Edge Detection Technique using HSI and Fuzzy Interference System

Sachin Chawla[1], Ajay Khokhar[2]

[1]M.Tech Scholar, Department of Electronics & Communication Engineering, SPGOI Rohtak (India)

[2]Assistant Professor, Department of Electronics & Communication Engineering, SPGOI Rohtak (India)

Abstract: ***The aim of image segmentation can be defined as partitioning an image into homogeneous regions in terms of the features of pixels extracted from the image. Image segmentation methods can be classified into four main categories: 1) clustering methods, 2) region-based methods, 3) hybrid methods, and 4) Bayesian. Digital image processing is a subset of the electronic domain wherein the image is converted to an array of small integers, called pixels, representing a physical quantity such as scene radiance, stored in a digital memory, and processed by computer or other digital hardware. Interest in digital image processing methods stems from two principals applications areas: improvement of pictorial information for human interpretation; and processing of image data for storage, transmission, and representation for autonomous machine perception. Edges characterize boundaries and edge detection is one of the most difficult tasks in image processing hence it is a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts and a jump in intensity from one pixel to the next can create major variation in the picture quality. Edge detection of an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. In many image processing applications, we have to use expert knowledge to overcome the difficulties (e.g. object recognition, scene analysis). Fuzzy set theory and fuzzy logic offer us powerful tools to represent and process human knowledge in form of fuzzy if-then rules. On the other side, many difficulties in image processing arise because the data/tasks/results are uncertain. This uncertainty, however, is not always due to the randomness but to the ambiguity and vagueness. Beside randomness which can be managed by probability theory we can distinguish between three other kinds of imperfection in the image processing. The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. For an image formation model, discontinuities in image brightness are likely to correspond to:-***
Discontinuities in depth, Discontinuities in surface orientation, Changes in material properties, Variations in scene illumination

Keywords: Fuzzy if-then rule, image segmentation, region-based method fuzzy set theory.

1. Introduction

A digital image is a representation of a two-dimensional image as a finite set of digital values. In image processing, the digitization process includes sampling and quantization of continuous data. The sampling process samples the intensity of the continuous-tone image, such as a monochrome, color or multi-spectrum image, at specific locations on a discrete grid. The grid defines the sampling resolution. The quantization process converts the continuous or analog values of intensity brightness into discrete data, which corresponds to the digital brightness value of each sample, ranging from black, through the grays, to white. A digitized sample is referred to as a picture element, or pixel. The digital image contains a fixed number of rows and columns of pixels. Pixels are like little tiles holding quantized values that represent the brightness at the points of the image. Pixels are parameterized by position, intensity and time. Typically, the pixels are stored in computer memory as a raster image or raster map, a two-dimensional array of small integers. Image is stored in numerical form which can be manipulated by a computer. A numerical image is divided into a matrix of pixels (picture elements).

Digital image processing allows one to enhance image features of interest while attenuating detail irrelevant to a given application, and then extract useful information about the scene from the enhanced image. Images are produced by a variety of physical devices, including still and video cameras, x-ray devices, electron microscopes, radar, and ultrasound, and used for a variety of purposes, including entertainment, medical, business (e.g. documents), industrial, military, civil (e.g. traffic), security, and scientific. The goal in each case is for an observer, human or machine, to extract useful information about the scene being imaged.

Colour television is an example of the additive nature of light. The screen of a colour TV is a large array of triangular patterns of red, green and blue phosphors that are individually modulated by an electron gun to obtain a specific intensity. The three colour intensities are used to make the correct colour. That is, the three primary colours of each triangle are added together and perceived by the human eye as a single colour. The colour characteristics are given by:

1) Brightness: basically, the intensity,
2) Hue: the colour corresponding to the dominant wavelength in a mixture of light waves (so effectively what humans call colour) and
3) Saturation: the relative purity or the amount of white light mixed with a hue. The pure spectrum colours are 100% saturated. Colours such as pink (red and white) or lavender (violet and white) are less saturated; the degree of saturation is inversely proportional to the amount of white light added. Hue and saturation taken together are called chromaticity and therefore, colour is characterized by its brightness and its chromaticity.

The amounts of red, green and blue needed to form a colour are called its tristimulus and denoted by X, Y and Z. A colour is specified by its trichromatic coefficients:

$$x = \frac{X}{X+Y+Z},$$

$$y = \frac{Y}{X+Y+Z}$$

and

$$z = \frac{Z}{X+Y+Z}.$$

Obviously, $x + y + z = 1$.

We can use the x (red) and y (green) axes, with z = 1-x-y, to represent a colour on the CIE diagram shown in Figure 1 (plate4). Thus, if a colour has 60% red and 30% green, then it must have 10% blue.

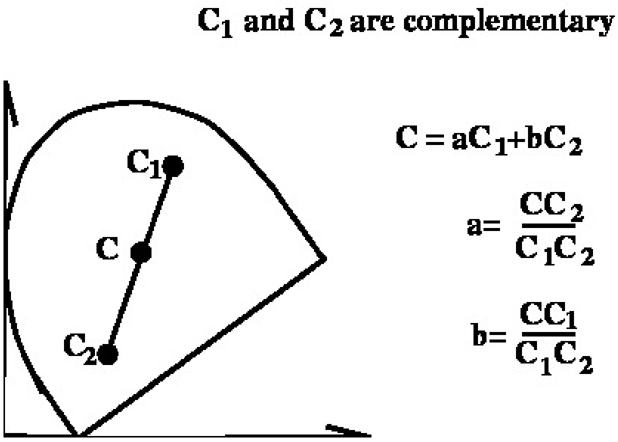
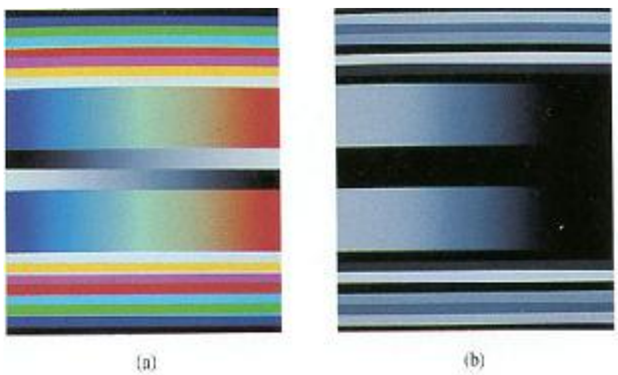


Figure 1: Combining 3 colours in the CIE diagram: the colour triangle

2. HSI Model

We are interested in the HSI colour model because it decouples intensity and colour information and because hue and saturation are closely related to the way humans perceive colour. Figure 2 shows an RGB colour test pattern. The test pattern consists, at the top, of 8 thin bands that contain black, followed by the pure primaries and secondaries and ending in white. Following these bands is a broad multi-colour band that ranges from blue to green to red. This band is followed by 2 gray value strips varying their intensity in opposite directions. These patterns then repeat themselves going the other way to form a square image.



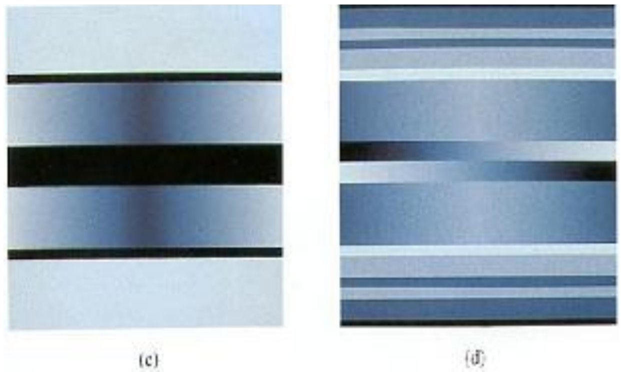


Figure 2: plate 8: (a) A RGB image and its (b) hue, (c) saturation and (d) intensity images

Figure 2 shows the transformations used to make the band of varying colour in Figure 2 a Note that this band is pure blue on the left end, pure red on the right end and equal parts red and blue plus twice the green in the middle.

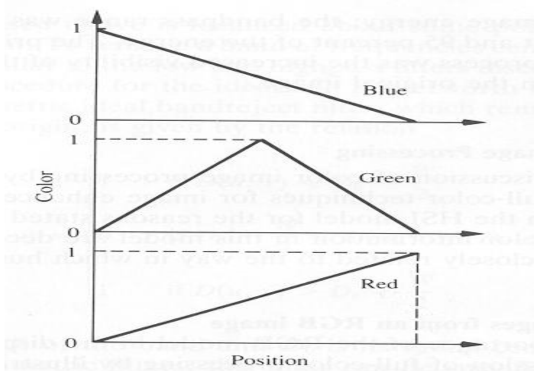


Figure 3: The colour transformations used to make the original colour band

Because intensity is decoupled from colour information using the HIS model, any monochrome enhancement technique discussed earlier can be used for enhancing full-colour images. Simply convert the image to HSI and enhance the I image and then convert back to RGB and display the resulting image. Figure 4a shows a full-colour image whose background detail is obscured. The image was converted to HSI and its component was histogram equalized. The result was converted back to RGB

Figure 4: (a) Original RGB image and (b) result after conversion to HSI, histogram equalization on and conversion back to RGB

The HSI model is useful for image processing that is based on the way humans perceive colour. In is useful, for example, in a system that deter mines the ripeness of fruits and vegetables. Artists also like the system to talk about tints (the colour of a hue less than 100% saturated), shades the colour from black to some pure hue) and tones (the colour in the triangle with white, black and a pure hue as the 3 vertices). We can perform RGB to HSI conversion using Matlab functions, hsi2rgb and rgb2hsi supplied by Gonzalez and Eddins and available on the course webpage. These functions are NOT built-in functions in Matlab.

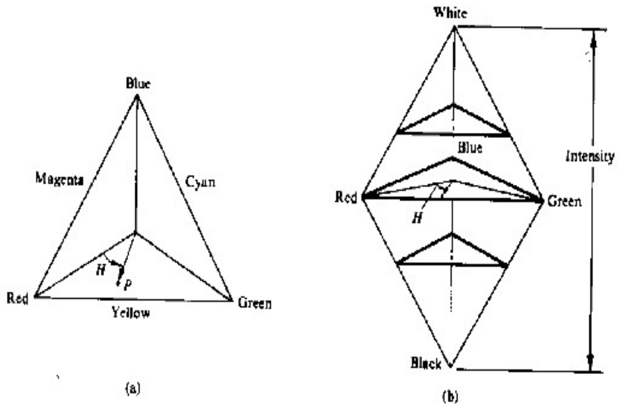


Figure 5: (a) The HSI Colour Triangle and (b) the HSI Colour Solid

(a) Colour Image (b) Hue (H) Image

(c) Saturation (S) Image (d) Intensity (I) Image

Figure 6: (a) The colour lena image, (b) its hue (H) image, (c) its saturation (S) image and (d) its intensity (I) image.

3. Conversion from RGB to HSI

The functions hsi2rgb and rgb2hsi use the algorithms described below.

- The colour components of HSI are defined with respect to the colour triangle shown in Figure 1.20. Since red, green and blue colours are the vertices of this triangle, we can obtain all the colour of the RGB cube.
- In the HSI model, Hue (H) is defined as an angle with respect to the red axis (the line from red to the middle vertical line).
- Saturation (S) is the degree to which the colour is diluted with white light and is proportional to the distance of colour P from the center of the triangle.
- Intensity (I) is measured with respect to a line perpendicular to the triangle and passing through its center. The intensities vary from black at the bottom to white at the top.
- Assuming R, G and B are normalized, i.e. 2 [0; 1], then, without derivation,
- the conversion equations for H, S and I (2 [0; 1] as well) are:

$$I = \frac{1}{3}(R+G+B),$$

$$S = 1 - \frac{3}{(R+G+B)}[\min(R,G,B)] \quad \text{and}$$

$$H = \cos^{-1}\left\{\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)]^{\frac{1}{2}}}\right\}.$$

We let $H = 360^\circ - H$ if $B/I > G/I$.

H is normalized, i.e. $H = H/360^\circ$

- If S=0, it is meaningless to talk about hue, H, angles as there is no colour information.
- If I=0, it is meaningless to talk about S, saturation as, again, there is no colour information

3.1 Image processing operations

Image processing operations can be roughly divided into three major categories:

a) Image Restoration
b) Image Enhancement
c) Image Compression
d) Image Segmentation

4. Image Segmentation as Edge Detection Operations

Segmentation is the process that subdivides an image into a number of uniformly homogeneous regions. Each homogeneous region is a constituent part or object in the entire scene. The objects on the land part of the scene need to be appropriately segmented and subsequently classified. Partitioning of an image is based on abrupt changes in gray level. If edges of the image can be extracted and linked, the region is described by the edge contour that contains it. The principal areas of interest within this category are the detection of edges of a digital image. An edge corresponds to local intensity discontinuities of an image. In the real world, the discontinuities reflect a rapid intensity change, such as the boundary between different regions, shadow boundaries, and abrupt changes in surface orientation and material properties. For example, edges represent the outline of a shape, the difference between the colors and pattern or texture. Therefore, edges can be used for boundary estimation and segmentation in scene understanding. They can also be used to find corresponding points in multiple images of the same scene. For instance, the fingerprint, human facial appearance and the body shape of an object are defined by edges in images. In a broad sense the term edge detection refers to the detection and localization of intensity discontinuities of these image properties. In a more restrictive sense, it only refers to localizations of significant change of intensity. Points of these localizations are called edges or edge elements. Edges are piecewise segmentation. They are both useful in computation of geometrical features such as shape or orientation. Edge detection is grounded on the assumption that physical 3-dimensional shapes in the scene, such as object boundaries and shadow boundaries, are clues for the characterization of the scene.

Figure 7: Image after and before segmentation

Since the overall goal is to locate edges in the real world via an image, the term edge detection is commonly used. An edge is not a physical entity, just like a shadow. It is where the picture ends and the wall starts, where the vertical and the horizontal surfaces of an object meet. If there were sensor with infinitely small footprints and zero-width point spread functions, an edge would be recorded between Pixels within in an image. In reality, what appears to be an edge from the distance may even contain other edges when close-up looked. The edge between a forest and a road in an aerial photo may not look like an edge any more in a image taken on the ground. In the ground image, edges may be found around each individual tree. If looked a few inches away from a tree, edges may be found within the texture on the bark of the tree. Edges are scale-dependent and an edge may contain other edges, but at a certain scale, an edge still has no width. If the edges in an image are identified accurately, all the objects are located and their basic properties such as area, perimeter and shape can be measured. Therefore edges are used for boundary estimation and segmentation in the scene. Since computer vision involves the identification and classification of objects in an image, edge detection is an essential tool.

4.1 Types of Edges

All edges are locally directional. Therefore, the goal in edge detection is to find out what occurred perpendicular to an edge. The following is a list of commonly found edges.

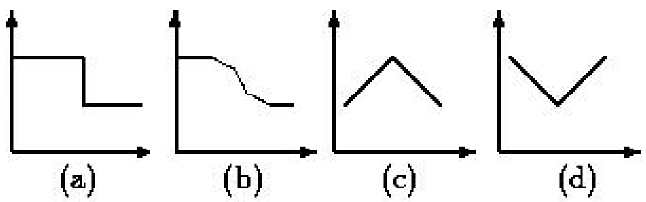


Figure 8: Types of Edges (a) Sharp step (b) Gradual step (c) Roof (d) Trough

A Sharp Step, as shown in Figure 8(a), is an idealization of an edge. Since an image is always band limited, this type of graph cannot ever occur. A Gradual Step, as shown in Figure 8(b), is very similar to a Sharp Step, but it has been smoothed out. The change in intensity is not as quick or sharp. A Roof, as show in Figure 8(c), is different than the first two edges. The derivative of this edge is discontinuous. A Roof can have a variety of sharpness, widths, and spatial extents. The Trough, also shown in Figure 8(d), is the inverse of a Roof. Edge detection is very useful in a number of contexts. Edges characterize object boundaries and are, therefore, useful for segmentation, registration, and identification of objects in scenes [1]. A straightforward example of edge detection is illustrated in Figure 7 Original picture has a uniform grey background. The edge enhanced version of the same image has dark lines outlining the three objects. Note that there is no way to tell which parts of the image are background and which are object, only the boundaries between the regions are identified.

The goal of the edge detection process in a digital image is to determine the frontiers of all represented objects, based on automatic processing of the color or gray level information in each present pixel. Edge detection has many applications in image processing and computer vision, and is an indispensable technique in both biological and robot vision [3]. The main objective of edge detection in image processing is to reduce data storage while at same time retaining its topological properties, to reduce transmission time and to facilitate the extraction of morphological outlines from the digitized image.

4.2 Criteria for Edge Detection

There are large numbers of edge detection operators available, each designed to be sensitive to certain types of edges. The Quality of edge detection can be measured from several criteria objectively. Some criteria are proposed in terms of mathematical measurement, some of them are based on application and implementation requirements. In all five cases a quantitative evaluation of performance requires use of images where the true edges are known.

a) **Good detection**: There should be a minimum number of false edges. Usually, edges are detected after a threshold operation. The high threshold will lead to less false edges, but it also reduces the number of true edges detected.
b) **Noise sensitivity:** The robust algorithm can detect edges in certain acceptable noise (Gaussian, Uniform and impulsive noise) environments. Actually, an edge detector detects and also amplifies the noise simultaneously. Strategic filtering, consistency checking and post processing (such as non-maximum suppression) can be used to reduce noise sensitivity.
c) **Good localization:** The edge location must be reported as close as possible to the correct position, i.e. edge localization accuracy (ELA).
d) **Orientation sensitivity:** The operator not only detects edge magnitude, but it also detects edge orientation correctly. Orientation can be used in post processing to connect edge segments, reject noise and suppress non-maximum edge magnitude.
e) **Speed and efficiency:** The algorithm should be fast enough to be usable in an image processing system. An algorithm that allows recursive implementation or separately processing can greatly improve efficiency.

Criteria of edge detection will help to evaluate the performance of edge detectors. Correspondingly, different techniques have been developed to find edges based upon the above criteria, which can be classified into linear and non linear techniques.

4.3 Various Techniques for Edge Detection

Edge detection of an image reduces significantly the amount of data and filters out information that may be regarded as less relevant, preserving the important structural properties of an image. Therefore, edges detected from its original image contain major information, which only needs a small amount of memory to store. The original image can be easily restored from its edge map. Various edge detection algorithms have been developed in the process of finding the perfect edge detector. However, most of them may be grouped into two categories, namely, gradient based edge detection and Laplacian-based edge detection. In the gradient based edge detection, we calculate an estimate of the gradient magnitude using the smoothing filter and use the calculated estimate to determine the position of the edges. In other words the gradient method detects the edges by looking for the maximum and the minimum in the first derivative of the image. In the Laplacian method we calculate the second derivative of the signal and the derivative magnitude is maximum when second derivative is zero, In short, Laplacian method searches for zero crossings in the second derivative of the image to find edges. The original image can be easily restored from its edges. Two main methods:-

a) **Gradient-based method**: Gradient-based methods (referred in Appendix A) detect edges by looking for maxima and minima in the first derivative of the image.
b) **Laplacian (zero-crossing) based method**: The Laplacian based methods search for zero crossings in the second derivative of the image in order to find edges, usually the zero-crossings of the Laplacian or the zero-crossings of a non-linear differential expression.

A number of edge detection techniques are available but there is no single detection method that performs well in every possible image context. Various edge detection techniques are used for edge detection like canny edge detection Krisch edge detection which are applied on various images. Choice of edge detector to be used depends upon the image properties like noise sensitivity, orientation sensitivity, speed and efficiency.

5. Fuzzy Logic used as Edge Detection

Fuzzy logic is a powerful problem-solving methodology with a myriad of applications in embedded control and information processing [2]. Fuzzy provides a remarkably simple way to draw definite conclusions from vague, ambiguous or imprecise information. In a sense, fuzzy logic resembles human decision making with its ability to work from approximate data and find precise solutions. As shown in Figure 1.29 fuzzy logic and probability theory are the most powerful tools to overcome the imperfection.

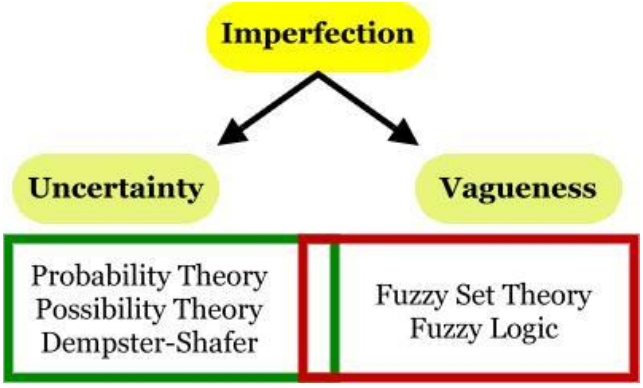


Figure 9: Imperfection and theories to handle it

Fuzzy image processing is not a unique theory. Fuzzy image processing is the collection of all approaches that understand, represent and process the images, their segments and features as fuzzy sets. The representation and processing depend on the selected fuzzy technique and on the problem to be solved. Fuzzy image processing has three main stages: image fuzzification, modification of membership values, and, if necessary, image defuzzification.

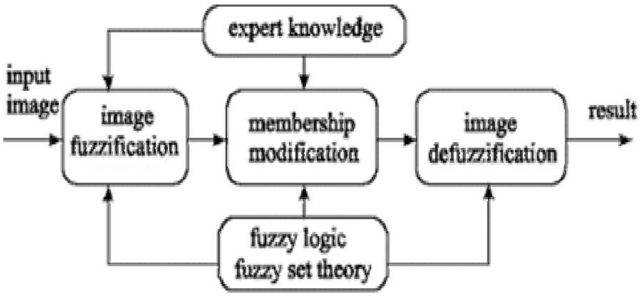


Figure 10: The general structure of fuzzy image processing

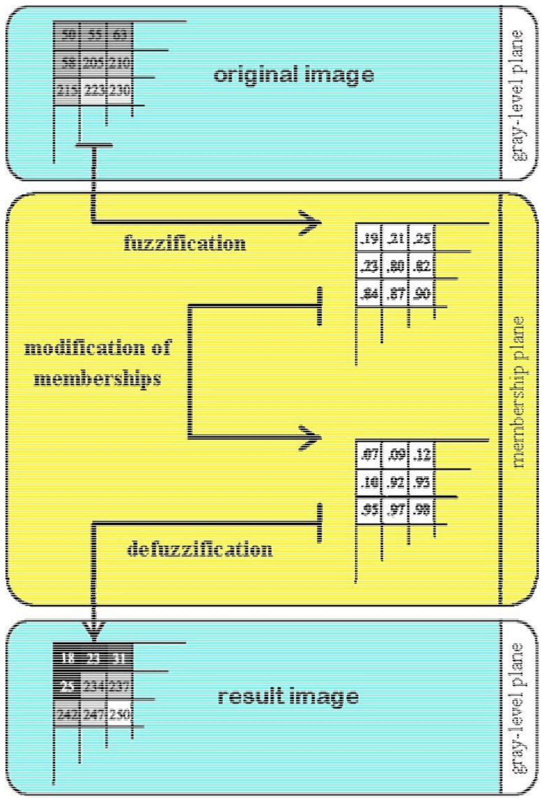


Figure 11: Steps of Fuzzy Image Processing

6. Result and Conclusion

Figure 12: Original Image

Figure 13: RGB Image

Figure 14: HSI Image

Figure 15: RGB Image-HSI Equalized

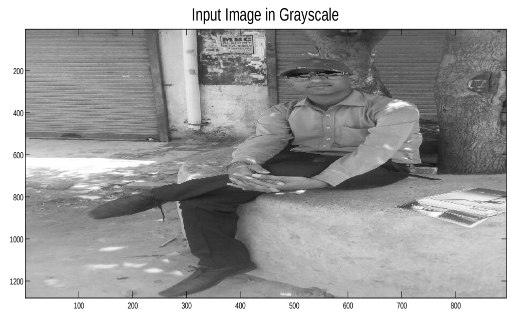


Figure 16: Input Image in Grayscale

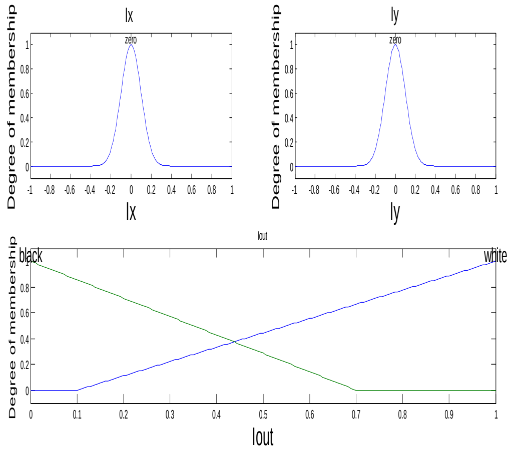


Figure 17: Fuzzyfier Rule with membership number

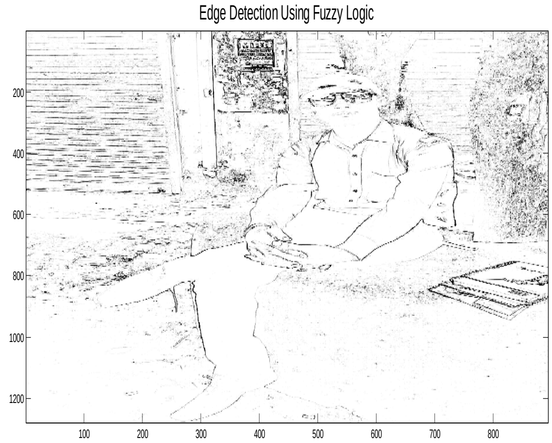


Figure18: Edge Detection Using Fuzzy Logic

Figure 19: Edge Detection using only HIS Function

In this thesis work, better algorithm has been proposed to improve the detection of edges by using HSI function & fuzzy rules. This algorithm is adaptable to various environments. The weights associated with each fuzzy rule were tuned to allow good results to be obtained while extracting edges of the image, where contrast varies a lot from one region to another. During the performance tests, however, all parameters were kept constant. If you use only HSI Function to detect the edge of the image, so quality if picture edge detected become very poor but when we use HIS and fuzzy both in one scenario the quality of edge detected quality much higher than the simple HIS. The results allow us to conclude that, the implemented FIS system presents greater robusteness to contrast and lighting variations, besides avoiding obtaining double edges. For future prospect, the proposed technique is to find more fine edges using fuzzy logic technique. In future, modification of fuzzy rules can produce better result. Further tuning of the weights associated to the fuzzy inference rules is still necessary to reduce even more inclusion in the output image of pixels not belonging to edges.

Our proposed technique is restricted only to gray scale images, this can be extended to color images in that case, and the detection would become significantly more complex

References

[1] Ng Geok See and Chan Khue hiang. "Edge Detection using supervised Learning and voting scheme". Nanyang Technological University, National university of Singapore, Singapore

[2] Hamid R. Tizhoosh, "Fuzzy Image Processing". www.pami.uwaterloo.ca/tizhoosh/edge.html

[3] Md. Shoiab Bhuiyan, Yuiji Iwahori, and Akira Iwata. "Optimal edge detection under difficult imaging conditions". Technical report, Educational Center for Information Processing and Dept. of Electrical and Computer Engineering, Nagoya Institute of Technology, Showa, Nagoya, 466-8555, JAPAN, URL:- http://www.center.nitech.ac.jp/people/bhuiyan /pub.html.

[4] Hamid R. Ezhoosh, "Fast Fuzzy Edge Detection". Pattern Recognition and Machine Intelligence Lab Systems Design Engineering, University of Waterloo Waterloo, Ontario, N2L 3G1, Canada.

[5] M. Hanmandlu, Rohan Raj Kalra, Vamsi Krishna Madasu, Shantaram Vasikarla. "Area based Novel Approach for Fuzzy Edge Detection". Department of Electrical Engineering, Indian Institute of Technology, Delhi.

[6] Cristiano Jacques Miosso, Adolfo Bauchspiess. "Fuzzy Inference System Applied to Edge Detection in Digital Images". Departamento de Engenharia Electrical Faculdade de Tecnologia Universidade de Braslia – UnB

[7] Jinbo Wu, Zhouping Yin, and Youlun Xiong, "The Fast Multilevel Fuzzy Edge Detection of Blurry Images".

[8] Renyan Zhang, Guoling Zhao, Li Su. "New edge detection method in image processing". College of Autom., Harbin Engineering University, China.

[9] Stamatia Giannarou, Tania Stathaki. **"Edge Detection Using Quantitative Combination of Multiple Operators".** Communications and Signal Processing Group, Imperial College LondonExhibition Road, SW7 2AZ London, UK

Author Profile

Sachin Chawla pursuing **MTECH** from SPGOI, Rohtak (2013-2015) and obtained **B.Tech in Electronics and Communication Engineering from** Maharshi Dayanand University. His main research interests includes: Next generation Wireless Networks and Image Processing.

Mr. Ajay Khokhar received the Masters degree in Electronics and Communications from **BRCM College of Engg. & Technology**, Bahal Bhiwani in 2011. He received his B.E degree in **Electronics and Communication from Vaish College of Engg. Rohtak**, India in 2006. At present he is working as Head of Deptt. ECE SPGOI Rohtakl. His main research interests includes: Next generation Wireless Networks and Image Processing. He has teaching experience of more than 8 years.