Antlion optimization and Whale optimization Algorithm for multilevel thresholding segmentation

Srikrishna Iyer[1]
School of Electronics Engineering
VIT University, Vellore
Tamil Nadu, India
srikrishna.iyer2015@vit.ac.in

Adwayt Pradeep Nadkarni[2]
School of Electronics Engineering
VIT University, Vellore
Tamil Nadu, India
adwaytpradeep.nadkarni2015@vit.ac.in

Padmini T.N.[3]
School of Electronics Engineering
VIT University, Vellore
Tamil Nadu, India
tnpadmini@vit.ac.in

Abstract—**Multi-level image segmentation is a critical task in image processing that involves multiple threshold values. As the high computational cost of an exhaustive search is inefficient and cumbersome, the optimal thresholds algorithms make for a better path to venture; hence a comparison of optimization algorithms to set the optimal thresholds is highly essential and beneficial. In this paper, a practical comparison is made to deduce the best optimization technique amongst the whale optimization and antlion optimization algorithm, to solve the multilevel threshold problem, to find the optimal multilevel thresholds. Otsu's function is maximized to perform optimized thresholding-based image segmentation. The experimental results showed that the Antlion optimization algorithm gave better performance in solving the problem for higher level multi-thresholding.**

Keywords—Multilevel image segmentation, multi-thresholding, whale optimization, antlion optimization algorithm, Otsu's function, image optimization

I. Introduction

Image segmentation has a major application in image processing and computer vision. It is the process of segmenting or dividing a digital image into multiple segments. This paper utilizes the most commonly used thresholding method for segmentation. Segmentation through multi-level thresholding segments an image into more than two thresholds which can be effectively used in complex image processing functions like object recognition and edge detection [1]. On the other hand, metaheuristic multi-thresholding algorithms do a better job at optimizing the task even when the number of thresholds increases. Some swarm intelligent techniques applied for optimizing the multi-threshold problem include Particle Swarm Optimization(PSO)[4], Genetic Algorithm[5], Ant Colony Optimization[6], Social Spider Optimization (SSO)[7], Firefly algorithm(FA)[8], Fuzzy adaptive swallow swarm optimization[9] and Moth Flame optimization [10]. For instance, Mohamed Abd El Aziz, Ahmed A. Ewees & Aboul Ella Hassanien (2017) presented an algorithm based on MFO (Moth Flame optimization) which utilized the Otsu's function to find optimal positions (thresholds) which maximized the fitness function. This paper introduces the application of two new meta-heuristic algorithms, whale optimization algorithm (WOA) [11] and antlion optimization algorithm (ALO) [12]. The WOA algorithm is mathematically modelled to mimic the behaviour of humpback whales to solve the global optimization problem. In this method, whales utilize two main approaches to hunt for its prey namely, shrinking encircling and bubble-net feeding method [13-15]. On the other hand, Mirjalili and Lewis (2016) had presented ALO (Ant-Lion optimization) algorithm which which is based on the hunting behaviour of antlions. This unique feeding behaviour is characterised by antlions randomly forming large pits in the sand to capture ants falling into such deep traps. The ant falls into the pit where the antlion hides at the bottom waiting for its prey to be captured. Therefore, the main contributions of this research include (1) Applying Whale optimization Algorithms (WOA) and the Antlion Optimization algorithm for selection of optimal multi-thresholds for multi-level image segmentation and (2) conduct a comparative study between WOA and ALO with implemented algorithms which include SSO, FA and FASSO. The simulation results of the WOA and ALO algorithms are compared using fitness function values, the peak to signal noise ratio (PSNR) and the structural similarity index (SSIM). The paper is arranged as follows: in Section 2, the problem definition of Otsu's method is specified. Later, in Section 3, WOA and MFO algorithm and their workflow are explained in detail. The experimental results based on benchmark images are illustrated in Section 4. Finally, the last section includes the conclusion and future scope.

II. Methodology

Consider a gray level image I which can be split into $M + 1$ groups for multi-level thresholding. Therefore, the t_m, where $m = 1, 2, 3\ldots\ldots M$ thresholds are required to split I to subgroups S_m in the following equation:

$$S_0 = \{I(i,j) \in \mathrm{I} \mid 0 \leq \mathrm{I(i,j)} \leq t_1 - 1\}$$

$$S_1 = \{I(i,j) \in \mathrm{I} \mid 0 \leq \mathrm{I(i,j)} \leq t_1 - 1\}$$

$$\ldots \ldots$$

$$S_m = \{I(i,j) \in \mathrm{I} \mid t_m \leq \mathrm{I(i,j)} \leq t_1 - 1\} \quad (1)$$

Where $I(i,j) is\ (i,j)^{th}$ pixel value and L is the gray levels of $I \in [0, L-1]$. The threshold values that form these groups

978-1-5386-8190-9 /19/$31.00 ©2019 IEEE

S_m is the main objective which can be found by maximizing equation:

$$t_1^*, t_2^*, \dots . t_M^* = \max\ O(t_1, \dots t_m) \quad (2)$$

Where $O(t_1, \dots t_m)$ is the Otsu's function for image thresholding which is defined as:

$$O = \sum_{i=0}^{M} B_i (n_i - n_1)^2 \quad (3)$$

$$B_i = \sum_{j=t_i}^{t_{i=1}-1} P_j \quad (4)$$

$$n_i = \sum_{j=t_i}^{t_{i=1}-1} i \frac{P_j}{B_j} \quad (5)$$

Where $P_i = {h_i}/{N_p}$, n_1 is the average intensity of image I with $t_0 = 0$ and $t_{M+1} =$ L. h_i is frequency and P_i is probability of the i^{th} gray level.

III. Proposed Algorithm For Multilevel Thresholding Problem

To optimize the process of multilevel thresholding, two algorithms are proposed in this section which maximizes the Otsu's function.

A. Whale Optimization Algorithm

In this paper, the spiral and shrinking encircling feeding manoeuvre is implemented to perform optimization. Using (2) and (3), optimized threshold values maximizing the Otsu's function are obtained using the WOA. The image being the input, dimension of each whale is the number of thresholds specified for performing mulithresholding .The optimal position X^* represents the optimal thresholds. The whales are represented as a matrix of real values mapped to thresholds as follows [11]:

$$X_{i,j} = \begin{bmatrix} x_{1,1} & x_{1,2} \cdots & x_{1,M} \\ \vdots & \vdots & \vdots \\ x_{N,1} & x_{N,2} \cdots & x_{N,M} \end{bmatrix} \quad (6)$$

Where $x_{i,1}, x_{i,2} \dots x_{i,M}$ corresponding to thresholds $t_1, t_2 \dots . t_M$ for a population of N whales. Each whale position is generated randomly in range [f_{min}, f_{max}], where f_{min} and f_{max} are the minimum and the maximum gray level values in the image histogram respectively. The positions are randomly generated by the following equation [11]:

$$x_{i,j} = f_{min} + rand(0, 1) \times (f_{max} - f_{min}) \quad (7)$$

Where, $x_{i,j} \in X_i$, and $j = (1,2 \dots M)$. The fitness function O_g for every whale and its corresponding optimal position X^* is computed. The whales optimize the best position X^* using shrinking encircling method as: [11]:

$$\overrightarrow{D_i} = |\vec{C} \otimes \vec{X}^*(v) - \vec{X}_i(v)| \quad (8)$$

$$\vec{X}_i(v+1) = \vec{X}^*(v) - \vec{A} \otimes \overrightarrow{D_i} \quad (9)$$

Where, $\otimes$ is element-wise multiplication, $\overrightarrow{D_i} = |\vec{X}^*(v)$ -$\vec{X}_i(v)|$ for the current iteration v. $\vec{A}$ and $\vec{C}$ are coefficient vectors computed as follows [11]:

$$\vec{A} = 2\vec{a} \otimes \vec{r} - \vec{a} \quad (10)$$

$$\vec{C} = 2\vec{r} \quad (11)$$

Where, $\vec{r}$ is a random vector in [0,1], the value of $\vec{a}$ is linearly decreased from 2 to 0 with iterations as:

$$\vec{a} = \vec{a} - v \frac{\vec{a}}{v_{max}} \quad (12)$$

Where, v_{max} is the maximum number of iterations. The spiraling bubble-net method for updating whale positions around X^* is given as [11]:

$$\vec{X}_i(v+1) = \overrightarrow{D'} \otimes e^{bh} \otimes \cos(2\pi h) + \vec{X}^*(v) \quad (13)$$

Where, $\overrightarrow{D'} = |\vec{X}^*(v)$ -$\vec{X}_i(v)|$, the shape of a logarithmic spiral is controlled by 'b' and $h \in [-1,1]$. The whales undergo optimization around the best position $\vec{X}^*$ simultaneously through models that include both shrinking circle and spiral path; therefore, (5), (6) and (10) are combined as [11]:

$$\vec{X}_i(v+1) = \begin{cases} \vec{X}^*(v) - \vec{A} \otimes \overrightarrow{D_i} & if\ p < 0.5 \\ \overrightarrow{D'} \otimes e^{bh} \otimes \cos(2\pi h) + \vec{X}^*(v) & if\ p \geq 0.5 \end{cases} \quad (14)$$

Where $p \in [-1,1]$ is probability of choosing either model. A slight modification to the encircling method is made so as to perform a global search during the exploration phase. Instead of using the best whale chosen up till now, position of a search agent is randomly selected and whale positions are updated using equation and the vector coefficient $\vec{A}$ is made greater than 1.

$$D_i = |\vec{C} \otimes \vec{X}_{rand} - \vec{X}_i(v)| \quad (15)$$

$$\vec{X}_i(v+1) = \vec{X}_{rand} - \vec{A} \otimes \overrightarrow{D_i} \quad (16)$$

Where X_{rand} is a random position vector from a population of whales. Thus, If p > 0.5 then use (13) otherwise use either (8) and (9) or (15) and (16) based on the value of A. Fig. 1 illustrates the flowchart of WOA implemented in this paper for multilevel thresholding.

B. Antlion Optimization Algorithm

The ALO algorithm mimics the interaction between antlions and ants in the trap [12]. Firstly, the random walk of the ant can be represented as:

$$\overrightarrow{X_i} = [0, \pi(2^* \vec{r}(v_1)\text{-}1), \pi(2^* \vec{r}(v_2)\text{-}1) \dots \pi(2^* \vec{r}(v_n)\text{-}1)] \quad (17)$$

Where Π function provides the cumulative sum of the random positions of the ants, ν is the random function and n is the number of iterations.

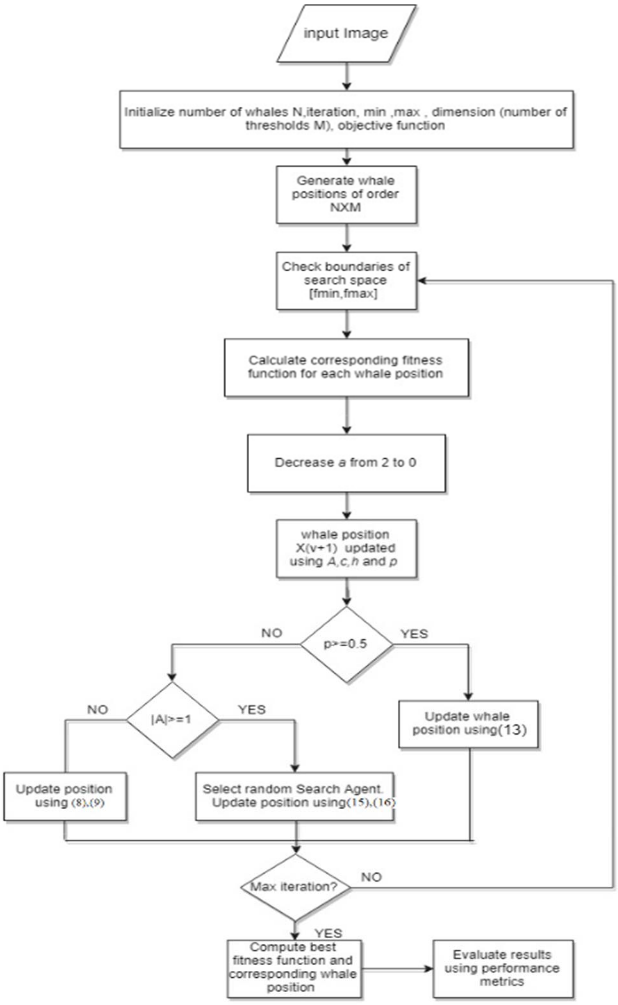

Fig. 1: Flowchart for multi-thresholding segmentation using WOA

The ants, however, keep moving and updating their positions and the positions defined in (14) are continuously updated but restrained within a well-defined search space. As a result, the equation is localized and normalized by:

$$X_j^{\nu} = \beta_j + \frac{\left(x_j^{\nu} - \alpha_j\right) \times (\gamma_j - \beta_j)}{(\gamma_j - \alpha_j)} \tag{18}$$

Where X_j^{ν} is the normalized, updated position of the v^{th} ant for j^{th} index within the search space, α_j is the minimum random walk possible, δ_j is the maximum random walk possible, β_j is the minimum of j^{th} variable possible for v^{th} iteration of ant positions and γ_j is the maximum of j^{th} variable possible for v^{th} iteration of ant positions. This guarantees that the values of random walks are completely random and well defined within the search space. The hiding Antlions waiting for prey/ants (objective/fitness function values) within search space are defined with their own positions and fitness function with respect to antlion given by $F_{POS\text{-}ANTLION}$ and $F_{OP\text{-}ANTLION}$:

$$F_{OP-ANT} = \begin{bmatrix} f([x_{1,1}, x_{1,2}, \cdots, x_{1,M}]) \\ f([x_{2,1}, x_{2,2}, \cdots, x_{2,M}]) \\ . \\ . \\ f([x_{N,1}, x_{N,2}, \cdots, x_{N,M}]) \end{bmatrix} \tag{19}$$

$$F_{OP-ANTLION} = \begin{bmatrix} f\left(\left[x'_{1,1}, x'_{1,2}, \cdots, x'_{1,M}\right]\right) \\ f\left(\left[x'_{2,1}, x'_{2,2}, \cdots, x'_{2,M}\right]\right) \\ . \\ . \\ f\left(\left[x'_{N,1}, x'_{N,2}, \cdots, x'_{N,M}\right]\right) \end{bmatrix} \tag{20}$$

Where $x_{i,1}, x_{i,2} \dots x_{i,M}$ and $x'_{i,1}, x'_{i,2} \dots x'_{i,M}$ corresponding to positions of ant and antlion respectively, N is the number of ants/ antlions and M is the number of variables/dimensions. As the defined antlions are of different dimensions and given the fact that stronger antlions or antlions with greater fitness functions make bigger pits, the chances that an ant will get caught in one is higher. Such antlions whose traps capture every ant per iteration are called elite antlions. Thus an ant that randomly walks around antlions is represented by:

$$Ant_j^{\nu} = \frac{(X_{AL}^{\nu}) + (X_{EL}^{\nu})}{2} \tag{21}$$

Where Ant_j^{ν} is the random walk of the ant, X_{AL}^{ν} is the random walk of the antlion and X_{EL}^{ν} is the random walk of the elite antlion and v is the iteration. The ALO algorithm is required to utilize the roulette wheel operator for selecting antlions based on their fitness during optimization [12]. To trap the ants however, the random walk of ants must be affected by the antlion's traps. This can be mathematically defined as:

$$\beta_j^{\nu} = \beta^{\nu} + AL_j^{\nu} \tag{22}$$

$$\gamma_j^{\nu} = \gamma^{\nu} + AL_j^{\nu} \tag{23}$$

It is now well established that antlions build traps proportional to its associated fitness function. If an ant that has been following a random path, falls into a trap, the antlion starts shooting sand outwards by creating an avalanche thereby preventing an escape and drawing the prey closer. Mathematically, the radius of the hypersphere is adaptively decreased by manipulating β andγ. It is defined as:

$$\beta^{\nu} = \frac{\beta^{\nu}}{\mathrm{I}} \tag{24}$$

$$\gamma^{\nu} = \frac{\gamma^{\nu}}{\mathrm{I}} \tag{25}$$

$$I = 10^{a} \frac{v}{\mathrm{T}} \tag{26}$$

Where, v is the current iteration value, a is the constant that depends on v called level of exploitation; T is the maximum possible iterations considered. The final stage of antlion hunting is when the ant reaches the bottom (center) of the pit and is caught by antlion. Mathematically, the fitter ant that gets trapped needs to be correspondingly fitter than the antlion. After this is done, the position of antlion is updated in

(27) to enhance chances to catch prey and all stages of the hunt are repeated only if f (A_j^v) > f (AL_j^v):

$$AL_j^v = A_j^v \tag{27}$$

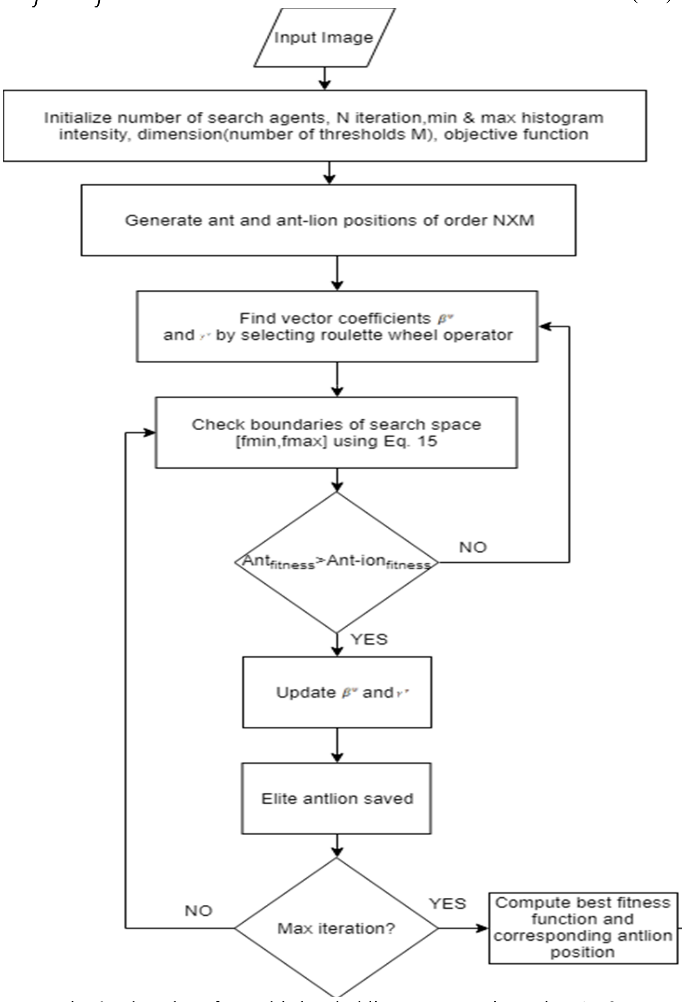

Fig. 2: Flowchart for multi-thresholding segmentation using ALO

IV. Results

In this section, the benchmark images are illustrated and the parameter for each algorithm is summarized. The performance metrics used to evaluate the quality of the segmentation process are:

A. *Benchmark images*

To test the proposed algorithms, eight grayscale images are used [16]. These images identified as monument, fireFig.hter, boatman, corn, starfish, pepper, sailboat, airplane, baboon is illustrated in Fig. 3-7.

B. *Experimental Settings*

A result comparison of the proposed methods WOA and ALO are performed with existing algorithms: SSO, FA, and FASSO. For uniformity, population size is fixed at 20, the number of iterations is 80 (with each algorithm run 10 times to obtain average value). The dimension of each search agent is the number of thresholds and input to each algorithm is the image. The parameters of each algorithm used in this paper are illustrated in Table 1. The experiments were carried out for four thresholds. Simulations were tested using "MATLAB R2016a" and implemented on a "Windows 64 bit" PC with Intel i7(2.6 GHz) processor and 16 GB memory.

Fig. 3: Benchmark test images "Monument" (left) , "Fire-fighter"(right)

Fig. 4: Benchmark test image "Boatman"(left), "Corn"(right)

Fig. 5: Benchmark test image "Starfish"(left), "Pepper"(right)

Fig. 6: Benchmark test image "Sailboat"(left), "Airplane"(right)

Fig. 7: Benchmark test image "Baboon"

C. *Quality metrics for the segmented images*

The accuracy of segmented image is based on three measures of performance which are briefly explained as follows:

a) Fitness function value: or evaluation function acts as a medium to deduce the closest or most optimum value of the desired problem.

b) Peak Signal to Noise Ratio (PSNR): It refers to the quality of the reconstructed image as a higher PSNR value indicates that the reconstructed image after segmentation is of higher quality. It is defined as:

$$PSNR = 20\log_{10}\frac{255}{RMSE}\ dB \qquad (26)$$

Where RMSE is the mean-squared error:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(I(i,j)-I'(i,j))^2}{MXN}} \qquad (27)$$

Where original and segmented images are *I* and *I'* respectively

c) Structural similarity index measure (SSIM) [17]: A higher value of SSIM denotes that the segmented image acheived higher performance.

$$SSIM(I,I') = \frac{(2\mu_I\mu_{I'}+c_1)(2\sigma_{I,I'}+c_2)}{(\mu_I^2+\mu_{I'}^2+c_1)(\sigma_I^2+\sigma_{I'}^2+c_2)} \qquad (28)$$

Where mean intensity of image *I* and *I'* are μ_I and $\mu_{I'}$. σ_I and $\sigma_{I'}$ is the standard deviation of *I* and *I'* respectively; $\sigma_{I,I'}$ is covariance of *I* and *I'*. *c1* and *c2* are constants whose values are *c1*=6.5025 and *c2*=58.52252[18].

TABLE I. Parameters set for all algorithms

Algorithm	Parameters	Value
WOA	*a*	[0, 2]
	b	1
	l	[−1, 1]
SSO	Probabilities of attraction or repulsion (*pm*)	0.7
	Lower female percent	65
	Upper female percent	90
FASSO	γ_{FA}	0.7
	β_{FA}	1
	α_{FA}	0.8
	Probabilities of attraction or repulsion (*pm*)	0.7
	Lower female percent	65
	Upper female percent	90
FA	γ_{FA}	0.7
	β_{FA}	1
	α_{FA}	0.8
ALO	*W*	[2,6]

D. Results

The results of the proposed algorithms are illustrated in Tables 2-6 and its comparison in fig. 8-10. Also, the segmented images using the optimal threshold values obtained from each of the algorithms are as shown in Fig.11-19. Table 2 shows the average fitness values obtained by WOA and ALO compared to values found using existing algorithms [19, 23]. Table 2 and Fig. 8 indicate that for all threshold values, ALO had the highest fitness function as compared to all the existing algorithms. For lower thresholds, M=2 & 3, WOA had the second highest fitness function while for higher thresholds M=4 & 5, FASSO performed second best. The average PSNR values as mentioned in table 3 increases with increase in the number of thresholds for all algorithms. From Fig. 9, we can observe that for higher threshold values of M=4 and 5, ALO performs the best with the highest average PSNR values of 19.463 and 19.799 respectively. However, for lower thresholds M=2 and 3, average PSNR values are closely contested between FA and SSO. The SSO has higher PSNR than FA for M=3 while for M=2, PSNR value of FA is slightly greater than SSO.

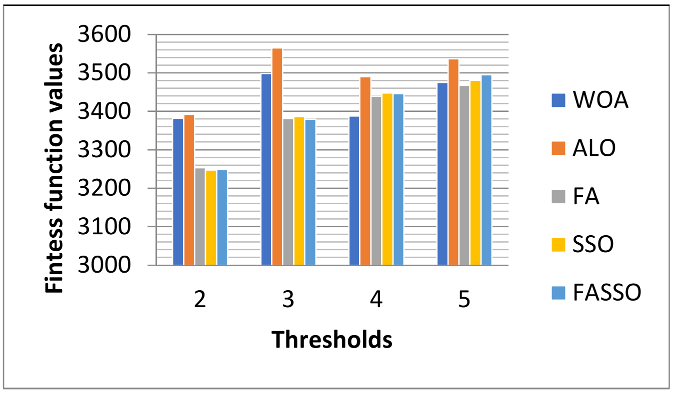

Fig. 8: The fitness function values of test images for threshold values M=2,3,4,5

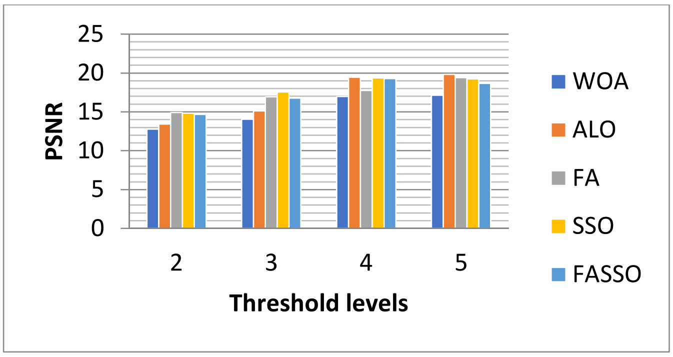

Fig. 9: The average PSNR values of test images for threshold values M=2,3,4,5

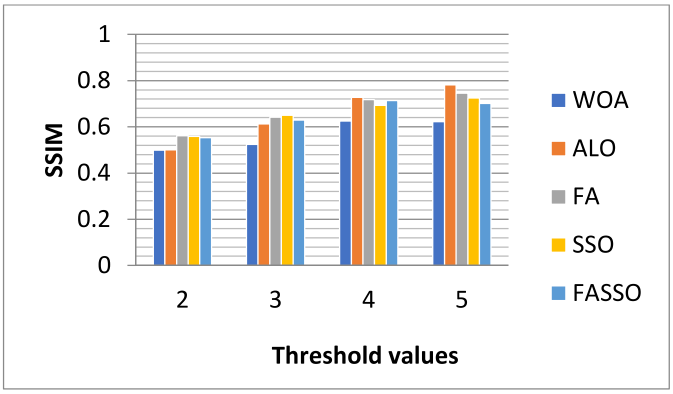

Fig. 10: The average SSIM measures of test images for threshold values, M=2,3,4,5

TABLE II: Average fitness function values

Image	M	Fitness values				
		WOA	ALO	FA	SSO	FASSO
sailboat	2	3317.92	3911.51	5067.0938	5051.35	5070.73
	3	3734.73	4106.32	5219.7479	5229.51	5218.89
	4	4502.78	5936.69	5288.3079	5295.81	5288.41
	5	4106.32	4775.2	5323.709	5333.68	5338.22
pepper	2	2530.54	2854.66	2436.7088	2429.44	2429.6
	3	2229.51	2309.92	2542.0552	2580.05	2513.54
	4	2406.64	3562.15	2624.0218	2620.44	2641.9
	5	2854.66	2111.37	2646.3557	2663.14	2639.61
airplane	2	2229.51	2106.63	1942.9876	1949.21	1937.4
	3	2132.55	2172.37	2023.2169	2016.17	2016.91
	4	2106.63	2144.01	2047.6726	2058.55	2059.62
	5	2172.37	2111.37	2058.1369	2068.4	2096.89
baboon	2	2106.63	2106.63	1545.1063	1539.76	1547.84
	3	2182.5	2144.01	1628.9734	1630.47	1634.05
	4	2144.01	2182.5	1674.267	1644.17	1655.9
	5	2395.39	2118.42	1680.8519	1678.63	1808.29
fire-fighter	2	4775.2	4502.78	4095.949	4099.08	4081.66
	3	4106.32	4106.32	4281.223	4273.42	4287.9
	4	4330.43	4775.21	4370.016	4396.64	4369.28
	5	3232.78	4502.78	4408.191	4446.26	4411.61
corn	2	3232.78	3734.73	3556.535	3560.86	3550.88
	3	3084.16	3392.34	3708.666	3695.66	3719.52
	4	3911.51	3911.51	3757.122	3774.97	3777.06
	5	3392.34	3084.17	3782.561	3810.94	3821.01
boatman	2	5936.69	5267.55	5067.0938	5051.35	5070.73
	3	5820.64	5173.2	5219.7479	5229.51	5218.89
	4	5173.2	5173.2	5288.3079	5295.81	5288.41
	5	5267.55	5936.69	5323.709	5333.68	5338.22
starfish	2	2106.63	2132.55	2093.031	2078.64	2090.94
	3	2679.51	2699.38	2206.062	2205.58	2186.23
	4	2111.37	2854.66	2229.118	2252.47	2252.22
	5	2111.37	2854.66	2265.752	2249.94	2272.46
Monument	2	3734.73	3911.51	3474.179	3468.53	3460.65
	3	3144.01	3232.78	3596.408	3615.77	3620.54
	4	3232.78	3600.7	3673.351	3691.68	3679.44
	5	3392.34	3734.73	3716.655	3742.24	3730.33

TABLE III: Average PSNR values for different segmentation techniques

Image	M	Average PSNR				
		WOA	ALO	FA	SSO	FASSO
sailboat	2	12.65	11.59	12.387	12.9733	12.5911
	3	12.814	13.1	14.7777	16.8979	13.9514
	4	17.25	14.63	18.124	19.2601	18.2197
	5	15.066	17.42	16.7468	17.857	19.2864
pepper	2	11.64	11.97	15.978	16.3635	16.1094
	3	11.94	15.76	18.2383	17.4433	18.1653
	4	15.48	17.163	19.3548	20.0213	19.8527
	5	18.48	17.83	21.0645	20.1802	19.8086
airplane	2	12.21	13.602	14.9586	14.5212	14.0105
	3	13.941	14.618	18.8623	19.3025	19.368
	4	16.094	14.154	20.7689	20.5376	20.9627
	5	19.52	15.003	19.8828	21.6164	15.7763
baboon	2	11.87	12.63	15.8376	15.1832	15.4887
	3	14.275	12.14	16.8615	18.6601	16.9365
	4	16.039	17.48	19.625	18.9211	18.4171
	5	18.15	16.98	19.0558	21.6425	15.8491
fire-fighter	2	13.73	11.453	15.2995	15.3727	15.1418
	3	13.78	14.36	16.7074	16.9031	17.6821
	4	16.021	18.16	18.9827	19.7417	18.3249
	5	19.819	17.583	20.3041	21.1293	19.4707
corn	2	13.35	13.31	13.8011	13.9347	14.0133
	3	13.23	14.401	16.0666	14.9637	15.4732
	4	16.137	14.34	16.6948	15.69	17.5301
	5	14.24	14.739	20.0627	16.6521	17.0998
boatman	2	9.81	12.33	12.387	12.9733	12.5911
	3	11.39	14.307	14.7777	16.8979	13.9514
	4	14.841	15.093	18.124	19.2601	18.2197
	5	19.23	15.003	16.7468	17.857	19.2064
starfish	2	13.088	12.112	17.7184	16.7979	16.4532
	3	16.39	14.328	18.062	18.6063	17.0253
	4	17	17.089	20.523	21.2755	21.9815
	5	15.37	17.735	21.1294	20.807	21.0581
Monument	2	11.82	12.77	15.763	15.376	15.7217
	3	14.74	14.41	17.9375	18.1777	18.4039
	4	15.68	15.63	19.7929	19.5899	19.9914
	5	18.88	17.9	20.7466	19.5003	20.5745

TABLE IV: Average SSIM for different segmentation techniques

Image	M	SSIM Values				
		WOA	ALO	FA	SSO	FASSO
sailboat	2	0.531	0.457	0.5487	0.5852	0.5439
	3	0.487	0.484	0.6348	0.6819	0.6012
	4	0.635	0.612	0.7846	0.7613	0.7749
	5	0.495	0.622	0.7876	0.8035	0.7696
pepper	2	0.545	0.554	0.6261	0.6173	0.6381
	3	0.571	0.644	0.6493	0.6759	0.658
	4	0.57	0.643	0.6495	0.6728	0.6837
	5	0.643	0.693	0.7627	0.7065	0.6859
airplane	2	0.659	0.746	0.7242	0.7015	0.7068
	3	0.754	0.775	0.8004	0.7822	0.7424
	4	0.69	0.768	0.8174	0.7624	0.7676
	5	0.764	0.803	0.7127	0.8	0.7314
baboon	2	0.459	0.47	0.6364	0.6102	0.6227
	3	0.499	0.54	0.6842	0.7267	0.6858
	4	0.622	0.675	0.7747	0.7438	0.7287
	5	0.707	0.604	0.7529	0.821	0.6295
Fire fighter	2	0.494	0.497	0.5397	0.5406	0.5084
	3	0.367	0.371	0.6327	0.6055	0.6457
	4	0.539	0.553	0.6729	0.6649	0.6158
	5	0.664	0.653	0.651	0.6959	0.6675
corn	2	0.37	0.365	0.3915	0.4037	0.4092
	3	0.369	0.42	0.5317	0.4583	0.4934
	4	0.544	0.416	0.5579	0.5035	0.6047
	5	0.411	0.439	0.7661	0.5532	0.5773
boatman	2	0.49	0.655	0.5487	0.5852	0.5439
	3	0.643	0.58	0.6348	0.6819	0.6012
	4	0.585	0.6	0.7846	0.7613	0.7749
	5	0.759	0.601	0.7876	0.8035	0.7696
starfish	2	0.381	0.353	0.5818	0.5624	0.5512
	3	0.49	0.4	0.6375	0.6495	0.6084
	4	0.514	0.526	0.704	0.719	0.7399
	5	0.494	0.555	0.7284	0.7141	0.7313
monument	2	0.384	0.328	0.4564	0.4251	0.4509
	3	0.416	0.69	0.5723	0.5938	0.6221
	4	0.542	0.501	0.7097	0.6428	0.7283
	5	0.609	0.611	0.7645	0.6259	0.7554

TABLE V: Threshold values obtained by the algorithm

Image	M	Threshold values				
		WOA	ALO	FA	SSO	FASSO
sailboat	2	68 138	96 118	108 203	111 188	103 200
	3	95 103 159	97 141 168	95 152 204	81 118 205	84 129 213
	4	58 99 143 209	44 105 170 175	85 129 163 205	34 70 123 198	65 96 152 205
	5	97 116 123 199 219	64 109 135 165 206	66 81 111 55 208	12 86 119 128 182	56 103 140 162 221
pepper	2	72 129	70 137	75 139	65 127	75 145
	3	60 75 131	44 135 198	70 118 165	77 131 176	71 104 153
	4	73 89 112 146	53 94 113 181	44 99 152 180	6 16 111 120	65 110 141 188
	5	62 70 109 151 211	42 78 111 144 175	58 80 128 147 190	42 82 105 119 220	43 78 133 178 205
airplane	2	75 113	79 145	113 175	97 165	117 173
	3	77 113 145	76 107 179	94 140 177	112 152 189	101 150 187
	4	79 99 123 198	81 115 127 152	66 90 159 201	2 12 121 159	78 78 107 140
	5	72 76 113 140 221	78 116 122 143 175	95 112 150 174 200	4 24 126 169 235	28 28 31 139 194
baboon	2	79 130	79 143	87 150	92 134	96 144
	3	82 122 208	57 81 126	99 128 166	86 115 149	53 106 161
	4	43 81 110 196	52 82 122 161	62 89 144 163	0 0 121 127	66 103 138 164
	5	45 85 104 130 202	80 99 106 130 180	71 108 136 169 194	26 26 81 107 155	36 36 71 123 123
Fire fighter	2	64 151	52 99	62 139	62 141	69 139
	3	97 124 158	97 130 198	42 94 142	26 92 161	37 84 164
	4	65 114 127 158	64 102 135 208	41 72 129 167	48 83 125 175	31 97 130 187
	5	40 92 106 150 206	26 99 143 155 185	54 92 137 177 187	42 68 93 149 196	22 72 86 138 199
corn	2	92 157	95 160	92 175	88 168	84 169
	3	91 97 194	93 135 197	66 114 182	84 138 209	77 132 189
	4	58 96 108 176	96 114 135 190	91 123 197 82	122 154 200 59	50 112 155 193
	5	43 117 128 133 171	91 100 114 138 200	37 66 110 179 219	72 114 132 178 204	67 95 120 172 225
boatman	2	105 109	63 120	108 203	111 188	103 200
	3	62 106 112	102 132 156	95 152 204	81 118 205	84 129 213
	4	102 115 139 204	102 115 136 185	85 129 163 205	34 70 123 198	65 96 152 205
	5	63 119 125 158 171	105 120 135 159 197	66 81 111 155 208	12 86 119 128 182	56 103 140 162 221
starfish	2	79 176	77 127	80 144	71 128	62 130
	3	71 125 186	88 125 152	53 128 159	54 124 155	57 134 180
	4	78 123 145 196	70 128 142 180	69 112 141 190	50 94 141 171	63 99 133 165
	5	58 78 85 135 147	70 93 102 136 195	27 101 121 139 167	53 96 132 173 243	41 107 137 156 197
monument	2	95 116	96 135	69 144	77 148	74 155
	3	81 101 177	82 118 154	48 113 161	52 109 159	43 94 151
	4	57 92 121 152	67 103 116 162	39 60 101 148	31 62 105 159	53 101 125 170
	5	53 93 122 150 206	49 95 108 134 202	44 76 133 178 187	31 60 80 129 178	30 58 78 123 184

Similar performance trends in terms of SSIM can be observed for higher threshold values of M=4 and 5. ALO performs the best with the highest average SSIM values of 0.728 and 0.781 respectively. However, for lower thresholds M=2 and 3, maximal SSIM values are closely contested between FA and SSO. The SSO has higher SSIM than FA for M=3 while for M=2, FA performs slightly better than SSO. From table 2, 3 and 4, we can also infer that the WOA is the least performing algorithm in terms of PSNR and SSIM values. ALO performs well for most test images, however, in some cases SSO or FA performs better. This is so because threshold values are generated randomly and each test image is taken as a different optimization problem. According to the No-Free lunch theorem [21], it is impossible to determine a single optimization algorithm suitable for numerous optimization algorithms because a segmented image is made by taking the average of gray level values grouped together under a class. Hence, the previously tested algorithms, FA and SSO were able to perform well in terms of PSNR and SSIM values for threshold numbers almost equal to the number of gray levels; while when the threshold number was increased greater than the number of gray levels, due to multimodality of histograms, FA and SSO performed poorly .For this case, the proposed ALO algorithm performed the best in terms of PSNR and SSIM values obtained after segmentation. The thresholded images obtained by all algorithms are given below:

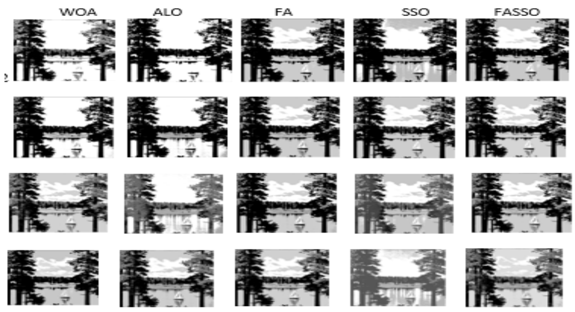

Fig. 11: Segmented images of "Sailboat" obtained by algorithms

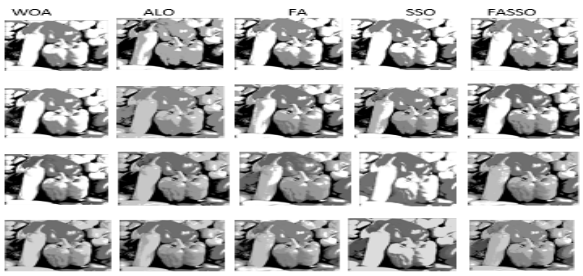

Fig. 12: Segmented images of "Pepper" obtained by algorithms

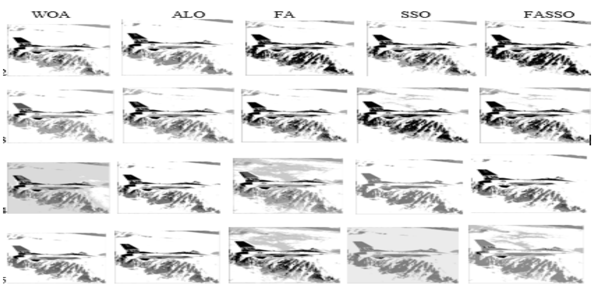

Fig. 13: Segmented images of "Airplane" obtained by algorithms

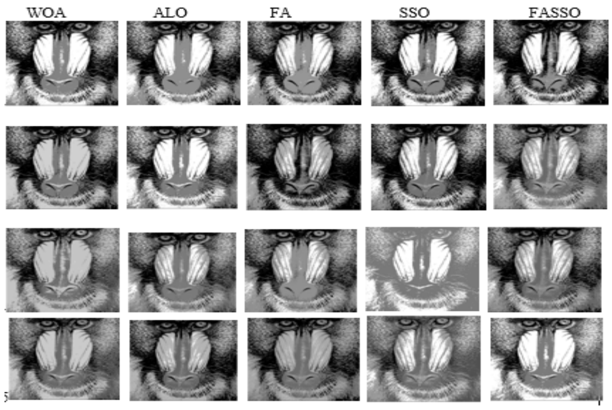

Fig. 14: Segmented images of "Baboon" obtained by algorithms

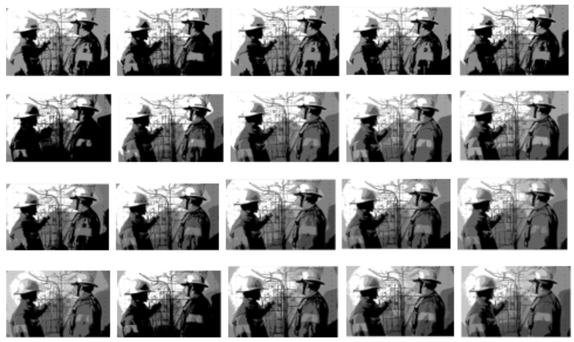

Fig. 15: Segmented images of "FireFig.hter" obtained by algorithms

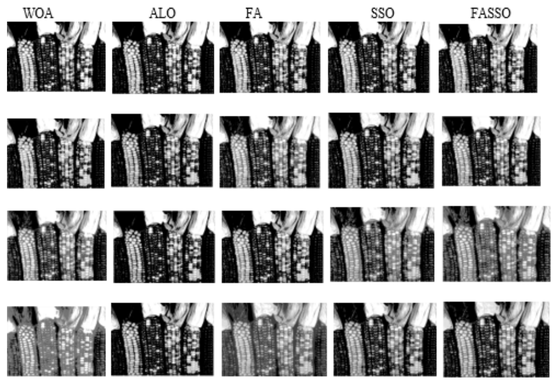

Fig. 16: Segmented images of "Corn" obtained by algorithms

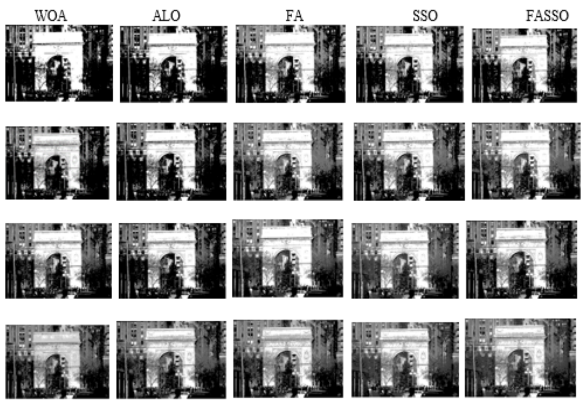

Fig. 17: Segmented images of "Monument" obtained by algorithms

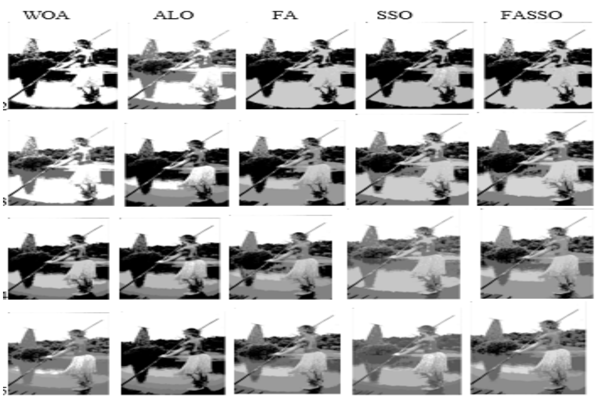

Fig. 18: Segmented images of "Sailboat" obtained by algorithms

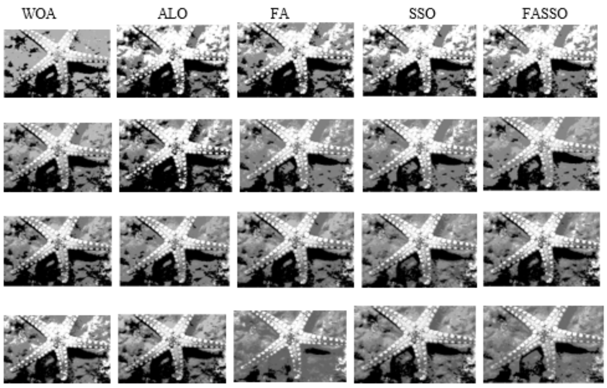

Fig. 19: Segmented images of "Starfish" obtained by algorithms

V. CONCLUSION

In this paper, optimal multi-thresholds for image segmentation are determined using the proposed meta-heuristic algorithms, Whale optimization (WOA) and Antlion optimization algorithm (ALO). The aim of the optimization problem is to maximize the Otsu's entropy. The results of which are compared to FA, SSO and FASSO algorithms using eight benchmark images. Every algorithm was evaluated, tested and compared based on three performance metrics, the fitness function, PSNR and SSIM. The results showed that ALO outperformed all other algorithms when maximizing the fitness function. Also, for higher levels of thresholding i.e. for 4 & 5 thresholds, PSNR and SSIM values are the highest for ALO, while for lower thresholding 2 & 3 , the existing algorithm, SSO performed the best. Generally speaking, we can conclude that the proposed application of ALO gives a better performance in terms of the fitness function, PSNR and SSIM for most optimization problems. Hence, ALO and SSO are high-performance algorithms for multi-thresholding image segmentation.

REFERENCES

[1] Bhandari, A. K., Kumar, A., & Singh, G. K. (2015). Modified artificial bee colony based computationally e□cient multilevel thresholding for satellite image segmentation using kapur's, otsu and tsallis functions. Expert Systems with Applications, 42(3), 1573–1601.

[2] Mohamed Abd El Aziz, Ahmed A. Ewees, Aboul Ella Hassanien, Mohammed Mudhsh and Shengwu Xiong .Multi-objective Whale Optimization Algorithm for Multilevel Thresholding Segmentation

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

[4] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In Neural Networks 1995. Proceedings. IEEE International Conference: Vol. 4(4) (pp. 1942–1948).

[5] Elsayed, S. M., Sarker, R. A., & Essam, D. L. (2014). A new genetic algorithm for solving optimization problems. Engineering Applications of Artificial Intelligence, 27, 57–69.

[6] Kaveh, A., & Talatahari, S. (2010). An improved ant colony optimization for con-strained engineering design problems. Engineering Computations, 27(1), 155–182.

[7] Cuevas, E., Cienfuegos, M., ZaldíVar, D., & PéRez-Cisneros, M. (2013). A swarm optimization algorithm inspired in the behaviour of the social-spider. Expert Systems with Applications, 40(16), 6374–6384.

[8] Yang, X.-S. (2009). Firefly algorithms for multimodal optimization. In International symposium on stochastic algorithms (pp. 169–178). Springer.

[9] Mehdi Neshat, Ghodrat Sepidname, A new hybrid optimization method inspired from swarm intelligence: fuzzy adaptive swallow swarm optimization algorithm (FASSO),Egyptian Informatics Journal, 16(3),2015,339-350

[10] Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowledge-Based Systems, 89, 228–249.

[11] Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. Advances in Engi-neering Software, 95, 51–67.

[12] Seyedali Mirjalili, The Antlion Optimizer,Advances in Engineering Software, 83,2015, 80-98.

[13] Kaveh, A., & Ghazaan, M. I. (2016). Enhanced whale optimization algorithm for siz-ing optimization of skeletal structures. Mechanics Based Design of Structures and Machines, 1–18.

[14] Cherukuri, S. K., & Rayapudi, S. R. (2016). A novel global MPP tracking of photo-voltaic system based on whale optimization algorithm. International Journal of Renewable Energy Development, 5(3).

[15] M. Sharawi, H. M. Zawbaa, E. Emary, H. M. Zawbaa and E. Emary, "Feature selection approach based on whale optimization algorithm," 2017 Ninth International Conference on Advanced Computational Intelligence (ICACI), Doha, 2017, pp. 163-168.

[16] Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In Computer vision, 2001. ICCV 2001. Proceedings. Eighth IEEE international conference on: Vol. 2 (pp. 416–423). IEEE.

[17] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality as-sessment: From error measurement to structural similarity. IEEE Transactions on Image Processing, 13(4), 600–612.

[18] Mlakar, U., Potocnik,ˇ B., & Brest, J. (2016). A hybrid differential evolution for optimal multilevel image thresholding. Expert Systems with Applications, 65, 221–232.

[19] Mohamed Abd El Aziz, Ahmed A. Ewees, Aboul Ella Hassanien,Whale Optimization Algorithm and Moth-Flame Optimization for multilevel thresholding image segmentation, Expert Systems with applications, 83,2017, 242-256.

[20] Goldbogen, J. A., Friedlaender, A. S., Calambokidis, J., McKenna, M. F., Simon, M., Nowacek, D. P. (2013). Integrative approaches to the study of baleen whale diving behavior, feeding performance, and foraging ecology. BioScience, 63(2), 90–100.

[21] Nowacek, D. P. (2013). Integrative approaches to the study of baleen whale diving behavior, feeding performance, and foraging ecology. BioScience, 63(2), 90–100.

[22] Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation, 1(1), 67–82.

[23] El Aziz M.A., Ewees A.A., Hassanien A.E., Mudhsh M., Xiong S. (2018) Multi-objective Whale Optimization Algorithm for Multilevel Thresholding Segmentation. In: Hassanien A., Oliva D. (eds) Advances in Soft Computing and Machine Learning in Image Processing. Studies in Computational Intelligence, vol 730. Springer, Cham.