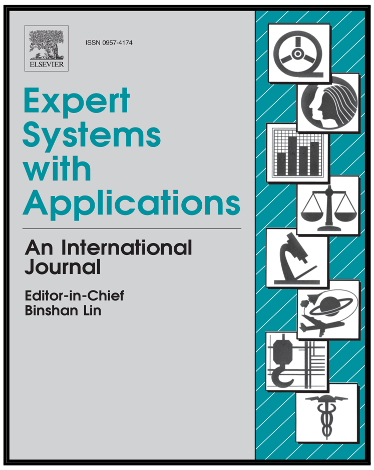
Accepted Manuscript



Multilevel thresholding using grey wolf optimizer for image segmentation

Abdul Kayom Md Khairuzzaman , Saurabh Chaudhury

PII: S0957-4174(17)30274-9
DOI: 10.1016/j.eswa.2017.04.029
Reference: ESWA 11262

To appear in: *Expert Systems With Applications*

Received date: 14 March 2016
Revised date: 14 April 2017
Accepted date: 15 April 2017

Please cite this article as: Abdul Kayom Md Khairuzzaman , Saurabh Chaudhury , Multilevel thresholding using grey wolf optimizer for image segmentation, *Expert Systems With Applications* (2017), doi: 10.1016/j.eswa.2017.04.029

Highlights:

- A new method of multilevel thresholding for image segmentation using Grey Wolf Optimizer (GWO) is proposed
- Two objective functions-Kapur's entropy and Otsu's between class variance are used
- The proposed method is more stable than PSO and BFO based methods
- Yields solutions of higher quality than PSO and BFO based methods
- Faster than BFO but slower than PSO

Point-to-Point responses to Reviewer Comments

Comments to Author:

There are still some typewriting errors (e.g., missing space between words - '…wolvesare…', 'ispresented', 'efficientalgorithms', 'functionas', 'methodsare', 'thresholdsincrease',).

Answer: The article is carefully rechecked and the typing errors are now corrected in the revised manuscript. We sincerely hope that there is no more typing error.

Section 1

Sentence 'A contour detection…into contour detection.' - Delete this sentence.

Answer: This sentence is now deleted from the revised manuscript as suggested.

Table 8

Correct caption of column 3 'GWOO' into 'GWO'.

Answer: The caption is now corrected in the revised manuscript.

Column 2 - add to <0.05 + sign in superscript.

Answer: '+' sign is added in the superscript of all <0.05 values.

ACCEPTED MANUSCRIPT

Multilevel thresholding using grey wolf optimizer for image segmentation

Abdul Kayom Md Khairuzzaman[1]

[1]Department of Electrical Engineering
National Institute of Technology (NIT) Silchar
Silchar-788010, Assam (India)
Email: kayomabdul@gmail.com

Saurabh Chaudhury[2,*]

[2]Department of Electrical Engineering
National Institute of Technology (NIT) Silchar
Silchar-788010, Assam (India)
Email: saurabh@nits.ac.in

***Corresponding author:**

Saurabh Chaudhury, Ph.D

Associate Professor,
Department of Electrical Engineering
National Institute of Technology (NIT) Silchar
Silchar-788010, Assam (India)
Email: saurabh@nits.ac.in
Phone (m): +91 9859939417

Abstract: Multilevel thresholding is one of the most important areas in the field of image segmentation. However, the computational complexity of multilevel thresholding increases exponentially with the increasing number of thresholds. To overcome this drawback, a new approach of multilevel thresholding based on Grey Wolf Optimizer (GWO) is proposed in this paper. GWO is inspired from the social and hunting behavior of the grey wolves. This metaheuristic algorithm is applied to multilevel thresholding problem using Kapur's *entropy* and Otsu's *between class variance* functions. The proposed method is tested on a set of standard test images. The performances of the proposed method are then compared with improved versions of PSO (Particle Swarm Optimization) and BFO (Bacterial Foraging Optimization) based multilevel thresholding methods. The quality of the segmented images is computed using Mean Structural SIMilarity (MSSIM) index. Experimental results suggest that the proposed method is more stable and yields solutions of higher quality than PSO and BFO based methods. Moreover, the proposed method is found to be faster than BFO but slower than the PSO based method.

Keywords: multilevel thresholding; image segmentation; grey wolf optimizer; Kapur's entropy; Otsu's threshold

1. Introduction

Segmentation partitions an image into some meaningful regions or extracts its boundaries. It is often the preprocessing stage of higher level processing such as, image analysis, and object recognition and computer vision. Therefore, the performance of higher level processing system depends on the accuracy of segmentation technique used.

Histogram based thresholding is a simple and the most popular image segmentation technique. Some survey on thresholding techniques can be found in (Mehmet Sezgin, 2004; Nikhil, R. Pal, 1993; P.K. Sahoo, S. Soltani, 1988). Thresholding techniques can be broadly classified into *local* and *global*. Global threshold selection techniques are most popular due to their simplicity and effectiveness. Among the global thresholding techniques, Kapur (Kapur, Sahoo, 1985) and Otsu (Otsu, 1979) methods are the most popular ones. Otsu method maximizes the *between class variance* function whereas, Kapur method maximizes *posterior entropy* of the segmented classes to find optimum thresholds. Computational complexity of Kapur and Otsu methods increases exponentially with the increasing number of thresholds due to exhaustive search used for finding the thresholds.

Liao et al. (Liao, Chen, 2001) proposed a fast recursive algorithm with a look up table for multilevel Otsu thresholding. Yin and Chen (Yin, 1997) proposed an iterative scheme for multilevel thresholding. Reddi et al.(Reddi, Rudin, & Keshavan, 1984) used a criterion function which was derived by assuming gray level histogram of an image to be a continuous probability function. Although these methods reduce the computational cost to some extent, processing time is still an issue. A statistical recursive algorithm proposed by Arora et.al (Arora, Acharya, Verma, & Panigrahi, 2008) uses mean and variance in order to perform multilevel image segmentation. It starts from the two extreme ends of the histogram and recursively apply the algorithm until there is no significant improvement in the segmented image. Although this method is very fast (<0.3 seconds), it assumes the image histogram as a Gaussian distribution and uses only the even number of thresholds.

Nature has always been an inspiration in solving computationally hard problems. Genetic Algorithm (GA) is applied in multilevel thresholding (Yin, 1999) using Otsu and Kapur functions. This method gives a very good segmentation with substantial reduction in the computational cost of multilevel thresholding. Differential Evolution (DE) along with Gaussian approximation function is applied in multilevel thresholding (Cuevas, Zaldivar, & Pérez-cisneros, 2010). A synergetic DE algorithm is applied in multilevel thresholding (Ali, Ahn, & Pant, 2014). This method outperforms Particle Swarm Optimization (PSO), GA, and DE based methods. The computational cost is greatly reduced but still it is high (~2 seconds). A hybrid DE algorithm using Otsu function is used for multilevel thresholding (Brest, Mlakar, & Poto, 2016). This method is very fast (~0.5 seconds). Maitra & Chatterjee has applied a comprehensive learning based hybrid PSO algorithm in multilevel thresholding (Maitra & Chatterjee, 2008a). This algorithm takes care of stagnation problem of PSO algorithm. Gao et al. (Gao, Xu, Sun, & Tang, 2010) has proposed a quantum behaved PSO algorithm for multilevel thresholding using Otsu function. Although this method has greatly reduced the computational complexity of multilevel thresholding, the computational time is still high (>5 seconds). Artificial bee colony algorithm is used to optimize a Gaussian mixture model for multilevel image segmentation (Cuevas, Sencion, Zaldivar, Perez-Cisneros, & Sossa, 2012). An improved version of bat algorithm is applied in multilevel thresholding (Alihodzic & Tuba, 2014). A cuckoo search algorithm is also applied in multilevel thresholding using Kapur's entropy as criterion function to segment satellite images (Bhandari, Singh, Kumar, & Singh, 2014). Sathya and Kayalvizhi have applied Bacterial Foraging Optimization (BFO) algorithm and its modified version (Sathya & Kayalvizhi, 2011b, 2011c) in multilevel thresholding using Otsu and Kapur functions. These two methods also substantially reduce the computational complexity of multilevel thresholding. But computational time is more than two seconds for Otsu function, whereas, it is more than six seconds for Kapur's function. Success of these nature inspired algorithms in multilevel thresholding applications and the quest for further reduction in computational complexity of multilevel thresholding have inspired the authors to investigate the merits of Grey Wolf Optimizer (GWO) algorithm in multilevel thresholding.

Multilevel thresholding has been applied in various image segmentation applications. For example, Maitra & Chatterjee has proposed a magnetic resonance image segmentation method using multilevel thresholding (Maitra & Chatterjee, 2008b). This method has applied BFO algorithm using Kapur's entropy function for multilevel thresholding. Similarly, Sathya and Kayalvizhi (Sathya & Kayalvizhi, 2011a) has presented a magnetic resonance image segmentation method based on BFO algorithm. Bhandari et al. (Bhandari et al., 2014) has applied multilevel thresholding using Cuckoo Search for segmenting satellite images. Sarkar et al. (Sarkar, Das, & Sinha, 2016) has proposed a multilevel thresholding based satellite hyper-spectral image segmentation technique using Renyi's entropy aided with differential evolution.

This paper presents a new method of multilevel thresholding based on Grey Wolf Optimizer (GWO). The proposed method selects the optimal set of thresholds using either Otsu's between class variance or Kapur's entropy function. The main contributions of this paper are: (1) the application of GWO for optimal multilevel thresholding using Otsu and Kapur methods. The result of experimentation suggest that GWO gives better result compared to BFO and PSO based methods, and (2) the computational complexity of multilevel thresholding is greatly reduced.

The remainder of the article is organized as follows. Section 2 describes the multilevel thresholding problem. It also describes Kapur's entropy and Otsu's between class variance functions for multilevel thresholding. Section 3 gives an overview of GWO followed by its mathematical model. Section 4 describes the proposed GWO based multilevel thresholding method followed by its pseudo code. Section 5 describes the experimental environment of the proposed method. Results and discussion are givenin section 6. Finally, the conclusions are provided in section 7.

2. Multilevel thresholding

Bi-level thresholding partitions an image into two classes: the object and the background. If the image is complex and contains multiple objects then bi-level thresholding is not very effective. In such a situation, multilevel thresholding is often used to segment an image. However, selecting the appropriate values of these thresholds is very important in obtaining a good segmentation. Optimal threshold selection techniques search for thresholds by optimizing (either minimizing or maximizing) an objective function. Kapur's entropy and Otsu's between class variance based methods are the two most widely used optimal thresholding techniques. In the following subsections we give a brief formulation of the above two multilevel thresholding techniques.

Let us assume that there are L numbers of gray levels in a given image which are in the range $\{0,1,2,...,(L-1)\}$. Let N be the total number of pixels in the image. A particular gray level i occurs n_i times in the image. The gray level histogram of the image can be normalized and regarded as a probability distribution. We define the probability of occurrence of a gray level i *as*, $p_i = n_i / N$. The following subsections describe Kapur's *entropy* and Otsu's *between class variance* functions in brief.

2.1. Otsu (between class variance) method

Otsu method (Otsu, 1979) is a nonparametric and unsupervised automatic threshold selection technique. It selects optimum thresholds by maximizing between class variance of the segmented classes. Let there be *m* number of thresholds $[t_1, t_2, ..., t_m]$ to be selected. These thresholds subdivide the image into $m+1$ classes: $C_0, C_1, C_2, \ldots, C_m$ by maximizing the objective function,

$$J_1(t_1, t_2, ..., t_m) = \sigma_0^2 + \sigma_1^2 + \sigma_2^2 + ... + \sigma_m^2 \tag{1}$$

Where

$$\sigma_0^2 = \omega_0(\mu_0 - \mu_T)^2, \omega_0 = \sum_{i=0}^{t_1-1} p_i, \ \mu_0 = \sum_{i=0}^{t_1-1} ip_i / \omega_0;$$

$$\sigma_1^2 = \omega_1(\mu_1 - \mu_T)^2, \omega_1 = \sum_{i=t_1}^{t_2-1} p_i, \ \mu_1 = \sum_{i=t_1}^{t_2-1} ip_i / \omega_1;$$

$$\sigma_2^2 = \omega_2(\mu_2 - \mu_T)^2, \omega_2 = \sum_{i=t_2}^{t_3-1} p_i, \ \mu_2 = \sum_{i=t_2}^{t_3-1} ip_i / \omega_2; \ldots$$

$$\sigma_m^2 = \omega_m(\mu_m - \mu_T)^2, \ \omega_m = \sum_{i=t_m}^{L-1} p_i, \ \mu_m = \sum_{i=t_m}^{L-1} ip_i / \omega_m.$$

$\sigma_0^2, \sigma_1^2, \sigma_2^2, \ldots, \sigma_m^2$ are the variances. $\omega_0, \omega_1, \omega_2 \ldots, \omega_m$ are the class probabilities. $\mu_0, \mu_1, \mu_2, \ldots,$ μ_m are the mean levels of the segmented classes: $C_0, C_1, C_2, \ldots, C_m$ respectively. If μ_T is the mean intensity for the whole image, then we have, $\omega_0\mu_0 + \omega_1\mu_1 + \omega_2\mu_2 + ... + \omega_m\mu_m = \mu_T$ and $\omega_0 + \omega_1 + \omega_2 + ... + \omega_m = 1$.

The equation (1) is maximized to find the optimum thresholds $t_1, t_2, \ldots, t_m$. Since the GWO algorithm minimizes an objective function, therefore, the objective function in equation (1) is converted to an equivalent minimization function by taking the inverse of it. The corresponding minimization function is taken as,

$$J_1^{'}(t_1, t_2, ..., t_m) = 1/(1 + J_1(t_1, t_2, ..., t_m)) \quad (2)$$

Minimizing the function in equation (2) is equivalent to maximizing the *between class variance* function in equation (1). After finding the optimum solution, the objective function value is converted back into the maximization form as in the equation (1) and presented in the paper.

2.2.Kapur's (entropy criterion) method

Kapur's method (Kapur, Sahoo, 1985) selects the optimum thresholds by maximizing the entropy of the segmented classes. It uses Shannon's concept of entropy. Shannon's function (Shannon, 1948) says that the information content of an event is inversely proportional to its probability of occurrence. Kapur et al. define the entropy of an image assuming that an image is entirely represented by its gray level histogram. If there are m number of thresholds $[t_1, t_2, ..., t_m]$ to be selected which subdivide the image into the classes: $C_0, C_1, C_2, \ldots, C_m$, then Kapur's method does it by maximizing the objective function,

$$J_2(t_1, t_2, ..., t_m) = H_0 + H_1 + H_2 + ... + H_m \quad (3)$$

Where,

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

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

$$H_2 = -\sum_{i=t_2}^{t_3-1} (p_i / \omega_2) \ln(p_i / \omega_2), \ \omega_2 = \sum_{i=t_2}^{t_3-1} p_i ; \ldots$$

$$H_m = -\sum_{i=t_m}^{L-1} (p_i / \omega_m) \ln(p_i / \omega_m), \ \omega_m = \sum_{i=t_m}^{L-1} p_i .$$

$H_0, H_1, H_2, \ldots, H_m$ are the Kapur's entropies. $\omega_0, \omega_1, \omega_2, \ldots, \omega_m$ are the class probabilities of the segmented classes: $C_0, C_1, C_2, \ldots, C_m$ respectively.

The equation (3) is maximized to find the optimum thresholds $t_1, t_2, \ldots, t_m$. Similar to the equation (1) the equation (3) is also converted to an equivalent minimization function by taking the inverse of it. The corresponding minimization function is taken as,

$$J_2^{'}(t_1, t_2, ..., t_m) = 1/(1 + J_2(t_1, t_2, ..., t_m)) \quad (4)$$

Minimization of the function in equation (4) corresponds to the maximization of Kapur's *entropy* function in equation (3). After finding the optimum solution, the objective function value is converted back into the maximization form as in the equation (3) and presented in the paper.

3. Grey Wolf Optimizer

Grey wolf optimizer (GWO) is a metaheuristic proposed by Mirjalili et al. (Mirjalili, Mohammad, & Lewis, 2014). A brief overview and the mathematical model of GWO are given in the following subsections.

3.1. Overview

Grey wolf optimizer is inspired by the social hierarchy and the hunting technique of grey wolves. Grey wolves live in a highly organized pack. The average pack size ranges from 5-12. Normally, there are four different ranks of wolves in a pack. These are alpha (α), beta (β), delta (δ), and omega (ω) wolves. The alphas (a male and a female) are the leaders of the pack and they dominate the whole pack. Other members of the pack are the followers of alphas. Besides the social hierarchy, hunting is also an interesting social behavior of the grey wolves. The main phases of grey wolf hunting are: (1) Tracking, chasing, and approaching the prey (2) Pursuing, encircling, and harassing the prey until it stops moving and (3) Attacking the prey.

3.2.Mathematical model of GWO

The mathematical model of the GWO as proposed by Mirjalili et al. (Mirjalili et al., 2014) is presented here in brief covering the social hierarchy and hunting strategy.

3.2.1. Social hierarchy

To model the social hierarchy of the grey wolves, the fittest solution is considered as the α. The next two best solutions are considered as the β and the δ. The rest are ω wolves. The optimization process is guided by α, β, and the δ. The ω wolves follow them.

3.2.2. Encircling of prey

Grey wolves encircle the prey during hunting. The encircling behavior is modeled as,

$$\vec{D} = |C \cdot \vec{X}_p(\mathrm{t}) - \vec{X}(\mathrm{t})| \tag{5}$$

$$\vec{X}(\mathrm{t}+1) = \vec{X}_p(\mathrm{t}) - A \cdot \vec{D} \tag{6}$$

Where t is the current iteration, A and C are the coefficients. $\vec{X}_p$ and $\vec{X}$ are the position vectors of the prey and a grey wolf respectively. The values of the coefficients A and C are calculated as,

$$A = 2a \cdot r_1 - a \tag{7}$$

$$C = 2 \cdot r_2 \tag{8}$$

The value of a is linearly decreased from 2 to 0 during the iterations, while, r_1 and r_2 are random numbers in the range $[0,1]$. Grey wolves update their positions around the prey using equations (5) and (6). In *m*-dimensional search space the grey wolves move in hyper-spheres around the best solution obtained so far.

3.2.3. Hunting

The hunting process is guided by the alpha and sometimes also by beta, and delta. It is assumed that the α, β, and the δ have better knowledge about the location of the prey (optimum solution). Therefore, the other wolves update their positions according to the positions of the alpha, beta, and the delta. The mathematical model for hunting is given as,

$$\vec{D}_\alpha = |C_1 \cdot \vec{X}_\alpha - \vec{X}|,\ \vec{D}_\beta = |C_2 \cdot \vec{X}_\beta - \vec{X}|,\ \vec{D}_\delta = |C_3 \cdot \vec{X}_\delta - \vec{X}| \tag{9}$$

$$\vec{X}_1 = \vec{X}_\alpha - A_1 \cdot \vec{D}_\alpha, \vec{X}_2 = \vec{X}_\beta - A_2 \cdot \vec{D}_\beta, \vec{X}_3 = \vec{X}_\delta - A_3 \cdot \vec{D}_\delta \tag{10}$$

Where $\vec{X}_\alpha$, $\vec{X}_\beta$, and $\vec{X}_\delta$ are the position vectors of alpha, beta, and the delta respectively. A_1, A_2, A_3, C_1, C_2, and C_3 are the coefficients calculated using the equations (7) and (8) with a different set of random numbers.

The position of a wolf is updated as,

$$\vec{X}(\mathrm{t+1}) = (\vec{X}_1 + \vec{X}_2 + \vec{X}_3)/3 \tag{11}$$

Equation (11) shows how a grey wolf (search agent) updates its position according to alpha, beta, and delta in a search space.

3.2.4. Attacking the prey (exploitation)

When the prey stops moving, grey wolves attack it to finish the hunting process. This is modeled by decreasing a from 2 to 0 during the iterations. As a is decreased, the fluctuation ranges of A also decreases. When $|A|<1$, the grey wolves attack the prey.

3.2.5. Search of prey (exploration)

Grey wolves diverge from each other in search of prey. When a prey is found, they converge to attack it. The searching process is guided by the alpha, beta, and delta. When the coefficient vector $|A|>1$ the grey wolves diverge (exploration) from each other in search of prey. The coefficient C in equation (8) also favors exploration by providing random weights to the prey. The natural obstacles that make it difficult for the grey wolves to approach a prey are modeled using this parameter. It allows the GWO to explore the solution space and helps in avoiding any local optima.

4. Methodology

4.1.The Proposed GWO based Multilevel thresholding method

Grey wolves represent the search agents and their positions represent the thresholds to be optimized. Therefore, depending upon the number of thresholds, the wolves move in 1-D, 2-D, 3-D, or hyper dimensional space by changing their position vectors. The positions of the grey wolves are first initialized randomly. Then the fitness of all the wolves is determined using the equation (2) or (4). The best three wolves are designated as alpha, beta, and the delta. All other wolves update their positions determined by the positions of the alpha, beta, and delta using equation (11). The positions of the wolves are updated if better positions are found. This process is repeated till the maximum number of iteration is completed. The position of the alpha wolf gives the desired thresholds. The pseudo code of the proposed multilevel thresholding method is presented in the following subsection.

4.2.Pseudo code of the proposed GWO based multilevel thresholding

Initialize the positions of the grey wolves, $\vec{X}_i \, (i = 1, 2, ..., n)$

Initialize the value of a *as 2.*

Calculate the coefficients A *and* C *using equations (7) and (8) respectively*

Calculate the objective function value of each wolf by using equation (2) for Otsu, or equation (4) for Kapur based method

$\vec{X}_\alpha$, $\vec{X}_\beta$, and $\vec{X}_\delta$ *are the positions of alpha, beta, and delta wolves*

WHILE *(* t *<Maximum number of iterations)*

    FOR *each search agent*

        Update the position of the wolf using the equation (11)

    END FOR

    Decrease linearly the values of a *from 2 to 0 during the iterations*

    Update A *and* C *using equations (7) and (8) respectively*

    Calculate the objective function value of each wolf by using the equation (2) for Otsu, or equation (4) for Kapur based method

    Update $\vec{X}_\alpha$, $\vec{X}_\beta$, and $\vec{X}_\delta$

    $t = t + 1$

5. Experimental environment

This section presents a brief description on the experimental set up of the proposed method. Firstly, a description on some benchmark test images is presented. Parameters of different algorithms taken into comparison with the proposed method are discussed. Finally the segmentation validation metric is briefly described.

In this study, eight test images from different databases are used. Lena, Peppers, Baboon, Man, and Airplane images are taken from USC-SIPI image database which are of size 512x512 each. While Starfish, Snake, and Zebra images are taken from BSD 500 image database and are of size 481x321 each. These test images are presented in figure 1.

Performance of the proposed GWO based multilevel thresholding method is compared with some widely used optimization algorithms such as, PSO and BFO. BFO is first proposed by Passino (Passino, 2002) and PSO is first proposed by Kennedy and Eberhurt (Kennedy & Eberhart, 1995). The proposed method is compared with an improved version of BFO (Sathya & Kayalvizhi, 2011b) and also with an improved version of PSO (Shi & Eberhart, 1998) that introduces an inertia parameter to the original PSO. The parameters of each algorithm that are used in this study are presented in table 1. All the algorithms are implemented in Matlab R2010a with Intel core-i7 CPU @ 3.40 GHz.

The quality of the segmented images are compared by using Mean Structural SIMilarity index (MSSIM) (Wang et al., 2004). The MSSIM index evaluates overall image quality and is calculated as,

$$MSSIM(X,Y) = \left(\sum_{j=1}^{M} SSIM(x_j, y_j) \right) / M \tag{12}$$

Where X and Y represents the reference and the segmented image respectively. x_j and y_j are the image contents at j^{th} local window and M is the total number of local windows in the image. The Structural Similarity index (SSIM) is given as,

$$SSIM(x,y) = (2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2) / (\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2) \tag{13}$$

Where μ_x and μ_y are the mean intensities of the image local windows defined by *x* and *y* respectively. σ_x and σ_y are the standard deviation of *x* and *y* respectively. σ_{xy} is the local correlation coefficient between *x* and *y*. C_1 and C_2 are constants. The values of MSSIM index are in the range [-1, 1]. A higher value of MSSIM index indicates a better quality of the segmented image.

6. Results and discussion

This section presents the experimental results of the proposed GWO based multilevel thresholding technique and its analysis in terms of solution quality, stability,and computational time. A statistical analysis is also performed to see the superiority of the proposed GWO based technique over the other efficient algorithms. All these are discussed in the following subsections.

6.1. Solution quality

The values of the objective function as obtained by GWO, BFO, and PSO based methods are presented in table 2 and the corresponding threshold values are presented in table 3 and table 4. In maximization algorithm, higher the value of the objective function better is the solution. For a small number of thresholds (such as, *m*=2 and 3), the values of the objective function obtained by all the algorithms are practically the same. Therefore, the results with the higher number of thresholds (such as, *m*=4 and 5) are only presented. It is seen from the table 2 that GWO based methods most often find higher value of objective function compared to BFO and PSO based methods.

Segmented images as obtained by GWO-Kapur method and GWO-Otsu method are presented in figure 2 and figure 3 respectively. The segmentation results for Lena image for GWO-Kapur and GWO-Otsu methods are presented in figure 4 and figure 5 respectively. The quality of the segmented images is then compared by using MSSIM index. The MSSIM index evaluates the visual similarity between the original image and the segmented image. The MSSIM index values of the segmented images obtained by all the methods are given in the table 5. MSSIM index gives a higher value when the segmented image is more similar to the original image. From the table 5 it is found that MSSIM index values of the segmented images by GWO based methods are higher than the BFO and PSO based methods. For example, the MSSIM index values in case of Lena image with five thresholds for Otsu based methods are 0.8069, 0.8035, and 0.8048 for GWO, BFO, and PSO respectively. It clearly shows that GWO based method gives higher quality segmentation compared to BFO and PSO based methods. It is also seen from table 5 that, the value of MSSIM index increases as the number of thresholds increase. This indicates that segmentation quality improves as the number of thresholds is increased. This is also evident from the figures 2 and 3 that the visual quality of the segmented images improves as the number of thresholds increase. The thresholds are fitted in the histogram of the segmented images in figure 4 and figure 5. These figures illustrate how the thresholds are able to segment the image into different classes.

As metaheuristic algorithms are stochastic in nature, the solution found at each run may not be identical. Therefore another test is performed to analyze the accuracy and stability of the algorithms using mean and standard deviation of the objective function values. A lower value of standard deviation indicates higher stability of the algorithm. Whereas, higher mean value of the objective function indicates higher accuracy. Each algorithm is run 100 times for each 16 cases to find the standard deviation and the mean value of the objective function. The mean values of objective functions are given in table 6. From the table it is observed that GWO based method most often finds higher mean value of the objective function than the BFO and PSO based methods. Hence, the proposed GWO based multilevel thresholding method gives more accurate segmentation than the BFO and PSO based methods. The standard deviation of the algorithms is given in table 7. It is evident from the table that GWO based method more often has a lower standard deviation than BFO and PSO based methods. Hence, GWO based method is more stable compared to BFO and PSO based methods.

6.2. Statistical analysis

A statistical analysis using Wilcoxon rank sum (Wilcoxon, 1945) test is performed at 5% significance level. The objective function values of the proposed method are compared with BFO and PSO based methods. All the algorithms are run 100 times for the statistical analysis. The null hypothesis is constructed as: there is no significant difference between the three algorithms. The alternative hypothesis considers that there is a significant difference between the three algorithms. The p and h values are presented in table 8. A value of p>0.05 (or h=0) indicates that the null hypothesis cannot be rejected. On the other hand, a value of p<0.05 (or h=1) means the null hypothesis can be rejected at 5% significance level. A p value where GWO performs better than the other algorithms is marked with '+' in the superscript and with '#' where GWO performance is similar or worse than the other algorithms. In the experiment using Otsu function, GWO based method produces better result in 16 out of 16 cases when compared with BFO based method and produces better result in 13 out of 16 cases when compared with PSO based method. Whereas, in the experiment using Kapur entropy function, GWO based method produces better result in 16 out of 16 cases when compared with BFO based method and produces better result in 15 out of 16 cases when compared with PSO based method. These results suggest that there is a significant difference between the three algorithms. In most cases GWO based multilevel thresholding algorithm performs better than the other two algorithms.

6.3.Computation time

Average CPU time is used to compare the computational complexity of the multilevel thresholding methods. The average CPU time for GWO, BFO, and PSO based methods are given in table 9. Each of the algorithms was run 100 times to find the average CPU time using Intel core-i7 CPU @ 3.40 GHz. As for example, in case of *Lena image* with *five thresholds*, the average CPU time for Kapur based method are 146.9, 181.2, and 47.0 milliseconds for GWO, BFO, and PSO respectively. Whereas, the average CPU time for Otsu based methods are 114.1, 279.7, and 40.6 milliseconds for GWO, BFO, and PSO respectively. From table 9 it is seen that GWO is faster than BFO but slower than PSO.

6.4.Advantages and disadvantages of the proposed method

The main advantages (strengths) of the proposed method are as follows. The proposed multilevel thresholding method is simple and easy to implement. Multilevel thresholding barely using the Kapur's (Kapur, Sahoo, 1985) or Otsu's (Otsu, 1979) method are computationally expensive because they search the solution space exhaustively. The proposed GWO based method greatly reduces the computational complexity of multilevel thresholding by optimizing the threshold searching process. The proposed method is tested on intensity images. The experimental results suggest that it performs quite well on intensity image segmentation problems.

The BFO algorithm gives very competitive result compared to other algorithms. But it has many parameters to adjust and it may be prone to be trapped in a local optimum when the dimension of the search space is high. The PSO algorithm is one of the best and arguably the most widely used optimization algorithm. It has fewer parameters to adjust and it is one of the fastest optimization algorithms. But it may fall into local optimum in higher dimensional search space. The GWO algorithm has few parameters to adjust. The best three solutions (alpha, beta, and delta) guide the searching process. Exploration and exploitation parameters are perfectly balanced which enables it to avoid local optimum which is the reason of the better performance of GWO over BFO and PSO.

The main disadvantages (weakness) of the proposed method are as follows. The proposed method only uses the intensity information of the image to perform segmentation. It makes the method sensitive to noise and inefficient for advanced image segmentation problems. Thresholding based methods generally do not perform satisfactorily with images that contain intensity inhomogeneity problem. For example, medical images that contain intensity inhogeneity, advanced segmentation techniques such as deformable models (Ciecholewski, 2016; Li et al., 2011; Li, Xu, Member, Gui, & Fox, 2010; Mcinerney & Terzopoulos, 1996) are used which combines geometry, physics and approximation theory to carry out segmentation.

7. Conclusion

In this study, the merit of Grey Wolf Optimizer (GWO) in multilevel thresholding problem is investigated. Kapur's *entropy* and Otsu's *between class variance* functions are considered for threshold selection. The computational complexity of multilevel thresholding increases many fold with the increasing number of thresholds. So, to mitigate this problem, a GWO based technique is used to speed up the threshold searching process. The proposed method is tested on some standard test images, such as, Lena, Peppers, Baboon, Man, Airplane, Starfish, Snake, and Zebra. The performance of the proposed method is then compared with PSO and BFO based methods. The experimental results suggest that the proposed GWO based method is more stable and yields solutions of higher quality than BFO and PSO based methods. The computational complexity of GWO, BFO, and PSO based methods are compared with the average CPU run time. The comparison shows that GWO is faster than BFO but slower than PSO.

The Kapur method and the Otsu method are very efficient for bi-level thresholding, while their computational complexity increases exponentially in case of multilevel thresholding. In order to make these methods practical for multilevel thresholding, GWO is used in this work. The segmentation results were analyzed using MSSIM index, the mean, and the standard deviation of the objective functions. The experimental study reveals that the GWO algorithm together with Kapur's entropy or Otsus's between class variance function can be effectively used for multilevel thresholding.

The results of the proposed GWO based multilevel thresholding method are promising. In future work, authors aim to use some spatial properties of images along with the intensity information to improve thresholding based image segmentation technique. And also aim to finding a simpler and more efficient GWO algorithm and apply it to image segmentation and computer vision problems. Further, the merit of GWO can be investigated using minimum cross entropy, Tsallis entropy, Renyi's entropy for multilevel thresholding.

References

Ali, M., Ahn, C. W., & Pant, M. (2014). Multi-level image thresholding by synergetic differential evolution. *Applied Soft Computing Journal*, *17*, 1–11. https://doi.org/10.1016/j.asoc.2013.11.018

Alihodzic, A., & Tuba, M. (2014). Improved bat algorithm applied to multilevel image thresholding. *Scientific World Journal*, *2014*. https://doi.org/10.1155/2014/176718

Arora, S., Acharya, J., Verma, A., & Panigrahi, P. K. (2008). Multilevel thresholding for image segmentation through a fast statistical recursive algorithm. *Pattern Recognition Letters*, *29*(2), 119–125. https://doi.org/10.1016/j.patrec.2007.09.005

Bhandari, A. K., Singh, V. K., Kumar, A., & Singh, G. K. (2014). 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), 3538–3560. https://doi.org/10.1016/j.eswa.2013.10.059

Brest, J., Mlakar, U., & Poto, B. (2016). A hybrid differential evolution for optimal multilevel image thresholding, *65*, 221–232. https://doi.org/10.1016/j.eswa.2016.08.046

Ciecholewski, M. (2016). An edge-based active contour model using an inflation / deflation force with a damping coefficient. *Expert Systems With Applications*, *44*, 22–36. https://doi.org/10.1016/j.eswa.2015.09.013

Cuevas, E., Sencion, F., Zaldivar, D., Perez-Cisneros, M., & Sossa, H. (2012). A multi-threshold segmentation approach based on artificial bee colony optimization. *Applied Intelligence*, *37*(3), 321–336. https://doi.org/10.1007/s10489-011-0330-z

Cuevas, E., Zaldivar, D., & Pérez-cisneros, M. (2010). Expert Systems with Applications A novel multi-threshold segmentation approach based on differential evolution optimization. *Expert Systems With Applications*, *37*(7), 5265–5271. https://doi.org/10.1016/j.eswa.2010.01.013

Gao, H., Xu, W., Sun, J., & Tang, Y. (2010). Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm. *IEEE Transactions on Instrumentation and Measurement*, *59*(4), 934–946. https://doi.org/10.1109/TIM.2009.2030931

Kapur, Sahoo, W. (1985). A New Method for Gray-Level Picture Thresholding Using the Entropy of the Histogram. *Computer Vision Graphics and Image Processing*, 273–285.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Neural Networks, 1995. Proceedings., IEEE International Conference on*, *4*, 1942–1948 vol.4. https://doi.org/10.1109/ICNN.1995.488968

Li, C., Huang, R., Ding, Z., Gatenby, J. C., Metaxas, D. N., & Gore, J. C. (2011). A Level Set Method for Image Segmentation in the Presence of Intensity Inhomogeneities With Application to MRI, *20*(7), 2007–2016.

Li, C., Xu, C., Member, S., Gui, C., & Fox, M. D. (2010). Distance Regularized Level Set Evolution and Its Application to Image Segmentation, *19*(12), 3243–3254.

Liao, Chen, C. (2001). A Fast Algorithm for Multilevel Thresholding. *Journal of Information and Engineering*, *727*, 713–727.

Maitra, M., & Chatterjee, A. (2008a). A hybrid cooperative-comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding. *Expert Systems with Applications*, *34*(2), 1341–1350. https://doi.org/10.1016/j.eswa.2007.01.002

Maitra, M., & Chatterjee, A. (2008b). A novel technique for multilevel optimal magnetic resonance brain image thresholding using bacterial foraging. *Measurement*, *41*(10), 1124–1134. https://doi.org/10.1016/j.measurement.2008.03.002

Mcinerney, T., & Terzopoulos, D. (1996). Deformable models in medical image analysis : a survey, *1*(2), 91–108.

Mehmet Sezgin, B. S. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, *13*(January), 146–165. https://doi.org/10.1117/1.1631316

Mirjalili, S., Mohammad, S., & Lewis, A. (2014). Advances in Engineering Software Grey Wolf Optimizer. *Advances in Engineering Software*, *69*, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

Nikhil, R. Pal, S. K. P. (1993). A Review Image Seg techniques. *Pattern Recognition*, *26*(9), 1277–1294.

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, *9*(1), 62–66.

P.K. Sahoo, S. Soltani, W. (1988). SURVEY A Survey of Thresholding Techniques *. *Computer*

Vision, Graphics, and Image Processing, 233–260.
Passino, K. M. (2002). Biomimicry of Bacterial Foraging. *IEEE Control System Magazine*, (June), 52–67.
Reddi, S. S., Rudin, S. F., & Keshavan, H. R. (1984). An Optimal Multiple Threshold Scheme for Image Segmentation. *IEEE Transactions on Systems, Man and Cybernetics*, *SMC-14*(4), 661–665. https://doi.org/10.1109/TSMC.1984.6313341
Sarkar, S., Das, S., & Sinha, S. (2016). Hyper-spectral image segmentation using Rényi entropy based multi-level thresholding aided with differential evolution. *Expert Systems With Applications*, *50*, 120–129. https://doi.org/10.1016/j.eswa.2015.11.016
Sathya, P. D., & Kayalvizhi, R. (2011a). Amended bacterial foraging algorithm for multilevel thresholding of magnetic resonance brain images. *Measurement*, *44*(10), 1828–1848. https://doi.org/10.1016/j.measurement.2011.09.005
Sathya, P. D., & Kayalvizhi, R. (2011b). Modified bacterial foraging algorithm based multilevel thresholding for image segmentation. *Engineering Applications of Artificial Intelligence*, *24*(4), 595–615. https://doi.org/10.1016/j.engappai.2010.12.001
Sathya, P. D., & Kayalvizhi, R. (2011c). Optimal multilevel thresholding using bacterial foraging algorithm. *Expert Systems with Applications*, *38*(12), 15549–15564. https://doi.org/10.1016/j.eswa.2011.06.004
Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(July 1928), 379–423. https://doi.org/10.1145/584091.584093
Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, 69–73. https://doi.org/10.1109/ICEC.1998.699146
Wang, Z., Bovik, A. C., Sheikh, H. R., Member, S., Simoncelli, E. P., & Member, S. (2004). Image Quality Assessment : From Error Visibility to Structural Similarity, *13*(4), 600–612.
Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *International Biometric Society*, *1*(6), 80–83.
Yin. (1999). A fast scheme for optimal thresholding using genetic algorithms. *Signal Processing*, *72*(2), 85–95. https://doi.org/10.1016/S0165-1684(98)00167-4
Yin, C. (1997). A fast iterative scheme for multilevel thresholding methods. *Signal Processing*, *60*, 305–313.

Table 1: Parameters of the algorithms

Algorithm	Parameter	Value
GWO	Number of grey wolves	12
	Number of iterations	100
PSO	Number of particles	12
	Number of iterations	100
	Maximum particle velocity	6
	Maximum inertia weight	0.9
	Minimum inertia weight	0.2
	Cognitive and social parameters	[0,2]
BFO	Number of bactera	12

Number of chemotactic steps	10
Number of reproductive steps	4
Number of ellimination steps	2
Probability of ellimination	0.95
Split ratio	0.5
Depth of attractant	0.1
Width of attractant	0.2
Height of repellant	0.1
Width of repellant	10

Table 2: The objective function values obtained by GWO, BFO, and PSO based methods

Test images	*m*	Kapur's objective values			Otsu's objective values		
		GWO	BFO	PSO	GWO	BFO	PSO
Lena	4	**18.0122**	18.0082	17.9877	**2191.84**	2191.47	**2191.84**
	5	**20.6093**	20.5975	20.6078	**2217.77**	2213.11	2215.99
Peppers	4	**18.7338**	18.7331	18.5541	**3151.99**	3151.84	3151.90
	5	**21.4016**	21.3955	21.3847	**3195.93**	3195.11	3195.90
Baboon	4	**18.1304**	18.1297	**18.1304**	**1692.14**	1691.79	**1692.14**
	5	**20.7896**	20.7583	20.7870	**1717.89**	1715.23	1717.85
Man	4	**18.6810**	18.6807	18.5997	**3210.64**	3210.57	**3210.64**
	5	**21.4749**	21.4680	21.3089	**3256.54**	3255.96	3254.45
Airplane	4	**18.3121**	18.3107	18.3115	**2070.07**	2069.22	**2070.07**
	5	**20.9088**	20.9018	20.9072	**2096.13**	2092.65	2096.10
Starfish	4	**19.0579**	19.0537	19.0563	**2869.43**	2869.40	2869.20
	5	**21.8090**	21.8068	21.8072	**2916.27**	2914.69	2915.72
Snake	4	**18.6376**	18.6352	18.6356	**1285.68**	1285.35	1285.42
	5	**21.3965**	21.3885	21.3835	**1315.66**	1315.34	1315.47
Zebra	4	**17.8857**	17.8805	17.8821	**1590.82**	1590.65	1590.74

5	**20.4511**	20.4491	20.4414	**1618.15**	1618.14	1618.12

Table 3:The threshold values obtained by GWO, BFO, and PSO based Kapur's entropy methods

Test images	*m*	Threshold values obtained by Kapur's Entropy methods		
		GWO	BFO	PSO
Lena	4	64,97,137,179	66,97,136,179	78,120,160,192
	5	63,94,128,163,194	61,95,127,164,194	62,93,126,162,194
Peppers	4	46,84,130,179	45,84,129,178	71,108,142,180
	5	43,76,111,144,181	42,75,108,142,182	45,78,115,149,184
Baboon	4	33,74,114,159	33,73,113,158	33,74,114,159
	5	33,70,105,139,173	33,71,109,147,179	33,66,101,136,171
Man	4	46,90,132,175	47,90,132,175	62,117,174,230
	5	46,89,131,174,230	43,87,130,174,230	46,89,131,172,206
Airplane	4	67,106,145,185	67,106,144,184	67,106,144,185
	5	60,89,123,155,187	57,89,121,154,186	62,92,125,157,188
Starfish	4	68,116,164,206	69,114,163,207	66,113,160,203
	5	56,93,132,170,209	56,95,135,174,212	54,92,131,170,208
Snake	4	75,123,168,211	74,122,165,211	75,123,167,211
	5	68,107,146,181,219	70,110,147,183,219	62,99,138,177,217
Zebra	4	92,134,168,207	89,134,168,208	91,134,169,207
	5	90,130,161,191,222	92,131,161,191,222	75,104,136,169,208

Table 4: The threshold values obtained by GWO, BFO, and PSO based Otsu's between class variance methods

Test images	*m*	Threshold values obtained by Otsu method		
		GWO	BFO	PSO
Lena	4	75,114,145,180	74,115,145,180	75,114,145,180
	5	73,109,136,160,188	67,104,126,151,186	67,97,124,151,183
Peppers	4	41,89,135,175	40,89,134,175	40,88,134,174
	5	39,80,118,150,182	41,81,120,149,183	39,81,119,151,183
Baboon	4	72,106,137,168	72,104,136,167	72,106,137,168
	5	67,99,125,149,174	57,88,115,141,170	67,98,124,148,174
Man	4	35,82,124,164	36,82,124,164	35,82,124,164
	5	28,65,100,134,172	30,66,103,136,174	26,61,95,129,164
Airplane	4	84,129,172,203	79,125,170,202	84,129,172,203
	5	69,107,143,180,205	57,104,141,182,203	70,108,144,180,205
Starfish	4	60,101,138,187	60,101,139,188	59,99,136,186
	5	52,86,117,150,194	26,39,102,158,178	51,86,117,150,189
Snake	4	70,102,130,167	70,100,129,168	70,102,130,169
	5	64,91,114,139,174	66,94,116,140,176	65,93,117,142,178
Zebra	4	84,112,142,199	84,111,141,199	85,113,143,200
	5	79,104,127,158,209	76,101,123,150,204	75,98,121,149,203

Table 5: Comparison of MSSIM index values of the segmented images obtained by GWO, BFO, and PSO based methods. A Higher value of MSSIM index indicates better quality of the segmented image.

Test images	*m*	MSSIM index values of Kapur methods			MSSIM index values of Otsu methods		
		GWO	BFO	PSO	GWO	BFO	PSO

Lena	4	**0.7765**	0.7716	0.7752	**0.7814**	0.7806	**0.7814**
	5	**0.7895**	0.7866	0.7869	**0.8069**	0.8035	0.8048
Peppers	4	**0.7227**	0.7206	0.7214	**0.7214**	0.7206	0.7206
	5	**0.7497**	0.7472	0.7482	**0.7526**	0.7435	0.7504
Baboon	4	**0.8014**	0.8012	**0.8014**	**0.8577**	0.8572	**0.8577**
	5	**0.8590**	0.8556	0.8558	**0.8868**	0.8827	0.8852
Man	4	**0.7093**	0.7022	0.7082	**0.7257**	0.7246	**0.7257**
	5	**0.7102**	0.7091	0.7089	**0.7731**	0.7682	0.7712
Airplane	4	**0.8605**	0.8596	0.8598	**0.8423**	0.8402	**0.8423**
	5	**0.8725**	0.8702	0.8722	**0.8622**	0.8604	0.8610
Starfish	4	**0.7455**	0.7441	0.7450	**0.7732**	0.7721	0.7727
	5	**0.7957**	0.7888	0.7929	**0.8188**	0.8177	0.8179
Snake	4	**0.8138**	0.8056	0.8102	**0.8708**	0.8700	0.8706
	5	**0.8496**	0.8469	0.8473	**0.9017**	0.8987	0.9009
Zebra	4	**0.7934**	0.7912	0.7933	**0.8393**	0.8369	0.8374
	5	**0.8041**	0.8002	0.8039	**0.8625**	0.8587	0.8621

Table 6: Mean values of the objective functions obtained by GWO, BFO, and PSO based methods at 100 run each

Images	*m*	Mean values of Kapur'sentropy			Mean Values of Otsu function		
		GWO	BFO	PSO	GWO	BFO	PSO
Lena	4	**18.0013**	17.9947	18.0008	**2191.84**	2189.76	**2191.84**
	5	**20.6073**	20.5675	20.6047	**2217.34**	2204.44	2217.21

Peppers	4	**18.7284**	18.7204	18.7049	**3151.98**	3151.63	**3151.98**
	5	**21.3881**	21.3517	21.3819	**3195.72**	3195.05	3195.66
Baboon	4	**18.1281**	18.1152	**18.1281**	**1692.14**	1689.41	**1692.14**
	5	**20.7854**	20.7450	20.7851	**1717.81**	1708.41	1717.75
Man	4	**18.6738**	18.6630	18.6728	**3210.62**	3209.95	**3210.62**
	5	**21.4388**	21.3527	21.3619	**3256.52**	3253.68	3256.41
Airplane	4	**18.3112**	18.3024	**18.3112**	**2069.94**	2065.89	**2069.94**
	5	**20.9030**	20.8609	20.9024	**2096.12**	2094.38	2096.04
Starfish	4	**19.0566**	19.0125	19.0421	**2869.08**	2867.25	2868.82
	5	**21.8045**	21.7824	21.7836	**2915.59**	2915.43	2915.46
Snake	4	**18.6340**	18.5632	18.6291	**1285.37**	1285.22	1285.31
	5	**21.3807**	21.2794	21.3472	**1315.13**	1315.02	1315.07
Zebra	4	**17.8819**	17.6284	17.7946	**1590.06**	1589.38	1589.76
	5	**20.4229**	20.3971	20.4135	**1617.56**	1617.19	1617.42

Table 7: Standard deviation of the objective functions obtained by GWO, BFO, and PSO based methods at 100 run each

Test images	*m*	Standard deviation of Kapur's entropy			Standard deviation of Otsu's function		
		GWO	BFO	PSO	GWO	BFO	PSO
Lena	4	**0.0067**	0.0190	0.0168	0.2101	**0.1217**	0.2614
	5	**0.0134**	0.1605	0.0313	**0.6718**	1.0531	0.9487
Peppers	4	**0.0028**	0.0233	0.0910	**0.0455**	0.1904	0.8347
	5	**0.0110**	0.0278	0.1178	**0.0965**	0.2617	1.3599

Baboon	4	**0.0026**	0.0136	0.0187	**0.2658**	1.0241	0.5534
	5	**0.0059**	0.0158	0.04554	**0.3430**	1.1605	0.6295
Man	4	**0.0011**	0.0526	0.0406	**0.0222**	0.2988	0.1636
	5	**0.0421**	0.0552	0.0811	**0.0317**	1.6689	1.3403
Airplane	4	**0.0021**	0.0264	0.0314	**0.2580**	1.9501	0.4872
	5	**0.0074**	0.0274	0.0362	**0.5038**	2.0132	1.5311
Starfish	4	**0.0018**	0.0224	0.0254	**0.4234**	0.4351	0.6027
	5	**0.0028**	0.0398	0.0501	**0.5790**	1.5473	1.6019
Snake	4	**0.0031**	0.0181	0.0209	0.6584	0.7323	**0.2477**
	5	**0.0138**	0.0210	0.0483	**0.9166**	0.9924	1.1310
Zebra	4	**0.0046**	0.0358	0.0107	**0.4894**	0.7374	0.5327
	5	**0.0104**	0.0997	0.0149	**1.0176**	2.1359	1.8710

Table 8: Statistical analysis (Wilcoxon Rank sum Test) of the metaheuristic algorithm based multilevel thresholding methods. 100 run for each of the cases. p= probability of the statistic. h=1 means the null hypothesis can be rejected at 5% level of significance. A p value where GWO performs better than the other algorithms is marked with '+' in the superscript and with '#' where GWO performance is similar or worse than the other algorithms.

Image	*m*	Otsu based method				Kapur based method			
		GWO vs. PSO		GWO vs. BFO		GWO vs. PSO		GWO vs. BFO	
		p	h	p	h	p	h	p	h
Lena	4	$>0.05^{\#}$	0	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
	5	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
Peppers	4	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
	5	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
Baboon	4	$>0.05^{\#}$	0	$<0.05^{+}$	1	$>0.05^{\#}$	0	$<0.05^{+}$	1
	5	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
Man	4	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
	5	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
Airplane	4	$>0.05^{\#}$	0	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
	5	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
Starfish	4	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1
	5	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1	$<0.05^{+}$	1

Snake	4	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1
	5	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1
Zebra	4	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1
	5	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1

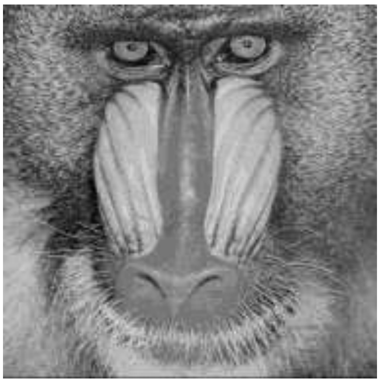
Table 9: Average CPU time (millisecond) of GWO, BFO, and PSO based methods at 100 run each

Test images	*m*	Kapur method			Otsu method		
		GWO	BFO	PSO	GWO	BFO	PSO
Lena	2	35.9	88.3	**29.1**	30.5	89.8	**25.8**
	3	61.5	102.0	**34.4**	51.6	88.3	**27.3**
	4	100.8	103.1	**46.2**	77.3	162.5	**29.7**
	5	146.9	181.2	**47.0**	114.1	279.7	**40.6**
Peppers	2	36.7	88.6	**32.0**	32.8	90.6	**25.4**
	3	62.5	125.8	**32.8**	52.3	82.8	**27.3**
	4	98.4	101.6	**44.5**	78.1	175.0	**30.5**
	5	148.4	181.1	**49.2**	110.2	268.0	**40.0**
Baboon	2	37.5	92.7	**28.9**	31.3	93.8	**25.8**
	3	64.8	106.3	**32.8**	48.4	96.9	**28.1**
	4	103.1	107.3	**46.8**	77.3	184.4	**32.0**
	5	151.6	174.2	**49.2**	114.8	293.0	**41.4**
Man	2	37.5	89.1	**28.1**	30.6	91.4	**25.8**
	3	65.6	96.9	**32.8**	48.4	84.4	**28.1**
	4	103.1	99.2	**46.1**	76.6	178.1	**37.5**
	5	153.1	169.5	**53.1**	109.4	269.5	**39.8**
Airplane	2	35.3	83.6	**29.7**	32.0	93.8	**26.6**
	3	61.7	95.3	**32.8**	48.4	94.4	**28.1**

	4	98.4	100.8	**46.9**	75.0	178.9	**38.3**
	5	151.6	186.7	**49.2**	107.0	288.3	**41.4**
Starfish	2	30.5	90.6	**20.8**	22.9	87.5	**18.0**
	3	56.3	98.3	**23.4**	40.6	88.2	**21.1**
	4	93.0	109.6	**36.7**	64.8	179.7	**28.1**
	5	146.1	179.2	**37.5**	99.2	273.4	**32.0**
Snake	2	30.5	87.8	**22.7**	23.1	85.2	**18.2**
	3	55.5	94.8	**25.0**	40.6	86.4	**21.1**
	4	93.8	96.9	**35.0**	63.4	186.7	**28.9**
	5	142.2	173.4	**36.6**	101.6	282.8	**32.0**
Zebra	2	29.7	80.5	**22.7**	23.4	86.7	**14.8**
	3	54.7	86.3	**23.4**	39.8	87.3	**18.2**
	4	90.6	89.1	**34.4**	64.8	170.8	**19.8**
	5	139.1	161.4	**37.3**	98.4	281.3	**24.2**

(a) (b)

(c) (d) (e)

(f) (g) (h)

Figure.1. Benchmark test images used to perform experiments. These images are taken from USC-SIPI (a-e) and BSD 500 (f-h) image databases. The images (a)-(e) are of size 512x512 each. And the images (f)-(h) are of size 481x321 each. (a) Lena, (b) Peppers, (c) Baboon, (d) Man, (e) Airplane, (f) Starfish, (g) Snake, and (h) Zebra.

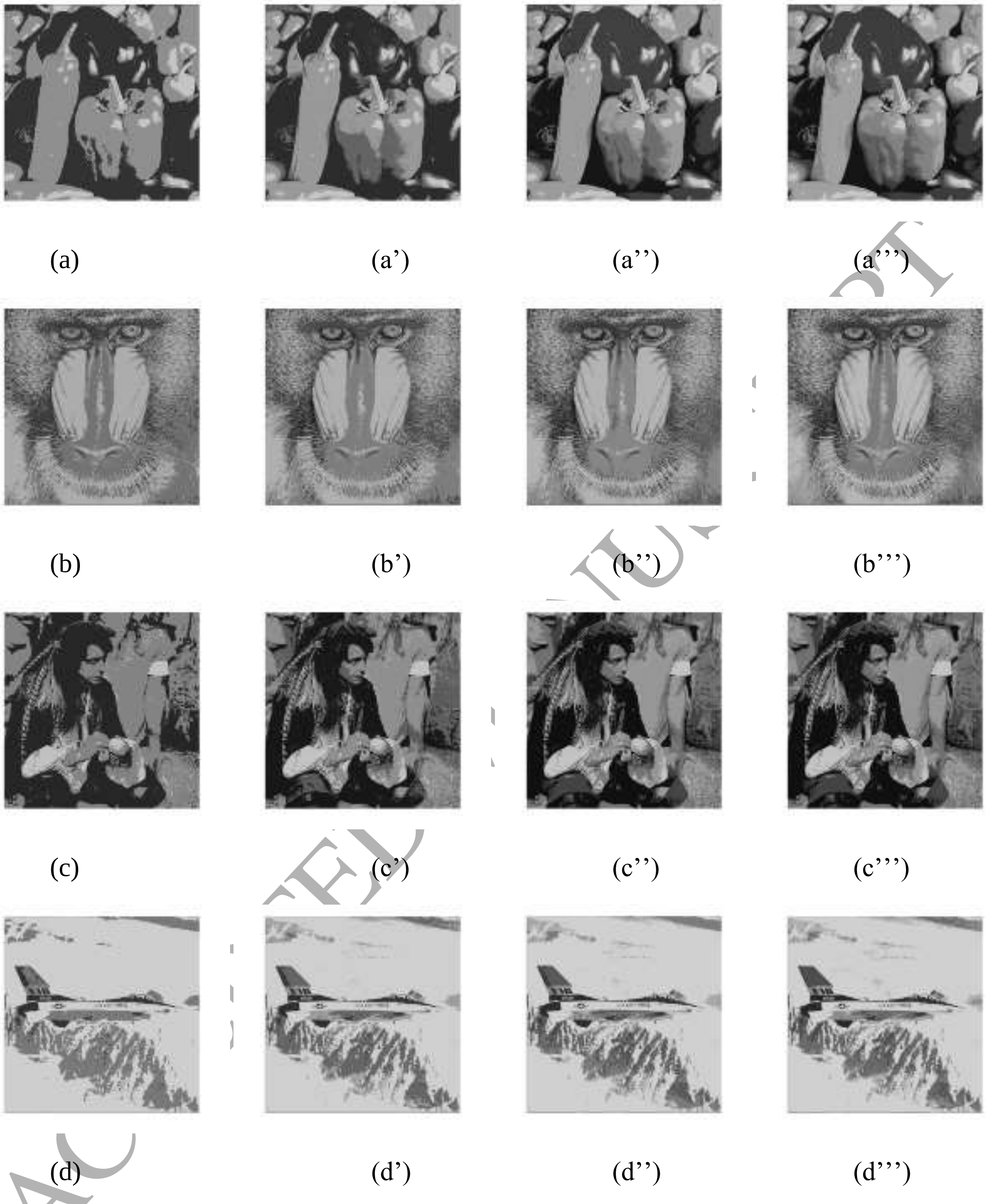
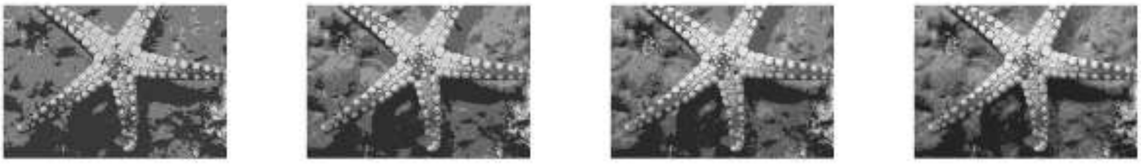


Figure.2. Segmented images obtained by GWO-Kapur multilevel thresholding method. (a)-(g) represent 3-level thresholding, (a’)-(g’) represent 4-level thresholding, (a’’)-(g’’) represent 5-level thresholding, and (a’’’)-(g’’’) represent 6-level thresholding.

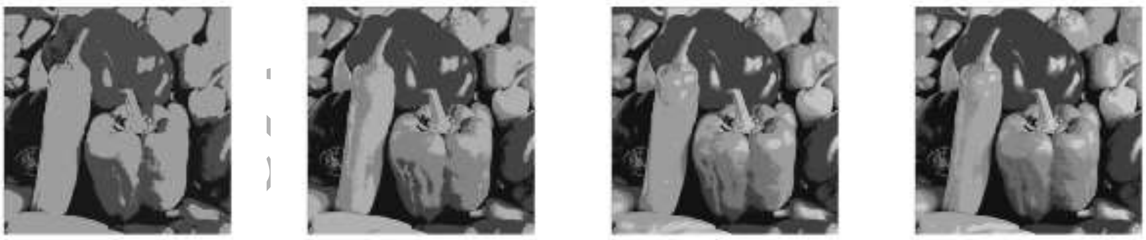


(e) (e') (e'') (e''')

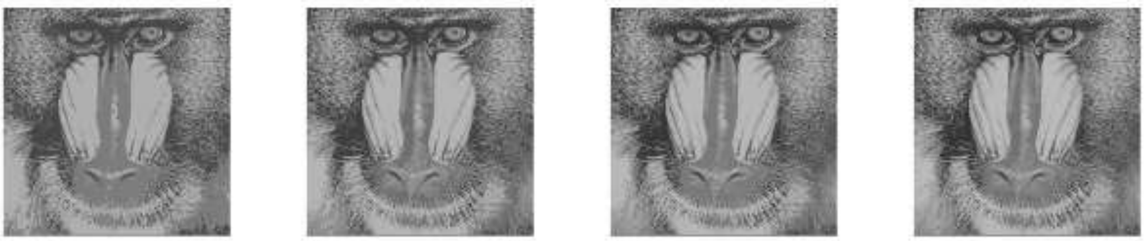
(f) (f') (f'') (f''')

(g) (g') (g'') (f''')

Figure.2. (continued)



(a) (a') (a'') (a''')

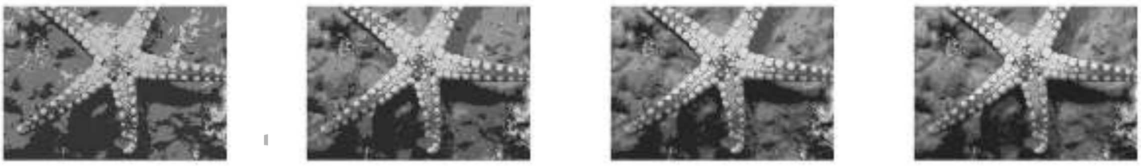


(b) (b') (b'') (b''')

(c) (c') (c'') (c''')

(d) (d') (d'') (d''')

Figure.3. Segmented images obtained by GWO-Otsu multilevel thresholding method. . (a)-(g) represent 3-level thresholding, (a')-(g') represent 4-level thresholding, (a'')-(g'') represent 5-level thresholding, and (a''')-(g''') represent 6-level thresholding.



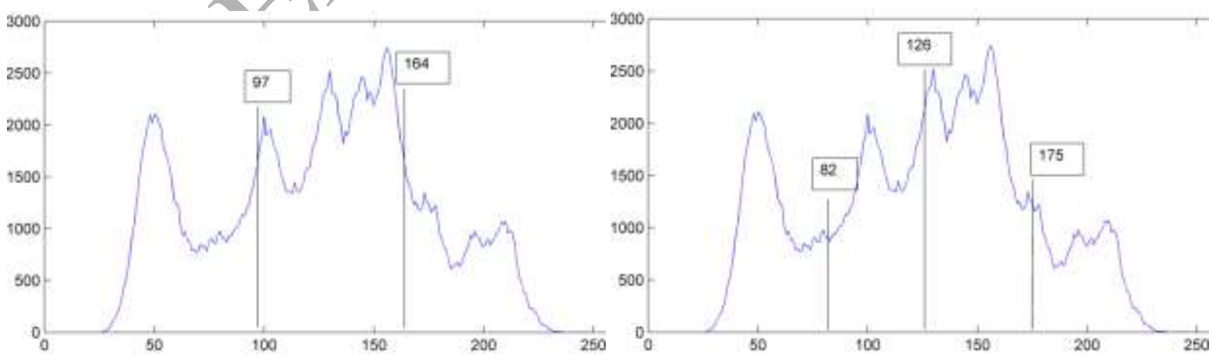
(e) (e') (e'') (e''')

(f) (f') (f'') (f''')

(g) (g’) (g’’) (g’’’)

Figure.3. (continued)

(a) (b)

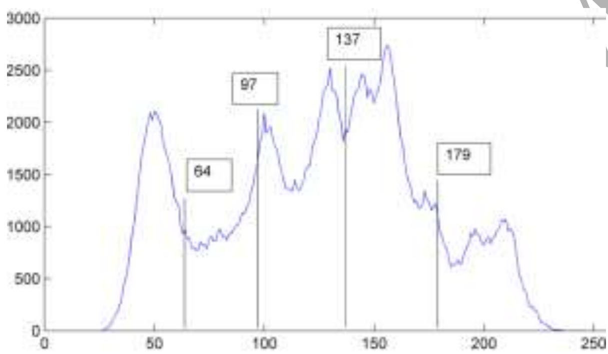


(a’) (b’)

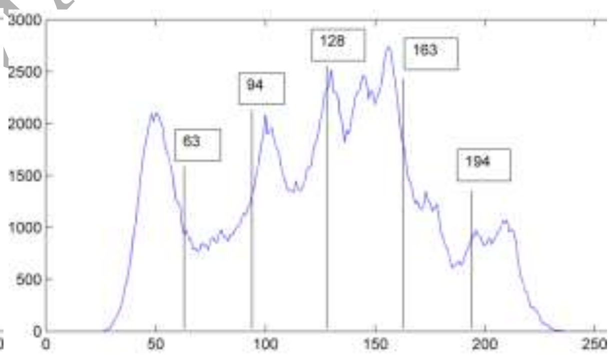
Figure.4. Segmentation results for Lena image by GWO-Kapur method. (a)-(d) represent segmented images into three, four, five, and six classes respectively. (a')-(d') represent the fitted histogram and the thresholds for the segmented images (a)-(d) respectively.

(c)

(d)



(c')

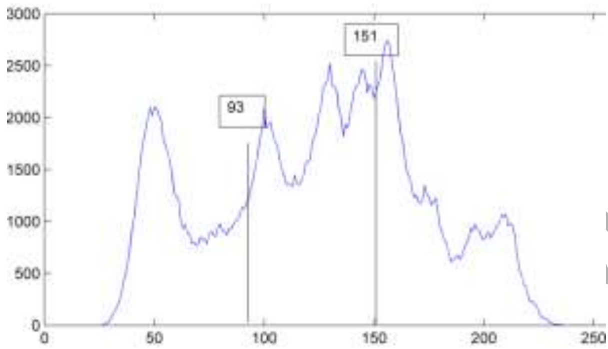


(d')

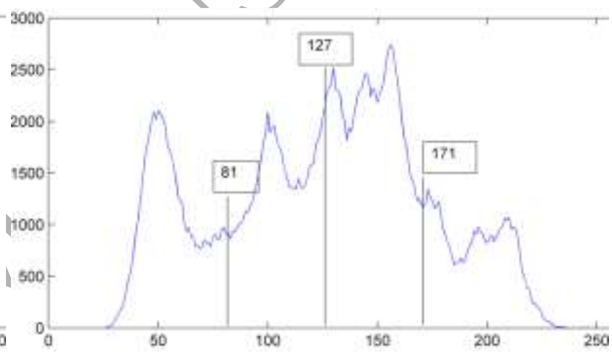
Figure.4. (continued)

(a)

(b)



(a’)

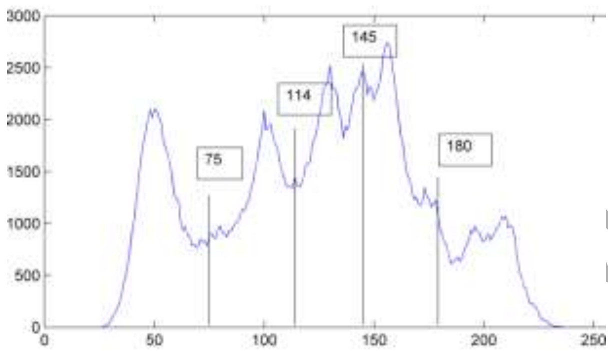


(b’)

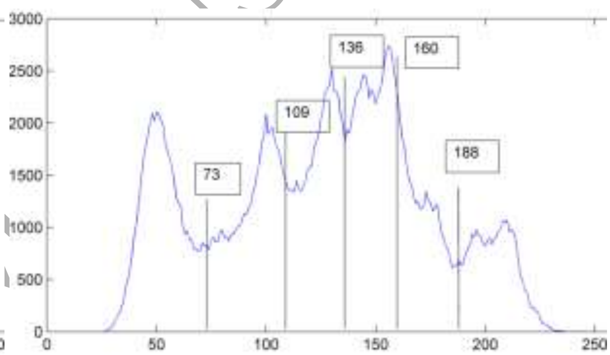
Figure.5. Segmentation results for Lena image by GWO-Otsu method. (a)-(d) represent segmented images into three, four, five, and six classes respectively. (a’)-(d’) represent the fitted histogram and the thresholds for the segmented images (a)-(d) respectively.

(c)

(d)



(c')



(d')

Figure.5. (continued)