

Fuzzy Logic Based Image Edge Detection Algorithm in MATLAB

Er Kiranpreet Kaur
Lecturer, ECE Deptt
BBSBEC, Fatehgarh Sahib

Er Vikram Mutenja
Lecturer, ECE, Deptt
SBSCET, Firozpur

Er Inderjeet Singh Gill
Lecturer, ECE Deptt
SBSCET, Firozpur

ABSTRACT

This paper reports the implementation, in MATLAB environment, of a very simple but efficient fuzzy logic based algorithm to detect the edges of an input image by scanning it throughout using a 2*2 pixel window. Also, a Graphical User Interface (GUI) in MATLAB has been designed to aid the loading of the image, and to display the resultant image at different intermediate levels of processing. Threshold level for the image can be set from the slider control of GUI. Fuzzy inference system designed has four inputs, which corresponds to four pixels of instantaneous scanning matrix, one output that tells whether the pixel under consideration is “black”, “white” or “edge” pixel. Rule base comprises of sixteen rules, which classify the target pixel. Algorithm for the noise removal has been implemented at different levels of processing. The resultant image from FIS is subjected to first and second derivative to trace the edges of the image and for their further refinement. The results of the implemented algorithm has been compared with the standard edge detection algorithm such as ‘Canny’, ‘Sobel’, ‘Prewit’ and ‘Roberts’. Main feature of the algorithm is that it has been designed by the smallest possible mask i.e. 2*2 unlike 3*3 or bigger masks found in the literature.

Categories and Subject Descriptors

Image processing and computer vision, Electronic vision, Artificial Intelligence, MATLAB

General Terms

Algorithms, Design, Experimentation.

Keywords

Fuzzy logic, Edge detection, digital image processing, feature extraction, noise removal, electronic vision, computer vision, comparison

1. INTRODUCTION

Modern time is an era of technology in which we now believe in the vision based intelligence. Penetration of computers into each area of the market and living has forced the designers to add the capability to see and analyze and to innovate more and more into the area of electronic vision or image processing. At the level of computational intelligence for electronic vision, many of the algorithms have been developed to extract different types of features from the image such as edges, segments and lot many other types of image features. Edge detection is a terminology in electronic vision, particularly in the areas of feature extraction, to refer to algorithms which aim at identifying points in a digital image at which the image brightness changes sharply or more

formally has discontinuities. The goal of edge detection is to locate the pixels in the image that correspond to the edges of the objects seen in the image. This is usually done with a first and/or second derivative measurement following by a comparison with threshold which marks the pixel as either belonging to an edge or not. The result is a binary image which contains only the detected edge pixels. The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. Discontinuities in image brightness are likely to correspond to discontinuities in depth, discontinuities in surface orientation, changes in material properties or variations in scene illumination. Shashank Mathur and Anil Ahlawat, presented a fuzzy relative pixel value algorithm for edge detection by checking the relative pixel values in 3*3 pixels mask for scanning of image using the windowing technique, which is subjected to a set of fuzzy conditions for the comparison of pixel values with adjacent pixels to check the pixel magnitude gradient in the window. However their technique was not rule based [1]. Yinghua Li, Bingqi Liu, and Bin Zhou of Ordnance Engineering College (China), presented Fuzzy technology as a newly rising technology used in many fields, especially in the image domain, and fuzzy enhancing technique as one important portion of the fuzzy technology. Based on this technology, they firstly set the image fuzzy characteristic plane of original image, secondly preceded the fuzzy enhancement, and then detected the edge by Sobel differential arithmetic[2]. Yasar Becerikli and Tayfun of Kocaeli University, Computer Engineering Department, Izmit, Turkey proposed that an edge detection is one of the most important tasks in image processing. They studied that image segmentation; registration and identification are based on edge detection. They proposed that fuzzy rules based algorithm is more flexible in handling thickness of edges in the final image [3]. Cristiano Jacques Miosso, Adolfo Bauchspiess presented that *First-order* linear filters constitute the algorithms most widely applied to edge detection in digital images but they don't allow good results to be obtained from images where the contrast varies a lot, due to non-uniform lighting, as it happens during acquisition of most part of natural images[4]. Dong-Su Kim, Wang-Heon Lee, In-So Kweon, Department of Electrical Engineering and Computer Science, KAIST, South Korea, presented an edge magnitude and direction scheme that uses 3 x 3 ideal binary pixel patterns and described a lookup table. They concluded that their algorithm didn't require any manual online threshold adjustment and was more suitable to the dynamic environment[5].

The work of this paper is concerned with the development of a fuzzy logic rules based algorithm for the detection of image edges. Scanning mask used is smallest possible i.e. 2*2 pixels window. Fuzzy Inference based system in MATLAB environment has been

developed, which is capable of detecting edges of an image. The rule-base of 16 rules has been designed to mark the pixel under consideration as *Black*, *White* or *Edge*. The noise removal algorithm has been implemented at intermediate and final levels of processing. The result has been compared with the standard algorithms

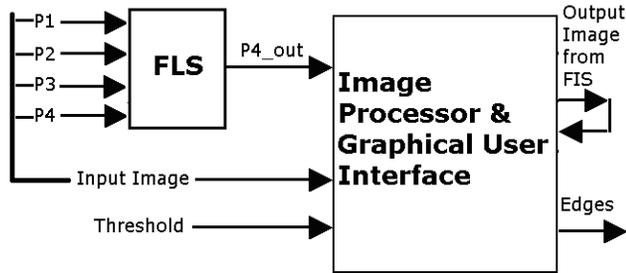


Figure 1. Basic Block Diagram

2. FUZZY LOGIC CONTROLLER

The designed fuzzy inference system is given four inputs and one output. The four inputs are the four pixel values of the window mask used. The triangular membership functions are used both for the inputs and the output. Two fuzzy sets are used for the input *Black* & *White* and three fuzzy sets are used for the output. Fuzzy sets for input and output variables are designed as shown in the table below

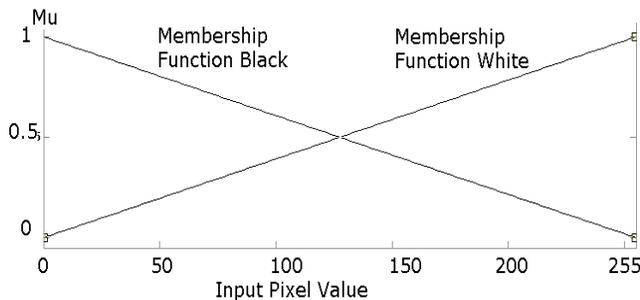


Figure 2. Fuzzy Sets of Input Pixels (P1-P4)

Table 1. Fuzzy Sets For I/O Variables

Fuzzy Input #1 = Pixel P1		
Name	Range	MF Type
Black	[0 0 255]	Triangular
White	[0 255 255]	Triangular
Fuzzy Input #2 = Pixel P2		
Name	Range	MF Type
Black	[0 0 255]	Triangular
White	[0 255 255]	Triangular
Fuzzy Input #3 = Pixel P3		
Name	Range	MF Type
Black	[0 0 255]	Triangular
White	[0 255 255]	Triangular
Fuzzy Input #4 = Pixel P4		
Name	Range	MF Type
Black	[0 0 255]	Triangular
White	[0 255 255]	Triangular
Fuzzy Output #1 = Pixel P4_out		
Name	Range	MF Type
Black	[0 3 5]	Triangular
Edge	[130 133 135]	Triangular
White	[249 252 255]	Triangular

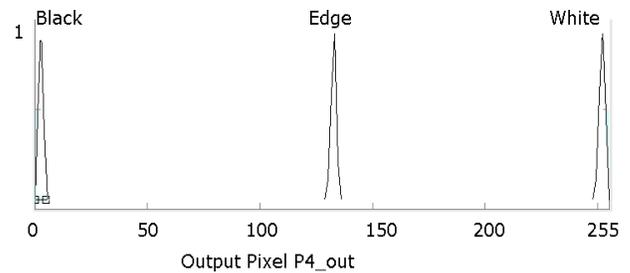


Figure 3. Fuzzy Sets of Output Pixel (P4_out)

A rule base of 16 rules is set for the various fuzzy conditions that can occur. Single output describes whether the output pixel i.e. P4 belongs to *White* fuzzy set, *Black* fuzzy set or *Edge* fuzzy set. Rules are enlisted in the form of a matrix in Table. 2.

3. NOISE REMOVAL

Noise removal is performed at different intermediate levels of processing. The idea of noise removal is to remove the pixels which have been falsely recognized as edge by the processing. Size of the scanning mask for this task is 3*3 pixels window. 3*3 pixels mask is slid over the whole image pixel by pixel row wise and the process continues till the time whole image is scanned for unwanted edge pixels. Fig. 2 shows p5 as falsely marked edge pixel as all the surrounding pixels i.e. p1, p2, p3, p4, p6, p7, p8 & p9 are white. Such types of falsely marked edge pixels are changed to *White* by the noise removal algorithm.

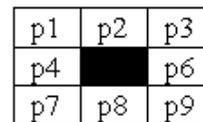


Figure 4. Falsely Marked Edge Pixel

Table 2. Fuzzy Rule Matrix

Fuzzy Inputs				Fuzzy Output
P1	P2	P3	P4	P4_out
B	B	B	B	B
B	B	B	W	E
B	B	W	B	E
B	B	W	W	E
B	W	B	B	E
B	W	B	W	E
B	W	W	B	E
B	W	W	W	W
W	B	B	B	E
W	B	B	W	E
W	B	W	B	E
W	B	W	W	E
W	W	B	B	E
W	W	B	W	E
W	W	W	B	E
W	W	W	W	W

4. ALGORITHM APPROACH

The algorithm is based on the subjection of a set of four pixels, part of a 2x2 window of an image to a set of fuzzy conditions which help to highlight all the edges that are associated with an image. . The fuzzy conditions help to test the relative values of pixels which can be present in case of presence on an edge. So the relative pixel values are instrumental in extracting all the edges associated to an image. The image is said to have an edge if the intensity variation in between the adjacent pixels is large. This task is accomplished with the help of sixteen rules. A Graphical User Interface is then designed in order to compare the results with the existing techniques.

The mask used for scanning image is shown below and an example is shown when P1, P2, P3, are white and P4 is Black then output is Black.

P1	P2
P3	P4

Figure 5. 2*2 Mask used for scanning

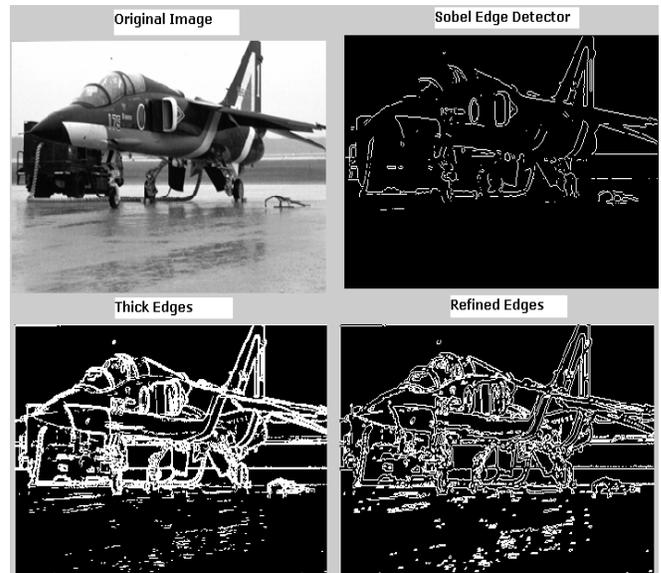
The mask is slid over an area of the input image, changes that P4 pixel's value and then shifts one pixel to the right and continues to the right until it reaches the end of a row. It then starts at the beginning of the next row & process continues till the whole image is scanned. When this mask is made to slid over the image, the output is generated by the fuzzy inference system based upon the rules and the value of the pixels P1, P2, P3 and P4.

- Crisp inputs for P1, P2, P3 & P4 are fuzzified into various FS, having conventional crisp membership functions i.e. Black & White.
- Firing strength is calculated using fuzzy t-norms operators (MIN or PRODUCT) on MFs.
- Fuzzy rules are fired for each crisp input.
- Aggregate resultant output FS for all fired rules is achieved by using MAX operator (s-norm).

- De-fuzzification is performed using the Centroid method.
- Crisp Output P4_out is the pixel value of the output image i.e. one containing the Edges, Black and White regions.
- First derivative is performed on the image output from FIS after the application of noise removal algorithm.
- Further refinement is performed by the second derivative and noise removal.

5. SIMULATION RESULTS

GUI designed for this application is shown in Fig. 11. Any of the standard edge detection algorithms (Sobel, Canny, Prewit & Roberts) can be selected for comparison from the *List Box* on GUI. Threshold level setting is done through the slider control of GUI. More the value of the slider, more of the edges will be traced, however, noise will also be increased. The developed fuzzy algorithm for image edge detection was tested for various images and the outputs were compared to the existing edge detection algorithms and it was observed that the outputs of this algorithm provide much more distinct marked edges and thus have better visual appearance than the standard existing. The sample output shown below in Fig.5.(a-c) compares the ‘‘Sobel’’ Edge detection algorithm and our fuzzy edge detection algorithm. It can be observed that the output that has been generated by the fuzzy method has found out the edges of the image more distinctly as compared to the ones that have been found out by the ‘‘Sobel’’ edge detection algorithm. Thus the Fuzzy rule based algorithm provides better edge detection and has an exhaustive set of fuzzy conditions which helps to extract the edges with a very high efficiency.

**Figure 6.** Results of our algorithm compared with Sobel

6. CONCLUSIONS

In this paper, emphasis has been to develop a very simple & small but a very efficient, fuzzy rule based edge detection algorithm to abridge the concepts of artificial intelligence and digital image processing. The algorithm and associated GUI has been developed in MATLAB environment. Comparisons were made with the various other edge detection algorithms that have already been developed. Displayed results have shown the accuracy of the edge

detection using the fuzzy rule based algorithm over the other algorithms. The fuzzy rule based algorithm has been successful in obtaining the edges that are present in an image after the its implementation and execution with various sets of images. Sample outputs have been shown to make the readers understand the accuracy of the algorithm. Thus developed algorithm exhibits tremendous scope of application in various areas of digital image processing.

7. FUTURE SCOPE

- a) To optimize the FLS, by fusion of other intelligent techniques like GA, ANN, PSO and / or ACO etc.
- b) To be designed using type-2 FS with expectation of better response.
- c) Presently, edge thickness is more than desirable. One can improve it by optimization of various fuzzy parameters.
- d) Further the size of mask can be increased from 2x2 to 3x3 or 4x4 and accordingly more rules can be set and results can be compared w.r.t. mask size.
- e) In our technique, the image is first to be converted into gray image. This limitation can be eliminated and algorithm can be applied directly to color images, and the detection would then become significantly more complex.

8. REFERENCES

- [1] Shashank Mathur, Anil Ahlawat, “Application Of Fuzzy Logic In Image Detection”, *International Conference "Intelligent Information and Engineering Systems" INFOS 2008, Varna, Bulgaria, June-July 2008*
- [2] Yinghua Li, Bingqi Liu, and Bin Zhou, “The Application Of Image Edge Detection by using Fuzzy Technique”, *in Conference “ Electronic Imaging and Multimedia Technology”, November 2004*
- [3] Yasar Becerikli1 and Tayfun M. Karan, “A New Fuzzy Approach for Edge Detection”, *Computational Intelligence and Bio inspired Systems”, June 2005.*
- [4] Cristiano Jacques Miosso, Adolfo Bauchspiess, “Fuzzy Inference System Applied to Edge Detection in Digital Images”, *in the proceedings of the V Brazilian Conference on Neural Networks pp. 481–486, April , 2001*
- [5] Dong-Su Kim, Wang-Heon Lee , In-So Kweon, “Automatic edge detection using 3x3 ideal binary pixel patterns and fuzzy-based edge thresholding,” *in Pattern Recognition Letters in 2004*