CHAPTER-1

INTRODUCTION

In today's world the visual information is transmitted in the form of digital images, but the image obtained after transmission is often degraded with some unwanted pattern or information called noise. The received image needs processing by applying linear or nonlinear filters [1] in order to restore to the original content before using in any applications. In digital image processing the edge detection algorithms are mainly used for image sharpening. The edge detection algorithm is designed to detect and highlight the discontinuities in the image intensity values. Edge detection has acquired enormous importance in computer vision research and for classification and recognition of objects. Edges consist of meaningful features used to reduce the amount of image size significantly and filter out the information that may be regarded as less relevant, preserving important structural properties of an image. Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. Edge detection is a challenging task in case of noisy images, since both the noise and the edges contain high frequency content.

In the field of image processing,the modern trend moves towards handling images effectively with more clarity and high performance.This image enhancement can be done either in spatial domain or in frequency domain.Low pass,high pass, band pass and notch filters are designed to remove the noise from the image or to enhance the quality of the image.

Currently, with the advances in electronic technologies, digital images have been used widely in our daily life. Unfortunately, digital images are sometimes corrupted by noises, which can significantly reduce the quality of the image.Digital images play an important role both in daily life applications such as Satellite television, Magnetic Resonance Imaging, Computer Tomography as well as in areas of research and technology such as geographical information systems and astronomy . Digital images are often corrupted by different types of noise during its acquisition and transmission phase.To enhance the quality of images various image enhancement or restoration techniques are used. Efficiency of every method depends on the quality of input images.

Image enhancement plays a vital role in the field of digital image processing since the noise is added very often with the original image. spatial filtering techniques like low pass,high pass,bandpass and notch with the help of convolution mask are often used to enhance the image with reduced noise.

The overall noise characteristics in an image depend on many factors, including the type of sensor, pixel dimensions, temperature, exposure time, and speed.The goal of image denoising is to remove the noise while retaining the important signal features. The denoising of a natural image corrupted by noise is an important problem in image processing. Image denoising still remains a challenge for researchers because noise removal introduces artifacts and causes blurring of the images. Normal edge detecting methods will yield a good performance when dealing with the clear images, i.e image with less or no noise.

The main problem comes here when there exists an image with impulse noise then, their performance will be degraded when we use normal edge detecting methods. Noise remains a ubiquitous and unwanted phenomenon that is inherent to many image acquisition. The noise we find in the image is due to a fundamental step in the image process called image acquisition.

Image acquisition is the action of retrieving an image from the source, usually a hardware based source for processing.In order to remove this type of noise , smoothing filters are often applied to the image to decrease the variance of the noise, in order to preserve as much as possible important details in the image.

Denoising the image still encounters some challenges when faced with high impulse noise that include loss of image details,blurring of the image and unsmoothed edges , which makes the edge detection process more difficult to attain.

The motivation in this is driven by following two rules:

- We have to resolve the challenges faced with the denoising methods by keeping as many details as possible , avoid blurring of the image , and preserve the sharper edges associated with boundaries.
- Contend with these challenges even in the presence of high-intensity impulsive noise.

If the filter or kernel size is small, filtering can be done more effectively in the spatial domain.if the kernel or convolution mask size becomes large,this would be tedious in the spatial domain.In such a case,the filtering could be done more effectively in the frequency domain.since the convolution becomes multiplication here,the processing would be done more effectively even with large kernel size.

There are several denoising and edge preservation methods that have been proposed
in the past. The filters that are commonly used includes;

- Standard median filter
- Total variation filter
- Bilateral filter

1.1 APPLICATIONS

Edge detection is essential for the following applications:

- object identification
- image segmentation
- feature extraction
- Pattern recognition

Object Identification:

Object identification is a very important part of automation. So one should first learn how to identify the objects. Object Identifiers or Locators are used to uniquely identify the objects used in the Application under Test while creating the test cases. Objects being identified accurately demands certain steps to be followed and is one of the most time consuming and error prone topic of automation.

Image Segmentation:

Image Processing or more specifically, *Digital Image Processing* is a process by which a digital image is processed using a set of algorithms. It involves a simple level task like noise removal to common tasks like identifying objects, person, text etc., to more complicated tasks like image classifications, emotion detection, anomaly detection, segmentation etc.

Feature Extraction:

Feature extraction is a part of the dimensionality reduction process, in which, an initial set of the raw data is divided and reduced to more manageable groups. So when you want to process it will be easier. The most important characteristic of these large data sets is that they have a large number of variables. These variables require a lot of computing resources to process. So Feature extraction helps to get the best feature from those big data sets by selecting and combining variables into features, thus, effectively reducing the amount of data. These features are easy to process, but still able to describe the actual data set with accuracy and originality.

Pattern Recognition:

Pattern recognition is the process of recognizing patterns by using a machine learning algorithm. Pattern recognition can be defined as the classification of data based on knowledge already gained or on statistical information extracted from patterns and/or their representation. One of the important aspects of pattern recognition is its application potential.

1.2 LITERATURE SURVEY:

Canny edge detection [1], perhaps one of the most useful and well-known method, is a multifaceted process that integrates Gaussian filtering for smoothing the image, intensity gradient, non-maximum suppression for edge thinning, thresholding and tracking of the edges to ensure edge connectivity and continuity. The holistically nested edge detection (HED) method [2], which is a robust edge detection method, uses convolutional neural networks and is based on image training and prediction through multi-scale and multi level feature learning. Such edge detection methods and related edge operators extract quite successfully edge information and yield a good performance when dealing with clean images; however, their performance is degraded in the presence of impulse noise, especially when it is of high intensity type. Such degradation could be overcome, but only with additional well thought out filtering steps. Neuro –fuzzy operator [3] is designed to detect edges in the presence of impulse noise, but its success is limited only for low intensity noise levels. A fast algorithm that detects edges in noisy images is proposed in [4], but preserving image details under different noise intensities was not its main focus.

There are several image denoising and edge-preserving methods that have been proposed in the past, with in-depth surveys on them provided in [5] and [6]. The filters that are commonly used include the standard median filter [7], total variation (TV) filter [8, 9], anisotropic diffusion filter [10, 11], bilateral filter [12], guided filter [13] and non-local mean filter [14]. Empirical evaluations reveal that the median filter performs best in the presence of impulse noise. There are some studies [15, 16] that were proposed for improving the performance of the median filter in high-intensity impulse noise. A comprehensive survey on switching median filters [17] provides a comparative assessment of denoising filters such as the standard median filter [7], the center weighted median filter (CWMF) [18], the weighted median filter [19], the adaptive switching median filter (ASMF) [20] and the modified decision based unsymmetric trimmed median filter (MDBUTMF) [21]. The results reported indicate that MDBUTMF [21] is better among them. Also, there are other switching based filters that were introduced in [22-25].

(ANDWP)[31] filter is used for removal of highly random valued impulse noise (RVIN). The proposed approach works in two phases. The first phase detects the contaminated pixels by making the differences between the test pixel and its all neighbor pixels aligned in four main directions in the 5 x 5 window. The second phase filters only the noisy pixels based on minimum variance of the four directional pixels. In literature Wilcoxon Test [9], [13] to detect edge pixels from a noisy image, find out the median of the neighbourhood pixels of centered pixel. Based on that statistic value decision will be made whether the pixel is edge pixel or not.

In literature [32]-[33], [34]-[36] to detect the edges of the noisy images first we need to perform regularization by selecting suitable filters to reduce the appropriate noise. Attempts to reduce the noise results in blurred, distorted edges and sometimes edge pixels may disappear. Operators used in noisy images are typically larger in scope, so they can average enough data to discount localized noisy pixels. Later needs to apply suitable edge detector [1],[37] to identify the edge of the restored image.

Classical methods of edge detection involve convolving the image with two-dimensional filters, which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions.Image edge detection is a process of locating the edges in an image which is important in finding the appropriate absolute gradient magnitude at point of an input image.

Detection of an edge in an image processing is an important step to understand the image features. Edges consist of meaningful features used to reduce the amount of image size significantly and filters out the information that may be regarded as less relevant ,preserving important structural properties of an image. Applying edge detection to image may significantly reduce the amount of data to be processed, solves the problem of the high volume of space images occupy in a computer memory. The problem of storage, transmission over the internet and bandwidth could be solved when edges are detected. Edge detection is a technology in image processing and computer vision, in the area of feature detection and feature extraction, which aims at identifying points in a digital image at which image brightness changes sharply or more formally, has discontinuities.

Edge detection mechanisms are used mainly in the following application areas Computer Vision, Medical image processing and Vehicle plate recognition. In a computer vision, Edge detectors are applied in order to identify the object or to recognize the objects based on recognition properties. In a medical image processing, the detection of tissue borders is of great importance in several MRI applications. The edge detection methods apply linear filters or non linear filters in order to detect edges. The derivative edge detection methods [32] covers to obtain the edges from original images directly which cannot obtain if applied directly on noisy images and methods [38]-[40] listed may obtain edge images directly from a noisy image without doing regularization.

1.3 TYPES OF NOISES:

Real images are often degraded by some random errors,this degradation is called as noise. The g(x,y) is a noisy image obtained by adding n(x,y) noise to the original image f(x,y).Noise can occur during image capture, transmission, or processing and may be dependent or independent on image content.
There are many kinds of noises and some of them are listed below:

Salt and Pepper Noise: This is also called data drop noise because statistically its drops the original data values. This noise is also referred to as salt and pepper noise. However the image is not fully corrupted by salt and pepper noise instead some pixel values are changed in the image.

This noise is seen in data transmission. Image pixel values are replaced by corrupted pixel values either maximum or minimum pixel value i.e., 255 ‘or’ 0 respectively.Salt and Pepper noise is added to an image by addition of both random bright (with 255 pixel value) and random dark (with 0 pixel value) all over the image.

Salt and Pepper noise generally corrupted the digital image by malfunctioning of pixel elements in camera sensors, faulty memory space in storage, errors in digitization process and many more.

Gaussian noise:It is also called electronic noise because it arises in amplifiers or detectors. Gaussian noise caused by natural sources such as thermal vibration of atoms and discrete nature of radiation of warm objects.

Gaussian noise generally disturbs the gray values in digital images. That is why the Gaussian noise model is essentially designed and characterized by its PDF or normalizes histogram with respect to gray value.
The Probability Density Function (PDF) of a Gaussian random variable z .

$$p(z) = (1/\sqrt{2\pi}\,\sigma)\ e^{-(z-\mu)2}/2\sigma^2$$

Where g = gray value, σ = standard deviation and μ = mean.

Impulse noise: This is also known as shot noise or binary noise also. It is caused by sudden disturbances in image signal. The noise is mostly caused by sensor and memory problems due to which the pixels are assigned incorrect minimum or maximum values.

The PDF is given in :

$$p(z) = \begin{cases} P_a & \text{for } z=a \\ P_b & \text{for } z=b \\ 0 & \text{otherwise.} \end{cases}$$

Photon Noise (Poisson Noise) : The appearance of this noise is seen due to the statistical nature of electromagnetic waves such as x-rays, visible lights and gamma rays. The x-ray and gamma ray sources emitted a number of photons per unit time.

These rays are injected into a patient's body from its source, in medical x-rays and gamma rays imaging systems. These sources are having random fluctuations of photons. The Resulting gathered image has spatial and temporal randomness. This noise is also called quantum (photon) noise or shot noise.

Brownian Noise (Fractal Noise): Colored noise has many names such as Brownian noise or pink noise or flicker noise or 1/f noise. In Brownian noise, power spectral density is proportional to square of frequency over an octave i.e., its power falls on ¼ the part (6 dB per octave).

Brownian noise caused by Brownian motion. Brownian motion seen due to the random movement of suspended particles in fluid. Brownian noise can also be generated from white noise.

Periodic Noise : This noise is generated from electronics interferences, especially in power signal during image acquisition. This noise has special characteristics like spatially dependent and sinusoidal in nature at multiples of specific frequency.

It appears in the form of conjugate spots in the frequency domain. It can be conveniently removed by using a narrow band reject filter or notch filter.

Structured noise: Structured noise is periodic, stationary or non stationary and aperiodic in nature. If this noise is stationary, it has fixed amplitude, frequency and phase. Structured noise caused by interferences among electronic components. Noise presents in communication channels are in two parts, unstructured noise (u) and structured noise (s).

structured noise is also called low rank noise. In signal processing, it is also more advantageable (more realistic) to consider noise models in a lower dimensionality space.

Speckle Noise: Speckle noise is considered as multiplicative noise. It is a granular noise that degrades the quality of images obtained by active image devices such as active radar and synthetic aperture radar (SAR) images. Due to random fluctuations in the return signal from an object in conventional radar that is not big as a single image processing element, speckle noise occurs.

Quantization noise: appearance is inherent in the amplitude quantization process. It is generally presented due to analog data converted into digital data.

Speckle Noise: This noise is multiplicative noise. Their appearance is seen in coherent imaging systems such as laser, radar and acoustics etc,. Speckle noise can exist similarly in an image as Gaussian noise. Its probability density function follows gamma distribution.

1.4 OBJECTIVE

1. To review the algorithms in the literature.
2. To study various noise models.
3. To study the algorithm for different Edge operators.
4. To develop new method for denoising and identify efficient edges.

1.5 SCOPE OF THE WORK

To achieve the above mentioned objectives the book is divided into five chapters.

The first chapter deals with Introduction, literature survey, objective of work and scope of the work. The second chapter gives the study of denoising algorithms. The third chapter discusses the edge detection methods using first order derivative, second order derivative and canny edge detection. The new algorithm is presented in chapter four. The subjective and objective analysis using the current algorithm is discussed in the fifth chapter. Conclusion is given in sixth chapter.

CHAPTER 2

DENOISING ALGORITHMS

The main aim of an image denoising algorithm is to achieve both noise reduction and feature preservation using the filters.
Types of denoising algorithms:

1. Mean Filter
2. Median Filter

Image denoising is to remove noise from a noisy image, so as to restore the true image. However, since noise, edge, and texture are high frequency components, it is difficult to distinguish them in the process of denoising and the denoised images could inevitably lose some details.

2.1 MEAN FILTER

2.1.1 INTRODUCTION

Mean filtering is a simple, intuitive and easy to implement method of smoothing images, i.e. reducing the amount of intensity variation between one pixel and the next. It is often used to reduce noise in images.

The idea of mean filtering is simply to replace each pixel value in an image with the mean value of its neighbors, including itself. This has the effect of eliminating pixel values which are unrepresentative of their surroundings. Mean filtering is usually thought of as a convolution filter. Like other convolutions it is based around a kernel which represents the shape and size of the neighborhood to be sampled when calculating the mean.

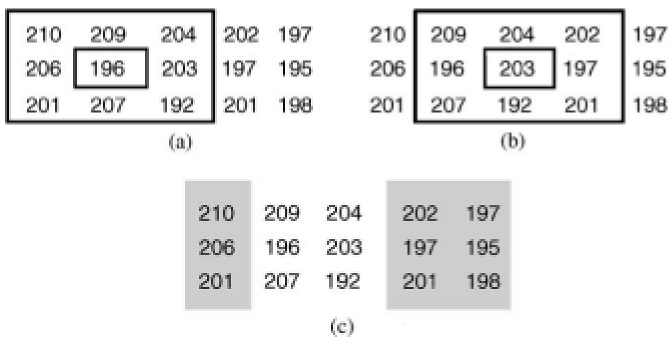
2.1.2 METHODOLOGY

Mean filtering is usually thought of as a convolution operation as the mask is successively moved across the image until every pixel has been covered.

Consider an image F ($P \times Q$), whose pixel gray levels are stored in a 2-D array say, Image[P][Q] such that image[0][0] and image[$P - 1$][$Q - 1$] contains the first and last pixels, respectively.In order to apply the mean-filter to this image over a rectangular neighborhood window $m \times n$ (m & n are odd positive integers), the image must be appropriately padded by some method like replication of boundary rows and columns [1] at the four sides of the input image.

ALGORITHM

Consider a pixel image[row][column]
For row = 0 to p-1
    For column=0 to q-1
        sum=sum of m x n pixels in neighborhood of considered pixels
    Image[row][column] = sum/(m x n)
This algorithm is applied to each and every pixel.
Consider the below example for better understanding



(a) Neighbourhood around 196 (b) neighborhood around 203
(c) un-shaded portion represents 6 out of 9 common pixels

2.1.3 RESULTS AND DISCUSSIONS

Figure. (a) Noise image

Figure. (b) output of mean filter

Fig 2.1.3.1 (a) is the image which is having the high impulse noise.
(b) is the image after processing with the mean filter.

If we observe both the images clearly, then we can see the image which is output of the mean filter has some blur i.e due to the loss of some information in the image.So if the intensity of noise is more in the image then applying the mean filter to that noise image leads to some information loss in it.

There are some potential problems with the mean filter:

1. A single pixel with a very unrepresentative value can significantly affect the average value of all the pixels in its neighborhood.
2. When the filter neighborhood straddles an edge, the filter will interpolate new values for pixels on the edge and so will blur that edge.This may be a problem if sharp edges are required in the output.

Mean filters it is found that the noise has been removed, and at the same time the image has got blurred. As the size of the window increases, the ability to remove noise increases at the expense of blurring of the image.

This has the effect of eliminating pixel values which are unrepresentative of their surroundings. Mean filtering is usually thought of as a convolution filter. Like other convolutions it is based around a kernel, which represents the shape and size of the neighborhood to be sampled when calculating the mean. Pixels values which are lying on the edges are not affected by zero padding and the noise which are lying on the edges are easily removed.

It is generally a sliding-window spatial filter that replaces the central value of the square window with the mean of all pixels values and the great disadvantage of this method is that it will make the picture blur. Pixels value which are lying on the edges are generally reduced due to the zero padding and the other great disadvantage of this filter is that it takes the mean even if the image is noise free.

2.2 MEDIAN FILTER

2.2.1 INTRODUCTION

The median filter is a nonlinear statistical filter that replaces the current pixel value with the median value of pixels in the neighboring region.

The intensity values of pixels in a small region within the size of the filter are examined, and the median intensity value is selected for the central pixel. Removing noise using the median filter does not reduce the difference in brightness of images, since the intensity values of the filtered image are taken from the original image.

The median filter is normally used to reduce noise in an image, somewhat like the mean filter. However, it often does a better job than the mean filter of preserving useful detail in the image.

2.2.2 METHODOLOGY

Mean filtering is usually thought of as a convolution operation as the mask is successively moved across the image until every pixel has been covered.

Consider an image ($P \times Q$), whose pixel gray levels are stored in a 2-D array say, Image[P][Q] such that image[0][0] and image[P − 1][Q − 1] contains the first and last pixels, respectively.In order to apply the median-filter to this image over a rectangular neighborhood window $m \times n$ (m & n are odd positive integers), the image must be appropriately padded by some method like replication of boundary rows and columns [1] at the four sides of the input image.

ALGORITHM

```
Consider a pixel image[row][column]
For row = 0 to p-1
        For column=0 to q-1
                array= append every pixels in the neighborhood of  considered pixel
                Sort the array
                Take the middlemost element and replace the considered pixel with median
        Image[row][column] = array[5]
```

This algorithm is applied to each and every pixel present in the image.

123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

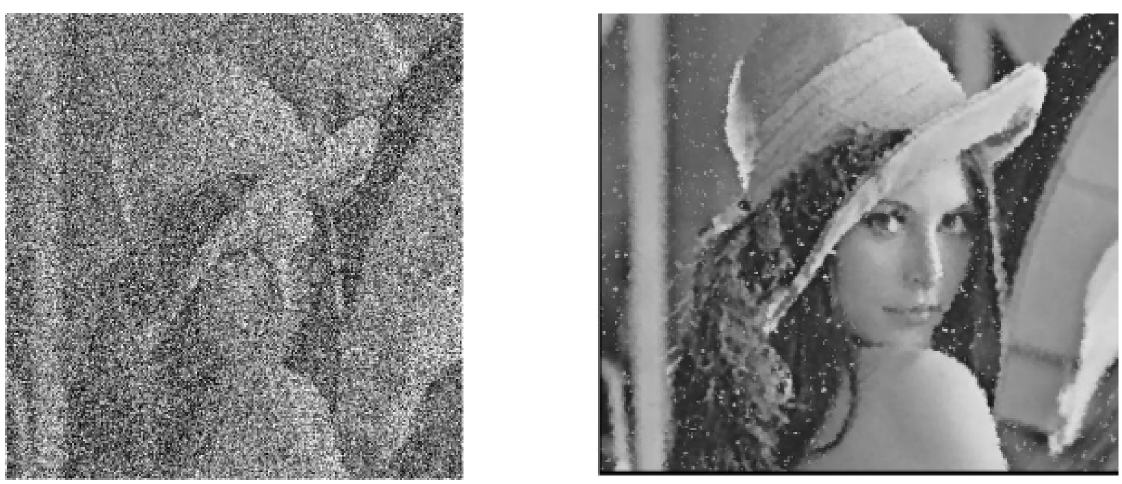
Neighbourhood values:

115, 119, 120, 123, 124,
125, 126, 127, 150

Median value: 124

Consider the above example for better understanding
1.choose the pixel in the above example it is 150.
2.Take the neighborhood of pixel 150 including itself.
3.sort all the values in the neighborhood of the current pixel.
4.take the middle value and then replace it with the current pixel.

2.2.3 RESULTS AND DISCUSSIONS



(a) (b)

Fig 2.2.3.1 (a) is the image which is having the high impulse noise.
(b) is the image after processing with the median filter.

(a)

(b)

Fig 2.2.3.2 (a) is the image which is having the high impulse noise.
(b) is the image after processing with the median filter.

If we observe both the images then we can find that the majority of the noise is removed , even though the intensity of noise in the image is very high,and also it makes all the data or information of the image keep as it is.

As there is no blur in the image after processing with the median filter it clearly means that no data or information about the images is lost during the processing with the median filter.

Considering the feature of local pixel distribution in the original image, the proposed method extracts noise pixels from the local pixel value, and makes full use of the filtering feature and the correlation between adjacent pixels.

2.3 SUMMARY

The mean filter is a filter which uses a mask over each pixel in the image. Each of the components of the pixels which fall under the mask are averaged together to form a single pixel. This new pixel is then used to replace the pixel in the image studied. The Mean Filter is poor at maintaining edges within the image.

A Median filter is a non-linear filter and is efficient to remove impulse noise.Median tends to preserve the sharpness of image edges while removing noise.The Median Filter is performed by taking the magnitude of all of the vectors within a mask and sorting the magnitudes.The pixel with the median magnitude is then used to replace the pixel studied.

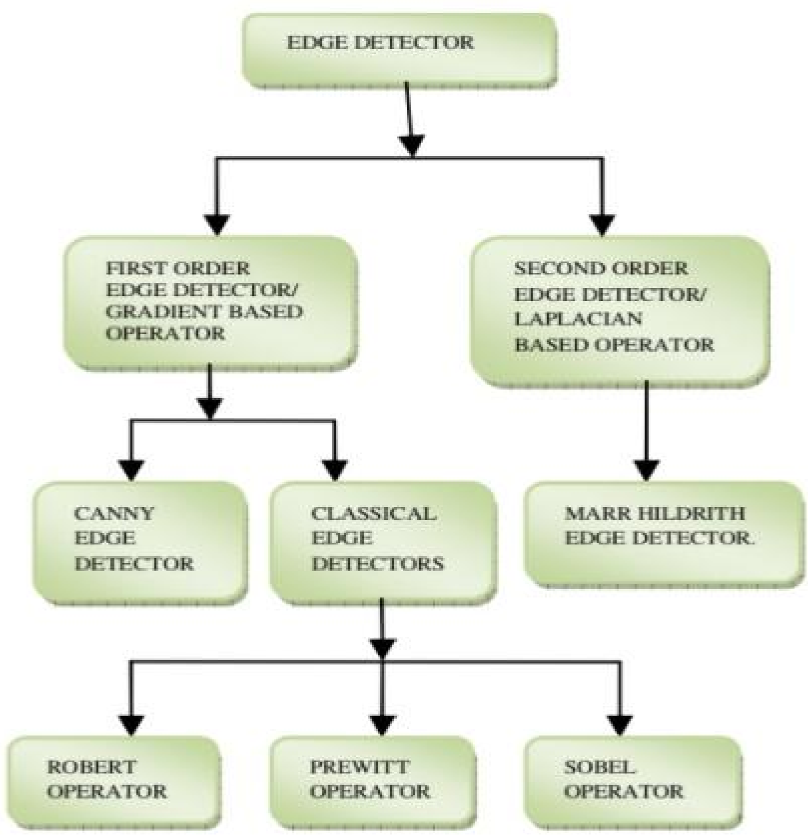
CHAPTER 3

BASIC EDGE DETECTION TECHNIQUES

INTRODUCTION:

Edge detection is a basic tool used in image processing, basically for feature detection and extraction, which aims to identify points in a digital image where brightness of image changes sharply and find discontinuities. The purpose of edge detection is significantly reducing the amount of data in an image and preserves the structural properties for further image processing. In a grey level image the edge is a local feature that, within a neighborhood, separates regions in each of which the gray level is more or less uniform within different values on the two sides of the edge. For a noisy image it is difficult to detect edges as both edge and noise contain high frequency contents which results in blurred and distorted results.

Edge detection makes use of differential operators to detect changes in the gradients of the grey levels. It is divided into two main categories:



3.1 FIRST ORDER EDGE DETECTION:

It is based on the use of a first order derivative, or can say gradient based. If I (i , j) be the input image, then image gradient is given by following formula

$$\nabla I(i,j) = \hat{\imath}\,\frac{\partial I(i,j)}{\partial i} + \hat{\jmath}\,\frac{\partial I(i,j)}{\partial j}$$

The gradient magnitude can be computed by the formula:

$$|G| = \sqrt{\left(\frac{\partial I}{\partial i}\right)^2 + \left(\frac{\partial I}{\partial j}\right)^2}$$

$$\text{OR} \quad |G| = \sqrt{{G_i}^2 + {G_j}^2}$$

Where: G_i is the gradient in the i direction.
G_j is the gradient in the j direction.
The gradient magnitude can be computed by the formula:

$$\theta = arctan\left(G_j / G_i\right)$$

The magnitude of gradient computed above gives edge strength and the gradient direction is always perpendicular to the direction of edge.

3.1.1 CANNY EDGE DETECTION:

The Canny edge detector[1] is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. Canny proposed an approach to edge detection that is optimal for step edges corrupted by white noise.
Canny edge detection can be explained in four steps:

- Noise Reduction
- Gradient Calculation
- Non-Maximum Suppression
- Edge Thresholding and Hysteresis
- Double threshold

Canny edge detectors have advanced algorithms derived from the previous work of Marr and Hildreth. It is an optimal edge detection technique that provides good detection, clear response and good localization. It is widely used in current image processing techniques with further improvements.

Canny edge detection algorithm :

STEP I: Noise reduction by smoothing
Noise contained in image is smoothed by convolving the input image I (i, j) with Gaussian filter G. Mathematically, the smooth resultant image is given by

$$F(i,j) = G * I(i,j)$$

Prewitt operators are simpler to operate as compared to sobel operators but more sensitive to noise in comparison with sobel operators.

STEP II: Finding gradients In this step we detect the edges where the change in grayscale intensity is maximum. Required areas are determined with the help of gradients of images. Sobel operator is used to determine the gradient at each pixel of a smoothened image. Sobel operators in i and j directions are given as

$$D_i = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad \text{And} \quad D_j = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

These sobel masks are convolved with smoothed image and giving gradients in i and j directions.

$$G_i = D_i * F(i,j) \quad \text{And} \quad G_j = D_j * F(i,j)$$

Therefore edge strength or magnitude of gradient of a pixel is given by

$$G = \sqrt{{G_i}^2 + {G_j}^2}$$

G^j And G^i are the gradients in the i-direction and j-directions respectively.

STEP III: Non maximum suppressions: Non maximum suppression is carried out to preserve all local maxima in the gradient image, and deleting everything else results in thin edges. For a pixel M (i, j):

- Firstly round the gradient direction nearest 45° ,then compare the gradient magnitude of the pixels in positive and negative gradient directions i.e. If the gradient direction is east then compare with the gradient of the pixels in east and west directions E (i, j) and W (i, j) respectively.
- If the edge strength of pixel M (i, j) is larger than that of E (i, j) and W (i, j), then preserve the value of gradient and mark M (i, j) as edge pixel, if not then suppress or remove.

STEP IV: Hysteresis thresholding: The output of non-maxima suppression still contains the local maxima created by noise. Instead choosing a single threshold, for avoiding the problem of streaking two thresholds 565 and 789 are used. For a pixel M (i, j) having gradient magnitude G following conditions exists to detect pixel as edge:

- If G < 789 then discard the edge.
- If G > than 565 keep the edge.
- If 789 < G < and565 and any of its neighbors in a 3 ×3 region around it have gradient magnitudes greater than565, keep the edge.
- If none of pixel (x, y)'s neighbors have high gradient magnitudes but at least one falls between 789 and, 565 search the 5 × 5 region to see if any of these pixels have a magnitude greater than thigh. If so, keep the edge.
- Else, discard the edge.

3.1.2 CLASSICAL EDGE DETECTION ALGORITHMS:

In general, classical edge detection algorithms rely mostly on the computation of image gradients i.e. identifying locations in the image for dark-to-light (or light-to-dark) intensity transitions. Hence, the worst case time complexity for most of them is O(2n) O (2 n).

3.1.2.1 SOBEL OPERATOR:

Sobel[27] operator is a discrete differentiation operator used to compute an approximation of the gradient of image intensity function for edge detection. At each pixel of an image, sobel operator gives either the corresponding gradient vector or normal to the vector. It convolves the input image with kernel and computes the gradient magnitude and direction. It uses following 3x3 two kernels:

-1	0	1
-2	0	2
-1	0	1

X-direction

-1	-2	1
0	0	0
1	2	1

Y-direction

As having larger mask, errors due to effects of noise are reduced by local averaging within the neighborhood of the mask.

$G_r(f(i, j)) = (f(i-1, j-1)) + 2(f(i-1, j)) + (f(i-1, j+1))$
$\quad - (f(i+1, j-1)) - 2(f(i+1, j)) - (f(i+1, j+1))$
$G_c(f(i, j)) = (- f(i-1, j-1) - 2(f(i, j-1)) - f(i+1, j-1))$
$\quad + (f(i-1, j+1) + 2(f(i ,j+1) + (f(i+1, j+1))$

Where,

- G_r represents gradient over the rows (i.e Y-Direction Kernel)
- G_c represents gradient over the column (i.e X-Direction Kernel)

Overall Gradient Magnitude is $G = \sqrt{} (G_r^2 + G$

3.1.2.2 PREWITT OPERATOR:

The Prewitt operator[30] is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function.

The function of Prewitt edge detector is almost same as of sobel detector but have different kernels:

1	1	1
0	0	0
-1	-1	-1

Y-direction

-1	0	1
-1	0	1
-1	0	1

X-direction

$G_r(f(i, j)) = (f(i-1, j-1)) + 2(f(i-1, j)) + (f(i-1, j+1))$
$\quad - (f(i+1, j-1)) - 2(f(i+1, j)) - (f(i+1, j+1))$
$G_c(f(i, j))=(- f(i-1, j-1) - (f(i, j-1)) - f(i+1, j-1))$
$\quad + (f(i-1, j+1) + (f(i ,j+1) + (f(i+1, j+1))$

Where,

- G_r represents gradient over the rows (i.e Y-Direction Kernel)
- G_c represents gradient over the column (i.e X-Direction Kernel)

3.1.2.3 ROBERTS OPERATOR:

It is a gradient based operator. It firstly computes the sum of the squares of the difference between diagonally adjacent pixels through discrete differentiation and then calculates the approximate gradient of the image. The input image is convolved with the default kernels of the operator and gradient magnitude and directions are computed. It uses following 2 x2 two kernels:

$$D_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{And} \quad D_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

The plus factor of this operator is its simplicity but having a small kernel it is highly sensitive to noise not and not much compatible with today's technology.

3.2 SECOND ORDER DERIVATIVE (LAPLACIAN FILTER) :

The Laplacian is a 2-D measure of the 2 nd derivative of an image. The Laplacian of an image highlights regions of rapid intensity change and is therefore often used for edge detection zero crossing edge detectors). The Laplacian is often applied to an image that has first been smoothed with something approximating a Gaussian smoothing filter in order to reduce its sensitivity to noise.The operator normally takes a single gray level image as input and produces another binary image as output. The zero crossing detector looks for places in the Laplacian of an image where the value of the Laplacian passes through zero i.e. points where the Laplacian changes sign. Such points often occur at edges in images i.e. points where the intensity of the image changes rapidly, but they also occur at places that are not as easy to associate with edges. It is best to think of the zero crossing detector as some sort of feature detector rather than as a specific edge detector.

3.2.1 METHODOLOGY

The derivative operator Laplacian for an Image is defined as shown in equations below

$$\Delta^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \qquad \ldots\ldots\ldots(6)$$

For X-direction, $$\frac{\partial^2 f}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2f(x,y) \qquad \ldots\ldots\ldots(7)$$

For Y-direction, $$\frac{\partial^2 f}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2f(x,y) \qquad \ldots\ldots\ldots(8)$$

By substituting, Equations (7) and (8) in (6), we obtain the equation (9)

$$\Delta^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y) \qquad \ldots(9)$$

3.3 SUMMARY

Edge detection is a technique of image processing used to identify points in a digital image with discontinuities, simply to say, sharp changes in the image brightness. These points where the image brightness varies sharply are called the edges (or boundaries) of the image.

Sobel Edge Detection: This uses a filter that gives more emphasis to the centre of the filter. It is one of the most commonly used edge detectors and helps reduce noise and provides differentiating, giving edge response simultaneously.

Laplacian detection performs second-order derivatives and hence are sensitive to noise. To avoid this sensitivity to noise.

Prewitt operator works like as first order derivate and calculates the difference of pixel intensities in a edge region. As the center column is of zero so it does not include the original values of an image but rather it calculates the difference of right and left pixel values around that edge. This increase the edge intensity and it become enhanced comparatively to the original image.

CHAPTER 4

A ROBUST EDGE DETECTION APPROACH IN THE PRESENCE OF HIGH IMPULSE NOISE INTENSITY THROUGH SWITCHING ADAPTIVE MEDIAN AND FIXED WEIGHTED MEAN FILTERING

This study introduces a robust edge detection method that relies on an integrated process for denoising images in the presence of high impulse noise. This process is shown to be resilient to impulse (or salt and pepper) noise even under high intensity levels. The proposed switching adaptive median and fixed weighted mean filter (SAMFWMF) is shown to yield optimal edge detection and edge detail preservation, an outcome we validate through high correlation, structural similarity index and peak signal to noise ratio measures. The non maximum suppression method is used to track the edges and overcome edge discontinuities and noisy pixels, especially in the presence of high-intensity noise levels.

4.1 INTRODUCTION

With the proposed method, boundary edges of filtered images are assumed to have high correlation with the original images; as such edges should track the true routes even under high-intensity impulse noise. Most of the current leading filters ensure a good performance on impulse noise reduction, but they still do not perform well on boundaries, especially in the presence of impulse noise with high-intensity levels.

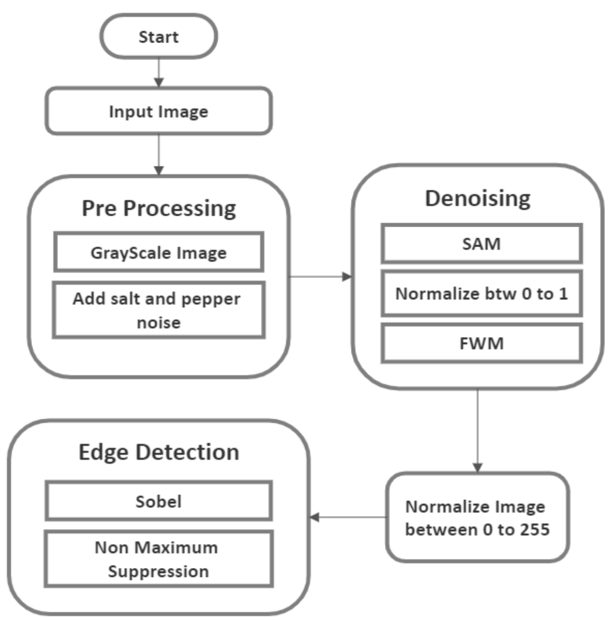


Fig 4.1.1. Block diagram of proposed Approach

4.2 METHODOLOGY

DENOISING METHODS

4.2.1 SWITCHING ADAPTIVE MEDIAN (SAM)

The procedural steps of this method embed the two main components of switching adaptive median (SAM) filtering and fixed weighted mean (FWM) filtering with additional shrinkage window to make up the proposed denoising method we refer to as SAMFWM.

Assume the noise model as given in (1), assuming normalization :

$$c = \begin{cases} 0 & Probability\ P_p \\ 1 & Probability\ P_s \\ 0 < c < 1 & Probability\ 1 - P_p - P_s \end{cases} \quad (1)$$

In this model, as expressed in [44], denotes the uncorrupted pixels, and where the corrupted pixels are assigned probability Ps for salt and Pp for pepper. In this normalized representation of the image, 0 being the minimum intensity denoted by I_{min} and 1 being the maximum intensity denoted by I_{max}

Within an initial sliding window, all pixels with 0 and 1 values are removed, and the median value of the remaining pixels with probability of $1\text{-}p_p\text{-}p_s$ as in (1) will replace the pixel being processed. If all of them are 0s, 1s or a combination of them, or if the variance of the pixels is much higher than the median value, then the size of the window is increased by 1 and the process is repeated until the window size reaches the predefined maximum window size. We assume the difference between pixel values is high when the variance is much higher than the median value ($\sigma > 2$Median), which could be an indication that an edge is present in that area. It thus checks the variance in bigger window sizes to validate whether such an edge does indeed exist or not. If there is an edge, the assumption is that the median value can detect it, otherwise the median value will be correlated to the texture found within the window.

Increasing the window size from its original 3x3 size is warranted only if the SAM step did not yield optimal results. Another adaptive additional window is set to overcome any remaining noise in white and black regions. By doing so, we avoided blurring the final SAMFWM image by increasing the size of the window in the SAM component.

ALGORITHM FOR SAM FILTER :

---

for $I_{i,j} \in$ image do

W_{min} = Minimum window size (3*3)

W_{max} = Maximum window size (5*5)

n = zeros(size(I))

for $I_{x,y} \in W$, $W_{min} \leq W \leq W_{max}$ do

S_{min} = Sort(W)

M = Median(S_{min})

L = Length(S_{min})

V = Variance(S_{min})

if $L \neq 0$ and $V > 2*M$ then

W = W_{max}

S_{max} = Sort(W)

M = Median(S_{max})

n(i,j) = M

I = n

STEPS :

1. Initialize the window to 3*3.
2. Sort the current window of the image and calculate the value of median of that window
3. Also, Calculate the Length(L) and Variance(V) of the window.
4. If the Length(L) is not equal to zero and Variance(V) Less than 2*Median then
    4.1. The central pixel value is replaced with the Median of that window .
5. Else if Length(L) is not equal to zero and Variance(V) greater than 2*Median then
    5.1. Window size is increased to 5*5.
    5.2. Repeat steps 2,3.
    5.3. The central pixel value is replaced with the Median of that window.
6. The resulting image is the resulting Switching Adaptive Median(SAM) Image.

4.2.2 FIXED WEIGHTED MEAN (FWM)

For the fixed mean filtered image, use a 2×2 window in a convolution manner, and check if the pixel being processed I(i,j) in that window (I(i, j), I(i, j+1), I(i+1, j), I(i+1, j+1)) is found corrupted (i.e I(i, j) = 0 or 1 (Normalized Value))

Using the weights selected on the basis of the two conditions described next, if salt or pepper (probability p_s or p_p) is detected, the new processed pixel would be assigned the new value as in (2). Otherwise, it leaves the pixels unchanged.

$$M_{new}(i,j) = \frac{\sum_{(x,y)\in S_{new(i,j)}} \omega_{x,y}\, I_{x,y}}{N-1} \qquad (2)$$

In this equation, N is 4, $S_{new(i,\ j)}$ = { I(i, j+1), I(i+1, j), I(i+1, j+1) }, with indices (i,j) indicating the positions of the corrupted pixels, and (x,y) are the coordinates of the pixels around it. In this proposed method, when the detected corrupted pixel occurs as salt or pepper (with probabilities p_s or p_p), the weights $w_{x,y}$ are directly selected based on the probability of occurrence 1 or 0 for neighboring pixels, according to one of these conditions:

- Assume the corrupted pixel has occurred (with the assumption that the window contains only 0 and 1). Then the weight is set to $w_{x,y}$ = 2 for the east and south pixels and $w_{x,y}$ = 1 for the southeast pixel
- If all of the neighboring pixels are equal then the weight is set to $w_{x,y}$ = 1 for all pixels

STEPS :

1. Initialize a 2*2 window for convolution of the image
2. Calculate Mean and Weighted Mean according to the above conditions
3. Calculate sum of the window
4. If the number of 1's are greater than number of 0's then
    4.1. $I_{x,y}$ is replaced with mean of the window.
  Else
    4.2. $I_{x,y}$ is replaced with weighted mean of the image.
5. If the sum is greater than or equal to 3 then the pixel value remains unchanged.

ALGORITHM FOR FIXED WEIGHTED MEAN (FWM)

---

for $I_{i,j} \in$ image do

W_{2*2} = 2*2 Window Size

S = Current Window

mean = $(\Sigma_{(x,y) \text{ in } S} I_{x,y})/3$

Wmean = $(\Sigma_{(x,y) \text{ in } S} I_{x,y} w_{x,y})/3$

sum = mean*3

if $I_{i,j}$ = 1 or 0 then

    if $I_{x,y}$ = 1 then

        count1++

    else

        count0++

    if count1 > count0 then

        $I_{i,j}$ = mean

    else

        $I_{i,j}$ = Wmean

    if sum $\geq$ **3**

        $I_{i,j}$ = mean

    else

        $I_{i,j}$ = Wmean

else

    $I_{i,j}$ = $I_{i,j}$

Where,

- I(i, j) refers to the image
- Mean refers to current mean of the window
- Wmean refers to weighted mean of the window
- S refers to current window of the image

EDGE DETECTION METHODS

4.2.3 SOBEL OPERATOR

After denoising the impulsive noise image using the Switching Adaptive Median and Fixed Weighted Mean Filter, the Image will clear of salt and pepper noise .The edge detection is performed on the filtered image using the Sobel operator by calculating the gradient over the x direction and y direction.

Sobel gradient operator is the gradient of a pixel is a weighted sum of pixels in the 3-by-3 neighborhood. For gradients in the vertical (y) direction and horizontal (x) the weights are :

-1	0	1
-2	0	2
-1	0	1

X-direction

-1	-2	1
0	0	0
1	2	1

Y-direction

$$G_r(f (i, j)) = (f (i-1, j-1)) + 2(f (i-1, j)) + (f (i-1, j+1)) - (f (i+1, j-1)) - 2(f (i+1, j)) - (f (i+1, j+1))$$

$$G_c(f (i, j))=(- f (i-1, j-1) - 2(f (i, j-1)) - f (i+1, j-1)) + (f (i-1, j+1) + 2(f (i ,j+1) + (f (i+1, j+1))$$

Where,

- G_r represents gradient over the rows (i.e Y-Direction Kernel)
- G_c represents gradient over the column (i.e X-Direction Kernel)

Overall Gradient Magnitude is $G = \sqrt{(G_r^2 + G_c^2)}$

4.2.4 NON MAXIMUM SUPPRESSION

This technique is used for edge thinning in the grayscale image. Edge strength is compared with the neighboring pixels according to gradient direction.

The whole process can be summarized as follows:

- Calculate the vertical and horizontal gradient.
    - The vertical and horizontal gradients are given by
      $G_r(f(i, j)) = (f (i-1, j-1)) + 2(f (i-1, j)) + (f (i-1, j+1))$
      $- (f (i+1, j-1)) - 2(f (i+1, j)) - (f (i+1, j+1))$
      $G_c(f(i,j)) = (- f (i-1, j-1) - 2(f (i, j-1)) - f (i+1, j-1))$
      $+ (f (i-1, j+1) + 2(f (i ,j+1) + (f (i+1, j+1))$
- Calculate the angle of the gradient
    - Angle = $\tan^{-1}(G_r / G_c)$

ALGORITHM FOR NON MAXIMUM SUPPRESSION

1. If the angle of gradient is 0 degrees, the gradient magnitude is checked in the east and west directions, and if it is more than the magnitude of pixels in these directions, it is considered on the edge
2. If the angle of gradient is 45 degrees, the gradient magnitude is checked in the northeast and southwest directions, and if it is more than magnitude of pixels in these directions, it is considered on the edge
3. If the angle of gradient is 90 degrees, the gradient magnitude is checked in the north and south directions and if it is more than the magnitude of pixels in these directions, it is considered on the edge
4. If the angle of gradient is 135 degrees, the gradient magnitude is checked in the northwest and southeast directions and if it is more than the magnitude of pixels in these directions, it is considered on the edge

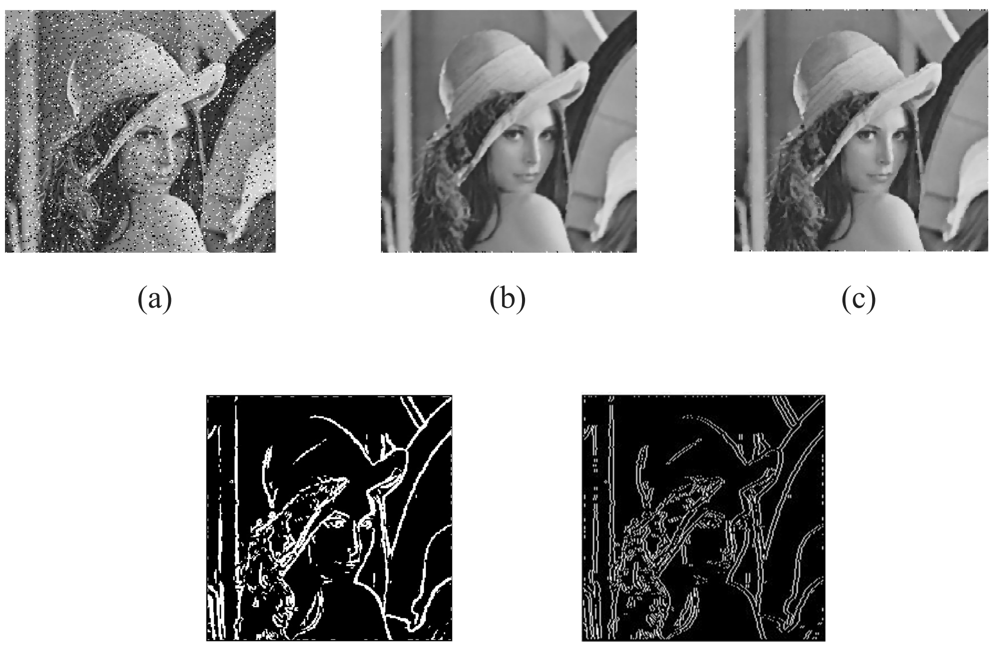
4.3 RESULTS

After applying the Switching Adaptive median and fixed weighted mean filter on tha highly impulsive noise image the noise is decreased a lot.

The SAM filter is applied on the whole image whereas the FWM filter is applied on the pixels that are let corrupted after applying the SAM Filter.

This usage of Combination of both filters gives the optimal Noise removed Image.

The whole output is shown below :



(a) (b) (c)

(d) (e)

Fig 4.3.1 : (a) Impulsive noise Image (b) SAM Image (c) FWM Image (d) Sobel Image (e) Non Maximum Suppression Image

The resulting Edge detected Image preserve edges better than the existing algorithms like Canny Edge Detection. This combination of SAM and FWM filters is shown to yield the best (i.e., highest) structural metrics than any other well-known denoising filter in the presence of different impulse noise intensities.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 EXPERIMENT SET

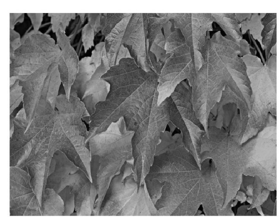
		
(a) Lena	(b) Camera Man	(c) Coins
		
(d) Flower	(e) Apple	(f) Dog
		
(g) Rose	(h) Bird	(i) Leaves

Table 5.1.1 Experiment Set

The Experiment Set consists of 9 images which were used for denoising and edge detection

5.2 PERFORMANCE METRICS

Three performance metric measures are used to compare the proposed method and various techniques, They are

1. MSE(Mean Square Error)
2. SSIM(Structural Similarity Index)
3. PSNR(Peak Signal-To-Noise Ratio)

1.MEAN SQUARE ERROR (MSE)

MSE value denotes the average difference of the pixels all over the image. A higher value of MSE designates a greater difference amid the original image and processed image. Nonetheless, it is indispensable to be extremely careful with the edges.

Formula for calculating MSE is as follows:

$$MSE = \frac{1}{mn}\sum_{0}^{m-1}\sum_{0}^{n-1}\|f(i,j) - g(i,j)\|^2$$

where

f represents the matrix data of our original image

g represents the matrix data of our degraded image in question

m is the numbers of rows of pixels of the images and i represents the index of that row

n is the number of columns of pixels of the image and j represents the index of column

2.STRUCTURAL SIMILARITY INDEX (SSIM)

An image quality metric that assesses the visual impact of three characteristics:

1.Luminance

2.contrast

3.structure

Numeric scalar with a single SSIM measurement.

Numeric array the same size as the input images. Each spatial element in the input image has an SSIM measurement along any channel or batch dimension.

The formula for calculating the SSIM[26] is as follows:

$$\mathrm{SSIM}(\mathbf{x},\mathbf{y}) = [l(\mathbf{x},\mathbf{y})]^{\alpha} \cdot [c(\mathbf{x},\mathbf{y})]^{\beta} \cdot [s(\mathbf{x},\mathbf{y})]^{\gamma}$$

Where,

l is Luminance comparison function

c is Contrast comparison function

s is Structure comparison function

3.PEAK SIGNAL-TO-NOISE RATIO (PSNR)

The term peak signal-to-noise ratio (PSNR) is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation. Because many signals have a very wide dynamic range, (ratio between the largest and smallest possible values of a changeable quantity) the PSNR is usually expressed in terms of the logarithmic decibel scale.

$$PSNR = 20\log_{10}\left(\frac{MAX_f}{\sqrt{MSE}}\right)$$

Where, MAX_f is the maximum possible pixel value of the image

MSE is the mean square error.

5.3 RESULTS

It is always preferable to have subjective and objective assessment to conclude whether the proposed method achieves the best results over the existing ones or not. In subjective assessment image quality is measured by the subjective evaluations of human observers. The evaluation may bek made using an absolute rating scale or by means of side-by-side comparison, with output images of different schemes. The objective assessment can be made by using various metrics. PSNR, MSE, SSIM are used for the purpose of objective analysis. Higher PSNR, higher SSIM indicates better image quality and lower MSE indicates batter image quality.

5.3.1 SUBJECTIVE ANALYSIS

Comparison of denoising Filters :

The proposed SAMFWM Filter images are brighter and sharper and have less noise than the Median filter and Mean Filter. The median filtered and mean filtered images are slightly on the side of smoothing the image while the proposed method kept details and the overall look is better than the median and mean filtered images.

The Figure 5.3.1.1 Shows the denoised images obtained by applying general Median filter and proposed SAMFWM method directly on a Lena image corrupted with 10% impulse noise. The image obtained by the proposed method is better in handling noise keeping the details intact.

The figure 5.3.1.3 , 5.3.1.6 shows the denoised images obtained by applying general Median filter and proposed SAMFWM method directly on a coin and Dog image corrupted with 10% impulse noise. The image obtained by applying the proposed method is brighter, sharper and has high contrast ,maintaining a more optimal image. The same for over mean filter.

All the figures of 5.3.1 shows the results of the denooised images on applying median filter and proposed SAMFWM method on 10% impulse noise.
The corrupted pixel undergoes through two processes SAM and FWM to correct the pixel thereby obtaining the optimal overall result.
The proposed method can even do better than the standard mean and median filter even in noisy situations.

(a) (b) (c)

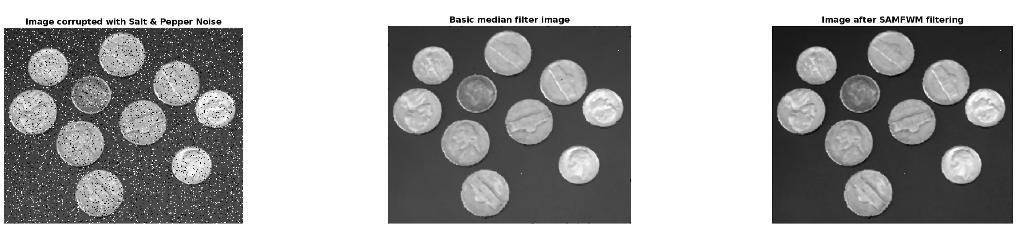
Fig 5.3.1.1 (a) Lena impulse noise image, (b) Lena Median Filter

(c) Proposed SAMFWM method

(d) (e) (f)

Fig 5.3.1.2 (d) CameraMan impulse noise image, (e) Median Filter

(f) Proposed SAMFWM method



(g) (h) (i)

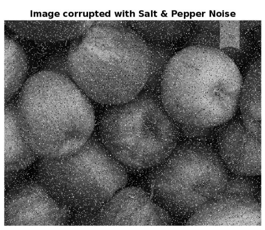
Fig 5.3.1.3 (g) Coins impulse noise image, (h) Median Filter

(i) Proposed SAMFWM method

(j) (k) (l)

Fig 5.3.1.4 (j) Flowers impulse noise image, (k) Median Filter

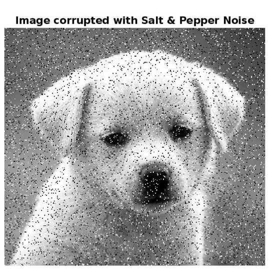
(l) Proposed SAMFWM method

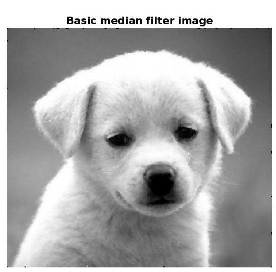


(m) (n) (o)

Fig 5.3.1.4 (m) Apple impulse noise image, (n) Median Filter

(o) Proposed SAMFWM method





(p) (q) (r)

Fig 5.3.1.6 (p) Dog impulse noise image, (q) Median Filter

(r) Proposed SAMFWM method

(s) (t) (u)

Fig 5.3.1.7 (s) Rose impulse noise image, (t) Median Filter

(u) Proposed SAMFWM method

(v) (w) (x)

Fig 5.3.1.8 (v) Bird impulse noise image, (w) Median Filter

(x) Proposed SAMFWM method

Comparison of Edge Detection :

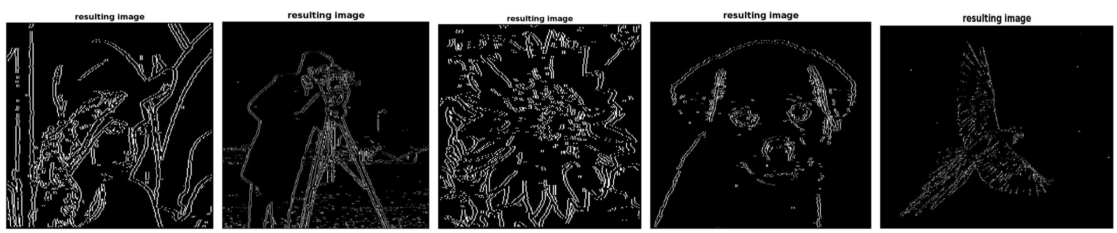
The Figure 5.3.1.9 Shows the edge images obtained by applying Proposed edge detection, sobel edge detection, canny edge detector directly on different images corrupted with 10% impulse noise. It is also observed that efficient edges are extracted by Proposed method only in all the regions of an image and works well for images corrupted by impulse noise.

The dog image of the three methods can be observed in figure 5.3.1.9. Canny edge detector failed to produce all the edges and even the edge connectivity is not good. The edges obtained by sobel are very thick. The proposed method keeps in check of those two disadvantages and maintaines good edge connectivity and thin edges over the image.

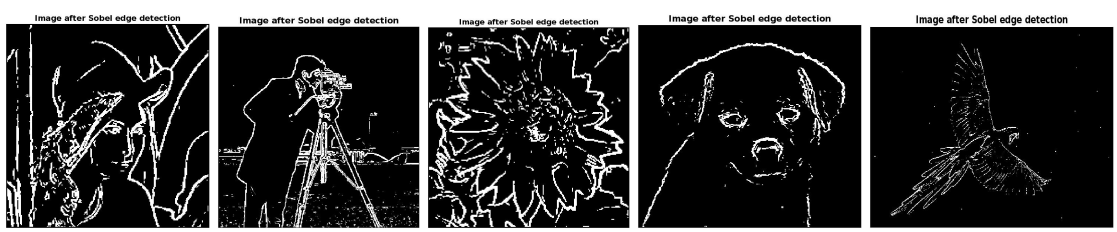
The same can be observed on the Lena, camera man image, flower, bird images.In all these different types of images proposed method aintains good edge connectivity and thinner edges.

Comparison between Sobel vs Canny vs Proposed algorithm

(a)



(b)



(c)

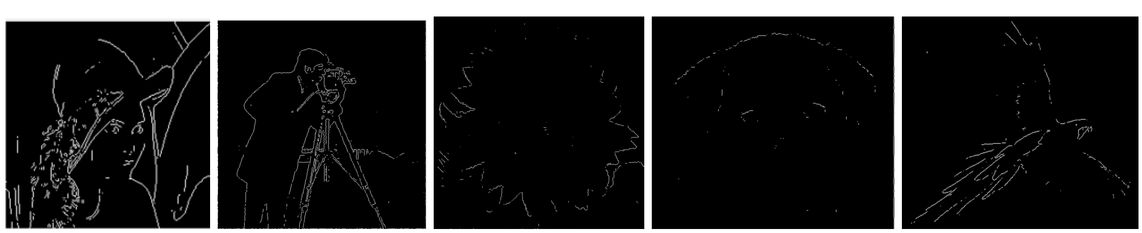


Fig 5.3.1.9 (a) Proposed Edge Detection (b) Sobel Edge Detection (c) Canny Edge Detection

It is clear that the proposed edge detection method is better than the sobel and canny edge detection Sobel obtained thick edges while canny lost the edges.The proposed method managed to thin edges while having the clear edges.

5.3.2 OBJECTIVE ANALYSIS

The Objective assessment is to compare the denoising methods and edge detectors applied on these denoised images. The Results obtained by applying median filter, mean filter, SAMFWM on several are presented in the tables I, II, III.

The metrics used to compare the three denoising filters are peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), mean square error (MSE).
The higher the value of PSNR and SSIM, lower the value of MSE higher is the image quality.

	Median Filter			Mean Filter			SAMFWM		
	Lena	Camera man	Coins	Lena	Camera man	Coins	Lena	Camera man	Coins
PSNR	-22.6093	-19.3298	-24.8395	-22.7016	-19.4319	-24.8415	-22.2171	-18.9183	-24.7506
MSE	175.6018	85.6996	304.7562	177.5012	89.4581	305.4332	163.0297	77.9519	298.5771
SSIM	0.7742	0.7583	0.5930	0.7698	0.7509	0.5904	0.7803	0.7648	0.5749

Table I: PSNR, MSE, SSIM results of Lena, Camera Man, Coins for Median Filter, Mean Filter, SAMFWM (proposed method)

	Median Filter			Mean Filter			SAMFWM		
	Flower	Apple	Dog	Flower	Apple	Dog	Flower	Apple	Dog
PSNR	-23.4927	-28.0134	-15.0488	-23..5004	-28.0111	-15.0443	-23.1065	-27.9472	-15.0384
MSE	223.4973	632.9079	31.9801	225.3428	630.1965	31.9876	204.4807	623.3279	31.9039
SSIM	0.7820	0.6855	0.9526	0.7698	0.6812	0.9501	0.7930	0.6882	0.9475

Table II: PSNR, MSE, SSIM results of Flower, Apple, Dog for Median Filter,
Mean Filter, SAMFWM (proposed method

	Median Filter			Mean Filter			SAMFWM		
	Rose	Bird	Leaves	Rose	Bird	Leaves	Rose	Bird	Leaves
PSNR	-19.0107	-13.3256	-23.4514	-18.6439	-13.2994	-23.4592	-18.1439	-13.2194	-23.2984
MSE	79.6280	21.5062	221.3834	71.6783	21.1891	221.9834	65.2220	20.9865	213.7191
SSIM	0.8824	0.8908	0.6940	0.8891	0.8919	0.6929	0.8793	0.8936	0.6955

Table III: PSNR, MSE, SSIM results of Rose, Bird, Leaves for Median Filter,
Mean Filter, SAMFWM (proposed method)

The tables I, II, III present the values of PSNR, MSE, SSIM of different images. Higher value of PSNR indicates better image with lower noise. The proposed method got better PSNR results than median and mean filters with the lower presence of noise and better image quality.

Lower the MSE value better is the image. In comparison of all the images MSE is lower for the proposed method than the median and mean filters which is an indication of better image quality.

Higher the SSIM the better the image is. SSIM gives the structural similarity index between the original image and the denoised image which shows the closeness to the original image. For every image SSIM is better for the proposed SAMFWM than the median and mean filter.

CHAPTER 6

CONCLUSION

A Robust switching adaptive median (SAM) filtering shrinkage window denoising method has been proposed. The filtered images are assumed to have high correlation with the original images as such edges should track the true routes even under high-intensity impulse noise. To measure the degree of edge preserving and image structural metrics, standard measures are computed in order to compare the performance of different filters including the proposed method to gauge the quality of image after the smoothing process is performed. Experiments shows that PSNR, SSIM, MSE is high in the proposed algorithm compared to existing. It identifies exactly the location of an edge pixel from the noisy image efficiently by considering more number of neighbors in different scale. The results obtained proved that the proposed method yielded a better performance after edge detection even in the presence of high intensity impulse noise

REFERENCES

[1] J. Canny, “A Computational Approach to Edge Detection,” IEEE Trans. Pattern Anal. Mach. Intell, vol.PAMI-8, no.6, Nov. 1986.

[2] S. Xie, Z. Tu, “Holistically-nested edge detection,” In Proc. IEEE Int. Conf. Comput. Vision, pp. 1395–1403 (2015).

[3] M. E. Yukse, “Edge detection in noisy images by neuro-fuzzy processing,” Elsevier, Int. J. Electron. Commun. vol.61 pp.82 – 89, 2007.

[4] Y. Shi, Q. Zhao, F. Guo, Y. Zhu, “A Fast Edge Detection Model In Presence of Impulse Noise,” in Proc. Springer 8th Int. Conf. Image Graph., Tianjin, China, vol.9217, pp. 64-74, Aug. 2015

[5] P. Jain, V. Tyagi, “A survey of edge-preserving image denoising methods,” Springer, Inform. Syst. Front. , vol.18, no.1, pp. 159–170, Feb. 2016.

[6] C. Pal, Amlan Chakrabarti, Ranjan Ghosh, “A Brief Survey of Recent Edge-Preserving Smoothing Algorithms on Digital Images,” Elsevier, Procedia Comput. Sci., pp.1-40, Mar. 2015.

[7] R. C. Gonzalez, R.E. Woods, “Digital image processing,” 3th ed., Upper Saddle River, NJ, Prentice-Hall, 2006.

[8] L. I. Rudin, S. Osher, E. Fatemi, “Nonlinear total variation based noise removal algorithms,” Elsevier, Physica D J., vol.60, no.1-4, pp.259–268, 1992.

[9] A. Chambolle, “An Algorithm for Total Variation Minimization and Applications,” J. Math. imaging Vis., vol. 20, no.1, pp.89-97, Jan. 2004.

[10] P. Perona, J. Malik, “Scale-Space and Edge Detection Using Anisotropic Diffusion,” IEEE Trans. Pattern Anal. Mach. Intell., vol.12, no.7, .Jul. 1990.

[11] G. Gerig, O. Kubler, R. Kikinis, F. A. Jolesz, “Nonlinear Anisotropic Filtering of MRI Data,” IEEE Trans. Med. Imag., vol.1, no.2, Jun. 1992.

[12] C. Tomasi, R. Manduchi, “Bilateral Filtering for Gray and Color Images”, in Proc. IEEE, 6th Int. Conf. comput. Vis., Bombay, India, Jan. 1998.

[13] K. He, J. Sun, X. Tang, “Guided Image Filtering,” IEEE Trans. Pattern Anal. Mach. Intell., vol.35, June. 2013.

[14] A. Buades, Bartomeu Coll, Jean-Michel Morel, “A non-local algorithm for image denoising,” in Proc. IEEE Computer society Conf. Comput. Vis. Pattern Recog., vol.2, pp.60- 65, Jun. 2005.

[15] R. Yang, L. Yin, M. Gabbouj, J. Astola, T. Neuvo, “Optimal weightedmmedian filtering under structural constraints” IEEE Trans. Signal Process., vol.43, no.3, pp. 591–604, 1995.

[16] G. Qiu, An improved recursive median filtering scheme for image processing”, IEEE Trans. Image Process., vol.5, no.4, pp. 646–648, 1996.
[17] S. Vishaga, S. L. Das, “A Survey on Switching Median Filters for Impulse Noise Removal,” in Proc. IEEE Int. Conf. Circuit, Power Comput. Tech., Mar. 2015.
[18] S. J. Ko and Y. H. Lee, “Center Weighted Median Filter,” IEEE Trans. Circuits Systs., vol. 3, no.9, pp 984-993, Sep. 1991.
[19] Y. Dong, S. Xu, “A New Directional Weighted Median Filter for Removal of Random-Valued Impulse Noise,” IEEE Signal Process. Lett., vol.
14, no. 3, Mar. 2007. [20] V. V. Khryaschev, A. L. Priorov, I. V. Apalkov, P. S. Zvonarev, “Impulse Denoising Using Adaptive Switching Median Filter,” in Proc. IEEE 2 th Int. Conf. Image process., Sep. 2005.
[21] S. Esakkirajan, T. Veerakumar, A. N. Subramanyam, C. H PremChand, “Removal of High Dendity Salt and Pepper noise Through Modified Decision based Unsymmetrical Trimmed Median Filter,” IEEE Signal Process. Lett., vol.18, no.5, May 2011.
[22] H. L. Eng, K. K. Ma, “Noise adaptive soft-switching median filter,” IEEE Trans. Image Process., vol.10, no.2, pp. 242–251, 2001.
[23] R. Yang, L. Yin, M. Gabbouj, J. Astola, T. Neuvo, “A New impulse detector for switching median filters,” IEEE Signal Process. Lett., vol.9, no.11, , pp. 360–363, 2002.
[24] P.-E. Ng, K.-K. Ma, “A Switching Median Filter With Boundary Discriminative Noise Detection for Extremely Corrupted Images,” IEEE Trans. Image Process., vol. 15, no. 6, pp.1506-1516, Jun. 2006.
[25] I. F. Jafar, R. A. AlNa’mneh, K. A. Darabkh, “Efficient Improvements on the BDND Filtering Algorithm for the Removal of High-Density Impulse Noise,” IEEE Trans. Image Process., Vol. 22, No. 3, Mar. 2013.
[26] Z. Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, Eero P. Simoncelli, “Image Quality Assessment: From Error Visibility to Structural Similarity,” IEEE Trans. Image Process., vol. 13, no. 4, Apr. 2004.
[27] A Descriptive Algorithm for Sobel Image Edge Detection, O. R. Vincent and O. Folorunso, Proceedings of Informing Science & IT Education Conference (InSITE), (2009).

[28] Median filtering frameworks for reducing impulse noise from grayscale digital images: a literature survey, Teoh and Ibrahim, S.H. Teoh, H. Ibrahim, 2012 Int. J. Future Comp. Commun., 1 (4) (Dec. 2012), pp. 323-326

[29] Nandhini and Nalini, 2014, M. Nandhini, T. Nalini, "Survey of image denoising algorithms" Int. J. Adv. Res. Comp. Sci., 5 (3) (March 2014), pp. 179-182

[30] S. Kaur and I. Singh, "Comparison between Edge Detection Techniques", International Journal of Computer Applications, vol. 145, no.15, (2016), pp. 15-18.

[31] J. K. Mandal and Somnath Mukhopadhyay "A Novel Technique for Removal of Random Valued Impulse Noise Using All Neighbor Directional Weighted Pixels (ANDWP)" Part of the Communications in Computer and Information Science book series (CCIS,volume 203).

[32].Gonzalez, R.C., Woods, R.E (2008). Digital Image Processing. Prentice Hall, Englewood Cliffs, NJ .

[33]. Pratt W.K (1978). Digital Image Processing, Wiley, NewYork.

[34]. R. H. Chan, C.W. Ho, and M. Nikolova (2005). "Salt-and-pepper noise removal by median-type noisedetectors and detail-preserving regularization". IEEE

[35]. P.-E. Ng and K.-K.Ma, (2006). "A switching medianfilter with boundary discriminative noise detection forextremely corrupted images," IEEE Trans. Image Process.,

Vol. 15, No. 6, pp. 1506–1516.[36]. Robert M. Haralickand Linda G Shapiro: Computer and Robot Vision,(1992). Addison-wesley publishing company.Inc, NewYork .

[37]. Huang, J.S., Tseng, D.H., (1988). "Statistical theory of edge detection". Journal of Computer Vision Graphics Image Processing, pp:337–346.

[38]. Olivier laligant and Frederic (2010). "A Nonlinear Derivative scheme applied to edge detection". IEEE Transactions on pattern analysis and machine intelligence, Vol. 32, No.2. pp. 242-257.

[39]. Dr. A. Srikrishna, Dr. B. Eswara Reddy and M.Pompapathi. "Non-Linear Noise Suppression Edge Detection Scheme for Noisy Images". Proceedings of the IEEE international conference on recent advances and innovations in engineering (ICRAIE - 2014), Organized by Poornima University, Jaipur. ISBN: 978-1-4799-4041- 7,DOI: Publisher: IEEE, pp:1-6.

[40]. Dong Hoon Lim., (2004). " Robust edge detection in noisy images". Journal of Computational Statistics & Data Analysis 50, pp. 803 – 812.