

Hybrid Algorithm for Edge Detection using Fuzzy Inference System

Mohammed Y. Kamil
College of Sciences
AL–Mustansiriyah University
Baghdad, Iraq

ABSTRACT

This paper presents a novel edge detection algorithm based on fuzzy inference system. Fuzzy inference system has been designed for three inputs using Gaussian membership functions, one output using Triangle membership function that expresses whether the pixel under consideration is “Low”, “Medium” or “High” pixel. Rules base comprises of twenty-seven rules, it applied a Mamdani FIS by taking a mask over the image of 3x3 size. The results obtained are compared with Canny edge method and performance parameters used are PSNR of true to false edges. Experimental results shows that the proposed method gives higher PSNR values when compared with Canny edge detection algorithm under all states.

Keywords

Edge detection, Fuzzy logic, Fuzzy inference system, image processing.

1. INTRODUCTION

Edge detection is the process of distinguishing outlines of an object and boundaries between objects and the background in the image. It is a main tool used in various applications in image processing, computer vision and pattern recognition [1]. Previously edge detection methods like Sobel, Prewitt and Robert were used based on the calculation of the intensity gradient magnitude at each image pixel but these techniques suffer from some drawbacks such as edge displacement [2]. Canny technique is commonly used in edge detection due to its better result compare to other traditional techniques [3]. Canny is gradient based method which used Gaussian filter for smoothing an image. The interchange between detection and location of edge pixels make a difficult inaccuracy. By changing threshold values, edge detection rate increases, but the accuracy of edge locating decrease [4]. The procedure for canny edge detection method are follows: (1) Convolve image with a Gaussian kernel to get smooth image. (2) Apply first difference gradient operator to compute edge strength then edge magnitude and direction. (3) Apply non-maximal or critical suppression to the Gradient magnitude. (4) Apply threshold to the non-maximal suppression image.

The fuzzy image processing is effort for the design of new processing images techniques, many techniques have been suggested for fuzzy logic-based edge detection [5-8].

2. FUZZY INFERENCE SYSTEM

Fuzzy inference system (FIS) is the process of formulating the diagraming from a given input to an output using fuzzy logic. The diagraming then provides a basis from which decisions can be made, or patterns discerned. The process of fuzzy inference involves all of the portions related to membership functions, fuzzy logic operators, and if-then rules [9]. There

are two types of FIS that can be implemented: Mamdani type and Sugeno-type. These two types of inference systems vary somewhat in the method outputs are determined. Fuzzy inference systems have been successfully applied in fields such as data classification, expert systems, image processing, and computer vision. Using a graphical user inference (GUI) editors in the fuzzy logic toolbox, we can build the rules set and define membership functions. The editors are used to edit and view the membership functions and rules.

Fuzzy inference system (FIS) in image processing takes three main stages: image fuzzification, modification of membership values, and image defuzzification. The coding of image data (fuzzification) and decoding of the results (defuzzification) are steps that make possible to process images with fuzzy techniques. The main control of fuzzy image processing is in the middle step (modification of membership values).

In the last years, there are many researchers used Fuzzy inference system in image processing field. O. Mendoza et. al. proposed a new FIS Type-1 and Type-2 method is implemented for the detection of edges [10]. S. Arora and A. Kaur modified edge detection technique using fuzzy inference system using traditional operators in combination with fuzzy logic [11]. A. Borkar and M. Atulkar trying to detect the edges of images by using FIS in MATLAB environment without determining threshold value [12, 13]. M. Yadav and K. Kashyap used fuzzy inference system (FIS) to detect edges by change of fuzzy logic rules recognized method which will be capable to detect the edges of image [14]. J. Kaur and P. Sethi reports the evaluation of Fuzzy logic implemented on image, used to implement different functions. FIS has been designed for four inputs, one output that tells whether the pixel under is low, medium or high pixel. And fuzzy rule comprises of eight rules [15].

3. PROPOSED FUZZY INFERENCE SYSTEM

In this paper we have proposed a new FIS for detecting the edges in digital images based on the gradient and magnitude of the gradient for image. This begins by segmenting the images into regions using mask (3x3) (see Figure 1).

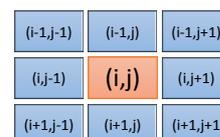


Fig. 1: Matrix mask 3x3.

A Mamdani FIS has been implemented used, with three inputs, one output and twenty-seven rules, based on the MATLAB Fuzzy Logic Toolbox 2012a. Figure 2 illustrated

the Fuzzy Inference Systems Editor. The FIS Editor displays common information about a fuzzy inference system include input variable on the left, and output variable on the right.

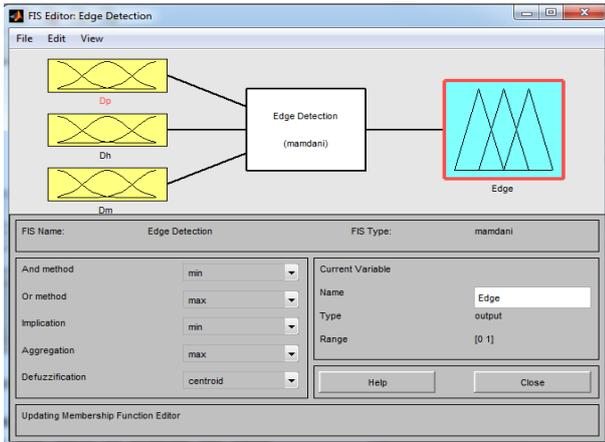


Fig. 2: FIS editor.

In FIS, D_p represented the gradient magnitude is calculated with $D_p = \sqrt{P_h^2 + P_v^2}$, where P_h and P_v produce from calculated through a bidimensional convolution operation for the original image by Prewitt operator [16]. Figure 3 shown the Prewitt filter measures for vertical edge component is calculated with kernel PF_x and the horizontal edge component is calculated with kernel PF_y .

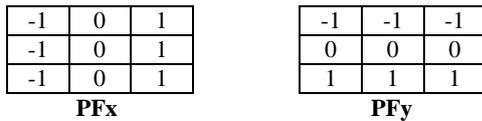


Fig. 3: Prewitt Operator.

If we define I_o as the original image, P_h and P_v (horizontal and vertical derivative approximations) are computed as $P_h = PF_x \otimes I_o$, and $P_v = PF_y \otimes I_o$, where \otimes represent the convolution operator.

The other two inputs, the high-pass filter (hpf) identifies the contrast of the image to guarantee the edge detection in relative low contrast regions. Also mean filter (mf) to detect image pixels belonging to regions of the input, are defined in Figure 4. Then D_h and D_m are separately defined as $D_h = hpf \otimes I_o$, and $D_m = mf \otimes I_o$.

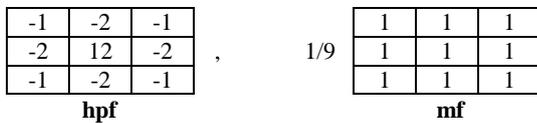


Fig. 4: High-pass filter (hpf) and mean filter (mf).

Three fuzzy sets are made up are implied variables “low”, “medium” and “high” in order to represent each variable’s intensities, the membership functions to all inputs for fuzzy sets are Gaussian, as shown in Figure 5. Figure 6 is the output membership functions variable Edge that also adjusted the range between 0 and 1, the fuzzy sets are Triangle function.

The twenty-seven fuzzy inference rules that allow to evaluate the input image, Figure 7 show surface viewer rule in FIS contain the rules are shown below:

1. IF (D_p is Low) and (D_h is Low) and (D_m is Low) THEN (Edge is High).

2. IF (D_p is Low) and (D_h is Medium) and (D_m is Low) THEN (Edge is High).
3. IF (D_p is Low) and (D_h is High) and (D_m is Low) THEN (Edge is High).
4. IF (D_p is Medium) and (D_h is Low) and (D_m is Low) THEN (Edge is High).
5. IF (D_p is Medium) and (D_h is Medium) and (D_m is Low) THEN (Edge is High).
6. IF (D_p is Medium) and (D_h is High) and (D_m is Low) THEN (Edge is Medium).
7. IF (D_p is High) and (D_h is Low) and (D_m is Low) THEN (Edge is High).
8. IF (D_p is High) and (D_h is Medium) and (D_m is Low) THEN (Edge is Low).
9. IF (D_p is High) and (D_h is High) and (D_m is Low) THEN (Edge is Low).
10. IF (D_p is Low) and (D_h is Low) and (D_m is Medium) THEN (Edge is High).
11. IF (D_p is Low) and (D_h is Medium) and (D_m is Medium) THEN (Edge is High).
12. IF (D_p is Low) and (D_h is High) and (D_m is Medium) THEN (Edge is Medium).
13. IF (D_p is Medium) and (D_h is Low) and (D_m is Medium) THEN (Edge is Medium).
14. IF (D_p is Medium) and (D_h is Medium) and (D_m is Medium) THEN (Edge is Medium).
15. IF (D_p is Medium) and (D_h is High) and (D_m is Medium) THEN (Edge is Low).
16. IF (D_p is High) and (D_h is Low) and (D_m is Medium) THEN (Edge is Medium).
17. IF (D_p is High) and (D_h is Medium) and (D_m is Medium) THEN (Edge is Low).
18. IF (D_p is High) and (D_h is High) and (D_m is Medium) THEN (Edge is Low).
19. IF (D_p is Low) and (D_h is Low) and (D_m is High) THEN (Edge is High).
20. IF (D_p is Low) and (D_h is Medium) and (D_m is High) THEN (Edge is Medium).
21. IF (D_p is Low) and (D_h is High) and (D_m is High) THEN (Edge is Medium).
22. IF (D_p is Medium) and (D_h is Low) and (D_m is High) THEN (Edge is Medium).
23. IF (D_p is Medium) and (D_h is Medium) and (D_m is High) THEN (Edge is Medium).
24. IF (D_p is Medium) and (D_h is High) and (D_m is High) THEN (Edge is Low).
25. IF (D_p is High) and (D_h is Low) and (D_m is High) THEN (Edge is Medium).
26. IF (D_p is High) and (D_h is Medium) and (D_m is High) THEN (Edge is Low).
27. IF (D_p is High) and (D_h is High) and (D_m is High) THEN (Edge is Low).

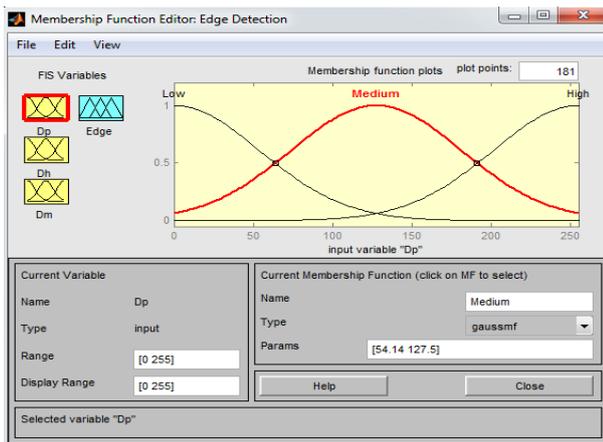


Fig. 5: Input membership function editor.

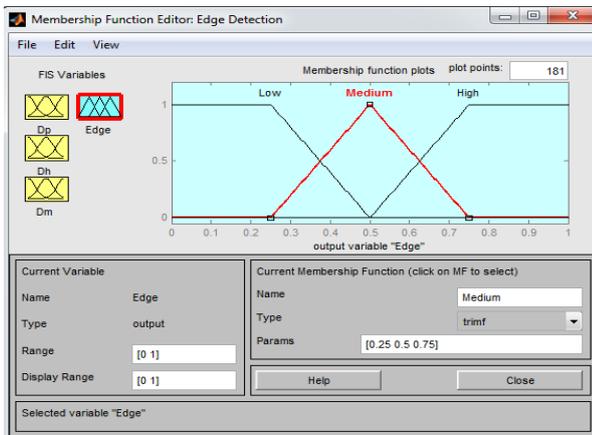


Fig. 6: Output membership function editor.

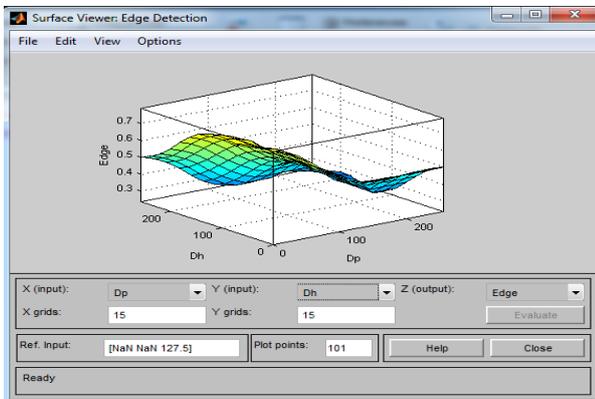


Fig. 7: Surface viewer window for rules.

4. RESULTS AND COMPARISON

In this part, the proposed algorithm is tested for various images (size, colour, formatting). The proposed method is compared with Canny method for Edge Detection, because it's most powerful edge detector [17]. In order to obtain an objective comparison of the images and performance of the proposed algorithms was evaluated in terms of the visual quality, we used histograms and peak signal-to-noise ratio (PSNR) to analysis of results [18]. The results of the tested images are shown in Figure 8 respectively. Table 1 shown the performance and comparative results, the proposed method is compared with Canny's algorithm. Results show that the proposed method gives higher PSNR than Canny's algorithm.

Table 1. Effect of PSNR on different test images.

Image	Size	PSNR Canny	PSNR Fuzzy
1 Peppers.png	512 x 384	8.708	8.768
2 Carpark.png	640 x 480	6.200	6.236
3 Westconcordaerial.png	369 x 394	4.910	4.948
4 Cell.tif	640 x 480	10.591	10.669
5 Mandi.tif	3039 x 2014	9.105	9.167
6 Spine.tif	490 x 367	20.985	21.170
7 Chess.bmp	675 x 507	5.614	5.653
8 Ibrahim.jpg	255 x 340	8.632	8.681
9 Trees.tif	350 x 258	11.689	11.756

5. CONCLUSIONS

The results shown that the proposed method gives higher PSNR values compared to Canny's method. It reduces false edge detection and identification of double edges are least, as well the marked pixel is nearer to the true edge. Fuzzy inference system presented the accuracy of the edge detection using the fuzzy IF-THEN rule over the other algorithms. This algorithm is appropriate for implementations in various parts of image processing such as medical imaging, face recognition and satellite image where boundaries of specific regions essential to be determined for further image analysis.

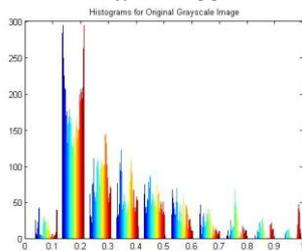
6. REFERENCES

- [1] Bezdek J. C. 1981. Pattern Recognition with Fuzzy Objective Functions. Plenum, New York.
- [2] J. Shah, N. Patel, H. Tandel, N. Sonia and G. Prajapati, "A Hybrid Approach For Edge Detection Using Fuzzy Logic And Canny Method", International Journal of Engineering Research & Technology, 2013, V. 2, Iss. 3.
- [3] J.F. Canny, "A computational approach to edge detection", IEEE Trans. on Pattern Analysis and Machine Intelligence, 1986, 8(6), pp. 679-698.
- [4] R. Wang, L. Gao, S. Yang, and Y. Liu, "An Edge detection method by combining fuzzy logic and neural networks," Machine Learning and Cybernetics, 2005, 7, 4539-4543, 18-21.
- [5] Tizhoosh H.R.2002. Fast fuzzy edge detection, Proceedings of Fuzzy Information Processing Society.
- [6] G. Mansoori and H. Eghbali., "Heuristic edge detection using fuzzy rule-based classifier", Journal of Intelligent and Fuzzy Systems, 2006, V. 17, N. 5, pp. 457-469.
- [7] Abdallah A. Alshennawy, and Ayman A. Aly. 2009. Edge Detection in Digital Images Using Fuzzy Logic Technique. World Academy of Science, Engineering and Technology 51, pp. 178-186.
- [8] D.O. Aborisade, "Novel Fuzzy logic Based Edge Detection Technique", International Journal of Advanced Science and Technology, 2011, Vol. 29.
- [9] M. Nachtgeael, D. Van derWeken, and E. E. Kerre, "Fuzzy techniques in image processing: Three case studies," Int. J. Comput. Anticipatory Syst., 2002, v. 12, pp. 89-104.
- [10] O. Mendoza, P. Melin, G. Sandoval, "Fuzzy Inference Systems Type-1 and Type-2 for Digital Images Edge Detection" Engineering Letters, 2007, 15:1, EL_15_1_7.
- [11] S. Arora and A. Kaur, "Modified Edge Detection Technique using Fuzzy Inference System", International Journal of Computer Applications, 2012, V.44, No.22.

- [12] A. Borkar and M. Atulkar, "Fuzzy Inference System for Image Processing", *International Journal of Advanced Research in Computer Engineering & Technology*, 2013, V. 2, Iss. 3.
- [13] A. Borkar and M. Atulkar, "Detection of Edges Using Fuzzy Inference System", *International Journal of Innovative Research in Computer and Communication Engineering*, 2013, V. 1, Iss. 1.
- [14] M. Yadav and K. Kashyap, "EDGE DETECTION THROUGH FUZZY INFERENCE SYSTEM", *International Journal of Engineering and Computer*, 2013, V. 2 Iss. 6, pp. 1855-1860.
- [15] J. Kaur and P. Sethi, "Evaluation of Fuzzy Inference System in Image Processing", *International Journal of Computer Applications*, 2013, V. 68, No.22.
- [16] Cristiano J. M., Adolfo B. 2001. Fuzzy Inference System Applied to Edge Detection in Digital Images. *Proceedings of the V Brazilian Conference on Neural Networks - V Congresso Brasileiro de Redes Neurais*, April, - PUC, Rio de Janeiro - RJ - Brazil, pp. 481-486.
- [17] G. Kumar and T. Jipeng, "Different Edge Detection Algorithms Comparison and Analysis on Handwritten Chinese Character Recognition", *International Journal of Computer Applications*, 2012, V. 47, No.17.
- [18] R. Zhu and Y. Wang, "Application of Improved Median Filter on Image Processing", *journal of computers*, 2012, V. 7, NO. 4.



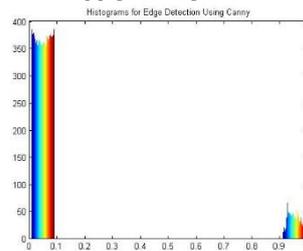
1a. original Peppers.



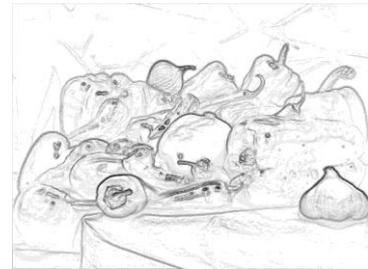
1d. histogram to 1a.



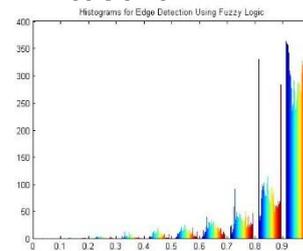
1b. apply Canny method.



1e. histogram to 1b.



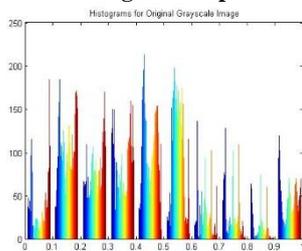
1c. apply proposed method.



1f. histogram to 1c.



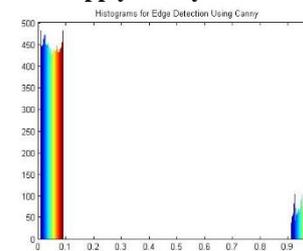
2a. original Carpark.



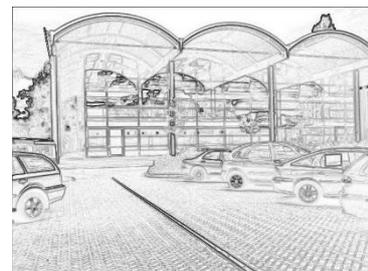
2d. histogram to 2a.



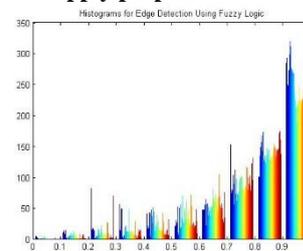
2b. apply Canny method.



2e. histogram to 2b.



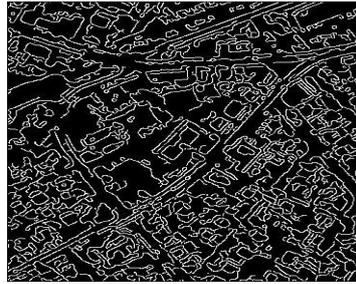
2c. apply proposed method.



2f. histogram to 2c.



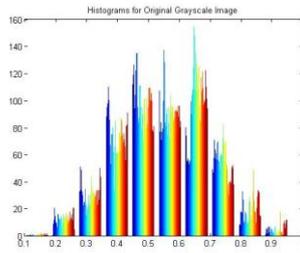
3a. original Westconcordaerial.



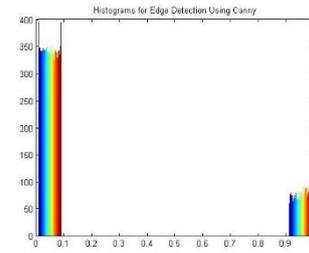
3b. apply Canny method.



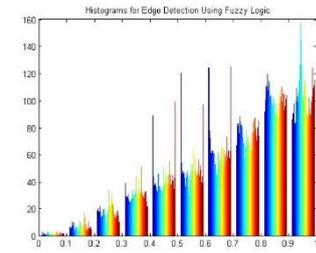
3c. apply proposed method.



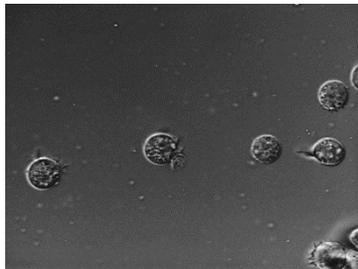
3d. histogram to 3a.



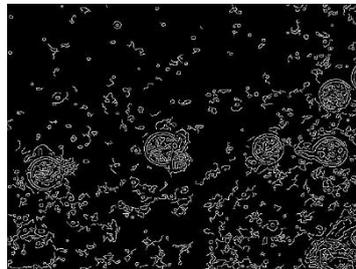
3e. histogram to 3b.



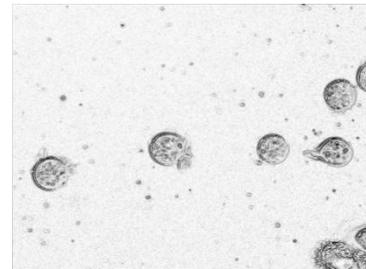
3f. histogram to 3c.



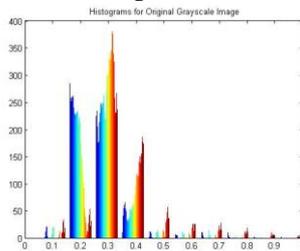
4a. original cell.



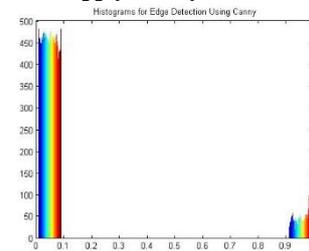
4b. apply Canny method.



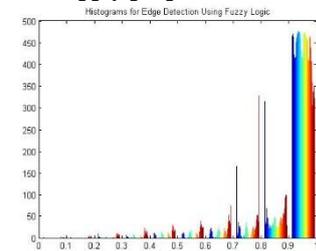
4c. apply proposed method.



4d. histogram to 4a.



4e. histogram to 4b.



4f. histogram to 4c.



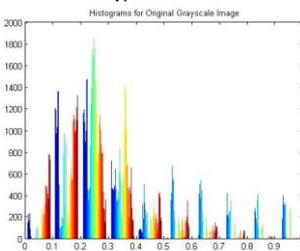
5a. original Mandi.



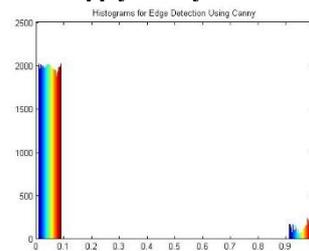
5b. apply Canny method.



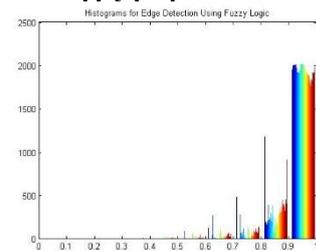
5c. apply proposed method.



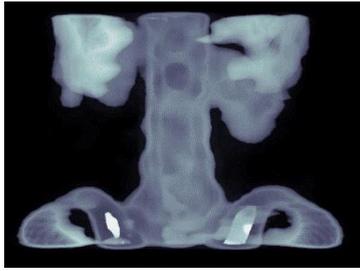
5d. histogram to 5a.



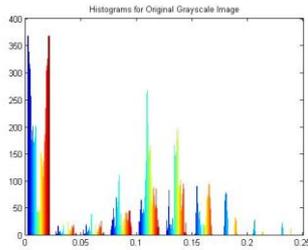
5e. histogram to 5b.



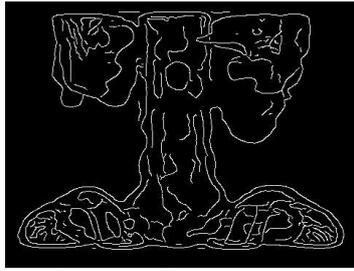
5f. histogram to 5c.



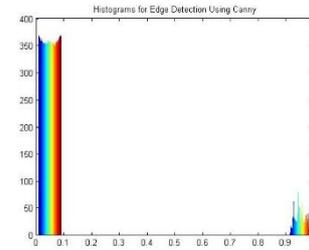
6a. original Spine.



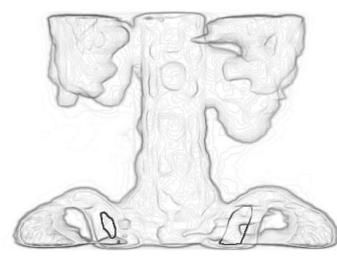
6d. histogram to 6a.



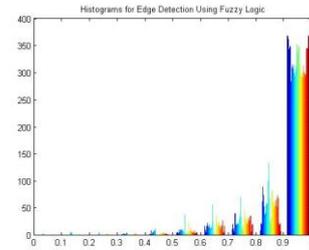
6b. apply Canny method.



6e. histogram to 6b.



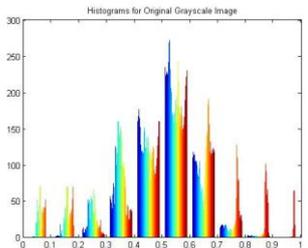
6c. apply proposed method.



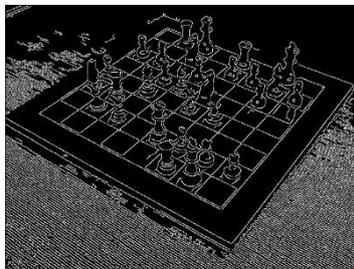
6f. histogram to 6c.



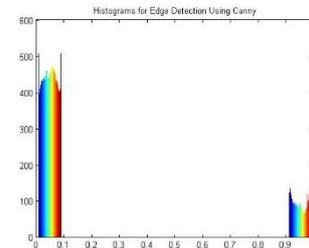
7a. original Chess.



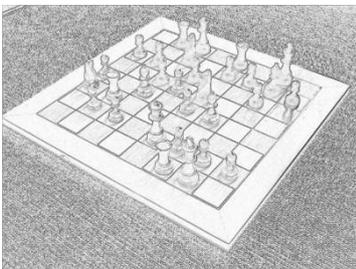
7d. histogram to 7a.



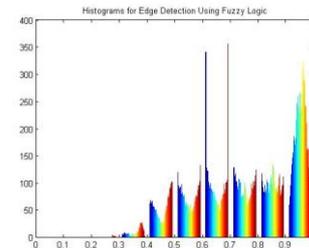
7b. apply Canny method.



7e. histogram to 7b.



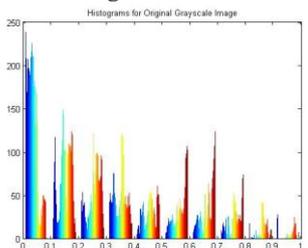
7c. apply proposed method.



7f. histogram to 7c.



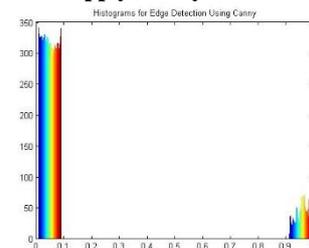
8a. original Ibrahim.



8d. histogram to 8a.



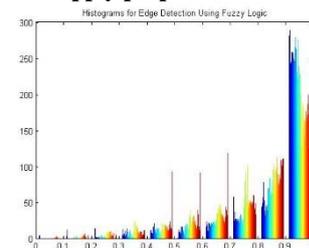
8b. apply Canny method.



8e. histogram to 8b.



8c. apply proposed method.



8f. histogram to 8c.

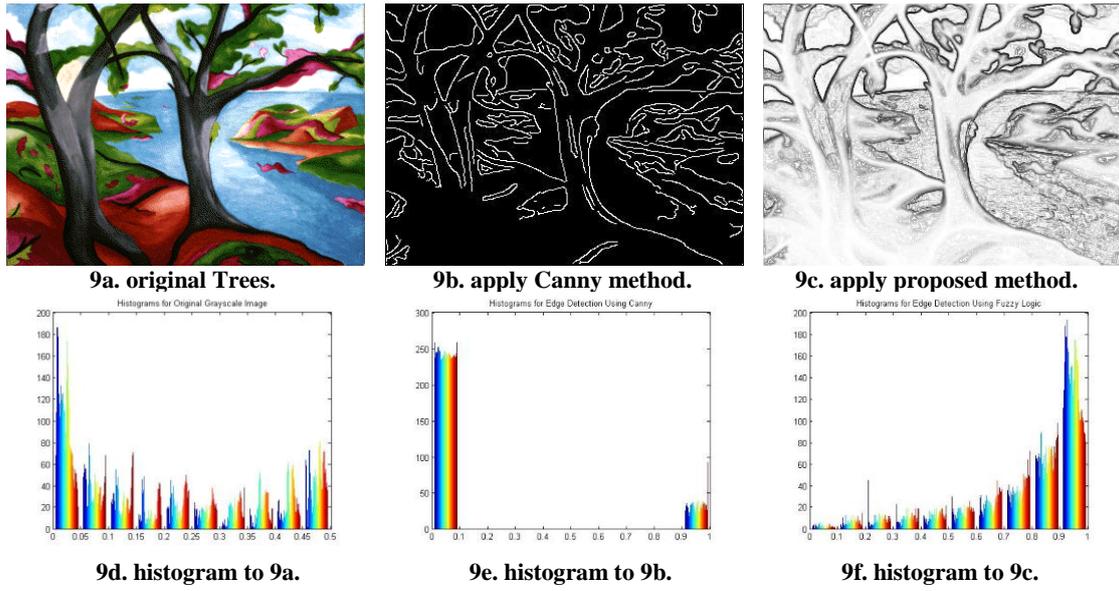


Fig. 8: Results for various images.