Artificial Intelligence Topics and Applications **1**(1), 2025, 1–21

Boosted Pathfinder Algorithm and Firefly Algorithm for Multi-level Thresholding Image Segmentation Using Fuzzy Entropy

Rehab A. Ibrahim [*,1], **Marwa Gaheen**[2], and **Hashim Jarar**[3]

[1]Department of Mathematics, Faculty of Science, Zagazig University, Zagazig, 44519, Egypt.
[2]Department of Computer, Damietta University, Damietta 34517, Egypt.
[3]Faculty of Information Technology, Middle East University, Jordan.

Abstract Image processing is very important for many applications in computer vision and digital intelligence society applications. This study presents an alternative multilevel thresholding algorithm for image segmentation. The developed approach aims to improve a metaheuristic (MH) technique called pathfinder (PF) algorithm using firefly algorithm (FA) as a local operator. In the proposed algorithm, called PF-FA, the FA is employed to boost the exploitation ability of the traditional PF algorithm, which is used as a local operator. To test the performance of the developed PF-FA approach, we use a number of gray-scale images. Moreover, the developed PF-FA method is compared to some well-known MH techniques that are generally employed for multilevel thresholding at different threshold levels. The experimental results revealed that the PF-FA has a competitive performance, and the applications of FA have a significant impact on the traditional PF algorithm.

Keywords: Digital intelligence society, Pathfinder Algorithm, Firefly algorithm, Multi-level thresholding, Image segmentation.

1 Introduction

Image segmentation approaches play critical roles in image processing techniques, and they are adopted in different applications and domains, such as medical image [1] remote sensing [2], medical images [3] and other computer vision domains [3, 4]. In general, the segmentation process is done by splitting a given image into classes, and these classes have the same properties, such as brightness, color, texture, contrast, and grey level.
In literature, different image segmentation techniques are adopted, for example, clustering algorithms [5], edge detection [6], region extraction [7], and thresholding methods [8]. Thresholding can be considered an efficient segmentation technique which confirmed its high efficiency in various segmentation approaches [9, 10]. It contains two different techniques, called bi-level and multi-level thresholding. In general, for the bi-level (BL)

*Corresponding author: rehab100r@yahoo.com.

Received: 05 May 2024; Revised: 11 November 2024; Accepted: 20 November 2024
Available Online: 18 December 2024
Artificial Intelligence Topics and Applications

technique, the image is divided into two classes; thus, it is not suitable if the segmented image has more than two classes. Where in multi-level technique, the image can be segmented into several classes. The multi-level thresholding (MLT) techniques are widely applied for in various computer vison applications. Traditional MLT techniques were implemented by applying image histograms to allocate the optimal values using minimization or maximization of the fitness function, for example, entropy and Otsu. But, these traditional multi-level algorithms face critical limitations in finding the optimal threshold values.

Therefore, in the recent decade, metaheuristics (MH) methods have been adopted in various applications [11, 12, 13, 14, 15, 16], including image segmentation using MLT technique, such as marine predators algorithm [17], whale optimization algorithm [18, 19], salp swarm algorithm [20], differential evolution [21], manta ray foraging optimizer [22], genetic algorithm [23] and others [10, 24, 25, 26, 9].

In this study, we introduce an alternative technique for MLT using a modified Pathfinder algorithm (PF). The PF optimizer is a new MH algorithm proposed by Yapici and Cetinkaya [27]. The PF algorithm was inspired by the collective movements of animal groups and simulates the leadership hierarchies of the swarms to find the food area. The PF algorithm has been used in some approaches. For example, Qi et al. [28] used a hybrid PF with the differential evolution (DE) to address constrained optimization problems.

The firefly algorithm (FA) is utilized to improve the PF algorithm's search ability. FA is a natural inspired optimization algorithm proposed by simulating fireflies flashing behavior [29, 30]. It has been widely utilized in different optimization tasks. For example, Nayak et al. [31] discussed the applications of the firefly algorithm in classification tasks. Jain and Katarya [32] employed a modified firefly algorithm for discovering opinion leaders from social media discussions. Zhou et al. [33] applied the firefly algorithm with fuzzy systems to forecast air overpressure. In [34], the authors applied the FA with the GA for scheduling cloud computing tasks. Moreover, Yang [35] discussed and studied the applications of the FA in different image processing applications. Furthermore, in [36], the FA was employed to enhance the performance of the WOA to solve scheduling problems. Additionally, the FA was employed in different image processing tasks such as [37, 38, 38, 39]

In short, our main objectives and contributions presented in this study can be stated as follows:

- Present an alternative technique for MLT approach using on a new version of the pathfinder algorithm based on the firefly algorithm.
- Enhance the search efficiency of the traditional PF method by employing the operators of the FA.
- Evaluate the proposed method, called PF-FA, with several images, and compare the proposed method with several existing metaheuristic algorithms to assess its high performance.

The rest sections of this paper are organized as follows. First, we present a simple review of the MLT studies, including MH algorithms in Section 2. In Section 3, we describe the problem formulation and the preliminaries of the PF and FA. In Section 4, we describe the proposed method. Section 5 introduces the experimental evaluation and outcomes. Finally, Section 6 introduces the conclusion and future works.

2 Related Work

In this section, a light review of the existing metaheuristic algorithms used for image segmentation using MLT is presented.

In [40], an improved salp swarm algorithm was proposed for MLT. The mouth flame optimizer (MFO) is used to enhance the SSA. This hybrid model, called SSAMFO, is applied for image segmentation with a number of greyscale Images. The evaluation experiments were implemented by comparing the SSAMFO to several existing MH methods; where the outcomes confirmed the good achievement of the SSAMFO. Alwerfali et al. [41] introduced a MLT approach using a modified spherical search optimizer (SSO) with fuzzy entropy. The sine cosine algorithm is applied as a local search for the SSO to improve its search ability. The introduced approach, called SSOSCA, had been applied to segment greyscale images, and it had been evaluated with a number of images. Extensive evaluation experiments and comparisons had been carried out to evaluate SSOSCA.

In [42], the subspace elimination optimization algorithm is applied for MLT. The authors used four images to test the proposed technique. They also compared the proposed approach to the PSO algorithm, and they concluded that the subspace elimination optimization algorithm showed better performance than the particle swarm optimization. Oliva et al. [43], proposed a MLT method, called MCET-CSA, by integrating minimum cross-entropy thresholding (MCET) with the CSA (crow search algorithm) . The MCET-CSA is employed by minimizing cross-entropy among classes. More so, the outcomes of the experiments showed that the MCET-CSA method has a better performance compared to the differential evolution and the harmony search algorithm.

In [44], the authors presented a modified differential evolution algorithm, namely synergetic differential evolution, for MLT. The experimental outcomes showed that the modified method performs better than several traditional MH algorithms in terms of finding optimal threshold values. In [45], the authors applied a new MH algorithm, marine predators algorithm (MPA) for MLT. The moth-flame optimization (MFO) algorithm is employed as a local search technique for the marine predators algorithm. This approach, called MPAMFO had been used for image segmentation, including COVID-19 CT images. A number of existing MH methods have been compared to the MPAMFO, and it achieved competitive performance.

In [46], the authors used quantum particle swarms algorithm for MLT with Otsu's fitness function. They used several images to assess the test the developed approach, and results showed that the proposed approach could improve the sped and the accuracy of the image segmentation process. In [47], the authors proposed two MH methods for MLT, called the whale optimization algorithm and MFO algorithm. The experimental evaluation showed that the MFO algorithm has better performance than all compared algorithms, including the whale optimization algorithm. In [48], the authors presented a MLT method using the human mental search algorithm. The human mental search algorithm had been tested with a number of images, and it had been compared to several existing methods to validate its performance.

Akay [49] presented a hybrid DE for MLT. The hybrid model was applied for obtaining the optimal threshold value of grey-level images. they used Otsu fitness function. The proposed method had been compared to several MH algorithms, and it obtained good results. Additionally, in [50], a modified artificial bee colony using the sine cosine algorithm was developed. He et al. [51] proposed a new modified variant of the FA. This method had been employed by minimizing Kapur entropy, cross-entropy, and intra-

class variances. Extensive experiments comparisons had been carried out to verify the superiority of the improved firefly algorithm over other MH approaches. Recently, in [24], the authors developed the MFO for MLT using Kapur/Tsallis entropy. The main idea is to utilize MFO for supporting the identification of the finest threshold for RGB images. It showed competitive performance compared to several algorithms. In [25], the authors studied the capability of several optimization algorithms for MLT applications. Several methods have been tested, including FA, DE, PSO, GA, ABC, and others. In [26], an efficient MLT method was proposed for COVID-19 images based on an enhanced version of the equilibrium optimizer. In the same manner, the marine predator algorithm (MPA) was used for COVID-19 image segmentation in [52]. There are other optimization methods that used for MLT in recent years, such as Manta Ray Foraging Optimizer [53], Slime mould algorithm [54], and enhanced version of the remora optimization algorithm [55].

3 Perminaly

In this section we present a brief description of the multi-level threshodling (MLT) problem, the firefly algorithm and the pathfinder algorithms.

3.1 Problem Definition

The mathematical definitions of the MLT problem is presented by considering I represents the gray-scale image which has $K+1$ classes. To segment I into its elements, a set of k thresholds $\{t_k, k = 1, 2, K\}$ are required to be determined, and this formulated as:

$$\begin{aligned} C_0 &= \{I_{ij} \mid 0 \leq I_{ij} \leq t_1 - 1\}, \\ C_1 &= \{I_{ij} \mid t_1 \leq I_{ij} \leq t_2 - 1\}, \\ &\cdots \\ C_K &= \{I_{ij} \mid t_K \leq I_{ij} \leq L - 1\} \end{aligned} \tag{1}$$

In Eq. (1), L denotes the maximum gray levels, C_K denotes the kth class of I. To find the threshold value, the MLT is considered as a maximization problem and defined as:

$$t_1^*, t_2^*, ..., t_K^* = \arg \max_{t_1,...,t_K} Fit(t_1, ..., t_K) \tag{2}$$

where Fit denotes the Fuzzy entropy that used as fitness function [56] since it has been established its performance in different image segmentation works [18, 57, 58]. It is formulated as:

$$Fit(t_1, ..., t_K) = \sum_{k=1}^{K} H_i \tag{3}$$

$$H_k = -\sum_{i=0}^{L-1} \frac{p_i \times \mu_k(i)}{P_k} \times ln(\frac{p_i \times \mu_k(i)}{P_k}), \tag{4}$$

$$P_k = \sum_{i=0}^{L-1} p_i \times \mu_k(i) \tag{5}$$

$$\mu_1(l) = \begin{cases} 1 & l \leq a_1 \\ \frac{l-c_1}{a_1-c_1} & a_1 \leq l \leq c_1 \\ 0 & l > c_1 \end{cases} \tag{6}$$

$$\mu_K(l) = \begin{cases} 1 & l \leq a_{K-1} \\ \frac{l-a_K}{c_K-a_K} & a_{K-1} < l \leq c_{K-1} \\ 0 & l > c_{K-1} \end{cases}, p_i = h(i)/N_p (0 < i < L-1) \quad (7)$$

where p_i denotes probability. N_p and $h(i)$ define total pixels' number and the pixels' number in I. $a_1, c_1, ..., a_{k-1}, c_{k-1}$ stands for the fuzzy parameters and $0 \leq a_1 \leq c_1 \leq ... \leq a_{K-1} \leq c_{K-1}$. Then $t_1 = \frac{a_1+c_1}{2}, t_2 = \frac{a_2+c_2}{2}, ..., t_{K-1} = \frac{a_{K-1}+c_{K-1}}{2}$.

3.2 Pathfinder Algorithm

The Pathfinder Algorithm (PFA) algorithm has recently been used as a new metaheuristic. Its inspiration is taken from the group moving and hunting of some animals and mimics the structure of leading of some swarms for reaching the finest region of food or prey. The general steps of the algorithm can be summarized as follows:

The first step is to generate the population. Then determining the leader of each swarm that means the pathfinder. Followed by modernizing the pathfinder location using the following equation:

$$x^{k+1} = x^k + 2r_1(x^k - x^{k+1}) + B \quad (8)$$

where $r_1 \in [0, 1]$ denotes a uniform arbitrary number, $B = u_1 e^(-2k)/k_m ax)$, u_1 is an arbitrary vector in the range [-1,1] and in general B represent random walk of members.

The next step is to compare the current pathfinder and the previous one, then modernizing the location of the individuals using the following equation:

$$x_i^{k+1} = x_i^k + R_1(x_j^k - x_i^k) + R_2(x^k - x_i^k) + \omega \quad (9)$$

where k is the current iteration, x_i, x_j exemplify the location vector of the ith and the jth members, and R_1 and R_2 are random numbers. Thereafter, the function value is evaluated and finds the best one according to which identify the best pathfinder.

3.3 Firefly optimization algorithm

In this subsection, the basic definition of the Firefly Algorithm (FA) are introduced [59]. There are two factors in FA that have the largest effect on its performance. These factors are the attractiveness and brightness, and attractiveness is updated using Eq. (10).

$$\beta = \beta_0 \times e^{(-\gamma r^2)} \quad (10)$$

where β denotes the attractiveness between ith solution and jth. While, $\beta_0 = 1$ is the attractiveness at distance $r = 0$ and γ denotes the light absorption coefficient and it is a constant, The distance r is defined as:

$$r_{ij} = ||x_i - x_j|| = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2} \quad (11)$$

where $x_{(i,k)}$ denotes the kth value of ith firefly x_i.
The movement of the i firefly is attracted to the more attractive one (i.e., x_j) and it is formulated as:

$$x_i = x_i + \beta \times (x_i - x_j) + \alpha \times \varepsilon_i \quad (12)$$

where $\alpha \in [0, 1]$ refers to a random value and $\varepsilon_i \in N(\mu, \sigma)$ denotes random vector numbers.

The Full description of FA is given in Algorithm 1. In the case of maximization optimization, the brightness is proportional to the fitness value [59].

Algorithm 1 The basic stepes of the original Firefly algorithm

Inputs: N number of solution, D dimensions of problem.
Determine the value of light absorption coefficient γ.
Generate initial population X.
Compute the fitness value $f(x), x = (x_1, x_2, \ldots, x_d)^T$ which represents light intensity I_i at x_i.
$t = 1$.
while $(t < t_{max})$ **do**
  for $i = 1 : n(allnfireflies)$ **do**
    for $j = 1 : i$ **do**
      if $I_i < I_j$ **then**
        Shift firefly i to j in dimension d.
      end if
      Update attractiveness using Eq. (12).
    end for
  end for
  Sort X using fitness value then determine the best one.
end while
Return the best solution.

4 Proposed PF-FA

In this section, the details of the PF-FA are introduced. The modified PF depends on employing the structure of the FA as a local search.

The developed PF-FA begin by generating X which contains N solutions using Eq. (13).

$$X_{i,j} = I_{min,j} + r_1 \times (I_{max,j} - I_{min,j}), \;\; j = 1, 2, ..., D, \tag{13}$$

in which $I_{min,j}$ and $I_{max,j}$ denotes the minimum and maximum gray values of I at jth dimensions. More so, $D = 2K$, where K represents to threshold level.

Thereafter, the fitness value Fit for X_i is computed for each solution using Eq. (2). Followed by finding the best solutions X_b. Thereafter, the solutions will be updated using either PF or FA depends on the probability (Pr_i) of each solution that computed as:

$$Pr_i = \frac{Fit_i}{\sum_{i=1}^{N} Fit_i} \tag{14}$$

$$X_i = \begin{cases} operators\, of\, PF & Pr_i > r1 \\ operators\, of\, FA & otherwise \end{cases} \tag{15}$$

where

$$r_s = min(Pr_i) + r_a \times (max(Pr_i) - min(Pr_i)),\; r_a \in [0, 1] \tag{16}$$

where the main goal of using Eq. (16) is to avoid fixing the value of r_s as a constant value. This leads to update $r1$ in automatic form.

However, Updating solution process is repeated till reaching the terminal conditions. Then using X_b to construct the fuzzy parameters which applied to construct the threshold value as $t_k = \frac{U_b^k + U_b^{k+1}}{2}$, where $k = 1 : 2 : K - 1$.

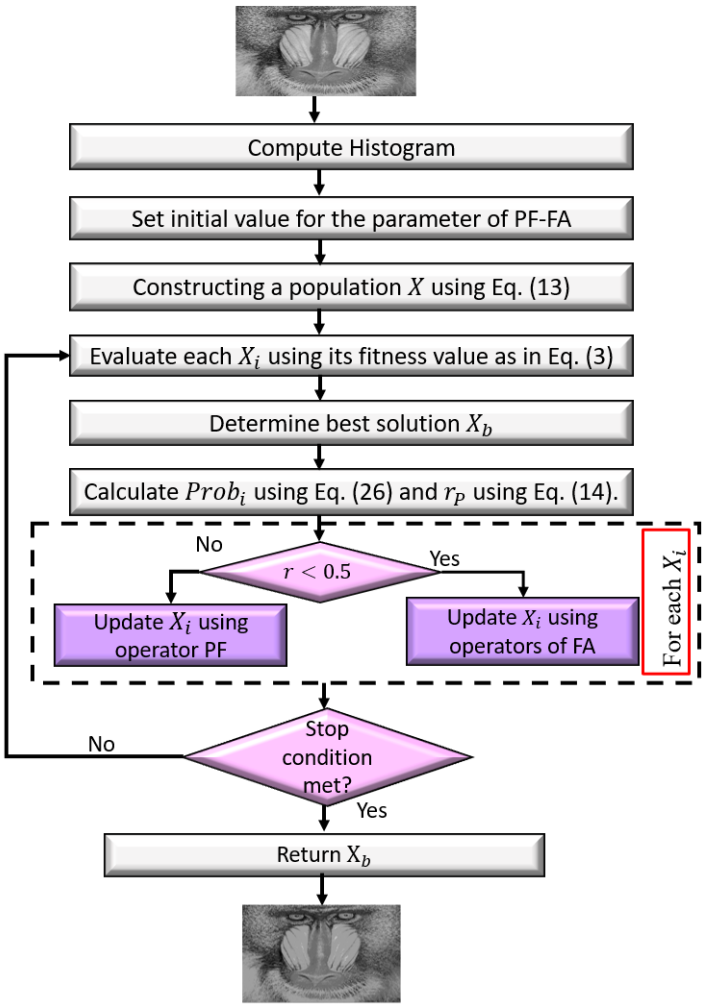


Figure 1: The basic structure and Framework of the PF-FA.

5 Experiments and Discussion

5.1 Performance metrics

This paper applies two performance metrics to assess the quality of the PF-FA algorithm, namely Structural Similarity Index (SSIM) [60] and Peak Signal-to-Noise Ratio (PSNR) [61] The PSNR and SSIM are most commonly applied to measure the quality of segmented images. The PSNR is easy to implement and mathematically convenient in optimization tasks as well as it has been widely tested and valid. Also, the SSIM is a popular measure in the field and applied using the images' contrast, luminance, and structure. It showed its ability to estimate content-dependent distortions and capture blurring and noise. They are calculated as in the following expressions:

$$SSIM(G, G_S) = \frac{(2\mu_G\mu_{G_S} + c_1)(2\sigma_{G,G_S} + c_2)}{(\mu_G^2 + \mu_{G_S}^2 + c_1)(\sigma_G^2 + \sigma_{G_S}^2 + c_2)} \tag{17}$$

In 17, μ_{G_S} (σ_{G_S}) and $\mu_G(\sigma_G)$ represent the standard deviation of G_S and G, respectively. σ_{G,G_S} denotes the covariance of G_S and G.

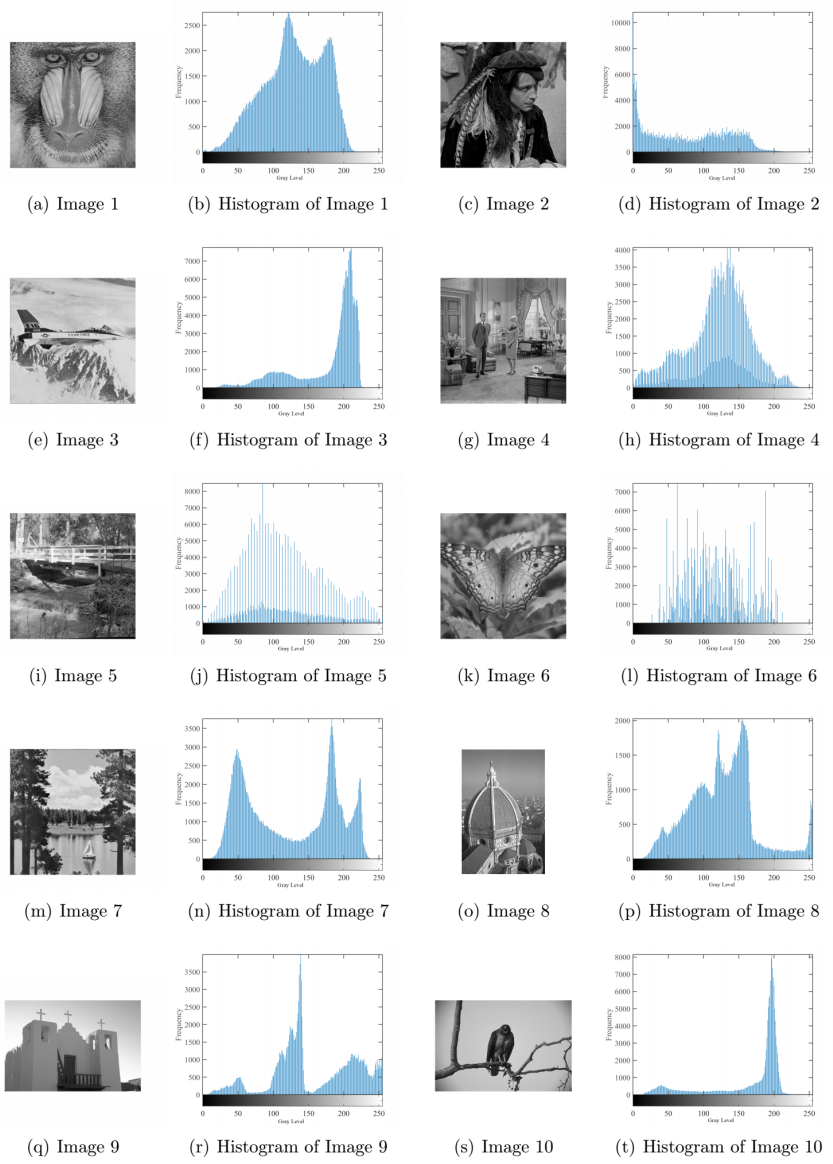
$$PSNR = 20log_{10}(\frac{255}{RMSE}), \tag{18}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N_r} \sum_{j=1}^{N_c} (G_{i,j} - G_S i, j)^2}{N_r \times N_c}}$$

In 18, $RMSE$ indicates the root mean-squared error. G represents the standard image. G_S represents the segmented images. $N_r \times N_c$ is the images size.

5.2 Results

In the experiment, ten images was used to evaluate the proposed PF-FA method in solving the segmentation problems, both PSNR and IGD were computed. Figure 2 shows the differences among these images. The proposed PF-FA and eight compared algorithms (i.e. PF, GWO, SCA, GOA, PSO, MFO) are evaluated in six thresholds values (i.e., 6, 8, 15, 17, 19, and 25) therefore, the total cases of segmentation in the experiment is 60 cases.



(a) Image 1 (b) Histogram of Image 1 (c) Image 2 (d) Histogram of Image 2

(e) Image 3 (f) Histogram of Image 3 (g) Image 4 (h) Histogram of Image 4

(i) Image 5 (j) Histogram of Image 5 (k) Image 6 (l) Histogram of Image 6

(m) Image 7 (n) Histogram of Image 7 (o) Image 8 (p) Histogram of Image 8

(q) Image 9 (r) Histogram of Image 9 (s) Image 10 (t) Histogram of Image 10

Figure 2: Images dataset and their histograms.

The PSNR results are recorded in Tables 1-2 as well as Figure 3. It presents the results

of all methods over all images. More details, the performance of the PF-FA at level 6 was similar to MFO whereas, they obtained the highest PSNR results in five images for each one whereas, PF got the third rank followed by GWO, PSO, and GOA.

The PSO obtained the first rank at threshold level 8, it got the best results in 40% of all images whereas, PF-FA and MFO showed similar performance in most images and obtained the best PSNR in two images for each one. GWO was ranked third followed by PF and GOA.

Moreover, the PF-FA and PSO achieved the highest PSNR results in four images for each one at level 15 whereas, the MFO got the best PSNR in two images. In this level, the performance of the PF was like GWO whereas the SCA got the worst performance. Furthermore, at level 17, PF-FA, PSO, and MFO algorithms got the best PSNR in three images followed by GWO and PF.

In addition, level 19 showed the superiority of the PF-FA algorithms. It achieved the highest PSNR in four images whereas, both PSO and MFO acted similarly and obtained the best PSNR in three images. Furthermore, both PF-FA and PSO showed the best results of PSNR in three images in threshold level 25. As well as, the GWO and MFO acted similarly and got the best values in two images. The worst PSNR at all levels were showed by the SCA.

Furthermore, Table 2 shows the average of the PSNR for all images at every thresholds. In this table the proposed PF-FA obtained the best average in 50% of the thresholds.

Table 1: PSNR results for all algorithms

K	Image No.	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
6	1	**14.906**	14.234	14.123	10.774	11.847	13.562	12.252
	2	**16.876**	15.720	15.612	12.476	12.495	15.455	16.874
	3	13.131	12.831	12.605	10.763	10.720	11.330	**13.203**
	4	**16.341**	16.244	16.331	10.776	11.028	16.025	15.307
	5	**13.224**	12.377	11.903	10.490	10.806	10.880	13.054
	6	12.569	11.752	12.183	10.374	10.955	11.507	**13.017**
	7	12.312	11.701	11.822	11.852	12.451	11.687	**14.078**
	8	14.894	14.213	14.019	10.787	10.551	13.397	**15.230**
	9	11.560	10.737	10.599	9.928	11.054	9.386	**13.196**
	10	**14.439**	13.914	14.424	12.049	13.122	13.073	14.199
8	1	**18.624**	17.814	17.706	14.666	17.051	17.989	16.840
	2	17.301	16.372	16.539	16.729	15.643	17.825	**18.266**
	3	15.842	15.180	15.964	13.050	14.909	**16.278**	16.175
	4	**18.278**	17.693	17.064	15.309	16.777	18.241	17.185
	5	15.968	15.843	16.157	15.042	15.748	**16.431**	16.054
	6	16.238	15.515	15.585	13.302	14.066	**17.116**	16.225
	7	15.859	15.575	15.544	15.067	15.139	**16.022**	15.704
	8	15.684	15.051	**16.899**	15.104	14.709	16.833	16.423

Table 1: PSNR results for all algorithms (*continued*)

K	Image No.	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
	9	15.071	14.971	15.424	13.215	14.237	14.987	**16.155**
	10	18.519	18.451	**19.316**	14.643	18.182	17.830	17.730
15	1	**23.074**	22.320	21.509	16.026	20.835	21.847	20.842
	2	22.007	21.694	22.187	17.383	20.035	**23.379**	20.748
	3	20.203	20.121	19.667	15.366	19.299	**23.026**	22.105
	4	21.529	21.402	21.685	15.872	19.882	**22.977**	21.057
	5	**21.700**	21.013	21.295	15.646	18.609	20.250	20.888
	6	19.860	19.796	20.510	13.732	17.751	23.115	**23.510**
	7	**21.253**	20.319	20.229	16.132	18.422	19.913	20.495
	8	**22.515**	21.147	21.299	15.678	18.722	21.748	22.505
	9	18.475	18.134	18.096	15.417	17.775	19.989	**21.206**
	10	21.502	20.562	21.467	16.760	19.492	**24.165**	21.719
17	1	**24.568**	23.937	23.075	15.709	22.315	23.525	21.207
	2	23.871	23.220	**24.048**	16.963	20.855	23.653	22.454
	3	22.316	21.539	20.658	15.191	20.903	23.306	**23.505**
	4	22.402	22.288	22.487	15.582	20.985	**24.194**	23.322
	5	**22.980**	22.000	22.868	16.430	20.365	22.892	22.909
	6	21.937	21.108	22.155	14.710	19.231	**23.480**	22.317
	7	21.619	21.425	21.414	15.900	20.145	22.094	**22.598**
	8	**23.611**	21.777	22.887	15.465	19.943	22.843	23.474
	9	19.457	19.309	19.356	15.154	18.916	21.320	**22.526**
	10	22.190	21.453	21.930	16.743	20.542	**23.223**	22.035
19	1	**24.921**	24.447	24.251	15.779	23.077	24.320	24.370
	2	**25.177**	24.579	24.971	18.023	22.273	25.050	24.647
	3	23.097	22.352	21.786	16.018	21.583	**24.976**	23.371
	4	**24.181**	23.567	23.913	15.229	21.438	23.916	23.451
	5	23.777	23.124	23.857	16.006	21.752	**24.064**	23.724
	6	23.495	22.962	23.623	14.915	20.327	24.136	**24.216**
	7	23.603	22.954	22.666	16.260	21.274	**24.346**	23.158
	8	24.072	23.123	23.879	16.487	20.465	24.208	**24.848**
	9	21.014	20.379	20.864	16.670	19.788	22.311	**24.358**
	10	**23.509**	22.748	22.756	16.513	21.452	23.108	23.434
25	1	26.914	26.887	26.732	16.734	25.759	**28.313**	26.519
	2	27.732	27.089	**28.058**	19.267	26.214	27.481	27.394

Table 1: PSNR results for all algorithms (*continued*)

K	Image No.	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
	3	25.310	24.978	23.930	15.478	23.908	27.545	**27.903**
	4	**28.798**	25.942	26.257	16.857	24.955	28.649	27.207
	5	26.090	26.021	26.906	16.767	24.826	**27.115**	26.562
	6	26.306	26.135	**27.280**	16.896	23.776	27.276	25.356
	7	**26.373**	25.911	25.971	16.729	24.731	26.051	25.698
	8	26.659	26.346	26.709	17.923	24.640	27.115	**27.163**
	9	24.540	24.189	24.435	16.417	23.307	**27.285**	25.832
	10	**26.543**	25.528	25.956	17.842	24.661	26.258	25.700

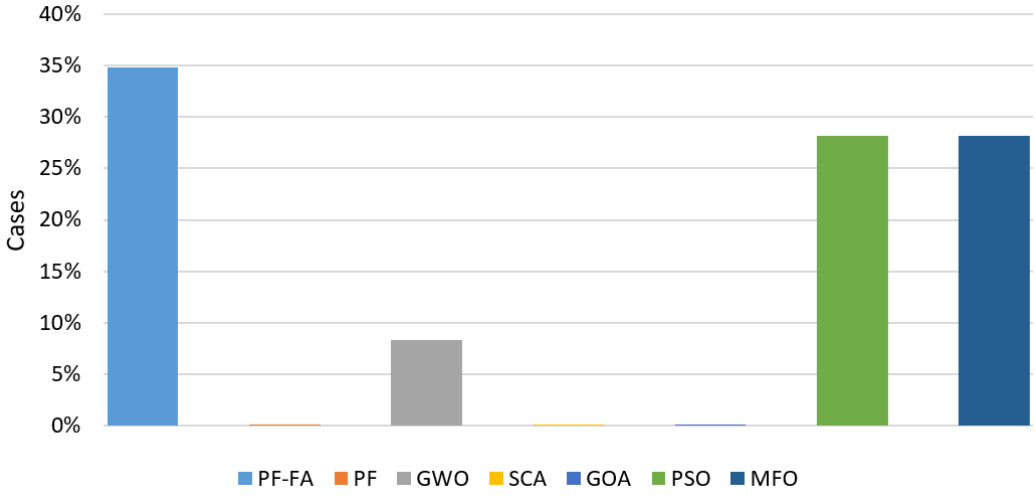


Figure 3: Best cases ratio achieved by the algorithms using PSNR measure.

Table 2: Average PSNR of the algorithms at each thresholds levels

	PF-FA	PF	GWI	SCA	GOA	PSO	MFO
6	14.025	13.372	13.362	11.027	11.503	12.630	**14.041**
8	16.758	16.247	16.620	14.613	15.646	**16.955**	16.676
15	**21.472**	20.651	20.794	15.801	19.082	21.424	21.408
17	**22.819**	21.805	22.088	15.785	20.420	22.785	22.695
19	23.685	23.023	23.257	16.190	21.343	**24.043**	23.958
25	**26.946**	25.903	26.224	17.091	24.678	26.939	26.427

The SSIM results of the methods are recorded in Tables 3-4 as well as Figure 4. This table shows that, the PF-FA got the best results of the SSIM in 43% of all images at all thresholds followed by PSO, GWO, and MFO, respectively. In detail, at levels 6, 8, 15, the PF-FA has the best values of the SSIM measure in four images at each level whereas,

the PSO was ranked second and has the best SSIM in three images at each level. In addition, MFO was ranked third at level 6 and 8 whereas, at level 15, it was ranked fifth after both GWO and PF, respectively.

At level 17, the PF-FA, GWO, and PSO achieved the first, second, and third ranks, respectively by achieving the best SSIM in three images for each one. The PF came in the fourth rank with only image.

The superiority of the PF-FA was showed at level 19 and 25, it got the best results of the SSIM in 60% and 50% of all images, respectively followed by GWO and MFO. The PSO was ranked fourth in both levels. Whereas, the SCA algorithm recorded the bad performance in all threshold's levels.

Furthermore, Table 4 shows the average of the SSIM for all images at every thresholds. In this table the proposed PF-FA obtained the best average in 66% of the thresholds.

Moreover, the results of the Friedman as a non-parametric test are listed in Tables 5-6 for both PSNR and SSIM measures. From these tables we can see that, there are significant differences between the PF-FA and the compared method in most cases whereas, it showed similar performance with the PSO in two cases as in the PSNR results.

Table 3: Comparative of algorithms based on SSIM values.

K	Image No.	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
6	1	**0.5492**	0.5216	0.5103	0.3994	0.4897	0.4156	0.3673
	2	0.4455	0.4179	0.4023	0.2248	0.3849	**0.5287**	0.2530
	3	0.6424	0.6148	0.6072	0.6113	0.6125	0.6476	**0.6987**
	4	**0.5773**	0.5497	0.5513	0.2871	0.5386	0.5255	0.2726
	5	**0.3994**	0.3468	0.3153	0.2370	0.2469	0.3983	0.2260
	6	0.3580	0.3304	0.3617	0.2679	0.3087	0.3509	**0.3656**
	7	0.4347	0.4071	0.4188	0.4016	0.3957	**0.5452**	0.3934
	8	0.6058	0.5782	0.5727	0.3885	0.5410	0.6057	**0.6109**
	9	0.6830	0.6555	0.7024	0.5829	0.5644	**0.7153**	0.5218
	10	**0.7811**	0.6519	0.6614	0.7687	0.6270	0.7796	0.7444
8	1	0.6708	**0.7119**	0.7044	0.5414	0.6805	0.6943	0.6371
	2	0.4884	0.4608	0.4567	0.4828	0.4037	0.4283	**0.4885**
	3	**0.7850**	0.7574	0.7528	0.6581	0.7520	0.7745	0.7772
	4	**0.6344**	0.6068	0.5889	0.5547	0.5731	0.6188	0.5839
	5	0.5052	0.5463	0.5653	0.5211	0.5366	**0.5704**	0.5540
	6	0.5601	0.5326	0.5342	0.4111	0.4460	**0.5842**	0.5334
	7	**0.5993**	0.5717	0.5685	0.5872	0.5367	0.5552	0.5526
	8	0.6087	0.6498	**0.7071**	0.6402	0.6364	0.6730	0.6679
	9	**0.8186**	0.7910	0.8046	0.7434	0.7806	0.7995	0.8106
	10	0.7754	0.7478	0.7632	0.7920	0.7384	**0.8345**	0.8218
15	1	**0.8574**	0.8298	0.8128	0.6043	0.7950	0.8076	0.7880

Table 3: SSIM results for all algorithms (*continued*)

K	Image No.	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
	2	**0.7040**	0.6765	0.7036	0.5882	0.5865	0.6523	0.5914
	3	0.8795	**0.8795**	0.8498	0.7061	0.8265	0.8287	0.7899
	4	0.7648	0.7372	0.7485	0.5844	0.6808	**0.7743**	0.7171
	5	**0.8038**	0.7762	0.7845	0.5519	0.6729	0.7339	0.7547
	6	0.7183	0.6907	0.7220	0.4335	0.6174	**0.7723**	0.7569
	7	**0.7964**	0.7688	0.7929	0.6150	0.6461	0.6852	0.6896
	8	0.8509	0.8233	0.8354	0.6835	0.7664	0.8275	**0.8577**
	9	0.8541	0.8265	**0.8542**	0.8011	0.8248	0.8429	0.8438
	10	0.8633	0.8357	0.8462	0.8193	0.8235	**0.8873**	0.8336
17	1	0.8223	**0.8634**	0.8434	0.5905	0.8306	0.8465	0.7954
	2	0.7550	0.7274	**0.7562**	0.5487	0.6155	0.6523	0.6851
	3	**0.8962**	0.8686	0.8666	0.7206	0.8503	0.8463	0.8096
	4	0.7863	0.7587	0.7722	0.5654	0.7264	**0.7997**	0.7788
	5	**0.8284**	0.8008	0.8275	0.6060	0.7448	0.8201	0.8231
	6	0.7702	0.7426	0.7747	0.4797	0.6767	**0.7836**	0.7532
	7	0.8165	0.7889	**0.8193**	0.6224	0.7288	0.7550	0.7745
	8	0.7932	0.8343	**0.8635**	0.6818	0.7922	0.8405	0.8517
	9	**0.8651**	0.8375	0.8549	0.8118	0.8308	0.8591	0.8528
	10	0.8116	0.8527	0.8550	0.8322	0.8533	**0.8783**	0.8610
19	1	0.8988	0.8712	0.8653	0.5989	0.8474	0.8599	**0.8995**
	2	0.7180	0.7591	**0.7881**	0.5924	0.6684	0.7400	0.7209
	3	**0.9052**	0.8776	0.8741	0.6996	0.8577	0.8401	0.8353
	4	**0.8204**	0.7928	0.8019	0.5564	0.7343	0.8036	0.7897
	5	**0.8577**	0.8301	0.8525	0.5769	0.7876	0.8503	0.8380
	6	0.7552	0.7963	**0.8093**	0.4920	0.7149	0.7970	0.7945
	7	0.8444	0.8168	0.8400	0.6469	0.7615	0.8147	**0.8512**
	8	**0.8852**	0.8576	0.8809	0.7275	0.8065	0.8677	0.8736
	9	**0.8801**	0.8525	0.8711	0.8115	0.8372	0.8686	0.8783
	10	**0.9098**	0.8822	0.8870	0.8297	0.8703	0.8959	0.8820
25	1	**0.9373**	0.9097	0.9058	0.6415	0.8951	0.9320	0.9046
	2	0.8646	0.8371	**0.8647**	0.6370	0.7923	0.8106	0.8202
	3	**0.9284**	0.9008	0.9014	0.7196	0.8830	0.8916	0.8720
	4	0.8719	0.8444	0.8544	0.6086	0.8226	0.8585	**0.8878**
	5	0.8519	0.8930	**0.9111**	0.6057	0.8664	0.9106	0.8944

Table 3: SSIM results for all algorithms (*continued*)

K	Image No.	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
	6	**0.8921**	0.8645	0.8864	0.5768	0.8223	0.8704	0.8317
	7	0.8876	0.8600	0.8777	0.6686	0.8439	0.8571	**0.8888**
	8	**0.9271**	0.8995	0.9145	0.7560	0.8838	0.9233	0.9000
	9	0.9090	0.8814	0.8934	0.8182	0.8740	**0.9112**	0.8894
	10	**0.9473**	0.9197	0.9223	0.8431	0.9025	0.9042	0.8845

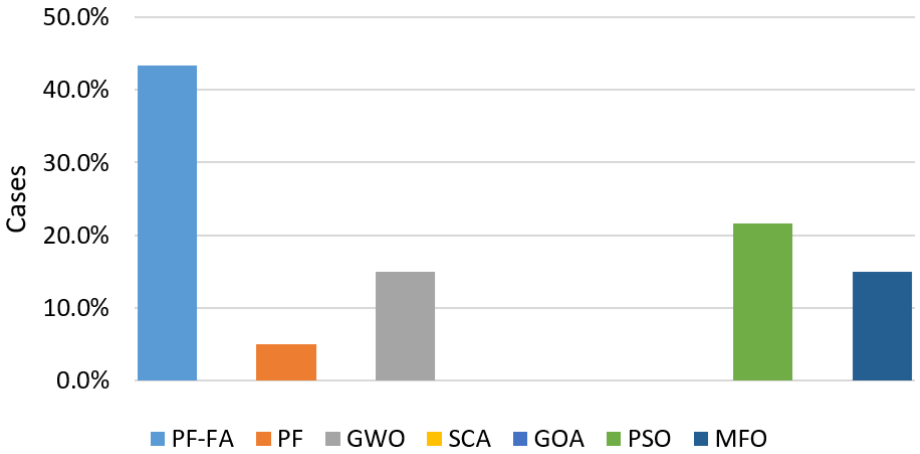


Figure 4: Ratio of the best cases achieved by each method using SSIM measure.

Table 4: Average of SSIM of the algorithm at each thresholds levels

	PF-FA	PF	GWI	SCA	GOA	PSO	MFO
6	0.548	0.507	0.510	0.417	0.471	**0.551**	0.445
8	**0.653**	0.638	0.645	0.593	0.608	0.652	0.643
15	**0.809**	0.784	0.795	0.639	0.724	0.781	0.762
17	0.814	0.808	**0.823**	0.646	0.765	0.808	0.799
19	**0.847**	0.834	0.847	0.653	0.789	0.834	0.836
25	**0.902**	0.881	0.893	0.688	0.859	0.887	0.877

In general, the PF-FA achieved the highest PSNR values in 35% of the tested images, followed by the PSO and MFO with 28% for each one. Whereas, in SSIM measure, the PF-FA has the highest SSIM in 43% of images followed whereas, the PSO has the second rank with 22% and the GWO and MFO with 15% for each one. The PF was ranked fifth both measures. The worst algorithm in all image was SCA, it recorded the low quality results in all measures. Moreover, Figure 5 shows the performance of the segmented Image 1 and Image 3 at level 19. We can be observed from this images that the quality of developed PF-FA is better than others methods in case of the two images (1 and 3).

Table 5: Friedman results of the PSNR measure

K	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
6	**6.40**	4.50	4.40	1.60	2.60	2.70	5.80
8	5.40	3.50	4.80	1.50	2.10	**5.90**	4.80
15	**5.70**	4.10	4.50	1.00	2.00	5.60	5.10
17	**6.00**	3.71	4.29	1.00	2.00	**6.00**	5.00
19	5.80	3.60	4.20	1.00	2.00	**6.00**	5.40
25	**6.00**	3.33	5.67	1.00	2.00	**6.00**	4.00

Table 6: Friedman results of the SSIM measure

K	PF-FA	PF	GWO	SCA	GOA	PSO	MFO
6	**6.10**	4.20	4.30	2.20	2.60	5.40	3.20
8	4.90	4.10	4.60	2.60	2.00	**5.20**	4.60
15	**6.25**	4.45	5.50	1.10	2.10	4.70	3.90
17	5.10	4.20	**5.70**	1.10	2.60	5.10	4.20
19	**6.00**	4.30	5.50	1.00	2.20	4.50	4.50
25	**6.10**	4.10	5.50	1.00	2.30	4.90	4.10

6 Conclusion

In the field of computer vision, image segmentation is considered as one of the primary processes. More so, MLT is a well-known and effective technique that widely used by various image segmentation applications due to its smooth and fast implementations. In this paper, we develop a new segmentation method depending on a hybrid of two metaheuristic algorithms, namely, Pathfinder Algorithm and Firefly algorithm. The main idea of the developed method, called PF-FA is to apply the operators of the Firefly algorithm to boost the search ability of the Pathfinder algorithm. We tested the PA-FA with different images; more so, we compared it to other metaheuristic algorithms. The evaluation results confirmed the competitive efficiency of the PF-FA which outperformed the original PF and FA algorithms, as well as other compared MH methods in terms of PSNR and SSIM. Based on this high quality, the PF-FA can be used to other optimization problems, such as job scheduling, feature selection, data clustering, and others in the future works.

Data Availability

The data used to support the findings of this study are available from the authors upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

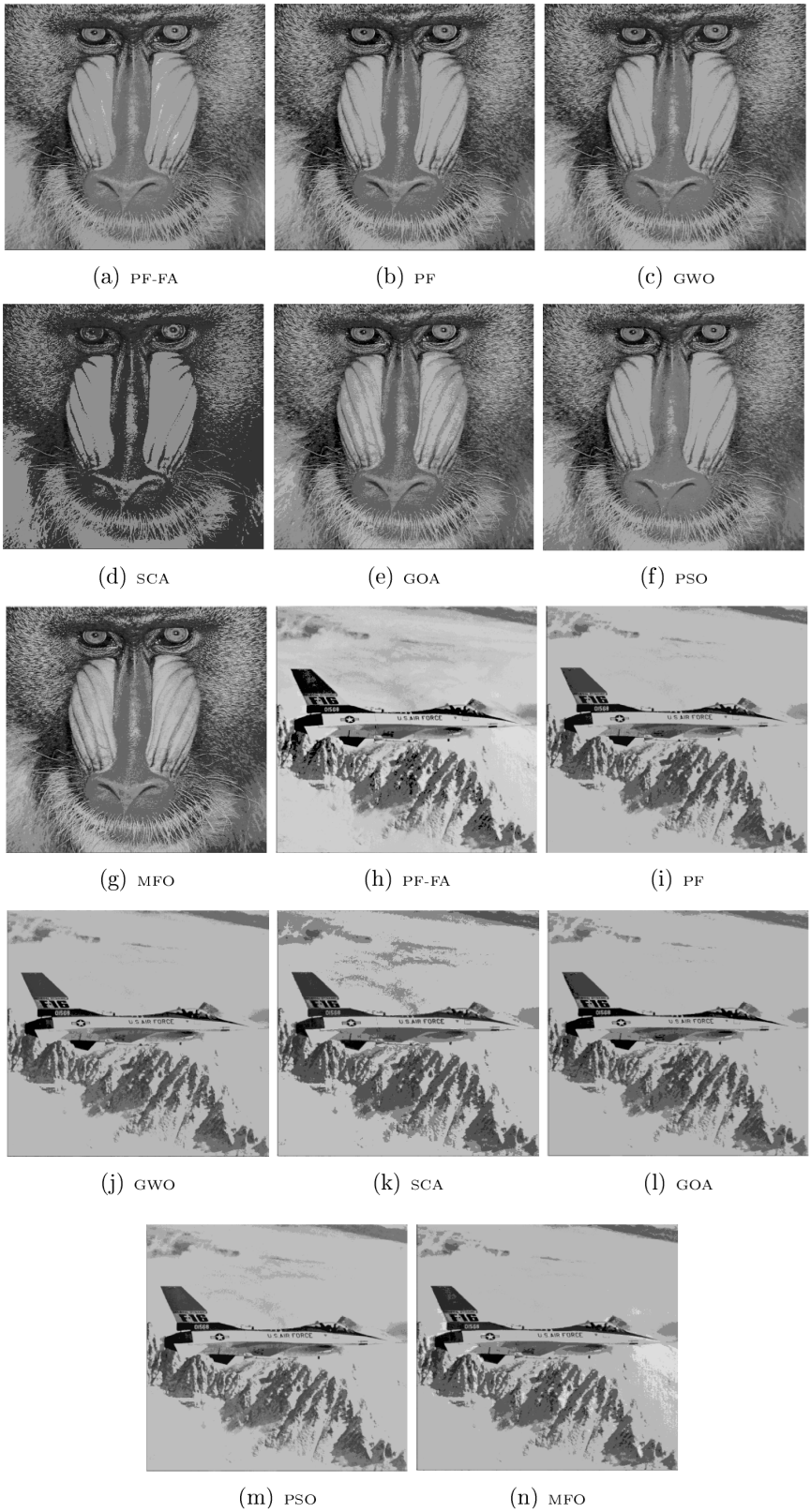


Figure 5: Segmented image I1 and I3 at threshold level 19.

References

[1] Sare Amerifar, Alireza Tavakoli Targhi, and Mohammad Mahdi Dehshibi. Iris the picture of health: Towards medical diagnosis of diseases based on iris pattern. In

2015 Tenth International Conference on Digital Information Management (ICDIM), pages 120–123. IEEE, 2015.

[2] Dalal AL-Alimi, Mohammed AA Al-qaness, Zhihua Cai, Abdelghani Dahou, Yuxiang Shao, and Sakinatu Issaka. Meta-learner hybrid models to classify hyperspectral images. *Remote Sensing*, 14(4):1038, 2022.

[3] Mohamed Abd Elaziz, Esraa Osama Abo Zaid, Mohammed AA Al-qaness, and Rehab Ali Ibrahim. Automatic superpixel-based clustering for color image segmentation using q-generalized pareto distribution under linear normalization and hunger games search. *Mathematics*, 9(19):2383, 2021.

[4] Mohamed Abd Elaziz, Diego Oliva, Ahmed A Ewees, and Shengwu Xiong. Multi-level thresholding-based grey scale image segmentation using multi-objective multi-verse optimizer. *Expert Systems with Applications*, 125:112–129, 2019.

[5] Mohamed Abd Elaziz, Esraa Osama Abo Zaid, Mohammed AA Al-qaness, and Rehab Ali Ibrahim. Automatic superpixel-based clustering for color image segmentation using q-generalized pareto distribution under linear normalization and hunger games search. *Mathematics*, 9(19):2383, 2021.

[6] MS Chelva and AK Samal. A comprehensive study of edge detection techniques in image processing applications using particle swarm optimization algorithm. *Indian J. Sci. Res*, 14(2):220–228, 2017.

[7] T Ramakrishnan and B Sankaragomathi. A professional estimate on the computed tomography brain tumor images using svm-smo for classification and mrg-gwo for segmentation. *Pattern Recognition Letters*, 94:163–171, 2017.

[8] Chengming Qi. Maximum entropy for image segmentation based on an adaptive particle swarm optimization. *Applied Mathematics & Information Sciences*, 8(6):3129, 2014.

[9] Venkatesan Rajinikanth, Nadaradjane Sri Madhava Raja, and Nilanjan Dey. *A Beginner's Guide to Multilevel Image Thresholding*. CRC Press, 2020.

[10] N Sri Madhava Raja, V Rajinikanth, and K Latha. Otsu based optimal multilevel image thresholding using firefly algorithm. *Modelling and Simulation in Engineering*, 2014, 2014.

[11] Mohammed AA Al-Qaness, Mohamed Abd Elaziz, and Ahmed A Ewees. Oil consumption forecasting using optimized adaptive neuro-fuzzy inference system based on sine cosine algorithm. *IEEE Access*, 6:68394–68402, 2018.

[12] Mohammed AA Al-qaness, Ahmed A Ewees, Hong Fan, and Mohamed Abd Elaziz. Optimized forecasting method for weekly influenza confirmed cases. *International Journal of Environmental Research and Public Health*, 17(10):3510, 2020.

[13] Abdelghani Dahou, Mohammed AA Al-qaness, Mohamed Abd Elaziz, and Ahmed Helmi. Human activity recognition in ioht applications using arithmetic optimization algorithm and deep learning. *Measurement*, 199:111445, 2022.

[14] Mohammed AA Al-qaness, Ahmed A Ewees, Hong Fan, Ayman Mutahar AlRassas, and Mohamed Abd Elaziz. Modified aquila optimizer for forecasting oil production. *Geo-spatial Information Science*, pages 1–17, 2022.

[15] Mohammed AA Al-qaness, Ahmed A Ewees, Hong Fan, Laith Abualigah, and Mohamed Abd Elaziz. Boosted anfis model using augmented marine predator algorithm with mutation operators for wind power forecasting. *Applied Energy*, 314:118851, 2022.

[16] Mohammed AA Al-Qaness, Ahmed A Ewees, Laith Abualigah, Ayman Mutahar AlRassas, Hung Vo Thanh, and Mohamed Abd Elaziz. Evaluating the applications of dendritic neuron model with metaheuristic optimization algorithms for crude-oil-production forecasting. *Entropy*, 24(11):1674, 2022.

[17] Essam H Houssein, Kashif Hussain, Laith Abualigah, Mohamed Abd Elaziz, Waleed Alomoush, Gaurav Dhiman, Youcef Djenouri, and Erik Cuevas. An improved opposition-based marine predators algorithm for global optimization and multilevel thresholding image segmentation. *Knowledge-Based Systems*, 229:107348, 2021.

[18] Mohamed Abd Elaziz, Songfeng Lu, and Sibo He. A multi-leader whale optimization algorithm for global optimization and image segmentation. *Expert Systems with Applications*, 175:114841, 2021.

[19] Mohamed Abd Elaziz, Neggaz Nabil, Reza Moghdani, Ahmed A Ewees, Erik Cuevas, and Songfeng Lu. Multilevel thresholding image segmentation based on improved volleyball premier league algorithm using whale optimization algorithm. *Multimedia Tools and Applications*, 80(8):12435–12468, 2021.

[20] Shikai Wang, Heming Jia, and Xiaoxu Peng. Modified salp swarm algorithm based multilevel thresholding for color image segmentation. *Math Biosci Eng*, 17:700–724, 2020.

[21] Soham Sarkar, Swagatam Das, and Sheli Sinha Chaudhuri. Hyper-spectral image segmentation using rényi entropy based multi-level thresholding aided with differential evolution. *Expert systems with applications*, 50:120–129, 2016.

[22] Dalia Yousri, Amr M AbdelAty, Mohammed AA Al-qaness, Ahmed A Ewees, Ahmed G Radwan, and Mohamed Abd Elaziz. Discrete fractional-order caputo method to overcome trapping in local optima: Manta ray foraging optimizer as a case study. *Expert Systems with Applications*, 192:116355, 2022.

[23] Genyun Sun, Aizhu Zhang, Yanjuan Yao, and Zhenjie Wang. A novel hybrid algorithm of gravitational search algorithm with genetic algorithm for multi-level thresholding. *Applied Soft Computing*, 46:703–730, 2016.

[24] V Rajinikanth, Seifedine Kadry, Rubén González Crespo, and Elena Verdú. A study on rgb image multi-thresholding using kapur/tsallis entropy and moth-flame algorithm. *International Journal of Interactive Multimedia & Artificial Intelligence*, 7(2), 2021.

[25] Simrandeep Singh, Nitin Mittal, Diksha Thakur, Harbinder Singh, Diego Oliva, and Anton Demin. Nature and biologically inspired image segmentation techniques. *Archives of Computational Methods in Engineering*, pages 1–28, 2021.

[26] Essam H Houssein, Bahaa El-din Helmy, Diego Oliva, Pradeep Jangir, M Premkumar, Ahmed A Elngar, and Hassan Shaban. An efficient multi-thresholding based covid-19 ct images segmentation approach using an improved equilibrium optimizer. *Biomedical Signal Processing and Control*, 73:103401, 2022.

[27] Hamza Yapici and Nurettin Cetinkaya. A new meta-heuristic optimizer: Pathfinder algorithm. *Applied Soft Computing*, 78:545–568, 2019.

[28] Xiangbo Qi, Zhonghu Yuan, and Yan Song. A hybrid pathfinder optimizer for unconstrained and constrained optimization problems. *Computational Intelligence and Neuroscience*, 2020, 2020.

[29] Xin-She Yang. Firefly algorithms for multimodal optimization. In *International symposium on stochastic algorithms*, pages 169–178. Springer, 2009.

[30] Xin-She Yang and Xingshi He. Firefly algorithm: recent advances and applications. *International journal of swarm intelligence*, 1(1):36–50, 2013.

[31] Janmenjoy Nayak, Kanithi Vakula, Paidi Dinesh, and Bighnaraj Naik. Applications and advancements of firefly algorithm in classification: An analytical perspective. In *Computational Intelligence in Pattern Recognition*, pages 1011–1028. Springer, 2020.

[32] Lokesh Jain and Rahul Katarya. Discover opinion leader in online social network using firefly algorithm. *Expert Systems with Applications*, 122:1–15, 2019.

[33] Jian Zhou, Atefeh Nekouie, Chelang A Arslan, Binh Thai Pham, and Mahdi Hasanipanah. Novel approach for forecasting the blast-induced aop using a hybrid fuzzy system and firefly algorithm. *Engineering with Computers*, pages 1–10, 2019.

[34] Aravind Rajagopalan, Devesh R Modale, and Radha Senthilkumar. Optimal scheduling of tasks in cloud computing using hybrid firefly-genetic algorithm. In *Advances in decision sciences, image processing, security and computer vision*, pages 678–687. Springer, 2020.

[35] Xin-She Yang. Firefly algorithm and its variants in digital image processing. *Applications of Firefly Algorithm and its Variants: Case Studies and New Developments*, 2019.

[36] Mohammed AA Al-qaness, Ahmed A Ewees, and Mohamed Abd Elaziz. Modified whale optimization algorithm for solving unrelated parallel machine scheduling problems. *Soft Computing*, pages 1–13, 2021.

[37] Nilanjan Dey, Jyotismita Chaki, Luminița Moraru, Simon Fong, and Xin-She Yang. Firefly algorithm and its variants in digital image processing: A comprehensive review. *Applications of firefly algorithm and its variants*, pages 1–28, 2020.

[38] Sayan Chakraborty, Nilanjan Dey, Sourav Samanta, Amira S Ashour, and Valentina E Balas. Firefly algorithm for optimized nonrigid demons registration. In *Bio-inspired computation and applications in image processing*, pages 221–237. Elsevier, 2016.

[39] Venkatesan Rajinikanth, Nilanjan Dey, Ergina Kavallieratou, and Hong Lin. Firefly algorithm-based kapur's thresholding and hough transform to extract leukocyte section from hematological images. In *Applications of firefly algorithm and its variants*, pages 221–235. Springer, 2020.

[40] Husein S Naji Alwerfali, Mohamed Abd Elaziz, Mohammed AA Al-Qaness, Aaqif Afzaal Abbasi, Songfeng Lu, Fang Liu, and Li Li. A multilevel image thresholding based on hybrid salp swarm algorithm and fuzzy entropy. *IEEE Access*, 7:181405–181422, 2019.

[41] Naji Alwerfali, S Husein, Mohammed AA Al-qaness, Mohamed Abd Elaziz, Ahmed A Ewees, Diego Oliva, and Songfeng Lu. Multi-level image thresholding based on modified spherical search optimizer and fuzzy entropy. *Entropy*, 22(3):328, 2020.

[42] Jun Qin, ChuTing Wang, and GuiHe Qin. A multilevel image thresholding method based on subspace elimination optimization. *Mathematical Problems in Engineering*, 2019, 2019.

[43] Diego Oliva, Salvador Hinojosa, Erik Cuevas, Gonzalo Pajares, Omar Avalos, and Jorge Gálvez. Cross entropy based thresholding for magnetic resonance brain images using crow search algorithm. *Expert Systems with Applications*, 79:164–180, 2017.

[44] Musrrat Ali, Chang Wook Ahn, and Millie Pant. Multi-level image thresholding by synergetic differential evolution. *Applied Soft Computing*, 17:1–11, 2014.

[45] M. A. Elaziz, A. A. Ewees, D. Yousri, H. S. N. Alwerfali, Q. A. Awad, S. Lu, and M. A. A. Al-qaness. An improved marine predators algorithm with fuzzy entropy for multi-level thresholding: Real world example of covid-19 ct image segmentation. *IEEE Access*, pages 1–1, 2020.

[46] Yourui Huang and Shuang Wang. Multilevel thresholding methods for image segmentation with otsu based on qpso. In *2008 Congress on Image and Signal Processing*, volume 3, pages 701–705. IEEE, 2008.

[47] Mohamed Abd El Aziz, Ahmed A Ewees, and Aboul Ella Hassanien. Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. *Expert Systems with Applications*, 83:242–256, 2017.

[48] Seyed Jalaleddin Mousavirad and Hossein Ebrahimpour-Komleh. Human mental search-based multilevel thresholding for image segmentation. *Applied Soft Computing*, 2019.

[49] Bahriye Akay. A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding. *Applied Soft Computing*, 13(6):3066–3091, 2013.

[50] Ahmed A Ewees, Mohamed Abd Elaziz, Mohammed AA Al-Qaness, Hassan A Khalil, and Sunghwan Kim. Improved artificial bee colony using sine-cosine algorithm for multi-level thresholding image segmentation. *IEEE Access*, 8:26304–26315, 2020.

[51] Lifang He and Songwei Huang. Modified firefly algorithm based multilevel thresholding for color image segmentation. *Neurocomputing*, 240:152–174, 2017.

[52] Mohamed Abd Elaziz, Ahmed A Ewees, Dalia Yousri, Husein S Naji Alwerfali, Qamar A Awad, Songfeng Lu, and Mohammed AA Al-Qaness. 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, 2020.

[53] Mohamed Abd Elaziz, Dalia Yousri, Mohammed AA Al-qaness, Amr M AbdelAty, Ahmed G Radwan, and Ahmed A Ewees. A grunwald–letnikov based manta ray foraging optimizer for global optimization and image segmentation. *Engineering Applications of Artificial Intelligence*, 98:104105, 2021.

[54] Xiaowei Chen, Hui Huang, Ali Asghar Heidari, Chuanyin Sun, Yinqiu Lv, Wenyong Gui, Guoxi Liang, Zhiyang Gu, Huiling Chen, Chengye Li, et al. An efficient multilevel thresholding image segmentation method based on the slime mould algorithm with bee foraging mechanism: A real case with lupus nephritis images. *Computers in Biology and Medicine*, 142:105179, 2022.

[55] Qingxin Liu, Ni Li, Heming Jia, Qi Qi, and Laith Abualigah. Modified remora optimization algorithm for global optimization and multilevel thresholding image segmentation. *Mathematics*, 10(7):1014, 2022.

[56] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[57] Mario Versaci and Francesco Carlo Morabito. Image edge detection: A new approach based on fuzzy entropy and fuzzy divergence. *International Journal of Fuzzy Systems*, 23(4):918–936, 2021.

[58] Shubham Mahajan, Nitin Mittal, and Amit Kant Pandit. Image segmentation using multilevel thresholding based on type ii fuzzy entropy and marine predators algorithm. *Multimedia Tools and Applications*, 80(13):19335–19359, 2021.

[59] Xin-She Yang. Firefly algorithm, levy flights and global optimization. In *Research and development in intelligent systems XXVI*, pages 209–218. Springer, 2010.

[60] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: From error measurement to structural similarity. *IEEE transactions on image processing*, 13(1), 2004.

[61] P. I. Yin. Multilevel minimum cross entropy threshold selection based on particle swarm optimization. *Applied Mathematics and Computation*, 184:503–892, 2007.