

Satellite Image Edge Detection Using Fuzzy Logic

¹Mrs.R.Shenbagavalli, ²Dr.K.Ramar

¹Assistant Professor ,

Department of Computer Science,

Rani Anna Govt College for Women Tirunelveli

Research scholar Mother Teresa Womens University, KodaiKanal.

²Principal, Einstein College of Engg&Tech

Tirunelveli-8

Abstract

In this paper Edge detection is developed for satellite image using fuzzy logic concept. Fuzzy logic helps to find and highlight all the edges associated with an image by checking the relative pixel values .Scanning of an image using the windowing technique takes place 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. After the testing of fuzzy conditions the appropriate values are allocated to the pixels in the window under testing to provide an image highlighted with all the associated edges.

Keywords: Introduction, Image pre-processing, Edge Enhancement, Fuzzy Logic, Experimental Results.

Date of Submission: 04, December, 2012



Date of Publication: 05, January 2013

I. Introduction:

Edge detection is a fundamental tool in image processing and computer vision, particularly in the areas of feature detection and feature extraction, which aim at identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities. The contrast is improved by increasing the difference across discontinuities of the image components .In order to improve the differences ,we have to detect them .The edge detection algorithm is designed to detect and highlight these discontinuities. 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.

The purpose of detecting sharp changes in image brightness is to capture important events and changes. 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.

In the ideal case, the result of applying an edge detector to an image may lead to a set of connected curves that indicate the boundaries of objects, the boundaries of surface markings as well as curves that correspond to discontinuities in surface orientation.

Thus, applying an edge detection algorithm to an image may significantly reduce the amount of data to be processed and may therefore filter out information that may be regarded as less relevant, while preserving the important structural properties of an image.If the edge detection step is successful, the subsequent task of interpreting the information contents in the original image may therefore be substantially simplified.

Image Pre-Processing:

Pre-processing is an operation with images at the lowest level of abstraction-both input and output are intensity images. Image Neighbouring pixels corresponding to one object in real images have essentially the same or similar brightness value, if so, a distorted pixels can be picked out from the image. It can usually be restored as an average value of neighbouring pixels.

Horizontal and Vertical Filtering

The horizontal filter will give the high frequency components along horizontal edges in the image. Similarly the vertical filter will give the high frequency components along vertical edges in the image. The horizontal and vertical filter will be useful to extract the linear features like roads, railways, river path etc along horizontal and vertical directions.

FUZZY LOGIC:

Fuzzy logic, is a version of first-order logic which allows the truth of a statement to be represented as a value between 0 and 1, rather than simply True (1) or False (0).[4]

To perform image processing using fuzzy logic, three stages must occur. First image fuzzification is used to modify the membership values of a specific data set or image. After the image data are transformed from gray-level plane to the membership plane using fuzzification, appropriate fuzzy techniques modify the membership values. This can be a fuzzy clustering, a fuzzy rule-based approach, or a fuzzy integration approach. Decoding of the results, called defuzzification, then results in an output image. “The main power of fuzzy image processing is in the modification of the fuzzy membership values,”[1]

The image in the fuzzy domain is represented by $I_F(x,y)$.The fuzzy image is processed by a fuzzy set of operators \mathcal{F} yielding a new fuzzy image $I_F'(x,y)$.Finally,the fuzzy image is mapped back to the original domain through the de-fuzzification function D .[9]

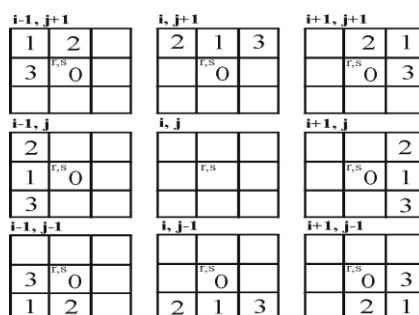
The processed image in the original domain is represented by $I(x,y)$:

$$F: I(x,y) \Rightarrow I_F(x,y) \quad (1)$$

$$\mathcal{F} [I_F(x,y)] \Rightarrow I_F'(x,y) \quad (2)$$

$$D: I_F'(x,y) \Rightarrow I'(x,y) \quad (3)$$

Fuzzy rules



Methodology:-

The image is said to have an edge if the intensity variation in between the adjacent pixels is large. There are many methods used for edge detection in an image . Roberts operator ,Prewitt operator,Sobel operator, Robinson operator,Kirsch operator,Laplace operator,Fuzzy operator.

In this paper,the fuzzy logic method has been used to detect an edge from the satellite image.The algorithm described below is based on the subjection of a set of nine pixels, part of a 3x3 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 FUZZY RELATIVE PIXEL VALUE ALGORITHM:

- Read an MxN image.
- The first set of nine pixels of a 3x3 window are chosen central pixel having values (2,2).
- After the initialization, the pixel values are subjected to the fuzzy conditions for edge existence as per the fuzzy rules in fig (a-h).
- After the subjection of the pixel values to the fuzzy conditions the algorithm generates an intermediate image.
- It is checked whether all pixels have been checked or now, if not then, first the horizontal coordinate pixels are checked. If all horizontal pixels have been checked then the vertical pixels are checked else the horizontal pixel is incremented to retrieve the next set of pixels of a window.
- After edge highlighting image is subjected to another set of condition with the help of which the unwanted parts of the output image are removed to generate an image which contains only the edges associated with the input image.

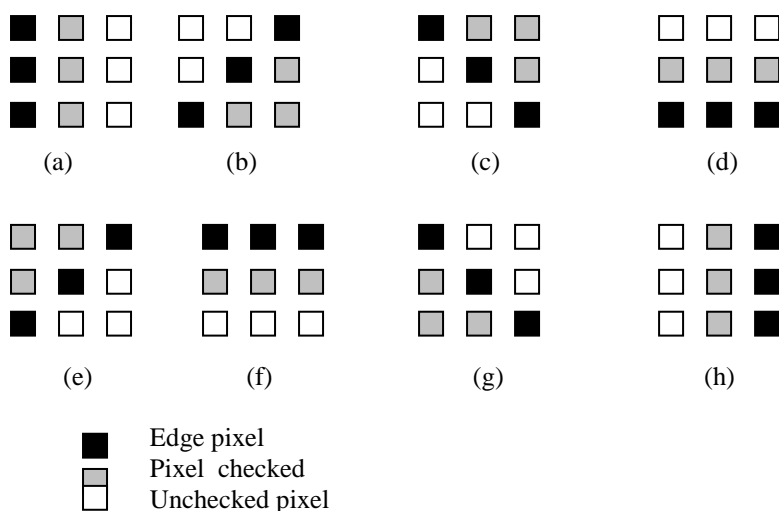
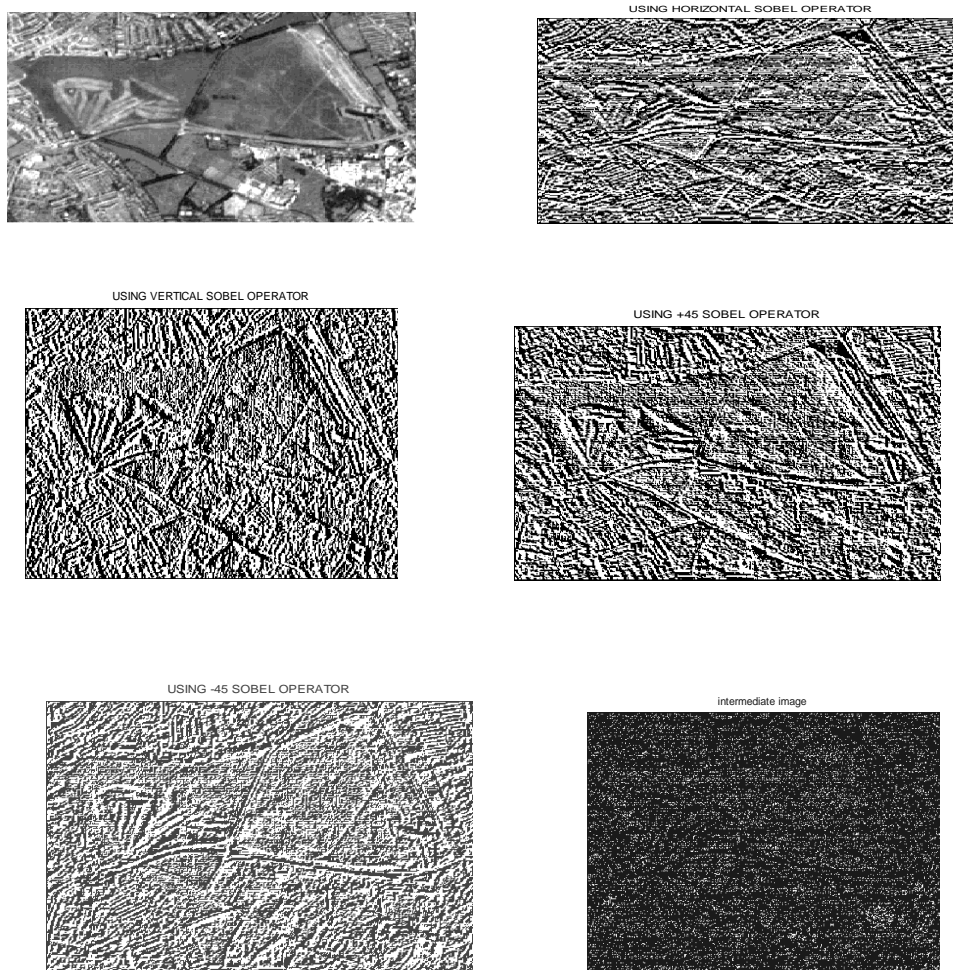
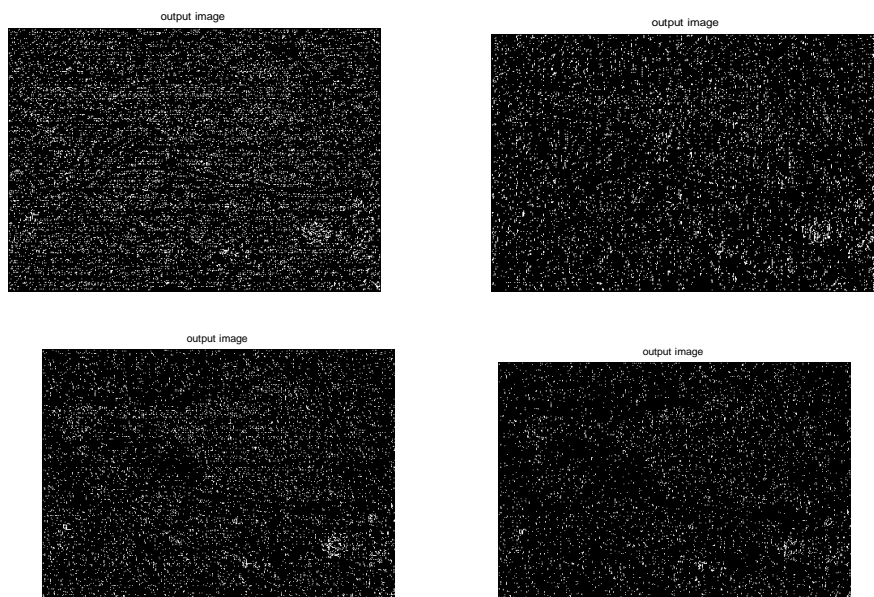


Fig 1 (a-h) fuzzy condition display

Experimental Results:

The fuzzy relative pixel value algorithm for image edge detection was tested for satellite images and the outputs were compared to the existing edge detection algorithm and it was observed that the outputs of this algorithm provide much more distinct marked edges and thus have better visual appearance than ones that are being used.





Conclusions:

In this paper Edge Detection and Edge Enhancement method has been described. The modified existing algorithm has been introduced to find the edges associated with an image. Fuzzy relative pixel value algorithm has been successful in obtaining the edges that are present in an image. Sample output have been shown to understand the accuracy of the algorithm.

References:

- [1]. Application of fuzzy logic on image edge detection by Shashank Mathur, Anil Ahlawat.
- [2]. H. H. Baker and T. O. Binford, "Depth from edge and intensity based stereo," in *Proc. Seventh Int. Joint Conf. Artificial Intelligence*, pp. 631–636, 1981.
- [3]. P. K. Saha, J. K. Udupa, and D. Odhner, "Scale-based fuzzy connected image segmentation: Theory, algorithms and validation," *Computer Vision and Image Understanding*, vol. 77, pp. 145–174, 2000.
- [4]. Mario I. Chacón. M "Fuzzy logic for image processing," *Advanced Fuzzy logic Techniques in industrial applications*, 2006.
- [5]. Terano T., Asai K., Sugeno M., *Fuzzy systems theory and its applications*.
- [6]. Image Processing and Edge Detection .Computer vision @Department of Computing ,Imperial College GZ Yang and DF Gillies
- [7]. Image processing and Edge detection by GZ Yang and DF Gillies, *Computer Vision* ,Department of Computing ,Imperial college
- [8]. Fuzzy Mathematics -Wikipedia
- [9]. Fuzzy Logic for Image Processing: Definition and Applications of a Fuzzy
- [10]. image Processing Scheme by Mario I. Chacón M

Coding:

```

clc;
clear all;
close all;
a=imread('E:\sip seminar\newcastle.jpg');
%a=imread('F:\ugc-appform_files\37870694_pic1.jpg');
display(a);
a=rgb2gray(a);
imshow(a)
a=double(a);
b=double(a);
[r c]=size(a);
for i=2:r-1

```

```
for j=2:c-1
  if a(i-1,j-1)>a(i-1,j)
    if a(i,j-1)>a(i,j)
      if a(i+1,j-1)>a(i+1,j)

        b(i-1,j)=0;
        b(i,j)=0;
        b(i+1,j)=0;

      end
    end
  end

  if a(i,j+1)>a(i-1,j+1)
    if a(i+1,j+1)>a(i,j)
      if a(i+1,j)>a(i+1,j-1)

        b(i-1,j+1)=0;
        b(i,j)=0;

        b(i+1,j-1)=0;
      end
    end
  end

  if a(i-1,j+1)>a(i-1,j)
    if a(i,j+1)>a(i,j)
      if a(i+1,j+1)>a(i+1,j)

        b(i-1,j)=0;
        b(i,j)=0;
        b(i+1,j)=0;

      end
    end
  end

  if a(i+1,j+1)>a(i,j+1)
    if a(i+1,j)>a(i,j)
      if a(i+1,j-1)>a(i,j-1)

        b(i,j+1)=0;
        b(i,j)=0;
        b(i,j-1) = 0;

      end
    end
  end

  if a(i-1,j+1)>a(i,j+1)
    if a(i-1,j)>a(i,j)
      if a(i-1,j-1)>a(i,j-1)
```

```

        b(i,j+1)=0;
        b(i,j)=0;
        b(i,j-1)=0;

    end

    end

    end

    if a(i+1,j)>a(i+1,j+1)
        if a(i+1,j-1)>a(i,j)
            if a(i,j-1)>a(i-1,j-1)

                b(i+1,j+1)=0;
                b(i,j)=0;
                b(i-1,j-1) = 0;

            end
        end
    end

    if a(i-1,j+1)>a(i,j)
        if a(i,j+1)>a(i+1,j+1)
            if a(i-1,j)>a(i-1,j-1)

                b(i,j)=0;
                b(i+1,j+1) = 0;
                b(i-1,j-1)=0;
            end
        end
    end

    if a(i-1,j)>a(i-1,j+1)
        if a(i-1,j-1)>a(i,j)
            if a(i,j-1)>a(i+1,j-1)

                b(i-1,j+1)=0;
                b(i,j)=0;
                b(i+1,j-1) = 0;

            end
        end
    end

    end
end
display(b);
figure,imshow(b),title('b image');
for i=2:r-1
    for j=2:c-1
        if (b(i-1,j)== 255 & b(i,j)==0 & b(i+1,j) == 255 & b(i,j-1)==255)
            b(i,j)= 255

        end
    end
end
figure,imshow(b),title('output image');

```