An Edge Detection Approach for Flame and Fire Images in Image Processing Using Bacterial Foraging Algorithm

Mr. Ashish Chaudhary
Research Scholar
Computer Science & Engineering
Galgotia's University
Greater Noida, India

Mr. Sahil Raheja
Asst. Professor
Computer Science & Engineering
Galgotia's University
Greater Noida, India

Abstract: **To characterize the boundaries and edge detection is not an easy task in image processing and it become very difficult when image is noisy. Edges are significant local or sharp change of intensity in an image and it occur on the boundary between two different regions in an image. In this paper we will work on color images of fire or flame. The determination of the fire and flame edges is the process of identifying a boundary between the two region where there is thermo chemical reaction and those without in the image ,this boundary is also known as edge of the flame or fire. Many conventional edge-detection methods have been tested to identify flame edges in color image, but the results achieved have been not so good when we use these methods on color image, they were very disappointing. Some research works related to flame and fire edge detection were reported for different applications; however these methods do not emphasize the continuity and clarity of the flame and fire edges. A Bacterial Foraging method is proposed to define flame and fire edges clearly and continuously. This algorithm worked on color image and produced the better edge result in a flame/fire image. In this modified bacterial foraging algorithm, a set of bacteria's are randomly initialized to image pixel then they will try to search their nutrients which are edges in this case. This whole process will be carried out in 3 steps: Chemotaxis, Reproduction and elimination & dispersal step. Swarming step is avoided to make the algorithm simpler and to reduce the complexity of the algorithm. As the algorithm works for high luminous flame images, it is enhanced to effectively find the edges for low luminous images also. Results for different flame images and video frames proved the effectiveness and robustness of the proposed algorithm.**

Keywords:—**Edge detection, feature extraction, flame, Noise, Bacterial Foraging Algorithm, Filter, Image, Edge analysis, Image processing, monitoring.**

I. INTRODUCTION

Edge detection is an important field in image processing. It can be applied to many applications like as segmentation of object, feature extraction, and identification of objects in an image. Edge detection can refer to the procedure of locating sharp discontinuities with in image. It is basically a fundamental tool used in most image processing applications to acquire information seen from the frames currently being a precursor step to feature extraction and object segmentation. This system detects boundaries of an object and limits between objects and the background in the image. One of the most essential applications is edge detection for image segmentation. Edge is basically a boundary between two homogeneous regions. As one of the important steps in flame and fire image processing, edge detection is mainly the predecessor and lays a foundation for other processing. There are many reasons which force to identify flame edges. First thing is that the flame or fire edges form a basis for the quantitative determination of a range of flame characteristic parameters like it's shape, it's size, location, and it's stability. Second thing is the definition of flame edges can reduce the amount of data processing and filter out unwanted information such as background noise within the image. In other words, edge detection can preserve the important structural properties of the flame and meanwhile shorten the processing time. And third, we can use edge detection to segment a group of flames. This is helpful for monitoring the multiple flames in some industrial instrument where a multi-burner system is used. Adkins [3] developed a software tool to analyze fire images, with which one can use a mouse to trace the flame edge. It is a normal edge-detection method, but it shows its importance and usefulness of the flame or fire edge detection. Such as fire detection or shape reconstruction in a complex background, or they have some limitations. Some flame edges detected are not clear and continuous, or do not match the actual flame shape. For the purpose of getting the flame's actual size and shape and the geometric characteristics, it is necessary to get the clear, continuous and, where possible, closed edge of the flame. In this paper, several conventional edge-detection methods have been tested to assess their effectiveness in flame edge identification. For the adjustment many parameters in the use of these methods, the results were still unsatisfactory when we these on color image.

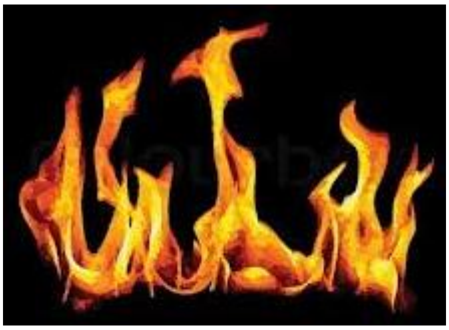

Figure1 (a)

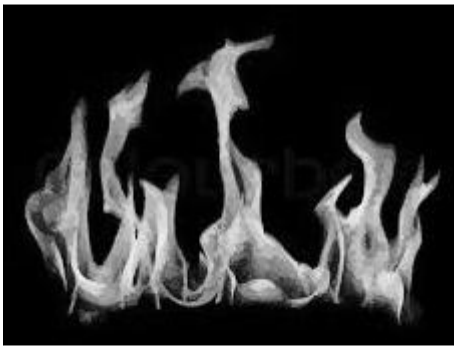

Figure1(b)

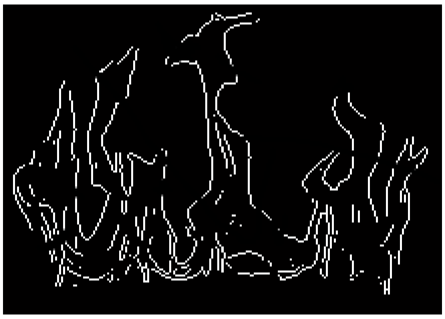

Figure1(c)

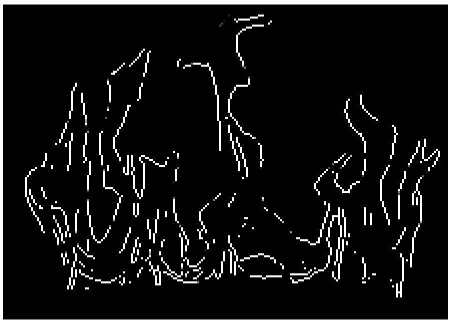

Figure1(d)

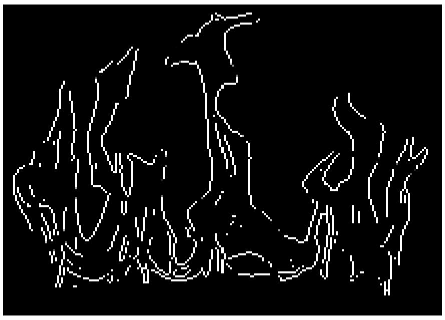

Figure1(e)

Figure 1. Results using the common edge-detection methods (a) Original image. (b) grayscale image (c) Sobel method. (c) Prewitt method. (d) Roberts method. (e) Canny method

.

According, to new computing algorithm is proposed in this paper to process an image and to identify flame/fire edges. The expected flame edge or boundaries should be continuous, and uninterrupted edge. However, as the results have shown, the edges we get using these methods are often disconnected, not clear and fragmented [Figure1(c)–(f)]; many methods can only detect a part of the flame edge or wrongly identify small and false edges that are obviously not the edges of the main flame. The results have therefore suggested that it is not always possible to get the ideal edges from real-life images of moderate complexity.

II TRADITIONAL METHODS OF EDGE DETECTION AND THEIR RESULTS

Edge detection is analyzed by using the mathematical representation of first order & second order derivatives. The first order finds the gradient & second order gives the magnitude of the edge. Several conventional edge-detection methods have been developed for finding the boundary of any object in image processing. The edge-detection methods that have been published may be grouped into two categories according to the computation of image gradients. These are the first-order and second-order derivatives. In the first category, edges are detected through computing a measure of edge strength with a first-order derivative expression. Roberts, Prewitt, and Sobel operators are the gradient based edge-detection [8]. The Canny edge-detection algorithm is an improved method using the Sobel operator. It is a powerful edge-detection method. In the second category, edges are detected by searching a second-order derivative expression over the image. In the present research, many common edge-detection methods have been used with suitable parameters to process typical flame images. Fig. 1(a)–(f) shows examples of results obtained by the conventional edge-detection methods along with the original image. The expected flame edge should be continuous, clear, and uninterrupted edge. However, in the results that we get using these methods are mostly disconnected and fragmented edges. Many methods only identify a part of the flame edge or wrongly identify small edges that are obviously not the edges of the main flame. The results have therefore suggested that it is not always possible to obtain ideal edges from complex real-life flame images. There are some other algorithms proposed for the flame/fire edge detection for various applications. Although we are unable to test all these methods, the published results have shown that these methods are not suitable for our purpose. It is therefore desirable to develop a dedicated edge-detection method for flame/fire image processing.

Sobel Operator

The Sobel operator is a discrete differentiation operator which computes the gradient at each point in an image for the intensity changes. The sobel operator consists of a pair of 3×3 convolution masks as shown in Figure 2.One mask is simply the other rotated by 90°.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Figure 2: Masks of Sobel Operator

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel, one mask for each of the two perpendicular orientations. The mask can be applied separately to the input image, for getting separate measurements of the gradient component in each orientation (call these

Gx and *Gy*). These can then be combined together to find the correct magnitude of the gradient at each point and the orientation of that gradient [3]. The gradient magnitude is given by:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

Typically, an approximate magnitude is computed using:

$$|G| = |Gx| + |Gy|$$

which is much faster to compute. The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by: q = arctan(*Gy* /*Gx*)

Robert's cross operator:

The Roberts Cross operator performs a simple, quick to compute, 2-D spatial gradient measurement on an image. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point. The operator consists of a pair of 2×2 convolution kernels as shown in Figure . One kernel is simply the other rotated by 90°[4]. This is very similar to the Sobel operator.

+1	0
0	-1

Gx

0	+1
-1	0

Gy

Figure 3. Mask used by Robert operator

These masks are designed to respond maximally to edges running at 45° to the pixel grid and one mask for each of the two perpendicular orientations. The angle of orientation of the edge giving rise to the spatial gradient (relative to the pixel grid orientation) is given by:

q = arctan(*Gy* /*Gx*) − 3p / 4

Prewitt's operator:

Prewitt operator is similar to the Sobel operator and is used for detecting vertical and horizontal edges in images.

-1	0	+1
-1	0	+1
-1	0	+1

Gx

+1	+1	+1
0	0	0
-1	-1	-1

Gy

Figure4. Mask used by Prewitt's operator

Canny Edge Detection Algorithm

Canny: The Canny edge detection algorithm has the following basic steps:

1. Noise is filtered and image is smoothed using Gaussian filter.
2. Edge strength is found by computing the gradient magnitude and angle of gradient vector for edge direction.
3. Non-maxima suppression is applied to the gradient magnitude to trace move along the edge direction and

suppress those pixel values that are not considered edge and thus resulting in thinning of edge.

4. Final step is to use hysteresis and connectivity analysis to detect and connect edges. If threshold value for edge detection is kept too low or too high there can be problem of either false positive or false edges. Canny algorithm provides the solution for this problem by using two thresholds: A low threshold and a high threshold

III. PROPOSED WORK

In general, a flame region has a stronger luminance in comparison to its background. As the algorithm discussed in the previous section of this paper, old methods works for superfluous flames with high luminosity and unable to detect the edges of fire/flame images with low intensity and maximum amount of noise. So we proposed an algorithm which produced the better result for flame and fire image with low intensity. Bacterial Foraging Algorithm is modified here to extract edges from noisy image. In this modified bacterial foraging algorithm, a set of bacteria's are randomly initialized to image pixel then they will try to search their nutrients which are edges in this case.

This whole process will be carried out in 3 steps: Chemotaxis, Reproduction and elimination & dispersal step. Swarming step is avoided to make the algorithm simpler and to reduce the complexity of the algorithm. Following are the steps to be followed to extract edges using modified bacterial foraging algorithm.

Search Space

In this case, since we are searching for edges in image so whole image is the search space. Image is 2-dimensional search space consisting of x-y coordinates of pixels. Image size is limited by its x y dimension so
the search space is finite in our case.

Defining the nutrient function

So what will be the nutrients for the bacteria. The nutrient function is based on the difference in intensities of a bacteria placed at a pixel position with respect to pixels in the surroundings. Here we have considered neighborhood pixels which are coming in window size of 3*3. Each bacterium will try to find out edges in these neighborhood pixels depending on the difference of those with their surrounding pixels.

4.3.4 Criteria for choosing the edge pixel

For a pixel to be considered as edge pixel, it should have average intensity difference greater than a threshold value with its surrounding pixels that are 8-neighborhoods in our case. Intensity difference will be taken as component wise difference in color images. This is explained with the following example:

P1	P2	P3
P4	P	P5
P6	P7	P8

Figure5: Apixel with its neighborhood pixel

Suppose we want to find whether pixel P with coordinates (x, y) is edge pixel or not, we will calculate component wise intensity difference with its neighborhood pixels in following way.

ID1=(|(x,y,1)-(x-1,y-1,1)|+|(x,y,1)–(x-1,y,1)|+|(x,y,1)–(x-1,y+1,1)|+|(x,y,1)–(x,y-1,1)|+|(x,y,1)–(x,y+1,1)|+ |(x,y,1)–(x+1,y-1,1)|+ |(x,y,1)–(x+1,y,1)|+ |(x,y,1)–(x+1,y+1,1|)/8;

ID2=(|(x,y,2)-(x-1,y-1,2)|+|(x,y,2)–(x-1,y,2)|+|(x,y,2)–(x-1,y+1,2)|+|(x,y,2)–(x,y-1,2)|+|(x,y,2)–(x,y+1,2)|+ |(x,y,2)–(x+1,y-1,2)|+ |(x,y,2)–(x+1,y,2)|+ |(x,y,2)–(x+1,y+1,2|)/8;

ID3=(|(x,y,3)-(x-1,y-1,3)|+|(x,y,3)–(x-1,y,3)|+|(x,y,3)–(x-1,y+1,3)|+|(x,y,3)–(x,y-1,3)|+|(x,y,3)–(x,y+1,3)|+ |(x,y,3)–(x+1,y-1,3)|+ |(x,y,3)–(x+1,y,3)|+ |(x,y,3)–(x+1,y+1,3|)/8;

Here ID1, ID2, ID3 refers to differences corresponding to red component, green component, blue component respectively.

These component wise differences will be added say in total variable. Now if this total is greater than a threshold value, then the pixel in interest that is P will be called as edge pixel otherwise not.

Chemotaxis

In this step, bacteria will move to next pixel position or it will change its direction (tumble). In this case, bacteria will move on to edge pixels i.e. its nutrient in its surrounding. It will check in its surrounding if it found with some edge pixels it will move on to the pixel that is having maximum intensity difference otherwise it will be dispersed to some other location.

Consider the following example:

P 1	P 2	P 3
P 4	P	P 5
P 6	P 7	P 8

Figure 6

Figure6: A pixel P with its neighborhood pixels in window size 3*3.

Suppose we are at pixel P and we have checked in its surrounding and we found with two pixels that satisfy the criteria to be an edge pixel then we will move to the pixel that has maximum intensity difference from its neighborhood. Suppose P3 has total intensity difference 40, P6 has 37 and threshold value is 30. Then we will move on to P3 leaving P6. So in next chemotactic step bacterium location will be at P3 and then it will check for edge pixel in its surrounding. In this way, bacteria will keep moving on to edge pixels thus making an edge-map.

Reproduction

In Reproduction step, least healthy bacteria die and gets replicated by healthiest bacteria. To model this situation in our case bacteria health is defined in terms of number of edge pixel it is getting in its surrounding. We will keep storing that number as health of bacteria. After N_c number of chemotaxis steps, the bacteria will be divided into two halves depending on S_r (reproduction ratio). In our case S_r is S/2. The last half of bacteria will gets replicated by first half bacteria positions.

Elimination and Dispersal

A bacterium will be assigned probability depending on number of pixels it found in its surrounding. A bacterium which is very bad in health will have high probability to disperse to some other random pixel position. Then it will again start in search of edge pixels from there.

IV. RESULT AND DISCUSSIONS

After implementing the algorithm as described, many of flame images were processed using this algorithm so as to check its correctness. We observed that the developed algorithm can successfully detect clear edges of the flame and remove the unnecessary edges, which traditional edge-detection methods cannot achieve.We tested the proposed method on color images that are corrupted with noise.Figure 7 where percentage of salt and pepper noise is 10%. It is not possible to remove noise completely, but we have tried to eliminate it as much as possible. Edge Detection is done on noisy images using bacteria foraging. The output depends on the selection on the parameters used in bacteria foraging that are No. of initial bacteria N_b, no. of chemotactic steps N_c, no. of elimination and dispersal event N_{ed}, no. of reproduction steps N_{re} threshold value T, reproduction ratio S_r. The result of the proposed edge detection method is being compared against the result of three other edge detection methods: Canny, Sobel and Prewitt.

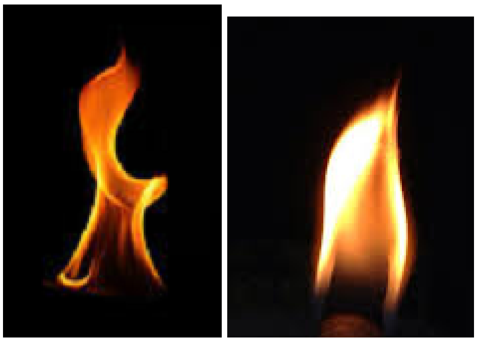

Figure7(a):Original Image

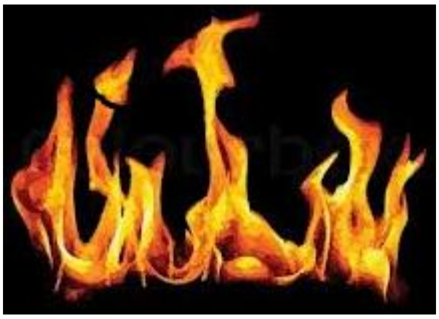

Figure7(b):OriginalImage

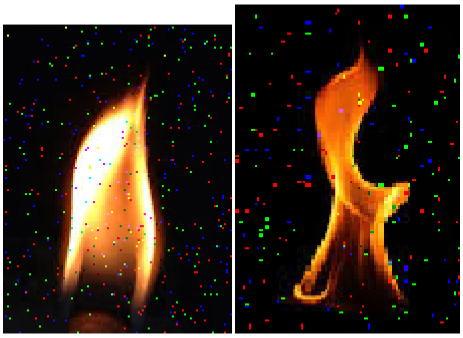

Figure7 (c) noisy image of 7(a)

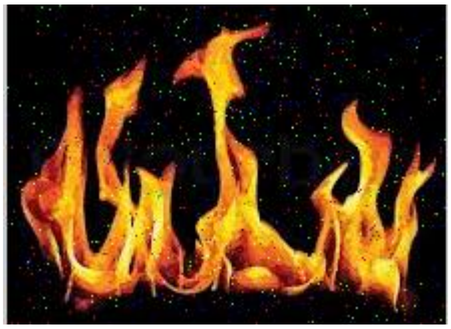

Figure7 (d) noisy image of 7 (b)

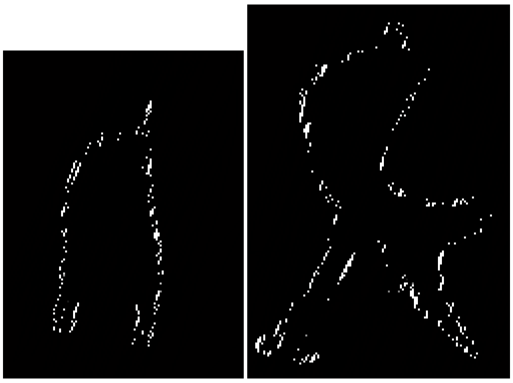

Figure 8(a) Nb=20 (Nb= number of bacteria)

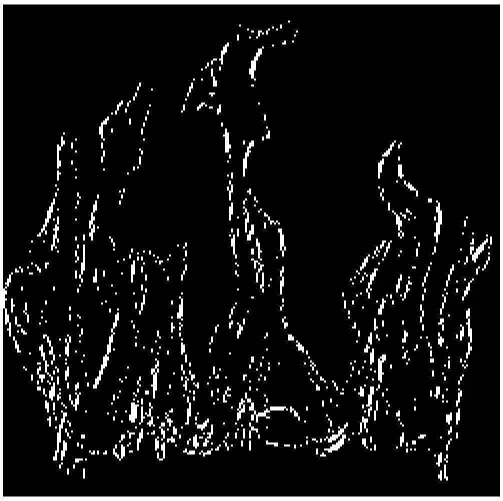

Figure 8(b) Nb=20 (Nb= number of bacteria)

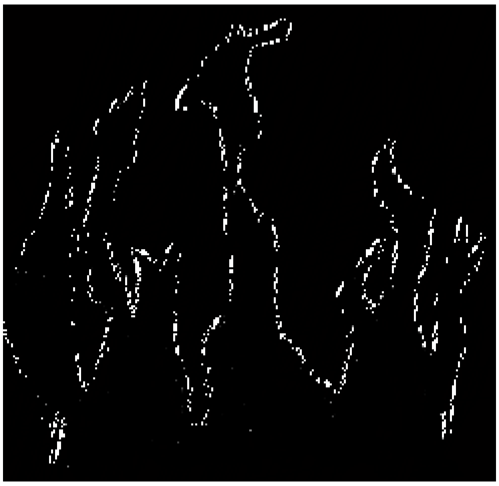

Figure 8(c) Nb=30 (Nb= number of Bacteria.

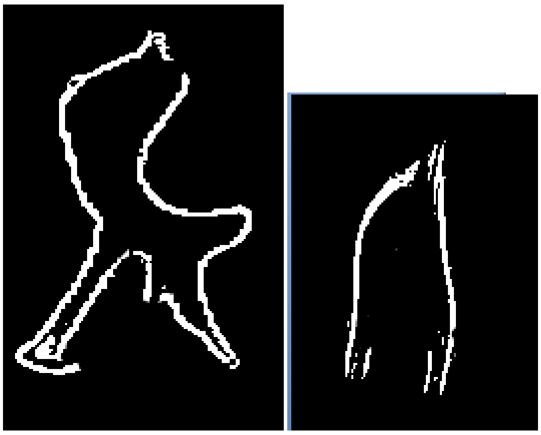

Figure9(a) Nb=40 (Nb= number of bacteria)

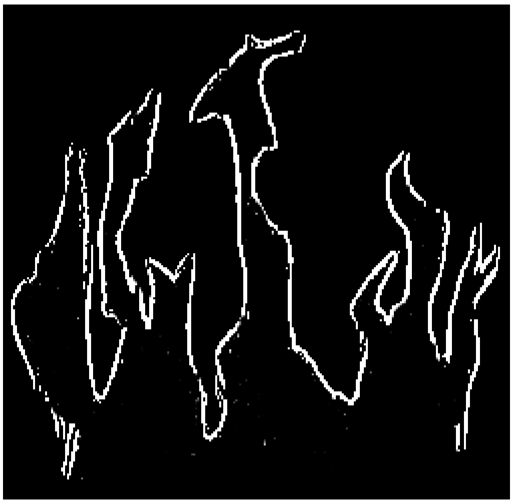

Figure9 (b) Nb=40 (Nb= number of bacteria)

Figure 8 and 9 are the Result of proposed approach for image taking different number of initial bacteria.

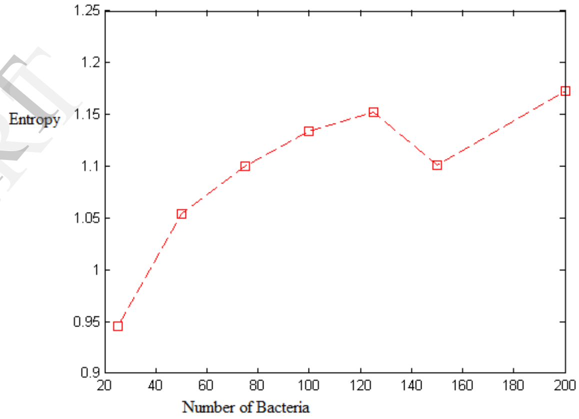

Figure 10(a): Entropy versus number of Bacteria (N_b)

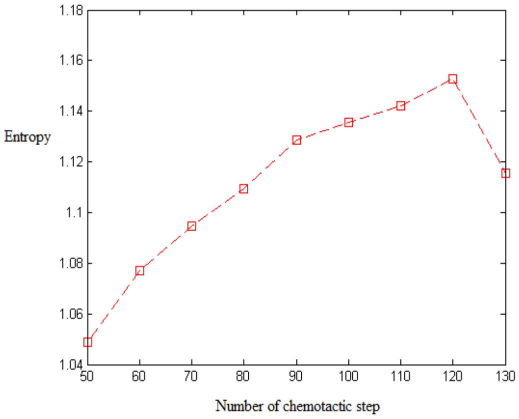

Fig10(b): Entropy versus number ofchemotactic steps (N_c)

Fig. 10(a) shows the variation of number of bacteria N_b with respect to Entropy measure. It means that if the no of bacteria is more than we get much clear edge. Figure 10(b) shows the variation of number of chemotactic steps (N_c) with respect to Entropy measure. Increasing the number of chemotactic step means increasing the lifetime of the bacteria. If bacteria are allowed to live longer then it will also try to explore more number of edges and thus requires longer time. To make bacteria to complete its life in a sufficient time with providing good number of edges and less noisy pixels, the number of chemotactic steps taken are 80. Fig 7,8,9 shows result of proposed approach for different images and for different number of chemotactic steps. The proposed algorithm makes it much easier to distinguish the flame region from the background. We can use the proposed edge-detection algorithm to improve the performance and characterize the geometric features of flames or fires image and, consequently, establish their relationship with combustion conditions such as air/fuel inputs and emissions. To assess the anti-noise effect of the algorithm, different types of noise are randomly added to the flame images before the images are processed.

V. CONCLUSION

After the flame characteristics are analyzed, a new flame edge-detection method has been developed and evaluated in comparison with conventional methods. As per the existing algorithm for detection of fire & flame edges giving much clear & continuous edges in colored image, when compared to the other traditional methods that has been used for processing & analysis of fire & flames captured in laboratories. The conventional algorithm works for the superfluous flames with high intensity & contrast depicted on the gray scale image with the results given. Hence our algorithm overcomes this constraint by identification of clear & continuous edges of any colored fire/flame images for processing in its respective identified area as also been depicted by the results. Experimental results have demonstrated that the algorithm developed is effective in identifying the edges of irregular flames in colored image. The advantage of this method is that the flame and fire edges detected are clear and continuous. The work presented was aimed for the processing of flame and fire images captured in laboratories. Further work is required to evaluate the performance of the algorithm in real-life flame detection scenarios.

VI. REFERENCES

[1] T. Celik, H. Demirel, H. Ozkaramanli and M.Uyguroglu, "Fire detection using statistical color modelin video sequences," Journal of Visual Communication and Image Representation, 18, 176-185 (2007).

[2] Qiu T, Y. Yan, and G. Lu, "A new edge detection algorithm for flame image processing," in Proc. IEEE I2MTC, Hangzhou, China, May 10–12, 2011, pp. 281–284.

[3] C. W. Adkins, "Users guide for fire image analysis system-Version 5.0: A Tool for measuring fire behavior characteristics," U.S. Dept. Agric., Forest Service, Southern Res. Station, Asheville, NC, Gen. Tech. Rep. SE93, 1995.

[4]Kevin M. Passino, "Biomimicry of Bacterial Foraging", IEEE Control Systems Magazine IEEE, 2002.

[5]Om Prakash Verma, Madasu Hanmandlu, Puneet Kumar, Sidharth Chhabra, Akhil Jindal, "A novel bacterial foraging technique for edge detection", ScienceDirect Pattern Recognition Letters 32 (2011) 1187–1196.

[6] B. C. Ko, K. H. Cheong, and J. Y. Nam, "Fire detection based on vision sensor and support vector machines," *Fire Safety J.*, vol. 44, no. 3, pp. 322–329, Apr. 2009

[7] Y. Yan, T. Qiu, G. Lu, M. M. Hossain, and G. Gilabertet, "Recent advances in 3D flame tomography," in *Proc. 6thWorld Congr. Ind. Process Tomogr.*, Beijing, China, Sep. 2010, pp. 1530–1539..

[8] B. U. Toreyin and A. E. Cetin, "Online detection of fire in video," in Proc. IEEE Conf. CVPR, 2007, pp. 1–5.

[9] D. Ziou and S. Tabbone, "Edge detection techniques: An overview," *Int. J. Pattern Recognit. Image Anal.*, vol. 8, no. 4, pp. 537–559, 1998.