International Journal of Modern Communication Technologies & Research (IJMCTR)
ISSN: 2321-0850, Volume-7, Issue-9, September 2019

An Edge Detection Filter by Using Firefly with ANN Algorithm

Anju Choudhry, Mr. Akhilesh Pandey

Abstract — Edge detection is a procedure of identifying and detecting sharpened disruption in an image. The disruption are unexpected changes in pixel intensity gray level value. The conventional approach of edge detection include convolving the image with an operator (2- D filter) which is produce to be sensitive to noise. Edge detector is a combination of crucial local image processing method to locate sharpened modification in the intensity value. Edge detection is an essential technique in different type of image processing applications like motion analysis, pattern recognition, object recognition, medical image processing etc. In this paper comparison is shown among the edge detection techniques under different conditions showing advantages and disadvantages of these algorithms. The Firefly algorithm with Artificial Neural Network has some characteristics that make it suitable for solving optimization problem, like higher converging speed and less computation rate.

Index Terms — Edge detection, Edge detection filter, Firefly algorithm, Grey level image.

I. INTRODUCTION

The Edge detection in digital images is one of the basic problems in the field of image processing [1] [4]. It is very important in image processing and computer vision. In common applications of machine vision and pattern recognition, discriminating the objects from their background is one of the important tasks. Edges determine location of objects of interest therefore their proper detection during low level processing conditions accuracy of image analysis performed in further stages of the processing.

An important property of the edge detection method is its ability to extract the accurate edge line with good orientation in the considered image, and much literature on edge detection has been published in the past two decades. However, there is not yet any general performance index to judge the performance of the edge detection methods.

Artificial Neural Network (ANN) has been successfully used in many computer-aided system of imaging [2]. Moreover, image segmentation and edge detection remains an important problem for all imaging applications, in which any computer assisted system will require edge detection and other techniques, such as watershed, snake modeling, region growing, and contour detection [2]. Additionally, ANN has been utilized to exploit its learning capability and training mechanisms to

Anju Choudhry, M.Tech Scholar, Department of Computer Science & Engineering, Kanpur Institute of Technology, Kanpur, India.
Akhilesh Pandey, Assistant Professor , Department of Computer Science & Engineering, Kanpur Institute of Technology, Kanpur, India.Classify images into content consistent regions to complete segmentation as well as edge detection [3]. The performance of edge detection method is always judged subjectively because different users have different requirements for the same image.

Although many algorithms have been conducted on edge detection for images, however higher computational cost and subjective image quality could be further improved. Therefore, the objective of this paper is to develop a fast ANN based edge detection algorithm for different images. Here, features based on horizontal, vertical, and diagonal differences are designed. Then, Canny edge detector will be taken as the training output. Lastly, we will get number of hidden layers and output threshold as optimized parameters. The edge detection image will be analyzed its quality subjectively and computational time compared to other traditional algorithms, such as Canny, Sobel, Prewitt, Roberts, and Laplacian of Gaussian (LoG). Results showed that the proposed algorithm provided better image quality while it has faster processing time around three times.

In this swarm intelligence optimization technique we assume that solution of an optimization problem can be represented as a firefly which glows proportionally to its quality in a considered problem setting.

Therefore, every brighter firefly attracts its partners, that prepare the search space being explored efficiently.

Some characteristics of the firefly algorithm with Artificial Neural Network that make it suitable for solving optimization problems are higher converging rate and flexibility.

In this paper, we proposed a novel edge detection method based on firefly algorithm with Artificial Neural Network and compared results by some other method like Canny [6], Sobel [7], Log [8] and Prewitt [9].

The rest of the paper is organized as follows: Section 2 describes the Firefly Algorithm, Artificial Neural network. Section 3 gives a detailed description of the proposed method. The experimental results are discussed in Section 4, and Section 5 concludes the paper.

II. Firefly Algorithm (FA)

Fireflies are one of the most special creatures in nature. Most of fireflies makes shortened and rhythmical flashes and have different flashing performance and these flashes are used by Fireflies for communication and attracting the probable prey. YANG used this behavior of fireflies and introduced Firefly Algorithm in 2008 [5].
In Firefly algorithm, there are three idealized rules:

1) All fireflies are unisex. Therefore, only single firefly will be attracted to other fireflies unconcerned of their sex; 2) Attractiveness is proportional to the brightness of the fireflies. So, for any two flashing fireflies, the less bright firefly will move towards the brighter one. The attractiveness is proportional to the brightness and both of them decreases as their distance increases. If there is no brighter firefly than a specific one, it will move aimlessly; **3)** the brightness of a firefly is influenced or resolved by the landscape of the objective function. For a maximization problem, the brightness can commonly be proportional to the value of the objective function [5].

The pseudo code for these three rules can be represented as **Fig.1** Firefly algorithm.

Initialize algorithm parameters:
*Max Gen: the maximum number of generations Objective function of f(**x**), where **x**=(x1,.,xd)T*
*Generate initial population of fireflies or **x**i (i=1, 2,. , n)*
*Define light intensity of **I**i at **x**i via f (**x**i)*
While *(t < Max Gen)*
For *i = 1 to n (all n fireflies);*
For *j=1 to n (all n fireflies)*
If *(Ij > Ii), move firefly i towards j;*
end if *Attractiveness varies with distance r via Exp [-γr2]; Evaluate new solutions and update light intensity;*
End for j;
End for i ;

Rank the fireflies and find the current best;

End while;

Post process results and visualization*;*

End procedure;

Figure1. Pseudo Code of the FA.

In FANN there are two important issues for variation of light intensity and the formulation of the attractiveness. For simplicity, it's assumed that the attractiveness of a firefly is determined by its brightness which associated with the objective function of the optimization problem. Since a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, we can now define the attractiveness of a firefly by:

$$B(r) = \beta_0 \, e^{-\gamma r^2} \quad (1)$$

where, β_0 is the attractiveness at r = 0 and is the light absorption coefficient at the source. It should be noted that the r_i,j which is described by equation 2, is the Cartesian distance between any two fireflies i and j at xi and X, where, x_i and x_j are the spatial coordinate of the fireflies i and j, respectively.

$$r_{i,j} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2} \quad (2)$$

The movement of a Firefly i, which is attracted to another more attractive Firefly j is determined by:

$$X_i = x_i + \beta_0 e^{-\gamma r_{i,j}^2}(x_i - x_j) + \alpha\left(rand - \frac{1}{2}\right) \quad (3)$$

Where, the second term is the attraction while the third term is randomization including randomization parameter α and the random number generator *rand which* its numbers are uniformly distributed in interval [0, 1]. For the most cases of implementations, $\beta_0 = 1$ and $\alpha \in [0, 1]$. The parameter γ characterizes the variation of the attractiveness and its value is important to determine the speed of the convergence and how the FA behaves. In the most applications, it typically varies from 0.01 to 100.

III. Artificial neural network

Neural network is a logical framework with multi-processing essential features that are linked through interconnection weights. The knowledge is presented by the interconnection weights, which are adjusted during the training phase. There are two typical categories of training process of a neural network: supervised learning and unsupervised learning. The first requires using both the input and the target values for each sample in the training set. The most common algorithm in this group is the back propagation, used in the Multi-Layer Perceptron (MLP), but it also includes most of the training methods for recurrent neural networks, time delay neural networks, and thrust basis networks. Unsupervised learning is used when the target pattern is not completely known (Quiza & Davim, 2011).

IV. Proposed method

A set of connected pixels that lie on the boundary between two regions is known an edge. One of the popular methods to edge detection is using a filter. For example, The Laplacian Edge Detection procedure uses a 2-dimensional linear filter to approximate the 2'd order derivative of pixel values of the image and the Sobel edge detection procedure uses two 2-dimensional linear filters to process vertical and horizontal edges individually.

In this work, to find a proper edge filter, we use a 3x3 filter that must be optimized to meet these conditions:

a) Sum of elements in *W* must be equal to zero this way pixels located in a monotonic area will lead to *E* (.) = 0.

b) All of edge pixels in the training sample image including vertical, horizontal and orthogonal edges must be resulted in *E* (.) value greater than *T*.

So, in our problem search space is 9-D. Before introducing our fitness function, it is necessary to write about our train image. Our proposed sample image and corresponding edge map are shown in Fig.2 [10].

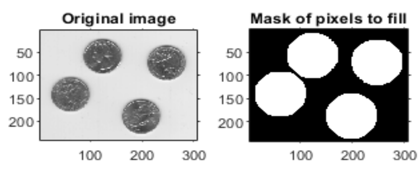

Figure 2: Sample image (right) and its edge map (left) used for Optimization of W by FA [9].

These two images are used as the input in the FA Training algorithm for optimizing the elements of *W*. The sample image has edges in all possible directions (vertical, horizontal and orthogonal). So it is expected that the trained filter can be successfully applied to other images for edge detection. To produce a binary edge map, a threshold value (*T*) is also necessary.

The threshold was specified by using a heuristic approach, based on visual inspection of the histogram.

The following algorithm can be used to obtain *T* automatically [1]:

1. Select an initial estimate for T.
2. Segment the image using *T*. This will produce two groups of pixels: *G1* consisting of all pixels with grey level values >*T* and *G2* consisting of pixels with values ≤ *T*.
3. Compute the average gray level values, μ1 and μ2 for the pixels in regions G1 and G2.
4. Compute a new threshold value:

For grayscale images, suppose that T_0=125.

The next step is introducing a proper fitness function to evaluation of possible solutions. The edge map of the input training image (Er) is available. So, the next step is a pixel by pixel comparison of (Er) and resulted edge map of the train image with resulted filter[10]. The number of differences between training image edge map and resulted edge map from output filer will shows the quality of output filter. The less difference is would indicate that the filter is suitable. This term counts unequal pixels in Er and Ew. Equation (6) will shows this rule,

As mentioned before, sum of elements in W must be zero. Considering these points, fitness function can be defined as:

$$T = \frac{1}{2}(\mu_1 + \mu_1) \quad (5)$$

5. Repeat steps 2 through 4 until the difference in *T* in Successive iterations is smaller than a predefined parameter T_0.

For grayscale images, suppose that T_0 =125.

As mentioned before, sum of elements in W must be zero. Considering these points, fitness function can be defined as:

$$F(w)_1 = \sum_{i,j}(E_r(i,j) \neq E_w(i,j)) \quad (6)$$

As mentioned before, sum of elements in W must be zero. Considering these points, fitness function can be defined as:

$$F(w) = abs(\sum_{k=1}^{9} w_k) + F(w)1 \quad (7)$$

Minimizing F will lead us to the optimum edge filter that can detect edges in all directions. The Procedure of proposed method can be determined as follows:

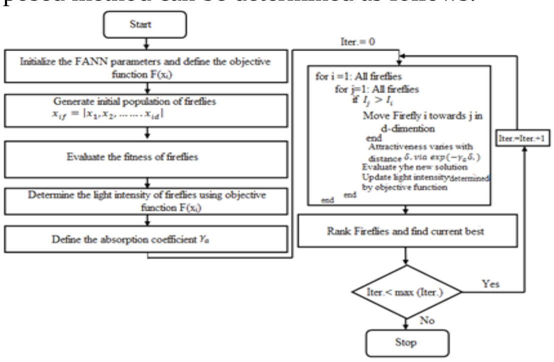

Figure 3: Flow chart of the Firefly Algorithm with ANN(FANN).

Step1: Initialize each Firefly with possible 9-D filter randomly and set FFA+ANN algorithm parameters.

Step2: for each firefly apply filter to train image and evaluate fitness function.

Step3: Move Firefly to a new position according to fitness value and evaluate new filters (the position of fireflies).

Step4: Judge whether the iteration number has been reached the maximum value. If so, deduction ends, otherwise, jump to the step 2.

Step5: finally, show edge detected image by applying optimal filter.

V. Experimental results

This is the result section, here we show the comparison among the performance of FFA and FFA+ANN on different images and comparisons between both algorithms are shown in the figures given below:

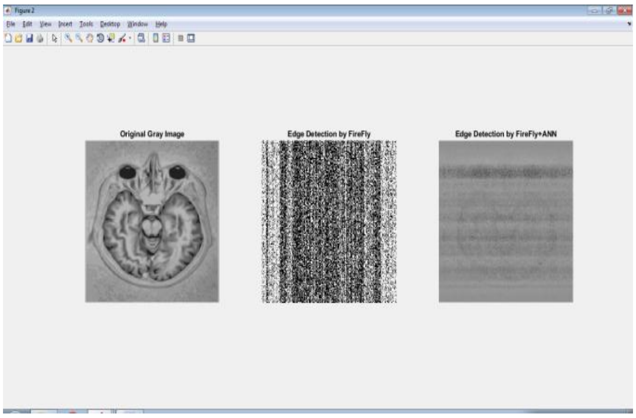

Figure 4 Original Gray Image of 'Brain', Edge detection by FFA and FFA+ANN.

Figure 2(a) shows the original gray image of 'Brain', (b) shows the edge detection by FFA and (c) shows the edge detection by proposed algorithm (FFA + ANN).

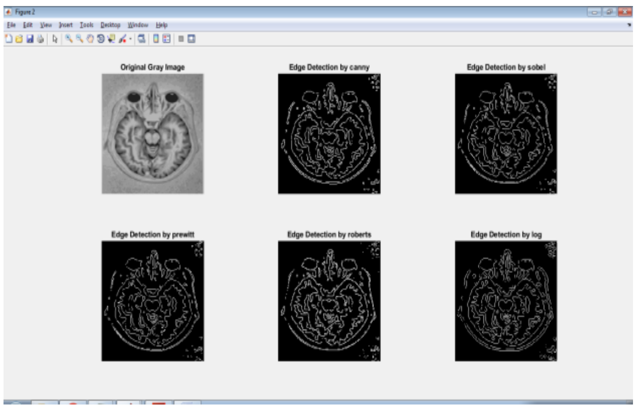

Figure 5 Comparison of our result and classical edge detection methods on a 'Brain' image.

Figure 3(a) shows the original gray image of 'Brain, (b) shows the edge detection by Canny edge detector (c) shows the edge detection by sobel edge detector (d) shows the edge detection by prewitt edge detector (e) shows the edge detection by roberts edge detector (d) shows the edge detection by log edge detector.

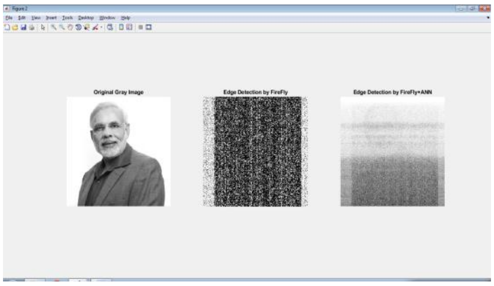

Figure 6 Original Gray Image of 'Modiji', Edge detection by FFA and FFA+ANN.

Figure 4 (a) shows the original gray image of 'Modiji', (b) shows the edge detection by FFA and (c) shows the edge detection by proposed algorithm (FFA + ANN).

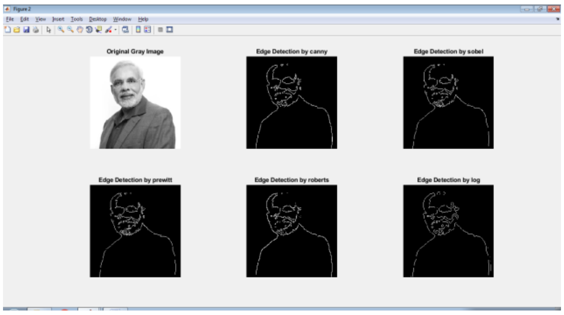

Figure 7 Comparison of our result and classical edge detection methods on a 'Modiji' image.

Figure 5(a) shows the original gray image of 'Modiji', (b) shows the edge detection by Canny edge detector (c) shows the edge detection by sobel edge detector (d) shows the edge detection by prewitt edge detector (e) shows the edge detection by roberts edge detector (d) shows the edge detection by log edge detector.

To objective evaluation of results we use peak signal to noise ratio method (PSNR), the higher value of the PSNR shows the better quality of the reconstructed image.

Method	PSNR		RMSE	
	Modiji	Brain	Modiji	Brain
Proposed work (FFA+ANN)	13.0530	15.3719	0.1411	0.0997
FireFly Algorithm	9.9468	11.1417	0.2249	0.1878
Canny	5.3202	5.9799	0.4502	0.4078
sobel	5.3150	5.9424	0.4506	0.4101
prewitt	5.3182	5.9407	0.4503	0.4102
roberts	5.3283	5.9598	0.4497	0.4090
log	5.3349	6.0273	0.4492	0.4049

Table 1: Comparison of proposed method with FireFly Algorithm, 'Canny', 'Sobel', 'Log','Prewitt' methods based on PSNR and RMSE criteria.

$$PSNR = 10\log_{10}\frac{(L-1)^2}{\frac{1}{MN}\sum_{r=0}^{M-1}\sum_{c=0}^{N-1}[E(r,c)-o(r,c)]^2} \qquad (4)$$

Where E(r,c) is the original image , o(r,c) is the resulted image and L is the number of gray level equal to 256 and [M,N] are the number of rows and cols of images.

And root mean square error (rmse) the lower value of the rmse shows the better quality of the reconstructed image and it can shows as follows:

$$\text{RMSE}=\sqrt{\frac{1}{MN}\sum_{r=0}^{M-1}\sum_{c=0}^{N-1}[E(r,c)-o(r,c)]^2} \qquad (5)$$

The evaluation of resulted image with PSNR and RMSE method showed in tables 1. The evaluation results show the resulted edge detected images are better in most cases. And it's proved the ability of proposed method for edge detection.

VI. Conclusion

In this work, we proposed a new filter FFA+ANN for edge detection of gray level images by using firefly algorithm. In the proposed method, we use a simple image and its edge map to design new filter. The proposed filter applies to different images and evaluated with different criteria. The obtained result and subjective and objective evaluation shows the efficiency of proposed filter to edge detection.

VII. References

1). R. C. Gonzalez, R. E. Woods, ''*Digital Image Processing*'' PrenticeHall, Upper Saddle River, New Jersey 2001.

2). Jiang, P. Trundle, and J. Ren, "Medical image analysis with artificial neural networks," Computerized, Medical Imaging and Graphics, vol. 34, pp. 617-631, 2010.

[3] C.-Y. Chang, "Contextual-based Hopfield neural network for medical image edge detection," Optical Engineering, vol. 45, pp. 037006-037006-9, 2006.

4). T. Young, J. J. Gerbrands, L. J. van Vliet, *Fundamentals of Image Processing*'' The Netherlands at the Delft University of Technology, 1998.

5). Yang, X. S.: Nature-Inspired etaheuristic Algorithms. Luniver Press, 2008.

6). Canny,J.F." A computational approach to edge detection", *IEEE Trans.Pattern Analysis Mach*. Intell. 8 .pp, 679–698, 1986.

7). Sobel, I.E., "Camera Models and Machine Perception," Ph.D. dissertation, Stanford University, Palo Alto, California, 1970.

8). Huertas A., Medioni G., "Detection of intensity changes with sub- pixel accuracy using Laplacian-Gaussianmasks" *IEEE Transactions on Pattern Analysis and Machine Intelligence*. PAMI.

9). Alipoor, M., Imandoost, S,. Haddadnia, J, "Designing Edge Detection Filters Using Particle Swarm Optimization", proceeding of ICEE, 2010

10). Akhilesh Pandey,Roushan Singh "A Classification Of Multilingual Document" International Journal of Science and Research (IJSR) Vol 3, No.7 July, 2014.

11). Akhilesh Pandey, Shashi Kant Shukla "Classification of Devnagari Numerals using Multiple Classifier" International Journal of Computer Trends and Technology (IJCTT) Vol12,No.4 Jun2014.

Anju Choudhry, M.Tech Scholar, Department of Computer Science & Engineering, Kanpur Institute of Technology, Kanpur, India.
Akhilesh Pandey, Assistant Professor , Department of Computer Science & Engineering, Kanpur Institute of Technology, Kanpur, India.