Automatic Multi-Level Thresholding Segmentation Based on Multi-Objective Optimization

[1]L. DJEROU, [2]N. KHELIL, [3]N. H. DEHIMI and [4]M. BATOUCHE
[1]LESIA Laboratory, University of Biskra, *ldjerou@yahoo.fr*, Algeria
[2]AM Laboratory, University of Biskra, *khelilna@yahoo.fr*, Algeria
[3]University of Oum El Bouaghi, *d.nouna@live.fr*, Algeria
[4] Misc laboratory, University of Constantine, *batouche@umc.edu.dz*, Algeria

***Abstract*-In this paper, we present a new multi-level image thresholding technique, called Automatic Threshold based on Multi-objective Optimization "*ATMO*" that combines the flexibility of multi-objective fitness functions with the power of a Binary Particle Swarm Optimization algorithm "*BPSO*", for searching the "optimum" number of the thresholds and simultaneously the optimal thresholds of three criteria: the between-class variances criterion, the minimum error criterion and the entropy criterion. Some examples of test images are presented to compare our segmentation method, based on the multi-objective optimization approach with Otsu's, Kapur's and Kittler's methods. Our experimental results show that the thresholding method based on multi-objective optimization is more efficient than the classical Otsu's, Kapur's and Kittler's methods.**

***Keywords:* Binary Particle Swarm Optimization, Image segmentation, Image thresholding, Multi-objective Optimization, Non-pare to approach.**

I. INTRODUCTION

Image segmentation is a low level image processing task that aims at partitioning an image into regions in order that each region groups contiguous pixels sharing similar attributes (intensity, color, etc.). It is a very important process because it is the first step of the image understanding process, and all others steps, such as feature extraction, classification and recognition, depend heavily on its results.

A wide variety of image segmentation techniques have been reported in the literature. A good review of these methods can be found in [14]. In general, these techniques can be categorized into thresholding, edge-based, region growing and clustering techniques.

Image thresholding is an important technique for image processing and pattern recognition that can be classified as bi-level thresholding and multi-level thresholding. Bi-level thresholding classifies the pixels of an image into two classes, one including those pixels with gray-levels above a certain threshold, the other including the rest. Multi-level thresholding divides the pixels into several classes. The pixels belonging to the same class have gray-levels within a specific range defined by several thresholds.

Various parametric and non-parametric thresholding methods and criteria have been proposed in order to perform bi-level thresholding [16-17]. They are extendable to multi-level thresholding as well. However, for optimal multi-level thresholding, existing algorithms are being trapped by an exhaustive search of all possible threshold subsets, then the objective function is evaluated of every possible placement of the thresholds and take the positions for which the objective function is optimal, then the objective function is evaluated of every possible placement of the thresholds and take the positions for which the objective function is optimal, thus implies to evaluate $\binom{n}{k}$ possibilities, which can be shown as:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{n(n-1)(n-2)...(n-k+1)}{k(k-1)...1} \geq \left(\frac{n}{k}\right)^k \tag{1}$$

As an example, for an image has L = 256 gray-levels and the number of classes is M = 5 (the number of thresholds is M-1=4), the thresholds can only be placed in the interval [1 . . 255]. The number of possibilities times the objective function has to be calculated is $\binom{255}{4} = 17200610505$. For real time implementations, the exhaustive search is therefore not a solution and faster algorithms, which find the optimal thresholds without checking every possible placement, are needed.

To overcome this problem, several techniques have been proposed. Some of them are designed especially for computation acceleration of a specific objective function, such as the Otsu's function, while other techniques are designed to be used with a general purpose. Among the last category, we can find dichotomization techniques, iterative schemes, reduction strategies, and the meta-heuristic techniques. A review of these methods can be found in [3 - 4]. In the literature several criteria to regularize the segmentation problem are presented [17]. However, there is no single criterion able to regularize the segmentation problem for all kinds of images [9]. Then, in order to have a good segmentation on more kinds of images, some criteria are used simultaneously. To optimize simultaneously these criteria, the multi-objective optimization techniques are used in image thresholding problem.

Multi-objective Optimization "MO" (also known as multicriterion) extends the optimization theory by permitting several design objectives to be optimized simultaneously [9].

A MO problem is solved in a way similar to the conventional Single-Objective "SO" problem. The goal is to find a set of values for the design variables that simultaneously optimizes several objectives (or cost) functions. In general, the solution obtained through a separate optimization of each objective (i.e. SO optimization) does not represent a feasible solution of the multi-objective problem.

The use of multi-objective problem approaches has been found in image segmentation methods [1] with clustering, histogram thresholding methods. There is also an attempt of using multi-objective approaches for evaluation of image segmentation methods. As compared to multi-objective clustering approaches, there is limited research endeavour of using methods with *MO* in classical histogram thresholding methods. The use of *MO* in image segmentation with thresholding techniques has been dominated by Nakid et al. [9-10]. They have proposed to find the optimal thresholds that allow to optimize a set of criteria as the objective functions. The aim is to increase the information on the positions of the optimal thresholds to obtain the correct segmentation.

There are different approaches to solving multi-objective optimization problems [18] e.g., aggregating, population based non-pareto and pareto-based techniques. In the aggregating approaches, the multi-objective problem is transformed into a single-objective one. The population based non-pareto approaches involve the use of several subpopulations as single-objective optimizers. Then, the sub-populations somehow exchange information or recombine among them-selves aiming to produce trade-offs among the different solutions previously generated for the objectives that were separately optimized. The pareto-based approaches use leader selection techniques based on Pareto dominance.

The use and the development of metaheuristics-based multi-objective optimization techniques have significantly grown in the last years.

Particle Swarm Optimization "*PSO*" [7] is a bio-inspired optimization algorithm inspired by the choreography of a bird flock. *PSO* has been found to be a very successful optimization approach both in single-objective and in multi-objective problems [2]. Binary Particle Swarm Optimization "*BPSO*" [6] is a variant of *PSO*, which was adapted to search in binary space.

Vector Evaluated Particle Swarm Optimization "*VEPSO*" [11] is a technique in the population based non-pareto approach which is inspired on the Vector Evaluated Genetic Algorithm "*VEGA*" [15]. *VEPSO* is a multi-swarm variant of *PSO*, in which different swarms are maintained for the different objectives.

In this paper, we propose a new AutomaticThreshold based on Multi-objective Optimization "*ATMO*" method that combines the flexibility of multi-objective fitness functions with the power of *BPSO* for searching vast combinatorial state spaces. The idea was inspired from the segmentation problematic:

1) There does not exist any thresholding criterion that is capable to produce an optimal thresholding result for all images. The use of non-Pareto multi-objective optimization aims to obtain good thresholding results independently of the image.
2) Finding the "optimum" number of thresholds, in a whole gray-level range, is usually a challenge since it requires a priori knowledge. However, despite the amount of research in this area, the outcome is still unsatisfactory [3]. The use of the *BPSO* aims to optimize the number of classes.

The organization of the paper is as follows. In section 2, *BPSO* is reviewed. In Section 3, the proposed approach "*ATMO*" is presented. In Section 4, we illustrate the obtained results through the proposed image thresholding algorithm. Finally, Section 5 concludes the paper.

II. Binary particle swarm optimization

In BPSO, the component values of particle's position are restricted to the set {0, 1}. The velocity is interpreted as a probability to change a bit from 0 to 1, or from 1 to 0 when updating the position of particles. This can be done using a sigmoid function, defined as:

$$sig(x) = \frac{1}{1+e^{-x}} \tag{2}$$

Hence, the equation for updating positions is the probabilistic update equation, namely [6],

$$x_{i,j}(t+1) = \begin{cases} 0 & if \quad r_j(t) \geq sig(v_{i,j}(t+1)) \\ 1 & if \quad r_j(t) < sig(v_{i,j}(t+1)) \end{cases} \tag{3}$$

Where : $r_j(t) \sim U(0,1)$ is a random number between 0 and 1, $x_{i,j}$ is a component value of particle's position and $v_{i,j}$ is a component value of particle's velocity, that is updated as using the following equation:

$$\begin{aligned} v_{i,j}(t+1) = \omega v_{i,j}(t) + c_1 r_{1,j}(t)(y_{i,j}(t) - x_{i,j}(t)) + \\ c_2 r_{2,j}(t)(\hat{y}_j(t) - x_{i,j}(i)) \end{aligned} \tag{4}$$

Here, $x_i(t)$ is the current position of the particle. $v_i(t)$ is the current velocity of the particle. $y_i(t)$ is the personal best position of the particle; the best position visited so far by the particle *i*. $\hat{y}(t)$ is the global best position of the swarm; the best position visited so far by the entire swarm. ω is the inertia weight which serves as a memory of previous velocities; the inertia weight controls the impact of the previous velocity [20]. The cognitive component

$y_i(t) - x_i(t)$ represents the particle's own experience as to where the best solution is. The social component $\hat{y}(t) - x_i(t)$ represents the belief of the entire swarm as to where the best solution is. c_1 and c_2 are acceleration constants and $r_1(t)$, $r_2(t) \sim U(0,1)$, where $U(0,1)$ is a random number between 0 and 1 [20].

Velocity updates can also be clamped through a user defined maximum velocity "*Vmax*" which would prevent them from exploding, thereby causing premature convergence [20].

III. Proposed approach

To solve our multi-objective problem, we adapt the *VEPSO* [11] method that consists in using a set of sub-swarms $\boldsymbol{S} = \{Ss_1, \ldots, Ss_p, \ldots, Ss_{Nc}\}$; Ss_p is the sub-swarm p, and Nc is the number of sub-swarm and it represents the number of criteria used f_p; $p = 1..Nc$. Each sub-swarm Ss_p is valued by using an algorithm *BPSO* that searches the optimal thresholds, by optimizing one of objective functions of the problem (thresholding criteria) f_p , which uses the gray-level thresholds as parameters. It starts with large number initial thresholds (gray-level range of pixels in the given image). Then, these thresholds are dynamically refined to improve the value of the objective function. The different sub-swarms communicate between them through the exchange of their better position by using the uniformity measure U [16].

A. Segmentation Criteria

In this approach, we use three threshold criteria and a selection operator of the best thresholds.

The threshold criteria can be described as follows: let there be N pixels in a given image, with gray-level range over $[0..L]$ and n_i denote the occurrence of gray-level i, giving a probability of gray-level i as:

$$p_i = \frac{n_i}{N} \tag{5}$$

Entropy Criterion

The Kapur's method [5] is based on the entropy theory. It consists in the maximization of the sum of entropies for each class, as follows:

$$f(t_1, t_2, \ldots, t_k) = H_0 + H_1 + \ldots + H_k \tag{6}$$

Where:

$$H_0 = -\sum_{i=0}^{t_1-1} \frac{p_i}{\omega_0} \ln \frac{p_i}{\omega_0} \;,\; \omega_0 = \sum_{i=0}^{t_1-1} p_i$$

$$H_1 = -\sum_{i=t_1}^{t_2-1} \frac{p_i}{\omega_1} \ln \frac{p_i}{\omega_1} \;,\; \omega_1 = \sum_{i=t_1}^{t_2-1} p_i$$

$$H_2 = -\sum_{i=t_2}^{t_3-1} \frac{p_i}{\omega_2} \ln \frac{p_i}{\omega_2} \;,\; \omega_2 = \sum_{i=t_2}^{t_3-1} p_i$$

$$H_k = -\sum_{i=t_k}^{L} \frac{p_i}{\omega_k} \ln \frac{p_i}{\omega_k} \;,\; \omega_k = \sum_{i=t_k}^{L} p_i$$

The optimal segmentation threshold vector $(t_1^*, t_2^*, \ldots, t_k^*)$ is that maximizing the total entropy:

$$(t_1^*, t_2^*, \ldots, t_k^*) = Arg\ \max_{0<t_1<t_2<\ldots<L} f\ (t_1, t_2, \ldots, t_k)$$

Between-Class Variance Criterion

The Otsu's method [13] is based on the discriminant analysis. It consists in the maximization of the between-class variance of the thresholded image as:

$$\begin{aligned} f(t_1, t_2, \ldots, t_k) = {} & \omega_0\omega_1(\mu_0 - \mu_1)^2 + \omega_0\omega_2(\mu_0 - \mu_2)^2 + \\ & \omega_0\omega_3(\mu_0 - \mu_3)^2 + \ldots + \omega_0\omega_k(\mu_0 - \mu_k)^2 + \\ & \omega_1\omega_2(\mu_1 - \mu_2)^2 + \omega_1\omega_3(\mu_1 - \mu_3)^2 + \ldots + \\ & \omega_1\omega_k(\mu_1 - \mu_k)^2 + \ldots + \omega_{k-1}\omega_k(\mu_{k-1} - \mu_k)^2 \end{aligned} \tag{7}$$

Where:

$$\omega_n = \sum_{i=t_n}^{t_{n+1}-1} p_i \;,\; \mu_n = \sum_{i=t_n}^{t_{n+1}-1} \frac{i \times p_i}{\omega_n} \quad \text{and} \quad 0 \le n \le k$$

The optimal segmentation threshold vector $(t_1^*, t_2^*, \ldots, t_k^*)$ makes the total variance maximum:

$$(t_1^*, t_2^*, \ldots, t_k^*) = Arg\ \max_{0<t_1<t_2<\ldots<L} f\ (t_1, t_2, \ldots, t_k)$$

Minimum Error Criterion

Kittler and Illingworth [8] proposed a method that assumes a parametric form of the histogram, which is fit to a mixture of Gaussians, aiming at minimizing the error between that parametric distribution and the actual histogram as follows:

$$f(t_1, t_2, \ldots, t_k) = 1 + 2 \times \sum_{i=0}^{k} \left(\omega_i \left(\ln \sigma_i - \ln \omega_i \right) \right) \tag{8}$$

Where: $\omega_n = \sum_{i=t_n}^{t_{n+1}-1} p(i)$

$$\sigma^2{}_n = \sum_{i=t_n}^{t_{n+1}-1} \frac{p(i) \times (i - \mu_n)^2}{\omega_n} \quad \text{and}$$

$$\mu_n = \sum_{i=t_n}^{t_{n+1}-1} \frac{p(i) \times i}{\omega_n} \quad , \quad 0 \le n \le k$$

The optimal segmentation threshold vector $(t_1^*, t_2^*, \ldots, t_k^*)$ is given by: $(t_1^*, t_2^*, \ldots, t_k^*) = Arg\ \min_{0<t_1<t_2<\ldots<L} f\ (t_1, t_2, \ldots, t_k)$

Selection Criterion

The uniformity measure U is used for evaluating the quality of thresholded image and eventually to select the best thresholds. The uniformity measure U is widely mentioned in the literature [16]:

$$U = 1 - 2k \frac{\sum_{j=0}^{k} \sum_{i \in C_j} (g_i - m_j)^2}{N(g_{\max} - g_{\min})^2} \tag{9}$$

Where: k is the number of thresholds, C_j is the segmented Class j, g_i is the gray-level of pixel i, m_j is the mean of the gray-levels of those pixels in segmented region j, N is the number of total pixels in the given image, g_{max} is the maximal gray-level of the pixels in the given image, g_{min} is the minimal gray-level of the pixels in the given image.
The value of the uniformity measure is between 0 and 1. A higher value of uniformity means that the quality of the thresholded image is better.

B. Proposed Algorithm
For the ease of describing the proposed algorithm, let us first define the following symbols:

- A is an image contains N pixels with gray-levels from 0 to L-1.
- Nt is the maximum number of thresholds, $Nt=L-1$.
- $T = \{t_k,\ k=1\ldots Nt\}$ is the set of Nt thresholds.
- Nc is the number of thresholding criteria used.
- $S = \{Ss_1,\ldots, Ss_p,\ldots, Ss_{Nc}\}$ is the swarm of Nc sub-swarms of the same size s, such that: $Ss_p = \left\{X_1^p,\ldots,X_i^p,\ldots,X_s^p\right\}$ is the sub-swarm p ; p=1..Nc.
- $X_i^p = (x_{i,1}^p,\ldots,x_{i,j}^p,\ldots,x_{i,Nt}^p)$ **i**ndicates the particle i of sub-swarm p, with $x_{i,k}^p \in \{0,1\}$, for j = 1,…, Nt such that, if $x_{i,k}^p$ = 1 then the corresponding t_k in T has been chosen to be part of the solution proposed by X_i^p. Otherwise, if $x_{i,k}^p$ = 0 then the corresponding t_k in T is not part of the solution proposed by X_i^p.
- n_i^p is the number of thresholds represented by particle X_i^p of sub-swarm Ss_p, such that: $n_i^p = \sum_{k=1}^{Nt} x_{i,k}^p$ with $n_i^p \le Nt$.
- T_i^p is the multi-threshold solution represented by particle X_i^p of sub-swarm Ss_p, such that: $T_i^p = (t_k)\forall k : x_{i,k}^p = 1$, with $T_i^p \subseteq T$.
- n_{gbest}^p is the number of thresholds represented by the global best particle X_{gbest}^p of sub-swarm Ss_p; $X_{gbest}^p = (x_{gbest,1}^p,\ldots,x_{gbest,j}^p,\ldots,x_{gbest,Nt}^p)$, such that: $n_{gbest}^p = \sum_{k=1}^{N_t} x_{gbest,k}^p$, with $n_{gbest}^p \le Nt$.
- T_{gbest}^p is the multi-threshold solution represented by global best particle X_{gbest}^p of sub-swarm Ss_p, such that: $T_{gbest}^p = (t_k)\forall k : x_{gbest,k}^p = 1$ with $T_{gbest}^p \subseteq T$
- *Tgbest* is the threshold best combination, in the swarm S that maximizes the uniformity measure U.
- *Ngbest* is the number of thresholds in the threshold best combination *Tgbest* in the swarm.
- p_{ini} is a user-specified probability defined in [12], which is used to initialize a particle position, X_i^p, as follows:

$$x_{i,k}^p = \begin{cases} 0 & if\ r_k(t) \ge p_{ini} \\ 1 & if\ r_k(t) < p_{ini} \end{cases} \tag{10}$$

Where: $r_k(t) \sim \bigcup(0,1)$. Obviously a large value for p_{ini} results in selecting most of the thresholds in T.

The algorithm works as follows: T is a set of thresholds, initialised by the integer values from 1 to L-1, L is the maximum gray-level in image A. The sub-swarm Ss_p of particles is then randomly initialized. The *BPSO* algorithm is then applied to find the "best" set of thresholds, T_{gbest}^p , from T, which optimizes the objective function f_p according to T_{gbest}^p . The different sub-swarms Ss_p; p=1..Nc, communicate between them, through the exchange of their better position (T_{gbest}^p , n_{gbest}^p), to decide whether the threshold best combination *Tgbest*, in the swarm, that maximizes the uniformity measure U. The algorithm is then repeated using the new T = *Tgbest* and the new Nt=*Ngbest*. When the particles of sub-swarm are then re-initialized; the first Nt elements of a particle are initialized according to (10), and rests are initialized by 0. When the termination criteria are met, T=*Tgbest* will be the resulting "optimum" set of the threshold best combination, in the whole gray-level range.

The *ATMO* algorithm is summarized below:

1) Initialize $T = \{t_k\ ,\ \ t_k = 1\ldots L\text{-}1\}$ is the set of Nt thresholds, $Nt= L-1$, for *a*n image with gray-levels from *0* to *L*.
2) For each sub-swarm Ss_p in S, p=1…,Nc
   a) Initialize the particle X_i^p, with $x_{i,k}^p \sim U\{0,1\}$; i = 1,…, s and k = 1,…, Nt using (10).
   b) Randomly initialize the velocity V_i^p of each particle X_i^p in Ss_p, such that $v_{i,k}^p \in$ [-5,5], i= 1,…, s and k = 1,…, Nt. The range of [-5,5] was set empirically.
3) For each particle X_i^p in Ss_p ; i= 1,…, s and p = 1,…, Nc
   a) Calculate the objective function according f_p.
   b) Apply the binary PSO velocity and position update equations, using (4) and (3), to find the *lbest*

solution (n_i^p, T_i^p) and the *gbest* solution $(n_{gbest}^p, T_{gbest}^p)$.

4) Repeat steps 3) until the termination criteria are met.
5) Calculate the uniformity measure $U(T_{gbest}^p)$, *p=1…Nc*, using (9).
6) Calculate the threshold best combination, in the swarm *S*, "*Tgbest*" and the number of thresholds in the threshold best combination "*Ngbest*"
   a) *Tgbest*= T_{gbest}^p ; U(T_{gbest}^p) $= \underset{h=1...NC}{Max}(U(T_{gbest}^h))$
   b) *Ngbest* = n_{gbest}^p
7) Re-initialize the particle X_i^p of sub-swarm Ss_p, with $x_{i,k}^p$ ~ *U{0,1}* *i* = 1,…, *s* and *k* = 1,…, *Nt*, using (10), and $x_{i,k}^p$ = 0, for *k* = *Nt*+1…, *L*-1.
8) Repeat steps 3) through 7) until termination criteria are met.

IV. Experimental results

To evaluate the performance of the proposed *ATMO* algorithm, we present some experiments with different kinds of images (see Fig. 1.); synthetic, natural and biomedical images where the actual number of classes "$C_{Optimal}$" for synthetic image was known in advance and the optimal range for number of classes "$C_{Optimal}$", natural and biomedical images, were based on a visual analysis survey conducted by a group of people [19] [12]. The algorithms are coded in Matlab version 7 and are run on a computer having Intel Core 2 Duo processor (3 GHZ) and 2 GB memory.

The *ATMO* parameters were empirically set; TABLE I summarizes values of parameters of *ATMO* with which we got good results. These values are applied for the segmentation of all test images.

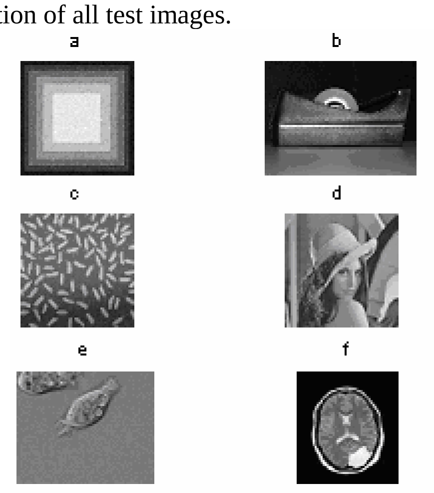

Fig. 1. Test images: (a) Square, (b) Tape, (c) Rice, (d) Lenna, (e) Cell and (f) MRI.

TABLE I
ATMO PARAMETERS

Parameters	Designation	values
p_{ini}	User-specified probability	0.75
Nc	Number of sub-swarm	3
s	Number of Particles in each sub-swarm	50
NI1	Number of Iterations; for step 4 of algorithm	50
NI 2	Number of Iterations; for step 8 of algorithm	10
ω	Inertia weight	0.72
c_1	Acceleration constant	1.8
c_2	Acceleration constant	1.4
Vmax	Maximum velocity	255

A. Evaluation of the Performance

In order to measure the performance of the segmentation, we used the criterion of Peak Signal to Noise Ratio "*PSNR*", which is used as a quality measurement between the original image and the thresholded image, the value is normally expressed in decibels (dB). The higher the *PSNR*, the better the quality of the thresholded, or reconstructed image. The *PSNR* is defined as:

$$PSNR = 20\log_{10}\left(\frac{255}{RMSE}\right) \tag{11}$$

Where *RMSE* is the root mean-squared error, which is defined as:

$$RMSE = \sqrt{\frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(I(i,j) - \hat{I}(i,j)\right)^2}$$

Where *I* and $\hat{I}$ are the original and the thresholded images, and *M*×*N* are the dimensions of the image.

Since the proposed algorithm is of stochastic type, its performance cannot be judged by the result of a single run. 50 different runs have been carried out, for each image, to reach valid conclusion about the performance of the algorithm. The dynamic behavior of the proposed method is also studied (see TABLE II) by calculating the mean and standard deviation concerning: the number of classes (C_{ATMO}), the uniformity measure *U* and the *PSNR* value. The higher standard deviation shows that the results of the experiment are unstable.

TABLE II
COMPUTATIONAL RESULT OF ATMO

Images	Size (in number of pixels)	$C_{Optimal}$	C_{ATMO}	*U*	*PSNR*
Square	250x250	6	6.22± 0.118	0.975± 0.026	20.91± 0.080
Tape	384x512	[3,4]	4.36± 0.122	0.912± 0.115	13.25± 0.112
Rice	256x256	[2,3]	3.51± 0.041	0.963± 0.024	10.31± 0.141
Lenna	256x256	[5,10]	6.64 ± 0.050	0.953 0.140	21.76± 0.123
Cell	159x 191	[2,3]	2.37± 0.025	0.988± 0.180	8.98± 0.016
MRI	178x 158	[4,8]	5.18 ± 0.287	0.896± 0.251	16.53± 0.125

From these results (TABLE II), it is clear that:

- The proposed *ATMO* method has small standard deviation values (for the C_{ATMO}, *U* and *PSNR*), for all test images, showing the stability of the proposed algorithm.
- The proposed *ATMO* algorithm found a correct number of classes, for the synthetic image, and a solution within the optimal range, for the natural and biomedical images.

B. Analysis of the Thresholded Results

In order to show the quality of the thresholded results in segmentation based on the simultaneous optimization of some criteria and their results when these criteria used separately, a comparison of *PSNR* values for the proposed *ATMO* method and Otsu's, Kapur's and Kittler's methods with exhaustive search is presented in TABLE III. The proposed approach automatically determines the "optimum" number of the thresholds as well as the adequate threshold values. However, the automatic determination of the threshold number still leaves a problem of the Otsu's, Kapur's and Kittler's methods. For this reason, the Otsu's, Kapur's and Kittler's methods are applied by varying the number of thresholds, then the optimal threshold number, which makes the objective function optimal (maximum for Otsu's or Kapur's functions and minimum for Kittler's function), is determined. The optimal threshold values obtained by these methods are shown in TABLE IV.

TABLE III
COMPARISON OF PSNR VALUES FOR METHODS UNDER EVALUTION

Images	Otsu	Kapur	Kittler	*ATMO*
Square	20.92	20.48	20.72	20.99
Tape	12.96	12.87	13.33	13.38
Rice	10.27	9.89	10.16	10.42
Lenna	21.80	20.76	20.55	21.88
Cell	8.97	8.90	8.92	8.99
IRM	15.93	16.62	15.84	16.76

TABLE IV
OPTIMAL THRESHOLD VALUES OBTAINED BY VARIOUS METHODS

Images	Otsu	Kapur	Kittler	*ATMO*
Square	35-80-126-170-212	38-76-122-160-214	52-100-127-168-209-	39-84-127-170-212
Tape	38-76-126	54-133-180	71-114-208	40-90-124
Rice	108-180	95-178	99-187	114-180
Lenna	43-79-104-144-177	43-80-104-141-177	35-74-100-152-186	69-110-135-166-197
Cell	93	124	111	98
IRM	23-70-90-173	23-71-103-181	25-70-110-176	23-70-101-176

From the results presented in TABLE III, it can be seen that, for almost all the images, the proposed *ATMO* algorithm gives the highest value of PSNR value. This performance is due to the inclusion of several criteria in the segmentation process.

For a visual understanding of the thresholding in the segmentation by the *ATMO* algorithm, the test images are thresholded and are shown in Fig. 2. The gray-level histograms of these images with threshold values are illustrated in Fig. 3.

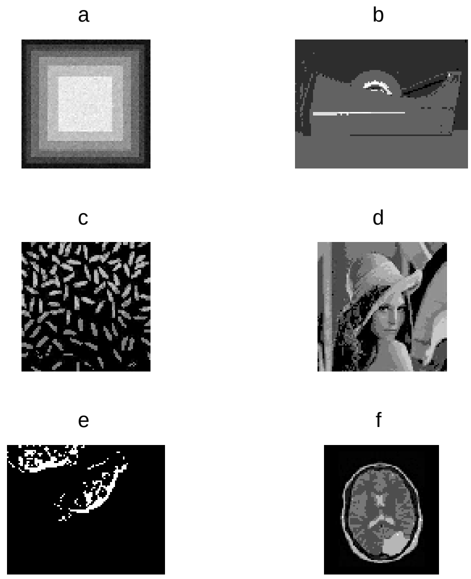

Fig. 2. Thresholded images: (a) Square, (b) Tape, (c) Rice, (d) Lenna, (e) Cell and (f) MRI.

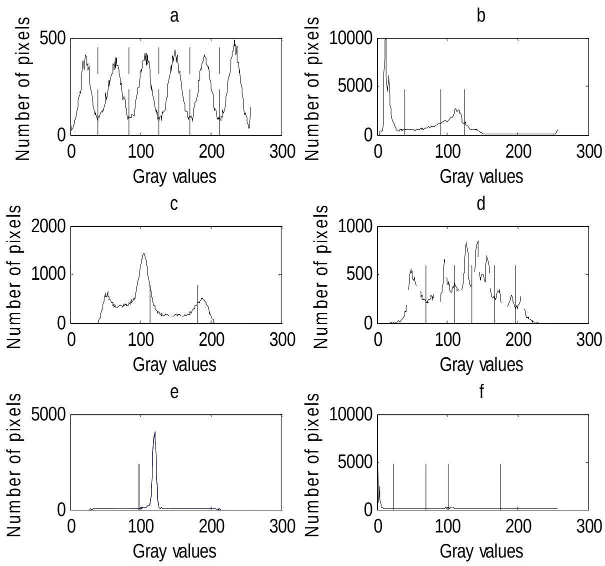

Fig. 3. Gray-level histograms with threshold values: (a) Square, (b) Tape, (c) Rice, (d) Lenna, (e) Cell and (f) MRI.

IV. CONCLUSION

In this paper, we have presented a non-supervised thresholding approach based on non-Pareto multi-objective optimization and particle swarm optimization, this approach enables to determinate the "optimum" number of the thresholds and simultaneously the optimal thresholds of three criteria: the between-class variances criterion, the minimum error criterion and the entropy criterion**.** The proposed method is validated by illustrative examples; comparison with the exhaustive search Otsu's, Kittler's and Kapur's methods showed the robustness of the proposed method, and its non dependence towards the kind of the image to be segmented, and also showed that image segmentation based on the simultaneous optimization of some criteria gives satisfactory results and increases the ability to apply one same technique to a wide variety of images.

In the future work, we will interest to the dynamic optimization problems. We will improve this approach by adding other segmentation criteria to treat image sequences.

ACKNOWLEDGMENT

This work was sponsored, in part, by the ANDRU (L'Agence Nationale pour le Développement de la Recherche Universitaire): Under the contract N° 8u07907.

REFERENCES

[1] C.W. Bong and R. Mandava, "Multiobjective Optimization Approaches in Image Segmentation - The Directions and Challenges," *Int. J. Advance Soft Comput. Appl*, Vol. 2, No. 1, pp. 40-64, 2010.

[2] C.A. Coello Coello, "Handling Multiple Objectives with Particle Swarm Optimization." IEEE Transactions on Evolutionary Computation, IEEE, Piscataway, NJ, 8 (3) 256-279, 2004.

[3] L. Djerou., H. Dehimi, N. Khelil and M. Batouche, "Using the BPSO algorithm in image segmentation for dynamic thresholding." in the proceeding of the International Conference on Bio-Inspired Computing: Theories and Applications, IEEE , pp.402-407, 2009.

[4] K. Hammouche, M. Diaf and P. Siarry, "A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem." *Journal of Engineering Applications of Artificial Intelligence***,** Vol. 23, 5, pp. 676-688, 2010.

[5] J.N. Kapur, P.K. Sahoo and A.K.C. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram." *Journal of Computer Vision Graphics Image Processing*, 29, pp. 273-285, 1985.

[6] J. Kennedy and R. Eberhart, "A Discrete Binary Version of the Particle Swarm Algorithm." *Proceedings of the Conference on Systems*, Man, and Cybernetics, pp. 4104-4109, 1997.

[7] J. Kennedy and R. Eberhart, "Particle Swarm Optimization." *Proceedings of IEEE International Conference on Neural Networks*, Perth, Australia, vol. 4, pp.1942-1948, 1995.

[8] J. Kittler and J. Illingworth, "Minimum Error Thresholding." *Pattern Recognition*, 19(1), pp 41-47, 1986.

[9] A., Nakib, H. Oulhadj and P. Siarry, "Image thresholding based on Pareto multiobjective optimization." *Engineering Applications of Artificial Intelligence*, 23 , pp 313-320, 2010.

[10] A. Nakib, H. Oulhadj and P. Siarry, "Image histogram thresholding based on multiobjective optimization. " *Signal Processing* 87, pp.2516-2534, 2007.

[11] E. Konstantinos Parsopoulos, K. Dimitris Tasoulis, and Michael N. Vrahatis. "Multiobjective optimization using parallel vector evaluated particle swarm optimization." In Proceedings of the IASTED International Conference on Artificial Intelligence and Applications (AIA 2004), volume 2, pages 823–828, Innsbruck, Austria, February ACTA Press, 2004.

[12] M. Omran, A. Salman and A. Engelbrecht, "Dynamic clustering using particle swarm optimization with application in image segmentation." *Pattern Analysis and Applications Journal*, vol. 8,4, 2006.J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, pp.68-73, 1892.

[13] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms." *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 9, 1, pp. 62-66, 1979.

[14] N.R. Pal, and S.K. Pal, "A review on image segmentation techniques." *Pattern Recognition* 9(26), pp. 1277-1294, 1993.

[15] J.D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithm." In the proceedings of the first Int. Conf. on Genetic Algorithms, Pittsburgh (USA), pp. 93-100, July 1985.

[16] P.K. Sahoo, S. Soltani, and A.K.C. Wong, " A survey of thresholding techniques. " *Computer Vision, Graphics, and Image Processing*, Vol. 41, pp. 233-260, 1988.

[17] M. Sezgin, B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation." Journal. of Electronic Imaging. Vol. 131, pp. 146-165, 2004.

[18] M. Reyes-Sierra, C.A.C. Coello, "Multi-objective Particle Swarm Optimizers: A survey of the State-of-the-Art.", Int. Journal of Comput. Intelligence Research, Vol. 2, 3, pp. 287-308, 2006.

[19] R.H. Turi, *Clustering-based color image segmentation*. Ph.D Thesis. Monash University, Australia. 2001.

[20] F. Van den Bergh, *An Analysis of Particle Swarm Optimizers*, PhD Thesis, Department of Computer Science, University of Pretoria 2002.

L. Djerou is a Doctor at computer Science Departement, University Med Khider at Biskra, Algeria. She holds a Graduate Diploma, Engineer in Informatics and Post Graduate Diploma: Magister. She obtained her Doctorat from the University of Biskra, Algeria. Her domain of interest is Emergent Computing and Complex Systems, Image Processing and Computer Vision, Metaheuristics and Nature Inspired Computing.

N. Khelil is a Doctor at Mathematics Department, University Med Khider at Biskra, Algeria. He holds a Graduate Diploma, DES in Mathematics and Post Graduate Diploma: Magister. He obtained his Doctorat from the University of Biskra, Algeria. His domain of interest is Numerical Analysis, Applied Mathematics, Metaheuristics and Nature Inspired Computing.

N.E.H. Dehimi is a Doctor degree student at the Department of Mathematics and Computer Science of the University of Oum El Bouaghi in Algeria. Her area of interest is Artificial Intelligence, Image processing and Computer Vision and the Simulation of population dynamics**.**

M. Batouche is a Professor at computer Science Departement, University of Constantine in Algeria. He obtained his PhD from the National Polytechnic Institute of Lorraine (INPL), University of Nancy, France. His domain of interest is Emergent Computing and Complex Systems, Image Processing and Computer Vision, Metaheuristics and Nature Inspired Computing.