A Novel Diagonal Class Entropy-Based Multilevel Image Thresholding Using Coral Reef Optimization

Sanjay Agrawal, *Member, IEEE*, Rutuparna Panda, and Ajith Abraham, Sr., *Senior Member, IEEE*

***Abstract*—In the normal image thresholding methods based on two-dimensional histogram, the edge information of the regions is not maintained because of the local averaging activity used. Moreover, the computation time increases with the increase in the level of thresholds. This paper focusses on retaining more edge information by calculating the image entropy along the diagonal regions of the gray level co-occurrence matrix inspired from the partitioned design structure matrix, which is a novel idea. In addition, the key to our success is the theoretical investigation of a novel diagonal class entropy (DCE) concept that utilizes the minimum area for computation. The benefits of the proposed method are: 1) improved results; 2) efficient to preserve more precise shape of the edges; and 3) the computation time decreases with the increase in the threshold levels. The optimal thresholds are obtained by minimizing the DCE using coral reef optimization (CRO). A first hand fitness function for multilevel image thresholding is derived. The fight for space and the efficient reproduction characteristics of the CRO makes it attractive for this application. Benchmark images from the Berkley segmentation dataset are taken to experiment. Our results are compared with other state-of-the-art thresholding methods. The results obtained are encouraging and may set the path for further investigation in the domain of multilevel thresholding.**

***Index Terms*—Coral reef optimization (CRO), design structure matrix (DSM), entropy, gray level co-occurrence matrix (GLCM), multilevel thresholding.**

I. Introduction

IMAGE segmentation is one of the vital steps in image processing. Many segmentation techniques are reported in the literature. Thresholding is one of the most normally used method. Thresholding is performed by dividing the pixels based on gray level information using different approaches. Sankur and Sezgin [1] classified the thresholding techniques into six categories depending on the information utilized. Histogram-based approaches utilize the curvature of the image histogram for determining the thresholds. Clustering-based approach groups the image into clusters, as background (BG) and foreground (FG) and then determines the threshold values. Object attributes-based approaches use the fuzzy similarity for thresholding.

Entropy-based approaches use the entropy of BG–FG regions for determining the thresholds. Spatial correlation approaches use the correlation between pixels for thresholding. In this paper, we propose a novel approach for image thresholding-based on both the entropy and spatial correlation.

Images with fewer details are thresholded using bi-level thresholding. However, images with larger details require multilevel thresholding. The problem of multilevel thresholding becomes complex when the number of threshold value increases. For any gray image, the number of gray levels varies from 1 to 256 (assuming 8-bit encoding). As the level of threshold increases, the search space dimension also increases in traditional methods. This warrants for a more appropriate technique to solve such problems. Now, evolutionary computing techniques have made their presence felt by solving such complex problems, especially when the search space dimension is very large. In this context, several thresholding techniques are proposed for multilevel thresholding, which uses different objective functions and different evolutionary computing techniques. Some of them are genetic algorithm [2], particle swarm optimization (PSO) [3], [4], ant colony optimization [5], harmony search [6], simulated annealing (SA) [7], and differential evolution (DE) [8] to name a few.

In this paper, we have experimented with one such latest evolutionary computing technique named coral reef optimization (CRO), which is based on the imitation of coral reproduction and coral reef formation [9]. The motivation behind the selection of the CRO for this application is its fight for space and efficient reproduction characteristics. Further, the application of CRO to multilevel thresholding problem is new. The CRO in its basic version is inspired by SA. It has mainly focused on the exploitation part of evolutionary algorithms. The main focus of this paper is to introduce a new objective function for multilevel thresholding. Researchers can freely choose any proven optimization technique for further investigation into the proposed objective function. Here, a new objective function is formulated to obtain optimal multiple threshold values. The objective function is derived from DCE. The DCE feature is extracted from the gray level co-occurrence matrix (GLCM) as a feature space of the image. The local transition entropies of BG and FG computed in local

Manuscript received March 22, 2018; revised May 21, 2018; accepted July 17, 2018. This paper was recommended by Associate Editor M. Celenk. *(Corresponding author: Rutuparna Panda.)*

S. Agrawal and R. Panda are with the Department of Electronics and Telecommunication Engineering, VSS University of Technology, Burla 768018, India (e-mail: agrawals_72@yahoo.com; r_ppanda@yahoo.co.in).

A. Abraham is with Machine Intelligence Research Laboratories, Washington, DC 98071-2259 USA (e-mail: ajith.abraham@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2859429

quadrants of the GLCM are summed up to give DCE. The optimum thresholds are obtained when DCE is minimized. For choosing the regions of GLCM to compute DCE, we use the concept of partitioned DSM. To the best of our knowledge, this application of DSM to image processing is new. The proposed method is validated using benchmark images from Berkley Segmentation Dataset. The results are compared with other state-of-the-art thresholding methods. It is observed that the computation time decreases with increase in levels of thresholds.

The rest of this paper is organized as follows. Section II presents the related work. Section III describes the proposed method. Section IV presents the results and discussions. Finally, Section V draws the concluding remarks of this paper.

II. Related Work

The research in the area of multilevel thresholding is still ongoing. We can also observe the application of different variations of evolutionary computation techniques for segmentation [10]. Roy *et al.* [11] presented a comparative analysis of multilevel thresholding schemes based on cuckoo search optimization technique. The authors, in their work, focused on the minimum cross-entropy criterion for thresholding. Raja *et al.* [12] implemented an improved PSO-based multilevel thresholding technique. The authors used Otsu's function for breast thermal images infected with cancer. They used the one-dimensional (1-D) histogram of the image and obtained the optimum thresholds by maximizing Otsu's criteria of between class variance.

The concept of entropy was explained in [13]–[15]. Brajevic and Tuba [16] applied CS and FF algorithms for the multilevel image thresholding problem. They used two different fitness functions, Kapur's maximum entropy thresholding criteria and multi Otsu between-class variance criteria for optimization. Kurban *et al.* [17] used Kapur's entropy to present a comparative study on evolutionary and swarm-based methods for multilevel color image thresholding. Liu *et al.* [18] proposed a modified PSO for multilevel thresholding by maximizing the Otsu's objective function.

Li *et al.* [19], [20] used dynamic-context cooperative quantum-behaved, partitioned and cooperative quantum-behaved PSO for medical image segmentation. They used these methods to optimize the parameters of Otsu segmentation for dividing the pixels into three or four classes only. The authors insisted on applying their methods to different kinds of segmentation problems. However, they have used 1-D Otsu segmentation method for medical images. Ali *et al.* [21] proposed a multilevel image thresholding technique by synergetic DE. It employs two criteria entropy and approximation of the normalized histogram of an image by combining Gaussian distribution to find the optimal thresholds.

From the above study, it is observed that the Otsu's method used for multilevel thresholding is based on the 1-D histogram. However, the performance of the Otsu scheme based on 1-D histogram may worsen for images with complex boundaries as the spatial correlation between the neighboring pixels is not considered. Therefore, researchers have proposed Otsu's method based on two-dimensional (2-D) histogram for bi-level image thresholding [22], [23]. A 2-D histogram is constructed using gray and the mean gray values. Then the maximum between class variance of object and BG area along the diagonal is computed to obtain the threshold value. However, the method is not efficient, as the local averaging is computed. When the image has complex boundaries, threshold selection by using the 2-D histogram method suffers from these flaws: 1) the internal area along the diagonal is not homogenous; 2) the shape of edge of the image is also not clear; and 3) the transition information between the FG and the BG area which is also available in the off-diagonal region is not taken into account.

This has motivated us to investigate a new algorithm for multilevel image thresholding. Even though several fitness functions are formulated derived from the entropy [24]–[30], still there is scope for new functions to be proposed for improvising the thresholding results. An automatic thresholding technique using entropy is also implemented in [26] and [27]. Many researchers have proposed methods of thresholding based on entropy derived from GLCM. Chang *et al.* [28] investigated thresholding based on Shannon's entropy and relative entropy. They used GLCM to compute local, joint, and global entropy for thresholding. A new class of entropy named DCE is introduced in this paper. The DCE feature is extracted from the GLCM as a feature space of the image. The GLCM regions are chosen based on partitioned DSM. The idea of partitioned DSM is used for the first time in image processing application.

III. Proposed Method

A block diagram of our proposed method is displayed in Fig. 1.

An image is considered as the input. The number of thresholds are taken as input from the user. The normalized GLCM is computed. The proposed objective function based on DCE is optimized using CRO. The optimal threshold values are obtained when DCE is minimized. The program automatically generates the thresholded output image. We have implemented the Otsu's multilevel threshold formula based on 2-D histogram for a comparison. We have also compared our results with our implementations of multilevel thresholding using Kapur's entropy [31]. Further, some current segmentation work based on similar entropic definitions, one nonthresholding-based method along with visual saliency map-based segmentation is also considered for comparison. A brief explanation on CRO is presented below.

A. Coral Reef Optimization

CRO suggested by Salcedo-Sanz *et al.* [9] is inspired by modeling and simulating corals formation and reproduction. Let $\tilde{A}$ represent a reef model containing $N \times M$ square grids. Each square (x, y) of $\tilde{A}$ can allocate a coral $\hat{C}_{x,y}$ representing different solutions to a problem. Each coral can be represented as strings of numbers in a specified alphabet $\breve{O}$. Initially, some squares in $\tilde{A}$ are randomly assigned to be occupied by corals. Some squares are intentionally kept empty so that new corals can stay and develop. The ratio between free to total

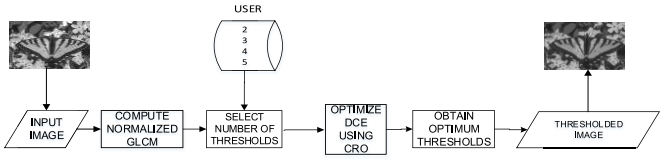

Fig. 1. Block diagram of our proposed method.

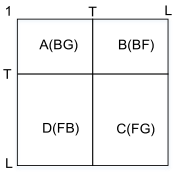

Fig. 2. GLCM quadrants for bi-level thresholding.

number of squares is an important parameter p. Usually, it is in the range [0, 1]. Each coral is associated with a fitness function $f(\hat{C}_{x,y})$: $\breve{O} \rightarrow R$ that is related to the problem objective function. The reef continues to grow as long as healthier corals exist and less healthy corals are discarded. A predetermined fraction (F_b) of existing corals is randomly selected as broadcast spawners and the remaining $(1 - F_b)$ as brooders.

In each iteration, a pair of corals from the spawners are selected (using any standard selection algorithm, e.g., Roulette wheel selection) to form larva by crossover. It is to be noted that a pair of corals can participate only once, for crossover, in each iteration. The brooders form larva using random mutation of the brooding-reproductive corals. This is used to achieve mutation for local searching. The larva formed from both the processes try to stay and develop in the reef. Each larva tries for a predetermined number of times (denoted as k) to occupy a square. When the patch is empty, the coral develops there. But if it is occupied by another coral, then the new larva fights for the space on the basis of its fitness function. If a larva fails to occupy, within the number of times, it is discarded.

The whole set of corals are sorted based on fitness values. A predetermined fraction F_a duplicates itself using budding or fragmentation and tries to stay in a different portion of the reef. At the end of iteration, with a predetermined probability (P_d), a fraction of corals is depredated thus creating space for new corals. This fraction is indicated as F_d on the basis of corals having worse fitness values. It is to be noted that any coral can be repeated for at most τ times in the reef. It is then eliminated and the space is freed. The above steps are repeated until the stopping criteria is met.

B. Diagonal Class Entropy

As we know, a single threshold value divides the GLCM into four quadrants. The four quadrants are classified as two classes: 1) local and 2) joint. The local quadrants conform to transitions in the BG or object only. The joint quadrants conform to transitions through the boundaries between BG and object or FG. The local transition entropies of BG and FG computed in local quadrants of the GLCM are summed up to give DCE. This notion is extended for multilevel thresholding. However, for choosing the regions of GLCM for multilevel thresholding, we have used here the concept of partitioned DSM. For higher levels of thresholding, e.g., from two to five, the BG and FG regions in the GLCM are not clearly distinguishable. Any selection of regions results either in overlapping or in exclusion. So only the diagonal regions of GLCM are used for computing the entropy. It is observed that with the increase in number of thresholds, the time taken to compute the optimum thresholds is reducing. This is because, the size of the GLCM is same for every level of the threshold, but the area of calculating the DCE reduces as the number of thresholds is increased.

It is known that, GLCM of an image is a square matrix, which represents distribution of co-occurring intensity values at a given offset. It is wise to reiterate that a single threshold value T divides a GLCM into four quadrants, which is further divided into two classes as shown in Fig. 2 [32].

Pixels having a gray level value greater than T correspond to FG and that less than T correspond to BG. Thus, the diagonal quadrants A and C of the matrix presents the local transitions in the BG and object or FG, respectively. On the other hand, quadrants B and D along the off-diagonal conform to joint transitions through boundaries between the BG and the object or FG [33]. Here, we focus only on the local transitions and define the diagonal class entropy (DCE) as

$$H_{\mathrm{DCE}}(T) = H_A(T) + H_C(T) \tag{1}$$

along the diagonal of the GLCM, where H denotes the entropy feature. $H_A(T)$ denotes the local transition entropy of BG and $H_C(T)$ denotes the local transition entropy of FG. Here, both the entropies are determined by threshold T, hence they are functions of T. These entropies are derived from GLCM and are defined later in this section.

Here, a partitioned DSM influences the selection of regions corresponding to the local transitions only. The GLCM is partitioned into many regions depending on the number of thresholding levels. Say, 3-level thresholding partitions the GLCM into 9 regions, 4-level thresholding partitions it into 16 regions and so on. To the best of our knowledge, there is no method available in the literature by which the GLCM regions corresponding to the local transitions can be selected. As observed in Fig. 2, for bi-level thresholding, the BG and FG area is clearly distinguishable. However, when this idea is used for multilevel thresholding, the BG and FG regions are not clearly distinguishable. Any selection of regions results either in overlapping or in exclusion. Hence, we suggest to choose the diagonal regions of the GLCM conforming to the local transitions, which is inspired by a partitioned DSM [34], [35].

A DSM is a square matrix, providing a compact representation of the relationships between the components of a system. It is also referred to as dependency structure matrix or dependency source matrix. The components are often labeled in the rows and/or in columns. Each element of the matrix represents the interaction of two components of the system. The off-diagonal elements are used to indicate relationships between the components. The elements along the diagonal are typically used to represent the relationship between the same components.

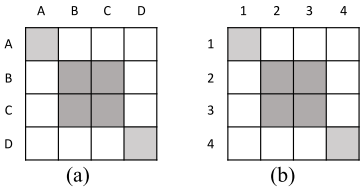

Fig. 3. DSM versus GLCM. (a) Partitioned DSM. (b) 4×4 GLCM for three-level thresholding.

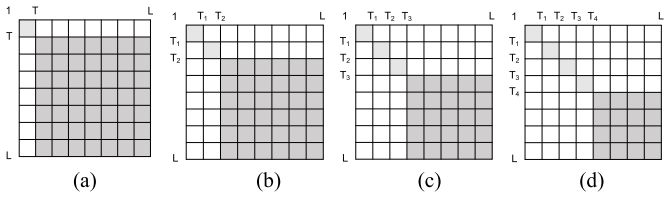

Fig. 4. GLCM regions based on partitioned DSM used for computing DCE for multilevel thresholding. (a) Bi-level thresholding. (b) Three-level thresholding. (c) Four-level thresholding. (d) Five-level thresholding.

The objective of DSM is to explore various complex system configurations with user-defined measures and requirements, to quantitatively determine, which components and/or system should be modularized to improve the total system performance. An important criterion for evaluating the matrix is that the components are permuted and reordered so that the matrix becomes lower triangular. The mutual dependency between different components is explored and additional cycles of dependency if any, are eliminated. Algorithms that have been developed to optimize the ordering of components and their combination into groups are called partition algorithms. When the partition algorithm is applied to a DSM, it is rearranged so that all dependencies fall either below the diagonal or within groups along the diagonal. The DSM is then said to be in block triangular form. Fig. 3(a) displays a partitioned block triangular DSM. This idea of partitioned DSM is used here. The cells of GLCM along the diagonal are grouped to compute the entropy for multilevel thresholding.

A GLCM for three level thresholding is shown in Fig. 3(b), where the cells along the diagonal are grouped into three regions. It is interesting to note that, to the best of our knowledge, DSM has not been used in image processing applications. This will certainly set a path for further research in this area. It is observed from Fig. 3 that the structure and characteristics of a GLCM are very much similar to DSM. A GLCM is also a square matrix representing the frequency of occurrence of different combinations of pixel intensities in an image. The number of rows and columns of the matrix is equal to the number of gray levels in the image.

Fortunately, the permutation and reordering of GLCM elements are not required here, as they are already distributed properly. Each matrix element of the GLCM contains the second-order statistics, probability values for changes between gray levels i and j for a particular displacement and angle. For a given distance, four angular GLCM are defined for $\theta = 0°, 45°, 90°$, and $135°$. It is already stated in the literature that computation of GLCM for the values of $\theta = 180°, 225°, 270°$, and $315°$ adds nothing significant. Hence, the final GLCM "G" is computed as

$$G = \left[g\left(d, 0^\circ\right) + g\left(d, 45^\circ\right) + g\left(d, 90^\circ\right) + g\left(d, 135^\circ\right)\right]/4 \quad (2)$$

where $g(\bullet)$ denotes GLCM in one direction only. Next, we normalize the final GLCM as

$$G(i,j) = g(i,j)/\sum_{i=1}^{L}\sum_{j=1}^{L} g(i,j) \quad (3)$$

to prevent a negative value occurring for the entropy.

A number of features are extracted from the GLCM [32]. In this paper, we use the entropy feature computed from the GLCM. Let L be the number of gray levels in the image. Then the size of GLCM will be $L \times L$. Let $G(i,j)$ represent an element of the matrix. Then the entropy feature from the matrix is computed as $H = -\sum_{i=1}^{L}\sum_{j=1}^{L} G(i,j) \times \log(G(i,j))$.

However, for bi-level thresholding, for a threshold value T, the DCE is computed as

$$H_A = -\sum_{i=1}^{T}\sum_{j=1}^{T} G(i,j) \times \log(G(i,j)) \quad (4)$$

$$H_C = -\sum_{i=T+1}^{L}\sum_{j=T+1}^{L} G(i,j) \times \log(G(i,j)) \quad (5)$$

$$H_{\mathrm{DCE}}(T) = H_A(T) + H_C(T). \quad (6)$$

When this formulation is extended to multilevel thresholding, we consider only the diagonal regions of the GLCM for computing the DCE for each level of thresholding as shown in Fig. 4. The optimum thresholds are obtained when DCE is minimized. We introduce here the theoretical formulation for multilevel thresholding using DCE. For $(K–1)$ thresholds $[T_1, T_2, \ldots, T_{K-1}]$ the DCE is computed as

$$\begin{aligned} H_{\mathrm{DCE}}(T_1, T_2, \ldots, T_{K-1}) &= -\left\{\sum_{i=1}^{T_1}\sum_{j=1}^{T_1} G(i,j) \times \log(G(i,j)) \right. \\ &\quad + \sum_{i=T_1+1}^{T_2}\sum_{j=T_1+1}^{T_2} G(i,j) \times \log(G(i,j)) \\ &\quad \vdots \\ &\quad \left. + \sum_{i=T_{K-1}+1}^{L}\sum_{j=T_{K-1}+1}^{L} G(i,j) \times \log(G(i,j))\right\}. \end{aligned} \quad (7)$$

The proposed objective function is

$$\begin{aligned} &\left\{T_{\mathrm{opt}_1}, T_{\mathrm{opt}_2}, \ldots, T_{\mathrm{opt}_{K-1}}\right\} \\ &= \arg \min_{1<T_1<T_2<\cdots<T_{K-1}<L} \{H_{\mathrm{DCE}}(T_1, T_2, \ldots, T_{K-1})\}. \end{aligned} \quad (8)$$

Note that K is the number of classes. It is observed from Fig. 4(b) and (c) that the area, while computing the entropy, is reduced as the number of thresholds is increased from two to three. Interestingly, the area is further reduced with higher levels of threshold [which is seen from Fig. 4 (d)]. This idea is more clearly explained using the following figure.

Pseudocode 1 Pseudo Code of the Proposed DCE Algorithm

Input: A gray scale image, Levels of thresholds (2,3,4,5)
Output: Thresholded image
Begin
1: Compute normalized GLCM using Eqn. (3)
2: Initialization of CRO vital parameters
Set broadcast probability (F_b) $\rightarrow$ 0.9
Set asexual reproduction probability (1-F_b) $\rightarrow$ 0.1
Occupancy parameter (p) $\rightarrow$ 0.6
Depredation probability (P_d) $\rightarrow$ 0.1
Reef initialization (N, M, L)
Reef size $\leftarrow$N$\times$M; No. of thresholds $\leftarrow$L
3: Evaluate initial fitness values using Eqn. (8)
4: **for** i=1 to number of iterations
5: larvae formation by broadcast spawning
6: larvae formation by brooding
7: larvae setting
8: budding or fragmentation
9: depredation
10: **end**
11: Optimal threshold values
End

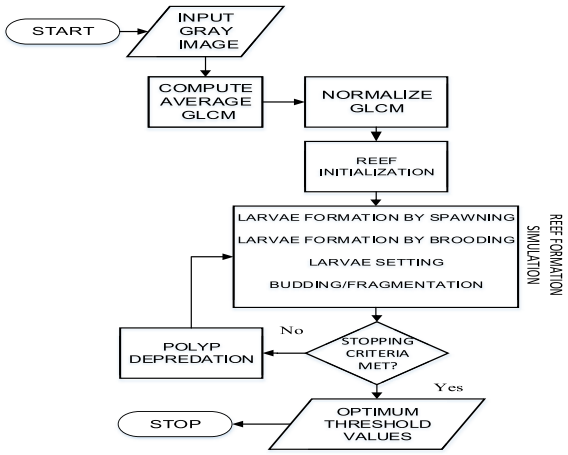

Fig. 5. Flow chart of the proposed technique.

Let the level of threshold be denoted by T. Then for 8$\times$8 GLCM, for $T = 1$, the number of GLCM elements used for computing DCE is 50, for $T = 2$, the number of GLCM elements used for computing DCE is 38, for $T = 3$, the number of GLCM elements used for computing DCE is 28, for $T = 4$, the number of GLCM elements used for computing DCE is 20 and similarly, for $T = 5$, the number of GLCM elements used for computing DCE is 14. With an increase in T, the number of GLCM elements for computing the entropy is reduced, which results in lesser number of additions and multiplications. As a result, our approach becomes very efficient while computing the entropy, as we go on increasing the threshold levels. This is another advantage of our proposed method. The algorithm of the proposed scheme is presented in Algorithm 1.

The flow chart of our proposed method is shown in Fig. 5.

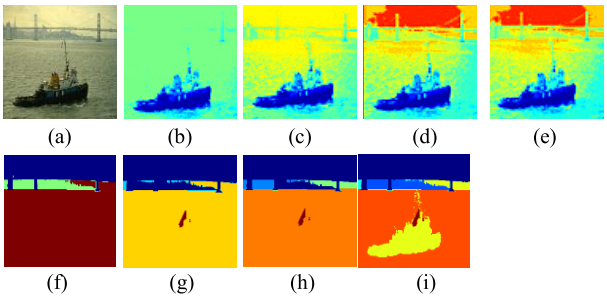

Fig. 6. Example image to experiment. (a) Original image with identification number 48 025 from BSD500 dataset. (b) Three-level thresholded image with DCE method using [125,158] threshold values. (c) Four-level thresholding using [119,144,161] threshold values. (d) Five-level thresholding using [108,145,170,215] threshold values. (e) Six-level thresholding using [102,133,156,173,215] threshold values. (f)–(i) Results of human segmentation.

We consider images from the Berkeley Segmentation dataset [36] to experiment. The 2-D histogram-based Otsu's technique is realized for obtaining optimal thresholding values utilizing (6). The thresholded image is obtained by applying the following reconstruction rule. For K optimum thresholds $[T_1, T_2, \ldots, T_K]$, pixels having intensity values less than T_1 preserve their values, pixels having gray levels between T_1 and T_2, are assigned T_1, pixels having gray levels between T_2 and T_3, are assigned T_2, similarly, pixels having gray levels between T_K and L, are assigned T_K.

IV. Results and Discussions

This section presents the results and discussions. The parameter setting for CRO is adopted using the guidelines provided in [9]. It is mentioned in the pseudo code for the algorithm. The findings are presented after 50 runs of each of the methods. We consider 30 number of iterations per independent run. We use 300 images from the Berkeley Segmentation Dataset (BSD300) [36] to experiment.

The dimension of longest side of each image is resized to 320 pixels and are converted to gray scale. The programs are implemented in MATLAB on a Core i5 processor with 4GB RAM. The results obtained using the proposed method are shown in boldface. In Fig. 6(a), the original image is displayed. Images are colored using color map *jet* in MATLAB for effective display of thresholded results.

The results are displayed for thresholding level $m = 2$, 3, 4, and 5. Fig. 6(b)–(e) represents the results obtained using our proposed method. It is observed that more details are visible with an increase in the thresholding level.

From Fig. 6, it is seen that the thresholded images obtained using our proposed method are close to the human segmentation results, when the number of thresholds is increased to five. This is due to the fact that the directional edge information is preserved in the proposed method. As observed from the images in Fig. 6, with lower levels of thresholds the different classes are overlapped. This may be due to the reconstruction rule presented under the algorithm section. The proposed objective function seems to be more effective, because it handles the local transitions in an efficient way.

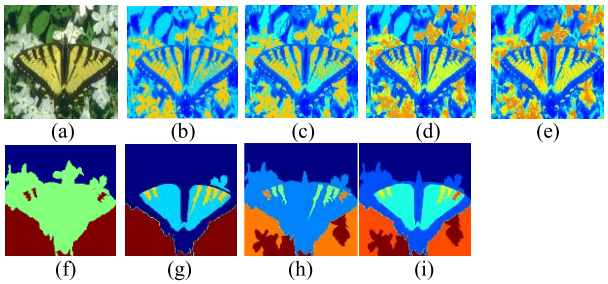

Fig. 7. Example image to experiment. (a) Original image with identification number 35 010 from BSD300 dataset. (b) Three-level thresholded image with DCE method using [86,173] threshold values. (c) Four-level thresholding using [62,99,173] threshold values. (d) Five-level thresholding using [55,89,147,186] threshold values. (e) Six-level thresholding using [53,69,100,165,191] threshold values. (f)–(i) Results of human segmentation.

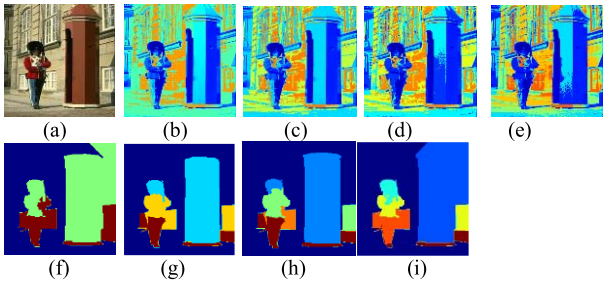

Fig. 8. Example image to experiment. (a) Original image with identification number 372 047 from BSD300 dataset. (b) Three-level thresholded image with DCE method using [115,184] threshold values. (c) Four-level thresholding using [86,143,191] threshold values. (d) Five-level thresholding using [44,93,152,189] threshold values. (e) Six-level thresholding using [39,87,129,174,200] threshold values. (f)–(i) Results of human segmentation.

Two more images 35 010 and 372 047 from BSD300 are added for more realistic performance results for image thresholding. Fig. 7(b)–(e) represents the results obtained using our proposed method. Similarly, Fig. 8(b)–(e) represents the results obtained using our proposed method.

Table I displays the average structured similarity index (SSIM) [37] and Table II shows the average feature similarity index (FSIM) [38] calculated over 300 images from BSD300 dataset. A higher value of SSIM and FSIM is desired for a better thresholding performance. It is observed from Tables I and II that the average SSIM and FSIM values are higher for our proposed technique. The reason may be the fact that the use of GLCM to compute DCE improves the correlation among the pixels. These values also increase with an increase in threshold levels. FSIM is also considered as an image quality assessment measure. It understands an image by its low level features like gradient magnitude and phase congruence. Interestingly, the gradient magnitude is better in this paper, because it is maximum along the diagonal of a GLCM. Therefore, in Table II, FSIM values are higher in our case.

Table III shows the comparison of average peak signal to noise ratio (PSNR) [39] values for all the methods. PSNR is useful for comparing images having different dynamic ranges. A high value of PSNR is desired for improved thresholding performance, which is evident for our proposed method.

TABLE I
AVERAGE SSIM VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**0.9203**	**0.9479**	**0.9701**	**0.9759**
2-D OTSU	0.8983	0.9372	0.9565	0.9692
Kapur's entropy	0.8758	0.9137	0.9325	0.9449
Saliency Map [44]	0.8467	0.8721	0.8925	0.8978
KL-MPSO [45]	0.8743	0.9005	0.9216	0.9271
Entropy based [46]	0.8283	0.8531	0.8731	0.8783

TABLE II
AVERAGE FSIM VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**0.9845**	**0.9879**	**0.9877**	**0.9906**
2-D OTSU	0.9720	0.9809	0.9849	0.9868
Kapur's entropy	0.9477	0.9564	0.9603	0.9621
Saliency Map [44]	0.9057	0.9089	0.9087	0.9114
KL-MPSO [45]	0.9353	0.9385	0.9383	0.9411
Entropy based [46]	0.8861	0.8891	0.8889	0.8915

TABLE III
AVERAGE PSNR VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**66.08**	**67.85**	**69.42**	**70.34**
2-D OTSU	64.62	66.30	67.52	68.52
Kapur's entropy	63.00	64.64	65.83	66.81
Saliency Map [44]	60.79	62.42	63.87	64.71
KL-MPSO [45]	62.78	64.46	65.95	66.82
Entropy based [46]	59.47	61.07	62.48	63.31

TABLE IV
AVERAGE CC VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**0.9370**	**0.9562**	**0.9697**	**0.9747**
2-D OTSU	0.8971	0.9400	0.9625	0.9730
Kapur's entropy	0.8747	0.9165	0.9384	0.9487
Saliency Map [44]	0.8620	0.8797	0.8921	0.8967
KL-MPSO [45]	0.8902	0.9084	0.9212	0.9260
Entropy based [46]	0.8433	0.8606	0.8727	0.8772

Possible reason may be its ability to preserve more information. Average cross correlation (CC) values are presented in Table IV. With an increase in the number of thresholds, the correlation value also increases. A high value of CC is desired, which is observed in our proposed method. The use of GLCM helps in increasing the CC values, which is inherent. It is seen that the proposed method outperforms the other methods.

We also use the segmentation indices for comparing results in Tables V–VIII. We have used the MATLAB code of the indices along with benchmark images from [40] and [41].

TABLE V
AVERAGE PRI VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**0.5197**	**0.6168**	**0.6512**	**0.6687**
2-D OTSU	0.4937	0.5860	0.6186	0.6353
Kapur's entropy	0.4814	0.5714	0.6031	0.6194
Saliency Map [44]	0.4781	0.5675	0.5991	0.6152
KL-MPSO [45]	0.4885	0.5798	0.6121	0.6286
Entropy based [46]	0.4677	0.5551	0.5861	0.6018

Probability rand index (PRI) [42] calculates the fraction of pairs of pixels whose labeling is consistent between the ground truth and the computed segmentation result. It is a measure of similarity between two images. Usually, its range is in [0,1]. The bigger is the value, the better is the thresholding result. Variation of information (VoI) [43] computes the amount of randomness in one segmented image by defining the distance between two segmentations in terms of average conditional entropy. Its range is in $[0, \infty)$. The smaller is the value, the better is the result. Global consistency error (GCE) computes the extent to which one segmentation can be observed as an improvement of the other [42]. The range is in [0, 1]. The smaller is the value, the better is the segmentation result. Boundary displacement error (BDE) computes the average error of displacement of boundary pixels between the segmented images. It is defined as the distance between the pixel and the nearest pixel in the other image boundary. A lower value of BDE is preferred for better segmentation [43].

It is observed that our proposed method performs better than the other methods. It is evident from Table V that the average PRI values are higher in our case. It is also noticed that the average PRI values increase with an increase in the number of threshold levels, which implies a good thresholding performance. A similar trend is observed in case of BDE values. The values get smaller with an increase in the threshold levels as shown in Table VI, which is expected. The average GCE and the average VoI values are smaller in our case, at a particular threshold level, as observed from Tables VII and VIII, respectively, which is desirable for better segmentation. The values of these two indices should get smaller, when the number of threshold levels increases. However, values of VoI and GCE are increasing. A probable reason for such a trend may be, the objective of multilevel thresholding technique is different from the traditional segmentation techniques. The segmentation techniques mainly focus on partitioning an image into multiple similar regions, whereas, multilevel thresholding is focussed on identifying multiple objects and boundaries.

The proposed method is compared with three more image segmentation methods [44]–[46], which includes other types of multilevel thresholding methods especially developed in recent years as well as a nonthresholding method with similar type of segmentation results. We have also compared our results with our implementations of 2-D Otsu and Kapur's entropy-based criteria. The results presented in Tables I–VIII illustrate that the performance of the proposed method is better as compared to other methods.

TABLE VI
AVERAGE BDE VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**10.1034**	**8.9887**	**8.6003**	**8.0798**
2-D OTSU	10.6086	9.4381	9.0303	8.4838
Kapur's entropy	10.8738	9.6741	9.2561	8.6959
Saliency Map [44]	10.9117	9.7078	9.2883	8.7262
KL-MPSO [45]	10.7096	9.5280	9.1163	8.5646
Entropy based [46]	11.1137	9.8876	9.4603	8.8878

TABLE VII
AVERAGE GCE VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**0.3531**	**0.5373**	**0.4900**	**0.4612**
2-D OTSU	0.3708	0.5642	0.5145	0.4843
Kapur's entropy	0.3801	0.5783	0.5274	0.4964
Saliency Map [44]	0.3813	0.5803	0.5292	0.4981
KL-MPSO [45]	0.3743	0.5695	0.5194	0.4889
Entropy based [46]	0.3884	0.5910	0.5390	0.5073

TABLE VIII
AVERAGE VOI VALUES FOR ALL THE ALGORITHMS (CALCULATED OVER 300 IMAGES FROM BSD300 DATASET)

ALGORITHM	NUMBER OF THRESHOLD LEVELS			
	2	3	4	5
DCE	**2.6518**	**3.2121**	**3.2433**	**3.3205**
2-D OTSU	2.7844	3.3727	3.4055	3.4865
Kapur's entropy	2.8540	3.4570	3.4906	3.5737
Saliency Map [44]	2.8639	3.4691	3.5028	3.5861
KL-MPSO [45]	2.8109	3.4048	3.4379	3.5197
Entropy based [46]	2.9170	3.5333	3.5676	3.6526

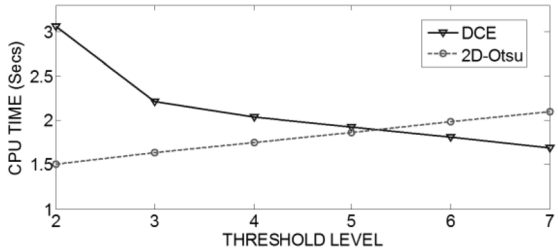

Fig. 9. CPU time comparison.

Fig. 9 shows the time required for computing the threshold values. It is seen that, in the Otsu's 2-D technique, the computation time increases with an increase in the number of thresholds, which is a general trend. From the graph, we observe that the Otsu 2-D technique is faster than our method for threshold level $m = 2, 3, 4, 5$. However, the proposed method supersedes the Otsu 2-D technique in terms of computation time for six and upper thresholding levels, an interesting trend is observed. The computation time decreases, as the number of threshold levels increase. The reason being the fact that the area of computing entropy decreases with an increase in levels of threshold (as explained in Section III). The computational complexity for multilevel thresholding using 2-D-Otsu

TABLE IX
COMPARISON BETWEEN DCE AND OTHER TECHNIQUES USING t-TESTS

Indices		2-D Otsu	Kapur's Ent.	Saliency map	KL-MPSO	Entropy based
PRI	t-value	16.64	16.65	16.47	4.47	16.24
	P-value	0.0034	0.0043	0.0023	0.0120	0.0036
GCE	t-value	-10.62	-10.61	-10.02	-3.05	-10.53
	P-value	0.0013	0.0013	0.0012	0.0271	0.0016
VOI	t-value	-18.22	-18.24	-18.19	-6.62	-17.98
	P-value	0.0026	0.0026	0.0027	0.0047	0.0022
BDE	t-value	-18.74	-18.89	-18.86	-3.89	-18.01
	P-value	0.0024	0.0024	0.0024	0.0301	0.0020

is $O\ (L^{k-1})$ where L is the number of gray levels and k is the number of classes. However, it is $O\ (L^{k-1}/k)$ for the proposed method. This is the reason why the computation time decreases when k increases. This may bring a new direction in research in multilevel thresholding.

In order to validate the proposed algorithm, statistical test (t-test) is carried out. Table IX presents the t values and the P values on the four indices of this two-tailed test with a significance level of 0.05 between the DCE and other techniques. The results indicate that the proposed method is significantly different from the other methods.

V. CONCLUSION

The traditional Otsu's method is based on 1-D histogram. Subsequently, it was also extended using the 2-D histogram for bi-level thresholding. In this paper, we have implemented the Otsu's method based on 2-D histogram. We have also experimented with our implementations of Kapur's entropy for multilevel thresholding. Further, we have proposed a new method called DCE, which uses the second-order statistics for computing the local entropy. The selection of diagonal regions for computing the entropy is inspired by the partitioned DSM. For bi-level thresholding, the local entropy feature is computed for BG and FG regions. However, for multilevel thresholding, it is difficult to identify the BG and FG regions. Hence, we use the diagonal regions of the GLCM and proposed the DCE for obtaining the thresholds. A new objective function is proposed with an automatic constraint repair mechanism for preventing the parameters from crossing the bounds. The objective function is minimized using CRO.

It is observed that the time required for obtaining optimum threshold decreases with an increase in the threshold levels. The reason being the fact that the number of elements of the GLCM considered for computing the DCE decreases. The performance parameters computed for comparison support our claim for a new multilevel thresholding technique. It has been shown that our method outperforms Otsu's method based on 2-D histogram and four other types of multilevel thresholding methods especially developed in recent years, as well as a nonthresholding method with similar type of segmentation results. As we increase the dimension to $K = 10$ or 20, the output image may result in over segmentation. The output may contain overlapped objects or overlapped boundaries. Further, we may not get proper solutions because the number of unique levels in the image may be less to find 10 or 20 thresholds. This is the limitation of the proposed method. This may set the path for new investigations in the domain of multilevel thresholding.

REFERENCES

[1] B. Sankur and M. Sezgin, "Image thresholding techniques: A survey over categories," *Pattern Recognit.*, vol. 34, no. 2, pp. 1573–1607, 2001.

[2] K. Hammouche, M. Diaf, and P. Siarry, "A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation," *Comput. Vis. Image Understand.*, vol. 109, no. 2, pp. 163–175, 2008.

[3] P.-Y. Yin, "Multilevel minimum cross entropy threshold selection based on particle swarm optimization," *Appl. Math. Comput.*, vol. 184, no. 2, pp. 503–513, 2007.

[4] M. H. Mozaffari and W.-S. Lee, "Convergent heterogeneous particle swarm optimisation algorithm for multilevel image thresholding segmentation," *IET Image Process.*, vol. 11, no. 8, pp. 605–619, Aug. 2017.

[5] Y.-C. Liang, H.-L. C. Angela, and C.-C. Chiuh, "Application of a hybrid ant colony optimization for the multilevel thresholding in image processing," in *Neural Information Processing*. Heidelberg, Germany: Springer, 2006, pp. 1183–1192.

[6] E. Cuevas, D. Zaldívar, and M. Perez-Cisneros, "Otsu and Kapur segmentation based on harmony search optimization," in *Applications of Evolutionary Computation in Image Processing and Pattern Recognition*. Cham, Switzerland: Springer, 2016, pp. 169–202.

[7] B. Karasulu and K. Serdar, "A simulated annealing-based optimal threshold determining method in edge-based segmentation of grayscale images," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2246–2259, 2011.

[8] S. Sarkar and S. Das, "Multilevel image thresholding based on 2D histogram and maximum Tsallis entropy—A differential evolution approach," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4788–4797, Dec. 2013.

[9] S. Salcedo-Sanz, J. D. Ser, I. Landa-Torres, S. Gil-López, and J. A. Portilla-Figueras, "The coral reefs optimization algorithm: A novel metaheuristic for efficiently solving optimization problems," *Sci. World J.*, Jul. 2014, Art. no. 739768.

[10] Z.-H. Zhan, Z. Jun, L. Yun, and S.-H. Chung, "Adaptive particle swarm optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 6, pp. 1362–1381, Dec. 2009.

[11] S. Roy *et al.*, "Comparative analysis of cuckoo search optimization-based multilevel image thresholding," in *Proc. Adv. Intell. Syst. Comput. (AISC)*, vol. 2, 2015, pp. 327–342.

[12] N. M. Raja, S. A. Sukanya, and Y. Nikita, "Improved PSO based multilevel thresholding for cancer infected breast thermal images using Otsu," in *Proc. Int. Conf. Comput. Commun. Converg. (ICCC)*, vol. 48, 2015, pp. 524–529.

[13] T.-H. S. Li, M.-C. Kao, and P.-H. Kuo, "Recognition system for home-service-related sign language using entropy-based K-means algorithm and ABC-based HMM," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 1, pp. 150–162, Jan. 2016.

[14] N. R. Pal and S. K. Pal, "Entropy: A new definition and its applications," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 21, no. 5, pp. 1260–1270, Sep./Oct. 1991.

[15] A. K. Wong and P. K. Sahoo, "A gray-level threshold selection method based on maximum entropy principle," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 19, no. 4, pp. 866–871, Jul./Aug. 1989.

[16] I. Brajevic and M. Tuba, "Cuckoo search and firefly algorithm applied to multilevel image thresholding," in *Cuckoo Search and Firefly Algorithm: Theory and Applications* (Studies in Computational Intelligence), X.-S. Yang, Ed. vol. 516. Cham, Switzerland: Springer Int., 2014, pp. 115–139.

[17] T. Kurban, P. Civicioglu, R. Kurban, and E. Besdok, "Comparison of evolutionary and swarm based computational techniques for multilevel color image thresholding," *Appl. Soft Comput.*, vol. 23, pp. 128–143, Oct. 2014.

[18] Y. Liu, C. Mu, W. Kou, and J. Liu, "Modified particle swarm optimization based multilevel thresholding for image segmentation," *Soft Comput.*, vol. 19, no. 5, pp. 1311–1327, May 2015.

[19] Y. Li, B. Xiaoyu, J. Licheng, and X. Yu, "Partitioned-cooperative quantum-behaved particle swarm optimization based on multilevel thresholding applied to medical image segmentation," *Appl. Soft Comput.*, vol. 56, pp. 345–356, Jul. 2017.

[20] Y. Li, J. Licheng, S. Ronghua, and S. Rustam, "Dynamic-context cooperative quantum-behaved particle swarm optimization based on multilevel thresholding applied to medical image segmentation," *Inf. Sci.*, vol. 294, pp. 408–422, Feb. 2015.

[21] M. Ali, C. W. Ahn, and M. Pant, "Multi-level image thresholding by synergetic differential evolution," *Appl. Soft Comput.*, vol. 17, pp. 1–11, Apr. 2014.
[22] H. D. Cheng, Y. H. Chen, and X. H. Jiang, "Thresholding using two-dimensional histogram and fuzzy entropy principle," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 732–735, Apr. 2000.
[23] F. Nie, W. Yonglin, P. Meisen, P. Guanghan, and Z. Pingfeng, "Two-dimensional extension of variance-based thresholding for image segmentation," *Multidimensional Syst. Signal Process.*, vol. 24, no. 3, pp. 485–501, 2013.
[24] C. Yan, N. Sang, and T. Zhang, "Local entropy-based transition region extraction and thresholding," *Pattern Recognit. Lett.*, vol. 24, no. 16, pp. 2935–2941, 2003.
[25] C.-I. Chang, D. Yingzi, J. Wang, S.-M. Guo, and P. D. Thouin, "Survey and comparative analysis of entropy and relative entropy thresholding techniques," *IEE Proc. Vis. Image Signal Process.*, vol. 153, no. 6, pp. 837–850, 2006.
[26] A. S. Abutaleb, "Automatic thresholding of gray-level pictures using two-dimensional entropy," *Comput. Vis. Graph. Image Process.*, vol. 47, no. 1, pp. 22–32, 1989.
[27] Y.-C. Liang and R.-C. Josue, "An automatic multilevel image thresholding using relative entropy and meta-heuristic algorithms," *Entropy*, vol. 15, no. 6, pp. 2181–2209, 2013.
[28] C.-I. Chang, C. Kebo, W. Jianwei, and M. L. G. Althouse, "A relative entropy-based approach to image thresholding," *Pattern Recognit.*, vol. 27, no. 9, pp. 1275–1289, 1994.
[29] D. Oliva *et al.*, "Image segmentation by minimum cross entropy using evolutionary methods," in *Soft Computing*. Heidelberg, Germany: Springer, 2017, pp. 1–20.
[30] S. J. Mousavirad and H. Ebrahimpour-Komleh, "Multilevel image thresholding using entropy of histogram and recently developed population-based metaheuristic algorithms," *Evol. Intell.*, vol. 10, nos. 1–2, pp. 45–75, 2017.
[31] A. K. Bhandari, K. S. Vineet, K. Anil, and K. S. Girish, "Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy," *Expert Syst. Appl.*, vol. 41, no. 7, pp. 3538–3560, 2014.
[32] F. Albregtsen, "Statistical texture measures computed from gray level co-occurrence matrices," Image Process. Lab., Dept. Informat., Univ. Oslo, Oslo, Norway, 1995.
[33] F. Nie, G. Chao, Y. Guo, and M. Gan, "Two-dimensional minimum local cross-entropy thresholding based on co-occurrence matrix," *Comput. Elect. Eng.*, vol. 37, no. 5, pp. 757–767, 2011.
[34] D. V. Steward, "The design structure system: A method for managing the design of complex systems," *IEEE Trans. Eng. Manag.*, vol. EM-28, no. 3, pp. 71–74, Aug. 1981.
[35] T. R. Browning, "Applying the design structure matrix to system decomposition and integration problems: A review and new directions," *IEEE Trans. Eng. Manag.*, vol. 48, no. 3, pp. 292–306, Aug. 2001.
[36] P. Arbelaez, F. Charless, and D. Martin. *The Berkeley Segmentation Dataset and Benchmark*. Accessed: Apr. 19, 2017. [Online]. Available: http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds
[37] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[38] L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.
[39] K. Hammouche, D. Moussa, and S. Patrick, "A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem," *Eng. Appl. Artif. Intell.*, vol. 23, no. 5, pp. 676–688, 2010.
[40] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jul. 2001, pp. 416–423.
[41] *Image Segmentation Benchmark Indices Package*. Accessed: Jan. 15, 2016. [Online]. Available: http://www.eecs.berkeley.edu/yang/software/lossy_segmentation/SegmentationBenchmark.zip
[42] J. Freixenet, X. Muñoz, D. Raba, J. Marti, and X. Cufí, "Yet another survey on image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 408–422.
[43] H. Gao, W. Xu, J. Sun, and Y. Tang, "Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 4, pp. 934–946, Apr. 2010.
[44] R. Achanta, F. Estrada, P. Wils, and S. Süsstrunk, "Salient region detection and segmentation," in *Proc. Comput. Vis. Syst.*, 2008, pp. 66–75.
[45] X. Zhao, M. Turk, W. Li, K.-C. Lien, and G. Wang, "A multilevel image thresholding segmentation algorithm based on two-dimensional K–L divergence and modified particle swarm optimization," *Appl. Soft Comput.*, vol. 48, pp. 151–159, Nov. 2016.
[46] A. B. Ishak, "A two-dimensional multilevel thresholding method for image segmentation," *Appl. Soft Comput.*, vol. 52, pp. 306–322, Mar. 2017.

Sanjay Agrawal (M'17) received the B.E. and M.Tech. degrees in electronics and telecommunication engineering from UCE Burla, Burla, India, in 1995 and 2003, respectively, and the Ph.D. (Engg.) degree from Sambalpur University, Sambalpur, India, in 2015.

He is currently an Associate Professor with the Department of Electronics and Telecommunication Engineering, VSS University of Technology, Burla. His current research interests include digital signal/image processing, biomedical engineering, and soft computing.

Rutuparna Panda received the B.Sc. (Engg.) and M.Sc. (Engg.) degrees in electronics and telecommunication engineering from UCE Burla, Burla, India, in 1985 and 1988, respectively, and the Ph.D. (Engg.) degree in electronics and telecommunication engineering from the Indian Institute of Technology Kharagpur, Kharagpur, India, in 1998.

He is currently a Professor with the Department of Electronics and Telecommunication Engineering, VSS University of Technology, Burla. He has over 120 papers in international/national journals and conferences. He has over 700 Google Scholar citations with an H-index of 16. His current research interests include digital signal/image processing, bioinformatics, biomedical engineering, very large scale integration signal processing, and soft computing.

Dr. Panda was a recipient of the Highly Cited Research Award from *Swarm and Evolutionary Computation Journal* (Elsevier), in 2016.

Ajith Abraham, Sr. (SM'07) received the Master of Science degree in control and automation from Nanyang Technological University, Singapore, in 1998, and the Ph.D. degree in computer science from Monash University, Melbourne, VIC, Australia, in 2001.

He is the Director of Machine Intelligence Research Laboratories, Washington, DC, USA, a Not-for-Profit Scientific Network for Innovation and Research Excellence connecting Industry and Academia. He is also a Research Professor with VSB-Technical University of Ostrava, Ostrava, Czech Republic. As an Investigator/Co-Investigator, he has won research grants worth over $100 Million from Australia, USA, EU, Italy, Czech Republic, and over 100 France, Malaysia, and China. One of his books was translated to Japanese and few other articles were translated to Russian and Chinese. He has over 1000 publications which are indexed by Scopus and over 700 are indexed by Thomson ISI Web of Science. He has authored over 700 publications originating from over 40 countries. He has over 30 000 Google Scholar citations with an *H*-index of 82. His current research interests include machine intelligence, cyber-physical systems, Internet of Things, network security, sensor networks, Web intelligence, Web services, data mining, and applied to various real world problems. He has authored/coauthored over 1300 research publications in the above areas, out of which there are over 100 books covering various aspects of computer science.

Dr. Abraham was a recipient of the Seven Best Paper Awards at prestigious international conferences held in Belgium, Canada, Bahrain, Czech Republic, China, and India for his research. Since 2008, he has been the Chair of IEEE Systems, Man, and Cybernetics Society Technical Committee on Soft Computing (which has over 200 members). He served as a Distinguished Lecturer of IEEE Computer Society representing Europe from 2011 to 2013. He is currently the Editor-in-Chief of *Engineering Applications of Artificial Intelligence*. He serves/served on the editorial board of over 15 international journals indexed by Thomson ISI. He is actively involved in the organization of several academic conferences, and some of them are now annual events. He has given over 100 plenary lectures and conference tutorials (in over 20 countries).