SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 12, No. 1, Feb 2015, 109-116

UDC: 004.93`12:004.942 **DOI: 10.2298/SJEE1501109P**

Fuzzy Logic and Fuzzy Set Theory Based Edge Detection Algorithm

Nebojša Perić[1]

Abstract: In this paper we will show a way how to detect edges in digital images. Edge detection is a fundamental part of many algorithms, both in image processing and in video processing. Therefore it is important that the algorithm is efficient and, if possible, fast to carry out. The fuzzy set theory based approach on edge detection is good for use when we need to make some kind of image segmentation, or when there is a need for edge classification (primary, secondary, ...). One example that motivated us is region labeling; this is a process by which the digital image is divided in units and each unit is given a unique label (sky, house, grass, …, etc.). To accomplish that, we need to have an intelligent system that will precisely determine the edges of the region. In this paper we will describe tools from image processing and fuzzy logic that we use for edge detection as well as the proposed algorithm.

Keywords: Fuzzy, Logic, Edge, Detection, Image, Processing.

1 Introduction

An image may be defined as two-dimensional function $f(x, y)$, where x and y are spatial coordinates, and the amplitude of f at any pair of coordinates (x, y) is called intensity of gray level of the image at that point. When x, y and the intensity values of f are all finite, discrete quantities, we call the image a digital image.

We should notice that every digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are called *picture elements*, *pels*, and *pixels*. *Pixel* is the term used most widely to denote the elements of digital image [1].

Now an image can be treated as a matrix of pixels, where each pixel in the matrix is represented by its intensity of grayscale value, which is a nonnegative number. Processing an image on pixel level is called processing in spatial domain.

Now we can define edges. Edge pixels are pixels at which the intensity of an image function changes abruptly, and edges (or edge segments) are sets of

[1]Group for Intelligent Systems, Faculty of Mathematics, University of Belgrade, Serbia;
E-mail: nebojscha.perich@gmail.com

connected edge pixels. Edge detectors are local image processing methods designed to detect edge pixels. A line may be viewed as an edge segment in which the intensity of the background on either side of the line is either much higher or much lower than the intensity of line pixels.

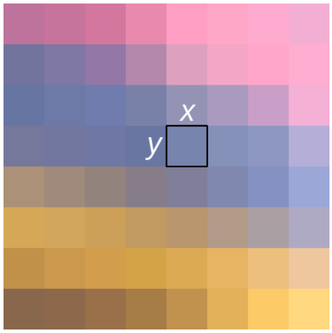

Fig 1 – *Pixel in digital image.*

Edge detection is a fundamental tool used in most image processing applications to obtain information from the frames as a precursor step to feature extraction and object segmentation. This process detects outlines of an object and boundaries between objects and the background in the image.

The basic edge-detection operator is a matrix- area gradient operation that determines the level of variance between different pixels. The edge-detection operator is calculated by forming a matrix centred on a pixel chosen as the center of the matrix area. If the value of this matrix area is above a given threshold, then the middle pixel is classified as an edge. Examples of gradient-based edge detectors are Roberts, Prewitt, and Sobel operators.

All the gradient-based algorithms have kernel operators that calculate the strength of the slope in directions which are orthogonal to each other, commonly vertical and horizontal. Later, the contributions of the different components of the slopes are combined to give the total value of the edge strength.

In that domain, as mentioned before, we have several algorithms for edge detection like Sobel algorithm, Prewitt algorithm, Roberts algorithm, Canny edge detection algorithm and others. The main problem for all of these algorithms is that they are „not perfect”. When we say not perfect we mean that one and the same edge detection algorithm cannot always be used for image preprocessing in more complex algorithms that require a previous edge detection. For example, Sobel algorithm is good for shape recognition, but is not good for real-time edge detection for video capture (instead of Sobel algorithm we use Canny). Therefore there is always a need to find a different, and perhaps, a better algorithm for this problem [2].

Another way to detect edges in a digital image is to use fuzzy logic (FL). Zadeh introduced the term *fuzzy logic* in his seminal work "Fuzzy sets," which described the mathematics of fuzzy set theory (1965). Plato laid the foundation for what would become fuzzy logic, indicating that there was a third region beyond True and False. It was Lukasiewicz who first proposed a systematic alternative to the bivalued logic of Aristotle. The third value Lukasiewicz proposed can be best translated as "possible," and he assigned it a numeric value between True and False. Later he explored four-valued logic and five-valued logic, and then he declared that, in principle, there was nothing to prevent the derivation of infinite-valued logic. FL provides the opportunity for modeling conditions that are inherently imprecisely defined. Fuzzy techniques in the form of approximate reasoning provide decision support and expert systems with powerful reasoning capabilities. The permissiveness of fuzziness in the human thought process suggests that much of the logic behind thought processing is not traditional two- valued logic or even multivalued logic, but logic with fuzzy truths, fuzzy connectiveness, and fuzzy rules of inference [3].

2 Fuzzy Logic and Fuzzy sets

A fuzzy set is an extension of a crisp set. Crisp sets allow only full membership or no membership at all, whereas fuzzy sets allow partial membership. In a crisp set, membership or non-membership of element *x* in set *A* is described by a characteristic function μ(x) [4]

$$\mu_A(x) = 1 \text{ if } x \in A \text{ and } \mu_A(x) = 0 \text{ if } x \notin A. \tag{1}$$

Fuzzy set theory extends this concept by defining partial membership. A fuzzy set *A* on a universe of discourse *U* is characterized by a membership function $\mu_A(x)$ that takes values in the interval [0, 1]. Fuzzy sets represent commonsense linguistic labels like *slow, fast, small, large, heavy, low, medium, high, tall,* etc. A given element can be a member of more than a one fuzzy set at a time. A fuzzy set *A* in *U* may be represented as a set of ordered pairs. Each pair consists of a generic element *x* and its grade of membership function:

$$A = \{(x, \mu(x)) \mid x \in U\}, \tag{2}$$

x is called support value if μ(x)>0. A membership function is essentially a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. As an example, consider a fuzzy set *tall*. Let the universe of discourse be heights from 40 inches to 90 inches. With a crisp set, all people with height 72 or more inches are considered tall, and all people with height of less than 72 inches are considered not tall. In the fuzzy set this is different. For every man we would calculate the degree, how much does he belong to the set of tall people.

The fuzzy approach can be used in image processing, especially for edge detection. 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.

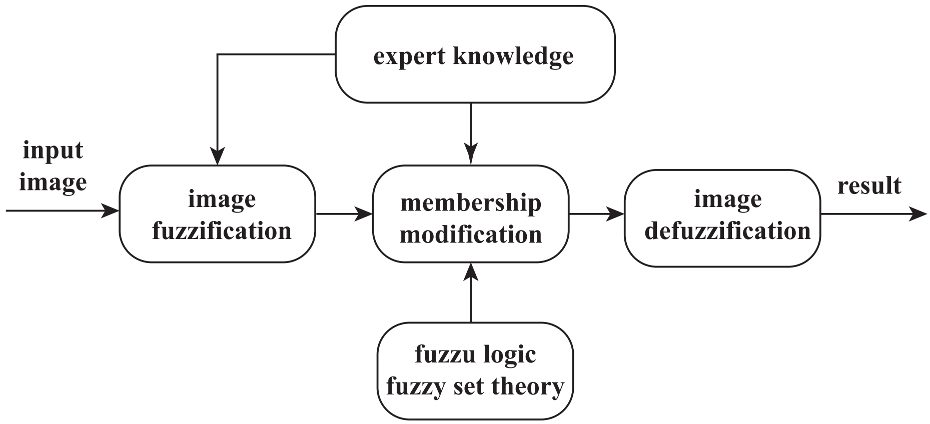

Fig 2 – *Main steps of fuzzy image processing.*

3 Edge Detection Algorithm

First of all, we need is to represent the digital image as a fuzzy set. This process is called fuzzification, while image defuzzification is a process in which we transform the values back to crisp values in the spatial domain. Zadeh in his work [4] introduced a mathematical framework for a new image understanding. An image *I* of size $M \times N$ and *L* gray levels can be considered as an array of fuzzy singletons, each having a value of membership denoting its degree of brightness relative to some brightness level l = 0, 1, …, *L*–1. In fuzzy notation image I can be represented [5]:

$$I = \bigcup_m \bigcup_n \mu_{mn} / g_{mn}, \quad m = 1, 2\ldots,M \quad \text{and} \quad n = 1, 2,\ldots,N, \tag{4}$$

where g_{mn} is the intensity of the (m,n)th pixel in the image and μ_{mn} its membership value. The membership function characterizes a suitable property of an image. For us, in our method, the property that is important is Edginess. We need to have a membership function that will give a degree of how much a specific pixel is belonging to the set of Edges? Now we see that the main idea is to create a fuzzy set out of our image, where each element of the fuzzy set is pair of pixel and edge membership value. For an image X size of $M \times N$ with L levels of gray intensities, we can create an Edge Image as following [6]:

$$Edge(X) = (L-1) \cdot \bigcup_m \bigcup_n \frac{\mu_{mn}}{g_{mn}}. \tag{5}$$

The key to determine how much a pixel is an edge pixel gives us the edge membership function. The Membership function is not unique, so there can be different membership functions, but they must meet the appropriate restrictions.

One membership function that can calculate the belonging to the set of edge pixels is given with:

$$\mu_{edge}(g(x,y)) = \frac{\Delta}{\sum_N \left| g(x,y) - g(i,j) \right| + \Delta}, \tag{6}$$

where g is the original image, and x and y are the coordinates of that pixel in image. In this connection, N is the neighborhood of pixel (x, y). The neighborhood is composed of pixels bordering our pixel on the position (x, y). Graphically it can be represented as:

$(x-1, y-1)$	$(x-1, y)$	$(x-1, y+1)$
$(x, y-1)$	(x, y)	$(x, y+1)$
$(x+1, y-1)$	$(x+1, y)$	$(x+1, y+1)$

Fig 3 – *Pixel* (x, y) *and its* 3×3 *neighborhood.*

Δ is a suitable number or constant. For Δ we usually take the maximal grayscale level $L - 1$ or the maximum grayscale intensity that appears in the neighborhood.

In order to more precisely show the relation between Δ and the sum of differences between the central pixel and its neighbors the membership function (6) can further be transformed into a terminated continued fraction:

$$\mu_{edge}(g(x,y)) = 1 - \frac{1}{1+\alpha}, \tag{7}$$

where α is:

$$\alpha = \frac{\sum_N \left| g(x,y) - g(i,j) \right|}{\Delta} \tag{8}$$

The Algorithm for edge detection based on fuzzy sets and fuzzy set theory is composed of the following steps:

Step 1: Load input Image I and determine its dimensions M = Width, N = Height.

Step 2: if image is color image convert it to grayscale. Iterate through the pixels of the image and set their intensities by the formula:

$$I(i,j) = \sqrt[2,2]{\begin{aligned}&0,2126 \times I(i,j).red^{2,2} + 0,7152 \times I(i,j).blue^{2.2} + \\ &+0,0722 \times I(i,j).green^{2,2}\end{aligned}}$$

Red, green and blue represent the color components of the pixels.

Step 3: Iterate through each pixel in the image and calculate the edge membership function μ_{edge} using (6) or (7). Remember the maximal value of $\mu_{edge.}$ that appeared in the image (MAX).

Step 4: For each pixel modify its membership value by dividing it with the MAX.

$$\mu_{edge}(I(i,j)) = \mu_{edge}(I(i,j)) / MAX .$$

Step 5: Generate the edges. For each pixel set the value of grayscale intensity by the formula:

$$I(i,j) = (L-1) \times \mu_{edge}\left(I(i,j)\right) .$$

4 Results and Examples

In this section we will see some applications of our algorithm on test images. This algorithm can be modified, in the way that we can classify edges, and to choose which one to display. That is good because we can describe, for example, primary edges as edges that are composed of pixels that have edge membership value greater than 0.75. In that manner we can also describe irrelevant edges and remove them if necessary. That is very useful if our input image had some noises that could have influenced on the final output.

Fig 4 – *Test image prior and after application of algorithm.*

Fig 5 – *Test image prior and after application of algorithm.*

5 Conclusion

The main advantage of this approach is that the algorithm is not so difficult to implement. Also the membership function can be used to classify edges. For example to order them from primary to irrelevant and to choose with which class of edges we want to operate with, or use in some more complex image processing. The main disadvantage of this, and any fuzzy image processing technique, is time complexity because we need to iterate several times through the image. This problem can be solved by taking advantage of the programming language in which we are coding, or we can merge steps 3 and 4 and do the operations of these steps in one iteration.

There are many possible fields of use for this approach, especially when we are dealing with sensitive and uncertain data. It could be used in molecular biology or in Medical laboratories for sample extractions, image improvement, image segmentation, noise removal, pattern recognition and as preprocessing method for more complex image processing algorithms.

6 References

[1] R. C Gonzalez, R. E. Woods: Digital image processing, 2nd Edition, Prentice Hall, Upper Saddle River, New Jersey, USA, 2002.

[2] N. R. Pal, K S. Pal: A Review on Image Segmentation Techniques, Pattern Recognition, Vol. 26, No. 9, 1993, pp. 1277 – 1294.

[3] Z. Chi, H. Yan, T. Pham, Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition, Advances in Fuzzy Systems – Applications and Theory, Vol. 10, World Scientific, 1996.

[4] L.A. Zadeh: Fuzzy Sets, Information and Control, Vol. 8, No. 3, 1965, pp. 338 – 353.

[5] G. Krell, B. Michaelis: On Fuzzy Enhancement of Megavoltage Images in Radiation Therapy, Sixth IEEE International Conference on Fuzzy Systems, 01–05 July 1997, Barcelona, Spain, Vol. 3, pp. 1398 – 1404.

[6] P. Haberacker: Praxis Der digitalen Bildverarbeitung und Mustererkennung, Carl Hanser Verlag, Munchen, Germany, 1995.