ARTICLE IN PRESS

Journal of King Saud University – Computer and Information Sciences xxx (xxxx) xxx

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Normalized square difference based multilevel thresholding technique for multispectral images using leader slime mould algorithm

Manoj Kumar Naik [a], Rutuparna Panda [b,*], Ajith Abraham [c]

[a] Faculty of Engineering and Technology, Siksha O Anusandhan, Bhubaneswar, Odisha 751030, India
[b] Dept. of Electronics and Telecommunication Engineering, Veer Surendra Sai University of Technology, Burla, Odisha 768018, India
[c] Machine Intelligence Research Labs, Scientific Network for Innovation and Research Excellence, WA 98071-2259, USA

ARTICLE INFO

Article history:
Received 23 June 2020
Revised 18 October 2020
Accepted 30 October 2020
Available online xxxx

Keywords:
Optimal multilevel thresholding
Multispectral image thresholding
Nature-inspired algorithm

ABSTRACT

The existing methodologies used for multilevel thresholding are not efficient in terms of both accuracy and computation time. Two-dimensional histogram-based techniques are better in terms of accuracy while they are computation intensive. The slime mould algorithm used for optimization mainly depends on the best leader and two randomly pooled slime moulds from the population, which leads to poor exploitation with more iteration to converge. These problems are solved in this paper by introducing a novel normalized square difference (NSD) based multilevel thresholding technique using a leader slime mould algorithm (LSMA). The contributions are many fold – i) a NSD based multilevel thresholding method is proposed using the gray level and normalized square difference (GLNSD) 2-D histogram with reduced computation; ii) LSMA is suggested; iii) 23 classical and 6 modern composition test functions from the IEEE CEC 2014 test suite are considered for evaluation of LSMA; iv) experiments on multispectral images are presented. The benefits are – i) reduces computations, ii) improves accuracy. The qualitative metrics used for analysis include – search history, trajectory, and average fitness history. Scalability analysis and statistical analysis (using Friedman's mean rank test) are presented. The proposal is compared with state-of-the-art techniques and found better.

1. Introduction

Thresholding is the popular methodology of image segmentation. Based on the color information of the image, the thresholding technique is classified as – gray level thresholding and color image thresholding. As the gray level images are a compressed version of color images, the former contains less information. The color images contain extra information, hue, and saturation (Haindl and Mikeš, 2016). This emphasizes the researcher to bend towards the color image thresholding nowadays. The color image thresholding has played a vital role in geographic information system (GIS), earth science research, and astronomy, where needs are to locate objects and boundaries in high dimensional multispectral satellite images. Therefore, the multispectral satellite image segmentation problem is crucial and interesting (Bhandari, 2015; Bhandari et al., 2014; Jia et al., 2019a).

The image thresholding is broadly classified as bi-level and multilevel based on the number of thresholds used to divide the image, to get a region of interest. The bi-level thresholding uses only one threshold value to divide the image into two classes – foreground and background class. However, the multilevel thresholding is an extension of bi-level thresholding, where more than one threshold values are used to divide the image into multiclass. The histogram-based approach of an image using statistical perception (entropy) is simple and practical to implement (Sezgin and Sankur, 2004). The first of the histogram-based thresholding approach is Otsu's method (Bhandari et al., 2016; Otsu, 1979), the threshold value is obtained by maximizing the between-class variance. The statistical perception called entropy is often used for the image thresholding. The popular entropy-based thresholding techniques based on 1-D histogram found in the literature are Kapur's entropy (Bhandari et al., 2016; Kapur et al., 1985), minimum cross-entropy (Li and Lee, 1993; Xu et al., 2019), Tsallis entropy (Agrawal et al., 2013; Portes de Albuquerque et al.,

* Corresponding author.
E-mail addresses: naik.manoj.kumar@gmail.com (M.K. Naik), r_ppanda@yahoo.co.in (R. Panda), ajith.abraham@ieee.org (A. Abraham).
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2020.10.030
1319-1578/

Please cite this article as: Manoj Kumar Naik, R. Panda and A. Abraham, Normalized square difference based multilevel thresholding technique for multispectral images using leader slime mould algorithm, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2020.10.030

2004), Renyi entropy (Sarkar et al., 2013) and Masi entropy (Jia et al., 2019b; Nie et al., 2017). The multilevel thresholding over the bi-level thresholding is recommended for better performance despite an increase in the computational complexity, because k threshold levels require a simple exhaustive search of $O\left(L^k\right)$ computation (Yin, 1999) for the gray scale image while $O\left(3L^k\right)$ for the multispectral image having 3 planes. Can thresholding methods based on 1-D histograms justify the thresholded image with the same threshold values, if two images of the similar histogram with different spatial locations are considered? This is the major drawback in 1-D histogram based thresholding approaches.

To overcome the drawback, a 2-D histogram is constructed to provide the spatial correlation information along with the gray level information. The 2-D histogram is formed by taking a correlation among the original gray level values and the local average gray level around the neighborhood. Some earlier thresholding on a 2-D histogram-based approaches are reported in (Abutaleb, 1989; Brink, 1992), at a cost of the computational complexity, the 2-D histogram-based approaches showed superior performance. Further, researchers proposed 2-D Otsu's method (Liu et al., 1991), 2-D Renyi's entropy (Sahoo and Arora, 2004), 2-D Tsaliis-Havrda-Charvát entropy (Sahoo and Arora, 2006), 2D Tsallis entropy (Sarkar and Das, 2013) and 2-D practical Masi entropy (Wunnava et al., 2020) based multilevel thresholding methods. The 2-D entropy based thresholding methods show superior performance as compared to 1-D entropy thresholding. However, the computational complexity for a k-level thresholded image, the exhaustive search increases from $O\left(L^k\right)$ to $O\left(L^{2k}\right)$ for the gray scale image while $O\left(3L^k\right)$ to $O\left(3L^{2k}\right)$ for the color image.

There are two major disadvantages arising in 2-D entropy-based multilevel thresholding, although it shows superior performance to 1-D entropy-based bi-level thresholding. The first disadvantage is that the computational cost exponentially increases with threshold levels. To overcome this, many researchers suggested recursive procedures (Liao et al., 2001; Yin and Chen, 1997) based on lookup tables. However, when the threshold level k increases, computation cost also increases. The second demerit is the exhaustive search process. This can be resolved by a good optimizer, more specifically, nature inspired optimization algorithm. It inherits the natural phenomenon to solve the problem. In this context, some recent advancements on reducing the computational complexity of multilevel thresholding using nature inspired algorithms are – krill herd optimization (Baby Resma and Nair, 2018), crow search algorithm (CSA) (Upadhyay and Chhabra, 2019), gray wolf optimization (GWO) (Khairuzzaman and Chaudhury, 2017), whale optimization algorithm (WOA) (El Aziz et al., 2018) and Harris hawks optimization (HHO) (Bao et al., 2019).

The motivation of the paper is primarily to propose a 2-D multilevel thresholding technique, to conserve the contextual information with reduced computational complexity. We introduce a normalized square difference (NSD) based multilevel thresholding method using the gray level & normalized square difference (GLNSD) 2-D histogram. The NSD based (k-level) multilevel thresholding requires only $O\left(L^{k+1}\right)$ computations for gray-level thresholding and $O\left(3L^{k+1}\right)$ for color image thresholding. This reduces the computational complexity by $(k-1)$ times, which is a significant achievement. The second prime motivation is to use an efficient optimizer to obtain the optimal threshold values for multilevel thresholding. Recently, a slime mould algorithm (SMA) (Li et al., 2020) is proposed using the mathematical modeling of attacking behavior and morphological divergences of the slime mould *Physarum polycephalum* for foraging. It shows quite impressive results on function optimization and engineering design problems. This inspired us to investigate the searching pattern of slime mould, the updating of search agents in SMA depends on the best candidate solution together with another two random candidates solutions. This may lead to an unexpected search space. Hence, we propose a leader slime mould algorithm (LSMA) by incorporating the three best-so-far candidates as a leader to guide the search process. The results of LSMA is compared with state-of-the-art optimizers – slime mould algorithm (SMA) (Li et al., 2020), equilibrium optimizer (EO) (Faramarzi et al., 2020), Harris hawks optimization (HHO) (Heidari et al., 2019), whale optimization algorithm (WOA) (Mirjalili and Lewis, 2016) and gray wolf optimizer GWO (Mirjalili et al., 2014) using 23 classical benchmark test functions from (Naik and Panda, 2016) and 6 composition functions from modern IEEE CEC2014 test suite (Liang et al., 2013). Further, the LSMA is utilized in NSD based multilevel thresholding for multispectral images, to obtain the optimal threshold values. The NSD based multilevel thresholding on high dimensional multispectral satellite images from *Landsat image gallery* ("Landsat Image Gallery", n.d.) using the LSMA is also compared with the SMA, EO, HHO, WOA, and GWO.

The main goals of this work are as follows:

I. A normalized square difference (NSD) based multilevel thresholding method is proposed using gray level and normalized square difference (GLNSD) 2-D histogram.
II. A leader slime mould algorithm (LSMA) is suggested by incorporating the leader (best-so-far candidates) in guiding the search process of the SMA. The LSMA evolves as 1st rank on Friedman's mean rank test when compared with recently developed nature inspire algorithms SMA, EO, HHO, WOA, and GWO. The proposed algorithm is evaluated by taking 23 classical test functions and 6 composition functions from IEEE CEC2014 test suite.
III. The proposed NSD-LSMA based multilevel thresholding technique is evaluated using some high dimensional multispectral images from the *Landsat image gallery*. It reveals that the LSMA based thresholding is evolved as a better method, when compared with SMA, EO, HHO, WOA, and GWO based thresholding.

The rest of this article is as follows. The proposed normalized square difference (NSD) based multilevel thresholding using gray level & normalized square difference (GLNSD) 2-D histogram is presented in Section 2. A leader slime mould algorithm (LSMA) is proposed, evaluated, and ranked with respect to the function optimization, in Section 3. A proposal of NSD-LSMA based multilevel thresholding method for multispectral images is presented in Section 4. The experimental results and discussions of the suggested multilevel thresholding are presented in Section 5. Finally, the conclusion is presented in Section 6.

2. Proposed objective function

In this section, one of the main contributions of the work is provided. A new objective function is introduced.

Let us assume an image I of dimension $M \times N$ with L gray level in the range $[0, L-1]$, which has the pixel intensity value at the coordinate (x, y) as $I(x, y)|x \in \{1, 2, \cdots, M\}, y \in \{1, 2, \cdots, N\}$. The color (RGB) image is defined as:

$$[I(x, y)] = [I_R(x, y), I_G(x, y), I_B(x, y)] \tag{1}$$

where $I_R(x, y), I_G(x, y), I_B(x, y)$ are the red, green, and blue components whose blend generates the color image. Let $I_c(x, y)$ represents a dummy component for red or green or blue color. Let $h(x, y)$ be the local average gray level values of $I_c(x,y)$ at coordinate (x,y) for a $w \times w$ neighborhood and is calculated as (Sahoo and Arora, 2004):

$$h(x,y) = \lfloor \frac{1}{w \times w} \sum_{m=-g}^{g} \sum_{n=-g}^{g} I_c(x+m, y+n) \rfloor \tag{2}$$

where g = [W/2] and $\lfloor \cdot \rfloor$ signifies the integer part of "·". The w is an odd number $w < \min(m,n)$.

We propose the normalized square difference (of $I_c(x,y)$ and $h(x,y)$) based 2-D histogram for multilevel thresholding technique. The square difference ($SD(x,y)$) is evaluated.

$$SD(x,y) = (I_c(x,y) - h(x,y))^2 \tag{3}$$

The normalized square difference ($NSD(x,y)$) is computed as:

$$NSD(x,y) = \lfloor \frac{(SD(x,y) - SD_{min}) \times G}{SD_{max} - SD_{min}} \rfloor \tag{4}$$

where $G(= \max I_c(x,y) | x \in \{1,2,\cdots,M\}, y \in \{1,2,\cdots,N\})$ is the maximum gray level value in I_c, the minimum and maximum value of $SD(x,y)$ are SD_{min} and SD_{max} , respectively and the $\lfloor \cdot \rfloor$ signifies the integer part of "·".

The gray level & normalized square difference (GLNSD) 2-D histogram is constructed as

$$p_{ij} = \frac{1}{MN} \{ q_{ij} | I(x,y) = i, NSD(x,y) = j; i,j \in (0, L-1) \} \tag{5}$$

where the q_{ij} is the occurrence times of pair (i,j) . Matrix representation of the GLNSD 2-D histogram plane is shown in Fig. 1. Fig. 1(a) shows a 2-D histogram plane for the threshold pair (S,T). It divides the plane into four quadrants {Q_1, Q_2, Q_3, Q_4}, where S is the threshold value from $NSD(x,y)$ value and T is the threshold value from intensity value $I_c(x,y)$. The Q_1 and Q_2 consist of most of the information needed for the thresholding compared to Q_3 and Q_4 because they have edge or noise information. So, first row quadrants Q_1 and Q_2 are meaningful for the thresholding application; classified as the foreground class C_f and the background class C_b.

The foreground and background class probabilities are given as:

$$P_f = \sum_{i=0}^{S-1} \sum_{j=0}^{T-1} p_{ij} \tag{6}$$

and

$$P_b = \sum_{i=0}^{S-1} \sum_{j=T}^{L-1} p_{ij} \tag{7}$$

where $S,T \in \{0,1,\cdots,L-1\}$. As the contribution quadrants Q_3 and Q_4 are insignificant, then $P_f \approx 1 - P_b$ (Sahoo and Arora, 2006).

The entropy dependent of a threshold value (S,T) are estimates for C_f using Eq. (8) and C_b using Eq. (9).

$$E_f = -\sum_{i=0}^{S-1} \sum_{j=0}^{T-1} \left(\frac{p_{ij}}{P_f}\right) ln \left(\frac{p_{ij}}{P_f}\right) \tag{8}$$

$$E_b = -\sum_{i=0}^{S-1} \sum_{j=T}^{L-1} \left(\frac{p_{ij}}{P_b}\right) ln \left(\frac{p_{ij}}{P_b}\right) \tag{9}$$

Then the optimal threshold value for bi-level thresholding is obtained by:

$$(S_{opt}, T_{opt}) = \arg\max \{E_f + E_b\} \tag{10}$$

subject to the conditions $0 < S < L-1$ and $0 < T < L-1$.

Let us consider the multilevel thresholding which divides the image into a set of K different classes such as $\{C_1, C_2, \cdots, C_K\}$ using $k(=K-1)$ threshold values $\{(S,T_1),(S,T_2),\cdots,(S,T_k)\}$, which conveyed more meaningful information than bi-level thresholding. The 2-D histogram plane for multilevel thresholding is shown in Fig. 1(b), in which only the first row K quadrants Q_1-Q_K are considered for the thresholding problem because they hold almost all information.

The classes $\{C_1, C_2, \cdots, C_K\}$ probability is evaluated as:

$$\begin{aligned} P_1 &= \sum_{i=0}^{S-1} \sum_{j=0}^{T_1-1} p_{ij} \\ P_2 &= \sum_{i=0}^{S-1} \sum_{j=T_1}^{T_2-1} p_{ij} \\ &\vdots \\ P_K &= \sum_{i=0}^{S-1} \sum_{j=T_k}^{L-1} p_{ij} \end{aligned} \tag{11}$$

where $0 < S < L-1$ and $0 < T_1 < T_2 < \cdots < T_k < L-1$. The second row K quadrants consist of the information related to the noise, and hence, are negligible. So, the summation of the probability of classes $\{C_1, C_2, \cdots, C_K\}$ related to the first row K quadrants Q_1-Q_K are approximated as:

$$\sum_{i=1}^{K} P_i \approx 1 \tag{12}$$

The entropy dependent on the threshold value $\{(S,T_1),(S,T_2),\cdots,(S,T_k)\}$ for various classes $\{C_1, C_2, \cdots, C_K\}$ are estimated as:

$$\begin{aligned} E_1 &= -\sum_{i=0}^{S-1} \sum_{j=0}^{T_1-1} \left(\frac{p_{ij}}{P_1}\right) ln \left(\frac{p_{ij}}{P_1}\right) \\ E_2 &= -\sum_{i=0}^{S-1} \sum_{j=T_1}^{T_2-1} \left(\frac{p_{ij}}{P_2}\right) ln \left(\frac{p_{ij}}{P_2}\right) \\ &\vdots \\ E_K &= -\sum_{i=0}^{S-1} \sum_{j=T_k}^{L-1} \left(\frac{p_{ij}}{P_K}\right) ln \left(\frac{p_{ij}}{P_K}\right) \end{aligned} \tag{13}$$

Then the optimal threshold value for the multilevel thresholding is obtained as:

$$\begin{aligned} &\{(S_{opt}, T_{1_{opt}}), (S_{opt}, T_{2_{opt}}), \cdots, (S_{opt}, T_{k_{opt}})\} \\ &= \arg\max \{E_1 + E_2 + \cdots + E_K\} \end{aligned} \tag{14}$$

subject to the conditions $0 < S < L-1$ and $0 < T_1 < T_2 < \cdots < T_k < L-1$.

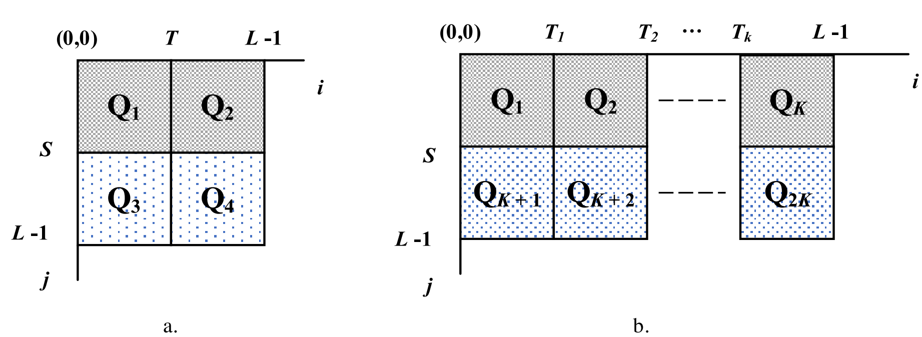

Fig. 1. GLNSD 2-D histogram plane. (a) Bi-level thresholding (b) Multilevel thresholding.

The Eq. (14) is a maximization problem and served to get the optimal threshold values by an exhaustive search. One can find remarkable differences here. Fig. 1(b) shows its worthiness over earlier approaches discussed in the introduction section. The entropic information is obtained by computing row-wise quadrants Q_1, Q_2, ⋯, Q_K, only. Interestingly, $O\left(L^{k+1}\right)$ computations are needed for multilevel thresholding of gray images while $O\left(3L^{k+1}\right)$ for color images. Whereas, for the existing methodologies, the number of computations required is $O\left(L^{2k}\right)$ and $O\left(3L^{2k}\right)$, respectively. Thus, computations are reduced by a factor of (k-1) in our case. Therefore, the significance of the proposal is explicit and it may enrich the image processing literature.

3. The proposed leader slime mould algorithm (LSMA)

At this moment, we need a good optimizer to obtain the optimal threshold values by utilizing the Eq. (14) as an objective function. In this context, here we introduce an efficient optimizer. The development of our leader slime mould algorithm (LSMA) is based on the modeling of approaching behavior of the slime mould algorithm (SMA) (Li et al., 2020) with leaders of the slime mould concentration. The SMA simulates the attacking behavior and morphological divergences of the slime mould *Physarum polycephalum* for foraging, which mainly depends on the best leader together with two slime mould randomly pooled from the overall populations.

Table 1
Parameter setting for various algorithms.

Algorithm	Parameters
LSMA / SMA	$N = 20$ and $z = 0.03$
EO	$N = 20$, $a_1 = 2$, $a_2 = 1$ and $GP = 0.5$
HHO	$N = 20$ and $\beta = 1.5$
WOA	$N = 20$, $a = [0, 2]$ and $b = 1$, $l = [-1, 1]$
GWO	$N = 20$and $a = [20]$

The updating rule of the SMA concentration for the ith slime mould $X_i(= \{x_i^1, x_i^2, \cdots, x_i^k\})$ for a k dimensional problem from N slime mould is modeled in (Li et al., 2020) as:

$$X_i(t+1) = \begin{cases} r_1 \cdot (UB - LB) + LB & r_1 < z \\ X_{GlobalBest}(t) + V_a \cdot (W \cdot X_{R1} - X_{R2}) & r_1 \geq z \text{ and } r_2 < p \\ V_b \cdot X_i(t) & r_1 \geq z \text{ and } r_2 \geq p \end{cases} \tag{15}$$

and

$$X_i(1) = r_1 \cdot (UB - LB) + LB \tag{16}$$

The r_1 and r_2 are random values in $[0, 1]$, t is the current iteration, UB is the upper boundary of the search space, LB is the lower boundary of the search space,V_a is the uniformly distributed velocity in a range of $[-a, a]$, $V_b$is the linearly decreasing velocity from 1 to 0, W is the weight of slime mould, p is the probability to determine the trajectory of the slime mould, $X_{GlobalBest}$ is the global best concentration current iteration t, X_{R1} and $X_{R2}$are the two slime mould randomly pooled from the N population, z is the elimination-and-dispersal rate of the slime mould which is fixed at 0.03 and $i \in 1, 2, \cdots, N$.

The p of the ith slime mould depends on its current fitness $f(X_i)$ and fitness of the global best concentration $f(X_{L1})$, which is formulated as:

$$p = \tanh |f(X_i) - f(X_{L1})| \tag{17}$$

The velocity V_a is uniformly distributed in the range $[-a, a]$ and V_b is uniformly distributed in the range $[-b, b]$. The a and b are determined as:

$$a = \operatorname{arctanh}\left(-\left(\frac{t}{t_{max}}\right) + 1\right) \tag{18}$$

and

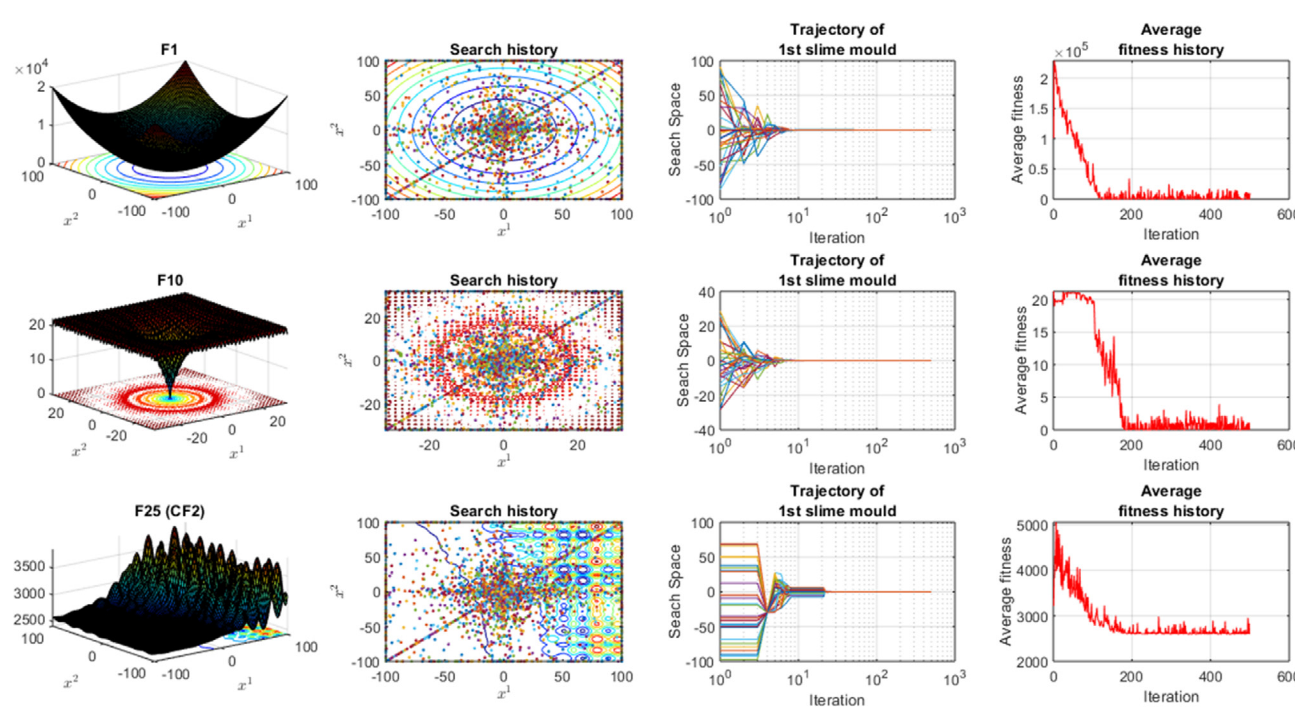

Fig. 2. Qualitative analysis for functions F1, F10, and F25(CF2).

$$b = 1 - \frac{t}{t_{max}} \tag{19}$$

where t_{max} represent the maximum iteration.

The W is determined from the local fitness value of slime mould. Let us sort the fitness value of the N slime mould in iteration t in ascending order for the minimization problem (or descending order for the maximization problem).

$$[sorted\_fitness, sort\_Index] = sort(f) \tag{20}$$

where $f = (f(X_1), f(X_2), \cdots, f(X_N))$

Then the W is determined as follows:

$$W(sort\_Index(l)) = \begin{cases} 1 + r_3 \cdot \log\left(\frac{f_{LocalBest} - sortf(l)}{f_{LocalBest} - f_{LocalWorst}} + 1\right) & 1 \le l \le \frac{N}{2} \\ 1 - r_3 \cdot \log\left(\frac{f_{LocalBest} - sortf(l)}{f_{LocalBest} - f_{LocalWorst}} + 1\right) & \frac{N}{2} < l \le N \end{cases} \tag{21}$$

$$f_{LocalBest} = sorted\_fitness(1) \tag{22}$$

$$f_{LocalWorst} = sorted\_fitness(N) \tag{23}$$

Table 2
Statistical results for test functions (Best value indicated by bold face).

	Function	Metric	LSMA	SMA	EO	HHO	GWO	WOA
Unimodal (scalable dimension)	F1	Ave	**0**	3.68E−244	6.07E−35	3.38E−92	1.61E−08	9.82E−17
		Std	**0**	**0**	1.20E−34	1.22E−91	3.00E−08	1.32E−16
	F2	Ave	**8.75E−247**	3.52E−138	1.06E−20	1.56E−47	2.85E−05	3.64E−10
		Std	**0**	1.96E−137	1.04E−20	6.50E−47	1.78E−05	3.00E−10
	F3	Ave	**0**	1.09E−261	7.45E−07	6.89E−69	3.85E−01	6.14E−04
		Std	**0**	**0**	2.91E−06	2.65E−68	4.38E−01	1.00E−03
	F4	Ave	**3.08E−267**	8.74E−135	2.85E−08	1.30E−45	1.42E−01	3.35E−04
		Std	**0**	3.81E−134	7.03E−08	5.36E−45	1.20E−01	3.42E−04
	F5	Ave	9.39E + 00	1.40E + 01	2.59E + 01	**3.04E−02**	2.87E + 01	2.71E + 01
		Std	1.28E + 01	1.32E + 01	1.55E−01	**3.67E−02**	2.93E−01	8.12E−01
	F6	Ave	3.70E−03	8.74E−03	**3.65E−04**	4.36E−04	3.42E + 00	1.06E + 00
		Std	1.69E−03	7.43E−03	1.20E−03	**8.17E−04**	5.09E−01	4.25E−01
	F7	Ave	**1.44E−04**	3.07E−04	1.59E−03	1.99E−04	6.35E−03	1.11E−02
		Std	**1.44E−04**	3.46E−04	7.83E−04	1.64E−04	4.07E−03	6.91E−03
Multimodal (scalable dimension)	F8	Ave	**−12569.31**	−12568.70	−8708.23	−12503.96	−38.42	−7669.29
		Std	**1.29E−01**	6.23E−01	5.92E + 02	3.58E + 02	5.93E + 00	4.93E + 02
	F9	Ave	**0**	**0**	**0**	**0**	2.64E + 01	2.56E + 01
		Std	**0**	**0**	**0**	**0**	2.75E + 01	1.66E + 01
	F10	Ave	**8.88E−16**	**8.88E−16**	1.33E−14	**8.88E−16**	7.50E−05	4.14E−09
		Std	**0**	**0**	3.02E−15	**0**	1.70E−04	3.61E−09
	F11	Ave	**0**	**0**	2.39E−04	**0**	1.75E−09	2.18E−16
		Std	**0**	**0**	1.33E−03	**0**	3.69E−09	2.90E−16
	F12	Ave	3.64E−03	1.03E−02	6.70E−03	**1.54E−05**	3.36E−01	4.72E−02
		Std	6.43E−03	1.13E−02	2.59E−02	**1.65E−05**	9.22E−02	1.95E−02
	F13	Ave	6.98E−02	1.24E−02	1.07E−01	**1.53E−04**	2.20E + 00	9.65E−01
		Std	1.32E−01	1.94E−02	1.05E−01	**3.21E−04**	2.79E−01	2.62E−01
Multimodal (fixed dimension)	F14	Ave	**0.9980**	**0.9980**	1.0620	1.5726	12.6705	1.1260
		Std	1.43E−11	5.65E−12	3.56E−01	1.24E + 00	**1.58E−13**	4.95E−01
	F15	Ave	6.19E−04	5.77E−04	2.38E−03	**3.42E−04**	7.11E−04	5.10E−04
		Std	1.76E−04	2.58E−04	5.99E−03	**3.82E−05**	4.49E−04	3.39E−04
	F16	Ave	**−1.0316**	**−1.0316**	**−1.0316**	**−1.0316**	−1.0265	**−1.0316**
		Std	5.54E−08	1.78E−09	**5.65E−16**	5.72E−09	1.13E−02	3.08E−10
	F17	Ave	**0.3979**	**0.3979**	**0.3979**	**0.3979**	0.7117	0.5476
		Std	1.04E−06	**1.43E−07**	0.00E + 00	8.07E−05	1.16E + 00	8.34E−01
	F18	Ave	**3.0000**	**3.0000**	**3.0000**	**3.0000**	4.7600	**3.0000**
		Std	1.41E−06	3.12E−10	**1.85E−15**	2.37E−06	6.74E + 00	3.04E−08
	F19	Ave	**−3.8628**	**−3.8628**	−3.8625	−3.8586	−3.5582	−3.8569
		Std	**3.65E−05**	**3.05E−06**	1.42E−03	5.66E−03	9.14E−01	3.51E−03
	F20	Ave	−3.2285	**−3.2754**	−3.2695	−3.0408	−2.3015	−2.7239
		Std	**5.88E−02**	5.95E−02	6.38E−02	1.30E−01	9.57E−01	6.34E−01
	F21	Ave	**−10.1529**	−10.1527	−8.2805	−5.4129	−3.9771	−4.7859
		Std	**2.52E−04**	4.02E−04	2.83E + 00	1.48E + 00	1.85E + 00	1.04E + 00
	F22	Ave	**−10.4027**	−10.4025	−8.6220	−5.8416	−4.4123	−4.8181
		Std	**2.02E−04**	3.86E−04	2.88E + 00	1.78E + 00	1.56E + 00	1.04E + 00
	F23	Ave	**−10.5361**	−10.5358	−9.1134	−5.1154	−4.1578	−5.1285
		Std	**2.67E−04**	7.21E−04	2.73E + 00	1.38E−02	1.83E + 00	1.45E−05
Composition	F24 (CF1)	Ave	**2500**	**2500**	2615.87	**2500**	2500.00	**2500**
		Std	**0**	**0**	4.73E−01	**0**	3.01E−03	4.70E−07
	F25 (CF2)	Ave	**2600**	**2600**	2600.03	2600.0003	2601.59	2600.45
		Std	**0**	**0**	1.38E−02	9.58E−04	6.21E−01	2.49E−01
	F26 (CF3)	Ave	**2700**	**2700**	2701.66	**2700**	2700.0001	**2700**
		Std	**0**	**0**	4.38E + 00	**0**	5.58E−05	5.04E−09
	F27 (CF4)	Ave	**2700.70**	2700.82	2742.19	2771.40	2800.0001	2710.42
		Std	1.72E−01	**1.49E−01**	4.99E + 01	4.55E + 01	2.16E−04	2.98E + 01
	F28 (CF5)	Ave	**2900**	**2900**	3358.14	**2900**	2900.00	**2900**
		Std	6.64E−13	1.08E−12	1.03E + 02	**0**	9.08E−04	3.69E−08
	F29 (CF6)	Ave	**3000**	**3000**	3836.14	**3000**	**3000.00**	**3000**
		Std	**0**	1.72E−12	1.86E + 02	**0**	1.30E−03	7.47E−08
Friedman's mean rank			**2.02**	2.21	3.67	2.90	5.76	4.45
Rank			**1**	2	4	3	6	5

The SMA shows promising results on the function optimization and engineering design problems. Still there is a hope of improvement in the search process given in Eq. (15). The updating of the ith slime mould concentration at iteration $(t+1)$ mainly depends on the global best slime mould concentration $(X_{GlobalBest})$ of the current iteration t and a deviation determined by the two random slime mould concentration $(X_{R1}$ and $X_{R2})$ pooled from the N population $(X = (X_1, X_2, \cdots, X_N))$. When the random slime mould is far from

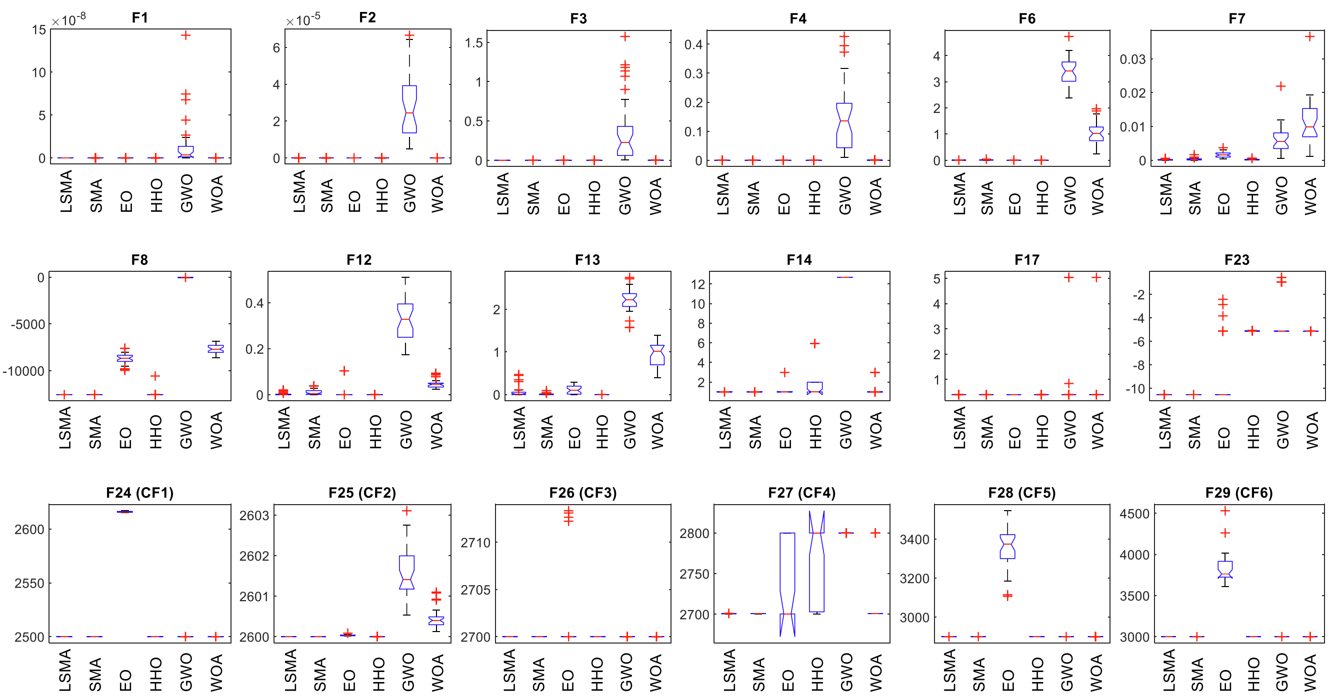

Fig. 3. Boxplot of 18 test functions.

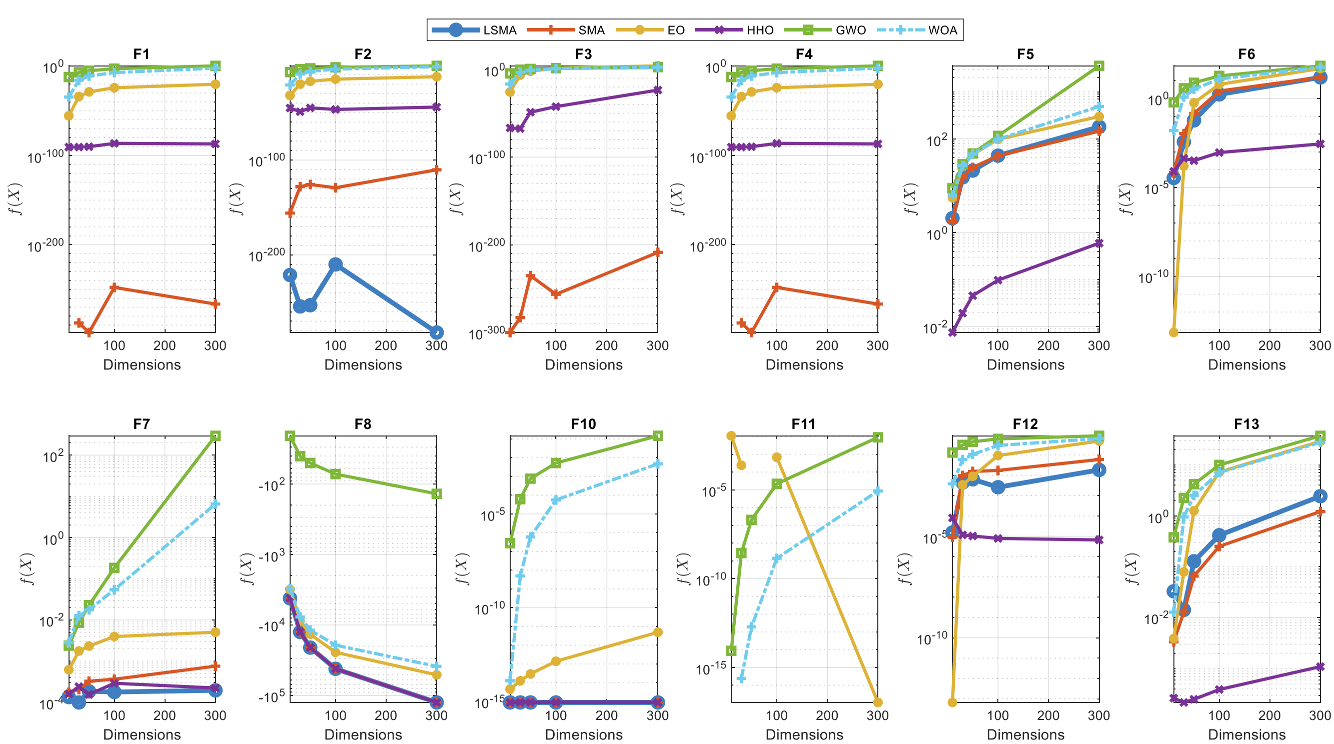

Fig. 4. Scalability results of 12 unimodal and multimodal scalable test functions.

the solution space, the randomization may take longer to reach an optimum solution. As inspired by GWO (Mirjalili et al., 2014), utilize three best-so-far candidates (alpha, beta, and gamma) to update the search positions of the wolves. EO (Faramarzi et al., 2020) utilizes the four best-so-far candidates as equilibrium candidates to reach the equilibrium state. We use three best-so-far candidates as the *leader* to update positions of the slime mould. So, we have coined as the leader slime mould algorithm (LSMA), that associate the thee best-so-far candidates as the *leader* with the random slime mould concentration (X_{R1}andX_{R2}) pooled from the N population. The global best concentration as the *leader1* ($L1$), the second and third best concentrations are named as the *leader2* ($L2$) and *leader3* ($L3$), respectively. Interestingly, this modeling makes the algorithm efficient enough to trade-off between the exploitation and the exploration more efficiently than before, which can achieve the optimal solutions.

The new updating rule of ith slime mould at iteration $(t+1)$ in LSMA is modeled as:

$$X_i(t+1) = r_1 \cdot (UB - LB) + LB, \text{ when } r_1 < z \tag{24.a}$$

$$X_i(t+1) = X_{L1}(t) + V_a \cdot ((W \cdot X_{L2} - X_{R1}) + (W \cdot X_{L3} - X_{R2})), \; r_1 \geq z \text{ and } r_2 < p \tag{24.b}$$

$$X_i(t+1) = V_b \cdot X_i(t), \text{ when } r_1 \geq z \text{ and } r_2 \geq p \tag{24.c}$$

3.1. Pseudocode of LSMA

In the beginning, identify the objective function (f), a dimension of the problem (k), the boundary of the search space (LB, UB), a population size of the slime mould (N)within a search space, maximum iteration (t_{max}) and elimination-and-dispersal rate (z) which is determined experimentally beforehand.

Begin:
Input: N, LB, UB, t_{max} and z.
Initialization: Initialize the N slime mould $X = (X_1, X_2, \cdots, X_N)$ using the Eq. (16) for a k dimensional problem and current iteration t as 1.
While $(t \leq t_{max})$
Evaluate the fitness of N slime mould using the objective function f.
Update the best-so-far candidate *leader's* concentrations X_{L1}, X_{L2} and X_{L3}.
Estimate the a using Eq. (18), b using Eq. (19) and W using Eq. (21).
For (each slime mould X_i)
Evaluate the p using the Eq. (17).
Generate the velocity V_a and V_b.
Randomly chose two slime mould X_{R1} and X_{R2} from the N slime mould present in search space.
Update the concentration of slime mould using Eq. (24).
End For
$t = t + 1$
End While
Output: Global best slime mould concentration X_{L1} and fitness of global best concertation $f(X_{L1})$.

3.2. Performance evaluation of LSMA

The performance evaluation of the proposed LSMA is carried out with the help of a set of 23 classical test functions (Naik and Panda, 2016) and 6 modern composition test functions from the IEEE CEC 2014 test suite (Liang et al., 2013). The test functions are classified into four categories as unimodal (F1-F7), multimodal with scalable dimensions (F8-F13), multimodal with fixed dimensions (F14-F23),

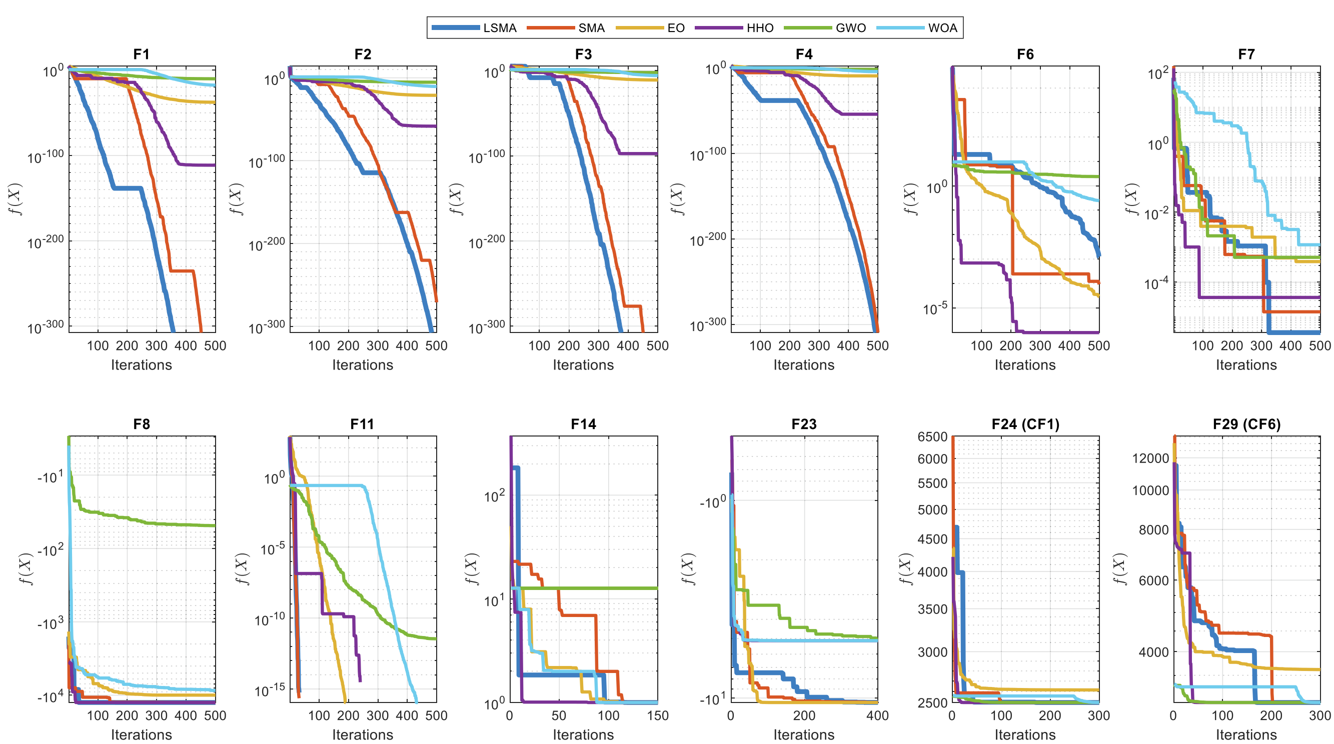

Fig. 5. Convergence curve of some test functions.

and composition test functions (F24-F29). The unimodal test functions are best described for a single optimal solution and help to understand the exploitative behavior of the optimization algorithm. However, the multimodal test functions have more than one optimal solution; therefore it helps to understand the exploration capability of the optimization algorithm. However, to mimic the real search problem, we take the composition test functions, a hybridization of unimodal and multimodal test functions, which are non-separable, multimodal, a variety of shapes in different regions and a large number of different local minima with different properties.

A results comparison of our proposed LSMA with some recently-developed optimizers SMA (Li et al., 2020), EO (Faramarzi et al., 2020), HHO (Heidari et al., 2019), WOA (Mirjalili and Lewis, 2016) and GWO (Mirjalili et al., 2014) algorithms based on an average value of the results '**Ave**' and standard deviation '**Std**' among the 31 independent runs are considered. The parameter settings are same as recommended by the original works of SMA (Li et al., 2020), EO (Faramarzi et al., 2020), HHO (Heidari et al., 2019), WOA (Mirjalili and Lewis, 2016) and GWO (Mirjalili et al., 2014), presented in Table 1. All algorithms are evaluated with the same population ($N = 20$) and the maximum iteration ($t_{max} = 500$) to maintain the uniformity in function evaluations as 10,000.

A qualitative analysis of LSMA for one function each from unimodal, multimodal, and composition test functions are presented in Fig. 2. The qualitative metric includes search history, trajectory, and average fitness history. The search history diagram presents the history of the slime mould positions in the search space as presented in the second column of Fig. 2, a higher concentration of slime mould positions are found near to the optimal solution. The next qualitative metric is the trajectory of the first slime mould for k dimension during the complete life cycle (i.e $t = 1$ to t= t_{max}), presented in the 3rd column of Fig. 2. The trajectory diagram

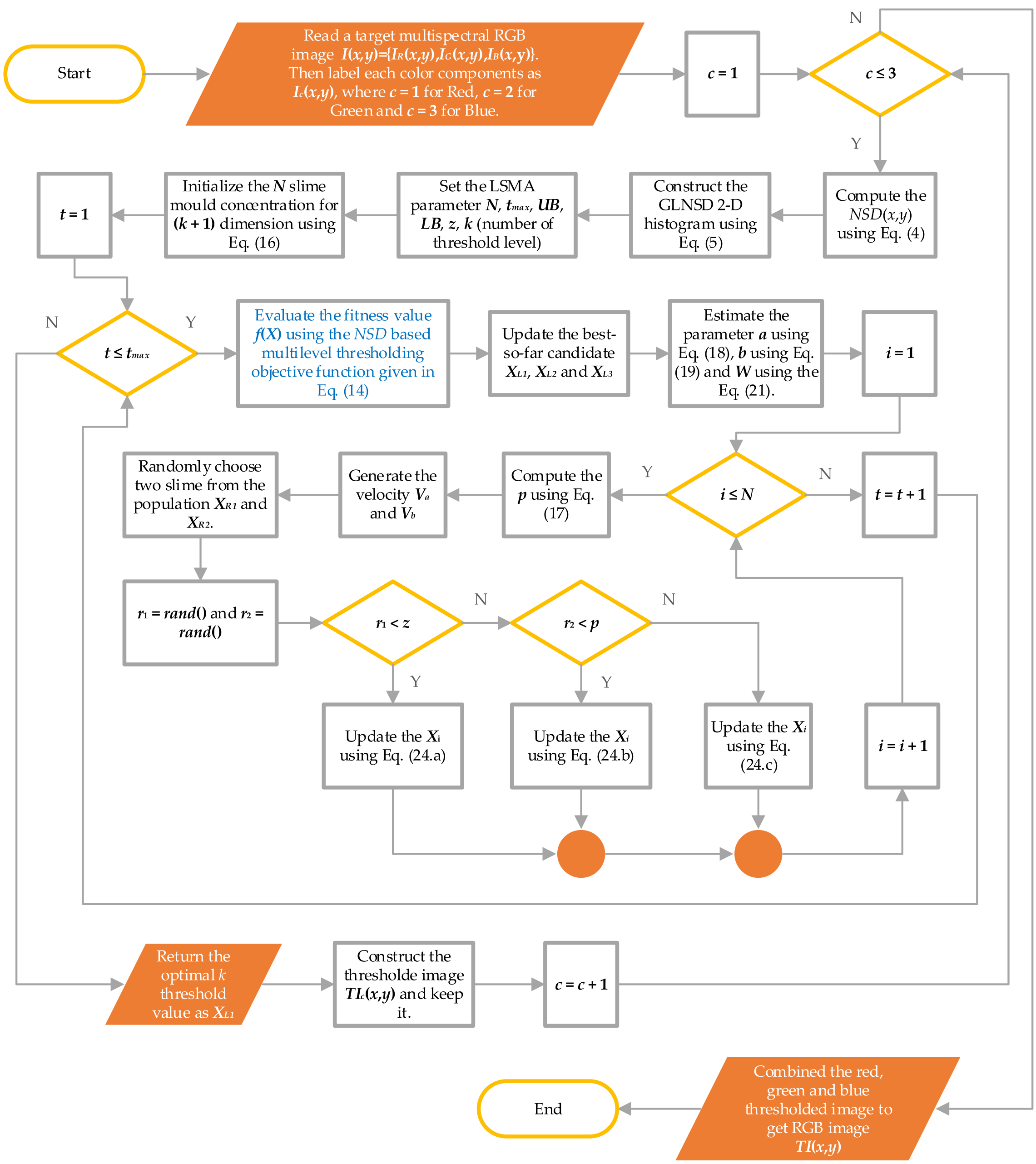

Fig. 6. Flowchart of the NSD-LSMA based multispectral multilevel thresholding.

reveals that during the initial generation slime moulds are widely dispersed in the search space. However, as the life cycle (iteration) goes on, they quickly converge to the optimal solution space. The last metric used for a qualitative analysis is an average fitness history of N slime mould during the complete life cycle, which is presented in the last column of Fig. 2. It reveals that a decreasing trend of the average fitness shows the collaborative effort of the slime mould to reach the optimal solution space.

Statistical results are presented in Table 2. The performance on the first category of unimodal functions (F1-F7) is quite encouraging; LSMA obtains the better optimal solutions in most of the test functions except F5 and F6, where it is just behind the HHO and EO. For the second category multimodal test functions with scalable dimensions (F8-F13), LSMA performs better for F8, performs the same as SMA, and HHO for F9-F11, lagged than HHO in F12 and F13. Mixed response for the third category multimodal test functions with fixed dimensions (F14-F23) is achieved. However, LSMA shows a better consistency to obtain the optimal solution. Finally, in the fourth category composition functions, LSMA performance is commendable and comparable with the SMA and HHO. The statistical analysis based on Friedman's mean rank (Derrac et al., 2011) is performed, considering the average value of all 29 test functions. Based on Friedman's mean rank of statistical results, the LSMA ranked one among other optimization algorithms.

A boxplot is presented in Fig. 3 to understand how the optimal solution is obtained during the 31 independent runs of the various optimization algorithms. LSMA showed better consistency among other optimization algorithms to obtain the optimal value. The impact of dimensions on optimization algorithm performance can be analyzed with a scalability analysis. For this purpose, we have taken varied dimensions as $k = 10, 30, 50, 100, 300$ for the experiments with $N = 20$ and $t_{max} = 500$. The unimodal and multimodal scalable test functions (F1-F13) are considered and average value results are compared in Fig. 4. From Fig. 4, it is evident that LSMA performance is better in the majority of test functions and just lagging behind HHO on test functions F5, F6, F12, and F13. This test reveals that LSMA may be used for high dimensional optimization problems.

The convergence curve of 12 test functions is presented in Fig. 5, to support how quickly the LSMA tracks the optimal solution during its lifetime. The first row of Fig. 5 shows the convergence curve of unimodal test functions, which reveals that the LSMA outperforms in test functions F1-F4 and F7. However, HHO has commendable convergence in F6. The second row of Fig. 5 presents the convergence curve of multimodal and composition test functions, which reveal that the LSMA has a mixed response with SMA, EO, and HHO. The study indicates that the LSMA may be utilized for both low/high dimensional optimization problems. This motivates us to use LSMA as both a low/high dimensional optimizer for the multispectral satellite image thresholding application.

4. The proposed NSD-LSMA based multilevel thresholding technique

In this section, we propose how to use the LSMA to obtain the optimal thresholds using NSD based objective function explained in section 2. For k threshold components on each color component $I_c(x,y) \in \{I_R(x,y), I_G(x,y), I_B(x,y)\}$, each slime mould in the LSMA is a $(k+1)$ dimensional thresholding vector of $X_i(t) = (\sigma_i, \tau_i)$. Note that$i = 1, 2, \cdots, N$, N is the population size, σ_i represents one threshold component from the $NSD(x,y)$, $\tau_i = (\tau_i^1, \tau_i^2, \cdots, \tau_i^k)$ represents the k number of threshold components for $I_c(x,y)$. The LSMA is used to obtain the threshold values of each color component using the *NSD* based multilevel thresholding objective function described in Eq. (14). After thresholding of each color component, we get $(k+1)$ gray levels for each red, green and blue component thresholded image TI_R, TI_G and TI_B, respectively. The red, green, and blue thresholded images are combined to get RGB thresholded image as:

$$[TI(x,y)] = [TI_R(x,y), TI_G(x,y), TI_B(x,y)] \tag{25}$$

Each color component thresholded image (TI_c) has at most ($K = k+1$) number of gray levels. So, RGB thresholded image ($TI(x,y)$) has a maximum number of K^3 gray levels, which is much smaller than the original RGB image. The process of the NSD-LSMA based multilevel thresholding is presented in Fig. 6.

5. Results and discussions

The performance of the NSD-LSMA based multilevel thresholding is demonstrated in this section. The simulation of the

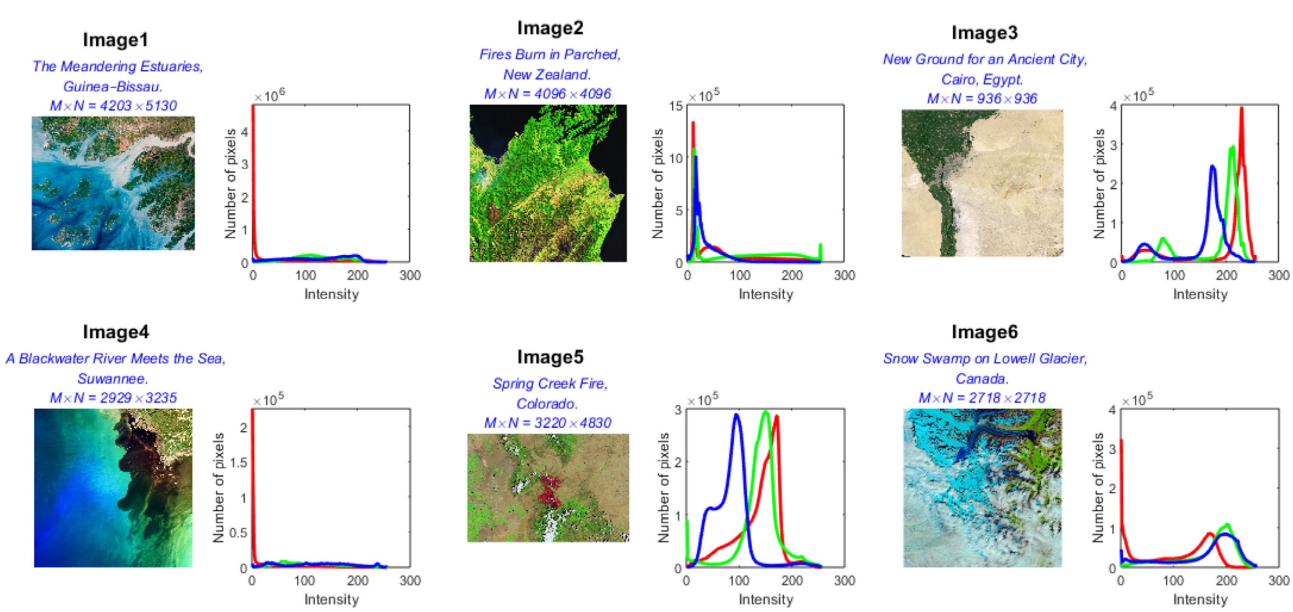

Fig. 7. The satellite test images and their corresponding histograms.

experiment is performed in MATLAB R2018b supported by Intel Core i3-6100U 2.3 GHz processor with 8 GB RAM running on Windows 10 environment. The experiments are carried out with high dimensional multispectral satellite images from the *Landsat image gallery* ("Landsat Image Gallery", n.d.). The satellite image features are condensed, rapidly varying from one zone to another zone. The satellite image is of high dimensions. This warrants us to use an efficient thresholding technique. This instigates us to choose the most recent efficient optimizers such as SMA (Li et al., 2020), EO (Faramarzi et al., 2020), HHO (Heidari et al., 2019), WOA (Mirjalili and Lewis, 2016) and GWO (Mirjalili et al., 2014) for a comparative analysis. All the algorithms SMA, EO, HHO, WOA and GWO are implemented using the proposed objective function.

A set of six selected test images from the *Landsat image gallery* are considered for the experiments, which are displayed in Fig. 7 with the place of imaging, size, and the corresponding histograms of RGB color components. All algorithms are run independently 31 times with a population size of 20 and a maximum iteration of 100 to maintain the consistency for the performance evaluation. The threshold dimensions are taken $(k = 4, 8)$ as a low dimensional, and $(k = 12)$ as a high dimensional experimental study. The parameters for the optimization algorithms are taken same as the simulations performed in Section 3.2 (see Table 1). For a comparative analysis, the well-known performance metrics of segmentation peak signal to noise ratio (PSNR) (Jia et al., 2019b), feature similarity (FSIM) (Zhang et al., 2011) and structural similarity

Table 3
Optimal PSNR, FSIM, and SSIM.

Test Images	K	Metric	LSMA	SMA	EO	HHO	WOA	GWO
Image1	4	PSNR	**27.1253**	27.1235	27.1152	27.1233	27.1209	27.0711
		FSIM	**0.9626**	0.9625	0.9624	0.9621	0.9622	0.9608
		SSIM	**0.9294**	0.9287	0.9293	0.9286	0.9292	0.9251
	8	PSNR	**31.9431**	31.9375	31.7870	31.9136	31.9457	30.9170
		FSIM	0.9880	0.9866	**0.9892**	0.9877	0.9871	0.9839
		SSIM	0.9718	0.9714	0.9688	**0.9721**	0.9714	0.9638
	12	PSNR	**34.8121**	34.4649	34.5641	34.8096	34.5571	33.1779
		FSIM	0.9938	0.9928	0.9930	**0.9961**	0.9950	0.9921
		SSIM	0.9835	0.9825	0.9834	0.9824	**0.9841**	0.9767
Image2	4	PSNR	27.1274	**27.1294**	27.1232	27.1289	27.1286	27.0436
		FSIM	**0.9941**	0.9940	0.9937	0.9940	0.9940	0.9927
		SSIM	**0.9416**	0.9390	0.9399	0.9392	0.9391	0.9433
	8	PSNR	32.2726	32.2607	32.0626	**32.3081**	32.3054	31.8053
		FSIM	**0.9987**	0.9985	0.9983	0.9985	0.9986	0.9981
		SSIM	**0.9783**	0.9775	0.9769	0.9763	0.9771	0.9720
	12	PSNR	35.1047	**35.3238**	34.7567	35.2951	35.2162	34.1196
		FSIM	**0.9993**	0.9989	0.9985	0.9989	0.9987	0.9988
		SSIM	**0.9867**	0.9857	0.9837	0.9838	0.9839	0.9811
Image3	4	PSNR	28.6963	28.6915	28.6707	**28.6971**	28.6966	28.6897
		FSIM	0.9592	0.9587	0.9570	0.9590	**0.9604**	0.9497
		SSIM	0.9260	0.9260	0.9257	0.9260	0.9255	**0.9266**
	8	PSNR	**33.1539**	33.1356	32.9699	33.1004	33.1014	32.1498
		FSIM	0.9884	0.9878	0.9870	0.9863	**0.9893**	0.9856
		SSIM	**0.9647**	0.9651	0.9633	0.9643	0.9646	0.9580
	12	PSNR	**35.9055**	35.9637	35.5257	35.7094	35.8112	34.6154
		FSIM	0.9935	0.9952	0.9937	0.9936	0.9937	**0.9961**
		SSIM	**0.9796**	0.9795	0.9771	0.9784	0.9788	0.9727
Image4	4	PSNR	26.4247	26.4237	26.3996	**26.4249**	26.4236	26.3524
		FSIM	0.9034	0.9032	0.9036	0.9032	0.9028	**0.9042**
		SSIM	0.9329	0.9330	**0.9338**	0.9329	0.9329	0.9316
	8	PSNR	31.2264	**31.2821**	31.1964	31.2418	31.2736	30.2136
		FSIM	0.9547	0.9552	**0.9560**	0.9547	0.9556	0.9519
		SSIM	0.9734	0.9732	0.9710	**0.9741**	0.9738	0.9608
	12	PSNR	**34.4007**	34.0473	33.6997	34.3080	34.0099	32.3060
		FSIM	**0.9801**	0.9790	0.9756	0.9785	0.9791	0.9689
		SSIM	**0.9856**	0.9848	0.9824	0.9849	0.9827	0.9733
Image5	4	PSNR	28.0560	28.0519	28.0299	**28.0563**	28.0559	28.0521
		FSIM	0.9737	0.9731	0.9736	0.9736	**0.9738**	**0.9738**
		SSIM	0.9432	0.9431	0.9415	0.9433	**0.9434**	0.9429
	8	PSNR	32.8669	32.8002	32.7167	**32.8759**	32.8484	32.5101
		FSIM	0.9921	0.9927	0.9911	0.9925	**0.9931**	0.9915
		SSIM	**0.9777**	0.9767	0.9771	0.9774	0.9775	0.9750
	12	PSNR	**35.7464**	35.5937	35.3063	35.7413	35.6511	34.6219
		FSIM	0.9972	**0.9974**	0.9969	0.9963	0.9973	0.9966
		SSIM	**0.9873**	0.9862	0.9855	0.9868	0.9870	0.9828
Image6	4	PSNR	26.7158	26.7134	26.7037	**26.7157**	26.7117	26.7076
		FSIM	0.9621	0.9634	0.9608	0.9622	**0.9637**	0.9634
		SSIM	**0.8890**	0.8875	0.8884	0.8887	0.8888	0.8894
	8	PSNR	**31.5393**	31.5267	31.3615	31.5503	31.5296	30.7903
		FSIM	0.9921	**0.9929**	0.9910	0.9922	0.9918	0.9889
		SSIM	**0.9515**	0.9517	**0.9515**	0.9513	0.9512	0.9435
	12	PSNR	34.4919	34.4280	33.8199	**34.5092**	34.2597	33.3243
		FSIM	0.9973	**0.9979**	0.9954	0.9975	0.9972	0.9967
		SSIM	**0.9739**	0.9728	0.9700	0.9733	0.9729	0.9656
Friedman's mean rank			**4.7778**	3.8981	2.5093	3.9907	4.0278	1.7963
Rank			**1**	4	5	3	2	6

(SSIM) (Zhou et al., 2004) are estimated. A higher PSNR, a value nearer to 1 for FSIM and SSIM means a better-thresholded image.

The statistical performance PSNR, FSIM, and SSIM based on the optimal threshold values of the test images (Image1 to Image 6) are presented in Table 3, the bold faces are the best results. From Table 3, it is evident that in the majority of the cases the NSD-LSMA based multilevel thresholding outperforms other optimization algorithms. From the Friedman mean rank (Derrac et al., 2011) statistical analysis considering the optimal PSNR, FSIM, and SSIM values of the test images (Image1-Image6), it is found that LSMA based approach ranked first in obtaining the optimal threshold values. Figs. 8–13 show the thresholded images and their corresponding histograms (test images are shown in Fig. 7). For an illustration, two from each threshold level 4, 8, and 12 are displayed. Our optimizer LSMA also enforces the thresholded image histogram to remain very close to the original one. Therefore, the

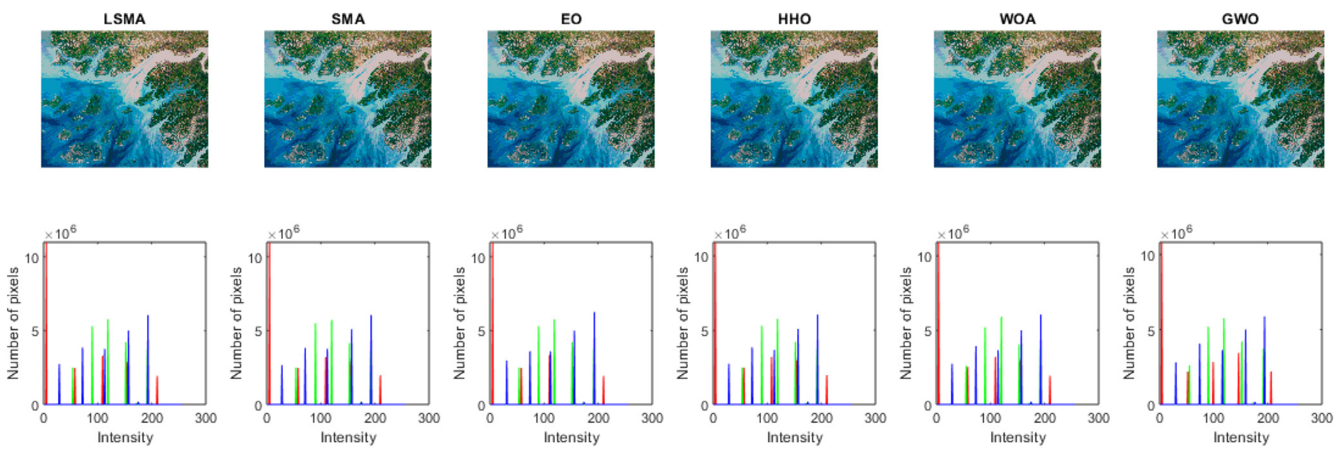

Fig. 8. For **Image1**, 4-level thresholded images, and their corresponding histograms.

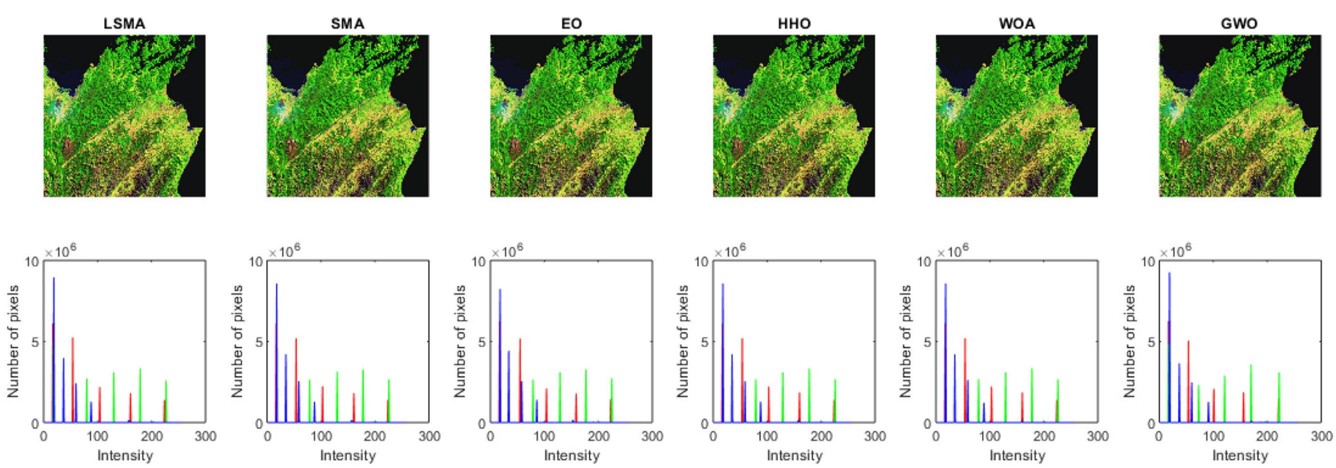

Fig. 9. For **Image2**, 4-level thresholded images, and their corresponding histograms.

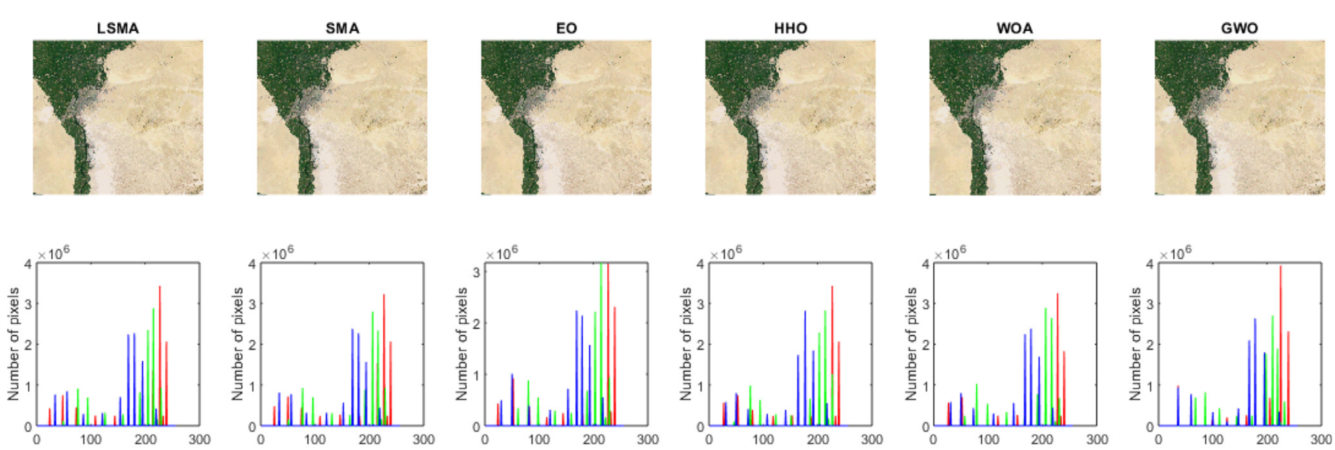

Fig. 10. For **Image3**, 8-level thresholded images, and their corresponding histograms.

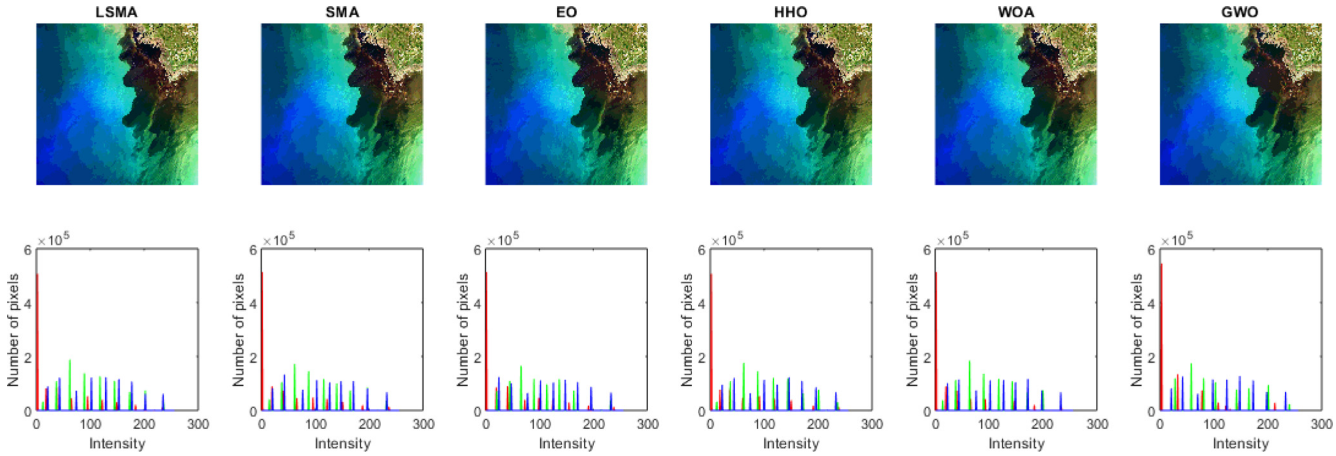

Fig. 11. For **Image4**, 8-level thresholded images, and their corresponding histograms.

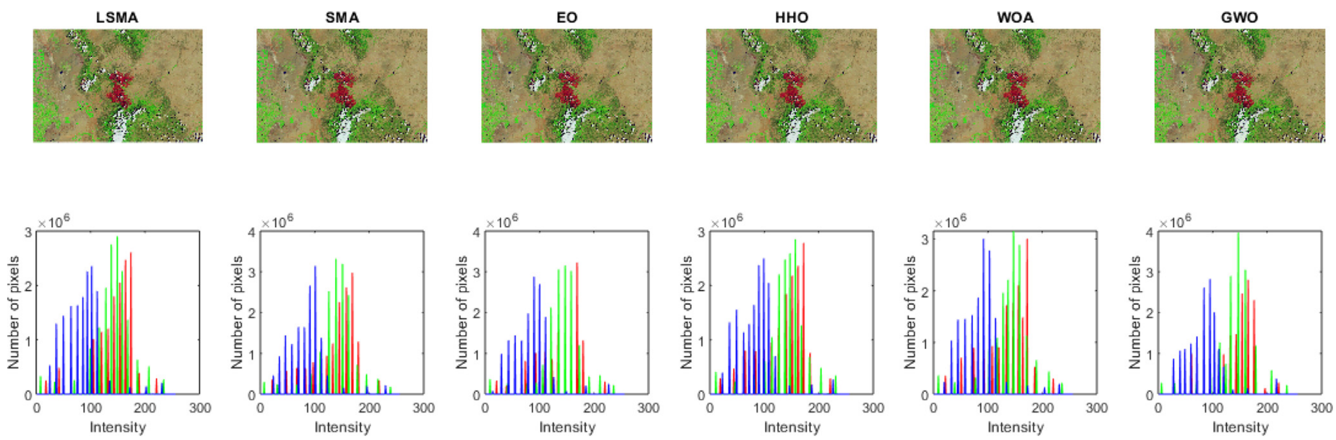

Fig. 12. For **Image5**, 12-level thresholded images, and their corresponding histograms.

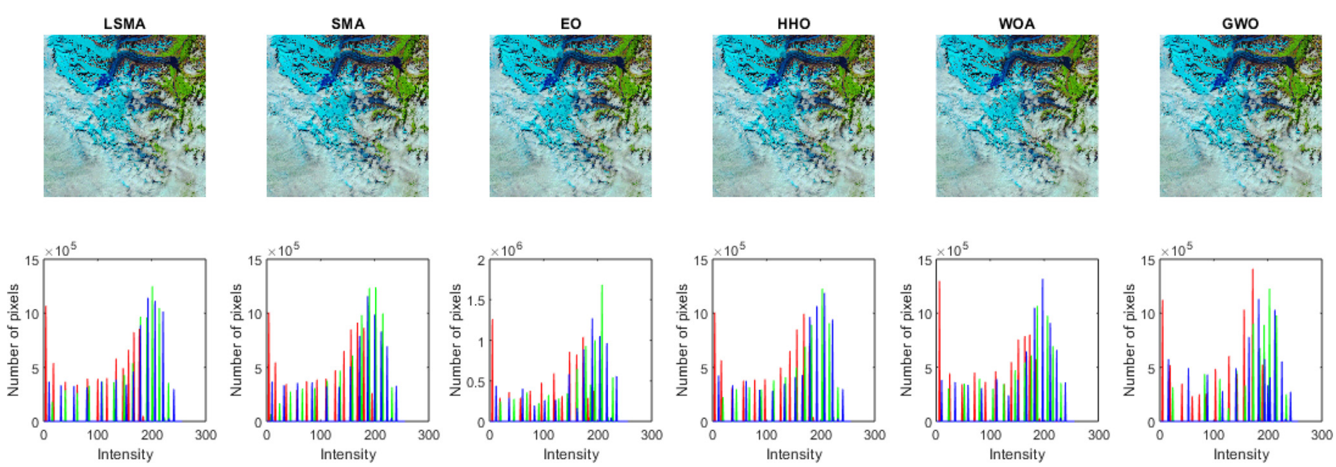

Fig. 13. For **Image6**, 12-level thresholded images, and their corresponding histograms.

NSD-LSMA based approach is competent for multilevel thresholding.

6. Conclusions

In this study, an efficient methodology in terms of both multi-level thresholding accuracy and time is fostered. It inherently includes the features for both exploration and exploitations with reduced computations. Profound differences are seen from a computation point of view. Even more interesting is its capability to handle both low/high dimension problems. Exemplar solutions are embodied in this paper to attract more readers. For completeness, both the statistical and numerical analysis is provided. From the results, it is revealed that the method also retains the contextual information; because the idea of squared difference based two-dimensional histogram takes care of this feature. The proposed method is well suited for the multiclass segmentation of satellite images. The reason may be due to its capability to handle high dimensional color images. To figure out, the method is ranked first while conducting the Friedman mean rank statistical analysis. Most of the PSNR, FSIM, and SSIM values are found optimal because it produced the best multiclass segmented outputs. Especially, it is quite efficient for computations, as opposed to the existing methods. The number of computations is reduced drastically by a factor of (k-1), where k is the number of threshold levels. Therefore, it would be more beneficial for high threshold levels. It means, realistically, our method would be very useful for multispectral color image analysis. Other future applications include – breast color thermogram analysis, brain MR image analysis, color image segmentation, etc.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abutaleb, A.S., 1989. Automatic thresholding of gray-level pictures using two-dimensional entropy. Comput. Vision Graph. Image Process. 47, 22–32. https://doi.org/10.1016/0734-189X(89)90051-0.

Agrawal, S., Panda, R., Bhuyan, S., Panigrahi, B.K., 2013. Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm. Swarm Evol. Comput. 11, 16–30. https://doi.org/10.1016/j.swevo.2013.02.001.

Baby Resma, K.P., Nair, M.S., 2018. Multilevel thresholding for image segmentation using Krill Herd Optimization algorithm. J. King Saud Univ. - Comput. Inf. Sci. https://doi.org/10.1016/j.jksuci.2018.04.007.

Bao, X., Jia, H., Lang, C., 2019. A novel hybrid Harris Hawks Optimization for Color Image Multilevel Thresholding Segmentation. IEEE Access 7, 76529–76546. https://doi.org/10.1109/ACCESS.2019.2921545.

Bhandari, A., 2015. Tsallis entropy based multilevel thresholding for colored satellite image segmentation using evolutionary algorithms. Expert Syst. Appl. https://doi.org/10.1016/j.eswa.2015.07.025.

Bhandari, A.K., Kumar, A., Chaudhary, S., Singh, G.K., 2016. A novel color image multilevel thresholding based segmentation using nature inspired optimization algorithms. Expert Syst. Appl. 63, 112–133. https://doi.org/10.1016/j.eswa.2016.06.044.

Bhandari, A.K., Singh, V.K., Kumar, A., Singh, G.K., 2014. Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy. Expert Syst. Appl. 41, 3538–3560. https://doi.org/10.1016/j.eswa.2013.10.059.

Brink, A.D., 1992. Thresholding of digital images using two-dimensional entropies. Pattern Recognit. 25, 803–808. https://doi.org/10.1016/0031-3203(92)90034-G.

Derrac, J., García, S., Molina, D., Herrera, F., 2011. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol. Comput. 1, 3–18. https://doi.org/10.1016/j.swevo.2011.02.002.

El Aziz, M.A., Ewees, A.A., Hassanien, A.E., Mudhsh, M., Xiong, S., 2018. Multi-objective whale optimization algorithm for multilevel thresholding segmentation. In: Hassanien, A.E., Oliva, D.A. (Eds.), Advances in Soft Computing and Machine Learning in Image Processing. Springer International Publishing, Cham, pp. 23–39. https://doi.org/10.1007/978-3-319-63754-9_2.

Faramarzi, A., Heidarinejad, M., Stephens, B., Mirjalili, S., 2020. Equilibrium optimizer: A novel optimization algorithm. Knowledge-Based Syst. 191. https://doi.org/10.1016/j.knosys.2019.105190.

Haindl, M., Mikeš, S., 2016. A competition in unsupervised color image segmentation. Pattern Recognit. 57, 136–151. https://doi.org/10.1016/j.patcog.2016.03.003.

Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., Chen, H., 2019. Harris hawks optimization: Algorithm and applications. Futur. Gener. Comput. Syst. 97, 849–872. https://doi.org/10.1016/j.future.2019.02.028.

Jia, H., Ma, J., Song, W., 2019a. Multilevel thresholding segmentation for color image using modified moth-flame optimization. IEEE Access 7, 44097–44134. https://doi.org/10.1109/ACCESS.2019.2908718.

Jia, Heming, Peng, X., Song, W., Oliva, D., Lang, C., Li, Y., Yao, Y., 2019b. Masi entropy for satellite color image segmentation using tournament-based Lévy multiverse optimization algorithm. Remote Sens. 11, 942. https://doi.org/10.3390/rs11080942.

Kapur, J.N.N., Sahoo, P.K.K., Wong, A.K.C.K.C., 1985. A new method for gray-level picture thresholding using the entropy of the histogram. Comput. Vision Graph. Image Process. 29, 273–285. https://doi.org/10.1016/0734-189X(85)90125-2.

Khairuzzaman, A.K.M., Chaudhury, S., 2017. Multilevel thresholding using grey wolf optimizer for image segmentation. Expert Syst. Appl. 86, 64–76. https://doi.org/10.1016/j.eswa.2017.04.029.

Landsat Image Gallery [WWW Document], n.d. URL https://landsat.visibleearth.nasa.gov/ (accessed 6.12.20).

Li, C.H., Lee, C.K., 1993. Minimum cross entropy thresholding. Pattern Recognit. 26, 617–625. https://doi.org/10.1016/0031-3203(93)90115-D.

Li, S., Chen, H., Wang, M., Heidari, A.A., Mirjalili, S., 2020. Slime mould algorithm: A new method for stochastic optimization. Futur. Gener. Comput. Syst. https://doi.org/10.1016/j.future.2020.03.055.

Liang, J.J., Qu, B.Y., Suganthan, P.N., 2013. Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization.

Liao, P.-S., Chen, T.-S., Chung, P.-C., 2001. A fast algorithm for multilevel thresholding. J. Inf. Sci. Eng. 17, 713–727.

Liu, J., Li, W., Tian, Y., 1991. Automatic thresholding of gray-level pictures using two-dimension Otsu method. In: China: 1991 International Conference on Circuits and Systems, pp. 325–327. https://doi.org/10.1109/CICCAS.1991.184351.

Mirjalili, S., Lewis, A., 2016. The whale optimization algorithm. Adv. Eng. Softw. 95, 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008.

Mirjalili, S., Mirjalili, S.M., Lewis, A., 2014. Grey wolf optimizer. Adv. Eng. Softw. 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007.

Naik, M.K., Panda, R., 2016. A novel adaptive cuckoo search algorithm for intrinsic discriminant analysis based face recognition. Appl. Soft Comput. 38, 661–675. https://doi.org/10.1016/j.asoc.2015.10.039.

Nie, F., Zhang, P., Li, J., Ding, D., 2017. A novel generalized entropy and its application in image thresholding. Signal Process. 134, 23–34. https://doi.org/10.1016/j.sigpro.2016.11.004.

Otsu, N., 1979. A threshold selection method from Gray-level histograms. IEEE Trans. Syst. Man. Cybern. 9, 62–66. https://doi.org/10.1109/TSMC.1979.4310076.

Portes de Albuquerque, M., Esquef, I.A., Gesualdi Mello, A.R., Portes de Albuquerque, M., 2004. Image thresholding using Tsallis entropy. Pattern Recognit. Lett. 25, 1059–1065. https://doi.org/10.1016/j.patrec.2004.03.003.

Sahoo, P.K., Arora, G., 2006. Image thresholding using two-dimensional Tsallis–Havrda–Charvát entropy. Pattern Recognit. Lett. 27, 520–528. https://doi.org/10.1016/j.patrec.2005.09.017.

Sahoo, P.K., Arora, G., 2004. A thresholding method based on two-dimensional Renyi's entropy. Pattern Recognit. 37, 1149–1161. https://doi.org/10.1016/j.patcog.2003.10.008.

Sarkar, S., Das, S., 2013. Multilevel image thresholding based on 2D histogram and maximum Tsallis entropy— a differential evolution approach. IEEE Trans. Image Process. 22, 4788–4797. https://doi.org/10.1109/TIP.2013.2277832.

Sarkar, S., Sen, N., Kundu, A., Das, S., Sinha Chaudhuri, S., 2013. A differential evolutionary multilevel segmentation of near infra-red images using Renyi's entropy BT - proceedings of the international conference on frontiers of intelligent computing: theory and applications (FICTA). Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 699–706.

Sezgin, M., Sankur, B., 2004. Survey over image thresholding techniques and quantitative performance evaluation. J. Electron. Imaging 13, 146–168. https://doi.org/10.1117/1.1631315.

Upadhyay, P., Chhabra, J.K., 2019. Kapur's entropy based optimal multilevel image segmentation using Crow Search Algorithm. Appl. Soft Comput. https://doi.org/10.1016/j.asoc.2019.105522.

Wunnava, A., Kumar Naik, M., Panda, R., Jena, B., Abraham, A., 2020. A differential evolutionary adaptive Harris hawks optimization for two dimensional practical Masi entropy-based multilevel image thresholding. J. King Saud Univ. - Comput. Inf. Sci. https://doi.org/10.1016/j.jksuci.2020.05.001.

Xu, L., Jia, H., Lang, C., Peng, X., Sun, K., 2019. A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution. IEEE Access 7, 19502–19538. https://doi.org/10.1109/ACCESS.2019.2896673.

Yin, P.-Y., 1999. A fast scheme for optimal thresholding using genetic algorithms. Signal Process. 72, 85–95. https://doi.org/10.1016/S0165-1684(98)00167-4.

Yin, P.-Y., Chen, L.-H., 1997. A fast iterative scheme for multilevel thresholding methods. Signal Process. 60, 305–313. https://doi.org/10.1016/S0165-1684(97)00080-7.

Zhang, L., Zhang, L., Mou, X., Zhang, D., Lin Zhang, L.Z., 2011. FSIM: a feature similarity index for image quality assessment. IEEE Trans. Image Process. 20, 2378–2386. https://doi.org/10.1109/TIP.2011.2109730.

Zhou, W., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13, 600–612. https://doi.org/10.1109/TIP.2003.819861.