Multilevel Threshold Image Segmentation of Brain Tumors Using Zebra Optimization Algorithm

Sarah Alhammad
Princess Nourah bint Abdulrahman University
Doaa Khafaga
Princess Nourah bint Abdulrahman University
Doaa Elshoura
Zagazig University
Khalid M. Hosny
k_hosny@yahoo.com
Zagazig University

Article

Keywords: Image segmentation, Brain tumor, Optimization, Kapur's entropy

Posted Date: February 29th, 2024

DOI: https://doi.org/10.21203/rs.3.rs-3941267/v1

Additional Declarations: No competing interests reported.

Multilevel Threshold Image Segmentation of Brain Tumors Using Zebra Optimization Algorithm

Sarah M. Alhammad[1], Doaa Sami Khafaga[1], Doaa El-Shoura[2], and Khalid M. Hosny[2*]

[1] Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O.Box 84428, Riyadh 11671, Saudi Arabia

[2] Department of Information Technology, Faculty of Computers and Informatics, Zagazig University, Zagazig 44519, Egypt.

Corresponding Author: (k_hosny@yahoo.com; k_hosny@zu.edu.eg)

Abstract: A Brain Tumor (BT), further known as an intracranial tumor, is a mass of abnormal tissue whose cells multiply and procreate uncontrolled and appear unaffected by those mechanisms that control normal cells, and it causes many people's deaths each year. BT is frequently detected using Magnetic Resonance Imaging (MRI) procedures. One of the greatest common techniques for segmenting medical images is Multilevel Thresholding (MT). MT received the researchers ' attention because of its simplicity, ease of use, and accuracy. Consequently, this paper uses the most recent Zebra Optimization Algorithm (ZOA) to deal with the MT problems of MRI images. The ZOA's performance has been evaluated on 10 MRI images with threshold levels up to 10 and evaluated against five different algorithms: Sine Cosine Algorithm (SCA), Arithmetic Optimization Algorithm (AOA), Flower Pollination Algorithm (FPA), Reptile Search Algorithm (RSA), and Marine Predators Algorithm (MPA). The experimental results, which included numerous performance metrics such as Mean Square Error (MSE), Peak Signal-To-Noise Ratio (PSNR), Feature Similarity Index Metric (FSIM), Normalized Correlation Coefficient (NCC), and fitness values, totally show that the ZOA outperforms all other algorithms based on Kapur's entropy for all the applied measures.

Keywords: Image segmentation; Brain tumor; Optimization; Kapur’s entropy

1. Introduction

BT, or brain cancer, is the development of unnatural cells in the human brain. The brain's structure is extremely complicated, and several regions control various nervous system activities. It may be cancerous or noncancerous. Malignant brain tumors can occur but are uncommon [1]. BT may arise in any part of the brain or skull, including its protective lining, the brain's backside (skull base), the brainstem, the sinuses, the nasal cavity, and several other areas [2]. MRI is a non-invasive medical imaging technique. Human organs may be seen in 2D and 3D with MRI. One of the most effective techniques is the MR imaging modality for classifying MRIs due to its precise images of brain tissues [3]. The most used technique for examining infected brain tissues is MRI.

Image segmentation is an essential step in image processing in various fields**,** including computer vision, agriculture, pattern recognition, medical imaging, robotic vision, and cryptography [4]. An image is segmented by extracting homogeneous sections with comparable texture, contrast, color, brightness, shape, and size using region-based, edge-based, feature-based clustering, and threshold-based segmentation techniques. The most common segmentation method is thresholding since it is straightforward, simple to use, quick, accurate, and requires little storage space [5]. There are two categories of threshold: bi-level and multilevel. A threshold value must be established within the first category to distinguish between an image's foreground and background. At the same time, the other class uses a histogram of pixel intensities to find more than two related locations in the image. [6]. The optimization issue of thresholding has been formulated using parametric or nonparametric methods [7]. With the parametric approach, the probability density function computes some parameters for each region to get the ideal threshold values. On the other hand, the nonparametric approach seeks to maximize a certain function., such as fuzzy entropy[8], Kapur's [7] (which maximizes class entropy), or Otsu function [9] (which maximizes between differences). Unfortunately, using these methods makes obtaining the best threshold values for ML exceedingly challenging and expensive computationally, especially as the threshold values increase.

Consequently, the demand for a powerful modern substitute was great. Because of the metaheuristic algorithms' notable performance in various domains, the researchers focus on these to solve the multilevel image segmentation issue [7]. Several research studies have used metaheuristic algorithms to address this issue, some of which will be covered in the following paragraphs.

Oliva et al. suggested an MT technique based on the harmony search optimization algorithm to segment digital images [9]. Houssein et al. [4]employed black widow optimization and the optimal threshold configuration using Otsu or Kapur's as an objective function to choose the ideal image threshold. Akay et al. [10] proposed a hybrid metaheuristic algorithm that combines levy flight with a teaching-learning-based artificial bee colony (TLABC) to show a new powerful variant, MTLABC, to calculate plant disease MT values. Five measures have been used to assess the effectiveness of this algorithm compared to other optimizers. The results of the testing confirmed its superiority to the others.

The authors in [11] give a self-adaptive method for adjusting the Dragonfly Optimization (DFO) parameters and suggest a technique using self-adaptive DFO (SADFO) for multilevel segmentation of digital images. The findings show that the suggested algorithm effectively searches the search space and finds the best threshold values. Khairuzzaman and Chaudhury [12] suggested a Grey Wolf Optimizer (GWO) technique for multilevel image thresholding. The threshold values are calculated using Otsu and Kapur's entropy methodology. Utilizing Kapur's entropy and Otsu thresholding, the GWO-based strategy outperforms bacterial foraging optimization (BFO) and particle swarm optimization (PSO) -based methods, according to the results of the experiments. Additionally, several additional metaheuristic algorithms have been effectively used for MT, including firefly algorithm (FA) [13], Whale Optimization Algorithm (WOA) [14], Cuckoo Search (CS) [15], Artificial Bee Colony (ABC) [16], The Wind Driven Optimization (WDO) [17], Krill Herd Optimization (KHO) [18], Moth Flame Optimization (MFO) [19], Marine Predators Algorithm (MPA) [20], Equilibrium Optimization Algorithm (EOA) [21], etc. Khalid AM et al. [22] presented a recently developed Coronavirus Disease Optimization Algorithm (COVIDOA). It outperformed traditional and modern competitors in terms of outcome effectiveness. In [23], a novel metaheuristic optimizer inspired by nature for image segmentation called RSA-SSA is based on RSA and the Salp Swarm Algorithm (SSA). Depended on COVIDOA combined with the Harris Hawks Optimization Algorithm (HHOA), a novel metaheuristic method for 2D and 3D medical grey image segmentation is presented in [24] to use both algorithms' benefits and avoid their weaknesses. In [25], COVIDOA is employed to address the issue of satellite image segmentation. Upadhyay and Chhabra proposed Kapur's entropy-based Crow Search Algorithm (CSA) to calculate optimal values of MT [26]. By combining the strengths of both techniques, Rather and Bala [27] suggested a gravitational search method and a constriction coefficient-based particle swarm optimization strategy to find the appropriate threshold values.

The important contributions of this study are outlined in the following:

- ZOA is demonstrated to handle multilevel thresholding segmentation.
- The Kapur's as an objective function was utilized to show a BT segmentation technique.

- The performance of the suggested approach is evaluated using various segmentation levels.
- ZOA is compared to several well-known metaheuristic methods.
- The MSE, PSNR, FSIM, NCC, and best fitness matrices are used to validate the efficacy of the segmentation technique.
- The suggested method may be improved to support more medical imaging diagnostics and be used for benchmark images.

The following research sections are organized: Section 2 discusses the MT segmentation. The proposed work is described in Section 3. Results and discussion are presented in Section 4. Conclusions and recommendations for future study are provided in Section 5.

2. Multilevel thresholding segmentation

Image thresholding applies a threshold value to the image's pixel intensity to convert a grayscale or color image into a binary image [28]. Pixels that fall below that threshold turn black, while those that rise beyond it turn white. Image thresholding is divided into bi-level and multilevel categories [29]. As mentioned below, Bi-level divides the image's pixels P into two areas (R1 and R2) utilizing a single threshold value (th):

$$
\begin{aligned}
&P \in R_1 \quad if \;\; 0 \leq P < th \\
&P \in R_2 \quad if \; th \leq P < L-1
\end{aligned}
\tag{1}
$$

Where L is the level of maximum intensity.

Contrarily, Multilevel uses several threshold values to split an image into multiple distinct regions, as shown next:

$$
\begin{aligned}
&P \in R_1 \qquad if \;\; 0 \leq P < th_1, \\
&P \in R_2 \qquad if \;\; th_1 \leq P < th_2, \\
&P \in R_{\,i} \qquad if \;\; th_i \leq P < th_{i+1}, \\
&P \in R_{\,k} \qquad if \;\; th_{k-1} \leq P < th_{L-1},
\end{aligned}
\tag{2}
$$

Where $\{th_1, th_2, ..., th_{k-1}\}$, indicates different threshold levels.

This paper uses a well-known MT technique to calculate the threshold values: Kapur’s entropy. This approach is briefly formulated in the subsection that comes next.

2.1 Kapur’s Entropy(Maximum Entropy Method)

Kapur's method for automatic thresholding is another unsupervised method that selects the ideal thresholds depending on the split classes' entropy [7]. Assuming that $\{th_1,\ th_2,\ \ldots\ th_n\}$ provides the group of thresholds that were applied to split the image into its several classes. Following that definition, the Kapur method's object function is:

$$H(th_1, th_2, \ldots, th_n) = H_0 + H_1 + \cdots + H_n \tag{3}$$

Where

$$H_0 = -\sum_{j=0}^{th_1-1} \frac{p_j}{w_0} \ln \frac{p_j}{w_0}, w_0 = \sum_{j=0}^{th_1-1} p_j \tag{4}$$

$$H_1 = -\sum_{j=th_1}^{th_2-1} \frac{p_j}{w_1} \ln \frac{p_j}{w_1}, w_1 = \sum_{j=th_1}^{th_2-1} p_j \tag{5}$$

$$H_n = -\sum_{j=th_n}^{L-1} \frac{p_j}{w_n} \ln \frac{p_j}{w_n}, w_n = \sum_{j=th_n}^{l-1} p_j \tag{6}$$

$H_0,\ H_1, \ldots, H_n$ Indicate to the entropies produced by each threshold value, w_0, w_1, w_n, refers to The proportion of pixels in each area to all of the image's pixels (the probability of each class).

The fitness function, as defined in equation (7), is maximized to get the best threshold values:

$$f_{Kapur}(th_1, th_2, \ldots, th_n) = \text{argmax}\ \{H\ (th_1, th_2, \ldots, th_n)\} \tag{7}$$

It is essential to remember that as the number of thresholds rises, the computing difficulty of the thresholding approach will exponentially climb. Kapur's entropy approach is ineffective for MT in such circumstances. So, it is suggested to enhance the accuracy and computational velocity of thresholding approaches by employing the ZOA based on Kapur's entropy. The final purpose of the suggested approach is to determine the best threshold values by maximizing the objective function represented by Equation (3).

3. Proposed segmentation approach

In 2022, the Zebra optimization algorithm suggested by TROJOVSKÁ et al. [30] is a new bio-inspired metaheuristic algorithm. Fundamentally, it was inspired by how zebras behave in the wild. Zebras' foraging behaviors are simulated, as well as their defense mechanism against assaults from predators.

A pioneer zebra allows other zebras to move to the forage in the foraging process. As a result, this pioneer zebra leads the herd's other zebras as they go over the plains [31]. The zebras' first line of defense against predators is to run away in a zigzag pattern. However, occasionally, they attempt to mislead or terrify the predator by assembling [32]. This section presents mathematical models of zebras' typical behaviors to mode ZOA.

1) Initialization

Using zebras to represent the population, the ZOA is a population-based optimizer. Every zebra is a potential solution to the issue and the place in which the zebras are in the search space for the issue. The position of every zebra in the search space calculates the decision variables' values. The zebras are distributed randomly in the search space at the beginning. Equation (8) contains a description of the ZOA population matrix.

$$X = \begin{bmatrix} X_1 \\ . \\ . \\ . \\ X_i \\ . \\ . \\ . \\ X_N \end{bmatrix} = \begin{bmatrix} X_{1,1} \ldots . X_{1,j} \ldots . X_{1,m} \\ . \quad . \quad . \\ . \quad . \quad . \\ . \quad . \quad . \\ X_{i,1} \ldots . X_{i,j} \ldots . X_{i,m} \\ . \quad . \quad . \\ . \quad . \quad . \\ . \quad . \quad . \\ X_{N,1} \ldots X_{N,j} \ldots X_{N,m} \end{bmatrix}_{M \times N} \tag{8}$$

The population of zebra is X; the i_{th} zebra is X_i; the j_{th} is the issue variable suggested by the ith zebra and represented by the value $X_{i,j}$. In the population, the number of zebras is N, and the number of decision variables is presented by m. Every zebra indicates a potential solution to the optimization issue. Consequently, the suggested values of every zebra for the issue variables may be used to assess the objective function. The fitness function's values are defined as a vector by Equation (9).

The zebra with the lowest value fitness function is the best potential solution for minimization issues. As an alternative, in maximization problems, the zebra with the greatest objective function value is the best potential solution. The best potential solution must also be determined by updating the objective function values due to the zebras' locations. As a result, Members of the ZOA population are updated in the two following stages in each iteration.

$$F=\begin{bmatrix} F_1 \\ \cdot \\ \cdot \\ \cdot \\ F_i \\ \cdot \\ \cdot \\ \cdot \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \cdot \\ \cdot \\ \cdot \\ F(X_i) \\ \cdot \\ \cdot \\ \cdot \\ F(X_N) \end{bmatrix}_{N\times 1} \tag{9}$$

Where F is fitness function values, and F_i, is the objective function value acquired for the i_{th} zebra.

2) Stage 1:Foraging behavior

The first stage involves updating population members depending on simulations of zebra behavior while foraging. Zebras typically eat sedges and grasses, but if their preferred foods are difficult to obtain, they may eat roots, buds, bark, leaves, and fruits. According to the quality and vegetation quantity, zebras might spend between 60 and 80 percent of their time eating [33]. One type of zebra, the plains zebra, is a forerunner among grazers. Consuming the canopy of higher grass with fewer nutrients creates circumstances for other species requiring shorter, below are healthier grasses. [31]. In ZOA, the pioneer zebra is the best population member, leading other population members to its location in the search space. Consequently, utilizing Equations (10) and (11) by updating the zebra's position in the foraging stage.

$$X_{i,j}^{new,p1} = X_{i,j} + r \cdot (PZ_j - I \cdot X_{i,j}) \tag{10}$$

$$X_i = \begin{cases} X_{i,j}^{new,p1}, & F_{i,j}^{new,p1} < F_{i,}, \\ X_{i,} & else \end{cases} \tag{11}$$

Where $X_i^{new,p1}$ is the new status of the i_{th} zebra depended on the first stage, $X_{i,j}^{new,p1}$ is its j_{th} dimension value, $F_{i,j}^{new,p1}$ it is a fitness function value. PZ refers to the pioneer zebra, and it is the ideal member. PZ_j is its j_{th} dimension. r is a random number in the range [0; 1]. I= round (1+rand), rand is a random number in the interval [0, 1]. As a result, if $I \in \{1,2\}$ and parameter I=2, there will be much further alterations in population mobility.

3) Stage 2: Defense strategies against predators

The second stage updates the location of ZOA population members in the search space using the zebra's defense models against predator assaults. Zebras are endangered by leopards, cheetahs, wild dogs, spotted hyenas, brown hyenas, and lions, their principal predators. When smaller predators like hyenas and dogs attack by banding together, they confuse and terrify the hunter. Therefore, zebras are more aggressive [32]. Zebras that approach water are also preyed upon by crocodiles [34]. Depending on the predator, zebras employ various strategies for defense. The zebra's defense against lion assaults is to flee in a zigzag form and make haphazard sideways turns [35]. In the ZOA design, it is believed that one of the two criteria below takes place with an equal probability:

a) When the lion attacks the zebra, it decides on an escape strategy. This method may be modeled utilizing the mode of S_1in the Equation (12).
b) When other predators attack zebras, they will choose the offensive strategy. This method may be modeled utilizing the mode S_2 in the equation (12).

Zebras' positions are updated, and the new location is approved for a zebra with a higher objective function value. This updating situation is modeled utilizing the Equation (13).

$$X_{i,j}^{new,p2} = \begin{cases} S_1 : x_{i,j} + R.(2r-1).\left(1-\frac{t}{T}\right).x_{i,j} & P_s \le 0.5; \\ S_2 : x_{i,j} + r.\left(AZ_j - I.x_{i,j}\right) & else \end{cases} \quad (12)$$

$$X_i = \begin{cases} X_{i,j}^{new,p2}, \quad F_{i,j}^{new,p2} < F_{i,} , \\ X_{i,} \quad\quad else \end{cases} \quad (13)$$

where $X_i^{new,p2}$ is the new status of the i_{th} zebra depended on the second stage, $X_{i,j}^{new,p2}$ is its j_{th} dimension value, $F_{i,j}^{new,p2}$ it is the value of the objective function, the iteration contour is t, The maximum number of iterations is T, while R is a fixed integer equal to 0.01,P_s, is the probability of selecting one of two randomly generated methods in the range [0; 1], The condition of the zebra that was attacked is AZ , and AZ_j is its j_{th} dimension.

We think this is the initial use of the ZOA for image segmentation in grayscale BT images. The flow chart of ZOA with Kapur's fitness function for MT segmentation of BT images is seen in Figure 1.

4. Results and discussion

An overview of the test datasets is presented at the beginning of this section. The parameter values for the suggested and other algorithms are then shown, and the evaluation measures employed to compare the outcomes are displayed. The results of evaluating the suggested method and its rivals are then displayed numerically. We completed a comparison of the results gathered in the end.

4.1 Dataset

The database used in this paper is obtained from the web (https://www.kaggle.com/). It is a famous, authentic, machine-learning engineer choice website. There are 253 images in all, with different sizes. They are in JPG format. Based on the existence of tumors, the dataset is divided into two classes: YES and NO. This selected dataset consists of 98 normal (NOT having Tumors) and 155 abnormal (Having multiple kinds of Tumors) MRI Images. From the dataset, figure 2 shows the sample images.

Our research includes segmenting ten grayscale images for BTs using 2, 4, 6, 8, and 10 threshold levels. These images were randomly chosen from the Kaggle dataset to validate the ZOA performance. Table 1 shows the original images and their histograms. It is significant to note that the images are given new names, such as img1, img2, img3, img4, and so on.

4.2 Parameter setting

The outcomes of the proposed algorithm's MT segmentation are assessed utilizing different criteria and compared to those of five well-known metaheuristic algorithms. These approaches are SCA [36], AOA [37], FPA [38], RSA [39], and MPA [20].

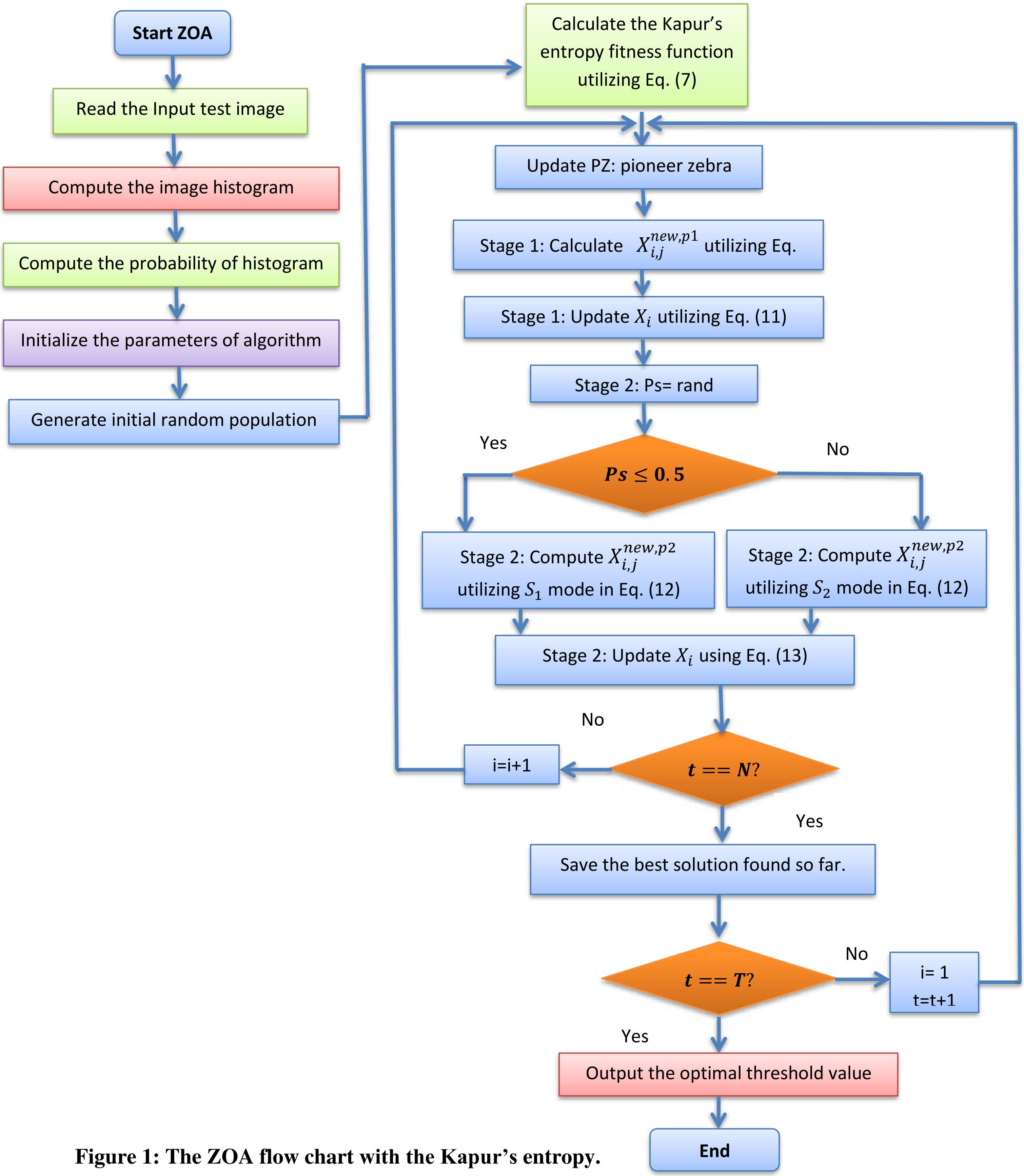


Figure 1: The ZOA flow chart with the Kapur's entropy.

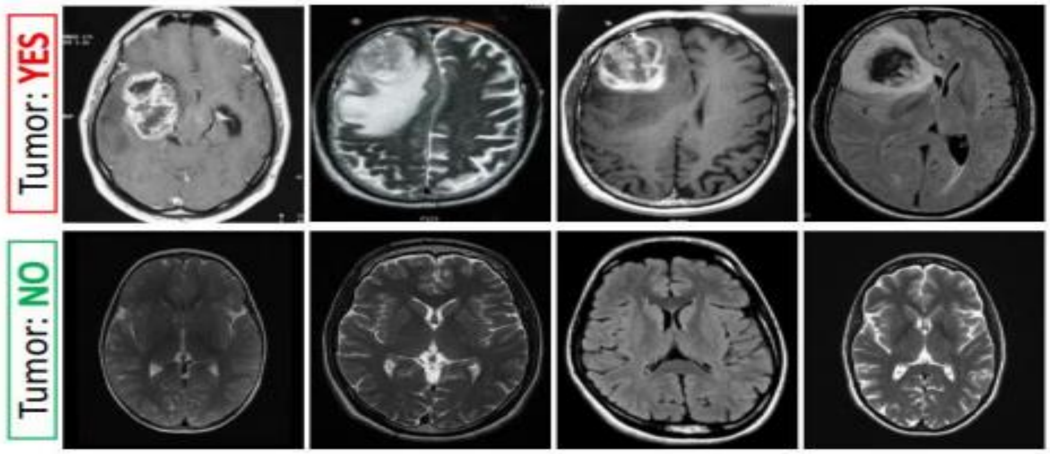


Figure 2: Different types of MRI image views

These algorithms were chosen for comparison for the following clarifications:

- They have demonstrated that they are better than others in solving optimization issues, particularly image segmentation.
- The majority of them have recently been published in reputable sources.
- On the MATLAB website (https://matlab.mathworks.com/), you may get their MATLAB implementations for free.

A laptop running Windows 10 Ultimate 64-bit and equipped with an Intel(R) Core(TM) i7-1065G7 CPU was used for all experiments. The MATLAB R2016b development environment was used to create each algorithm.

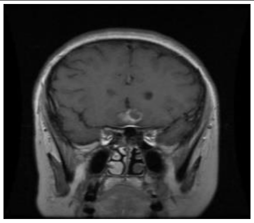
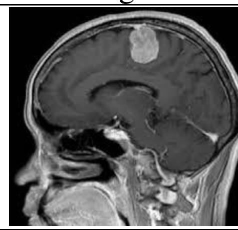
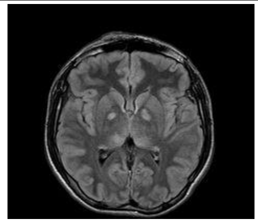
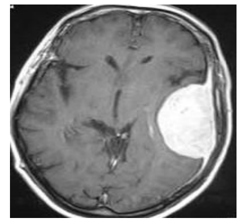
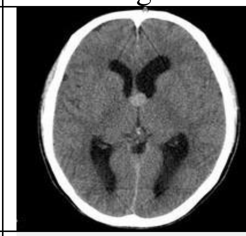
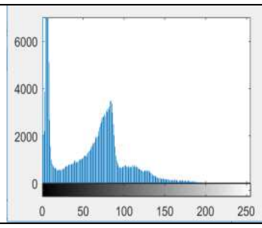
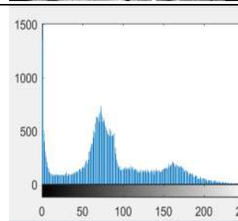
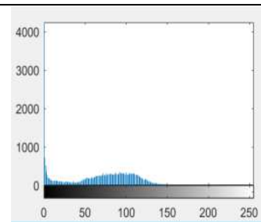
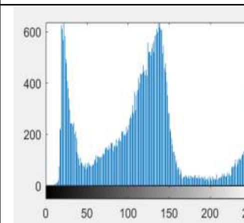
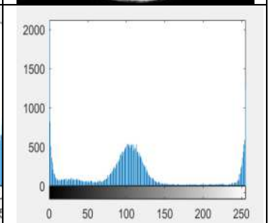
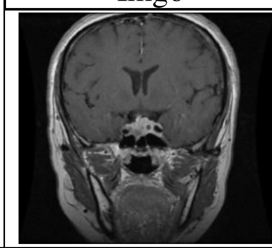
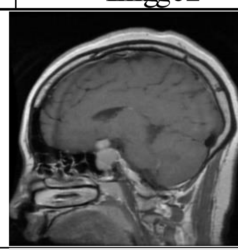
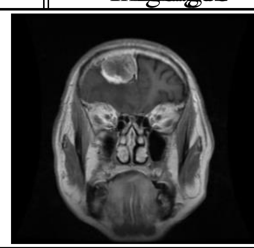
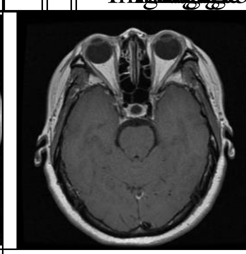
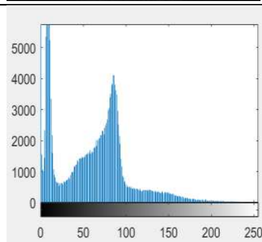
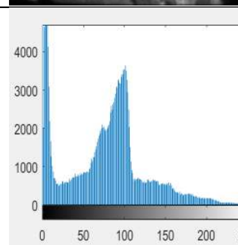
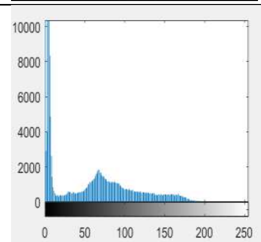
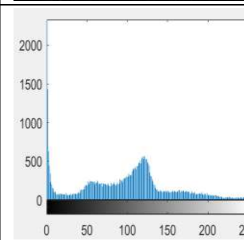
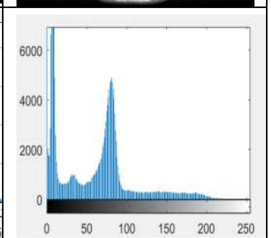
As mentioned above, each algorithm is evaluated using 30 independent runs with a population size of 100, where the maximum number of iterations is 200 for each test image of BT. The simulation environment is the same for all algorithms.

4.3 performance evaluation criteria

Various performance measures assess the suggested algorithm's performance involving MSE, PSNR, FSIM, NCC, and best fitness.

The segmented images' quality is assessed using MSE, PSNR, FSIM, and NCC. On the other hand, the best fitness is used to demonstrate that the suggested algorithm can find the best solutions.

Table 1: The histograms for the test images

<table>
<tr><td rowspan="2">Image</td><td>Img1</td><td>Img2</td><td>Img3</td><td>Img4</td><td>Img5</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Histogram</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td></tr>
<tr><td rowspan="2">Image</td><td>Img6</td><td>Img7</td><td>Img8</td><td>Img9</td><td>Img10</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Histogram</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td></tr>
</table>

4.3.1 Best Fitness

Using Kapur's entropy, the suggested and state-of-the-art algorithms' maximum fitness is estimated and tested using equation (7). The average fitness values within 30 runs for each threshold level for each image are calculated, and the results are recorded in Table 2 and graphically shown in Figure 3. This table indicates that the proposed algorithm outperforms the other alternatives, which is confirmed based on the results in Figure 3. With a value of 28.8729, the proposed approach outperforms the alternatives by at least 0.1324, as seen in Figure 3.

4.3.2 Mean square error (MSE)

MSE is usually utilized to estimate the variance between the segmented and original images. The formula used to calculate it is as follows:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} [F(i,j) - f(i,j)]^2 \qquad (8)$$

Here, $F(i,j)$, $f(i,j)$ are the original and segmented images, the intensity level within the row, i^{th} and columnj^{th}, respectively. The image's row and column numbers are M and N, respectively.

4.3.3 Peak Signal-to-Noise Ratio (PSNR)

In this subsection, another metric named PSNR is commonly used to quantify image quality. It is the ratio of the square of the maximum grey level 255^2. The following formula is used to calculate the MSE between the original and the separated one:

$$PSNR = 10 \left(\frac{255^2}{MSE}\right) \qquad (9)$$

The equation described above is used to calculate MSE. For better quality, the PSNR must be raised.

4.3.4 Feature Similarity Index (FSIM)

FSIM is used to estimate the structural similarity of the two images based on what follows:

$$FSIM(F,f) = \frac{\sum_{x \in \Omega} S_L\ (X).PC_m(x)}{\sum_{x \in \Omega} PC_m(x)} \qquad (10)$$

Where S_L (X) refers to the resemblance between the two images, PC is the phase congruence, and Ω relates to the image's spatial domain. The maximum possible value of the FSIM, representing total similarity, is 1. The higher FSIM value enhances the thresholding process's performance [40].

4.3.5 Normalized Correlation Coefficient (NCC)

NCC is a metric for assessing how closely two images are related. The absolute value of NCC ranges from 0 to 1. One indicates the strongest potential association between the two images, whereas zero indicates no relationship. The higher the absolute value of NCC, the stronger the relationship between the two images. NCC between the original image $F(i,j)$and segmented images $f(i,j)$ is estimated in accordance as follows:

$$NCC = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [F(i,j) \times f(i,j)]}{\sqrt{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [F(i,j) \times f(i,j)] \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [F(i,j) \times f(i,j)]}} \qquad (11)$$

4.4 Experimental results

The numerical results of testing the ZOA to determine the best threshold values using Kapur's fitness function are highlighted in this subsection. These results are assessed against the state-of-the-art SCA, AOA, FPA, RSA, and MPA algorithms. The tests used 2, 4, 6, 8, and 10 thresholds. The ZOA algorithm with Kapur's entropy was run, and the outcomes were compared with the other algorithms. Table 3 shows ZOA segmented images for all BT images used in the tests. Tables 4-7 show the results of the appropriate MSE, PSNR, FSIM, and NCC evaluation matrices. These tables have bolded the greatest thresholding approach values, which yield the best outcomes. They display the best quality segmentation. The average values of MSE, PSNR, FSIM, and NCC evaluation matrices were displayed in Figures 4, 5, 6 & 7.

Table 2: Fitness values of each algorithm under Kapur's entropy.

Test image	nTh	SCA	AOA	FPA	RSA	MPA	ZOA
Img1	2	12.0807	**12.0907**	**12.0907**	11.9809	**12.0907**	**12.0907**
	4	21.8396	21.6925	**21.8801**	21.6050	**21.8801**	**21.8801**
	6	29.6614	29.5510	29.7238	29.8628	29.7322	**29.9680**
	8	36.7660	**38.8317**	37.1337	37.0877	37.4446	37.4166
	10	43.9659	**45.2420**	44.1519	43.5919	44.3185	44.4672
Img2	2	**12.4260**	12.4225	**12.4260**	12.4087	**12.4260**	**12.4260**
	4	22.1317	22.1274	**22.1663**	21.9652	**22.1663**	**22.1663**
	6	29.7890	29.1274	29.9539	29.6526	**30.0276**	29.9193
	8	35.9424	**37.6257**	36.9308	37.0007	36.9131	37.3305
	10	43.2628	43.3453	41.3679	43.8728	43.2315	**43.8913**
Img3	2	**10.2221**	10.204	**10.2221**	10.0529	**10.2221**	**10.2221**
	4	18.8586	18.8795	**18.9166**	18.7887	18.8142	**18.9166**
	6	26.6899	26.5984	26.8431	26.5760	26.1241	**26.8720**
	8	33.4857	**34.2795**	34.2568	34.0277	34.0389	**34.2795**
	10	39.8159	**40.5444**	40.2314	40.1098	40.1234	40.4451
Img4	2	**13.1024**	13.0938	**13.1024**	13.0416	**13.1024**	**13.1024**
	4	22.6776	22.6455	22.7131	21.9397	**22.7138**	**22.7138**
	6	30.7881	30.0182	30.9241	30.7327	30.9241	**30.9597**
	8	37.3841	**38.7449**	38.4387	38.2305	38.2297	38.5191
	10	43.5987	44.8538	44.0285	44.9987	44.8948	**45.0726**
Img5	2	11.9675	11.9642	**11.9682**	11.7959	11.9642	**11.9682**
	4	20.4993	20.0093	**20.5740**	20.3509	20.5711	**20.5740**
	6	28.0808	**28.7821**	28.2837	28.2916	28.3557	28.3557
	8	35.1713	34.5437	35.8234	35.6227	35.6165	**35.8947**
	10	41.6987	41.3942	42.4819	42.6144	42.2798	**42.7074**
Img6	2	**12.7062**	**12.7062**	**12.7062**	12.6821	**12.7062**	**12.7062**
	4	22.4811	22.4310	**22.4863**	21.9261	**22.4863**	**22.4863**
	6	30.4813	**30.7530**	30.5756	29.3541	30.5961	30.6161

	8	36.7548	37.4939	37.7926	37.9748	37.9233	**38.0177**
	10	43.5233	44.6108	44.6684	44.8716	**45.0171**	45.0112
Img7	**2**	**13.0047**	**13.0047**	13.0032	12.9626	**13.0047**	**13.0047**
	4	22.6331	22.5175	**22.7020**	22.1212	**22.7020**	**22.7020**
	6	30.5778	30.2289	30.8634	30.1794	**30.8931**	30.8807
	8	37.6307	38.2416	**38.4186**	37.8019	38.3925	38.3864
	10	43.6754	44.3671	44.0749	44.3021	44.6499	**45.4110**
Img8	**2**	**11.4076**	11.3966	**11.4076**	11.3994	**11.4076**	**11.4076**
	4	20.0455	20.7358	**20.7831**	20.5834	20.5216	**20.7831**
	6	29.0164	28.6556	28.2748	28.3560	28.9591	**29.3003**
	8	36.3548	**37.1143**	36.7356	36.6530	36.8674	36.8226
	10	42.1779	43.7029	43.4951	41.2204	43.5305	**43.7227**
Img9	**2**	**12.3072**	**12.3072**	**12.3072**	12.1296	**12.3072**	**12.3072**
	4	21.2804	21.2421	**21.4721**	21.1199	**21.4721**	21.3282
	6	28.6406	29.1634	29.3303	28.8561	29.3431	**29.3618**
	8	35.4072	36.2774	36.7168	36.2721	36.7435	**36.8779**
	10	42.9643	**44.2927**	43.5428	43.2321	43.7835	43.9739
Img10	**2**	**12.5139**	**12.5139**	**12.5139**	12.4615	12.3641	**12.5139**
	4	21.8791	21.6838	**21.9119**	21.3207	21.8733	**21.9119**
	6	29.6436	**30.0951**	29.8369	29.5629	29.7813	29.9657
	8	36.2431	36.8848	37.4300	37.4752	37.5030	**37.5447**
	10	42.5771	43.5028	43.2005	43.8804	44.0688	**44.4436**

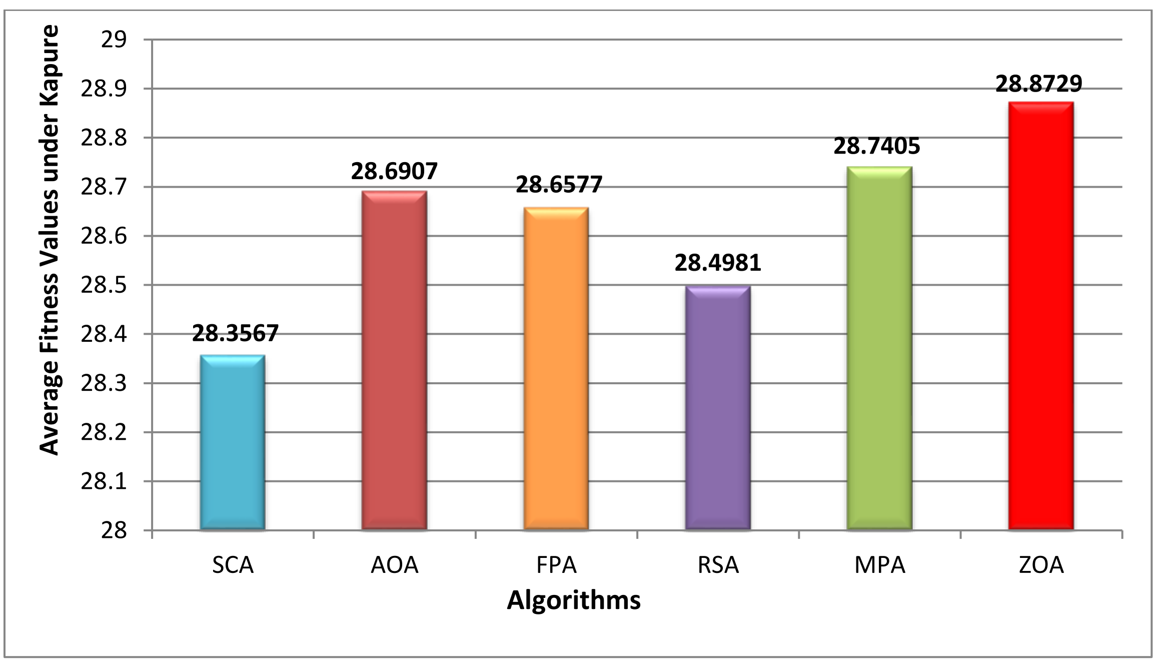
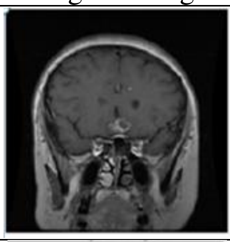
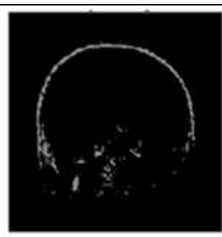
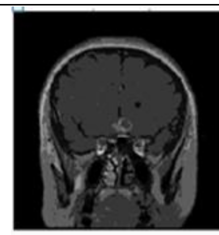
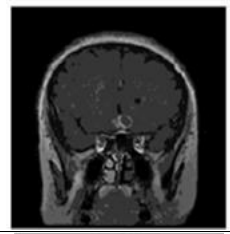
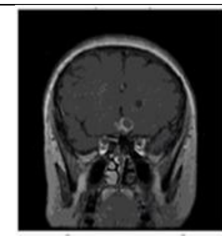
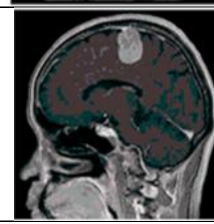
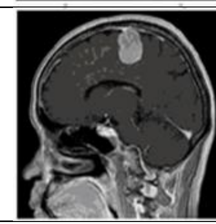
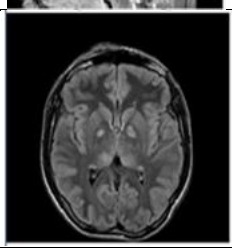
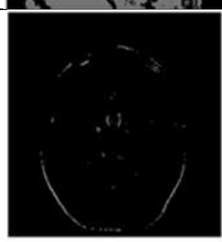
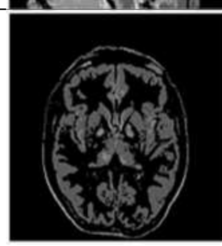
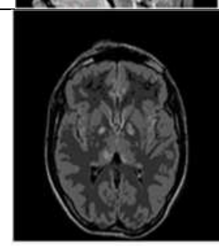
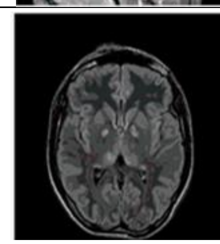
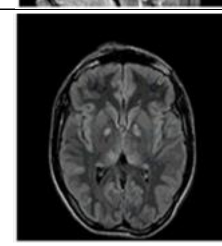
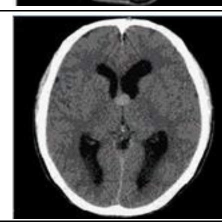
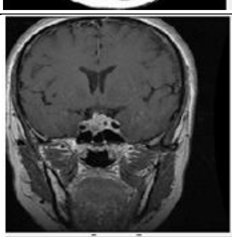
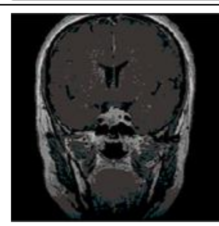
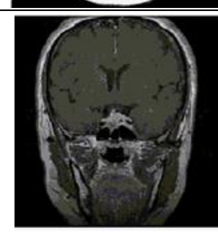
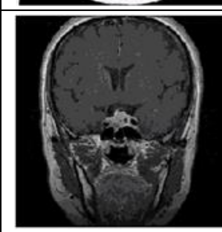


Figure 3: Average fitness values under Kapur's entropy.

Higher mean values for PSNR, FSIM, and NCC indicate a more accurate and efficient method. At the same time, the lowest mean value reflects the ideal MSE value. The values for the MSE metric are listed in Table 4. The lowest value corresponds to the best MSE result. Notably, ZOA outperforms all competing algorithms (as previously indicated). The SCA has higher MSE values with all threshold levels in img3, 4, 5, and 8. The AOA in img3, 5, and 10 the MPA in img2, 3, and 4. Table 5 shows the values of PSNR for each algorithm. An increase in the value of PSNR indicates better segmentation quality. The ZOA generally outperforms all other algorithms.

Table 3: ZOA-segmented images were collected using Kapur's entropy at Th = 2, 4, 6, 8, and 10.

	Original Image	Th2	Th4	Th6	Th8	Th10
Img1						
Img2						
Img3						
Img4						
Img5						

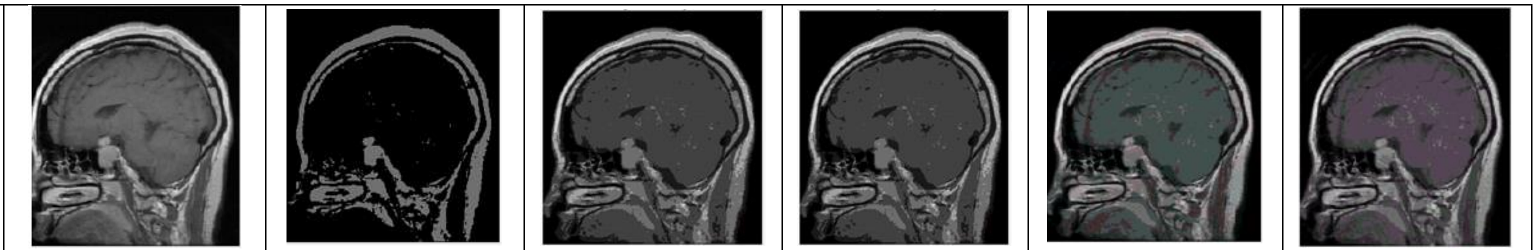
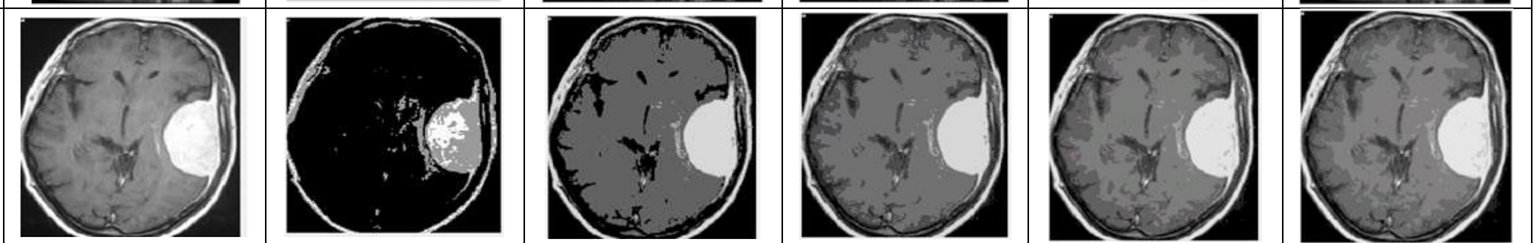
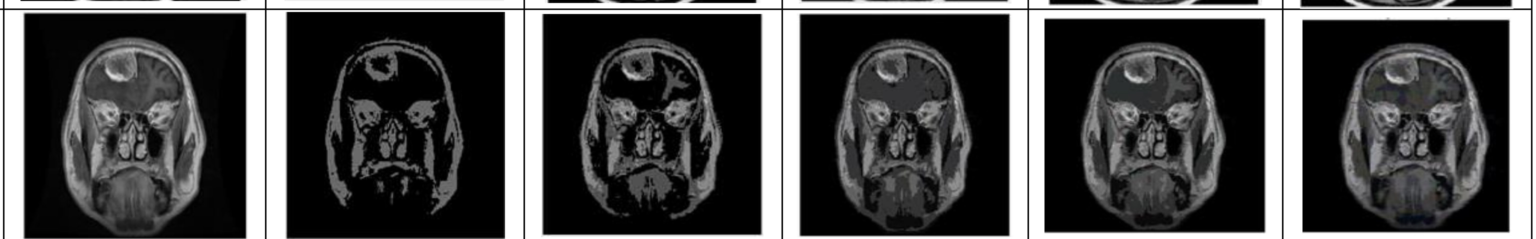
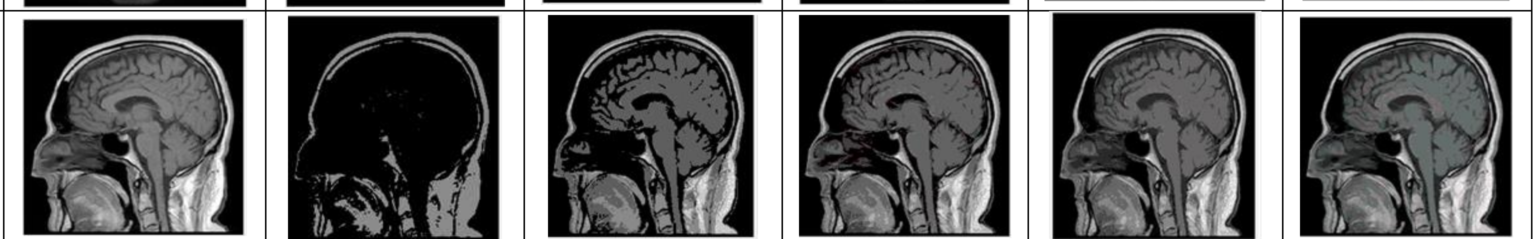
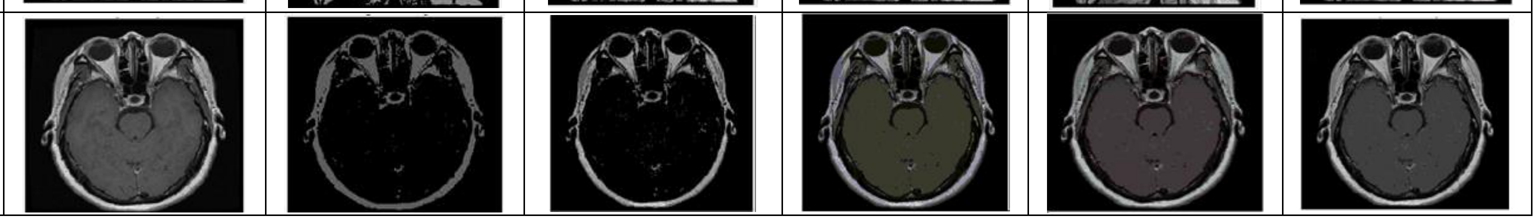
Img6	
Img7	
Img8	
Img9	
Img10	

Table 6 presents the FSIM measure's values. This statistic investigates and evaluates how well an image's characteristics remain after processing. Compared to other algorithms, the ZOA outperforms FSIM across various test images. The SCA and AOA have lower values with all threshold levels in img2, the RSA in img3, and the MPA in img5. The FPA provides superior outcomes in img1 (level 2 and 8), img2 (level 4), and img3, 4, 6, 7, 9 (level 2).

Table 7 shows the NCC measure's values. The results of the suggested algorithm (ZOA) surpass other comparable algorithms. The SCA offers higher value at only one level in img1 (level 4) and img2 (level 8), and the MPA in img1 and 2 (level 6).

Table 4: Comparison of the ZOA and the other chosen algorithms based on MSE values.

Test image	nTh	SCA	AOA	FPA	RSA	MPA	ZOA
Img1	2	158.9184	158.9730	**158.8184**	**158.8184**	**158.8184**	**158.8184**
	4	150.6175	151.1256	151.1357	**150.2485**	**150.2485**	**150.2485**
	6	128.2198	126.7281	130.7769	124.9438	121.8598	**121.5801**
	8	103.2709	**100.3865**	122.3274	112.3153	104.8607	115.2156
	10	**93.9872**	95.5439	104.1153	100.2496	101.0415	99.8731
Img2	2	**185.8820**	185.8843	**185.8820**	**185.8820**	185.8920	**185.8820**
	4	**172.8410**	173.7744	173.0474	173.8279	173.8279	**172.8410**
	6	144.8680	**140.6737**	142.8834	141.5046	140.8789	140.7902
	8	123.4662	**118.2418**	123.3949	120.4741	120.8153	120.6064
	10	125.6806	111.7685	122.5241	**109.6258**	112.2712	112.7342
Img3	2	115.0809	114.1015	112.3535	102.0809	105.0809	**101.3603**
	4	98.1351	96.3442	97.0728	97.3713	97.4273	**96.2571**
	6	85.2379	83.8884	86.3520	**83.6428**	84.9011	87.8579
	8	77.6914	78.2026	75.4648	**72.8540**	75.0055	75.2209
	10	71.1113	69.3038	72.3845	69.3338	70.1814	**69.0123**
Img4	2	240.7926	241.3617	**239.2642**	240.7926	240.7926	**239.2642**
	4	227.8380	**215.1193**	225.7677	222.3153	232.5773	232.5258
	6	200.0589	196.2060	209.6711	206.6069	201.7654	**198.0113**
	8	184.4940	183.1071	185.5429	181.8710	182.0644	**180.7149**
	10	156.9026	151.3074	**147.9278**	155.6690	151.5179	154.3534
Img5	2	149.4571	151.2278	151.8433	149.3324	**149.2775**	**149.2775**
	4	130.8769	131.4485	132.9525	131.0438	132.4367	**130.0146**
	6	116.6040	114.6245	121.7776	118.3011	115.6622	**112.5541**
	8	105.4833	104.5415	103.7967	100.6640	99.8435	**99.0314**
	10	89.1913	99.5615	109.3469	**87.4443**	90.8170	90.4431
Img6	2	**182.0543**	182.7423	**182.0543**	**182.0543**	**182.0543**	**182.0543**
	4	177.6079	**175.4338**	178.6294	177.4306	175.4396	**175.4338**
	6	**150.3105**	157.1096	156.8394	156.9755	152.2338	151.9911
	8	143.1729	**141.4241**	145.1713	145.8810	143.0406	142.6774
	10	127.1759	128.4959	123.9883	128.5125	123.5053	**122.6050**
Img7	2	**189.8107**	190.3056	**189.8107**	189.9107	**189.8107**	**189.8107**
	4	182.7078	181.9576	181.7776	182.5508	183.5508	**181.5169**
	6	**158.1542**	161.3669	159.5180	159.3732	158.7823	159.0487
	8	131.6746	**129.2275**	130.8862	129.6114	129.5844	129.4773
	10	107.9638	105.4667	104.8557	105.5416	106.1923	**101.0960**
Img8	2	116.5343	**116.3623**	116.5343	116.5343	116.5343	**116.3623**
	4	106.1678	**101.0733**	107.3096	106.1679	106.4024	**101.0733**
	6	92.8709	**88.0760**	98.6512	93.6621	93.0794	93.4030
	8	83.5777	77.4737	91.5508	80.5134	80.1301	**72.8663**
	10	77.4180	72.6584	75.9731	**72.0232**	79.2080	**72.0232**
Img9	2	**166.3062**	166.5052	**166.3062**	**166.3062**	166.4062	**166.3062**
	4	156.9205	**145.9900**	152.4625	150.6013	154.6013	**145.9900**
	6	132.5298	132.7664	131.2131	131.7976	**130.8455**	**130.8455**
	8	116.8653	**103.2348**	119.4200	110.8698	106.6334	106.6334
	10	101.1996	**95.4203**	104.7913	106.8391	101.3695	96.6834

Img10	2	**161.8955**	162.5941	**161.8955**	**161.8955**	**161.8955**	**161.8955**
	4	156.5049	156.7960	157.6506	157.6582	156.9000	**155.6506**
	6	125.9771	128.0638	129.6961	**125.2697**	126.0015	125.3477
	8	115.5229	108.9977	119.6643	108.3111	110.7810	**106.7972**
	10	105.3970	103.5830	106.1347	99.6079	**98.6649**	107.2852

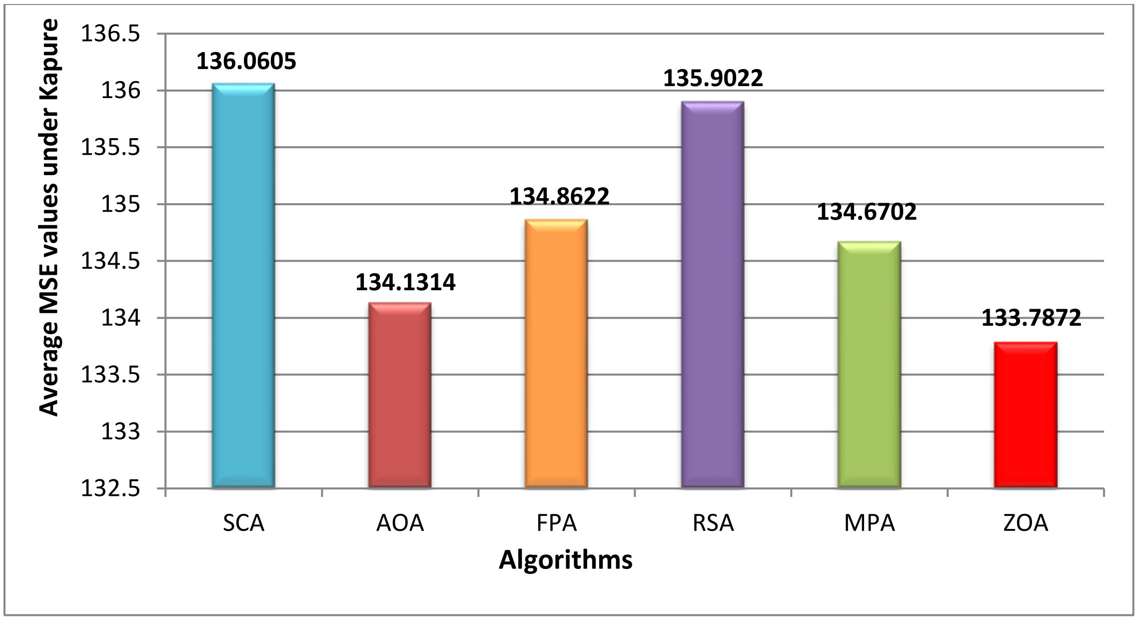


Figure 4: Average MSE values under Kapur's entropy.

Table 5: Comparison of the ZOA and the other chosen algorithms based on PSNR values.

Test image	nTh	SCA	AOA	FPA	RSA	MPA	ZOA
Img1	2	12.7471	12.7471	12.7471	12.6848	12.7471	**12.8183**
	4	15.0176	14.8114	**15.0192**	14.7081	**15.0192**	**15.0192**
	6	22.5411	22.6504	22.5936	22.5551	22.0833	**22.6557**
	8	23.7300	24.7821	24.9547	23.4878	**25.3969**	25.2366
	10	26.6923	26.6978	26.3374	26.0816	26.1906	**26.9054**
Img2	2	**12.6767**	**12.6767**	**12.6767**	12.6123	**12.6767**	**12.6767**
	4	14.2883	14.3071	**14.4601**	13.2545	**14.4601**	14.2313
	6	16.2999	16.6810	16.9298	16.2513	16.8182	**16.9178**
	8	22.5249	23.0783	22.4650	22.1342	22.9744	**23.1394**
	10	23.8556	24.1439	24.2633	23.9517	25.2077	**25.8249**
Img3	2	12.7527	12.7961	12.8527	12.4403	12.5527	**12.9240**
	4	17.9551	18.3204	18.7061	18.1205	**18.9130**	18.4773
	6	21.0305	22.2626	22.5094	21.3402	24.0256	**24.2776**
	8	26.0279	26.0848	**26.9187**	26.8023	26.8631	26.7131
	10	27.8435	27.8708	29.1585	28.3610	29.5500	**29.8941**
Img4	2	8.1774	8.1938	8.1774	8.1814	8.1774	**8.9486**
	4	15.5236	15.2448	**15.7510**	15.4380	15.7342	15.5054
	6	20.8448	20.9670	20.2456	20.1942	20.4212	**21.0877**

	8	21.0956	21.6204	**22.8059**	21.0683	21.1634	22.3447
	10	23.2095	23.3921	23.7549	22.9426	23.4765	**23.9048**
Img5	**2**	10.9634	**10.9804**	10.9764	10.8829	10.9387	10.7476
	4	14.1724	14.2774	14.4525	13.9324	14.2218	**14.9138**
	6	21.1262	21.8209	21.7229	21.0335	21.5056	**21.1856**
	8	24.2878	25.5083	25.3659	**25.8079**	25.3043	25.3554
	10	25.9979	26.4837	26.1989	26.5837	25.9813	**26.8997**
Img6	**2**	**13.4176**	**13.4176**	**13.4176**	13.2284	**13.4176**	**13.4176**
	4	13.8482	13.7285	**13.8828**	13.1702	**13.8828**	**13.8828**
	6	20.1718	20.6392	**20.6893**	20.2348	20.7542	20.5237
	8	23.8570	23.7860	23.8592	23.6289	23.7054	**24.0529**
	10	25.9809	26.2272	26.0466	26.4161	25.4226	**26.5368**
Img7	**2**	11.6278	**11.9278**	**11.9278**	11.4647	11.6990	**11.9278**
	4	14.3567	14.4046	**14.4051**	13.8897	**14.4051**	14.2919
	6	19.8022	19.8429	19.8426	19.9944	19.9736	**20.7633**
	8	22.1720	23.2138	24.0183	24.0307	22.9915	**24.0836**
	10	26.0396	**26.7837**	26.2050	26.4197	26.2594	26.1105
Img8	**2**	15.5618	15.5718	**15.5818**	**15.5818**	15.5718	**15.5818**
	4	18.2220	18.7753	18.3995	19.1268	18.3665	**19.1487**
	6	**23.8465**	23.8195	23.7007	23.8142	22.6753	23.5534
	8	25.2627	25.0474	25.8032	26.4506	25.9070	**26.6684**
	10	27.7807	27.7929	**27.7934**	27.3962	27.0745	27.6497
Img9	**2**	**10.8764**	**10.8764**	**10.8764**	10.6993	10.7764	**10.8764**
	4	14.4877	15.2963	15.4840	14.7240	14.4840	**15.8081**
	6	20.6718	20.2588	21.3626	19.8411	21.0694	**21.9869**
	8	23.0694	24.0250	24.4688	24.0619	**24.6194**	24.1193
	10	24.9186	25.7913	25.5257	26.4721	25.3593	**26.9659**
Img10	**2**	**13.8809**	**13.8809**	**13.8809**	13.8264	**13.8809**	13.6521
	4	14.6914	14.4964	14.5890	**15.3831**	14.5779	14.3603
	6	22.9921	23.6463	23.8233	24.0942	23.2075	**24.9205**
	8	25.0440	26.7488	24.8468	24.3925	24.9766	**26.6923**
	10	26.4371	**27.1063**	26.9600	26.7359	26.3264	27.0527
		19.60797	19.91008	19.98868	19.71857	19.87574	20.26464

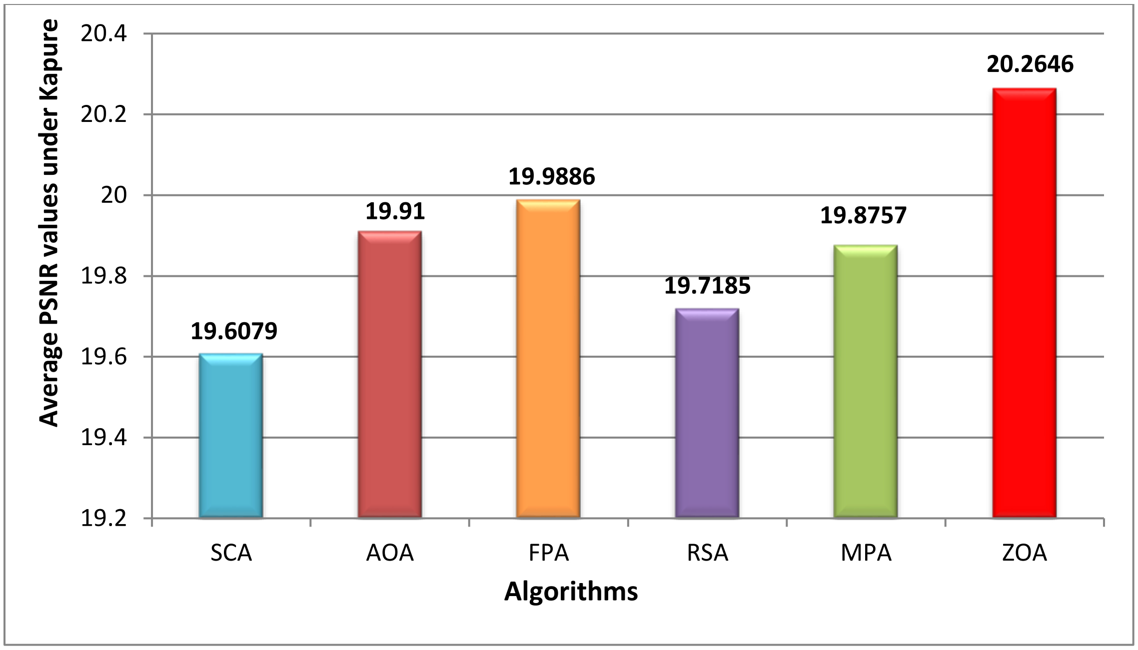


Figure 5: Average PSNR values under Kapur's entropy.

Table 6: Comparison of the ZOA and the other chosen algorithms based on FSIM values.

Test image	nTh	SCA	AOA	FPA	RSA	MPA	ZOA
Img1	2	**0.6115**	**0.6115**	**0.6115**	0.6077	**0.6115**	**0.6115**
	4	0.6964	**0.6967**	0.6766	0.6760	0.6766	0.6766
	6	0.8178	0.8495	0.8602	0.8516	0.7818	**0.8790**
	8	0.8306	0.8865	**0.8918**	0.8725	0.8474	**0.8918**
	10	0.9068	0.9285	0.9337	0.9114	0.9297	**0.9330**
Img2	2	0.5786	0.5786	0.5786	**0.5793**	0.5786	0.5786
	4	0.7007	0.7211	**0.7229**	0.6562	**0.7229**	**0.7229**
	6	0.7025	0.7628	0.7534	**0.8510**	0.7589	0.7974
	8	0.8945	0.8697	0.8366	0.8744	0.9030	**0.9304**
	10	0.9289	0.9173	0.9190	0.8828	0.9395	**0.9432**
Img3	2	0.5779	**0.5843**	**0.5843**	0.5826	**0.5843**	**0.5843**
	4	**0.6877**	0.6759	0.6753	0.6208	0.6533	0.6753
	6	0.7232	0.8184	0.8272	0.7690	0.8173	**0.8297**
	8	0.7608	0.8644	0.8875	0.8433	0.8645	**0.9087**
	10	0.7994	0.8784	0.9301	0.9079	0.8988	**0.9213**
Img4	2	0.5627	**0.5709**	**0.5709**	0.5105	**0.5709**	**0.5709**
	4	0.6722	0.6832	0.6018	**0.6299**	0.6995	**0.6995**
	6	0.8313	**0.8427**	0.8218	0.8062	0.7959	0.8352
	8	0.8346	0.8311	0.8726	0.8543	0.8503	**0.8797**
	10	0.8549	0.8891	0.8935	0.8778	0.8859	**0.8990**
Img5	2	0.6550	0.6453	0.6456	**0.6480**	0.6441	0.6456
	4	0.6600	0.6787	0.6743	0.6935	0.6879	**0.7021**
	6	0.6708	**0.8905**	0.8576	0.7645	0.8632	0.8660

	8	0.7045	0.8901	0.9185	0.9043	0.9110	**0.9223**
	10	0.8317	0.9316	0.9443	0.9144	0.9187	**0.9549**
Img6	**2**	**0.6407**	**0.6407**	**0.6407**	0.6405	**0.6407**	**0.6407**
	4	**0.6968**	0.6918	0.6863	0.6561	0.6863	0.6863
	6	0.7733	0.8121	0.8403	0.7693	0.8409	**0.8620**
	8	0.7909	0.9184	0.9121	0.8141	0.9010	**0.9456**
	10	0.8371	0.9297	0.9434	0.9397	0.9395	**0.9445**
Img7	**2**	**0.6352**	**0.6352**	**0.6352**	0.6280	**0.6352**	**0.6352**
	4	0.6979	0.6738	0.6876	0.6728	0.6876	**0.7391**
	6	0.7932	0.8532	0.8120	0.8152	0.8100	**0.8620**
	8	0.8252	0.8791	0.9044	0.8918	0.8893	**0.9480**
	10	0.8705	0.9236	0.9367	0.9049	0.8929	**0.9326**
Img8	**2**	0.6978	0.6978	0.6978	**0.6991**	0.6978	0.6978
	4	0.7675	0.7749	0.7776	**0.7957**	0.7791	0.7801
	6	0.8451	0.8835	0.8819	0.8722	0.8217	**0.8839**
	8	0.8611	0.9066	0.9196	0.9253	0.9187	**0.9263**
	10	0.8667	0.9192	0.9443	0.9126	0.9390	**0.9608**
Img9	**2**	**0.5544**	**0.5544**	**0.5544**	0.5414	**0.5544**	**0.5544**
	4	0.7304	0.6154	0.6196	**0.7479**	0.6196	0.7200
	6	0.7541	0.7342	0.8478	0.8188	0.8422	**0.8521**
	8	0.8203	0.8686	0.9040	0.8616	0.8905	**0.9151**
	10	0.8923	0.9239	0.9308	0.9238	0.9327	**0.9334**
Img10	**2**	0.6793	0.6793	0.6793	**0.6896**	0.6793	0.6793
	4	0.7455	0.7312	**0.7487**	0.7023	0.7466	**0.7487**
	6	0.8304	0.8850	0.9005	0.9038	0.8972	**0.9187**
	8	**0.9347**	0.9245	0.9334	0.9102	0.9294	0.9311
	10	0.9485	0.9335	0.9506	0.9260	0.9354	**0.9598**

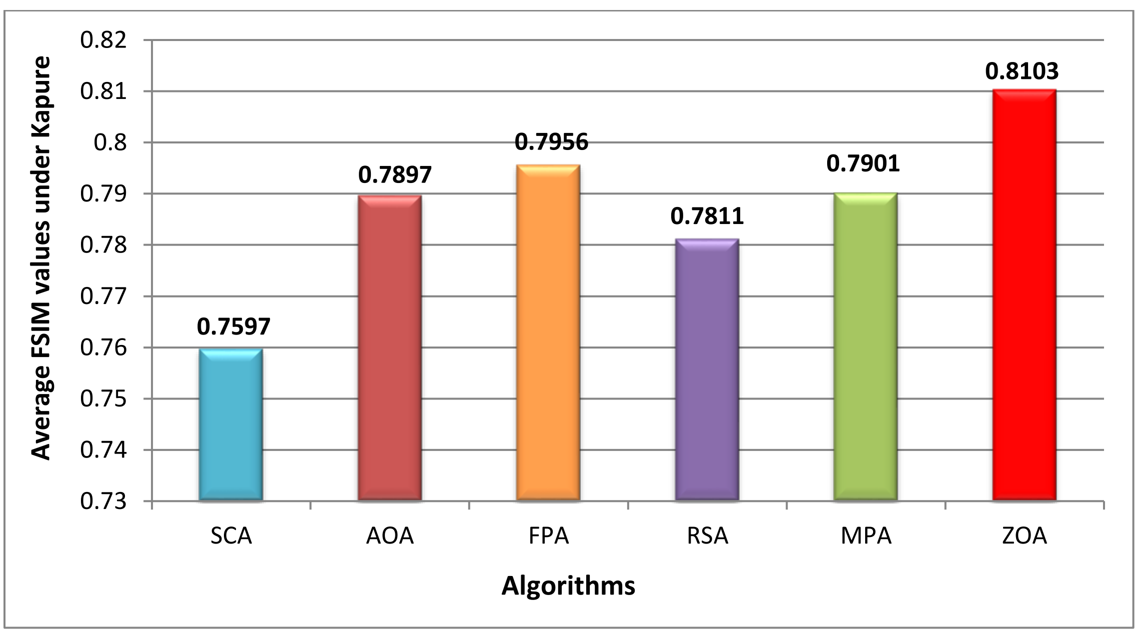


Figure 6: Average FSIM values under Kapur's entropy.

Table 7: Comparison of the ZOA and the other chosen algorithms based on NCC values.

Test image	nTh	SCA	AOA	FPA	RSA	MPA	ZOA
Img1	2	**0.4567**	**0.4567**	**0.4567**	**0.4567**	0.4434	**0.4567**
	4	**0.7234**	0.7048	0.7186	0.7186	0.6971	0.7186
	6	0.9670	0.9603	0.9695	0.7221	**0.9730**	0.9725
	8	0.9520	0.9621	0.9721	0.9679	0.9535	**0.9723**
	10	0.9564	0.9819	0.9878	**0.9875**	0.9643	0.9870
Img2	2	**0.8024**	**0.8024**	**0.8024**	**0.8024**	0.7980	**0.8024**
	4	0.8586	0.8590	**0.8626**	**0.8626**	0.8271	**0.8626**
	6	0.9057	0.8937	0.8715	0.8697	**0.9137**	0.9011
	8	**0.9737**	0.9661	0.9633	**0.9737**	0.9776	0.9735
	10	0.9750	0.9710	0.9821	0.9848	0.9820	**0.9857**
Img3	2	**0.2445**	0.2391	**0.2445**	**0.2445**	0.2391	**0.2445**
	4	0.8287	0.8340	0.8057	0.7642	0.8609	**0.8760**
	6	0.9182	0.9721	0.9755	0.9727	0.9596	**0.9768**
	8	0.9646	0.9817	0.9866	0.9816	0.9600	**0.9824**
	10	0.9792	0.9723	0.9913	0.9881	**0.9916**	0.9903
Img4	2	0.7403	0.7442	0.7403	0.7403	**0.7498**	0.7477
	4	0.8397	0.8337	0.8490	0.8477	0.8641	**0.8842**
	6	0.9549	**0.9858**	0.9836	0.9809	0.9816	0.9801
	8	0.9752	0.9892	0.9903	0.9876	0.9873	**0.9919**
	10	0.9885	0.9907	0.9915	0.9923	0.9901	**0.9927**
Img5	2	0.7822	0.7835	0.7827	0.7812	**0.7910**	0.7485
	4	0.8666	0.8707	0.8724	0.8741	0.8713	**0.8908**
	6	0.9564	0.9544	0.9525	0.9512	0.9551	**0.9630**
	8	0.9601	0.9863	0.9726	0.9819	**0.9878**	0.9850
	10	0.9740	0.9900	0.9862	0.9830	0.9905	**0.9951**
Img6	2	0.6651	0.6651	0.6651	0.6651	0.6486	**0.7041**
	4	0.7019	0.6940	0.7041	0.7041	0.7352	**0.7493**
	6	0.8489	0.8415	0.8461	0.8466	**0.9028**	0.8811
	8	0.8815	0.8938	**0.9766**	0.8725	0.9264	0.9373
	10	0.9845	0.9483	0.9879	0.9866	0.9878	**0.9957**
Img7	2	0.6957	0.6957	0.6957	0.6957	0.6629	**0.7995**
	4	0.7394	0.7317	0.7328	0.7328	0.7603	**0.8019**
	6	0.9476	0.9607	0.9476	0.9478	0.9570	**0.9818**
	8	0.9749	0.9632	0.9841	0.9737	0.9831	**0.9844**
	10	0.9895	**0.9954**	0.9903	0.9734	0.9889	0.9906
Img8	2	0.7756	0.7756	0.7756	0.7756	**0.7793**	0.7756
	4	0.8781	0.8634	**0.8841**	0.8834	0.8288	0.8787
	6	0.9692	0.9663	0.9777	0.9304	0.9820	**0.9872**
	8	0.9855	0.9801	0.9861	0.9866	0.9889	**0.9903**
	10	0.9809	0.9813	0.9816	0.9807	**0.9850**	0.9810
Img9	2	0.6810	0.6810	0.6810	0.6810	0.7104	**0.7178**
	4	0.7184	0.7264	0.7193	0.7193	0.7325	**0.7707**
	6	0.9588	0.9689	**0.9729**	0.9591	0.9539	0.9605
	8	0.9846	0.9739	0.9826	0.9830	0.9790	**0.9832**
	10	0.9914	0.9879	0.9912	0.9912	**0.9935**	0.9921
	2	0.7321	0.7321	0.7321	0.7321	**0.7935**	0.7645

Img10	**4**	0.7685	0.8599	0.7645	0.7641	0.8098	**0.8735**
	6	0.9050	0.9025	0.9120	0.9050	0.9295	**0.9124**
	8	0.9879	0.9811	0.9859	0.9829	0.9868	**0.9911**
	10	0.9827	0.9886	0.9914	0.9829	0.9953	**0.9945**

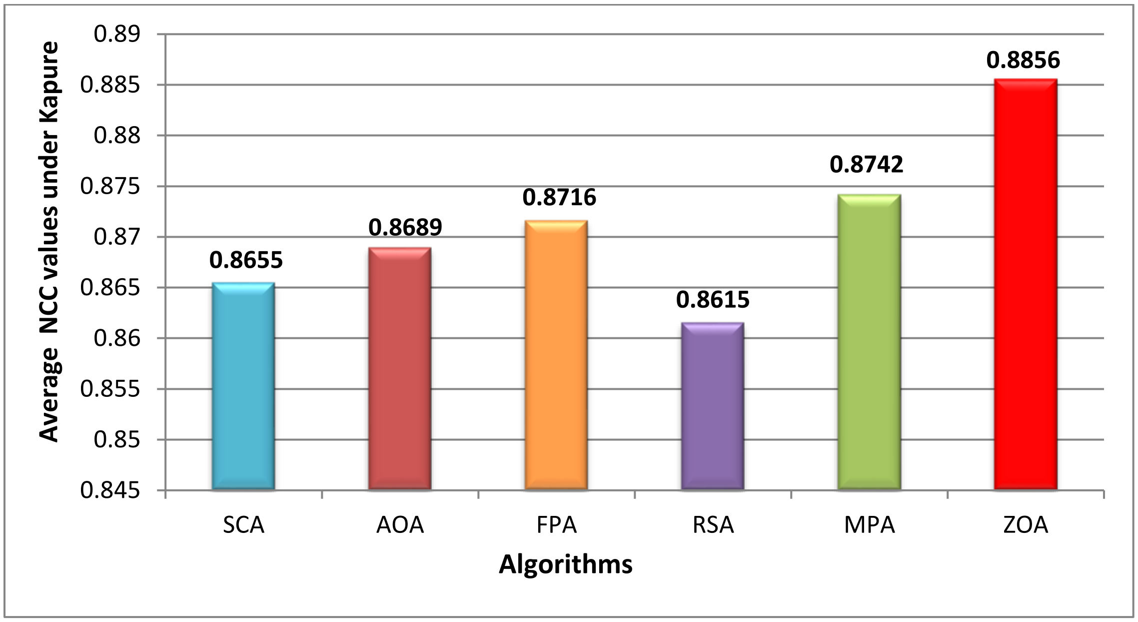


Figure 7: Average NCC values under Kapur's entropy.

5- Conclusion and future work

BTs are among the most common cancers; early detection can greatly reduce the associated death rate. Image segmentation is significant to any Computer-Aided Diagnosis system to extract interest regions from BT images to improve the categorization phase. The thresholding technique is one of the most successful and effective image segmentation techniques. This study discusses the problem of determining the most appropriate threshold values for MT image segmentation. The ZOA with Kapur's fitness function was applied to a group of grayscale BT images. The ZOA's performance is tested utilizing 10 BT images and evaluated against five metaheuristic algorithms, SCA, AOA, FPA, RSA, and MPA, with threshold levels between 2 and 10. Various measures, including MSE, PSNR, FSIM, and NCC, have evaluated ZOA's performance. The experiments' results demonstrated that the suggested

ZOA with Kapur's fitness function exceeds the others in terms of the segmentation metrics MSE, PSNR, FSIM, and NCC.

Future work may involve the novel hybridization of ZOA with one of the other metaheuristics techniques. Additionally, the segmentation of color images may be done using the proposed algorithm.

Abbreviations

BT Brain Tumor

MRI Magnetic Resonance Imaging

MT Multilevel Thresholding

ZOA Zebra Optimization Algorithm

SCA Sine Cosine Algorithm

AOA Arithmetic Optimization Algorithm

FPA Flower Pollination Algorithm

RSA Reptile Search Algorithm

MPA Marine Predators Algorithm

MSE Mean Square Error

PSNR Peak Signal-To-Noise Ratio

FSIM Feature Similarity Index Metric

NCC Normalized Correlation Coefficient

TLABC Teaching-Learning-based Artificial Bee Colony

DFO Dragonfly Optimization

SADFO self-adaptive DFO

GWO Grey Wolf Optimizer

PSO Particle Swarm Optimization

BFO Bacterial Foraging Optimization

FA Firefly Algorithm

WOA Whale Optimization Algorithm

CS Cuckoo Search

ABC artificial bee colony

WDO Wind Driven Optimization

KHO Krill Herd Optimization

MFO Moth Flame Optimization

MPA Marine Predators Algorithm

EOA Equilibrium Optimization Algorithm

COVIDOA Coronavirus Disease Optimization Algorithm

SSA Salp Swarm Algorithm

HHOA Harris Hawks Optimization Algorithm

CSA Crow Search Algorithm

Declarations

- **Availability of Data and Materials**
  Data and Materials is available upon a reasonable request from the corresponding author, Khalid M. Hosny (k_hosny@zu.edu.eg).

- **Competing interests**

  No conflict of interest exists.

- **Funding**

  Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R442), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

- **Authors' contributions:**

Sarah M. Alhammad	Project administration; Funding acquisition; Formal analysis
Doaa Sami Khafaga:	Resources; Curation; Visualization
Doaa Elshora:	Conceptualization; Methodology; Software; Writing Original Draft
Khalid M. Hosny:	Conceptualization; Validation; Writing Review & Editing; Supervision

- **Acknowledgments**
  Non

- **Correspondences:**
  Correspondence and requests for materials should be addressed to Khalid M. Hosny (k_hosny@zu.edu.eg).

References

1. Van Meir, E.G., et al., *Exciting new advances in neuro-oncology: the avenue to a cure for malignant glioma.* CA: a cancer journal for clinicians, 2010. **60**(3): p. 166-193.
2. Louis, D.N., et al., *The 2016 World Health Organization classification of central nervous system tumors: a summary.* Acta neuropathologica, 2016. **131**: pp. 803-820.
3. Usman, M., et al., *Retrospective motion correction in multishot MRI using generative adversarial network.* Scientific Reports, 2020. **10**(1): p. 4786.
4. Houssein, E.H., et al., *A novel black widow optimization algorithm for multilevel thresholding image segmentation.* Expert Systems with Applications, 2021. **167**: p. 114159.
5. Hosny, K.M., et al., *Deep Learning and Optimization-Based Methods for Skin Lesions Segmentation: A Review.* IEEE Access, 2023.
6. Ma, B.J., et al., *Manta ray foraging optimizer-based image segmentation with a two-strategy enhancement.* Knowledge-Based Systems, 2023. **262**: p. 110247.
7. Kapur, J.N., P.K. Sahoo, and A.K. Wong, *A new method for gray-level picture thresholding using the entropy of the histogram.* Computer vision, graphics, and image processing, 1985. **29**(3): p. 273-285.
8. Oliva, D. et al., *Fuzzy entropy approaches for image segmentation.* Metaheuristic algorithms for image segmentation: theory and applications, 2019: p. 141-147.
9. Oliva, D., et al., *Multilevel thresholding segmentation based on harmony search optimization.* Journal of Applied Mathematics, 2013. **2013**.
10. Akay, R., et al., *Multilevel thresholding segmentation of color plant disease images using metaheuristic optimization algorithms.* Neural Computing and Applications, 2022. **34**(2): p. 1161-1179.
11. Sambandam, R.K. and S. Jayaraman, *Self-adaptive dragonfly based optimal thresholding for multilevel segmentation of digital images.* Journal of King Saud University-Computer and Information Sciences, 2018. **30**(4): p. 449-461.
12. Khairuzzaman, A.K.M. and S. Chaudhury, *Multilevel thresholding using grey wolf optimizer for image segmentation.* Expert Systems with Applications, 2017. **86**: p. 64-76.
13. He, L. and S. Huang, *Modified firefly algorithm based multilevel thresholding for color image segmentation.* Neurocomputing, 2017. **240**: p. 152-174.
14. Mirjalili, S. and A. Lewis, *The whale optimization algorithm.* Advances in engineering software, 2016. **95**: p. 51-67.
15. Pare, S., et al., *An efficient method for multilevel color image thresholding using cuckoo search algorithm based on minimum cross entropy.* Applied Soft Computing, 2017. **61**: p. 570-592.
16. Gao, H., et al., *A multilevel thresholding image segmentation based on an improved artificial bee colony algorithm.* Computers & Electrical Engineering, 2018. **70**: p. 931-938.
17. Kotte, S., R.K. Pullakura, and S.K. Injeti, *Optimal multilevel thresholding selection for brain MRI image segmentation based on adaptive wind driven optimization.* Measurement, 2018. **130**: p. 340-361.
18. Beevi, S., M.S. Nair, and G. Bindu, *Automatic segmentation of cell nuclei using Krill Herd optimization based multi-thresholding and localized active contour model.* Biocybernetics and Biomedical Engineering, 2016. **36**(4): p. 584-596.
19. Abd El Aziz, M., A.A. Ewees, and A.E. Hassanien, *Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation.* Expert Systems with Applications, 2017. **83**: p. 242-256.

20. Faramarzi, A., et al., *Marine Predators Algorithm: A nature-inspired metaheuristic.* Expert systems with applications, 2020. **152**: p. 113377.
21. Abdel-Basset, M., V. Chang, and R. Mohamed, *A novel equilibrium optimization algorithm for multi-thresholding image segmentation problems.* Neural Computing and Applications, 2021. **33**: p. 10685-10718.
22. Khalid, A.M., K.M. Hosny, and S. Mirjalili, *COVIDOA: a novel evolutionary optimization algorithm based on coronavirus disease replication lifecycle.* Neural Computing and Applications, 2022. **34**(24): p. 22465-22492.
23. Abualigah, L., et al., *Improved Reptile Search algorithm by Salp Swarm algorithm for medical image segmentation.* Journal of bionic engineering, 2023: p. 1-25.
24. Hosny, K.M., et al., *Multilevel segmentation of 2D and volumetric medical images using hybrid Coronavirus Optimization Algorithm.* Computers in Biology and Medicine, 2022. **150**: p. 106003.
25. Hosny, K.M., et al., *Multilevel thresholding satellite image segmentation using chaotic coronavirus optimization algorithm with hybrid fitness function.* Neural Computing and Applications, 2023. **35**(1): p. 855-886.
26. Upadhyay, P. and J.K. Chhabra, *Kapur's entropy-based optimal multilevel image segmentation using crow search algorithm.* Applied soft computing, 2020. **97**: p. 105522.
27. Rather, S.A. and P.S. Bala, *Constriction coefficient based particle swarm optimization and gravitational search algorithm for multilevel image thresholding.* Expert Systems, 2021. **38**(7): p. e12717.
28. Sezgin, M. and B.l. Sankur, *Survey over image thresholding techniques and quantitative performance evaluation.* Journal of Electronic Imaging, 2004. **13**(1): p. 146-168.
29. Ewees, A.A., M. Abd Elaziz, and D. Oliva, *Image segmentation via multilevel thresholding using hybrid optimization algorithms.* Journal of Electronic Imaging, 2018. **27**(6): p. 063008-063008.
30. Trojovská, E., M. Dehghani, and P. Trojovský, *Zebra optimization algorithm: A new bio-inspired optimization algorithm for solving optimization algorithm.* IEEE Access, 2022. **10**: p. 49445-49473.
31. Pastor, J., Y. Cohen, and N.T. Hobbs, *The roles of large herbivores in ecosystem nutrient cycles.* CONSERVATION BIOLOGY SERIES-CAMBRIDGE-, 2006. **11**: p. 289.
32. Caro, T., et al., *The function of zebra stripes.* Nature Communications, 2014. **5**(1): p. 3535.
33. Estes, R.D., *The behavior guide to African mammals: including hoofed mammals, carnivores, primates*. 2012: Univ of California Press.
34. Kennedy, A.S. and V. Kennedy, *Animals of the Masai Mara*. Vol. 5. 2013: Princeton University Press.
35. Wilson, A.M., et al., *Biomechanics of predator-prey arms race in lion, zebra, cheetah, and impala.* Nature, 2018. **554**(7691): p. 183-188.
36. Mirjalili, S., *SCA: a sine cosine algorithm for solving optimization problems.* Knowledge-based systems, 2016. **96**: p. 120-133.
37. Abualigah, L., et al., *The arithmetic optimization algorithm.* Computer methods in applied mechanics and engineering, 2021. **376**: p. 113609.
38. Yang, X.-S. *Flower pollination algorithm for global optimization*. In *International Conference on unconventional computing and natural computation*. 2012. Springer.
39. Abualigah, L., et al., *Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer.* Expert Systems with Applications, 2022. **191**: p. 116158.
40. Zhang, L., et al., *FSIM: A feature similarity index for image quality assessment.* IEEE Transactions on Image Processing, 2011. **20**(8): p. 2378-2386.