Expert Systems With Applications 168 (2021) 114426

Contents lists available at ScienceDirect

Expert Systems With Applications

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

A novel evolutionary row class entropy based optimal multi-level thresholding technique for brain MR images

Rutuparna Panda [a,*], Leena Samantaray [b], Akankshya Das [a], Sanjay Agrawal [a], Ajith Abraham [c]

[a] Department of Electronics and Tele-communication Engineering, VSS Univ. of Technology, Burla 768018, India
[b] Department of Electronics and Communication Engineering, ABIT, Cuttack, India
[c] Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, Washington 98071-2259, USA

ARTICLE INFO

Keywords:
Expert system
Optimal multi-level image thresholding
Adaptive Cuckoo Search
Squirrel Search Algorithm
Row Class Entropy

ABSTRACT

The local averaging technique adopted for the construction of 2D histogram in Otsu's method fails to preserve the edge information. Further, the consideration of the diagonal pixels only results in the loss of information. These make the 2D Otsu method of multi-level thresholding inefficient to retain the spatial correlation information. Although the computation of 2D histogram based on gray gradient information is a better way to threshold an image, it faces a backlash due to the high magnitude peaks. To solve these problems, we suggest a new normalized local variance (NLV) method for constructing 2D histogram using the local variance followed by a novel evolutionary row class entropy (ERCE) method for optimal multi-level image thresholding, which tries to preserve maximum spatial information through normalization of the local variance. A new optimization technique called hybrid Adaptive Cuckoo Search-Squirrel Search Algorithm (ACS-SSA) is also introduced. A new fitness function is suggested. The standard CEC 2005 benchmark test functions are used to validate the performance of our proposed ACS-SSA technique. The optimum threshold values obtained are used to segment 100 slices of T-2 weighted axial brain MR images (taken from the Harvard Medical School database). Several performance evaluation metrics are computed to compare the performance of our method with the state-of-the-art methods. The analysis of the results shows that ERCE method outperforms other methods. This method may set a new direction in the multilevel image thresholding research.

1. Introduction

Medical image analysis is a very important contribution of image processing to the present-day society. Image processing seems to be an ever-growing and ever–improving area with endless scopes and possibilities. The effective solutions provided to age old existing problems indicate the dire need of implementation of efficient image processing techniques in the field of medicine. Segmentation of image is a primary step in image analysis. The efficiency of these techniques depends on the accuracy of image segmentation (Dora et al., 2017). A segmented image is represented by each of its constituent parts (Despotović et al., 2015; Mittal & Saraswat, 2018). Thus, image segmentation is the first step that precedes image analysis, disease diagnosis and treatment planning. The thresholding technique is the easiest method of image segmentation. In recent times, magnetic resonance (MR) images have gained greater popularity for image analysis of soft tissues like the brain and spine. This property may be attributed to the superior quality of the images obtained. Here, an external contrast agent is usually not required, because it provides a natural contrast based on varying properties of tissues or blood. It is used for pre-surgical and post-surgical treatment planning of various diseases due to its sensitivity to the differentiation of various neurological tissues (Maitra & Chatterjee, 2008). Its non-intrusive nature and non-usage of ionized radiations saves patients from any fatality in the long run.

Segmentation of brain tissue partitions the brain mainly into white matter (WM), gray matter (GM) and cerebrospinal fluid (CSF) together with its various abnormalities (if present). However, brain tissue segmentation can be a tedious process due to the highly complicated and overtly sensitive nature of this organ. Moreover, the size and location of the constituent parts may vary from patient to patient. Thus, the use of a simple yet effective segmentation technique can address this issue without further complicating it. Several methods for image segmentation have been proposed over the years. As far as we know, thresholding is the simplest known technique for image segmentation.

* Corresponding author.
E-mail addresses: r_ppanda@yahoo.co.in (R. Panda), leena_sam@redeffmail.com (L. Samantaray), ajith.abraham@ieee.org (A. Abraham).

https://doi.org/10.1016/j.eswa.2020.114426
Received 19 July 2019; Received in revised form 29 November 2020; Accepted 29 November 2020
Available online 10 December 2020

Thresholding is the technique of segmenting an image by comparing each pixel intensity value with one or more predetermined threshold values obtained by optimizing a cost function. Expert systems are developed for thresholding based segmentation methods (Dora et al., 2017; Mittal & Saraswat, 2018). The operation of thresholding is befitting for images having multimodal histograms (Panda et al., 2017). An extensive analysis of various methods involved in thresholding has been discussed by Sezgin and Sankur (2004). More on segmentation using thresholding methods is found in (Chatterjee et al., 2012; Nakib et al., 2010).

The use of only one threshold value results in an image with 2 segments. One segment carries pixels whose values are below the threshold value while the other segment contains pixel values above the threshold value. The object of interest is the foreground; the rest of the pixels constitute the background. Thus, it is rightly termed as bi-level thresholding. However, for brain tissue segmentation, bi-level thresholding lacks in imparting useful information, due to the intricacies and complexities involved in the brain. Multi-level thresholding, where two or more threshold values are used, resulting in an image with multiple classes (can give an appropriate solution). Here, each class represents a component of the image and conveys meaningful information.

Several statistical information (first order, second order and higher order statistics) is used to construct the histograms. The main aim of thresholding technique is to obtain the optimum threshold values, which can accurately segment the image under consideration. This segmented image must produce object boundaries clearly. The optimum threshold values are obtained by optimizing a certain fitness function. Later, several performance evaluations are carried out, statistical parameters are used to determine the accuracy of segmentation.

Even though many methods are suggested for multilevel thresholding, ample scope is there for an improvement. The limitations in the conventional 2D histogram based Otsu's method are – inconsideration of the off diagonal information, local averaging to construct the 2D histogram results in loss of high frequency information. To supplement the 2D histogram based Otsu's method, the authors in (Panda et al., 2017) suggested an evolutionary gray gradient algorithm (EGGA) for multilevel thresholding. The gray level gradient magnitude based 2D histogram utilizes the gradient magnitude. The gradient magnitude should be filtered using anisotropic diffusion method before constructing the histogram to prevent information loss. Nevertheless, the authors have not used filtering as a preprocessing step. Thus, the 2D thresholding methods possess some demerits. The 2D Otsu method uses info from the background and the object while ignoring the edge information. The EGGA method never used filtering (which is a difficult task) before thresholding. The use of a suitable filter with appropriate parameters is a difficult task. The construction of 2-D histogram presented in EGGA method used gray gradient magnitude with an idea to retain spatial correlation. However, without filtering, if one uses the gray gradient information directly, then it may lead to poor thresholding due to the presence of high magnitude peaks at some positions of the image.

These limitations motivated us to suggest a novel normalized local variance (NLV) method for construction of 2D histogram using the local variance. It is noteworthy to mention here that the local variance carries the spatial information. While constructing the 2D histogram, the contextual information is retained computing the local variance in a predefined neighbourhood. Hence, NLV based scheme is suggested for construction of the 2D histogram. Furthermore, an evolutionary row class entropy (ERCE) method is proposed to compute the entropy values from the 2D histogram (row wise). Especially, the ERCE based multilevel thresholding method is deployed to obtain improved multi-level threshold outputs.

The structure of the paper is as follows. Section 2 presents the related works. Section 3 presents the proposed methodology. A new hybrid optimization technique is presented in section 4. Results are discussed in section 5. Finally, the conclusions are given in section 6.

2. Related work

The use of 1D histograms of thresholding purpose makes the process very simple and easy. Several researchers have used this method in thresholding applications (Bhandari et al., 2014, 2015; Raja et al., 2015; Hadjidimitriou et al., 2014). These 1D histograms, use first order statistics information like average or variance of a single pixel at a time without considering the relationship between pixels. Maitra and Chatterjee (2008) proposed a new optimal thresholding algorithm using Bacteria foraging (BF) technique. The performance of the concerning algorithm called 'BACTFOR' is based upon the behavior *of E.Coli* bacteria to locate nutrient rich food. They have evaluated the efficiency using several benchmark test functions. They have also analyzed the performance of the algorithm in analysis of MR brain image by optimizing the objective functions (Kapur's entropy and 1D Otsu method). The use of fixed step size in chemotaxis (bacteria foraging) decreases the efficiency of the algorithm. This algorithm has been further enhanced in Sathya and Kayalvizhi (2011a). Here, the authors have implemented an adaptive step size in in chemotaxis (bacteria foraging) the previously existing technique i.e., BF. Maximization of Kapur's entropy and minimization of between class variance were used as objective functions. The use of an adaptive step size in chemotaxis (bacteria foraging) may have resulted in lesser execution time. These authors have also proposed an amended bacteria foraging (ABF) technique in Sathya and Kayalvizhi (2011b). The performance of the algorithm on T2-weighted axial images was compared with that of Genetic algorithm (GA), Particle Swarm Optimization (PSO) and BF. The authors claimed that ABF technique was faster and gave lower standard deviations as compared to existing BF.

In 2014, Manikandan et al. (2014) proposed real coded genetic algorithm. A simulated binary cross over enhanced the performance of the algorithm. This technique was found to have better consistency than Nelder-Mead simplex, PSO, BF and ABF. Priyedarsini et al. (2017) have performed MR brain image segmentation by implementing Kapur's entropy based social group optimization (SGO) technique. The authors have taken this method a step further by implementing pre-processing steps like skull-stripping and post-processing steps (extraction of the tumor using watershed segmentation). Such inclusions obviously resulted in better values of performance evaluation metrics. Rajinikanth et al. (2017) have followed a similar approach. Their method involves skull-stripping followed by optimization of entropy-based objective functions (Kapur's entropy, Shannon's function and Tsallis entropy) using teaching learning-based optimization (TLBO) technique. The post processing step involves active contour segmentation for extraction of tumor. The researchers concluded that Shannon's entropy based TLBO outperformed the rest of the objective functions. In 2017, minimum cross entropy based on a crow search algorithm (MCET-CSA) has been given in Oliva et al. (2017). In this paper, there is a minimization of the cross entropy using the proposed optimization technique. This method was further compared with differential evolution (DE) and harmony search (HS), and has been claimed to be better than them. Recently, Kotte et al. (2018) have proposed an adaptive wind driven optimization (AWDO) technique. This paper uses Kapur's entropy and Otsu's method to evaluate the fitness function. The adaptive nature of the steps increases the overall performance of the optimization technique and makes it better than the earlier proposed WDO. However, due to the lack of spatial information correlation among neighboring pixels 1D histogram-based methods lag behind in giving accurate segmentation results. Several soft computing techniques have been proposed in recent times in order to perform multilevel thresholding of images (Suresha & Parthasarathy, 2018; Xu et al., 2019; Liang et al., 2019).

In order to overcome the issues associated with first-order statistics, researchers have come up with 2D and 3D histograms. The properties of two or more pixel values that occur at different locations and the relationship between them is estimated in higher order statistics. 1D based objective functions perform inefficiently whilst using them with 2D or

3D histograms. Several objective functions have been proposed to deal with higher order statistics. Panda et al. (2017) proposed an evolutionary gray gradient information method (EGGA). Agrawal et al. (2017) have proposed an absolute intensity difference based technique which requires the intensity difference information for thresholding. An adaptive coral reef optimization technique was used to obtain the optimal threshold values. This method was found to be very effective in dealing with intensity inhomogeneity. An interesting amalgamation of both the intensity and edge magnitude techniques has been proposed by Kaur et al. (2018). It even implements GLCM for computing multiple thresholds. These threshold values have been obtained using mutation based particle swarm optimization (MPSO). The authors claim this new technique to outperform all other approaches in all aspects. Feng et al. (2017) proposed a new method for 3D histograms. They have constructed a 3D Otsu histogram based on several information like optimal pixel threshold value and the mean and median of neighboring pixels. The segmented image underwent a fast local Laplacian filtering (FLLP) in order to obtain a filtered image.

3. Proposed methodology

In the 2D Otsu method, a 2D histogram of the concern image is created using the local average versus the gray levels. However, only the diagonal quadrants are considered for the threshold selection. The exclusion of the off-diagonal elements may miss out on any important information in these areas. Moreover, the diagonal region considered is not smooth, has a limitation on providing the edge information of the image. These issues are addressed in Panda et al. (2017). The authors proposed a thresholding technique based on the gradient information of an image. They constructed a 2D histogram by using the gray levels vs the absolute difference of the pixel intensity. The algorithm preserves the relevant information while giving a lesser computational complexity. It is clearly analyzed that the number of quadrants formed decreases. Therefore, the number of computations decreases. This algorithm also preserves the edge information of the image.

However, analysis of the histogram plot reveals high non-uniformity in the peak values of certain magnitudes, this peak value becomes very high which may give rise to magnitude limiting problems. This is a limitation to improving the segmentation accuracy. Hence, it is always advisable to normalize these parameters manually while construction of the histogram. Such a step eliminates any confusion, prevent any loss of information.

In this section, a novel ERCE based multi-level thresholding method is proposed to obtain improved multi-level threshold outputs.

Fig. 1 displays the block diagram of the suggested ERCE technique. The brain MR image together with the desired number of threshold levels (m) are given as the input. The segmented image is obtained as the output. A 2D histogram of the input image is formed using the proposed NLV method. The suggested ERCE method is used to calculate the row class entropy using the 2D histogram. To be precise, row class entropy values are calculated from the 2D histogram constructed using the normalized local variance information. The optimized row class entropy of this histogram is computed using the newly proposed hybrid ACS-SSA technique. The threshold values, thus obtained, are used to segment the brain MR image. The computation of our proposed algorithm is discussed below.

Let I represent a gray level image of size $M \times N$. It is assumed that there is a total of L gray levels in the image. The pixel intensity is represented as: $g = \{1, 2, ..., L\}$

Now, let $f(x, y)$ indicates the gray level of the image at coordinates $\times$ and y. Here, the maximum value of $\times$ is M and that of y is N. $h(x,y)$ is the local average gray value denoted as:

$$h(x,y) = \lfloor \frac{1}{w \times w} \sum_{a=-l}^{l} \sum_{b=-l}^{l} f(x+a, y+b) \rfloor \quad (1)$$

where, $l = \lfloor w/2 \rfloor$ and $w < min(M,N)$. Note that $w$is usually taken as an odd number.

The local variance $g(x, y)$ is represented as:

$$g(x,y) = (f(x,y) - h(x,y))^2 \quad (2)$$

This local variance is further normalized to eliminate the high magnitude peaks. The normalized local variance is

$$g_n(x,y) = \frac{(g(x,y) - g_{min}) \times K}{g_{max} - g_{min}} \quad (3)$$

where g_{max} and g_{min} are the maximum and minimum values of the local variance, respectively. Here, K is a constant whose value is assumed to be fixed at 256.

Let $f(x, y) = i$, $g_n(x, y) = j$, the time of occurrence of the pair $(i,j) = q_{ij}$.

And so, the frequency of occurrence of (i,j) is

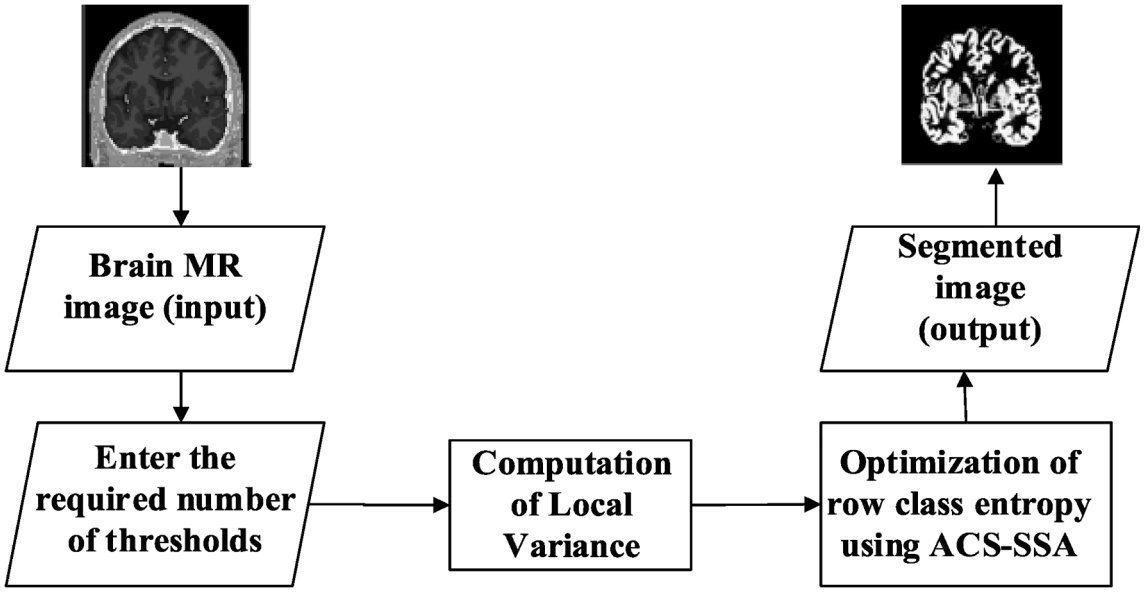

Fig. 1. Block diagram of the proposed segmentation technique.

$$p_{ij} = \frac{q_{ij}}{M \times N} \; 1 \leqslant i \leqslant L \text{ and } 1 \leqslant j \leqslant L \tag{4}$$

In Fig. 2, we clearly see that for bi-level thresholding, the 2D histogram is partitioned into 4 quadrants. There is only a single threshold value at *(S, T)*. The first two quadrants convey the edge information. The 1st quadrant is assumed to be the background (class C_1) and the 2nd quadrant as the foreground (class C_2).

The probability distribution of $C_1$is

$$P_1(C_1) = \sum_{i=1}^{S} \sum_{j=1}^{T} p_{ij} \tag{5}$$

Similarly, the probability distribution of $C_2$is

$$P_2(C_2) = \sum_{i=1}^{S} \sum_{j=T+1}^{L} p_{ij} \tag{6}$$

The class probabilities of C_1 and$C_2$are given as:

$$C_1 : \left\{ \frac{p_{ij}}{P_1}, i = 1, 2, ..., S; j = 1, 2, ..., T \right\} and\, C_2 : \left\{ \frac{p_{ij}}{P_2}, i = 1, 2, ..., S; j = T+1, T+2, ..., L \right\}$$

The entropy, dependent on a threshold value (*S, T*), is formulated as:

$$E_1(S,T) = -\sum_{i=1}^{S} \sum_{j=1}^{T} \left(\frac{p_{ij}}{P_1}\right) ln\left(\frac{p_{ij}}{P_1}\right) \tag{7}$$

and

$$E_2(S,T) = -\sum_{i=1}^{S} \sum_{j=T+1}^{L} \left(\frac{p_{ij}}{P_2}\right) ln\left(\frac{p_{ij}}{P_2}\right) \tag{8}$$

From the additive property of entropy, we get:

$$E_{Total}(S,T) = E_1(S,T) + E_2(S,T) \tag{9}$$

Thus, the new objective function is obtained by maximizing this total entropy as

$$(S_{opt}, T_{opt}) = argmax\{E_{Total}(S,T)\} \tag{10}$$

From Fig. 3, it is observed that for 2 threshold values, the histogram is divided into 6 regions. It is the intersection of (*S, T_1*) and (*S, T_2*) values. All the information is found along the edges. In addition, due to normalization the information loss is minimum. Similarly, for 3 threshold values, the entire region gets divided into 8 sub-regions, and so on.

It is assumed that for multi-level thresholding, the histogram has

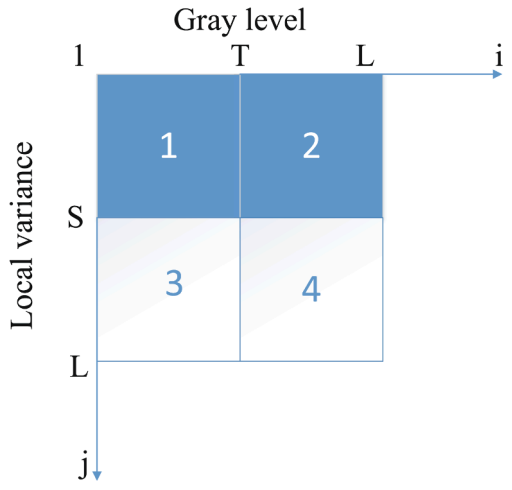

Fig.2. Graphical presentation of 2-D histogram (for bi-level thresholding).

Fig.3. Graphical presentation of 2D histogram (for multi-level thresholding).

been divided into '*k*' number of classes. Thus, the probability distribution of classes, $C_1, C_2, ..., C_k$ are:

$$P_1(C_1) = \sum_{i=1}^{S} \sum_{j=1}^{T_1} p_{ij}$$

$$P_2(C_2) = \sum_{i=1}^{S} \sum_{j=T_1+1}^{T_2} p_{ij}$$

...

$$P_k(C_k) = \sum_{i=1}^{S} \sum_{j=T_{k-1}+1}^{L} p_{ij} \tag{11}$$

The row class entropies are given as:

$$E_1(S,T_1) = -\sum_{i=1}^{S} \sum_{j=1}^{T_1} \left(\frac{p_{ij}}{P_1}\right) ln\left(\frac{p_{ij}}{P_1}\right)$$

$$E_2(S,T_2) = -\sum_{i=1}^{S} \sum_{j=T_1+1}^{T_2} \left(\frac{p_{ij}}{P_2}\right) ln\left(\frac{p_{ij}}{P_2}\right)$$

...

$$E_k(S,T_{k-1}) = -\sum_{i=1}^{S} \sum_{j=T_{k-1}+1}^{L} \left(\frac{p_{ij}}{P_k}\right) ln\left(\frac{p_{ij}}{P_k}\right) \tag{12}$$

On adding, the total Row Class Entropy (RCE) is obtained as:

$$E_{Total}(ST_1, ST_2, ..., ST_{k-1}) = E_1(S,T_1) + E_2(S,T_2) + ... + E_k(S,T_{k-1}) \tag{13}$$

The maximization of the above total entropy results in the new objective function and is defined as:

$$\left(S_{opt}T_{opt_1}, S_{opt}T_{opt_2}, ..., S_{opt}T_{opt_{k-1}}\right) = \underset{1 \leqslant ST_i \leqslant k-1}{argmax} \{E_{Total}(ST_1, ST_2, ..., ST_{k-1})\} \tag{14}$$

Fig. 4 visually represents our proposed 2D histogram. The original image is taken from Berkeley Segmentation Dataset, displayed in Fig. 4 (a). Its corresponding 3 dimensional view of the 2-dimensional histogram constructed using Otsu method is given in Fig. 4(b). Fig. 4(c) and (d) are the 2-D histogram plots constructed using EGGA and local variance methods, respectively. It can be seen that all the information is available along the diagonal while using the first method. In the second method, information is pushed to the edge which makes the data extraction task simpler. In our proposed method, the normalization of the high magnitude peaks definitely succeeds in preserving a greater amount of information. In this work, Eq. (14) is used as the objective

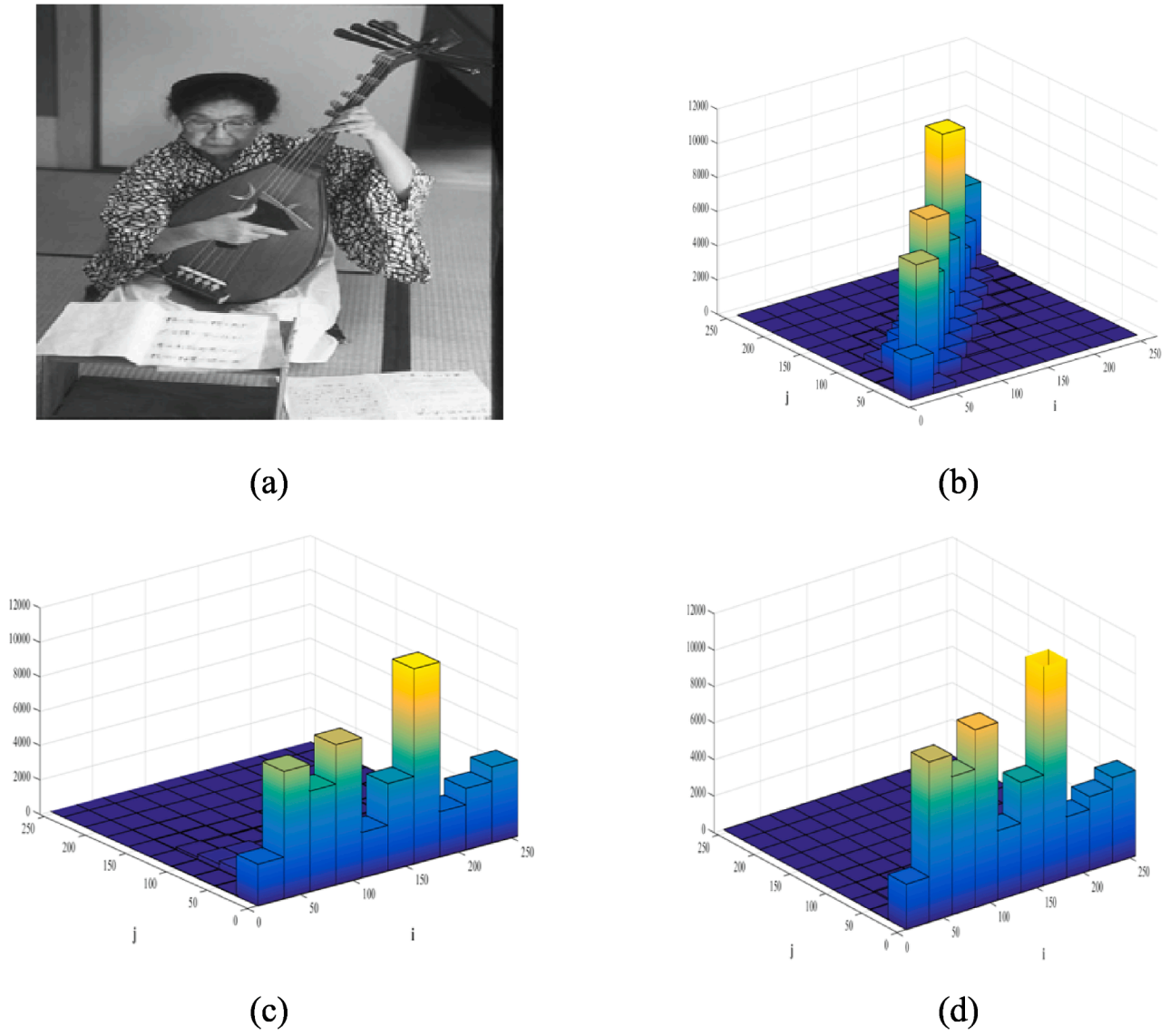

Fig. 4. Representation of 2D histogram for (a) training image 'sitar' from BSD data set (b) 2D Otsu method (c) EGGA method (d) NLV method.

function.

4. Optimization techniques

4.1. Cuckoo search algorithm (CS)

CS (Yang & Deb, 2009) is based on the aggressive reproduction strategy of the cuckoo birds. These birds display an intelligent behavior during reproduction where they very tactfully lay eggs in the nest of other horde birds. Their intelligence can be attributed to the mimicking of color and pattern of eggs along with the time of egg-laying by the female cuckoo with other horde birds. The authors have assumed 3 important points: i) each cuckoo lays only one egg at a time and that too in a random manner, ii) only the best nests get transferred to the next generation and iii) available host nests are fixed and their probability of discovering the alien egg is given by $pa \in [0,1]$. The aim of the algorithm is to get the nest with best eggs. The best nests are the new eggs from Cuckoo bird. Thus, out of all the solutions (eggs in nests), the nest with a cuckoo egg (objective function) having a maximum fitness value gives the best solution. They have used Levy steps using Mantegna's algorithm to explore the search space. The new solution is modelled as:

$$s_j(t+1) = s_j(t) + \alpha \oplus Levy(\lambda) \tag{15}$$

where, $s_j\ (t) = j^{th}$ Cuckoo's current search space; α = constant (greater than 0 and usually fixed at 1); ⊕ = entry-wise multiplication; and $j = 1, 2,\ldots, N$ (number of host nests).

The Levy flight takes a random walk from Levy distribution; the Levy step is consequently calculated. The major drawback of this algorithm is sub-optimal convergence. Due to the use of fixed step sizes, the convergence parameters may get caught up in local maxima or minima. This attribute also results in greater convergence time.

4.2. Adaptive cuckoo search (ACS)

Naik and Panda (2016) have come up with an interesting solution to address the gap area in CS. They have replaced the fixed Levy steps by adaptive steps. The steps are called adaptive due to their characteristic feature of decrement in the step size with increment in the number of iterations. This adaptive step size is calculated using:

$$step_j(t+1) = \frac{1}{t}^{\left|\left((bestf(t)-f_j(t))/(bestf(t)-worsf(t))\right)\right|} \tag{16}$$

where, t = number of generations of Cuckoo search; $f_j(t)$ = fitness value of the objective function, j denotes the j^{th} Cuckoo, $j = 1, 2, \ldots, N$, *bestf(t)* = best fitness value at t; *worsf(t)* = worst fitness value at t. The convergence value becomes better with the implementation of adaptive step size and approaches the global optimum solution in lesser time as compared to CS. The new optimum solution is given as:

$$s_j(t+1) = s_j(t) + rand \times step_j(t+1) \tag{17}$$

4.3. Squirrel search algorithm (SSA)

This new algorithm proposed by Jain et al. (2019) is based on the dynamic foraging strategy of southern flying squirrels. These squirrels follow an intelligent means of locomotion in the form of gliding. Gliding facilitates this mammal in covering more distances in less time with the consumption of minimum amount of energy. Extreme weather conditions, scarcity of food resources and higher risk of predation (due to loss of forest cover) make survival a challenging task for the squirrels during winter. Hickory nuts fulfill greater energy requirements than acorn nuts. Thus, these squirrels save up the hickory nuts for consumption in winter when their body demands greater energy and in order to do so they feed on the acorn nuts which are abundantly available during favorable conditions. Depending on the current and consequent locations of the mammals foraging procedure is divided into 3 categories:

Flying squirrels present on acorn trees and moving towards hickory trees.

$$s_{at}(t+1) = s_{at}(t) + dg \times gc \times (s_{ht}(t) - s_{at}(t)) \tag{18}$$

Those present on normal trees and approaching acorn trees.

$$s_{nt}(t+1) = s_{nt}(t) + dg \times gc \times (s_{at}(t) - s_{nt}(t)) \tag{19}$$

Those squirrels present on normal trees and gliding towards hickory trees.

$$s_{nt}(t+1) = s_{nt}(t) + dg \times gc \times (s_{ht}(t) - s_{nt}(t)) \tag{20}$$

where, dg and gc are the gliding distance and gliding coefficient which are constants. It must be noted that the above equations are valid if the predation probability constant is satisfied else random allocation occurs. A seasonal constant monitors the end of winter when the mammals are again free to travel and get randomly dispersed. The performance of the algorithm is evaluated on 26 standard benchmark functions and compared with genetic algorithm (GA), particle swarm optimization (PSO), firefly optimization (FF), multi-verse algorithm (MVA), krill herd (KH). SSA outperformed the rest. The analysis of the experimental results shows that the convergence rate is appreciable. However, search space exploration could have been better. A majority of the graphs reflect a sudden fall in the convergence rate and this steep slope is a negative indicator of good exploitation quality.

4.4. Proposed hybrid ACS-SSA

In ACS, the introduction of adaptive steps maintains an exploration and exploitation balance. Here, the search strategy is conveyed as a series of steps with each step being influenced by the outcome of the previous steps. This methodology has proven excellent for a quicker convergence. However, there is no inbuilt constraint in the ACS to evaluate the search strategy when the abandon probability becomes very less (less than 0.1). Hence, the dynamic foraging behavior of SSA is supplemented to achieve better results.

When $rand < pa$,

$$s_j(t+1) = s_j(t) + rand \times fc \times (bestf(t) - f_j(t)) \tag{21}$$

Here, we have incorporated a constant called flying coefficient fctogether with the difference between the current and the best fitness values. This helps in augmenting the exploitation capacity of the algorithm. We have fixed the value of fcat 1.9. For abandon probability greater than 0.1, the original equation of ACS is kept intact.

Pseudo-code for ACS-SSA:
Begin:
Define input parameters
Define an objective function
Declare the number of host nests N
Random initialization of host nests (d-dimensional) as $s_j = (s_j^1, s_j^2, \ldots, s_j^d)$ *for* $j = 1$ *to N*
Calculate value of objective function for the host nest for 1st iteration
while(termination criterion is not satisfied)
Begin
Calculate bestf and worsf
Then calculate the step size using Eq. (16)
Calculation of new position of Cuckoo nest
if (rand < pa)
use Eq. (21)
else
use Eq. (17)
Objective function of host nests is then evaluated
Determine the best nest comparison with other nests
Abandonment of worst nests
End
Best solutions are obtained in descending order
End

4.5. Performance evaluation of ACS-SSA

In order to evaluate the performance of our proposed algorithm, it is tested using 14 benchmark test functions. They are from CEC 2005 benchmark test functions (García et al., 2009). These test functions have again been divided into 2 categories: unimodal and multimodal. These are given in Tables 1 and 2, respectively. Table 3 represents expanded test functions. The proposed algorithm is compared with that of CS, ACS and SSA.

Where $F_7(X)$ is the Griewank's Function

$$F_7(X) = \frac{Z_i^2}{4000} - \prod_{i=1}^{d} cos\left(\frac{Y_i}{\sqrt{i}}\right) + 1$$

$F_6(X)$ is Rosenbrock's Function

$$F_6(X) = \sum_{i=1}^{d-1}\left(100\left(Z_i^2 - Z_{i+1}\right)^2 + \left(Z_i - 1\right)^2\right)$$

$$F(W,X) = 0.5 + \frac{\left(sin^2\left(\sqrt{W^2 + X^2}\right) - 0.5\right)}{\left(1 + 0.001\left(X^2 + W^2\right)\right)^2}$$

In Table 4, we present the various control parameters used in CS, ACS, SSA and hybrid ACS-SSA. The results obtained are the best of results of over 50 independent runs for each test function. These control parameters are selected based on the output obtained by extensive simulation.

Smaller steps result in greater convergence time while large steps give low precision results. Our algorithm strikes a perfect balance between the both. The performances of different algorithms are displayed in Figs. 5–10. Fig. 5 gives the performance comparison of different algorithms for fitness function F3. It is clearly visible that SSA converges faster than other algorithms. However, fast convergence is not always the only criteria to judge an algorithm. On detailed analysis, it is seen that the slope of SSA is steep; the steps taken are huge and undesirable. Thus, here the exploration and exploitation capacity is very poor. This may impair the quality of results. On the other hand, our proposed algorithm maintains a healthy exploration–exploitation balance together with a good convergence rate. A similar notion is again observed for the fitness function F4. Huge steps badly influence the output quality of SSA. However, the adaptive steps of ACS coupled with the gliding coefficient condition of SSA make the proposed hybrid technique a decent

Table 1
Unimodal functions ($d = 30$).

Function	Description	Range of 'X'	f_bias	Global optimal
Shifted sphere function	$F_1(X) = \sum_{i=1}^{d} Z_i^2 + f\_bias_1$	$[-100, 100]^d$	$F_1(X^*) = f\_bias_1 = -450$	$X^* = 0$
Shifted Schwefel's problem 1.2	$F_2(X) = \sum_{i=1}^{d} \left(\sum_{j=1}^{i} Z_j\right)^2 + f\_bias_2$	$[-100, 100]^d$	$F_2(X^*) = f\_bias_2 = -450$	$X^* = 0$
Shifted rotated high conditioned elliptic function	$F_3(X) = \sum_{i=1}^{d} (10^6)\frac{i-1}{d-1} Y_i^2 + f\_bias_3$	$[-100, 100]^d$	$F_3(X^*) = f\_bias_3 = -450$	$X^* = 0$
Shifted Schwefel's problem 1.2 with noise in fitness	$F_4(X) = \left(\sum_{i=1}^{d} \left(\sum_{j=1}^{i} Z_j\right)^2\right) * (1 + 0.4\lvert N(0,1)\rvert) + f\_bias_4$	$[-100, 100]^d$	$F_4(X^*) = f\_bias_4 = -450$	$X^* = 0$
Schwefel's problem 2.6 with global optimum on bounds	$F_5(X) = max(\lvert A_i X - B_i\rvert) + f\_bias_5$	$[-100, 100]^d$	$F_5(X^*) = f\_bias_5 = -310$	$X^* = 0$

Table 2
Multimodal functions ($d = 30$).

Function	Description	Range of 'X'	f_bias	Global optimal
Shifted Rosenbrock's Function	$F_6(X) = \sum_{i=1}^{d-1} \left(100(Z_i^2 - Z_{i+1})^2 + (Z_i - 1)^2\right) + f\_bias6$	$[-100, 100]^d$	$F_6(X^*) = f\_bias6 = 390$	$X^* = 0$
Shifted rotated Griewank's Function without bound	$F_7(X) = \sum_{i=1}^{d} \frac{Z_i^2}{4000} - \prod_{i=1}^{d} cos\left(\frac{Y_i}{\sqrt{i}}\right) + 1 + f\_bias_7$	$[0, 600]^d$	$F_7(X^*) = f\_bias7 = -180$	$X^* = 0$
Shifted rotated Ackley's with global optimum on bounds	$F_8(X) = -20exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} P_i^2}\right) - exp\left(\frac{1}{d}\sum_{i=1}^{d} cos(2\pi P_i)\right) + 20 + e + f\_bias_8$	$[-32, 32]^d$	$F_8(X^*) = f\_bias8 = -140$	$X^* = 0$
Shifted Rastrigin's Function	$F_9(X) = \sum_{i=1}^{d} \left((Z_i)^2 - 10cos(2\pi Z_i) + 10\right) + f\_bias9$	$[-5, 5]^d$	$F_9(X^*) = f\_bias9 = -330$	$X^* = 0$
Shifted Rotated Rastrigin's Function	$F_{10}(X) = \sum_{i=1}^{d} \left((R_i)^2 - 10cos(2\pi R_i) + 10\right) + f\_bias10$	$[-5, 5]^d$	$F_{10}(X^*) = f\_bias10 = -330$	$X^* = 0$
Shifted Rotated Weierstrass's Function	$F_{11}(X) = \sum_{i=1}^{d} \left(\sum_{k=0}^{20} 0.5^k cos\left(2\pi 3^k (U_i + 0.5)\right)\right) - d\sum_{k=0}^{20} 0.5^k cos\left(2\pi 3^k .0.5\right) + f\_bias11$	$[-0.5, 0.5]^d$	$F_{11}(X^*) = f\_bias11 = 90$	$X^* = 0$
Schwefel's problem 2.13	$F_{12}(X) = \sum_{i=1}^{d} (A_i - B_i(X))^2 + f\_bias12$	$[-\pi, \pi]^d$	$F_{12}(X^*) = f\_bias12 = -460$	$X^* = 0$

Table 3
Expanded functions ($d = 30$).

Function	Description	Range of 'X'	f_bias	Global optimal
Shift Expanded Griewank's plus Rosenbrock's Function	$F_{13}(X) = F_7(F_6(Z_1, Z_2)) + F_7(F_6(Z_2, Z_3)) + \ldots\ldots + F_7(F_6(Z_{d-1}, Z_d)) + F_7(F_6(Z_d, Z_1)) + f\_bias_{13}$	$[-5, 5]^d$	$F_{13}(X^*) = f\_bias13 = -130$	$X^* = 0$
Shift Expanded Scaffer's Function	$F_{14}(X) = EF(Z_1, Z_2, \ldots.Z_d) = F(Z_1, Z_2) + F(Z_2, Z_3) + \ldots\ldots + F(Z_{d-1}, Z_d) + F(Z_d, Z_1) + f\_bias_{14}$	$[-100, 100]^d$	$F_{14}(X^*) = f\_bias14 = -300$	$X^* = 0$

Table 4
Parameters chosen.

Algorithm	Parameters
CS	N = 25, α = 1, λ = 1.5, pa = 0.25, tmax = 2000
ACS	N = 25, pa = 0.25, tmax = 2000
SSA	N = 25, gc = 1.9, dg = 9 to 25, pp = 0.1, tmax = 2000
ACS-SSA	N = 25, pa = 0.1, fc = 1.9, tmax = 2000

algorithm.

In Fig. 7, it is seen that SSA fails to explore the entire search space. All other algorithms exhibit good search capability. ACS takes lesser convergence time as compared to CS due to the presence of less number of parameters in the later. Moreover, ACS-SSA out beats the others by having a faster convergence rate. This may be attributed to the introduction of a new condition. Even in Fig. 8, our algorithm outperforms the rest. SSA with its poor search strategy is unable to converge down. This may be due to the limitations of the algorithm. While our method does it efficiently in less time. For fitness function F11, it is observed that ACS performs more efficiently than the rest (Fig. 9). For fitness function F15, ACS-SSA performs well in less time (Fig. 10).

The performance of the optimization technique can be evaluated by using various statistical parameters like best value, mean value and standard deviation. The best fitness value is the best result and in our case it is the minimum fitness value. The mean value denotes the average of all fitness values for a given fitness function. Standard deviation reflects the value by which the obtained result is different from the ideal value. This parameter must be as less as possible. The average time column is inserted to track down the time taken by each algorithm for a specific fitness function. The lesser is the convergence time, the better is the performance of the algorithm. However, it is not the sole criteria. A decent convergence time is always appreciable. These results are an average of over 50 independent runs. The best results are displayed in bold face.

Table 5 gives the performance comparison of all the 4 algorithms for unimodal fitness functions (1–5). These fitness functions have been discussed earlier in table 1. The best, mean and standard deviation values are obtained after a series of 50 iterations for each technique.

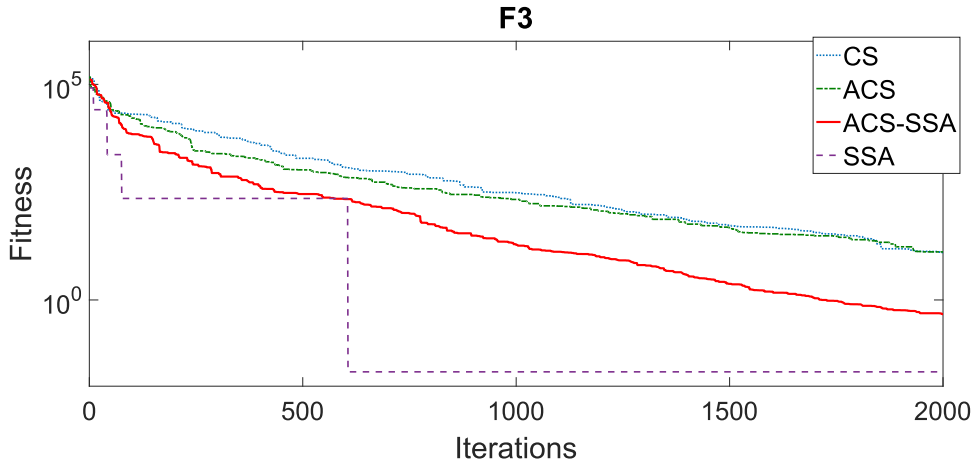

Fig. 5. Iteration vs fitness value of fitness function number 3.

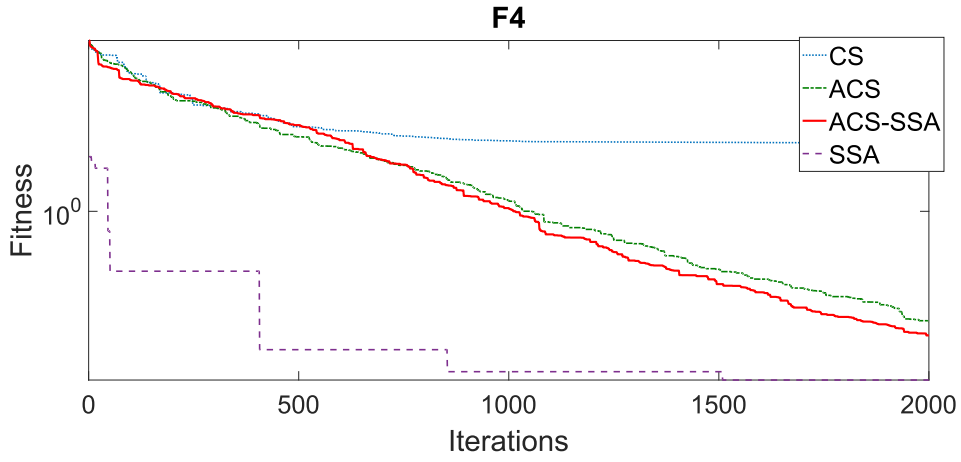

Fig. 6. Iteration vs fitness value of fitness function number 4.

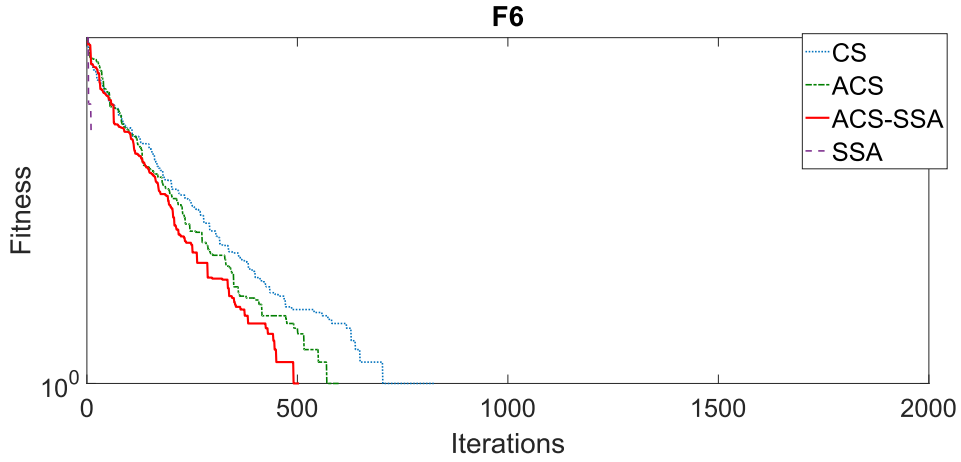

Fig. 7. Iteration vs fitness value of fitness function number 6.

ACS-SSA performs better than the rest of the techniques. Table 6 gives the results comparison for multimodal fitness functions (6–12). Even here ACS-SSA outperforms. Extended multimodal fitness functions 13 and 14 with their results are given in Table 7.

5. Multi-level thresholding of brain MR images using ERCE technique

Here, 100 slices (slice #20 to slice #119) of brain MR images with T2 modality from the Harvard Medical School database are used to evaluate our proposed technique. T2 weighted structure is preferred over other available modalities, because it yields better results for intra-tissue

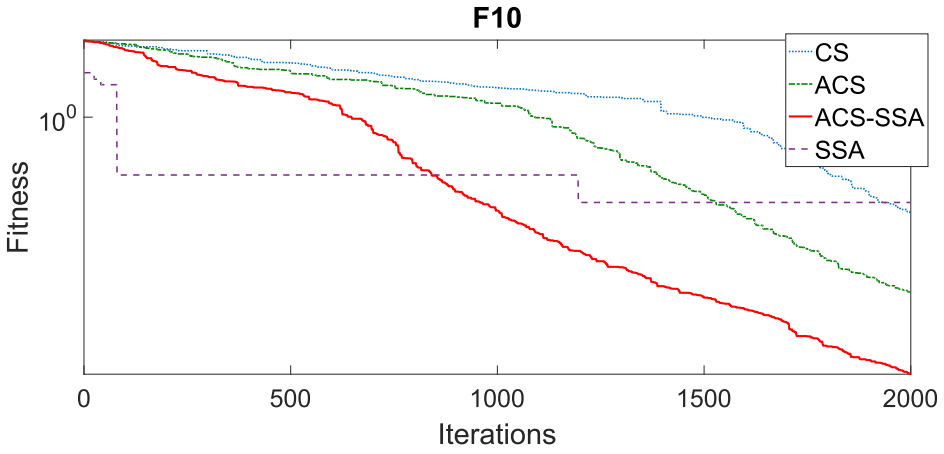

Fig. 8. Iteration vs fitness value of fitness function number 10.

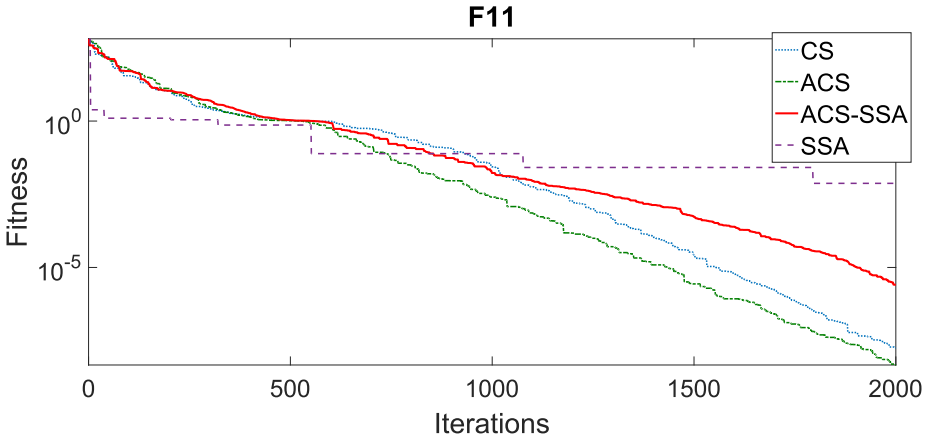

Fig. 9. Iteration vs fitness value of fitness function number 11.

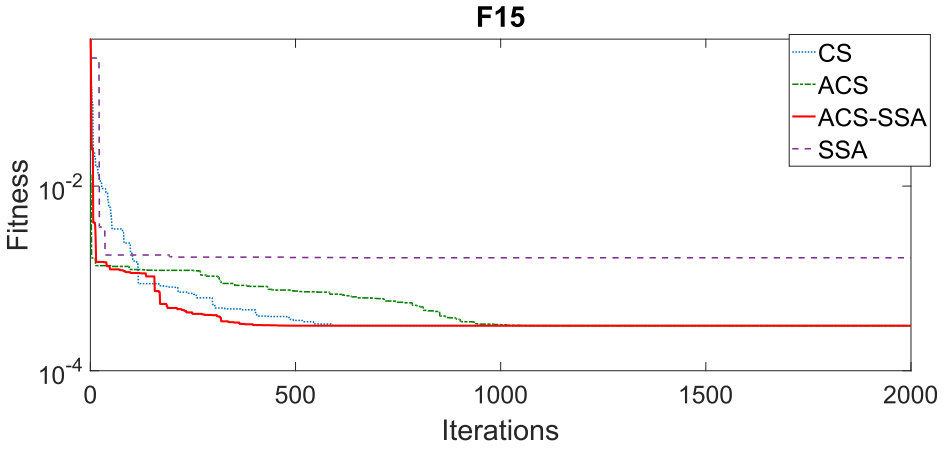

Fig. 10. Iteration vs fitness value of fitness function number 15.

segmentation. The experiment is performed using a computer having a 4 GB RAM, Intel core i-5 processor, Windows 10 operating system. The uniqueness of our paper lies in the method of construction of the 2D histogram of the input image. In order to evaluate the robustness of this technique, the results are compared with other 2D histogram construction methods like counterpart EGGA and the conventional 2D Otsu method. Our proposed soft computing technique is compared with 3 different established techniques. The new algorithm (adaptive cuckoo search-squirrel search algorithm) which is the combination of both the techniques was found to perform better than the individual algorithms (it is comprised of). Adaptive cuckoo search is a much more robust and efficient technique than a cuckoo search algorithm (Naik & Panda, 2016). Squirrel search is a new algorithm in the field of soft computing. The hybridization of both the algorithms is a significant improvement in the area of exploration and exploitation of search space. This alludes to the results obtained in the tables mentioned. In order to gauge the efficiency of our method, 8 different performance evaluation metrics are verified. This new algorithm is found to surpass its counterparts in the

Table 5
Statistical Performance Analysis.

Function	Algorithm	Best value	Mean value	Std. deviation	Avg. time
F_1	CS	3.02E−23	3.79E−21	8.56E−21	5.76E + 00
	ACS	2.98E−23	3.73E−21	8.43E−21	**4.56E + 00**
	SSA	2.28E−11	5.04E−10	8.04E−10	5.88E + 00
	ACS-SSA	**2.94E−23**	**3.67E−21**	**8.23E−21**	4.76E + 00
F_2	CS	1.96E−13	1.07E−11	1.71E−11	6.02E + 00
	ACS	1.87E−13	1.02E−11	1.63E−11	**4.78E + 00**
	SSA	1.44E−05	4.75E−05	2.36E−05	6.26E + 00
	ACS-SSA	**1.81E−13**	**9.89E−12**	**1.55E−11**	4.92E + 00
F_3	CS	4.96E−01	1.42E + 01	7.30E + 00	2.16E + 01
	ACS	2.12E−01	7.21E + 00	3.83E + 00	**2.02E + 01**
	SSA	5.21E−01	1.49E + 01	7.66E + 00	2.27E + 01
	ACS-SSA	**2.08E−01**	**7.06E + 00**	**3.75E + 00**	2.17E + 01
F_4	CS	1.59E−01	4.54E + 00	2.22E + 00	9.76E + 00
	ACS	2.45E−02	1.60E + 00	1.36E + 00	**7.80E + 00**
	SSA	1.67E-01	4.77E + 00	2.33E + 00	1.02E + 01
	ACS-SSA	**2.40E−02**	**1.57E + 00**	**1.33E + 00**	7.95E + 00
F_5	CS	1.31E−01	2.70E + 01	1.68E + 01	7.89E + 00
	ACS	8.28E−02	2.25E + 01	1.64E + 01	**5.56E + 00**
	SSA	1.37E−01	2.84E + 01	1.77E + 01	8.28E + 00
	ACS-SSA	**8.11E−02**	**2.20E + 01**	**1.61E + 01**	5.85E + 00

Bold values indicate best results.

Table 6
Statistical Performance Analysis.

Function	Algorithm	Best value	Mean value	Std. deviation	Avg. time
F_6	CS	1.50E−02	3.36E−02	2.15E−02	6.74E + 00
	ACS	5.17E−03	1.59E−02	5.36E−03	**5.59E + 00**
	SSA	1.57E−02	3.53E−02	2.26E−02	7.07E + 00
	ACS-SSA	**5.07E−03**	**1.56E−02**	**5.25E-03**	5.88E + 00
F_7	CS	−9.31E + 03	−8.13E + 03	5.50E + 02	6.48E + 00
	ACS	−9.53E + 03	−8.84E + 03	2.93E + 02	**5.35E + 00**
	SSA	−9.34E + 03	−7.97E + 03	5.62E + 02	6.80E + 00
	ACS-SSA	**−9.77E + 03**	**−9.28E + 03**	**2.79E + 02**	5.84E + 00
F_8	CS	3.67E−01	5.87E + 01	1.79E + 01	6.36E + 00
	ACS	3.30E−01	5.22E + 01	1.83E + 01	**5.22E + 00**
	SSA	3.74E−01	5.99E + 01	1.04E + 01	6.68E + 00
	ACS-SSA	**3.14E−01**	**4.97E + 01**	**9.87E + 00**	5.52E + 00
F_9	CS	1.04E−04	5.27E−01	6.86E−01	6.31E + 00
	ACS	2.54E−11	8.63E−08	2.44E−07	**5.07E + 00**
	SSA	1.09E−04	5.53E−01	7.20E−01	6.62E + 00
	ACS-SSA	**2.49E−11**	**8.45E−08**	**2.40E−07**	5.97E + 00
F_{10}	CS	4.59E−10	2.57E−03	1.75E−03	6.86E + 00
	ACS	0.00E + 00	4.64E−04	1.28E−03	**5.67E + 00**
	SSA	4.82E−10	2.70E−03	1.77E−03	7.20E + 00
	ACS-SSA	**0.00E + 00**	**4.54E−04**	**1.22E−03**	6.56E + 00
F_{11}	CS	8.47E−08	2.64E−01	3.06E−01	1.88E + 01
	ACS	2.97E−20	6.49E−03	3.56E−02	**1.68E + 01**
	SSA	8.90E−08	2.77E−01	3.21E−01	1.98E + 01
	ACS-SSA	**2.91E−20**	**6.36E−03**	**3.49E−02**	1.75E + 01
F_{12}	CS	2.71E−09	4.94E−01	2.01E + 00	1.88E + 01
	ACS	1.11E−21	3.44E−04	1.88E−03	**1.59E + 01**
	SSA	2.84E−09	5.19E−01	2.11E + 00	1.97E + 01
	ACS-SSA	**1.09E−21**	**3.38E−04**	**1.84E−03**	1.66E + 01

Bold values indicate best results.

majority of the cases. An optimization algorithm requires an objective function, which upon maximization or minimization yields the optimal values. Here, 2D row class entropy is used as the objective function, this when maximized gives the threshold values. It may be noted that since random values are used to generate the initial solution in an optimization technique, the solution obtained in the first iteration is usually unreliable, and thus, the results that are considered in this paper are obtained after 100 complete iterations of each individual algorithm.

5.1. Results and discussion

There are 2 methods of image quality assessment (IQA) – i) Subjective methods: these are based on human perception and ii) Objective methods: these evaluate a fixed set of assessment algorithms (Kuo et al., 2016). Structural Similarity index (SSIM) is an efficient objective IQA metric with low computation complexity. SSIM takes into consideration the structure, contrast and brightness of the images. Thus, a higher SSIM value means a better threshold image. Table 8 gives the average SSIM values of the threshold image computed using different methods. The results indicate that ACS-SSA gives greater SSIM value than other soft computing methods at the same threshold level. This value is found to increase with the increase in threshold levels. On further analysis at threshold level 4, we see that the performance of ACS-SSA is better than others. The superiority of ACS-SSA is established by its better search space exploration and exploitation capability. We can also see that the results obtained for ERCE using ACS-SSA at threshold level 4 is greater than EGGA and 2D Otsu method. The reason may be attributed to the preservation of edge information without truncation of very high values in ERCE method. In Table 9, we discuss the FSIM values of the thresholded image. Feature Similarity Index (FSIM) is a novel image quality assessment index where the gradient magnitude (GM) and the phase congruency (PC) of an image are taken into consideration (Zhang et al., 2011). A high FSIM value is desirable for a good segmented image. It can be noted that with the increase in threshold levels, the FSIM value increases.

We also note that the values are consistently high for our proposed method. It is better than 2D Otsu and EGGA. Incidentally, for FSIM, all the information is pushed to the edge which enhances the gradient magnitude. Moreover, the normalization of this edge gradient preserves

Table 7
Statistical Performance Analysis.

Function	Algorithm	Best value	Mean value	Std. deviation	Avg. time
F_{13}	CS	9.39E−01	9.39E−01	7.67E−08	1.76E + 00
	ACS	9.39E−01	9.36E−01	1.21E−16	**1.65E + 00**
	SSA	9.86E−01	9.86E−01	8.05E−08	1.84E + 00
	ACS-SSA	**9.20E−01**	**9.20E−01**	**1.18E−16**	1.72E + 00
F_{14}	CS	5.72E−04	6.77E−04	7.12E−04	5.82E−01
	ACS	4.21E−04	6.43E−04	1.04E−05	**5.05E−01**
	SSA	6.01E−04	7.11E−04	7.47E−04	6.11E−01
	ACS-SSA	**4.13E−04**	**6.30E−04**	**1.02E−05**	5.35E−01

Bold values indicate best results.

Table 8
SSIM values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	0.92850	0.96701	0.99274
	ACS	0.93755	0.99046	**0.99456**
	SS	0.92632	0.95331	0.99200
	ACS-SSA	0.96307	0.99123	0.99149
3	CS	0.92903	0.90129	0.99255
	ACS	0.95954	0.99557	0.99551
	SS	0.93350	0.94119	0.96638
	ACS-SSA	0.96992	0.99499	**0.99561**
4	CS	0.94758	0.98296	0.99365
	ACS	0.97397	0.99684	0.99567
	SS	0.94276	0.95434	0.97777
	ACS-SSA	0.97194	0.99607	**0.99681**
5	CS	0.95483	0.98495	0.99187
	ACS	0.97919	0.99649	0.99714
	SS	0.95263	0.96887	0.98196
	ACS-SSA	0.99619	0.99672	**0.99743**

Bold values indicate best results.

Table 9
FSIM values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	0.92060	0.97722	0.99136
	ACS	0.97720	0.99170	0.99140
	SS	0.92441	0.96692	0.99143
	ACS-SSA	0.96507	0.99720	**0.99227**
3	CS	0.93657	0.98596	0.99542
	ACS	0.97732	0.99612	0.99574
	SS	0.93574	0.97142	0.99329
	ACS-SSA	0.97976	0.99956	0.99612
4	CS	0.96376	0.99268	0.99640
	ACS	0.97843	0.99719	0.99780
	SS	0.94021	0.98298	0.99600
	ACS-SSA	0.97902	0.99948	**0.99894**
5	CS	0.97899	0.99305	0.99689
	ACS	0.98419	0.99864	0.99919
	SS	0.95987	0.98885	0.99641
	ACS-SSA	0.97912	0.99928	**0.99928**

Bold values indicate best results.

all the information with greater efficiency and a higher GM directly indicates a high FSIM value.

Table 10 shows the average contrast values of all the algorithms using all soft computing techniques discussed in this paper. It is seen that, out of all the algorithms, performance of 2D Otsu is least satisfactory and ERCE method achieves the highest result. Similarly, out of all CS, ACS, SS and ACS-SSA, our proposed technique gives the best value. It is also to be noted that these values gradually increase with the increase in threshold levels.

Table 10
Contrast values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	1.0534e + 03	1.2871e + 03	1.7710e + 03
	ACS	1.1078e + 03	1.6045e + 03	**1.7854e + 03**
	SS	1.0361e + 03	1.2333e + 03	1.7640e + 03
	ACS-SSA	1.2502e + 03	1.7423e + 03	1.7709e + 03
3	CS	1.2988e + 03	1.6279e + 03	1.7513e + 03
	ACS	1.4542e + 03	1.7489e + 03	1.7596e + 03
	SS	1.2162e + 03	1.3977e + 03	1.7506e + 03
	ACS-SSA	1.4642e + 03	1.7913e + 03	**1.7710e + 03**
4	CS	1.5050e + 03	1.7180e + 03	1.7759e + 03
	ACS	1.5632e + 03	1.8276e + 03	1.7531e + 03
	SS	1.4995e + 03	1.7199e + 03	1.7683e + 03
	ACS-SSA	1.5460e + 03	1.7807e + 03	**1.7831e + 03**
5	CS	1.4999e + 03	1.7681e + 03	1.4886e + 03
	ACS	1.5699e + 03	1.7853e + 03	1.7897e + 03
	SS	1.5215e + 03	1.7454e + 03	1.7699e + 03
	ACS-SSA	1.5703e + 03	1.7999e + 03	**1.7939e + 03**

Bold values indicate best results.

A good segmentation technique demands for the preservation and enhancement of the brightness information. Considering ERCE optimized by ACS-SSA at threshold level 4, the results that we get are greater than EGGA and 2-D Otsu method. Even when we consider only ERCE, the performance is better when ACS-SSA is used as compared to CS, ACS, and SSA, respectively.

Table 11 gives another contrast enhancement metric called Inverse Difference Moment (IDM). Its value is the inverse of the contrast values that have been discussed earlier. Hence, a lesser IDM value denotes a better segmented image. Interestingly, the IDM values attained for ERCE are consistently high. Table 12 gives the comparison result of average Peak to Signal Ratio (PSNR) values. PSNR is a very simple, efficient and widely used performance evaluation metric. The PSNR of a thresholded image can be calculated by the formula:

$$PSNR(m,n) = 10log_{10}(255^2/MSE(m,n)) \tag{22}$$

where, 'm' is the reference image and 'n' is the test image.

Due to the reason that PSNR is a ratio of signal to noise, a higher value PSNR indicates greater information and lesser noise in the thresholded image. In this Table 12, it is clearly visible that ERCE outperforms rest of the methods employed. It is evident that with the increase in threshold levels, we get better results. The incorporation of ACS-SSA in order to maximize the fitness function gives the best results. Upon rigorous analysis, it is seen that the average PSNR value for

Table 11
IDM values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	0.04838	0.03986	0.03161
	ACS	0.04581	0.03494	0.03177
	SS	0.04595	0.03989	0.03256
	ACS-SSA	0.04220	0.03246	**0.03141**
3	CS	0.04431	0.03898	0.03161
	ACS	0.04281	0.03694	0.03173
	SS	0.04448	0.03940	0.03188
	ACS-SSA	0.03998	0.03202	**0.03134**
4	CS	0.04371	0.03777	0.03178
	ACS	0.04201	0.03215	0.03284
	SS	0.04387	0.03832	0.03176
	ACS-SSA	0.03862	0.03307	**0.03111**
5	CS	0.46174	0.03635	0.04002
	ACS	0.42203	0.03519	0.03225
	SS	0.43114	0.36698	0.03264
	ACS-SSA	0.38306	0.03254	**0.03172**

Bold values indicate best results.

Table 12
PSNR values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	61.3948	71.2345	78.4678
	ACS	68.6554	73.3968	**80.4099**
	SS	60.3691	63.2486	64.5984
	ACS-SSA	70.8092	78.3079	78.6158
3	CS	64.3366	76.2456	79.3880
	ACS	71.9392	77.6037	**81.8682**
	SS	62.1141	66.6716	66.9589
	ACS-SSA	73.2122	80.9955	79.5156
4	CS	69.3196	78.5960	80.4690
	ACS	72.3621	81.3539	80.9610
	SS	65.6963	69.3342	70.9358
	ACS-SSA	76.1065	81.1720	**81.3127**
5	CS	73.1796	79.4628	76.4351
	ACS	73.1641	80.8673	81.3857
	SS	67.1472	70.5621	72.0067
	ACS-SSA	76.6438	82.4576	**84.5077**

Bold values indicate best results.

ERCE obtained using ACS-SSA at threshold level 4 is 11% higher than that obtained from 2D Otsu methods. The high value is a clear indication of the fact that ERCE preserves maximum information than the conventional 2D Otsu method. The reason may be due to the non-consideration of the off-diagonals eliminate the sizeable amount of information in 2D Otsu's method. Similarly, comparison of the PSNR value for ERCE and EGGA shows that EGGA lags behind ERCE by almost 3%. The possible reason may be the loss of information due to the presence of high magnitude peaks. It is reiterated that the results obtained while using ERCE method is the best. The PSNR values of an image have a direct relationship with the SSIM values (Hore & Ziou, 2010). Thus, the results obtained is at par with the above statement. A good image segmentation technique tries to preserve information from the original image as much as possible.

Contrast is another feature of an image which, when preserved or enhanced to improve the quality of the resulting output image. Table 13 gives the average Absolute Mean Brightness Error (AMBE) values of the 100 slices. AMBE is an IQA which is widely used in contrast enhancement method. It gives the relation between the mean of the input image and the mean of output image. Although our paper does not focus on contrast enhancement techniques, the evaluation of AMBE values can help us gauge the amount of brightness that gets preserved in the entire process. It is calculated using the formula:

$$AMBE(I,J) = \frac{1}{1 + |E(I) - E(J)|} \tag{23}$$

where, *E(I)* is the mean brightness of the input image, *E(J)* gives the mean brightness of the output image. Thus, lesser is the value of the absolute difference, the smaller is the brightness loss. This in turn demands higher AMBE value. Hence, a high AMBE value indicates a better quality of the output image. It is clearly observed that we get better AMBE values on increasing the number of threshold levels. Our proposed optimization technique ACS-SSA, when applied, out beats rest other combinations for most of the times. The results obtained from the implementation of ACS-SSA at threshold level 4 with ERCE as the histogram construction method is around 5% better than the results obtained using the 2D Otsu method. This clearly indicates that picking up only the diagonal elements does a damage to the brightness information of the image. Similarly, it is found that results for ERCE is better than EGGA. This means that the normalization of the histogram peaks preserves better brightness than EGGA. Another important area to be considered while IQA is related particularly to edge enhancement. An important metric regarding this characteristic of an image is Edge Based Contrast Measure (EBCM). This is an efficient parameter with high sensitivity to edges (Beghdadi & Le Negrate, 1989). Another interesting characteristic of this objective metric is that it is based on human perception mechanism (Sumaiya & Kumari, 2014).

Table 14 gives the average EBCM values of the output images taken over 100 slices from Harvard Medical School database. EBCM is calculated using the formula:

$$EBCM = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{c(i,j)}{MN} \tag{24}$$

where $c(i,j)$ is the contrast of the image and $M \times N$ is the size of the image.

It is observed that the quality of results gets better with the increase in the threshold levels, and thus, the edges get preserved. However, it must be noted that there is also a limit to the threshold levels, because a very high number of threshold levels may result in an over segmented image (where the non-edges may also appear as edges). Considering the values at threshold level 4, we can see that ACS-SSA gives the best results as compared to its counterparts. It is seen that these results are consistently best for ERCE. Upon an in-depth analysis, it is seen that the results obtained for ERCE at threshold level 4 using ACS-SSA is greater than 2-D Otsu and EGGA. Thus, it is concluded that edge information is efficiently preserved using our proposed technique.

The normalized contrast measure is another effective parameter to

Table 13
AMBE values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	0.15494	0.17854	0.16531
	ACS	0.17756	0.19636	0.19711
	SS	0.15351	0.17140	0.17742
	ACS-SSA	0.19354	0.20003	**0.20047**
3	CS	0.16415	0.19361	0.19826
	ACS	0.18788	0.19869	0.19935
	SS	0.16567	0.18118	0.19256
	ACS-SSA	0.19635	0.20051	**0.20138**
4	CS	0.16842	0.19369	0.19374
	ACS	0.19850	0.19983	0.20101
	SS	0.16948	0.19014	0.19985
	ACS-SSA	0.20082	0.20088	**0.20152**
5	CS	0.19081	0.20140	0.19756
	ACS	0.19959	0.20146	0.20197
	SS	0.17563	0.20078	0.19881
	ACS-SSA	0.20099	0.20570	**0.20211**

Bold values indicate best results.

Table 14
EBCM values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	0.15494	0.17854	0.16531
	ACS	0.17756	0.19636	0.19711
	SS	0.15351	0.17140	0.17742
	ACS-SSA	0.19354	0.20003	**0.20047**
3	CS	0.16415	0.19361	0.19826
	ACS	0.18788	0.19869	0.19935
	SS	0.16567	0.18118	0.19256
	ACS-SSA	0.19635	0.20051	**0.20138**
4	CS	0.16842	0.19369	0.19374
	ACS	0.19850	0.19983	0.20101
	SS	0.16948	0.19014	0.19985
	ACS-SSA	0.20082	0.20088	**0.20152**
5	CS	0.19081	0.20140	0.19756
	ACS	0.19959	0.20146	0.20197
	SS	0.17563	0.20078	0.19881
	ACS-SSA	0.20099	0.20570	**0.20211**

Bold values indicate best results.

measure the contrast enhancement in an image. We have tested this technique using this metric. Interestingly, this metric is a function of EBCM and is given as:

$$CM_n = \frac{1}{1+\frac{1-EBCM(J)}{1-EBCM(I)}} \tag{25}$$

where *EBCM (J)* gives the EBCM value of output image and *EBCM (I)* is the EBCM value of the input image. Thus, the greater is the value ofCM_n, the better is the resulting image. The table shows that the best values are obtained for ERCE. Taking ERCE method for construction of histogram, different optimization technique gives different values and it is distinct that our proposed soft computing technique dominates over other methods. Apart from these, it can also be analyzed that CM_nvalues (see Table 15).

increase with increasing the number of threshold levels. Taking the average of the results obtained from ERCE method using ACS-SSA at threshold levels 1, 2, 3 and 4, it is found that our proposed method is better than both EGGA and 2-D Otsu method. This result is justified since CM_n is a function of EBCM and the EBCM values for our proposed method are the best. Thus, the resulting image quality is greatly enhanced with greater human perceptivity.

Table 16 gives the comparison results of normalized discrete entropy (DE_n) of the segmented image. The authors have considered evaluating this parameter since it is a standard IQA for determining the contrast enhancement of an image. For instance, if we take I as the input image, J as the output image and DE as their discrete entropies then $DE_n$is calculated as:

$$DE_n = \frac{1}{1+\frac{log256-DE(J)}{log256-DE(I)}} \tag{26}$$

Thus, a higher DE_n value signals better entropy of the output image and hence better segmentation. From the Table 16, it is found that the DE_nvalues are higher for ERCE for most of the iterations. Greater DE_nvalue means greater details are preserved. This serves our purpose of segmentation where we want to reflect the hidden details in the segmented image. Again, here, as the number of threshold levels increases, the metric value also increases. When the fitness function is optimized for ERCE using ACS-SSA, satisfactory results are obtained. Greater information is preserved, because normalization of the histograms does not allow the system in getting away with high magnitude peaks. However, it is also seen that in certain cases, the performance of EGGA is better than ERCE. Ironically, the possible reason may be the normalization method in NLV. Possibly in ERCE for those incidences where the peak value is low, normalization of the histogram may cause a greater loss to the image than any significant advantage.

Table 15
CM_n values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	0.48560	0.49006	0.48868
	ACS	0.49238	0.49816	0.49845
	SS	0.48554	0.48967	0.48991
	ACS-SSA	0.49729	0.49931	**0.49944**
3	CS	0.48834	0.49731	0.49875
	ACS	0.49554	0.49889	0.49910
	SS	0.48825	0.49003	0.48954
	ACS-SSA	0.49816	0.49946	**0.49973**
4	CS	0.48962	0.49733	0.49735
	ACS	0.49883	0.49924	0.49916
	SS	0.49196	0.49519	0.49614
	ACS-SSA	0.49955	0.49957	**0.49977**
5	CS	0.49625	0.49974	0.49854
	ACS	0.49917	0.49976	0.49992
	SS	0.49335	0.49688	0.49660
	ACS-SSA	0.49969	0.49979	**0.49999**

Bold values indicate best results.

Table 16
DE_n values (average of 100 slices).

M	Optimization technique	Algorithm		
		2-D Otsu	EGGA	ERCE
2	CS	0.24764	0.28566	0.29075
	ACS	0.32385	0.45061	0.45341
	SS	0.30074	0.35964	0.36722
	ACS-SSA	0.41188	0.47353	**0.47319**
3	CS	0.28167	0.41405	0.42405
	ACS	0.37909	0.46549	0.46698
	SS	0.31220	0.40571	0.40417
	ACS-SSA	0.42818	0.47373	**0.47475**
4	CS	0.28790	0.42232	0.42880
	ACS	0.43929	0.47037	0.48045
	SS	0.35661	0.41038	0.42085
	ACS-SSA	0.47487	0.47583	**0.48699**
5	CS	0.39721	0.43439	0.43493
	ACS	0.46208	0.48372	0.48366
	SS	0.39323	0.42667	0.43230
	ACS-SSA	0.48275	0.48521	**0.49381**

Bold values indicate best results.

Box plots have also been implemented in this paper. These are given in Figs. 11 and 12. These are an interesting and easy method to monitor the performance of any method and also compare it with other methods. Fig. 11 (a) gives the box plot of average SSIM values for 2-D Otsu method optimized with CS, ACS, SS and ACS-SSA. The performance of ACS-SSA is the best both visually and experimentally. Similarly, Fig. 11 (b) and (c) are the box plots for EGGA and ERCE respectively. Fig. 12 is the box plot for PSNR values where Fig. 12 (a), (b) and (c) are for 2-D Otsu, EGGA and ERCE, respectively. It can be clearly observed that for most of the cases ERCE is the best among its counterparts. Its characteristic feature of manually normalizing the histogram may be the possible reason. The performance of EGGA is also not far behind. However, ERCE fine tunes the available values from EGGA. Moreover, the introduction of a new condition for abandon probability makes our proposed hybrid ACS-SSA out beat other soft computing techniques for most of the times. However, for certain cases, ACS performs better than our proposed hybrid. The possible reason may be that ACS is already a powerful optimization tool and its introduction of the new condition may not be always necessary.

The brain MR image segmentation at threshold levels 1, 2, 3 and 4 for a single slice (slice #074) computed using hybrid ACS-SSA optimization technique, and having a fitness function obtained from the ERCE histogram construction technique, are given in Fig. 13. The image obtained at threshold level 1 does not carry much information. Here, the edges are not distinct and it could be possibly eliminated for any further analysis. This is a gap area of bi-level thresholding. The complexities of the brain cannot be accurately reflected using a single threshold level. For this reason, multilevel thresholding operations of the brain MR image are performed. It is to be noted from the figures that the quality of visual perceptivity of these images increases with the increase in threshold levels. The image which is segmented at threshold level 4 clearly distinguishes various edges and visually seems very informative. Interestingly, we see that the quality of visual perception of the thresholded image and their performance evaluation using several metrics goes hand in hand. In the former, image segmented at threshold level 4 gives the best visual perception. Later, the best results for the performance evaluation metrics are obtained (at the same threshold level 4).

6. Conclusion

A new method of constructing the 2-D histogram is investigated in this paper. The normalization of the local variance information preserves the high frequency content of an image, because the magnitude limiting problem is eliminated. The contributions of this paper are

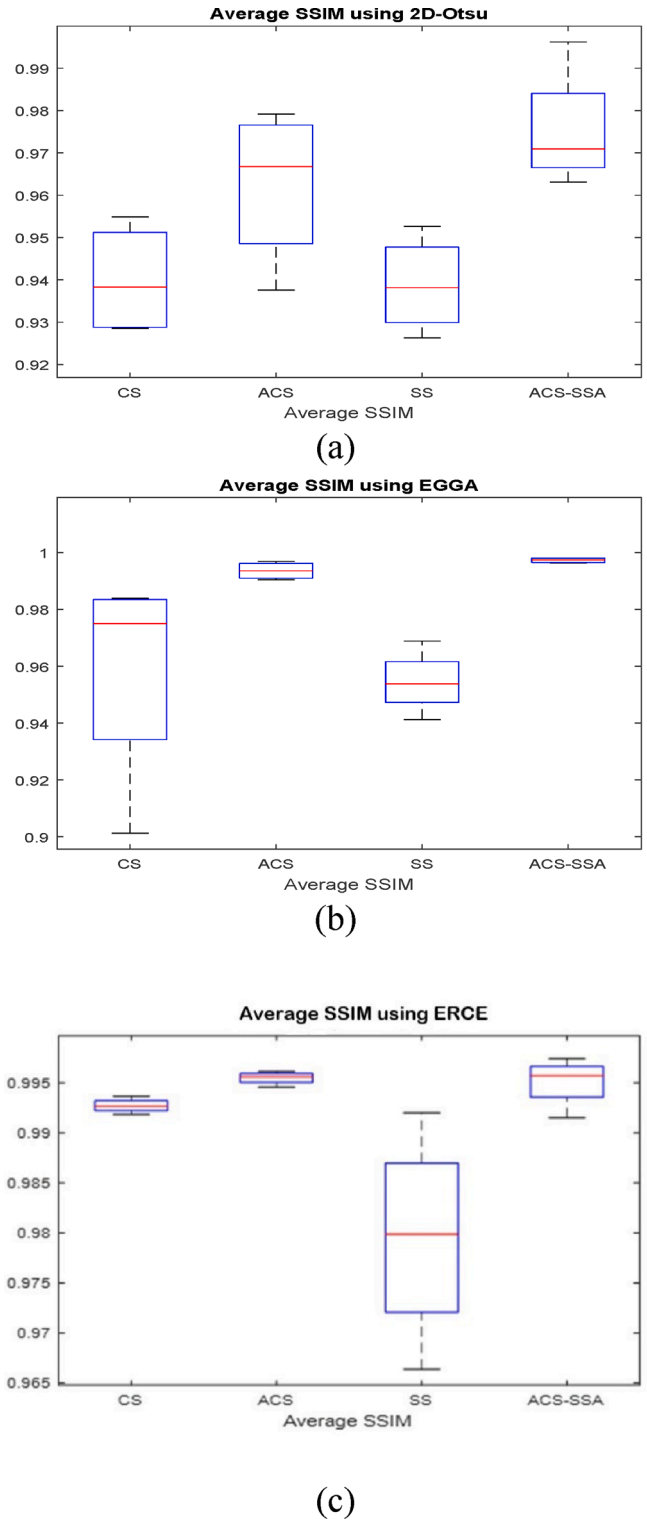

Fig. 11. Box-plot representation of average SSIM value for (a) 2-D Otsu (b) EGGA (c) ERCE.

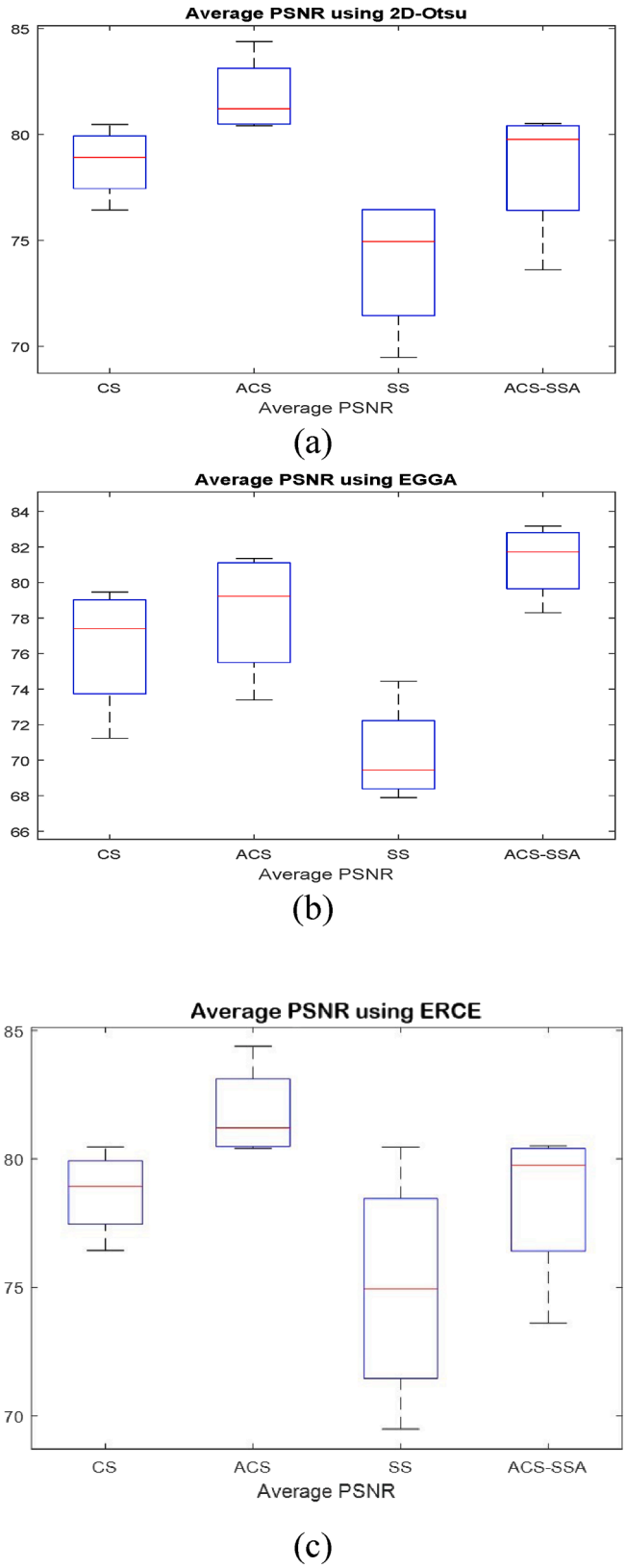

Fig. 12. Box plot of average PSNR values for (a) 2-D Otsu (b) EGGA (c) ERCE.

manifold – 1) construction of 2-D histogram using ERCE method, 2) a firsthand fitness function, 3) investigation of a new hybrid ACS-SSA method for optimization, 4) an entropic method for multilevel thresholding using 2-D row class Entropy, 5) its application to brain MR image thresholding. The incorporation of a flying coefficient together with a new condition for the evaluation of the abandon probability in ACS technique effectively enhances the exploration and exploitation of the newly proposed algorithm (hybrid ACS-SSA). Our suggested hybrid ACS-SSA algorithm performs better than the others – CS, ACS and SSA, which is implicit from the best, mean and standard deviation values shown in Tables 5-7. Moreover, it exhibits a better convergence rate. The resulting threshold values could effectually segment the brain MR image

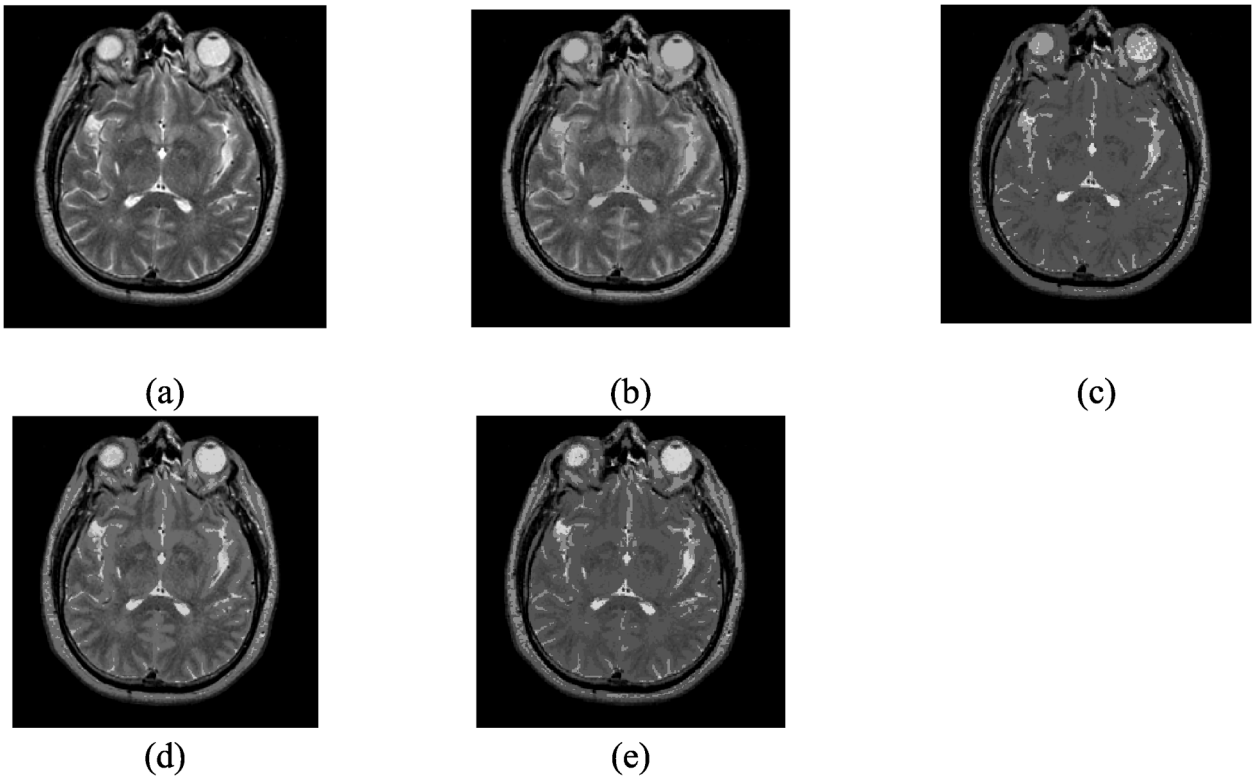

Fig. 13. Segmented brain MR images after multilevel thresholding (a) original image (slice#074) (b) image thresholded at level-1 [143, 176] (c) image thresholded at level-2 [82, 151, 221] (d) image thresholded at level-3 [107, 155, 183, 210] (e) image thresholded at level-4 [73, 84, 137, 200, 213].

into several components. In addition, it preserves the edge information well suited for clinical diagnosis. Various performance evaluation metrics are computed which shows the competency of our proposed method. Finally, it is concluded that the proposed method fare well in all aspects and may significantly contribute to the field of brain MR image segmentation using multilevel thresholding. The new fitness function proposed may be useful for many more image processing applications like color image segmentation, registration etc.

CRediT authorship contribution statement

Rutuparna Panda: Conceptualization, Formal analysis, Methodology. **Leena Samantaray:** Methodology, Investigation. **Akankshya Das:** Software, Validation, Writing - original draft. **Sanjay Agrawal:** . **Ajith Abraham:** Supervision.

Declaration of Competing Interest

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

References

Agrawal, S., Panda, R., Samantaray, L., & Abraham, A. (2017). A novel automated absolute intensity difference based technique for optimal MR brain image thresholding. Journal of King Saud University-Computer and Information Sciences.

Beghdadi, A., & Le Negrate, A. (1989). Contrast enhancement technique based on local detection of edges. *Computer Vision Graphics Image Processing, 46*(2), 162–174.

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

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

Chatterjee, A., Siarry, P., Nakib, A., & Blanc, R. (2012). An improved biogeography based optimization approach for segmentation of human head CT-scan images employing fuzzy entropy. *Engineering Applications of Artificial Intelligence, 25*(8), 1698–1709.

Despotović, I., Goossens, B., & Philips, W. (2015). MRI segmentation of the human brain: challenges, methods, and applications. Computational and Mathematical Methods in Medicine.

Dora, L., Agrawal, S., Panda, R., & Abraham, A. (2017). State-of-the-art methods for brain tissue segmentation: A review. *IEEE Reviews in Biomedical Engineering, 10*, 235–249.

Feng, Y., Zhao, H., Li, X., Zhang, X., & Li, H. (2017). A multi-scale 3D Otsu thresholding algorithm for medical image segmentation. *Digital Signal Processing, 60*, 186–199.

García, S., Molina, D., Lozano, M., & Herrera, F. (2009). A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization. *Journal of Heuristics, 15*(6), 617.

Hadjidimitriou, S., Charisis, V., Sergiadis, G., Hadjileontiadis, L., Kouloulias, V., Papathanasiou, M., … Kelekis, D. (2014). Fast histogram-based brain segmentation from T1-weighted MR images using morphological processing and geometric criteria. *Physica Medica, 30*, 88–95.

Hore, A., & Ziou, D. (2010). Image quality metrics: PSNR vs. In *SSIM. 20th International Conference on Pattern Recog* (pp. 2366–2369).

Jain, M., Singh, V., & Rani, A. (2019). A novel nature-inspired algorithm for optimization: Squirrel search algorithm. *Swarm and Evolutionary Computation, 44*, 148–175.

Kaur, T., Saini, B. S., & Gupta, S. (2018). A joint intensity and edge magnitude-based multilevel thresholding algorithm for the automatic segmentation of pathological MR brain images. *Neural Computing and Application, 30*(4), 1317–1340.

Kotte, S., Pullakura, R. K., & Injeti, S. K. (2018). Optimal multilevel thresholding selection for brain MRI image segmentation based on adaptive wind driven optimization. *Measurement, 130*, 340–361.

Kuo, T. Y., Su, P. C., & Tsai, C. M. (2016). Improved visual information fidelity based on sensitivity characteristics of digital images. *Journal of Vision Communication and Image Representation, 40*, 76–84.

Liang, H., Jia, H., Xing, Z., Ma, J., & Peng, X. (2019). Modified grasshopper algorithm-based multilevel thresholding for color image segmentation. *IEEE Access, 7*, 11258–11295.

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

Manikandan, S., Ramar, K., Iruthayarajan, M. W., & Srinivasagan, K. G. (2014). Multilevel thresholding for segmentation of medical brain images using real coded genetic algorithm. *Measurement, 47*, 558–568.

Mittal, H., & Saraswat, M. (2018). An optimum multi-level image thresholding segmentation using non-local means 2D histogram and exponential Kbest

gravitational search algorithm. *Engineering Applications of Artificial Intelligence, 71*, 226–235.
Naik, M. K., & Panda, R. (2016). A novel adaptive cuckoo search algorithm for intrinsic discriminant analysis based face recognition. *Applied Soft Computing, 38*, 661–675.
Nakib, A., Oulhadj, H., & Siarry, P. (2010). Image thresholding based on Pareto multiobjective optimization. *Engineering Applications of Artificial Intelligence, 23*(3), 313–320.
Oliva, D., Hinojosa, S., Cuevas, E., Pajares, G., Avalos, O., & Gálvez, J. (2017). Cross entropy based thresholding for magnetic resonance brain images using Crow Search Algorithm. *Expert System with Application, 79*, 164–180.
Panda, R., Agrawal, S., Samantaray, L., & Abraham, A. (2017). An evolutionary gray gradient algorithm for multilevel thresholding of brain MR images using soft computing techniques. *Applied Soft Computing, 50*, 94–108.
Priyedarsni, P., Nandhini, B., Catherine, A.R., Sahana, K., & Sundaravadivu, K. (2017). September. Soft-computing assisted tool to extract tumor section from brain MR images. In IEEE. Int. Conf. Power, Control, Signal Instrument Eng. (ICPCSI) (pp. 2776–2780).
Raja, N. S. M., Sukanya, S. A., & Nikita, Y. (2015). Improved PSO based multi-level thresholding for cancer infected breast thermal images using Otsu. *Procedia Computer Science, 48*, 524–529.
Rajinikanth, V., Satapathy, S. C., Fernandes, S. L., & Nachiappan, S. (2017). Entropy based segmentation of tumor from brain MR images–a study with teaching learning based optimization. *Pattern Recoginition Letters, 94*, 87–95.
Sathya, P. D., & Kayalvizhi, R. (2011a). Amended bacterial foraging algorithm for multilevel thresholding of magnetic resonance brain images. *Measurement, 44*(10), 1828–1848.
Sathya, P. D., & Kayalvizhi, R. (2011b). Optimal segmentation of brain MRI based on adaptive bacterial foraging algorithm. *Neurocomputing, 74*(14–15), 2299–2313.
Sezgin, M., & Sankur, B. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging, 13*(1), 146–166.
Sumaiya, M. N., & Kumari, R. S. S. (2014). Unsupervised edge enhancement algorithm for SAR images using exploitation of wavelet transform coefficients. In Int. Conf. Comm. Net. Tech. (pp. 57–61).
Suresha, H. S., & Parthasarathy, S. S. (2018). Diagnosis of alzheimer disease using fast independent component analysis and Otsu multi-level thresholding. *International Journal of Intelligent Engineering and Systems, 11*(5), 74–83.
Xu, L., Jia, H., Lang, C., Peng, X., & Sun, K. (2019). A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution. *IEEE Access, 7*, 19502–19538.
Yang, X. S., & Deb, S. (2009). Cuckoo search via Lévy flights, 2009. In World Congress on Nature & Bio. Insp. Comput. (NaBIC) (pp. 210–214).
Zhang, L., Zhang, L., Mou, X., & Zhang, D. (2011). FSIM: A feature similarity index for image quality assessment. *IEEE Transactions on Image Processing, 20*(8), 2378–2386.