A Novel Approach for Edge Detection

S.Lakshmi†, Dr.V.Sankaranarayanan††

†Assistant Professor, Jeppiaar Engineering College, Chennai, India
††Director, Crescent Engineering College, Director (Rtd.), RCC, Anna University, Chennai, India

Summary
A novel edge detection algorithm is presented in this paper. The proposed method is based on the neighborhood similarity of a pixel using a pre-defined intensity range and median filter which can be implemented easily and effectively. Many experiments were carried out to evaluate and compare the performance of the proposed algorithm. This new detector outperforms the previously available classical edge detectors.

Key words:
Image processing, Noise Cleaning, Median Filter and Edge detection

1. Introduction

Since the edges confer the shape for the object, there are plenty of algorithms in the literature for processing the image or scene. This is the forefront of all the subsequent progression in image processing. Color images provide more information than gray scale images. There are numerous techniques available in the literature to detect meaningful edges based on the approaches such as genetic algorithms, fuzzy logic, and neural network, Bayesian approach, wavelet domain, mathematical morphology and so on. Research in automatic edge detection is an active field because it is used in many different applications in image processing such as diagnosis in medical imaging, topographical recognition and automated inspection of machine assemblies.[1]. Edge functions look for rapid changes in intensity. When the first derivative of the intensity is larger in magnitude than some threshold, or the second derivative of the intensity has a zero crossing, this often suggests an edge is present, because an edge is a curve that follows a path of rapid change in image intensity [2]. Edges are often associated with the boundaries of objects in a scene [3]. Traditional approaches using classical edge detectors fail when the images are noisy.

The organization of this paper is as follows: The image noise and the classification of noise reduction filters are explained in section 2. The noise pixels are identified in the first stage. Then the noise reduction technique is formulated in section 3. Section 4 shows the edge detection process. Section 5 provides the number of experimental results to show the performance of the new edge detection technique. Finally conclusions are depicted in Section 6.

2. Types of Noise and its Classification

Noise is any unwanted information that infects images easily. Even though we can not see the noise when we look at an image or video, there is some form of noise present in the image which degrades the quality of the image considerably. Different kinds of noise can be produced from various typical sources which include flecks of dust on the lens or inside the camera, or with digital cameras, faulty CCD elements. Some of them are random noise, fixed pattern and banding noise. Random noise is characterized by intensity and color fluctuations above and below the actual image intensity. Fixed pattern noise surrounds hot pixels. Hot pixels are pixel bits that are more intense than others surrounding it and are much brighter than random noise fluctuations. Long exposures and high temperatures cause fixed pattern noise to appear. Banding noise can also increase for certain white balances, depending on camera model [4].
In digital images, the Gaussian and impulse noises played a major role to contaminate and sometimes ruin the image or photograph fruitfully. Impulse noise is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware or transmission in a noisy channel. In salt and pepper noise (also known as random noise or independent noise or speckle), pixels in the image are vastly different in color from their surrounding pixels. The defining characteristic is that the color of a noisy pixel bears no relation to the color of surrounding pixels. Generally this type of noise will only affect a small number of image pixels. When viewed, the image contains dark and white dots, hence the term salt and pepper noise. In Gaussian noise (dependent noise), an amount of noise is added to every part of the picture. Each pixel in the image will be changed from its original value by a (usually) small amount. Taking a plot of the amount of distortion of a pixel against the frequency with which it occurs produces a Gaussian distribution of noise [5].

Manuscript received April 5, 2010
Manuscript revised April 20, 2010

2.1. Noise in Digital Image

Let us consider an image I of size m x n and an inspected image X of same size.

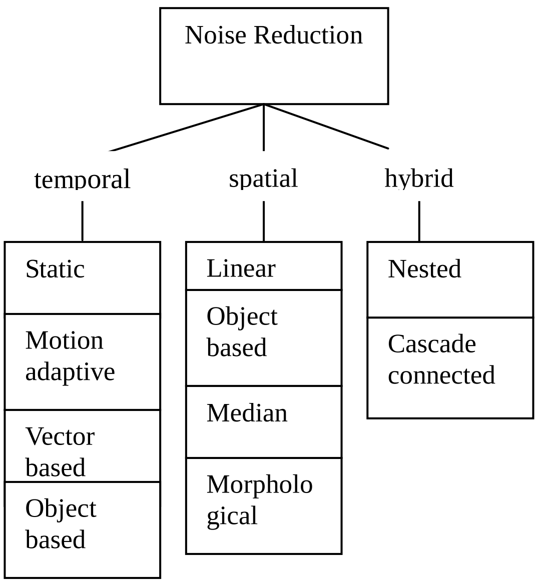
$$X = I + N$$

where N is noise with random values.

$$X_{i,j} = \begin{cases} N_{i,j} & \text{with Probability } P \\ I_{i,j} & \text{with Probability } 1-P \end{cases}$$

where i = 1,2,…..m and j = 1,2,…..n and 0<= P <= 1.

2.2. Classification of Noise Filter



The special filters are mainly classified into two groups, ie., non-adaptive and adaptive filters. Non-adaptive filters are neglecting the local properties of the image whereas the adaptive filters are considering the local properties to remove the noise from the image.

2.3 Noise Reduction Technique

Reduction of noise is one of the most important processes to raise the quality of the image. The design of an efficient, robust and computationally effective edge preserving denoising algorithm is a widely studied and yet unsolved problem [5].To recover the image from its noise there exits many filtering techniques which are application oriented. Some filtering techniques have better performance than the others according to noise category.

2.3.1. Some of the Existing Filtering Techniques

To reduce the noise from the image there exist many mean filtering techniques which are described below.

1. Arithmetic Mean Filtering Technique:

This is a simple and straightforward mean filtering technique. Let G be the given image and pixel x be a noisy pixel. The noisy pixel should have the value 0(black) or 255(white). Taking a moving kernel window of size m x m centered the pixel x, Compute the average value of the corrupted image I and replace the corrupted pixel value by the average value.

2. Geometric Mean Filtering Technique:

In this technique each corrupted pixel is replaced by the pixel which value can be calculated by using the geometric formula.

3. Harmonic Mean

This technique gives good result for salt noise and fails for pepper noise.

4. Median Filtering Technique

This technique gives very good results when we compare with other filtering techniques. medfilt2() command is used in matlab to filter the noise from the given image. In this technique the middle value will be taken to remove the noisy pixel in the m x n filtering window.
For removing the noise from the image, the median filter gives the better results and it overcomes all three mean filtering techniques.

3. Proposed Approach

When taking the noisy pixel values ie., 255 to calculate arithmetic mean in the filtering window, it provides some erroneous result. In the proposed approach arithmetic mean will be calculated by ignoring the noisy pixel ie. 0 and 255. So this new approach outperforms the existing methods.

Calculation of Intensity

The intensity of each and every pixel in the RGB color space of the input image by using the formula I = R+G+B/3.

Identify Noisy Pixel

Generally, the pixels are divided into four categories, They are the foreground’s pixels, the background’s pixels, the isolated noise pixels and the noise points attached to image's margins.[6].

For 8 bit images, if a pixel is corrupted, it is replaced by positive and negative impulse values. For example, the noisy pixel takes '0' for a negative impulse and '255' for a positive impulse.

Let x(i,j) be the pixel value on the position (i,j) in the input image G, if x(i,j) = 0 or 255, then we set C(i,j) = 1 .Otherwise C(i,j) = pixel value.

Algorithm

Level 1: Steps to remove the Noise

1: For each pixel x in the image do

i. Take a moving kernel window of size mxn (3x3) around the pixel x.
ii. Find out the Arithmetic mean from the pixels of the kernel ignoring the pixels with the values 255 and 0.
iii. Calculate the Arithmetic mean value after ignoring the maximum and minimum value of the pixels i.e., noisy pixel.
iv. Replace the pixel x with the calculated arithmetic mean.
v. We get the noise free image

4. Edge Detection Process

Level 2: Steps to detect Edges

Two basic properties of image intensity values are the basis for current segmentation algorithms; similarity and discontinuity [3]. The principal approaches in the similarity category are based on partitioning an image into regions that are similar to a set of predefined criteria. The approach for discontinuity algorithms is to partition an image based on abrupt changes in intensity.

Based on the following assumptions we can prove that the detected edges are good enough for further image processing applications.

1. Similarity of pixel values and
2. Pre-delineated intensity value

Let us assume that L represents a list of pixel values of the m x n (7x7 or 9x9) moving kernel window in the input image. Using the pre-delineated intensity value P, the list L is partitioned into two sub lists L1 & L2 delimited by the middle edge pixel 'x_m' where m = 1,2,….n. First we have to calculate the median. Here the x_m is the median value for the filtering window.

i. Take a moving kernel window of size mxn (7x7 or 9x9) around the pixel x.
ii. Calculate the median from the pixels of the kernel that to be placed as the middle element of the kernel.
iii. Assign a value to the variable P for ex., 20, 25, which may or may not be less than the total number of similar pixels in the kernel.

To identify the edge pixels, check the following condition

```
if ((x_i <= (x_m + P)) &&
        (x_i >= (x_m – P)))
            c[i,j] = 1
else
            c[i,j] = 0
```

iv. Now we can get the edge pixels which have the value 1 and all other pixels have the value 0.Here we are getting thick edges that means the edge pixel array can also consist of other pixels that satisfies the condition in the image. To avoid that we have to calculate the similarity measure by using the formula S = P/No. If the ratio of S <= 1 we get the thin edges or otherwise we get very thick edges.
v. Finally we get the output image framed by using the detected edge pixels which forms the boundaries of the object.

5. Experimental Results

The proposed approach is tested using the Lena, peppers, flower, baboon and parrot color images. Fig 1 shows the original image. Fig 2 is the grayscale image which is the conversion of the original image.Fig3 is the salt and pepper noise added image. Fig 4 shows the output image after removing the salt and pepper noise .Fig 5 is the edge image without taking the similarity ratio S = P/No. The value of S should be approximately equal to 1. If the value of S is greater than 1, we will get the thick edges. Fig 6 is also the edge image taking the similarity ratio. Figures from 7 to 10 are edge images using prewitt, log, sobel and canny operators to detect edges. When we use general arithmetic mean filtering technique to clear the noise, the output image is not at all a pleasing one. However, in this proposed arithmetic mean filtering technique, the output will be a visually pleasing one.

Fig-1 Original Image

Fig-2 Grayscale Image

Fig-3 Salt and Pepper Noisy Image

Fig-4 Proposed Arithmetic Mean Filter

Fig-5 Detected Edges by Proposed Approach

Fig-6 Detected Edges by Proposed Approach

Original Image

Proposed Arithmetic Mean Filter

Original Image

Proposed Edge Detector

Comparison Results

Fig-7 Prewitt

Fig-8 Log

Fig-9 Sobel

Fig-10 Canny

Conclusion

In this paper we have developed a new filtering technique for noise removal which is better than the existing filtering techniques for salt and pepper noise.

While removing the noisy pixel we should preserve the details of edge information as well as spatial resolution. The proposed filtering algorithm meets these conditions without any negligence. Using the new filtering technique, the edges of the object is detected by using the similarity criteria. In future, they can be upgraded them according to achieve better performance.

References

[1] Evelyn Brannock, Michael Wees,"A Synopsis of Recent Work in Edge Detection using the DWT", IEEE, 2008, pages 515 -520.

[2] Matlab Documentation, *Image Processing Toolbox User's Guide*, Release 7.1, The MathWorks, Inc. 3 Apple Hill Drive, Natick, MA 01760-2098, 2007.

[3] James R. Parker, *Algorithms for Image Processing and Computer Vision*, JohnWiley and Sons, Inc., 1997 pages 1-28.

[4] Rafeal C. Gonzalez, and Richard E. Woods, *Digital Image Processing*, Prentice Hall, 2007.

[5] "Patch-Based Bilateral Filter and Local M-Smoothere for Image Denoising",ESANN'2009 proceedings, European Symposium on Artificial Neural Networks – Advances in Computational Intelligence and Learning. Bruges (Belgium),Apr 2009.

[6] YANG Ji-hong, ZHANG Min," A New Filtering Algorithm for Removing Salt- and – Pepper Noise",2009, International Conference on Environmental Science and Information Application Technology, Pages 355-358.

[7] N.R.Mohtar,Nor Hazlyna Harun, M.Y.Mashor,H.Roseline, Nazahah Mustafa, R.Adollah,H.Adilah,N.F.Mohd Nasir ," Image Enhancement Techniques Using Local,Global,Bright,Dar and Partial Contrast Stretching For Acute Leukemia Images", Proceedings of the World Congress on Engineering,Vol I,WCE, July 1 - 3,2009,London,U.K.

[8] Dong Hu,Xianzhong Tian,"A Multidirections algorithm for Edge Detection Based on Fuzzy Mathematical Morphology",Proceedings of the 16th International Conference on Artificial Reality and Telexistence – Workshops (ICAT'06),2006,IEEE