Applied Soft Computing Journal 89 (2020) 106063

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

An efficient krill herd algorithm for color image multilevel thresholding segmentation problem

Lifang He [a,*], Songwei Huang [b]

[a] Department of Electronics and Communication Engineering, Kunming University of Science and Technology, Kunming 650093, China
[b] Department of Mineral Processing, Kunming University of Science and Technology, Kunming 650093, China

ARTICLE INFO

Article history:
Received 29 September 2018
Received in revised form 26 December 2019
Accepted 30 December 2019
Available online 3 January 2020

Keywords:
Image segmentation
Color image multilevel thresholding segmentation
Swarm intelligence algorithm
Krill herd algorithm

ABSTRACT

The conventional thresholding methods are very efficient for bi-level thresholding, but the computational complexity may be excessively high for color image multilevel thresholding. Color image multilevel thresholding segmentation can be considered as a constrained optimization problem, therefore swarm intelligence algorithms are widely used to reduce the complexity. In this paper, an efficient krill herd (EKH) algorithm is proposed to search optimal thresholding values at different level for color images and the Otsu's method, Kapur's entropy and Tsallis entropy are employed as objective functions. Seven different algorithms, KH without any genetic operators (KH I), KH with crossover operator (KH II), KH with crossover and mutation operators (KH IV), modified firefly algorithm (MFA), modified grasshopper optimization algorithm (MGOA), bat algorithm (BA) and water cycle algorithm (WCA), are compared with the EKH algorithm. Experiments are performed on ten color benchmark images in terms of optimal threshold values, objective values, PSNR, SSIM and standard deviation of the objective values at different levels. The experimental results show that the presented EKH algorithm is superior to the other algorithms for color image multilevel thresholding segmentation. On the other hand, Kapur's entropy is found to be more accurate and robust for color image multilevel thresholding segmentation.

1. Introduction

Image segmentation plays an important and effective role in image processing and pattern recognition. The process of image segmentation of a given image is to divide the image into several objects and regions and the goal is to separate objects from backgrounds. So far a large number of image segmentation methods have been proposed by many scholars and researchers, such as thresholding [1], clustering-based method [2], edge detection [3], region-growing method [4], Graph partitioning-based methods [5]. Among all the existing segmentation methods [2–4], thresholding is an efficient and popular techniques because of its simplicity and high efficiency [6,7]. However, color image thresholding segmentation is still a challenging task as it includes 3-D histogram unlike the 1-D histogram of gray-scale images.

Thresholding plays a very effective role in the field of image segmentation. It can be classified into two groups: bi-level and multi-level thresholding. The simplest thresholding method is bi-level thresholding segmentation that splits an image into two classes by searching a single thresholding value, while dividing the image into many different parts by setting multiple threshold values is called multilevel thresholding [8,9]. If a given color image is divided into multiple regions according to color by setting numerous thresholding values, then it is known as color image multilevel thresholding. Because color images have different color components and a variety of color intensity, there is a strong requirement for color image multilevel thresholding segmentation problem.

Over the years, various thresholding segmentation methods have been proposed [10,11]. In 1979, Otsu's method [1] named after Nobuyuki Otsu, is presented for thresholding segmentation. It exhaustively searches optimal threshold values by maximizing the between-class variance. In 1985, researchers Kapur et al. [12] introduced a new technique for gray-level image thresholding using entropy of the histogram called Kapur entropy technique. Tsai [10] proposed a novel method for gray-level image by using the moment-preserving principle called Tsallis entropy method widely used for image thresholding segmentation. In 1993, Li and Lee [13] selected optimal thresholding values by minimizing the cross entropy between the original image and the segmented image. These thresholding segmentation methods can be easily extended to multilevel thresholding. At present, multilevel thresholding methods draw may attention in image segmentation and widely used in numerous applications, for example, Synthetic Aperture Radar (SAR) image segmentation [14,15], medical

* Corresponding author.
E-mail address: Lifang@kmust.edu.cn (L. He).

https://doi.org/10.1016/j.asoc.2020.106063
1568-4946/

Initialize the population P of NP krill randomly, V_f , $D^{\max}$ and $N^{\max}$
Calculate objective function for each krill according to its initial position
Initial generation, k=0
While k< MaxGenerations do
Sort the population according to their objective function values.
Update inertia weights via equation (39).
Update C^{best} parameter via equation (40).
for i=1:NP(all krill) do
Perform the following motion calculation.
Motion induced by other individuals
Foraging motion
Physical diffusion
Update the krill position in the search space by equation (37).
Then update the krill position in the search space by equation (41).
Calculate objective function for each krill according to its new position.
end for i
G=G+1.
end while
Results and visualization.

Fig. 1. The pseudo code of EKH algorithm.

image processing [16,17], satellite image segmentation [18–20], infrared image segmentation [21,22]. However, computational costs of the conventional multilevel thresholding algorithms will quickly increase for multilevel thresholding problem with increasing thresholding values as they search the best thresholding values exhaustively to optimize the objective function. Therefore, it brings obstacles to the real time application.

The process of searching optimal thresholding values for multilevel thresholding segmentation can be considered as a constrained optimization problem. To overcome the computational complexity issues, over the years, many swarm intelligence (SI) algorithms and their modified algorithms have been used for multilevel thresholding segmentation, e.g., particle swarm optimization (PSO) algorithm [23–25], genetic algorithm (GA) [26,27], differential evolution (DE) algorithm [28–30], artificial bee colony (ABC) algorithm [31–33], cuckoo search (CS) algorithm [6,7,20,34], firefly algorithm (FA) [34–36], wind driven optimization (WDO) [7] etc. The inspiration of these algorithms often comes from nature, especially biological systems. Now the most common swarm intelligence algorithms include PSO simulate the social behavior of bird flock or fish school [37–39], GA mimics the process of natural selection [40,41], ABC simulates the foraging behavior of honey bees [42–44], CS algorithm was inspired by the behavior of some cuckoo species of laying their eggs in the nests of other species [45,46], FA algorithm mimics the flashing behavior of fireflies in nature [47–50], WDO algorithm was inspired by the motion of atmospheric [51,52].

In recent years, color image multilevel thresholding segmentation methods using swarm intelligence algorithms have also been widely used for image segmentation. Pare et al. [53] proposed an efficient method for color image multilevel thresholding segmentation using cuckoo search algorithm based on minimum cross entropy. Bhandari et al. [54] applied DE, WDO, PSO

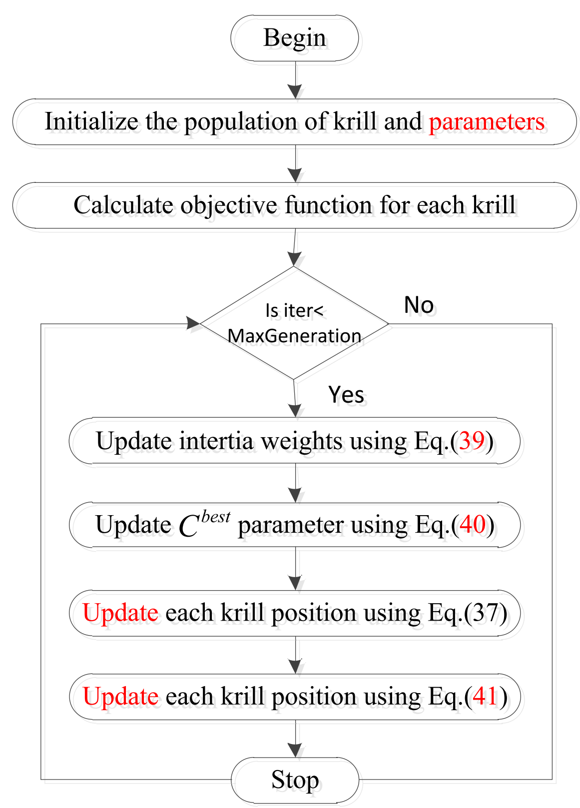

Fig. 2. The flowchart of EKH algorithm.

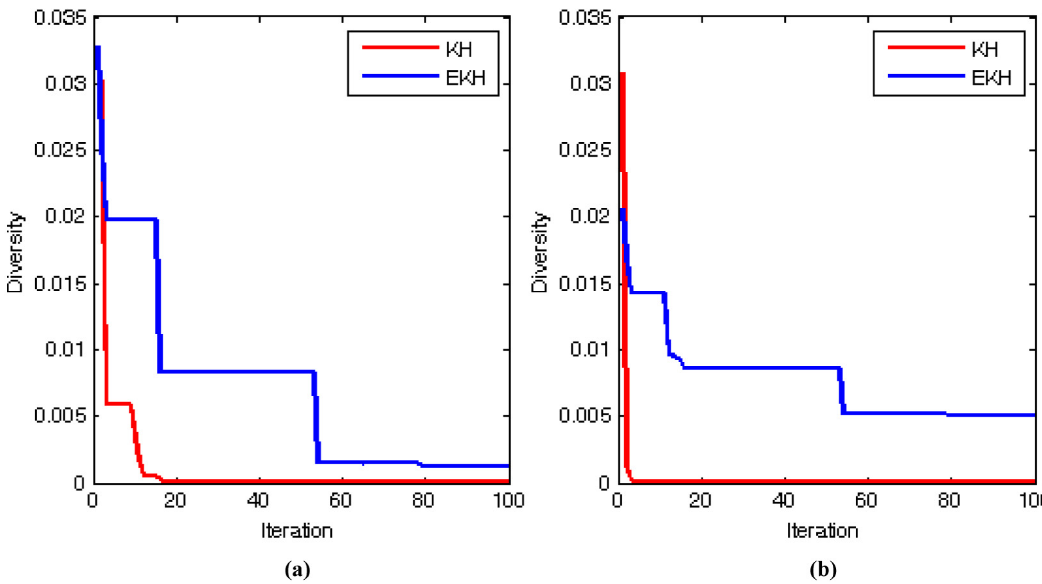

Fig. 3. The diversity curve of EKH and KH algorithm: (a) for f_1, (b) for f_2.

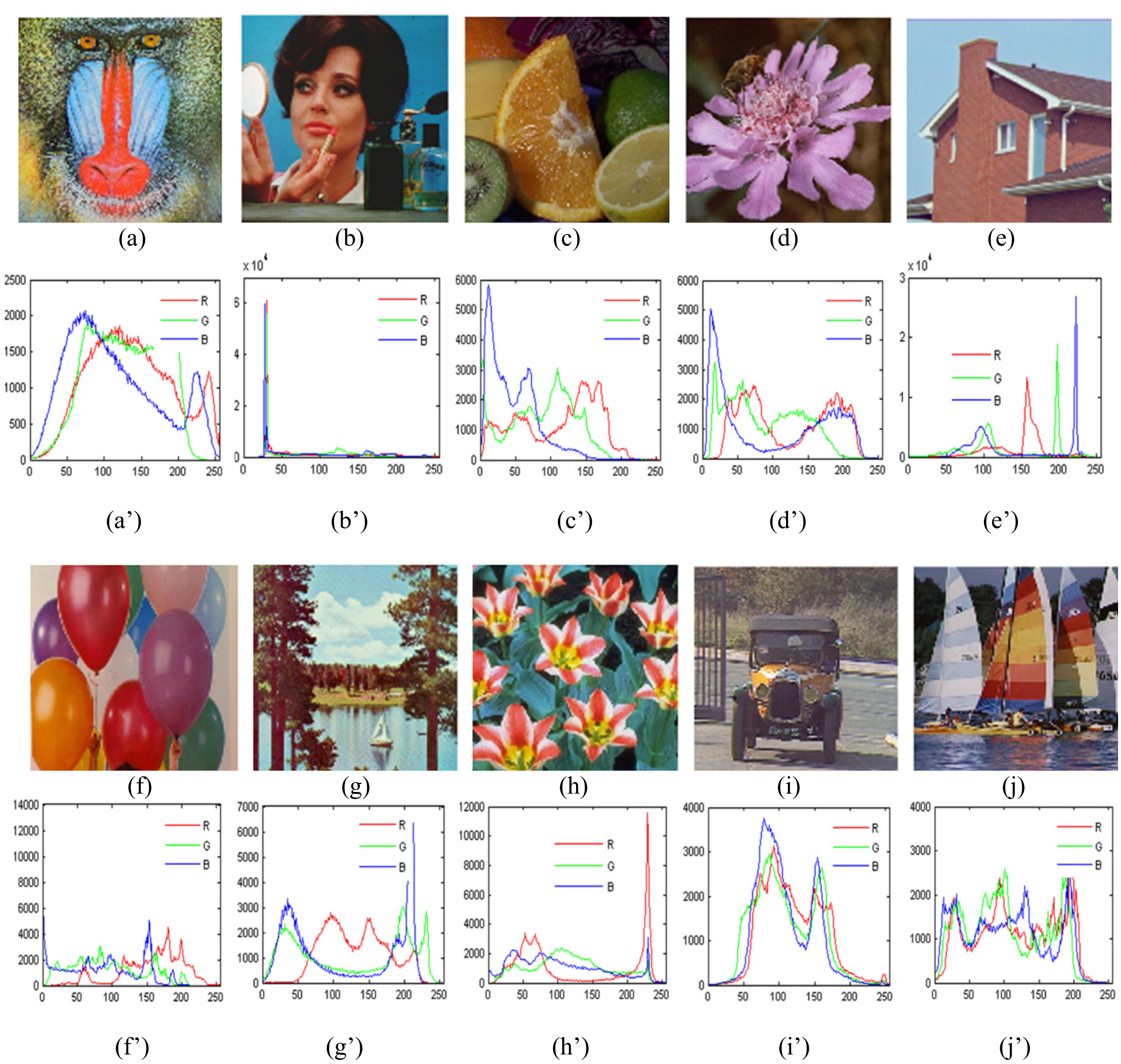

Fig. 4. The ten test images (a) baboon, (b) donna, (c) fruit, (d) flower, (e) house, (f) pallon, (g) sailboat, (h) tulips, (i) voit, (j) yacht; and corresponding histograms (a’) baboon, (b’) donna, (c’) fruit, (d’) flower, (e’) house, (f’) pallon, (g’) sailboat, (h’) tulips, (i’) voit, (j’) yacht.

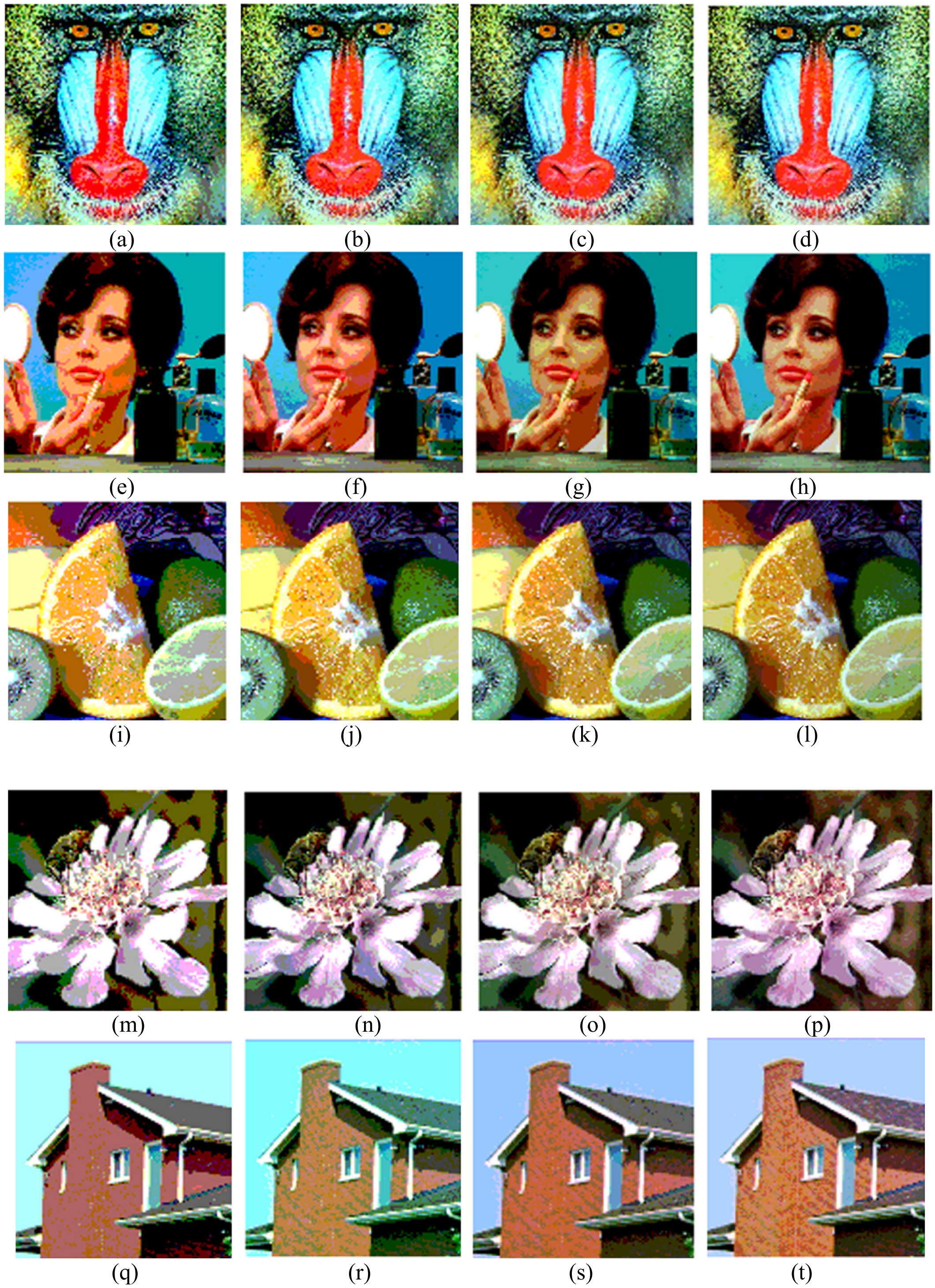

Fig. 5. For m = 3, 4, 5 and 6, images (a)–(d) for Baboon, (e)–(h) for Donna, (i)–(l) for Fruit, (m)–(p) for Flower, (q)–(t) for House, using EKH algorithm based on Otsu's method.

and CS algorithms for color satellite image multilevel segmentation. Liu et al. [55] presented a novel multilevel thresholding method for color image segmentation using cooperative bacterial foraging algorithm (CBFA), the performance of the proposed algorithm has been tested with various standard test images and compared with traditional Bacterial Foraging Algorithm. Dey et al. [56] proposed quantum inspired ant colony optimization, quantum inspired differential evolution and quantum inspired particle swarm optimization for color image multilevel thresholding segmentation. Rajinikanth and Couceiro [57] proposed new segmentation method based on RGB histogram using Brownian search based Firefly Algorithm (BFA), Lévy search based Firefly Algorithm (LFA), and Firefly Algorithm (FA). Sarkar et al. [58] proposed a new multilevel thresholding method based on minimum cross entropy and differential evolution for color image. Kurban et al. [59] have introduced evolutionary and swarm-based

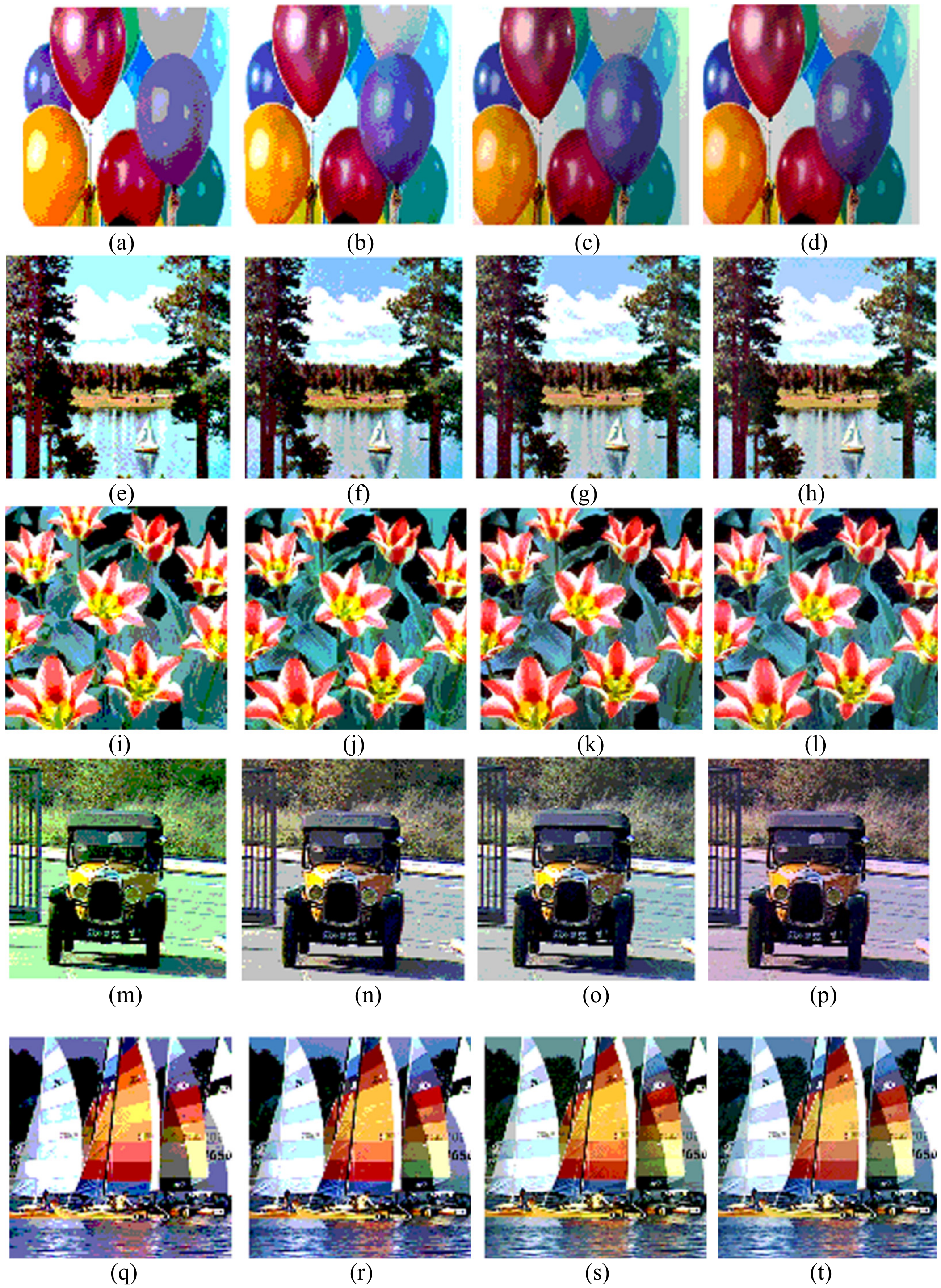

Fig. 6. For m = 3, 4, 5 and 6, images (a)–(d) for Pallon, (e)–(h) for Sailboat, (i)–(l) for Tulips, (m)–(p) for Voit, (q)–(t) for Yacht, using EKH algorithm based on Otsu's method.

optimization algorithms for color image multilevel thresholding segmentation.

In 2012, Gandomi and Alavi [60] presented a new swarm intelligence algorithm named krill herd (KH) algorithm which is used to mimic the herding behavior of krill individuals. In the KH algorithm, during the course of movement the minimum distances of each krill from food and highest density of the herd are defined as the objective function. Many simulation results in the related literatures reveal that the KH algorithm outperforms some well-known swarm intelligence algorithms such as GA, PSO, DE, ABC, ACO, CS algorithms for solving optimization tasks [60–63], so it has attained a lot of interest and has been widely employed for solving several practical and complex optimization problems.

To improve the performance of KH algorithm, Gandomi and Alavi [60] incorporate crossover and mutation operator into the algorithm which are inspired from the DE algorithm and four different KH algorithms are presented that are basic KH without

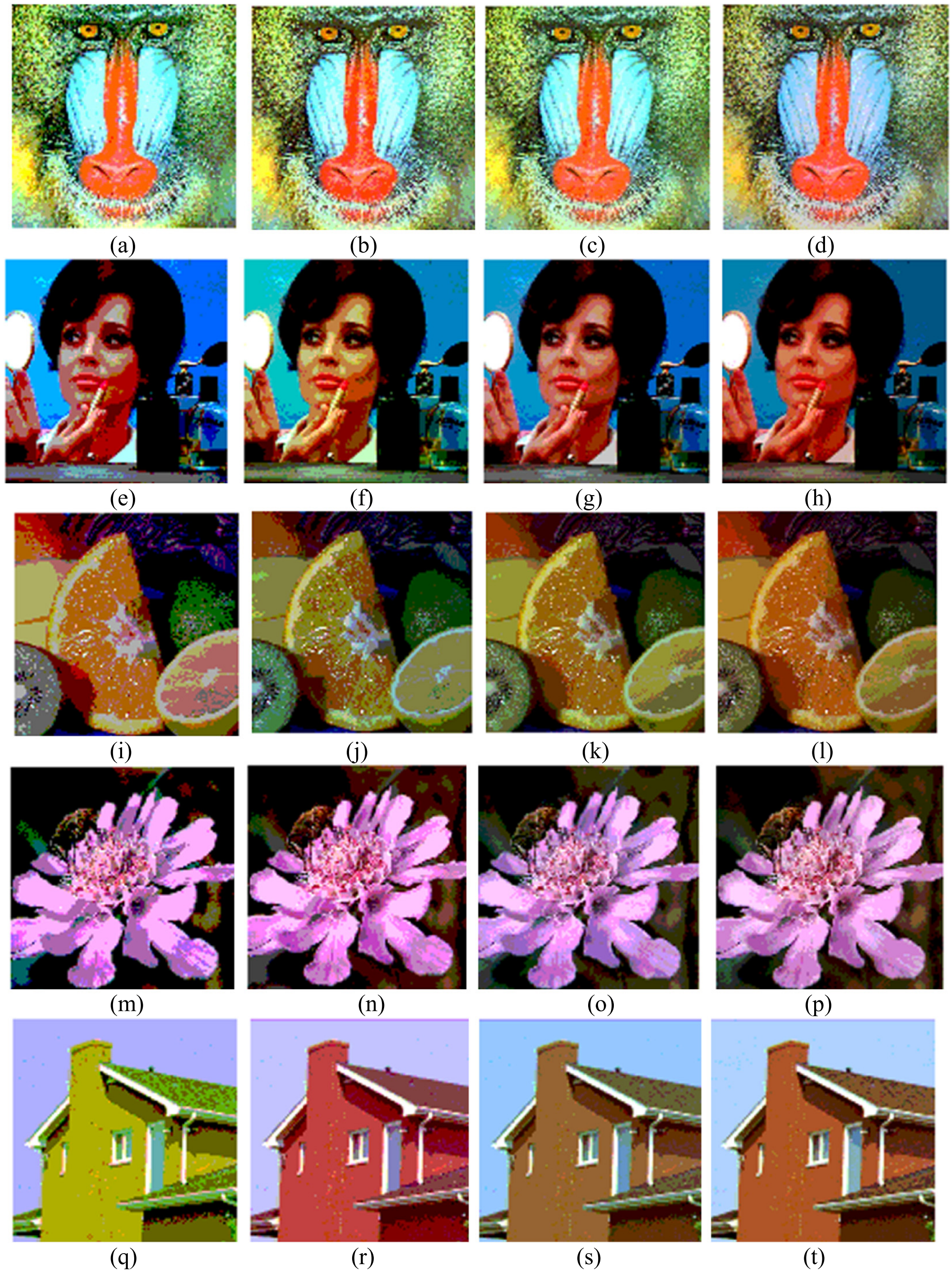

Fig. 7. For m = 3, 4, 5 and 6, images (a)–(d) for Baboon, (e)–(h) for Donna, (i)–(l) for Fruit, (m)–(p) for Flower, (q)–(t) for House, using EKH algorithm based on Kapur's entropy.

any genetic operators (KH I), KH with crossover operator (KH II), KH with mutation operator (KH III) and KH with crossover and mutation operators (KH IV). The algorithms are tested using several well-known benchmark problems and compared with the other optimization algorithms. The results show that the KH II shows the best performance in term of robustness, KH I and KH IV are better than the other algorithms for convergence performance.

The Otsu's, Kapur's entropy and Tsallis entropy method are three common thresholding techniques and can be easily extended to multilevel thresholding segmentation. They are very effective for bi-level thresholding segmentation, but they are very time consuming for multilevel thresholding segmentation problem because they exhaustively find the optimal thresholding values by maximizing the objective function. Therefore, several swarm intelligence algorithms have been coupled with the conventional Otsu's method, Kapur's entropy and Tsallis entropy methods for multilevel image segmentation over the years. In

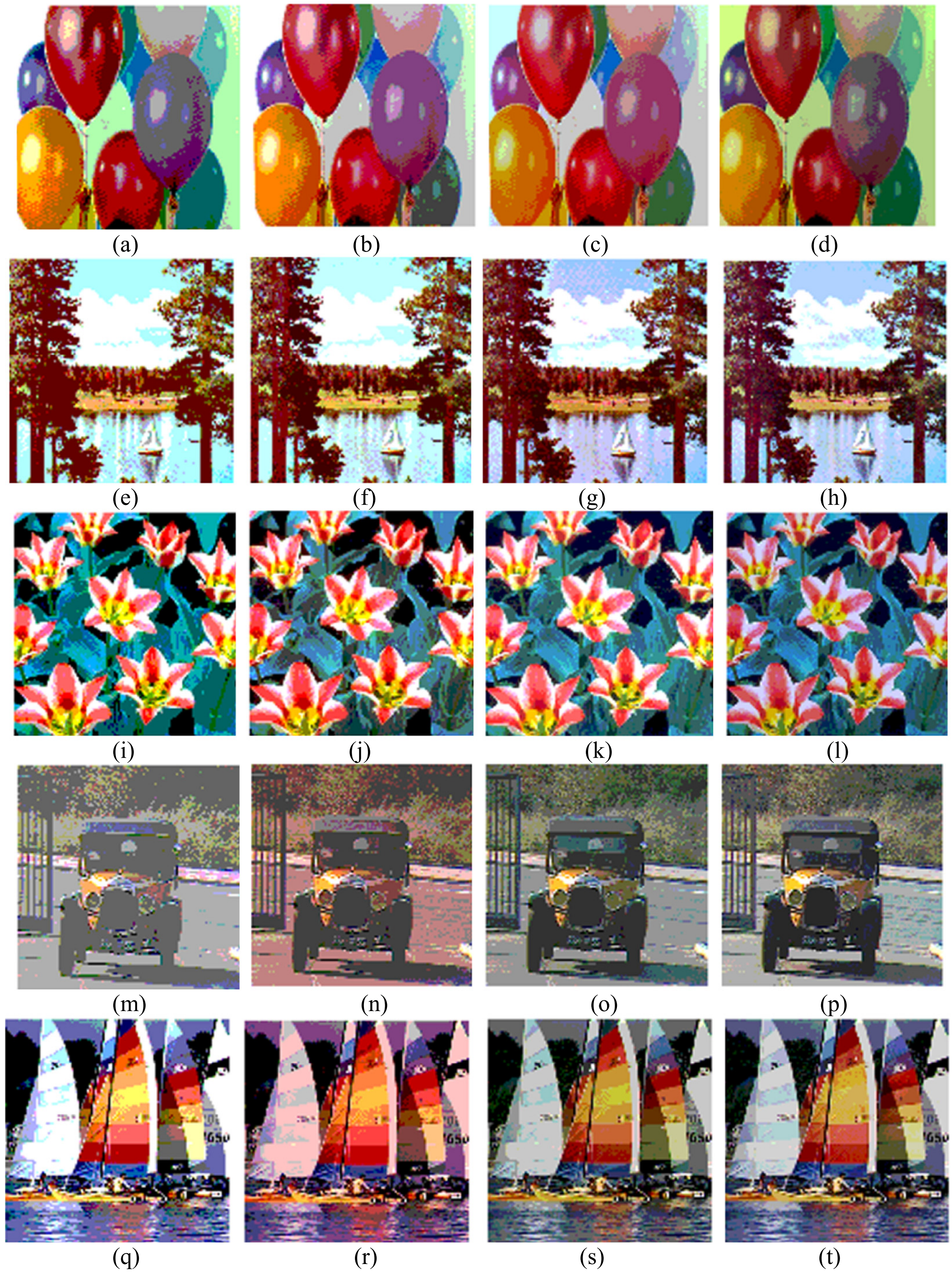

Fig. 8. For m = 3, 4, 5 and 6, images (a)–(d) for Pallon, (e)–(h) for Sailboat, (i)–(l) for Tulips, (m)–(p) for Voit, (q)–(t) for Yacht, using EKH algorithm based on Kapur's entropy.

2011, Sathya and Kayalvizhi [64] proposed a novel multilevel image segmentation method using bacterial foraging (BF) algorithm. The method search optimal thresholding values for maximizing the Kapur's and Otsu's objective functions. Experimental results show that the method has better performance. After that, in 2013, Akay [9] has introduced a new method, in which PSO and ABC algorithm have been used to find optimal threshold values for multilevel thresholding. Kapur's entropy and Otsu's method have been investigated as objective functions. In addition, Bhandari et al. [19] presented a modified artificial bee colony (MABC) algorithm based satellite image segmentation using Kapur's entropy, Otsu and Tsallis entropy as objective function to search the optimal multilevel thresholds. In 2017, Suresh and Lal [65] introduced a chaotic Darwinian Particle Swarm Optimization (CDPSO) algorithm for satellite image multilevel segmentation. Tsallis entropy and minimum cross entropy were used as objective functions.

The basic KH algorithm has two problems that are premature convergence and high computational complexity. Based on the above analysis, in order to modify the performance of KH

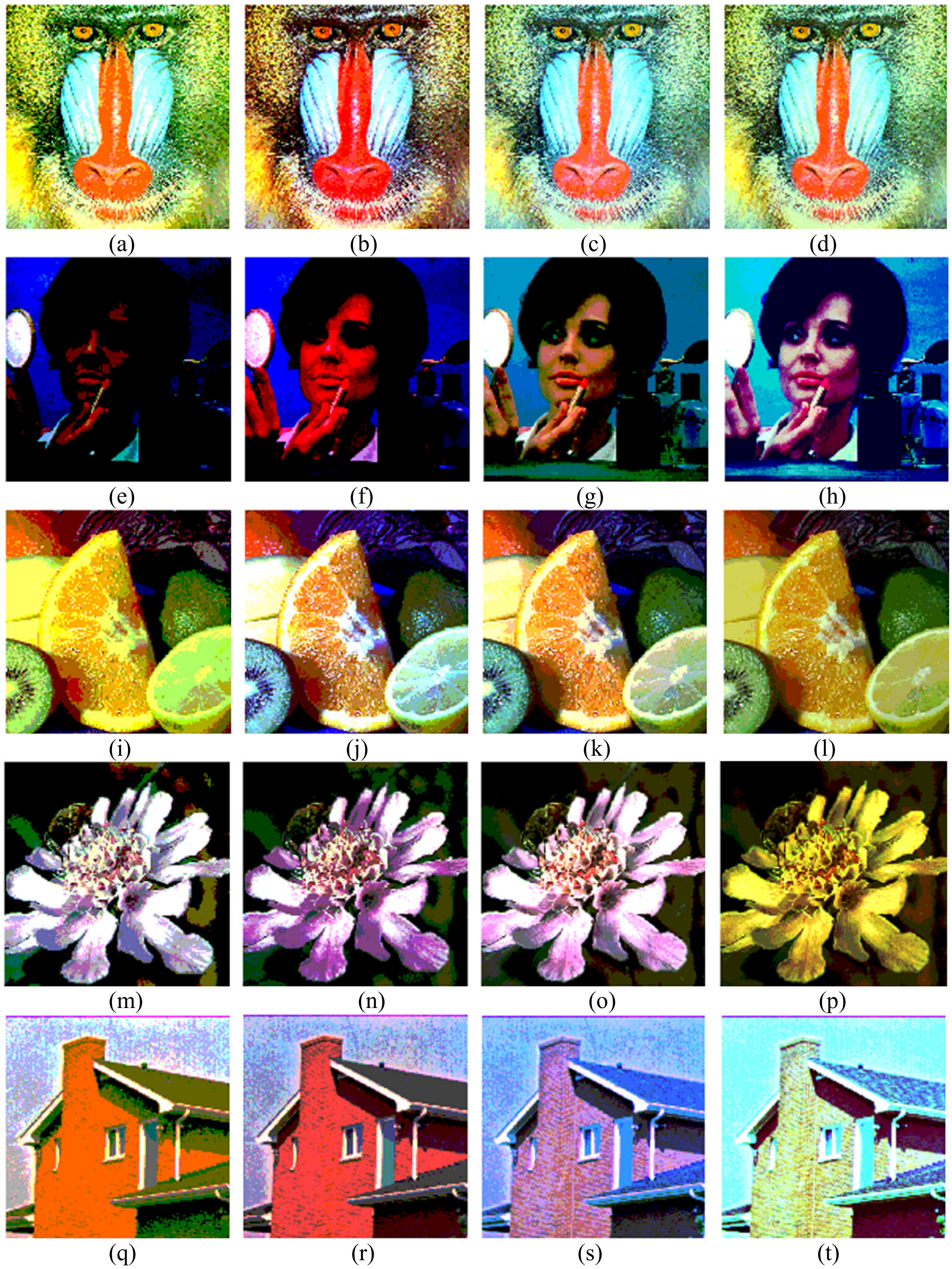

Fig. 9. For m = 3, 4, 5 and 6, images (a)–(d) for Baboon, (e)–(h) for Donna, (i)–(l) for Fruit, (m)–(p) for Flower, (q)–(t) for House, using EKH algorithm based on Tsallis entropy.

algorithm, KH II, KH III and KH IV are proposed. Although these improved KH algorithms improved the performance of the basic KH algorithm, the improved KH algorithm requires fast searching speeds and avoiding premature convergence and not to be implemented. To overcome these shortcomings, in this paper an efficient Krill Herd (EKH) algorithm are presented and used for color image multilevel thresholding segmentation. The inertia weights $\left(\omega_n, \omega_f\right)$ determine the convergence of KH algorithm, so a novel update method of inertia weights with the iteration numbers is presented in this paper. Furthermore the adaptive update scheme of C^{best} is proposed and a new position update method is introduced into the movement course of krill individuals.

Over the years, many swarm intelligence algorithms have been reported for image multilevel thresholding segmentation problem. In order to test the performance of the segmentation methods, the researchers have used different swarm intelligence algorithms with different objective functions for gray image or color image segmentation and given the results of the optimal thresholding value, optimal objective function, peak signal-to-noise ratio (PSNR), mean square error (MSE), feature similarity index (FSIM), structural similarity index (SSIM), and standard

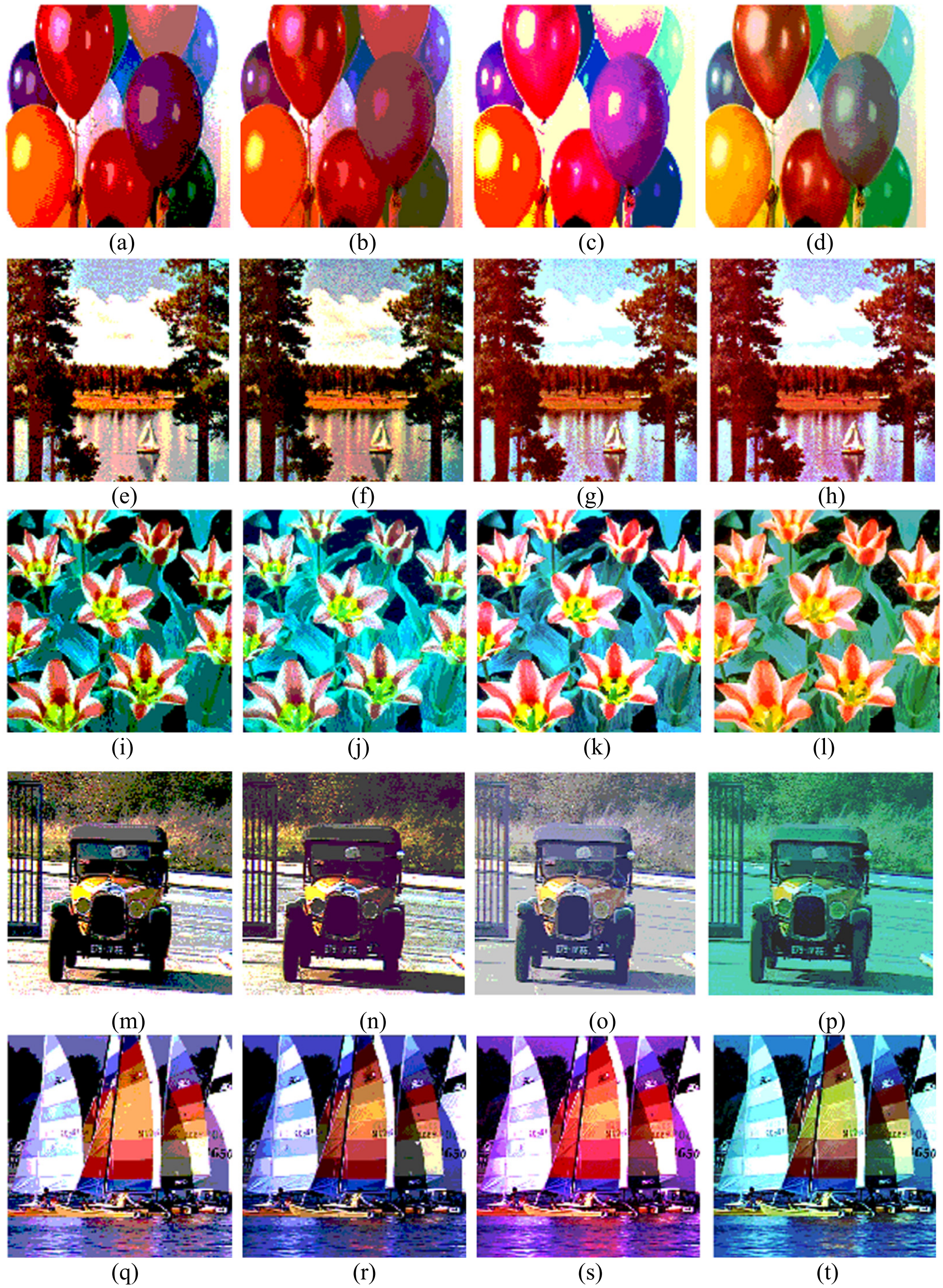

Fig. 10. For m = 3, 4, 5 and 6, images (a)–(d) for Pallon, (e)–(h) for Sailboat, (i)–(l) for Tulips, (m)–(p) for Voit, (q)–(t) for Yacht, using EKH algorithm based on Tsallis entropy.

Table 1
Benchmark functions.

Functions	Domains	Optimal	Dimension
$f_1(x) = \sum_{i=1}^{n} i x_i^2$	[−5.12,5.12]	0	30
$f_2(x) = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10)$	[−100,100]	0	30

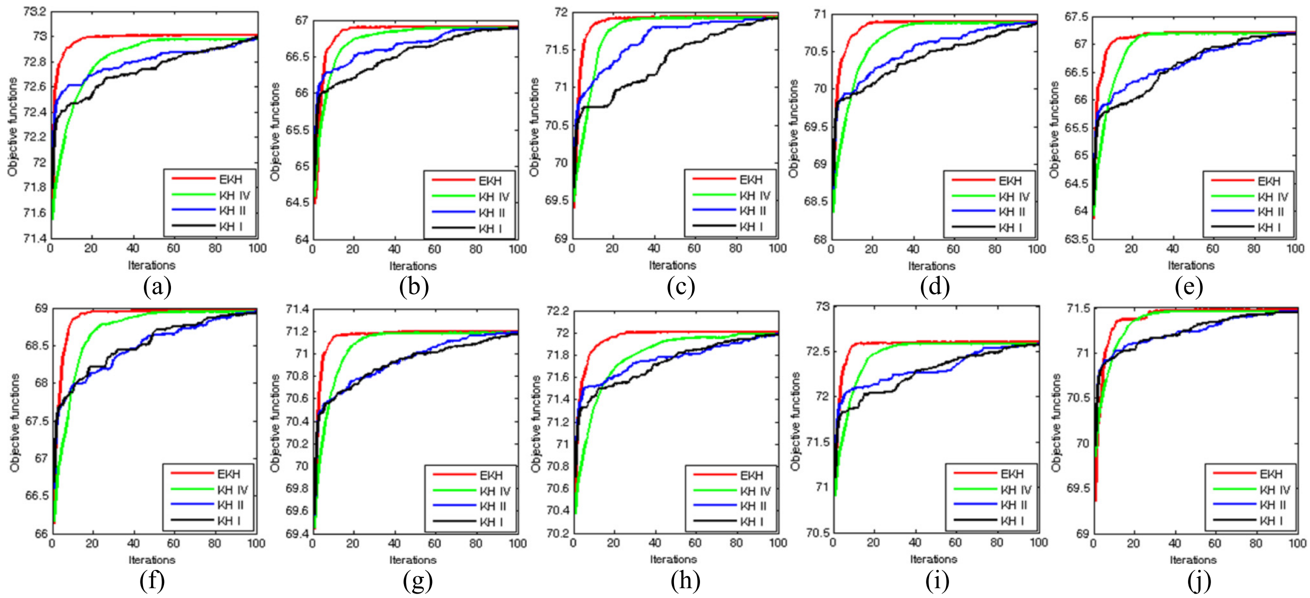

Fig. 11. The convergence curve of EKH, KH?, KH II and KH IV algorithm for 6 level of ten color test images using Kapur's entropy.

Table 2
Parameters of the comparison algorithms.

Algorithm	Parameters	Values
MFA	α	0.01
	β_0	1
	γ	1
BA	β	(0,1)
WCA	μ	0.1
MGOA	c_{max}	1
	c_{min}	0.0004
	Levy	0.8

deviation (SD). Furthermore, the results of the algorithm have been compared with other swarm intelligence algorithms.

For segmentation of practical application, a color image is more acceptable than the gray-scale images. Therefore, motivated by the wide range of application, this paper presented a new color image multilevel thresholding segmentation method based on the EKH algorithm using three different objective functions, such as Otsu's method, Kapur's entropy and Tsallis entropy. The method is used for maximizing between-class variance, Kapur's entropy and Tsallis entropy to find optimal thresholding values. The performance of EKH algorithm for color image multilevel segmentation is measured in terms of optimal thresholding values, objective functions, PSNR, SSIM and standard deviation of the objective values and compared with KH I algorithm, KH II algorithm, KH IV algorithm, modified firefly algorithm (MFA), modified grasshopper optimization algorithm (MGOA), bat algorithm (BA) and water cycle algorithm (WCA). In order to compare the performance of the algorithms, the convergence curves are also drawn.

The organization of this paper is as follows: Section 2 describes related work in this area. Section 3 gives brief explanations of the Otsu's method, Kapur's entropy and Tsallis entropy. The detailed presentation of the basic KH and EKH algorithm is presented in Section 4. Subsequently, the experiments and results of KH and EKH algorithms for multilevel thresholding segmentation are shown in Section 5 to verify the efficiency of the method. Finally, some conclusions and suggestions for future research are represented in Section 6.

2. Related work

Image segmentation is a very important research topic in image processing. In the last few decades, numerous works have been carried out and a large number of methods have been presented. Among many algorithms, thresholding has been widely used due to its simplicity. The traditional thresholding segmentation approaches, such as between-class variance, Kapur's entropy, minimum cross entropy and Tsallis entropy, search exhaustively the optimal threshold values to optimize the objective functions. Although those thresholding methods can find the accurate threshold values, they are computationally expensive when extended to multilevel thresholding. The procedure of image segmentation is considered as a constrained optimization problem. Therefore, evolutionary and swarm intelligence algorithms are preferred to solve the multilevel thresholding problem. For example, the particle swarm optimization (PSO) has been used in many literatures for multilevel segmentation problem [23–25]. Hammouche et al. [66] applied genetic algorithm (GA), particle swarm optimization (PSO), differential evolution (DE), ant colony optimization (ACO), simulated annealing (SA) and tabu search (TS) algorithm to the multilevel thresholding problem. Additionally, there are many evolutionary and swarm intelligence algorithms for multilevel thresholding with artificial bee colony (ABC) algorithm [31–33], cuckoo search (CS) algorithm [6,7,20,34], firefly algorithm (FA) [34–36], wind driven optimization (WDO) [7]. Besides, in the past two years there are many recent algorithms are also used to solve multilevel segmentation problem. El Aziz et al. [67] proposed multilevel thresholding methods based on Whale Optimization Algorithm (WOA) and Moth-Flame Optimization (MFO) for image segmentation. A new method of multilevel thresholding for image segmentation using fireworks algorithm is presented [68]. Tuba et al. [69] applied recent elephant herding optimization algorithm for multilevel thresholding

Table 3
Comparison of optimal thresholds and objective values obtained by EKH and KH I algorithms using Otsu's method.

m	EKH				KH I			
	R	G	B	f ($\times 10^3$)	R	G	B	f ($\times 10^3$)
Baboon								
3	93,143,197	84,125,167	71,121,182	8.33180	93,143,197	84,125,167	71,121,182	8.33180
4	81,121,161,207	69,103,138,173	60,98,140,192	8.57374	81,121,161,207	69,103,138,173	60,98,140,192	8.57374
5	71,104,136,170,211	61,92,121,150,179	52,83,116,153,198	**8.70349**	72,102,134,169,210	63,95,124,152,179	51,85,114,154,198	8.70209
6	65,95,123,150,179,215	56,83,107,132,158,183	46,73,100,130,163,202	**8.77985**	63,90,117,144,174,212	52,77,111,135,160,186	50,76,107,135,167,205	8.77563
Donna								
3	55,107,167	61,112,177	63,124,178	10.32967	55,107,167	61,112,177	63,124,178	10.32967
4	47,87,136,183	56,100,139,193	53,93,137,182	10.56005	47,87,136,183	56,100,139,193	53,93,137,182	10.56005
5	46,82,125,167,207	49,83,113,144,196	52,91,134,173,210	**10.67661**	44,80,121,165,204	50,81,115,144,195	52,91,134,172,221	10.67510
6	44,74,109,142,175,211	48,79,107,130,153,199	43,71,103,140,174,211	**10.74667**	42,69,103,138171,209	46,74,101,125,147,197	42,67,98,134,170,208	10.74256
Fruit								
3	46,98,149	37,85,127	27,56,97	6.03146	46,98,149	37,85,127	27,56,97	6.03146
4	41,84,126,159	30,67,99,132	25,50,78,113	6.18859	41,84,126,159	30,67,99,132	25,50,78,113	6.18859
5	39,77,113,142,167	27,60,88,115,141	22,43,64,87,120	**6.26963**	39,80,120,148,172	27,63,92,117,142	21,40,62,91,121	6.26817
6	36,69,102,132,157,183	26,57,84,107,129,154	22,42,62,81,105,134	**6.31930**	33,67,95,126,153,179	27,63,88,112,131,156	25,48,66,83,109137	6.30507
Flower								
3	66,120,176	45,89,137	45,109,173	11.89602	66,120,176	45, 89,137	45,109,173	11.89602
4	64,109,156,189	42,79,116,150	37,89,145,185	12.06825	64,109,156,189	42,79,116,150	37,89,145,185	12.06825
5	56,84,123,163,192	36,62,92,124,154	33,75,124,162,193	**12.17086**	59,88,126,161,193	39,65,95,125,154	33,71,118,159,191	12.16930
6	50,72,98,133,167,194	35,59,86,113,137,162	25,50,89,132,166,195	**12.23024**	52,78,103,137,170,198	34,56,87,116,143,165	26,50,90,135,169,202	12.22572
House								
3	101,140,186	78,127,175	87,132,192	7.75031	101,140,186	78,127,175	87,132,192	7.75031
4	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823
5	85,111,140,163,191	52,85,128,174,213	81,99,126,165,202	**7.89108**	82,109,137,163,190	49,82,127,177,212	81,94,125,167,200	7.88961
6	79,101,119,143,164,192	49,78,102,132,176,213	77,92,105,131,168,203	**7.92699**	73,96,114,139,162,193	47,77,107,133,179,211	80,94,111,134,172,200	7.92287
Pallon								
3	93,148,187	50,91,140	34,79,127	7.09724	93,148,187	50,91,140	34,79,127	7.09724
4	90,138,167,194	41,72,104,145	22,54,88,130	7.24319	90,138,167,194	41,72,104,145	22,54,88,130	7.24319
5	89,134,161,185,207	39,69,99,137,175	21,53,86,126,167	**7.34898**	91,137,164,185,206	39,72,101,139,177	23,56,84,127,166	7.34758
6	87,126,147,167,185,205	37,65,90,114,144,178	19,48,78,106,135,168	**7.41616**	85,130,154,171,187,206	35,67,99,119,148,180	17,53,82,101,138,167	7.41207
Sailboat								
3	96,133,177	58,118,180	52,104,169	13.21162	96,133,177	58,118,180	52,104,169	13.21162
4	91,124,156,188	51,101,160,206	46,87,145,192	13.40812	91,124,156,188	51,101,160,206	46,87,145,192	13.40812
5	84,108,134,161,191	40,76,122,170,208	37,65,106,156,195	**13.52215**	83,107,138,161,189	40,79,125,174,209	39,68,110,159,196	13.52091
6	79,99,119,141,164,192	34,63,98,139,178,210	33,55,84,122,164,196	**13.58331**	83,105,123,145,165,190	36,70,103,142,181,210	35,59,91,125,168, 198	13.579.09
Tulips								
3	57,119,195	69,119,173	60,114,175	12.53545	57,119,195	69,119,173	60,114,175	12.53545
4	53,90,148,205	61,101,136,182	53,94,136,187	12.74659	53,90,148,205	61,101,136,182	53,94,136,187	12.74659
5	43,66,104,160,208	57,93,123,156,194	48,82,115,151,195	**12.86765**	44,66,102,154,205	59,93,126,157,197	46,80,118,154,197	12.86631
6	41,62,87,130,178,213	46,78,105,131,162,198	33,61,90,121,156,197	**12.94470**	44,65,96,137,182,216	52,86,112,138,168,202	37,65,96,125,161,201	12.94045
Voit								
3	94,135,182	72,105,143	87,126,170	4.27650	94,135,182	72,105,143	87,126,170	4.27650
4	83,112,145,187	71,102,137,177	75,100,132,172	4.43773	83,112,145,187	71,102,137,177	75,100,132,172	4.43773
5	80,105,132,159,196	66,91,116,145,180	69,89,110,138,174	**4.52374**	81,107,133,162,199	67,91,118,147,184	68,88,109,138,178	4.52201
6	76,97,117,139,163,198	66,91,116,144,179,254	66,85,103,125,147,176	**4.55294**	78,100,123,144,165,199	68,96,123,150,184,253	69,90,106,133,155,181	4.54883
Yacht								
3	62,114,164	59,110,158	52,102,157	9.62894	62,114,164	59,110,158	52,102,157	9.62894
4	56,97,136,176	50,86,125,166	45,85,120,163	9.83344	56,97,136,176	50,86,125,166	45,85,120,163	9.83344
5	47,80,110,144,178	38,66,93,128,167	43,79,111,141,174	**9.93049**	49,82,114,148,180	43,70,99,131,167	41,78,108,139,173	9.92900
6	46,78,105,133,160,185	32,58,86,114,145,174	28,55,86,115,144,176	**9.99804**	50,82,102,129,154,181	30,55,86,119,151,176	30,58,89,118,147,173	9.97957

segmentation by Kapur's and Otsu's method. Experimental results show that evolutionary and swarm intelligence algorithms can solve the problem of traditional thresholding segmentation methods and has better performances.

Color images have different color components and a variety of color intensity which makes segmentation very challenging. Therefore, color image multilevel segmentation problem based on evolutionary and swarm intelligence algorithms has got more attention in recent years, differential evolution (DE) [54,56], wind driven optimization (WDO) [54], particle swarm optimization (PSO) [54,56], cuckoo search (CS) [54], bacterial foraging algorithm (BFA) [55], ant colony optimization (ACO) [56] and firefly algorithm (FA) [57]. Further, in 2017, He and Huang [70] proposed a modified firefly algorithm (MFA) to search multilevel threshold values for color image. In 2018 bat algorithm (BA) [71] are used to search the optimal thresholding values based on Kapur's entropy, Tsallis entropy, Otsu's method, Shannon entropy, Renyi entropy. Kandhway et al. [72] applied a recently developed water cycle algorithm (WCA) for color image multilevel segmentation. In 2019, Liang et al. [73] proposed a modified grasshopper optimization algorithm (GOA) for color image multilevel segmentation. Xu et al. [74] presented a novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution.

Krill herd (KH) algorithm is a novel swarm intelligence algorithm that has better performance. However, it is easy to trap local optimization and cause premature convergence. To overcome these shortcomings, this paper propose an novel krill herd algorithm for color image multilevel, called efficient krill herd (EKH) algorithm, which is based on the a new linear decreasing inertia weights (ω_n, ω_f), an adaptive update scheme of C^{best} and Firefly algorithm. The EKH algorithm makes full advantages of novel linear decreasing inertia weights (ω_n, ω_f), C^{best} adaptive update scheme and a new position update method of krill individuals to enhance the krill individuals global search ability, the global convergence ability and avoid trapping into local optima. To prove the validity of the EKH algorithm, the algorithm is applied for color image multilevel thresholding segmentation based on Otsu's method, Kapur's entropy and Tsallis entropy. Firstly, the EKH algorithm is compared with KH without any genetic operators (KH I), KH with crossover operator (KH II) and KH with crossover and mutation operators (KH IV), are the EKH algorithm. Secondly, the four recent algorithms for color image multilevel thresholding segmentation, modified firefly algorithm (MFA) [70], bat algorithm (BA) [71], water cycle algorithm (WCA) [72] and modified grasshopper optimization algorithm (MGOA) [73], are compared with the EKH algorithm. Firefly algorithm is a new swarm intelligence algorithm inspired the flashing phenomenon

Table 4
Comparison of optimal thresholds and objective values obtained by KH II and KH IV algorithms using Otsu's method.

m	KH II				KH IV			
	R	G	B	f ($\times 10^3$)	R	G	B	f ($\times 10^3$)
Baboon								
3	93,143,197	84,125,167	71,121,182	8.33180	93,143,197	84,125,167	71,121,182	8.33180
4	81,121,161,207	69,103,138,173	60,98,140,192	8.57374	81,121,161,207	69,103,138,173	60,98,140,192	8.57374
5	70,103,136,170,210	61,92,120,149,180	52,82,115,152,198	8.70315	71,106,137,171,211	61,93,122,152,179	52,84,117,155,198	8.70314
6	63,97,124,148,174,215	55,82,107,129,156,182	45,70,98,129,160,201	8.77808	65,95,123,150,179,215	56,83,107,132,158,183	46,73,100,130,163,202	8.77805
Donna								
3	55,107,167	61,112,177	63,124,178	10.32967	55,107,167	61,112,177	63,124,178	10.32967
4	47,87,136,183	56,100,139,193	53,93,137,182	10.56005	47,87,136,183	56,100,139,193	53,93,137,182	10.56005
5	45,81,125,167,208	48,82,112,144,195	52,91,134,173,215	10.67621	47,83,125,169,207	49,84,114,145,197	53,93,135,173,210	10.67620
6	43,72,106,138,173,211	46,76,105,128,154,198	42,70,106,142,175,212	10.74495	44,77,112,145,179,212	49,83,109,135,156,199	43,75,106,142,178,212	10.74491
Fruit								
3	46,98,149	37,85,127	27,56,97	6.03146	46,98,149	37,85,127	27,56,97	6.03146
4	41,84,126,159	30,67,99,132	25,50,78,113	6.18859	41,84,126,159	30,67,99,132	25,50,78,113	6.18859
5	38,77,112,142,166	26,59,88,114,140	22,43,64,88,123	6.26931	39,77,113,142,167	27,60,88,115,141	22,43,64,87,120	6.26929
6	34,68,101,129,159,183	28,59,85,110,131,157	24,45,65,83,106,135	6.26871	37,73,105,136,159,184	27,59,88,109,132,155	23,46,64,85,109,135	6.26868
Flower								
3	66,120,176	45, 89,137	45,109,173	11.89602	66,120,176	45, 89,137	45,109,173	11.89602
4	64,109,156,189	42,79,116,150	37,89,145,185	12.06825	64,109,156,189	42,79,116,150	37,89,145,185	12.06825
5	57,85,124,163,192	36,61,90,122,153	34,76,124,162,193	12.17054	56,86,124,165192	37,63,94,125,154	33,78,124,163194	12.17053
6	53,77,108,144,172,196	31,54,81,109,134,160	24,50,90,134,167,196	12.22867	51,75,104,139,173,197	38,65,89,117,139,165	26,52,93,135,169,196	12.22863
House								
3	101,140,186	78,127,175	87,132,192	7.75031	101,140,186	78,127,175	87,132,192	7.75031
4	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823
5	86,111,139,163,191	51,84,129,174,214	80,99,126,166,204	7.89079	85,113,141,164,191	52,87,129,176,213	81,99,128,167,203	7.89078
6	83,104,123,144,163,192	47,78,105,133,174,214	75,91,104,134,171,201	7.92548	79,105,123,146,166,193	50,80,106,135,177,214	77,94,109,135,169,203	7.92545
Pallon								
3	93,148,187	50,91,140	34,79,127	7.09724	93,148,187	50,91,140	34,79,127	7.09724
4	90,138,167,194	41,72,104,145	22,54,88,130	7.24319	90,138,167,194	41,72,104,145	22,54,88,130	7.24319
5	89,134,160,184,207	39,69,99,138,176	22,54,86,126,166	7.34863	89,134,163,187,207	40,71,99,138,175	23,55,87,126,167	7.34862
6	86,129,150,168,186,205	36,62,93,116,145,179	18,49,80,107,136,166	7.41458	89,129,153,170187,205	38,69,94,116,147,179	21,51,80,110,139,169	7.41455
Sailboat								
3	96,133,177	58,118,180	52,104,169	13.21162	96,133,177	58,118,180	52,104,169	13.21162
4	91,124,156,188	51,101,160,206	46,87,145,192	13.40812	91,124,156,188	51,101,160,206	46,87,145,192	13.40812
5	83,107,133,161,191	40,75,121,170,208	37,65,107,158,195	13.52184	85,109,136,162,191	40,78,124,170,208	37,67,107,158,195	13.52183
6	79,102,121,144,164,192	33,60,95,137,176,209	33,58,86,125,165,196	13.58188	81,103,124,143,165,192	35,66,99,145,182,211	35,57,88,125,168,197	13.58185
Tulips								
3	57,119,195	69,119,173	60,114,175	12.53545	57,119,195	69,119,173	60,114,175	12.53545
4	53,90,148,205	61,101,136,182	53,94,136,187	12.74659	53,90,148,205	61,101,136,182	53,94,136,187	12.74659
5	43,67,105,161,208	57,93,123,156,196	47,82,113,152,196	12.86731	43,68,106,161,208	57,95,124,157,194	49,83,117,152,195	12.86730
6	42,64,93,131,183,214	47,83,108,134,163,199	34,63,92,124,156,196	12.94319	43,66,89,135,182,213	48,82,109,135,164,198	34,66,92,125,157,197	12.94315
Voit								
3	94,135,182	72,105,143	87,126,170	4.27650	94,135,182	72,105,143	87,126,170	4.27650
4	83,112,145,187	71,102,137,177	75,100,132,172	4.43773	83,112,145,187	71,102,137,177	75,100,132,172	4.43773
5	80,105,131,159,195	66,91,116,144,179	70,90,110,138,174	4.52343	80,105,134,160,197	66,93,117,146,180	69,90,111,139,175	4.52342
6	76,98,118,138,163,195	65,94,119,146,181,254	65,82,99,120,145,176	4.55144	77,99,119142,166,198	66,95,118,147,182,254	67,89,106,126,150,177	4.55141
Yacht								
3	62,114,164	59,110,158	52,102,157	9.62894	62,114,164	59,110,158	52,102,157	9.62894
4	56,97,136,176	50,86,125,166	45,85,120,163	9.83344	56,97,136,176	50,86,125,166	45,85,120,163	9.83344
5	46,79,110,144,178	38,66,93,129,168	43,79,110,139,173	9.93015	47,83,111,144,178	38,67,95,129,167	43,79,113,143,175	9.93014
6	47,76,104,134,161,184	33,59,87,116,148,174	29,59,87,117,145,177	9.99619	48,79,109,136,164,185	352,60,89,117,147,175	29,57,89,118,146,179	9.99615

of fireflies that better performance. In order to improve the performance of FA algorithm, a modified firefly algorithm (MFA) is presented [70]. Bat algorithm (BA) is a novel population-based evolutionary algorithm inspired by the behavior of the echolocation. Due to its simple concept, BA has been widely applied to various engineering applications. Water cycle algorithm (WCA) is a new metaheuristic technique that models the water flow cycle from streams to rivers and from rivers to sea. In 2017, Mirjalili Seyedali et al. [75] proposed the grasshopper optimization algorithm (GOA) inspired by grasshopper swarms. In order to balance the exploration and exploitation of GOA, the Levy flight is used to improve GOA algorithm and proposed modified grasshopper optimization algorithm (MGOA) [73]. Although these algorithms have their own advantages, No-Free-Lunch has proved that no algorithm can solve all optimization problems, so there is no perfect optimization algorithm [76]. In order to further balance the exploration and exploitation and improve the performance of the algorithm, an efficient krill herd algorithm (EKH) is proposed in this paper.

3. Problem assessment of multilevel thresholding

Thresholding is a very useful method for image segmentation. This method classifies the pixels of gray image or colored image into different classes or sets. For bi-level thresholding, the optimal threshold value (th) should be searched in a manner that follows Eq. (1).

$$\begin{array}{llll} C_1 \leftarrow p & if & 0 \leq p < th \\ C_2 \leftarrow p & if & th \leq p < L-1 \end{array} \tag{1}$$

where p is one of the pixels belonging to image I of size $m \times n$. C_1 and C_2 indicate the two different classes which are separated by pixel p, th represents the threshold value.

The bi-level thresholding problem can be extended to multilevel thresholding by following:

$$\begin{array}{llll} C_1 \leftarrow p & if & 0 \leq p < th_1 \\ C_2 \leftarrow p & if & th_1 \leq p < th_2 \\ C_i \leftarrow p & if & th_i \leq p < th_{i+1} \\ C_n \leftarrow p & if & th_n \leq p < L-1 \end{array} \tag{2}$$

where, $th_1, th_2, \ldots, th_k$ are various threshold values. The key of bi-level and multilevel thresholding is to select the th values that appropriately partition an image into different segment. In this paper, the three most widely used objective functions Otsu's method, Kapur's entropy and Tsallis entropy have been considered to find the threshold values.

Table 5
Comparison of optimal thresholds and objective values obtained by MFA and MGOA algorithms using Otsu's method.

m	MFA				MGOA			
	R	G	B	f ($\times 10^3$)	R	G	B	f ($\times 10^3$)
Baboon								
3	93,143,197	84,125,167	71,121,182	8.33180	93,143,197	84,125,167	71,121,182	8.33180
4	81,121,161,207	69,103,138,173	60,98,140,192	8.57374	81,121,161,207	69,103,138,173	60,98,140,192	8.57374
5	71,104,136,170,211	61,92,121,150,179	52,83,116,153,198	8.70349	71,105,136,170,211	61,92,122,150,179	52,85,116,153,198	8.70340
6	65,96,123,150,179,215	56,83,107,132,158,183	46,73,100,130,163,202	8.77976	66,96,124,150,179,215	56,85,108,132,158,183	46,73,100,131,163,203	8.77945
Donna								
3	55,107,167	61,112,177	63,124,178	10.32967	55,107,167	61,112,177	63,124,178	10.32967
4	47,87,136,183	56,100,139,193	53,93,137,182	10.56005	47,87,136,183	56,100,139,193	53,93,137,182	10.56005
5	46,82,125,167,207	49,83,113,144,196	52,91,134,173,210	10.67661	47,82,125,167,207	49,83,113,146,196	52,91,134,173,211	10.67652
6	44,74,109,142,175,211	48,80,107,130,153,199	43,71,103,140,174,211	10.74656	46,74,109,144,175,211	48,80,107,131,153,200	43,73,103,140,175,211	10.74626
Fruit								
3	46,98,149	37,85,127	27,56,97	6.03146	46,98,149	37,85,127	27,56,97	6.03146
4	41,84,126,159	30,67,99,132	25,50,78,113	6.18859	41,84,126,159	30,67,99,132	25,50,78,113	6.18859
5	39,77,113,142,167	27,60,88,115,141	22,43,64,87,120	6.26963	41,77,113,142,167	27,61,88,115,141	22,43,64,88,120	**6.26952**
6	38,69,102,132,157,183	26,57,84,107,129,154	22,42,62,81,105,134	6.31918	37,69,103,132,158,183	27,59,85,107,129,154	22,43,62,82,106,134	6.31888
Flower								
3	66,120,176	45,89,137	45,109,173	11.89602	66,120,176	45, 89,137	45,109,173	11.89602
4	64,109,156,189	42,79,116,150	37,89,145,185	12.06825	64,109,156,189	42,79,116,150	37,89,145,185	12.06825
5	56,84,123,163,192	36,62,92,124,154	33,75,124,162,193	12.17086	56,84,124,163,192	36,63,92,124,154	34,75,124,162,193	12.17077
6	50,72,98,133,167,194	35,59,86,113,137,162	26,50,89,132,166,195	12.23015	52,72,99,133,167,194	36,60,87,113,137,162	26,51,90,132,166,195	12.22983
House								
3	101,140,186	78,127,175	87,132,192	7.75031	101,140,186	78,127,175	87,132,192	7.75031
4	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823
5	85,111,140,163, 191	52,85,128,174,213	81,99,126,165,202	7.89108	85,111,140,163, 192	52,86,128,174,213	81,99,127,165,202	7.89098
6	79,101,119,143,164,193	49,78,102,132,176,213	77,92,105,131,168,203	7.92688	79,102,120,143,165,192	49,78,102,132,178,214	77,93,106,131,168,204	**7.92659**
Pallon								
3	93,148,187	50,91,140	34,79,127	7.09724	93,148,187	50,91,140	34,79,127	7.09724
4	90,138,167,194	41,72,104,145	22,54,88,130	7.24319	90,138,167,194	41,72,104,145	22,54,88,130	7.24319
5	89,134,161,185, 207	39,69,99,137,175	21,53,86,126,167	7.34898	89,134,161,187, 207	40,69,99,137,175	22,53,86,126,167	7.34888
6	88,126,147,167,185,205	37,65,90,114,144,178	19,48,78,106,135,168	7.41596	89,127,148,167,185,205	38,66,90,115,144,178	19,50,79,106,136,168	7.41584
Sailboat								
3	96,133,177	58,118,180	52,104,169	13.21162	96,133,177	58,118,180	52,104,169	13.21162
4	91,124,156,188	51,101,160,206	46,87,145,192	13.40812	91,124,156,188	51,101,160,206	46,87,145,192	13.40812
5	84,108,134,161,191	40,76,122,170,208	37,65,106,156,195	13.52215	85,108,134,161,191	40,76,122,171,208	37,65,106,156,196	13.52205
6	79,99,119,141,164,193	34,63,98,139,178,210	33,55,84,122,164,196	13.58322	80,99,119,142,165,192	36,63,99,139,179,210	34,55,84,124,164,197	13.58290
Tulips								
3	57,119,195	69,119,173	60,114,175	12.53545	57,119,195	69,119,173	60,114,175	12.53545
4	53,90,148,205	61,101,136,182	53,94,136,187	12.74659	53,90,148,205	61,101,136,182	53,94,136,187	12.74659
5	43,66,104,160,208	57,93,123,156,194	48,82,115,151,195	12.86765	45,66,104,160,208	57,94,123,156,194	49,82,115,151,195	12.86754
6	41,62,87,130,178,213	46,78,105,131,162,200	33,61,90,121,156,197	12.94457	41,62,88,130,179,214	47,78,106,131,163,198	33,61,91,121,156,199	12.94429
Voit								
3	94,135,182	72,105,143	87,126,170	4.27650	94,135,182	72,105,143	87,126,170	4.27650
4	83,112,145,187	71,102,137,177	75,100,132,172	4.43773	83,112,145,187	71,102,137,177	75,100,132,172	4.43773
5	80,105,132,159,196	66,91,116,145,180	69,89,110,138,174	4.52374	80,105,133,159,196	66,91,117,145,180	70,89,110,138,174	4.52365
6	76,97,117,139,163,198	67,91,116,144,179,254	66,85,103,125,147,176	4.55285	78,97,117,141,163,198	66,92,116,145,180,254	66,85,104,126,148,176	4.55252
Yacht								
3	62,114,164	59,110,158	52,102,157	9.62894	62,114,164	59,110,158	52,102,157	9.62894
4	56,97,136,176	50,86,125,166	45,85,120,163	9.83344	56,97,136,176	50,86,125,166	45,85,120,163	9.83344
5	47,80,110,144,178	38,66,93,128,167	43,79,111,141,174	9.93049	47,80,110,145,178	38,67,93,128,167	43,79,112,141,174	9.93040
6	46,78,105,133,160,186	32,58,86,114,145,174	28,55,86,115,144,176	9.99793	48,79,105,133,160,185	32,59,87,114,146,174	29,56,86,115,144,177	9.99763

3.1. Between-class variance method (Otsu's method)

In the Otsu's method, the optimal threshold values are searched by maximizing the between-class variance of the different parts.

Assume an image has N pixels and can be represented in L gray levels, and the number of pixels with gray level i is presented by f_i. The occurrence probability of gray level i is defined as

$$p_i^c = \frac{f_i^c}{N}, p_i^c \geq 0, \sum_{i=0}^{L-1} p_i^c = 1, \tag{3}$$

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

where i is a gray level between the range 0 to $L-1$, c represents the image component. RGB color image has three separate image component that are red, green and blue.

For bi-level thresholding, the original image is divided into two classed by the optimal threshold t^{*c}, and the between-class variance of two classes is defined by Eq. (4):

$$f^c(t) = \sigma_0^c + \sigma_1^c \tag{4}$$

$$\sigma_0^c = \omega_0^c \left(\mu_0^c - \mu_T^c\right)^2, \qquad \sigma_1^c = \omega_1^c \left(\mu_1^c - \mu_T^c\right)^2 \tag{5}$$

where

$$\mu_T^c = \sum_{i=0}^{L-1} i p_i^c$$

and μ_T^c is the mean levels of whole image.

The mean levels of two classes can be obtained using Eq. (6):

$$\mu_0^c = \sum_{i=0}^{t-1} \frac{i p_i^c}{\omega_0^c}, \qquad \mu_1^c = \sum_{i=t}^{L-1} \frac{i p_i^c}{\omega_1^c} \tag{6}$$

The cumulative probabilities of two classes can be presented as follows:

$$\omega_0^c = \sum_{i=0}^{t-1} p_i^c, \qquad \omega_1^c = \sum_{i=t}^{L-1} p_i^c \tag{7}$$

The Otsu's method searches the optimal threshold t^{*c} by maximizing the between-class variance:

$$t^{*c} = \arg\max(\sigma_0^c + \sigma_1^c) \tag{8}$$

For Otsu's method, the extension of the original method to multi-level thresholding is referred to as the Multi Otsu method. Assume an image is divided into m classes and has $m-1$ thresholding values, the extended between-class variance function can

Table 6
Comparison of optimal thresholds and objective values obtained by WCA and BA algorithms using Otsu's method.

m	WCA				BA			
	R	G	B	f ($\times 10^3$)	R	G	B	f ($\times 10^3$)
Baboon								
3	93,143,197	84,125,167	71,121,182	8.33180	93,143,197	84,125,167	71,121,182	8.33180
4	81,121,161,207	69,103,138,173	60,98,140,192	8.57374	81,121,161,207	69,103,138,173	60,98,140,192	8.57374
5	72,104,137,170,211	63,92,121,150,179	52,83,117,153,199	8.70332	70,103,136,170,210	62,92,123,150,179	53,84,116,154,198	8.70323
6	66,95,125,151,179,215	57,83,108,134,158,183	47,74,101,130,163,203	8.77844	67,95,125,151,179,215	57,85,108,132,158,184	46,75,101,130,163,204	8.77813
Donna								
3	55,107,167	61,112,177	63,124,178	10.32967	55,107,167	61,112,177	63,124,178	10.32967
4	47,87,136,183	56,100,139,193	53,93,137,182	10.56005	47,87,136,183	56,100,139,193	53,93,137,182	10.56005
5	46,83,125,168,207	50,83,113,145,196	52,91,134,174,211	10.67646	48,82,126,167,207	49,84,114,145,196	53,91,135,173,211	10.67636
6	45,74,111,143,175,211	49,80,108,130,154,199	43,72,105,140,175,211	10.74595	45,75,111,142,175,212	50,79,109,131,153,199	44,72,104,141,174,211	10.74564
Fruit								
3	46,98,149	37,85,127	27,56,97	6.03146	46,98,149	37,85,127	27,56,97	6.03146
4	41,84,126,159	30,67,99,132	25,50,78,113	6.18859	41,84,126,159	30,67,99,132	25,50,78,113	6.18859
5	40,77,113,143,167	27,61,89,115,141	22,43,64,87,122	6.26945	38,77,112,142,166	29,60,88,116,141	23,43,65,87,121	6.26937
6	37,71,102,133,157,183	26,58,85,108,130,154	22,43,62,81,107,135	6.31858	38,70,102,134,157,183	26,57,86,107,130,155	23,43,63,81,107,134	6.31827
Flower								
3	66,120,176	45,89,137	45,109,173	11.89602	66,120,176	45, 89,137	45,109,173	11.89602
4	64,109,156,189	42,79,116,150	37,89,145,185	12.06825	64,109,156,189	42,79,116,150	37,89,145,185	12.06825
5	57,84,123,163,193	36,63,92,125,154	33,75,124,163,194	12.17068	58,84,123,164,192	37,63,92,125,154	33,77,125,162,193	12.17061
6	51,72,99,135,167,194	35,60,87,114,137,163	25,50,90,133,166,197	12.22952	48,71,97,133,166,194	35,61,86,114,138,163	27,51,90,132,167,195	12.22921
House								
3	101,140,186	78,127,175	87,132,192	7.75031	101,140,186	78,127,175	87,132,192	7.75031
4	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823	100,136,163,191	52, 85,128,176	82,102,140,195	7.82823
5	85,112,141,163,191	52,85,129,174,215	81,99,127,165,202	7.89089	86,112,140,164,191	52,85,129,175,214	81,99,128,165,203	7.89082
6	80,103,120,143,164,192	50,78,103,132,177,213	77,92,106,131,169,205	7.92625	81,102,120,143,165,192	49,80,102,134,177,213	79,93,106,131,169,203	7.92595
Pallon								
3	93,148,187	50,91,140	34,79,127	7.09724	93,148,187	50,91,140	34,79,127	7.09724
4	90,138,167,194	41,72,104,145	22,54,88,130	7.24319	90,138,167,194	41,72,104,145	22,54,88,130	7.24319
5	89,135,162,185,207	39,69,99,137,177	22,54,86,126,167	7.34880	90,135,161,186,207	39,70,99,138,176	21,53,88,127,167	7.34873
6	88,127,148,167,186,205	37,67,91,114,145,178	20,48,79,106,135,170	7.41524	87,128,148,167,187,205	39,67,90,115,144,179	19,50,80,106,135,169	7.41513
Sailboat								
3	96,133,177	58,118,180	52,104,169	13.21162	96,133,177	58,118,180	52,104,169	13.21162
4	91,124,156,188	51,101,160,206	46,87,145,192	13.40812	91,124,156,188	51,101,160,206	46,87,145,192	13.40812
5	85,109,134,161,191	40,76,124,170,208	37,66,106,157,195	13.52197	85,109,135,161,191	40,77,124,170,208	38,65,107,156,196	13.52189
6	79,100,121,141,165,192	35,64,98,139,178,212	34,56,85,122,164,197	13.58255	81,99,121,141,165,192	35,64,99,139,178,211	35,56,84,123,165,196	13.58227
Tulips								
3	57,119,195	69,119,173	60,114,175	12.53545	57,119,195	69,119,173	60,114,175	12.53545
4	53,90,148,205	61,101,136,182	53,94,136,187	12.74659	53,90,148,205	61,101,136,182	53,94,136,187	12.74659
5	43,67,105,160,208	57,93,123,158,194	48,82,116,152,195	12.86747	44,67,104,161,208	57,93,125,156,195	50,82,115,152,195	12.86740
6	43,62,86,130,177,213	48,78,106,131,163,198	34,61,92,122,156,197	12.94397	43,63,87,131,178,214	47,80,105,133,162,198	34,62,91,121,158,197	12.94365
Voit								
3	94,135,182	72,105,143	87,126,170	4.27650	94,135,182	72,105,143	87,126,170	4.27650
4	83,112,145,187	71,102,137,177	75,100,132,172	4.43773	83,112,145,187	71,102,137,177	75,100,132,172	4.43773
5	80,106,132,160,196	66,91,117,145,181	69,90,111,138,174	4.52356	82,105,132,160,196	66,92,117,146,180	71,89,110,138,175	4.52348
6	77,99,117,140,163,198	67,91,117,146,179,254	66,86,105,125,147,177	4.55221	78,97,119,140,163,198	67,92,117,146,179,254	68,86,103,125,148,177	4.55192
Yacht								
3	62,114,164	59,110,158	52,102,157	9.62894	62,114,164	59,110,158	52,102,157	9.62894
4	56,97,136,176	50,86,125,166	45,85,120,163	9.83344	56,97,136,176	50,86,125,166	45,85,120,163	9.83344
5	48,81,110,144,178	38,66,93,129,168	45,79,111,141,174	9.93037	48,81,110,145,178	38,68,93,129,167	44,79,112,141,175	9.93023
6	47,78,107,134,160,185	32,59,87,114,146,175	29,56,87,115,145,176	9.99731	48,78,107,134,160,185	32,59,87,115,145,176	28,56,87,117,144,178	9.99702

be described by (9):

$$f^c(t) = \sum_{i=0}^{m-1} \sigma_i^c \tag{9}$$

The optimal thresholding values $\left(t_1^{*c}, t_2^{*c}, \ldots, t_{m-1}^{*c}\right)$ are calculated by maximizing σ_B^c as follows:

$$\left(t_1^{*c}, t_2^{*c}, \ldots, t_{m-1}^{*c}\right) = \underset{0 \le t_1^c \le \cdots t_{m-1}^c \le L-1}{\arg} \max \left\{\sigma_B^c \left(t_1^c, t_2^c, \ldots, t_{m-1}^c\right)\right\} \tag{10}$$

where $\sigma_B^c = \sigma_0^c + \sigma_1^c + \cdots + \sigma_{m-1}^c$.

The sigma terms are as follows:

$$\sigma_0^c = \omega_0^c(\mu_0^c - \mu_T^c)^2, \sigma_1^c = \omega_1^c(\mu_1^c - \mu_T^c)^2, \ldots, \sigma_{m-1}^c = \omega_{m-1}^c(\mu_{m-1}^c - \mu_T^c)^2 \tag{11}$$

The mean levels of m classes are given by

$$\mu_0^c = \sum_{i=0}^{t_1-1} ip_i^c/\omega_0^c, \mu_1^c = \sum_{i=t_1}^{t_2-1} ip_i^c/\omega_1^c, \ldots, \mu_{m-1}^c = \sum_{i=t_{m-1}}^{L-1} ip_i^c/\omega_{m-1}^c \tag{12}$$

Similar to bi-level segmentation, the c value of RGB image and gray scale image is $c = 1, 2, 3$ and $c = 1$ respectively for multi Otsu segmentation.

3.2. Kapur's entropy method

The Kapur's entropy criterion is based on the probability distribution of the gray level histogram. The method can search optimal thresholding values when the entropy is maximizing. It was firstly employed in determining the optimal threshold values for bi-level thresholding. Let there be L gray levels and N pixels in a given image, then the Kapur's entropy of two classes may be described by (13):

$$f(t) = H_0^c + H_1^c, \qquad c = \begin{cases} 1, 2, 3 & if \quad RGB\ image \\ 1 & if \quad gray\ scale\ image \end{cases} \tag{13}$$

where, H_0 and H_1 are calculated using the following formula:

$$\begin{aligned} H_0^c &= -\sum_{i=0}^{t-1} \frac{p_i^c}{\omega_0^c} \ln \frac{p_i^c}{\omega_0^c}, \qquad \omega_0^c = \sum_{i=0}^{t-1} p_i^c, \\ H_1^c &= -\sum_{i=t}^{L-1} \frac{p_i^c}{\omega_c^1} \ln \frac{p_i^c}{\omega_c^1}, \qquad \omega_c^1 = \sum_{i=t}^{L-1} p_i^c \end{aligned} \tag{14}$$

Table 7
Comparison of PSNR and SSIM obtained by EKH, KH I, KH II and KH IV algorithms using Otsu's method.

Image	m	EKH		KH I		KH II		KH IV	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Baboon	3	16.80716	0.84245	16.80716	0.84245	16.80716	0.84245	16.80716	0.84245
	4	16.80836	0.84945	16.80836	0.84945	16.80836	0.84945	16.80836	0.84945
	5	16.80992	0.90455	16.80809	0.90286	16.80925	0.90402	16.80892	0.90365
	6	16.81193	0.91001	16.80822	0.90722	16.80994	0.90876	16.80943	0.90798
Donna	3	20.16255	0.86492	20.16255	0.86492	20.16255	0.86492	20.16255	0.86492
	4	20.16399	0.88252	20.16399	0.88252	20.16399	0.88252	20.16399	0.88252
	5	20.15689	0.91602	20.15458	0.93411	20.15635	0.93558	20.15616	0.93509
	6	20.15917	0.92516	20.15543	0.92249	20.15805	0.92401	20.1511	0.92375
Fruit	3	21.74544	0.68011	21.74544	0.68011	21.74544	0.68011	21.74544	0.68011
	4	21.74662	0.70181	21.74662	0.70181	21.74662	0.70181	21.74662	0.70181
	5	21.74844	0.77284	21.77695	0.76985	21.77803	0.77235	21.77803	0.77235
	6	21.75992	0.82004	21.75602	0.81764	21.75885	0.81901	21.75861	0.81870
Flower	3	18.65565	0.84662	18.65565	0.84662	18.65565	0.84662	18.65516	0.84587
	4	18.65683	0.87125	18.65683	0.87125	18.65683	0.87125	18.65683	0.87125
	5	18.65856	0.90225	18.65636	0.89973	18.65807	0.90180	18.65786	0.90154
	6	18.66059	0.91149	18.65682	0.90863	18.65875	0.91031	18.65822	0.90963
House	3	15.37113	0.85188	15.37113	0.85188	15.37113	0.85188	15.37113	0.85188
	4	15.37236	0.87494	15.37236	0.87494	15.37236	0.87494	15.37236	0.87494
	5	15.37418	0.88916	15.37208	0.88773	15.37334	0.88878	15.37346	0.88843
	6	15.37624	0.89615	15.37251	0.89318	15.37412	0.89474	15.37385	0.89407
Pallon	3	18.35243	0.78392	18.35243	0.78392	18.35243	0.78392	18.35243	0.78392
	4	18.35427	0.78486	18.35427	0.78486	18.35427	0.78486	18.35427	0.78486
	5	18.35613	0.87235	18.35198	0.87066	18.35240	0.87188	18.35218	0.87135
	6	18.35870	0.88157	18.35430	0.87998	18.35446	0.88096	18.35406	0.88026
Sailboat	3	16.46503	0.85674	16.46503	0.85674	16.46503	0.85674	16.46503	0.85674
	4	16.46616	0.87891	16.46616	0.87891	16.46616	0.87891	16.46616	0.87891
	5	16.46777	0.91276	16.46554	0.91099	16.46702	0.91218	16.46713	0.91184
	6	16.47009	0.92051	16.46605	0.91788	16.46815	0.91900	16.46776	0.91833
Tulips	3	17.44694	0.90046	17.44694	0.90046	17.44694	0.90046	17.44694	0.90046
	4	17.44851	0.90286	17.44851	0.90286	17.44851	0.90286	17.44851	0.90286
	5	17.44985	0.94209	17.44752	0.93975	17.44908	0.94179	17.44901	0.94145
	6	17.45132	0.94901	17.44757	0.94632	17.44923	0.94781	17.44930	0.94709
Voit	3	18.97092	0.72260	18.97092	0.72260	18.97092	0.72260	18.97092	0.72260
	4	18.97224	0.74391	18.97224	0.74391	18.97224	0.74391	18.97224	0.74391
	5	18.97389	0.81435	18.97200	0.81207	18.97368	0.81378	18.97342	0.81337
	6	18.97579	0.82671	18.97181	0.82419	18.97372	0.82556	18.97330	0.82485
Yacht	3	18.49263	0.85616	18.49263	0.85616	18.49263	0.85616	18.49263	0.85616
	4	18.49398	0.86673	18.49398	0.86673	18.49398	0.86673	18.49398	0.86673
	5	18.49588	0.88987	18.49376	0.88827	18.49526	0.88934	18.49500	0.88901
	6	18.49776	0.90049	18.49382	0.89772	18.49589	0.89918	18.49537	0.89836

where H_0^c and H_1^c is the Kapur's entropy of class 0 and class 1 respectively, p_i^c represents the probability distribution of the intensity levels. $\omega_0^c(th)$ and $\omega_1^c(th)$ are the probability distribution for class 0 and class 1.

The probability distribution p_i^c is defined as

$$p_i^c = \frac{h_i^c}{\sum_{i=0}^{L-1} h^c(i)}, \qquad i = 0, \dots, L-1 \tag{15}$$

The optimal threshold value t^{*c} can be found by maximizing (16):

$$t^{*c} = \arg\max(H_0^c + H_1^c) \tag{16}$$

Further, the Kapur's entropy can also be easily extended for multilevel thresholding problem using the Eq. (17)

$$f(t) = \sum_{i=0}^{m-1} H_i^c, \qquad c = \begin{cases} 1,2,3 & \textit{if} \quad \textit{RGB image} \\ 1 & \textit{if} \quad \textit{gray scale image} \end{cases} \tag{17}$$

In Eq. (17), the image is divided into m classes by $m-1$ thresholding values. The Kapur's entropy can be easily extended to multilevel image thresholding and is formulated as

$$\begin{aligned} H_0^c &= -\sum_{i=0}^{t_1-1} \frac{p_i^c}{\omega_0^c} \ln \frac{p_i^c}{\omega_0^c}, \qquad \omega_0^c = \sum_{i=0}^{t_1-1} p_i^c \\ H_1^c &= -\sum_{i=t_1}^{t_2-1} \frac{p_i^c}{\omega_1^c} \ln \frac{p_i^c}{\omega_1^c}, \qquad \omega_1^c = \sum_{i=t_1}^{t_2-1} p_i^c \\ H_j^c &= -\sum_{i=t_j}^{t_{j+1}-1} \frac{p_i^c}{\omega_j^c} \ln \frac{p_i^c}{\omega_j^c}, \qquad \omega_j^c = \sum_{i=t_j}^{t_{j+1}-1} p_i^c \\ H_{m-1}^c &= -\sum_{i=t_{m-1}}^{L-1} \frac{p_i^c}{\omega_{m-1}^c} \ln \frac{p_i^c}{\omega_{m-1}^c}, \qquad \omega_{m-1}^c = \sum_{i=t_{m-1}}^{L-1} p_i^c \end{aligned} \tag{18}$$

The optimal multilevel thresholding problem is considered as a multi-dimension optimization problem. In order to find $m-1$ optimal threshold values $[t_1, t_2, \dots, t_{m-1}]$ for a given image, the

Table 8
Comparison of PSNR and SSIM obtained by MFA, MGOA, WCA and BA algorithms using Otsu's method.

Image	m	MFA		MGOA		WCA		BA	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Baboon	3	16.80716	0.84245	16.80716	0.84245	16.80716	0.84245	16.80716	0.84245
	4	16.80836	0.84945	16.80836	0.84945	16.80836	0.84945	16.80836	0.84945
	5	16.80992	0.90455	16.80973	0.90439	16.80956	0.90422	16.80939	0.90405
	6	16.81183	0.90994	16.81157	0.90971	16.81123	0.90939	16.81091	0.90906
Donna	3	20.16255	0.86492	20.16255	0.86492	20.16255	0.86492	20.16255	0.86492
	4	20.16399	0.88252	20.16399	0.88252	20.16399	0.88252	20.16399	0.88252
	5	20.15689	0.91602	20.15671	0.91585	20.15655	0.91571	20.15637	0.91551
	6	20.15908	0.92510	20.15881	0.92487	20.15848	0.92454	20.15812	0.92420
Fruit	3	21.74544	0.68011	21.74544	0.68011	21.74544	0.68011	21.74544	0.68011
	4	21.74662	0.70181	21.74662	0.70181	21.74662	0.70181	21.74662	0.70181
	5	21.74844	0.77284	21.74825	0.77267	21.74807	0.77251	21.74791	0.77235
	6	21.75988	0.81996	21.75954	0.81974	21.75921	0.81941	21.75886	0.81908
Flower	3	18.65565	0.84662	18.65565	0.84662	18.65565	0.84662	18.65516	0.84587
	4	18.65683	0.87125	18.65683	0.87125	18.65683	0.87125	18.65683	0.87125
	5	18.65856	0.90225	18.65837	0.90208	18.65821	0.90191	18.65805	0.90177
	6	18.66051	0.91144	18.66022	0.91118	18.65989	0.91086	18.65956	0.91053
House	3	15.37113	0.85188	15.37113	0.85188	15.37113	0.85188	15.37113	0.85188
	4	15.37236	0.87494	15.37236	0.87494	15.37236	0.87494	15.37236	0.87494
	5	15.37418	0.88916	15.37401	0.88900	15.37381	0.88882	15.37364	0.88864
	6	15.37615	0.89610	15.37586	0.89585	15.37556	0.89552	15.37517	0.89518
Pallon	3	18.35243	0.78392	18.35243	0.78392	18.35243	0.78392	18.35243	0.78392
	4	18.35427	0.78486	18.35427	0.78486	18.35427	0.78486	18.35427	0.78486
	5	18.35613	0.87235	18.35594	0.87217	18.35577	0.87204	18.35561	0.87186
	6	18.35862	0.88151	18.35832	0.88127	18.35798	0.88096	18.35765	0.88061
Sailboat	3	16.46503	0.85674	16.46503	0.85674	16.46503	0.85674	16.46503	0.85674
	4	16.46616	0.87891	16.46616	0.87891	16.46616	0.87891	16.46616	0.87891
	5	16.46777	0.91276	16.46758	0.91260	16.46741	0.91242	16.46752	0.91227
	6	16.47001	0.92044	16.46972	0.92021	16.46938	0.91989	16.46901	0.91954
Tulips	3	17.44694	0.90046	17.44694	0.90046	17.44694	0.90046	17.44694	0.90046
	4	17.44851	0.90286	17.44851	0.90286	17.44851	0.90286	17.44851	0.90286
	5	17.44985	0.94209	17.44968	0.94192	17.44948	0.94175	17.44932	0.94160
	6	17.45125	0.94895	17.45096	0.94869	17.45065	0.94839	17.45025	0.94803
Voit	3	18.97092	0.72260	18.97092	0.72260	18.97092	0.72260	18.97092	0.72260
	4	18.97224	0.74391	18.97224	0.74391	18.97224	0.74391	18.97224	0.74391
	5	18.97389	0.81435	18.97371	0.81420	18.97355	0.81401	18.97342	0.81383
	6	18.97572	0.82664	18.97544	0.82642	18.97510	0.82608	18.97481	0.82575
Yacht	3	18.49263	0.85616	18.49263	0.85616	18.49263	0.85616	18.49263	0.85616
	4	18.49398	0.86673	18.49398	0.86673	18.49398	0.86673	18.49398	0.86673
	5	18.49588	0.88987	18.49571	0.88969	18.49552	0.88954	18.49531	0.88936
	6	18.49770	0.90045	18.49741	0.90017	18.49708	0.89986	18.49668	0.99950

objective function is maximized by the Eq. (19):

$$\left(t_1^{*c}, t_2^{*c}, \dots, t_{m-1}^{*c}\right) = \arg\max\Big(\sum_{i=0}^{m-1} H_i^c\Big) \tag{19}$$

3.3. Tsallis entropy

In statistical thermodynamics, physically entropy is associated with the measure of disorder in a system. But Shannon redefined the entropy concept of Boltzmann as a measure of uncertainty regarding the information content of a system. The Shannon entropy is described by (20):

$$S = -\sum_{i=1}^{k} p_i \ln(p_i) \tag{20}$$

Shannon entropy has the extensive property (additivity):

$$S(A+B) = S(A) + S(B) \tag{21}$$

Based on multi-fractal theory, Tsallis has proposed a generalization of BGS statistics, and the expression can be described as:

$$S_q = \frac{1 - \sum_{i=1}^{k} (p_i)^q}{q-1} \tag{22}$$

where k is the total number of possibilities of the system, q represents an entropic index that characterizes the degree of non-extensivity, p_i ranges from 0 to 1 which denotes the probability of the modeled system to be in state i.

Tsallis entropy of the system can be calculated by a pseudo additivity entropic rule using Eq. (23):

$$S_q(A^c + B^c) = S_q(A^c) + S_q(B^c) + (1-q).S_q(A^c).S_q(B^c)$$
$$c = \begin{cases} 1, 2, 3 & if \quad RGB\ image \\ 1 & if \quad gray\ scale\ image \end{cases} \tag{23}$$

The concept of Tsallis entropy is used for thresholding of the image. Assume a given image has N pixels and L gray levels, and p_i is the probability distributions of pixels at level i. For bi-level thresholding, the image is divided into two classes, class A and class B by a threshold value t, the probability distributions of class A and B class are given by

$$p_A^c = \frac{p_0^c}{P^{c_A}}, \frac{p_1^c}{P^{c_A}}, \dots, \frac{p_{t-1}^c}{P^{c_A}} \quad and \quad p_B = \frac{p_t^c}{P^{c_B}}, \frac{p_{t+1}^c}{P^{c_B}}, \dots, \frac{p_{L-1}^c}{P^{c_B}} \tag{24}$$

Table 9
Comparison of standard deviation obtained by EKH, KH I, KH II and KH IV algorithms using Otsu's method.

Image	m	Standard deviation			
		EKH	KH I	KH II	KH IV
Baboon	3	**1.28932e−005**	2.34652e−4	2.09872e−4	1.98752e−4
	4	**2.43896e−004**	0.00367	0.00323	0.00268
	5	**0.00423**	0.00989	0.00931	0.00856
	6	**0.00872**	0.08504	0.08062	0.07415
Donna	3	**2.08561e−005**	2.95629e−4	2.61080e−4	2.38876e−4
	4	**2.89564e−004**	0.00534	0.00510	0.00429
	5	**0.00578**	0.01182	0.01090	0.00989
	6	**0.00967**	0.08878	0.08750	0.08411
Fruit	3	**1.11553e−005**	3.05476e−4	2.81899e−4	2.64537e−4
	4	**1..32465e-004**	0.00235	0.00221	0.00185
	5	**0.00388**	0.00875	0.00863	0.00784
	6	**0.00720**	0.08034	0.07985	0.07882
Flower	3	**2.23406e−005**	2.89954e−4	2.85763e−4	2.79556e−4
	4	**2.92547e−004**	0.00558	0.00539	0.00480
	5	**0.00543**	0.01104	0.01088	0.00976
	6	**0.00934**	0.08967	0.08869	0.08413
House	3	**1.22904e−005**	2.32188e−4	2.10712e−4	1.98655e−4
	4	**2.30254e−004**	0.00309	0.00288	0.00251
	5	**0.00446**	0.00936	0.00918	0.00887
	6	**0.00856**	0.08749	0.08156	0.07608
Pallon	3	**1.53048e−005**	3.61750e−4	3.47655e−4	3.19577e−4
	4	**1..68056e-004**	0.00328	0.00305	0.00283
	5	**0.00472**	0.00889	0.00872	0.00853
	6	**0.00836**	0.07260	0.07095	0.06903
Sailboat	3	**3.01179e−005**	3.94465e−4	3.92260e−4	3.89498e−4
	4	**3.32115e−004**	0.00605	0.00542	0.00492
	5	**0.00678**	0.01374	0.01250	0.01137
	6	**0.00987**	0.09472	0.09209	0.09121
Tulips	3	**1.78375e−005**	2.52006e−4	2.45192e−4	2.34808e−4
	4	**2.67833e−004**	0.00466	0.00375	0.00332
	5	**0.00506**	0.01033	0.00996	0.00942
	6	**0.00912**	0.09341	0.09125	0.08873
Voit	3	**1.40846e−005**	2.45290e−4	2.23906e−4	2.10472e−4
	4	**2.52845e−004**	0.00456	0.00345	0.00299
	5	**0.00504**	0.01123	0.01045	0.00958
	6	**0.00935**	0.08866	0.08629	0.08297
Yacht	3	**2.10545e−005**	2.98946e−4	2.87483e−4	2.66865e−4
	4	**2.98903e−004**	0.00647	0.00529	0.00472
	5	**0.00531**	0.01265	0.01040	0.00956
	6	**0.00913**	0.09075	0.08922	0.08787

where $P^{c_A} = \sum_{i=0}^{t-1} p_i^c$ and $P^{c_B} = \sum_{i=t}^{L-1} p_i^c$

Now the Tsallis entropy of class A class B is defined as:

$$S_q^{c_A}(t) = \frac{1 - \sum_{i=0}^{t-1}(p_i^c/P^{c_A})^q}{q-1}$$
$$S_q^{c_B}(t) = \frac{1 - \sum_{i=t}^{L-1}(p_i^c/P^{c_B})^q}{q-1} \tag{25}$$

The total Tsallis entropy $S_q(t)$ of the image is defined as:

$$\begin{aligned} S_q(t) &= S_q^{c_A}(t) + S_q^{c_B}(t) + (1-q) \times S_q^{c_A}(t) \times S_q^{c_B}(t) \\ &= \frac{1 - \sum_{i=0}^{t-1}(p_i^c/P^{c_A})^q}{q-1} + \frac{1 - \sum_{i=t}^{L-1}(p_i^c/P^{c_B})^q}{q-1} \\ &\quad +(1-q) \times \frac{1 - \sum_{i=0}^{t-1}(p_i^c/P^{c_A})^q}{q-1} \times \frac{1 - \sum_{i=t}^{L-1}(p_i^c/P^{c_B})^q}{q-1} \end{aligned} \tag{26}$$

The optimal threshold values are obtained by maximizing the Tsallis entropy $S_q(t)$:

$$t^{*c} = \arg\max(S_q^{c_A}(t) + S_q^{c_B}(t) + (1-q) \times S_q^{c_A}(t) \times S_q^{c_B}(t)) \tag{27}$$

Subject to following constraint:

$$\left|P^{c_A} + P^{c_B}\right| - 1 < S < 1 - \left|P^{c_A} + P^{c_B}\right| \tag{28}$$

The Tsallis entropy method can be easily extended for multi-level thresholding using Eq. (29)

$$\begin{aligned} \left[t_1^{*c}, t_2^{*c}, \ldots, t_{m-1}^{*c}\right] &= \arg\max(S_q^{c_1}(t) + S_q^{c_2}(t) \\ &\quad + \cdots + S_q^{c_m}(t) + (1+q).S_q^{c_1}(t).S_q^{c_1}(t)\cdots S_q^{c_m}(t)) \end{aligned} \tag{29}$$

where

$$S_q^{c_1}(t) = \frac{1 - \sum_{i=0}^{t_1-1}(p_i^c/P^{c_1})^q}{q-1}$$

$$S_q^{c_2}(t) = \frac{1 - \sum_{i=t_1}^{t_2-1}(p_i^c/P^{c_2})^q}{q-1} \quad and$$

$$S_q^{c_m}(t) = \frac{1 - \sum_{i=t_{m-1}}^{L-1}(p_i^c/P^{c_m})^q}{q-1}$$

Table 10
Comparison of standard deviation obtained by MFA, MGOA, WCA and BA algorithms using Otsu's method.

Image	m	Standard deviation			
		MFA	MGOA	WCA	BA
Baboon	3	1.28940e−005	1.29855e−005	1.30337e−005	1.30873e−005
	4	2.43895e−004	2.44590e−004	2.45129e−004	2.45674e−004
	5	0.00422	0.00475	0.00515	0.00547
	6	0.00881	0.00923	0.00991	0.01044
Donna	3	2.08556e−005	2.09267e−005	2.09724e−005	2.14212e−005
	4	2.89569e−004	2.90258e−004	2.90860e−004	2.91402e−004
	5	0.00580	0.00626	0.00668	0.00690
	6	0.00977	0.01045	0.01113	0.01142
Fruit	3	1.11562e−005	1.12336e−005	1.12826e−005	1.13292e−005
	4	1.32469e−004	1.33153e−004	1.33664e−004	1.34284e−004
	5	0.00389	0.00434	0.00463	0.00499
	6	0.00728	0.00805	0.00853	0.00860
Flower	3	2.23414e−005	2.24205e−005	2.24886e−005	2.25352e−005
	4	2.92546e−004	2.93250e−004	2.93806e−004	2.94304e−004
	5	0.00544	0.00585	0.00623	0.00664
	6	0.00932	0.01013	0.01082	0.01146
House	3	1.22896e−005	1.23589e−005	1.24090e−005	1.24588e−005
	4	2.30257e−004	2.30942e−004	2.31478e−004	2.31967e−004
	5	0.00448	0.00480	0.00523	0.00561
	6	0.00853	0.00924	0.00989	0.01035
Pallon	3	1.53054e−005	1.53704e−005	1.54220e−005	1.54753e−005
	4	1..68063e−004	1..68841e−004	1..69327e−004	1..69830e−004
	5	0.00474	0.00504	0.00548	0.00592
	6	0.00839	0.00903	0.00975	0.01038
Sailboat	3	3.01185e−005	3.01765e−005	3.02248e−005	3.02714e−005
	4	3.32119e−004	3.32781e−004	3.33270e−004	3.33784e−004
	5	0.00681	0.00729	0.00774	0.00805
	6	0.00998	0.01065	0.01079	0.01148
Tulips	3	1.78373e−005	1.78919e−005	1.79420e−005	1.79947e−005
	4	2.67840e−004	2.68489e−004	2.69085e−004	2.69561e−004
	5	0.00504	0.00548	0.00587	0.00623
	6	0.00919	0.00981	0.01047	0.01093
Voit	3	1.40854e−005	1.41378e−005	1.41866e−005	1.42362e−005
	4	2.52842e−004	2.53466e−004	2.53987e−004	2.54473e−004
	5	0.00505	0.00542	0.00581	0.00602
	6	0.00946	0.00993	0.01056	0.01098
Yacht	3	2.10549e−005	2.11066e−005	2.11558e−005	2.12057e−005
	4	2.98902e−004	2.99582e−004	3.00124e−004	3.00619e−004
	5	0.00532	0.00571	0.00616	0.00645
	6	0.00922	0.00980	0.01048	0.01095

Subject to following constraint:

$$\begin{aligned} &\left|P^{c_1}+P^{c_2}\right|-1<S^{c_1}<1-\left|P^{c_1}+P^{c_2}\right|,\\ &\left|P^{c_2}+P^{c_3}\right|-1<S^{c_2}<1-\left|P^{c_2}+P^{c_3}\right|\\ \&&\left|P^{c_{m-1}}+P^{c_m}\right|-1<S^{c_{m-1}}<1-\left|P^{c_{m-1}}+P^{c_m}\right| \end{aligned} \tag{30}$$

4. Brief explanations of the algorithms in the study

4.1. Krill herd algorithm

In this paper, a modern bio-based swarm intelligence optimization method is described which is based on the simulation of the herding of the krill swarms called Krill herd (KH). The distance of each krill from the food and from the highest density of the krill swarm decides the fitness of each agent. In KH, the position of each individual is affected by the following three main actions:

i. movement affected by other krill individuals;
ii. foraging activity; and
iii. random diffusion.

In KH, the n dimensional decision space can be presented using the following Lagrangian model:

$$\frac{dX_i}{dt} = N_i + F_i + D_i \tag{31}$$

where N_i is the motion led by other krill individuals, F_i is the foraging motion, and D_i is the physical diffusion of the ith krill individual.

The direction of motion of each krill individual, α_i, is affected by the three factors: target swarm density(target effect), local swarm density (local effect) and a repulsive swarm density (repulsive effect). The movement of krill can be stated as:

$$N_i^{new} = N^{\max}\alpha_i + \omega_n N_i^{old} \tag{32}$$

where, $\alpha_i = \alpha_i^{local} + \alpha_i^{t\arg et}$, and $N^{\max}$ is the maximum induced speed, ω_n is the inertia weight of the motion that is in the range of [0, 1], N_i^{old} is the last motion induced, α_i^{local} is the local effect provided by the neighbors and $\alpha_i^{t\arg et}$ is the target direction effect provided by the best krill individual.

The effect of the individual krill with the best fitness on the ith individual krill is given by Eq. (33).

$$\alpha_i^{t\arg et} = C^{best}\hat{K}_{i,best}\hat{X}_{i,best} \tag{33}$$

Table 11
Comparison of optimal thresholds and objective values obtained by EKH and KH I algorithms using Kapur's entropy.

m	EKH				KH I			
	R	G	B	f	R	G	B	f
Baboon								
3	71,132,193	54,108,160	66,125,183	47.82243	71,132,193	54,108,160	66,125,183	47.82243
4	39,92,146,198	43,87,129,171	49,99,149,198	56.72957	39,92,146,198	43,87,129,171	49,99,149,198	56.72957
5	34,77,120,163,206	30,63,101,138,175	40,83,125,167,208	**65.15325**	33,73,117,160,203	30,62,100,137,174	39,81,123,165,207	65.14093
6	28,64,101,139,176,214	30,63,99,135,171,209	32,68,105,141,177,213	**73.00435**	30,61,97,134,171,211	32,60,96,131,170,209	30,62,99,135,173,211	72.97168
Donna								
3	77,142,207	95,135,171	70,113,155	43.10597	77,142,207	95,135,171	70,113,155	43.10597
4	68,116,164,208	72,105,137,171	70,113,154,198	51.56375	68,116,164,208	72,105,137,171	70,113,154,198	51.56375
5	66,102,138,173,209	72,105,137,171,208	57,90,122,155,198	**59.49548**	67,101,136,174,208	70,103,137,171,207	57,90,121,154,198	59.48255
6	64,93,121,149,178,210	72,105,137,171,197,229	60,98,136,171,201,226	**66.91194**	60,87,116,145,176,210	69,100,132,165,195,228	56,93,132,165,199,225	66.88146
Fruit								
3	64,124,211	67,128,187	44,96,166	46.96690	64,124,211	67,128,187	44,96,166	46.96690
4	64,122,183,214	53,100,151,187	40,82,120,167	55.95746	64,122,183,214	53,100,151,187	40, 82,120,167	55.95746
5	44,89,134,183,214	45,84,120,154,187	40,81,116,155,184	**64.35767**	48,93,135,183,214	48,87,124,155,187	37,77,115,153,184	64.33873
6	38,73,110,145,183,214	45,84,120,154,186,206	39,80,111,143,167,197	**71.93213**	36,70,105,141,181,213	47,88,125,156,188,206	37,84,116,148,169,197	71.90422
Flower								
3	90,140,180	69,131,200	57,121,173	46.54285	90,140,180	69,131,200	57,121,173	46.54285
4	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291
5	61,98,134,164,194	44,80,121,161,200	37,72,109,143,184	**63.24557**	59,95,131,163,194	46,82,123,163,200	36,70,107,142,184	63.23279
6	54,83,109,139,168,196	42,74,106,139,172,205	37,71,103,135,166,197	**70.89569**	56,86,113,144,169,197	39,71,100,133,168,205	36,76,107,138,167,197	70.86540
House								
3	84,150,181	50,93,204	117,165,213	42.97286	84,150,181	50,93,204	117,165,213	42.97286
4	49,89,150,181	82,122,190,204	83,118,166,213	51.40947	49, 89,150,181	82,122,190,204	83,118,166,213	51.40947
5	49,89,150,181,206	48,88,124,190,204	83,117,150,181,213	**59.38592**	49,89,146,179,205	51,89,126,192,204	82,119,150,183,214	59.36604
6	49,84,117,151,181,206	48,88,120,154,191,204	83,115,139,163,188,214	**67.20589**	48,79,109,150178,205	47,93,125,157193,205	83,112,132,159,187,213	67.17615
Pallon								
3	108,155,202	53,105,151	54,101,160	45.09195	108,155,202	53,105,151	54,101,160	45.09195
4	51,111,156,202	46,86,124,168	40,78,116,160	53.58439	51,111,156,202	46, 86,124,168	40, 78,116,160	53.58439
5	50,72,107,163,236	41,75,109,143,176	37,72,106,139,160	**61.21343**	53,74,108,165,236	43,78,110,142,176	34,74,102,137,160	61.19226
6	70,105,139,171, 204,236	35,64,93,124,152, 178	37,72,106,139160, 193	**68.96115**	71,108,148,173, 205,236	38,68,99,127,154,176	35,68,99,135,158,193	68.92984
Sailboat								
3	40,114,172	63,118,177	64,113,171	47.02108	40,114,172	63,118,177	64,113,171	47.02108
4	40,91,135,179	53,96,140,183	52,92,133,175	55.67322	40,91,135,179	53,96,140,183	52,92,133,175	55.67322
5	40,80,114,149,184	51,93,134,175,207	48,80,113,146,178	**63.63414**	42,82,117,152,185	48,87,128,172,206	45,76,109,142,178	63.61567
6	40,77,108,138,166,194	40,73,108,143,176,209	27,57,86,116,147,178	**71.20125**	42,75,105,130,163,192	36,73,113,146,179,208	23,54,90,121,150,178	71.16914
Tulips								
3	92,150,201	71,127,180	60,115,172	47.42015	92,150,201	71,127,180	60,115,172	47.42015
4	47,95,152,201	63,108,152,194	51,97,141,185	56.15928	47,95,152,201	63,108,152,194	51,97,141,185	56.15928
5	46,90,127,167,206	46,81,120,158,196	22,63,104,146,188	**64.46129**	47,91,131,168,206	44,78,117,156,195	24,65,103,149,189	64.44301
6	46,90,127,167,206,239	46,81,119,156,194,234	22,58,94,129,164,198	**72.00804**	46,83,122,164,203,238	44,78,113,147,190,235	22,63,100,133,168,199	71.97496
Voit								
3	52,121,188	37,112,197	50,111,175	46.89495	52,121,188	37,112,197	50,111,175	46.89495
4	51,101,145,192	37,106,173,211	50,109,169,207	56.14887	51,101,145,192	37,106,173,211	50,109,169,207	56.14887
5	51,98,139,181,215	37,80,125,175,213	48,89,125,170,207	**64.69912**	51,96,135,180,213	37,82,129,175,211	50,94,131,171,207	64.67976
6	51,86,118,150,183,216	36,73,109,143,176,213	45,78,111,142,172,211	**72.59936**	47,83,113,146,181,216	37,75,113,148,180,213	42,77,108,146,169,205	72.56653
Yacht								
3	62,111,162	59,108,156	52,101,152	46.75981	62,111,162	59,108,156	52,101,152	46.75981
4	47,85,126,168	59,108,153,201	51,100,150,213	55.34433	47, 85,126,168	59,108,153,201	51,100,150,213	55.34433
5	45,82,123,165,208	41,78,116,159,201	46,92,136,176,213	**63.60461**	45,81,119,163,208	43,80,119,157,202	49,96,140,178,213	63.58958
6	41,74,108,142,176,208	38,71,107,138,169,201	36,71,106,142,178,213	**71.47703**	38,71,103,137,172,209	40,78,109,137,169,201	38,78,110,147,181,214	71.44368

Table 12
Comparison of optimal thresholds and objective values obtained by KH II and KH IV algorithms using Kapur's entropy.

m	KH II				KH IV			
	R	G	B	f	R	G	B	f
Baboon								
3	71,132,193	54,108,160	66,125,183	47.82243	71,132,193	54,108,160	66,125,183	47.82243
4	39,92,146,198	43,87,129,171	49,99,149,198	56.72957	39,92,146,198	43,87,129,171	49,99,149,198	56.72957
5	34,76,119,162,205	30,63,100,137,174	43,87,128,169,211	65.14935	34,75,121,164,207	30,64,102,139,174	41,85,129,170,209	65.14846
6	28,62,101,139,174,213	30,60,97,132,171,209	31,69,102,140,175,211	72.98830	29,64,102,141,178,214	30,65,98,134,173,210	31,70,105,143,177,213	72.98648
Donna								
3	77,142,207	95,135,171	70,113,155	43.10597	77,142,207	95,135,171	70,113,155	43.10597
4	68,116,164,208	72,105,137,171	70,113,154,198	51.56375	68,116,164,208	72,105,137,171	70,113,154,198	51.56375
5	67,103,138,174,209	70,103,137,171,208	57,92,124,155,198	59.48793	67,102,137,176,208	70,104,138,172,208	57,93,125,156,199	59.48711
6	61,89,118,147,178,210	70,100,134,172,197,229	65,102,139,171,201,226	66.89563	65,92,123,150,179,210	73,104,136,174,198,229	62,100,137,173,204,227	66.89355
Fruit								
3	64,124,211	67,128,187	44,96,166	46.96690	64,124,211	67,128,187	44,96,166	46.96690
4	64,122,183,214	53,100,151,187	40, 82,120,167	55.95746	64,122,183,214	53,100,151,187	40, 82,120,167	55.95746
5	46,92,136,183,214	48,86,122,155,187	40,81,116,155,187	64.35264	44,93,137,184,214	45,86,124,156,188	41,83,117,156,184	64.35092
6	37,74,113,148,183,214	46,87,123,155,186,206	40,81,115,146,169,197	71.91825	37,76,112,147,185,214	45,847,122,156,188,206	40,83,112,145,169,197	71.91612
Flower								
3	90,140,180	69,131,200	57,121,173	46.54285	90,140,180	69,131,200	57,121,173	46.54285
4	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291
5	61,98,135,165,195	45,81,121,162,200	37,72,111,144 185	63.24184	61,99,137,165,194	45,80,123,162,200	37,74,100,145,184	63.24089
6	53,83,109,138,164,196	42,76,102,136, 170,205	36,72,105,138, 168,197	70.88135	54,83,109,139, 168,196	42,74,106,139, 172,205	37,71,103,135, 166,197	70.87961
House								
3	84,150,181	50,93,204	117,165,213	42.97286	84,150,181	50,93,204	117,165,213	42.97286
4	49, 89,150,181	82,122,190,204	83,118,166,213	51.40947	49, 89,150,181	82,122,190,204	83,118,166,213	51.40947
5	49,90,152,182,207	49,89,124,191,204	84,117,149,181,213	59.38023	49,91,152,182,207	48,90,126,190,205	83,118,152,182,214	59.37937
6	49,80,115,151,179,206	47,87,123,156,191,204	83,112,137,160,187,214	67.19081	50,83,119,153,181,206	48,91,121,156,193,204	83,116,141,165,189,215	67.18883

(continued on next page)

Table 12 (*continued*).

m	KH II				KH IV			
	R	G	B	f	R	G	B	f
Pallon								
3	108,155,202	53,105,151	54,101,160	45.09195	108,155,202	53,105,151	54,101,160	45.09195
4	51,111,156,202	46, 86,124,168	40, 78,116,160	53.58439	51,111,156,202	46, 86,124,168	40, 78,116,160	53.58439
5	51,70,107,163,236	40,73,108,142,176	37,70,105,138,160	61.20726	50,71,109,164,236	41,76,109,144,177	37,72,108,139,160	61.20624
6	67,107,140,171,204,236	37,66,96,125,153,178	36,69,102,137,160,193	68.94740	70,106,140,173,207,236	35,66,94,126,154,179	38,75,109,142,163,193	68.94536
Sailboat								
3	40,114,172	63,118,177	64,113,171	47.02108	40,114,172	63,118,177	64,113,171	47.02108
4	40,91,135,179	53,96,140,183	52,92,133,175	55.67322	40,91,135,179	53,96,140,183	52,92,133,175	55.67322
5	40,81,116,150,185	48,88,130,172,207	46,80,112,145,178	63.62738	40,83,114,150,184	51,96,137,177,207	48,83,115,147,178	63.62620
6	40,74,105,136,164,193	37,72,104,141,175,208	24,57,88,118,149,178	71.18445	41,79,111,141,166,193	40,76,109,145,177,209	27,60,87,119,149,179	71.18267
Tulips								
3	92,150,201	71,127,180	60,115,172	47.42015	92,150,201	71,127,180	60,115,172	47.42015
4	47,95,152,201	63,108,152,194	51,97,141,185	56.15928	47,95,152,201	63,108,152,194	51,97,141,185	56.15928
5	46,90,128,168,207	46,79,119,157,196	23,64,103,144,188	64.45494	46,92,128,169,206	46,83,121,159,196	22,65,106,147,188	64.45384
6	46,88,124,163,205,239	45,78,115,155 195,234	21,56,95,133,165,199	71.99159	47,93,129,170,207,239	47,84,121,159,196,234	23,59,97,131,166,199	71.98951
Voit								
3	52,121,188	37,112,197	50,111,175	46.89495	52,121,188	37,112,197	50,111,175	46.89495
4	51,101,145,192	37,106,173,211	50,109,169,207	56.14887	51,101,145,192	37,106,173,211	50,109,169,207	56.14887
5	51,99,141,182,215	37,82,126,175,213	48,92,127,171,211	64.69382	51,99,140,183,215	37,81,126,177,213	48,89,127,172,207	64.69285
6	50,83,117,151,185,216	35,71,107,145,179,213	44,78,112,144,171,207	72.58486	53,88,121,153,185,216	37,75,112,145,179,213	45,81,113,145,173,213	72.58271
Yacht								
3	62,111,162	59,108,156	52,101,152	46.75981	62,111,162	59,108,156	52,101,152	46.75981
4	47, 85,126,168	59,108,153,201	51,100,150,213	55.34433	47, 85,126,168	59,108,153,201	51,100,150,213	55.34433
5	46,83,123,164,208	40,76,115,157,201	46,93,138,176,213	63.59840	45,80,124,166,208	41,79,117,160,201	46,94,138,176,214	63.59738
6	41,73,107,140,173,208	35,70,106,135,168,201	36,69,109,145,179,213	71.46227	41,77,111,145,176,210	38,75,109,139,171,202	37,73,107,145,179,214	71.46033

Table 13
Comparison of optimal thresholds and objective values obtained by MFA and MGOA algorithms using Kapur's entropy.

m	MFA				MGOA			
	R	G	B	f	R	G	B	f
Baboon								
3	71,132,193	54,108,160	66,125,183	47.82243	71,132,193	54,108,160	66,125,183	47.82243
4	39,92,146,198	43,87,129,171	49,99,149,198	56.72957	39,92,146,198	43,87,129,171	49,99,149,198	56.72957
5	34,77,120,163,206	30,63,101,138,175	40,83,125,167,208	65.15325	34,78,120,163,206	30,63,102,138,175	41,83,125,167,208	65.15227
6	28,65,101,139,176,214	30,63,99,135,171,209	32,68,105,141,177,213	73.00498	29,65,101,140,176,214	32,63,99,136,171,209	33,69,105,141,177,214	73.00153
Donna								
3	77,142,207	95,135,171	70,113,155	43.10597	77,142,207	95,135,171	70,113,155	43.10597
4	68,116,164,208	72,105,137,171	70,113,154,198	51.56375	68,116,164,208	72,105,137,171	70,113,154,198	51.56375
5	66,102,138,173,209	72,105,137,171,208	57,90,122,155,198	59.49548	67,102,138,173,209	73,105,137,171,208	57,90,122,156,198	59.49452
6	64,93,121,149,178,210	72,105,137,171,197,229	60,98,136,171,201,27	66.91253	64,95,122,149,178,210	73,105,138,171,197,230	60,99,136,172,202,226	66.90913
Fruit								
3	64,124,211	67,128,187	44,96,166	46.96690	64,124,211	67,128,187	44,96,166	46.96690
4	64,122,183,214	53,100,151,187	40,82,120,167	55.95746	64,122,183,214	53,100,151,187	40, 82,120,167	55.95746
5	44,89,134,183,214	45,84,120,154,187	40,81,116,155,184	64.35767	44,89,135,183,214	45,84,122,154,187	41,81,116,155,184	64.35668
6	38,73,110,145,183,214	45,84,120,154,186,206	39,82,111,143,167,197	71.93285	39,74,111,145,183,214	46,84,120,156,186,206	39,80,112,144,167,198	71.92929
Flower								
3	90,140,180	69,131,200	57,121,173	46.54285	90,140,180	69,131,200	57,121,173	46.54285
4	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291
5	61,98,134,164,194	44,80,121,161,200	37,72,109,143,184	63.24557	61,99,134,164,194	44,81,121,161,200	37,72,109,144,184	63.24459
6	55,83,109,139,168,196	42,74,106,139,172,205	37,71,103,135,166,197	70.89642	56,83,110,139,169,196	43,75,106,139,173,205	37,71,105,135,166,198	70.89285
House								
3	84,150,181	50,93,204	117,165,213	42.97286	84,150,181	50,93,204	117,165,213	42.97286
4	49,89,150,181	82,122,190,204	83,118,166,213	51.40947	49, 89,150,181	82,122,190,204	83,118,166,213	51.40947
5	49,89,150,181,206	48,88,124,190,204	83,117,150,181, 213	59.38592	49,89,150,181,206	48,88,124,190,204	83,117,150,181,213	59.38496
6	49,84,117,151,181,206	49,88,120,154,191,204	83,115,139,163,188,214	67.20656	51,84,118,151,181,206	48,89,120,155,192,204	84,116,139,164,188,214	67.20298
Pallon								
3	108,155,202	53,105,151	54,101,160	45.09195	108,155,202	53,105,151	54,101,160	45.09195
4	51,111,156,202	46,86,124,168	40,78,116,160	53.58439	51,111,156,202	46, 86,124,168	40, 78,116,160	53.58439
5	50,72,107,163,236	41,75,109,143,176	37,72,106,139,160	61.21343	51,72,107,163,236	41,76,109,143,176	38,72,106,139,160	61.21244
6	71,105,139,171,204,236	35,64,93,124,152,178	37,72,106,139160,193	68.96176	71,105,140,172,204,236	35,65,93,126,152,178	37,73,107,139161,193	68.95829
Sailboat								
3	40,114,172	63,118,177	64,113,171	47.02108	40,114,172	63,118,177	64,113,171	47.02108
4	40,91,135,179	53,96,140,183	52,92,133,175	55.67322	40,91,135,179	53,96,140,183	52,92,133,175	55.67322
5	40,80,114,149,184	51,93,134,175,207	48,80,113,146,178	63.63414	40,80,115,149,184	52,93,134,175,207	48,80,113,146,179	63.63312
6	40,77,108,138,166,194	40,73,108,143,176,209	27,57,86,116,148,178	71.20186	41,78,109,138,166,194	40,74,108,144,176,210	27,57,88,117,147,178	71.19837
Tulips								
3	92,150,201	71,127,180	60,115,172	47.42015	92,150,201	71,127,180	60,115,172	47.42015
4	47,95,152,201	63,108,152,194	51,97,141,185	56.15928	47,95,152,201	63,108,152,194	51,97,141,185	56.15928
5	46,90,127,167,206	46,81,120,158,196	22,63,104,146,188	64.46129	46,90,127,168,206	46,81,121,158,196	22,64,104,146,188	64.46030
6	46,90,127,167,206,239	46,81,119,156,194,235	22,58,94,129,164,198	72.00862	46,91,128,167,207,239	47,81,121,156,194,234	23,58,94,129,165,199	72.00511
Voit								
3	52,121,188	37,112,197	50,111,175	46.89495	52,121,188	37,112,197	50,111,175	46.89495
4	51,101,145,192	37,106,173,211	50,109,169,207	56.14887	51,101,145,192	37,106,173,211	50,109,169,207	56.14887
5	51,98,139,181,215	37,80,125,175,213	48,89,125,170,207	64.69912	51,98,139,182,215	38,80,125,175,213	48,89,126,170,207	64.69811
6	51,86,118,150,183,216	37,73,109,143,176,213	45,78,111,142,172,211	72.59992	51,86,119,150,183,218	36,73,111,143,177,213	45,78,112,143,172,212	72.59640
Yacht								
3	62,111,162	59,108,156	52,101,152	46.75981	62,111,162	59,108,156	52,101,152	46.75981
4	47,85,126,168	59,108,153,201	51,100,150,213	55.34433	47, 85,126,168	59,108,153,201	51,100,150,213	55.34433
5	45,82,123,165,208	41,78,116,159,201	46,92,136,176,213	63.60461	46,82,123,165,208	41,78,116,160,201	46,92,136,176,214	63.60362
6	41,74,108,142,177,208	38,71,107,138,169,201	36,71,106,142,178,213	71.47771	42,75,108,143,176,208	38,72,108,138,169,202	36,71,107,142,178,215	71.47412

Table 14
Comparison of optimal thresholds and objective values obtained by WCA and BA algorithms using Kapur's entropy.

m	WCA				BA			
	R	G	B	f	R	G	B	f
Baboon								
3	71,132,193	54,108,160	66,125,183	47.82243	71,132,193	54,108,160	66,125,183	47.82243
4	39,92,146,198	43,87,129,171	49,99,149,198	56.72957	39,92,146,198	43,87,129,171	49,99,149,198	56.72957
5	35,77,121,163,206	30,63,101,140,175	40,83,126,167,209	65.15144	35,78,120,164,206	31,63,102,138,176	42,83,125,168,208	65.15062
6	29,64,102,139,178,214	30,65,99,136,171,210	33,68,105,142,178,214	72.99887	29,65,101,141,176,215	32,63,100,136,172,210	32,69,107,141,177,215	72.99647
Donna								
3	77,142,207	95,135,171	70,113,155	43.10597	77,142,207	95,135,171	70,113,155	43.10597
4	68,116,164,208	72,105,137,171	70,113,154,198	51.56375	68,116,164,208	72,105,137,171	70,113,154,198	51.56375
5	66,102,139,174,209	73,105,138,171,208	57,90,122,157,198	59.49363	67,103,138,174,209	72,105,138,171,210	57,91,123,155,199	59.49282
6	65,94,121,150,178,210	73,105,138,171,197,230	60,99,136,173,202,226	66.90639	63,93,121,151178,212	72,104,136,170,198,229	62,98,137,171,203,226	66.90398
Fruit								
3	64,124,211	67,128,187	44,96,166	46.96690	64,124,211	67,128,187	44,96,166	46.96690
4	64,122,183,214	53,100,151,187	40,82,120,167	55.95746	64,122,183,214	53,100,151,187	40, 82,120,167	55.95746
5	44,89,135,183,215	45,86,120,154,187	41,82,116,155,184	64.35595	44,90,134,184,215	46,84,121,154,188	42,81,116,155,185	64.35503
6	40,73,111,145,185,214	46,84,121,156,186,206	39,81,112,143,167,198	71.92654	39,73,112,146,183,215	46,84,121,155,186,207	41,80,113,144,167,197	71.92416
Flower								
3	90,140,180	69,131,200	57,121,173	46.54285	90,140,180	69,131,200	57,121,173	46.54285
4	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291	62,101,142,181	61,106,150,200	49, 93,137,181	55.25291
5	61,99,134,165,194	44,80,122,161,201	37,72,109,145,184	63.24371	62,98,134,165,195	44,81,122,161,201	38,72,111,143,184	63.24295
6	56,83,109,140,168,197	43,74,106,140,172,206	38,71,105,135,166,198	70.89065	55,84,111,139,169,196	43,74,108,139,172,207	38,72,103,136,167,198	70.88769
House								
3	84,150,181	50,93,204	117,165,213	42.97286	84,150,181	50,93,204	117,165,213	42.97286
4	49,89,150,181	82,122,190,204	83,118,166,213	51.40947	49, 89,150,181	82,122,190,204	83,118,166,213	51.40947
5	49,90,150,181,207	48,88,125,191,204	85,117,150,181,213	59.38411	49,90,151,182,206	49,88,124,191,205	84,118,151,181,213	59.38343
6	50,84,118,152,181,207	48,89,120,155,192,204	84,115,141,163,188,215	67.20092	50,85,118,151,182,207	49,88,122,154,193,204	83,116,139,164,189,216	67.19793
Pallon								
3	108,155,202	53,105,151	54,101,160	45.09195	108,155,202	53,105,151	54,101,160	45.09195
4	51,111,156,202	46,86,124,168	40,78,116,160	53.58439	51,111,156,202	46, 86,124,168	40, 78,116,160	53.58439
5	51,72,107,164,236	41,76,109,144,176	37,72,108,139,160	61.21155	50,72,107,163,236	41,75,109,143,176	37,72,106,139,160	61.21074
6	71,106,140,171,205,236	35,65,94,125,152,179	38,73,106,139,160,195	68.95563	70,105,139,171,204,236	35,64,93,124,152,178	37,72,106,139,160,193	68.95317
Sailboat								
3	40,114,172	63,118,177	64,113,171	47.02108	40,114,172	63,118,177	64,113,171	47.02108
4	40,91,135,179	53,96,140,183	52,92,133,175	55.67322	40,91,135,179	53,96,140,183	52,92,133,175	55.67322
5	41,80,115,149,184	51,94,134,175,208	49,80,113,147,178	63.63237	41,80,115,150,184	52,94,135,175,207	49,81,113,147,178	63.63148
6	40,78,108,139,166,196	41,73,109,144,177,209	28,58,86,117,148,178	71.19574	42,77,109,139,166,195	40,75,108,143,177,210	28,58,87,116,147,179	71.19327
Tulips								
3	92,150,201	71,127,180	60,115,172	47.42015	92,150,201	71,127,180	60,115,172	47.42015
4	47,95,152,201	63,108,152,194	51,97,141,185	56.15928	47,95,152,201	63,108,152,194	51,97,141,185	56.15928
5	47,90,128,167,206	46,82,121,158,196	22,63,105,147,188	64.45946	47,91,127,168,206	46,81,122,158,197	22,63,105,147,189	64.45858
6	46,91,127,168,206,241	47,83,119,157,194,234	23,58,95,130,165,199	72.00247	47,90,129,167,208,239	46,82,119,157,194,236	23,59,94,130,165,199	72.00006
Voit								
3	52,121,188	37,112,197	50,111,175	46.89495	52,121,188	37,112,197	50,111,175	46.89495
4	51,101,145,192	37,106,173,211	50,109,169,207	56.14887	51,101,145,192	37,106,173,211	50,109,169,207	56.14887
5	51,99,140,181,215	37,80,126,175,214	49,89,125,170,208	64.69724	52,98,140,181,216	38,80,126,175,214	49,89,126,170,208	64.69645
6	53,86,119,150,183,217	37,74,110,143,177,214	45,79,112,142,173,212	72.59388	53,86,118,152,183,217	36,73,111,144,177,214	47,79,111,142,174,211	72.59134
Yacht								
3	62,111,162	59,108,156	52,101,152	46.75981	62,111,162	59,108,156	52,101,152	46.75981
4	47,85,126,168	59,108,153,201	51,100,150,213	55.34433	47, 85,126,168	59,108,153,201	51,100,150,213	55.34433
5	46,82,123,166,208	43,78,116,159,201	46,92,137,176,214	63.60272	46,82,124,165,209	42,78,116,161,201	47,92,137,176,214	63.60198
6	41,76,108,143,177,208	39,72,108,139,169,201	37,72,106,142,178,215	71.47142	43,74,109,141,176,210	39,71,108,139,169,202	36,73,106,143,178,215	71.46895

where, C^{best} is the effective coefficient of the krill individual with the best fitness to the ith krill individual.

The parameter of C^{best} is defined as Eq. (34).

$$C^{best} = 2\left(rand + \frac{I}{I_{\max}}\right) \tag{34}$$

where $rand$ is a random values between 0 and 1 and it is for enhancing exploration, I is the actual iteration number and $I_{\max}$ is the maximum number of iterations.

The item of foraging motion is decided by the two main effective components: the food location and its previous experience. For the ith krill individual, the motion is formulated as:

$$F_i = V_f \beta_i + \omega_f F_i^{old} \tag{35}$$

where, $\beta_i = \beta_i^{food} + \beta_i^{best}$, and V_f is the foraging speed, ω_f is the inertia weight of the foraging motion, F_i^{old} is the last foraging motion, β_i^{food} is the food attractive and β_i^{best} is the effect of the best fitness of the ith krill so far.

The random diffusion of the krill individuals is essentially a random process. This motion can be presented according to a maximum diffusion speed and a random directional vector. The equation is given as:

$$D_i = D_{\max}\delta \tag{36}$$

where D_i is the maximum diffusion speed, and δ is the random directional vector and its arrays are random numbers in $[-1, 1]$.

Hence, the equation of location update of krill i can be presented:

$$X_i(t + \Delta t) = X_i(t) + \Delta t \frac{dX_i}{dt} \tag{37}$$

It should be noted that Δt is an important parameters and should be adjusted in terms of the optimization problem. This is because this parameter can be treated as a scale factor of the speed vector. The value of Δt is completely depends on the given search space C_t and it can be presented by the following formula:

$$\Delta t = C_t \sum_{j=1}^{NV} (UB_j - LB_j) \tag{38}$$

where NV is the total number of variables, and LB_j and UB_j are lower and upper bounds of the jth variables. So the value of parameter Δt has critical effect on the performance of KH. We can adjust the parameter Δt by adjusting the parameter C_t, UB_j and LB_j.

In KH, the parameters of inertia weights (ω_n, ω_f) are very powerful that represent the variations of the global optimal attraction. The values of (ω_n, ω_f) decide the convergence speed

Table 15
Comparison of PSNR and SSIM obtained by EKH, KH I, KH II and KH IV algorithms using Kapur's entropy.

Image	m	EKH		KH I		KH II		KH IV	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Baboon	3	**16.81227**	**0.87998**	16.81227	0.87998	16.81227	0.87998	16.81227	0.87998
	4	**16.81381**	**0.88368**	16.81381	0.88368	16.81381	0.88368	16.81381	0.88368
	5	**16.81590**	**0.94180**	16.81442	0.93871	16.81521	0.94090	16.81507	0.94050
	6	**16.81856**	**0.96437**	16.81594	0.96154	16.81761	0.96303	16.81715	0.96217
Donna	3	**20.15640**	**0.86095**	20.15640	0.86095	20.15640	0.86095	20.15640	0.86095
	4	**20.15780**	**0.88108**	20.15780	0.88108	20.15780	0.88108	20.15780	0.88108
	5	**20.15961**	**0.91624**	20.15761	0.91443	20.15893	0.91574	20.15866	0.91539
	6	**20.16215**	**0.92076**	20.15980	0.91783	20.16095	0.91969	20.16043	0.91886
Fruit	3	**21.75639**	**0.85262**	21.75639	0.85262	21.75639	0.85262	21.75639	0.85262
	4	**21.75822**	**0.86458**	21.75822	0.86458	21.75822	0.86458	21.75803	0.86412
	5	**21.75995**	**0.93161**	21.75814	0.92975	21.75922	0.93107	21.75846	0.92945
	6	**21.76244**	**0.94344**	21.76013	0.94104	21.76175	0.94251	21.76175	0.94251
Flower	3	**18.65517**	**0.87416**	18.65517	0.87416	18.65517	0.87416	18.65517	0.87416
	4	**18.65691**	**0.88711**	18.65691	0.88711	18.65691	0.88711	18.65691	0.88711
	5	**18.65876**	**0.93231**	18.65697	0.92942	18.65807	0.93077	18.65779	0.93014
	6	**18.66105**	**0.94352**	18.65879	0.94026	18.66036	0.94235	18.65975	0.94142
House	3	**15.37225**	**0.85331**	15.37225	0.85331	15.37225	0.85331	15.37225	0.85331
	4	**15.37479**	**0.87095**	15.37479	0.87095	15.37479	0.87095	15.37479	0.87095
	5	**15.37655**	**0.92356**	15.37482	0.92176	15.37606	0.92312	15.37580	0.92278
	6	**15.37895**	**0.93161**	15.37664	0.92906	15.37801	0.93127	15.37754	0.93035
Pallon	3	**18.35515**	**0.84324**	18.35515	0.84324	18.35515	0.84324	18.35515	0.84324
	4	**18.35674**	**0.86352**	18.35674	0.86352	18.35674	0.86352	18.35674	0.86352
	5	**18.35822**	**0.91200**	18.35706	0.91021	18.35765	0.91164	18.35738	0.91112
	6	**18.35987**	**0.93428**	18.34730	0.93122	18.34875	0.93260	18.34814	0.93157
Sailboat	3	**16.47373**	**0.89558**	16.47373	0.89558	16.47373	0.89558	16.47373	0.89558
	4	**16.47499**	**0.90354**	16.47499	0.90354	16.47499	0.90354	16.47499	0.90354
	5	**16.47685**	**0.92950**	16.47547	0.92776	16.47645	0.92888	16.47619	0.92824
	6	**16.47872**	**0.94040**	16.46808	0.93887	16.47019	0.93951	16.46952	0.93847
Tulips	3	**17.44759**	**0.89211**	17.44759	0.89211	17.44759	0.89211	17.44759	0.89211
	4	**17.44928**	**0.90784**	17.44928	0.90784	17.44928	0.90784	17.44928	0.90784
	5	**17.45096**	**0.95069**	17.44881	0.94851	17.44994	0.94975	17.44976	0.94923
	6	**17.45305**	**0.96615**	17.44977	0.96303	17.45159	0.96430	17.45163	0.96345
Voit	3	**18.97051**	**0.87195**	18.97051	0.87195	18.97051	0.87195	18.97051	0.87195
	4	**18.97231**	**0.88314**	18.97231	0.88314	18.97231	0.88314	18.97231	0.88314
	5	**18.97410**	**0.93316**	18.97254	0.93112	18.97362	0.93243	18.97335	0.93184
	6	**18.97615**	**0.94457**	18.97216	0.93204	18.97289	0.94369	18.97312	0.94275
Yacht	3	**18.49345**	**0.85109**	18.49345	0.85109	18.49345	0.85109	18.49345	0.85109
	4	**18.49523**	**0.88988**	18.49523	0.88988	18.49523	0.88988	18.49523	0.88988
	5	**18.49785**	**0.95810**	18.49584	0.95546	18.49713	0.95709	18.49694	0.95657
	6	**18.50933**	**0.96339**	18.49621	0.96139	18.50806	0.96226	18.50759	0.96132

and how to update the position of the krill. In the basic KH, the inertia weights (ω_n, ω_f) are set to 0.9 at the early search stage to highlight exploration. Finally, they are linearly decreased to 0.1 to stimulate exploitation.

The parameter of foraging speed V_f decides the convergence speed. However, it is set to 0.02 in the basic KH. In order to improve the performance of the KH algorithm, we can modify the V_f parameter.

In addition, the position of an individual krill is decided by movement affected by other krill individuals, foraging activity and random diffusion. So we can modify the above three ways of krill to can improve the performance of the KH algorithm.

4.2. Improved Krill herd algorithm

Intensification and diversification are two important characteristics of the metaheuristic algorithms [77]. Intensification searches around the current best solutions and selects the best candidate points. The diversification process allows the optimizer to explore the search space more efficiently. The parameters of inertia weights $\left(\omega_n, \omega_f\right)$ represent the variations of the global optimal attraction that affect the convergence speed and the position update of the krill individuals in the KH algorithm. In the basic KH algorithm, the inertia weights $\left(\omega_n, \omega_f\right)$ are set to a large value of 0.9 at initial search state to emphasize the exploration and are linearly decreased to 0.1 at the end to encourage exploitation [60]. Inspired by the experience of our lives we hope that the krill herds explore the global are at the beginning. With the increasing of the number of iterations krill herds are encouraged for the local search ability. In the final stage the krill just need to carefully search a local area without any exploration. Through the above analysis, the first improvement is proposing a new linear decreasing inertia weights. We make $\left(\omega_n, \omega_f\right)$ have the following formulation:

$$\omega = 0.9 - ((0.9 - 0.1)/MI) * j \tag{39}$$

where [0.1, 0.9] is the range of inertia weight, MI is the maximum number of iteration, and j is the present number of iteration, ω_n and ω_f are linearly decreasing form 0.9 to 0.1 with iteration.

The KH algorithm is proposed for solving global optimization problems [60]. It has powerful ability in exploration, but at times it may trap into some local optima. Therefore, to address this

Table 16
Comparison of PSNR and SSIM obtained by MFA, MGOA, WCA and BA algorithms using Kapur's entropy.

Image	m	MFA		MGOA		WCA		BA	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Baboon	3	16.81227	0.87998	16.81227	0.87998	16.81227	0.87998	16.81227	0.87998
	4	16.81381	0.88368	16.81381	0.88368	16.81381	0.88368	16.81381	0.88368
	5	16.81590	0.94180	16.81578	0.94171	16.81569	0.94162	16.81540	0.94133
	6	16.81850	0.96432	16.81820	0.96408	16.81784	0.96371	16.81757	0.96339
Donna	3	20.15640	0.86095	20.15640	0.86095	20.15640	0.86095	20.15640	0.86095
	4	20.15780	0.88108	20.15780	0.88108	20.15780	0.88108	20.15780	0.88108
	5	20.15961	0.91624	20.15949	0.91614	20.15942	0.91605	20.15911	0.91576
	6	20.16207	0.92073	20.16178	0.92149	20.16142	0.92006	20.16116	0.91977
Fruit	3	21.75639	0.85262	21.75639	0.85262	21.75639	0.85262	21.75639	0.85262
	4	21.75822	0.86458	21.75822	0.86458	21.75822	0.86458	21.75803	0.86412
	5	21.75995	0.93161	21.75982	0.93151	21.75976	0.93145	21.75945	0.93114
	6	21.76239	0.94337	21.76208	0.94313	21.76171	0.94276	21.76142	0.94246
Flower	3	18.65517	0.87416	18.65517	0.87416	18.65517	0.87416	18.65517	0.87416
	4	18.65691	0.88711	18.65691	0.88711	18.65691	0.88711	18.65691	0.88711
	5	18.65876	0.93231	18.65864	0.93220	18.65864	0.93213	18.65826	0.93182
	6	18.66096	0.94346	18.66069	0.94321	18.66033	0.94286	18.66007	0.94253
House	3	15.37225	0.85331	15.37225	0.85331	15.37225	0.85331	15.37225	0.85331
	4	15.37479	0.87095	15.37479	0.87095	15.37479	0.87095	15.37479	0.87095
	5	15.37655	0.92356	15.37642	0.92347	15.37635	0.92339	15.37605	0.92309
	6	15.37886	0.93155	15.37858	0.93130	15.37827	0.93097	15.37793	0.93060
Pallon	3	18.35515	0.84324	18.35515	0.84324	18.35515	0.84324	18.35515	0.84324
	4	18.35674	0.86352	18.35674	0.86352	18.35674	0.86352	18.35674	0.86352
	5	18.35822	0.91200	18.35812	0.91188	18.35801	0.91183	18.35774	0.91152
	6	18.35980	0.93421	18.35951	0.93400	18.35918	0.93361	18.35887	0.93329
Sailboat	3	16.47373	0.89558	16.47373	0.89558	16.47373	0.89558	16.47373	0.89558
	4	16.47499	0.90354	16.47499	0.90354	16.47499	0.90354	16.47499	0.90354
	5	16.47685	0.92950	16.47674	0.92939	16.47665	0.92932	16.47635	0.92901
	6	16.47866	0.94032	16.47835	0.94011	16.47804	0.93974	16.47774	0.93942
Tulips	3	17.44759	0.89211	17.44759	0.89211	17.44759	0.89211	17.44759	0.89211
	4	17.44928	0.90784	17.44928	0.90784	17.44928	0.90784	17.44928	0.90784
	5	17.45096	0.95069	17.45083	0.95059	17.45078	0.95051	17.45047	0.95020
	6	17.45294	0.96608	17.45272	0.96585	17.45233	0.96548	17.45205	0.96518
Voit	3	18.97051	0.87195	18.97051	0.87195	18.97051	0.87195	18.97051	0.87195
	4	18.97231	0.88314	18.97231	0.88314	18.97231	0.88314	18.97231	0.88314
	5	18.97410	0.93316	18.97397	0.93305	18.97388	0.93297	18.97362	0.93268
	6	18.97606	0.94453	18.97582	0.94428	18.97543	0.94390	18.97516	0.94358
Yacht	3	18.49345	0.85109	18.49345	0.85109	18.49345	0.85109	18.49345	0.85109
	4	18.49523	0.88988	18.49523	0.88988	18.49523	0.88988	18.49523	0.88988
	5	18.49785	0.95810	18.49773	0.95797	18.49765	0.95791	18.49738	0.95759
	6	18.50927	0.96334	18.50901	0.96308	18.50862	0.96274	18.50830	0.96241

problem, the second improvement is presenting the adaptive update strategy of C^{best} parameter to help the krill individuals get out of the possible local optima regions. In the new strategy, C^{best} is a large value in the initial stage of finding the optimal value that the KH algorithm has the strong ability of exploration (global search), and it gradually reduce with the increase of iteration in order to accurately search the optimal value. The adaptive update scheme of C^{best} used herein is formulated as:

$$C^{best} = 2 * (1 - j/MI) \tag{40}$$

where j is the number of iteration and MI is the maximum number of iteration.

Moreover, the third improvement is to insert a new position update method of krill individuals into KH algorithm with the aim of accelerating its global convergence speed. The i krill individual firstly update position vector by Eq. (37), then move to the global optimal value at interval of $t + \Delta t$ using Eq. (40). The firefly algorithm is one of the novel swarm intelligence algorithm proposed by [78]. Many simulation results in the related literatures reveals that the FA algorithm has better performance in relation to other swarm intelligence techniques, so it has attained a lot of interest and has been widely employed for solving several practical and complex optimization problems [79–82]. In FA algorithm, the third term is randomization that avoids trapping into local optimal solution, so the randomization with α being the randomization parameter is introduced into the Eq. (41).

$$X_i(t+\Delta t) = X_i(t+\Delta t) + (X_{gbest} - X_i(t+\Delta t)) + \alpha \times (rand - 1/2) \tag{41}$$

where X_{gbest} is the position of best krill individual of the entire firefly swarm and $rand$ is a random number generator uniformly distributed in [0,1].

The pseudo code of the EKH algorithm is shown in Fig. 1 and the flowchart depicting this algorithm is given in Fig. 2.

4.3. Modified firefly algorithm

In order to balance the local search and global search ability of firefly algorithm and avoid falling into local optimal value, a modified firefly algorithm (MFA) is presented by He and Huang [70].

In the basic firefly algorithm (FA), the initialization of fireflies is performed randomly. The chaotic Tent map has better properties of ergodicity and randomicity than random distribution, so chaotic Tent map is introduced to the initialization phase to

Table 17
Comparison of standard deviation obtained by EKH, KH I, KH II and KH IV algorithms using Kapur's entropy.

Image	m	Standard deviation			
		EKH	KH I	KH II	KH IV
Baboon	3	**8.75423e−006**	9.06682e−005	8.76410e−005	8.64637e−005
	4	**8.34280e−005**	6.95983e−004	6.82104e−004	6.69968e−004
	5	**7.46708e−004**	0.00214	0.00202	0.00193
	6	**0.00134**	0.00879	0.00848	0.00810
Donna	3	**9.15572e−006**	9.95857e−005	9.90213e−005	8.88716e−005
	4	**8.59064e−005**	7.12187e−004	6.99203e−004	6.82618e−004
	5	**7.99170e−004**	0.00356	0.00325	0.00232
	6	**0.00196**	0.00946	0.00922	0.00893
Fruit	3	**8.36415e−006**	8.87238e−005	8.54891e−005	8.42019e−005
	4	**8.21359e−005**	6.74938e−004	6.63705e−004	6.55994e−004
	5	**7.22107e−004**	0.00185	0.00166	0.00142
	6	**0.00102**	0.00762	0.00743	0.00725
Flower	3	**9.47683e−006**	9.80337e−005	9.76430e−005	9.72548e−005
	4	**8.86842e−005**	7.39016e−004	7.28377e−004	7.26135e−004
	5	**8.15584e−004**	0.00463	0.00435	0.00406
	6	**0.00245**	0.00998	0.00965	0.00947
House	3	**8.66758e−006**	8.98146e−005	8.77807e−005	8.63135e−005
	4	**8.20907e−005**	6.74598e−004	6.63462e−004	6.58792e−004
	5	**7.26794e−004**	0.00197	0.00176	0.00152
	6	**0.00122**	0.00835	0.00812	0.00808
Pallon	3	**8.89624e−006**	9.25919e−005	9.04617e−005	8.93405e−005
	4	**8.54297e−005**	7.02813e−004	6.93870e−004	6.78606e−004
	5	**7.60375e−004**	0.00254	0.00232	0.00215
	6	**0.00153**	0.00890	0.00875	0.00857
Sailboat	3	**1.24668e−005**	1.14889e−004	1.02556e−004	9.98463e−005
	4	**9.34225e−005**	8.92674e−004	8.81267e−004	8.70343e−004
	5	**9.14981e−004**	0.00566	0.00537	0.00509
	6	**0.00360**	0.01224	0.01205	0.01189
Tulips	3	**9.06564e−006**	9.56780e−005	9.38723e−005	9.29535e−005
	4	**8.55923e−005**	7.08781e−004	6.93642e−004	6.72076e−004
	5	**7.76982e−004**	0.00289	0.00265	0.00247
	6	**0.00171**	0.00912	0.00896	0.00880
Voit	3	**8.86537e−006**	9.13467e−005	8.88934e−005	8.72359e−005
	4	**8.46365e−005**	6.85322e−004	6.73901e−004	6.62805e−004
	5	**7.63723e−004**	0.00230	0.00216	0.00195
	6	**0.00224**	0.00957	0.00936	0.00918
Yacht	3	**9.16094e−006**	9.97967e−005	9.95885e−005	9.93776e−005
	4	**8.60357e−005**	7.16789e−004	7.04668e−004	6.90237e−004
	5	**8.02455e−004**	0.00390	0.00378	0.00356
	6	**0.00230**	0.00958	0.00939	0.00915

improve the diversity of fireflies and enhance global search ability in the MFA.

In the movement phase of basic FA, each firefly do local search and do not memorize any history of global best in each iteration process that causes they miss their best situations. Therefore, FA has better the ability of intensification and the local convergence speed is very good, but it easily fallen into local optimum and premature. To solve the problem, the new movement equation is proposed in MFA algorithm which is inspired by particle swarm optimization (PSO) algorithm. The ideal of global search method of PSO is incorporated into the MFA algorithm.

4.4. Computational complexity

In this section, the computational complexity of the proposed EKH algorithm is discussed and compared with the basic KH algorithm. The computational complexity includes time complexity and space complexity.

4.4.1. Time complexity

The time complexity of KH algorithm mainly depends on two steps which are motion calculation and updating krill positions [60]. Thus, the time complexity can be defined as follows:

$$O(KH) = O(t(O(motion\ \ neighbors) + O(position\ \ update))) \tag{42}$$

$$\begin{aligned} O(KH) &= O(t(n^2 \times d + n \times d)) = O(tn^2 d + tnd) \\ &= O(tn^2 d) \end{aligned} \tag{43}$$

where t is the maximum number of iterations, n is the number of krill, and d is the dimension of given problem.

4.4.2. Space complexity

The space complexity of KH algorithm is the amount of memory space required to solve an instance of the computational problem, which is considered the initialization process of the krill individuals. Therefore, the total space complexity of KH algorithm is $O(n \times d)$ [83].

In the basic KH algorithm, the time complexity is mainly determined by the process of updating krill positions. A new position update method of krill individuals is proposed in the EKH algorithm. Accurately speaking, the presented method moves the krill individual to the current global optimal value in the each iteration which accelerates its convergence speed. Therefore, compared with the basic KH algorithm, the EKH algorithm reduces the

Table 18
Comparison of standard deviation obtained by MFA, MGOA, WCA and BA algorithms using Kapur's entropy.

Image	m	Standard deviation			
		MFA	MGOA	WCA	BA
Baboon	3	8.75432e−006	8.75508e−006	8.75591e−006	8.75667e−006
	4	8.34285e−005	8.34359e−005	8.34420e−005	8.34495e−005
	5	7.46712e−004	7.46769e−004	7.46833e−004	7.46867e−004
	6	0.00141	0.00175	0.00218	0.00252
Donna	3	9.15578e−006	9.15662e−006	9.15723e−006	9.15776e−006
	4	8.59074e−005	8.59178e−005	8.59260e−005	8.59342e−005
	5	7.99172e−004	7.99243e−004	7.99290e−004	7.99337e−004
	6	0.00199	0.00236	0.00268	0.00302
Fruit	3	8.36421e−006	8.36493e−006	8.36576e−006	8.36645e−006
	4	8.21365e−005	8.21446e−005	8.21523e−005	8.21598e−005
	5	7.22103e−004	7.22171e−004	7.22246e−004	7.22280e−004
	6	0.00108	0.00146	0.00180	0.00209
Flower	3	9.47692e−006	9.47751e−006	9.47823e−006	9.47866e−006
	4	8.86849e−005	8.86937e−005	8.87032e−005	8.87084-005
	5	8.15582e−004	8.15652e−004	8.15689e−004	8.15733e−004
	6	0.00256	0.00280	0.00316	0.00328
House	3	8.66768e−006	8.66852e−006	8.66941e−006	8.67036e−006
	4	8.20902e−005	8.21021e−005	8.21089e−005	8.21168e−005
	5	7.26799e−004	7.26853e−004	7.26916e−004	7.26982e−004
	6	0.00125	0.00122	0.00122	0.00122
Pallon	3	8.89634e−006	8.89733e−006	8.89780e−006	8.89867e−006
	4	8.54310e−005	8.54388e−005	8.54476e−005	8.54534e−005
	5	7.60383e−004	7.60455e−004	7.60502e−004	7.60536e−004
	6	0.00148	0.00189	0.00224	0.00261
Sailboat	3	1.24676e−005	1.24760e−005	1.24845e−005	1.24908e−005
	4	9.34237e−005	9.34342e−005	9.34399e−005	9.34476e−005
	5	9.14987e−004	9.15046e−004	9.15088e−004	9.15134e−004
	6	0.00369	0.00395	0.00432	0.00463
Tulips	3	9.06575e−006	9.06645e−006	9.06712e−006	9.06778e−006
	4	8.55929e−005	8.56042e−005	8.56135e−005	8.56190e−005
	5	7.76980e−004	7.77042e−004	7.77083e−004	7.77136e−004
	6	0.00177	0.00231	0.00256	0.00288
Voit	3	8.86529e−006	8.86642e−006	8.86735e−006	8.86812e−006
	4	8.46370e−005	8.46471e−005	8.46558e−005	8.46592e−005
	5	7.63721e−004	7.63788e−004	7.63841e−004	7.63905e−004
	6	0.00229	0.00261	0.00299	0.00346
Yacht	3	9.16093e−006	9.16170e−006	9.16249e−006	9.16338e−006
	4	8.60352e−005	8.60448e−005	8.60503e−005	8.60585e−005
	5	8.02459e−004	8.02534e−004	8.02589e−004	8.02683e−004
	6	0.00236	0.00242	0.00271	0.00303

time-consuming of the process of updating krill position. This significant result demonstrates that the presented EKH algorithm can reduce CPU time to some extent.

4.5. Swarm diversity

Swarm diversity is a key important performance for evolutionary and swarm intelligence algorithms. Diversity is calculated as the average hamming distance for the population through generations. Low diversity is considered for being the main reason for premature convergence. In order to controlling the diversity of KH algorithm and preventing premature convergence, the EHK algorithm is proposed in this paper. Fig. 3 compares the diversity between the presented EKH and the basic KH algorithm on the benchmark functions and the benchmark functions are listed in Table 1. It can be seen form Fig. 3 the diversity of the presented EKH algorithm changed slowly in the early stage but the values is larger than the KH algorithm, which indicated the EKH algorithm has stronger global searching ability and can avoid falling into local optimal solution. In the late stages of search, the diversity of the EKH algorithm is yet higher than the KH algorithm. To sum up, the diversity of the EKH algorithm is better than the KH algorithm.

5. Experiments and results

Experimental results have been presented in this section. Multilevel thresholding for color image segmentation is one of the most widely used techniques that search the optimal thresholding values for each RGB components (i.e., red, green and blue) within the range $[0, L-1]$. The optimal objective function values (f) of color image is equal to the sum of the optimal objective function values of R, G and B components. In this study, based on the EKH algorithm with three different objective functions (Otsu, Kapur and Tsallis entropy) has been employed for color image multilevel thresholding segmentation problem. Firstly, the results of the EKH algorithm are compared with the KH I, KH II and KH IV algorithm. Secondly, the EKH algorithm is compared with the four recent algorithms for color image multilevel thresholding segmentation, modified firefly algorithm (MFA), modified grasshopper optimization algorithm (MGOA), water cycle algorithm (WCA) and Bat algorithm (BA). The performance of the presented algorithm has been evaluated using optimal threshold values, optimal objective values, PSNR, SSIM and standard deviation of the objective values at different levels. The overall results achieved are shown in Tables 3–29 and in Figs. 5–11.

Table 19
Comparison of optimal thresholds and objective values obtained by EKH and KH I algorithms using Tsallis entropy.

m	EKH				KH I			
	R	G	B	f	R	G	B	f
Baboon								
3	74,112,158	54,96,148	89,153,204	**3.88830**	74,112,158	54,96,148	89,153,204	3.88830
4	57,87,123,166	79,121,149,177	70,120,146,193	**4.96277**	57,87,123,166	79,121,149,177	70,120,146,193	4.96277
5	53,83,117,149,198	36,64,93,123,158	32,57,82,118,190	**5.98741**	50,87,120,154,195	39,67,98,126,162	30,61,86,121,193	5.98725
6	39,71,102,128,156,200	29,58,84,107,132,158	39,62,87,121,197,216	**6.99560**	39,71,97,120,165,210	26,58,90,103,137,160	45,66,92,128,192,221	6.99529
Donna								
3	198,213,234	171,188,235	163,176,190	**3.88436**	198,213,234	171,188,235	163,176,190	3.88436
4	156,168,183,197	171,187,232,236	139,153,163,176	**4.95577**	156,168,183,197	171,187,232,236	139,153,163,176	4.95577
5	153,166,182,193,210	93,105,154,184,202	95,111,155,176,205	**5.95896**	155,172,186,196,213	91,108,159,187,204	97,115,159,180,208	5.95881
6	135,148,168,182,194,216	108,116,124,130,144,176	27,48,72,93,128,194	**6.98406**	136,149,168,188,198,223	109,119,128,135,149,184	29,51,75,98,133,201	6.98376
Fruit								
3	48,98,148	54,85,113	54,179,194	**3.88856**	48,98,148	54,85,113	54,179,194	3.88856
4	45,94,139,156	55,87,108,123	23,46,61,71	**4.96267**	45,94,139,156	55,87,108,123	23,46,61,71	4.96267
5	44,90,124,139,149	41,66,96,113,134	25,47,62,73,165	**5.98732**	42,95,128,142,153	45,69,98,116,137	28,51,66,76,169	5.98715
6	45,94,133,148,164,184	51,98,125,143,172,196	32,50,74,92,113,203	**6.99501**	48,96,137,151,167,190	59,102,129,146,177,201	36,57,78,99,116,207	6.99472
Flower								
3	89,157,186	65,112,138	68,127,181	**3.88884**	89,157,186	65,112,138	68,127,181	3.88884
4	93,141,176,193	56,128,141,155	71,127,170,193	**4.96249**	93,141,176,193	56,128,141,155	71,127,170,193	4.96249
5	66,121,152,166,175	46,87,113,131,150	60,113,148,164,175	**5.98758**	69,126,155,171,177	51,89,117,135,154	64,118,153,168,179	5.98743
6	60,95,151,172,184,195	47,88,110,125,141,157	125,207,224,227, 236,244	**6.99557**	65,97,156,174,189,198	49,95,118,126,143,159	133,215,225,229, 237,248	6.99528
House								
3	106,150,158	55,190,197	127,209,220	**3.88885**	106,150,158	55,190,197	127,209,220	3.88885
4	88,149,157,162	78,157,196,207	84,131,193,221	**4.96289**	88,149,157,162	78,157,196,207	84,131,193,221	4.96289
5	88,149,156,162,180	53,87,106,196,203	68,80,89,94,99	**5.98651**	91,147,158,167,184	55,91,109,199,208	73,83,91,96,104	5.98634
6	48,88,131,157,162,180	50,81,98,112,166,204	68,81,89,94,98,104	**6.99427**	52,91,135,160,165,188	55,83,99,115,170,210	71,85,92,97,100,109	6.99397
Pallon								
3	113,144,178	86,151,163	92,136,150	**3.88883**	113,144,178	86,151,163	92,136,150	3.88883
4	51,112,158,182	57,143,160,167	77,127,148,152	**4.96282**	51,112,158,182	57,143,160,167	77,127,148,152	4.96282
5	72,102,120,133,146	59,113,123,135,148	32,46,82,89,157	**5.98507**	75,109,126,136,150	63,117,128,139,153	35,53,84,95,160	5.98490
6	73,103,134,158,172,182	27,48,70,89,117,165	55,77,96,120,150,157	**6.99540**	78,106,139,162,178,189	30,51,72,93,125,168	55,77,96,120,150,157	6.99510
Sailboat								
3	95,130,158	86,162,200	90,167,204	**3.88886**	95,130,158	86,162,200	90,167,204	3.88886
4	94,130,150,170	79,149,195,208	92,166,201,209	**4.96291**	94,130,150,170	79,149,195,208	92,166,201,209	4.96291
5	37,82,106,145,180	71,133,175,188,194	76,133,169,180,185	**5.98749**	40,85,110,152,185	74,138,179,193,198	79,138,173,182,187	5.98734
6	37,74,96,117,149,179	71,132,176,189,195,199	59,114,154,176, 184,187	**6.99566**	39,78,99,124,155,185	77,137,180,190,197,208	62,117,158,179, 188,196	6.99537
Tulips								
3	93,200,227	72,127,155	68,127,163	**3.88881**	93,200,227	72,127,155	68,127,163	3.88881
4	99,181,225,228	45,76,107,139	22,62,91,134	**4.96287**	99,181,225,228	45,76,107,139	22,62,91,134	4.96287
5	48,135,163,179,187	47,79,102,124,154	34,63,81,105,139	**5.98733**	49,141,168,183,195	49,82,106,126,158	35,66,84,108,145	5.98718
6	38,61,88,102,137,168	37,62,87,107,168,203	44,74,106,130,176,214	**6.99551**	41,65,89,109,141,175	41,65,89,113,172,209	46,77,109,135,183,220	6.99521
Voit								
3	96,132,155	83,133,156	90,131,151	**3.88879**	96,132,155	83,133,156	90,131,151	3.88879
4	52,129,147,164	78,123,147,159	47,129,148,155	**4.96260**	52,129,147,164	78,123,147,159	47,129,148,155	4.96260
5	51,83,104,142,191	37,67,93,134,178	30,49,84,111,175	**5.98730**	54,87,109,146,197	39,73,97,136,182	33,51,87,118,180	5.98714
6	7,123,218,221,241,252	31,59,83,102,144,208	24,65,96,124,160,198	**6.99498**	12,127,219,228243,254	35,62,85,108,147,215	28,67,100,129,163,205	6.99467
Yacht								
3	65,111,169	66,129,170	54,99,139	**3.88886**	65,111,169	66,129,170	54,99,139	3.88886
4	75,137,167,190	67,133,161,182	45,87,121,159	**4.96292**	75,137,167,190	67,133,161,182	45,87,121,159	4.96292
5	39,65,94,120,163	65,132,148,160,172	33,63,89,114,133	**5.98749**	43,67,99,127,169	67,137,155,165,175	35,66,92,116,139	5.98734
6	56,117,156,179,190,198	43,70,96,112,131,163	34,62,79,99,126,166	**6.99545**	59,124,159,181,192,204	46,73,98,114,139,169	36,67,82,101,130,174	6.99516

Table 20
Comparison of optimal thresholds and objective values obtained by KH II and KH IV algorithms using Tsallis entropy.

m	KH II				KH IV			
	R	G	B	f	R	G	B	f
Baboon								
3	74,112,158	54,96,148	89,153,204	3.88830	74,112,158	54,96,148	89,153,204	3.88830
4	57,87,123,166	79,121,149,177	70,120,146,193	4.96277	57,87,123,166	79,121,149,177	70,120,146,193	4.96277
5	51,84,115,152,197	37,62,97,125,158	31,56,84,115,191	5.98739	52,85,115,152,196	38,65,99,128,159	32,57,85,117,192	5.98738
6	42,73,103,129,161,203	30,60,85,111,133,161	40,65,89,123,198,219	6.99553	41,75,105,129,158,204	31,59,86,109,134,161	42,65,89,123,199,219	6.99551
Donna								
3	198,213,234	171,188,235	163,176,190	3.88436	198,213,234	171,188,235	163,176,190	3.88436
4	156,168,183,197	171,187,232,236	139,153,163,176	4.95577	156,168,183,197	171,187,232,236	139,153,163,176	4.95577
5	152,168,184,196,211	95,106,157,186,203	97,113,158,178,206	5.95895	154,167,185,195,211	95,105,156,187,205	99,112,157,176,208	5.95894
6	133,146,167,182,193,211	108,116,123,129,140,181	29,51,73,96,133,195	6.98398	137,152,171,185,198,219	109,118,128,134,148,180	30,50,74,97,132,197	6.98396
Fruit								
3	48,98,148	54,85,113	54,179,194	3.88856	48,98,148	54,85,113	54,179,194	3.88856
4	45,94,139,156	55,87,108,123	23,46,61,71	4.96267	45,94,139,156	55,87,108,123	23,46,61,71	4.96267
5	43,93,125,139,150	41,67,98,115,135	27,49,63,74,166	5.98731	46,92,125,142,151	43,68,97,116,136	26,48,65,75,167	5.98730
6	46,97,137,152,165,186	54,99,128,145,173,199	34,53,76,93,115,205	6.99492	47,95,138,149,167,186	54,101,127,144,175,198	35,52,77,93,115,206	6.99490
Flower								
3	89,157,186	65,112,138	68,127,181	3.88884	89,157,186	65,112,138	68,127,181	3.88884
4	93,141,176,193	56,128,141,155	71,127,170,193	4.96249	93,141,176,193	56,128,141,155	71,127,170,193	4.96249
5	67,125,154,168,176	48,89,116,135,151	62,116,151,165,177	5.98756	68,124,157,167,176	47,89,117,133,153	63,115,149,166,178	5.98755
6	61,97,153,177,188,197	50,90,115,128,145,160	130,211,225,229, 237,245	6.99550	60,95,151,172,184,195	50,90,113,128,145,159	131,214,227,232, 239,249	6.99548
House								
3	106,150,158	55,190,197	127,209,220	3.88885	106,150,158	55,190,197	127,209,220	3.88885
4	88,149,157,162	78,157,196,207	84,131,193,221	4.96289	88,149,157,162	78,157,196,207	84,131,193,221	4.96289
5	87,152,158,164,181	51,88,108,199,203	69,82,90,95,101	5.98649	88,149,156,162,180	53,87,106,196,203	71,82,89,95,102	5.98648
6	48,90,132,158,168,185	52,85,102,114,168,205	69,85,91,96,100,108	6.99419	49,90,135,159,166,183	52,83,102,113,168,210	75,87,90,95,99,106	6.99417

(*continued on next page*)

Table 20 (*continued*).

m	KH II				KH IV			
	R	G	B	f	R	G	B	f
Pallon								
3	113,144,178	86,151,163	92,136,150	3.88883	113,144,178	86,151,163	92,136,150	3.88883
4	51,112,158,182	57,143,160,167	77,127,148,152	4.96282	51,112,158,182	57,143,160,167	77,127,148,152	4.96282
5	71,105,122,135,147	62,114,125,136,148	30,47,83,92,158	5.98505	73,104,123,136,147	61,117,125,136,151	33,49,85,92,158	5.98504
6	74,105,138,159,175,183	29,450,73,91,121,167	58,79,98,123,152,159	6.99532	75,104,137,161,173,186	29,49,72,90,120,167	58,78,98,122,151,160	6.99530
Sailboat								
3	95,130,158	86,162,200	90,167,204	3.88886	95,130,158	86,162,200	90,167,204	3.88886
4	94,130,150,170	79,149,195,208	92,166,201,209	4.96291	94,130,150,170	79,149,195,208	92,166,201,209	4.96291
5	36,84,109,147,181	72,134,178,189,195	78,135,170,181,186	5.98747	39,85,108,147,183	72,136,177,189,197	78,136,171,181,188	5.98746
6	39,75,98,118,151,182	73,136,177,188,197,201	60,116,156,177, 185,189	6.99558	38,77,99,119,153181	73,133,178,192,197,201	61,117,156,177, 185,190	6.99556
Tulips								
3	93,200,227	72,127,155	68,127,163	3.88881	93,200,227	72,127,155	68,127,163	3.88881
4	99,181,225,228	45,76,107,139	22,62,91,134	4.96287	99,181,225,228	45,76,107,139	22,62,91,134	4.96287
5	46,137,165,180,189	49,80,105,126,156	35,66,84,107,142	5.98732	49,138,166,180,191	49,81,104,127,157	37,65,83,109,142	5.98731
6	39,66,91,105,141,173	38,63,90,110,172,207	45,76,107,132,179,216	6.99544	41,65,90,105,140,174	39,65,89,110,171,205	45,75,107,135,179,215	6.99542
Voit								
3	96,132,155	83,133,156	90,131,151	3.88879	96,132,155	83,133,156	90,131,151	3.88879
4	52,129,147,164	78,123,147,159	47,129,148,155	4.96260	52,129,147,164	78,123,147,159	47,129,148,155	4.96260
5	52,85,106,145,192	38,68,95,136,179	31,51,85,113,178	5.98729	53,86,105,146,193	39,70,95,136,181	32,50,87,115,178	5.98728
6	9,124,219,223,244,252	32,61,84,104,145,209	25,66,97,125,162,199	6.99491	8,124,220,224,245,252	31,63,85,104,146,208	26,65,99,125,162,200	6.99489
Yacht								
3	65,111,169	66,129,170	54,99,139	3.88886	65,111,169	66,129,170	54,99,139	3.88886
4	75,137,167,190	67,133,161,182	45,87,121,159	4.96292	75,137,167,190	67,133,161,182	45,87,121,159	4.96292
5	38,67,96,121,165	66,133,152,162,173	35,64,90,116,134	5.98748	42,67,96,121,166	68,135,149,165,174	34,65,95,117,134	5.98747
6	57,119,158,180,191,201	44,73,97,115,133,166	36,63,82,101,129,168	6.99537	57,121,157,180,193,203	46,72,99,115,134,165	35,64,83,103,128,169	6.99535

Table 21
Comparison of optimal thresholds and objective values obtained by MFA and MGOA algorithms using Tsallis entropy.

m	MFA				MGOA			
	R	G	B	f	R	G	B	f
Baboon								
3	74,112,158	54,96,148	89,153,204	3.88830	74,112,158	54,96,148	89,153,204	3.88830
4	57,87,123,166	79,121,149,177	70,120,146,193	4.96277	57,87,123,166	79,121,149,177	70,120,146,193	4.96277
5	53,83,117,149,198	36,64,93,123,158	32,57,82,118,190	5.98741	53,84,117,149,198	36,64,93,124,158	33,57,82,118,190	5.987405
6	39,71,102,129,156,200	29,58,84,107,132,158	39,62,87,121,197,216	6.99559	40,71,103,128,157,200	29,58,84,108,133,159	39,63,87,123,197,216	6.99558
Donna								
3	198,213,234	171,188,235	163,176,190	3.88436	198,213,234	171,188,235	163,176,190	3.88436
4	156,168,183,197	171,187,232,236	139,153,163,176	4.95577	156,168,183,197	171,187,232,236	139,153,163,176	4.95577
5	153,166,182,193,210	93,105,154,184,202	95,111,155,176,205	5.95896	156,172,186,196,213	91,108,159,189,204	97,115,159,180,209	5.958805
6	135,148,168,182, 194,217	108,116,124,130,144,176	27,48,72,93,128,194	6.98405	138,149,168,189, 198,223	109,120,129,135,149,185	29,51,76,98,134,202	6.98374
Fruit								
3	48,98,148	54,85,113	54,179,194	3.88856	48,98,148	54,85,113	54,179,194	3.88856
4	45,94,139,156	55,87,108,123	23,46,61,71	4.96267	45,94,139,156	55,87,108,123	23,46,61,71	4.96267
5	44,90,124,139,149	41,66,96,113,134	25,47,62,73,165	5.98732	44,92,124,139,149	41,66,96,114,134	26,47,62,73,165	5.987315
6	45,94,133,148,164,184	52,98,125,143,172,196	32,50,74,92,113,203	6.99500	46,95,133,149,164,184	52,98,126,143,173,196	33,50,75,92,114,203	6.99499
Flower								
3	89,157,186	65,112,138	68,127,181	3.88884	89,157,186	65,112,138	68,127,181	3.88884
4	93,141,176,193	56,128,141,155	71,127,170,193	4.96249	93,141,176,193	56,128,141,155	71,127,170,193	4.96249
5	66,121,152,166,175	46,87,113,131,150	60,113,148,164,175	5.98758	66,121,153,166,175	47,87,113,131,150	60,113,148,164,177	5.987575
6	60,95,151,172,184,195	47,88,110,125,141,157	126,207,224,227, 236,244	6.99557	60,95,152,172,185,196	48,89,110,126,141,157	126,207,225,227, 237,244	6.99555
House								
3	106,150,158	55,190,197	127,209,220	3.88885	106,150,158	55,190,197	127,209,220	3.88885
4	88,149,157,162	78,157,196,207	84,131,193,221	4.96289	88,149,157,162	78,157,196,207	84,131,193,221	4.96289
5	88,149,156,162,180	53,87,106,196,203	68,80,89,94,99	5.98651	89,149,156,162,180	53,87,108,196,203	68,80,89,94,100	5.986505
6	48,88,131,157,162,180	50,81,98,112,166,204	68,81,89,94,98,104	6.99426	46,88,131,157,163,180	49,81,98,111,167,204	68,80,89,93,99,104	6.99425
Pallon								
3	113,144,178	86,151,163	92,136,150	3.88883	113,144,178	86,151,163	92,136,150	3.88883
4	51,112,158,182	57,143,160,167	77,127,148,152	4.96282	51,112,158,182	57,143,160,167	77,127,148,152	4.96282
5	72,102,120,133,146	59,113,123,135,148	32,46,82,89,157	5.98507	71,102,120,133,146	59,112,123,135,148	32,47,82,89,157	5.985065
6	73,103,134,158,172, 183	27,48,70,89,117,165	55,77,96,120,150,157	6.99539	74,103,135,159,172, 182	28,48,70,88,117,164	56,77,96,121,150,158	6.99538
Sailboat								
3	95,130,158	86,162,200	90,167,204	3.88886	95,130,158	86,162,200	90,167,204	3.88886
4	94,130,150,170	79,149,195,208	92,166,201,209	4.96291	94,130,150,170	79,149,195,208	92,166,201,209	4.96291
5	37,82,106,145,180	71,133,175,188,194	76,133,169,180,185	5.98749	40,84,110,152,185	74,138,179,192,198	79,138,173,182,188	5.987335
6	37,74,96,117,149,179	71,132,177,189,195,199	59,114,154,176,184,187	6.99566	41,78,99,124,156,185	76,137,180,189,197,207	63,117,158,178188,197	6.99535
Tulips								
3	93,200,227	72,127,155	68,127,163	3.88881	93,200,227	72,127,155	68,127,163	3.88881
4	99,181,225,228	45,76,107,139	22,62,91,134	4.96287	99,181,225,228	45,76,107,139	22,62,91,134	4.96287
5	48,135,163,179,187	47,79,102,124,154	34,63,81,105,139	5.98733	48,135,163,179,186	47,79,103,124,154	35,63,81,105,139	5.987325
6	38,61,88,102,137,168	37,62,87,107,168,203	45,74,106,130,176,214	6.99550	39,61,88,103,138,168	37,63,87,108,168,205	45,74,106,130,177,215	6.99549
Voit								
3	96,132,155	83,133,156	90,131,151	3.88879	96,132,155	83,133,156	90,131,151	3.88879
4	52,129,147,164	78,123,147,159	47,129,148,155	4.96260	52,129,147,164	78,123,147,159	47,129,148,155	4.96260
5	51,83,104,142,191	37,67,93,134,178	30,49,84,111,175	5.98730	51,82,104,142,191	38,67,93,134,178	30,49,84,111,177	5.987295
6	8,123,218,221,241,252	31,59,83,102,144,208	24,65,96,124,160,198	6.99497	8,125,218,221,242,252	31,59,84,103,144,209	25,65,97,124,160,199	6.99497
Yacht								
3	65,111,169	66,129,170	54,99,139	3.88886	65,111,169	66,129,170	54,99,139	3.88886
4	75,137,167,190	67,133,161,182	45,87,121,159	4.96292	75,137,167,190	67,133,161,182	45,87,121,159	4.96292
5	39,65,94,120,163	65,132,148,160,172	33,63,89,114,133	5.98749	39,65,94,121,163	65,132,147,160,172	33,63,89,114,132	5.987485
6	56,117,156,179, 190,198	43,71,96,112,131,163	34,62,79,99,126,166	6.99545	57,117,157,180, 190,198	43,71,96,112,131,165	36,62,79,100,126,167	6.99543

Table 22
Comparison of optimal thresholds and objective values obtained by WCA and BA algorithms using Tsallis entropy.

m	WCA				BA			
	R	G	B	f	R	G	B	f
Baboon								
3	74,112,158	54,96,148	89,153,204	**3.88830**	74,112,158	54,96,148	89,153,204	3.88830
4	57,87,123,166	79,121,149,177	70,120,146,193	**4.96277**	57,87,123,166	79,121,149,177	70,120,146,193	4.96277
5	55,83,117,149,199	36,65,94,123,158	32,57,83,119,190	5.98740	51,83,117,148,198	37,65,93,124,158	32,57,83,118,192	5.98739
6	40,71,103,128,158,200	30,58,84,107,133,160	39,63,87,122,198,217	6.99556	41,72,103,128,157,200	29,58,84,109,133,159	37,61,87,120,197,216	6.99555
Donna								
3	198,213,234	171,188,235	163,176,190	**3.88436**	198,213,234	171,188,235	163,176,190	3.88436
4	156,168,183,197	171,187,232,236	139,153,163,176	**4.95577**	156,168,183,197	171,187,232,236	139,153,163,176	4.95577
5	153,165,182,192,210	94,105,154,184,203	95,111,156,177,205	5.95895	154,167,182,193,211	91,105,152,183,202	97,111,156,176,206	5.95894
6	136,148,169,182,196,216	107,116,124,131,145,177	27,50,72,93,129,196	6.98402	136,148,170,182,195,217	109,116,124,131,145,177	26,48,71,95,127,193	6.98401
Fruit								
3	48,98,148	54,85,113	54,179,194	**3.88856**	48,98,148	54,85,113	54,179,194	3.88856
4	45,94,139,156	55,87,108,123	23,46,61,71	**4.96267**	45,94,139,156	55,87,108,123	23,46,61,71	4.96267
5	44,91,124,139,149	41,67,96,113,134	25,47,63,73,165	5.98731	44,91,126,139,149	41,67,96,114,135	26,47,63,73,166	5.98730
6	47,94,133,149,164,185	51,97,125,143,171,194	33,51,74,93,113,202	6.99497	45,95,133,147,164,187	51,98,126,144,172,197	32,52,74,92,114,205	6.99496
Flower								
3	89,157,186	65,112,138	68,127,181	**3.88884**	89,157,186	65,112,138	68,127,181	3.88884
4	93,141,176,193	56,128,141,155	71,127,170,193	**4.96249**	93,141,176,193	56,128,141,155	71,127,170,193	4.96249
5	67,121,152,167,175	46,88,113,133,150	61,113,149,164,175	5.98757	66,121,151,166,174	46,85,113,131,151	60,114,148,164,176	5.98756
6	62,95,152,173,184,195	47,89,110,125,143,158	127,207,225,227, 236,245	6.99552	62,95,152,172,185,196	47,88,111,127,141,158	126,207,224,228, 236,245	6.99553
House								
3	106,150,158	55,190,197	127,209,220	**3.88885**	106,150,158	55,190,197	127,209,220	3.88885
4	88,149,157,162	78,157,196,207	84,131,193,221	**4.96289**	88,149,157,162	78,157,196,207	84,131,193,221	4.96289
5	88,148,156,163,180	51,87,106,195,203	68,79,89,93,100	5.98650	89,149,156,162,181	54,88,107,197,203	68,80,87,94,100	5.98649
6	47,88,132,157,164,180	50,80,98,111,166,202	68,81,90,94,99,105	6.99423	49,88,131,159,163,182	50,82,98,112,167,205	69,81,89,92,99,106	6.99421
Pallon								
3	113,144,178	86,151,163	92,136,150	**3.88883**	113,144,178	86,151,163	92,136,150	3.88883
4	51,112,158,182	57,143,160,167	77,127,148,152	**4.96282**	51,112,158,182	57,143,160,167	77,127,148,152	4.96282
5	71,102,119,133,146	59,115,123,136,148	33,46,82,90,157	5.98506	71,102,118,133,145	59,114,123,136,148	35,46,82,90,157	5.98505
6	74,105,134,158,172,183	28,48,70,90,118,166	54,77,96,122,150,158	6.99536	72,105,134,158,173,183	27,48,72,90,117,166	56,77,97,120,150,158	6.99535
Sailboat								
3	95,130,158	86,162,200	90,167,204	**3.88886**	95,130,158	86,162,200	90,167,204	3.88886
4	94,130,150,170	79,149,195,208	92,166,201,209	**4.96291**	94,130,150,170	79,149,195,208	92,166,201,209	4.96291
5	38,82,107,145,180	71,134,175,189,194	76,133,168,180,184	5.98748	37,82,107,145,182	72,135,175,187,194	76,133,170,180,186	5.98747
6	38,74,96,118,150,180	70,132,175,190,195,199	59,116,155,176, 184,189	6.99562	37,75,96,117,150,182	72,133,177,189,195,201	58,114,153,176, 184,188	6.99561
Tulips								
3	93,200,227	72,127,155	68,127,163	**3.88881**	93,200,227	72,127,155	68,127,163	3.88881
4	99,181,225,228	45,76,107,139	22,62,91,134	**4.96287**	99,181,225,228	45,76,107,139	22,62,91,134	4.96287
5	48,135,163,180,189	47,80,102,124,155	32,63,81,104,139	5.98732	49,136,163,179,187	47,78,102,126,155	34,64,81,105,137	5.98731
6	37,61,89,102,137,170	37,62,88,107,169,204	44,75,106,132,176,215	6.99548	40,62,89,102,138,169	37,63,87,109,168,205	45,74,107,131,178,214	6.99545
Voit								
3	96,132,155	83,133,156	90,131,151	3.88879	96,132,155	83,133,156	90,131,151	3.88879
4	52,129,147,164	78,123,147,159	47,129,148,155	4.96260	52,129,147,164	78,123,147,159	47,129,148,155	4.96260
5	50,83,104,142,192	37,69,93,134,178	30,50,84,112,175	5.98729	51,82,104,140,191	37,68,94,134,180	32,49,84,112,175	5.98728
6	7,123,217,221,240,254	31,57,83,101,144,207	24,66,96,125,160,196	6.99494	8,125,218,221,241,253	32,60,85,102,144,210	25,65,96,126,161,198	6.99493
Yacht								
3	65,111,169	66,129,170	54,99,139	**3.88886**	65,111,169	66,129,170	54,99,139	3.88886
4	75,137,167,190	67,133,161,182	45,87,121,159	**4.96292**	75,137,167,190	67,133,161,182	45,87,121,159	4.96292
5	39,65,94,120,163	65,132,148,160,172	33,63,89,114,133	5.98748	39,65,94,122,164	65,131,148,162,172	35,63,89,116,133	5.98747
6	56,119,156,180,191,199	42,70,96,113,132,163	34,64,79,99,125,167	6.99541	56,115,156,178,190,200	42,70,96,111,130,163	35,62,79,100,128,167	6.99540

5.1. Image dataset

In order to obtain better comparison of the segmented results, the experiments have been done on ten different color well-known test images. The test images and their corresponding histograms are shown in Fig. 4. All the images are 512×512 size. For color image multilevel thresholding segmentation, the number of thresholds (m) used in this experiment are 3-level, 4-level, 5-level and 6-level. Basically, the swarm intelligence optimization algorithms have randomized behavior. Therefore, all the experiments were repeated for 30 times for each image and for each level.

5.2. Experiment setup

In this paper, the EKH, KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms have been used individually for color image multilevel thresholding segmentation. Three well-known objective functions named Otsu's method, Kapur's entropy and Tsallis entropy are employed as objective functions. In all the experiments, the population size was set to be 25 and the maximum number of iteration is 100. The other parameter of the KH algorithm, foraging speed V_f was set to 0.02, $D_{\max}$ was set to 0.005, $N_{\max}$ was set to 0.01 and α was set to 0.01 [60]. The parameters of MFA [70], MGOA [73], WCA [72] and BA [71] are shown in Table 2. The experiments are developed using MATLAB Release 2014a.

5.3. Segmented image quality measures

For compare the performance of KH I, KH II, KH IV, MFA, MGOA, WCA, BA and EKH algorithm based multilevel thresholding using three different objective functions, in addition to the optimal threshold values and objective function values, the parameters of peak signal to noise ratio (PSNR) and structural similarity index (SSIM) values have been also used in this paper. The PSNR and SSIM values are average values of 30 times, and the optimal threshold values are best values of 30 times.

The term of the peak signal to noise ratio (PSNR) is an important performance indicator for multilevel thresholding segmentation which is used to compute the peak signal to noise ratio the between the original image and the segmented image. The PSNR value (in dB) is calculated as [84]:

$$PSNR\ (in\ dB) = 20\log_{10}(255/RMSE) \tag{44}$$

where

$$RMSE = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(I\left(i,j\right)-I'\left(i,j\right)\right)^2}{MN}} \tag{45}$$

Table 23
Comparison of PSNR and SSIM obtained by EKH, KH I, KH II and KH IV algorithms using Tsallis entropy.

Image	m	EKH		KH I		KH II		KH IV	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Baboon	3	**16.81210**	**0.82788**	16.81210	0.82788	16.81210	0.82788	16.81210	0.82788
	4	**16.81309**	**0.85707**	16.81309	0.85707	16.81309	0.85707	16.81309	0.85707
	5	**16.81508**	**0.90692**	16.81402	0.90372	16.81453	0.90589	16.81442	0.90542
	6	**16.81734**	**0.94376**	16.81489	0.94104	16.81655	0.96283	16.81626	0.94273
Donna	3	**20.12316**	**0.80656**	20.12316	0.80656	20.12316	0.80656	20.12316	0.80656
	4	**20.14389**	**0.85887**	20.14389	0.85887	20.14389	0.85887	20.14389	0.85887
	5	**20.15091**	**0.89832**	20.14968	0.89552	20.15026	0.89760	20.15008	0.89784
	6	**20.16023**	**0.91674**	20.15783	0.91402	20.15940	0.91577	20.15898	0.91540
Fruit	3	**21.75392**	**0.84863**	21.75392	0.84863	21.75392	0.84863	21.75392	0.84863
	4	**21.75532**	**0.85375**	21.75532	0.85375	21.75532	0.85375	21.75532	0.85375
	5	**21.75819**	**0.90391**	21.75684	0.90076	21.75748	0.90298	21.75730	0.90249
	6	**21.76136**	**0.93295**	21.75892	0.93024	21.76017	0.93207	21.75980	0.93163
Flower	3	**18.65681**	**0.84106**	18.65681	0.84106	18.65681	0.84106	18.65681	0.84106
	4	**18.65694**	**0.88712**	18.65694	0.88712	18.65694	0.88712	18.65694	0.88712
	5	**18.65862**	**0.92637**	18.65708	0.92335	18.65794	0.92548	18.65775	0.92495
	6	**18.66105**	**0.94352**	18.65859	0.94087	18.66018	0.94256	18.65979	0.94216
House	3	**15.37162**	**0.79738**	15.37162	0.79738	15.37162	0.79738	15.37162	0.79738
	4	**15.37295**	**0.85269**	15.37295	0.85269	15.37295	0.85269	15.37295	0.85269
	5	**15.37583**	**0.90398**	15.37445	0.90387	15.37521	0.90312	15.37503	0.90349
	6	**15.37903**	**0.92494**	15.37649	0.92223	15.37812	0.92405	15.37770	0.92368
Pallon	3	**18.35033**	**0.80845**	18.35033	0.80845	18.35033	0.80845	18.35033	0.80845
	4	**18.35261**	**0.83449**	18.35104	0.83449	18.35261	0.83449	18.35261	0.83449
	5	**18.35752**	**0.90277**	18.35601	0.90031	18.35694	0.90202	18.35677	0.90159
	6	**18.35906**	**0.92670**	18.35671	0.92396	18.35820	0.92582	18.35782	0.92532
Sailboat	3	**16.46198**	**0.84848**	16.46198	0.84848	16.46198	0.84848	16.46198	0.84848
	4	**16.46393**	**0.88218**	16.46393	0.88218	16.46393	0.88218	16.46393	0.88218
	5	**16.46723**	**0.90137**	16.46582	0.89845	16.46655	0.90053	16.46637	0.90014
	6	**16.47356**	**0.92678**	16.47118	0.92402	16.47277	0.92584	16.47238	0.92545
Tulips	3	**17.44468**	**0.86061**	16.46198	0.84848	16.46198	0.84848	16.46198	0.84848
	4	**17.44657**	**0.88059**	17.44657	0.88059	17.44657	0.88059	17.44657	0.88059
	5	**17.44956**	**0.92189**	17.44801	0.91894	17.44889	0.92102	17.44867	0.92058
	6	**17.45636**	**0.94150**	17.45389	0.94078	17.45547	0.94055	17.45511	0.94004
Voit	3	**18.97068**	**0.87030**	18.97068	0.87030	18.97068	0.87030	18.97068	0.87030
	4	**18.97151**	**0.87976**	18.97151	0.87976	18.97151	0.87976	18.97151	0.87976
	5	**18.97356**	**0.90168**	18.97193	0.89874	18.97290	0.90085	18.97271	0.90037
	6	**18.97586**	**0.93143**	18.97342	0.92865	18.97500	0.93051	18.97461	0.93006
Yacht	3	**18.49558**	**0.85494**	18.49558	0.85494	18.49558	0.85494	18.49558	0.85494
	4	**18.49603**	**0.88351**	18.49603	0.88351	18.49603	0.88351	18.49603	0.88351
	5	**18.49848**	**0.93348**	18.49685	0.93059	18.49786	0.93067	18.49767	0.93020
	6	**18.50369**	**0.95320**	18.50124	0.95042	18.50281	0.95228	18.50247	0.95175

where, M, N is the size of the image, I is the original image, and I' is the segmented image. A higher PSNR value indicates better quality of the segmented result.

The parameter of structural similarity (SSIM) describes the structural similarity of the original image and the segmented image. The SSIM index can be defined as [85,86]:

$$SSIM(I, I') = \frac{(2\mu_I \mu_{I'} + c_1)(2\sigma_{II'} + c_2)}{(\mu_I^2 + \mu_{I'}^2 + c_1)(\sigma_I^2 + \sigma_{I'}^2 + c_2)} \tag{46}$$

where μ_I is the average of I, $\mu_{I'}$ is the average of I', σ_I^2 is the variance of I, $\sigma_{I'}^2$ is the variance of I, $\sigma_{II'}$ is the covariance of I and I', two variables $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ stabilize the division with weak denominator, L is the dynamic range of the pixel-values, $k_1 = 0.01$ and $k_2 = 0.03$ by default [87].

To eliminate stochastic discrepancy, the experiments are independently carried out 30 times for each image and for each threshold value. To analyze the stability of the krill herd algorithms for color image multilevel segmentation problem, the mean μ and standard deviation *std* are defined as

$$\mu = \frac{\sum_{i=1}^{k} \sigma_i}{k} \tag{47}$$

$$std = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(\sigma_i - \mu)^2} \tag{48}$$

where k is the number of runs for each algorithm ($k = 30$), σ_i is the best objective function value obtained by the$i$th run of the algorithm. The standard deviation values of the best objective function obtained by Otsu's method, Kapur's entropy and Tsallis entropy based KH I, KH II, KH IV, MFA, MGOA, WCA, BA and EKH algorithms for 30 runs are given in Tables 9, 10, 17, 18, 25 and 26, respectively.

5.4. Experiment 1: Maximizing between-class variance

In the first part of experiments, the between-class variance was used as the objective function that was maximized based on the KH I, KH II, KH IV, MFA, MGOA, WCA, BA and EKH algorithm for color image multilevel segmentation problem to search optimal thresholding values. The method was applied to the ten color test images. The objective function values and optimal threshold values of R, G, and B components obtained by KH I, KH II, KH

Table 24
Comparison of PSNR and SSIM obtained by MFA, MGOA, WCA and BA algorithms using Tsallis entropy.

Image	m	MFA		MGOA		WCA		BA	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Baboon	3	16.81210	0.82788	16.81210	0.82788	16.81210	0.82788	16.81210	0.82788
	4	16.81309	0.85707	16.81309	0.85707	16.81309	0.85707	16.81309	0.85707
	5	16.81508	0.90692	16.81497	0.90678	**16.81508**	**0.90692**	**16.81508**	**0.90692**
	6	16.81730	0.94374	16.81706	0.94340	**16.81734**	**0.94376**	**16.81734**	**0.94376**
Donna	3	20.12316	0.80656	20.12316	0.80656	20.12316	0.80656	20.12316	0.80656
	4	20.14389	0.85887	20.14389	0.85887	20.14389	0.85887	20.14389	0.85887
	5	20.15091	0.89832	20.15079	0.89818	**20.15091**	**0.89832**	**20.15091**	**0.89832**
	6	20.16017	0.91663	20.15994	0.91640	**20.16023**	**0.91674**	**20.16023**	**0.91674**
Fruit	3	21.75392	0.84863	21.75392	0.84863	21.75392	0.84863	21.75392	0.84863
	4	21.75532	0.85375	21.75532	0.85375	21.75532	0.85375	21.75532	0.85375
	5	21.75819	0.90391	21.75808	0.90377	**21.75819**	**0.90391**	**21.75819**	**0.90391**
	6	21.76129	0.93286	21.76109	0.93258	**21.76136**	**0.93295**	**21.76136**	**0.93295**
Flower	3	18.65681	0.84106	18.65681	0.84106	18.65681	0.84106	18.65681	0.84106
	4	18.65694	0.88712	18.65694	0.88712	18.65694	0.88712	18.65694	0.88712
	5	18.65862	0.92637	18.65852	0.92633	**18.65862**	**0.92637**	**18.65862**	**0.92637**
	6	18.66096	0.94342	18.66079	0.94316	**18.66105**	**0.94352**	**18.66105**	**0.94352**
House	3	15.37162	0.79738	15.37162	0.79738	15.37162	0.79738	15.37162	0.79738
	4	15.37295	0.85269	15.37295	0.85269	15.37295	0.85269	15.37295	0.85269
	5	15.37583	0.90398	15.37572	0.90383	**15.37583**	**0.90398**	**15.37583**	**0.90398**
	6	15.37898	0.92485	15.37874	0.92460	**15.37903**	**0.92494**	**15.37903**	**0.92494**
Pallon	3	18.35033	0.80845	18.35033	0.80845	18.35033	0.80845	18.35033	0.80845
	4	18.35261	0.83449	18.35104	0.83449	18.35261	0.83449	18.35261	0.83449
	5	18.35752	0.90277	18.35741	0.90265	**18.35752**	**0.90277**	**18.35752**	**0.90277**
	6	18.35901	0.92662	18.35877	0.92636	**18.35906**	**0.92670**	**18.35906**	**0.92670**
Sailboat	3	16.46198	0.84848	16.46198	0.84848	16.46198	0.84848	16.46198	0.84848
	4	16.46393	0.88218	16.46393	0.88218	16.46393	0.88218	16.46393	0.88218
	5	16.46723	0.90137	16.46713	0.90125	**16.46723**	**0.90137**	**16.46723**	**0.90137**
	6	16.47348	0.92674	16.47328	0.92641	**16.47356**	**0.92678**	**16.47356**	**0.92678**
Tulips	3	17.44468	0.86061	16.46198	0.84848	16.46198	0.84848	16.46198	0.84848
	4	17.44657	0.88059	17.44657	0.88059	17.44657	0.88059	17.44657	0.88059
	5	17.44956	0.92189	17.44945	0.92177	**17.44956**	**0.92189**	**17.44956**	**0.92189**
	6	17.45635	0.94136	17.45608	0.94114	**17.45636**	**0.94150**	**17.45636**	**0.94150**
Voit	3	18.97068	0.87030	18.97068	0.87030	18.97068	0.87030	18.97068	0.87030
	4	18.97151	0.87976	18.97151	0.87976	18.97151	0.87976	18.97151	0.87976
	5	18.97356	0.90168	18.97345	0.90154	**18.97356**	**0.90168**	**18.97356**	**0.90168**
	6	18.97579	0.93140	18.97550	0.93108	**18.97586**	**0.93143**	**18.97586**	**0.93143**
Yacht	3	18.49558	0.85494	18.49558	0.85494	18.49558	0.85494	18.49558	0.85494
	4	18.49603	0.88351	18.49603	0.88351	18.49603	0.88351	18.49603	0.88351
	5	18.49848	0.93348	18.49839	0.93335	**18.49848**	**0.93348**	**18.49848**	**0.93348**
	6	18.50362	0.95315	18.50340	0.95286	**18.50369**	**0.95320**	**18.50369**	**0.95320**

IV and EKH algorithm are presented in Tables 3–4, considering four different threshold values $m = 3, 4, 5, 6$. The results of MFA, MGOA, WCA and BA algorithm are shown in Tables 5–6.

Furthermore, the PSNR and SSIM values acquired using all algorithms are shown in Tables 7–8. Tables 9–10 show the standard deviation values of the best objective function for the EKH, KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithm. Figs. 5 and 6 give the segmented results based EKH algorithm using between-class variance for the ten color test images with various threshold values ($m = 3$–6).

5.5. Experiment 2: Maximizing Kapur's entropy

In the second part of experiments, the Kapur's entropy has been given to the KH I, KH II, KH IV, MFA, MGOA, WCA, BA and EKH algorithm as the objective function to be maximized. The number of thresholds, optimal threshold values and corresponding objective function obtained from the eight algorithms over the entire test images are depicted in Tables 11–14. Tables 15–16 list the values of PSNR (dB) and SSIM of the ten test images applying all algorithms. Tables 17–18 give the standard deviation values of the best objective function for KH I, KH II, KH IV, MFA, MGOA, WCA, BA and EKH algorithm. Further, Figs. 7 and 8 show the segmented images for four different threshold values $m = 3, 4, 5, 6$ using Kapur's entropy function based on the EKH algorithm for ten test color images, respectively.

5.6. Experiment 3: Maximizing Tsallis entropy

In the third part of experiments, the performance evaluation of Tsallis entropy for color image multilevel segmentation is discussed. The number of thresholds, optimal threshold values and corresponding objective function based on Tsallis entropy using each evaluated algorithms are reported in Tables 19–22 for $m = 3, 4, 5, 6$. The PSNR (dB) and SSIM values obtained with the presented EKH algorithm are listed in Tables 23–24 and compared with the results obtained by using KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms respectively. The standard deviation values of the best objective function for all the algorithms are shown in Tables 25 and 26. Figs. 9 and 10 represented the segmented results for various threshold levels ($m = 3$–6) using the EKH algorithm.

5.7. Results analysis

In this section, two comparisons are made of EKH, KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms based on Otsu's method, Kapur's entropy and Tsallis entropy in solving color image multilevel segmentation problem.

Firstly, the segmented performance of EKH algorithm are compared with KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms in terms of optimal threshold values, objective function

Table 25
Comparison of standard deviation obtained by EKH, KH I, KH II and KH IV algorithms using Tsallis entropy.

Image	m	Standard deviation			
		EKH	KH I	KH II	KH IV
Baboon	3	**1.39576e−005**	2.37890e−4	2.11246e−4	1.98962e−4
	4	**2.48932e−004**	0.00387	0.00336	0.00278
	5	**0.00441**	0.00985	0.00942	0.00876
	6	**0.00869**	0.08493	0.08058	0.07407
Donna	3	**2.15691e−005**	2.98872e−4	2.65158e−4	2.35764e−4
	4	**2.87872e−004**	0.00586	0.00489	0.00504
	5	**0.00489**	0.01198	0.01007	0.01020
	6	**0.00899**	0.08765	0.08577	0.08584
Fruit	3	**1.24873e−005**	3.36740e−4	3.01544e−4	2.89673e−4
	4	**1.40741e−004**	0.00308	0.00123	0.00189
	5	**0.00467**	0.00902	0.00645	0.00788
	6	**0.00876**	0.07652	0.05890	0.06263
Flower	3	**2.42542e−005**	2.87633e−4	2.70345e−4	2.75487e−4
	4	**3.12475e−004**	0.00675	0.00438	0.00525
	5	**0.00564**	0.01359	0.00986	0.01017
	6	**0.00967**	0.08245	0.07035	0.07358
House	3	**1.30674e−005**	2.31783e−4	1.97652e−4	2.06577e−4
	4	**2.43670e−004**	0.00289	0.00167	0.00202
	5	**0.00502**	0.00988	0.00834	0.00886
	6	**0.00903**	0.08464	0.07865	0.08135
Pallon	3	**1.67534e−005**	3.83466e−4	2.75692e−4	2.93435e−4
	4	**1.70336e−004**	0.00378	0.00239	0.00281
	5	**0.00523**	0.00902	0.00788	0.00814
	6	**0.00852**	0.07560	0.06289	0.06573
Sailboat	3	**2.98776e−005**	3.87654e−4	3.45693e−4	3.56832e−4
	4	**3.36712e−004**	0.00702	0.00538	0.00593
	5	**0.00713**	0.02046	0.01356	0.01445
	6	**0.00899**	0.09543	0.08867	0.08924
Tulips	3	**1.80564e−005**	2.05673e−4	2.00782e−4	2.02320e−4
	4	**2.72356e−004**	0.00560	0.00267	0.00335
	5	**0.00654**	0.02045	0.01022	0.00983
	6	**0.00876**	0.09673	0.08974	0.08996
Voit	3	**1.52367e−005**	2.67823e−4	2.34275e−4	2.39874e−4
	4	**2.60347e−004**	0.00574	0.00408	0.00436
	5	**0.00560**	0.01245	0.00976	0.01033
	6	**0.00967**	0.08964	0.08543	0.08432
Yacht	3	**2.20448e−005**	2.97425e−4	2.65435e−4	2.65566e−4
	4	**2.99032e−004**	0.00782	0.00467	0.00534
	5	**0.00408**	0.01346	0.00978	0.01025
	6	**0.00895**	0.09347	0.08760	0.08845

values, Peak signal to noise ratio (PSNR) values, Structural similarity index (SSIM) values and standard deviation values of the best objective function. As the stochastic nature of meta-heuristic optimization algorithms, the experiments are conducted over 30 runs.

The objective function values of Otsu, Kapur's entropy and Tsallis entropy for all the algorithms are shown in Tables 3–6, Tables 11–14 and Tables 19–22, respectively. It can been seen from the tables that the EKH algorithm based methods give the best values than the KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms. For example, in the case of Baboon image segmentation using Otsu method (for $m = 6$), the objective function values are 8.77985, 8.77563, 8.77808, 8.77805, 8.77976, 8.77945, 8.77844 and 8.77813 for EKH, KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms, respectively. These experimental results show that the EKH algorithm can obtain better objective function values than the other seven algorithms.

In order to test the stability of the presented EKH algorithm, the standard deviation *std* is also used in this paper. The *std* values of EKH, KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms based on three methods are listed in Tables 9–10, 17–18 and 25–26. From the tables, it can be seen that the EKH algorithm based all methods has lower *std* values compared with the other algorithms, which shows the better stability of the EKH algorithm. In addition to the above performance indicators, PSNR and SSIM are used for all the algorithms. The higher values of the two indicators show the better segmentation performance. The PSNR and SSIM values obtained by all the algorithms are presented in Tables 7–8, Tables 15–16 and Tables 23–24, respectively. It is can be seen that the EKH algorithm based on Otsu, Kapur's entropy and Tsallis entropy obtains better PSNR and SSIM values than the other algorithms for $m = 5, 6$. The results show that the EKH algorithm has precise search ability.

The process of searching optimal thresholding values for color image multilevel thresholding segmentation can be considered as a constrained optimization problem. The thresholding value determines the accuracy of image segmentation. Therefore, the quality of image segmentation is based on the performance of the meta-heuristic algorithms. The proposed EKH algorithm presented a new linear decreasing inertia weights (ω_n, ω_f), an adaptive update scheme of C^{best} and a new position update method of krill individuals that can better balance the exploration and exploitation compared with the other algorithms. So the presented EKH algorithm is an efficient technique for color image multilevel thresholding segmentation.

Based on the result of the above analysis, the EKH algorithm using Otsu's method, Kapur's entropy and Tsallis entropy has better performance than the KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithms for color image multilevel segmentation problem.

Table 26
Comparison of standard deviation obtained by MFA, MGOA, WCA and BA algorithms using Tsallis entropy.

Image	m	Standard deviation			
		MFA	MGOA	WCA	BA
Baboon	3	1.39578e−005	1.39684e−005	1.39764e−005	1.39858e−005
	4	2.48936e−004	2.49013e−004	2.49088e−004	2.49157e−004
	5	0.00447	0.00485	0.00532	0.00587
	6	0.00871	0.00905	0.00947	0.00989
Donna	3	2.15693e−005	2.15805e−005	2.15893e−005	2.15966e−005
	4	2.87879e−004	2.87941e−004	2.88012e−004	2.88086e−004
	5	0.00492	0.00527	0.00563	0.00602
	6	0.00902	0.00932	0.00961	0.00989
Fruit	3	1.24882e−005	1.24964e−005	1.25052e−005	1.25137e−005
	4	1.40745e−004	1.40807e−004	1.40868e−004	1.40940e−004
	5	0.00469	0.00499	0.00540	0.00565
	6	0.00879	0.00921	0.00966	0.00992
Flower	3	2.42548e−005	2.42643e−005	2.42727e−005	2.42812e−005
	4	3.12479e−004	3.12552e−004	3.12616e−004	3.12670e−004
	5	0.00568	0.00613	0.00658	0.00706
	6	0.00978	0.01013	0.01056	0.01078
House	3	1.30675e−005	1.30785e−005	1.30846e−005	1.30933e−005
	4	2.43673e−004	2.43751e−004	2.43824e−004	2.43885e−004
	5	0.00505	0.00545	0.00588	0.00641
	6	0.00911	0.00955	0.00994	0.01039
Pallon	3	1.67532e−005	1.67662e−005	1.67698e−005	1.67756e−005
	4	1.70343e−004	1.70398e−004	1.70466e−004	1.70513e−004
	5	0.00527	0.00561	0.00598	0.00636
	6	0.00862	0.00904	0.00951	0.01002
Sailboat	3	2.98779e−005	2.98865e−005	2.98972e−005	2.99068e−005
	4	3.36718e−004	3.36785e−004	3.36867e−004	3.36975e−004
	5	0.00716	0.00755	0.00787	0.00828
	6	0.00908	0.00941	0.00979	0.01022
Tulips	3	1.80570e−005	1.80680e−005	1.80768e−005	1.80856e−005
	4	2.72363e−004	2.72433e−004	2.72490e−004	2.72564e−004
	5	0.00658	0.00690	0.00746	0.00782
	6	0.00889	0.00927	0.00978	0.01031
Voit	3	1.52374e−005	1.52455e−005	1.52553e−005	1.52657e−005
	4	2.60354e−004	2.60415e−004	2.60473e−004	2.60547e−004
	5	0.00561	0.00598	0.00638	0.00684
	6	0.00979	0.01023	0.01058	0.01109
Yacht	3	2.20459e−005	2.20570e−005	2.20651e−005	2.20729e−005
	4	2.99035e−004	2.99122e−004	2.99198e−004	2.99269e−004
	5	0.00410	0.00452	0.00487	0.00518
	6	0.00906	0.00921	0.00968	0.01013

Table 27
Comparison of optimal PSNR (dB) and SSIM values obtained by EKH algorithms using OTSU, Kapur's and Tsallis entropy.

Image	m	PSNR (OTSU) EKH	PSNR (Kapur) EKH	PSNR (Tsallis) EKH	SSIM (OTSU) EKH	SSIM (Kapur) EKH	SSIM (Tsallis) EKH
Baboon	3	16.80716	**16.81227**	16.81210	0.84245	**0.87998**	0.82788
	4	16.80836	**16.81381**	16.81309	0.84945	**0.88368**	0.85707
	5	16.80992	**16.81590**	16.81508	0.90455	**0.94180**	0.90692
	6	16.81193	**16.81856**	16.81734	0.91001	**0.96437**	0.94376
Donna	3	20.16255	**20.15640**	20.12316	0.86492	**0.86095**	0.80656
	4	20.16399	**20.15780**	20.14389	0.88252	**0.88108**	0.85887
	5	20.15689	**20.15961**	20.15091	0.91602	**0.91624**	0.89832
	6	20.15917	**20.16215**	20.16023	0.92516	**0.92076**	0.91674
Fruit	3	21.74544	**21.75639**	21.75392	0.68011	**0.85262**	0.84863
	4	21.74662	**21.75822**	21.75532	0.70181	**0.86458**	0.85375
	5	21.74844	**21.75995**	21.75819	0.77284	**0.93161**	0.90391
	6	21.75992	**21.76244**	21.76136	0.82004	**0.94344**	0.93295
Flower	3	18.65565	**18.65517**	18.65681	0.84662	**0.87416**	0.84106
	4	18.65683	**18.65691**	18.65694	0.87125	**0.88711**	0.88712
	5	18.65856	**18.65876**	18.65862	0.90225	**0.93231**	0.92637
	6	18.66059	**18.66105**	18.66105	0.91149	**0.94352**	0.94352

(continued on next page)

Secondly, in order to obtain an effective method for color image multilevel segmentation, the EKH algorithm based on Otsu's method are compared with Kapur's entropy and Tsallis entropy. The higher value of PSNR and SSIM shows a better quality of

Table 27 (*continued*).

Image	m	PSNR (OTSU) EKH	PSNR (Kapur) EKH	PSNR (Tsallis) EKH	SSIM (OTSU) EKH	SSIM (Kapur) EKH	SSIM (Tsallis) EKH
House	3	15.37113	**15.37225**	15.37162	0.85188	**0.85331**	0.79738
	4	15.37236	**15.37479**	15.37295	0.87494	**0.87095**	0.85269
	5	15.37418	**15.37655**	15.37583	0.88916	**0.92356**	0.90398
	6	15.37624	**15.37895**	15.37903	0.89615	**0.93161**	0.92494
Pallon	3	18.35243	**18.35515**	18.35033	0.78392	**0.84324**	0.80845
	4	18.35427	**18.35674**	18.35261	0.78486	**0.86352**	0.83449
	5	18.35613	**18.35822**	18.35752	0.87235	**0.91200**	0.90277
	6	18.35870	**18.35987**	18.35906	0.88157	**0.93428**	0.92670
Sailboat	3	16.46503	**16.47373**	16.46198	0.85674	**0.89558**	0.84848
	4	16.46616	**16.47499**	16.46393	0.87891	**0.90354**	0.88218
	5	16.46777	**16.47685**	16.46723	0.91276	**0.92950**	0.90137
	6	16.47009	**16.47872**	16.47356	0.92051	**0.94040**	0.92678
Tulips	3	17.44694	**17.44759**	17.44468	0.90046	**0.89211**	0.86061
	4	17.44851	**17.44928**	17.44657	0.90286	**0.90784**	0.88059
	5	17.44985	**17.45096**	17.44956	0.94209	**0.95069**	0.92189
	6	17.45132	**17.45305**	17.45636	0.94901	**0.96615**	0.94150
Voit	3	18.97092	**18.97051**	18.97068	0.72260	**0.87195**	0.87030
	4	18.97224	**18.97231**	18.97151	0.74391	**0.88314**	0.87976
	5	18.97389	**18.97410**	18.97356	0.81435	**0.93316**	0.90168
	6	18.97579	**18.97615**	18.97586	0.82671	**0.94457**	0.93143
Yacht	3	18.49263	**18.49345**	18.49558	0.85616	**0.85109**	0.85494
	4	18.49398	**18.49523**	18.49603	0.86673	**0.88988**	0.88351
	5	18.49588	**18.49785**	18.49848	0.88987	**0.95810**	0.93348
	6	18.49776	**18.50933**	18.50369	0.90049	**0.96339**	0.95320

thresholding segmentation. The PSNR and SSIM values obtained by the EKH algorithm based on Otsu, Kapur's entropy and Tsallis entropy are listed in Table 27. It can be clearly identified that the presented EKH algorithm based on Kapur's entropy provides greater PSNR and SSIM values than Otsu's method and Tsallis entropy. For example, on comparing the PSNR values, Kapur's entropy method gives better results in 40 out of 40 cases (10 images and 4 threshold values). From Tables 9–10, 17–18 and 25–26, it is also observed that the EKH algorithm based on Kapur's entropy gives lower standard deviation value than Otsu's method and Tsallis entropy. To sum up, the proposed EKH algorithm based on Kapur's entropy gives better experimental results. Therefore, the EKH algorithm based on Kapur's entropy is an effective method for color image multilevel segmentation.

In order to show the effectiveness of the presented EKH algorithm for color image multilevel segmentation, the convergence curve of best objective function values are drawn in Fig. 10. It can be clearly observed from Fig. 10 the presented EKH algorithm can obtain best objective function values (kapur's entropy) in less number of iterations compared with KH I, KH II, KH IV algorithm and has better convergence property.

Moreover, in order to further test the performance of the proposed EKH algorithm, a statistical analysis of the test results is performed in this paper. "Wilcoxon's rank sum test" is a non-parametric statistical test that can check whether one of two independent samples tends to have larger values than the other and does not depend on the specific form of population distribution, so it is used to evaluate the significant difference between algorithms [88]. To reduce the statistical errors, the experiments are repeated for 30 times at significance level 5%. The PSNR values of all the algorithms based on Otsu's method, Kapur's entropy and Tsallis entropy are compared. The alternative hypothesis assumes that there is a significant difference between the two algorithms being compared and the results of which is represented as "$p < 0.05$ and $h = 1$". The null hypothesis considers that there is no significant difference between the algorithms and the "p" values are given in numerical form. A value of $p > 0.05$ indicates that the null hypothesis cannot be rejected. A value of $p < 0.05$ means the null hypothesis can be rejected at the 5% significance level.

The experimental results are shown in Tables 28–29. From the experimental results, we can see that EKH based method gives the satisfied results in general. For example, in the circumstance of Otsu technique, the proposed EKH method gives better results in 40 out of 40 cases for KH I, 40 cases for KH II, 40 cases for KHIV, 40 cases for KHIV, 40 cases for MFA, 39 cases for MGOA, 38 cases for WCA, 40 cases for BA. In terms of Tsallis technique, the EKH based method outperforms in 39 out of 40 cases for KH I and KHIV. All the other algorithms show a significant difference with IDA based method. To sum up, we can see that the proposed EKH algorithm has the better performance with statistical significance.

6. Conclusion

In this paper, to overcome the drawbacks of conventional color image multilevel thresholding segmentation algorithm, a novel color image multilevel segmentation method based on an efficient krill herd (EKH) algorithm is proposed. The method using three different objective functions, between-class variance, Kapur's entropy and Tsallis entropy, are employed to maximize the objective function to search the optimal threshold values for color image multilevel thresholding problem. The EKH algorithm has been tested with ten color test images considering four different threshold values $m = 3, 4, 5, 6$ and the performance has been compared with basic KH I, KH II, KH IV algorithm and the four best recent algorithms, namely MFA, MGOA, WCA and BA. The performance measures include optimal threshold values, objective functions, PSNR, SSIM and standard deviation values.

As a comparative study, we can see that the EKH algorithm based on Otsu's method, Kapur's entropy and Tsallis entropy is superior to KH I, KH II, KH IV, MFA, MGOA, WCA and BA algorithm for color image multilevel thresholding segmentation and the presented EKH algorithm using Kapur's entropy is superior to the EKH algorithm using Otsu's method and Tsallis entropy in terms of PSNR, SSIM and standard deviation values. In addition, the convergence curves of the four different algorithms based on Kapur's entropy for 6 levels are drawn for all test images and show that the EKH algorithm converges quickly than the KH I, KH II, KH IV algorithms. A non-parametric statistical test called "Wilcoxon's rank sum test" is used for statistical analysis. The results indicated that the EKH algorithm has better performance with statistical significance. As a result, the EKH algorithm

Table 28
Statistical analysis for experimental results based on Otsu's method and Kapur's entropy.

Image	m	Otsu														Kapur													
		EKH vs KHI		EKH vs KH II		EKH vs KH IV		EKH vs MFA		EKH vs MGOA		EKH vs WCA		EKH vs BA		EKH vs KH I		EKH vs KH II		EKH vs KH IV		EKH vs MFA		EKH vs MGOA		EKH vs WCA		EKH vs BA	
		p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h
Baboon	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Donna	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Fruit	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Flower	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	0.06235	0	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
House	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	0.03893	0	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Pallon	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Sailboat	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Tulips	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Voit	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Yacht	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	0.04659	0	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1

Table 29
Statistical analysis for experimental results based on Tsallis entropy.

Image	m	Tsallis														EKH		EKH	
		EKH vs KH I		EKH vs KH II		EKH vs KH IV		EKH vs MFA		EKH vs MGOA		EKH vs WCA		EKH vs BA		Kapur vs Otsu		Kapur vs Tsallis	
		p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h	p	h
Baboon	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Donna	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	0.07534	0	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Fruit	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Flower	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
House	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Pallon	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Sailboat	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Tulips	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	0.05647	0	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Voit	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
Yacht	3	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	4	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	5	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1
	6	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1	<0.05	1

based on Kapur's entropy is an efficient method for color image multilevel segmentation.

As a scope of further research, the other objective functions will be introduced to the EKH algorithm for color image multilevel segmentation problem. In addition, the EKH algorithm is an efficient optimization algorithm and can be also used to various types of complex and real-time image processing applications. For other researchers further work is to be carried out to present improved krill herd algorithm for multilevel image segmentation problem and for more complex image processing and practical engineering problems.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2020.106063.

CRediT authorship contribution statement

Lifang He: Conceptualization, Data curation, Investigation, Methodology, Project administration, Resouces, Software, Validation, Writing - original draft. **Songwei Huang:** Formal analysis, Funding acquisition, Writing - review & editing.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 51204077).

References

[1] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst. Man Cybern. 9 (1) (1979) 62–66.

[2] L. Barghout, J. Sheynin, Real-world scene perception and perceptual organization: Lessons from computer vision, J. Vis. 13 (9) (2013) 709.

[3] T. Lindeberg, M.X. Li, Segmentation and classification of edges using minimum description length approximation and complementary junction cues, Comput. Vis. Image Underst. 67 (1) (1997) 88–98.

[4] R. Nock, F. Nielsen, Statistical region merging, IEEE Trans. Pattern Anal. Mach. Intell. 26 (11) (2004) 1452–1458.

[5] L. Grady, E.L. Schwartz, Isoperimetric graph partitioning for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 28 (3) (2006) 469–475.

[6] S. Agrawal, R. Panda, S. Bhuyan, B.K. Panigrahi, Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm, Swarm Evol. Comput. 11 (2013) 16–30.

[7] A.K. Bhandari, V.K. Singh, A. Kumar, G.K. Singh, Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy, Expert Syst. Appl. 41 (7) (2014) 3538–3560.

[8] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, J. Electron. Imaging 13 (1) (2004) 146–166.

[9] B. Akay, A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding, Appl. Soft Comput. 13 (6) (2013) 3066–3091.

[10] W.H. Tsai, Moment-preserving thresolding: A new approach, Comput. Vis. Graph. Image Process. 29 (3) (1985) 377–393.

[11] A.K. Bhandari, A. Kumar, G.K. Singh, Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using Kapur's, Otsu and Tsallis functions, Expert Syst. Appl. 42 (3) (2015) 1573–1601.

[12] J.N. Kapur, P.K. Sahoo, A.K. Wong, A new method for gray-level picture thresholding using the entropy of the histogram, Comput. Vis. Graph. Image Process. 29 (3) (1985) 273–285.

[13] C.H. Li, C.K. Lee, Minimum cross entropy thresholding, Pattern Recognit. 26 (4) (1993) 617–625.

[14] D. Mera, J.M. Cotos, J. Varela-Pet, O. Garcia-Pineda, Adaptive thresholding algorithm based on SAR images and wind data to segment oil spills along the northwest coast of the Iberian Peninsula, Mar. Pollut. Bull. 64 (10) (2012) 2090–2096.

[15] A. El-Zaart, A.A. Ghosn, SAR images thresholding for oil spill detection, in: Processing Electronics, Communications and Photonics Conference, SIECPC, 2013, pp. 1–5.

[16] M. Maitra, A. Chatterjee, A novel technique for multilevel optimal magnetic resonance brain image thresholding using bacterial foraging, Measurement 41 (10) (2008) 1124–1134.

[17] P.D. Sathya, R. Kayalvizhi, Optimal segmentation of brain MRI based on adaptive bacterial foraging algorithm, Neurocomputing 74 (14–15) (2011) 2299–2313.

[18] R. Sammouda, N. Adgaba, A. Touir, A. Al-Ghamdi, Agriculture satellite image segmentation using a modified artificial hopfield neural network, Comput. Hum. Behav. 30 (2014) 436–441.

[19] A.K. Bhandari, A. Kumar, G.K. Singh, Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using Kapur's, Otsu and Tsallis functions, Expert Syst. Appl. 42 (3) (2015) 1573–1601.

[20] S. Suresh, S. Lal, An efficient cuckoo search algorithm based multilevel thresholding for segmentation of satellite images using different objective functions, Expert Syst. Appl. 58 (2016) 184–209.

[21] D. Feng, S. Wenkang, C. Liangzhou, D. Yong, Z. Zhenfu, Infrared image segmentation with 2-d maximum entropy method based on particle swarm optimization (PSO), Pattern Recognit. Lett. 26 (5) (2005) 597–603.

[22] J. Liu, Y. Liu, Q. Ge, An adaptive fuzzy clustering algorithm based on multi-threshold for infrared image segmentation, in: Processing CCF Chinese Conference on Computer Vision, 2015, pp. 277–286.

[23] M. Maitra, A. Chatterjee, A hybrid cooperative–comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding, Expert Syst. Appl. 34 (2) (2008) 1341–1350.

[24] H. Gao, W. Xu, J. Sun, Y. Tang, Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm, IEEE Trans. Instrum. Meas. 59 (4) (2010) 934–946.

[25] Y. Liu, C. Mu, W. Kou, J. Liu, Modified particle swarm optimization-based multilevel thresholding for image segmentation, Soft Comput. 19 (5) (2015) 1311–1327.

[26] W.B. Tao, J.W. Tian, J. Liu, Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm, Pattern Recognit. Lett. 24 (16) (2003) 3069–3078.

[27] K. Hammouche, M. Diaf, P. Siarry, A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation, Comput. Vis. Image Underst. 109 (2) (2008) 163–175.

[28] E. Cuevas, D. Zaldivar, M. Pérez-Cisneros, A novel multi-threshold segmentation approach based on differential evolution optimization, Expert Syst. Appl. 37 (7) (2010) 5265–5271.

[29] S. Sarkar, G.R. Patra, S. Das, A differential evolution based approach for multilevel image segmentation using minimum cross entropy thresholding, in: Processing International Conference on Swarm, Evolutionary, and Memetic Computing, 2011, pp. 51–58.

[30] H.V.H. Ayala, F.M. dos Santos, V.C. Mariani, L. dos Santos Coelho, Image thresholding segmentation based on a novel beta differential evolution approach, Expert Syst. Appl. 42 (4) (2015) 2136–2142.

[31] M.H. Horng, Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation, Expert Syst. Appl. 38 (11) (2011) 13785–13791.

[32] E. Cuevas, F. Sención, D. Zaldivar, M. Pérez-Cisneros, H. Sossa, A multi-threshold segmentation approach based on artificial bee colony optimization, Appl. Intell. 37 (3) (2012) 321–336.

[33] Y. Zhang, L. Wu, Optimal multi-level thresholding based on maximum Tsallis entropy via an artificial bee colony approach, Entropy 13 (4) (2011) 841–859.

[34] I. Brajevic, M. Tuba, Cuckoo search and firefly algorithm applied to multilevel image thresholding, in: Cuckoo Search and Firefly Algorithm, Springer, Cham., 2014, pp. 115–139.

[35] M.H. Horng, R.J. Liou, Multilevel minimum cross entropy threshold selection based on the firefly algorithm, Expert Syst. Appl. 38 (12) (2011) 14805–14811.

[36] K. Chen, Y. Zhou, Z. Zhang, M. Dai, Y. Chao, J. Shi, Multilevel image segmentation based on an improved firefly algorithm, Math. Probl. Eng. 2016 (2016) 285–296.

[37] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Processing IEEE International Conference on Neural Networks, Western Australia, 1995, pp. 1942–1948.

[38] J. Kennedy, The particle swarm: social adaptation of knowledge, in: Processing IEEE International Conference on Evolutionary Computation, 1997, pp. 303–308.

[39] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: Processing IEEE World Congress on Computational Intelligence, 1998, pp. 69–73.

[40] A.E. Eiben, P.E. Raue, Z. Ruttkay, Genetic algorithms with multi-parent recombination, in: Processing International Conference on Parallel Problem Solving from Nature, 1994, pp. 78–87.

[41] D.E. Goldberg, J.H. Holland, Genetic algorithms and machine learning, Mach. Learn. 3 (2) (1988) 95–99.

[42] D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm, Appl. Soft Comput. 8 (1) (2008) 687–697.

[43] D. Karaboga, B. Akay, A comparative study of artificial bee colony algorithm, Appl. Math. Comput. 214 (1) (2009) 108–132.

[44] W. Gao, S. Liu, Improved artificial bee colony algorithm for global optimization, Inform. Process. Lett. 111 (17) (2011) 871–882.

[45] X.S. Yang, S. Deb, Cuckoo search via Lévy flights, in: Processing Nature & Biologically Inspired Computing, NaBIC, 2009, pp. 210–214.

[46] X.S. Yang, S. Deb, Engineering optimisation by cuckoo search, Int. J. Math. Model. Numer. Optim. 1 (4) (2010) 330–343.

[47] X.S. Yang, Firefly algorithm, in: Nature-Inspired Metaheuristic Algorithms, Luniver Press, Frome, 2010, pp. 81–89.

[48] X.S. Yang, Firefly algorithm, stochastic test functions and design optimisation, Int. J. Bio-Inspired Comput. 2 (2) (2010) 78–84.

[49] A.H. Gandomi, X.S. Yang, A.H. Alavi, Mixed variable structural optimization using firefly algorithm, Comput. Struct. 89 (23–24) (2011) 2325–2336.

[50] A.H. Gandomi, X.S. Yang, S. Talatahari, A.H. Alavi, Firefly algorithm with chaos, Commun. Nonlinear Sci. Numer. Simul. 18 (1) (2013) 89–98.

[51] Z. Bayraktar, M. Komurcu, D.H. Werner, Wind Driven Optimization (WDO): A novel nature-inspired optimization algorithm and its application to electromagnetics, in: Processing Antennas and Propagation Society International Symposium, APSURSI, 2010, pp. 1–4.

[52] Z. Bayraktar, M. Komurcu, J.A. Bossard, D.H. Werner, The wind driven optimization technique and its application in electromagnetics, IEEE Trans. Antennas Propag. 61 (5) (2013) 2745–2757.

[53] S. Pare, A. Kumar, V. Bajaj, G.K. Singh, An efficient method for multilevel color image thresholding using cuckoo search algorithm based on minimum cross entropy, Appl. Soft Comput. 61 (2017) 570–592.

[54] A.K. Bhandari, A. Kumar, S. Chaudhary, G.K. Singh, A novel color image multilevel thresholding based segmentation using nature inspired optimization algorithms, Expert Syst. Appl. 63 (2016) 112–133.

[55] Y. Liu, K. Hu, Y. Zhu, H. Chen, Color image segmentation using multilevel thresholding-cooperative bacterial foraging algorithm, in: Processing Cyber Technology in Automation, Control, and Intelligent Systems, CYBER, 2015, pp. 181–185.

[56] S. Dey, S. Bhattacharyya, U. Maulik, New quantum inspired meta-heuristic techniques for multi-level colour image thresholding, Appl. Soft Comput. 46 (2016) 677–702.

[57] V. Rajinikanth, M.S. Couceiro, RGB histogram based color image segmentation using firefly algorithm, Procedia Comput. Sci. 46 (2015) 1449–1457.

[58] S. Sarkar, S. Das, S.S. Chaudhuri, A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution, Pattern Recognit. Lett. 54 (2015) 27–35.

[59] T. Kurban, P. Civicioglu, R. Kurban, E. Besdok, Comparison of evolutionary and swarm based computational techniques for multilevel color image thresholding, Appl. Soft Comput. 23 (2014) 128–143.

[60] A.H. Gandomi, A.H. Alavi, Krill herd: a new bio-inspired optimization algorithm, Commun. Nonlinear Sci. Numer. Simul. 17 (12) (2012) 4831–4845.

[61] L. Guo, G.G. Wang, A.H. Gandomi, A.H. Alavi, H. Duan, A new improved krill herd algorithm for global numerical optimization, Neurocomputing 138 (2014) 392–402.

[62] G.G. Wang, A.H. Gandomi, A.H. Alavi, An effective krill herd algorithm with migration operator in biogeography-based optimization, Appl. Math. Model. 38 (9–10) (2014) 2454–2462.

[63] G.G. Wang, L. Guo, A.H. Gandomi, G.S. Hao, H. Wang, Chaotic krill herd algorithm, Inform. Sci. 274 (2014) 17–34.

[64] P.D. Sathya, R. Kayalvizhi, Optimal multilevel thresholding using bacterial foraging algorithm, Expert Syst. Appl. 38 (12) (2011) 15549–15564.

[65] S. Suresh, S. Lal, Multilevel thresholding based on chaotic darwinian particle swarm optimization for segmentation of satellite images, Appl. Soft Comput. 55 (2017) 503–522.

[66] K. Hammouche, M. Diaf, P. Siarry, A comparative study of various metaheuristic techniques applied to the multilevel thresholding problem, Eng. Appl. Artif. Intell. 23 (5) (2010) 676–688.

[67] M.A. El Aziz, A.A. Ewees, A.E. Hassanien, Whale optimization algorithm and moth-flame optimization for multilevel thrholding image segmentation, Expert Syst. Appl. 83 (2017) 242–256.

[68] M. Tuba, N. Bacanin, A. Alihodzic, Multilevel image thresholding by fireworks algorithm, in: 2015 25th International Conference Radioelektronika, RADIOELEKTRONIKA, 2015, pp. 326–330.

[69] E. Tuba, A. Alihodzic, M. Tuba, Multilevel image thresholding using elephant herding optimization algorithm, in: 2017 14th International Conference on Engineering of Modern Electric Systems, EMES, 2017, pp. 240–243.

[70] L. He, S. Huang, Modified firefly algorithm based multilevel thresholding for color image segmentation, Neurocomputing 240 (2017) 152–174.
[71] S. Pare, A.K. Bhandari, A. Kumar, G.K. Singh, Rényi's entropy and Bat algorithm based color image multilevel thresholding, in: Mach. Intel. Signal Anal., Springer, Singapore, 2019, pp. 71–84.
[72] P. Kandhway, A.K. Bhandari, A water cycle algorithm-based multilevel thresholding system for color image segmentation using masi entropy, Circuits Systems Signal Process. (2018) 1–49.
[73] H. Liang, H. Jia, Z. Xing, J. Ma, X. Peng, Modified grasshopper algorithm based multilevel thresholding for color image segmentation, IEEE Access (2019).
[74] L. Xu, H. Jia, C. Lang, X. Peng, K. Sun, A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution, IEEE Access 7 (2019) 19502–19538.
[75] S. Saremi, S. Mirjalili, A. Lewis, Grasshopper optimisation algorithm: theory and application, Adv. Eng. Softw. 105 (2017) 30–47.
[76] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1) (1997) 67–82.
[77] A.H. Gandomi, X.S. Yang, A.H. Alavi, Cuckoo search algorithm: a meta-heuristic approach to solve structural optimization problems, Eng. Comput. 29 (1) (2013) 17–35.
[78] X.S. Yang, Firefly algorithm, levy flights and global optimization, in: Research and Development in Intelligent Systems XXVI, Springer, London, 2010, pp. 209–218.
[79] S. Łukasik, S. Zak, Firefly algorithm for continuous constrained optimization tasks, in: Processing International Conference on Computational Collective Intelligence, 2009, pp. 97–106.
[80] I. Fister, I. Fister Jr., X.S. Yang, J. Brest, A comprehensive review of firefly algorithms, Swarm Evol. Comput. 13 (2013) 34–46.
[81] S. Debbarma, L.C. Saikia, N. Sinha, Solution to automatic generation control problem using firefly algorithm optimized IλDμ controller, ISA Trans. 53 (2) (2014) 358–366.
[82] H. Shareef, A.A. Ibrahim, N. Salman, A. Mohamed, W.L. Ai, Power quality and reliability enhancement in distribution systems via optimum network reconfiguration by using quantum firefly algorithm, Int. J. Electr. Power Energy Syst. 58 (2014) 160–169.
[83] G. Dhiman, V. Kumar, Emperor penguin optimizer: A bio-inspired algorithm for engineering problems, Knowl.-Based Syst. 159 (2018) 20–50.
[84] B. Sowmya, B.S. Rani, Colour image segmentation using fuzzy clustering techniques and competitive neural network, Appl. Soft Comput. 11 (3) (2011) 3170–3178.
[85] D. Oliva, E. Cuevas, G. Pajares, D. Zaldivar, V. Osuna, A multilevel thresholding algorithm using electromagnetism optimization, Neurocomputing 139 (2014) 357–381.
[86] A.K. Bhandari, A. Kumar, G.K. Singh, Tsallis entropy based multilevel thresholding for colored satellite image segmentation using evolutionary algorithms, Expert Syst. Appl. 42 (22) (2015) 8707–8730.
[87] Z. Wang, E.P. Simoncelli, A.C. Bovik, Multiscale structural similarity for image quality assessment, in: Processing Asilomar Conference on Signals, Systems and Computers, 2003, pp. 1398–1402.
[88] F. Wilcoxon, Individual comparisons by ranking methods, Biom. Bull. 1 (6) (1945) 80–83.