Poonamdeep Kaur, Raman Maini / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 4, Jul-Aug 2013, pp.1872-1880

BFO Based Multithresholding Edge Detection Technique

Poonamdeep Kaur, Raman Maini, Phd.
Research Scholar University college of Engineering, Punjabi University, Patiala
Professor University college of Engineering, Punjabi University, Patiala

ABSTRACT

Bacterial foraging optimization algorithm (BFOA) has been widely accepted as a global optimization algorithm of current interest for optimization and control. It has already drawn the attention of researchers because of its efficiency in solving real-world optimization problems arising in several application domains. In this paper, we propose the use of BFOA in an integrated manner with traditional image segmentation techniques to provide an efficient edge detection technique for selected natural images. The developed experimental results are compared with the results of other known existing edge detection techniques based on K-means Clustering and Otsu's threshold selection methods. The comparison is done on the basis of qualitative and quantitative analysis. Such that qualitative analysis is based on visual inspection and quantitative analysis is done by using performance parameters like peak Signal-to-Noise Ratio (PSNR), mean square error (MSE) and agreement indicator. From experimental results it has been concluded that the proposed method is capable of achieving satisfactory results and performs better than other methods in study.

Keywords: thresholding, image segmentation, Bacteria Foraging Optimization algorithms, edge detection.

I. INTRODUCTION

Edge detection is a process used in most processing applications to capture the significant properties of objects in the image. These properties include discontinuities in the physical, geometrical and photometrical characteristics of objects. Edge detection can be used for region segmentation, feature extraction and object boundary description. Thus important information about the object such as the structure and the topology of an object in any given image is provided by an edge which helps in object recognition and segmentation. In case of analyzing and understanding the information in an acquired image, it is necessary to extract the area in which the objective material is included. Edge detection is a problem of fundamental importance in image analysis, and there are probably more algorithms in the literature for enhancing and detecting edges than any other single subject [6]. Edge Detection Methods frequently use thresholds to decide whether a pixel belongs to object or background in any image i.e., can be considered as an edge pixel or not. However, the thresholds negatively affect the image segmentation results often in accuracy. Therefore, the tuning values of the thresholding algorithms must be set up carefully. A multilevel thresholding method which allows the determination of the appropriate number of thresholds as well as the adequate threshold values was proposed in [6]. It is based on the one of the most efficient techniques for edge detection, which is entropy-based thresholding. In [1] authors proposed the use of GAs in an integrated manner with traditional image segmentation techniques to provide an efficient segmentation and edges detection for selected natural images. An edge detection method applicable to gray level images in which an objective function is used to extract edge points as well as their directions was proposed in [5]. Adaptive thresholding technique is used thereafter to suppress non maximum points and to highlight edge points. The proposed method gave better results than other methods however proposed method is bit slow as compared to other methods. Despite the large number of edge detection techniques presently available, no general methods admit a unique solution because contest may take part in the decision phase. It follows that general solutions are not possible, and each proposed technique can be used only to solve class of problems [1].

This research proposed a new method for edge detection problem using Bacteria Foraging Optimization Algorithm (BFO). The contribution of this work is the use of BFO to build a reliable and accurate method of edge detection by tuning a set of parameters available in BFO. One reason for using this kind of method is mainly related with the BFO ability to deal with complex search spaces in situations where only minimum knowledge is available. Moreover, the proposed BFOA method for edge detection is compared with other traditional methods in the literature. The applicability of the proposed methods has been tested on a large class of images using MATLAB toolbox; the obtained results are encouraging, performing well in case of various natural images.

II. WHAT IS BFO?

One of the most successful areas of application has been the use of BFO to solve a wide variety of difficult numerical optimization problems.

Bacterial foraging optimization algorithm (BFOA) has been widely accepted as a global optimization Algorithm inspired by the social foraging behaviour of Escherichia coli. Because of its efficiency in solving real-world optimization problems arising in several application domains it has already drawn the attention of researchers. It a population-based numerical optimization algorithm used to solve many complex optimization problems. Optimization problems defined by functions for which derivatives are unavailable or available at a prohibitive cost are appearing more and more frequently in computational science and engineering. As in our proposed technique multilevel thresholding is used to obtain better segmentation which leads to better edge detection results. Selecting correct threshold is a critical issue and results in increased computational complexity. For optimization and control we proposed nature-inspired optimization algorithms BFO to build a reliable and accurate method to tune a set of parameters for a traditional filter. Application of group foraging strategy of a swarm of E.coli bacteria in multi-optimal function optimization is the key idea of the new algorithm. Bacteria search for nutrients in a manner to maximize energy obtained per unit time. BFO complement existing optimization methods nicely in that they require no gradient information and are much less likely to get trapped in local minima on multimodal surfaces. In addition, BFO has been shown to be quite insensitive to the presence of noise[3].

- In BFO, a number of e_coli bacteria are randomly initialized to different numerical values such that each possible solution corresponds to individual in the population. Total number of bacteria generates what is called population. Each bacteria in the population is assigned a fitness value according to how good a solution to the problem based on a given fitness function. These solutions are taken and are regenerated by updating the numerical values assigned to individuals by a hope that new solutions generated are better than the old solutions and generation is complete. In this way optimal solution is obtained by moving whole population in one group.
- A number of generations are there to achieve optimal results. Such that at each generation, each individual is evaluated for its fitness on the basis of fitness function. If fitness criteria is satisfied then optimal results are achieved otherwise there are population reproduction takes place. On the basis of evaluation criteria the bacteria with small fitness are eliminated. Such that population is reproduced by reproduction and elimination of individuals are performed by Elimination and Dispersal step.
- All the bacteria are ordering in ascending order based on the value of fitness function (or nutrient function). According to the reproduction step, only the bacteria with higher fitness value can survive and population is reinitialized over the entire search space by moving bacteria unit step. This keeps the swarm size constant. Then fitness value is again computed against each regenerated individual.
- According to the possibility rules and considering the different positions of bacteria, It is possible that bacteria sticks in the last place and unable to navigates the entire search space; In this case, by using the elimination-dispersal step, these bacteria can be removed from the cycle of searching, or distributed their accumulation in the area [7].

III. EDGE DETECTION USING BFO

BFO algorithm consists of different parameters to segment the image. It can dynamically change the parameters to achieve the best performance. The edge detection based on Bacteria forging optimization process may be summarized as follows:

1. Once the initial population is available, the genetic cycle begins as shown in Figure.
2. The quality of the edge detection results for each parameter set is then evaluated. If the maximum edge detection quality for the current population is above a predefined threshold value or another stopping criterion, then, the cycle will terminate. Alternatively, if the performance is not achieved, the reproduction and elimination dispersal steps are applied to the high strength individuals in the population, which yield a new set of population that has better performance.
3. The new population is supplied back to the edge detection process, and thus, the cycle begins again. Each pass through the loop (Edge detection-Fitness-Reproduction, and Elimination dispersal) is known as generation. The cycle shown in Figure continues until the maximum fitness or termination criteria are achieved.

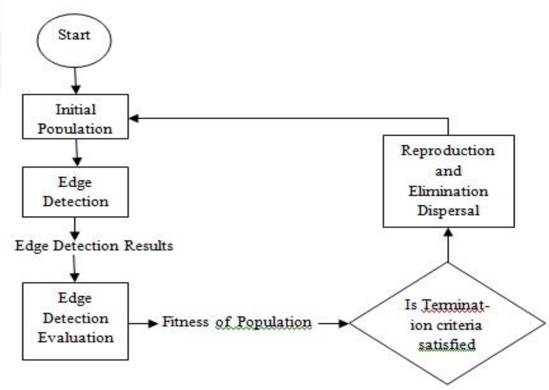


Figure1: Flow Chart of BFO based Edge Detection Method

3.1 Steps in Proposed Edge Detection Technique

Our edge detection technique is proposed by implementing different steps in MATLAB R2008b and is explained in detail.

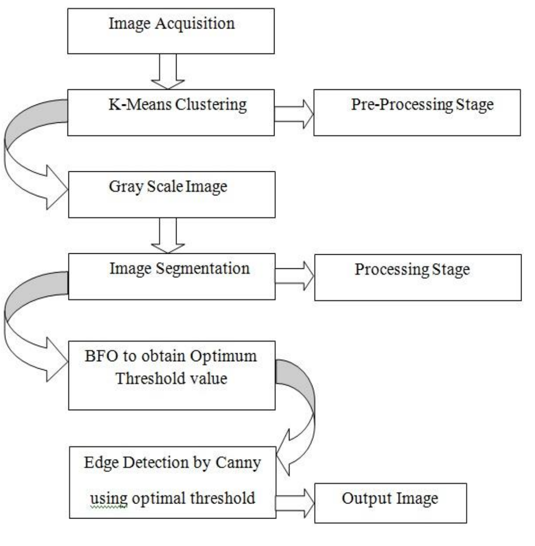


Figure2: The Flow Chart represents the steps of execution in proposed technique

3.1.1 Image Acquisition

In order to implement the proposed edge detector, the first step is to acquire a digital image on which we can apply various steps to get required simulation results. On the basis of which one can further compare the results.

3.1.2 K-Means Clustering

After acquiring the image next step is applying K-Means Clustering on acquired image. It is a pre-processing phase of edge detection technique which includes image similarity based on the distance function, connected component labelling, edge tracing, and initial fitness evaluation. The K-Means method is numerical, unsupervised, non-deterministic and iterative. K-Means clustering generates a specific number of disjoint, flat (non-hierarchical) clusters. The regions with similar pixels are grouped into regions as a final result of segmentation or clustering. At the end of clustering procedure, a label is assigned to each pixel that take a value between 1 and K, such that, each possible region of F can be represented by a matrix, and it corresponds to a pixel of the subset or partition.

3.1.3 Conversion to Gray scale

After applying K-means clustering next step in our proposed technique is conversion of color segmented image into gray scale image for further processing. A gray scale (or gray level) image is simply one in which the only colors are shades of gray. The reason for differentiating such images from any other sort of color image is that less information needs to be provided for each pixel.

3.1.4 Image segmentation

As after previous step implementation gray scale image is obtained. Next step in our proposed technique is segmentation of image into different parts using multilevel thresholding technique. The aim of an effective segmentation is to separate objects from the background and to differentiate pixels having nearby values for improving the contrast. Multilevel thresholding is a process that segments a gray-level image into several distinct regions. This technique determines more than one threshold for the given image and segments the image into certain brightness regions, which correspond to one background and several objects.

3.1.5 Bacteria Foraging Optimization Algorithm

As in our proposed technique to obtain consistent segmentation results multilevel thresholding is used which results in exponential increase in computational complexity and segmentation performance instability. To eliminate such problems, evolutionary techniques have been applied in solving multilevel thresholding problem. So in next step we propose novel optimal multilevel thresholding algorithm based on the foraging behaviour of bacteria. This algorithm selects the optimal threshold value based on the objective function.

Step1: Initialization of variables

i. Number of bacteria (S) to be used for finding the optimal solution.
ii. Dimension of search area (p).
iii. Specifying the total number of iterations (It).
Iv N_c is the number of chemotactic steps taken by each bacterium before reproduction.
v. initializing step size i.e., C(i)
v. Specifying swim size (N_s) i.e., maximum number of steps taken by each bacterium when it moves from low nutrient area to high nutrient area.
viN_{re} And N_{ed} is the number of reproduction and elimination dispersal events.
vii. Specifying (P_{ed}) is the probability of elimination and dispersal.

Parameters	Value
Population size	10
Length of E_coli	8 bytes
Maximum generation	10
Unit step size	.001
Elimination Percentage	10

Step2: Random initialization of each bacterium in population

After initialization of different variables, next step is to randomly initialize each bacterium within the search area p. Such that each value assigned to bacterium indicates the set of possible threshold values. Then original image I is threshold using two threshold values i.e., $th_1$and $th_2$and threshold image I_b is obtained for further processing as shown in eq3.2

$$GT = \sum_{i=1}^{m} \sum_{j=1}^{n} |I_{(i,j)}|_{Canny}$$
$$I_b = (I > th_1 \ \& \ I < th_2)$$

Step3: Calculate Objective Function

In next step of the algorithm using ith bacterium value as threshold canny edge detector is used to detect edges given by equation. Objective function is calculated for each ith bacterium using equation 3.4. Any ith bacteria at the jth chemo tactic, kth reproduction and lth elimination stage is θ^i(j,k,l) and its corresponding objective value is given by J(i,j,k,l). Objective function in this equation is developed as the ratio of edges obtained by canny using each bacterium value as threshold to the total number of edges found in the given image.

$I_{ci} = |I_{bi}|_{Canny\ (k_i)}$

Where k_i is the threshold value or the randomly assigned value to ith bacterium.

J = $\sum_{i=1}^{m-1}(I_{c(i)}|\ Z)$

Where m = Total number of threshold values

And, Z = total edges found in original image

The algorithm works as follows:

1. Starting of the Elimination-dispersal loop
2. Starting of the Reproduction loop
3. Starting of the chemotaxis loop

a) i = 1, 2, ...S, calculate objective function J

b) J_{old} (i, j, k, l) is served as J_{old} so as to compare with other J_{new} values.

cθ^i (j+1,k,l) = θ^i(j,k,l) + C(i)

This results in a step size C(i) for ith bacterium in only one direction.

e) Calculate J_{new} (i, j+1, k, l)

d) Swim

Let m = 0 (counter for swim length)

While m<N_s

m=m+1;

If J_{old} < J_{new} then J_{old} = J_{new} (i, j+1, k, l)

This θ^i (j+1,k,l) is used to calculate J_{new} (i, j+1, k, l)

Else n = N_s

e) Go to the next bacterium (i+1) till all the bacteria undergo chemo taxis.

4. Chemotaxis

If j<N_c,, go to step 3 and continue chemotaxis since the life of bacteria is not over else go to the reproduction stage.

5. Reproduction

a) For the given k and l, and for each i = 1, 2, 3, S, J^i_{health} = $\sum_{j=1}^{N_c+1} J(i,j,k,l)$ be the health of ith bacterium. The bacteria are stored according to descending order of J^i_{health} .

b) The bacteria with the lowest J_J^i_{health} values die and other bacteria with highest values split and the copies that are made are placed at the same location as their parent.

6. If k < N_{re}, go to step 2 to start the next generation in the chemotactic loop else go to step 7.

7. Elimination - dispersal:

For i = 1, 2, S a random number (rand) is generated and if rand ≤ P_{ed}, then that bacterium gets eliminated and dispersed to a new random location, else the bacterium remains at its original location.

8. If l < N_{re} go to step 1 else stop.

3.1.6 Edge Detection using Canny Edge Detector

After selection of the fittest threshold value next step is to detect edges using this threshold value. Edge detector used in our proposed technique is canny edge detector. The canny edge detector is selected because of its advantages over other edge detection techniques. It is more widely considered as the standard algorithm for edge detection. Canny produces better edge detection results specially in noise conditions because it is not sensitive to noise. It is a multistage optimal edge detector whose main aim is to localize the edges means and it marks points as an edge in such a way that there is not much difference between the actual and marked edges. Secondly it takes care that there is no such situation occurs in which it doesn't respond and it doesn't miss points which are edges. The third thing is that it marks an edge only once means single response to existing edge in an image.

On the basis of above criteria it performs different steps such as:

It filters out useless data and noise, hence smoothes the image before detecting edges by using Gaussian filter. This step is involved in edge detection as a pre-processing step.

In next step the detector calculates the gradient at each point in the image. Generally Sobel is used for finding approximate gradient which helps in finding the edge by detecting the largest increase from light to dark pixels and also direction or orientation.

Hence magnitude of gradient is given by:

$$|G| = \sqrt{(G_x)^2 + (G_y)^2}$$

Direction of gradient is given by:

$$\theta = \arctan\left({G_y}/{G_x} \right)$$

In the next step canny edge detector suppress the non-maxima pixels of edge. Such points where gradient is not maximum are removed and not considered as a part of edge. Therefore we iterate through each pixel and their gradient value as well as the orientation of gradient at each point is checked.

IV. COMPARISON WITH OTHER METHODS

4.1 Entropy Based Image Thresholding

Entropy is the basic concept used in information theory. It is the key measure of amount of information contained in a digit from the information source. It is defined in the context of a probabilistic model, Such that

$$I(X) = \log\left[{}^{1}/_{P(X)}\right] = -\log[P(X)] \quad ..... \quad (1)$$

Where I(X) is the self-information of event X which occurs with the probability P(X). Thus it is clear from above equation that amount of self information is inversely related to its probability. So If an event has probability P(X) = 1, we get no information from the occurrence of the event, i.e. I(X) = 0.

Shannon entropy is the measure of mean information of information source and is denoted by S(X). Let set of source symbols with values in the set of integers {i = 0............k} where k=255

And probabilities is denoted by p where p = {p_1, p_2, p_3..... p_k} such that $\sum_{i=0}^{k} p_i = 1$, $0 \le p_i \le 1$ condition must satisfied by this set of probabilities. Thus Shanon entropy is given by:

$$S(Z) = -\sum_{i=1}^{k} p \log p_i \quad ..... (2)$$

Where pi is the probability that the observation at any site s = (x, y) ϵ S and k the total no of states. Shannon entropy has an additive property such that if a system can decomposed into two independent systems A and B then S(A + B) = S(A) +S(B) exists, so it has been shown to be restricted to the Boltzmann-Gibbs-Shannon (BGS) statistics.

The Tsallis entropy is a generalization of the standard Boltzmann–Gibbs entropy and is a very useful extension for non-extensive physical systems. For set of all source probabilities P_i

With condition$\sum_{i=0}^{k} p_i = 1$, 0 Tsallis entropy is given by:

$$S_q(p_i) = \frac{1}{q-1}(1-\sum_{i=1}^{k} p_i^q) \quad .... \; ... \; ..... \quad (3)$$

Where real parameter q is called the entropic index that describes the degree of non-extensibility. In the limit a q→1, the usual Boltzmann-Gibbs entropy is recovered, such that

$$S_{BG} = S_1(p) = -k \sum_{i=1}^{k} P_i \log p_i \quad ..... \; ... \; ... \; ... \quad (4)$$

Tsallis entropy has a non-extensive property for statistical independent systems, defined by the following rule

$$S_q(A+B) = S_q(A) + S_q(B) + (1-q).S_q(A).S_q(B) \quad ... \; .... \; .... \; .... \quad (5)$$

Consider a digital image f(x, y) such that x ϵ {1,2,………M} and y ϵ {1,2,………N} of size M×N histogram be h(i) with f as the amplitude (brightness) of the image at the real coordinate position (x, y). Let f(x,y) represents gray value of the pixel located at the point (x, y). For the sake of convenience, we denote the set of all gray levels {0,1,2,3,4,5,……. ,255} as G(global threshold). Such threshold value is selected by using gray level histogram of the image method. The optimal threshold value denoted by t* is obtained by optimizing the function results from gray level distribution of the image [6].

Let t be the threshold value and B = {b0, b1} be a pair of binary gray levels such that b belongs to G and with value 0 and 1 respectively. If otherwise, the result of thresholding an image function f(x, y) at gray level t is a binary function ft(x, y) such that $f_t(x,y)=b_0$ if f_t $(x,y) \le t$ otherwise, f_t $(x, y)=b_1$

As we have observed, we have defined information strictly in terms of the probabilities of events. Therefore, let us suppose that we have a set of probabilities (a probability distribution) P = {p_1, p_2, p_3.............................p_k} for image with k gray-levels. From this we derive two probability distributions, one for the object (class A) and the other for the background (class B), given by:

$$P_A : \frac{P_1}{P_A}, \frac{P_2}{P_A}, ..., \frac{P_t}{P_A},$$

$$P_B : \frac{P_{i+1}}{P_B}, \frac{P_{i+2}}{P_B}, ... \frac{P_K}{P_B} \text{ And}$$

Where

$$P_A = \sum_{i=1}^{t} P_i, \; P_B = \sum_{i=t+1}^{k} P_i \quad ..... \quad (6)$$

The Tsallis entropy of order q for each distribution is defined as:

$$S_q^A(t) = \frac{1}{q-1}(1 - \sum_{i=1}^{k} P_A^q) \text{ And}$$

$$S_q^B(t) = \frac{1}{q-1}(1 - \sum_{i=t+1}^{k} P_{B)}^q \quad ..... \quad (7)$$

Allowing the pseudo – additive property as defined in equation (2), Tsallis entropy is formulated as the sum each entropy. Thus Tsallis entropy is dependent upon the threshold value t for both background and foreground we try to maximize the sum of entropies of two classes i.e., background class and object class. The luminance level t that maximizes the function is considered to be the optimum threshold value, when Tsallis entropy $S_q(t)$ is maximized [7]. Hence optimum threshold value is

$$t^* = Arg\, max_{t=G} S_q^A(t) + S_q^B(t) + (1-q).S_q^A(t).S_q^B(t)] \quad .... \; ...... \; ...... \quad (8)$$

The threshold value equals to the same value found by Shannon's method when q→1.

4.2 OSTU'S METHOD

This method is considered as a variation of iterative thresholding method. It is a clustering method based upon maximizing the between class variance. It is based upon defining well defined threshold classes as clusters with clusters lying tightly adjacent to each other and there is a minimal overlap.
The within class variance can be defined as summation of variance of each class as [4]:

$$\sigma^2_{within}(T) = n_B(T)\sigma^2_B(T) + n_0(T)\sigma^2_0(T) \quad .... \quad ...(9)$$

$n_B(T)$ = sum of pixel values in background

$n_0(T)$ = sum of pixel values in foreground

$\sigma^2_B(T)$ = variance of pixels in background

$\sigma^2_0(T)$ = variance of pixels in foreground

From the proved methods the between class variance is given as [4]:

$$\sigma^2_{between}(T) = n_B(T)n_0(T)[\mu_B(T) - \mu_0(T)]^2 \quad ..(10)$$

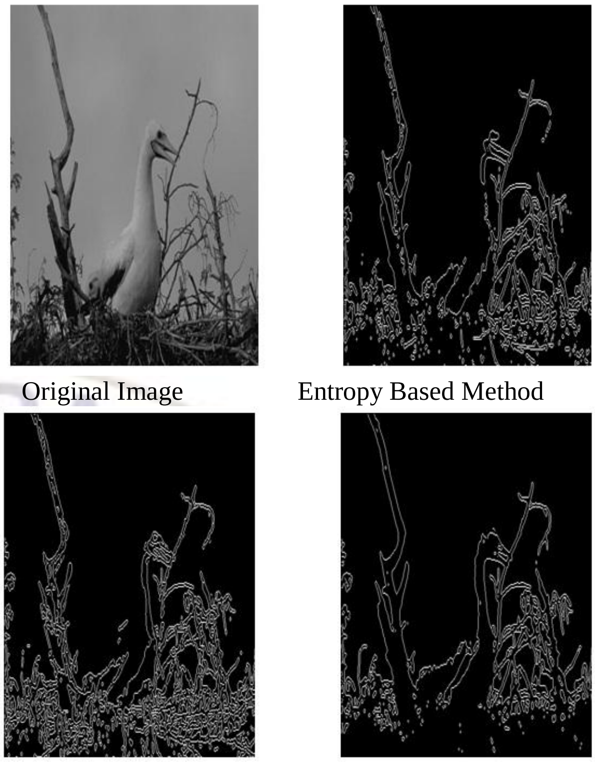
Where μ_B and μ_0 are cluster means

To find an optimal threshold value is to maximize the between class variance and this is relatively an easier calculation than calculation within class variance which would be minimized by maximizing between class variance. This method also works by assuming an image with a bimodal histogram [2].

V. EXPERIMENTAL RESULTS

The proposed edge detector along with other methods such as edge detector based on Entropy and Otsu method are simulated on MATLAB R2008b. The experiments are performed on 20 test images (taken from the database of images Berkeley Segmentation Dataset, MATLAB test images and this database contains a set of 200 gray or color images of natural scenes along with boundary detection results based on human perception) and simulation results on a few natural images (i.e., Test1, Test2, Test3 and Test4) are shown in figure 3 to 6. The complex images such as Test1, Test2, Test3, Test4, Test5 and Test6 images are taken instead of Lena and Cameraman images to comprehensively evaluate the performance of the proposed edge detector. To analyse the performance of various edge detectors peak Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE) and Agreement Indicator are taken as performance measures.
Results of performance evaluation of various Edge Detectors have been shown qualitatively as well as quantitatively.

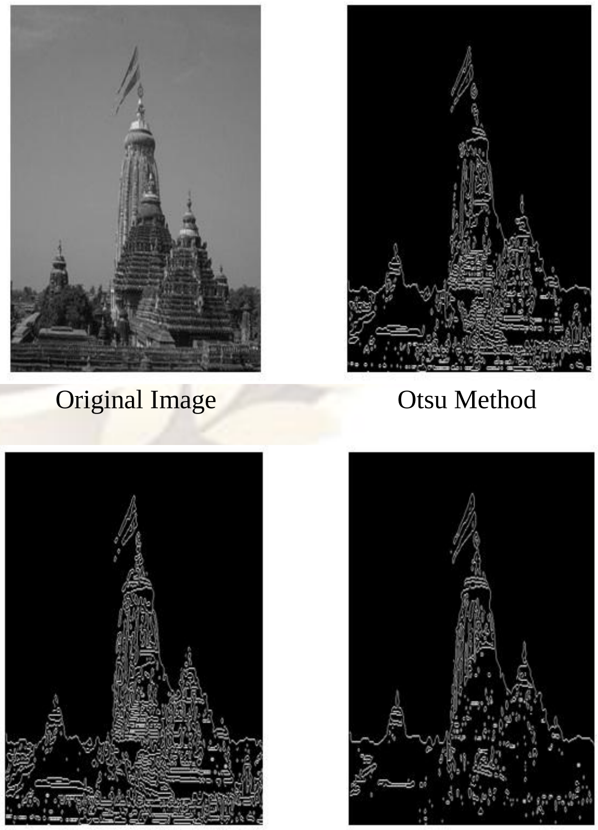
5.1 Visual Results of Different Edge Detectors on different Test Images



Original Image — Entropy Based Method

Otsu method — Proposed method

Figure3: Qualitative performance of various Methods for Test1 image



Original Image — Otsu Method

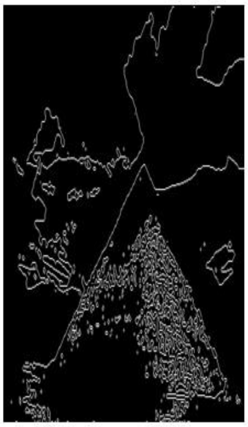
Entropy Based Method — Proposed method

Figure4: Qualitative performance of various Methods for Test2 image


Original Image Otsu Method


Entropy Based Method Proposed method

Figure5: Qualitative performance of various Methods for Test3 image

Original Image Otsu Method


Entropy based Method Proposed method

Figure6: Qualitative performance of various Methods for Test4 image

Several experiments are tested to demonstrate the performance of the proposed edge detection algorithm. The image size and total number of generations for BFOA run are shown in Table2.

TABLE2: Image Size and Number of Generations For BFO Runs

Image Name	Size	Iterations
Test1.jpg	481×321	10
Test2.jpg	481×321	10
Test3.jpg	481×321	10
Test4.jpg	481×321	10

5.2 Quantitative Results of Various Edge Detectors on Selected Test Images

TABLE3: Agreement Indicators of the Tested Images

Image Name	Entropy Based Method	Otsu Method	Proposed Method
Test1	0.0434	0.0610	0.1325
Test2	0.0427	0.0506	0.1034
Test3	0.0286	0.0514	0.0798
Test4	0.0497	0.0503	0.0798

TABLE4: PSNR Values of the Tested Images

Image Name	Entropy Based Method	Otsu's Method	Proposed Method
Test1	18.4931	18.5237	18.5249
Test2	20.3144	20.3470	20.3471
Test3	19.4122	19.4393	19.4395
Test4	22.3871	21.4158	22.4187

TABLE5: MSE Values of the Tested Images

Image Name	Entropy Based Method	Otsu's Method	Proposed Method
Test1	920.4272	920.6864	919.9711
Test2	605.0253	605.0320	604.8440
Test3	745.6463	745.6763	744.4997
Test4	375.5628	375.7497	375.2891

5.3 Discussions

The quantitative performance of all the Edge Detectors in terms of different quantitative metrics on various test images is shown in Table 3 to Table 5. To evaluate the performance of our proposed edge detector natural images are taken from Berkeley Segmentation Dataset. The simulation results of different edge detection methods are shown in Fig 3 to Fig 8. From the above results following can be observed.

- Detected edges are perfectly connected at junctions in case of proposed method but in case of rest of methods no or very few edges are connected at junctions.
- Qualitative results demonstrate that edges are on the exact position as in original image.
- In addition proposed method provides better performance in terms of all metrics such as higher PSNR value, Lower MSE and higher Agreement Indicator.

In summary, from qualitative and quantitative parameters it has been concluded that proposed technique is accurate and provides better results in all the tests.

VI. CONCLUSION

A combination of K-Means clustering and Bacteria Foraging optimization algorithm based edge detector has been developed and implemented to produce better edge detection results than traditional detectors. Such that BFO algorithm is used to choose the optimal threshold value from multiple threshold values for canny edge detector to produce more accurate and satisfactory edge detection results. Results of the proposed edge detector and the edge detectors based on Entropy and Otsu's method were presented both qualitatively and quantitatively.

From the qualitative and quantitative results of edge detectors on natural images, It is concluded that the proposed edge detector clearly outperform all the other methods in study. It has been observed that proposed method obtain optimal results but with high computational effort. So the BFO based technique produced more accurate results than other studied techniques. Finally obtained results indicate that the proposed method have a high effectiveness on a large category of image applications. From the results of proposed edge detection on natural images following observations were made by visual inspection.

- It has been observed from final results that the detected edges are perfectly connected at junctions in proposed method.
- Qualitative results demonstrate that edges are on the exact position as in original image.

Also from quantitative parameter values it has been observed that proposed edge detector performs better than other edge detectors in study. Above observations demonstrates that proposed method is capable of achieving satisfactory results. The effectiveness of method is checked by simulating the test images on MATLAB. The proposed method provides the superior edge detection results to existing edge detection techniques based on the Entropy & Otsu's Thresholding method. In addition, the proposed method provides better quality in visualization by obtaining maximum PSNR value.

REFERENCES

[1] Alaa Sheta,Malik S. Braik ,Sultan Aljahdali "Genetic Algorithms: A Tool for Image Segmentation" International Conference on Multimedia Computing and Systems (ICMCS) IEEE, May 2012.

[2] B. Poornima., Y. Ramadevi , T. Sridevi ,"Threshold based edge detection algorithm", IACSIT International Journal of Engineering and Technology, Vol. 3, No. 4,August 2011

[3] B.Sathya, R.Manavalan "Image Segmentation by Clustering Methods: Performance Analysis" International Journal of Computer Applications, Volume 29, No.11, September 2011

[4] Jain.S.Petrou,"Thresholding", Sections 3.2.1, 3.2.2, December, 2009

[5] Manisha Kaushal et. al. "Adaptive Thresholding for Edge Detection in Gray Scale Images" International Journal of Engineering Science and Technology, Vol. 2(6), pp. 2077-2082, 2010

[6] Mohamed A.EI-Sayed "A New Algorithm Based Entropic Threshold for Edge Detection in Images" IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 1, pp. 71-78 September 2011

[7] Mohamed. A. El-Sayed, S. Abdel-Khalek, and Eman Abdel-Aziz "Study of Efficient Technique Based On 2D Tsallis Entropy For Image Thresholding" *International Journal on Computer Science and Engineering (IJCSE)*, Vol. 3, Issue 9, pp. 3125-3138 ,September 2011

[8] Muthukrishnan.R and M.Radha "Edge Detection Techniques For Image Segmentation "International Journal of Computer Science & Information Technology (IJCSIT), Vol. 3, No. 6, 2011

[9] Nitin Kumar Jhankal, Dipak Adhyaru "Bacterial foraging optimization algorithm: A Derivative free technique" Institute Of Technology, Nirma University, Ahmedabad,Gujarat IEEE , pp. 1-4, December,2011.

[10] P.D. Sathya and R. Kayalvizhi "Optimum Multilevel Image Thresholding Based on Tsallis Entropy Method with Bacterial Foraging Algorithm" *IJCSI International Journal of Computer Science Issues*, Vol. 7, Issue 5, pp. 336-343, September 2010.

[11] Ping-Sung Liao, Tse-Sheng Chen and Pau-ChooChung "A Fast Algorithm for Multilevel Thresholding" *Journel of Information Science And Engineering*, 17(5), pp. 713-727 2001.

[12] Saiful Islam, Dr. Majidul Ahmed " Implementation of Image Segmentation for Natural Images using Clustering Methods" *International Journal of Emerging Technology and Advanced Engineering*, Volume 3, Issue 3, pp. 175-180, March 2013.