



# Improved Reptile Search Algorithm by Salp Swarm Algorithm for Medical Image Segmentation

Laith Abualigah<sup>1,2,3,4,5</sup> · Mahmoud Habash<sup>6</sup> · Essam Said Hanandeh<sup>7</sup> · Ahmad MohdAziz Hussein<sup>8</sup> · Mohammad Al Shinwan<sup>9</sup> · Raed Abu Zitar<sup>10</sup> · Heming Jia<sup>11</sup>

Received: 21 October 2022 / Revised: 24 December 2022 / Accepted: 4 January 2023 / Published online: 7 February 2023  
© Jilin University 2023

## Abstract

This study proposes a novel nature-inspired meta-heuristic optimizer based on the Reptile Search Algorithm combed with Salp Swarm Algorithm for image segmentation using gray-scale multi-level thresholding, called RSA-SSA. The proposed method introduces a better search space to find the optimal solution at each iteration. However, we proposed RSA-SSA to avoid the searching problem in the same area and determine the optimal multi-level thresholds. The obtained solutions by the proposed method are represented using the image histogram. The proposed RSA-SSA employed Otsu's variance class function to get the best threshold values at each level. The performance measure for the proposed method is valid by detecting fitness function, structural similarity index, peak signal-to-noise ratio, and Friedman ranking test. Several benchmark images of COVID-19 validate the performance of the proposed RSA-SSA. The results showed that the proposed RSA-SSA outperformed other metaheuristics optimization algorithms published in the literature.

**Keywords** Bioinspired · Reptile Search Algorithm · Salp Swarm Algorithm · Multi-level thresholding · Image segmentation · Meta-heuristic algorithm

✉ Laith Abualigah  
aligah.2020@gmail.com

Mahmoud Habash  
Habash@yahoo.com

Essam Said Hanandeh  
hanandeh@zu.edu.jo

Ahmad MohdAziz Hussein  
amihussein@uqu.edu.sa

Mohammad Al Shinwan  
moshanwan@yahoo.com

Raed Abu Zitar  
raed.zitar@sorbonne.ae

Heming Jia  
jjaheminglucky99@126.com

<sup>1</sup> Computer Science Department, Prince Hussein Bin Abdullah Faculty for Information Technology, Al Al-Bayt University, Mafraq 25113, Jordan

<sup>2</sup> Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan

<sup>3</sup> Faculty of Information Technology, Middle East University, Amman 11831, Jordan

<sup>4</sup> Applied Science Research Center, Applied Science Private University, Amman 11931, Jordan

<sup>5</sup> School of Computer Sciences, Universiti Sains Malaysia, 11800 Pulau Pinang, Malaysia

<sup>6</sup> Amman Arab University, Amman, Jordan

<sup>7</sup> Department of Computer Information System, Zarqa University, P.O. Box 13132, Zarqa, Jordan

<sup>8</sup> Deanship of E-Learning and Distance Education, Umm Al-Qura University, Makkah 21955, Saudi Arabia

<sup>9</sup> Faculty of Information Technology, Applied Science Private University, Amman 11931, Jordan

<sup>10</sup> Sorbonne Center of Artificial Intelligence, Sorbonne University-Abu Dhabi, Abu Dhabi, United Arab Emirates

<sup>11</sup> School of Information Engineering, Sanming University, Sanming 365004, China

## 1 Introduction

Image segmentation is the base of processing actions in each image and machine vision [1]. The segmentation system is becoming more favored and has been used in domains such as bio-medical, urban vehicles, farming production, etc. Image segmentation is required when component selection or tracking image details is needed [2]. In the meantime, the rate of succeeding image processing stages, such as recognition, is involved in the image segmentation outcome [3]. Thus, the segmentation of images has evolved into an intimate rapprochement with our stamina, accuracy, and efficiency directly impacting the result of the practical application, so the segmentation of images is significantly essential [4, 5].

Over the past decades, multiple segmentation techniques have been suggested. Since the per image contains additional typical facts, its category is various. There are mainly four segmentation approaches: thresholding, boundary-based, and hybrid approaches [6, 7]. Among image segmentation approaches, the thresholding approach has obtained wide alert because of its clarity, storage space, fast time processing, and ease of manipulation. The histogram is necessary for the threshold approach's outcomes. It is frequently used to categorize the pixels in a scene. There are two options for choosing an image threshold: bi-level and multi-level thresholding [8, 9]. First, the image is divided into two classes based on a threshold weight. It divides pixels into two categories based on how far they deviate from a threshold weight: most prevalent pixels and less thick pixels. They are frequently used to separate the images' foreground and backdrop. The process of dividing images into two classes is unacceptable, especially when the image is more complex and contains multiple entities with the same range of gray-scale [10, 11].

Thus, it is essential to boost the threshold statuses from bi-level to multi-level [12]. The multi-level threshold (MTH) divides an image into multi categories belonging to multiple entities in the image [13, 14]. This way, it is feasible to indicate more details and entities from the segmented image. Thresholding techniques affect the selection of many thresholds utilizing some features illustrated in images. Among the thresholding techniques, Tsallis, Kapur, and Otsu are the most popular ones [15]. Otsu technique maximizes the variance between classes. Kapur techniques maximize the rear entropy of the split categories to see the best thresholds [16, 17].

Beneath the possibility of segmenting intricate images, the multi-level thresholding image segmentation technique can split the prey territory accurately [18–21]. Regardless, as the numeral of thresholds grows, the added dose of technology increases and the processing duration will be slow.

It is a vital and challenging duty for classic comprehensive techniques due to the high computational outlay [22, 23]. Accordingly, the image threshold technique is sometimes mixed with an intelligent optimization technique to fix the technique's weaknesses—for example, genetic, simulated annealing, and so on [24, 25]. Also, many unused methods can be used to solve the segmentation problem, such as African Vultures Optimization Algorithm [26], Farmland Fertility [27], Artificial Gorilla Troops Optimizer [28], Prairie Dog Optimization Algorithm [29], Differential Evolution [30], Butterfly Optimization Algorithm [31, 32], Gazelle Optimization Algorithm [33], and Mountain Gazelle Optimizer [34].

In this paper, we improved the performance of the Reptile Search Algorithm (RSA) with Salp Swarm Algorithm (SSA) to determine the optimal thresholds in multi-level image segmentation. The proposed method aims to tackle the main problems raised by the original methods, like premature convergence and unbalance between the search process. Several benchmark images are used to assess the efficiency of the proposed method (RSA-SSA), and the results are compared with other well-known optimization techniques in the literature. The fitness function, square root error, signal-to-noise ratio, and Friedman ranking test are a few commonly used measures to evaluate the effectiveness of the proposed segmentation method.

The goal of multi-level thresholding is to find the ideal threshold values that divide an image's pixels into many groups; these values can be found using meta-heuristic algorithm optimization techniques. Optimizing meta-heuristic algorithms is challenging as it requires the model to reach the optimal thresholds to segment images more accurately. The conventional methods work when there are fewer thresholds, but there is an issue when there are more thresholds. The suggested RSA optimized by SSA avoids becoming trapped in a local optimum and can discover the best solution for the image to address this issue and improve segmentation accuracy. In RSA, there is a problem with the exploitation phase. It is the possibility of repetition in generating or stopping at the exact solutions. So, when we get to that state, we'll boost that stage with the SSA. The SSA algorithm has the advantage of efficient area exploitation and speed in area exploitation.

The main objective of image segmentation is to segment the image more efficiently and highlight the importance of the items in images that are getting from datasets. The segmentation process is significant and crucial in many functions, including image processing. An image is divided into classes pertaining to various objects using a multi-level threshold. Consequently, extracting more data and objects from the segmented image is feasible. An optimization challenge is determining the best thresholding for image

segmentation in a multi-level thresholding condition. The reptile search approach (RSA), a unique algorithm, is utilized in our work to handle this problem with the SSA technique. In our study, we try to improve the RSA to identify the best threshold for image segmentation, which is done by boosting the SSA. The proposal seeks to identify the best threshold value to maximize Otsu's worth. This study aids anyone interested in image segmentation and optimization methods by providing theoretical and practical value to researchers when achieving goals.

The remaining sections are structured as follows: Sect. 2 summarizes previous and ongoing studies on image segmentation using multi-level thresholding and meta-heuristic algorithms. Section 3 presents the study approach for image segmentation by multi-level thresholding with meta-heuristic algorithms. Section 4 illustrates the testing and analysis of the suggested approach and compares the outcomes of the proposed method and another previous method. Section 5 presents the paper's conclusion, followed by ideas and inspirations for additional research in algorithms.

## 2 Related Work

The technique of grouping pixels into various groups is one of the most significant variables influencing image processing. It enables us to recognize and distinguish the objects in the image [35]. The Threshold approach is the most straightforward of the several techniques used to segment the images. The image histogram and the number of thresholds have the most considerable effects on the threshold technique. To divide the graph into separate classes, thresholds are utilized. The pixels are split into two groups in the case of a unique threshold known as a bi-level threshold. It is referred to as a multi-level threshold if there are more than two classes and more than one threshold. Finding the optimal thresholds in the image related to the number of classes is the key challenge with multi-level thresholds (or objects in the image) [36].

Several metrics, including variance and entropy, are utilized to identify the best thresholds. Contrast is used in the Otsu technique [53], whereas the Kapur method employs entropy [37]. Meta-heuristic algorithms (MA) are required

when the number of thresholds exceeds one; for the most part, they produce exact results [38].

Image Segmentation is a base of the critical pretreatment stages in computer science. In the present-day, image is a helpful manner of knowledge asset and contact for mortal beings. The application of the technique is evolving more favored, and it has been used in diverse fields. Image segmentation is required when element selection or hunting of image knowledge is required [39]. Image segmentation is an algorithm that separates the image into some disjoint areas by characteristics such as gray, colored, spatial surface and geometric form. These characteristics indicate character or nearness in the same area but offer specific distinctions among various areas [40].

Multi-level thresholding is a segmentation technique called threshold that divides pixels into various classes based on their level of intensity and following one or more threshold values. Given that it is one of the most straightforward partitioning methods, this method's simplicity is one of its most significant advantages. This is because the segmented images produced by the threshold have a tiny storage footprint, quick processing times, and are simple to process. The histogram and the thresholds specified are two significant aspects considered by segmentation using thresholds. There are two kinds of thresholds; the bi-level threshold is the first type. The second sort of threshold is a multi-level threshold, which is employed when more than one threshold value is used [41].

Meta-heuristic optimization algorithms are built around two fundamental tasks: (1) exploration, which refers to the algorithm's capacity to examine non-local (global) locations. (2) Exploitation: this refers to the algorithm's capacity to discover superior optimization solutions within those overall solutions. A suitable balance between these tactics is required (exploration and exploitation) [42]. In addition to evolutionary algorithms, swarm intelligence algorithms and physics-based or human-based approaches are among the most significant classes on which meta-heuristics are built.

In this paper [43], a multi-threshold boosting of outline segment selection in the medical image is presented in this study. First, according to various grades of racket between other medical images, the gray approach is employed to study the span of correlation between a per pixel. Besides, multiple thresholds are a group to supply a limited band

**Table 1** Related studies

References	Methods	Type of images
[44]	Bloch quantum artificial bee colony	Gray-level image thresholds
[50]	CCACO+ (HCS, VCS)	Medical image segmentation
[51]	Fiction CLACO-based, scilicet CLACO-MIS	X-ray of COVID-19
[16]	Marine Predator algorithm + (GA and PSO)	Medical image segmentation
[36]	MPA optimization + MFO	Medical image segmentation

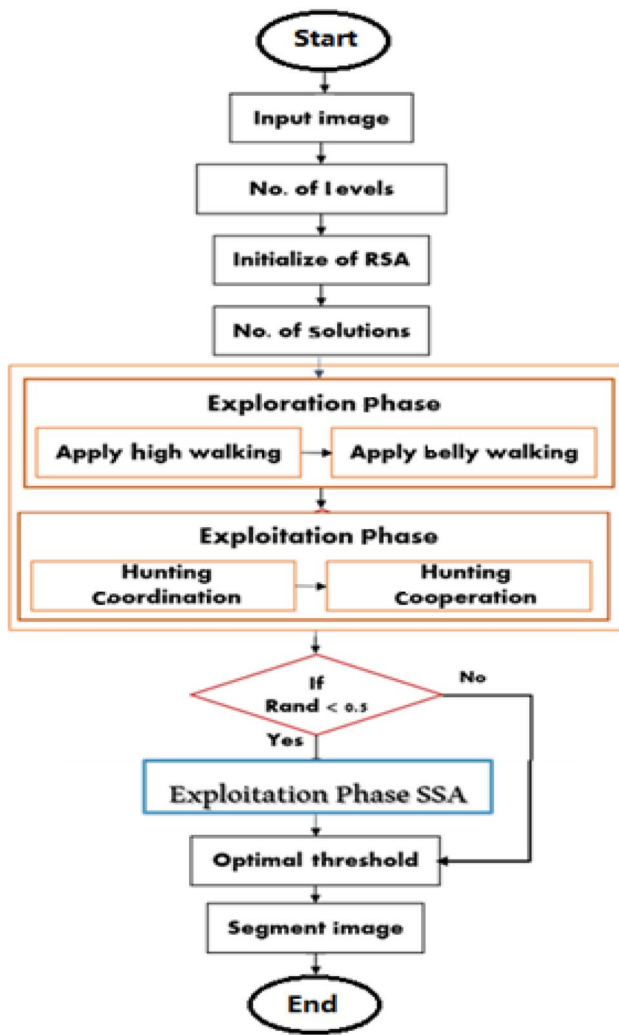


Fig. 1 The proposed method

restraint before image outline selection. The Otsu function is employed to explore the related space, and the target segmentation threshold of the medical image outline component is obtained after a majority of the restraint threshold of outline components in the medical image. These thresholds are searched more quickly through genetic algorithms. Meanwhile, processes such as extract, crossover, modification, etc., are employed to form inhabitants better adaptive to the new domain in randomly developed inhabitants. The golden quote technique is operated to translate various histograms in medical images. Meanwhile, the mean technique is employed to get the last threshold of outline component selection in the medical image.

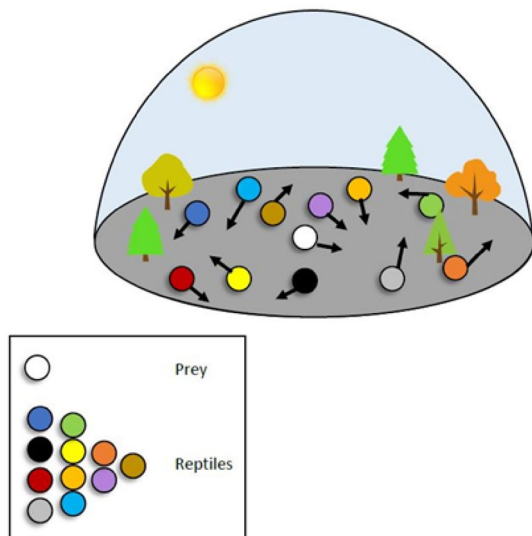
Therefore, outline components are selected by the multi-threshold optimization in the medical image. Finally, testing outcomes indicate that the suggested technique decreases the attribute extent and the attribute selection time, which means

that this method has immediate relevancy and supplies a specific theoretical base for future medical image studies. Although this research has reached acceptable outcomes, many points must be improved and in-depth. The real selection technique cannot ultimately be automated. Because of the complication of determining tissues and organs in (medical images), the technique must be finished manually.

In this paper [44], quantum encoding and the idea of the quantum web are combined into Bloch globe to create a boosting Bloch quantum artificial bee colony approach. The approach has the most extraordinary rapidity and accuracy by assuring heightened dimensional multi-modal standard procedures. The approach is devoted to various kinds of gray-level image thresholds, and various approaches affirm the significance of the enhanced approach. In the first position, we utilize the Bloch spheroidal ordinary of IBQABC approach for encoding the initial conception, which can bypass the randomness of quantum stature in developing binary encoding. Lastly, the quantum entanglement concept is in the invention of unique nectar enhancement procedure, which can realize by the emotional modification of grade in enhancing the process and dynamically adjusting its standing on the externals of Bloch. The approach greatly enhances the capability to bounce out of the regional best and the general tracking effectively of the bee colony approach. Although this paper has reached suitable outcomes, other factual procedures could not be assessed using this technique, such as variance between classes (Otsu), crossover entropy, etc. This paper also did not have been involved in operating color images.

In this paper [45], an enhanced technique named CCACO is suggested depending on HCS, VCS, and enhanced choice means. Among them, HCS enhances the conjunction rate of the technique in decoding the continued optimization issue. VCS and the enhanced selecting mean enhanced the capability to evade dropping into LO and conjunction accurateness. By approximating CCACO, actual ACOR, basic technique, and enhanced technique on 30 standard procedures, all testing outcomes indicate briefer conjunction speed, high confluence accuracy, and more muscular capability to evade dropping into LO are the heart benefits of CCACO. But the study, we report that the issue of the increased computational sophistication is still. So, they must consider orienting high-performance computing techniques into CCACO to enhance computing effectively.

This paper [46] suggests a fiction CLACO-based, scilicet CLACO-MIS, and uses it to segment the X-ray of COVID-19, enhance the diagnoses class of COVID-19, and obtain segmentation outcomes with an increased rate. The Cauchy transformation is involved in the end stage of ant foraging, which improves the search for ants. CLACO resembled ten of techniques. At last, they studied and resembled all the testing outcomes again employing WSRT and FT, which



**Fig. 2** Encircling the prey

powerfully depict that CLACO includes sweetened searching, further progress in conjunction with speed, and an enhanced capability to bounce out of the regional optimum. Regardless, due to the preface of this transformation and the ungenerous Levy transformation, the time complexity of CLACO MIS advances and the CPU calculation time needed for its segmentation has a companion attachment.

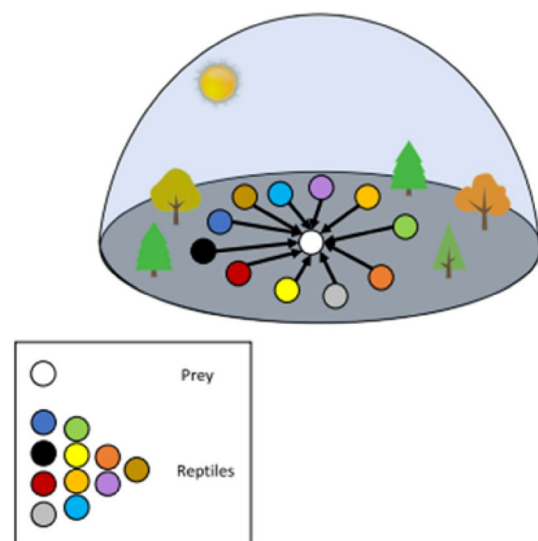
In Ref. [47], the researcher evaluated the performance of the Marine Predator algorithm on several tests at randomly generated landscapes, the real-world engineering design problem for ventilation fields and the premises energy performance, several engineering standards, and CEC-BC-2017 tests. The algorithm was compared to other optimization algorithms: the first is GA and PSO; the second is GSA, CS, and SSA, which are recent algorithms; and the third is CMA-ES, SHADE, and LSHADE-cnEpSin. The Marine Predator algorithm came in second place as the best performance approach, outperforming LSHADE-cnEpSin regarding outcomes. One of the winning algorithms in the CEC 2017 competition is the Marine Predator algorithm. Several optimization techniques were compared to the algorithm. The first is GA and PSO, one of the algorithms undergoing extensive research. The second is the more contemporary algorithms, GSA, CS, and SSA. The third group consists of the performance optimizers CMA-ES, SHADE, and LSHADE-cnEpSin. The Marine Predator algorithm came in second place as the best performance approach and had results that were competitive with LSHADE-cnEpSin. The Marine Predator algorithm was one of the CEC 2017 competition's winners.

This study compares the proposed algorithm to various optimization algorithms [40]. The first is GA and PSO, which are one of the algorithms that have been thoroughly

investigated. The second is GSA, CS, and SSA, which are recent algorithms. The third is CMA-ES, SHADE, and LSHADE-cnEpSin, performance optimizers. They also took first place in the IEEE CEC contest. The Marine Predator algorithm came in second place as the best performance approach and had results that were competitive with LSHADE-cnEpSin. The Marine Predator algorithm was one of the CEC 2017 competition's winners. The proposed approach aids specialists in image segmentation by an intelligent platform to establish threshold values, according to the findings of this study. Based on the suggested approach's high performance, it may be used in subsequent studies for various applications, including (1) feature selection, (2) multi-objective method, (3) global optimization, and (4) cloud computing.

A successful multi-level threshold (MTH) method is reported for image segmentation, including COVID-19 CT scans and medical image segmentation [48]. The Marine Predatory Animal (MPA) algorithm is recommended. A new SI approach is MPA. The suggestion offers the first application for usage in image segmentation, according to the researchers' study. The MPA optimization procedure makes use of the (MFO) algorithm. The new plan is referred to as MPAMFO. The proposal was assessed using a range of photographs since it contained cross-sectional coronavirus images (COVID-19). All tests were performed well and consistently, according to the results. Additionally, several comparisons were used to demonstrate how better the MPAMFO proposal was than many other algorithms, such as PSO, GWO, and CS, regarding SSIM, fitness value, and PSNR.

The authors introduced an efficient multi-level threshold (MTH) method for image segmentation, including segmentation of COVID-19 CT images and medical images [49].



**Fig. 3** Attacking the prey

The Marine Predator Animal (MPA) algorithm has been proposed. A new SI technique is MPA. The suggestion offers the first usage for image segmentation, according to a study by the researchers. The MPA optimization method employs the (MFO) algorithm. MPAMFO is the name of the new proposal. The inclusion of cross-sectional coronavirus images in the proposal was examined using a range of images (COVID-19). The results indicated consistent and successful performance across all exams. Additionally, comparisons were used to demonstrate the MPAMFO proposal's superiority over various algorithms, including PSO, GWO, and CS, regarding SSIM, fitness value, and PSNR.

An efficient multi-level threshold method for image segmentation, including segmentation of COVID-19 CT images and medical images [6]. The Marine Predator Algorithm (MPA) algorithm has been proposed. A new SI technique is MPA. The suggestion offers the first usage for image segmentation, according to a study by the researchers. The MPA optimization method employed the MFO algorithm. MPAMFO is the name of the new proposal. Due to the inclusion of cross-sectional Coronavirus images in the proposal, it was examined using a range of images (COVID-19). Results indicated consistent and successful performance across all exams. Additionally, numerous comparisons were used to demonstrate the MPAMFO proposal's superiority over various algorithms, including PSO, GWO, and CS, in terms of SSIM, fitness value, and PSNR., as well as evaluating the fitness function. This means that hybridization of the SSA gives better results than it does separately. An overview of the related studies is given in Table 1.

The main objective of this study was to segment the image more efficiently and highlight the importance of the items in images obtained from datasets. In this work, we try to detect the optimal threshold to segment the image. We are

expected to be able to identify the best threshold for image segmentation. In addition, we will also present a promising formulation using boosting algorithms that frame the problem of identifying the best threshold for segmentation.

### 3 The Proposed Method

To address the issue raised in problem 1.2, it is essential to establish the best multi-level thresholds for image segmentation. The best threshold values are determined by maximizing a standard, such as the Otsu standard. The ideal threshold value for the Otsu-based method maximizes the variance between classes.

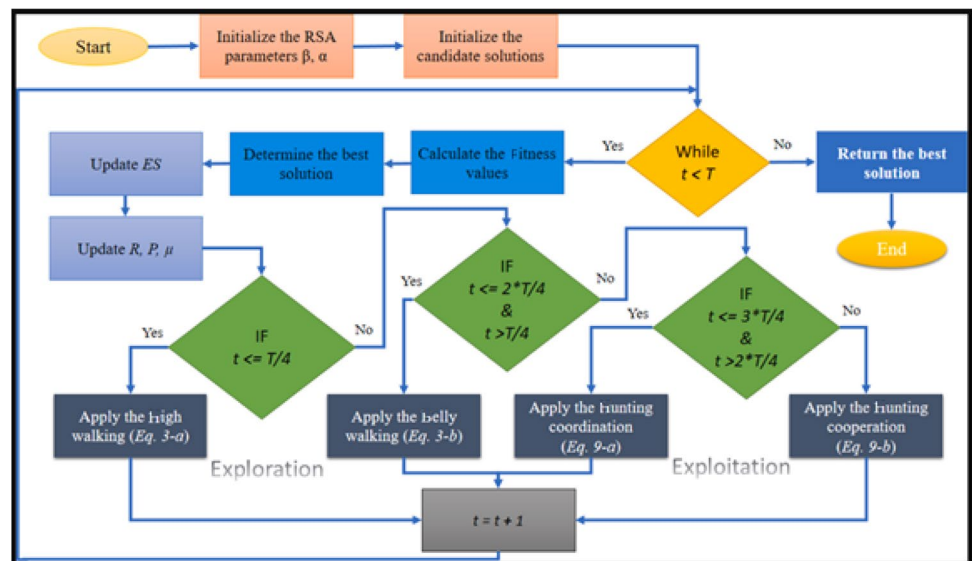
Therefore, this section will outline the problem's formulation, RSA, and its key components. To find the best fitness, exploration and exploitation will be improved for each solution, starting with the initialized parameter. Additionally, as illustrated in Fig. 1, SSA will start with the initialized parameter and will be strengthened with each RSA less than the rand parameter solution.

#### 3.1 Formulation of the Problem

Thresholding is a useful method of image segmentation [38]. It classifies all pixels in an image (color or gray) into various classes. As we know, a bi-level threshold is not useful with a complex image [52]. It is used to segment images only into two objects. So, it can be extended to multi-level thresholding as follows:

- $C1 \leftarrow p$  if  $0 \leq p < th1$
- $C2 \leftarrow p$  if  $th1 \leq p < th2$
- $Ci \leftarrow p$  if  $thi \leq p < thi + 1$
- $Cn \leftarrow p$  if  $thn \leq p < L - 1$

Fig. 4 Flowchart of the reptile search algorithm (RSA)



Bi-level/multi-level thresholding aims to determine the  $i$ th values which appropriately segment an image into various classes. Where  $(1, \dots, n)$  are a number of threshold values,  $P$  is the value of the gray level. In our study, Otsu’s function is used.

**3.1.1 Between-Class Variance Method (Otsu’s method)**

Searching for optimal threshold values in Otsu’s method is by maximizing the between-class variance. Assume an image consisting of  $N$  pixels represented in  $L$  gray levels, and the number of pixels with gray level  $i$  is presented by  $f_i$  [53]. The probability of gray level  $i$  is defined as in Eq. (1).

$$p_i^c = \frac{f_i^c}{N}, p_i^c \geq 0, \sum_{i=0}^{L-1} p_i^c = 1, \tag{1}$$

$$c = \begin{cases} 1, 2, 3 & \text{if } RGB \text{ image} \\ 1, 2, 3 & \text{if gray - scale image} \end{cases}$$

where  $i$  is a gray level between 0 to  $L - 1$ ,  $c$  represents the image component. RGB color images have three separate components: red, green, and blue [54].

Assume an image is divided into  $m$  classes and has  $m - 1$  thresholding values. Otsu’s method extended the bi-level thresholding to multi-level thresholding. The extended between-class variance function can be described in Eq. (2):

$$f^c(t) = \sum_{i=0}^{m-1} \sigma_i^c. \tag{2}$$

The optimal thresholding values  $(t_1^{*c}, t_2^{*c}, t_3^{*c}, \dots, t_{m-1}^{*c})$  are calculated by maximizing  $\sigma_B^c$  as follow Eq. (3).

$$t_1^{*c}, t_2^{*c}, t_3^{*c}, \dots, t_{m-1}^{*c} = \arg \max_{0 \leq t_1, \dots, t_{m-1} \leq L-1} \{ \sigma_B^c(t_1^c, t_2^c, \dots, t_{m-1}^c) \} \tag{3}$$

**3.2 Reptile Search Algorithm (RSA)**

RSA draws inspiration from crocodiles’ natural surroundings, hunting, and social behavior. The two phases of crocodile behavior—exploration (global) and exploitation (local)—consist of encircling and hunting the victim. These mechanisms are mathematically represented to provide the suggested RSA and carry out the optimization procedures. Since RSA is a population-based and gradient-free method, it can be applied to simple and complex optimization problems subject to predetermined restrictions. Crocodiles benefit from cohesive groups because they increase their hardiness and allow for active cooperation [55].

Crocodiles’ sleek bodies increased their speed. They can move more swiftly because it makes moving easier. When they walk, they elevate their feet to the side, which causes

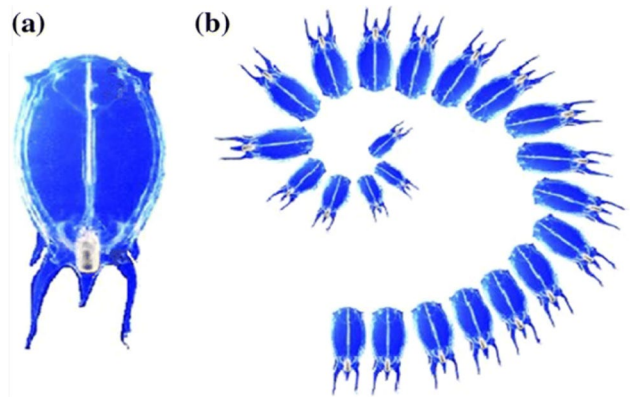


Fig. 5 Behavior of Salp swarm algorithms

them to move more quickly. Since animals typically travel from one area to another via walking, their feet constitute a distinguishing characteristic. The following are the primary descriptions of the Crocodile.

**3.2.1 Initialization Phase**

In RSA, the optimization process begins with a collection of stochastically produced candidate solutions ( $X$ ), as stated in Eq. (4), and the best-obtained solution is regarded as roughly the optimum in each iteration.

$$X = \begin{bmatrix} x_{1,1} & \dots & x_{1,j} & x_{1,n} \\ x_{2,1} & & x_{2,j} & x_{2,n} \\ \vdots & \ddots & \vdots & \\ x_{N-1,1} & \dots & x_{N-1,j} & x_{N-1,n-1} \\ x_{N,1} & & x_{N,j} & x_{N,n} \end{bmatrix} \tag{4}$$

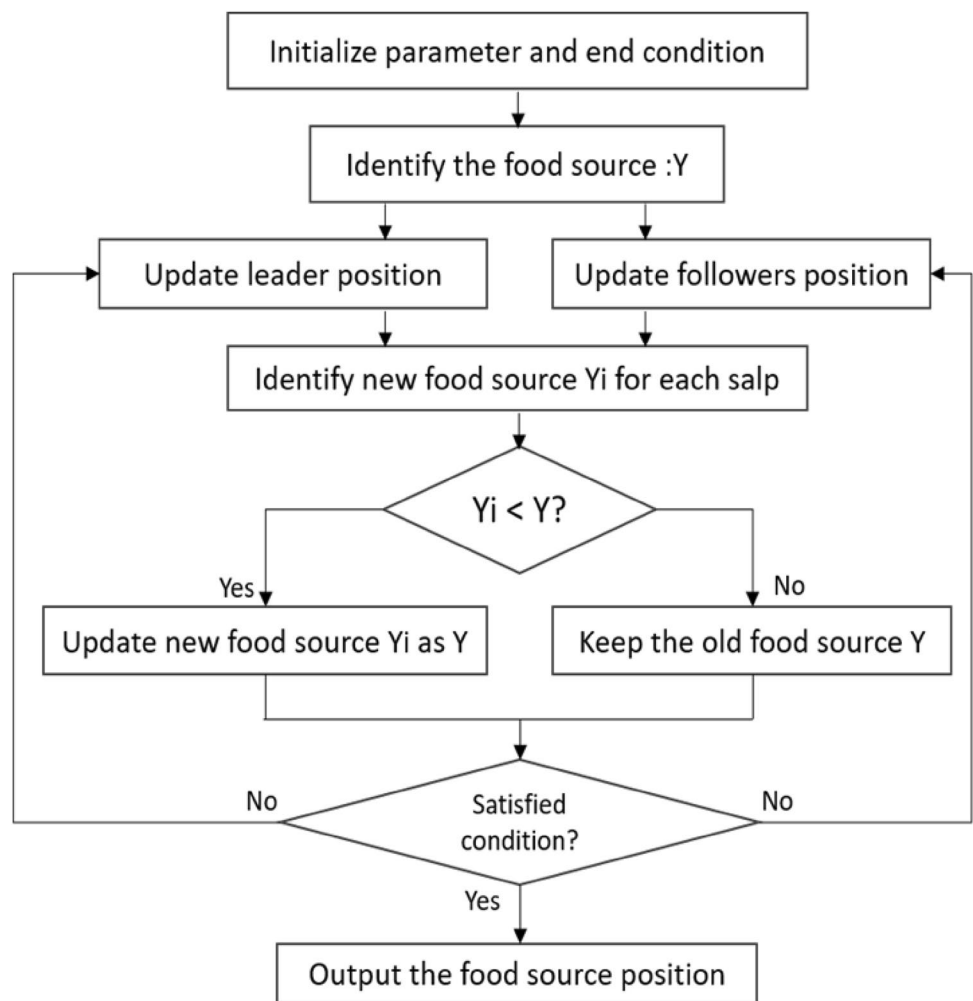
$$x_{(i,j)} = \text{rand} + (\text{UB} - \text{LB}) + \text{LB}, J = 1, 2, \dots, n, \tag{5}$$

where  $N$  is the number of candidate solutions,  $X$  is a set of the candidate solutions that are produced at random using Eq. (5),  $x_{i,j}$  signifies the  $j$ th location of the  $i$ th solution, and  $n$  denotes the dimension size of the given problem.

**3.2.2 Exploration Strategy (Encircling)**

Crocodiles in this section have two movements during the encircling high walking and belly walking, following the encircling habit. Contrary to another search phase, these movements (high and belly walking) make it difficult for crocodiles to approach the intended prey due to their disturbance (hunting phase). As a result, the exploratory search uncovers a sizable search space, as illustrated in Fig. 2, and may eventually be able to locate the density area. In this figure, the search process is clearly going in a wide

**Fig. 6** Salp swarm algorithm flowchart



area. Additionally, at this optimization level, the exploration mechanisms are used to support the other phase of the search—hunting and exploration—through vast and dispersed research [56].

The RSA can switch between the encircling (exploration) and hunting (extraction) search phases, and it does so based on four different parameters. Based on two primary search algorithms, the RSA exploration mechanisms investigate the search regions and seek a better solution (high walking strategy and belly walking strategy). There are two prerequisites for moving on to this stage of the search. The belly walking movement strategy depends on  $t \geq 2(T/4)$  and  $t > T/4$ , whereas the high walking movement strategy depends on  $t \leq T/4$  as shown briefly in Sect. 3.2.3.

This indicates that this requirement will be met for almost half of the exploration iterations (high walking) and the other half (belly walking). These are two techniques for exploration searches. Observe that utilizing Eq. (6), the stochastic scaling coefficient is investigated for the element to provide more diverse solutions and explore the diverse location.

$$\begin{aligned}
 &tx_{(i,j)}(t + 1) \\
 &= \begin{cases} \text{best}_j(t) \times \eta_{(i,j)}(t) \times \beta - R_{(i,j)}(t) \times \text{rand}, t \leq \frac{T}{4} \\ \text{best}_j(t) \times x_{(r_1,j)} \times ES(t) \times \text{rand}, t \leq 2\frac{T}{4} \text{ and } t > \frac{T}{4} \end{cases} \quad (6)
 \end{aligned}$$

### 3.2.3 Exploitation Strategy (Hunting)

Crocodiles use coordination and collaboration as their primary hunting techniques, both indicated by their hunting (exploitation) behavior. These tactics all refer to various intensification methods that focus on the exploitation search (locally). Contrary to encircling processes, as depicted in Fig. 3, crocodile techniques (hunting coordination and cooperation) allow them to approach the target prey readily due to their intensity. This figure clearly shows that the search focuses on the targeted area and narrows to the main aim. As a result, the exploitation search eventually finds the ideal solution, possibly after numerous attempts. Additionally, at this optimization stage, the exploitation mechanisms are



used to perform an intensified search close to the ideal solution and highlight communication between them.

The RSA exploitation mechanisms use two primary search techniques (hunting coordination and cooperation), depicted in Eq. (7), to explore the search space and identify the best solution. When  $t \leq 3(T/4)$  and  $t > 2(T/4)$ , the hunting coordination method is used to search; otherwise, the hunting cooperation approach is used when  $t \leq T$  and  $t > 3(T/4)$ .

$$x_{(i,j)}(t+1) = \begin{cases} \text{best}_j(t) \times P_{(i,j)}(t) \times \text{rand}, & t \leq 3\frac{T}{4} \text{ and } t > 2\frac{T}{4} \\ \text{best}_j(t) - \eta_{(i,j)}(t) \times \epsilon - R_{(i,j)}(t) \times \text{rand}, & t \leq T \text{ and } t > 3\frac{T}{4} \end{cases} \quad (7)$$

### 3.2.4 Reptile Search Algorithm (RSA)

The optimization process starts by creating a random set of candidate solutions to investigate potential placements for

the nearly ideal solution. According to the procedures of the suggested RSA at each iteration, each solution swaps out its spots from the best-obtained solution (Fig. 4).

The search procedures are separated into two basic approaches (exploration and exploitation) with the following four strategies to highlight exploration and exploitation.

**3.2.4.1 High Walking** In the exploration phase, the reptile accepts the prey area and chooses the best hunting area by the high walking movement strategy performed when  $t \leq T/4$ , using the following Eq. (8).

$$x_{(i,j)}(t+1) = \text{best}_j(t) \times \eta_{(i,j)}(t) \times \beta - R_{(i,j)}(t) \times \text{rand}, t \leq \frac{T}{4} \quad (8)$$

where  $x_{((i,j))}(t+1)$  represents the solution produced by the initial search strategy at iteration  $t$ . Where  $\text{rand}$  stands for a random number between 0 and 1,  $\text{Best}_j(t)$  stands for the  $j$ th location in the best solution so far, and  $\eta(i,j)$  stands for the

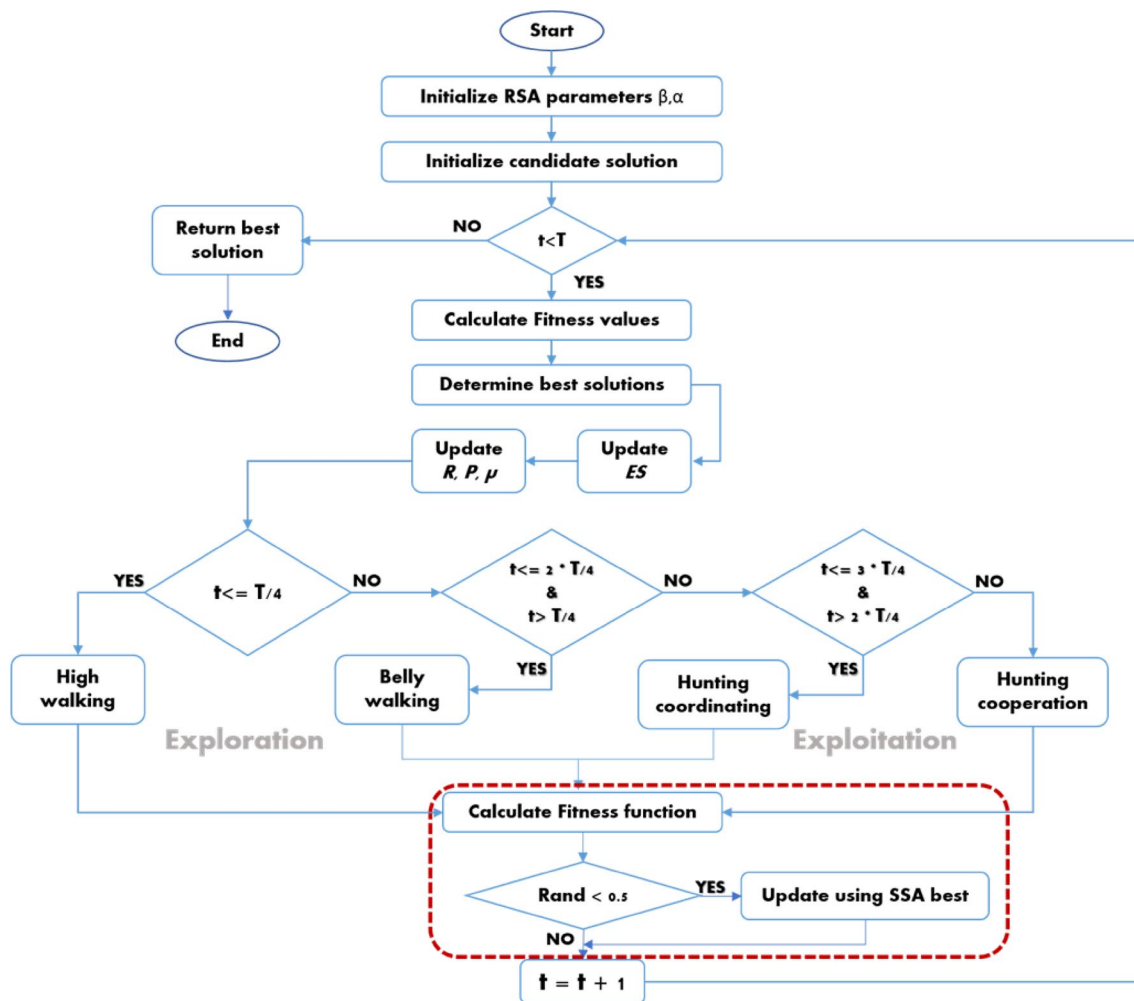


Fig. 7 The proposed hybrid algorithm (RSA-SSA)

hunting operator for the  $j$ th place in the  $i$ th solution, which is derived using Eq. (8).  $\beta$  is a sensitive parameter that, with a fixed value of 0.1, determines the exploration accuracy (high walking) during the encircling phase over the course of iterations. The search region is reduced using the Reduce function ( $R(i,j)$ ), derived using Eqs. (9–10).  $\text{rand}$  is the name for a random number.  $t$  and  $T$  are the current iteration and the maximum number of iterations, respectively.

$$\eta_{(i,j)}(t) = \text{best}_j(t) \times P_{(i,j)}, \tag{9}$$

$$R_{(i,j)}(t) = \frac{\text{best}_j(t) \times x_{(r2,j)}}{\text{best}_j(t) \times \epsilon}. \tag{10}$$

Using Eq. (11) and as mentioned in approach 3,  $P_{(i,j)}$  is the percentage difference between the  $j$ th position of the best-obtained solution and the  $j$ th position of the present solution.

**3.2.4.2 Belly Walking Exploration** When the prey area is discovered and  $t \leq T/4$ , the Reptile belly walking movement strategy is performed, prepares the area, and watches prey, using the following Eq. (11).

$$x_{(i,j)}(t + 1) = \text{best}_j(t) \times x_{(r1,j)} \times ES(t) \times \text{rand}, t \leq 2\frac{T}{4} \text{ and } t > \frac{T}{4} \tag{11}$$

When  $r1$  is a random number between  $[1N]$ ,  $x_{(r1,j)}$  refers to the  $i$ th solution’s random position. The number of potential solutions is  $N$ . Equation (12) is used to calculate the

probability ratio known as evolutionary sense ( $ES(t)$ ), which randomly decreases in value from 2 throughout the course of the repetitions.

$$ES(t) = 2 \times r_3 \times \left(1 - \frac{1}{T}\right), \tag{12}$$

These two search methods uncover a large search region; they may locate the density area after multiple attempts. Additionally, during this optimization level, the high and belly walking exploration mechanisms help the other phase of the search process (hunting/exploration) through thorough and dispersed investigation.

**3.2.4.3 Hunting of Coordination Exploitation** When  $t \leq T/2$ , candidate solutions try to widen the search space, and when  $t > T/2$ , they try to converge on the nearly ideal solution. The Reptile is prepared to attack when the prey region is precisely specified, and Eq. (13) hunting coordination approach is used when  $t \leq 3(T/4)$  and  $t > 2(T/4)$ .

$$x_{(i,j)}(t + 1) = \text{best}_j(t) \times P_{(i,j)}(t) \times \text{rand}, t \leq 3\frac{T}{4} \text{ and } t > 2\frac{T}{4} \tag{13}$$

where  $\text{Best}_j(t)$  is the  $j$ th position in the best-obtained solution so far,  $P_{(i,j)}$  is the percentage difference between the  $j$ th position of the best-obtained solution and the  $j$ th position of the current solution, which is calculated using Eq. (14).

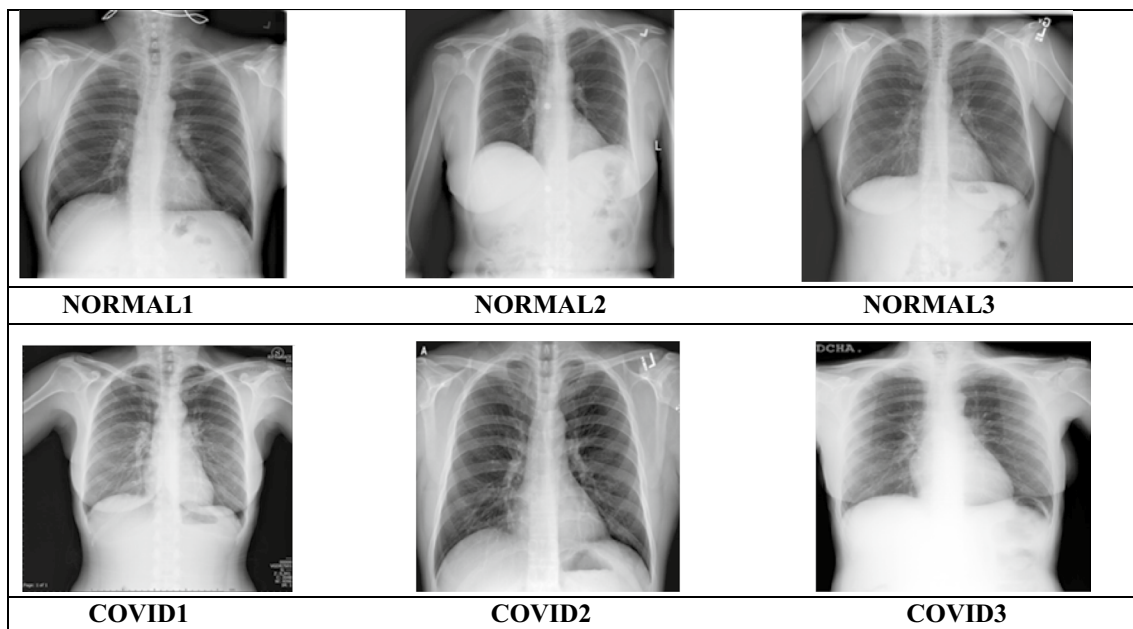


Fig. 8 Source images

$$P(t) = \alpha + \frac{x_{(i,j)} - M_{(x_i)}}{\text{best}_j(t) \times (\text{UB}_j - \text{LB}_j) + \epsilon}, \tag{14}$$

where  $M_{(x_i)}$ , as in Eq. (14), is the average positions of the  $i$ th solution, which is calculated using Eq. (15).  $\text{UB}(j)$  and  $\text{LB}(j)$  are the  $j$ th position's upper and lower limits, respectively. In this study, the sensitive parameter  $\alpha$ , which is fixed at 0.1, governs the exploration accuracy (the difference between candidate solutions) for the hunting cooperation during iterations.

$$M_{(x_i)} = \frac{1}{n} \sum_{j=1}^n x_{(i,j)}. \tag{15}$$

**3.2.4.4 Hunting of Cooperation Exploitation** The hunting cooperation strategy is performed when  $t \leq T$  and  $t > 3(T/4)$ , as in Eq. (16). Finally, the RSA is stopped when it meets the end criterion.

$$x_{(i,j)}(t + 1) = \text{best}_j(t) - \eta_{(i,j)}(t) \times \epsilon - R_{(i,j)}(t) \times \text{rand} \tag{16}$$

Hunting coordination and collaboration are exploitation search systems' attempts to avoid becoming caught in the local optima. These processes help the exploration search retain diversity among the potential solutions while identifying the best solution. To obtain a stochastic value at each iteration, they carefully defined two parameters (i.e., and) and continued research during the first and last iterations. This section is applicable when local optima become stagnant, especially in later iterations.

### 3.3 Salp Swarm Algorithm (SSA)

Marine animals come in over a million species, often categorized by a central database [57]. Most of these creatures share similar traits and habits, including moving, looking for food, and communicating. The Salp, a member of the Salpidae family, is one of these creatures. The Salp resembles a jellyfish in terms of shape. It has a cylindrical shape, a gelatinous body, and holes at the end that aid in pumping water. Figure 5a depicts the Salp's form when it feeds via

internal filters. A Salps chain, as seen in Fig. 5b, is a Salp grouping. This habit aids the Salps in improved foraging and mobility. Swarming is one of the characteristics that marine animals share, as was previously established.

The Salp algorithm is one of the meta-heuristic algorithms that has effectively resolved numerous optimization issues across numerous disciplines. Engineering design, image processing, wireless networks, machine learning, and other applications are some of the Salp algorithm's most significant application fields.

Many optimization problems can be solved using the basic SSA procedure. The population is necessary for the operation of this method. It was recommended in [57]. As previously indicated, this algorithm's behavior in looking for food may be successful because it relies on the swarm during the search phase. The swarm's common objective is to find food sources inside the search region.

Mathematically, a leader and followers make up the swarm [58]. The leader directs the actions of his followers. Figure 6 depicts the Salp Swarm Algorithm's flow and the fundamental steps in its functioning. Where  $X$  is a two-dimensional matrix that depicts a swarm of  $n$  Salps according to Eq. (17). Each Salp's fitness is calculated to determine which is the best (meaning, defines the leader for the swarm). Using Eq. (18), the leader position will be updated.

$$x_i = \begin{bmatrix} X_1^1 & X_1^1 & \dots & X_{1,d}^1 \\ X_1^2 & X_2^2 & \dots & X_d^2 \\ \vdots & \ddots & \ddots & \vdots \\ X_1^n & X_d^n & \dots & X_d^n \end{bmatrix} \tag{17}$$

$$X_i^1 = \begin{cases} y_i + ((ub_i - lb_i)r_2 + lb_i) & r_4 < 0.5 \\ y_i + ((ub_i - lb_i)r_2 + lb_i) & r_4 \geq 0.5 \end{cases}, \tag{18}$$

where  $y_i$  is the food site in the  $i$ th dimension and  $x_i^1$  represents the position of the leader (first) Salp in that dimension  $i$ th. Equation (19) determines the coefficient  $r_1$ , where  $lb_i$  and  $ub_i$  represent the dimensions' lower and upper bounds. Random values between [0, 1] make up  $r_2$  and  $r_3$ .

$$r_1 = e^{-\left(\frac{4t}{L}\right)^2} \tag{19}$$

$l$  is the running iteration, and  $L$  is the upper iteration. Because the balance of exploration and exploitation depends on it during the search process, the coefficient  $r_1$  is particularly important in SSA. Regarding the followers, Eq. (20) displays their updated positions:

$$x_i^j = \frac{1}{2} \lambda t^2 + \delta_0 t. \tag{20}$$

In the case when  $j > 2$ ,  $x_i^j$  refers to the location of the  $n$ th Salp in the  $i$ th dimension,  $\delta_0$  denotes an initial speed,  $t$  denotes

**Table 2** Parameters of the comparison algorithms

Algorithm	Parameters setting
SSA	$c_2$ and $c_3$ are random values [1, 0]
MPA	$FAD_S=0.2, P=0.5$
PSO	$wMax=0.9, wMin=0.2, c_1=2, c_2=2$
WOA	$a [2, 0], b=1, t [-1, 1]$
AO	$\text{alpha}=0.1, \text{delta}=0.1;$
RSA	$\beta=0.1, \alpha=0.005$

**Table 3** Comparison of PSNR for COVID, NORMAL cases images

	<i>K</i>		AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO	
COVID-1	2	MEAN	15.99636	13.3542	13.62139	15.14325	14.72392	14.52838	13.05388	
		STD	0.457807	2.879741	0.812893	0.749422	1.631453	1.673632	2.029729	
	3	MEAN	15.72325	14.98874	15.67748	15.71619	14.23023	13.94943	15.28932	
		STD	0.495808	0.921241	1.989558	1.363071	2.637373	1.214327	0.895679	
	4	MEAN	15.19492	16.37215	15.97784	15.04494	15.0158	15.72654	16.09853	
		STD	0.46831	0.722467	1.623999	1.678525	1.901651	1.567743	2.587032	
	5	MEAN	16.54481	15.90312	15.12889	16.76509	16.44464	15.85798	16.77462	
		STD	1.204432	0.038961	0.329087	1.181614	2.838713	2.173416	1.481966	
	6	MEAN	18.28938	19.67572	18.86296	19.66328	19.26783	19.64373	18.07998	
		STD	1.275444	0.94462	2.296606	0.666739	1.238636	1.033956	0.754216	
	COVID-2	2	MEAN	13.8528	10.73882	12.54894	13.69693	12.29714	13.4655	13.62295
			STD	1.496287	1.366289	0.476997	0.242637	1.038113	1.055034	1.208031
3		MEAN	14.80407	14.23211	14.14794	14.98948	15.20345	13.91394	13.86499	
		STD	1.583597	0.634901	2.904722	1.510022	1.901503	1.308217	1.412318	
4		MEAN	14.53891	16.79614	14.21837	16.70445	16.6797	16.73104	15.00825	
		STD	1.354547	1.131132	1.439741	0.90755	1.039551	1.303573	1.218771	
5		MEAN	16.165	16.43886	14.73026	16.19848	16.3406	16.94531	16.38499	
		STD	2.742593	2.081787	0.719713	1.894364	1.553394	0.760369	0.378739	
6		MEAN	17.21162	18.17566	17.41433	18.82411	18.58039	18.90498	16.0777	
		STD	1.564371	1.306123	1.135714	3.055779	1.001871	1.209981	3.618802	
COVID-3		2	MEAN	11.16933	10.83157	13.62884	13.2988	13.44292	13.65758	13.38808
			STD	2.423073	1.894036	1.82483	1.966076	0.399431	2.046013	2.060039
	3	MEAN	14.51449	14.58047	15.36989	14.33983	9.564635	13.0029	14.58103	
		STD	2.245033	0.743261	2.143962	3.068949	1.48891	1.104968	1.434026	
	4	MEAN	12.26771	13.33694	14.83319	16.27265	15.88782	14.39665	16.21477	
		STD	5.378315	1.03325	1.313469	0.990431	0.49478	3.322576	2.463693	
	5	MEAN	16.92287	16.55397	15.69802	18.9417	17.38048	18.89579	16.25714	
		STD	2.17281	0.595055	4.706241	2.841534	3.999957	0.458308	1.322994	
	6	MEAN	19.66963	17.27966	17.61387	19.93219	16.1862	17.99446	16.34047	
		STD	0.207749	2.820685	3.020165	3.267133	2.373307	0.647447	3.535097	
	NORMAL-1	2	MEAN	12.41414	13.20414	13.3123	13.63814	12.82645	13.45965	12.31845
			STD	1.680167	0.367283	0.446509	0.739487	1.176126	0.783467	0.975626
3		MEAN	12.8001	13.70984	13.49773	13.5575	12.57109	13.53202	13.6597	
		STD	0.488647	1.151328	1.183424	0.329	1.186902	0.254708	0.687869	
4		MEAN	15.56469	16.81564	15.85846	16.73248	16.22043	16.70589	15.2646	
		STD	1.06833	0.572113	0.25025	1.715065	1.773017	0.641843	1.405736	
5		MEAN	17.53325	16.5834	17.17805	17.52708	16.83676	15.74463	17.49363	
		STD	1.051604	0.62701	0.81215	1.218011	0.363412	2.034782	1.371984	
6		MEAN	17.66664	17.96638	17.39476	17.93091	16.7057	17.83272	16.94448	
		STD	0.420154	0.821384	0.978986	0.209366	4.475576	1.238946	0.542374	
NORMAL-2		2	MEAN	12.9072	10.79158	10.38172	12.13743	13.80757	12.8793	11.77823
			STD	1.361359	1.587784	2.577119	2.73055	1.030454	2.083797	1.065328
	3	MEAN	13.13795	14.5805	13.57049	15.49957	14.69812	15.50122	11.78228	
		STD	1.35674	0.882141	1.143955	1.740162	0.467485	1.854953	1.787442	
	4	MEAN	13.68826	15.74899	14.15722	14.53005	14.17531	13.91692	13.78059	
		STD	0.843746	0.798077	0.315345	1.40499	2.915623	3.24002	2.866384	
	5	MEAN	15.42665	15.33481	15.90974	16.49487	15.7174	14.87574	15.65947	
		STD	2.277014	0.440318	1.399815	1.471268	1.511312	0.533327	0.369938	
	6	MEAN	17.83979	17.92491	17.5945	17.69739	16.11937	13.98299	14.5756	
		STD	1.339437	0.432668	2.802622	0.455714	2.988868	3.624765	2.384023	

**Table 3** (continued)

	<i>K</i>		AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
NORMAL-3	2	MEAN	14.42355	12.84311	14.35452	14.14685	12.93748	12.12876	14.5101
		STD	0.738135	0.794227	0.287968	0.672658	1.209149	2.750113	0.46632
	3	MEAN	15.42658	12.89072	15.01992	15.64615	14.82943	15.18174	14.93892
		STD	0.860717	0.887621	2.062571	1.24102	0.906443	0.800181	1.654095
	4	MEAN	17.04607	17.28479	16.01375	16.50957	16.10153	16.86906	15.16286
		STD	2.543886	0.364581	1.267257	1.242821	1.077229	1.999883	2.028429
	5	MEAN	17.05958	17.4702	18.02356	18.8066	17.19482	18.76523	18.93854
		STD	0.848082	0.970442	1.067506	1.323133	1.095208	1.032608	2.1991
	6	MEAN	16.56732	18.3278	17.29078	18.43022	17.67516	18.47673	18.52612
		STD	0.761405	1.595749	0.4339	0.695685	1.428555	0.030226	1.08701
Mean			15.02604	15.03353	15.16173	15.98876	15.09539	15.26292	15.09392
Ranking			7	6	3	1	4	2	5

the passage of time, and  $= \frac{\delta_{final}}{\delta_0}$ , where  $\delta_0 = \frac{x-x_0}{t}$ . The duration of optimization indicates iteration. As a result, there is a 1 in the difference between iterations. The following equation is used to solve this problem while keeping in mind the idea that  $\delta_0 = 0$ .

$$x_i^j = \frac{1}{2}(x_i^j + x_i^{j-1}) \tag{21}$$

in when  $j > 2$ . When the Salp departs from the search space, we can sometimes bring them back by applying Eq. (21).

$$x_i^j = \begin{cases} l^i & \text{if } x_i^j \leq l^i \\ u^j & \text{if } x_i^j \geq u^j \\ x_i^j & \text{if } x_i^j \leq l^i \end{cases} \tag{22}$$

In optimization algorithms, terms like exploration and exploitation, diversity and intensification, international and local searches, and a few word pairs are frequently used. One of those pairs is at least present in the optimization algorithms. Exploration and exploitation will be covered in this section. In the search area, exploration seeks to find uncharted territory.

Exploitation's goal is to improve on successful and previously known solutions. Due to the search process not stopping at one level, the level of the ideal solution can be found. The  $c1$  coefficient is in charge of maintaining a balance between exploration and exploitation. It is calculated using the formula Eq. (21).

### 3.4 The Proposed Hybrid Algorithm (RSA-SSA)

The study presents a hybrid approach for segmenting images. Based on the RSA and SSA algorithms, the suggested algorithm, dubbed RSA-SSA, uses Otsu's technique as its objective function. The ideal multi-level thresholding values that optimize Otsu's goal function are found using the proposed hybrid technique. The suggested hybrid technique depends on SSA use to improve RSA performance.

The RSA narrows the search region by selecting the optimal solution. Next, the SSA optimizes each agent less than the restricted base. Consequently, after calculating an input image's histogram, a random population of solutions is generated (threshold values). The RSA then refreshes this population using levels that are represented by four different motions. Then, using Otsu's method, the best solution is picked from the population (the best solutions from the RSA algorithm).

After determining the globally optimal solution (SSA algorithm's output), all preceding steps are repeated until the halting conditions are satisfied. The SSA uses the result from the RSA (worst solution) to start calculating the minimal thresholding value, and the population's solutions are then optimized using the method covered in Sect. 3.3.2. Figure 7 depicts the RSA-concluding SSA's procedures.

## 4 Experiments and Results

In this section, the experiment is used to assess the performance of the RSA-SSA ability for segmentation NORMAL and COVID-19. Comparable algorithms include the Marine

**Table 4** SSIM of COVID and NORMAL cases images

	<i>K</i>		AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO	
COVID-1	2	MEAN	0.42965	0.40297	0.376196	0.45871	0.40297	0.444653	0.460506	
		STD	0.03405	0.05994	0.03036	0.01114	0.059947	0.023983	0.024026	
	3	MEAN	0.49806	0.44377	0.53224	0.41162	0.443777	0.395412	0.467073	
		STD	0.00533	0.0407	0.07087	0.10924	0.040703	0.04514	0.015658	
	4	MEAN	0.46394	0.50754	0.53656	0.46277	0.507542	0.475494	0.533161	
		STD	0.0213	0.01404	0.03246	0.0421	0.014043	0.021184	0.05675	
	5	MEAN	0.5298	0.5217	0.5346	0.50467	0.52172	0.561966	0.555674	
		STD	0.03282	0.04483	0.02549	0.07967	0.044831	0.048949	0.046626	
	6	MEAN	0.54274	0.58835	0.55735	0.62441	0.588358	0.576641	0.591855	
		STD	0.05497	0.03155	0.0859	0.091	0.0314	0.052018	0.019438	
	COVID-2	2	MEAN	0.50924	0.42224	0.42558	0.37087	0.422248	0.360834	0.439632
			STD	0.01713	0.06783	0.05705	0.09721	0.067832	0.134443	0.073197
3		MEAN	0.48167	0.41479	0.51593	0.45614	0.414796	0.460597	0.418115	
		STD	0.08559	0.0175	0.07921	0.08515	0.01758	0.089635	0.056822	
4		MEAN	0.52968	0.54112	0.49385	0.52068	0.541124	0.546894	0.480378	
		STD	0.12498	0.03782	0.1051	0.01472	0.037822	0.017256	0.063804	
5		MEAN	0.58313	0.6066	0.55032	0.62619	0.60667	0.586562	0.535726	
		STD	0.08215	0.02334	0.08076	0.00756	0.023342	0.004549	0.052696	
6		MEAN	0.57667	0.62786	0.56212	0.62452	0.627816	0.646609	0.597459	
		STD	0.00633	0.02681	0.08871	0.02341	0.026981	0.032317	0.066562	
COVID-3		2	MEAN	0.373967	0.339013	0.495861	0.512224	0.339003	0.56946	0.450022
			STD	0.142947	0.091252	0.080724	0.016876	0.091282	0.062944	0.080966
	3	MEAN	0.530732	0.551748	0.610586	0.280487	0.551768	0.455782	0.558857	
		STD	0.125136	0.022145	0.029913	0.08828	0.022165	0.027358	0.124729	
	4	MEAN	0.437222	0.465512	0.523524	0.601395	0.465522	0.531071	0.585213	
		STD	0.324238	0.031438	0.049375	0.029156	0.031428	0.156657	0.104579	
	5	MEAN	0.636554	0.589412	0.519042	0.563354	0.589412	0.672963	0.601557	
		STD	0.062811	0.010422	0.172	0.081168	0.010432	0.027416	0.023636	
	6	MEAN	0.688038	0.602256	0.63167	0.552424	0.602256	0.635441	0.565898	
		STD	0.022429	0.080002	0.073866	0.096402	0.080002	0.03781	0.100983	
	NORMAL-1	2	MEAN	0.377072	0.433945	0.425974	0.448437	0.60622	0.439367	0.45975
			STD	0.146139	0.084738	0.05663	0.082113	0.059084	0.072743	0.040375
3		MEAN	0.487931	0.495169	0.466183	0.461924	0.675019	0.350841	0.485511	
		STD	0.052161	0.090261	0.09588	0.028882	0.031402	0.186978	0.069474	
4		MEAN	0.449308	0.497662	0.500883	0.494075	0.63306	0.43292	0.405417	
		STD	0.069547	0.053722	0.040597	0.13073	0.03235	0.074601	0.032333	
5		MEAN	0.622331	0.586756	0.527278	0.454854	0.694551	0.575692	0.406074	
		STD	0.014378	0.043584	0.078913	0.04693	0.054632	0.068585	0.225934	
6		MEAN	0.521997	0.582859	0.549776	0.506106	0.661963	0.426425	0.46821	
		STD	0.060609	0.03107	0.088384	0.125498	0.077946	0.160948	0.194706	
NORMAL-2		2	MEAN	0.440137	0.443875	0.448085	0.380422	0.433945	0.425825	0.361047
			STD	0.108188	0.064598	0.031175	0.042702	0.084738	0.041938	0.024426
	3	MEAN	0.478387	0.499265	0.441933	0.506195	0.495169	0.518201	0.507956	
		STD	0.071283	0.042977	0.040663	0.03042	0.090261	0.041512	0.050541	
	4	MEAN	0.457856	0.585825	0.600818	0.497838	0.497662	0.574744	0.507555	
		STD	0.029951	0.044236	0.048041	0.062741	0.053722	0.017915	0.06425	
	5	MEAN	0.637749	0.590795	0.571383	0.554001	0.586756	0.58265	0.578656	
		STD	0.010376	0.081043	0.081152	0.04878	0.043584	0.044166	0.042985	
	6	MEAN	0.620989	0.610829	0.657615	0.554396	0.582859	0.579489	0.577239	
		STD	0.080172	0.021174	0.094464	0.080524	0.03107	0.042093	0.075686	

**Table 4** (continued)

	<i>K</i>		AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
NORMAL-3	2	MEAN	0.579632	0.525593	0.470218	0.496973	0.443875	0.528933	0.55688
		STD	0.067865	0.084187	0.209488	0.121069	0.064598	0.144974	0.043189
3	MEAN	0.659135	0.637236	0.659628	0.671445	0.499265	0.578848	0.623327	
	STD	0.043666	0.017713	0.0457	0.090691	0.042977	0.125382	0.065171	
4	MEAN	0.727387	0.723672	0.70059	0.724118	0.585825	0.660755	0.688485	
	STD	0.051295	0.035443	0.012628	0.049346	0.044236	0.089128	0.021552	
5	MEAN	0.671559	0.666467	0.750507	0.702174	0.590795	0.701738	0.693761	
	STD	0.088484	0.039542	0.046332	0.044752	0.081043	0.059367	0.068361	
6	MEAN	0.746381	0.745939	0.749069	0.726662	0.610829	0.750085	0.745973	
	STD	0.022251	0.073073	0.061598	0.055838	0.021174	0.024152	0.016091	

**Table 5** Friedman ranking test for the methods of PSNR for COVID-19 experiment

PSNR_rank	<i>K</i>	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
COVID1	2	1	5	6	2	3	7	4
	3	1	3	5	2	4	6	7
	4	4	7	2	5	1	3	6
	5	3	5	6	2	1	4	7
	6	7	1	4	2	5	3	6
	SUM	14	15	23	13	14	19	21
	Mean rank	2.8	3	4.6	2.6	2.8	3.8	4.2
Final rank	2	3	6	1	2	4	5	
COVID2	2	1	5	7	2	4	6	3
	3	2	5	4	1	6	3	7
	4	3	7	6	2	4	1	5
	5	3	7	6	4	2	1	5
	6	6	3	5	2	1	7	4
	SUM	15	19	25	11	17	12	23
	Mean rank	3	3.8	5	2.2	3.4	2.4	4.6
Final rank	3	5	7	1	4	2	6	
COVID3	2	5	7	2	4	6	1	3
	3	4	3	1	5	6	7	2
	4	6	5	4	1	2	3	7
	5	7	5	6	1	4	3	2
	6	1	5	4	2	7	3	6
	SUM	20	24	17	13	25	16	15
	Mean rank	4	4.8	3.4	2.6	5	3.2	3
Final rank	5	6	4	1	7	3	2	

**Table 6** Summary of Friedman ranking test for the methods of PSNR for the COVID-19 experiment

PSNR_rank	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
COVID1	2	3	6	1	2	4	5
COVID2	3	5	7	1	4	2	6
COVID3	5	6	4	1	7	3	2

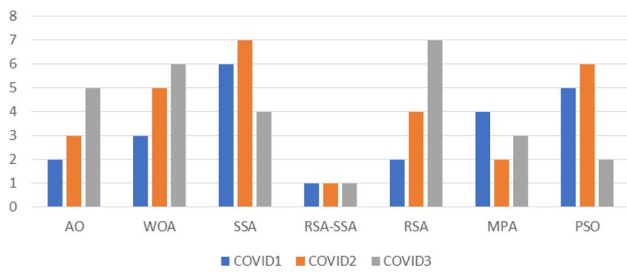


Fig. 9 Friedman ranking in term PSNR chart (COVID-19)

Predators Algorithm (MPA) [47], Salp Swarm Algorithm (SSA) [57], Whale Optimization Algorithm (WOA) [59], Particle Swarm Optimization Algorithm (PSO) [60], Aquila Optimization Algorithm (AO) [61], and the Reptile Search Algorithm (RSA) [55]. These algorithms demonstrated their

effectiveness as image segmentation optimization techniques in the literature.

### 4.1 Image Dataset

The trials were carried out on six photographs of well-known test images for a better comparison of the segmentation results, as shown in Fig. 8. Each image with a size of (256 × 256). Five thresholds (2–6) were used in this research's multi-level image segmentation procedure. Optimization systems using swarm intelligence usually display random behavior. Each experiment was carried out 100 times for each image and level in three times run.

In this study, the algorithms will be tested using the dataset's standard gray images of (COVID-19, NORMAL) X-rays, which were released on the link on January 4, 2022: <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database>

Table 7 Friedman ranking test for the methods of PSNR for NORMAL experiment

PSNR_rank	K	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
NORMAL1	2	5	7	4	1	6	3	2
	3	5	1	4	3	2	6	7
	4	7	1	3	2	4	5	6
	5	1	4	2	3	5	7	6
	6	2	7	3	4	1	6	5
	SUM	17	13	16	13	18	25	21
	Mean rank	3.4	2.6	3.2	2.6	3.6	5	4.2
Final rank	3	1	2	1	4	6	5	
NORMAL2	2	1	7	6	3	5	2	4
	3	3	6	5	2	1	7	4
	4	7	1	2	4	6	5	3
	5	3	4	6	1	2	5	7
	6	2	1	7	3	5	6	4
	SUM	15	17	24	13	18	19	17
	Mean rank	3	3.4	4.8	2.6	3.6	3.8	3.4
Final rank	2	3	6	1	4	5	3	
NORMAL3	2	2	5	6	3	1	4	7
	3	3	7	2	1	4	5	6
	4	2	1	6	4	5	3	7
	5	4	3	7	2	6	5	1
	6	7	4	6	3	5	2	1
	SUM	16	17	26	13	21	19	14
	Mean rank	3.2	3.4	5.2	2.6	4.2	3.8	2.8
Final rank	3	4	7	1	6	5	2	

Table 8 Summary of Friedman ranking test for the methods of PSNR for the NORMAL experiment

PSNR_rank	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
NORMAL1	3	1	2	1	4	6	5
NORMAL2	2	3	6	1	4	5	3
NORMAL3	3	4	7	1	6	5	2



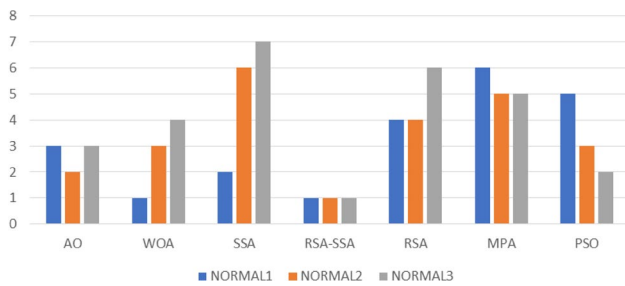


Fig. 10 Friedman ranking in term PSNR chart (NORMAL)

### 4.2 Segmented Image Quality Measures

The performance of the algorithms will be compared using different metrics, including PSNR, SSIM, and Friedman ranking test.

1. The peak signal-to-noise ratio (PSNR) measure is used to measure the variance between the reference image and segmented image [62], and it depends on the value of intensity in:

$$PSNR = 20 \log_{10} \left( \frac{255}{RMSE} \right), \text{ (indB)}. \tag{22}$$

RMSE refers to the root-mean-squared error detected as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^M \sum_{j=1}^Q (I(i,j) - Seg(i,j))^2}{M * Q}}. \tag{23}$$

The terms  $I$  and  $Seg$  denote the original and segmented images, respectively, of sizes  $M$  and  $Q$ . The highest  $PSNR$  value represents the highest level of performance for the segmentation algorithms.

Table 9 Friedman ranking test for the methods of SSIM for the COVID-19 experiment

SSIM_rank	K	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
COVID1	2	4	5	6	1	3	7	2
	3	2	4	1	3	5	6	7
	4	6	7	1	4	3	5	2
	5	5	7	4	1	6	2	3
	6	7	2	6	4	5	3	1
	SUM	22	20	18	13	22	19	13
	Mean rank	4.4	4	3.6	2.6	4.4	3.8	2.6
Final rank	5	4	2	1	5	3	1	
COVID2	2	1	5	4	3	6	7	2
	3	3	7	2	1	4	5	6
	4	4	2	5	1	6	3	7
	5	7	1	5	4	2	3	6
	6	4	3	7	1	5	2	6
	SUM	14	15	21	10	23	19	26
	Mean rank	2.8	3	4.2	2	4.6	3.8	5.2
Final rank	2	3	5	1	6	4	7	
COVID3	2	4	6	7	5	2	1	3
	3	7	3	1	5	4	6	2
	4	6	7	4	1	5	3	2
	5	3	5	6	1	4	2	7
	6	2	7	4	1	4	3	5
	SUM	19	23	17	13	19	15	16
	Mean rank	3.8	4.6	3.4	2.6	3.8	3	3.2
Final rank	5	6	4	1	5	2	3	

Table 10 Summary of Friedman ranking test for the methods of SSIM for the COVID-19 experiment

SSIM_rank	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
COVID1	5	4	2	1	5	3	1
COVID2	2	3	5	1	6	4	7
COVID3	5	6	4	1	5	2	3

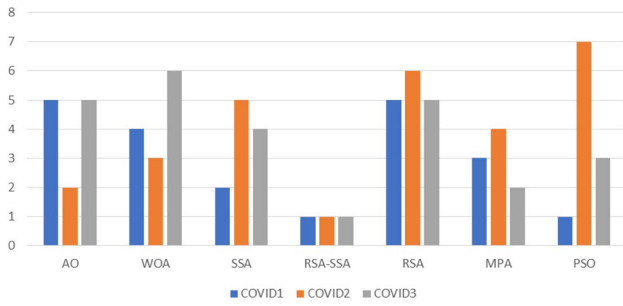


Fig. 11 Friedman ranking in term SSIM chart (COVID-19)

2. The structural similarity index (SSIM): is used to assess how comparable ( $I$ ) and (Seg) images are [63], and is defined as:

$$SSMI(I, Seg) = \frac{(2\mu_I\mu_{Seg} + C_1)(2\sigma_{I,Seg} + C_2)}{(\mu_I^2 + \mu_{Seg}^2 + C_1)(\sigma_I^2 + \sigma_{Seg}^2 + C_2)} \quad (24)$$

$\mu_I$  and  $\mu_{Seg}$  denote the mean intensity of ( $I$ ) and ( $Seg$ ) and the standard deviation of ( $I$ ) and ( $Seg$ ), respectively.  $\sigma_I$ ,  $Seg$  also denotes the variance of ( $I$ ) and ( $Seg$ ). Constants  $c_1$  and  $c_2$  have values of 6.5025 and 58.52252, respectively. The highest SSIM value indicates better performance.

3. Friedman ranking test: the one-way ANOVA with repeated measures is a non-parametric alternative to the Friedman ranking test. When the dependent variable being measured is ordinal, it is utilized to examine changes across groups. Additionally, it can be applied to continuous data that deviates from the presumptions required to conduct a one-way ANOVA with repeated measures [43].

Table 11 Friedman ranking test for the methods of SSIM for NORMAL experiment

SSIM_rank	K	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
NORMAL1	2	6	4	5	2	1	3	7
	3	7	2	5	1	3	6	4
	4	4	3	2	1	6	5	7
	5	1	2	7	5	4	3	6
	6	4	2	3	1	6	7	5
	SUM	18	13	19	10	20	23	22
	Mean rank	3.6	2.6	3.8	2	4	4.6	4.4
Final rank	3	2	4	1	5	7	6	
NORMAL2	2	7	2	1	4	3	5	6
	3	5	4	6	1	2	7	3
	4	6	7	1	2	3	4	5
	5	1	3	6	2	5	4	7
	6	2	3	1	4	7	5	6
	SUM	17	15	15	13	20	20	25
	Mean rank	3.4	3	3	2.6	4	4	5
Final rank	3	2	2	1	4	4	5	
NORMAL3	2	2	5	6	1	3	4	7
	3	2	7	1	3	4	6	5
	4	7	3	4	1	2	6	5
	5	5	6	2	1	3	7	4
	6	4	5	7	2	3	1	6
	SUM	15	23	16	8	15	20	23
	Mean rank	3	4.6	3.2	1.6	3	4	4.6
Final rank	2	5	3	1	2	4	5	

Table 12 Summary of Friedman ranking test for the methods of SSIM for the NORMAL experiment

SSIM_rank	AO	WOA	SSA	RSA-SSA	RSA	MPA	PSO
NORMAL1	3	2	4	1	5	7	6
NORMAL2	3	2	2	1	4	4	5
NORMAL3	2	5	3	1	2	4	5

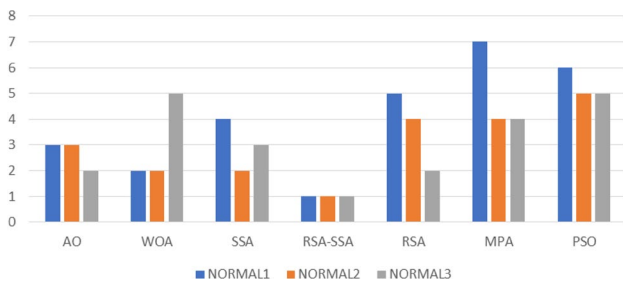


Fig. 12 Friedman ranking in term SSIM chart (NORMAL)

### 4.3 Experiment Setup

The comparison of six different algorithms, the Marine Predators Algorithm (MPA), Salp Swarm Algorithm (SSA), Whale Optimization Algorithm (WOA), Particle Swarm Optimization (PSO), Aquila Optimizer (AO), and the novel of the RSA. The studies were carried out on six well-known test photos, as shown in Fig. 8. To compare the segmentation results more effectively, those that are used in the research have the same cessation conditions (maximum iterations set to 100), with a total run of 10 per algorithm, and the size

of the population "set to (25). The parameters of all algorithms are shown in Table 2. For performance evaluation of all experiments on the test, images are done by the number of thresholds": 2, 3, 4, 5 "and 6 as in [37, 64, 65]. The selection for such thresholds was to illustrate the algorithm's performance (RSA-SSA) compared with the traditional segmentation algorithms based on the swarm. All algorithms are programmed in Matlab2017 and implemented in the "Windows 10—64bit" environment on a computer with Intel Corei7 10th Gen (1.99 GHz) processor" 1T" memory and SSD (128).

### 4.4 Benchmark Images

To test each method, this study uses six photos—three COVID-19 case images and three NORMAL case images.

The dataset used to create these photos was made available on the website <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database> on April 9, 2021. Figure 8 shows these images have NORMAL1, NORMAL2, NORMAL3, COVID1, COVID2, and COVID3. All have a dimension of 256 × 256 and the PNG extension.

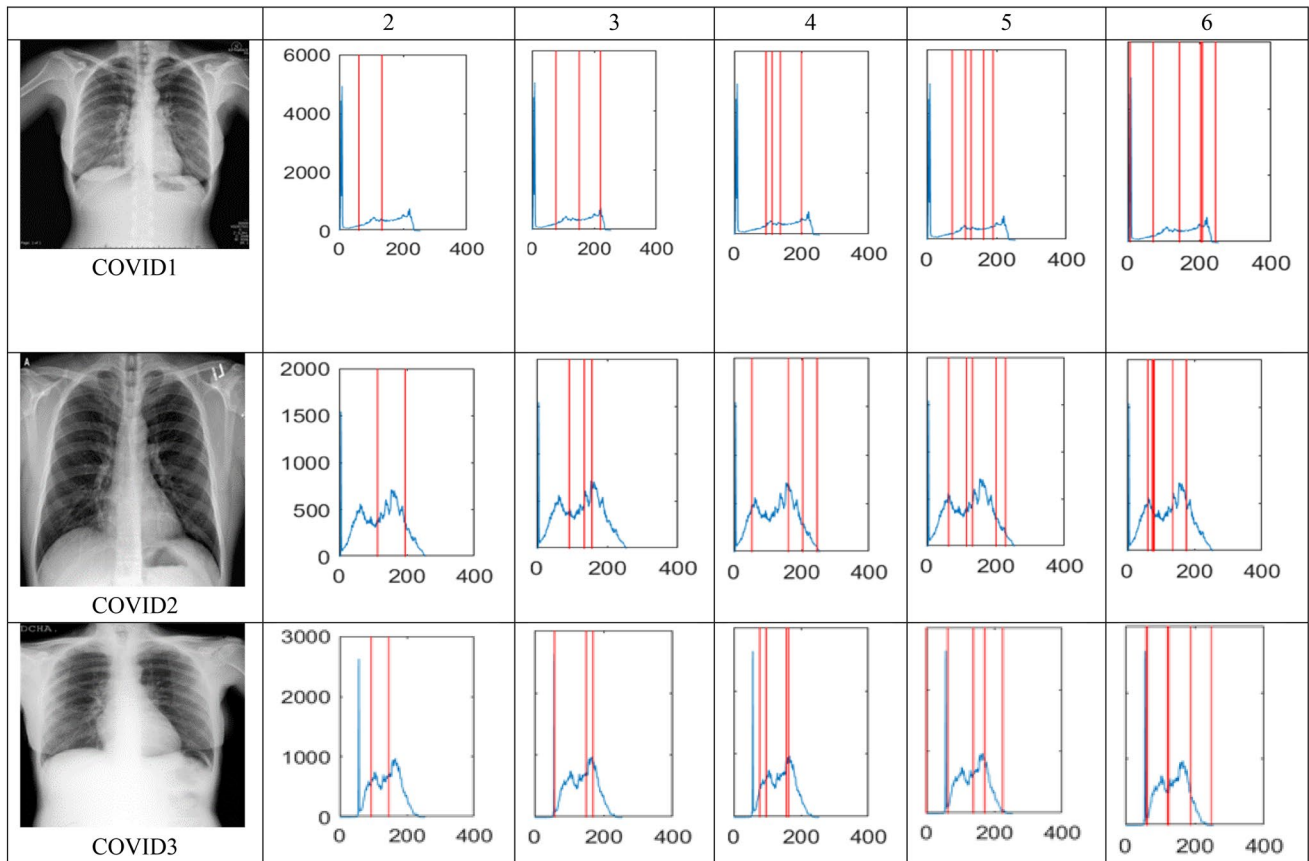


Fig. 13 Histogram for COVID images experiments for proposed technique in threshold 2,3,4,5,6

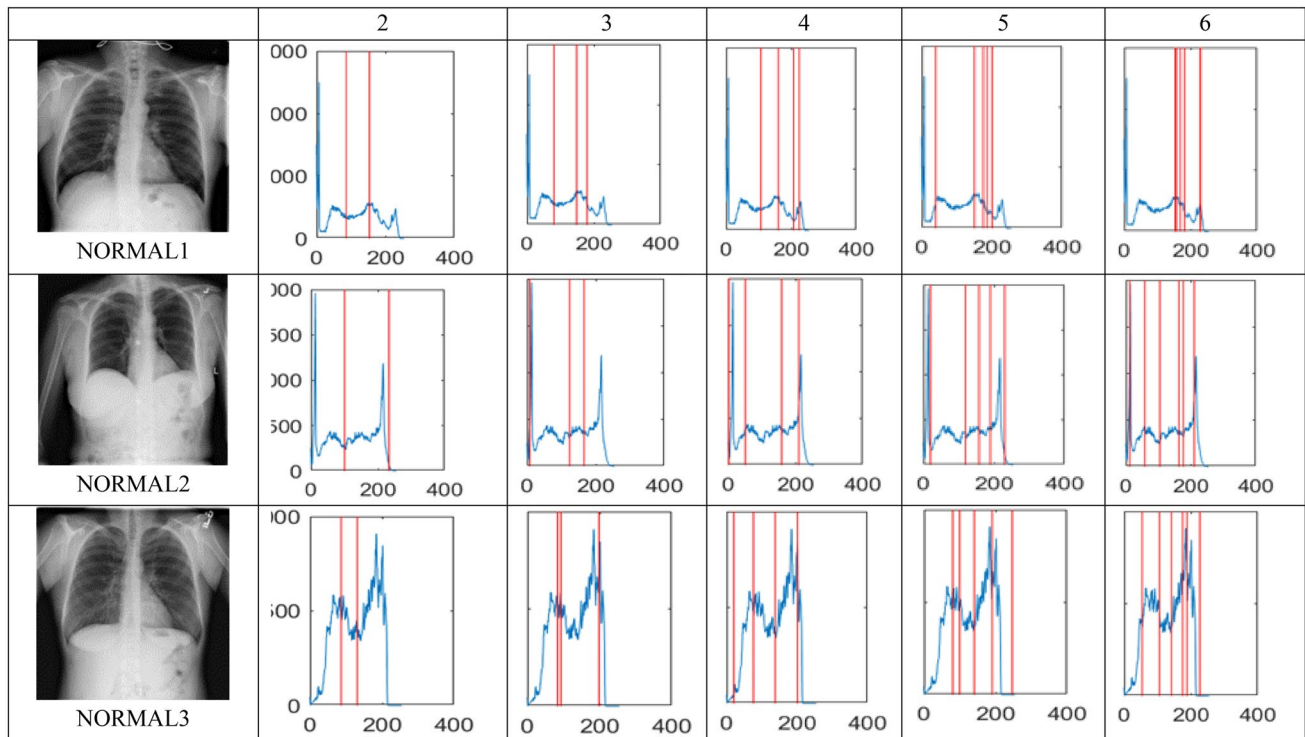


Fig. 14 Histogram for NORMAL images experiments for proposed technique in threshold 2,3,4,5,6

### 4.5 The Results and Discussions

In our trials, the objective function that was maximized based on the WOA, MPA, SSA, AO, RSA-SSA, and PSO algorithm chest X-ray Images for COVID-19 Cases was Otsu's technique (between-class variance). Six test photos were used to test the approach. They were thinking about the thresholds of 2, 3, 4, 5, and 6. For the WOA, MPA, SSA, AO, RSA-SSA, and PSO algorithms, the PSNR, SSIM, and Friedman ranking test values are provided. The results of the comparing approaches are provided in terms of PSNR using the Otsu function, as seen in Table 3. The findings unmistakably demonstrated that the proposed method outperformed all comparison methodologies, which speaks to the proposed method's capacity to accurately segment the images and select the best

threshold values. The proposed method came out on top in the Friedman ranking test, followed by MPA, WOA, SSA, AO, and PSO. The results of the comparison approaches are expressed in terms of SSIM using the Otsu function, as seen in Table 4. The findings showed that the suggested technique outperformed other comparison methods, demonstrating its searchability in segmenting the tested images and choosing the suitable threshold values to produce precise segmented images. The proposed method came out on top in the Friedman ranking test, followed by the SSA, AO, WOA, MPA, and PSO.

Table 5 shows each algorithm's degree at each threshold level and image. Degree 1 means the method has the best degree and more accurate performance of PSNR measure of COVID-19 cases images.

We can summarize the results of Table 5 as in Table 6. Table 6 shows the levels for all algorithms in Table 5 sorted

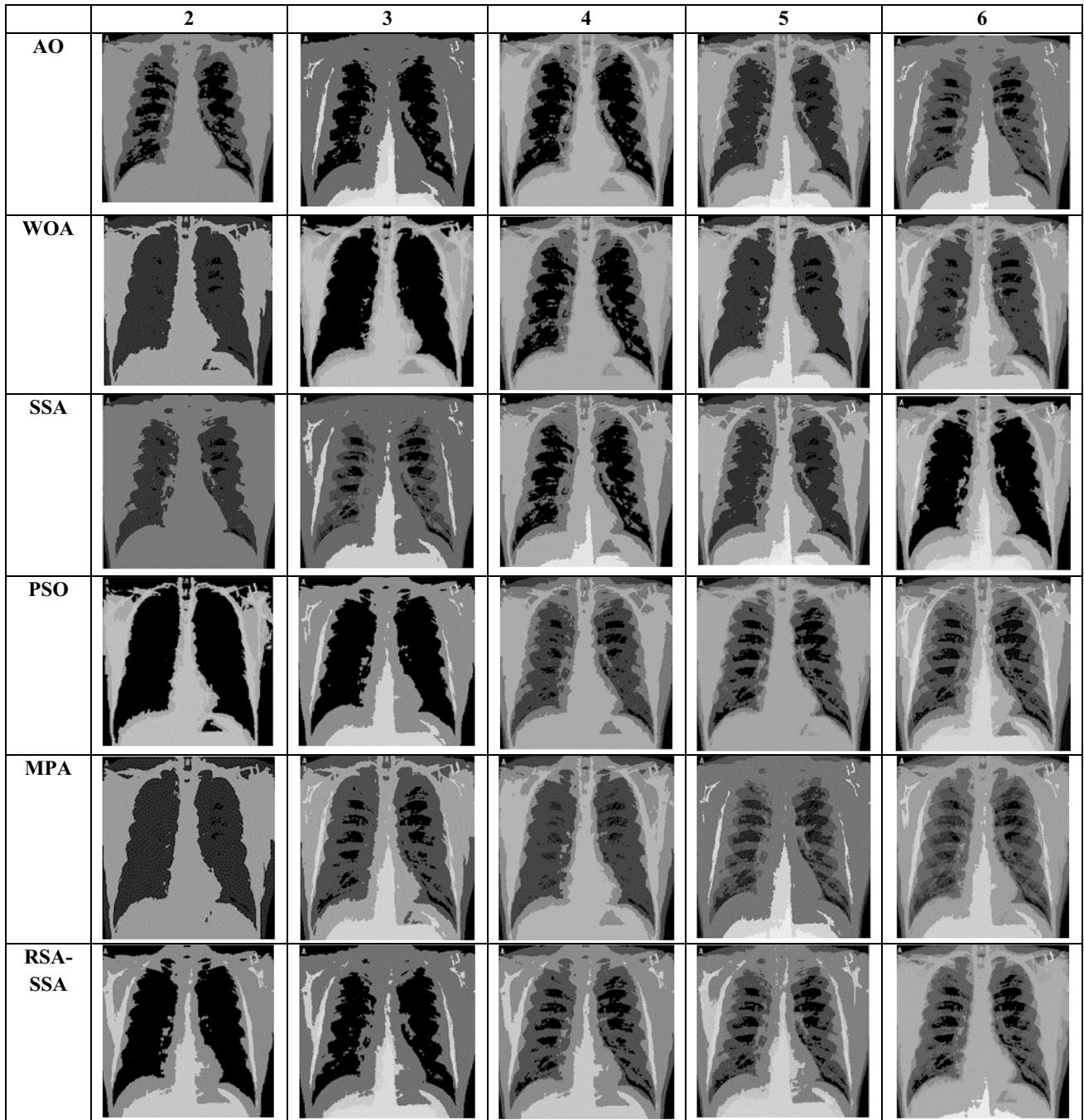


Fig. 15 COVID2 segment


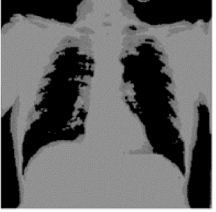





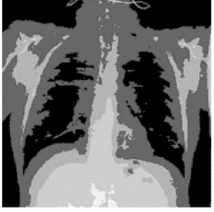

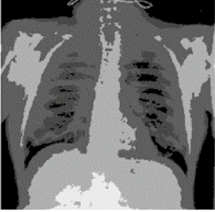


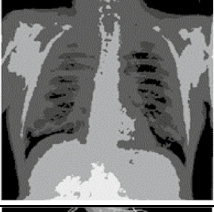
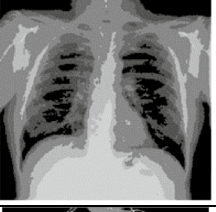
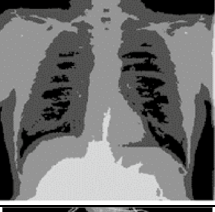
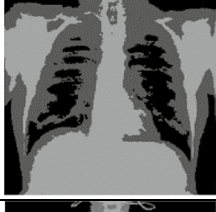
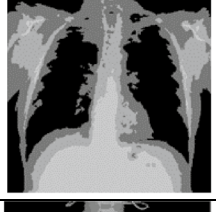
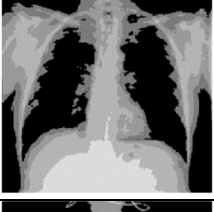

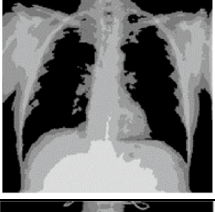

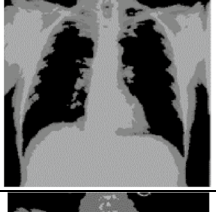


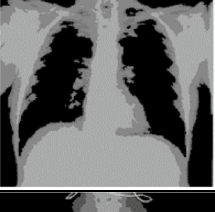





	2	3	4	5	6
AO					
WOA					
SSA					
PSO					
MPA					
RSA-SSA					

Fig. 16 NORMAL1 segment

depending on the final rank of the Friedman ranking test of PSNR measure of COVID-19 cases images, as shown in Fig. 9.

This table shows the degree for each algorithm at each threshold level and each image. Degree 1 means the method has the best degree and more accurate performance of PSNR measure of NORMAL cases images.

We can summarize the results of Table 7 as in Table 8. Table 8 shows the levels for all algorithms in Table 7 sorted depending on the final rank of the Friedman ranking test of PSNR measure of NORMAL cases images, as shown in Fig. 10.

We can summarize the results of Table 9 as in Table 10. Table 10 shows the levels for all algorithms in Table 9 sorted depending on the final rank of the Friedman ranking test of SSIM measure of COVID-19 cases images, as shown in Fig. 11.

This table shows the degree for each algorithm at each threshold level and each image. Degree 1 means the method has the best and more accurate performance of SSIM measure of NORMAL cases images.

We can summarize the results of Table 11 as in Table 12. Table 12 shows the levels for all algorithms in Table 11 sorted depending on the final rank of the Friedman ranking test of SSIM measure of NORMAL cases images, as shown in Fig. 12.

The discussion depends on each measure given in this section, and Figs. 13 and 14 show the image's histogram for the algorithm (RSA-SSA) at  $k$  equal to 2, 3, 4, 5, and 6. Figures 15 and 16 show the compression techniques used in different thresholds.

The limitations of the proposed method can be stated as follows: the tested images can be changed to other image classes, the parameters of the used methods can be investigated to get the best results and configuration, and the proposed method can be investigated further to validate its performance to solve other optimization problems.

## 5 Conclusion and Future Work

Identifying the optimal or best threshold for image segmentation is a challenge in this study's scenario of multi-layer thresholding. The term optimization difficulty has been used to describe this issue. As a fitness function, Otsu's function has been used. Consequently, a multi-level threshold image segmentation RSA-SSA technique was proposed. This issue has been resolved using the suggested algorithm. Its goal is to establish the ideal threshold value that improves Otsu's function. The algorithm's outcomes have been contrasted with those of the MPA, SSA, WOA, AO, and PSO algorithms. The following metrics have been used to assess algorithm performance:

PSNR, SSIM, and Friedman ranking test. We used six benchmark images for COVID-19 and NORMAL cases in all experiments. The study finds that the RSA-SSA suggested approach performs well during the segmentation. This superiority is mainly because it avoids becoming stuck at a local optimum. All of this suggests that this method has a strong capacity to locate the image's ideal solution. The Future Work advocated using the technique to address various issues and applications involving image processing, including (visualization, computer vision, computer-aided diagnostics, image classification, etc.). The finesses of the images are a very significant difficulty in image segmentation in all of these applications. Low resolutions, excessive noise, or poor contrast may bring on this issue. Pre-processing the photos is essential to increase the quality of that situation and achieve the best segmentation results.

**Data Availability Statements** Data is available from the authors upon reasonable request.

## Declarations

**Conflict of interest** The authors declare that there is no conflict of interest regarding the publication of this paper.

**Ethical approval** This article does not contain any studies with human participants or animals performed by any of the authors.

**Informed consent** Informed consent was obtained from all individual participants included in the study.

## References

1. Bhanu, B., Lee, S., & Ming, J. (1995). Adaptive image segmentation using a genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(12), 1543–1567.
2. Otair, M., Hasan, O. A., & Abualigah, L. (2022). The effect of using minimum decreasing technique on enhancing the quality of lossy compressed images. *Multimedia Tools and Applications*, 1–32.
3. Abualigah, L., Almotairi, K.H., & Elaziz, M. A. (2022). Multi-level thresholding image segmentation using meta-heuristic optimization algorithms: Comparative analysis, open challenges and new trends. *Applied Intelligence*, 1–51.
4. Kaur, D., & Kaur, Y. (2014). Various image segmentation techniques: A review. *International Journal of Computer Science and Mobile Computing*, 3(5), 809–814.
5. Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
6. Ibrahim, A., Franz, B., Ahmad, Z., Healy, R., Knobelspiesse, K., Gao, B.-C., Proctor, C., & Zhai, P.-W. (2018). Atmospheric correction for hyperspectral ocean color retrieval with application to the Hyperspectral Imager for the Coastal Ocean (HICO). *Remote Sensing of Environment*, 204, 60–75.

7. Abualigah, L., Al-Okbi, N. K., Elaziz, M. A., & Houssein, E. H. (2022). Boosting Marine Predators Algorithm by Salp Swarm Algorithm for multilevel thresholding image segmentation. *Multimedia Tools and Applications*, 81(12), 16707–16742.
8. Yin, P.-Y., & Chen, L.-H. (1997). A fast iterative scheme for multilevel thresholding methods. *Signal Processing*, 60(3), 305–313.
9. Horng, M.-H. (2011). Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation. *Expert Systems with Applications*, 38(11), 13785–13791.
10. Abualigah, L., Al-Okbi, N. K. (2022). A comparative analysis using multilevel thresholding image segmentation problems. In *Handbook of moth-flame optimization algorithm: Variants, hybrids, improvements, and applications* (p. 241).
11. Liu, Q., Li, N., Jia, H., Qi, Q., & Abualigah, L. (2022). Modified remora optimization algorithm for global optimization and multilevel thresholding image segmentation. *Mathematics*, 10(7), 1014.
12. Lin, S., Jia, H., Abualigah, L., & Altalhi, M. (2021). Enhanced slime mould algorithm for multilevel thresholding image segmentation using entropy measures. *Entropy*, 23(12), 1700.
13. Chakraborty, S., Sharma, S., Saha, A. K., & Saha, A. (2022). A novel improved whale optimization algorithm to solve numerical optimization and real-world applications. *Artificial Intelligence Review*, 1–112.
14. Sharma, S., Saha, A. K., Majumder, A., & Nama, S. (2021). MPBOA-A novel hybrid butterfly optimization algorithm with symbiosis organisms search for global optimization and image segmentation. *Multimedia Tools and Applications*, 80(8), 12035–12076.
15. Abualigah, L., Diabat, A., Sumari, P., & Gandomi, A. H. (2021). A novel evolutionary arithmetic optimization algorithm for multilevel thresholding segmentation of covid-19 ct images. *Processes*, 9(7), 1155.
16. Houssein, E. H., Hussain, K., Abualigah, L., Abd Elaziz, M., Alomoush, W., Dhiman, G., Djenouri, Y., & Cuevas, E. (2021). An improved opposition-based marine predators algorithm for global optimization and multilevel thresholding image segmentation. *Knowledge-Based Systems*, 229, 107348.
17. Ye, Z., Yang, J., Wang, M., Zong, X., Yan, L., & Liu, W. (2018). 2D Tsallis entropy for image segmentation based on modified chaotic bat algorithm. *Entropy*, 20(4), 239.
18. Chakraborty, S., Saha, A. K., Nama, S., & Debnath, S. (2021). COVID-19 X-ray image segmentation by modified whale optimization algorithm with population reduction. *Computers in Biology and Medicine*, 139, 104984.
19. Jia, H., Xing, Z., & Song, W. (2019). Three dimensional pulse coupled neural network based on hybrid optimization algorithm for oil pollution image segmentation. *Remote Sensing*, 11(9), 1046.
20. Nama, S., Saha, A. K., & Sharma, S. (2022). Performance upgradation of symbiotic organisms search by backtracking search algorithm. *Journal of Ambient Intelligence and Humanized Computing*, 13(12), 5505–5546.
21. Sharma, S., Saha, A. K., & Lohar, G. (2022). Optimization of weight and cost of cantilever retaining wall by a hybrid metaheuristic algorithm. *Engineering with Computers*, 38(4), 2897–2923.
22. Li, L., Sun, L., Guo, J., Qi, J., Xu, B., & Li, S. (2017). Modified discrete grey wolf optimizer algorithm for multilevel image thresholding. *Computational Intelligence and Neuroscience*.
23. Chakraborty, S., Saha, A. K., Sharma, S., Chakraborty, R., Debnath, S. (2021). A hybrid whale optimization algorithm for global optimization. *Journal of Ambient Intelligence and Humanized Computing*, 1–37.
24. Chouhan, S. S., Kaul, A., & Singh, U. P. (2019). Image segmentation using computational intelligence techniques. *Archives of Computational Methods in Engineering*, 26(3), 533–596.
25. Al-Gburi, Z. D. S., & Kurnaz, S. (2022). Optical disk segmentation in human retina images with golden eagle optimizer. *Optik*, 271, 170103.
26. Abdollahzadeh, B., Gharehchopogh, F. S., & Mirjalili, S. (2021). African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. *Computers and Industrial Engineering*, 158, 107408.
27. Shayanfar, H., & Gharehchopogh, F. S. (2018). Farmland fertility: A new metaheuristic algorithm for solving continuous optimization problems. *Applied Soft Computing*, 71, 728–746.
28. Abdollahzadeh, B., Soleimani Gharehchopogh, F., & Mirjalili, S. (2021). Artificial gorilla troops optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems. *International Journal of Intelligent Systems*, 36(10), 5887–5958.
29. Ezugwu, A. E., Agushaka, J. O., Abualigah, L., Mirjalili, S., & Gandomi, A. H. (2022). Prairie dog optimization algorithm. *Neural Computing and Applications*, 34(22), 20017–20065.
30. Chakraborty, S., Saha, A. K., Ezugwu, A. E., Agushaka, J. O., Zitar, R. A., & Abualigah, L. (2022). Differential evolution and its applications in image processing problems: A comprehensive review. *Archives of Computational Methods in Engineering*, 1–56.
31. Sharma, S., Khodadadi, N., Saha, A. K., Gharehchopogh, F. S., & Mirjalili, S. (2022). Non-dominated sorting advanced butterfly optimization algorithm for multi-objective problems. *Journal of Bionic Engineering*, 1–25.
32. Sharma, S., Saha, A. K., Roy, S., Mirjalili, S., & Nama, S. (2022). A mixed sine cosine butterfly optimization algorithm for global optimization and its application. *Cluster Computing*, 25(6), 4573–4600.
33. Agushaka, J. O., Ezugwu, A. E., & Abualigah, L. (2022). Gazelle Optimization Algorithm: A novel nature-inspired metaheuristic optimizer. *Neural Computing and Applications*, 1–33.
34. Abdollahzadeh, B., Gharehchopogh, F. S., Khodadadi, N., & Mirjalili, S. (2022). Mountain Gazelle Optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems. *Advances in Engineering Software*, 174, 103282.
35. Abd El Aziz, M., Ewees, A. A., & Hassanien, A. E. (2017). Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. *Expert Systems with Applications*, 83, 242–256.
36. Abd Elaziz, M., Ewees, A. A., & Oliva, D. (2020). Hyper-heuristic method for multilevel thresholding image segmentation. *Expert Systems with Applications*, 146, 113201.
37. Bhandari, A. K., Singh, V. K., Kumar, A., & Singh, G. K. (2014). Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy. *Expert Systems with Applications*, 41(7), 3538–3560.
38. Pare, S., Kumar, A., Singh, G. K., & Bajaj, V. (2020). Image segmentation using multilevel thresholding: A research review. *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, 44(1), 1–29.
39. Ewees, A. A., Abualigah, L., Yousri, D., Sahlol, A. T., Al-qaness, M. A., Alshathri, S., & Elaziz, M. A. (2021). Modified artificial ecosystem-based optimization for multilevel thresholding image segmentation. *Mathematics*, 9(19), 2363.
40. Abd Elaziz, M., Ewees, A. A., Yousri, D., Alwerfali, H. S. N., Awad, Q. A., Lu, S., & Al-Qaness, M. A. (2020). An improved Marine Predators Algorithm with fuzzy entropy for multi-level thresholding: Real world example of COVID-19 CT image segmentation. *IEEE Access*, 8, 125306–125330.
41. He, L., & Huang, S. (2020). An efficient krill herd algorithm for color image multilevel thresholding segmentation problem. *Applied Soft Computing*, 89, 106063.



42. Moser, G., & Zerubia, J. (2018). *Mathematical models for remote sensing image processing: Signals and communication technology*. Berlin: Springer.
43. Liang, H., Jia, H., Xing, Z., Ma, J., & Peng, X. (2019). Modified grasshopper algorithm-based multilevel thresholding for color image segmentation. *IEEE Access*, 7, 11258–11295.
44. Huo, F., Sun, X., & Ren, W. (2020). Multilevel image threshold segmentation using an improved Bloch quantum artificial bee colony algorithm. *Multimedia Tools and Applications*, 79(3), 2447–2471.
45. Houssein, E. H., Helmy, B.E.-D., Oliva, D., Elngar, A. A., & Shaban, H. (2021). A novel black widow optimization algorithm for multilevel thresholding image segmentation. *Expert Systems with Applications*, 167, 114159.
46. Liu, L., Zhao, D., Yu, F., Heidari, A. A., Li, C., Ouyang, J., Chen, H., Mafarja, M., Turabieh, H., & Pan, J. (2021). Ant colony optimization with Cauchy and greedy Levy mutations for multilevel COVID 19 X-ray image segmentation. *Computers in Biology and Medicine*, 136, 104609.
47. Faramarzi, A., Heidarinejad, M., Mirjalili, S., & Gandomi, A. H. (2020). Marine predators algorithm: a nature-inspired metaheuristic. *Expert systems with applications*, 152, 113377.
48. Yousri, D., Abd Elaziz, M., Abualigah, L., Oliva, D., Al-Qaness, M. A., & Ewees, A. A. (2021). COVID-19 X-ray images classification based on enhanced fractional-order cuckoo search optimizer using heavy-tailed distributions. *Applied Soft Computing*, 101, 107052.
49. Hsu, D.-Z., Chen, Y.-W., Chu, P.-Y., Periasamy, S., & Liu, M.-Y. (2013). Protective effect of 3, 4-methylenedioxyphenol (sesamol) on stress-related mucosal disease in rats. *BioMed Research International*.
50. Zhao, S., Wang, P., Heidari, A. A., Chen, H., Turabieh, H., Mafarja, M., & Li, C. (2021). Multilevel threshold image segmentation with diffusion association slime mould algorithm and Renyi's entropy for chronic obstructive pulmonary disease. *Computers in Biology and Medicine*, 134, 104427.
51. Zhao, D., Liu, L., Yu, F., Heidari, A. A., Wang, M., Liang, G., Muhammad, K., & Chen, H. (2021). Chaotic random spare ant colony optimization for multi-threshold image segmentation of 2D Kapur entropy. *Knowledge-Based Systems*, 216, 106510.
52. Geiger, D., & Yuille, A. (1991). A common framework for image segmentation. *International Journal of Computer Vision*, 6(3), 227–243.
53. Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62–66.
54. Chen, Q., Zhao, L., Lu, J., Kuang, G., Wang, N., & Jiang, Y. (2012). Modified two-dimensional Otsu image segmentation algorithm and fast realisation. *IET Image Processing*, 6(4), 426–433.
55. Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z. W., & Gandomi, A. H. (2022). Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer. *Expert Systems with Applications*, 191, 116158.
56. Almotairi, K. H., & Abualigah, L. (2022). Hybrid reptile search algorithm and remora optimization algorithm for optimization tasks and data clustering. *Symmetry*, 14(3), 458.
57. Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Advances in Engineering Software*, 114, 163–191.
58. Tubishat, M., Ja'afar, S., Alswaiti, M., Mirjalili, S., Idris, N., Ismail, M. A., & Omar, M. S. (2021). Dynamic Salp Swarm Algorithm for feature selection. *Expert Systems with Applications*, 164, 113873.
59. Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software*, 95, 51–67.
60. Wang, D., Tan, D., & Liu, L. (2018). Particle swarm optimization algorithm: An overview. *Soft Computing*, 22(2), 387–408.
61. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-Qaness, M. A., & Gandomi, A. H. (2021). Aquila optimizer: A novel meta-heuristic optimization algorithm. *Computers and Industrial Engineering*, 157, 107250.
62. PirahanSiah, F., Abdullah, S. N. H. S., & Sahran, S. (2010). Adaptive image segmentation based on peak signal-to-noise ratio for a license plate recognition system. In *Book adaptive image segmentation based on peak signal-to-noise ratio for a license plate recognition system* (pp. 468–472). IEEE. <https://ieeexplore.ieee.org/xpl/conhome/5729855/proceeding>
63. Brunet, D., Vrscay, E. R., & Wang, Z. (2011). On the mathematical properties of the structural similarity index. *IEEE Transactions on Image Processing*, 21(4), 1488–1499.
64. Oquadfel, S., & Taleb-Ahmed, A. (2016). Social spiders optimization and flower pollination algorithm for multilevel image thresholding: A performance study. *Expert Systems with Applications*, 55, 566–584.
65. Akay, B. (2013). A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding. *Applied Soft Computing*, 13(6), 3066–3091.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.