Multimedia Tools and Applications (2023) 82:32711–32753
https://doi.org/10.1007/s11042-023-14637-1

Hybrid SCCSA: An efficient multilevel thresholding for enhanced image segmentation

A. Renugambal[1] · K. Selva Bhuvaneswari[2] · A. Tamilarasan[3]

Received: 11 November 2021 / Revised: 22 April 2022 / Accepted: 3 February 2023 /
Published online: 2 March 2023

Abstract

In a variety of image processing applications, multilevel thresholding image segmentation has gotten a lot of interest. When traditional approaches are utilised, however, the process of obtaining the ideal threshold values takes time. Despite the fact that Hybrid metaheuristic methods can be used to overcome these limits, such approaches may be ineffective when dealing with a local solution. The present study proposes a multi-level image thresholding based hybridization strategy based Sine-Cosine Crow Search Algorithm(SCCSA) to make more efficient image segmentation. The main limitation of the classical Crow Search Algorithm (CSA) is that search agents sometimes do not produce the best solutions. To update a solution to the best solution, each search agent can use Sine-Cosine Algorithm (SCA) movements to update its position accordingly. This ensures a good balance between two goals (exploration and exploitation) would improve the efficiency of the search algorithm. The optimal threshold values are searched by the chosen objective functions of the otsu's and kapur's entropy approaches. The hybrid algorithm is evaluated in 12 standard image sets and then compared with the performance of other state-of-the-art algorithms such as ICSA, SCA, CSA and ABC. Experimental results showed that, in different metrics of the output such as objective function values, PSNR, STD values, Mean, SSIM, FSIM and CPU time, the proposed algorithm is consistently higher than other algorithms. In addition, the wilcoxon test is performed using the proposed algorithm to detect the significant differences between the other algorithms. The findings indicated that the proposed SCCSA succeeds with other well-known algorithms and has dominance over robust, accurate and convergent values.

✉ A. Renugambal
renutjn2016@gmail.com

[1] Department of Mathematics, University College of Engineering Kancheepuram, Kanchipuram, Tamilnadu 631552, India

[2] Department of Computer Science and Engineering, University College of Engineering Kancheepuram, Kanchipuram, Tamilnadu 631552, India

[3] Department of Mechanical Engineering, Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya, Kanchipuram, Tamilnadu 631561, India

Keywords Multilevel thresholding · Image segmentation · Hybridization strategy · Metaheuristics · SCCSA algorithm

1 Introduction

Segmentation of images is a central method in many applications and innovations for image, video and a vision of computers [22]. It is a crucial stage in the image processing process that separates an image into numerous segments in order to boost the significance of the picture representation in the places where the image segments are divided [1]. For computer-based vision systems, these important regions are easier to evaluate, including defence [17, 52] medical [15, 44], object detection [21], quality inspection [11], crack detection [36], and remote sensing [10, 45]. Four popular methods of segmentation are: thresholding, region-based strategies, edge identification, and relaxation techniques to preserve connectivity [28]. The thresholding-based strategy, in particular, has piqued the interest of researchers due to its efficiency and ease of implementation [18]. The thresholding-based technique can be categorized into bi-level and multi-level thresholding. Bi-level thresholding, as the name implies, employs a single threshold value to divide an image into two uniform foreground(object) and background sections. When an image contains a variety of objects with varying intensities, the bi-level threshold was unable to distinguish between them. Multi-level thresholding, on the other hand, separates an image into multiple zones based on pixel intensities [13]. Multilevel thresholding techniques are widely employed in a variety of fields of image processing and computer vision. Satellite image processing, synthetic aperture radar image segmentation and medical image analysis are just a few of the many significant and intriguing uses of multilayer thresholding. However, choosing the right threshold values is still the most difficult part, and more research is needed to figure that out. In the literature, there are a variety of thresholding strategies. Some are self-contained, while others require user intervention. Manually segmenting a large number of images is a time-consuming operation. Manually segmenting a large number of photos is not always practical, and it can also be inaccurate. As a result, automatic segmentation techniques are gaining a lot of support nowadays. Techniques for setting thresholds were utilised using two approaches: parametric and non-parametric approaches, try to identify the best threshold values. For classifying image classes in a parametric approach, various parameters of a probability density function should be determined. However, this method is computationally expensive and time consuming, whereas the non parametric method optimises multiple factors such as the error rate, entropy, and others to obtain the best threshold values [33]. Nonparametric approaches are choose thresholds by optimization (minimizing or maximizing) of certain feature parameter functions and easier to implement computationally rather than parametric approaches [33]. As a segmentation strategy, multi-level image thresholding has recently become a powerful technique. The threshold of the multilevel divides an image into various regions. Bi-level thresholds are the easiest form, provided that only a threshold value is chosen. But, with increasing the number of threshold levels, this is becoming more complicated. In fact, numerical complexity is exponential and contributes to an increase in threshold values over a long period of time [34]. Otsu and Kapur's entropy are two state-of-the-art thresholding methods among the existing image thresholding approaches. Otsu and Kapur's entropy help them find the best threshold values based on a set of rules. Otsu maximises the histogram classes' variance, while Kapur's entropy maximises the histogram's entropy. The above-mentioned thresholding approaches can readily be expanded to multilevel thresholding segmentation. The exhaustive search, on the other hand, makes finding the best threshold values inefficient, and the time complexity grows exponentially as the number of thresholds grows. Metaheuristic algorithms are widely employed in multilevel thresholding situations to solve the aforementioned limitation. Such algorithms can mimic natural phenomena to solve the problems of this difficult optimization, to avoid the local optimum by exploring randomization in the search space repeatedly. There were several algorithms widely used to solve multilevel thresholds. Popular meta-heuristic algorithms are included. The most common approaches, including genetic [25], particle swarm [47], honey bee mating [26], artificial bee colony [14], firefly [27], ant colony [53], differential evolution [46], cuckoo Search [3], and bacterial foraging [48] algorithms, have been addressed for the past decade. A newly proposed algorithms, such as whale [8, 37], gray wolf [30, 31], moth swarm [56], animal migration [43], spider monkey [4], krill herd [9], harmony search [41], spherical search [39], flower pollination [49], bat [54], teaching-learning method [50], and elephant herding [51] in the last couple of years have been suggested for multilevel thresholding. The limits on convergence speed or accuracy are all of these methods and all experiments aim to match these two main aspects. But all the algorithms can still be trapped in local optima, which have a significant impact on their segmentation quality. Each optimization approach may face different local solutions; therefore, combining two optimization algorithms is able to escape individual local solutions. Hybridization of one or more algorithms is today the latest trend in research to solve problems of high dimensionality. These algorithms are extremely capable of overcoming one algorithm's poor exploration abilities and the other algorithm's poor exploitability [23]. In fact, the hybridization of algorithms is a practical way to overcome the constraint by growing its performance in terms of convergence and solution consistency [23].

2 Related work

Currently, hybrid algorithms to address the multilevel thresholding problem have been suggested. Ewees et al. [20] introduced a multi-level hybrid WOAPSO threshold algorithm based upon two objectives: fuzzy entropy and otsu's functions. Results showed that the WOAPSO algorithm is highly efficient in demonstrating high competitiveness in almost every aspect of the criteria in compared with other seven algorithms. Mlakar et al. [35] introduced a multi-level image threshold otsu's hybrid hjDE algorithm. Eleven real-life images has been evaluated and compared to the algorithms CS, DE, jDE, ABC, and PSO. In 2017, Dehshibi et al. [16]implemented a new BFHS hybrid algorithm with otsu's and kapur entropy criterions utilizing two set (standard and satellite) of images. Research findings show that the algorithm's ability to select multiple thresholds is important, as compared to the other algorithms that address the same problem. In addition, BFHS < HS < BF < GA is the order of the CPU time from low to high. A hybrid SCABC algorithm, which hybridize ABC to improve the level of exploitation and exploration in the classical ABC algorithm with SCA algorithm, proposed by Gupta and Deep [24]. A better SCABC search capacity as compared with traditional ABC, SCA and The overall analysis therefore suggests that SCABC be better than SCA and ABC algorithms. Aziz et al. [19] introduced a nature inspired behaviour of fireflies and real spider based hybrid FASSO algorithm to achieve the maximum between class variance criteria. Experimental results revealed the utility of the FASSO algorithm and provide comparatively lesser CPU time for quicker convergence. Ahmadi et al. [5] developed a hybrid BMO-DE optimization algorithm focused on bird mating optimization (BMO) and differential evolutionary (DE) strategies utilizing the methods of kapur's and otsu's. The algorithm has better results in solutions accuracy compared with other popular evolutionary algorithms, such as GA, PSO, BF and MBF. Yue and Zhang [55] incorporated hybrid invasive weed bat algorithm (IWBA) for the selection of optimal thresholds. The comparatives tests show that the IWBA algorithm is better and more efficient than the GSA, PSO-GSA and BA algorithms. According to the literature reviewed, in recent years few researchers have been interested in finding the solutions of newly developed hybrid algorithms to multilevel thresholding problems and have become an active area of research. Despite the fact that significant work has been done in this field, good image segmentation remains elusive for practitioners, owing to the following two factors: To begin with, there is limited unanimity on what criteria should be used to assess image segmentation quality. It can be difficult to achieve a fair balance between objective metrics that are entirely based on the underlying statistics of imaging data and subjective measures that attempt to empirically approximate human experience. Second, there has been a lack of consensus on acceptable models for a unified representation of image segments in the search for objective metrics. Many researchers have strong reasons for developing strategies that provide near-optimal metaheuristic search through a wide search space, despite the enormous effort required. Therefore, this paper proposes a new hybridized sine-cosine crow search algorithm (SCCSA) for multi-level thresholding. In SCCSA, the combined architecture and operators of the two separate algorithms (CSA and SCA) have made it possible to find a reasonable compromise between exploration and exploitation capabilities. A new insight into the work is the hybridization of CSA and SCA algorithms to boost the solution consistency. The hybridization of one or more search algorithms not only increases their search capacity, but also leads to their optimization and solves further variations of problems to some degree. For this work, two objective functions, like otsu's and kapur's entropy criteria, were used. The three main contributions will be summarized as follows:

- Propose a SCCSA algorithm for multilevel thresholds of two sets, including the objectives of otsu's and kapur's.
- Incorporating the other swarm algorithms for multi-level thresholding, namely ICSA,SCA, CSA and ABC.
- Comprehensive qualitative and quantitative comparisons of the SCCSA with other algorithms to validate the findings.

The remainder of the paper is accordingly arranged. Section 2 explains the problem formulation. A background of hybrid SCCSA algorithm is covered in Section 3. Section 4 describes the approach of proposed multilevel thresholding based SCCSA algorithm. Experimental work and evaluation, Results and discussion are discussed respectively in Section 5 and 6. Consequently, Section 7 addresses the conclusions and future work.

3 Problem formulation

Traditionally, multi-level thresholding approaches used to grayscale photos placed thresholds on the histogram of the images. The intensities that are placed between two thresholds are assumed to belong to the same segment as the intensities that are placed between two thresholds. Two thresholding strategies are briefly explained in Sections 3.1 and 3.2.

3.1 Based on Otsu's method

The Otsu's approach is one of the most popular approaches to two and multiple thresholds based on the finding of the optimum threshold, which can be defined as the number of sigma functions for each region in the following equation by optimizing the segmented region of interclass variation [40].

$$f(t) = \sigma_1 + \sigma_2; \sigma_1 = \omega_0(\mu_0 - \mu_T)^2 \text{and } \sigma_2 = \omega_1(\mu_1 - \mu_T)^2 \tag{1}$$

Where μT in the above equation denotes the mean image amplitude and for the thresholding of two levels. Mean can be described as of every class [12].

$$\mu_0 = \sum_{i=0}^{t-1} \frac{ip_i}{\omega_0} \text{ and } \mu_1 = \sum_{i=t}^{L-1} \frac{ip_i}{\omega_1} \tag{2}$$

Through optimizing interclass variances the desired threshold can be reached [12].

$$(t^*) = argmax(f(t)) \tag{3}$$

The same strategy is used for multilevel threshold problems [12].

$$f(t) = \sum_{i=0}^{m} \sigma_i \tag{4}$$

The σ could be extended [12].

$$\begin{gathered} \sigma_1 = \omega_1(\mu_1 - \mu_T)^2, \\ \sigma_2 = \omega_2(\mu_2 - \mu_T)^2 \\ \sigma_j = \omega_j\left(\mu_j - \mu_T\right)^2 \\ \sigma_m = \omega_m(\mu_m - \mu_T)^2 \end{gathered} \tag{5}$$

$$\mu_0 = \sum_{i=0}^{t_1-1} \frac{ip_i}{\omega_i}; \mu_1 = \sum_{i=t_1}^{t_2-1} \frac{ip_i}{\omega_i}; \mu_j = \sum_{i=t_j+1}^{L-1} \frac{ip_i}{\omega_i}; \mu_1 = \sum_{i=t_1}^{t_2-1} \frac{ip_i}{\omega_i} and \mu_m = \sum_{i=t_m}^{t_j+1-1} \frac{ip_i}{\omega_i} \tag{6}$$

Optimizing the objective function of eq. 7 as follows will achieve the desired threshold value [12].

$$(t^*) = \arg\ \max\left(\sum_{i=0}^{m} \sigma_i\right) \tag{7}$$

3.2 Based on Kapur's entropy

The kapur's method maximizes the entropic measurement of the segmented histogram for the distribution of the region. Kapur's proposed two distributions of probability, one for the object and the other for the background. Kapur's entropy describes an essentially gray level histogram image [29]. The entropy of kapur determines the image completely defined by its accompanying histogram at the gray level [29]. Note that there are several thresholds that separate the image into several parts (t1,t2,t3, ...,tm). Consequently, kapur's entropy is obtained by means of the following equation [56]:

$$Max\ J\ (t1, t2, t3, \ldots\ldots, tm) = H0 + H1 + H2 + \ldots + Hm \tag{8}$$

where.

$$\begin{aligned} H_1 &= -\sum_{i=0}^{t_1-1} (p_i/\omega_0)\ln(p_i/\omega_0), \omega_0 = \sum_{i=0}^{t_1-1} p_i; H_3 = -\sum_{i=t_2}^{t_3-1} (p_i/\omega_2)\ln(p_i/\omega_2), \omega_2 = \sum_{i=t_2}^{t_3-1} p_i; \\ H_2 &= -\sum_{i=t_1}^{t_2-1} (p_i/\omega_1)\ln(p_i/\omega_1), \omega_1 = \sum_{i=t_1}^{t_2-1} p_i; H_m = -\sum_{i=t_m}^{K-1} (p_i/\omega_m)\ln(p_i/\omega_m), \omega_m = \sum_{i=t_m}^{t_m-1} p_i \end{aligned} \tag{9}$$

Where H_1, H_2,...,H_m reflect entropy values and ω_0, ω_1, ω_2,..., ω_m indicate probabilities of segmented class C_0, C_1, C_2,...,C_m, respectively [29]. In both of the above methods, there are constraints which are defined as follows as: t1 < t2 < t3 < ... < tm.

4 Hybrid sine-cosine crow search algorithm (SCCSA)

4.1 Overview of crow search algorithm

Askarzadeh is developed a population-dependent metaheuristic optimization algorithm (i.e. CSA-crow search algorithm [7], which is simulated by intelligent crow behaviour. It is based on the concealed location of the excess food stock [7]. Crow is moral in stealing food from other birds. Keep watching other birds find out where they are hiding their food. This would allow crow to take food from other birds if they left the hiding place [7]. This action has encouraged by crows to develop algorithm called the CSA. Due to various awareness of other birds, crows change the location on the basis of the following formula [32].

$$X_i^{t+1} = \begin{cases} X_i^t + r_i \times f l_i^t \times |m_i^t - X_i^t|, & r_i < AP_i^t \\ randomposition & otherwise \end{cases} \tag{10}$$

Where AP_i^t is the jth crow's consciousness. When the victim bird realizes the crow i follow, it tries to get the crow to a random place. Keep in mind that a crow j is randomly chosen for each crow i to change crow ith location. Figure 1 illustrates the crow's position update in CSA within the search space.

4.2 Overview of sine-cosine algorithm

A mathematical motivation focused on the trigonometric functions of the sine and cosine is proposed in 2016 by Mirjalili [2]. With the help of the following formula, the SCA updates the location of the particles within solution space to the position of the best solution [32].

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times \sin(r_2) \times |r_3 P_i^t - X_i^t|, & r_4 < 0.5 \\ X_i^t + r_1 \times \cos(r_2) \times |r_3 P_i^t - X_{ii}^t|, & r_4 \geq 0.5 \end{cases} \tag{11}$$

Here, X_i^t refers the position at which it stands, $P_i^t$is the best solution position and r1,r2, and r3 numbers are produced randomly between zero and one. r1 demonstrates the direction for the update, r2 defines the distance for the update, r3 guarantees a proper balance between emphasis and desalination by creating a random weight, and r4 chooses a movement of the sine or cosine [32]. The effect of difference between the motions on the next position with the range in [−2, 2] in the sine and cosine movements are shown in Fig. 2.

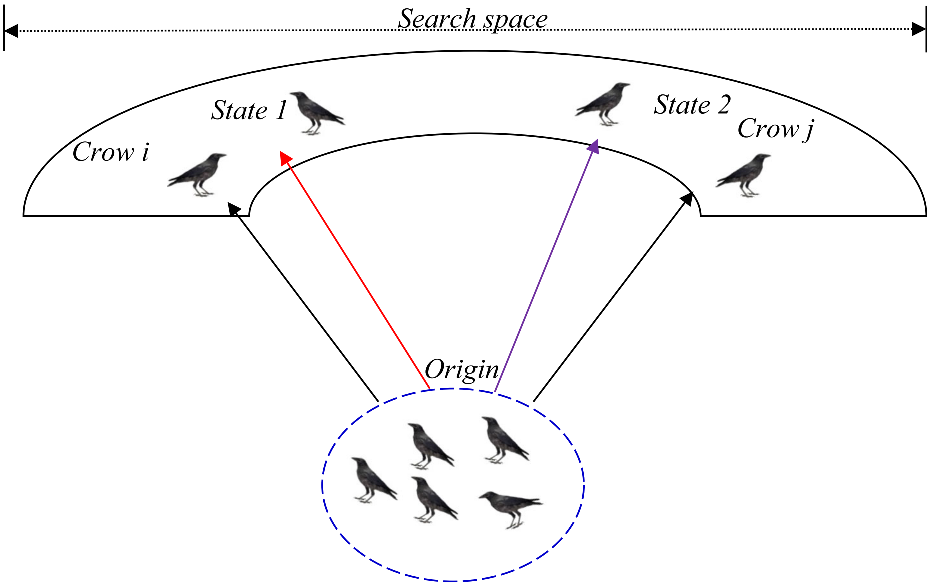


Fig. 1 Crows position update in CSA within the search space

4.3 Hybridization of sine-cosine and crow search algorithm

The main consideration of the proposed hybrid algorithm (i.e. SCCSA) uses the CSA [32]. The CSA first drawback is that the search agents do not necessarily adopt the best solution they have ever achieved [32]. Further, if $r_i \leq AP_i^t$ is implemented, the search agents change their location to a random place in solution space, which reduces the CSA's efficiency [32]. Therefore, first of all, it is considered to increase the efficiency of CSA to update and solution to the best solution to date or on the basis of the random status of the search agent as follows [32]

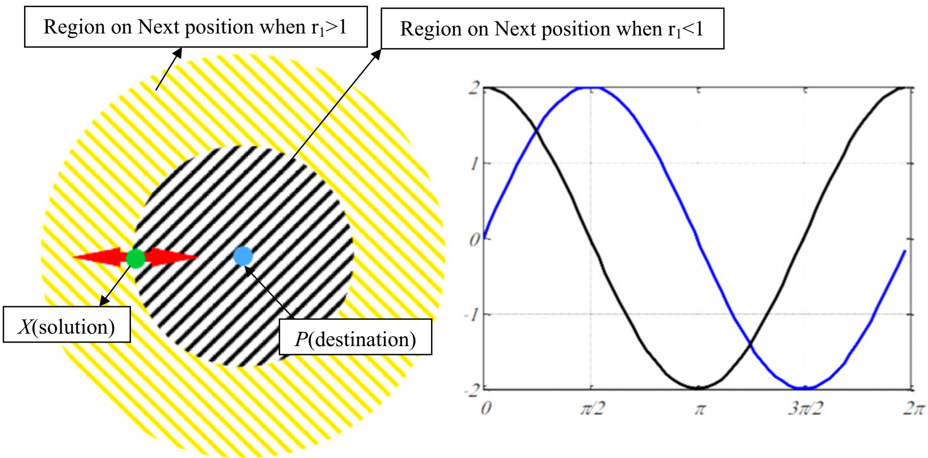


Fig. 2 Effects of sine and cosine in Eq. 11 on the next position with the range from −2 to 2 [32].

$$X_i^{t+1} = \begin{cases} \text{update the position based on the position of the best solution} & r_1 < 0.5 \\ \text{update the position toward a randmly chosen search agent} & r_1 \geq 0.5 \end{cases} \quad (12)$$

Where r1 is a 0 to 1. Then, a CSA updating procedure or SCA movements can be used by each search agent to update its position accordingly [32].

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times \sin(r_2) \times |r_3 P_i^t - X_i^t|, & r_4 < 0.3 \\ X_i^t + r_1 \times \cos(r_2) \times |r_3 P_i^t - X_i^t|, & 0.3 \leq r_4 \leq 0.6 \\ X_i^t + r_1 \times fl_i^t \times |m_i^t - X_i^t|, & r_4 \geq 0.6 \end{cases} \quad (13)$$

These steps ensure that all search agents are intelligent and do not create random, low-quality solutions [32]. Therefore, a different approach can be adopted by each search agent in the solution area, thereby increasing its searching capacity. To maximize the use of metaheuristics, ensuring an efficient trade between exploration and exploitation is important [32]. The solution space must be exploited in the first course of iterations; however, we focus more on exploitation in the final iterations [32]. To this effect, the following method is used in SCCSA throughout focus more on experimentation in the first and last iterations [32]

$$r_1 = a - t\frac{a}{T} \quad (14)$$

Where t represents current iteration, a is a constant and T is the maximum generation.

5 Proposed multilevel thresholding based SCCSA algorithm

As described in the previous section, SCCSA was selected for multilevel image thresholding. The SCCSA algorithm has been proved to be more efficient for optimizing the objective function in the large search space, when an optimum solution is required [32]. The SCCSA algorithm proposed is designed to identify optimal thresholds within gray [0, L-1] levels by optimizing the function of either otsu's or kapur's entropy. In the proposed algorithm, the N crow individuals (the size of the flock) are believed to occupy randomly a position in d-dimensional space and it is supposed that in their first positions they hide their food. Within the algorithm, the sine-cosine algorithm based movements were applied and accordingly all the crows are generating new positions and updating their memories. The best fitness was then evaluated using either Eq. (7)-otsu's function or Eq. (9)-kapur's entropy is bound by constraint [0,255]. Such step is repeated until the best fitness is achieved (i.e. the maximum objective function value). The block diagram and the complete flow diagram of the proposed SCCSA multi-level threshold image segmentation algorithm are shown in Figs. 3 and 4.

6 Experimental work and evaluation

This section discusses experimental work. To evaluate and validate the proposed algorithm, twelve standard bench mark images such as Starfish (https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images/Gy/12003.html), Lena,

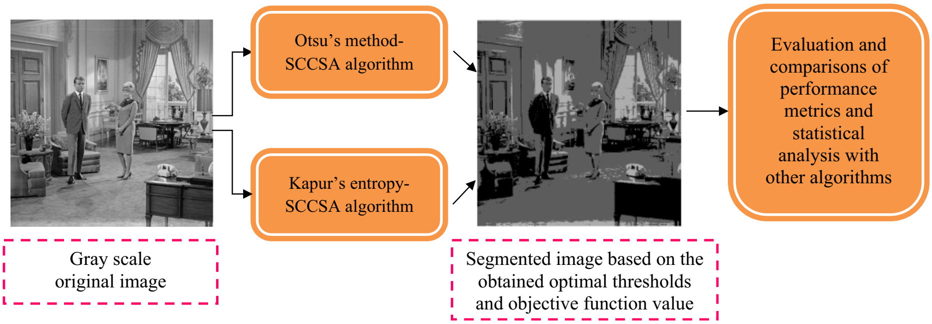


Fig. 3 Block diagram for the proposed SCCSA based multilevel thresholding

Airplane, Cameraman, Hunter, and Living room (http://www.imageprocessingplace.com/root_files_V3/image_databases.htm), Baboon, Pepper and Lake (http://sipi.usc.edu/database/database.php?volume=misc&image=35#top), Boat (http://decsai.ugr.es/cvg/dbimagenes/g512.php) and Gold hill(https://homepages.cae.wisc.edu/~ece533/images/) were taken from different image datasets. All images in JPEG format with size 512 × 512. Table 1 illustrates all original images with their histogram. Due to their multimodal histogram, most images are difficult to segment. In order to achieve better results for such multimodal histograms, sophisticated multi-level threshold segmentation is necessary for those images.

To examine the viability of the proposed SCCSA algorithm in relation to other state-of -the-art ICSA, SCA, CSA and ABC algorithms for optimally evaluated objective function and performance measures. These algorithms were individually implemented and their search efficiency was tested using the Matlab 2014 environment on the Intel(R) Core(TM) i3-6006 U CPU 8GB RAM at 2 GHz running in Windows 10. The search route for crows in SCCSA, bees in ICSA, agents in SCA, crows in CSA and bees in ABC is extremely different and can be merely applied in the formulated problem. These algorithms have been initialised in similar conditions, but have a very different search nature for the same objective function [10]. For each algorithm, various control parametres have been examined to better explore and explore the features of all algorithms and this helps to bring each algorithm to a faster convergence. The selected control parameters of each algorithm for SCCSA, ICSA, SCA, CSA and ABC are taken from the original reference. The optimum solution is found for the set of m threshold values in every candidate solution in all algorithms [6, 38]. Thus, within [0, L-1] at each aspect at the initialization stage of all algorithms, a population of the solution is formed alone. The population is created first, then the solution of the population is allocated to a fitness value and then the optimum threshold values for its selection criteria are calculated by each algorithm. The elapsed CPU time to reach the desired accuracy with each algorithm is considered. In this way, the stopping condition for all algorithms is based on the objective function value and is not on the number of iterations. Since the evolutionary and swarm-based algorithms used include randomness and the initial solutions were randomly created for each run, all experiments for each threshold number and image were repeated for 30 times to ensure the credibility of the statistics. The between class variance and entropy of the Kapur was attempted to be maximized for the given iteration number for all algorithms. The 2–5 level

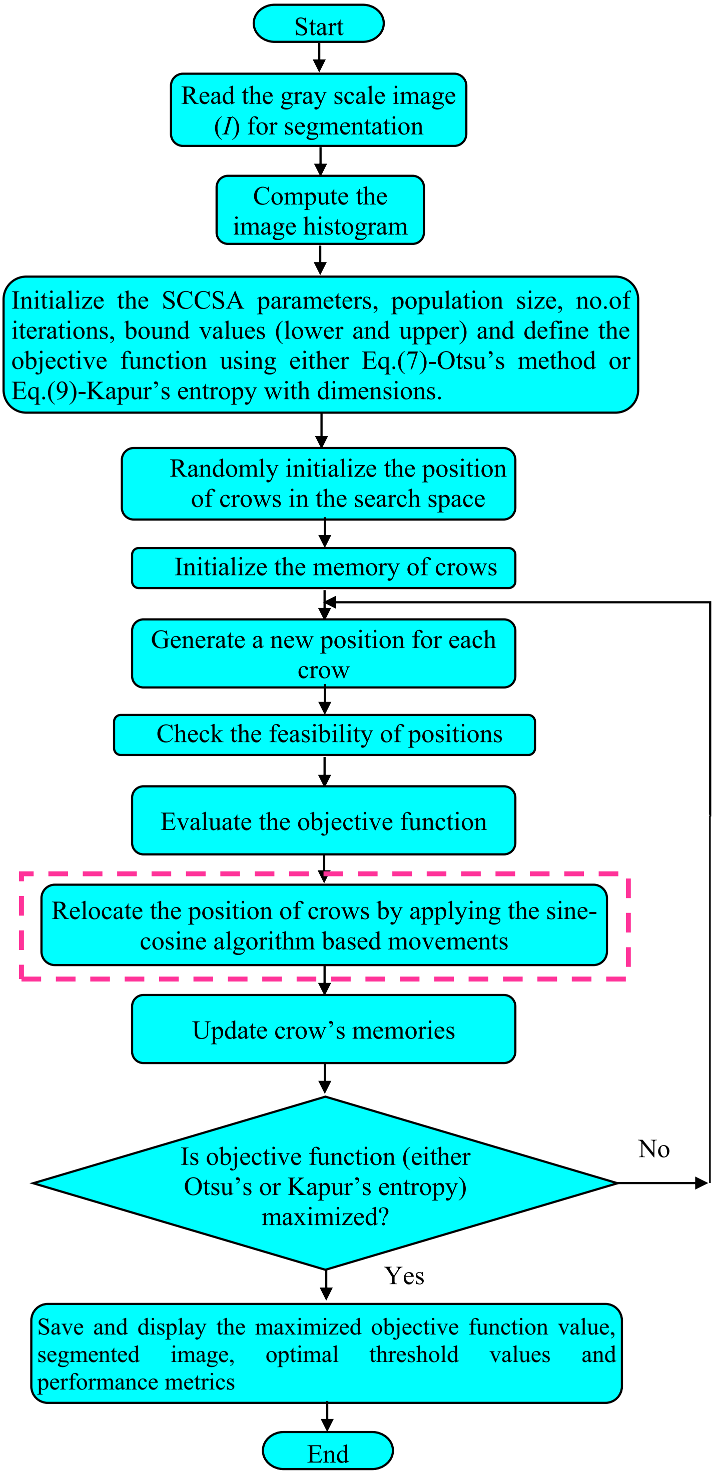
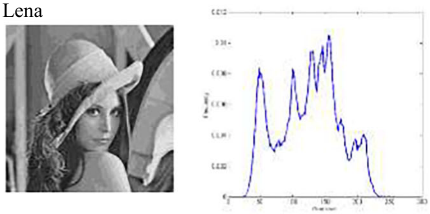
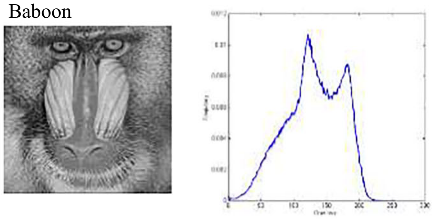
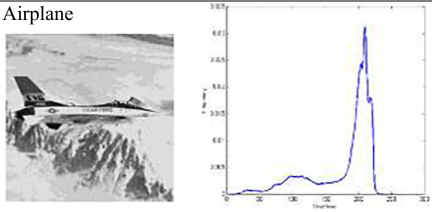
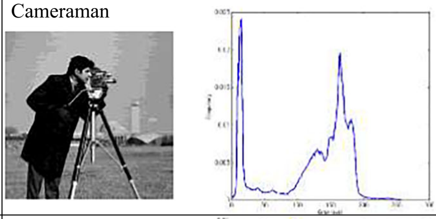
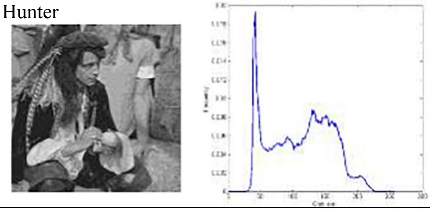
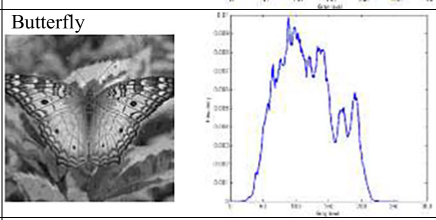
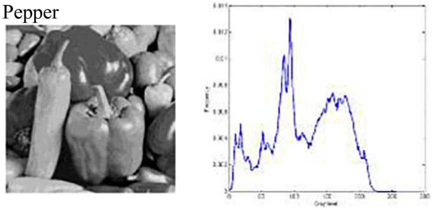
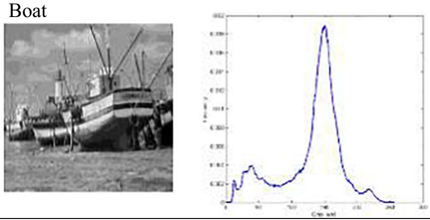
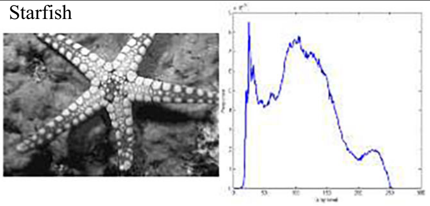
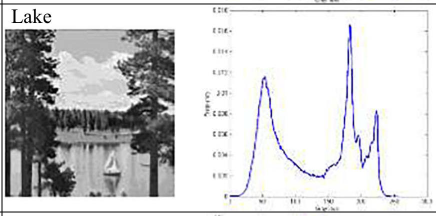
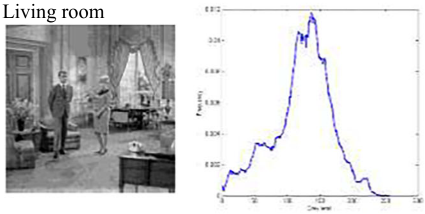
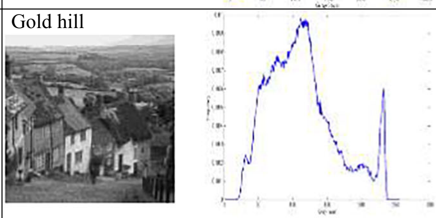


Fig. 4 Flow chart of multilevel thresholding based SCCSA algorithm

Table 1 Original gray scale test images and related histograms

Lena 	Baboon 
Airplane 	Cameraman 
Hunter 	Butterfly 
Pepper 	Boat 
Starfish 	Lake 
Living room 	Gold hill 

thresholding was considered for all running environment algorithms for the purpose of visualizing better perception and fidelity assessment of segmented images.

Generic output metrics are measured as: PSNR, standard deviation, mean, SSIM, FSIM, in subsequence to the values of final objective function values(Jmax), thresholds and CPU time. The stability and efficiency of all the algorithms are evaluated by the mean and standard deviation (STD) values for the each objective function by the following Eq. (15) below [8]:

$$\mu = \frac{\sum_{i=1}^{k} \sigma_i}{k}, STD = \sqrt{\sum_{i=1}^{n} \frac{(\sigma_i - \mu)}{k}} \tag{15}$$

Where, σi- best fitness value of the ith run of the algorithm. μ- mean value of σ and k- the number of runs for each stochastic (i.e. k = 30 times). The lower STD value here means that the algorithm uses the objective function to have greater stability. In the decibel (dB) unit, PSNR values are determined for measurements of the dissimilarity between the original and the segmented images. The consistency metric thus shows the degree of similarity between the segmented and the original on the basis of MSE of each pixel [42]:

$$PSNR = 20\log_{10}\left(\frac{255}{RMSE}\right)_{,\text{(in dB)}} \tag{16}$$

where,

$$RMSE = \sqrt{\frac{\sum_{i=1}^{X} \sum_{j=1}^{Y} (I(i,j) - Seg(i,j))^2}{X \cdot Y}} \tag{17}$$

where, 255 is the maximum gray value, I and Seg are original and segmented images of size X.Y, respectively. Generally, the higher value of the PSNR indicates the good quality of segmentation. The SSIM is used to assess the visual similarity between the original and the reconstructed images. This index combines comparisons of luminance, contrast and structure. In addition, it satisfies symmetry, constraint and unique maximum properties. The SSIM metric can be modeled as follows [42].

$$SSIM(I, Seg) = \frac{\left(2\mu_I \mu_{Seg} + c_1\right)\left(2\sigma_{I,Seg} + c_2\right)}{\left(\mu_I^2 + \mu_{Seg}^2 + c_1\right)\left(\sigma_I^2 + \sigma_{Seg}^2 + c_2\right)} \tag{18}$$

where μI - mean intensity of the image I, μSeg - mean of the image Seg, σ_I^2- the variance of I. σ_{Seg}^2- the variance of Seg. $\sigma_{I,\,Seg}$- the covariance of I and Seg. The c1 and c2 are the constants, and are included to avoid instability when $\mu_I^2 + \mu_{Seg}^2$ are very close to zero, which are mathematically represented as: c1 = (k1L)2 and c2 = (k2L)2. By default, k1 = 0.01 and k2 = 0.03 were taken for computation. And, L is the gray level number in the image. Improved performance is achieved when the SSIM metric is reached at a higher value. The FSIM metric is being used to determine and measure the resemblance among the two images as [8]:

$$FSIM = \frac{\sum_{X \in \Omega} S_L(X) PC_m(X)}{\sum_{X \in \Omega} PC_m(X)} \tag{19}$$

where

$$\begin{aligned} S_L(X) &= S_{PC}(X)S_G(X); \\ S_{PC}(X) &= \frac{2PC_1(X)PC_2(X)+T_1}{PC_1^2(X)+PC_2^2(X)+T_1}; \\ S_G(X) &= \frac{2G_1(X)G_2(X)+T_2}{G_1^2(X)+G_2^2(X)+T_2} \end{aligned} \tag{20}$$

The constants here are T_1 and T_2. $T_1 = 0.85$ and $T_2 = 160$ values were selected. The G is the gradient of the image and is defined as mathematically

$$G = \sqrt{G_x^2 + G_y^2} \tag{21}$$

The PC is the compatibility of the phase and expresses itself as:

$$PC(X) = \frac{E(X)}{(\varepsilon + \sum nA_n(X))} \tag{22}$$

An(X) indicates the localized intensity on scale n and E(X) indicates the magnitude of response vector at position X on scale n and ε is a small positive constant. The higher FSIM valve is seen as enhanced threshold approach efficiency.

7 Results and discussion

Population-based SCCSA, ICSA, SCA, CSA and ABC algorithms have been tested on 12 sets of different test images in this experimental study. The sample of visualization results after applying the proposed SCCSA algorithm to the tested image (i.e. Boat) is illustrated in Tables 2 to 3 for the representation purposes. Table 2 presents the results of Otsu's method and, as shown in Table 3 for Kapur's entropy results, the threshold values range from 2 to 5 levels. Figuring out, Tables 2 and 3 provide detailed information on the convergence characteristics of the SCCSA algorithm, the different optimal threshold values associated with the histogram and the segmented image. It is clearly noticed from Tables 2–3, The qualitative evaluation of visual effects shows that the gray scale images are gradually segmented with threshold numbers that from $m = 2$ to $m = 5$. With higher threshold numbers ($m = 5$), each image's quality is significantly greater and provides a clearer information than $m = 2$, 3 and 4 for all the images tested for both methods, with a greater number of thresholds ($m = 5$). Similarly, the same visual results were observed in all other images tested were observed. Further, the multi-level thresholding graphical segmentation results of the SCCSA and the other compared algorithms (ICSA, SCA, CSA and ABC) are shown in Figs.5 (otsu's method) and 6 (kapur's entropy) for star image with $m = 5$ threshold level. From these figures, it can be found that the proposed SCCSA algorithm achieved the best segmentation performance compared with other algorithms. Besides, sample of otsu's and kapur's based segmented result of images (i.e. airplane, peppers, lake and goldhill) using the hybridized SCCSA algorithm for $m = 2$–5 thresholds. Such figures show that, under different thresholds, the SCCSA showed strong segmentation outcomes for different images. Furthermore, these figures show that the segmented images are better for an increasing threshold Figs 6 and 7.

Quantitative results are compared and illustrated in Tables 4-10 as well as in Figs. 8-11 with four other state-of-the-art algorithms. For all the five algorithms, the optimal objective function values (i.e. for otsu's and kapur's) are reported in Table 4 over 30 evaluation runs for each

Table 2 SCCSA results on boat image using otsu's method

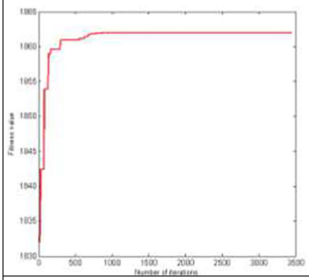
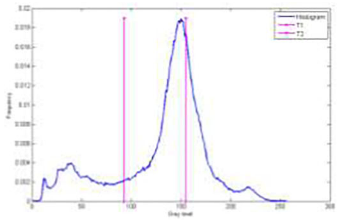
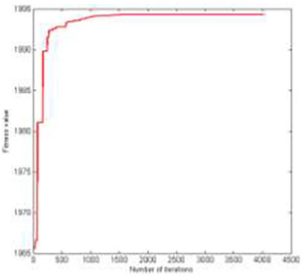
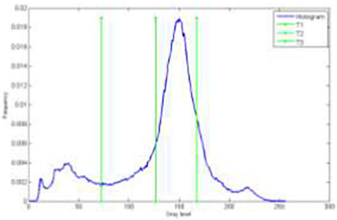
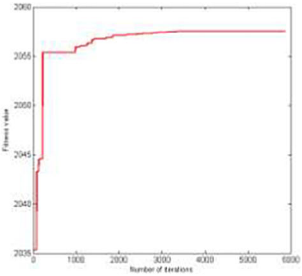
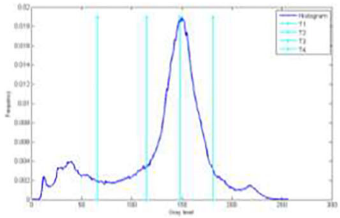
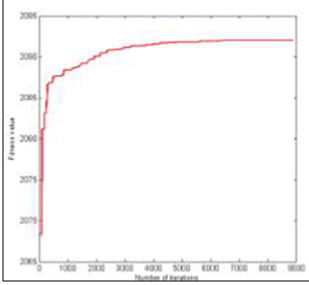
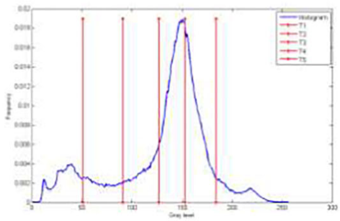
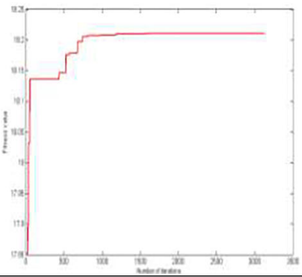
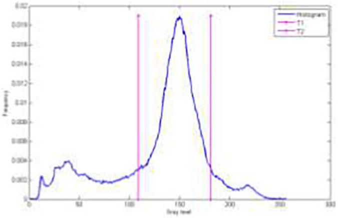
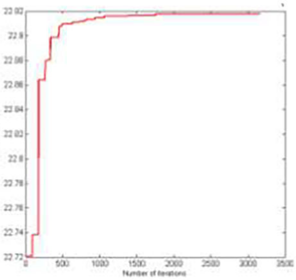
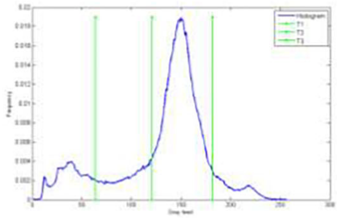
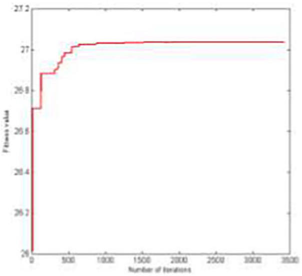
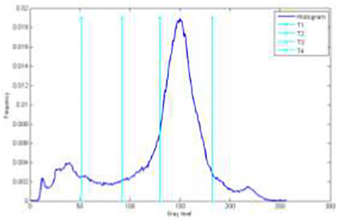
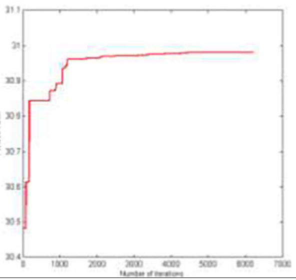
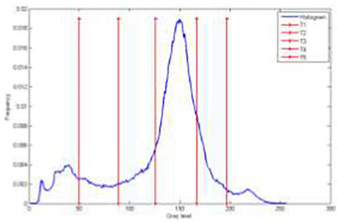
Convergence plot	Histogram	Segmented image
m=2 		
m=3 		
m=4 		
m=5 		

image being tested. Since, the two methods are a maximization problem, each target function should be as high as possible to achieve the optimal threshold. It is apparent from Table 4, that all algorithms performed were almost equal in the increase of thresholds (from $m = 2$ to $m =$

Table 3 SCCSA results on boat image using kapur's entropy

Convergence plot	Histogram	Segmented image
m=2 		
m=3 		
m=4 		
m=5 		

5). In addition, for images tested, the SCCSA offers higher objective function values than all other algorithms for both methods. However, the otsu's based obtained SCCSA objective function values for the pepper image were lower when compared with the ICSA algorithm for

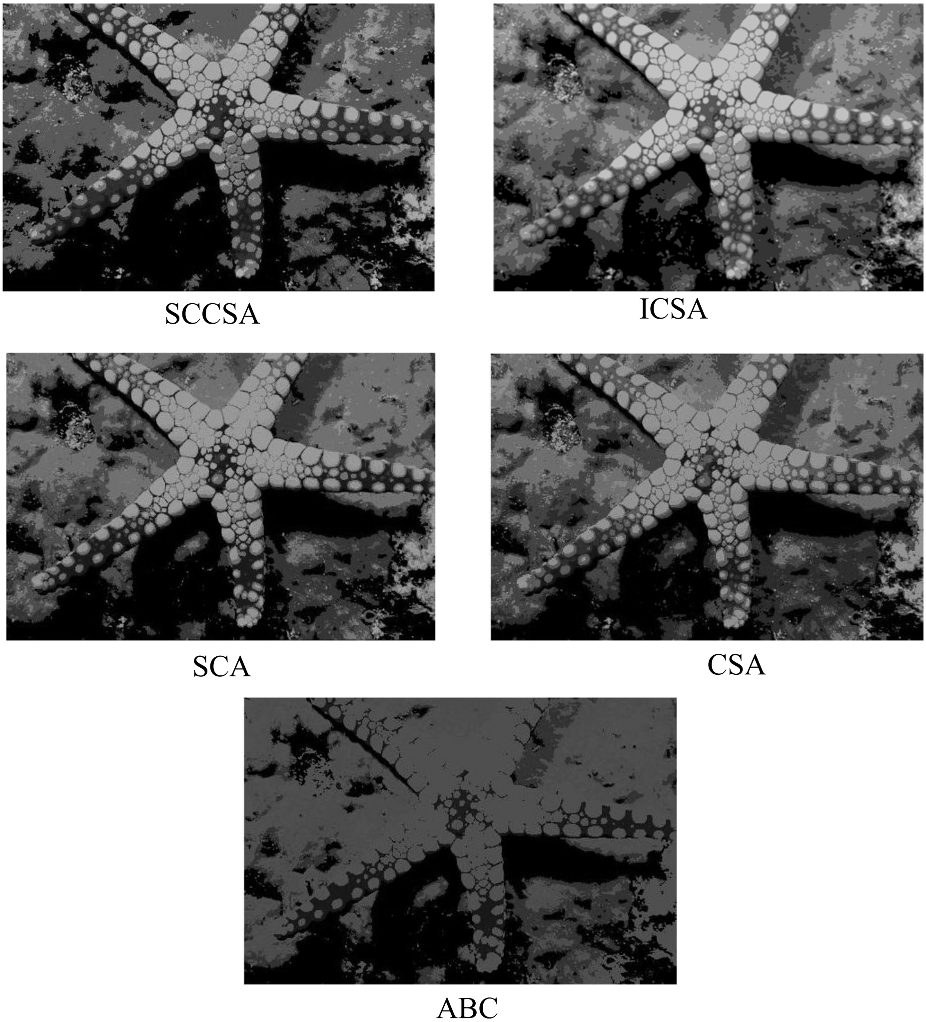


Fig. 5 Segmented images of starfish at $m = 5$ using SCCSA, ICSA, SCA, CSA and ABC based on otsu's method

the $m = 4$ threshold level. Whereas, the kapur's entropy results from the results of objective function values in the SCCSA on the hunter image was produced by a lower value than the ICSA algorithm when the threshold level was $m = 5$. This occurred due to the randomness of the different swarm approaches, the results may vary in some cases. Tables 5 and 6 sum up the chosen thresholds for the test images in the total range of the gray values obtained from the otsu's and kapur's approaches for determining the consistency of the optimum threshold values. The table results show that the dedicated threshold detection results are not exactly similar (i.e. this is due to the stochastic and random design of all algorithms) and that the values are in many cases more scattered and broader. Compared to other algorithms, the numerical results from the PSNR results of the proposed SCCSA algorithm are presented in

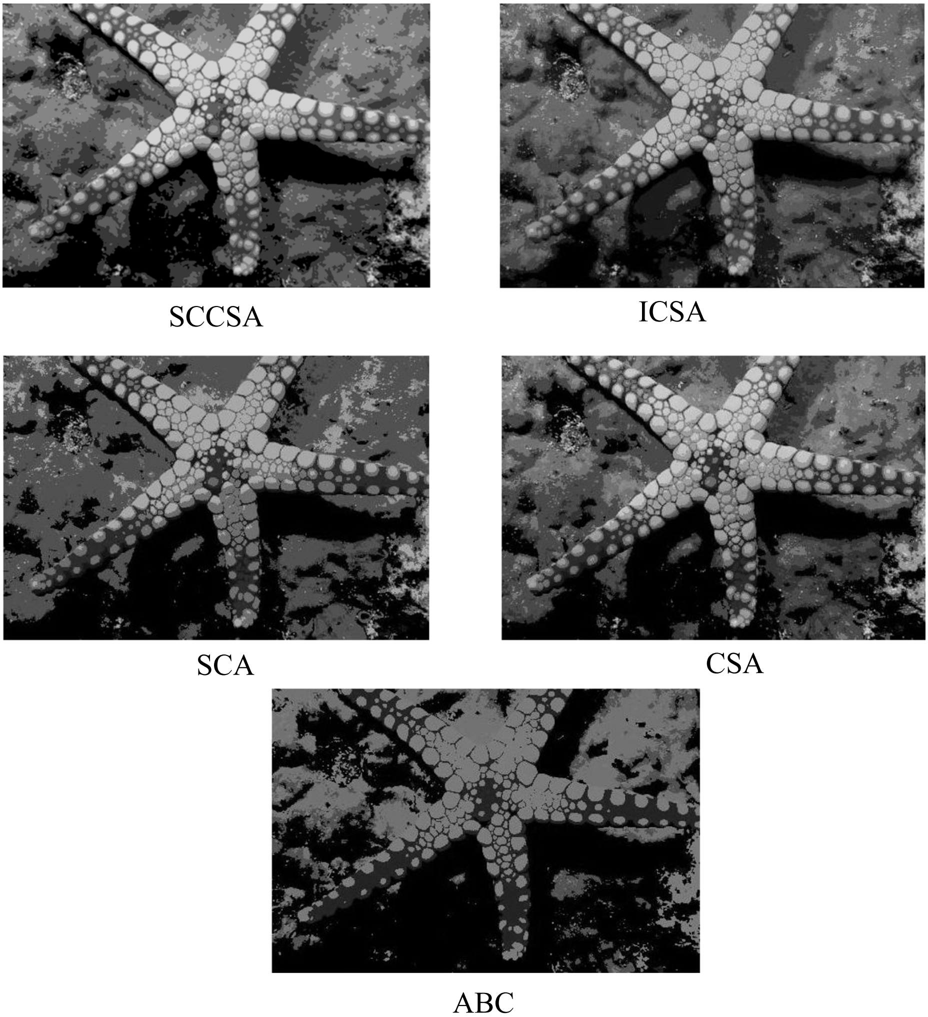


Fig. 6 Segmented images of starfish at $m = 5$ using SCCSA, ICSA, SCA, CSA and ABC based on kapur's entropy

Table 7. PSNR values are observed with increasing threshold values, from $m = 2$ to $m = 5$, as can be seen in Table 7. Moreover, Table 7 identifies several aspects. In otsu's method, average PSNR values compared to SCCSA are improved by 26.17%, 46.16%, 65.26% and 96.69% respectively with ICSA, SCA, CSA and ABC. In the case of Kapur's method, average PSNR values are compared with the SCCSA algorithm by 24.56% for ICSA, 48.57% for SCA, 69.37% for CSA and 96.08% for ABC algorithms.

The values of SSIM and FSIM are evaluated using eqs. (18 and 19) for each image of all algorithms with threshold numbers from 2 to 5 for the entropy methods of otsu's and kapur's. The obtained values of SSIM values is given in Table 8. The SSIM values for the image sample of boats as shown in Figs. 8 and 9 were obtained. These figures show that the SCCSA

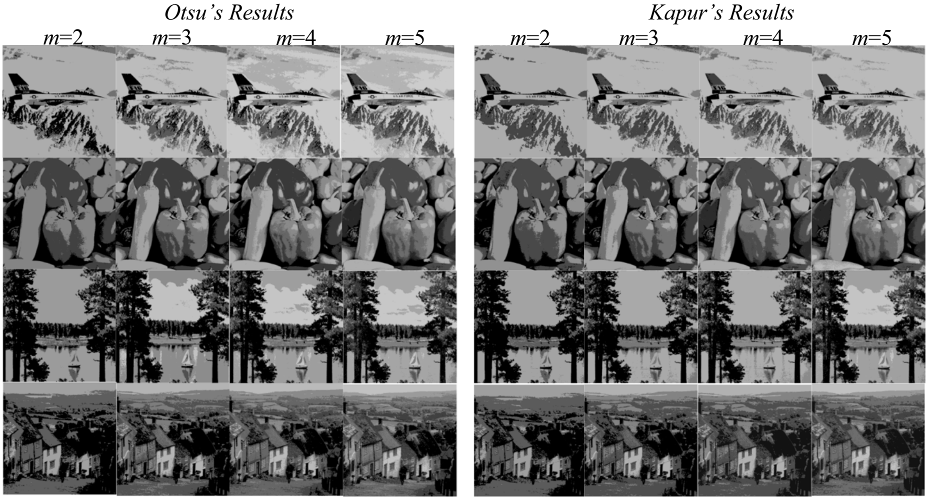


Fig. 7 Sample of otsu's and kapur's based segmented results of images using the hybridized SCCSA algorithm for $m = 2$–5 thresholds. From top to bottom: Airplane, Peppers, Lake and Goldhill

results in higher values than the rest of the algorithms, which means that the SCCSA algorithm is more sensitive to the threshold increase for the both methods. All other images tested have been observed with the same trend. Table 8 displays the FSIM values obtained. Then, the FSIM values for the sample of starfish image as depicted in Fig. 10 for the otsu's method and Fig. 11 for kapur's entropy are then tested for visual similarity between the original and the segmented images. With both methods, it is noteworthy that SCCSA produces higher FSIM values for all threshold levels. Similar findings were witnessed for all other images tested Table 9.

Calculating the CPU time required by any algorithm is important because real-time applications need rapid execution. Accordingly, the computational efficiencies of all algorithms are compared using the best CPU time (in seconds) required to converge on the solution as stated in in four different threshold levels is shown in Table 10. Since the comparison results of CPU time values in the proposed SCCSA are lower in both methods than those of ICSA, SCA, CSA and ABC algorithms. In regard of CPU time for SCCSA algorithm is primarily due to each search agent acquiring experience (exploration and exploitation mechanisms) from the population in the search region. This capability enables hybrid SCCSA to perform efficient searches within the search space and greatly reduce computation time. Although increasing the number of thresholds increases the CPU time of all algorithms, when compared to other algorithms, specifically the CPU time of SCCSA, it has the lowest growth time. The Mean values over 30 runs are provided by five algorithms for each image with threshold numbers from 2 to 5 in Table 11. In terms of mean, this parameter shows the average value of the objective function values during iterations, reflecting to some extent algorithm stability (Sun et al. 2017). The result of the Mean values is almost same when $m = 2$ and 3 threshold values. In SCCSA, when the m value is greater than 3, the mean value of the goal function is mostly higher than those of other algorithms. The proposed SCCSA therefore provides a more accurate segmentation of the images tested. In addition to checking the stability of the proposed SCCSA with four other algorithms, the values and standard deviation (i.e. STD) are calculated using Eq. (15) and given in Table 12. From the tabular findings, it was

Table 4 Comparative analysis of optimal objective function values

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	1962.457	1961.308	1960.506	1958.395	1957.933	12.3545	11.2176	10.1821	9.9769	9.5616
	3	2163.355	2161.953	2161.780	2160.699	2159.142	16.8950	15.5989	14.9000	12.9405	12.5463
	4	2221.988	2220.215	2219.245	2218.216	2218.095	18.8295	18.4322	17.9885	16.9535	15.8659
	5	2230.889	2229.853	2228.323	2227.596	2225.502	21.2145	20.6041	20.3765	19.2690	18.1181
Baboon	2	1555.789	1554.065	1553.936	1552.808	1550.391	13.2178	8.60610	10.1881	10.3577	11.6125
	3	1647.087	1646.947	1645.354	1643.444	1642.960	15.2516	15.3424	14.9489	12.0887	10.2657
	4	1701.297	1700.024	1699.269	1698.724	1696.122	18.3674	18.6226	17.8264	17.1002	14.4936
	5	1728.765	1729.539	1727.961	1726.870	1724.973	22.2375	21.6190	20.8389	19.4468	18.9204
Airplane	2	1930.708	1928.150	1927.252	1926.199	1925.751	12.1115	11.0016	9.2438	8.4926	7.7446
	3	2007.552	2006.417	2005.161	2004.386	2004.041	15.0871	13.7481	13.6704	12.0887	11.1723
	4	2054.145	2053.176	2051.554	2050.075	2049.520	18.3806	17.0785	16.4669	14.4848	13.9993
	5	2079.989	2078.501	2076.871	2074.297	2074.265	20.8713	19.7998	18.1044	17.6572	15.1792
Camera man	2	3651.956	3650.501	3649.151	3648.404	3647.335	12.4897	11.6844	9.9177	8.1921	7.8284
	3	3727.875	3726.792	3725.788	3723.098	3722.604	15.9435	14.4591	13.0347	12.2355	10.8729
	4	3782.466	3780.059	3779.608	3778.266	3777.207	20.8745	19.8959	19.3133	18.8953	17.8765
	5	3813.799	3812.828	3811.371	3809.964	3808.411	23.3201	22.3793	21.6988	20.1464	19.4189
Hunter	2	3064.211	3063.907	3061.301	3059.286	3059.123	17.5449	16.6616	15.9189	14.2444	12.8004
	3	3213.447	3111.723	3110.540	3109.343	3108.622	21.9171	20.1284	19.7517	18.7601	17.5006
	4	3269.517	3268.049	3267.383	3266.652	3265.993	26.3767	25.0259	24.4299	23.3685	22.0844
	5	3308.142	3307.590	3306.032	3305.694	3304.254	29.2975	30.0875	28.8459	27.6313	26.6166
Butterfly	2	1554.887	1554.613	1553.643	1553.633	1553.549	12.7468	12.4619	12.2401	11.6700	11.2134
	3	1671.834	1671.283	1671.148	1670.954	1670.427	16.7482	16.4732	15.6565	15.6278	14.9822
	4	1713.558	1713.293	1712.796	1712.564	1712.253	19.3044	19.2385	18.9652	18.6704	17.8046
	5	1738.429	1738.319	1737.929	1737.634	1737.571	21.7393	21.3441	21.0585	20.4098	20.4049
Pepper	2	2513.916	2512.796	2511.442	2509.209	2508.049	12.9346	10.1813	10.0457	8.3555	8.1918
	3	2683.371	2682.660	2681.059	2678.697	2680.513	15.8843	14.4981	12.7318	12.1873	10.0537
	4	2739.327	2740.566	2738.537	2737.289	2736.925	18.4418	17.3791	16.8309	15.4949	15.3387
	5	2786.154	2786.071	2785.237	2784.001	2782.380	21.0368	18.8059	17.8590	16.9774	16.4932
Boat	2	1865.284	1864.893	1864.702	1864.275	1863.383	19.0404	18.5802	18.2166	18.1280	18.0137
	3	1998.000	1997.434	1996.823	1996.726	1995.696	23.4064	23.1986	23.0553	22.1099	22.1087
	4	2060.247	2059.809	2059.251	2058.661	2058.285	28.9171	27.2867	27.5442	27.5283	27.4537

Table 4 (continued)

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
	5	2093.802	2092.669	2092.339	2091.471	2091.043	30.8679	30.7472	30.3323	30.0507	30.0112
Starfish	2	2492.857	2492.698	2492.235	2491.233	2491.205	19.3936	18.9279	18.1992	17.7630	17.6876
	3	2718.143	2717.977	2717.651	2717.382	2717.327	23.9641	23.5019	23.2895	23.2712	23.0793
	4	2803.993	2803.021	2802.697	2801.947	2801.913	28.8215	28.1928	28.1299	28.0158	27.8604
	5	2849.841	2849.269	2848.429	2846.925	2846.834	32.1378	32.0082	30.8290	30.6995	30.6773
Lake	2	3970.961	3970.854	3970.592	3969.897	3969.430	17.4339	17.0336	16.9694	16.8309	16.6487
	3	3981.275	3980.933	3980.717	3980.531	3979.746	22.8559	22.7820	22.3283	22.3021	21.8161
	4	4097.250	4096.523	4095.605	4094.230	4092.178	27.5915	27.3603	27.0902	26.6011	26.2195
	5	4123.861	4122.146	4120.604	4119.347	4118.229	31.3392	31.0120	30.3676	30.0938	29.8387
Living room	2	1630.735	1630.029	1629.275	1628.488	1628.008	12.8070	12.6888	12.3765	11.8488	11.5522
	3	1748.533	1747.773	1747.337	1746.843	1746.129	15.9944	15.9339	15.7284	15.4110	14.9857
	4	1861.196	1861.028	1860.030	1859.590	1859.010	18.9539	18.8948	18.2005	17.5603	17.4670
	5	1916.253	1916.079	1915.505	1915.405	1915.390	21.9476	20.5804	20.5674	20.3083	20.1457
Gold hill	2	1564.535	1564.393	1564.352	1563.976	1563.889	20.7311	19.3878	19.2095	19.0497	18.0884
	3	1608.738	1608.331	1608.305	1607.269	1607.190	24.3793	24.2813	23.8632	22.5487	22.4793
	4	1694.260	1694.248	1693.395	1693.337	1693.287	28.5311	28.3043	27.9324	27.1769	27.1280
	5	1712.832	1711.654	1711.535	1710.502	1710.117	31.5955	31.0201	30.5959	30.3334	29.6248

Table 5 Comparative analysis of best threshold values obtained by otsu's method

Test image	*m*	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	88, 156	91, 152	92, 151	92, 152	94, 149
	3	77,126, 171	80, 126, 171	58, 111, 172	79, 125, 170	79, 127, 170
	4	75, 114, 145, 185	75,114, 146, 181	51, 98, 142, 176	73, 112, 144, 179	78, 112, 134, 175
	5	65,93,121,149,182	70,109,136,160,188	49,103,128,152,186	71,107,134,158,186	79,110,140,167,188
Baboon	2	101,141	97, 144	87, 148	97, 149	96, 149
	3	83,122, 159	85, 125, 161	100, 148, 182	85, 125, 161	85, 126, 166
	4	71, 105, 136, 167	72, 106, 137, 168	71, 112,138, 184	71, 105, 136, 167	79, 105, 140, 174
	5	66,98,124,148,174	67, 99, 125,149,174	74,102,138,158,181	66, 97, 123,147,173	74,104,134,161,180
Airplane	2	112, 172	116, 177	91, 213	113, 173	117, 174
	3	89,141, 188	85, 142, 194	96, 138, 215	92, 114, 190	99, 158, 193
	4	83, 128, 172, 202	81,122,171,206	58, 112, 138, 203	84, 129, 172, 203	84, 125, 168, 201
	5	67,106,142,179,204	65,114,139,175,210	72, 95, 131,148,206	68,106,143,180,205	60,101,138,177,204
Camera man	2	70, 144	70, 143	71, 145	72, 144	71, 143
	3	57, 116, 154	67, 118, 155	53, 114, 162	59, 119, 156	71, 134, 166
	4	41, 94, 139, 169	45, 91, 128, 171	49, 112, 148, 169	42, 95, 140, 170	65, 121, 147, 172
	5	36, 83,122, 149,173	40, 85,125,151,171	44, 92,140,173,204	36, 82,122,149,173	45,78, 121,146,172
Hunter	2	49, 115	51, 116	64, 122	51, 116	52, 115
	3	31, 91, 143	36, 86, 135	49, 86, 142	36, 86, 135	39, 86, 135
	4	30, 72, 111, 146	27, 73, 121, 156	33, 84, 125, 160	27, 65, 104, 143	36, 84, 130, 157
	5	33, 94,123,150,181	29, 53, 88,122,152	30, 70,110,146,180	22, 53,88,112,152	37, 85,125,154,177
Butterfly	2	100, 152	94, 149	94, 168	97, 151	99, 150
	3	81, 118, 159	75, 126, 160	84, 123, 174	82, 119, 160	79, 119, 164
	4	73, 101, 129, 163	71, 98, 133, 162	72, 114, 141, 180	71, 102, 130,163	80, 115, 145, 178
	5	75,104,125,151,179	71, 99,125,156,180	63, 99,126,160,185	62, 77,109,137,167	75,106,129,157,180
Pepper	2	65, 132	71, 138	67, 134	72, 138	76, 144
	3	61, 116, 163	65, 122, 169	62, 118, 165	65, 122, 169	72, 124, 171
	4	57, 102, 138, 172	49, 88, 129, 171	47, 85, 125, 168	50, 88, 124, 170	57, 92, 130, 172
	5	39,74,108,142,174	48,85,118,150,179	42,78,112,145,176	45, 81,114,149,189	56, 84,115,150,179
Boat	2	87, 148	99, 172	77, 189	102, 189	93, 155
	3	75, 136, 176	95, 157, 189	65, 154, 192	69, 110, 168	73, 126, 167
	4	68, 81, 148, 180	62, 102, 149, 186	64, 98, 131, 170	72, 94, 135, 197	65, 114, 147, 201
	5	48, 96,118,155,186	39, 80,146,187,200	49, 73,126,158,194	66, 90,111,170,202	54,77,139,168,202
Starfish	2	82, 155	83, 157	81, 155	82, 159	86, 160

Table 5 (continued)

Test image	m	SCCSA	ICSA	SCA	CSA	ABC
	3	69, 120, 177	65, 125, 171	61, 118, 175	70, 114, 178	74, 109, 182
	4	61, 101, 138, 186	55, 98, 139, 181	62, 111, 138, 189	59, 105, 138, 183	49, 113, 149, 195
	5	52, 86, 117,150,193	60, 98, 132,159,192	63, 89, 116,154,176	61, 91, 120,159,189	52,82, 142, 168,204
Lake	2	88, 155	86, 158	81, 151	85, 154	88, 155
	3	80, 140, 193	72, 152, 195	74, 145, 200	78, 140, 194	80, 140, 193
	4	70, 112, 158, 197	76, 110, 162, 188	66, 115, 169, 223	67, 120, 158, 201	71, 155, 160, 203
	5	59, 88, 126,165,199	52, 91, 132,169,207	57, 88, 127,166,200	53, 95, 137,166,202	76,111,149,186,208
Living room	2	87, 145	87, 145	84, 152	86, 145	88, 145
	3	73, 128, 163	75, 128, 163	86, 141, 162	76, 153, 163	81, 127, 165
	4	47, 99, 134, 170	56, 97, 132, 168	59, 109, 136, 161	56, 101, 132, 168	62, 97, 143, 178
	5	47, 89,122,148,180	45, 88,120,146,178	61,102,135,166,198	49, 88, 123,155,178	56, 98, 128,156,190
Gold hill	2	94, 161	87, 161	86, 166	84, 165	94, 161
	3	82, 125, 179	80, 146, 177	72, 138,171	63, 137, 182	70, 146, 180
	4	69, 102, 138, 186	57, 112, 147, 182	56, 89, 144, 183	61, 101, 148, 194	58, 95, 138, 190
	5	64,101,125,155,193	53, 98, 147,187,201	46, 76, 150,177,203	63, 91, 117,147,201	63, 91, 117,171,203

Table 6 Comparative analysis of best threshold values obtained by kapur's entropy

Test image	*m*	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	97, 165	99, 165	97, 162	96, 163	80, 150
	3	82, 97, 165	86, 151, 162	88, 156,168	83, 96, 163	70, 109, 160
	4	97, 82, 127,177	92, 129, 162, 191	90, 101, 125, 173	60, 80, 125, 173	56,100,144,182
	5	92,64,97,138,179	94,115,145,170,197	93, 75, 119, 149, 189	91, 71,109,144,185	94,79,115,148,186
Baboon	2	79, 143	61, 126	79, 142	81, 144	76, 146
	3	49, 101, 153	38, 82, 166	44, 99, 154	51, 103, 154	72,130, 181
	4	43, 98, 152, 231	27, 70, 103, 149	39, 74, 114, 159	43, 83, 121, 163	65, 121, 153, 180
	5	31,72,113,159,231	17, 68, 89,120, 164	24, 59, 110, 158, 213	38, 72, 106, 139, 172	73,110,142,161,198
Airplane	2	68,172	69, 171	70, 173	70, 171	80, 175
	3	68, 125, 181	61, 137, 168	68, 126, 182	65, 129, 181	72,121, 191
	4	63, 103, 143, 184	47, 84, 157, 196	68, 126, 182, 232	68, 124, 189, 233	74, 129, 162, 188
	5	59,89,122,154,186	62,121,157,190,212	64, 114, 149, 187, 232	64, 104, 143,184,229	81,118,144,167,192
Camera man	2	125, 196	115, 197	128, 193	127, 196	115, 196
	3	43, 101, 196	45, 112, 190	44, 104, 189	44, 103, 191	46, 138, 192
	4	43, 97, 145, 196	42, 96, 145, 198	44, 97, 146, 197	44, 96, 146, 196	77, 116, 151, 202
	5	24, 61, 99,145,196	27, 61, 99, 146, 198	28, 83,118,154,197	24, 60, 98, 146, 196	24, 95, 121,156,198
Hunter	2	110, 180	83, 186	88, 177	92, 179	83, 179
	3	83, 129, 181	78, 135, 179	59, 118, 178	59, 117, 179	85, 128, 166
	4	22, 84, 131, 181	42, 80, 156, 226	45, 89, 173, 224	44, 89, 133, 179	74, 131, 174, 200
	5	22,74,111,147,183	38, 81, 134,171,213	32, 67, 112, 156, 215	44, 89, 133, 179,222	90,120,164,190,219
Butterfly	2	93, 151	96, 171	97, 212	98, 203	95, 141
	3	93, 151, 226	80, 174, 217	27, 126, 233	28, 125, 242	63, 124, 177
	4	17, 93, 151, 226	25, 150, 180, 214	27, 96, 144, 213	27, 96, 147, 213	21, 113, 162, 188
	5	17,73,114,157,226	41, 91, 137,168,217	27, 83, 108, 162, 220	24, 88, 118, 152, 213	92,116,142,157,182
Pepper	2	73, 145	75, 143	71, 154	66, 143	79, 146
	3	63, 114, 165	36, 98, 159	65, 121, 169	61, 112, 162	104, 144, 180
	4	43, 77, 125, 171	37, 89, 127, 175	71, 137, 151, 253	62, 112, 163, 227	57, 110, 162, 199
	5	40,74,108,147,187	45, 77, 98, 147, 205	65, 122, 152,168, 252	48, 85, 127, 171,227	70,116,138,169,200
Boat	2	109, 181	104, 172	109, 181	120, 201	116, 172
	3	64, 121, 182	70, 129, 170	68, 108, 179	59, 122, 194	62, 114, 187
	4	52, 92, 130, 182	57, 81, 128, 178	72, 97, 144, 192	69, 91, 142, 187	61, 85, 124, 188
	5	50,89,126,167,197	58, 71, 137,186,201	67, 90, 131,182, 201	59, 92, 134, 166, 201	48, 87, 128,164,198
Starfish	2	92, 171	85, 160	86, 154	77, 1632	91, 172

Table 6 (continued)

Test image	m	SCCSA	ICSA	SCA	CSA	ABC
	3	75, 130, 184	90, 169, 222	76, 134, 165	64, 135, 176	89, 142, 214
	4	66, 112, 158, 201	58, 86, 119, 168	71, 90, 129, 175	62, 125, 134, 172	80, 124, 160, 221
	5	56,94,138,171,210	61,131,162,190,221	65, 89, 120, 154, 184	55, 86, 138, 154, 188	67,115,149,171,201
Lake	2	93, 167	98, 168	91, 163	87, 153	94, 163
	3	75, 122, 169	72,125, 161	72,125, 174	73, 120, 170	73, 150, 197
	4	66, 102, 138, 172	70, 111, 140, 173	77, 134, 202, 222	69, 112, 156, 195	41, 121, 166, 200
	5	61,90,118,160,196	84, 99, 133, 167	60, 100, 137, 177	52, 96, 131, 166, 198	46, 93, 128,165,197
Living room	2	94, 175	90, 170	81, 180	89, 170	86, 175
	3	47, 103, 175	47, 98, 174	60, 128, 192	47, 103, 175	73, 158, 187
	4	47,98,149,162,197	47, 103, 159, 196	50, 100, 150, 195	51 98, 149, 197	59, 124, 172, 202
	5	42,85,124,162,197	45, 92, 139,168,236	48, 98, 140, 187, 235	42, 85, 125, 162, 197	72, 97, 119,158,195
Gold hill	2	95, 158	89,181	81, 165	90, 179	91, 168
	3	78, 131, 177	77, 121, 158	79, 109, 162	84, 134, 185	96, 168, 184
	4	18, 78, 131, 177	78, 120, 177, 202	82,110,173,208	85, 113, 193, 220	88, 125, 187, 223
	5	18,67,109,152,192	49, 97, 131,165,199	57,84,134,177,204	64, 101, 148,197,217	55, 91, 137,195,227

Table 7 Comparative analysis of PSNR values

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	15.8192	15.8026	15.7654	15.1121	14.6554	14.9034	14.8460	13.7499	13.4055	13.3616
	3	17.9672	17.6769	17.5617	17.2032	16.1477	17.8956	17.3513	16.9954	16.7213	15.6293
	4	18.8628	18.4810	18.4430	18.3006	17.2281	19.9039	19.5930	18.8596	18.7646	18.1737
	5	19.8209	19.1411	19.0017	18.5589	18.0566	22.5213	22.1503	21.0225	20.8146	20.2459
Baboon	2	15.9388	15.7707	15.2288	15.1820	14.1010	16.9172	16.5132	16.4580	15.4345	15.3507
	3	18.7138	18.6785	17.4915	16.2294	15.0029	17.4213	17.3244	16.3714	16.3218	15.8415
	4	20.7679	19.4664	19.2628	18.9986	18.8252	19.5614	18.6417	18.4682	18.1578	17.4158
	5	21.9640	20.6425	20.1991	20.0643	19.5316	21.8682	20.9757	20.4319	20.0888	19.7712
Airplane	2	16.9228	16.3264	16.0838	14.4314	14.0499	15.9573	14.8713	14.2001	14.0932	14.0507
	3	19.8284	18.4938	17.7321	17.4928	17.1108	18.8202	18.4322	18.2459	17.5620	17.5194
	4	21.8647	20.3201	19.6238	19.5852	19.4502	19.7741	18.7162	18.3690	17.9976	17.9183
	5	24.9543	24.8281	24.7678	23.9707	22.3178	21.8920	20.8626	20.0575	19.4260	19.0111
Camera man	2	18.9001	18.5808	17.4176	16.9154	16.3532	13.2778	13.1921	12.7554	12.2659	12.2105
	3	20.1090	19.7078	19.6728	19.4948	19.0838	15.5058	14.5985	14.5448	13.5660	13.0031
	4	22.2698	21.7469	21.6547	20.9445	20.8539	21.2745	21.1015	21.0653	19.9061	19.1297
	5	23.9255	23.7053	22.6353	22.6325	22.3384	21.9993	21.7099	20.7112	20.4059	19.5427
Hunter	2	17.9814	16.3306	16.2524	16.1924	16.0897	15.8138	15.3599	15.3088	14.3246	14.0571
	3	21.3420	20.4781	19.6505	19.1213	19.0007	19.9687	19.9320	19.7711	19.2914	17.6994
	4	22.5776	21.2994	20.3060	20.2598	19.6053	22.1004	20.9832	20.5262	20.5077	20.1714
	5	24.0261	23.4674	22.8216	21.6221	20.5325	22.2121	22.1723	21.0887	20.8646	20.7284
Butterfly	2	14.0210	13.6401	13.4954	12.3840	12.0522	10.8342	9.2276	9.1402	9.1092	9.0422
	3	17.3248	16.5397	15.3016	14.9611	14.2379	14.9969	14.5283	14.0574	13.8179	13.1564
	4	20.8467	20.0310	19.5202	18.8975	18.2849	17.3476	16.5819	16.1553	15.1531	15.0811
	5	21.9812	21.9280	21.8508	21.6551	20.7922	20.9654	20.8560	19.8233	19.5387	18.0099
Pepper	2	17.1292	16.6648	16.6464	16.5381	15.3962	17.5842	17.3881	16.2941	16.0823	16.0817
	3	19.8426	17.8945	17.7755	17.5475	17.3098	18.5697	18.3748	17.8878	17.6821	17.2188
	4	21.8538	21.2640	20.9231	20.7289	19.4495	19.7973	19.6689	19.2896	18.4943	17.1329
	5	23.4567	23.0812	22.2759	22.2118	21.0533	20.7911	19.7789	18.5180	18.0216	17.1331
Boat	2	15.5351	15.4530	15.3968	15.3641	15.3214	14.4443	14.4431	14.4196	14.4068	14.3962
	3	18.4955	18.4811	18.4552	18.3813	18.3223	17.5732	17.4485	17.4264	17.3982	17.3645
	4	20.3357	20.2594	20.2513	20.2080	20.1776	19.4562	19.4267	19.4210	19.3310	19.3214

Table 7 (continued)

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
	5	22.2518	22.2243	22.1636	22.1388	22.0972	20.6001	20.5149	20.5107	20.4668	20.4587
Starfish	2	14.3792	14.2903	14.2637	14.2565	14.1610	14.5007	14.4578	14.3803	14.3122	14.2612
	3	17.5112	17.4824	17.4772	17.3559	17.3402	18.1791	18.1635	18.0924	18.0417	18.0193
	4	19.7771	19.7631	19.6659	19.6625	19.6220	18.8592	18.8112	18.7337	18.7290	18.7154
	5	20.3715	20.3611	20.3416	20.2695	20.1491	21.0976	21.0881	21.0405	21.0195	20.9399
Lake	2	15.6380	15.2513	14.7510	14.4022	14.1947	14.9177	14.6421	13.5409	13.3311	13.0086
	3	17.9170	18.3842	18.3272	17.8718	16.0296	16.9363	16.4671	16.0902	15.1847	14.7874
	4	19.5970	18.4905	18.2801	18.2444	18.1309	17.9838	16.9804	16.5232	16.2965	16.2712
	5	20.9204	20.8998	20.4411	20.4011	19.4256	19.7463	19.1502	19.1025	18.4541	18.3313
Living room	2	16.8739	15.9323	14.3932	14.1013	14.0385	15.9921	15.8947	15.4431	15.3067	13.6287
	3	19.3731	18.9031	18.8479	18.1476	17.4651	18.5031	18.2919	17.7130	17.3861	16.2105
	4	21.4711	20.0752	20.0715	19.9351	19.5822	20.8727	19.6402	19.5679	19.4203	19.4115
	5	23.9213	23.6248	22.6857	22.3532	21.5381	22.8377	22.2302	21.9605	21.7680	20.2113
Gold hill	2	13.1576	13.1297	13.1000	13.0791	13.0780	14.4217	14.3724	14.3029	14.2545	14.1844
	3	17.0292	17.0216	17.0062	17.0020	16.1278	16.7683	16.7264	16.7253	16.7142	16.5905
	4	18.2518	18.2249	18.1422	18.0920	18.0738	17.9380	17.8678	17.8315	17.7031	17.6741
	5	19.2874	19.2596	19.1789	19.1176	19.0815	20.1394	20.1293	20.0883	20.0077	19.9490

Table 8 Comparative analysis of SSIM values obtained by otsu's and kapur's methods

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	0.54627	0.54600	0.54582	0.54496	0.54490	0.52515	0.52511	0.52510	0.52496	0.52393
	3	0.63528	0.63479	0.63476	0.63474	0.63417	0.62440	0.62413	0.62378	0.62362	0.62320
	4	0.68012	0.67934	0.67923	0.67921	0.67860	0.68071	0.68001	0.67993	0.67979	0.67943
	5	0.70358	0.70327	0.70195	0.70154	0.70137	0.70944	0.70831	0.70824	0.70822	0.70812
Baboon	2	0.58770	0.58758	0.58755	0.58751	0.58686	0.62399	0.61517	0.61491	0.61483	0.61454
	3	0.68013	0.67895	0.67858	0.67855	0.67844	0.73572	0.73493	0.73488	0.73415	0.73386
	4	0.77228	0.77208	0.77176	0.77088	0.77057	0.78511	0.78490	0.78476	0.78371	0.78347
	5	0.81219	0.81211	0.81167	0.81161	0.81059	0.83460	0.83441	0.83346	0.83332	0.83278
Airplane	2	0.71756	0.71563	0.71260	0.70747	0.70559	0.69138	0.69119	0.69095	0.69001	0.68994
	3	0.74473	0.74247	0.73699	0.73644	0.73302	0.79622	0.79611	0.79582	0.79563	0.79471
	4	0.80296	0.80136	0.80066	0.79512	0.79299	0.81855	0.81834	0.81778	0.81777	0.81763
	5	0.81678	0.81135	0.80978	0.80763	0.80202	0.80949	0.80931	0.80915	0.80881	0.80854
Camera man	2	0.59656	0.59636	0.59577	0.59551	0.59541	0.54922	0.54660	0.54095	0.53734	0.53586
	3	0.63720	0.63681	0.63676	0.63623	0.63576	0.62440	0.62431	0.62385	0.61158	0.61083
	4	0.65206	0.65181	0.65157	0.65106	0.65083	0.69154	0.68630	0.68554	0.68094	0.67990
	5	0.68482	0.68429	0.68406	0.68366	0.68300	0.70622	0.70484	0.70290	0.69982	0.69798
Hunter	2	0.45229	0.45110	0.45108	0.45051	0.45045	0.35937	0.35929	0.35925	0.35897	0.35820
	3	0.57158	0.56158	0.56103	0.56083	0.56036	0.49925	0.49903	0.49844	0.49839	0.49818
	4	0.65361	0.64812	0.64756	0.64741	0.64728	0.57520	0.57409	0.57376	0.57374	0.57371
	5	0.66249	0.65189	0.65157	0.65106	0.65102	0.61036	0.60983	0.60980	0.60927	0.60876
Butterfly	2	0.64593	0.64388	0.64299	0.63910	0.63813	0.75691	0.75404	0.74935	0.74771	0.74598
	3	0.68812	0.68281	0.68199	0.67994	0.67062	0.87249	0.86872	0.86187	0.86014	0.85912
	4	0.71478	0.71357	0.71106	0.70519	0.70474	0.88989	0.88681	0.87841	0.87733	0.87710
	5	0.72344	0.72314	0.72037	0.71681	0.71382	0.92396	0.92212	0.91409	0.91199	0.91094
Pepper	2	0.61997	0.61362	0.61313	0.61379	0.61416	0.60084	0.60024	0.60013	0.59994	0.59963
	3	0.66056	0.66054	0.66023	0.65957	0.65933	0.67042	0.66959	0.66929	0.66870	0.66861
	4	0.71447	0.70353	0.70224	0.70216	0.70213	0.71937	0.71819	0.71813	0.71808	0.71803
	5	0.76789	0.76279	0.76214	0.76210	0.76129	0.71047	0.71022	0.70982	0.70955	0.70950
Boat	2	0.54630	0.54075	0.53679	0.53536	0.53246	0.51878	0.51834	0.51664	0.50973	0.50731
	3	0.65215	0.63964	0.63400	0.63321	0.63272	0.60718	0.60567	0.60443	0.59718	0.58829
	4	0.67302	0.66223	0.66153	0.66015	0.65917	0.66747	0.66603	0.66289	0.66110	0.66055

Table 8 (continued)

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
	5	0.80585	0.79872	0.79169	0.79052	0.78893	0.69410	0.69218	0.68725	0.68586	0.68202
Starfish	2	0.65288	0.64754	0.64597	0.64403	0.64296	0.63260	0.63235	0.63110	0.62178	0.61795
	3	0.69551	0.69223	0.69170	0.69039	0.68301	0.72334	0.72044	0.71887	0.71200	0.71004
	4	0.71598	0.71264	0.71081	0.70135	0.69872	0.77085	0.76764	0.76244	0.75870	0.75839
	5	0.73611	0.73330	0.73301	0.73071	0.72556	0.81586	0.81493	0.81121	0.80806	0.80473
Lake	2	0.53336	0.51909	0.51844	0.49371	0.47357	0.51510	0.51492	0.51349	0.50072	0.49894
	3	0.56676	0.53549	0.52067	0.51556	0.50963	0.51762	0.51573	0.50606	0.50503	0.50306
	4	0.62028	0.58833	0.58048	0.57336	0.57268	0.52276	0.51850	0.51727	0.51395	0.51286
	5	0.59840	0.58752	0.56229	0.54001	0.53975	0.53126	0.52834	0.51519	0.51256	0.51168
Living room	2	0.70459	0.69933	0.69752	0.69170	0.68784	0.49308	0.49296	0.49292	0.49286	0.49204
	3	0.70851	0.70849	0.70802	0.70128	0.70077	0.61038	0.61037	0.60980	0.60944	0.60904
	4	0.73130	0.72841	0.71301	0.71229	0.71196	0.64347	0.64299	0.64273	0.64241	0.64198
	5	0.74393	0.74310	0.73706	0.73187	0.72911	0.68697	0.68639	0.68595	0.68583	0.68562
Gold hill	2	0.29506	0.29395	0.28916	0.28106	0.28062	0.18948	0.18496	0.18107	0.17838	0.17619
	3	0.54967	0.54851	0.54467	0.54396	0.53224	0.30415	0.29940	0.29786	0.29500	0.29011
	4	0.55407	0.55104	0.54303	0.54157	0.54076	0.31735	0.31685	0.31178	0.30694	0.30445
	5	0.64172	0.63615	0.63378	0.62743	0.62565	0.36088	0.35894	0.35440	0.35438	0.34623

Table 9 Comparative analysis of FSIM values obtained by otsu's and kapur's methods

Test image	m	otsu's method					kapur's Entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	0.82044	0.81783	0.81678	0.81559	0.81345	0.74036	0.73956	0.73855	0.73548	0.73473
	3	0.77122	0.77078	0.77062	0.76919	0.76492	0.74754	0.74563	0.74325	0.74204	0.73997
	4	0.77635	0.77535	0.77448	0.77072	0.77037	0.81823	0.81741	0.81455	0.81419	0.81290
	5	0.84558	0.84388	0.84078	0.83881	0.83874	0.83769	0.83757	0.83457	0.83119	0.82920
Baboon	2	0.80314	0.80209	0.79746	0.79586	0.79420	0.79287	0.79154	0.79085	0.78992	0.78801
	3	0.83654	0.83582	0.83405	0.83356	0.83233	0.79833	0.79560	0.79551	0.79510	0.78917
	4	0.86251	0.86053	0.86036	0.86028	0.85723	0.83846	0.83607	0.83575	0.83504	0.83002
	5	0.87722	0.87708	0.87629	0.87281	0.87215	0.84774	0.84410	0.84367	0.84128	0.84059
Airplane	2	0.81977	0.81875	0.81308	0.81289	0.81216	0.79845	0.79745	0.79514	0.79313	0.78961
	3	0.83909	0.83811	0.83400	0.83200	0.83130	0.85191	0.85100	0.85057	0.84872	0.84658
	4	0.83975	0.83542	0.83430	0.83422	0.83291	0.85946	0.85846	0.85589	0.85517	0.85322
	5	0.89238	0.89094	0.89052	0.89006	0.88831	0.85901	0.85880	0.85791	0.85673	0.85458
Camera man	2	0.82721	0.82619	0.82600	0.82514	0.82267	0.82348	0.82210	0.82006	0.81995	0.81863
	3	0.84668	0.84505	0.84405	0.84271	0.83804	0.84568	0.84171	0.84115	0.84080	0.83852
	4	0.90601	0.90412	0.90190	0.90168	0.89887	0.89335	0.89246	0.88835	0.88750	0.88722
	5	0.86166	0.86013	0.85728	0.85676	0.85538	0.84851	0.84842	0.84499	0.84485	0.84268
Hunter	2	0.79065	0.78948	0.78845	0.78837	0.78415	0.81537	0.81493	0.81478	0.81202	0.81155
	3	0.86209	0.85654	0.85647	0.85620	0.85554	0.80812	0.80733	0.80387	0.80321	0.80065
	4	0.74059	0.73665	0.73659	0.73608	0.73217	0.80992	0.80919	0.80632	0.80459	0.80128
	5	0.79115	0.79033	0.78832	0.78792	0.78605	0.87768	0.87708	0.87679	0.87307	0.87275
Butterfly	2	0.71411	0.71347	0.70802	0.70798	0.70598	0.77710	0.77448	0.77293	0.77023	0.76998
	3	0.80473	0.80397	0.79975	0.79849	0.79557	0.71526	0.71493	0.71230	0.71086	0.70958
	4	0.88367	0.88193	0.88120	0.88014	0.87878	0.77960	0.77845	0.77673	0.77643	0.77055
	5	0.91861	0.91466	0.91360	0.91114	0.90939	0.86567	0.86437	0.86422	0.86328	0.85892
Pepper	2	0.81681	0.81593	0.81566	0.81526	0.80945	0.70862	0.70640	0.70611	0.70553	0.70050
	3	0.78662	0.78582	0.78493	0.78420	0.78032	0.76862	0.76743	0.76335	0.76015	0.75968
	4	0.84665	0.84625	0.84493	0.84465	0.83750	0.80218	0.80033	0.79925	0.79773	0.79261
	5	0.83189	0.82937	0.82887	0.82783	0.82717	0.82644	0.82214	0.82148	0.81933	0.81896
Boats	2	0.75914	0.75799	0.75711	0.75188	0.75098	0.75656	0.75640	0.75541	0.75236	0.75235
	3	0.79127	0.79067	0.79049	0.78828	0.78537	0.84009	0.83924	0.83889	0.83627	0.83549
	4	0.80355	0.80284	0.80243	0.79833	0.79718	0.87705	0.87377	0.86974	0.86860	0.86803

Table 9 (continued)

Test image	m	otsu's method					kapur's Entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
	5	0.87666	0.87650	0.87597	0.87217	0.87112	0.89848	0.89772	0.89338	0.89062	0.89018
Starfish	2	0.77688	0.77397	0.77209	0.76940	0.76875	0.61861	0.61473	0.61376	0.61208	0.60933
	3	0.77135	0.77042	0.76826	0.76757	0.76752	0.69571	0.69522	0.69453	0.69413	0.69365
	4	0.79446	0.79379	0.78948	0.78873	0.78710	0.74875	0.74830	0.74682	0.74496	0.74204
	5	0.80689	0.80563	0.80490	0.80349	0.80192	0.79847	0.79733	0.79519	0.79097	0.79055
Lake	2	0.80961	0.80625	0.80412	0.80360	0.80221	0.79485	0.79407	0.79346	0.79105	0.78741
	3	0.82249	0.82202	0.81819	0.81674	0.81609	0.83772	0.83423	0.83305	0.83062	0.82899
	4	0.86400	0.86280	0.86239	0.86010	0.85748	0.87490	0.87435	0.87309	0.87260	0.87245
	5	0.89101	0.89012	0.88720	0.88709	0.88551	0.90587	0.90310	0.90046	0.89988	0.89799
Living room	2	0.81518	0.81498	0.81351	0.80907	0.80593	0.72051	0.71938	0.71917	0.71816	0.71757
	3	0.77538	0.77526	0.77476	0.77166	0.77045	0.81737	0.81708	0.81644	0.81292	0.80967
	4	0.83832	0.83806	0.83527	0.83512	0.83478	0.84566	0.84314	0.84234	0.84142	0.84060
	5	0.69970	0.69768	0.69386	0.69359	0.69080	0.89809	0.89620	0.89562	0.89482	0.89352
Gold hill	2	0.75448	0.75319	0.75317	0.74696	0.74581	0.53791	0.53340	0.52978	0.52914	0.52900
	3	0.78887	0.78721	0.78511	0.78416	0.78309	0.54401	0.54156	0.53865	0.53799	0.53591
	4	0.73766	0.73522	0.73212	0.72994	0.72885	0.66382	0.66323	0.66273	0.65945	0.65874
	5	0.81936	0.81888	0.81865	0.81668	0.81590	0.73811	0.73577	0.73542	0.73396	0.73161

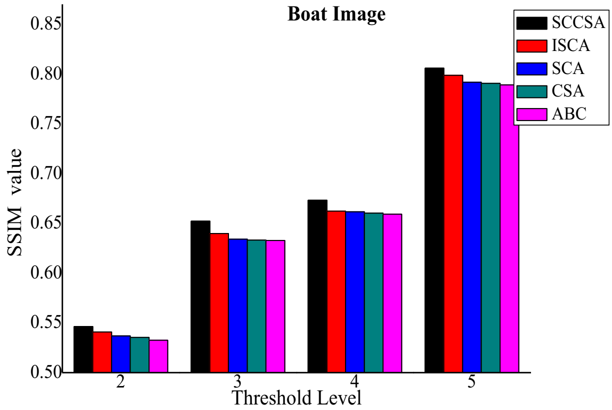
Table 10 Comparative analysis of CPU time values

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	3.88468	3.88926	3.90192	3.91583	3.92527	3.2377	3.2579	3.2587	3.2612	3.2690
	3	4.03131	4.07391	4.16927	4.26154	4.29439	3.6808	3.8136	3.8314	3.8821	3.8974
	4	4.86244	4.95354	5.02031	5.03373	5.47816	4.0833	4.4237	4.8069	5.0487	5.3830
	5	5.23309	5.48963	5.50097	5.55365	5.79959	5.0263	5.0998	5.4310	5.6663	5.8579
Baboon	2	3.35522	3.88532	3.91084	3.92332	3.92919	3.7889	3.7975	3.8002	3.8041	3.8451
	3	4.06252	4.10395	4.24422	4.24901	4.31248	3.5848	3.6976	3.7418	3.7427	3.9388
	4	5.45708	5.48490	5.65898	5.67820	6.02710	4.5403	5.0067	5.0297	5.4515	5.4749
	5	5.38682	5.51922	5.73740	5.76397	6.08500	5.1416	5.1452	5.2465	5.4064	5.4747
Airplane	2	3.56841	3.88536	3.89327	3.93129	3.93737	3.0805	3.2317	3.2605	3.2619	3.2760
	3	4.02795	4.03861	4.06677	4.16876	4.28114	3.4040	3.6877	3.7611	3.8737	3.9162
	4	5.00149	5.26576	5.63523	5.72507	5.92812	4.1408	4.5979	4.6584	5.0583	5.4398
	5	5.24005	5.50684	5.92374	6.02438	6.07948	5.0966	5.1732	5.3109	5.4571	5.7967
Camera man	2	3.22483	3.22879	3.22997	3.25530	3.26653	3.2283	3.2344	3.2501	3.2711	3.9367
	3	4.26611	4.27193	4.30107	4.35570	4.51252	3.6882	3.6996	3.7011	3.7342	3.9469
	4	5.04413	5.32569	5.58158	5.70500	5.95217	4.1339	4.4524	4.7131	4.7241	4.8746
	5	6.05852	6.16150	6.93601	7.03719	7.05301	4.8290	4.8650	5.2314	5.6779	5.8331
Hunter	2	3.56599	3.57764	3.58909	3.61393	3.61994	3.2218	3.2274	3.2349	3.2629	3.5584
	3	4.71327	4.85175	4.85923	4.88466	4.92808	3.0989	3.7007	3.7315	3.7806	3.9220
	4	4.93708	4.99273	5.06136	5.18548	5.64201	4.5246	4.8196	4.8277	5.1370	5.3581
	5	5.56994	5.94397	6.21173	6.23578	6.36902	5.1532	5.3543	5.4536	5.5005	5.5627
Butterfly	2	3.01190	3.03018	3.03096	3.04415	3.04865	3.8969	3.8970	3.9020	3.9344	3.9346
	3	4.21791	4.22000	4.22694	4.41035	4.45854	3.1434	3.1620	3.1622	3.2403	3.4070
	4	4.83144	5.22748	5.77763	5.99816	6.23701	5.4273	5.4362	5.4372	5.7131	5.7819
	5	5.80706	6.04830	6.28818	6.30587	6.31969	5.0036	5.3991	5.6349	5.6603	5.7606
Pepper	2	3.76724	3.76837	3.80766	3.81691	3.82281	3.0020	3.0058	3.0245	3.0403	3.0522
	3	4.50927	4.58416	4.59595	4.64538	4.65178	3.3742	3.4418	3.4541	3.5295	3.6315
	4	4.93687	5.29254	5.39648	5.88040	5.90884	4.7559	5.1678	5.4389	5.8334	6.0897
	5	5.99574	6.03386	6.34378	6.35056	6.60893	4.8175	4.9429	5.0977	5.7204	5.7503
Boat	2	3.70026	3.70619	3.71900	3.72452	3.72746	3.1130	3.1293	3.1299	3.1317	3.1328
	3	4.88207	4.90601	4.95090	5.01687	5.04609	3.4749	3.6022	3.6632	3.6870	3.7118
	4	5.19341	5.42969	5.64355	5.76946	6.10003	4.3294	5.1246	5.2144	5.4229	5.4509

Table 10 (continued)

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
	5	5.49593	5.58840	5.61906	6.19261	6.20686	4.6326	4.6854	4.7105	4.7224	5.0487
Starfish	2	3.26242	3.26493	3.27390	3.28940	3.29121	3.0281	3.0299	3.0409	3.0701	3.0765
	3	4.06834	4.13202	4.15856	4.21307	4.24648	3.2613	3.4405	3.4926	3.5118	3.5283
	4	4.52410	4.92455	4.93001	5.16207	5.35671	4.6994	4.7624	4.8772	4.9760	5.1024
	5	5.79231	5.90482	6.12002	6.47202	6.54152	4.8615	5.2636	5.3321	5.6188	5.7532
Lake	2	3.43174	3.43182	3.43835	3.44053	3.45324	3.0987	3.1217	3.1240	3.1339	3.1343
	3	4.54202	4.57458	4.57779	4.74149	4.76610	3.5872	3.6035	3.6139	3.8029	3.8634
	4	4.48429	4.51703	4.67299	5.18472	5.44057	4.7105	4.9851	5.3559	5.9117	6.0499
	5	5.66761	5.90395	6.22495	6.28255	6.30926	4.7285	4.8826	4.9614	5.0008	5.4069
Living room	2	3.55164	3.56651	3.59646	3.60119	3.60455	3.1294	3.1381	3.1426	3.1447	3.1618
	3	4.20652	4.29205	4.38918	4.40935	4.43837	3.6738	3.7061	3.7240	3.7711	3.8232
	4	4.31596	4.38085	4.88019	5.06464	5.64287	4.9104	5.3001	5.3664	5.3962	6.0357
	5	5.71209	5.82153	5.91490	6.11022	6.13687	5.2154	5.2910	5.5159	5.8524	5.9444
Gold hill	2	3.23727	3.28171	3.28342	3.28902	3.29125	3.3315	3.3673	3.3719	3.3726	3.3728
	3	4.23327	4.33893	4.36654	4.46446	4.49094	3.5788	3.6895	3.7352	3.7874	3.8192
	4	4.76976	4.96255	5.32166	5.58101	5.76544	5.5254	5.6051	5.7398	5.9152	6.0024
	5	5.74533	5.76118	5.85369	6.04653	6.32998	5.2885	5.3972	5.4006	5.6564	5.6797

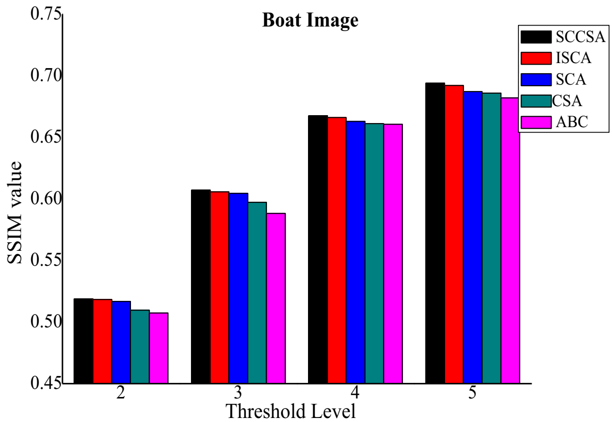
Fig. 8 Comparison of SSIM values for different algorithms with SCCSA using Otsu's method



concluded that SCCSA apparently produced lower STD values more frequently with rest of other algorithms. The supported results are indicate that the SCCSA algorithm's stability is better achieved.

Furthermore, a statistical based wilcoxon test is performed to compare the meaningful difference between the performances of five algorithms at a 5% significance level. Subsequently, the objective function values (both methods) of the SCCSA algorithm is compared with the other four algorithms, such as ICSA, SCA, CSA and ABC. The values of the two target functions are not considered to be significantly different in a null hypothesis. The alternative hypothesis takes into account both values and differences between the two methods. Typically adequate justification against the null hypothesis may be considered if p values are less than 0.05. The demonstrated test results of the wilcoxon test with p value for evaluation, and from Table 13, it is evident that SCCSA outperforms other algorithms in almost all test cases because of the p value <0.05, which proves to be more statistically significant. In addition, there is a considerable difference between SCCSA and the other four algorithms is observed. As a whole, the tabular findings of quantitative results and figures with

Fig. 9 Comparison of SSIM values for different algorithms with SCCSA using Kapur's entropy



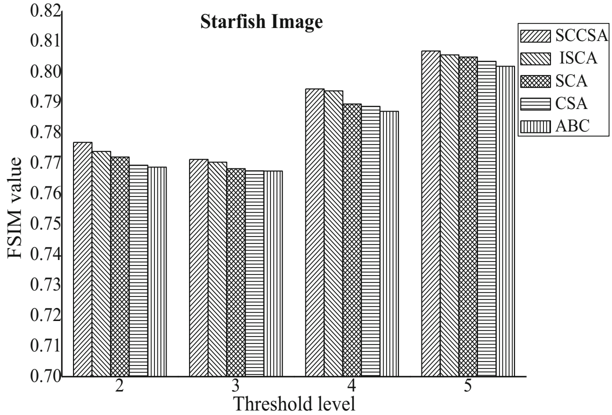


Fig. 10 Comparison of FSIM values for different algorithms with SCCSA using Otsu's method

proposed multilevel thresholding based SCCSA algorithm outperformed as compared with ICSA, SCA, CSA and ABC algorithms.

8 Conclusion and future work

In this work, we introduced a new hybrid sine cosine crow search algorithm for multilevel image thresholding using the objective functions of otsu's and kapur's. A standard 12 set of gray images were used for testing of proposed algorithm. The efficacy of the SCCSA algorithm was evaluated by testing and comparing algorithms (ICSA, SCA, CSA and ABC) for best threshold values, PSNR, SSIM, FSIM, and CPU time. Results show that the hybridized SCCSA algorithm can achieve very obvious superiority and competitive achievements in comparison to other algorithms in either method particularly in terms of qualitative and quantitative aspects. Further, the wilcoxon test results between SCCSA and other algorithms show that the difference with other algorithms is significant. Finally, the promising

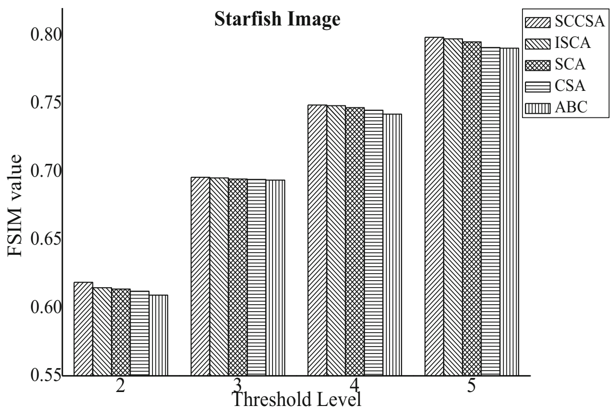


Fig. 11 Comparison of FSIM values for different algorithms with SCCSA using Kapur's entropy

Table 11 Comparative analysis of mean values

Test image	*m*	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	1962.457	1961.308	1960.506	1958.277	1957.856	12.3545	11.1255	10.0985	9.8950	9.4831
	3	2163.355	2161.799	2161.577	2160.548	2159.125	16.8950	15.4708	14.7777	12.8342	12.4433
	4	2221.658	2220.204	2219.181	2218.188	2218.034	18.6749	18.2809	17.8408	16.8143	15.7356
	5	2230.755	2229.745	2228.308	2227.572	2225.479	21.0403	20.4349	20.2092	19.1108	17.9693
Baboon	2	1555.789	1554.065	1553.936	1552.109	1550.106	13.2178	12.6812	13.3327	12.2160	11.7472
	3	1647.087	1646.654	1645.349	1643.444	1642.752	15.2516	15.3424	14.8262	11.9895	10.1814
	4	1701.215	1700.011	1699.221	1698.655	1696.105	18.2166	18.4697	17.6801	16.9598	14.3746
	5	1728.666	1729.476	1727.713	1726.576	1724.733	22.0549	21.4415	20.6678	19.2871	18.7650
Airplane	2	1930.708	1928.150	1927.252	1926.360	1925.111	12.1115	10.9113	9.1679	8.4229	7.6810
	3	2007.516	2006.376	2005.152	2004.324	2004.012	15.0871	13.6352	13.5582	11.9894	11.0806
	4	2054.125	2053.156	2051.533	2050.054	2049.500	18.2297	16.9383	16.3317	14.3659	13.8844
	5	2079.968	2078.480	2076.850	2074.277	2074.244	20.6999	19.6372	17.9557	17.5122	15.0546
Camera man	2	3651.956	3650.501	3649.151	3648.507	3647.225	12.4897	11.5885	9.8363	8.1248	7.7641
	3	3727.837	3726.754	3725.751	3723.061	3722.567	15.9435	14.3404	12.9277	12.1350	10.7837
	4	3782.428	3780.021	3779.570	3778.228	3777.169	20.7031	19.7325	19.1547	18.7402	17.7297
	5	3813.761	3812.790	3811.333	3809.926	3808.373	23.1286	22.1956	21.5206	19.9809	19.2594
Hunter	2	3064.211	3063.907	3061.301	3059.247	3059.105	17.5449	16.5248	15.7882	14.1275	12.6953
	3	3213.447	3111.692	3110.509	3109.312	3108.591	21.9171	19.9631	19.5895	18.6061	17.3569
	4	3269.484	3268.016	3267.350	3266.619	3265.960	26.1601	24.8204	24.2293	23.1766	21.9031
	5	3308.109	3307.557	3305.998	3305.661	3304.221	30.0487	29.2454	28.6091	27.4044	26.3981
Butterfly	2	1554.887	1554.613	1553.643	1553.510	1553.316	12.7468	12.3596	12.1396	11.5742	11.1214
	3	1671.817	1671.267	1671.131	1670.937	1670.410	16.6106	16.4732	15.5280	15.4994	14.8592
	4	1713.541	1713.276	1712.779	1712.547	1712.235	19.1459	19.0806	18.8095	18.5171	17.6584
	5	1738.411	1738.301	1737.912	1737.616	1737.554	21.5608	21.1688	20.8856	20.2422	20.2373
Pepper	2	2513.916	2512.796	2511.442	2509.175	2508.018	12.9346	10.0977	9.9632	8.2869	8.1246
	3	2683.371	2682.634	2681.032	2678.670	2680.486	15.8843	14.3791	12.6273	12.0872	9.9711
	4	2740.538	2739.300	2738.509	2737.262	2736.898	18.4418	17.2364	16.6927	15.3676	15.2128
	5	2787.609	2786.374	2785.209	2783.973	2782.352	20.8641	18.6515	17.7124	16.8380	16.3578
Boat	2	1865.266	1864.874	1864.683	1864.256	1863.364	19.0404	18.4276	18.0670	17.9791	17.8658
	3	1997.980	1997.414	1996.803	1996.706	1995.677	23.2142	23.0081	22.8660	21.9284	21.9272
	4	2060.226	2059.809	2059.231	2058.640	2058.264	28.0548	27.6880	27.3180	27.3023	27.2283

Table 11 (continued)

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
	5	2093.781	2092.648	2092.318	2091.450	2091.022	30.6144	30.4948	30.0833	29.8040	29.7648
Starfish	2	2492.857	2492.698	2492.235	2491.225	2491.140	19.3936	18.7725	18.0498	17.6171	17.5424
	3	2718.116	2717.950	2717.624	2717.355	2717.300	23.7674	23.3090	23.0983	23.0801	22.8898
	4	2803.563	2803.215	2802.669	2801.919	2801.885	28.4857	27.9613	27.8989	27.7858	27.6316
	5	2849.412	2849.241	2848.401	2846.896	2846.806	31.8739	31.7454	30.5759	30.4474	30.4254
Lake	2	3970.961	3970.854	3970.592	3969.512	3969.430	17.4339	17.0336	16.8301	16.6927	16.5120
	3	3981.235	3980.893	3980.677	3980.491	3979.706	22.8559	22.5950	22.1450	22.1190	21.6369
	4	4097.209	4096.482	4095.564	4094.189	4092.137	27.3649	27.1357	26.8678	26.3827	26.0042
	5	4123.820	4122.105	4120.563	4119.306	4118.188	31.0819	30.7573	30.1182	29.8467	29.5937
Living room	2	1630.735	1630.029	1629.275	1628.353	1628.005	12.8070	12.5846	12.2749	11.7515	11.4573
	3	1748.515	1747.755	1747.320	1746.826	1746.112	15.8631	15.8031	15.5993	15.2844	14.8626
	4	1861.078	1861.010	1860.012	1859.572	1858.992	18.7983	18.7397	18.0511	17.4161	17.3236
	5	1916.234	1916.059	1915.486	1915.386	1915.370	21.7673	20.4114	20.3986	20.1415	19.9803
Gold hill	2	1564.535	1564.393	1564.336	1563.961	1563.873	20.7311	19.2286	19.0517	18.8932	17.9399
	3	1608.738	1608.315	1608.289	1607.253	1607.174	24.3793	24.0820	23.6672	22.3635	22.2947
	4	1694.244	1694.231	1693.378	1693.320	1693.271	28.2968	28.0719	27.7030	26.9537	26.9052
	5	1712.795	1711.814	1711.518	1710.485	1710.099	31.3361	30.7654	30.3447	30.0843	29.3816

Table 12 Comparative analysis of STD values

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
Lena	2	0.00E+00	0.00E+00	0.00E+00	8.29E-04	1.12E-03	3.13E-11	3.85E-04	5.22E-04	6.79E-04	2.40E-03
	3	8.37E-08	8.46E-04	1.15E-03	1.16E-03	1.66E-03	7.74E-13	1.25E-03	1.76E-03	2.88E-03	4.08E-03
	4	2.75E-04	4.95E-04	1.11E-03	1.22E-03	1.43E-03	1.14E-04	5.40E-04	1.10E-03	1.17E-03	8.72E-03
	5	8.10E-04	1.14E-03	1.19E-03	1.48E-03	1.54E-03	2.06E-04	7.99E-04	2.31E-03	3.29E-03	5.54E-03
Baboon	2	0.00E+00	0.00E+00	0.00E+00	1.24E-03	1.28E-03	1.43E-09	3.20E-04	3.71E-04	5.31E-04	1.26E-03
	3	0.00E+00	7.18E-04	1.26E-03	1.88E-07	1.99E-03	2.33E-12	4.72E-10	6.87E-04	1.43E-03	2.80E-03
	4	2.65E-04	0.00E+00	7.16E-04	9.81E-04	1.14E-03	2.61E-04	3.80E-04	8.72E-04	1.02E-03	1.40E-03
	5	7.94E-04	1.52E-03	1.94E-03	5.15E-03	6.13E-03	9.31E-04	3.16E-03	3.47E-03	4.01E-03	5.10E-03
Airplane	2	0.00E+00	0.00E+00	0.00E+00	4.21E-04	5.86E-04	0.00E+00	8.54E-04	9.46E-04	1.22E-03	1.79E-03
	3	2.69E-04	3.37E-04	9.75E-04	1.32E-03	1.66E-03	6.41E-14	2.94E-03	2.95E-03	2.99E-03	6.69E-03
	4	9.08E-05	1.06E-04	2.63E-04	6.83E-04	1.00E-03	6.05E-04	6.99E-04	7.19E-04	8.85E-04	8.23E-03
	5	7.34E-04	4.51E-03	5.15E-03	5.73E-03	6.38E-03	1.24E-04	3.07E-04	2.25E-03	2.26E-03	3.73E-03
Camera man	2	0.00E+00	0.00E+00	0.00E+00	1.22E-03	1.32E-03	3.96E-15	9.36E-04	9.37E-04	1.10E-03	1.33E-03
	3	1.27E-03	3.11E-03	3.58E-03	4.06E-03	5.72E-03	6.30E-15	2.03E-03	3.29E-03	4.03E-03	4.10E-03
	4	2.04E-04	2.72E-04	6.45E-04	7.83E-04	6.78E-03	1.89E-04	4.41E-04	5.54E-04	6.13E-04	6.58E-03
	5	3.95E-04	6.97E-04	8.80E-04	3.38E-03	5.90E-03	4.67E-04	1.29E-03	1.85E-03	4.37E-03	5.40E-03
Hunter	2	0.00E+00	0.00E+00	0.00E+00	9.40E-04	1.33E-03	1.11E-13	9.11E-04	2.39E-04	8.32E-05	3.68E-04
	3	0.00E+00	4.88E-04	2.88E-03	4.02E-03	4.35E-03	3.15E-15	3.32E-03	4.13E-03	2.28E-03	1.61E-03
	4	2.35E-04	3.39E-04	7.81E-04	8.42E-04	5.58E-03	7.08E-04	4.46E-04	8.07E-04	9.42E-05	7.55E-04
	5	2.26E-04	7.47E-04	3.43E-03	3.60E-03	4.86E-03	4.41E-04	9.81E-04	2.40E-03	2.81E-03	5.82E-03
Butterfly	2	0.00E+00	0.00E+00	0.00E+00	1.34E-03	1.37E-03	4.63E-13	1.05E-03	1.13E-03	1.32E-03	1.94E-03
	3	3.15E-04	4.37E-04	1.51E-03	1.79E-03	5.59E-03	5.72E-04	2.11E-12	3.43E-03	4.16E-03	6.18E-03
	4	4.57E-04	7.93E-04	1.06E-03	1.21E-03	1.81E-03	2.66E-04	4.48E-04	5.18E-04	6.80E-04	7.37E-04
	5	5.69E-04	1.58E-03	1.79E-03	3.71E-03	4.97E-03	5.77E-05	1.12E-03	3.39E-03	6.06E-03	6.47E-03
Pepper	2	0.00E+00	0.00E+00	0.00E+00	1.14E-03	1.26E-03	5.25E-09	5.29E-04	8.93E-04	9.00E-04	1.29E-03
	3	7.94E-10	8.05E-04	1.06E-03	1.58E-03	2.29E-03	2.24E-14	5.34E-04	1.29E-03	2.45E-03	3.89E-03
	4	5.43E-04	5.82E-04	6.08E-04	1.16E-03	2.57E-03	1.92E-13	5.30E-04	5.45E-04	5.53E-04	8.73E-04
	5	7.06E-04	1.43E-03	3.98E-03	5.47E-03	6.00E-03	3.01E-08	2.40E-03	3.03E-03	4.24E-03	5.02E-03
Boat	2	3.44E-04	6.80E-04	8.95E-04	9.77E-04	1.37E-03	3.85E-12	4.22E-04	9.82E-04	1.28E-03	1.62E-03
	3	1.28E-04	8.12E-04	1.53E-03	2.12E-03	2.31E-03	1.83E-04	3.45E-04	4.68E-04	2.35E-03	2.88E-03
	4	4.31E-04	4.93E-09	1.22E-03	1.27E-03	3.47E-03	4.64E-04	6.87E-04	1.14E-03	1.20E-03	1.81E-03

Table 12 (continued)

Test image	m	otsu's method					kapur's entropy				
		SCCSA	ICSA	SCA	CSA	ABC	SCCSA	ICSA	SCA	CSA	ABC
	5	1.57E-04	2.56E-03	3.21E-03	3.75E-03	6.41E-03	1.27E-03	1.43E-03	2.01E-03	2.33E-03	4.28E-03
Starfish	2	0.00E+00	0.00E+00	0.00E+00	1.10E-03	2.29E-03	1.60E-14	6.01E-04	1.07E-03	1.31E-03	2.36E-03
	3	9.46E-05	1.55E-04	2.96E-03	3.64E-03	3.74E-03	1.16E-04	4.84E-04	2.37E-03	3.03E-03	5.13E-03
	4	2.91E-04	8.18E-04	9.61E-04	9.88E-04	6.02E-03	1.65E-04	6.55E-04	7.38E-04	1.11E-03	1.14E-03
	5	8.87E-05	1.30E-03	2.25E-03	3.36E-03	4.42E-03	4.98E-05	6.24E-04	7.67E-04	1.33E-03	5.28E-03
Lake	2	0.00E+00	0.00E+00	0.00E+00	1.09E-03	1.25E-03	1.94E-09	6.57E-14	9.65E-04	9.97E-04	1.28E-03
	3	4.31E-04	1.91E-03	2.47E-03	2.79E-03	4.23E-03	3.51E-12	6.01E-04	8.89E-04	9.22E-04	6.95E-03
	4	1.57E-04	2.02E-04	6.71E-04	1.14E-03	7.92E-03	1.47E-04	2.47E-04	3.96E-04	1.16E-03	2.90E-03
	5	5.42E-04	8.47E-04	4.06E-03	4.37E-03	4.86E-03	1.04E-03	1.69E-03	2.10E-03	3.62E-03	4.39E-03
Living room	2	0.00E+00	0.00E+00	0.00E+00	7.35E-04	1.57E-03	2.93E-14	6.97E-04	9.90E-04	1.25E-03	2.48E-03
	3	5.76E-04	1.90E-03	2.59E-03	2.98E-03	4.52E-03	9.20E-04	1.60E-03	2.12E-03	3.24E-03	4.17E-03
	4	1.62E-04	4.46E-04	8.52E-04	1.24E-03	2.54E-03	2.81E-04	9.59E-04	9.88E-04	1.04E-03	7.82E-03
	5	5.73E-04	8.72E-04	2.06E-03	2.48E-03	5.77E-03	4.79E-04	9.98E-04	5.27E-03	6.22E-03	6.42E-03
Gold hill	2	2.48E-12	0.00E+00	5.45E-04	1.05E-03	1.77E-03	4.46E-11	6.51E-04	7.80E-04	1.16E-03	1.27E-03
	3	5.36E-09	9.66E-04	2.29E-03	3.69E-03	4.20E-03	1.58E-15	2.93E-03	3.80E-03	3.80E-03	3.98E-03
	4	4.23E-04	4.68E-04	6.07E-04	9.59E-04	6.10E-03	1.90E-04	5.39E-04	6.47E-04	7.21E-04	5.62E-03
	5	5.75E-04	6.96E-04	5.28E-03	5.51E-03	5.70E-03	2.02E-04	7.90E-04	9.59E-04	1.17E-03	4.52E-03

Table 13 *p*-values produced by Wilcoxon test comparing SCCSA vs. other algorithms based on objective function values

Test image	m	otsu's method				kapur's entropy			
		SCCSA vs. ICSA	SCCSA vs. SCA	SCCSA vs. CSA	SCCSA vs. ABC	SCCSA vs. ICSA	SCCSA vs. SCA	SCCSA vs. CSA	SCCSA vs. ABC
Lena	2	1.925E-03	1.629E-09	2.656E-08	3.358E-08	2.722E-09	2.264E-10	5.464E-07	2.723E-07
	3	3.130E-05	5.290E-08	2.964E-11	3.051E-11	3.719E-01	3.484E-11	2.767E-10	4.309E-06
	4	4.525E-08	3.725E-01	2.791E-10	6.940E-11	7.135E-05	2.318E-09	4.256E-02	4.691E-06
	5	2.306E-10	2.670E-05	2.827E-11	2.631E-11	6.097E-05	2.545E-11	1.151E-10	5.676E-01
Baboon	2	5.890E-04	7.414E-11	8.195E-10	6.304E-11	2.680E-07	2.362E-08	7.976E-08	4.978E-10
	3	6.969E-08	5.474E-07	5.518E-01	1.995E-04	5.632E-05	3.402E-01	5.039E-09	2.082E-09
	4	2.278E-08	1.397E-04	4.980E-01	6.904E-08	1.145E-06	1.612E-11	1.157E-10	1.692E-08
	5	7.531E-05	3.386E-02	2.704E-05	7.477E-05	1.020E-05	2.663E-08	2.618E-10	2.201E-11
Airplane	2	2.627E-09	6.794E-08	2.562E-08	5.783E-08	9.965E-10	1.395E-08	1.118E-05	5.428E-05
	3	6.290E-05	2.421E-04	2.704E-04	2.635E-11	3.536E-08	5.249E-10	2.152E-08	1.106E-08
	4	7.968E-05	3.604E-02	3.559E-10	3.459E-10	1.482E-08	1.453E-08	5.641E-01	2.385E-10
	5	7.164E-06	2.897E-11	2.531E-09	2.558E-03	6.051E-11	1.109E-08	8.997E-10	7.592E-11
Camera man	2	2.232E-11	2.112E-07	2.017E-10	2.363E-01	4.136E-11	4.754E-08	3.254E-02	8.140E-11
	3	1.288E-08	5.067E-11	7.932E-09	8.072E-10	2.984E-11	2.454E-11	5.564E-10	7.390E-08
	4	1.006E-08	1.891E-01	1.703E-10	2.028E-05	3.893E-09	8.577E-10	2.479E-04	6.089E-04
	5	2.059E-05	2.330E-08	2.137E-04	2.086E-11	2.754E-10	2.723E-01	2.245E-04	4.767E-05
Hunter	2	4.029E-11	1.163E-11	9.666E-11	8.912E-04	4.479E-09	2.356E-08	5.755E-08	1.941E-11
	3	3.095E-01	2.175E-08	1.711E-08	1.118E-05	2.731E-02	2.701E-05	2.884E-07	2.334E-01
	4	6.167E-01	1.139E-11	6.615E-09	1.946E-04	2.931E-04	8.464E-11	2.583E-10	1.506E-08
	5	2.512E-05	2.817E-04	2.101E-07	1.571E-08	4.256E-01	1.247E-11	2.358E-04	2.444E-10
Butterfly	2	1.118E-05	8.912E-04	8.915E-10	3.340E-01	7.064E-04	7.986E-02	8.556E-11	6.032E-08
	3	5.658E-07	5.957E-09	1.504E-08	7.970E-10	7.630E-07	4.354E-04	3.862E-05	4.409E-08
	4	3.397E-08	1.688E-08	2.802E-08	1.840E-08	6.465E-10	7.036E-04	1.600E-02	6.176E-06
	5	7.584E-11	7.613E-11	5.663E-07	7.579E-11	3.288E-05	7.085E-04	1.896E-05	4.912E-05
Pepper	2	2.811E-08	6.810E-08	4.153E-11	5.993E-11	9.618E-09	3.351E-08	1.237E-08	1.672E-08
	3	3.041E-11	2.921E-11	2.407E-01	5.826E-08	5.246E-05	7.216E-03	2.817E-08	1.998E-06
	4	2.234E-10	1.407E-10	5.302E-13	2.185E-08	1.224E-06	4.291E-06	2.185E-02	3.387E-07
	5	2.770E-11	2.826E-11	2.311E-11	2.726E-05	2.390E-11	2.465E-11	2.909E-11	2.125E-11
Boat	2	6.895E-01	9.496E-11	4.602E-10	9.017E-10	2.145E-08	8.337E-11	7.831E-11	7.902E-11
	3	6.895E-05	9.587E-05	1.223E-05	1.010E-08	6.047E-04	5.988E-01	3.714E-08	7.842E-10
	4	1.531E-11	6.484E-07	1.576E-01	6.325E-09	1.379E-08	2.082E-08	6.465E-11	7.463E-08

Table 13 (continued)

Test image	*m*	otsu's method				kapur's entropy			
		SCCSA vs. ICSA	SCCSA vs. SCA	SCCSA vs. CSA	SCCSA vs. ABC	SCCSA vs. ICSA	SCCSA vs. SCA	SCCSA vs. CSA	SCCSA vs. ABC
	5	2.635E-03	5.845E-11	7.591E-11	7.587E-11	2.638E-07	4.979E-10	3.080E-05	1.564E-06
Starfish	2	2.548E-06	4.386E-10	4.556E-06	3.205E-08	4.618E-09	3.291E-08	5.737E-10	4.556E-08
	3	1.118E-11	1.223E-05	9.715E-11	7.598E-11	6.556E-01	5.372E-09	2.539E-05	5.458E-04
	4	9.588E-05	2.877E-08	7.855E-07	6.269E-07	2.625E-04	1.160E-10	3.559E-06	1.827E-10
	5	7.263E-10	9.587E-05	1.224E-05	4.510E-09	4.578E-11	8.245E-05	9.781E-11	2.705E-04
Lake	2	8.271E-11	2.180E-11	5.636E-05	1.118E-05	6.951E-11	5.998E-11	7.248E-11	8.374E-11
	3	4.851E-01	1.938E-10	2.959E-11	7.927E-10	1.880E-08	6.140E-09	3.155E-08	1.206E-09
	4	2.802E-08	7.808E-04	3.480E-01	8.627E-09	5.762E-08	7.489E-11	1.077E-01	7.639E-10
	5	7.579E-05	3.019E-08	5.956E-08	7.600E-11	7.211E-06	4.243E-05	3.424E-07	1.806E-05
Living room	2	7.064E-11	1.103E-05	1.449E-11	6.025E-07	3.814E-08	3.103E-07	8.509E-11	2.375E-08
	3	3.541E-08	3.145E-11	6.060E-09	2.547E-01	2.454E-10	2.466E-11	3.764E-05	2.717E-11
	4	6.343E-08	3.743E-08	1.780E-08	1.224E-05	2.777E-10	7.545E-11	2.572E-08	3.231E-09
	5	1.226E-05	2.219E-01	7.611E-11	1.137E-11	2.727E-01	2.699E-10	7.441E-05	2.454E-11
Gold hill	2	3.844E-08	5.964E-04	7.459E-08	4.855E-08	8.438E-10	7.755E-11	1.146E-11	4.424E-08
	3	2.379E-11	5.247E-09	6.033E-09	4.129E-10	7.740E-05	5.291E-09	2.512E-08	3.146E-11
	4	5.871E-10	4.605E-12	9.200E-12	6.337E-07	1.737E-08	1.685E-10	3.086E-08	5.080E-01
	5	2.225E-11	8.788E-10	4.578E-01	2.924E-11	1.163E-05	5.075E-05	4.999E-05	2.464E-11

results showed that the proposed SCCSA hybrid algorithm is more feasible and easily implementable for gray image segmentation effectively. Although the findings are indicative of a good result with the proposed SCCSA algorithm, the aim of the analysis is not to generate a proposed algorithm that would be able to overcome all algorithms with the currently available algorithms, but to show that for this reason the SCCSA can be seen as an attracting alternative. Future work involves incorporating with other popular entropy-based segmentation methods such as Fuzzy entropy, Tsallis entropy, Minimal cross-entropy, Renyi entropy, Masi entropy and Shannon entropy for medical and satellite images.

Declarations

Conflict of interest The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Abualigah L, Al-Okbi NK, Elaziz MA, Houssein EH (2022) Boosting marine predators algorithm by Salp swarm algorithm for multilevel thresholding image segmentation. Multimed tools Appl 1–36. https://doi.org/10.1007/s11042-022-12001-3
2. Abualigah L, Ewees AA, Al-qaness MAA et al (2022) Boosting arithmetic optimization algorithm by sine cosine algorithm and levy flight distribution for solving engineering optimization problems. Neural Comput Applic 34:8823–8852. https://doi.org/10.1007/s00521-022-06906-1
3. Agrawal S, Panda R, Bhuyan S, Panigrahi BK (2013) Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm. Swarm Evol Comput 11:16–30. https://doi.org/10.1016/j.swevo.2013.02.001
4. Agrawal V, Rastogi R, Tiwari DC (2018) Spider monkey optimization: a survey. Int J Syst Assur Eng Manag 9:929–941. https://doi.org/10.1007/s13198-017-0685-6
5. Ahmadi M, Kazemi K, Aarabi A, Niknam T, Helfroush MS (2019) Image segmentation using multilevel thresholding based on modified bird mating optimization. Multimed Tools Appl 78:23003–23027. https://doi.org/10.1007/s11042-019-7515-6
6. Alwerfali HSN, Abd Elaziz M, Al-Qaness MAA et al (2019) A multilevel image thresholding based on hybrid salp swarm algorithm and fuzzy entropy. IEEE Access 7:181405–181422. https://doi.org/10.1109/access.2019.2959325
7. Askarzadeh A (2016) A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput Struct 169:1–12. https://doi.org/10.1016/j.compstruc.2016.03.001
8. Aziz MAE, Ewees AA, Hassanien AE (2017) Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. Expert Syst Appl 83:242–256. https://doi.org/10.1016/j.eswa.2017.04.023
9. Baby Resma KP, Nair MS (2021) Multilevel thresholding for image segmentation using krill herd optimization algorithm. J King Saud Univ - Comput Inf Sci 33:528–541. https://doi.org/10.1016/j.jksuci.2018.04.007
10. Bhandari AK, Kumar A, Singh GK (2015) Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using Kapur's, Otsu and Tsallis functions. Expert Syst Appl 42:1573–1601. https://doi.org/10.1016/j.eswa.2014.09.049
11. Bhargava A, Bansal A (2018) Fruits and vegetables quality evaluation using computer vision: a review. J King Saud Univ - Comput Inf Sci https://doi.org/10.1016/j.jksuci.2018.06.002
12. Chakraborty F, Roy PK, Nandi D (2019) Oppositional elephant herding optimization with dynamic Cauchy mutation for multilevel image thresholding. Evol Intell 12:445–467. https://doi.org/10.1007/s12065-019-00238-1
13. Chen X, Huang H, Heidari AA, Sun C, Lv Y, Gui W, Liang G, Gu Z, Chen H, Li C, Chen P (2022) An efficient multilevel thresholding image segmentation method based on the slime mould algorithm with bee foraging mechanism: a real case with lupus nephritis images. Comput Biol Med 142:105179. https://doi.org/10.1016/j.compbiomed.2021.105179

14. Cuevas E, Sención F, Zaldivar D, Pérez-Cisneros M, Sossa H (2012) A multi-threshold segmentation approach based on artificial bee Colony optimization. Appl Intell 37:321–336. https://doi.org/10.1007/s10489-011-0330-z
15. Das S, Nayak GK, Saba L et al (2022) An artificial intelligence framework and its bias for brain tumor segmentation: a narrative review. Comput Biol Med 143:105273. https://doi.org/10.1016/j.compbiomed.2022.105273
16. Dehshibi MM, Sourizaei M, Fazlali M, Talaee O, Samadyar H, Shanbehzadeh J (2017) A hybrid bio-inspired learning algorithm for image segmentation using multilevel thresholding. Multimed Tools Appl 76:15951–15986. https://doi.org/10.1007/s11042-016-3891-3
17. Du E, Ives R, van Nevel A, She J-H (2011) advanced image processing for defense and security applications. EURASIP J Adv Signal Process 2010: https://doi.org/10.1155/2010/432972
18. Duan L, Yang S, Zhang D (2021) Multilevel thresholding using an improved cuckoo search algorithm for image segmentation. J Supercomput 77:6734–6753. https://doi.org/10.1007/s11227-020-03566-7
19. El Aziz MA, Ewees AA, Hassanien AE (2016) Hybrid swarms optimization based image segmentation. In: Hybrid soft computing for image segmentation. Springer International Publishing, Cham, pp 1–21
20. Ewees AA, Abd Elaziz M, Oliva D (2018) Image segmentation via multilevel thresholding using hybrid optimization algorithms. J Electron Imag 27:1. https://doi.org/10.1117/1.jei.27.6.063008
21. Fayaz S, Parah SA, Qureshi GJ (2022) Underwater object detection: architectures and algorithms – a comprehensive review. Multimed Tools Appl 81:20871–20916. https://doi.org/10.1007/s11042-022-12502-1
22. Gonzalez RC, Woods RE (2008) Digital image processing: international edition, 3rd edn. Pearson, Upper Saddle River, NJ
23. Grosan C, Abraham A (2007) Hybrid evolutionary algorithms: methodologies, architectures, and reviews. In: Hybrid evolutionary algorithms. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 1–17
24. Gupta S, Deep K (2020) Hybrid sine cosine artificial bee colony algorithm for global optimization and image segmentation. Neural Comput Applic 32:9521–9543. https://doi.org/10.1007/s00521-019-04465-6
25. Hammouche K, Diaf M, Siarry P (2010) A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem. Eng Appl Artif Intell 23:676–688. https://doi.org/10.1016/j.engappai.2009.09.011
26. Horng M-H (2010) Multilevel minimum cross entropy threshold selection based on the honey bee mating optimization. Expert Syst Appl 37:4580–4592. https://doi.org/10.1016/j.eswa.2009.12.050
27. Horng M-H, Liou R-J (2011) Multilevel minimum cross entropy threshold selection based on the firefly algorithm. Expert Syst Appl 38:14805–14811. https://doi.org/10.1016/j.eswa.2011.05.069
28. Jiang Z, Zou F, Chen D, Kang J (2021) An improved teaching–learning-based optimization for multilevel thresholding image segmentation. Arab J Sci Eng 46:8371–8396. https://doi.org/10.1007/s13369-021-05483-0
29. Kapur JN, Sahoo PK, Wong AKC (1985) A new method for gray-level picture thresholding using the entropy of the histogram. Comput Vis Graph Image Proc 29:273–285. https://doi.org/10.1016/0734-189x(85)90125-2
30. Karakoyun M, Gülcü Ş, Kodaz H (2021) D-MOSG: discrete multi-objective shuffled gray wolf optimizer for multi-level image thresholding. Eng Sci Technol Int J 24:1455–1466. https://doi.org/10.1016/j.jestch.2021.03.011
31. Khairuzzaman AKM, Chaudhury S (2017) Multilevel thresholding using grey wolf optimizer for image segmentation. Expert Syst Appl 86:64–76. https://doi.org/10.1016/j.eswa.2017.04.029
32. Khalilpourazari S, Pasandideh SHR (2020) Sine–cosine crow search algorithm: theory and applications. Neural Comput Applic 32:7725–7742. https://doi.org/10.1007/s00521-019-04530-0
33. Kotte S, Rajesh Kumar P, Injeti SK (2018) An efficient approach for optimal multilevel thresholding selection for gray scale images based on improved differential search algorithm. Ain Shams Eng J 9:1043–1067. https://doi.org/10.1016/j.asej.2016.06.007
34. Mahajan S, Mittal N, Pandit AK (2021) Image segmentation using multilevel thresholding based on type II fuzzy entropy and marine predators algorithm. Multimed Tools Appl 80:19335–19359. https://doi.org/10.1007/s11042-021-10641-5
35. Mlakar U, Potočnik B, Brest J (2016) A hybrid differential evolution for optimal multilevel image thresholding. Expert Syst Appl 65:221–232. https://doi.org/10.1016/j.eswa.2016.08.046
36. Mohan A, Poobal S (2018) Crack detection using image processing: a critical review and analysis. Alex Eng J 57:787–798. https://doi.org/10.1016/j.aej.2017.01.020
37. Mousavirad SJ, Ebrahimpour-Komleh H (2017) Multilevel image thresholding using entropy of histogram and recently developed population-based metaheuristic algorithms. Evol Intell 10:45–75. https://doi.org/10.1007/s12065-017-0152-y
38. Mousavirad SJ, Ebrahimpour-Komleh H (2020) Human mental search-based multilevel thresholding for image segmentation. Appl Soft Comput 97:105427. https://doi.org/10.1016/j.asoc.2019.04.002

39. Naji Alwerfali HS, AA Al-Qaness M, Abd Elaziz M et al (2020) Multi-level image thresholding based on modified spherical search optimizer and fuzzy entropy. Entropy (Basel) 22:328. https://doi.org/10.3390/e22030328
40. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9:62–66. https://doi.org/10.1109/tsmc.1979.4310076
41. Ouadfel S, Taleb-Ahmed A (2016) Performance study of harmony search algorithm for multilevel thresholding. J Intell Syst 25:473–513. https://doi.org/10.1515/jisys-2014-0147
42. Ouadfel S, Taleb-Ahmed A (2016) Social spiders optimization and flower pollination algorithm for multilevel image thresholding: a performance study. Expert Syst Appl 55:566–584. https://doi.org/10.1016/j.eswa.2016.02.024
43. Rahkar Farshi T (2019) A multilevel image thresholding using the animal migration optimization algorithm. Iran J Comput Sci 2:9–22. https://doi.org/10.1007/s42044-018-0022-5
44. Renugambal A, Selva Bhuvaneswari K (2021) Kapur's entropy based hybridised WCMFO algorithm for brain MR image segmentation. IETE J res 1–20. https://doi.org/10.1080/03772063.2021.1906765
45. Saranya K, Selva Bhuvaneswari K (2022) Semantic annotation of land cover remote sensing images using fuzzy CNN. Intell autom soft comput 33:399–414. https://doi.org/10.32604/iasc.2022.023149
46. Sarkar S, Sen N, Kundu A, das S, Sinha Chaudhuri S (2013) A differential evolutionary multilevel segmentation of near infra-red images using Renyi's entropy. In: Advances in intelligent systems and computing. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 699–706
47. Sathya PD, Kayalvizhi R (2010) PSO-based Tsallis thresholding selection procedure for image segmentation. Int J Comput Appl 5:39–46. https://doi.org/10.5120/903-1279
48. Sathya PD, Kayalvizhi R (2011) Optimal multilevel thresholding using bacterial foraging algorithm. Expert Syst Appl 38:15549–15564. https://doi.org/10.1016/j.eswa.2011.06.004
49. Shen L, Fan C, Huang X (2018) Multi-level image thresholding using modified flower pollination algorithm. IEEE Access 6:30508–30519. https://doi.org/10.1109/access.2018.2837062
50. Singh Gill H, Singh Khehra B, Singh A, Kaur L (2019) Teaching-learning-based optimization algorithm to minimize cross entropy for selecting multilevel threshold values. Egypt Inform J 20:11–25. https://doi.org/10.1016/j.eij.2018.03.006
51. Tuba E, Alihodzic A, Tuba M (2017) Multilevel image thresholding using elephant herding optimization algorithm. In: 2017 14th International Conference on Engineering of Modern Electric Systems (EMES). IEEE
52. Yamini B, Sabitha R (2022) Image steganalysis: real-time adaptive colour image segmentation for hidden message retrieval and Matthew's correlation coefficient calculation. Int J Inf Comput Secur 17:83. https://doi.org/10.1504/ijics.2022.121292
53. Ye Z, Zheng Z, Yu X, Ning X (2006) Automatic threshold selection based on ant colony optimization algorithm. In: 2005 International conference on neural networks and brain. IEEE
54. Ye Z-W, Wang M-W, Liu W, Chen S-B (2015) Fuzzy entropy based optimal thresholding using bat algorithm. Appl Soft Comput 31:381–395. https://doi.org/10.1016/j.asoc.2015.02.012
55. Yue X, Zhang H (2020) A multi-level image thresholding approach using Otsu based on the improved invasive weed optimization algorithm. Sig Imag Video Proc 14:575–582. https://doi.org/10.1007/s11760-019-01585-3
56. Zhou Y, Yang X, Ling Y, Zhang J (2018) Meta-heuristic moth swarm algorithm for multilevel thresholding image segmentation. Multimed Tools Appl 77:23699–23727. https://doi.org/10.1007/s11042-018-5637-x