

Edge magnitude based multilevel thresholding using Cuckoo search technique

Rutuparna Panda ^{*}, Sanjay Agrawal, Sudipta Bhuyan

Department of Electronics and Telecommunication Engineering, VSS University of Technology, Burla 768018, India



ARTICLE INFO

Keywords:

Co-occurrence matrix
Multilevel thresholding
Cuckoo search technique

ABSTRACT

Multilevel thresholding technique is popular and extensively used in the field of image processing. In this paper, a multilevel threshold selection is proposed based on edge magnitude of an image. The gray level co-occurrence matrix (second order statistics) of the image is used for obtaining multilevel thresholds by optimizing the edge magnitude using Cuckoo search technique. New theoretical formulation for objective functions is introduced. Key to our success is to exploit the correlation among gray levels in an image for improved thresholding performance. Apart from qualitative improvements the method also provides us optimal threshold values. Results are compared with histogram (first order statistics) based between-class variance method for multilevel thresholding. It is observed that the results of our proposed method are encouraging both qualitatively and quantitatively.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Interpretation of any image requires the image to be properly segmented into meaningful regions. So multi level thresholding plays a very important role in image processing. One of the very basic, easy and efficient techniques used in image segmentation is gray level thresholding. It classifies the pixels of the image into classes based on their gray levels. But the main problem in thresholding is to choose a proper threshold value. Many types of statistical properties like maximum likelihood (Kurita, Otsu, & Abdelmalek, 1992), moment (Tsai, 1985), entropy (Kapur, Sahoo, & Wong, 1985), and between-class variance (Otsu, 1979) from image histogram have been used for choosing a threshold. Otsu (1979) considered the image histogram consisting of two Gaussian distribution representing the object and background. He then used the technique of maximizing the between-class variance for selecting a threshold. Kittler and Illingworth (1986) used the technique of selecting a threshold value that minimized the error in the Bayes sense. Pun (1980, 1981) used the maximum entropy as optimal criteria for thresholding. Sahoo, Soltani, Wong, and Chen (1988) presented a survey of thresholding techniques. Their study mainly focused on some automatic global thresholding methods using uniformity and shape measures. They stated that one of the drawbacks of point dependent thresholding technique is its dependence on first order statistics i.e. histogram of the image. But thresholding results can be improved if second order statistics i.e. co-

occurrence matrix of image can be used. Lee, Chung, and Park (1990) presented a comparative performance study of several global thresholding techniques for segmentation. They studied five global thresholding algorithms for segmentation. They observed that most algorithms are suitable for images with bimodal histograms only. They insisted for a more sophisticated and reliable technique to get a good threshold for segmentation. Sezgin and Sankur (2004) presented a survey over various thresholding techniques and their quantitative performance evaluation. They categorized the thresholding methods according to the information used such as histogram shape, entropy, object attributes, spatial correlation and local gray level surface. But they mainly focused on document image applications.

Chang, Du, Wang, Guo, and Thouin (2006) presented a survey and comparative analysis of entropy and relative entropy thresholding methods. They presented eight different entropy based information theoretic methods evaluated by shape and uniformity. But they observed that information conveyed by histogram is not sufficient to select a proper threshold value because it does not take into consideration image spatial correlation. It discards any correlation among gray levels due to which images with similar histogram may result in same threshold value. This has motivated us to use gray level co-occurrence matrix (GLCM) for selecting proper thresholds to capture transitions between gray levels. It can successfully describe and capture image correlation which is necessary to improve thresholding performance. Many studies are available in the literature where different properties of images have been derived from GLCM for obtaining an optimal threshold (Chanda & Majumder, 1988; Gonzalez & Woods, 1992; Li, Cheriet, & Ching Suen, 2005; Mokji & Abu Bakar, 2007). The co-occurrence

* Corresponding author. Tel.: +91 663 2430211; fax: +91 663 2430204.

E-mail addresses: r_ppanda@yahoo.co.in (R. Panda), agravals_72@yahoo.com (S. Agrawal), sudiptabhuyan@gmail.com (S. Bhuyan).

matrix has been used for computing many types of entropies like local, global, joint and relative to select a proper threshold value (Pal & Pal, 1989; Clausi & Jernigan, 1998; Fritz, 2008). However, the edge information in the co-occurrence matrix has not been used effectively. So we are using this method where some statistical features are derived from the GLCM based on the edge information. The features are optimized using Cuckoo search technique and results are compared with that of histogram based between-class variance method. The reason for using Otsu's method in our paper is its establishment as one of the most successful technique in image thresholding. We are using a relatively new optimization method named Cuckoo search technique. In summary, we mainly focus on a comparative study between edge magnitude based multilevel thresholding vs. histogram based multilevel thresholding scheme.

The paper is organized as follows: Section 2 explains the idea of gray level co-occurrence matrix. Section 3 describes the concept of Cuckoo search technique. Section 4 discusses the proposed technique. Results and discussions are presented in Section 5. Section 6 is the conclusion.

2. Gray level co-occurrence matrix

Gray level co-occurrence matrix (GLCM) plays an important role in image processing. The gray level co-occurrence matrix provides us an idea about how often different combinations of pixel intensity values occur in an image. Basically second order statistical features are obtained from GLCM. The frequency of appearance of one gray level in linear spatial relationship with another gray level within the same area is obtained from this matrix. Note that the co-occurrence matrix is computed based on relative distance between pixel pairs and their orientations. In GLCM, the number of rows and columns is equal to the number of gray levels. Interestingly, a matrix element $m(a,b|d,\theta)$ contains the second order statistical probability values for changes between gray levels a and b at a particular displacement d and at a particular angle θ . Let ' I ' be an image with L gray levels in the range $0 \dots L - 1$. Let $d = (a, b)$ denote an integer valued displacement vector, which specifies the relative position of the pixels at coordinates (x, y) and $(x + a, y + b)$. A gray level co-occurrence matrix M is a $L \times L$ matrix whose (i, j) element is the number of pairs of pixels of ' I ' in relative position d such that the first pixel has gray level ' i ' and the second pixel has gray level ' j '. So the matrix M involves counts of pairs of neighboring pixels. Simple relationships exist among certain pairs of the estimated probability distribution $p(d, \theta)$ as shown in Fig. 1. Note that the distance d is taken as 1 to simplify and reduce the computational complexity.

Let $p^T(d, \theta)$ denote the transpose of matrix $p(d, \theta)$.

Then $p(d, 0^\circ) = p^T(d, 180^\circ)$, $p(d, 45^\circ) = p^T(d, 225^\circ)$, $p(d, 90^\circ) = p^T(d, 270^\circ)$, $p(d, 135^\circ) = p^T(d, 315^\circ)$. Thus the computation of $p(d, 180^\circ)$, $p(d, 225^\circ)$, $p(d, 270^\circ)$, $p(d, 315^\circ)$ adds nothing significant to the GLCM. So M is formed for each of four quantized directions 0° , 45° , 90° , and 135° only. Then the final GLCM is calculated by taking their average [15]:

$$\text{GLCM} = \frac{[M(d, 0^\circ) + M(d, 45^\circ) + M(d, 90^\circ) + M(d, 135^\circ)]}{4} \quad (1)$$

Thresholding an image by a single level threshold value ' T ' divides the image co-occurrence matrix into four regions, bb , bf , ff and fb as shown in Fig. 2.

In region bb (background), the gray level transition within the background is represented. In region ff (foreground), the gray level transition within the foreground is represented. Region bf and fb represent joint transitions across boundaries between background and foreground. Thus, the method of threshold selection using

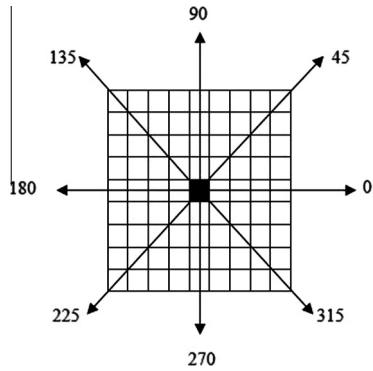


Fig. 1. Geometry for relationship among probability distribution $p(d, \theta)$.

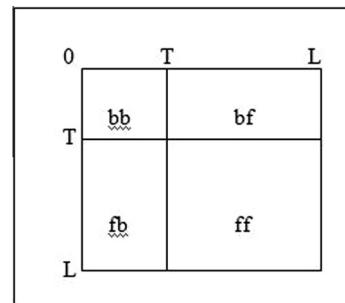


Fig. 2. Four regions of a co-occurrence matrix.

GLCM for single level thresholding can be grouped into two classes only i.e. local and joint regions. Many features can be extracted from the GLCM (Haralick, Shanmugam, & Dinstein, 1973; Haralick & Shapiro, 1992). But it has been described that six of them are more relevant (Otsu, 1979): contrast, correlation, energy, entropy, variance and inverse difference moment. All these features are computed on the basis of frequency of pixel pair. But another feature can be extracted from the GLCM on the basis of gray value difference of the pixel pair called edge magnitude 'q' (Mokji & Abu Bakar, 2007). The value of the edge magnitude is obtained from the position of the pixel pair in the GLCM. The edge magnitude increases diagonally and it is zero along the diagonal of the GLCM. Among the features given above, only the contrast computation carries the edge magnitude information. This information is then used for the thresholding process. For single level thresholding, the threshold ' T ' is calculated as (Mokji & Abu Bakar, 2007):

$$T = \frac{1}{\eta} \sum_{m=0}^{L-1-q} \sum_{n=m+q}^{L-1} \left(\frac{m+n}{2} \right) \text{GLCM}(m, n) \quad (2)$$

where

$$\eta = \sum_{m=0}^{L-1-q} \sum_{n=m+q}^{L-1} \text{GLCM}(m, n) \quad (3)$$

The summation range computes the threshold in a specific area in the GLCM which is constrained by $n - m \geq q$. In the GLCM, the pixel pairs whose edge magnitude is greater than or equal to ' q ' are involved in computation of threshold. η is defined as these total number of pixel pairs within the GLCM with edge magnitude greater than or equal to ' q '. Selection of ' q ' is crucial for improving thresholding results, because an optimum ' q ' value will get the computation area on the object's boundary. Further, the computation is restricted to upper triangle of the GLCM due to

its symmetrical features. This idea is extended here for multi-level thresholding. Equations for calculating the optimum thresholds are derived and presented in Section 4.

3. Cuckoo search algorithm

Yang and Deb (2009, 2010) developed the Cuckoo search (CS) algorithm which is a nature inspired evolutionary optimization algorithm. The algorithm is based on the parasitic breeding behavior of the Cuckoo bird. The Cuckoo bird lays its egg in the nests of other host birds. The host bird takes care of this egg presuming it as its own egg. If the host bird recognizes this egg, then it either destroys this egg or abandons the nest and builds a new nest at some new location as described in Chakraverty and Kumar (2011). The flow chart for the algorithm is presented in Fig. 3. The flow chart is explained below.

1. A set of host nests represent a generation. Each nest carries an egg as the solution.
2. The best nests with high quality eggs or solutions from each generation are carried over to the next generation.

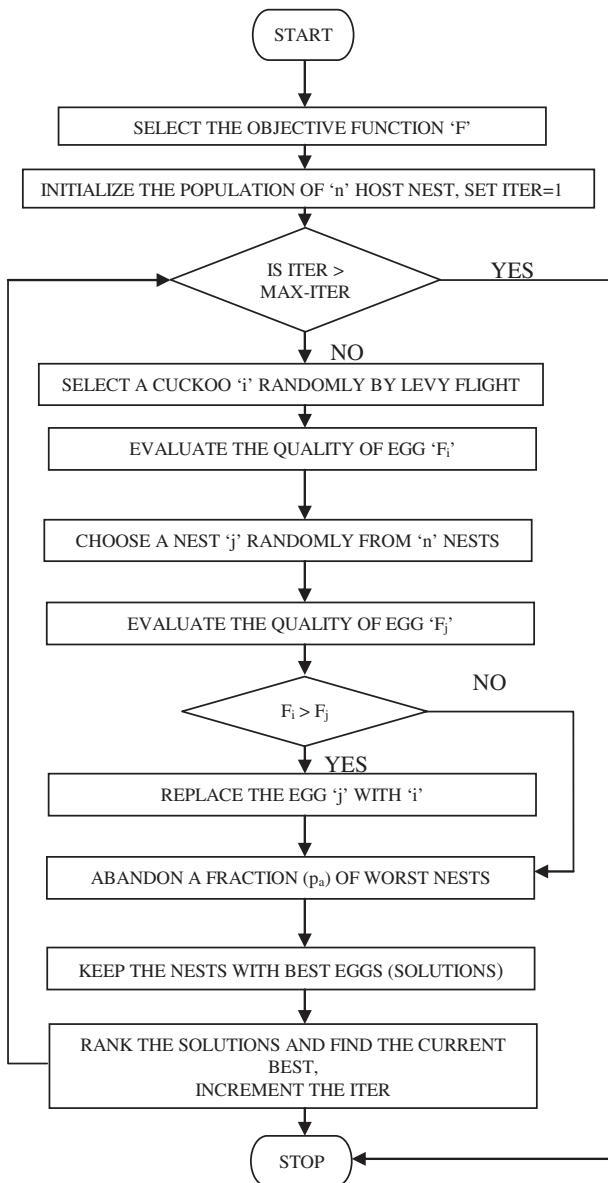


Fig. 3. Flow chart of Cuckoo search algorithm.

3. The number of host nests is fixed in each generation. An egg (solution) is chosen from a randomly selected nest. The quality of the egg is improved by generating a new egg by a random move on the selected egg. The new egg then is compared with an existing egg of another randomly chosen nest. If it is better, then the new egg replaces the existing egg.
4. Mutation probability ' p_a ' represents the probability that a Cuckoo's egg is recognized by the host bird. Such nests are discarded and removed from further calculations.

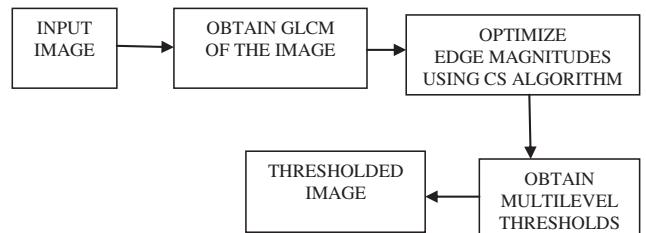


Fig. 4. Block diagram of the proposed methodology.

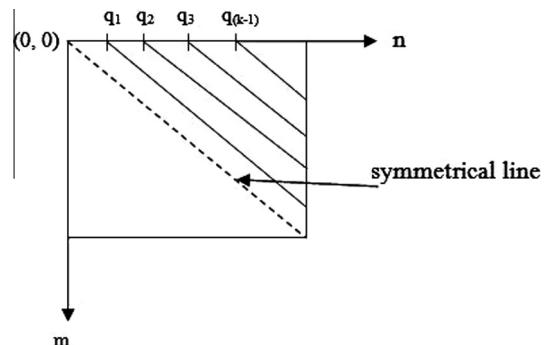


Fig. 5. Multiple threshold calculation area.

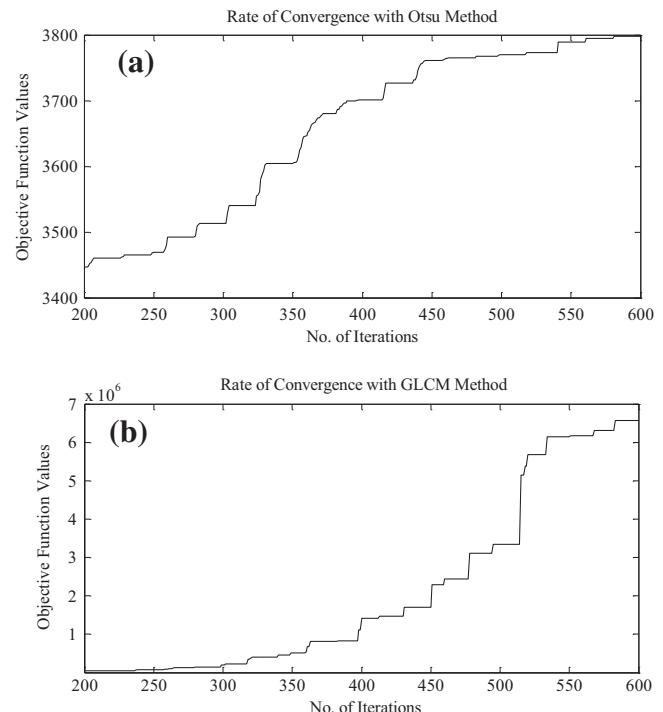
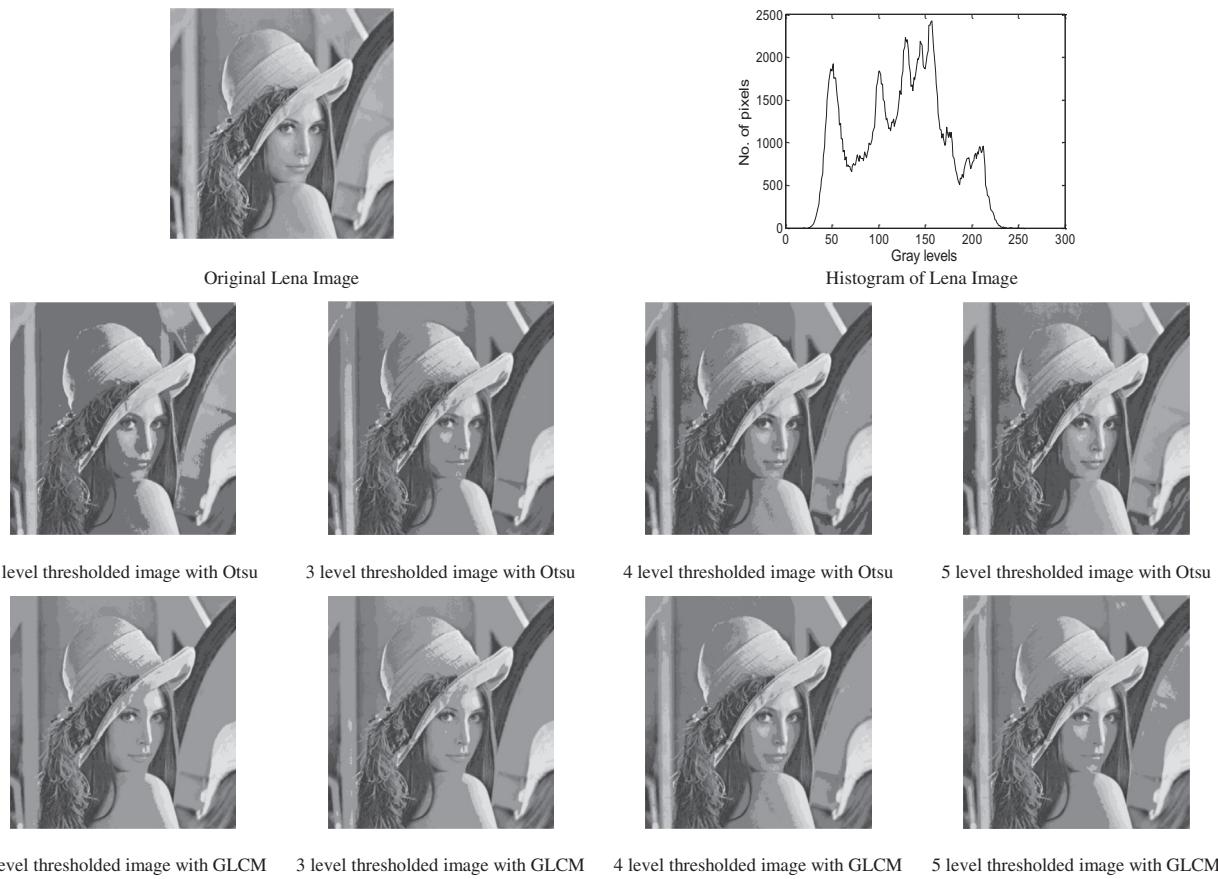
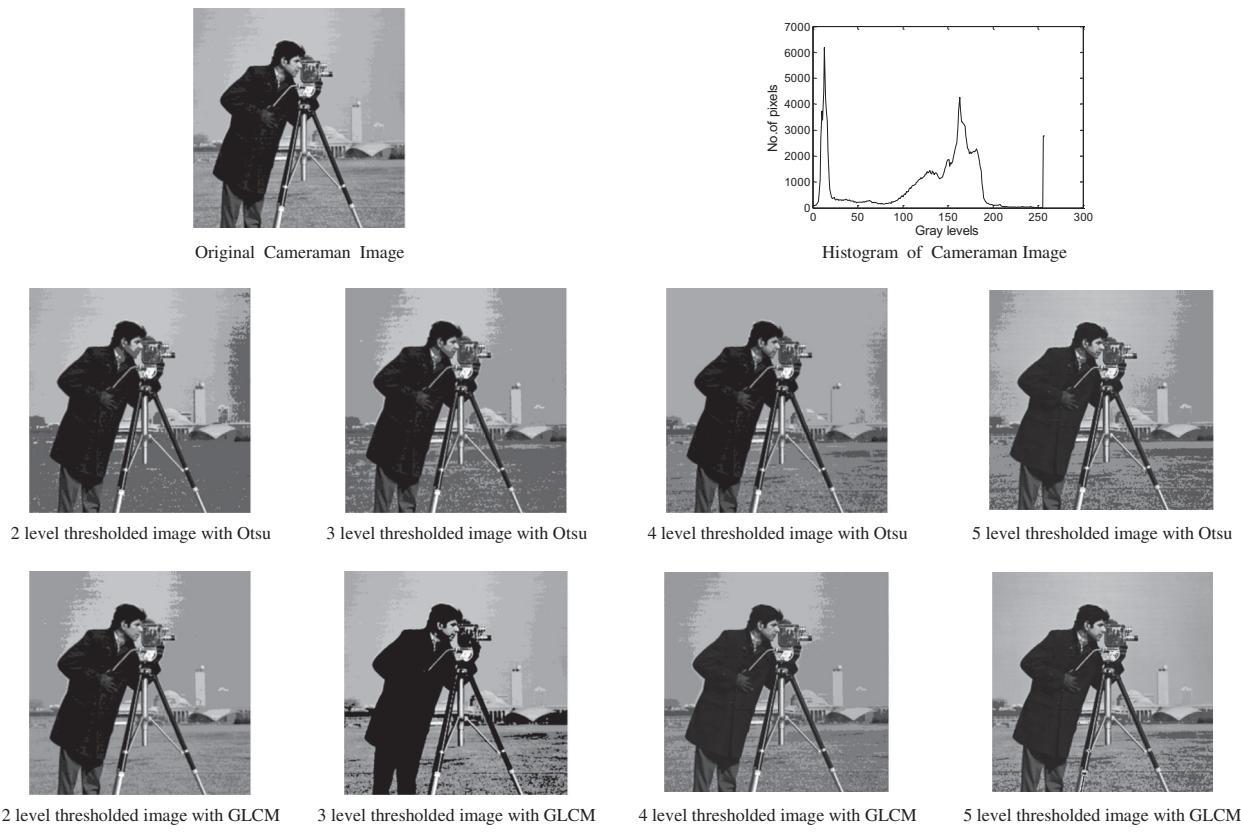
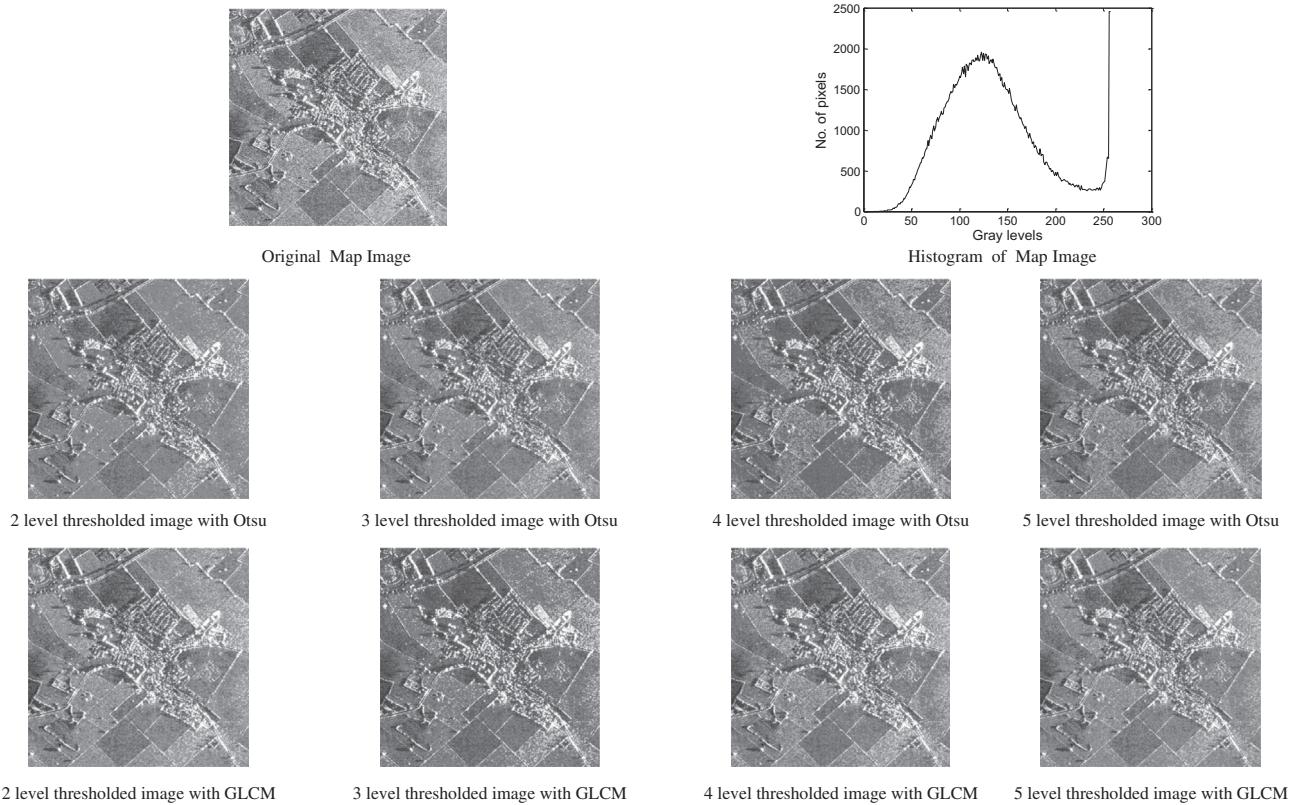
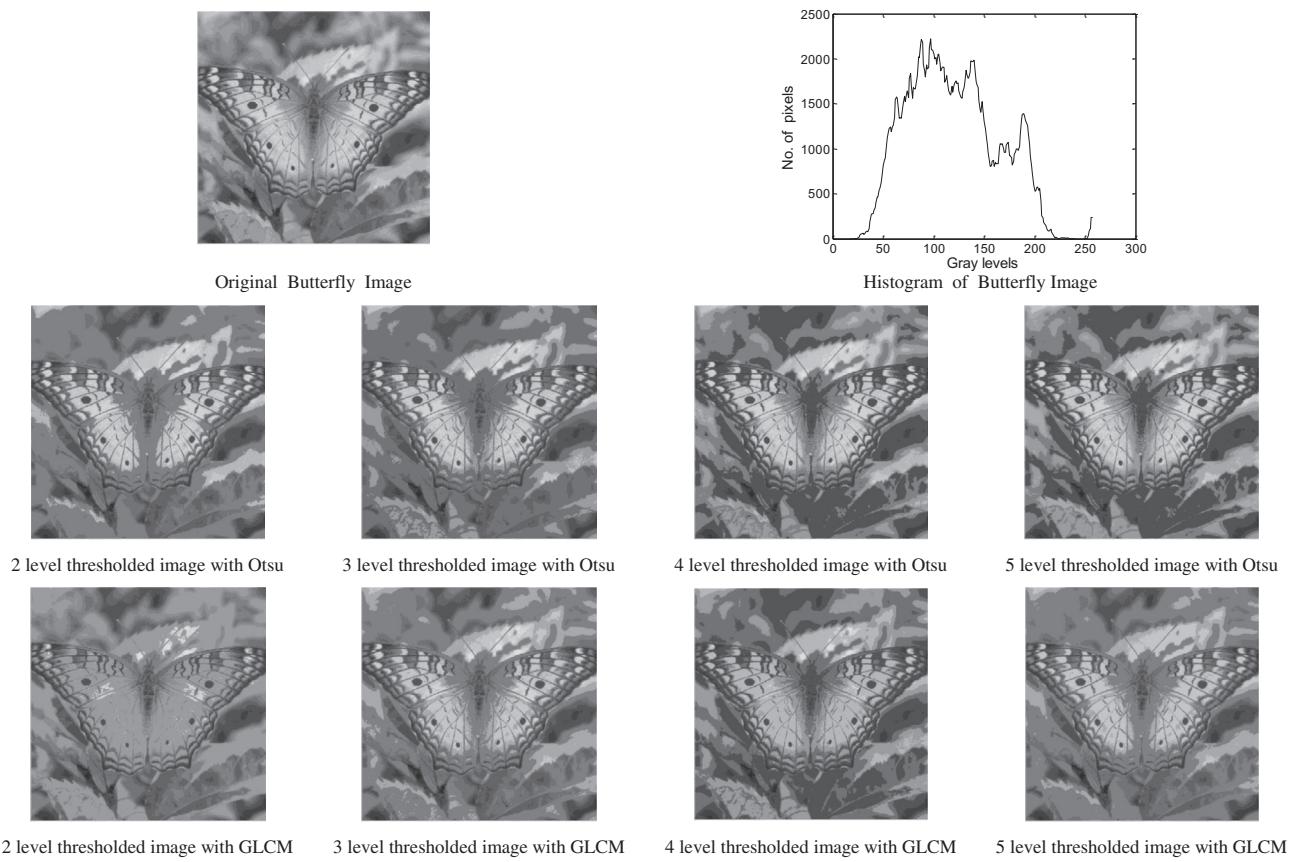


Fig. 6. (a) The rate of convergence with histogram based method. (b) The rate of convergence with edge magnitude based method.

**Fig. 7.** Results of thresholding using Otsu and GLCM methods.**Fig. 8.** Results of thresholding using Otsu and GLCM methods.

**Fig. 9.** Results of thresholding using Otsu and GLCM methods.**Fig. 10.** Results of thresholding using Otsu and GLCM methods.

The Cuckoo search algorithm is used here to optimize different edge magnitudes 'q' which determines the multilevel thresholds.

4. Proposed technique

In this paper, edge magnitudes obtained from GLCM is used for multilevel thresholding. Generally, the real life images do not contain binary information. The gray level histogram of such images is multimodal in nature. Hence, single level thresholding does not yield satisfactory results. That's why we consider here multi-level thresholding by using edge magnitudes from GLCM. The key to our success is that we have exploited the correlation among gray levels in an image for improving thresholding performance. Let an image I contains L gray levels from 0 to $L - 1$. Multilevel thresholding separates the pixels of the image into K classes C_1, C_2, \dots, C_k by setting thresholds t_1, t_2, \dots, t_{k-1} . The threshold t_0 is equal to 0 and t_k is equal to $L - 1$. The block diagram of the proposed technique is presented in Fig. 4.

The GLCM of the input image is computed using (1). The edge information which is found in GLCM is used here for thresholding. Here we introduce new objective functions for obtaining optimal threshold values. There is a strong need to maximize the multiple functional $f(\cdot)$ to get optimal threshold values. To be more specific, our aim is to get the best ' q_k ' out of a selection of [1, 254]. Therefore, the dimension of the problem depends on the number of threshold values. This has motivated the authors to employ a heuristic iterative optimization scheme that modifies individuals in a population of contestants. In this context, we propose objective functions well suited for heuristic search algorithms. The thresholding method used here is by optimizing an objective function as presented in

(4). The optimal thresholds are the one which maximizes the objective function

$$[t_{1opt}, t_{2opt}, \dots, t_{k-1opt}] = \arg \max \{f(q_1, q_2, \dots, q_{k-1})\} \quad (4)$$

subject to the following constraints

$$0 < t_{1opt} < t_{2opt} < \dots < t_{(k-1)opt} < L - 1 \quad (5)$$

The thresholds obtained are optimum when the summation over 'q' is maximized. The edge magnitude 'q' is optimized using the Cuckoo search algorithm. The number of nests is chosen 30. The dimension of search space is chosen as per the number of thresholds. The number of iterations is chosen and the population of nests is initialized. The fitness value is also initialized. Then a solution i.e. 'q' is randomly chosen and compared with all other solutions. The solution with the best fitness is then chosen as the best solution. The multiple threshold values are then calculated as:

$$t_{1opt} = \arg \max \left(\frac{1}{\eta_1} \sum_{m=0}^{q_1} \sum_{n=q_1+1}^{q_2} \left(\frac{m+n}{2} \right) \text{GLCM}(m, n) \right) \quad (6)$$

$$t_{2opt} = \arg \max \left(\frac{1}{\eta_2} \sum_{m=q_1+1}^{q_2} \sum_{n=q_2+1}^{q_3} \left(\frac{m+n}{2} \right) \text{GLCM}(m, n) \right) \quad (7)$$

and

$$t_{k-1opt} = \arg \max \left(\frac{1}{\eta_{k-1}} \sum_{m=q_{k-2}+1}^{q_{k-1}} \sum_{n=q_{k-1}+1}^{L-1} \left(\frac{m+n}{2} \right) \text{GLCM}(m, n) \right) \quad (8)$$

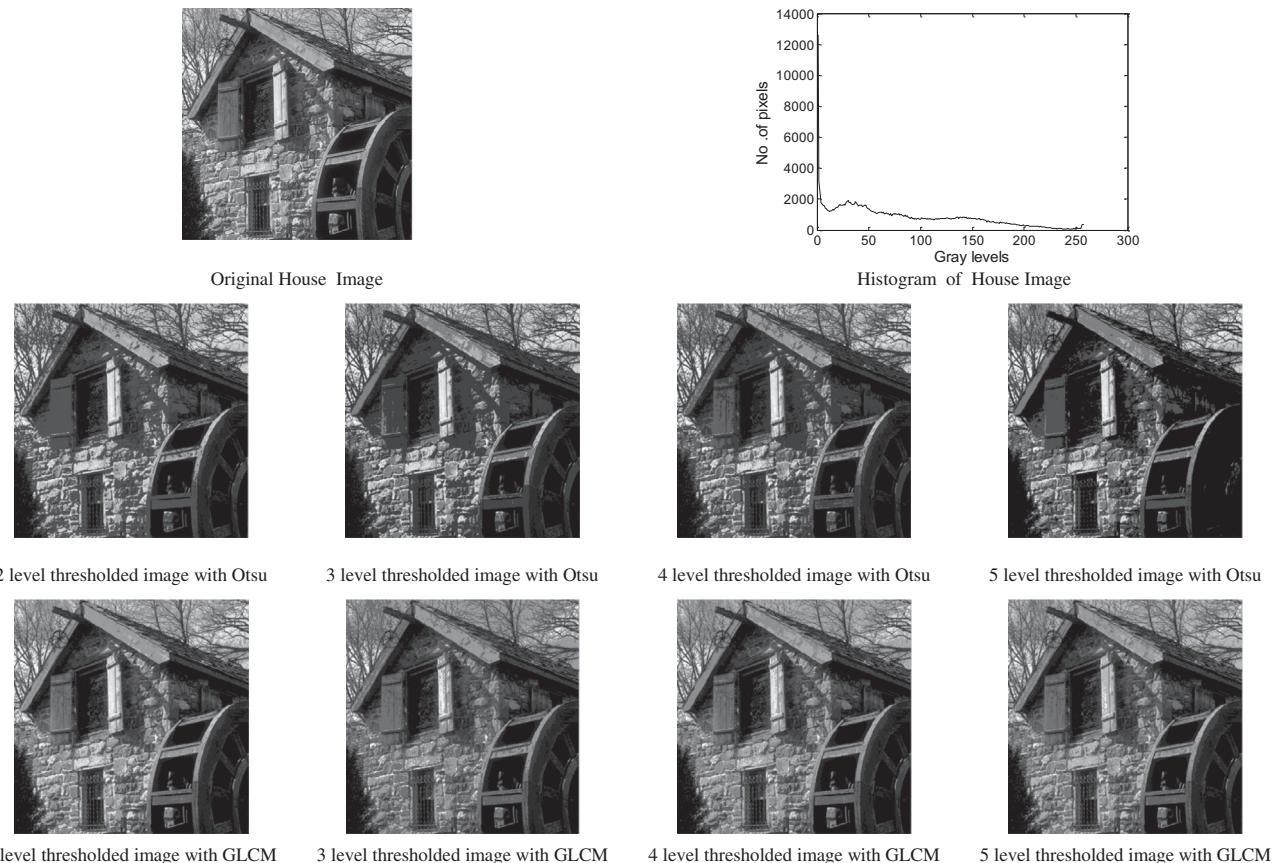


Fig. 11. Results of thresholding using Otsu and GLCM methods.

where

$$\eta_1 = \sum_{m=0}^{q_1} \sum_{n=q_1+1}^{q_2} \text{GLCM}(m, n) \quad (9)$$

$$\eta_2 = \sum_{m=q_1+1}^{q_2} \sum_{n=q_2+1}^{q_3} \text{GLCM}(m, n) \quad (10)$$

$$\eta_{k-1} = \sum_{m=q_{k-2}+1}^{q_{k-1}} \sum_{n=q_{k-1}+1}^{L-1} \text{GLCM}(m, n) \quad (11)$$

In the above equations, q_1, q_2, \dots, q_{k-1} represent the edge magnitudes corresponding to multiple threshold values. These are optimized using Cuckoo search algorithm and then the corresponding threshold values are calculated by using (6)–(8). The thresholds obtained are optimum when the summation over ‘ q ’ is maximized. The multiple threshold calculation area is displayed in Fig. 5. The symmetrical line represents the area where the edge magnitude is zero. The computation area is restricted to upper triangle only due to symmetry of GLCM.

It may be reiterated the fact that edge magnitudes q_1, q_2, \dots, q_{k-1} are explicitly free from any spatial distribution information. Interestingly, they only provide us corresponding optimal threshold values. Alternately, the problem is a maximization problem. It is needed to maximize multiple functional $f(\cdot)$. Therefore it is wise to choose a heuristic search algorithm. These ideas are presented in this section. Here Cuckoo search strategy is used to maximize the multiple functional $f(\cdot)$.

Edge magnitude based multi-level thresholding technique is used here. For a comparison, Otsu's histogram based between-class variance method [4] has also been implemented for multi-level

thresholding. We have also optimized the between class variance for multilevel thresholding by using Cuckoo search technique. A separate objective function (19) is used for histogram based 2between-class variance method. Then the comparison is made between our method and the histogram based between-class variance method.

The Otsu's histogram based between-class variance method is described as below [4]. Let the pixels of an image be represented in L gray levels $[0, 1, 2, \dots, L - 1]$. The gray level histogram of the image is normalized and treated as probability distribution:

$$p_i = \frac{n_i}{N}, \quad p_i \geq 0 \quad \text{and} \quad \sum_{i=0}^{L-1} p_i = 1 \quad (12)$$

where n_i is the number of pixels at level i and N is the total number of pixels in the image. If we assume single level thresholding, then a threshold ‘ t ’ will divide the image into two classes, C_0 and C_1 . C_0 will have pixels with gray levels $[0 \text{ to } t]$ and C_1 will have pixels with gray levels $[t + 1 \text{ to } L - 1]$. Now the probability distribution for each class can be given as,

$$C_0 : \frac{p_0}{\omega_0(t)}, \dots, \frac{p_t}{\omega_0(t)} \quad \text{and} \quad C_1 : \frac{p_{t+1}}{\omega_1(t)}, \dots, \frac{p_{L-1}}{\omega_1(t)} \quad (13)$$

where

$$\omega_0(t) = \sum_{i=0}^t p_i \quad \text{and} \quad \omega_1(t) = \sum_{i=t+1}^{L-1} p_i \quad (14)$$

and the class mean levels can be defined as,

$$\mu_0 = \sum_{i=0}^t \frac{ip_i}{\omega_0(t)} \quad \text{and} \quad \mu_1 = \sum_{i=t+1}^{L-1} \frac{ip_i}{\omega_1(t)} \quad (15)$$

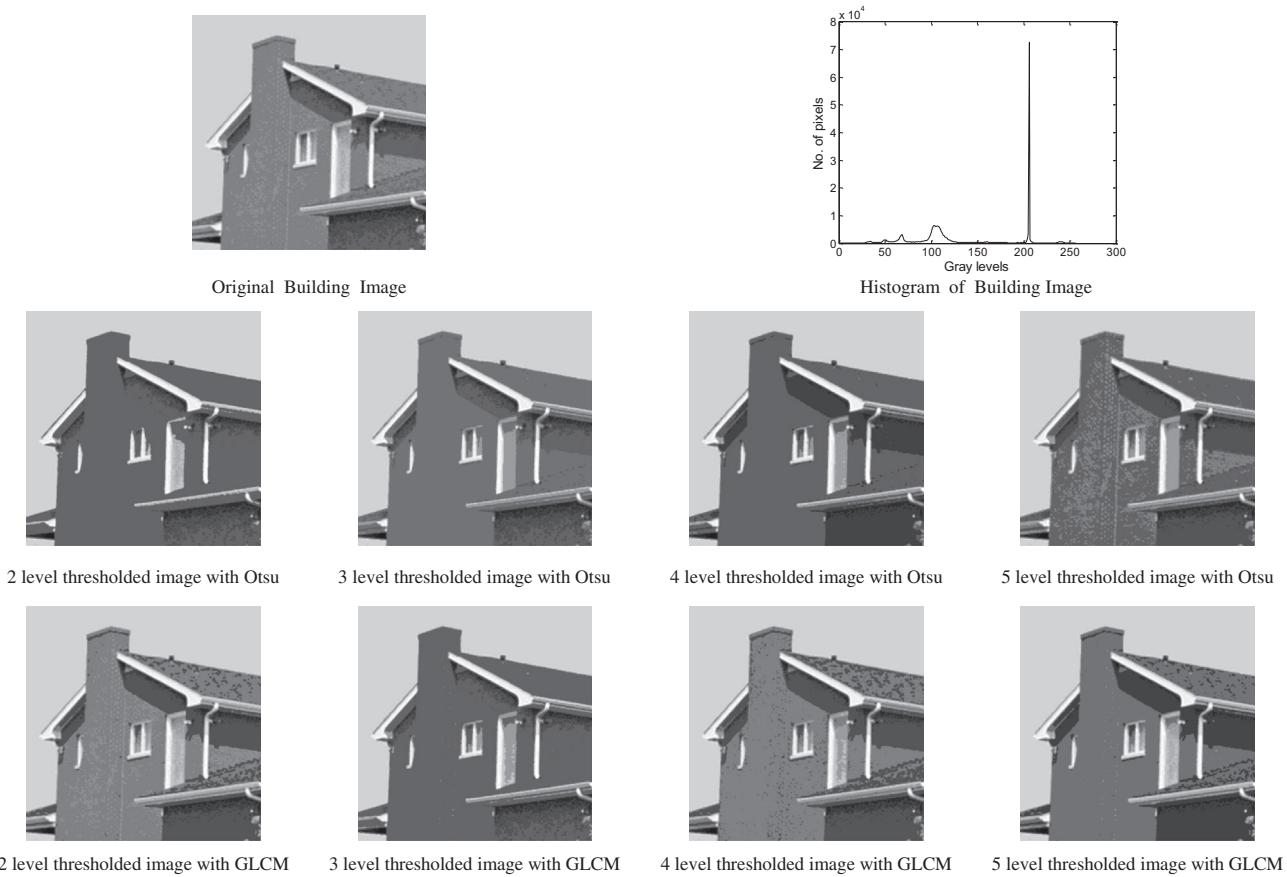


Fig. 12. Results of thresholding using Otsu and GLCM methods.

ω and μ represent the zeroth and first order moments of the histogram. Let μ_T represent the total mean level of the entire image, then it can be easily verified that

$$\omega_0\mu_0 + \omega_1\mu_1 = \mu_T \quad \text{and} \quad \omega_0 + \omega_1 = 1 \quad (16)$$

The between class variance as defined by Otsu is given as,

$$\sigma_B^2 = \sigma_0^2 + \sigma_1^2 \quad (17)$$

where

$$\sigma_0^2 = \omega_0(\mu_0 - \mu_T)^2 \quad \text{and} \quad \sigma_1^2 = \omega_1(\mu_1 - \mu_T)^2 \quad (18)$$

Based on the above formulation, an optimum threshold ‘t’ for single level thresholding is selected such that the between class variance σ_B^2 is maximized. This formulation has been extended (Akay, 2012; Sathya & Kayalvizhi, 2011) and implemented for multi-level thresholding as well. The optimal thresholds ($t_{1opt}, t_{2opt}, \dots, t_{k-1opt}$) are selected by maximizing the between class variance σ_B^2 as,

$$(t_{1opt}, t_{2opt}, \dots, t_{k-1opt}) = \arg \max \{\sigma_B^2(t_1, t_2, \dots, t_{k-1})\} \quad (19)$$

where

$$\sigma_B^2 = \sigma_0^2 + \sigma_1^2 + \dots + \sigma_{k-1}^2 \quad (20)$$

and

$$\sigma_0^2 = \omega_0(\mu_0 - \mu_T)^2, \quad (21)$$

$$\sigma_1^2 = \omega_1(\mu_1 - \mu_T)^2, \quad (22)$$

and

$$\sigma_{k-1}^2 = \omega_{k-1}(\mu_{k-1} - \mu_T)^2. \quad (23)$$

This divides the image into ‘k’ classes with ‘k – 1’ thresholds as $C_0[0 \text{ to } t_1]$, $C_1[t_1 + 1 \text{ to } t_2]$ and $C_{k-1}[t_{k-1} + 1 \text{ to } L - 1]$. It is important to mention here that the objective function to be maximized by Cuckoo search using edge magnitude method is represented by (4) and using histogram based method is represented by (19). It is observed that both the objective functions are different. This warrants us to evoke many more interesting ideas regarding the performance of both the objective functions. More discussions on this issue are presented in the following Section 5.

5. Results and discussions

The results obtained from both edge magnitude technique and histogram method is presented here. The experiments are carried out on an Intel dual core processor with 1.75 GHz and 1 GB RAM, running under the Windows 7 operating system. The algorithms are developed using MATLAB release 2009. The parameters chosen for Cuckoo search algorithm are: Number of nests-30, Number of iterations-20, Mutation probability- 0.25 and scale factor-1.5. The test images are thresholded for levels 2–5 using both methods. Nine different images are considered for this experiment. Lena and cameraman images are two well known test images which are used in this experiment. Figs. 7–15 display thresholded images. As observed from the histogram of images, they are multimodal in nature. This makes them suitable for multilevel thresholding application. The thresholded images are obtained by following the rules as mentioned below:

For 2-level thresholding, let $t_{1opt} = T_1$ and $t_{2opt} = T_2$. Then the thresholded image \tilde{I} having gray levels 0 to $L - 1$ will have gray levels:

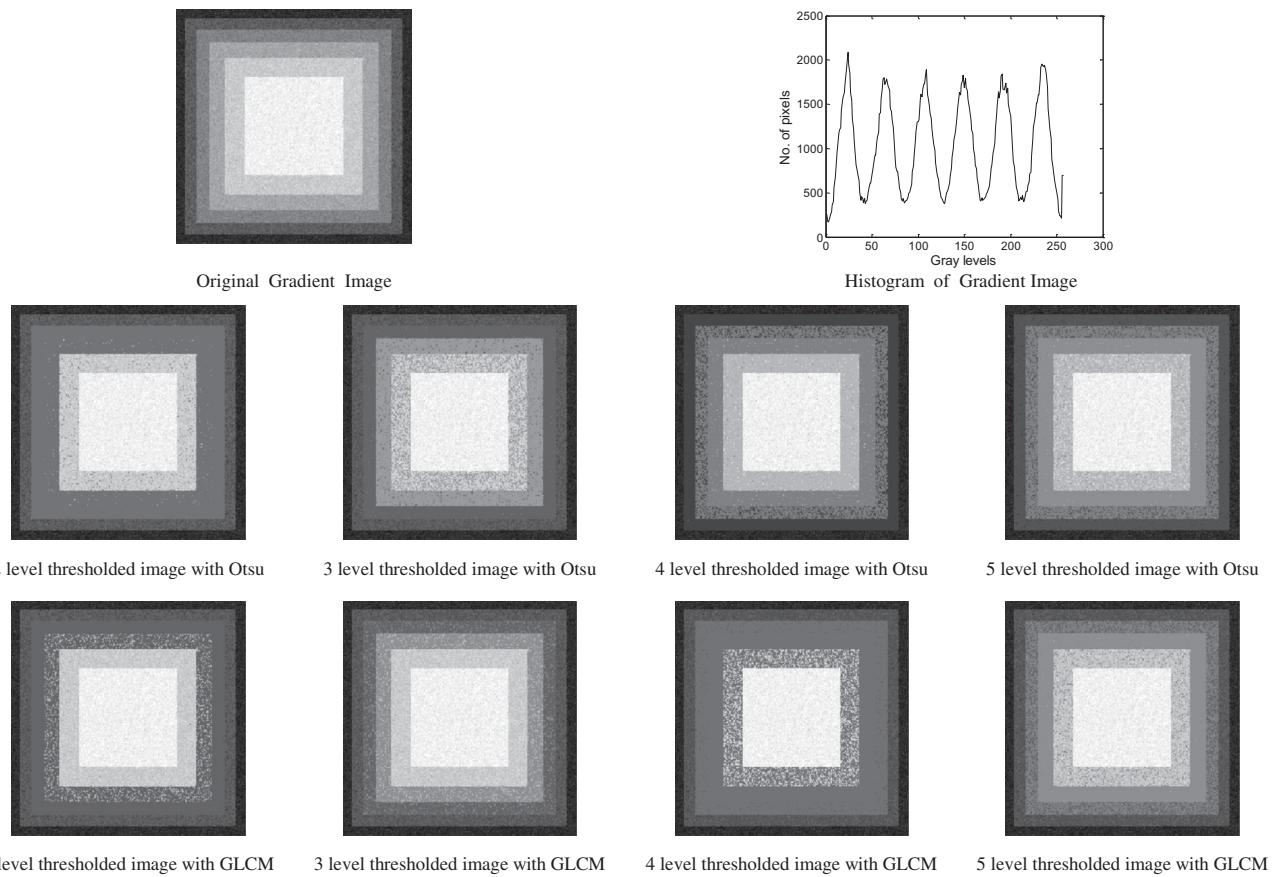
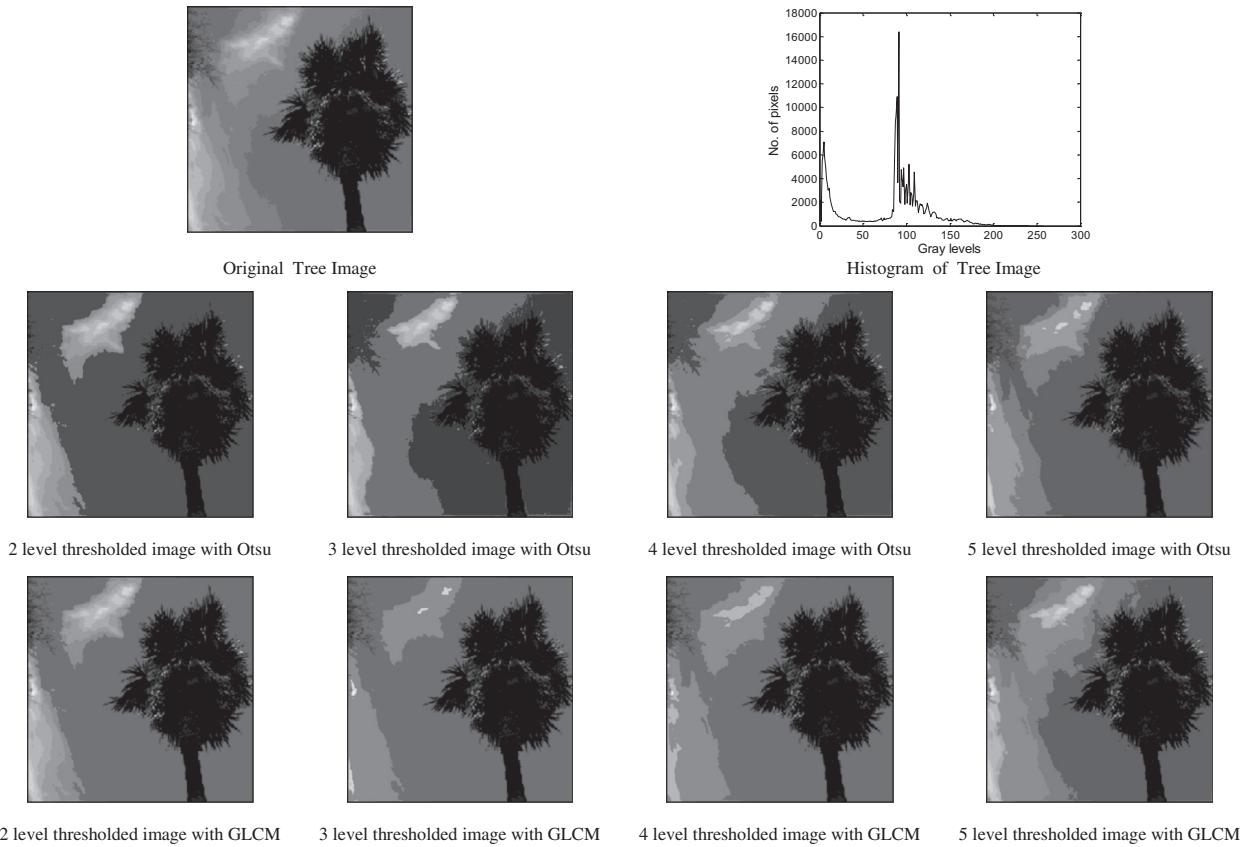
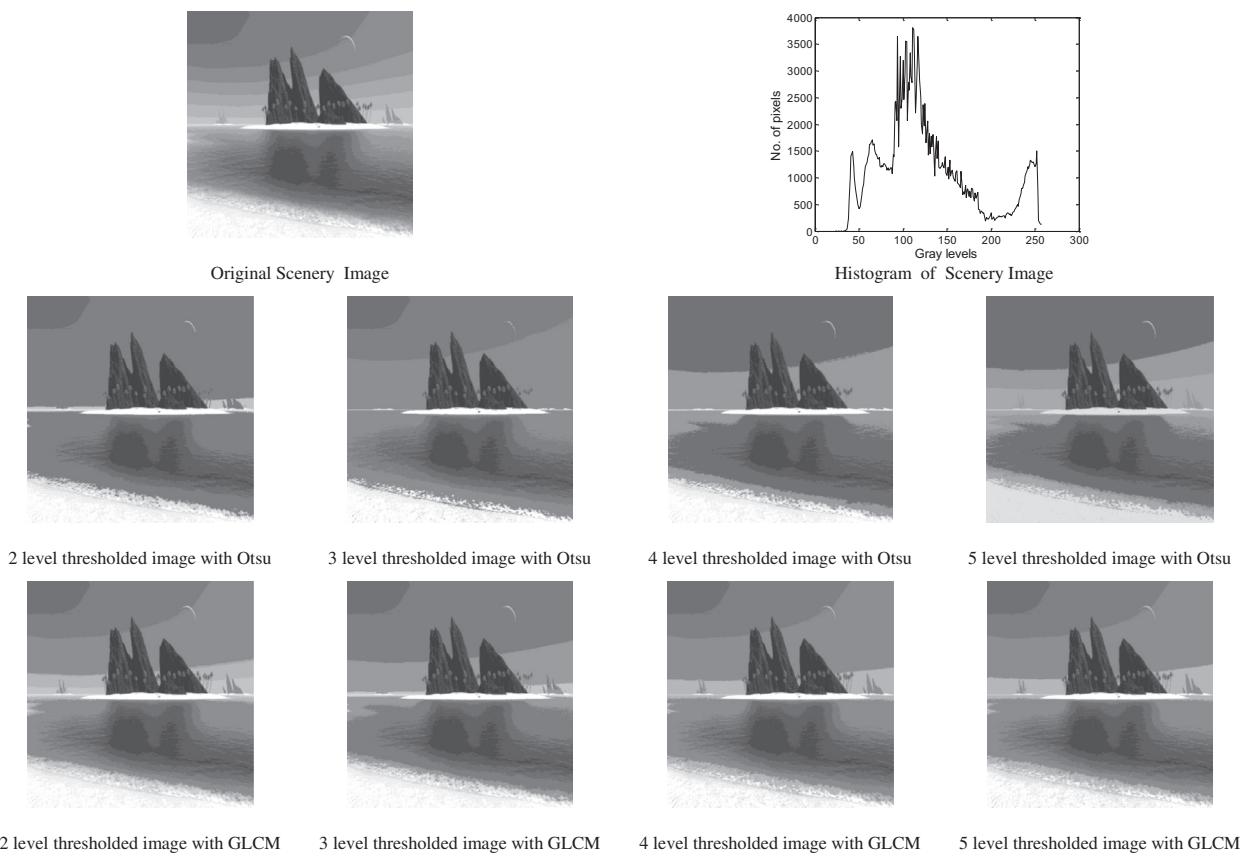


Fig. 13. Results of thresholding using Otsu and GLCM methods.

**Fig. 14.** Results of thresholding using Otsu and GLCM methods.**Fig. 15.** Results of thresholding using Otsu and GLCM methods.

T , for $0 < T \leq T_1$
 T_1 , for $T_1 < T \leq T_2$ and
 $T, T_2 < T < L - 1$

Similar rules are used for higher levels of thresholding. The result with histogram based method is also presented here to provide a comparison with our proposed technique. The reason for such a choice is due to the fact that histogram based methods are recently used (Akay, 2012; Sathy & Kayalvizhi, 2011) for multilevel thresholding technique. It is observed that images thresholded with GLCM method are qualitatively better than histogram based method. Tables 1 and 2 presents result for nine different types of images. Table 1 presents the best objective function value and the corresponding threshold values. Various measures as given in Table 2 are defined to justify our results quantitatively. Best objective function value for both methods using Cuckoo search technique is displayed in Table 1. It is observed that GLCM method gives higher objective function values than histogram based method. This is due to the fact that both objective functions are differently formulated. They are presented in Eqs. (4) and (19), respectively. In our case (4), the optimum thresholds are obtained when the summation over ' q ' is maximized. The edge magnitude ' q ' directly relates optimum threshold values. Here summation over ' q ' leads to a higher objective function value, which is desirable. Hence, it is believed that the proposed objective functions (Eqs. (6)–(8)) are useful for multilevel thresholding applications.

PSNR (Peak Signal to Noise Ratio) as given in Table 2 gives the similarity of the original image with respect to the thresholded

image at a particular level based on the root mean square error (RMSE) of each pixel as explained in (Akay, 2012).

$$PSNR(\text{in } dB) = 20\log_{10}\left(\frac{255}{RMSE}\right) \quad (24)$$

It is observed, from Table 2, that our proposed method gives a higher value of PSNR as compared to histogram based scheme. Higher is the PSNR value, better is the scheme. In our case, we get higher threshold values which leads to a reduced RMSE. To be precise, thresholded image \tilde{I} is more similar to the original image I by using our proposed method. SSIM (Structural Similarity) index (Wang, Bovik, Sheikh, & Simoncelli, 2004) evaluates the visual similarity between the original image I and the thresholded image \tilde{I} at a particular level. The SSIM index is calculated as:

$$SSIM(I, \tilde{I}) = \frac{(2\mu_I\mu_{\tilde{I}} + C1)(2\sigma_{I\tilde{I}} + C2)}{(\mu_I^2 + \mu_{\tilde{I}}^2 + C1)(\sigma_I^2 + \sigma_{\tilde{I}}^2 + C2)} \quad (25)$$

where μ_I is the average of I , $\mu_{\tilde{I}}$ is the average of \tilde{I} , σ_I^2 is the variance of I , $\sigma_{\tilde{I}}^2$ is the variance of \tilde{I} , $\sigma_{I\tilde{I}}^2$ is the variance of I and \tilde{I} . $C1$ and $C2$ are constants and are taken as 0.065. It is observed from Table 2 that SSIM is higher for our proposed method. In this sense, the proposed technique produces results which are visually better than the histogram based method. FSIM (Feature Similarity) index as in (Zhang, Zhang, Mou, & Zhang, 2011) is also used to measure the similarity between the two images. The FSIM index is calculated as:

$$FSIM = \frac{\sum_{X \in \Omega} S_L(X) \cdot PC_m(X)}{\sum_{X \in \Omega} PC_m(X)} \quad (26)$$

Table 1
Comparison of best objective function values and corresponding threshold values for test images.

Images	m	Best objective function value		Corresponding threshold values	
		GLCM	Histogram	GLCM	Histogram
Lena	2	2.7763e+006	1.9446e+003	129,184	87,146
	3	5.5273e+006	2.1047e+003	129,180,193	83,117,188
	4	5.3354e+006	2.1464e+003	127,165,210,229	59,98,136,165
	5	2.7116e+006	2.1814e+003	127,166,209,212,229	58,98,130,148,173
	2	6.5722e+006	3.7017e+003	130,175	73,150
Cameraman	3	6.8132e+006	3.7979e+003	10,106,123	78,139,174
	4	6.5001e+006	3.8850e+003	126,165,207,234	58,111,148,194
	5	3.2919e+006	3.9235e+003	126,165,202,215,234	52,107,134,172,216
	2	2.2037e+007	1.8126e+003	130,170	115,181
	3	2.8774e+007	1.9480e+003	67,95,164	114,157,202
Map	4	5.4579e+007	2.0139e+003	108,139,157,188	96,145,180,214
	5	1.1138e+007	2.0694e+003	126,165,199,215,233	81,105,146,174,228
	2	7.2972e+004	1.5930e+003	128,198	98,156
	3	7.1362e+003	1.6793e+003	59,148,213	90,137,171
	4	5.6535e+004	1.7148e+003	97,143,179,196	54,97,136,164
Butterfly	5	9.5764e+004	1.7556e+003	97,150,179,218,220	58,98,130,148,173
	2	7.8245e+006	3.3587e+003	139,173	62,128
	3	1.0721e+007	3.5665e+003	82,136,213	39,96,153
	4	1.6377e+006	3.6477e+003	121,152,208,232	39,78,139,178
	5	1.1964e+006	3.6877e+003	83,105,137,166,228	15,48,108,163,191
House	2	5.6087e+005	3.1634e+003	53,94	78,154
	3	7.6375e+005	3.2361e+003	67,130,174	79,137,186
	4	5.7984e+005	3.2530e+003	55,98,135,165	39,78,139,178
	5	1.7167e+005	3.2756e+003	66,131,156,188,222	49,78,110,129,181
	2	7.8714e+005	4.7429e+003	69,155	89,172
Building	3	3.5161e+005	4.9369e+003	66,113,153	67,132,187
	4	5.2038e+005	5.0633e+003	57,67,92,194	45,101,168,206
	5	1.7621e+005	5.1428e+003	64,100,119,180,199	54,97,118,174,200
	2	1.8482e+005	2.0517e+003	82,118	54,118
	3	2.1375e+005	2.1081e+003	84,120,190	39,94,139
Gradient	4	1.7339e+005	2.1413e+003	81,111,164,217	54,97,136,164
	5	7.5248e+005	2.1602e+003	73,99,111,126,154	49,78,110,129,181
	2	2.2706e+006	2.5062e+003	123,157	110,181
	3	2.2614e+006	2.6214e+003	105,140,181	96,119,216
	4	2.2270e+006	2.7534e+003	127,165,210,231	58,111,148,194
Scenery	5	1.0915e+006	2.7878e+003	123,163,175,212,234	90,130,156,212,242

where Ω is the whole image spatial domain. $S_L(X)$ is the overall similarity of two images, $PC_m(X)$ is the maximum phase congruence between two images. Interestingly, $FSIM$ is also higher for our proposed method (as depicted in Table 2). Contrast (CON) and Inverse Difference Moment (IDM) (Baraldi & Parmiggiani, 1995) are also used to describe the GLCM parameters. The contrast is defined as:

$$CON = \sum_{p=0}^{L-1} p^2 \left\{ \sum_{m=0}^{L-1} \sum_{n=0}^{L-1} C(m, n) \right\}, \quad (27)$$

$$|m - n| = p$$

Here, $C(m, n)$ represents the normalized GLCM. A high value of contrast implies better correlation between the first order statistics contrast and GLCM contrast, which is higher for our proposed method. Results obtained by using edge magnitude based method yields better contrast values as shown in Table 2.

The IDM measures the homogeneity of the image and is calculated as:

$$IDM = \sum_{m=0}^{L-1} \sum_{n=0}^{L-1} \left[1/(1 + (m - n)^2) \right] C(m, n) \quad (28)$$

Here, $C(m, n)$ represents normalized GLCM. It is inversely correlated to contrast. It is lower for our proposed method, which is desirable. In summary, the proposed method has shown improved results with respect to different measures.

Fig. 6(a) displays the rate of convergence curve for histogram based method, using CS algorithm, for the cameraman image thresholded at level 3. Fig. 6(b) displays the rate of convergence curve

for edge magnitude based method, using CS algorithm, for the cameraman image thresholded at level 3. It is observed that for same number of iterations, the objective function value attains a much higher value in our proposed method as shown in Fig. 6(b). Although both objective functions are different, it is noteworthy to mention here that the rate of convergence with GLCM seems to be competitive.

The best objective function values are displayed in Table 1. For cameraman image with threshold level 3, the best objective function value is **6.5722e + 006** by using GLCM method. Whereas the best objective function value is **3.7979e + 003** for histogram based method. The maximum objective function values are obtained at around 600 iterations in both cases by using CS algorithm. At 200 iterations, the objective function value is **0.15e + 006** by using GLCM method. Whereas the objective function value is **3.450e + 003** for histogram based method at 200 iterations. It is seen that the objective function values always remain higher for our case. From these results, it is found that the proposed objective functions are well suited for multilevel thresholding applications. Further, these functions are very much compatible with meta-heuristic search algorithm like CS algorithm. In general, it is believed that objective functions with higher values in maximization problems are more suited for different engineering applications (Sarkar, Das, & Chaudhuri, 2012; Sarkar, Patra, & Das, 2011).

6. Conclusion

Edge magnitudes from GLCM are effectively used for multi-level thresholding. Cuckoo search algorithm is efficiently used for maximizing edge information for selection of optimized thresholds.

Table 2
Comparison of evaluation parameters for test images.

Images	m	PSNR		SSIM		FSIM		Contrast		IDM	
		GLCM	OTSU	GLCM	OTSU	GLCM	OTSU	GLCM	OTSU	GLCM	OTSU
Lena	2	23.4362	20.6751	0.9720	0.9528	0.9723	0.9751	710.78	616.10	0.0542	0.0647
	3	23.9096	22.0513	0.9751	0.9685	0.9733	0.9813	718.19	552.14	0.0645	0.0714
	4	24.6523	23.4636	0.9821	0.9789	0.9896	0.9889	756.90	598.07	0.0795	0.0815
	5	25.0232	24.5300	0.9901	0.9833	0.9921	0.9903	731.09	631.85	0.0820	0.0852
	2	22.6829	18.7141	0.9765	0.9417	0.9894	0.9713	1.1462e+003	818.55	0.0484	0.0613
Cameraman	3	21.5246	20.2893	0.9681	0.9632	0.9871	0.9859	1.2823e+003	793.57	0.0506	0.0631
	4	22.2643	21.8086	0.9783	0.9756	0.9922	0.9903	1.0923e+003	726.41	0.0587	0.0719
	5	24.8527	22.9834	0.9793	0.9609	0.9977	0.9921	994.65	897.03	0.0698	0.0843
	2	27.3341	21.1621	0.9880	0.9555	0.9989	0.9941	1.6424e+003	1258.25	0.0343	0.0414
	3	25.4532	23.5431	0.9828	0.9756	0.9927	0.9973	1.3678e+003	963.70	0.0414	0.0477
Map	4	26.9922	21.5160	0.9892	0.9671	0.9989	0.9958	1.1592e+003	922.42	0.0440	0.0571
	5	25.9476	22.7437	0.9850	0.9768	0.9985	0.9983	1.0221e+003	910.84	0.0455	0.0680
	2	22.1818	22.1187	0.9633	0.9605	0.9764	0.9682	824.44	651.30	0.0648	0.0758
	3	24.6534	23.1557	0.9814	0.9712	0.9798	0.9766	688.40	671.65	0.0731	0.0869
	4	23.1703	22.0814	0.9714	0.9711	0.9772	0.9821	719.02	592.3512	0.0852	0.1046
Butterfly	5	23.0677	22.4705	0.9803	0.9787	0.9820	0.9763	637.67	582.05	0.0772	0.1136
	2	29.7929	22.6828	0.9934	0.9715	0.9975	0.9946	1.528e+003	130.66	0.0356	0.0421
	3	21.7246	21.2280	0.9623	0.9611	0.9963	0.9926	861.52	125.59	0.0545	0.0594
	4	27.0956	21.6956	0.9892	0.9648	0.9988	0.9955	1.209e+003	1.109e+003	0.0401	0.0589
	5	26.1473	20.5667	0.9868	0.9556	0.9990	0.9948	1.0475e+003	946.04	0.0523	0.0676
House	2	29.7972	21.5648	0.9944	0.9718	0.9961	0.9837	394.85	385.488	0.0696	0.0835
	3	23.0342	22.5782	0.9630	0.9782	0.9830	0.9869	280.5727	248.87	0.0926	0.0966
	4	25.8330	21.3740	0.9887	0.9752	0.9941	0.9853	321.03	315.66	0.0839	0.0952
	5	26.0921	23.7377	0.9921	0.9842	0.9989	0.9934	398.70	303.32	0.0865	0.0988
	2	19.8745	19.7654	0.9571	0.9607	0.9608	0.9570	445.70	441.48	0.0584	0.0592
Building	3	23.5664	21.2212	0.9840	0.9734	0.9694	0.9629	396.62	379.31	0.0586	0.0612
	4	20.0465	19.8707	0.9676	0.9671	0.9732	0.9701	567.92	352.95	0.0633	0.0690
	5	23.8159	23.2174	0.9826	0.9842	0.9833	0.9861	408.90	384.70	0.0629	0.0652
	2	27.0019	18.4057	0.9870	0.9114	0.9861	0.9781	381.47	345.71	0.0824	0.1007
	3	24.3007	18.8835	0.9750	0.9208	0.9742	0.9672	395.83	336.05	0.0655	0.1123
Tree	4	24.9428	21.1269	0.9794	0.9532	0.9756	0.9741	385.32	290.69	0.0602	0.0914
	5	27.4881	24.7188	0.9892	0.9773	0.9832	0.9759	355.41	299.60	0.0592	0.0904
	2	30.6529	21.8091	0.9949	0.9643	0.9882	0.9731	610.99	546.69	0.0697	0.0722
	3	26.0174	20.5080	0.9873	0.9539	0.9811	0.9599	536.53	527.68	0.0714	0.0766
	4	27.5035	22.3451	0.9899	0.9723	0.9865	0.9756	752.08	723.98	0.0753	0.0769
Scenery	5	27.6448	22.4499	0.9903	0.9759	0.9857	0.9738	763.11	586.80	0.0754	0.0818

Results are then compared with the histogram based between class variance method used for multi-level thresholding. From the results, we observe that the proposed edge magnitude based method is proved to be one of the contestants for multi-level thresholding. The Cuckoo search algorithm seems to be very effective as it involves few parameters for tuning. Similarity, Fidelity and Contrast related measures presented in Table 2 reveal the fact that our method outperforms histogram based method.

References

- Akay, B. (2012). A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding. *Applied Soft Computing*, in press. <http://dx.doi.org/10.1016/j.asoc.2012.03.072>.
- Baraldi, A., & Parmiggiani, F. (1995). An investigation of the textural characteristics associated with gray level co-occurrence matrix statistical parameters. *IEEE Transactions on Geoscience and Remote Sensing*, 33(2), 293–304.
- Chakraverty, S., & Kumar, A. (2011). Design optimization for reliable embedded system using cuckoo search. *International Conference on Electronics Computer Technology*, 1, 264–268.
- Chanda, B., & Majumder, D. D. (1988). A note on the use of the gray-level co-occurrence matrix in threshold selection. *Signal Processing*, 15, 149–167.
- Chang, C. I., Du, Y., Wang, J., Guo, S. M., & Thouin, P. D. (2006). Survey and comparative analysis of entropy and relative entropy thresholding techniques. *IEE Proceedings Vision Image Signal Processing*, 153(6), 837–850.
- Clausi, A. D., & Jernigan, Ed. M. (1998). A fast method to determine co-occurrence texture features. *IEEE Transactions on Geoscience and Remote Sensing*, 36(1), 298–300.
- Fritz, A. (2008). Statistical texture measures computed from gray level co-occurrence matrices. Image Processing Laboratory, Department of Informatics, University of Oslo.
- Gonzalez, R., & Woods, J. (1992). *Digital image processing*. Digital image processing: Addison-Wesley.
- Haralick, R., Shanmugam, K., & Dinstein, I. (1973). Texture features for image classification. *IEEE Transactions on SMC*, 3(6), 610–621.
- Haralick, R., & Shapiro, L. G. (1992). *Computer and robot vision-I*. Addison-Wesley.
- Kapur, J. N., Sahoo, P. K., & Wong, A. K. C. (1985). A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision Graphics and Image Processing*, 29, 273–285.
- Kittler, J., & Illingworth, J. (1986). Minimum error thresholding. *IEEE Transactions on Pattern Recognition*, 19(1), 41–47.
- Kurita, T., Otsu, N., & Abdelmalek, N. (1992). Maximum likelihood thresholding based on population mixture models. *Pattern Recognition*, 25(10), 1231–1240.
- Lee, S. U., Chung, S. Y., & Park, R. H. (1990). A comparative performance study of several global thresholding techniques for segmentation. *Computer Vision Graphics and Image Processing*, 52, 171–190.
- Li, Y., Cheriet, M., & Ching Suen, Y. (2005). A threshold selection method based on multi-scale and gray level co-occurrence matrix analysis. *Proceedings of International Conference on Document Analysis and Recognition*, 2, 167–172.
- Mokji, M. M., & Abu Bakar, S. A. R. (2007). Adaptive thresholding based on co-occurrence matrix edge information. *Journal of Computer*, 2, 44–52.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on SMC*, 9(1), 62–66.
- Pal, N. R., & Pal, S. K. (1989). Entropic thresholding. *Signal Processing*, 16, 97–108.
- Pun, T. (1980). A new method for grey-level picture thresholding using the entropy of the histogram. *Signal Processing*, 2, 223–237.
- Pun, T. (1981). Entropic thresholding: A new approach. *Computer Graphics and Image Processing*, 16, 210–239.
- Sahoo, P. K., Soltani, S., Wong, A. K. C., & Chen, Y. C. (1988). A survey of thresholding techniques. *Computer Vision Graphics and Image Processing*, 41, 233–260.
- Sarkar, S., Das, S., & Chaudhuri, S. S. (2012). Multilevel image thresholding based on Tsallis entropy and differential evolution. *Swarm, evolutionary, and memetic computing. Lecture notes in computer science* (7077). (pp. 17–24).
- Sarkar, S., Patra, G. R., & Das, S. (2011). A differential evolution based approach for multilevel image segmentation using minimum cross entropy thresholding. *Swarm, evolutionary, and memetic computing. Lecture notes in computer science* (7076). (pp. 51–58).
- Sathy, P. D., & Kayalvizhi, R. (2011). Optimal multilevel thresholding using bacterial foraging algorithm. *Expert Systems with Applications*, 38(12), 15549–15564.
- Sezgin, M., & Sankur, B. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1), 146–165.
- Tsai, W. (1985). Moment-preserving thresholding: A new approach. *Computer Vision Graphics and Image Processing*, 29, 377–393.
- Wang, Z., Bovik, C. A., Sheikh, R. H., & Simoncelli, P. Eero. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612.
- Yang, X. S., & Deb, S. (2009). Cuckoo search via levy flights. In *World congress on nature & biologically inspired computing* (Vol. 54, pp. 210–214).
- Yang, X. S., & Deb, S. (2010). Engineering optimization by cuckoo search. *International Journal of Mathematical Modeling and Numerical Optimization*, 1(4), 330–343.
- 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.