ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

MULTILEVEL THRESHOLDING OF COLOR IMAGE SEGMENTATION USING MEMORY-BASED GREY WOLF OPTIMIZER WITH OTSU METHOD, KAPUR, AND M.MASI ENTROPY

I Made Satria Bimantara[1*], Anny Yuniarti[2]

[1,2]Departemen Teknik Informatika, Institut Teknologi Sepuluh Nopember

email: satriabimantara.md@gmail.com[1], anny@if.its.ac.id[2]

Abstract

Determining the optimal threshold value for image segmentation has become more attention in recent years because of its varied uses. Otsu-based thresholding methods, minimum cross entropy, and Kapur entropy are efficient for solving bi-level thresholding image segmentation problems (BL-ISP), but not with multi-level thresholding image segmentation problems (ML-ISP). The main problem is exponentially increasing computational complexity. This study uses the memory-based Gray Wolf Optimizer (mGWO) to determine the optimal threshold value for solving ML-ISP on RGB images. The mGWO method is a variant of the standard grey wolf optimizer (GWO) that utilizes the best track record of each individual grey wolf for the global exploration and local exploitation phases of the problem solution space. The solution candidates are represented by each grey wolf using the image intensity values and optimized according to mGWO characteristics. Three objective functions, namely the Otsu method, Kapur Entropy, and M.Masi Entropy are used to evaluate the solutions generated in the optimization process. The GridSearch method is used to determine the optimal parameter combination of each method based on 10 training images. Evaluation of the performance of the mGWO method was measured using several benchmark images and compared with five standard swarm intelligence (SI) methods as benchmarks. Analysis of the results was carried out qualitatively and quantitatively based on the average PSNR, RMSE, SSIM, UQI, fitness value, and CPU processing time from 30 tests. The results were analyzed further with the Wilcoxon signed-rank test. The experimental results show that the performance of the mGWO method outperforms the benchmark method in most experiments and metrics. The mGWO variant also proved to be superior to the standard GWO in resolving multi-level color image segmentation problems. The mGWO performance results are also compared with other state-of-the-art SI methods in solving ML-ISP on grayscale images and was able to outperform those methods in most experiments when combined with the Otsu method and Kapur Entropy.

Keywords : Multilevel Thresholding, Color Image Segmentation, Memory-based Grey Wolf Optimizer, Otsu Method, Kapur Entropy, M.Masi Entropy

Received: 06-06-2023 | **Revised:** 24-07-2023 | **Accepted:** 26-07-2023
DOI: https://doi.org/10.23887/janapati.v12i2.62874

INTRODUCTION

Image segmentation plays an important role in advanced image processing [1], [2] and computer vision [3], [4]. Image segmentation has been utilized in terms of satellite imaging [5], automatic target recognition [3], [6], [7], and medical image analysis [8]–[11]. Good image segmentation can determine the performance of advanced image analysis [6]. The thresholding method is the most commonly used approach to perform image segmentation [3]–[8], [12]–[18]. The thresholding method is commonly used for image segmentation because of its simplicity and efficiency [3], [6], [7], [15], [19].

The main objective of the thresholding method is to determine the optimal threshold value so that it can divide the image into several regions based on the pixel intensity value of an image. Thresholding methods can be divided into two based on the number of values taken from the histogram of an image, namely bi-level thresholding and multilevel thresholding [3], [6], [18]. When the selected threshold value is one, it is known as bi-level thresholding, whereas when more than one threshold value is selected, it is known as multilevel thresholding [6].

The non-parametric approach to the thresholding method which uses certain criteria to obtain the optimum threshold value has been proven to be better for solving BL-ISP [6]. Otsu's between class variance, minimum cross entropy, Kapur entropy are some of the criteria commonly used to complete BL-ISP [3], [6], [13]. Although these criteria have proven to be very efficient in solving BL-ISP on grayscale images, this approach has proven to be inefficient [20] and impractical [3] to be used to solve ML-ISP. Computational complexity will increase exponentially [7], [14], [15], [20] as the number of specified thresholds increases [3], [4], [6], [7], [14], [15] and performance levels tend to decrease [6]. This is because all possible threshold value pairs must be tried thoroughly in order to meet the specified criteria. Therefore, determining the optimal threshold value at ML-ISP in a short time is a challenge [7].

Determining the optimal threshold value for ML-ISP is included in the NP-hard combinatorial optimization problem [6], [7], [13], [18] and has been a challenge in the last few decades [7], [18]. Several approaches have been proposed to solve this problem, including using an SI-based metaheuristic optimization algorithm. The metaheuristic algorithm is proven to be more efficient in finding the optimal threshold value for solving ML-ISP when compared to exhaustive search [3], [6], [13], [15]. SI-based metaheuristic algorithms have been widely used to reduce computational complexity and have proven to be more accurate in solving ML-ISP when compared to evolutionary algorithms [3].

The Otsu, Kapur Entropy, and M.Masi Entropy methods have each been used as an objective function to solve ML-ISP on grayscale images with various proposed SI-based methods. Otsu's method with GA-PSO [20], MFO [18], KHO [15], WOA [18], GWO [14] and improved WOA [4] have been tried before. Kapur Entropy with WOA-SMA [8], multistage hybrid SI optimization algorithm [6], DA [3], GWO [14] and KHO [15] have been tested before. M.Masi Entropy with GWO [7] and PSO [3] have been tested before.

Problems arise when there is no one optimization method that can provide the same solution for all optimization problems referring to the No Free Lunch (NFL) theory [7]. Several previous studies that utilized the SI method to solve ML-ISP [3], [4], [6]–[9], [15], [20], [21] only tested the method they proposed using one function. just be objective. In fact, it is important to test the robustness and consistency of the performance of the proposed method against different objective functions. Thus, it can be guaranteed that the performance of the proposed SI method is stable against several objective functions used [7].

In addition, many studies that apply the SI method to solve ML-ISP only focus on grayscale images as their test images [2], [3], [6]–[9], [14], [15], [17], [18], [20], [21]. In fact, color images can provide a better description of an image than grayscale images [13]. Research related to the completion of ML-ISP using SI on color images is very little found [4], [13]. Ma dan Yue (2022) [4] have implemented a variant of WOA to solve ML-ISP on color images. However, this study did not explain in detail the steps to complete ML-ISP on color images with the proposed WOA variant. The explanation given in this study is only based on grayscale images using the Otsu method. Borjigin dan Sahoo (2019) [13] have implemented PSO with the objective function Tsallis-Havrda-Charvát *Entropy* to solve ML-ISP on color images.

Borjigin and Sahoo's research [13] has inspired this study to adapt ML-ISP solutions to color images using mGWO [21] and GWO [22] as well as three different objective functions to measure the performance stability of the two methods for solving ML-ISP. The SI method that has been applied in previous studies to solve ML-ISP still has some drawbacks, such as early convergence, stuck at local optimum values, and low convergence speed [4], [6], [20]. Therefore, images with good segmentation cannot be obtained with the threshold values obtained [8].

mGWO and GWO are used as proposals in this study because they can balance the exploration and exploitation in solving optimization problems, so as to avoid local optimum values [7]. In addition, GWO can reduce computation time greatly when compared to other optimization methods [14]. In fact, mGWO [21] has been proven to be able to solve global optimization problems better in terms of search efficiency, solution accuracy, and convergence rate when compared to standard GWO [22].

The discussion that has been described in the previous section has motivated this research to take place. This study utilizes the GWO [22] and mGWO [21] methods to solve ML-ISP on RGB color images. Three different objective functions, namely the Otsu method, Kapur Entropy and M.Masi Entropy are used in the evaluation to see the performance stability of the two methods compared to the four SI methods as benchmarks, namely genetic algorithm (GA), particle swarm optimization (PSO), whale optimization algorithm (WOA), and slime mold algorithm (SMA).

GA is implemented because it can substantially reduce computational costs in solving ML-ISP [14]. WOA is used because it is proven to be able to provide the best results in terms of exploration capabilities [8], [18]. In addition, this method has fewer parameter configurations with a simple framework and can avoid local optimum values [18]. SMA is used because it has been proven to be significantly successful in solving optimization problems in the continuous domain when compared to other algorithms [8], [23]. PSO is implemented because it has global optimization capabilities [1], is simple [13] and can achieve convergence in a relatively short time [3], [13].

The main contributions of this research are as follows:

(1) This study proposes ML-ISP solutions for RGB color images in the mGWO and GWO frameworks besides using grayscale images.
(2) Three different objective functions namely the Otsu Method, Kapur Entropy, and M.Masi Entropy were tested on mGWO and GWO to measure the stability of their performance on ML-ISP
(3) The performance of the method implemented in this study was measured using qualitative and quantitative analysis using benchmark images from the USC-SIPI image database. Qualitative analysis was carried out by segmenting the six test images with each optimal threshold for each level. Quantitative analysis was carried out by calculating the fitness, RMSE, PSNR, SSIM, UQI, and CPU time values of each objective function.
(4) Hyperparameter tuning based on GridSearch is performed to obtain the optimal parameter combination of each SI method involved with the aim of maximizing the Otsu method.
(5) Statistical analysis using the Wilcoxon signed-rank test was used to test the significance of differences in the quantitative measurements of the GWO and mGWO methods against the benchmark method assigned to the test images.
(6) Comparing the results of the mGWO and GWO performance tests on grayscale images with other state-of-the-art methods in terms of fitness values and CPU Time (seconds).

MULTILEVEL THRESHOLDING FOR COLOR IMAGE SEGMENTATION

This section describes the definition of multilevel thresholding for mathematical image segmentation based on the Otsu, Kapur Entropy, and M.Masi Entropy methods.

Assume that there is an *i*-th 2D grayscale image as C_i^{gray} sized $R \times K$ with gray level $G = \{0,1,2,\dots,L-1\}$. The R value represents the number of rows, while the K value represents the number of columns. So, an *i*-th RGB color image as C_i^{RGB} can be defined as a function vector [13] $\vec{f_i}(x,y): R \times K: \rightarrow C_i^r \times C_i^g \times C_i^b$, such that:

$$C_i^{RGB} = [\vec{f_i}(x,y)] = [C_i^r, C_i^g, C_i^b]$$

with C_i^r, C_i^g, C_i^b each represents the red, green, and blue components (channels) of an image whose combinations can generate any displayable color [13]. Therefore, an RGB color image is a 3D array of color pixels with size $R \times K \times 3$ [13]. C_i^x notation is used to show any channel (RGB or grayscale) of an *i*-th image.

Suppose N_i is the total number of pixels in C_i^x with n_j is the number of occurrences of the jth gray level. Normalized histogram of C_i^x is a probability distribution of each $g \in G$. The probability that the jth gray level occurs at C_i^x is defined according to Equation 1. The main objective of multilevel thresholding is to find a number of *m* optimal thresholds $\{t_1, t_2, \dots, t_m\}$ so split C_i^x into the $m+1$ regions or segment that meet predetermined criteria or objective functions (Otsu Method, Kapur Entropy, or M.Masi Entropy). Suppose $m+1$ regions from C_i^x is defined as $W^{(i)} = \left\{W_0^{(i)}, W_1^{(i)}, \dots, W_m^{(i)}\right\}$ with the range of gray level values of the pixels contained in $W_j^{(i)}$ is defined according to Equation 2. $g_{(x,y)}$ value in Equation 2 is the gray level of the pixels in the (x,y) coordinates from a 2D image C_i^x.

$$P_j^{(i)} = \frac{n_j}{N_i}, \left(0 \le P_j^{(i)} \le 1\right) \wedge \left(\sum_{k=0}^{L-1} P_k^{(i)} = 1\right) \quad (1)$$

$$\begin{aligned} W_0^{(i)} &= \{g_{(x,y)} \in C_i \mid 0 \le g_{(x,y)} \le t_1 - 1\} \\ W_1^{(i)} &= \{g_{(x,y)} \in C_i \mid t_1 \le g_{(x,y)} \le t_2 - 1\} \\ W_2^{(i)} &= \{g_{(x,y)} \in C_i \mid t_2 \le g_{(x,y)} \le t_3 - 1\} \\ &\dots \\ W_m^{(i)} &= \{g_{(x,y)} \in C_i \mid t_m \le g_{(x,y)} \le L - 1\} \end{aligned} \quad (2)$$

Otsu Method

Suppose $F_{Otsu}(t_1, t_2, \dots, t_m)$ is a function that accepts several *m* thresholds $\{t_1, t_2, \dots, t_m\}$ so that split C_i^x into $m+1$ regions according to Otsu's criteria. C_i^x image can be segmented properly using a threshold $\{t_1, t_2, \dots, t_m\}$ when it produces the maximum $F_{Otsu}(t_1, t_2, \dots, t_m)$ value among all the existing *m* thresholds combinations. The Otsu method maximizes the value between class variance according to Equation 3. σ_j value is calculated using Equation 4. ω_j value represents the sum of the probabilities of selecting pixels in the $W_j^{(i)}$ region which is calculated using Equation 5. μ_j value is the average pixel intensity value in the $W_j^{(i)}$ region which is calculated using Equation 6. μ_T value is the average pixel intensity value in C_i^x which is calculated using Equation 7.

$$F_{Otsu}(t_1, t_2, \ldots, t_m) = \sum_{j=0}^{m} \sigma_j = \sigma_0 + \sigma_1 + \cdots + \sigma_m \quad (3)$$

$$\sigma_0 = \omega_0(\mu_0 - \mu_T)^2$$
$$\sigma_1 = \omega_1(\mu_1 - \mu_T)^2 \quad (4)$$
$$\ldots$$
$$\sigma_m = \omega_m(\mu_m - \mu_T)^2$$

$$\omega_j = \sum_{z=t_j}^{t_{(j+1)}-1} P_z^{(i)} \quad (5)$$

$$\mu_j = \sum_{z=t_j}^{t_{(j+1)}-1} j \times \left(\frac{P_z^{(i)}}{\omega_z}\right) \quad (6)$$

$$\mu_T = \sum_{j=0}^{m} (\omega_j \mu_j) \quad (7)$$

Kapur Entropy

Suppose $F_{Kapur}(t_1, t_2, \ldots, t_m)$ is a function that accepts several *m* thresholds $\{t_1, t_2, \ldots, t_m\}$ so that split the C_i^x into $m+1$ regions according to Kapur Entropy criteria. C_i^x image can be segmented properly using a threshold $\{t_1, t_2, \ldots, t_m\}$ when it produces the maximum $F_{Kapur}(t_1, t_2, \ldots, t_m)$ value among all the existing *m* thresholds combinations. The Kapur Entropy maximizes the variance value in $W^{(i)}$ by using the entropy value according to Equation 8. En_j value is the entropy value of the $W_j^{(i)}$ region which is calculated using Equation 9.

$$F_{Kapur}(t_1, t_2, \ldots, t_m) = \sum_{j=0}^{m} En_j = En_0 + En_1 + \cdots + En_m \quad (8)$$

$$En_0 = -\sum_{z=0}^{t_1-1} \frac{P_z^{(i)}}{w_0} \ln\left(\frac{P_z^{(i)}}{w_0}\right), w_0 = \sum_{z=0}^{t_1-1} P_z^{(i)}$$
$$En_1 = -\sum_{z=t_1}^{t_2-1} \frac{P_z^{(i)}}{w_1} \ln\left(\frac{P_z^{(i)}}{w_1}\right), w_1 = \sum_{z=t_1}^{t_2-1} P_z^{(i)} \quad (9)$$
$$\ldots$$
$$En_m = -\sum_{z=t_m}^{L-1} \frac{P_z^{(i)}}{w_m} \ln\left(\frac{P_z^{(i)}}{w_m}\right), w_m = \sum_{z=t_m}^{L-1} P_z^{(i)}$$

M.Masi Entropy

Suppose $F_{Masi}(t_1, t_2, \ldots, t_m)$ is a function that accepts several *m* thresholds $\{t_1, t_2, \ldots, t_m\}$ so that split the C_i^x into $m+1$ regions according to M.Masi Entropy criteria. C_i^x image can be segmented properly using a threshold $\{t_1, t_2, \ldots, t_m\}$ when it produces the maximum $F_{Masi}(t_1, t_2, \ldots, t_m)$ value among all the existing *m* thresholds combinations. The M.Masi Entropy maximizes the variance value in $W^{(i)}$ by using the entropy value according to Equation 10 [3]. MME_j value is the M.Masi Entropy for $W_j^{(i)}$ which is calculated using Equation 11. φ_j value on MME_j calculation is calculated using Equation 12. The value of α in Equation 11 can be determined through experiments with a value range of -1 to 3 intervals of 0.1 [3]. The value of $\alpha < 1$ has been proven to produce good and stable segmented image quality [3].

$$F_{Masi}(t_1, t_2, \ldots, t_m) = \sum_{j=0}^{m} MME_j = MME_0 + MME_1 + \cdots + MME_m \quad (10)$$

$$MME_0 = \frac{\log(1-(1-\alpha)\times\varphi_0)}{(1-\alpha)}$$
$$MME_1 = \frac{\log(1-(1-\alpha)\times\varphi_1)}{(1-\alpha)} \quad (11)$$
$$\ldots$$
$$MME_m = \frac{\log(1-(1-\alpha)\times\varphi_m)}{(1-\alpha)}$$

$$\varphi_0 = \sum_{z=0}^{t_1-1} \frac{P_z^{(i)}}{w_0} \log\left(\frac{P_z^{(i)}}{w_0}\right), w_0 = \sum_{z=0}^{t_1-1} P_z^{(i)}$$
$$\varphi_1 = \sum_{z=t_1}^{t_2-1} \frac{P_z^{(i)}}{w_1} \log\left(\frac{P_z^{(i)}}{w_1}\right), w_1 = \sum_{z=t_1}^{t_2-1} P_z^{(i)} \quad (12)$$
$$\ldots$$
$$\varphi_m = \sum_{z=t_m}^{L-1} \frac{P_z^{(i)}}{w_m} \log\left(\frac{P_z^{(i)}}{w_m}\right), w_m = \sum_{z=t_m}^{L-1} P_z^{(i)}$$

MATERIAL AND METHODS

This section describes the steps taken to answer the research objectives along with the dataset used in this study as shown in Figure 1.

Benchmark Images

This study uses a standard benchmark dataset from the USC-SIPI image database and Berkeley BSDS 300. There are 10 images as training data and 6 images as test data. The training data is a combination of several image files selected from the train and test folders on the Berkeley BSDS 300 source. These files were chosen because they have multimodal histogram characteristics so that the optimal hyperparameters of each model can be selected objectively. The training data is used for the SI model development process for ML-ISP including the hyperparameter tuning process for each model. The test data that is used in this study are Airplane F16, Lena, Man, Mandrill (baboon), Peppers, and Sailboat on lake. The test data is used to evaluate the performance of the method used. Table 1 displays the names of the image files used as training data, while Figure 2 shows the pixel intensity histogram of the test data.

Grey Wolf Optimizer (GWO)

GWO is a method proposed by Mirjaili et al (2014) [22]. The GWO optimization method is inspired by social intelligence in hunting prey

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

and the social hierarchy of the gray wolf (*Canis lupus*) [14], [22], [24]. Gray wolves live in groups with 5 – 12 wolves in each group [14]. There are four levels or hierarchies in one group, namely alpha (α), betha (β), delta (δ), and omega (ω) wolves.

Alpha wolves are the highest level in this group whose job is to make decisions about hunting, where to sleep, when to wake up and so on. This wolf dominates the pack. The beta wolf is the second level after the alpha whose job is to help the alpha wolf make decisions and other group activities. The deltha wolf is the third level after betha whose duties are scout, caretaker, hunter, guard and scout. The omega wolf is the lowest level of this group which must submit to orders from all other dominant wolves [24].

Social hierarchy on GWO

The social hierarchy in a pack of gray wolves is divided into α, β, δ, and ω wolves. In the mathematical modeling of the GWO method, the α, β, and δ wolves each represent the first, second and third best solutions [24]. The optimization process in this method is guided by the solutions produced by the three wolves, while the remaining ω wolves follow them [14].

Encircling of prey in GWO

During the hunt for prey, the gray wolves surround their prey. This prey encirclement process is modeled mathematically according to Equations 13 and 14. The t-th iteration is expressed by the value (t). The values of $\vec{A}$ and $\vec{C}$ are calculated using Equations 15 and 16, respectively. $\overrightarrow{X_p^{(t)}}$ represents the position vector of the prey at iteration (t), whereas $\overrightarrow{X_j^{(t)}}$ represents the vector position of the gray wolf in iteration (t). Vector $\vec{a}$ decreases linearly in each iteration starting from 2 to 0 which is calculated using Equation 17 [21]. The maximum number of iterations is represented by T. The two random vectors in the interval [0,1] are represented by $\vec{r_1}$ and $\vec{r_2}$.

Table 1. List Files as Train Data

File name	Dimension (pixel)	Color Space
101087.jpg	321 x 481	
3096.jpg		
106020.jpg		
112082.jpg		
113016.jpg		
118035.jpg	481 x 321	Gray
119082.jpg		
189003.jpg		
231015.jpg		
296059.jpg		

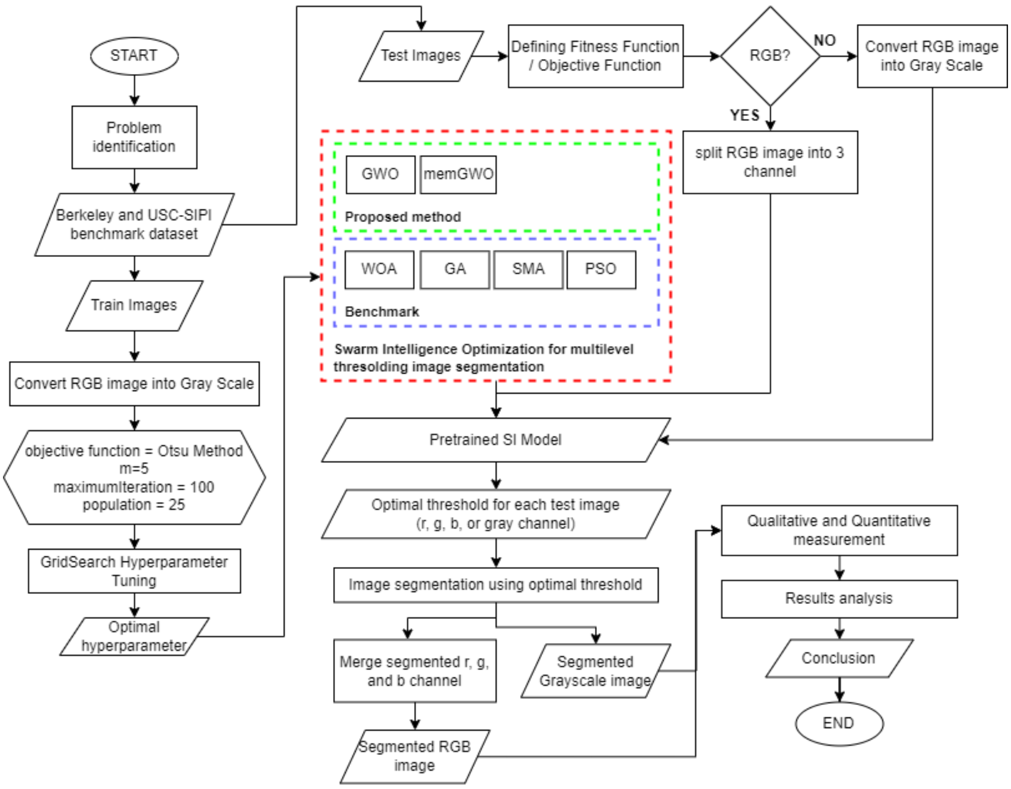


Figure 1. Research Flowchart

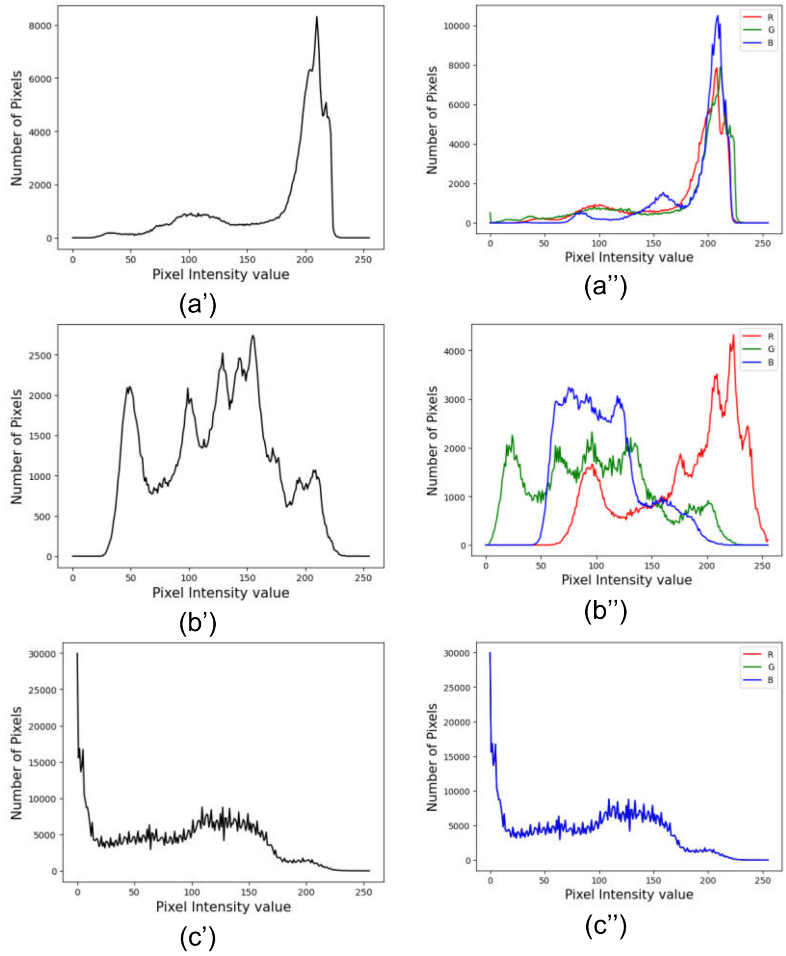


Figure 2. The benchmark (RGB) image data used for evaluating the performance of the method was obtained from the USC-SIPI image database. Each image in (a) – (f) has a size of 512 x 512 pixels along with their respective grayscale histograms (a') – (f') and RGB histograms (a'') – (f'') with (a) Airplane F-16, (b) Lena, (c) Man, (d) Mandrill (Baboon), (e) Peppers, and (f) Sailboat on lake.

The gray wolf moves in hypercubes or hyper-spheres around the best solution (α wolf) for a solution space of *m* dimension [14], [24].

$$\vec{D} = \left|\vec{C} \cdot \overrightarrow{X_p^{(t)}} - \overrightarrow{X_J^{(t)}}\right| \quad (13)$$

$$\overrightarrow{X_J^{(t+1)}} = \overrightarrow{X_p^{(t)}} - \vec{A} \cdot \vec{D} \quad (14)$$

$$\vec{A} = 2\vec{a} \cdot \overrightarrow{r_1} - \vec{a} \quad (15)$$

$$\vec{C} = 2 \cdot \overrightarrow{r_2} \quad (16)$$

$$\vec{a} = 2\left(1 - \frac{t}{T}\right) \quad (17)$$

Hunting process in GWO

The α, β, and δ wolves guide the hunting process of all wolves in a group. These three wolves are assumed to have better knowledge of the potential location of a prey (optimal solution). Hence all the other wolves updated their positions based on the information on the three wolves. The mathematical model for the prey hunting process is according to Equations 18 and 19. The position vectors of the α, β, and δ wolves in (t) iteration are represented by $\overrightarrow{X_\alpha^{(t)}}$, $\overrightarrow{X_\beta^{(t)}}$, and $\overrightarrow{X_\delta^{(t)}}$ respectively. Vektor $\overrightarrow{A_\iota}$ dan $\overrightarrow{C_\iota}$ are calculated using Equations 15 and 16 with different sets of random numbers, respectively. The position vector of each gray wolf is updated using Equation 20.

$$\begin{aligned} \overrightarrow{D_\alpha} &= \left|\overrightarrow{C_1} \cdot \overrightarrow{X_\alpha^{(t)}} - \overrightarrow{X_J^{(t)}}\right| \\ \overrightarrow{D_\beta} &= \left|\overrightarrow{C_2} \cdot \overrightarrow{X_\beta^{(t)}} - \overrightarrow{X_J^{(t)}}\right| \\ \overrightarrow{D_\delta} &= \left|\overrightarrow{C_3} \cdot \overrightarrow{X_\delta^{(t)}} - \overrightarrow{X_J^{(t)}}\right| \end{aligned} \quad (18)$$

$$\begin{aligned} \overrightarrow{X_1^{(t)}} &= \overrightarrow{X_\alpha^{(t)}} - \overrightarrow{A_1} \cdot \overrightarrow{D_\alpha} \\ \overrightarrow{X_2^{(t)}} &= \overrightarrow{X_\beta^{(t)}} - \overrightarrow{A_2} \cdot \overrightarrow{D_\beta} \\ \overrightarrow{X_3^{(t)}} &= \overrightarrow{X_\delta^{(t)}} - \overrightarrow{A_3} \cdot \overrightarrow{D_\delta} \end{aligned} \quad (19)$$

$$\overrightarrow{X_J^{(t+1)}} = \frac{\overrightarrow{X_1^{(t)}} + \overrightarrow{X_2^{(t)}} + \overrightarrow{X_3^{(t)}}}{3} \quad (20)$$

Attacking the prey (exploitation) in GWO

Vector $\vec{A}$ and vector $\vec{C}$ in the above equation are used to store the exploration and

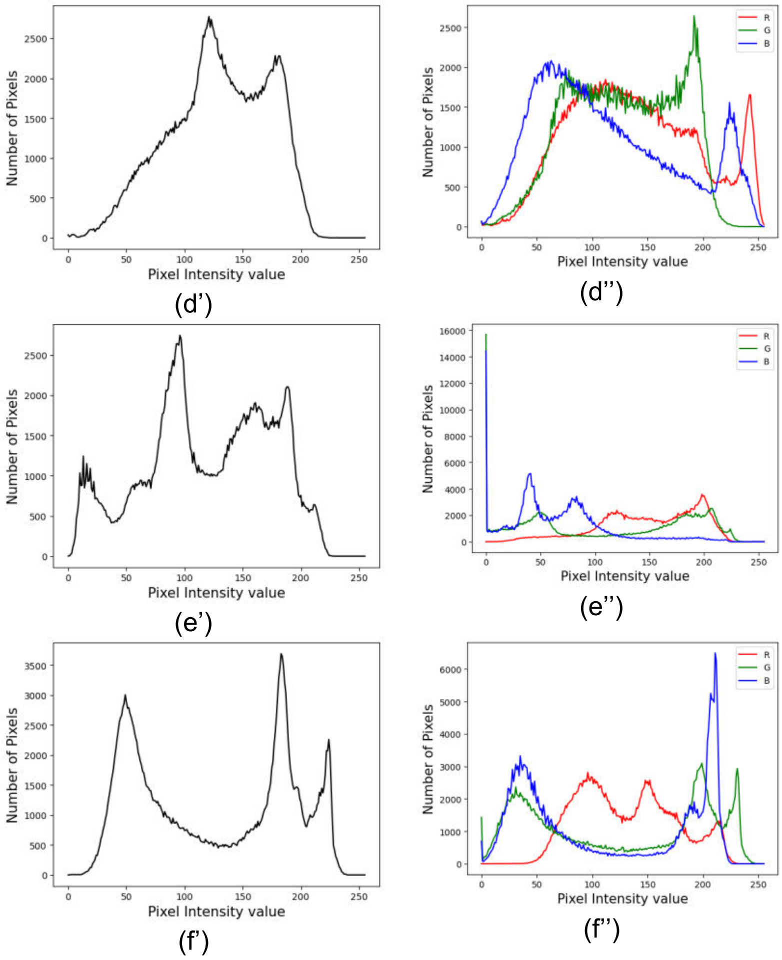


Figure 2. (Continued).

exploitation abilities of wolves [21]. The process of hunting prey from wolves is completed by attacking prey by wolves. This process is modeled by reducing the value of the vector $\vec{a}$ in the range 2 to 0 during the iteration process. The fluctuation range of $\vec{A}$ will decrease if $\vec{a}$ is also decrease. When $\left|\overrightarrow{A^{(t)}}\right| < 1$ and/or $\left|\overrightarrow{C^{(t)}}\right| < 1$, then a pack of wolves attacks the prey [14], [21]. When $\left|\overrightarrow{A^{(t)}}\right| > 1$ and/or $\left|\overrightarrow{C^{(t)}}\right| > 1$, then the new search area explored by the pack of wolves can avoid being stuck at the local optimum [14], [21]. Exploration and exploitation operators are balanced by using a value transition from vector $\vec{a}$ at each (t) iteration.

GWO for solving ML-ISP

The GWO algorithm is used in this study to find the optimal threshold value (represented by the position of the wolf) at the mth level so that it can be used to segment images with a maximum of $m + 1$ regions. One wolf in GWO represents a solution for which you want to find the optimal value. The input of this process is the histogram of the image to be segmented, while the output of this process is the optimal position vector of the wolf as X^* which represents the optimal threshold value.

The positional vector representation of each *i-th* wolf is written as $\overrightarrow{X_i}$ which is initialized according to Equation 22. The total number of wolves initialized is written as N. The value $x_{(i,j)} \in \overrightarrow{X_i^{(t)}}$ is the threshold value represented by the gray level of an image. At the beginning of the iteration $(t = 0)$, the fitness of all wolves is calculated with a predetermined objective function, namely using the Otsu method, Kapur Entropy, or M.Masi Entropy respectively with Equations 3, 8 or 10. Three wolves with fitness values optimal is then defined as $\overrightarrow{X_\alpha^{(0)}}$, $\overrightarrow{X_\beta^{(0)}}$, dan $\overrightarrow{X_\delta^{(0)}}$. For each iteration during a predetermined maximum iteration *T*, each wolf updates its position taking into account the positions of the three α, β, and δ wolves using Equation 20. Before updating the positions, vector $\overrightarrow{A_i}$, vector $\overrightarrow{C_i}$, and vector $\vec{a}$ is calculated using Equations 15, 16, and 17 respectively.

Algorithm 1: Implementing GWO for finding *m* optimal of thresholds using Otsu Method, Kapur Entropy, or M.Masi Entropy
Input:
$N_{wolf} \leftarrow$ Grey wolf population size
$T_{max} \leftarrow$ total number of maximum iteration
$F_{func} \leftarrow$ type of objective function either Otsu Method, Kapur Entropy, or M.Masi Entropy
Output: Optimal individual $\overrightarrow{X_{\alpha}^{(T_{max})}}$ and best fitness values $Fit\left(\overrightarrow{X_{\alpha}^{(T_{max})}}\right)$
Initialization:
$GreyWolfs \leftarrow$ Initialize 2D matrix of N_{wolf} individual grey wolf position using Equation 22.
Calculate each individual fitness value using F_{func}
Find $\overrightarrow{X_{\alpha}^{(0)}}$, $\overrightarrow{X_{\beta}^{(0)}}$, dan $\overrightarrow{X_{\delta}^{(0)}}$
// GWO optimization steps for ML-ISP
FOR t in range(0, T_{max}) DO
Update $\vec{a}$ using Equation 17.
FOR individual in $GreyWolfs$ DO
Update $\overrightarrow{A_x}$ and $\overrightarrow{C_x}$ using Equation15 and 16.
Calculate $\overrightarrow{X_1^{(t)}}, \overrightarrow{X_2^{(t)}}, \overrightarrow{X_3^{(t)}}$ using Equation 19
Update $\overrightarrow{X_j^{(t)}}$ using Equation 20
Update $\overrightarrow{X_j^{(t)}}$ boundaries solution space using Equation 21
ENDFOR
Update each individual fitness value using F_{func}
Update $\overrightarrow{X_{\alpha}^{(t)}}$, $\overrightarrow{X_{\beta}^{(t)}}$, dan $\overrightarrow{X_{\delta}^{(t)}}$
ENDFOR
Return $\overrightarrow{X_{\alpha}^{(T_{max})}}$, $Fit\left(\overrightarrow{X_{\alpha}^{(T_{max})}}\right)$

Figure 3. GWO pseudocode for ML-ISP

The updated vector $\overrightarrow{X_i^{(t)}}$ can be outside the constraints of ML-ISP when the value $x_{(i,j)}$ is outside the range of gray level G. Therefore, Equation 21 is used to adjust $\overrightarrow{X_i^{(t)}}$ to be in the problem solution space in this study. Then, the fitness value of each *i-th* wolf is written as $Fit(\overrightarrow{X_i})$ which is calculated in the same way as when t =0. The position vectors of the wolves α, β, and δ at each iteration $\overrightarrow{X_{\alpha}^{(t)}}$, $\overrightarrow{X_{\beta}^{(t)}}$, and $\overrightarrow{X_{\delta}^{(t)}}$ are updated based on the three wolves with optimal fitness.

$$\overrightarrow{X_i^{(0)}} = \left[x_{(i,1)}, x_{(i,2)}, \dots, x_{(i,m)}\right], (i = 1,2,\dots,N) \wedge \left(0 < x_{(i,1)\dots} < x_{(i,m)} < L\right) \qquad (22)$$
$$x_{(i,j)} = randInt(0,(L-1))$$

GWO Pseudocode for solving ML-ISP

Pseudocode of the GWO Algorithm to solve ML-ISP is presented as in Figure 3.

Memory-based Grey Wolf Optimizer (mGWO)

$$\overrightarrow{X_j^{(t+1)}} = \begin{cases} L - round\left(random(0,1) * randomInteger(0,L)\right), \left(x_{(i,j)}\right) > L \\ 0, \left(x_{(i,j)}\right) < 0 \\ x_{(i,j)}, otherwise \end{cases}, \forall_{x_{(i,j)} \in \overrightarrow{X_j^{(t+1)}}} \qquad (21)$$

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

Algorithm 3: Segmenting process of C_i using m optimal threshold			
Input: *2D pixels array of image* C_i, m *optimal threshold T*			
Output: *segmented image* S_i			
Initialization:			
	row ← *row shape of* C_i		
	col ← *col shape of* C_i		
	regionThres ← *dictionary()*		
	flatCi ← *convert 2D of* C_i *into 1D array*		
// Get lower and upper bound for each threshold boundaries			
FOR *idx in range(0, m+1)* **DO**			
	IF *idx is 0* **DO**		
		bb_thres ← 0 ba_thres ← $T_{idx} - 1$	
	ELIF *idx is m* **DO**		
		bb_thres ← T_{idx-1} ba_thres ← $L - 2$	
	ELSE		
		bb_thres ← T_{idx-1} ba_thres ← $T_{idx} - 1$	
	$regionThres_{(idx)} \leftarrow [bb, ba]$		
	ENDIF		
ENDFOR			
// Convert each pixel values of C_i to correspondent bb and ba			
FOR *idx, pixel in enumerate(flatCi)* **DO**			
	FOR *regionId, interval in regionThres* **DO**		
		IF *pixel* $\geq$ $interval_{(0)}$ **AND** *pixel* $\leq$ $interval_{(1)}$ **DO**	
			$flatCi_{(idx)} \leftarrow interval_{(1)} + 1$
		ENDIF	
	ENDFOR		
ENDFOR			
S_i ← *reshape flatCi to row and col dimension*			
Return S_i			

Figure 5. Pseudocode to get the segmented image from C_i

The GWO proposed by Mirjaili et al (2014) [22] is considered vulnerable to being stuck at the optimum local value when used to solve multimodal problems because the update process from $\overrightarrow{X_i^{(t)}}$ only depends on three leader wolves [21]. The wolf pack will find it difficult to get out of the optimum locale when the three wolves are trapped in the optimum locale because they only depend on them [21]. Therefore, Gupta and Deep (2020) [21] proposed an update process of $\overrightarrow{X_i^{(t)}}$ integrating the best track record of each wolf with the positions of the three lead wolves. It aims with the intention of involving the best knowledge of each wolf in the process of exploring the solution space to guide the wolf pack to explore or move into a promising solution space and not get stuck at local optimum values [21].

All the symbols in this section are the same as those in the previous section. In mGWO, the encircling prey process is updated using Equation 23 [21]. Vector $\overrightarrow{X_{pbest\ (j)}^{(t)}}$ is the best position vector stored in the memory of the *j-th* wolf in the (t) iteration. The process of hunting prey involving the three α, β, and δ wolves is updated using Equation 24 [21]. Equation 24 [21] is proposed with the intention of imitating the thought that each wolf may have information on prey and use it to explore and retrace the surrounding wolf area using $\overrightarrow{X_{pbest\ (j)}^{(t)}}$.

Algorithm 2: Implementing mGWO for finding m optimal of thresholds using Otsu Method, Kapur Entropy, or M.Masi Entropy			
Input:			
	N_{wolf} ← Grey wolf population size T_{max} ← total number of maximum iteration CR ← crossover rate F_{func} ← type of objective function either Otsu Method, Kapur Entropy, or M.Masi Entropy		
Output: Optimal individual $\overrightarrow{X_\alpha^{(T_{max})}}$ and best fitness values $Fit\left(\overrightarrow{X_\alpha^{(T_{max})}}\right)$			
Initialization:			
	$GreyWolfs$ ← Initialize 2D matrix of N_{wolf} individual grey wolf position using Equation 22. $memoryGreyWolfs$ ← Save the initial best position of $GreyWolfs$ $pBestFitness$ ← Calculate each individual fitness value using F_{func} Find $\overrightarrow{X_\alpha^{(0)}}$, $\overrightarrow{X_\beta^{(0)}}$, dan $\overrightarrow{X_\delta^{(0)}}$		
// mGWO optimization steps for ML-ISP			
FOR t in range(0, T_{max}) DO			
	Update $\vec{a}$ and $k_{(t)}$ using Equation 17 and 26.		
	FOR individual in $GreyWolfs$ DO		
		Update $\overrightarrow{A_x}$ and $\overrightarrow{C_x}$ using Equation15 and 16.	
		IF $rand(0{,}1) < CR$ DO	
			Calculate $\overrightarrow{X_1^{(t)}}, \overrightarrow{X_2^{(t)}}, \overrightarrow{X_3^{(t)}}$ using Equation 19 Update $\overrightarrow{X_j^{(t)}}$ using Equation 24
		ELSE	
			Find two of $\overrightarrow{X_{rand\ (j)}^{(t)}}$ randomly Update $\overrightarrow{X_j^{(t)}}$ using Equation 25
		ENDIF	
		Update $\overrightarrow{X_j^{(t)}}$ boundaries solution space using Equation 21	
	ENDFOR		
	Update each individual fitness value using F_{func}		
	FOR $i = 1,2, \ldots N_{wolf}$ DO		
		IF $Fit\left(\overrightarrow{X_i^{(t)}}\right) > pBestFitness_{\ (i)}$ DO	
			$memoryGreyWolfs_{(i)}^{(t)} = \overrightarrow{X_i^{(t)}}$ $pBestFitness_{(i)} = Fit\left(\overrightarrow{X_i^{(t)}}\right)$
		ENDIF	
	ENDFOR		
	Update $\overrightarrow{X_\alpha^{(t)}}$, $\overrightarrow{X_\beta^{(t)}}$, dan $\overrightarrow{X_\delta^{(t)}}$		
ENDFOR			
Return $\overrightarrow{X_\alpha^{(T_{max})}}$, $Fit\left(\overrightarrow{X_\alpha^{(T_{max})}}\right)$			

In Equation 25, two wolf positions are chosen at random which are written as $\overrightarrow{X_{rand\ (J)}^{(t+1)}}$. The k value is the scale factor used to adjust the effect of subtracting the two position vectors. The value of k decreases linearly from 1 to 0 in each iteration which is updated using Equation 26. The crossover process is then carried out in the update stage $\overrightarrow{X_J^{(t+1)}}$. The process combines information from the three best wolves with each individual wolf according to Equation 27. The crossover probability value is written as *CR* and random numbers in the range 0 to 1 with a uniform distribution are written as r.

$$\overrightarrow{X_J^{(t+1)}} = \overrightarrow{X_p^{(t)}} - \vec{A} \cdot \vec{D} = \overrightarrow{X_p^{(t)}} - \vec{A} \cdot \left|\vec{C} \cdot \overrightarrow{X_p^{(t)}} - \overrightarrow{X_{pbest\ (J)}^{(t)}}\right| \quad (23)$$

$$\overrightarrow{Z_J^{(t+1)}} = \frac{\overrightarrow{S_1^{(t)}} + \overrightarrow{S_2^{(t)}} + \overrightarrow{S_3^{(t)}}}{3}$$
$$\overrightarrow{S_1^{(t)}} = \overrightarrow{X_\alpha^{(t)}} - \overrightarrow{A_1} \cdot \left|\overrightarrow{C_1} \cdot \overrightarrow{X_\alpha^{(t)}} - \overrightarrow{X_{pbest\ (J)}^{(t)}}\right|$$
$$\overrightarrow{S_2^{(t)}} = \overrightarrow{X_\beta^{(t)}} - \overrightarrow{A_2} \cdot \left|\overrightarrow{C_2} \cdot \overrightarrow{X_\beta^{(t)}} - \overrightarrow{X_{pbest\ (J)}^{(t)}}\right| \quad (24)$$
$$\overrightarrow{S_3^{(t)}} = \overrightarrow{X_\delta^{(t)}} - \overrightarrow{A_3} \cdot \left|\overrightarrow{C_3} \cdot \overrightarrow{X_\delta^{(t)}} - \overrightarrow{X_{pbest\ (J)}^{(t)}}\right|$$

$$\overrightarrow{P_J^{(t+1)}} = \overrightarrow{X_{pbest\ (J)}^{(t)}} + k_{(t+1)} \cdot \left(\overrightarrow{X_{rand\ (J)}^{(t+1)}} - \overrightarrow{X_{rand\ (J)}^{(t+1)}}\right) \quad (25)$$

$$k_{(t)} = \left(1 - \frac{t}{T}\right) \quad (26)$$

$$\overrightarrow{X_J^{(t+1)}} = \begin{cases} \overrightarrow{Z_J^{(t+1)}}, r < CR \\ \overrightarrow{P_J^{(t+1)}}, otherwise \end{cases} \quad (27)$$

mGWO pseudocode for solving ML-ISP

Pseudocode of the mGWO Algorithm to solve ML-ISP is presented as in Figure 4.

Image Segmentation with Optimal Threshold

Each channel in the C_i^x test image is then segmented using the optimal threshold value that has been obtained from the SI method-based optimization process. The pseudocode in Figure 5 is used to divide the C_i^x image into $m+1$ regions using optimal $\{t_1, t_2, \ldots, t_m\}$. The C_i^x image that has been segmented with the optimal m threshold is denoted as S_i^{RGB} for RGB color images and S_i^{gray} for grayscale images.

S_i^{RGB} is obtained by combining the segmentation results from C_i^r, C_i^g, and C_i^b. Assume thresholding levels for each channel *r*, *g*, and *b* in the RGB image, namely m_r, m_g, and m_b. The SI method is implemented to obtain the optimal threshold value for each channel. Each optimal threshold value is used to segment each channel using the pseudocode in Figure 5. The segmented image for each channel is denoted as S_i^r, S_i^g, and S_i^b such that:

$S_i^{RGB} = [S_i^r, S_i^g, S_i^b]$

Therefore, S_i^{RGB} has the most $m_r \times m_g \times m_b$ color levels and fewer than C_i^{RGB} [13].

EXPERIMENTS

Experiments in this study were conducted to measure the performance stability of the GWO and mGWO methods to solve ML-ISP using three different objective functions, namely the Otsu method, Kapur Entropy, and M.Masi Entropy. As a comparison, another standard IS optimization method is involved, namely the Genetic Algorithm (GA) [20], [25]–[28], Particle Swarm Optimization (PSO) [1], [9], [13], Whale Optimization Algorithm (WOA) [8], [12], [18], and Slime Mould Algorithm (SMA) [8], [23]. All comparison methods used are standard SI methods and not variants or optimized versions of these methods.

For the comparison to be fair, all methods used in this study use optimization stopping criteria, namely the maximum number of iterations is 100, with a population of 25 solutions, and the number of trials for each method is 30 [18]. The number of thresholds evaluated for each image in the test data is 2, 3, 4, and 5 as in previous studies [3], [4], [6]–[8], [14], [15], [18].

All methods are programmed and evaluated using the Python3.10 programming language which is implemented in the device environment Windows 10 – 64 bit, Intel Core i7-8565U CPU @1.80GHz and 8GB of RAM.

GridSearch Hyperparameter Tuning

The hyperparameter tuning process was carried out to obtain the optimal parameter combination for each method (as shown in Table 2) used in this study. It aims to obtain a fair comparison of performance metrics for each method at the evaluation stage. This study uses the GridSearch scheme for the hyperparameter tuning process. The GridSearch method looks for all possible combinations of each hyperparameter value and then gets the parameter combination that gives the most optimal results based on the predefined metrics.

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

Several criteria are set the same in this process to get the optimal combination of parameters from each method used. The metric used is the average fitness value of all training data. The number of thresholds used is 5. The objective function used is the Otsu method.

Evaluation Metrics

Qualitative and quantitative evaluation was carried out on each S_i segmented image. Qualitative measurement is done by visualizing $S_i^{RGB/gray}$ for each threshold level used and comparing it with the visualization of the original $C_i^{RGB/gray}$ image. Meanwhile, quantitative measurements are carried out using six metrics, namely peak signal to noise ratio (PSNR), root mean square error (RMSE), structured similarity index metrix (SSIM), universal quality index (UQI), fitness value and CPU processing time. (in seconds).

PSNR

PSNR measures the ratio between the maximum squared gray level and the mean square error (MSE) value. PSNR basically calculates the difference between S_i and C_i using the pixel intensity value of an image [18]. The PSNR is calculated using Equation 28 with the MSE value calculated using Equation 29. The gray level pixels at coordinates (x, y) of the segmented image are represented by $S_{(x,y)}$, while those of the original image are represented by $C_{(x,y)}$.

$$PSNR_{(C,S)} = 10log_{10}\left(\frac{255\times255}{MSE_{(C,S)}}\right) \quad (28)$$

$$MSE_{(C,S)} = \frac{\sum_{x=1}^{R}\sum_{y=1}^{K}|S_{(x,y)}-C_{(x,y)}|}{R\times K} \quad (29)$$

RMSE

RMSE measures the square root of MSE. The RMSE value is calculated using Equation 30.

$$RMSE(C,S) = \sqrt{\frac{\sum_{x=1}^{R}\sum_{y=1}^{K}\left(S_{(x,y)}-C_{(x,y)}\right)^2}{R\times K}} \quad (30)$$

SSIM

The structure of the images that are compared between S_i and C_i cannot be measured using only PSNR. PSNR only measures the comparison of errors between two images [8]. SSIM is used to measure the similarity, distortion, and brightness between the two images. SSIM is calculated using Equation 31. The μ_C and μ_s values are the average intensity values of C_i and S_i, respectively. The values of σ_C and σ_S are standard deviations of C_i and S_i, respectively. The value of $cov_{(C,S)}$ is the covariance between C_i and S_i. The values of a and b are constants of 6.5025 and 58.52252 respectively [18].

$$SSIM_{(C,S)} = \frac{(2\mu_S\mu_C+a)(2cov_{(C,S)}+b)}{(\mu_C{}^2+\mu_S{}^2+a)(\sigma_C{}^2+\sigma_S{}^2+b)} \quad (31)$$

Table 2. List of parameters along with the solution space for each method

Methods	Parameter	Search space
GA	Crossover rate (cr)	[0.5, 0.6, 0.7, 0.8]
	Mutation rate (mr)	[0.05, 0.1, 0.15, 0.2]
PSO	Cognitive learning parameter (φ_1)	[2.2, 2.4, 2.6, 2.8]
	Social learning parameter (φ_2)	[1.1, 1.3, 1.5, 1.7]
	Inertia (ω)	[0.5, 0.8]
WOA	Constanta $(cons)$	[1,2,3,4,5]
mGWO	Crossover rate (CR)	[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
SMA	Probability slime will search to random solution (Z)	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7]

UQI

UQI is similar to the SSIM measurement which measures the quality of S_i based on structural similarities between C_i and S_i. UQI is measured by Equation 32.

$$UQI_{(C,S)} = \frac{4cov_{(C,S)}\mu_C\mu_S}{(\mu_C{}^2+\mu_S{}^2)(\sigma_C{}^2+\sigma_S{}^2)} \quad (32)$$

Fitness values

The fitness value is used to measure the performance of the method used against the objective function used. The Otsu method, Kapur Entropy, and M.Masi Entropy are each used to measure the fitness value of each method, each of which is calculated using Equations 3, 8 and 10.

CPU Processing Time

CPU processing time is measured to measure the efficiency of the optimization process time of each method for the results it obtains. To get fair results, calculating CPU processing time starts from the moment the method starts optimizing the optimal threshold value until it gets it, without measuring other

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

computational processes involved (such as variable declarations, value initialization, and so on).

RESULT AND DISCUSSION

This section presents the results of the GWO and mGWO for solving ML-ISP using the Otsu, Kapur Entropy, and M.Masi Entropy methods as objective functions. These results were analyzed from the qualitative and quantitative aspects. The average value of each quantitative metric used is calculated from a total of 30 experiments conducted for each method. The results shown in this section are obtained from testing on RGB images. However, this study also tested grayscale images to obtain comparable results with state-of-the-art methods for solving IS-based ML-ISP from previous studies.

Parameter Setting

This section presents the results of the GWO and mGWO for solving ML-ISP using the Otsu, Kapur Entropy, and M.Masi Entropy methods as objective functions. These results were analyzed from the qualitative and quantitative aspects. The average value of each quantitative metric used is calculated from a total of 30 experiments conducted for each method. The results shown in this section are obtained from testing on RGB images. However, this study also tested grayscale images to obtain comparable results with state-of-the-art methods for solving IS-based ML-ISP from previous studies.

Quantitative Analysis Results

This section describes the performance results of the GWO and mGWO methods to solve ML-ISP with the Otsu, Kapur Entropy, and M.Masi Entropy method objective functions. Table 4 – Table 11 each displays the average value of PSNR, RMSE, SSIM, UQI, CPU Time, and Fitness (for the three RGB channels) of each method. The higher the value of PSNR, SSIM, UQI and Fitness; and the lower the RMSE and CPU Time values, the better the performance of an SI method for solving ML-ISP. Values in bold on the measurement results show the best results.

All methods were tested on RGB and grayscale images. The measurement results displayed in this section are only for RGB image format, while the measurement results displayed for grayscale images are only the average value of fitness and CPU Time as a comparison with other state-of-the-art methods.

The sum of the best performance for each method from 24 total experiments for each metric is summarized in Figure 6. Based on Figure 6 the majority of mGWO outperformed the performance of other methods in almost all metrics. In fact, even the standard GWO method was able to obtain the best performance after mGWO. This shows that the GWO method and its variant, mGWO, are stable when tested with different objective functions.

The interesting thing is that the performance of the GWO method can be matched or even surpassed by the PSO method. For example, based on Figure 6, PSO can offset GWO in terms of SSIM when using the Otsu method. When using Kapur Entropy, PSO was able to outperform GWO in terms of PSNR, RMSE, SSIM and UQI. When using M.Masi Entropy, PSO outperforms GWO in terms of PSNR, RMSE, and SSIM.

The results of testing the average fitness value of mGWO and GWO on grayscale images as shown in Table 12 also show the same thing as testing RGB images. The mGWO and GWO methods got 14 and 6 best results respectively from a total of 24 experiments. PSO cannot match the performance of the two methods because it only gets the 4 best results. In fact, mGWO was able to outperform the other methods in terms of CPU processing time for most experiments when using the Otsu method as the objective function, as shown in Table 13.

Experimental results on RGB and grayscale images show that the mGWO and GWO methods are able to solve intensity-based ML-ISP well. The advantage of solving ML-ISP on RGB or grayscale images using the GWO method is that it is simple and easy to implement [14] compared to a thorough search using only the Otsu, Kapur Entropy, or M.Masi Entropy methods. However, the mGWO method provides more accurate performance than the standard GWO [21]. This is because mGWO is able to increase the global exploration phase, local exploitation, and balance the two during the search for prey [21], so as to avoid local optimum values [7]. In addition, the existence of a new prey hunting mechanism in mGWO can have an impact on wolf packs to explore new areas that are more promising for solutions [21].

The GWO method can produce higher quality solutions when compared to other SI benchmark methods [7]. GWO can balance the exploration and exploitation phases so that it can find better solutions [14]. Parameter configuration of other SI benchmark methods which are relatively more than GWO can cause these methods to get stuck at local optimum when solving problems with high dimensional solution spaces, such as PSO. [14]. The success of finding solutions from GWO is heavily influenced by the α, β, and δ wolves [14].

In standard GWO, the prey hunting phase is only guided by the best three wolves, namely α, β, and δ wolves. These three wolves might get stuck at local optimum values when the optimization problems being solved are multimodal [21]. It will be difficult for a pack of wolves to get out of the local optimum when the process of hunting for prey depends only on the three best wolves. In mGWO, this problem is solved by utilizing the best track record of each individual gray wolf during the prey hunting phase. This allows for a collaborative information exchange mechanism between each individual and the wolf pack so that the search for optimal solutions can take place efficiently [21]. The best track record of knowledge from each individual wolf is used as a guide besides using the three best wolves to get a more promising solution space and to maintain balance between exploitation and exploration [21]. This is in accordance with the results in this study.

The process of updating solutions by being guided by the best solutions and utilizing the best track record of each individual has proven to perform better in solving ML-ISP on RGB and grayscale images. This is shown through the results of this study and is summarized in Figure 6. The GWO method utilizes solutions from α, β, and δ wolves in each iteration in the process of updating the wolf's position in hunting the prey [22]. The PSO method utilizes the best track record of each particle and utilizes the best global position of a set of particles in updating the position and velocity vectors of each particle. The mGWO method updates the wolf's position by combining the GWO and PSO mechanisms. The α, β, and δ wolves and the track record of each wolf are used to guide the solution update process on mGWO. The three methods, mGWO, GWO, and PSO, are the three methods that performed best in this study compared to other methods.

Qualitative Analysis Results

This section presents a qualitative analysis. Figure 1 – Figure 4 in supplementary fsiles displays the segmented RGB images of each method for the number of thresholds, namely 2, 3, 4, and 5. We also record the optimal threshold values obtained from each method for the three channels on the the images as a supplementary file. To make comprehensive comparisons, the proposed method is also analyzed qualitatively on the grayscale test images and their graylevel histograms. The results displayed are only the results of grayscale image segmentation at level 3, as shown in Figure 5 and its best threshold in Figure 6.

The visualization results of the segmented images shown show that the optimal threshold values generated by the SI method based on the Otsu, Kapur Entropy, and M.Masi Entropy methods are able to properly separate several classes in the RGB and grayscale test images. The results of 3-level grayscale image segmentation on C6 can show important components that should be in C6 images, such as the sky, trees, sailboats, lakes, parks, and shaped clouds. These components can be segmented properly with optimal threshold values obtained from the SI-based optimization method. The results of the segmentation also do not overlap between components. The results of the 3-level RGB image segmentation as shown in Figure 2 (in supplementary files) show that the objective function with Kapur Entropy can produce segmented images that are relatively brighter and not blurry when compared to the Otsu method. This can be seen in the RGB C2 – C6 image results which have been segmented.

Wilcoxon Signed Rank Test Results

The Wilcoxon signed-rank test was performed as a statistical analysis at the 5% significance level. The fitness and PSNR values generated by the objective function of the Otsu method, Kapur Entropy, and M.Masi Entropy of each method are compared to one another. Each method was run 30 times for this analysis. The null hypothesis (H_0) and the alternative hypothesis (H_a) used are as follows [6], [14].

H_0: The difference between sample pairs is not significant

H_a: The difference between pairs of samples is significant

If the p-values are less than 0.05 (H_a), then the null hypothesis can be rejected at the 5% significance level. Conversely, if the p-values are more than 0.05 (H_0), then the null hypothesis is accepted [14].

We calculate *p*-values using the Wilcoxon signed-rank test on the fitness and PSNR value metrics between the mGWO method and the comparison method to solve the multi-level color image segmentation problem. The results are presented as a supplementary file. The p-values of mGWO which show better results than other methods are marked with a sign (*). Based on those results, when combined with the Otsu method, mGWO obtained significantly better results than SMA, WOA, GA, PSO, and GWO of 24, 23, 22, 13, and 0 respectively out of a total of 24 experiments. When combined with Kapur Entropy, mGWO obtained significantly better results than SMA, WOA, GA, PSO, and GWO by 23, 16, 18, 13, and 3 respectively out of a total of 24 experiments. When combined with M.Masi Entropy, mGWO obtained significantly better results than SMA, WOA, GA, PSO, and GWO of 23, 22, 24, 12, and 2 respectively out of a total of 24 experiments. Although the performance of mGWO is better than GWO, in most statistics it does not show a significant difference.

Comparison with other state-of-the-art algorithms

Table 14 and Table 15 present a comparative analysis of mGWO (proposed) against state-of-the-art SI methods from previous studies used to solve ML-ISP. The method being compared are KHO [15], WOA, MFO [18], and GWO [14]. Comparisons were made to the test image and the same number of thresholds by measuring the average value of fitness and CPU Time (in seconds). The test image used is a grayscale image because the studies being compared have not reported their results with RGB imagery and for a fair comparison.

The first comparison was made by comparing the performance of the SI method in terms of objective function values when using the Otsu method. The proposed mGWO method was able to give the best results for 12 out of a total of 24 experiments. This result outperforms the results given by the GWO [14] and KHO [15] by 3 and 9 out of a total 24 experiments, respectively. Furthermore, the mGWO method was able to outperform all test cases at various levels when compared to WOA and MFO [18]. The mGWO method is also proven to provide better performance when compared to GWO [14] in the majority of test cases. Some methods like KHO [15], WOA, MFO [18], and GWO [14] does not report the research results at several threshold levels from the same test image. The performance of these methods has not been tested for solving ML-ISP on RGB images.

The second comparison was made by comparing the performance of the SI method in terms of objective function values when using Kapur Entropy. In contrast to the results of the Otsu method, the KHO method [15] did not perform better when compared to the mGWO method in this study. The mGWO and GWO methods [14] respectively gave the best results in 17 and 4 out of 24 experiments. This also shows that the mGWO method is also proven to provide better performance when compared to GWO [14].

The third comparison was made by comparing the performance of the SI method in terms of CPU Time when using the Otsu and Kapur Entropy methods as the objective function. The GWO method [14] can provide the shortest computation time when compared to mGWO, KHO [15], WOA, and MFO [18] in most test cases. The longest CPU processing time was generated by WOA and MFO [18] in most test cases according to Table 14. CPU Time from GWO [14] is relatively faster when compared to mGWO because in mGWO there are several additional processes that were not previously available in standard GWO [14], [22]. Some of them, namely the process of initializing the matrix to store the best track record, the process of updating the track record of each individual wolf, the process of updating the best fitness value of each grey wolf individu in each iteration and the crossover process in hunting prey [21]. This causes the computational time of the mGWO to increase when compared to the standard GWO [14], [22].

CONCLUSION

Determining the optimal threshold value for solving color ML-ISP can be viewed as an optimization problem using the Otsu, Kapur Entropy, and M.Masi Entropy methods as objective functions. Therefore, the SI method, namely mGWO as a variant of GWO, is proposed to solve this problem. The objective of this method is to determine the optimal threshold value for each channel by maximizing the specified objective function. This study compared the experimental results of mGWO against the PSO, GA, WOA, and SMA methods using six metrics, namely PSNR, RMSE, SSIM, UQI, fitness value, and CPU time (seconds).

The experimental results show that the performance of the majority of mGWO and GWO is superior to other methods in almost all metrics, but the mGWO method is still better than GWO. In addition, mGWO performance is stable when tested with different objective functions. In fact, the results of a comparison of the mGWO method against state-of-the-art WOA and MFO to solve ML-ISP on grayscale images show the best performance of a total of 24 experiments. The increase in the global exploration and local exploitation phases of mGWO can help find a better optimal threshold value for solving color ML-ISP. Statistical testing using the Wilcoxon signed-rank test showed that mGWO gave a significant difference in results compared to other methods in most experiments.

The next research will examine the performance of mGWO to solve color ML-ISP with the multi-objective optimization problem paradigm. In addition, a dynamic approach in determining the optimal number of threshold levels for color ML-ISP will be designed and implemented to obtain better segmentation results.

REFERENCES

[1] X. Chen, Miao Pu, and Q. Bu, “Image Segmentation Algorithm Based on Particle Swarm Optimization with K-Means Optimization,” in *IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, 2019, pp. 156–159.

[2] S. Tongbram, B. A. Shimray, L. S. Singh, and N. Dhanachandra, “A novel image segmentation approach using fcm and whale optimization algorithm,” *J Ambient Intell Humaniz Comput*, 2021, doi: 10.1007/s12652-020-02762-w.

[3] A. K. M. Khairuzzaman and S. Chaudhury, “Masi entropy based multilevel thresholding for image segmentation,” *Multimed Tools Appl*, vol. 78, no. 23, pp. 33573–33591, Dec. 2019, doi: 10.1007/s11042-019-08117-8.

[4] G. Ma and X. Yue, “An improved whale optimization algorithm based on multilevel threshold image segmentation using the Otsu method,” *Eng Appl Artif Intell*, vol. 113, Aug. 2022, doi: 10.1016/j.engappai.2022.104960.

[5] J. Rahaman and M. Sing, “An efficient multilevel thresholding based satellite image segmentation approach using a new adaptive cuckoo search algorithm,” *Expert Syst Appl*, vol. 174, Jul. 2021, doi: 10.1016/j.eswa.2021.114633.

[6] P. Upadhyay and J. K. Chhabra, “Multilevel thresholding based image segmentation using new multistage hybrid optimization algorithm,” *J Ambient Intell Humaniz Comput*, vol. 12, no. 1, pp. 1081–1098, Jan. 2021, doi: 10.1007/s12652-020-02143-3.

[7] B. S. Khehra, A. Singh, and L. Kaur, “M. Masi Entropy- and Grey Wolf Optimizer-Based Multilevel Thresholding Approach for Image Segmentation,” *Journal of The Institution of Engineers (India): Series B*, vol. 103, no. 5, pp. 1619–1642, Oct. 2022, doi: 10.1007/s40031-022-00740-8.

[8] M. Abdel-Basset, V. Chang, and R. Mohamed, “HSMA_WOA: A hybrid novel Slime mould algorithm with whale optimization algorithm for tackling the image segmentation problem of chest X-ray images,” *Applied Soft Computing Journal*, vol. 95, Oct. 2020, doi: 10.1016/j.asoc.2020.106642.

[9] M. Sharif, J. Amin, M. Raza, M. Yasmin, and S. C. Satapathy, “An integrated design of particle swarm optimization (PSO) with fusion of features for detection of brain tumor,” *Pattern Recognit Lett*, vol. 129, pp. 150–157, Jan. 2020, doi: 10.1016/j.patrec.2019.11.017.

[10] F. Orujov, R. Maskeliūnas, R. Damaševičius, and W. Wei, “Fuzzy based image edge detection algorithm for blood vessel detection in retinal images,” *Applied Soft Computing Journal*, vol. 94, Sep. 2020, doi: 10.1016/j.asoc.2020.106452.

[11] C. J. J. Sheela and G. Suganthi, “Morphological edge detection and brain tumor segmentation in Magnetic Resonance (MR) images based on region growing and performance evaluation of modified Fuzzy C-Means (FCM) algorithm,” *Multimed Tools Appl*, vol. 79, no. 25–26, pp. 17483–17496, Jul. 2020, doi: 10.1007/s11042-020-08636-9.

[12] C. Lang and H. Jia, “Kapur’s entropy for color image segmentation based on a hybrid whale optimization algorithm,” *Entropy*, vol. 21, no. 3, Mar. 2019, doi: 10.3390/e21030318.

[13] S. Borjigin and P. K. Sahoo, “Color image segmentation based on multi-level Tsallis–Havrda–Charvát entropy and 2D histogram using PSO algorithms,” *Pattern Recognit*, vol. 92, pp. 107–118, Aug. 2019, doi: 10.1016/j.patcog.2019.03.011.

[14] A. K. M. Khairuzzaman and S. Chaudhury, “Multilevel thresholding using grey wolf optimizer for image segmentation,” *Expert Syst Appl*, vol. 86, pp. 64–76, Nov. 2017, doi: 10.1016/j.eswa.2017.04.029.

[15] K. P. Baby Resma and M. S. Nair, “Multilevel thresholding for image segmentation using Krill Herd Optimization algorithm,” *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 5, pp. 528–541, Jun. 2021, doi: 10.1016/j.jksuci.2018.04.007.

[16] S. Arora, J. Acharya, A. Verma, and P. K. Panigrahi, “Multilevel thresholding for image segmentation through a fast statistical recursive algorithm,” *Pattern Recognit Lett*, vol. 29, no. 2, pp. 119–

125, Jan. 2008, doi: 10.1016/j.patrec.2007.09.005.

[17] M. Ameur, M. Habba, and Y. Jabrane, "A comparative study of nature inspired optimization algorithms on multilevel thresholding image segmentation," *Multimed Tools Appl*, vol. 78, no. 24, pp. 34353–34372, Dec. 2019, doi: 10.1007/s11042-019-08133-8.

[18] M. A. El Aziz, A. A. Ewees, and A. E. Hassanien, "Whale Optimization Algorithm and Moth-Flame Optimization for multilevel thresholding image segmentation," *Expert Syst Appl*, vol. 83, pp. 242–256, Oct. 2017, doi: 10.1016/j.eswa.2017.04.023.

[19] B. Lei and J. Fan, "Image thresholding segmentation method based on minimum square rough entropy," *Applied Soft Computing Journal*, vol. 84, Nov. 2019, doi: 10.1016/j.asoc.2019.105687.

[20] D. T. Hidayat, Isnan, and M. A. Fauzi, "Optimum Multilevel Thresholding Hybrid GA-PSO By Algorithm," *Journal of Computer Science and Information*, vol. 6, no. 1, pp. 1–5, 2013.

[21] S. Gupta and K. Deep, "A memory-based Grey Wolf Optimizer for global optimization tasks," *Applied Soft Computing Journal*, vol. 93, Aug. 2020, doi: 10.1016/j.asoc.2020.106367.

[22] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[23] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Generation Computer Systems*, vol. 111, pp. 300–323, Oct. 2020, doi: 10.1016/j.future.2020.03.055.

[24] Suyanto, A. Arifianto, R. Rita, and S. Andi, *Evolutionary Machine Learning Pembelajaran Mesin Otonom Berbasis Komputasi Evolusioner*. Bandung: Informatika Bandung, 2020.

[25] I. W. Supriana, M. A. Raharja, I. M. Bimantara, and D. Bramantya, "IMPLEMENTASI DUA MODEL CROSSOVER PADA ALGORITMA GENETIKA UNTUK OPTIMASI PENGGUNAAN RUANG PERKULIAHAN," *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, vol. 4, no. 2, pp. 167–177, Oct. 2021, [Online]. Available: http://jurnal.stiki-indonesia.ac.id/index.php/jurnalresistor

[26] W. Chen, K. Ramos, K. N. Mullaguri, and A. S. Wu, "Genetic Algorithms For Extractive Summarization," May 2021, [Online]. Available: http://arxiv.org/abs/2105.02365

[27] D. Kristiadi and R. Hartanto, "Genetic Algorithm for lecturing schedule optimization," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 13, no. 1, pp. 83–94, Jan. 2019, doi: 10.22146/ijccs.43038.

[28] Y. S. Chaudhari, V. W. Dmello, S. S. Shah, and P. Bhangale, "Autonomous Timetable System Using Genetic Algorithm," in *Proceedings of the Fourth International Conference on Smart Systems and Inventive Technology (ICSSIT-2022)*, Institute of Electrical and Electronics Engineers (IEEE), Feb. 2022, pp. 1687–1694. doi: 10.1109/icssit53264.2022.9716370.

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

Table 4. PSNR average measurement of all methods on RGB test images

Images	m	Mean PSNR values from Otsu Method as Objective Function					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	14.524	14.485	14.518	14.396	**14.526**	14.525
	3	**16.204**	15.607	16.028	15.320	16.158	16.164
	4	17.346	16.296	17.323	16.286	17.458	**17.482**
	5	18.683	17.322	18.276	17.137	18.709	**18.758**
C2	2	15.436	14.803	15.427	13.945	**15.438**	15.438
	3	17.431	16.739	17.392	15.727	17.443	**17.445**
	4	**19.430**	18.155	18.944	16.638	19.418	19.411
	5	**20.833**	19.266	20.174	17.926	20.650	20.687
C3	2	**11.561**	11.166	11.494	10.811	11.560	11.560
	3	14.218	13.702	13.914	12.410	**14.225**	14.223
	4	17.768	15.706	16.473	14.164	17.880	**17.937**
	5	19.526	17.309	18.203	15.671	**19.826**	19.802
C4	2	14.767	14.334	14.615	13.595	14.798	**14.799**
	3	16.976	16.197	16.738	15.076	17.012	**17.012**
	4	18.268	17.272	17.836	16.301	18.473	**18.500**
	5	19.345	18.472	18.706	17.311	19.612	**19.728**
C5	2	13.856	13.546	13.751	12.797	**13.857**	13.857
	3	15.217	15.097	15.095	14.502	15.290	**15.313**
	4	17.279	16.658	16.753	15.791	17.607	**17.607**
	5	18.483	17.889	17.916	16.842	18.876	**18.897**
C6	2	14.805	14.220	14.848	13.516	14.799	**14.856**
	3	16.509	15.998	16.387	15.246	16.501	**16.527**
	4	18.760	17.149	18.413	16.259	18.899	**18.922**
	5	19.876	18.314	19.320	17.289	**19.959**	19.955
Images	m	**Mean PSNR values from Kapur Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	**14.112**	13.988	14.036	11.584	14.064	14.111
	3	**14.974**	14.747	14.950	14.757	14.951	14.953
	4	15.405	15.228	15.340	**15.974**	15.372	15.372
	5	15.789	15.911	15.591	**16.767**	15.646	15.655
C2	2	15.460	14.797	15.455	7.402	**15.468**	15.468
	3	**17.976**	16.834	17.948	11.945	17.958	17.961
	4	**19.789**	18.427	19.562	15.799	19.608	19.609
	5	21.766	19.598	21.587	17.475	21.843	**21.943**
C3	2	13.201	12.516	**13.224**	3.321	13.201	13.200
	3	16.243	14.846	**16.267**	10.490	16.261	16.254
	4	18.235	16.393	18.140	13.891	18.291	**18.295**
	5	**19.214**	17.419	19.164	14.891	19.180	19.110
C4	2	13.792	13.525	13.777	5.408	13.792	**13.793**
	3	16.185	15.070	16.145	12.219	**16.263**	16.262
	4	17.339	16.872	17.514	15.274	17.660	**17.673**
	5	18.629	17.877	18.615	16.331	18.716	**18.742**
C5	2	**13.149**	12.866	13.101	4.248	13.106	13.104
	3	15.096	14.861	15.102	12.460	15.136	**15.139**
	4	16.841	16.248	16.907	15.242	**16.985**	16.980
	5	17.873	17.331	17.923	15.993	**18.025**	17.957
C6	2	13.066	13.633	13.116	5.157	13.768	**14.145**
	3	15.931	14.860	**15.934**	13.715	15.928	15.926
	4	17.076	16.443	17.046	15.087	**17.077**	17.075
	5	17.932	17.839	18.109	16.187	18.175	**18.294**
Images	m	**Mean PSNR values from M.Masi Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	14.524	14.401	14.522	14.470	**14.525**	14.525
	3	**16.196**	15.643	16.077	15.348	16.163	16.162

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

Table 4. (continued)

Images	m	Mean PSNR values from M.Masi Entropy as Objective Function					
		PSO	GA	WOA	SMA	GWO	mGWO
C1	4	17.346	16.595	17.230	16.335	17.472	**17.482**
	5	18.784	17.775	18.271	16.991	18.754	**18.786**
C2	2	15.436	14.727	15.427	13.718	15.438	**15.438**
	3	17.441	16.892	17.375	15.665	17.443	**17.444**
	4	**19.448**	17.944	19.008	16.556	19.411	19.399
	5	**20.860**	19.262	20.221	17.959	20.697	20.713
C3	2	11.559	11.102	11.448	10.847	11.560	**11.560**
	3	14.212	13.558	13.864	12.777	14.225	**14.225**
	4	17.646	15.987	16.544	14.054	**17.931**	17.929
	5	19.341	17.555	18.264	15.100	**19.822**	19.821
C4	2	**14.802**	14.437	14.748	13.784	14.757	14.798
	3	16.907	15.975	16.805	15.163	16.982	**17.010**
	4	18.343	17.453	17.973	16.584	**18.480**	18.465
	5	19.176	18.646	18.705	17.229	19.681	**19.744**
C5	2	**13.857**	13.555	13.790	12.879	13.857	13.857
	3	15.198	14.887	15.177	14.541	15.240	**15.311**
	4	17.293	16.577	16.899	15.765	17.576	**17.585**
	5	18.438	17.628	18.000	16.728	18.862	**18.916**
C6	2	**14.856**	14.331	14.851	13.684	14.856	14.855
	3	16.527	15.979	16.349	15.220	16.510	**16.527**
	4	18.757	17.082	18.495	16.248	**18.925**	18.919
	5	19.784	18.433	19.269	16.939	19.930	**19.955**

Table 5. RMSE average measurement of all methods on RGB test images

Images	m	Mean RMSE values from Otsu Method as Objective Function					
		PSO	GA	WOA	SMA	GWO	mGWO
C1	2	47.898	48.131	47.934	48.647	**47.892**	47.894
	3	**39.481**	42.442	40.295	43.844	39.685	39.660
	4	34.643	39.145	34.716	39.417	34.171	**34.076**
	5	29.709	35.068	31.148	35.692	29.594	**29.423**
C2	2	43.127	46.486	43.171	51.675	**43.116**	43.117
	3	34.276	37.229	34.431	41.859	34.227	**34.220**
	4	**27.233**	31.629	28.836	37.741	27.269	27.289
	5	**23.173**	27.980	25.053	32.621	23.663	23.561
C3	2	**67.376**	70.816	67.931	73.835	67.382	67.382
	3	49.619	52.917	51.456	61.647	**49.581**	49.587
	4	32.979	42.081	38.754	50.536	32.568	**32.336**
	5	26.946	35.072	31.605	42.319	**26.016**	26.090
C4	2	46.582	48.998	47.470	53.416	46.416	**46.409**
	3	36.122	39.577	37.173	45.127	35.971	**35.971**
	4	31.140	34.992	32.782	39.242	30.405	**30.307**
	5	27.525	30.542	29.666	34.908	26.675	**26.312**
C5	2	51.727	53.651	52.482	58.743	**51.726**	51.727
	3	44.239	44.940	44.883	48.315	43.863	**43.743**
	4	34.907	37.529	37.181	41.532	33.590	**33.588**
	5	30.387	32.636	32.549	36.839	29.026	**28.954**
C6	2	46.402	49.666	46.147	54.063	46.438	**46.106**
	3	38.115	40.486	38.675	44.247	38.154	**38.038**
	4	29.438	35.513	30.671	39.349	28.949	**28.871**
	5	25.872	31.089	27.652	35.000	**25.623**	25.634
Images	m	**Mean RMSE values from Kapur Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	**50.227**	50.997	50.683	67.367	50.519	50.234
	3	**45.483**	46.740	45.610	46.762	45.600	45.593

Table 5. (continued)

Images	m	Mean RMSE values from Kapur Entropy as Objective Function					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	4	43.282	44.245	43.608	**40.960**	43.443	43.447
	5	41.414	40.966	42.363	**37.319**	42.098	42.053
C2	2	43.008	46.513	43.030	110.832	**42.967**	42.969
	3	**32.193**	36.818	32.296	68.879	32.259	32.246
	4	**26.134**	30.668	26.820	43.742	26.677	26.676
	5	20.812	26.823	21.267	34.638	20.633	**20.388**
C3	2	55.784	60.465	**55.633**	174.108	55.782	55.790
	3	39.299	46.255	**39.193**	79.512	39.220	39.250
	4	31.257	38.778	31.622	51.813	31.044	**31.030**
	5	**27.926**	34.489	28.094	46.261	28.030	28.258
C4	2	52.112	53.885	52.204	136.980	52.110	**52.109**
	3	39.568	45.152	39.748	65.653	**39.212**	39.216
	4	34.659	36.728	33.957	44.302	33.386	**33.336**
	5	29.869	32.687	29.909	39.255	29.561	**29.475**
C5	2	**56.125**	58.138	56.427	156.645	56.398	56.406
	3	44.852	46.244	44.817	64.608	44.640	**44.625**
	4	36.692	39.428	36.412	44.730	**36.081**	36.105
	5	32.584	34.886	32.395	40.974	**32.015**	32.267
C6	2	56.658	53.240	56.362	141.201	52.479	**50.234**
	3	40.735	46.231	**40.723**	54.054	40.749	40.762
	4	35.708	38.518	35.828	45.315	**35.701**	35.712
	5	32.370	32.842	31.736	39.778	31.495	**31.069**

Images	m	Mean RMSE values from M.Masi Entropy as Objective Function					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	47.901	48.594	47.913	48.249	**47.894**	47.895
	3	**39.516**	42.195	40.061	43.668	39.664	39.670
	4	34.637	37.966	35.097	39.149	34.115	**34.077**
	5	29.360	33.236	31.142	36.426	29.436	**29.328**
C2	2	43.128	46.901	43.168	53.015	43.117	**43.116**
	3	34.236	36.549	34.500	42.219	34.230	**34.226**
	4	**27.175**	32.442	28.619	38.083	27.289	27.328
	5	**23.101**	27.866	24.908	32.435	23.536	23.492
C3	2	67.391	71.354	68.318	73.794	67.382	**67.381**
	3	49.654	53.937	51.790	59.396	49.580	**49.579**
	4	33.459	40.801	38.287	51.129	**32.358**	32.366
	5	27.550	34.011	31.431	45.312	**26.028**	26.030
C4	2	**46.393**	48.417	46.702	52.260	46.649	46.415
	3	36.416	40.616	36.869	44.679	36.102	**35.977**
	4	30.868	34.261	32.250	37.928	**30.378**	30.432
	5	28.070	29.923	29.672	35.212	26.457	**26.263**
C5	2	**51.726**	53.596	52.140	58.107	51.726	51.727
	3	44.338	46.030	44.443	48.051	44.124	**43.751**
	4	34.849	37.930	36.532	41.682	33.711	**33.679**
	5	30.559	33.601	32.176	37.499	29.072	**28.890**
C6	2	**46.106**	49.036	46.131	52.980	46.106	46.110
	3	38.038	40.565	38.848	44.385	38.114	**38.033**
	4	29.439	35.806	30.401	39.401	**28.859**	28.880
	5	26.160	30.671	27.857	36.414	25.709	**25.633**

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

Table 6. SSIM average measurement of all methods on RGB test images

Images	m	Mean SSIM values from Otsu Method as Objective Function					
		PSO	GA	WOA	SMA	GWO	mGWO
C1	2	0.781	0.769	**0.781**	0.752	0.781	0.781
	3	0.733	**0.762**	0.736	0.759	0.735	0.734
	4	0.722	0.755	0.720	**0.757**	0.717	0.717
	5	0.731	**0.755**	0.727	0.751	0.727	0.727
C2	2	**0.639**	0.620	0.639	0.596	0.639	0.639
	3	0.699	0.674	0.699	0.639	**0.700**	0.700
	4	0.752	0.711	0.749	0.668	**0.754**	0.754
	5	**0.784**	0.739	0.776	0.693	0.784	0.784
C3	2	**0.439**	0.431	0.438	0.415	0.439	0.439
	3	**0.535**	0.511	0.528	0.465	0.535	0.535
	4	0.613	0.571	0.589	0.512	0.615	**0.616**
	5	0.676	0.618	0.640	0.562	**0.679**	0.679
C4	2	0.695	0.670	0.689	0.628	0.695	**0.695**
	3	0.775	0.738	0.768	0.689	0.776	**0.776**
	4	0.819	0.778	0.806	0.735	0.825	**0.825**
	5	0.852	0.809	0.835	0.770	0.857	**0.859**
C5	2	0.561	0.546	0.558	0.518	**0.561**	0.561
	3	0.605	0.590	0.600	0.560	0.605	**0.606**
	4	0.663	0.633	0.646	0.594	**0.665**	0.665
	5	0.703	0.665	0.682	0.624	0.708	**0.709**
C6	2	0.663	0.636	0.665	0.606	0.663	**0.665**
	3	**0.729**	0.699	0.726	0.652	0.728	0.728
	4	**0.761**	0.726	0.755	0.685	0.760	0.761
	5	0.804	0.744	0.790	0.712	**0.806**	0.806
Images	m	**Mean SSIM values from Kapur Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	0.768	0.749	0.765	0.650	0.766	**0.768**
	3	0.789	0.766	0.791	0.731	**0.791**	0.791
	4	0.795	0.771	0.798	0.740	**0.799**	0.799
	5	0.792	0.767	**0.799**	0.749	0.799	0.799
C2	2	**0.643**	0.616	0.643	0.467	0.643	0.643
	3	0.699	0.662	0.700	0.574	**0.700**	0.700
	4	0.732	0.705	**0.733**	0.640	0.732	0.731
	5	0.779	0.733	0.778	0.679	0.783	**0.784**
C3	2	**0.458**	0.437	0.458	0.194	0.458	0.458
	3	0.544	0.516	0.544	0.422	0.544	**0.544**
	4	0.623	0.566	0.618	0.497	0.625	**0.626**
	5	**0.652**	0.602	0.645	0.528	0.646	0.644
C4	2	**0.662**	0.642	0.662	0.159	0.662	0.662
	3	0.747	0.700	0.746	0.572	**0.749**	0.749
	4	0.788	0.754	0.791	0.682	0.795	**0.796**
	5	0.829	0.785	0.828	0.715	0.830	**0.831**
C5	2	**0.537**	0.526	0.535	0.275	0.535	0.535
	3	**0.604**	0.581	0.603	0.523	0.603	0.603
	4	0.647	0.614	0.647	0.575	**0.647**	0.647
	5	0.675	0.648	0.672	0.604	**0.676**	0.674
C6	2	0.609	0.611	0.611	0.298	0.628	**0.638**
	3	0.698	0.656	**0.698**	0.601	0.697	0.697
	4	**0.745**	0.705	0.744	0.643	0.743	0.743
	5	**0.781**	0.737	0.777	0.670	0.778	0.778
Images	m	**Mean SSIM values from M.Masi Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	0.781	0.768	**0.781**	0.752	0.781	0.781
	3	0.734	0.755	0.737	**0.755**	0.734	0.734

Table 6. (continued)

Images	m	Mean SSIM values from M.Masi Entropy as Objective Function					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	4	0.721	0.758	0.722	**0.759**	0.717	0.717
	5	0.730	**0.760**	0.725	0.757	0.727	0.728
C2	2	0.639	0.621	0.639	0.587	**0.639**	0.639
	3	**0.700**	0.674	0.698	0.639	0.699	0.699
	4	0.753	0.712	0.749	0.667	0.754	**0.754**
	5	**0.785**	0.737	0.777	0.698	0.785	0.784
C3	2	0.439	0.430	0.437	0.412	**0.439**	0.439
	3	0.535	0.511	0.525	0.475	0.535	**0.535**
	4	0.612	0.573	0.591	0.518	**0.616**	0.616
	5	0.671	0.622	0.642	0.551	**0.679**	0.679
C4	2	**0.695**	0.673	0.694	0.633	0.694	0.695
	3	0.773	0.737	0.770	0.691	0.775	**0.776**
	4	0.822	0.781	0.811	0.739	**0.825**	0.824
	5	0.849	0.812	0.834	0.763	0.858	**0.860**
C5	2	**0.561**	0.547	0.558	0.519	0.561	0.561
	3	0.604	0.588	0.602	0.561	0.604	**0.606**
	4	0.662	0.635	0.648	0.595	**0.664**	0.664
	5	0.700	0.661	0.683	0.625	0.708	**0.709**
C6	2	**0.665**	0.642	0.665	0.612	0.665	0.665
	3	**0.728**	0.693	0.725	0.657	0.727	0.728
	4	**0.762**	0.717	0.757	0.689	0.761	0.761
	5	0.804	0.744	0.788	0.713	0.805	**0.806**

Table 7. UQI average measurement of all methods on RGB test images

Images	m	Mean UQI values from Otsu Method as Objective Function					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	0.941	0.939	0.941	0.938	**0.941**	0.941
	3	**0.959**	0.953	0.957	0.949	0.958	0.958
	4	0.968	0.961	0.969	0.958	0.969	**0.969**
	5	0.976	0.968	0.975	0.965	0.977	**0.977**
C2	2	0.895	0.887	**0.896**	0.870	0.895	0.895
	3	0.930	0.919	**0.930**	0.897	0.930	0.930
	4	0.952	0.938	0.950	0.915	0.952	**0.952**
	5	0.962	0.949	0.961	0.929	0.962	**0.962**
C3	2	**0.682**	0.669	0.680	0.653	0.682	0.682
	3	**0.752**	0.736	0.746	0.698	0.752	0.752
	4	**0.798**	0.774	0.783	0.734	0.797	0.797
	5	**0.827**	0.801	0.811	0.768	0.825	0.825
C4	2	0.893	0.885	0.890	0.868	0.893	**0.893**
	3	0.930	0.918	0.927	0.896	**0.931**	0.931
	4	0.949	0.935	0.943	0.916	0.950	**0.950**
	5	0.961	0.948	0.954	0.932	0.962	**0.963**
C5	2	0.801	0.796	0.799	0.769	**0.801**	0.801
	3	0.833	0.832	0.830	0.810	0.833	**0.834**
	4	0.874	0.861	0.863	0.837	**0.874**	0.874
	5	**0.896**	0.877	0.883	0.852	0.896	0.896
C6	2	0.852	0.835	**0.853**	0.818	0.852	0.853
	3	**0.899**	0.884	0.896	0.855	0.898	0.899
	4	**0.924**	0.904	0.920	0.876	0.923	0.923
	5	0.944	0.919	0.935	0.896	0.944	**0.944**
Images	m	**Mean UQI values from Kapur Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	0.933	0.932	0.932	0.901	0.932	**0.933**
	3	**0.949**	0.946	0.949	0.944	0.949	0.949

Table 7. (continued)

Images	m	Mean UQI values from Kapur Entropy as Objective Function					
		PSO	GA	WOA	SMA	GWO	mGWO
C1	4	**0.957**	0.953	0.956	0.955	0.957	0.957
	5	0.962	0.960	0.960	**0.962**	0.961	0.961
C2	2	0.895	0.882	**0.895**	0.694	0.895	0.895
	3	0.930	0.913	**0.930**	0.822	0.930	0.930
	4	0.943	0.936	**0.947**	0.890	0.945	0.944
	5	0.962	0.946	**0.965**	0.919	0.964	0.964
C3	2	0.691	0.681	**0.691**	0.409	0.691	0.691
	3	0.754	0.732	**0.754**	0.655	0.754	0.754
	4	**0.789**	0.761	0.788	0.722	0.789	0.789
	5	**0.804**	0.782	0.803	0.742	0.799	0.797
C4	2	**0.877**	0.870	0.877	0.603	0.877	0.877
	3	0.921	0.903	0.921	0.840	**0.922**	0.922
	4	0.939	0.928	0.940	0.898	0.942	**0.942**
	5	0.954	0.940	0.953	0.914	0.954	**0.954**
C5	2	**0.774**	0.769	0.772	0.508	0.772	0.772
	3	0.834	0.818	**0.834**	0.771	0.834	0.834
	4	0.863	0.845	**0.863**	0.818	0.862	0.862
	5	**0.884**	0.865	0.883	0.834	0.883	0.880
C6	2	0.809	0.821	0.810	0.592	0.821	**0.828**
	3	0.880	0.860	**0.881**	0.829	0.880	0.879
	4	**0.913**	0.894	0.913	0.860	0.912	0.912
	5	**0.933**	0.915	0.933	0.878	0.932	0.932

Images	m	Mean UQI values from M.Masi Entropy as Objective Function					
		PSO	GA	WOA	SMA	GWO	mGWO
C1	2	0.941	0.939	0.941	0.938	**0.941**	0.941
	3	**0.959**	0.953	0.958	0.949	0.958	0.958
	4	0.969	0.962	0.968	0.959	0.969	**0.969**
	5	0.977	0.970	0.975	0.964	0.977	**0.977**
C2	2	0.895	0.888	**0.896**	0.864	0.895	0.895
	3	0.930	0.920	**0.930**	0.899	0.930	0.930
	4	0.952	0.938	0.951	0.914	**0.952**	0.952
	5	0.962	0.948	0.961	0.930	0.962	**0.962**
C3	2	0.682	0.668	0.679	0.651	**0.682**	0.682
	3	0.752	0.734	0.745	0.706	0.752	**0.752**
	4	0.797	0.774	0.784	0.738	0.797	**0.797**
	5	0.825	0.801	0.812	0.760	**0.826**	0.825
C4	2	**0.893**	0.886	0.893	0.870	0.893	0.893
	3	0.930	0.916	0.928	0.898	0.930	**0.931**
	4	0.949	0.937	0.945	0.918	**0.950**	0.950
	5	0.960	0.949	0.954	0.929	0.962	**0.963**
C5	2	**0.801**	0.793	0.800	0.771	0.801	0.801
	3	0.833	0.830	0.831	0.808	0.833	**0.834**
	4	0.873	0.861	0.865	0.835	0.874	**0.874**
	5	0.895	0.877	0.884	0.850	0.896	**0.896**
C6	2	0.853	0.838	**0.853**	0.822	0.853	0.853
	3	**0.899**	0.882	0.895	0.855	0.898	0.899
	4	**0.924**	0.899	0.921	0.881	0.923	0.923
	5	0.943	0.915	0.935	0.895	0.943	**0.944**

Table 8. CPU Time (seconds) average measurement of all methods on RGB test images

Images	m	Mean CPU Time values from Otsu Method as Objective Function					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	3.611	3.226	3.042	3.426	**1.189**	2.999
	3	3.580	3.776	3.666	4.018	**1.649**	3.497
	4	3.833	3.281	3.671	3.705	**1.586**	3.751
	5	3.860	3.623	3.876	4.147	**1.873**	4.162
C2	2	3.688	**3.227**	3.430	3.537	3.454	4.170
	3	3.817	**3.310**	3.380	3.700	3.609	4.356
	4	3.838	**3.385**	3.496	3.647	3.785	4.405
	5	4.043	**3.441**	3.562	3.680	4.323	4.391
C3	2	4.453	**4.002**	4.158	4.220	4.774	4.590
	3	4.403	**4.133**	4.141	4.309	4.843	5.055
	4	4.359	**4.225**	4.537	4.607	4.918	4.952
	5	4.403	**4.235**	4.555	4.879	5.191	4.985
C4	2	3.728	**3.493**	3.533	3.836	4.507	4.269
	3	3.749	**3.363**	3.518	3.688	4.569	4.340
	4	3.836	**3.387**	3.488	3.758	4.506	4.337
	5	3.710	**3.480**	3.623	3.672	4.723	4.333
C5	2	3.597	**3.236**	3.363	3.671	4.442	4.119
	3	3.663	**3.377**	3.446	3.646	4.383	4.328
	4	**3.045**	3.430	3.559	3.683	4.548	4.315
	5	3.560	**3.371**	3.492	3.704	4.490	4.222
C6	2	**2.872**	3.246	3.290	3.495	4.413	3.541
	3	3.353	**3.266**	3.369	3.444	4.288	3.937
	4	**2.432**	3.353	3.517	3.736	4.204	3.457
	5	**2.042**	3.357	3.426	3.547	4.478	3.640
Images	m	**Mean CPU Time values from Kapur Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	8.377	8.798	7.202	8.471	**3.694**	7.995
	3	8.701	8.838	8.989	9.606	**3.747**	7.759
	4	8.655	8.628	8.835	8.887	**6.765**	8.891
	5	8.598	8.166	**7.979**	8.418	8.057	9.075
C2	2	8.955	**7.823**	8.144	8.131	8.860	8.868
	3	8.697	8.129	8.229	**8.030**	9.111	9.094
	4	9.288	8.730	**8.549**	9.194	9.151	9.451
	5	8.907	**8.531**	8.718	8.737	9.507	9.341
C3	2	10.237	10.346	10.138	**9.892**	10.567	10.596
	3	**9.966**	10.232	10.012	10.025	10.805	10.900
	4	10.097	10.266	**9.706**	10.280	11.078	10.646
	5	9.981	**9.715**	9.754	9.911	10.914	10.567
C4	2	9.037	9.009	8.973	**8.826**	10.264	9.466
	3	9.312	**9.075**	9.140	9.566	10.417	9.723
	4	**8.882**	9.092	8.981	9.310	10.222	9.732
	5	**7.591**	8.873	8.975	9.236	9.846	8.913
C5	2	**7.845**	8.644	8.695	8.422	9.681	8.731
	3	**5.422**	8.446	8.789	7.252	8.737	7.592
	4	4.102	6.970	8.423	**3.982**	9.046	8.022
	5	**3.437**	3.891	4.621	4.133	7.817	5.128
C6	2	**3.363**	3.941	3.771	3.866	7.958	3.691
	3	**2.942**	3.799	3.947	4.030	5.261	3.630
	4	7.687	3.886	3.909	3.686	3.797	**3.323**
	5	**2.377**	3.670	3.949	2.875	3.836	7.254
Images	m	**Mean CPU Time from M.Masi Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**SMA**	**GWO**	**mGWO**
C1	2	3.580	3.459	3.563	3.559	**1.545**	3.529
	3	3.603	3.732	3.583	4.319	**1.527**	3.624

Table 8. (continued)

Images	m	Mean CPU Time values from M.Masi Entropy as Objective Function					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	4	3.893	3.488	3.781	**1.729**	3.977	4.193
	5	3.832	3.650	3.803	**2.950**	4.056	4.225
C2	2	3.805	**3.139**	3.650	3.623	3.530	4.345
	3	3.877	**3.314**	3.660	3.612	3.646	4.293
	4	4.015	**3.407**	3.728	4.200	3.688	4.329
	5	4.130	**3.395**	3.814	4.192	3.879	4.516
C3	2	4.441	4.123	**4.095**	4.718	4.419	4.746
	3	4.546	**4.160**	4.165	4.774	4.557	4.799
	4	4.367	4.057	**3.973**	5.088	4.474	5.012
	5	4.353	4.290	**4.156**	5.084	4.588	5.026
C4	2	3.641	3.383	**3.308**	4.246	3.800	4.316
	3	3.676	**3.349**	3.483	4.495	3.607	4.443
	4	3.792	**3.423**	3.449	4.618	3.723	4.240
	5	3.910	3.384	**2.547**	4.619	3.738	4.300
C5	2	3.680	3.399	**1.428**	4.403	3.756	4.316
	3	3.497	3.386	**1.405**	4.355	3.714	4.099
	4	3.305	3.449	**1.480**	4.286	3.684	4.232
	5	3.210	3.386	**1.544**	4.262	3.783	3.985
C6	2	2.909	3.251	**1.398**	4.091	3.536	3.903
	3	3.225	3.290	**1.472**	4.187	3.579	3.309
	4	2.291	3.288	**1.272**	4.339	3.656	3.290
	5	1.910	3.185	**1.089**	4.271	3.320	3.715

Table 9. Fitness average measurement of all methods on RGB test images on R, G, and B channels with the objective function of the Otsu method

Images	m	Mean Fitness values in R Channel					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	2	930.762	919.050	930.740	**930.781**	890.590	930.781
	3	970.840	956.487	970.807	**971.128**	927.140	971.128
	4	993.832	973.031	994.306	994.220	949.327	**994.897**
	5	1004.898	987.546	1005.243	1006.147	964.461	**1006.660**
C2	2	**1004.914**	984.406	1004.875	1004.914	922.195	1004.914
	3	1069.910	1044.729	1069.859	**1070.116**	970.054	1070.116
	4	1096.416	1069.717	1092.732	**1097.085**	1021.011	1097.074
	5	1115.080	1088.962	1107.648	1115.600	1026.495	**1116.256**
C3	2	**2947.386**	2916.430	2947.320	2947.385	2809.961	2947.385
	3	3126.543	3061.943	3108.334	3126.722	2922.787	**3126.724**
	4	3208.291	3142.718	3188.828	3208.640	3011.447	**3208.647**
	5	3253.597	3194.452	3225.206	**3254.497**	3100.713	3252.983
C4	2	3359.863	3316.971	3359.746	**3359.864**	3183.303	3359.864
	3	3555.663	3497.021	3536.021	3555.743	3380.285	**3555.749**
	4	3642.833	3577.633	3615.202	3642.624	3466.711	**3645.612**
	5	3686.684	3636.917	3649.479	3690.552	3556.294	**3693.733**
C5	2	**1656.201**	1626.875	1639.955	1656.201	1493.777	1656.201
	3	1777.165	1730.146	1764.577	1777.168	1628.346	**1777.170**
	4	1848.684	1800.352	1828.548	**1848.959**	1704.287	1848.954
	5	1880.803	1840.142	1859.133	1880.731	1759.992	**1881.365**
C6	2	5640.404	5609.146	5640.416	**5640.462**	5546.601	5640.462
	3	5740.058	5712.943	5736.878	5740.294	5662.826	**5740.298**
	4	5794.173	5751.694	5788.670	5796.739	5687.418	**5796.740**
	5	5827.737	5783.710	5805.951	**5830.047**	5745.419	5829.838
Images	m	**Mean Fitness values in G Channel**					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	2436.207	2419.379	2436.307	2436.358	2371.651	**2436.358**
	3	2533.905	2507.081	2517.659	**2534.075**	2450.084	2534.075

Table 9. (continued)

Images	m	Mean Fitness values in G Channel					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	4	2578.356	2549.020	2581.847	2587.570	2499.664	**2587.613**
	5	2611.414	2582.319	2600.576	2616.790	2546.931	**2618.815**
C2	2	**2393.647**	2356.467	2393.620	2393.647	2223.839	2393.647
	3	2597.968	2541.443	2597.723	2597.979	2381.003	**2597.979**
	4	2675.494	2614.947	2667.393	**2675.653**	2463.977	2675.647
	5	2709.496	2656.818	2702.795	**2709.778**	2552.235	2709.777
C3	2	2947.385	2884.685	2947.320	**2947.386**	2781.179	2947.386
	3	3126.663	3071.220	3114.387	3126.722	2891.408	**3126.722**
	4	3207.865	3139.325	3174.867	3205.916	3019.545	**3208.641**
	5	3251.522	3191.409	3209.510	3254.481	3114.993	**3254.537**
C4	2	2004.697	1969.444	1983.767	**2004.700**	1891.101	2004.699
	3	2113.172	2074.714	2098.886	2113.660	1986.858	**2113.662**
	4	2166.672	2125.517	2142.863	**2169.763**	2054.947	2169.755
	5	2195.747	2157.027	2171.431	2197.531	2093.953	**2202.933**
C5	2	5203.917	5176.518	5203.887	**5203.923**	5105.570	5203.921
	3	5382.551	5319.872	5370.345	**5382.576**	5245.609	5382.572
	4	5476.443	5414.126	5456.186	5479.696	5324.046	**5479.710**
	5	5520.237	5455.785	5491.853	5525.365	5386.948	**5525.386**
C6	2	5622.879	5594.550	5636.606	5636.630	5487.121	**5636.631**
	3	5776.054	5735.206	5757.919	5772.034	5668.646	**5776.703**
	4	5861.275	5805.208	5849.251	5868.059	5719.417	**5868.073**
	5	5916.046	5850.727	5899.985	**5919.268**	5784.950	5919.268

Images	m	Mean Fitness values in B Channel					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	2	1787.898	1771.365	1787.841	**1787.924**	1743.623	1787.923
	3	1856.065	1836.678	1856.242	1856.409	1806.024	**1856.439**
	4	1900.109	1867.016	1901.257	1901.873	1840.299	**1901.877**
	5	1921.447	1889.702	1921.333	1923.262	1862.135	**1923.362**
C2	2	2181.629	2151.060	2181.737	**2181.867**	2108.303	2181.865
	3	2269.323	2232.839	2270.203	**2270.494**	2195.854	2270.493
	4	2316.704	2282.422	2319.591	**2320.314**	2232.543	2320.302
	5	2336.270	2305.161	2338.494	**2339.777**	2276.813	2339.687
C3	2	2947.385	2920.441	2933.170	2947.385	2824.493	**2947.386**
	3	3126.676	3066.389	3114.553	**3126.723**	2952.785	3126.721
	4	3208.182	3156.424	3173.948	3208.635	3030.060	**3208.643**
	5	3251.788	3188.168	3234.641	3254.533	3100.400	**3254.549**
C4	2	2632.667	2610.907	2608.297	**2640.869**	2511.415	2640.866
	3	2828.958	2773.313	2822.615	**2829.212**	2681.343	2829.210
	4	2905.125	2859.266	2890.916	2922.725	2765.213	**2922.730**
	5	2958.774	2902.184	2950.859	**2970.229**	2835.987	2968.653
C5	2	**1759.218**	1726.617	1759.163	1759.218	1637.222	1759.218
	3	1876.237	1842.664	1864.025	1881.253	1762.638	**1885.459**
	4	1929.065	1897.497	1909.297	**1936.467**	1828.987	1936.467
	5	1954.269	1924.743	1948.911	1964.592	1868.647	**1965.604**
C6	2	**1589.872**	1560.152	1589.702	1581.595	1513.927	1589.872
	3	1694.029	1660.467	1693.884	**1694.067**	1568.224	1694.065
	4	1742.400	1700.177	1735.440	1740.932	1635.498	**1742.548**
	5	1770.146	1727.333	1762.456	**1770.879**	1670.836	1770.872

Table 10. Fitness average measurement of all methods on RGB test images on R, G, and B channels with the objective function of the Kapur Entropy

Images	m	Mean Fitness values in R Channel					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	11.516	11.386	11.513	11.515	11.067	**11.516**
	3	14.738	14.347	14.747	14.750	13.757	**14.750**
	4	17.444	16.970	17.475	**17.488**	16.057	17.488
	5	19.944	19.254	20.004	20.063	17.629	**20.073**
C2	2	**11.612**	11.543	11.612	11.612	5.047	11.612
	3	14.466	14.178	14.465	14.467	9.490	**14.467**
	4	16.984	16.651	16.967	16.989	14.346	**16.993**
	5	19.448	18.778	19.457	**19.481**	16.863	19.481
C3	2	**12.635**	12.525	12.635	12.635	5.215	12.635
	3	15.809	15.549	15.807	15.810	13.748	**15.810**
	4	18.683	18.299	18.579	**18.686**	17.267	18.686
	5	21.458	20.960	21.470	21.529	19.156	**21.536**
C4	2	**12.954**	12.895	12.953	12.954	5.529	12.954
	3	16.134	15.957	16.135	**16.136**	14.194	16.136
	4	19.103	18.808	19.104	**19.106**	17.834	19.106
	5	21.897	21.380	21.907	**21.926**	19.903	21.926
C5	2	12.467	12.404	12.473	**12.473**	5.139	12.473
	3	15.616	15.370	15.615	15.617	13.779	**15.617**
	4	18.431	18.079	18.434	18.442	17.034	**18.442**
	5	21.042	20.593	21.048	21.064	18.832	**21.065**
C6	2	**12.354**	12.242	12.353	12.354	5.000	12.354
	3	15.515	15.295	15.515	15.516	14.108	**15.516**
	4	18.358	17.971	18.370	18.382	16.837	**18.383**
	5	20.928	20.493	20.936	20.966	18.749	**20.968**
Images	m	**Mean Fitness values in G Channel**					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	**12.527**	12.448	12.527	12.527	5.462	12.527
	3	15.806	15.497	15.806	**15.808**	14.542	15.808
	4	18.676	18.275	18.670	18.681	17.111	**18.682**
	5	21.372	20.884	21.276	**21.384**	19.175	21.383
C2	2	**12.699**	12.608	12.699	12.699	9.471	12.699
	3	15.766	15.504	15.764	**15.766**	14.749	15.766
	4	18.587	18.199	18.579	**18.587**	17.037	18.587
	5	21.235	20.567	21.218	21.239	19.317	**21.239**
C3	2	**12.635**	12.492	12.634	12.635	5.431	12.635
	3	15.808	15.526	15.808	15.810	13.054	**15.810**
	4	18.684	18.340	18.678	18.685	17.453	**18.686**
	5	21.475	20.941	21.358	21.431	19.224	**21.543**
C4	2	**12.374**	12.277	12.374	12.374	6.336	12.374
	3	15.506	15.325	15.503	**15.507**	14.176	15.507
	4	18.307	18.003	18.226	18.329	16.749	**18.329**
	5	21.015	20.455	21.022	**21.039**	18.800	21.035
C5	2	12.631	12.495	12.631	**12.631**	5.404	12.631
	3	15.773	15.520	15.775	**15.777**	13.600	15.777
	4	18.591	18.310	18.597	**18.600**	17.067	18.599
	5	21.161	20.642	21.154	21.169	18.791	**21.201**
C6	2	**12.913**	12.828	12.913	12.913	5.746	12.913
	3	16.142	15.863	16.142	16.144	15.101	**16.144**
	4	19.034	18.645	19.043	19.050	17.296	**19.050**
	5	21.711	21.259	21.715	21.728	19.622	**21.731**
Images	m	**Mean Fitness values in B Channel**					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	12.141	12.059	12.141	**12.141**	11.852	12.141

Table 10. (continued)

Images	m	Mean Fitness values in B Channel					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	3	15.339	15.096	15.342	15.344	14.584	**15.344**
	4	18.076	17.596	18.098	**18.109**	16.633	18.109
	5	20.539	20.012	20.555	20.591	18.690	**20.592**
C2	2	12.068	11.993	12.068	**12.069**	11.747	12.069
	3	15.009	14.768	15.021	**15.023**	14.404	15.023
	4	17.617	17.311	17.662	17.667	16.248	**17.669**
	5	20.040	19.605	20.088	20.131	18.036	**20.156**
C3	2	**12.635**	12.468	12.634	12.635	5.436	12.635
	3	15.807	15.537	15.808	15.810	14.174	**15.810**
	4	18.679	18.307	18.677	18.686	17.447	**18.686**
	5	21.453	20.815	21.352	21.504	19.172	**21.532**
C4	2	**12.795**	12.741	12.795	12.795	5.587	12.795
	3	15.993	15.856	15.994	15.996	14.264	**15.996**
	4	19.065	18.762	19.062	**19.065**	17.924	19.065
	5	21.936	21.397	21.931	**21.944**	19.950	21.943
C5	2	12.516	12.413	12.516	**12.516**	5.087	12.516
	3	15.539	15.298	15.438	15.539	14.318	**15.539**
	4	18.400	18.039	18.402	18.407	16.929	**18.407**
	5	20.951	20.498	20.864	20.962	18.529	**20.964**
C6	2	**12.107**	11.953	12.103	12.066	7.700	12.044
	3	15.354	14.946	15.352	**15.354**	14.271	15.354
	4	18.230	17.722	18.222	**18.232**	16.410	18.232
	5	20.888	20.261	20.884	20.896	18.297	**20.896**

Table 11. Fitness average measurement of all methods on RGB test images on R, G, and B channels with the objective function of the M.Masi Entropy

Images	m	Mean Fitness values in R Channel					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	2	930.731	917.873	930.752	**930.781**	879.106	930.781
	3	970.797	956.869	970.970	**971.129**	932.104	971.128
	4	993.744	973.081	992.437	**994.897**	946.431	994.880
	5	1004.753	987.794	1004.667	1006.508	970.986	**1006.577**
C2	2	1004.911	990.974	1004.843	**1004.914**	884.842	1004.914
	3	1069.878	1043.914	1067.697	1070.117	972.743	**1070.118**
	4	1096.556	1071.061	1092.815	**1097.065**	1023.235	1097.040
	5	1115.079	1081.324	1111.937	1115.595	1041.978	**1116.904**
C3	2	2947.385	2917.452	2947.332	2947.385	2792.647	**2947.386**
	3	3126.628	3066.076	3114.574	3126.722	2959.915	**3126.725**
	4	3207.917	3150.512	3188.400	3208.630	3039.085	**3208.639**
	5	3253.366	3203.172	3218.331	**3254.531**	3074.506	3254.508
C4	2	3359.861	3316.511	3359.758	**3359.865**	3224.688	3359.865
	3	3555.446	3502.350	3548.716	**3555.756**	3359.682	3555.754
	4	3645.202	3575.818	3617.772	3645.612	3474.785	**3645.617**
	5	3686.227	3633.776	3661.294	3692.160	3521.240	**3693.722**
C5	2	**1656.201**	1635.364	1656.169	1656.201	1517.512	1656.201
	3	1777.169	1740.934	1760.767	**1777.171**	1644.619	1777.169
	4	1848.726	1799.461	1831.552	**1848.964**	1717.972	1848.956
	5	1878.275	1841.351	1859.689	**1881.368**	1763.625	1881.365
C6	2	**5640.462**	5613.692	5640.442	5640.462	5568.398	5640.462
	3	5739.726	5706.655	5730.245	**5740.297**	5644.144	5740.295
	4	5795.379	5748.885	5783.229	5796.738	5703.263	**5796.739**
	5	5825.148	5784.647	5807.802	**5831.017**	5736.652	5831.017
Images	m	Mean Fitness values in G Channel					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	2	2436.358	2412.740	2436.331	**2436.358**	2377.766	2436.358

Table 11. (continued)

Images	m	Mean Fitness values in G Channel					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	3	2533.604	2505.838	2530.507	**2534.074**	2455.863	2534.073
	4	2579.730	2554.088	2578.224	2587.584	2513.719	**2587.598**
	5	2615.780	2581.615	2604.602	2618.867	2549.214	**2618.874**
C2	2	**2393.647**	2356.674	2393.609	2393.647	2205.639	2393.647
	3	2597.968	2524.988	2597.735	2597.978	2397.751	**2597.981**
	4	2675.516	2610.729	2669.697	**2675.657**	2472.629	2675.656
	5	2709.477	2657.052	2697.428	**2709.783**	2554.147	2709.776
C3	2	2947.366	2902.541	2947.316	**2947.386**	2780.989	2947.386
	3	3126.669	3076.910	3108.234	**3126.722**	2943.562	3126.720
	4	3207.544	3150.131	3180.082	**3208.637**	3029.700	3208.621
	5	3247.182	3202.194	3222.764	3254.471	3078.551	**3254.525**
C4	2	2004.694	1983.123	1994.227	2004.698	1860.440	**2004.700**
	3	2111.910	2071.018	2098.996	2110.028	1987.027	**2113.658**
	4	2163.809	2132.032	2148.707	**2169.776**	2041.162	2167.896
	5	2192.779	2157.626	2173.026	2201.837	2082.568	**2202.911**
C5	2	**5203.925**	5180.416	5203.865	5203.923	5113.816	5203.919
	3	5382.496	5323.371	5370.450	5382.574	5260.021	**5382.569**
	4	5474.564	5416.974	5466.213	5479.699	5299.996	**5479.697**
	5	5517.672	5469.007	5500.483	5524.891	5387.711	**5525.361**
C6	2	5636.629	5593.344	5636.573	**5636.631**	5516.568	5636.631
	3	5776.053	5733.161	5757.918	5776.703	5666.116	**5776.704**
	4	5863.682	5807.840	5858.133	**5868.075**	5740.463	5868.065
	5	5914.152	5857.316	5893.090	5917.574	5776.310	**5919.307**

Images	m	Mean Fitness values in B Channel					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	1787.891	1773.193	1787.887	**1787.923**	1733.958	1787.920
	3	1856.003	1831.535	1856.265	1856.429	1809.108	**1856.433**
	4	1900.046	1868.262	1901.251	1901.869	1842.929	**1901.900**
	5	1920.789	1893.597	1922.045	1923.295	1867.291	**1923.380**
C2	2	2181.686	2144.013	2181.797	2181.865	2100.755	**2181.867**
	3	2269.028	2235.984	2270.142	2270.488	2203.302	**2270.497**
	4	2316.356	2280.854	2319.137	**2320.329**	2229.249	2320.325
	5	2336.473	2306.348	2338.228	2339.664	2267.259	**2339.668**
C3	2	2947.384	2917.453	2919.010	2947.385	2806.220	**2947.386**
	3	3126.660	3074.673	3102.451	**3126.724**	2927.363	3126.722
	4	3207.539	3141.863	3178.042	**3208.645**	3031.222	3208.636
	5	3245.990	3197.677	3232.636	3254.506	3099.236	**3254.544**
C4	2	**2640.869**	2604.334	2640.737	2624.620	2503.454	2640.869
	3	2821.331	2778.981	2822.713	**2829.212**	2659.081	2829.206
	4	2915.753	2858.170	2906.365	**2919.722**	2764.932	2919.643
	5	2953.380	2907.666	2931.250	2967.004	2830.765	**2970.217**
C5	2	**1759.218**	1733.750	1726.893	1759.218	1652.187	1759.218
	3	1872.617	1840.584	1880.956	1872.835	1748.975	**1885.459**
	4	1926.926	1901.020	1912.204	1934.760	1809.410	**1934.762**
	5	1953.181	1926.460	1945.912	1964.533	1868.084	**1966.602**
C6	2	**1589.872**	1568.114	1589.785	1589.872	1507.598	1589.872
	3	1694.036	1653.310	1693.881	1690.597	1605.672	**1694.067**
	4	1742.298	1698.067	1737.111	1742.551	1627.272	**1742.553**
	5	**1770.235**	1729.862	1760.964	1768.058	1668.050	1769.006

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

Table 12. Measurement of the average fitness of all methods on grayscale test images

Images	m	Mean Fitness values from Otsu Method as Objective Function					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	1948.708	1933.456	1948.684	**1948.718**	1899.311	1948.718
	3	2024.618	2002.246	2024.680	2024.823	1968.045	**2024.828**
	4	2068.925	2040.525	2068.739	2069.969	2012.155	**2069.976**
	5	2093.558	2064.616	2094.259	2095.748	2037.178	**2095.894**
C2	2	1961.753	1934.919	1961.796	1961.821	1860.228	**1961.822**
	3	2127.064	2083.094	2128.144	2128.299	1971.929	**2128.299**
	4	2188.941	2146.156	2190.858	**2191.870**	2062.143	2191.870
	5	2213.168	2170.434	2215.334	2217.139	2092.731	**2217.501**
C3	2	**2947.386**	2905.317	2947.268	2947.386	2794.870	2947.385
	3	3126.660	3068.972	3126.551	3126.722	2907.334	**3126.724**
	4	3206.902	3139.550	3197.226	**3208.636**	3028.555	3208.623
	5	3252.227	3194.267	3221.194	3253.023	3077.447	**3254.536**
C4	2	1549.025	1526.467	1548.960	**1549.027**	1451.079	1549.027
	3	1638.286	1607.980	1624.148	1636.582	1523.064	**1639.462**
	4	1689.293	1650.866	1667.464	1693.111	1559.626	**1693.113**
	5	1708.055	1678.080	1684.396	1717.071	1617.407	**1718.173**
C5	2	2532.097	2499.146	2532.073	**2532.098**	2410.568	2532.098
	3	2702.688	2641.840	2697.320	**2703.333**	2558.624	2697.625
	4	2761.601	2721.074	2730.366	**2766.087**	2652.294	2765.924
	5	2802.491	2755.961	2776.141	2810.495	2687.477	**2810.506**
C6	2	**3975.630**	3955.353	3975.587	3975.630	3896.051	3975.624
	3	**4113.418**	4059.638	4113.212	4108.823	3987.953	4113.412
	4	4182.113	4124.539	4167.574	**4182.129**	4057.375	4182.099
	5	4217.800	4158.330	4211.061	4217.449	4107.980	**4217.993**
Images	m	**Mean Fitness values from Kapur Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	12.211	12.145	12.211	**12.211**	11.985	12.211
	3	15.500	15.252	15.502	15.504	14.676	**15.504**
	4	18.295	17.884	18.298	**18.311**	16.843	18.311
	5	20.870	20.295	20.867	20.902	18.599	**20.907**
C2	2	12.346	12.219	12.346	**12.346**	11.966	12.346
	3	15.314	15.136	15.317	**15.318**	14.530	15.318
	4	17.981	17.704	**18.007**	18.003	16.902	18.005
	5	20.527	20.028	20.586	20.609	18.018	**20.609**
C3	2	**12.635**	12.520	12.634	12.635	5.215	12.635
	3	15.810	15.503	15.808	**15.810**	12.835	15.810
	4	**18.686**	18.342	18.677	18.685	17.222	18.685
	5	21.492	20.896	21.469	**21.547**	19.890	21.546
C4	2	**12.218**	12.143	12.218	12.218	5.780	12.218
	3	15.278	15.056	15.277	**15.279**	12.966	15.279
	4	18.123	17.775	17.932	**18.124**	16.570	18.124
	5	20.762	20.190	20.756	20.698	18.367	**20.788**
C5	2	**12.635**	12.530	12.634	12.635	5.263	12.635
	3	15.688	15.518	15.688	15.689	12.888	**15.689**
	4	18.488	18.248	18.505	18.524	17.255	**18.529**
	5	21.222	20.788	21.258	**21.280**	19.364	21.279
C6	2	**12.525**	12.410	12.525	12.525	6.749	12.525
	3	15.565	15.336	15.565	**15.566**	14.609	15.566
	4	18.362	17.976	18.361	**18.368**	16.966	18.368
	5	**21.021**	20.345	20.940	20.987	18.831	20.994
Images	m	**Mean Fitness values from M.Masi Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	1948.707	1931.621	1948.702	**1948.718**	1915.388	1948.718
	3	2024.607	2003.156	2024.650	2024.824	1973.501	**2024.827**

Table 12. (continued)

Images	m	Mean Fitness values from M.Masi Entropy as Objective Function					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	4	2069.015	2043.645	2068.722	2070.010	2006.877	**2070.053**
	5	2093.320	2063.197	2093.977	2095.876	2034.592	**2095.879**
C2	2	1961.738	1931.822	1961.795	**1961.822**	1884.466	1961.822
	3	2127.348	2082.519	2128.012	**2128.299**	1996.689	2128.297
	4	2188.138	2143.223	2189.047	**2191.864**	2067.696	2191.860
	5	2213.486	2182.660	2214.424	2217.367	2119.535	**2217.462**
C3	2	**2947.386**	2887.823	2919.039	2947.385	2791.990	2947.386
	3	3126.693	3071.772	3114.506	**3126.722**	2936.842	3126.721
	4	3208.067	3142.152	3183.510	3208.638	2992.283	**3208.648**
	5	3247.404	3189.892	3214.547	3253.017	3087.796	**3254.477**
C4	2	1549.024	1517.362	1548.990	**1549.027**	1430.280	1549.027
	3	1635.852	1606.022	1603.138	**1639.465**	1510.104	1639.458
	4	1683.206	1647.763	1667.403	1691.357	1588.507	**1691.430**
	5	1712.370	1678.225	1684.892	1718.001	1613.917	**1718.860**
C5	2	**2532.098**	2483.827	2518.639	2532.098	2395.733	2532.098
	3	2703.316	2644.461	2686.068	2697.625	2535.022	**2703.331**
	4	2759.585	2728.074	2746.195	**2766.167**	2644.864	2765.993
	5	2798.609	2766.935	2779.814	2809.024	2688.460	**2810.500**
C6	2	**3975.630**	3934.453	3975.507	3975.628	3895.119	3975.630
	3	**4113.419**	4075.050	4108.515	4113.414	4008.455	4113.414
	4	4182.076	4123.197	4179.263	4182.122	4040.780	**4182.126**
	5	4217.926	4166.966	4210.173	**4217.990**	4128.990	4217.988

Table 13. Measurement of the average CPU Time (seconds) of all methods on grayscale test images

Images	m	Mean CPU Time values from Otsu Method as Objective Function					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	2	0.672	1.149	**0.568**	1.515	0.605	1.130
	3	1.374	1.200	1.238	1.535	1.270	**1.110**
	4	1.337	1.238	1.292	1.678	1.350	**1.000**
	5	1.429	1.275	1.376	1.649	1.484	**0.839**
C2	2	1.367	1.218	1.246	1.541	1.301	**0.763**
	3	1.386	1.238	1.254	1.715	1.327	**0.770**
	4	1.479	1.229	1.361	1.617	1.376	**0.767**
	5	1.354	1.378	1.327	1.721	1.406	**1.032**
C3	2	1.655	**1.526**	1.591	1.854	1.652	1.658
	3	1.664	1.519	1.580	1.866	1.625	**0.715**
	4	1.662	1.569	1.603	1.915	1.647	**0.753**
	5	1.645	1.530	1.599	1.901	1.709	**0.734**
C4	2	1.341	1.233	1.180	1.597	1.314	**0.603**
	3	1.391	1.265	1.270	1.600	1.366	**0.621**
	4	1.413	1.235	1.301	1.602	1.391	**0.628**
	5	1.392	1.256	1.304	1.618	1.365	**0.629**
C5	2	1.333	1.197	1.241	1.535	1.289	**0.597**
	3	1.352	1.218	1.282	1.576	1.306	**0.622**
	4	1.369	1.218	1.302	1.568	1.361	**0.621**
	5	1.386	1.247	1.315	1.689	1.365	**0.644**
C6	2	1.348	1.253	1.249	1.557	1.328	**0.595**
	3	1.418	**1.243**	1.318	1.562	1.367	1.793
	4	1.370	**1.245**	1.278	1.279	1.379	2.037
	5	1.380	1.025	1.258	0.949	1.377	**0.765**
Images	m	**Mean CPU Time values from Kapur Entropy as Objective Function**					
		PSO	**GA**	**WOA**	**GWO**	**SMA**	**mGWO**
C1	2	2.864	**0.991**	4.393	2.495	2.219	2.426
	3	3.741	3.520	4.085	4.098	**2.195**	2.213

Table 13. (continued)

Images	m	Mean CPU Time values from Kapur Entropy as Objective Function					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	4	3.915	3.502	3.940	4.413	1.987	**1.817**
	5	3.565	3.608	4.549	3.911	1.684	**1.674**
C2	2	3.819	3.572	4.115	3.827	**1.413**	1.689
	3	3.746	3.373	4.039	4.046	**1.650**	2.316
	4	4.247	3.807	4.424	4.185	**1.694**	5.102
	5	3.754	4.001	4.098	4.087	**1.780**	1.832
C3	2	5.240	4.987	4.727	5.495	**1.549**	1.580
	3	5.494	5.093	3.526	5.209	**1.458**	1.613
	4	5.327	5.018	2.166	5.401	1.670	**1.584**
	5	5.541	5.765	**1.362**	5.656	1.734	1.615
C4	2	4.021	4.273	**1.246**	4.439	1.479	1.354
	3	4.569	4.142	2.009	5.636	**1.294**	1.391
	4	4.426	5.106	1.448	4.241	1.407	**1.382**
	5	4.403	4.152	**1.307**	4.401	1.492	1.387
C5	2	4.484	3.999	1.253	4.616	**1.023**	4.625
	3	4.554	4.508	1.798	4.337	**1.095**	2.953
	4	4.005	4.159	1.736	4.538	1.047	**0.975**
	5	4.585	3.907	1.062	4.692	1.042	**0.825**
C6	2	3.772	4.393	1.740	3.523	1.066	**0.834**
	3	2.874	3.415	2.185	2.758	**0.877**	0.878
	4	2.651	2.525	2.639	2.623	**0.905**	1.207
	5	2.060	2.572	2.222	2.191	**0.969**	1.043

Images	m	Mean CPU Time values from M.Masi Entropy as Objective Function					
		PSO	GA	WOA	GWO	SMA	mGWO
C1	2	1.811	**0.748**	1.894	1.994	1.647	1.138
	3	1.891	**1.024**	1.598	2.294	1.751	1.087
	4	1.923	**0.876**	1.546	2.000	1.916	0.956
	5	1.682	0.877	1.645	1.981	1.672	**0.841**
C2	2	1.723	**0.744**	1.633	1.865	1.637	0.770
	3	1.748	**0.604**	1.664	1.962	1.664	0.809
	4	1.862	**0.577**	1.887	2.207	1.802	0.766
	5	1.952	**0.572**	1.609	2.338	1.911	1.053
C3	2	2.328	**0.774**	2.233	2.501	2.250	1.594
	3	2.495	**0.634**	2.176	2.653	2.352	0.722
	4	2.369	**0.611**	2.298	2.657	2.461	0.730
	5	2.575	**0.540**	2.370	2.689	2.584	0.743
C4	2	2.095	**0.591**	1.750	2.341	1.998	0.603
	3	2.030	0.639	1.772	2.274	2.014	**0.613**
	4	1.828	**0.511**	2.028	2.114	1.895	0.623
	5	2.229	**0.584**	2.349	3.122	1.970	0.622
C5	2	2.262	0.620	1.802	2.257	2.475	**0.601**
	3	2.063	**0.559**	1.841	2.566	2.116	0.626
	4	2.038	**0.527**	2.040	2.410	1.929	0.616
	5	2.235	**0.480**	1.828	2.508	2.269	0.625
C6	2	1.942	**0.501**	1.708	2.125	1.804	0.583
	3	1.954	**0.471**	1.738	2.157	1.752	1.996
	4	1.908	**0.568**	2.027	2.453	1.860	1.986
	5	2.289	**0.446**	1.511	2.167	2.235	0.622

Table 14. Comparison of the performance of the mGWO proposed methods based on the Otsu method with state-of-the-art KHO [15], WOA, MFO [18], and GWO [14] to solve ML-ISP on grayscale test images. The value in bold is the best value.

Images	m	Mean Fitness value				
		GWO [14]	KHO [15]	WOA [18]	MFO [18]	Proposed
C1	2	-	1845.4988	1942.845	1945.1633	**1948.718**
	3	-	1930.3687	2022.589	1996.8834	**2024.828**
	4	2069.94	1959.0374	2054.8458	2061.949	**2069.976**
	5	**2096.12**	2003.1864	2074.8249	2080.169	2095.894
C2	2	-	1953.4586	-	-	**1961.822**
	3	-	**2130.6885**	-	-	2128.299
	4	2191.84	**2202.0374**	-	-	2191.870
	5	2217.34	**2219.1162**	-	-	2217.501
C3	2	-	**3045.8975**	-	-	2947.385
	3	-	**3219.0868**	-	-	3126.724
	4	3210.62	**3300.7269**	-	-	3208.623
	5	3256.52	**3315.9942**	-	-	3254.536
C4	2	-	1519.2687	1545.9279	1538.8138	**1549.027**
	3	-	**1639.6722**	1635.7034	1592.9889	1639.462
	4	1692.14	1691.9691	1669.8319	1647.9387	**1693.113**
	5	1717.81	**1726.3814**	1682.4839	1705.9335	1718.173
C5	2	-	2469.3327	2433.3641	2435.5069	**2532.098**
	3	-	2627.0498	2493.1884	2574.7041	**2697.625**
	4	**3151.98**	2704.9586	2632.9086	2647.6704	2765.924
	5	**3195.72**	2737.8597	2682.0104	2669.4779	2810.506
C6	2	-	-	3968.9561	3970.0579	**3975.624**
	3	-	-	4040.3573	4095.6016	**4113.412**
	4	-	-	4154.3274	4176.697	**4182.099**
	5	-	-	4136.5779	4178.6363	**4217.993**
Images	m	**Mean CPU Time (seconds) value**				
		GWO [14]	**KHO [15]**	**WOA [18]**	**MFO [18]**	**Proposed**
C1	2	**0.032**	2.2392	3.8	3.74	1.130
	3	**0.0484**	2.2857	4.82	4.48	1.110
	4	**0.075**	2.2943	5.4	5.32	1.000
	5	**0.107**	2.3269	6	5.95	0.839
C2	2	**0.035**	2.2584	-	-	0.763
	3	**0.0516**	2.3007	-	-	0.770
	4	**0.0773**	2.3306	-	-	0.767
	5	**0.1141**	2.3314	-	-	1.032
C3	2	**0.0306**	2.2639	-	-	1.658
	3	**0.0484**	2.3022	-	-	0.715
	4	**0.0766**	2.3082	-	-	0.753
	5	**0.1094**	2.3106	-	-	0.734
C4	2	**0.313**	2.2837	3.56	3.74	0.603
	3	**0.484**	2.2894	3.84	4.6	0.621
	4	0.773	2.3263	4.11	5.25	**0.628**
	5	**0.1148**	2.3305	4.25	6.09	0.629
C5	2	**0.0328**	0.9185	3.98	4.43	0.597
	3	**0.0523**	0.9296	3.17	4.51	0.622
	4	**0.0781**	0.9317	4.36	5.26	0.621
	5	**0.1102**	0.9669	4.2	5.94	0.644
C6	2	-	-	2.23	3.71	**0.595**
	3	-	-	3.81	4.43	1.793
	4	-	-	4.55	5.23	2.037
	5	-	-	5.28	5.97	**0.765**

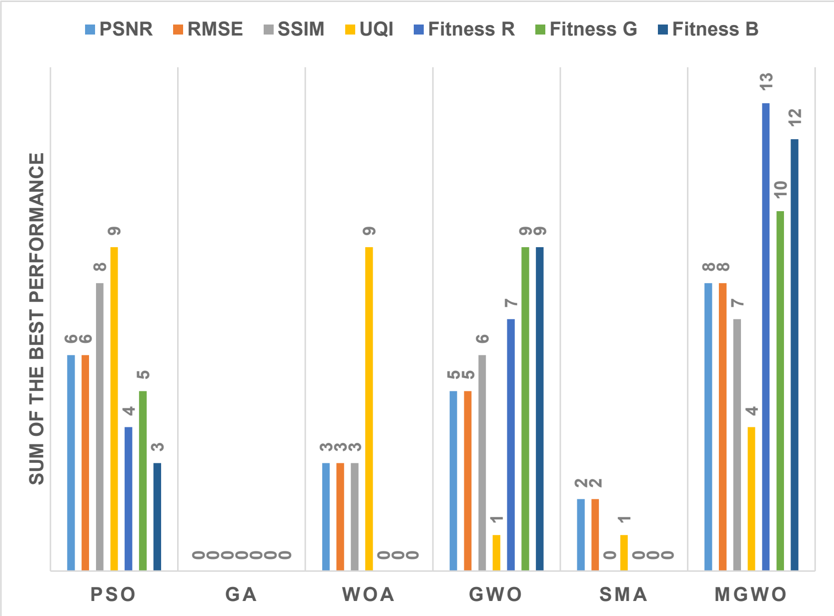
ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

Table 15. Comparison of the performance of the Kapur Entropy-based mGWO proposed methods with state-of-the-art KHO [15] and GWO [14] to solve ML-ISP on grayscale test images.

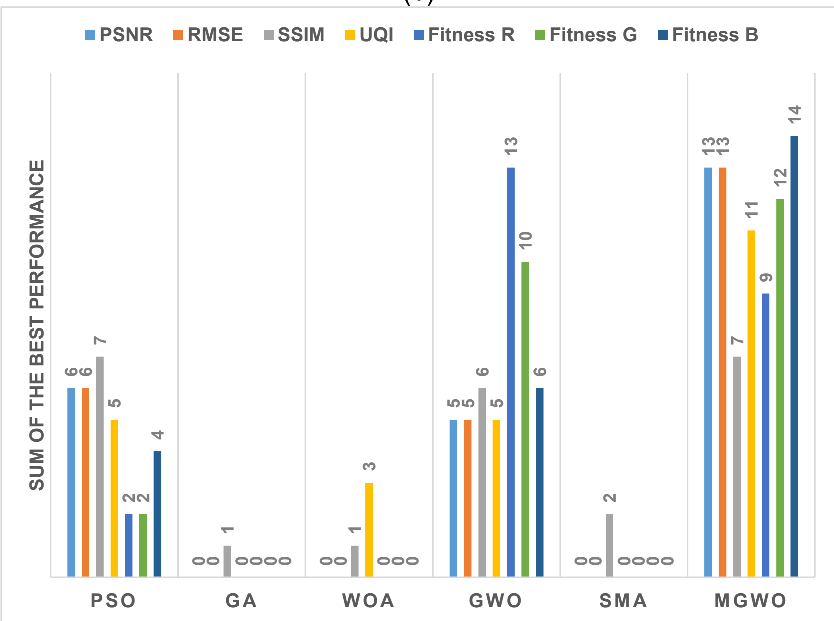
Image	m	Mean Fitness value			Image	m	Mean CPU Time (s)		
		KHO [15]	GWO [14]	Proposed			KHO [15]	GWO [14]	Proposed
C1	2	12.182	-	**12.211**	C1	2	2.263	**0.0353**	2.426
	3	10.463	-	**15.504**		3	2.2724	**0.0617**	2.213
	4	18.182	18.311	**18.311**		4	2.2981	**0.0984**	1.817
	5	20.773	20.903	**20.907**		5	2.3231	**0.1516**	1.674
C2	2	**12.370**	-	12.346	C2	2	2.2645	**0.0359**	1.689
	3	15.261	-	**15.318**		3	2.277	**0.0615**	2.316
	4	**18.043**	18.001	18.005		4	2.2964	**0.1008**	5.102
	5	20.161	20.607	**20.609**		5	2.3218	**0.1469**	1.832
C3	2	12.377	-	**12.635**	C3	2	2.2726	**0.0375**	1.580
	3	15.555	-	**15.810**		3	2.2872	**0.0656**	1.613
	4	18.467	18.674	**18.685**		4	2.3212	**0.1031**	1.584
	5	20.973	21.439	**21.546**		5	2.3291	**0.1531**	1.615
C4	2	**12.238**	-	12.218	C4	2	2.2572	**0.0375**	1.354
	3	15.238	-	**15.279**		3	2.2746	**0.0656**	1.391
	4	18.047	**18.128**	18.124		4	2.986	**0.1031**	1.382
	5	20.643	20.785	**20.788**		5	2.3117	**0.1531**	1.387
C5	2	12.351	-	**12.635**	C5	2	0.9694	**0.0367**	4.625
	3	15.551	-	**15.689**		3	0.9778	**0.0625**	2.953
	4	18.282	**18.728**	18.529		4	1.0022	**0.0984**	0.975
	5	20.926	**21.388**	21.279		5	1.0047	**0.1484**	0.825
C6	2	-	-	**12.525**	C6	2	-	-	**0.834**
	3	-	-	**15.566**		3	-	-	**0.878**
	4	-	-	**18.368**		4	-	-	**1.207**
	5	-	**20.994**	20.987		5	-	-	**1.043**

(a)

Figure 6. Sum of the best performance (out of a total of 24 experiments) of each method on each metric to measure performance stability when tested using different objective functions named (a) Otsu Method, (b) Kapur Entropy, and (c) M.Masi Entropy on RGB image



(b)



(c)

Figure 6. (Continued)