A new Optimization-Based Image Segmentation method By Particle Swarm Optimization

Fahd M. A. Mohsen
Computer & Mathematics Dept.
Faculty of Science, Ibb University
Ibb, Yemen

Mohiy M. Hadhoud
Faculty of Computers & Information
Minufiya University
Shebin Elkom, Egypt

Khalid Amin
Faculty of Computers & Information
Minufiya University
Shebin Elkom, Egypt

Abstract— **This paper proposes a new multilevel thresholding method segmenting images based on particle swarm optimization (PSO). In the proposed method, the thresholding problem is treated as an optimization problem, and solved by using the principle of PSO. The algorithm of PSO is used to find the best values of thresholds that can give us an appropriate partition for a target image according to a fitness function. in this paper, a new quantitative evaluation function is proposed based on the information theory. The new evaluation function is used as an objective function for the algorithm of PSO in the proposed method. Because quantitative evaluation functions deal with segmented images as a set of regions, the target image is divided into a set of regions and not to a set of classes during the different stages of our method (where a region is a group of connected pixels having the same range of gray levels). The proposed method has been tested on different images, and the experimental results demonstrate its effectiveness.**

Keywords- Thresholding-based segmentation; Particle swarm optimization; Quantitative image segmentation evaluation.

I. Introduction

Recently, swarm intelligence (SI) has been applied in numerous fields including optimization [1]. One of SI methods performing well in solving optimization problems is particle swarm optimization (PSO). PSO is a stochastic search method that was developed in 1995 [1] based on the sociological behavior of bird flocking. The algorithm of PSO is easy to implement and has been successfully applied to solve a wide range of optimization problems in many fields such as image processing fields including image segmentation. Image segmentation is a low-level image processing task aiming at partitioning an image into homogeneous regions [2]. The result of image segmentation is a set of regions that collectively cover the entire image, or a set of contours extracted from the image. All of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture [3]. Image segmentation methods have been classified into numerous categories of which region and thersholding based segmentations.

Thresholding based segmentation is definitely one of the most popular and effective approaches used in image segmentation [5]. Over the years a wide range of thresholding techniques has been developed and considerable research continues nowadays. Marcello et al. in [6] has classified the thresholding techniques into two groups, local and global thresholding techniques, and the global ones again classified according to the information they exploit into histogram shape-based methods, clustering-based methods, entropy-based methods, object attribute-based methods and spatial methods. All the thresholding techniques involve a bi-level thresholding and a multilevel thresholding. The main objective of thresholding is to determine a threshold for bi-level thresholding or several thresholds for multilevel thresholding giving a suitable classification for pixels in an image. The simplest problem will be a bi-level thresholding one, where only one threshold, which separates the pixels into only two classes, is selected and the image able to be segmented by thresholding it at this value. This can facilitate to generate a binary image where all pixels having gray levels higher than the threshold are assigned to one class and pixels having gray levels lower than the threshold are assigned to another class. However the problem gets more and more complex when we try to achieve segmentation with greater detail by multilevel thresholding. Then the image segmentation problem becomes a multi-class classification problem where, based on the determined thresholds, pixels having gray levels within a specified range are grouped into one class. Determination of appropriate threshold values, that can segment the image efficiently, is the most important task involved in thresholding techniques. Over the years many method has been developed to solve this problem [5, 7]. The determination of appropriate thresholds is still the most difficult task in the thresholding techniques and it is still a challenge and a hot research topic for the researchers.

Now, the PSO technique has been used to solve the problem of thresholding based segmentation. Zahara et al. in [8] combined a hybrid Nelder–Mead simplex search method and the PSO technique to solve the objective functions of Gaussian curve fitting and the Otsu's method. This combination is applied to image thresholding with multi-modal histograms. In [9] a multilevel threshold selection based on PSO was proposed. The PSO technique was used to find near-optimal thresholds by minimizing the cross entropy between the original image and its thresholded version. Maitra et al. [10] proposed a new thresholding algorithm for histogram-based image segmentation using a hybrid cooperative-comprehensive learning based on the PSO algorithm. In their algorithm, the entropy criterion has been used as a fitness function and near optimal threshold values have been searched out through maximizing the normal entropy function. Zhiwei et al. in [11] presented a new method to select image threshold automatically based on PSO. They employed the PSO technique to deal with the criteria of Otsu's for Bi-level thresholding image segmentation and Wei et al. in [12] combined the PSO technique with Otsu's for multilevel thresholding image segmentation. Sathya et al. [13] presented a new histogram-based method for multilevel thresholding segmentation based on PSO. This method used Tsallis entropy as a fitness function. The PSO algorithm is used to find the near optimal threshold values that maximize the Tsallis objective function. Hongmei et al. in [14] proposed a multilevel thresholding method segmenting images based on the maximum entropy and an improved PSO. First, the parameters and the evolutionary process of the basic PSO have been improved, and then the combinations of near optimal thresholds are searched out by combining the improved PSO with maximum entropy.

The goal of this paper is to propose a new multilevel thresholding image segmentation method based on the idea of PSO. The PSO algorithm will try to find the near-optimal threshold values that can give us a near optimal segmentation for a target image according to a fitness function. A new quantitative evaluation function will be proposed and used as a fitness function for the algorithm of PSO. The new function manipulates with a segmented image as a set of regions not a set of classes. So, in the prooposed method, a given image is segmented into a set of regions, where a region is a group of connected pixels having a specific range of gray levels.

This paper is organized as follows. Section 2 gives an overview of the PSO method. Section 3.A gives an overview of image segmentation evaluation and section 3.B proposed a new quantitative evaluation function. Section 4 presents the proposed method. In Section 5, the experimental results are presented, and, finally, the conclusions are stated in Section 6.

II. Particle Swarm Optimization

Particle swarm optimization (PSO) is a population-based optimization algorithm modeled after the simulation of social behavior of birds in a flock [6, 15]. The algorithm of PSO is initialized with a group of random particles and then searches for optima by updating generations. Each particle is flown through the search space having its position adjusted based on its distance from its own personal best position and the distance from the best particle of the swarm. The performance of each particle, i.e. how close the particle is from the global optimum, is measured using a fitness function which depends on the optimization problem.

Each particle, i, flies through an n–dimensional search space, R^n , and maintains the following information:

- x_i, the current position of i^{th} particle (x - vector),
- p_i, the personal best position of i^{th} particle (p - vector), and
- v_i, the current velocity of i^{th} particle i (v - vector).

The personal best position associated with a particle, i, is the best position that the particle has visited so far. If *f* denotes the fitness function, then the personal best of i at a time step t is updated as:

$$p_i(t+1) = \begin{cases} p_i(t) & \text{if} \quad f(x_i(t+1)) \geq f(p_i(t)) \\ x_i(t+1) & \text{if} \quad f(x_i(t+1)) < f(p_i(t)) \end{cases} \qquad (1)$$

If the position of the global best particle is denoted by *gbest* , then :

$$gbest \in \{ p_1(t),\ p_2(t), \ldots, p_m(t) \} = \min\{ f(p_1(t)),\ f(p_2(t)), \ldots, f(p_m(t)) \} \qquad (2)$$

The velocity updates are calculated as a linear combination of position and velocity vectors. Thus, the velocity of particle i is updated using equation (3) and the position of particle i is updated using equation (4).

$$v_i(t+1) = w\, v_i(t) + c_1 r_1(p_i(t) - x_i(t)) + c_2 r_2 (gbest - x_i(t)) \qquad (3)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \qquad (4)$$

In the formula, w is the inertia weight [16], c1 and c2 are the acceleration constants, r_1 and r_2 are random numbers in the range [0,1] and Vi must be in the range [$-V_{max}$, V_{max}], where V_{max} is the maximum velocity.

III. Image Segmentation Evaluation

A. Related Work

In order to measure the performance of image segmentation methods without human interactions, we need an evaluation criterion. There are many image segmentation evaluation functions have been presented in the literature. Those functions have been divided into numerous types including quantitative evaluation measures [17, 18, 19]. Borsotti et al. in [18] empirically proposed a quantitative evaluation function, Q , such as:

$$Q(I) = \frac{1}{1000 \times Na} \sqrt{n} \sum_{j=1}^{n} \left[\frac{e_j^2}{1 + \log L(R_j)} + \left(\frac{M(L(R_j))}{L(R_j)} \right)^2 \right] \qquad (5)$$

Where I is a given image, Na is the number of pixels in I, n is the number of regions, R_j denotes to j^{th} region, $L(R_j)$ is the number of pixels in j^{th} region and e_j is the color error of j^{th} region. The first term of Equation 5 is a normalization factor and the second term penalizes results with too many regions. The last term in Q function penalizes simultaneously regions with big color error and small regions.

Zhang et al. [19] proposed an information theoretic approach for segmentation evaluation, E function, based on entropy theory. Given a segmented image, they define V_j as the set of all possible values for the luminance in the region j and let $L_m(R_j)$ denote the number of pixels in the region R_j that have luminance of m. The entropy for the j^{th} region is defined such as:

$$H_v(R_j) = -\sum_{m\in V_j} \frac{L_m(R_j)}{L(R_j)} \log \frac{L_m(R_j)}{L(R_j)} \tag{6}$$

Next, they define the expected region entropy of the segmented image, $H_r(I)$, such as:

$$H_r(I) = -\sum_{j=1}^{n}\left(\frac{L(R_j)}{Na}\right)H_v(R_j) \tag{7}$$

As it is seen, $H_r(I)$ is simply the expected entropy across all regions where each regions has weight (or probability) proportional to its area. The expected region entropy serves in a similar capacity to the term involving the squared color error used in Q function. The expected region entropy must be combined with another term or factor that penalizes segmentations having a large number of regions since there would otherwise be a strong bias to over-segment the image. Instead of penalizing the expected region entropy, they instead introduced the layout entropy, $H_l(I)$, such as:

$$H_l(I) = -\sum_{j=1}^{n} \frac{L(R_j)}{Na}\log\frac{L(R_j)}{Na} \tag{8}$$

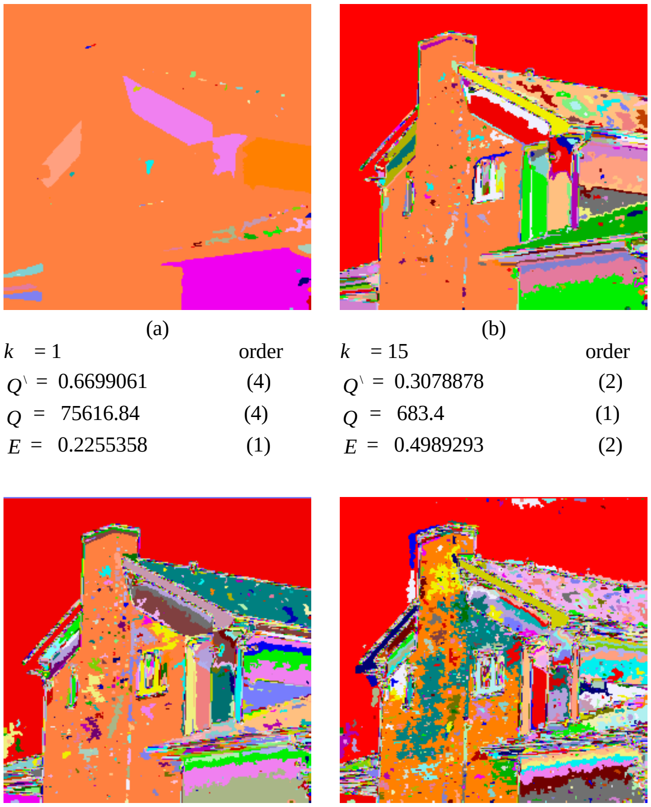
Finally, they define their evaluation function, E, where a lower entropy value means a better segmentation, as follows:

$$E = H_l(I) + H_v(I) \tag{9}$$

For more study about the image segmentation evaluation methods, the reader is advised to refer to Zhang [20] and Zhang et al. [21]. They have explained a lot of evaluation methods with detailed information.

B. A new Quantitative Evaluation Function

This section presents a new quantitative evaluation function, $Q^{\backslash}$, based on the information theory. The new function is equivalent to Q function. The main problem of Q function is it was designed empirically. This makes Q function produces less balancing between the homogeneities and number of regions in a segmented image. Figure 1 illustrates this problem. Figures 1(a, b, c, d) show the segmentation results produced through multithresolding the image of House using k=1, k=10, k=15 and k=30 respectively (where k represents the number of thresholds – Note that the gray levels were divided into equal interval according to k). Figure 1(a) shows a segmented image with many merged regions. Figure 1(b) shows a segmented image with fewer details. Figure 1(c) shows a segmented image with more details reflecting more homogenous regions. Figure 1(d) shows the segmentation with many noises. Each figure has its evaluation values computed using $Q^{\backslash}$, Q and E functions respectively. From those results, we can see that the differences between the evaluations of Q function are very large especially between the evaluations of Figure 1(b) and Figure 1(d). These differences don't reflect the visual judgment of the segmentation quality. This is due to the fact that Q function produces less balancing between the number and homogeneities of regions where it was designed empirically. The results shown in Figure 1 illustrate that the evaluations obtained by $Q^{\backslash}$ function, which will be described bellow, are the same as that obtained by Q, but the order and scores given by $Q^{\backslash}$ reflect the visual evaluation of segmentation quality. This is due to $Q^{\backslash}$ function produces more balancing between the number of regions and their homogeneities. To show that, see the difference between the evaluations of $Q^{\backslash}$ function shown in Figure 1(b) and Figure 1(d). The balancing of $Q^{\backslash}$ isn't the optimal but it is acceptable so far.



(a)

k	= 1	order
$Q^{\backslash}$	= 0.6699061	(4)
Q	= 75616.84	(4)
E	= 0.2255358	(1)

(b)

k	= 15	order
$Q^{\backslash}$	= 0.3078878	(2)
Q	= 683.4	(1)
E	= 0.4989293	(2)

(c)

k	= 10	order
$Q^{\backslash}$	= 0.2663163	(1)
Q	= 3425.93	(2)
E	= 0.5604761	(3)

(d)

k	= 30	order
$Q^{\backslash}$	= 0.4008721	(3)
Q	= 54283.53	(3)
E	= 0.6775197	(4)

Figure 1: The segmentation results of House image though multi-thresholding it into regions using:
(a). *k*=1. (b). *k*=10. (c). *k*=28. (d). *k*=30.

The orders of segmented images produced by Q & $Q^{\backslash}$ functions are both acceptable but the order of $Q^{\backslash}$ function is the best. The order of the segmented images according to E function is k=1, k=10, k=15, k=30. This order gives Figure 1(a) the best segmentation. This is due to the fact that E function performs well in evaluating segmented images having explicit objects (real-world objects) [21] – Note: $Q^{\backslash}$, Q and E functions are computed according to the gray levels and the values of $Q^{\backslash}$ and E are scaled to be in the range [0,10].

In describing the $Q^{\backslash}$ evaluation function, considered here, we will use the following notations. Let I be a given image and let Na be the number of pixels in the given image. We will use

R_j to denote jth region, and use $L(R_j)$ to denote the area of jth region (as measured by the number of pixels). The homogeneity of the region R_j is computed according to the color errors. Let $e(px_{i,j})$ be the color error of ith pixel in jth region, we define $e(px_{i,j})$ according to the mean of a region as:

$$e(px_{i,j}) = mean_j - g(px_{i,j}) \tag{10}$$

Where $g(px_{i,j})$ is the gray level of $px_{i,j}$ and $mean_j$ is the mean of jth region computed using the gray levels of pixels being in this region. So, we will define a new group, Re_j , containing the integers of color errors of the pixels included in jth region and it is defined such as:

$$\mathrm{Re}_j = \{\mathrm{int}(e(px_{i,j})) : px_{i,j} \in R_j\} \tag{11}$$

Let Re_j is split into two groups Re'_j & Re''_j , where Re'_j contains the integers of color errors of the near similar pixels (homogeneous) and Re''_j contains the integers of the rest color errors, we define Re'_j such as:

$$\mathrm{Re}'_j = \{\mathrm{int}(e(px_{i,j})) : px_{i,j} \in R_j \;\&\; abs(e(px_{i,j})) < d\} \tag{12}$$

Where d is a constant value. Empirically, we found that the proper value of d falls in the range [0, 0.3] and the value d=0.15 gives us good evaluation results. Also, we found that the value of d can be detected mathematically according to the homogeneity of a given image - we will illustrate this principle later. Thus, d divides the pixels of regions in a segmented image into near similar and non-similar pixels according to the color errors. Consequently, we have I'_e & I''_e. If n denotes to the number of regions in a segmented image, then:

$$I'_e = \bigcup_{i=1}^{n} R'e_i \tag{13}$$

$$I'_e = \bigcup_{i=1}^{n} R''e_i \tag{14}$$

$Q^{\backslash}$ function as E function has two components, $H_l^{\backslash}$ & H_e , where $H_l^{\backslash}$ represents the layout entropy and H_e represents the entropy of color errors. The working principle of $Q^{\backslash}$ function is a lower entropy value means a better segmentation result and vise versa, and it is computed such as:

$$Q^{\backslash} = w_1 H_l^{\backslash}(I) + w_2 H_e(I)\} \tag{15}$$

Where, w_1 & w_2 are weighting parameters. Empirically, we found that $Q^{\backslash}$ function gives us a good evaluation if the value of w_1 is greater than the value of w_2. Throughout this work we will set w_1 = 0.55 and w_2 = 0.45. The users can tailor these parameters according to their objectives. The first term, $H_l^{\backslash}$, represents the layout entropy, and it is computed such as:

$$H_l^{\backslash} = -\sum_{j=1}^{n} \frac{L(R'e_j)}{Na} \log_2 \frac{L(R'e_j)}{Na} \tag{16}$$

From the information coding theory, the above function indicates the number of bits needed to specify a region id in a segmented image according to the number of near similar pixels in different regions. The second term, H_e , represents the entropy of color errors in the overall segmented image and it is computed such as:

$$H_e = -\sum_{z=\min_e}^{\max_e} \frac{L(z)}{Na} \log_2 \frac{L(z)}{Na} \tag{17}$$

Where, $\min_e$ and $\max_e$ represent the minimum and maximum color errors in I''_e respectively. From the information coding theory, H_e indicates the number of bits needed to encode the color errors of non-similar pixels. The entropy of color errors serves in a similar way to the term involving the squared color error used in Q function.

The two entropies of $H_l^{\backslash}$ & H_e components are computed according to the number of pixels in the overall image, Na. This make $H_l^{\backslash}$ and H_e are complements to each other making $Q^{\backslash}$ function gives more balancing between the number of regions and their homogeneities. The value of $H_l^{\backslash}$ is increased if the number of regions in a segmented image increases and vise versa and the value of H_e is increased if the homogeneities of regions decrease and vise versa. In the case of a segmented image with one pixel per region, the first term is maximized to a very large value and the second term is minimized to zero. This means that $Q^{\backslash}$ function gives a large value in the case of non-segmented image.

- *d* detection

The value of d splits the pixels in a region into near-similar (homogeneous) and non-similar (non-homogeneous) so this value can be extracted mathematically according to the homogeneity of the overall image, $H(I)$. Entropy theory is one of the ways used to measure the homogeneity of images. Let $g_{\max}$ is the maximum gray level in a given image, I, then the homogeneity of I is computed depending on the image entropy as follows:

$$H(I) = -\sum_{z=0}^{g_{\max}} \frac{L(z)}{Na} \log_2 \frac{L(z)}{Na} \tag{18}$$

In the case of images having 256 gray levels, the number of bits needed to represent a pixel is 8 bits per pixel. So if the value of $H(I)$ is less than 4 bits per pixels, this means that the pixels in the given image are near-similar. In this case, the value of d is equaled to *zero*. In other cases the value of d is equaled to ($H(I)$ – 4). Empirically, we found that the proper value of d falls in the range [0, 0.3]. So the value of d after it has been computed is scaled to be in that range.

IV. A NEW IMAGE SEGMENTATION METHOD BASED ON PSO

This section develops a new multilevel thresholding method segmenting images using the principle of PSO. The new method will be named PSOTH. The algorithm of PSO, in PSOTH, tries to find near-optimal values of thresholds that can give us a near-optimal segmentation. Consequently, the PSO algorithm initializes a random swarm of m particles, where each one has its k thresholds, and flies them on a search space to look after the target partition according to a fitness function. The PSO algorithm uses $Q^\backslash$ function as a fitness function and the gray levels {g_{min}, ..., g_{max}} as a search space, where g_{min} and g_{max} are the minimum and maximum gray levels in a given image respectively.

A. PSO Representation

One of the key issues in designing a successful PSO algorithm is the representation step, i.e. finding a suitable mapping between a problem and PSO particles. Figure 2 shows the PSO representation that is used in PSOTH. As it is shown, PSO has the following information:

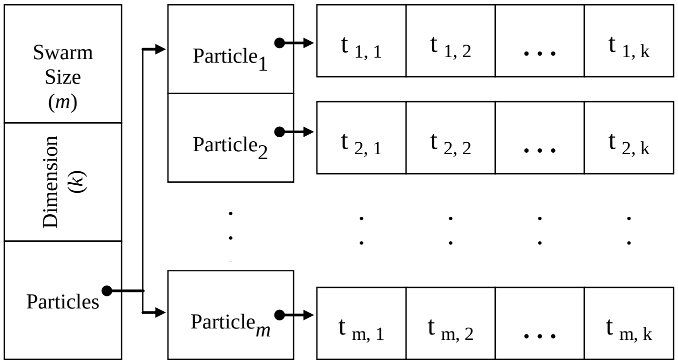


Figure 2: PSO Representation in the PSOTH Method

In PSO of PSOTH method, a single particle represents k thresholds. That is, each vector x_i is constructed such as x_i = ($t_{i,1}$, . . . , $t_{i,j}$, . . . , $t_{i,k}$) where $t_{i,j}$ refers to the jth threshold value of the ith particle. Therefore, in the PSOTH method, after applying the thresholding to each particle, a swarm represents a number of segmented images.

B. Fitness Function

In PSOTH, the algorithm of PSO uses the $Q^\backslash$ evaluation function presented above as a fitness function. The idea behind using $Q^\backslash$ function can be formulated as: the smaller the value of $Q^\backslash$, the better is the segmentation result. Consequently, The PSO algorithm, in the PSOTH method, looks after the threshold values that minimize the value of $Q^\backslash$. It should be noted that $Q^\backslash$ function manipulates with a segmented image as a set of regions. So, in PSOTH, a given image is thresholded into a set of regions (where a region is a group of connected pixels located in the same rang of gray levels). Let there be L gray levels in the given image and these gray levels are in the range M={g_{min}, …, g_{max}}, So the PSO algorithm tries to obtain a near optimal *k*-dimensional vector {t_1, t_2, ..., t_k} $\subset$ M that thresholds the given image into regions which can minimize the value of $Q^\backslash$ function shown in Equation 15.

C. PSOTH Algorithm

Firstly, the algorithm of PSOTH starts by applying (3x3) low pass filter to remove a noise from a target image, after that, every particle in the swarm is initialized randomly to contain k thresholds. Secondly, the target image is thresholed and divided into regions using the threshold values of each particle separately, and the fitness function, which is the $Q^\backslash$ evaluation function, is calculated for each particle. Once the fitness values have been found the global best solution, *gbest*, is computed and the updates of PSO velocities and vectors are then done. This procedure is repeated until the number of iterations has been satisfied. The algorithm of PSOTH is summarized in Algorithm 1.

```
Algorithm 1: PSOTH Algorithm
1- Do (3 x 3) Low pass filter.
2- Initialize randomly each particle to contain k threshold values.
3- For each particle, i, Sort the threshold values.          // Optional
3- Repeat the following Until the number of iterations has been
   satisfied
   (a) For each particle i :
       (i) Threshold the target image into regions using the
           threshold values of particle i.
       (ii) Compute the fitness of particle i by using $Q^\backslash$
           function.
   (b) Find the global best solution (gbest).
   (c) For each particle i, Update the threshold values.
4- Fix the gbest as a best solution.
5- Merge small regions of the best solution.
```

The algorithm initializes the vectors of each particle randomly. The threshold values of each particle are initialized in the range [g_{min}, g_{max}]. The thresholds are sorted in our method optionally. The sorting step is added to increase the symmetric between the thresholds in different particles.

To threshold and group a target image into regions, the principle of normal region based segmentation is applied as follows. The image is scanned from left to right and from top to bottom and the current pixel, px, is used as a seed point growing with its neighboring pixels according to a connectivity and a similarity condition. Here, the 4-connectivty is used and the similarity condition is formulated using the range of gray levels as follows. If g(px) falls in a rang [$t_{i,j}$, $t_{i,j+1}$] then a neighbored pixel N(px) append to the region of px, if g(N(px)) also falls in the rang [$t_{i,j}$, $t_{i,j+1}$]. Where, [$t_{i,j}$, $t_{i,j+1}$] is the rang of gray levels specified by two threshold values $t_{i,j}$ and $t_{i,j+1}$ of ith particle, g(px) is the gray level of px, and g(N(px)) refers to the gray level of a pixel neighboring to the region px, in 4-connectivity. The algorithm of thresholding and grouping the target image into regions is summarized in Algorithm 2.

The fitness of every particle is evaluated by using the quantitative evaluation function, $Q^\backslash$, shown in Equation 15. $Q^\backslash$ function uses the information stored in the list of regions to

```
Algorithm 2:  Grouping a target image into regions.
- For each pixel, px, in the target image.
-   If   Pixel, px, is ungrouped Then
        (i)   Grow px with its neighboring pixels according to the threshold values of particle i.
        (ii) - Add the new region to the list of regions.
-   End if
- End For
```

get the fitness of a particle. The gbest takes the position of the particle having the best fitness value. Once the gbest has been found, the threshold values of each particle are updated using Equation (3) and (4) in 2-dimention space (i, j), where i represents ith particle and j represents jth threshold.

Step 3 of PSOTH algorithm is repeated until the stopping criterion has been satisfied. The segmented image of the global best solution, *gbest*, after finishing the final iteration, is chosen as a best solution. For the best segmentation, merging small regions must be done. In the merging step, the final list of regions is scanned to merge regions having pixels less than a predefined number (e.g. 10 pixels), with their neighboring regions that satisfy a similarity condition. The similarity condition that is used in merging processes is the smallest difference between means. In the algorithm of PSOTH, the number of final regions is determined after merging small regions step.

V. Experimental Results

In order to prove the efficiency and accuracy of PSOTH, it has been applied to many different images. In this section we will present the results produced through applying the PSOTH method to the three images shown in Figure 3. These images have been selected to test the algorithm of PSOTH, and to compare it with other algorithms. All the test images are 256x256 pixels in size. This section is organized as follows. Section A presents the examination of $Q^{\backslash}$ function. Section B shows the tracing steps of PSOTH algorithm and Section C shows the performance measuring of PSOTH method.

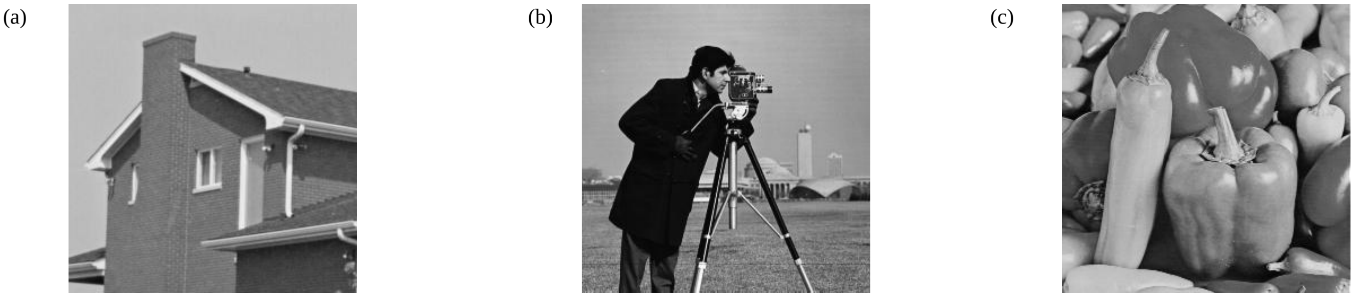


Figure 3:   Test Images   (a). House.   (b). Cameraman.   (c). Peppers.

A. $Q^{\backslash}$ *Function Examination*

$Q^{\backslash}$ function was examined through applying it to evaluate the segmentation results produced through segmenting the test images using the principle of normal region-based segmentation. Here, the normal region based segmentation segments the test images through multithresholding them into a group of regions using *k* thresholds, where *k* thresholds divide the gray levels into a number of gray level ranges with equal interval. Consequently, a test image scanned from left to right and left to bottom, and the connected pixels that fall in the same range of gray level grouped into a single region. This action is repeated for n=40 iterations. In each iteration, a different segmentation is produced, where k is set to the number of iteration so k = {1, 2, …, n}.

Figure 4 shows the best segmentations that were chosen by $Q^{\backslash}$ function. To show the efficiency of $Q^{\backslash}$ function for choosing the best segmentation, the results produced by $Q^{\backslash}$ will be compared with the results chosen by Q function. Figure 5 shows the best segmentations that were produced by Q function. With human visual perception, it can be seen that the results produced by $Q^{\backslash}$ function are better than the results produced by Q function.

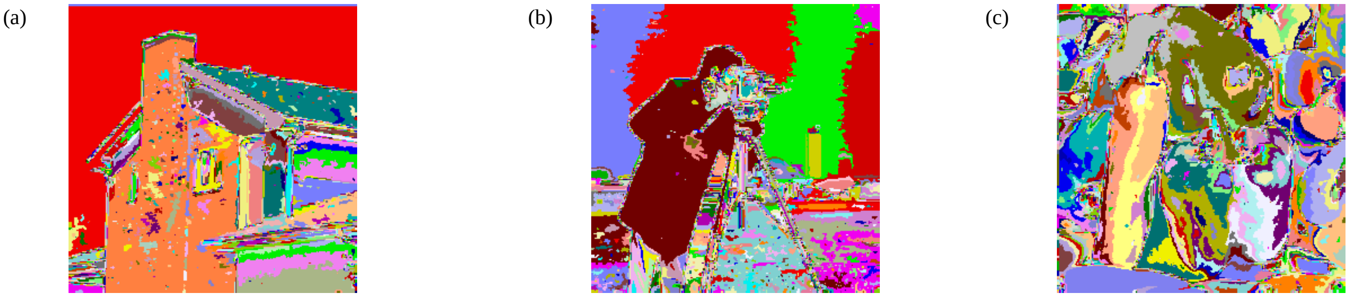


Figure 4: The best segmentations choosing by $Q^{\backslash}$ function for   (a). Cameraman image.   (b). House Image   (c).Peppers image.

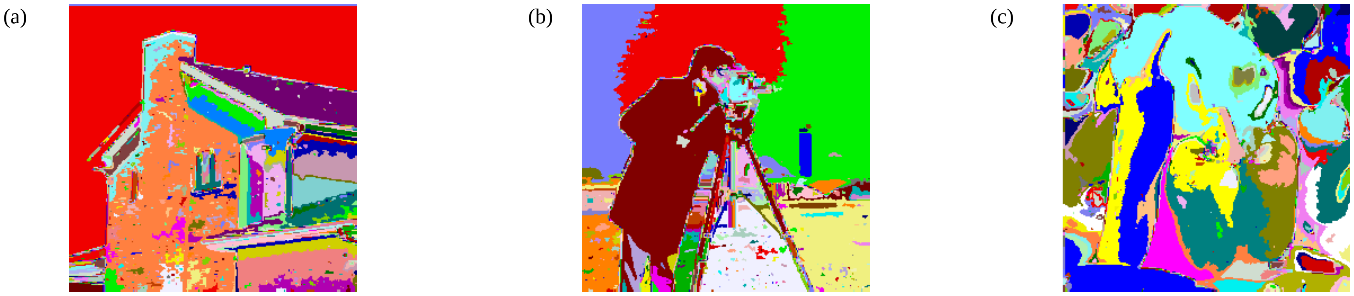


Figure 5: The best segmentations choosing by Q function for (a). Cameraman image. (b). House Image (c).Peppers image.

B. Tracing Steps of PSOTH Algorithm

For tracing purpose, the algorithm of PSOTH is used to get a near optimal segmentation for Cameraman image. Firstly, the algorithm starts by applying (3x3) low pass filter, Figure 6(a) shows the filtered image, and, then, the particles are randomly initialized. The x-vector and v-vector values are initialized in the ranges [g_{min}, g_{max}] and [V_{max}, -V_{max}] respectively, where g_{min} = 13, g_{max} =233 for Cameraman image.

For the PSO algorithm, in this experiment, 20 particles are trained for 50 iterations. The PSO parameters are initially set as follows. V_{max} = 3, w = 1 decreased over time to 0.4 and C_1 = C_2 = 1.49. The particles are trained with k=10. Empirically, we found that the proper value of k falls in the range [7, 12], where if we increase the value of k, more details will be produced and if we decrease it, more flat regions will be produced.

The algorithm of PSO starts by segmenting the target image using the gray level ranges of each particle and computing the fitness of each one using the $Q^{\backslash}$ evaluation function. The flow chart shown in Figure 7 illustrates the values of $Q^{\backslash}$ related to each particle in the first iteration. In the flow chart, the best particle, 2nd particle, is white in color and the worst particle, 11th particle, is black in color. Figure 6(b) and Figure 6(c) show the best and worst segmentation in the first iteration, i.e. illustrating that the segmentation of 2nd particle is clearly better than the segmentation of 11th particle. After that, the particle having the best fitness value is taken as *gbest*. Finally the velocities and positions of each particle are updated respectively according to the updating functions. These steps are repeated until the number of iterations has been satisfied. Figure 8 illustrates for the target image how the fitness of PSO improves over time. The fitness value, as measured using $Q^{\backslash}$ function, improved from the initial 0.2720554 to 0.1326954.

After that, the segmented image of *gbest* is chosen as a near optimal segmentation (the best solution), Figure 6(d) shows the best solution choosing by PSO. For the best solution small regions having pixels less than 10 pixels are merged with their neighboring regions. Figure 6(e) shows the best solution after merging small regions. Clearly, we can see that the segmentation in Figure 6(e) is better than the segmentation shown in Figure 6(c). This means that the segmentation of House image was improved over item according to the values produced by $Q^{\backslash}$ function. Finally, Figure 6(f) shows the regions of the best solution after filling with their means.

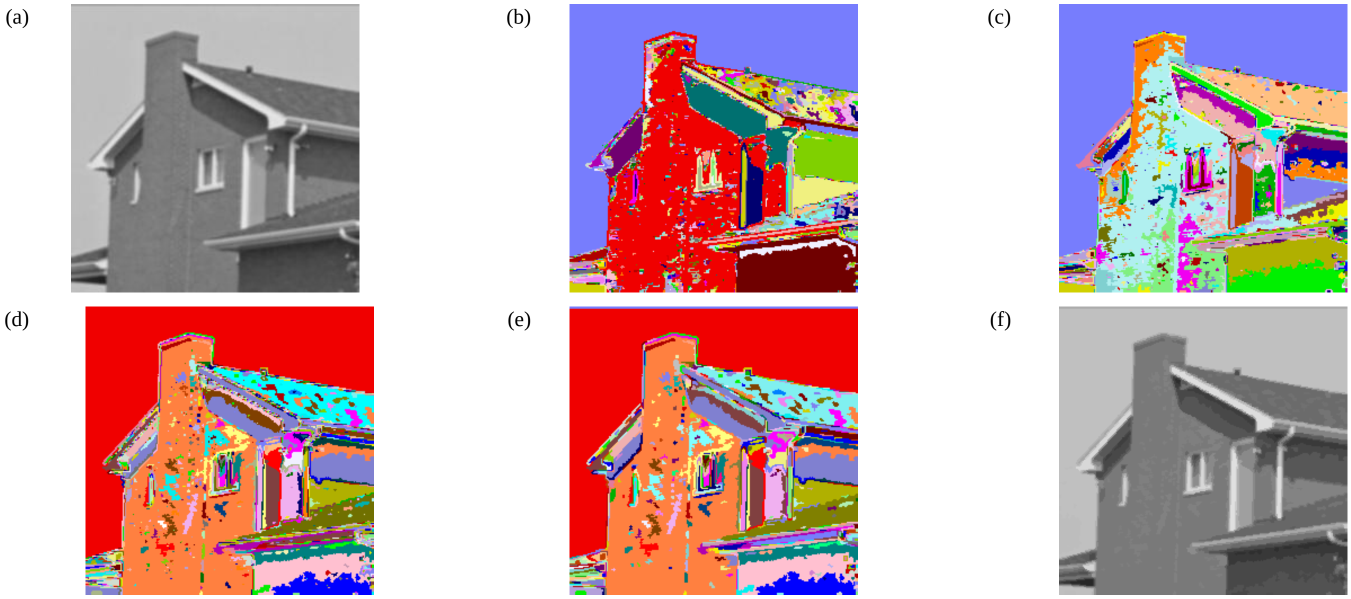


Figure 6: Tracing steps of PSOTH algorithm.

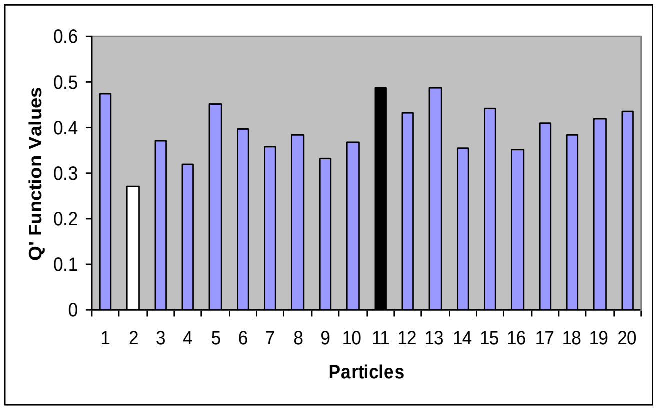


Figure 7: Fitness of Particles in First Iteration.

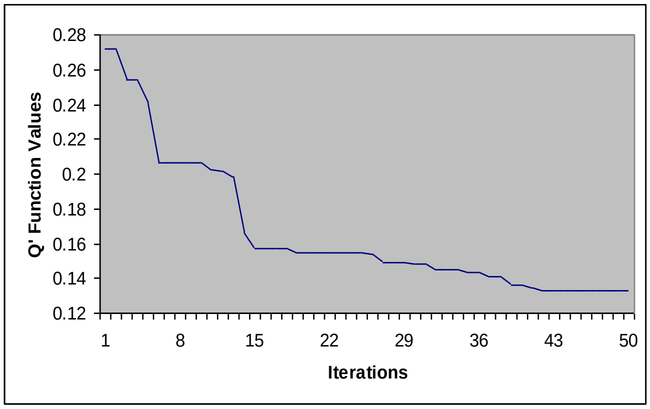


Figure 8: PSOTH Performance on House Image

C. Performance Measuring

Comparing different segmentation algorithms with each other is difficult mainly because they differ in the properties they try to satisfy [33]. For the comparison of the PSOTH method, the segmentation results of it will be compared with the results produced by implementing the algorithm of *k*-means and the normal region-based segmentation as discussed in previous section.

In all the test images, for PSOTH method, 20 particles were trained for 50 iterations, k=10, Vmax = 3, w = 1 decreased over time to 0.4, c1 = c2 = 1.49 and for the fitness function, $Q^{\backslash}$ function, w1 = 0.55, w2 = 0.45. For the k-means algorithm, 100 iterations were used and k=10 centroids selected randomly in the range [gmin, gmax] of each image separately. For region based method, n=40 iterations was used and k= {1, 2, …, n} depending on the number of iteration.

Figures 9(a, b, c) illustrate the segmented images produced through applying the PSOTH method to the test images, while Figures 9(d, e, f) and Figures 9(g, h, i) illustrate the segmented images obtained from applying the k-mean algorithm and the region based segmentation to the test images respectively. With human visual perception, it can be seen that the results shown in the figures clearly illustrate that the segmented images obtained by the PSOTH method are better than the segmented images obtained by the region based segmentation and the k-means algorithm. This means that the PSO algorithm is a good optimization method can be used to segment images if it is complicated with a good fitness function such as $Q^{\backslash}$ function.

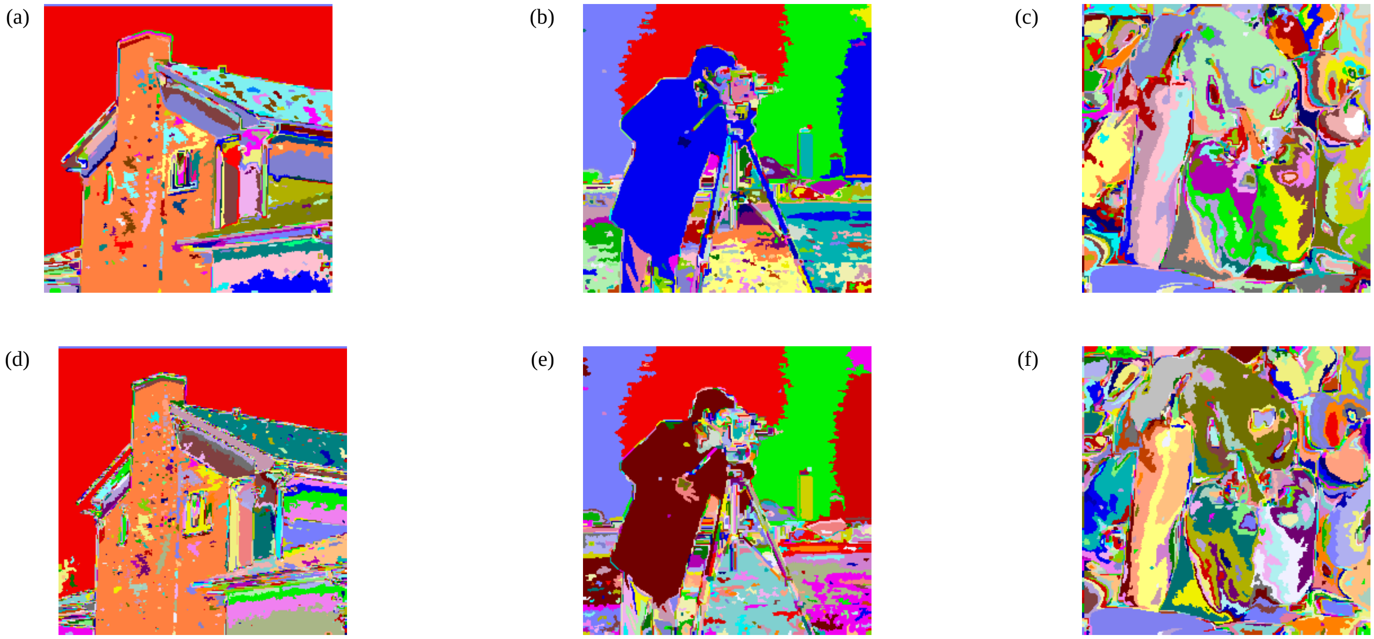


Figure 9: The best Segmentation of Test Images Produced by (a, b, c). PSOTH method. (d, e, f). Normal region based algorithm. (g, h, i) *k*-means algorithm.

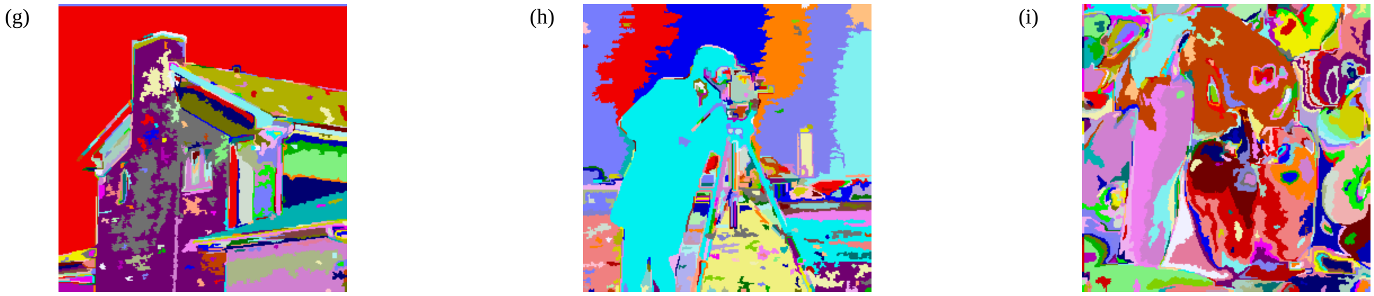


Figure 9: Continue;

- Evaluation Measuring

Table 1 summarizes the values of $Q^{\backslash}$ function produced by applying the normal region based segmentation, *k*-means algorithm and PSOTH method to the test images respectively (the values in Table 1 are the evaluations of segmented images in Figure 9). The results show that, in all cases, the *k*-means algorithm performed better than the region based segmentation method, and the PSOTH method performed better than both the region based method and *k*-means algorithm. Note that the values of $Q^{\backslash}$ function were scaled to be in the range [0, 10].

TABLE I. COMPARISON BETWEEN REGION BASED, K-MEANS AND PSOTH

Images	**Region Based**	***k*-means**	**PSOTH**
House	0.2030503	0.1806134	0.1326954
Cameraman	0.1972899	0.1503304	0.1463228
Peppers	0.1692429	0.1587412	0.1372648

VI. CONCLUSION

In this work, PSO has been used to produce a new optimization-based image segmentation method, PSOTH. In the PSOTH method, the algorithm of PSO tries to find a near optimal segmentation for a given image using a fitness function. PSO is a flexible optimization method, where many objective functions can be used. For this reason, a new quantitative evaluation function for segmented images has been proposed in this paper. So in the PSOTH method, the new evaluation function has been used as a fitness function for the algorithm of PSO. The experimental results have illustrated that the efficiencies of the PSOTH method and the new evaluation function.

REFERENCES

[1] E. Bonabeau, M. Dorigo and G. Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, NY: Oxford University Press, 1999.

[2] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in the Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, Vol. 4, pp. 1942-1948, 1995.

[3] R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis, John Wiley & Sons, New-York, 1973.

[4] A. A. Younes, I. Truck, and H. Akdaj, "Color Image Profiling Using Fuzzy Sets," Turk J Elec. Engin., Vol.13, No.3, 2005.

[5] M. Sezgin, B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation, " J. Electron. Imaging , Vol. 13, No. 1, pp. 146-165, 2004.

[6] J. Marcello, F. Marques and F. Eugenio, "Evaluation of thresholding techniques applied to oceanographic remote sensing imagery," SPIE, 5573, pp. 96-103, 2004.

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

[8] E. Zahara, S. S. Fan and D. Tsai, "Optimal multi-thresholding using a hybrid optimization approach," Pattern Recognition Letters, Elsevier, Vol. 26, pp. 1082–1095, 2005.

[9] P. Yin, "Multilevel minimum cross entropy threshold selection based on particle swarm optimization," Applied Mathematics and Computation, Vol. 184, pp. 503–513, 2007.

[10] M. Maitra and A. Chatterjee, "A hybrid cooperative–comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding," Expert Systems with Applications, Vol. 34, pp. 1341–1350, 2008.

[11] Y. zhiwei, C. hongwei, L wei and Z. jinping, "Automatic threshold selection based on Particle Swarm Optimization algorithm," in the proccesdngs International Conference on Intelligent Computation Technology and Automation, pp. 36-39, 2008

[12] C. Wei and F. Kangling, "Multilevel Thresholding Algorithm Based on Particle Swarm Optimization for Image Segmentation," in the Proceedings of the 27th Chinese Control Conference, July 16-18, Kunming, Yunnan, China, pp. 348-351, 2008

[13] P. D. Sathya, R. Kayalvizhi, "PSO-Based Tsallis Thresholding Selection Procedure for Image Segmentation," International Journal of Computer Applications, Vol. 5, No. 4, pp. 39-46, 2010.

[14] T. Hongmei, W. Cuixia, H. Liying, and W. Xia, "Image Segmentation Based on Improved PSO," the proceedings of the International Conference on Computer and Communication Technologies in Agriculture Engineering(CCTAE2010), pp. 191-194, 2010.

[15] J. Kennedy, and R. Eberhart, Swarm Intelligence, San Francisco: Morgan Kaufmann Publishers, 2001.

[16] Y. Shi, and R. Eberhart, "A modified particle swarm optimizer," in the Proceedings of the IEEE International Conference on Evolutionary Computation, Piscataway, NJ, pp. 69-73, 1998.

[17] Liu J., Yang Y. H., "Multiresolution Color Image Segmentation," IEEE Trans. on PAMI, Vol. 16, No. 7, pp. 689-700, 1994.

[18] Borsotti M., Campadelli P., Schettini R., "Quantitative Evaluation of Color Image Segmentation Results," Pattern Recognition Letters, Elsevier, vol.19, pp. 741-747, 1998.

[19] H. Zhang, J. Fritts, and S. Goldman, “An entropy-based objective evaluation method for image segmentation,” in Proc. SPIE- Storage and Retrieval Methods and Applications for Multimedia, 2004.

[20] Y. J. Zhang, "A survey on evaluation methods for image segmentation," Pattern Recognition, Elsevier, Vol. 29, No. 8, pp. 1335–1346, 1996.

[21] Hui Zhang, Jason E. Fritts, and Sally A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," Computer Vision and Image Understanding, Vol. 110, No. 2, pp.260-280, 2008