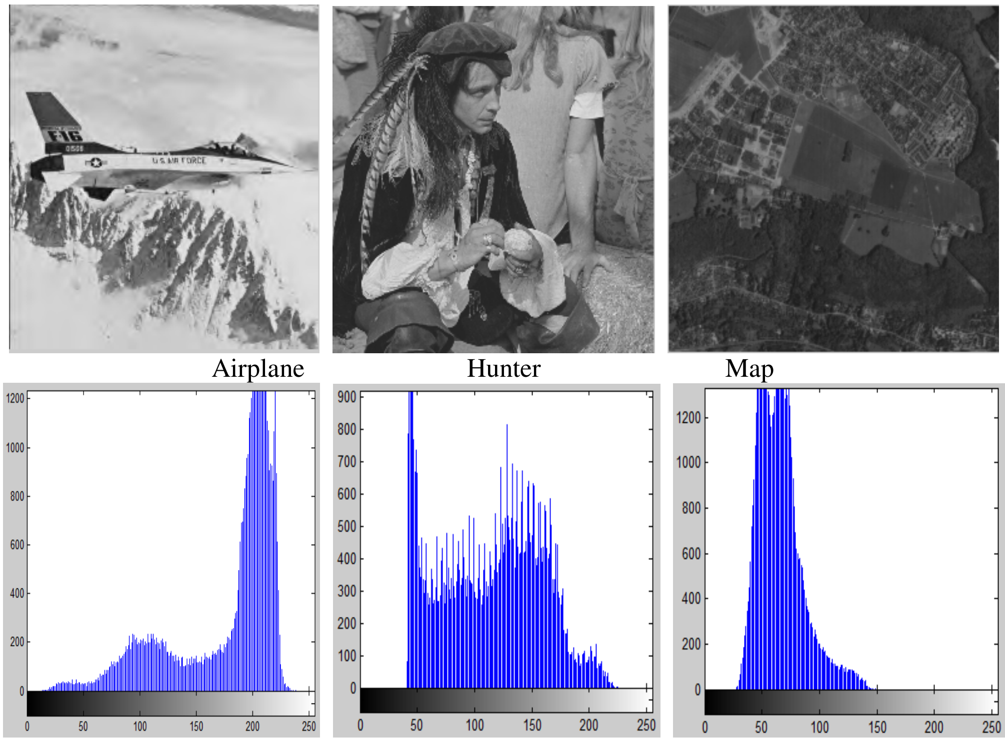

Figure 1. Benchmark images with their histograms.

The performance of the proposed methods is evaluated by comparing their results with a few popular methods such as GA and PSO. As well, MMPSO is a parameterized algorithm depends on many factors like cognitive, social and inertial weights. They were chosen in reference to several works focusing on the convergence analysis of the traditional PSO [87, 83] to give result in faster convergence (Table 8).

TABLE VIII. TABLE 1. INITIAL PARAMETERS OF THE PSO AND MMPSO

Parameters	PSO	MMPSO

International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.203-213
ISSN 2356-5608

Num of Iterations	8	8
Population	200	200
c_1	1.5	1.5
c_2	1.5	1.5
w	1.2	1.2
V_{max}	2	2
V_{min}	-2	-2
X_{max}	255	255
X_{min}	0	0

The computational time and threshold values are among the most important indicators which determinate the ability of the algorithm [88]. For many applications, such as medical images, in particular MRI images, data are considerably large in most cases so using a high speed and highly efficient algorithm is preferable. Therefore, the evaluation of the execution time of the CPU process and threshold values seems essentially and necessary to determinate the performance of the new method.

A. *CPU processing time*

PSO is bio-inspired stochastic algorithm and is like all evolutionary algorithms based random initialization. So, results are not similar in each run. For this, MMPSO and PSO algorithms are executed 20 times and the average CPU process time values are brought in Table 9 below:

TABLE IX. THE AVERAGE CPU PROCESS TIME OF DIFFERENT SEGMENTATION METHODS

Image	Thresholds	MMPSO (s)	PSO (s)	GA (s)
Airplane	2	0.4318	0.4382	0.8113
	3	0.5012	0.4844	0.8642
	4	0.5502	0.5516	0.9189
	5	0.6066	0.6065	0.9862
Hunter	2	0.3814	0.3966	0.7013
	3	0.4766	0.4761	0.7862
	4	0.5516	0.5517	0.9014
	5	0.5913	0.6031	0.9914
Map	2	0.3014	0.3158	0.6821
	3	0.3897	0.3875	0.7532
	4	0.4719	0.4882	0.8852
	5	0.5032	0.5142	0.9365

In the literature, it has been proven that the PSO requires less CPU processing time to find the threshold values in comparison to GA and this is clear in refer to Table 9 above. It is very clearly that MMPSO presents the best processing time. It is able to found in less CPU time than PSO and of course GA the threshold values. It is given that the difference between PSO and MMPSO is not huge for most of CPU process time values, but each time we increase the number of thresholds, the difference will be notable. Also, whenever the image contains more detail, greater the difference is significant, i.e., the gap CPU process time between PSO and MMPSO is higher for the map image than the airplane and hunter images.

B. *Threshold values*

The aim of the proposed algorithm is to determinate the best threshold values the optimal problem. Since all evolutionary methods among of them MMPSO and PSO are stochastic and random, the results are not completely the same in each run and in each number of threshold. For this, different level for image segmentation are applied and classified in Table 10 below.

TABLE X. AVERAGE THRESHOLDS OF DIFFERENT SEGMENTATION ALGORITHM

Image	Th	MMPSO	PSO	GA
Airplane	2	116,176	117,174	116,175
	3	95,149,196	99,158,193	86, 133, 204
	4	84,130,173,203	84,125,168,201	71,119,164,200
	5	78,115,150,187,206	60,101,138,177,204	84,124,164,188,204
Hunter	2	54, 118	52,116	51,115
	3	42,88,144	39,86,135	36,89,133
	4	36,88,133,159	36,84,130,157	39,93,142,163
	5	41,84,128,159,184	37,85,125,154,177	39,94,130,169,204

International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.203-213
ISSN 2356-5608

Map	2	61,91	60,90	56,85
	3	59,78,104	57,75,101	57,74,97
	4	54,68,84,108	52,68,82,103	49,66,78,100
	5	47,57,64,76,98	46,56,63,75,97	44,55,60,72,95

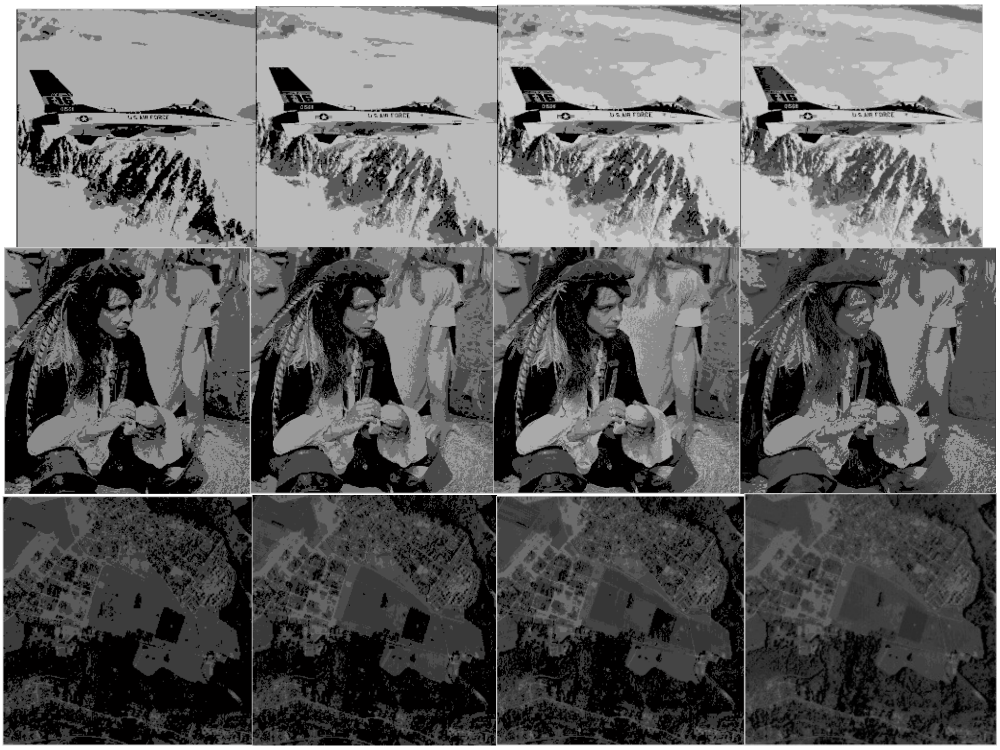

Figure 2. Results of segmentation with 2, 3, 4, 5 thresholds, respectively from (left to right)

In fig 2, it gives results of different segmented images with various threshold levels. Qualitative results given below shows those images with higher level of segmentation have more detail than others. Along the same lines, Table 10 shows that for 6-level threshold, it is almost the MMPSO gives the best threshold values.

V. CONCLUSION

In this paper, MMPSO inspired Particle Swarm Optimization algorithm for multilevel thresholding is developed. This method is able to determine optimal threshold values from complex gray-level images. In this purpose, a new fitness function is developed to ensure best threshold values in less CPU process time. Experimental results demonstrated by computing optimal threshold values in 4 different levels (3, 4, 5 and 6 levels) for three different benchmark images. Results indicate that the MMPSO is more efficient than PSO and GA. In particular, this method is better when the level of segmentation increase and the image is with more details. Moreover, due to the low computational complexity of the algorithm, this algorithm will be applied to classify the MRI medical images.

REFERENCES

[1] A. Melouah, A.; A Novel Region Growing Segmentation Algorithm for Mass Extraction in Mammograms. In: Amine, A., Ait Mohamed, O., Bellatreche, L. (eds). Modeling Approaches and Algorithms for Advanced Computer Applications Studies in Computational Intelligence, pp 95 - 104 (2013)

[2] J. Chakraborty, S. Mukhopadhyay, V. Singla, N. Khandelwal, R.M. Rangayyan; Detection of masses in mammograms using region growing controlled by multilevel thresholding, Rome, 25th International Symposium on Computer-Based Medical Systems (CBMS),20 - 22 June, pp. 1 – 6 (2012)

[3] R. Dragon, J. Ostermann, L. Van Gool; Robust Realtime Motion-Split-And-Merge for Motion Segmentation. In: Computer Science, GCPR 2013. 35th German Conference, Saarbrücken, Germany, September (3) 6, pp. 425 - 434 (2013)

[4] D. Chaudhuri, A. Agrawal; Split-and-merge Procedure for Image Segmentation using Bimodality Detection Approach, Defence Science Journal, (60) 3, pp. 290 - 301 (2010)

[5] X. Cao, W. Ding, S. Hu, L. Su; Image Segmentation Based on Edge Growth. In: Software Engineering & Digital Media Technology, ICITSE 2012, International Conference, pp 541 - 548 (2013)

[6] M. Sharif, M. Raza, S. Mohsin; Face recognition using edge information and DCT, Sindh Univ. Res. Jour. (Sci. Ser.), (43) 2, pp. 209 - 214 (2011)

[7] T. Baakek, A. Chikh Mohamed; Interactive Image Segmentation Based on Graph Cuts and Automatic Multilevel Thresholding for Brain Images, Journal of Medical Imaging and Health Informatics, (4) 1, pp. 36 - 42 (2014)

[8] F. Martin-Rodriguez; New Tools for Gray Level Histogram Analysis, Applications in Segmentation. In: Image analysis and recognition, ICIAR 2013. 10th International Conference, Póvoa do Varzim, Portugal, June 26 - 28, pp 326 - 335 (2013)

[9] L. Qifang, O. Zhe, C. Xin, Z. Yongquan; A Multilevel Threshold Image Segmentation Algorithm Based on Glowworm Swarm Optimization, Journal of Computational Information Systems, (10) 4, 1621 - 1628 (2014)

[10] R.V. Kulkarni, G.K. Venayagamoorthy; Bio-inspired algorithms for autonomous deployment and localization of sensor nodes. IEEE Transactions, SMC-C40, (6), 663 - 675 (2010)

[11] F. Hamdaoui, A. Ladgham, A. Sakly, A. Mtibaa; A new images segmentation method based on modified PSO algorithm. International Journal of Imaging Systems and Technology, 23 (3), pp. 265 - 271 (2013)

[12] A. Ladgham, F. Hamdaoui, A. Sakly, A. Mtibaa; Fast MR brain image segmentation based on modified Shuffled Frog Leaping Algorithm. Signal, Image and Video Processing (2013). DOI 10.1007/s11760-013-0546-y

[13] H.J. Sun, T.Q. Deng, Y.Y. Jiao; Remote Sensing Image Segmentation Based on Rough Entropy. In: Advances in Swarm Intelligence ICSI (2013). 4th International Conference, Harbin, China, June 12 - 15, pp 411 - 419 (2013)

[14] S. Sarkar, N. Sen, A. Kundu, S. Das, S.S. Chaudhuri; A Differential Evolutionary Multilevel Segmentation of Near Infra-Red Images Using Renyi's Entropy. In: Advances in Intelligent Systems and Computing FICTA, 2013. International Conference on Frontiers of Intelligent Computing: Theory and Applications, pp 699 - 706 (2013)

[15] J.F. Daisne, M. Sibomana, A. Bol, T. Doumont, M. Lonneux, V. Grégoire ; Tri-dimensional automatic segmentation of PET volumes based on measured source-to-background ratios: influence of reconstruction algorithm. Radiotherapy and oncology 69, pp. 247 - 250 (2003)

[16] D.Y. Huang, T.W. Lin, W.C. Hu; Automatic Multilevel Thresholding Based On Two-Stage Otsu's Method with Cluster Determination by Valley Estimation. International Journal of Innovative Computing, Information and Control, 7 (10), pp. 5631 - 5644 (2011)

[17] Z. Ningning, Y. Tingting, Z. Shaobai; An Improved FCM Medical Image Segmentation Algorithm Based on MMTD, Computational and Mathematical Methods in Medicine, (2014), http://dx.doi.org/10.1155/2014/690349

[18] M. Yasmin, S. Mohsin, M. Sharif, M. Raza, S. Masood; Brain Image Analysis: A Survey, World Applied Sciences Journal, (19), pp. 1484 - 1494 (2012)

[19] M. Raza, M.Sharif, M. Yasmin, S. Masood, S. Mohsin; Brain image representation and rendering: A survey, Research Journal of Applied Sciences, Engineering and Technology, (4) 18, 3274 - 3282 (2012)

[20] M. Al-azawi; Image Thresholding using Histogram Fuzzy Approximation. International Journal of Computer Applications, 83, (9), pp. 36-40 (2013)

[21] A. Nakib, S. Roman, H. Oulhadj, P. Siarry; Fast brain MRI segmentation based on two-dimensional survival exponential entropy and particle swarm optimization, Proceedings of the 29th Annual International Conference of the IEEE EMBS, Cité Internationale, Lyon, France, August 23-26, 2007, pp. 5563-5566 (2007)

[22] N. Otsu; A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man and Cybernetics, 9 (1), pp 62 - 66 (1979)

[23] C. Yao, H.J. Chen; Automated retinal blood vessels segmentation based on simplified PCNN and fast 2D-Otsu algorithm. Journal of Central South University of Technology 16, pp 640 - 646 (2009)

[24] D.Y. Huang, C.H. Wang; Optimal multi-level thresholding using a two-stage Otsu optimization approach. Pattern Recognition Letters, 30 (3), pp 275 - 284 (2009)

[25] B.F. Wu, Y.L. Chen, C.C. Chiu; Recursive algorithms for image segmentation based on a discriminant criterion. International Journal of Signal Processing, (1), pp 55 - 60 (2004)

[26] K. Hammouche, M. Diaf, P. Siarry; A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem. Engineering Applications of Artificial Intelligence (23) pp 676 - 688 (2010)

[27] K. Hammouche, M. Diaf, P. Siarry; A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation. Computer Vision and Image Understanding, (109) pp 163 - 175 (2008)

[28] W.B. Tao, J.W Tian, J. Liu; Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm. Pattern Recognition Letters, 24 (16), 3069 – 3078 (2003)

[29] Z. Yang, Z. Pu, Z.Qi; Relative entropy multilevel thresholding method based on genetic optimization. Neural Networks and Signal Processing, In proceedings of the 2003 International Conference on. IEEE, (1), pp. 583 - 586 (2003)

[30] E. Hancer, C. Ozturk, D. Karaboga; Artificial Bee Colony Based Image clustering Method, Evolutionary Computation, IEEE International Congres, Brisbane, QLD, 10 – 15 June, pp. 1 – 5 (2012)

[31] Y. Zhang, L. Wu; Optimal Multi-Level Thresholding Based on Maximum Tsallis Entropy via an Artificial Bee Colony Approach. Entropy (2011). doi:10.3390/e13040841

[32] R. Geng; Color Image Segmentation Based on Self-Organizing Maps, Advances in Key Engineering Materials, Advanced Materials Research, (214), pp. 693 – 698 (2011)

[33] A.K. Bhandari, V.K. Singh, A. Kumar, G.K. Singh; Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur ' s entropy, Expert Systems with Applications, (41) 7, pp. 3538- 3560 (2014)

[34] H. Gao, S. Kwong, J. Yang, J. Cao; Particle swarm optimization based on intermediate disturbance strategy algorithm and its application in multi-threshold image segmentation, Information Sciences, (250) 20, pp. 82 − 112 (2013)

[35] P. Ghamisi, M.S. Couceiro, J.A. Benediktsson, N.M.F. Ferreira; An efficient method for segmentation of images based on fractional calculus and natural selection. Expert Systems with Applications, (39), pp 12407 - 12417 (2012)

[36] J. Tillett, T.M. Rao, F. Sahin, R. Rao, S. Brockport; Darwinian Particle Swarm Optimization. In: Proceedings of the 2nd Indian international conference on artificial intelligence 2005 pp. 1474 - 1487 (2005)

[37] M.S Couceiro, N.M.F. Ferreira, J.A.T, Machado; In Fractional order Darwinian Particle Swarm Optimization. In FSS'11 (2011). Symposium on fractional signals and systems, November 4 - 5, Coimbra, Portugal

[38] J.H. Holland; Adaptation in Natural and Artificial systems, Ann Arbor: The University of Michigan Press, (1975)

[39] D.E. Goldberg ; Algorithmes Génétiques : Exploration, optimisation et apprentissage automatique, Edition Wesley, (1989)

[40] J.H. Holland; Genetic Algorithms, pour la science, Edition of Scientific American, (179), pp. 44 - 50 (1992)

[41] K.F. Man, K.S. Tang, S. Kwong; Genetic Algorithms: Concepts and Applications, IEEE Transactions on Industrial Electronics, 43 (5), pp 519 - 534 (1996)

[42] L.M. Schmitt; Fundamental study: Theory of genetic algorithms, Theoretical Computer Science, (259), pp. 1 - 61 (2001)

[43] A. Petrowski; Une introduction à l'optimisation par algorithmes génétiques, http://www-inf.int-evry.fr/~ap/EC-tutoriel/Tutoriel.html, (2001)

[44] B.D. Phulpagar, S.S. Kulkarni; Image Segmentation using Genetic Algorithm for Four Gray Classes, IEEE International Conference on Energy, Automation and Signal, pp. 1 - 4, December (2011)

[45] B.D. Phulpagar, R.S. Bichkar; Segmentation of Noisy Binary Images Containing Circular and Elliptical Objects using Genetic Algorithms, IJCA, 66 (22), pp. 1 - 7, March (2013)

[46] K. Janc, J. Tarasiuk, A.S, Bonnet, P. Lipinski; Genetic algorithms as a useful tool for trabecular and cortical bone segmentation, Comput Methods Programs Biomed., 111 (1), pp. 72 - 83. doi: 10.1016/j.cmpb.2013.03.012 (2013)

[47] S. Manikandan, K. Ramar, I.M. Willjuice, K.G. Srinivasagan; Multilevel thresholding for segmentation of medical brain images using real coded genetic algorithm, Measurement, (47), pp. 558 - 568 (2014)

[48] M. Dorigo, L.M. Gambardella; Guest editorial special on ant colony optimization IEEE Transactions on evolutionary computation, 6(4), pp. 317 - 319 (2002)

[49] A. Ajith, G. Crina, R. Vitorino; Stigmergic Optimization, Studies in Computational Intelligence, (31), pp. 1 - 299 (2006)

[50] R. Beckers, J.L. Deneubourg, S. Goss; Trails and U- Turns in the Selection of a Path by the Ant Lasius Niger, J. Theor. Biol., (159), pp. 397 - 415 (1992)

[51] S. Goss, S. Aron, J.L. Deneubourg, J. M Pasteels; Self- Organized Shortcuts in the Argentine Ant. Naturwissenchaften, (76), pp. 579 - 581 (1989)

[52] M. Dorigo, V. Maniezzo, V. Colorni; Ant System : optimization by a colony of cooperating agents, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 26 (1), pp. 29 - 41 (1996)

[53] A. Colorni, M. Dorigo, V. Maniezzo ; Distributed Optimization by Ant Colonies, In : Proceedings of the First European Conference on Artificial Life, pp. 134 - 142, MIT Press, Paris, France, December (1991)

[54] A.A. Mousa, I.M. El-Desoky; Stability of Pareto optimal allocation of land reclamation by multistage decision-based multipheromone ant colony optimization. Swarm and Evolutionary Computation, (13), pp. 13 - 21 (2013)

[55] Y.C. Liang, Y.C. Yin; Optimal multilevel thresholding using a hybrid ant colony system. Journal of the Chinese Institute of Industrial Engineers, 28 (1), pp. 20 - 33 (2011)

[56] L. Ma, K. Wang, D. Zhang; A universal texture segmentation and representation scheme based on ant colony optimization for iris image processing. Computers and Mathematics with Applications, 57 (11) 12, pp. 1862 - 1866 (2009)

[57] W. Tao, H. Jin, L. Liu; Object segmentation using ant colony optimization algorithm and fuzzy entropy. Pattern Recognition Letters, 28 (7), pp. 788 - 796 (2007)

[58] D. Karaboga; An idea based on honey bee swarm for numerical optimization, Technical Report TR06, Computer Engineering Department, Erciyes University, Turkey, (2005)

[59] B. Basturk, D. Karaboga; An artificial bee colony (abc) algorithm for numeric function optimization, IEEE Swarm Intelligence Symposium, Indianapolis, Indiana, USA, May (2006)

[60] D. Karaboga, B. Basturk; On The Performance of Artificial Bee Colony (ABC) Algorithm, Applied Soft Computing, (8) 1, pp. 687 - 697 (2008)

[61] D. Karaboga, B. Basturk; Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems, Foundations of Fuzzy Logic and Soft Computing, Lecture Notes in Computer Science, (45) 29, pp 789 - 798 (2007)

[62] A. Hadidi, S.K. Azad, S.K. Azad; Structural optimization using artificial bee colony algorithm, The second International Conference on Engineering Optimization, Lisbon, Portugal, September 6 - 9 (2010)

[63] V. Tereshko, A. Loengarov; Collective Decision-Making in Honeybee Foraging Dynamics, Computing and Information Systems Journal, (9) 3 (2005)

[64] M.H. Horng; Multilevel Minimum Cross Entropy Thresholding using Artificial Bee Colony Algorithm, Telkomnika, (11) 9, pp. 5229 – 5236 (2013)

[65] B. Akay; A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding, Applied Soft Computing, 13 (6), pp. 3066 - 3091 (2013)

[66] K. Charansiriphaisan, S. Chiewchanwattana, K. Sunat; A Comparative Study of Improved Artificial Bee Colony Algorithms Applied to Multilevel Image Thresholding, Mathematical Problems in Engineering, (2013), http://dx.doi.org/10.1155/2013/927591

[67] Y.F. Cao, Y.H. Xiao, W.Y. Yu, Y.C. Chen; Multi-level Threshold Image Segmentation Based on PSNR using Artificial Bee Colony Algorithm, Research Journal of Applied Sciences, Engineering and Technology 4 (2), 104 - 107 (2012)

[68] M.H. Horng, T.W. Jiang; Multilevel Image Thresholding Selection Using the Artificial Bee Colony Algorithm, Artificial Intelligence and Computational Intelligence, Lecture Notes in Computer Science, (6320), pp. 318 – 325 (2010)

[69] M.M. Eusuff, K.E. Lansey; Optimization of water distribution network design using the shuffled frog leaping algorithm. J.Water Resour. Plan. Manag. 129 (3), 210 - 225 (2003)

[70] Q.Y. Duan, V.K. Gupta, S. Sorooshian; Shuffled complex evolution approach for effective and efficient global minimization. J. Optimization Theory Appns, (76), pp. 502 – 521 (1993)

[71] C. Fang, L. Chang; An effective shuffled frog-leaping algorithm for resource-constrained project scheduling problem, Computers & Operations Research, (39) 5, pp. 890 - 901 (2012)

[72] M.R. Narimani; A New Modified Shuffle Frog Leaping Algorithm for Non-Smooth Economic Dispatch, World Applied Sciences Journal, (12) 6, pp. 803 – 814 (2011)

[73] N. Wang, X. Li, X.H. Chen; Fast three-dimensional Otsu thresholding with shuffled frog-leaping algorithm, Pattern Recognition Letters, Meta-heuristic Intelligence Based Image Processing, (31) 13, pp. 1809 – 1815 (2010)

[74] S.Y. Liong, M. Atiquzzaman; Optimal design of water distribution network using shuffled complex evolution. J. Instn. Engrs, (44), pp 93 – 107 (2004).

[75] Y.J. Gu, Z.H. Jia, X.Z. Qin, J Yang, S.N. Pang; Image Segmentation Algorithm based on Shuffled Frog-leaping with FCM, Communications Technology, (2011)

[76] C.S. Yang, L.Y. Chuang, C.H. Ke; A combination of shuffled Frog–Leaping algorithm and genetic algorithm for gene selection. J. Adv. Comput. Intell. Intell. Inf. (12) 3, pp. 218 - 226 (2008)

[77] M.H. Horng; Multilevel image threshold selection based on the shuffled frog-leaping algorithm, Journal of Chemical and Pharmaceutical Research, (5) 9, pp. 599 – 605 (2013)

[78] S. Ouadfel, S. Meshoul; A Fully Adaptive and Hybrid Method for Image Segmentation Using Multilevel Thresholding, International Journal of Image, Graphics and Signal Processing (IJIGSP), (5) 1, pp. 46 - 57 (2013)

[79] M.H. Horng; Multilevel image thresholding by using the shuffled frog-leaping optimization algorithm, Nano, Information Technology and Reliability (NASNIT), 2011 15th North-East Asia Symposium, Macao, 24-26 Oct. 2011, pp. 144 - 149 (2011)

[80] K. Jiehong, M. Ma; Image thresholding segmentation based on frog leaping algorithm and Ostu method. Yunnan University (Natural Science Edition). pp. 634 – 640, (2012)

[81] J. Liu, Z. Li, X. Hu, Y. Chen; Multiobjective optizition shuffled frog-leaping biclustering, Proceedings of 2011 IEEE International Conference on Bioinformatics and Biomedicine Workshops, Atlanta, 12-15 November, pp. 151 – 156 (2011)

[82] A. Bharuri, A. Bhaduri; Color Image Segmentation Using Clonal Selection-Based Shuffled Frog Leaping Algorithm, Advances in Recent Technologies in Communication and Computing, 2009. ARTCom '09. International Conference on, Kottayam, Kerala, 27-28 Oct. 2009, pp. 517 – 520, (2009)

[83] M.S. Couceiro, J.M.A. Luz, C.M. Figueiredo, N.M.F. Ferreira, G. Dias; Parameter estimation for a mathematical model of the golf putting. In WACI'10 (2010). Workshop applications of computational intelligence ISEC-IPC, December 2, Coimbra, Portugal pp. 1-8 (2010)

[84] M.S. Couceiro, N.M.F. Ferreira, J.A.T. Machado; Application of fractional algoritms in the control of a robotic bird. Journal of Communications in Nonlinear Science and Numerical Simulation – Special Issue, 15(4), pp 895–910 (2010)

[85] R.C. Eberhart, J. Kennedy; A new optimizer using particle swarm theory, in: 6th Symposium on Micro Machine and Human Science 1995, pp. 39–43 (1995)

ISSN 2356-5608

[86] J. Kennedy, R.C. Eberhart; Particle swarm optimization. In IEEE Int. Conf. Neural Netw (4), pp 1942-1948 (1995)

[87] M. Jiang, Y.P. Luo, S.Y. Yang; Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm. Information Processing Letters, 102(1), 8–16 (2007)

[88] J. Fan, M. Han, J. Wang; Single point iterative weighted fuzzy C-means clustering algorithm for remote sensing image segmentation. Pattern Recognition, 42, 2527–2540 (2009)