A project report on

FUZZY BASED GAUSSIAN NOISE REDUCTION IN COLOR IMAGE

Submitted in the partial fulfillment of the requirements for the award of Degree of

Master of Technology

In

Software Engineering

By

Deepankar Yadav

University Roll No. 2K15/SWE/08

Under the Guidance of

Dr. O. P. Verma

(Professor & Former H.O.D., Department of Computer Science and Engineering, DTU)

2015-2017

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

DELHI-110042, INDIA

CERTIFICATE

This is to certify that **Deepankar Yadav (2K15/SWE/08)** has carried out the major project entitled "**Fuzzy Based Gaussian Noise Reduction in Color Image**" in partial fulfillment of the requirements for the award of Master of Technology Degree in Software Engineering during session 2015-2017 at Delhi Technical University.

The major project bonafide piece of work carried out and completed under my supervision and guidance. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University/Institute for the award of any degree or diploma.

Dr. O. P. Verma
Professor & Former H.O.D.
Department of Computer Science and Engineering
Delhi Technological University
Delhi - 110042

DECLARATION

I hereby declare that the major project entitled "**Fuzzy Based Gaussian Noise Reduction in Color Image**" which is being submitted to Delhi Technological University, Delhi in partial fulfillment of requirements for the award of degree of Master of Technology (Software Engineering) is an authentic work carried out by me. The material contained in the report has not been submitted to any university or institution for the award of any degree.

Deepankar Yadav

2K15/SWE/08

ACKNOWLEDGEMENT

I would like to take this opportunity to express my appreciation and gratitude to all those who have helped me directly or indirectly towards the successful completion of this work.

I would like to express my sincere gratitude to my guide **Dr. O. P. Verma**, Professor and Former H.O.D., Department of Computer Science and Engineering, Delhi Technological University, Delhi whose benevolent guidance, encouragement, constant support and valuable inputs were always there for me throughout the course of my work. Without his continuous support and interest, this thesis would not have been the same as presented here.

I thank God for making all this possible, my parent and friends for their constant support and encouragement throughout the project work.

Deepankar Yadav
2K15/SWE/08

ABSTRACT

An elementary problem of image restoration is to reduce the noise from an image as much as possible. The noise reduction procedures depends upon the type of noise degrading the image. In this work effort are made to reduce Gaussian noise. To do same efforts are made to develop fuzzy based filters for reduction of Gaussian noise in colour images. In the proposed method first we determine the level of Gaussian noise in the local window with the help of the standard deviation of local window in comparison to standard deviation of all local windows contained within image and mean of local window and its nearby windows. Then we compute adaptive distances between colour component pairs of the central pixel and the neighbourhood pixel and fuzzify these adaptive distances. These fuzzified adaptive distances used to determine the weights of everey color component of each neighbourhood pixels contained within local window. The weighted average of color components weights of all the neighbourhood pixels helps in computing the correction term for the Gaussian filter. The results on several color images prove the efficacy of the proposed fuzzy filters.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Overview

Generally degradation in an image take place because of several kinds of noises. Noise in an image may contaminate in the course of image acquisition, image transmission, image storage and retrieval. In the image processing, noise is interpreted as displacement of signal intensities from their actual values. An elementary problem of image restoration is to reduce the noise from an image as much as possible in the same time preserving image's sharp details constituting of edges, corners etc. Also denoising an Image is an essential pre-processing task before performing other image processing task like feature extraction, image segmentation, texture analysis etc. The noise reduction procedures depends upon the type of noise degrading the image. The most usually occurring noises generally classified as: Additive noises like Gaussian Noise and Impulse Noise and multiplicative noises like speckle noise. Reduction of noise in an image is a very significant issue. In this thesis efforts are made to reduce Gaussian noise. For this purpose efforts are made to develop fuzzy based filters for reduction of Gaussian noise in colour images. Fuzzy based filters are more advantageous over classical filters even as they preserve the image structure. Also, fuzzy filters offers ease in implementation because of simple fuzzy rules that illustrate a specific noise.

1.2 The Problem Statement

The main objective of image restoration is to successfully reduce the Noise in an Image in the same time sustaining its fundamental framework which includes edges, corners and other fine details to a possible extent. In the present work a fuzzy based filter to reduce Gaussian Noise in color image has been proposed.

1.3 Literature Review

Degradation of image features take place because of several kind noise at the time of image acquisition, image transmission and image storing. The most usually occurring noises generally classified as: Additive noises like Gaussian Noise and Impulse Noise and multiplicative noises like speckle noise. Noises could also be categorized widely in three category substitutive noise, additive noise and multiplicative noise. Substitutive noise also called as impulsive noise such as salt and pepper noise. Additive Gaussian noise is an example of additive noise. Additive Gaussian Noise follows Gaussian probability distribution and additive in nature that is why it called as Additive White Gaussian Noise. Researchers have been suggested numerous linear and nonlinear filtering procedures for the purpose of image denoising. To discard Gaussian Kind noises generally linear filters are much effective but generally these filters have a trend to blur the image's edges. While nonlinear filters are well suitable in the reduction of substitutive noises. In the last few years numerous nonlinear filters had been constructed using classical as well as fuzzy methods. In the last few years many such methods have been appeared. Modern advancement in fuzzy theory permits diverse prospects for evolving novel image restoration approaches for example on the basis of classical median filter the fuzzy median filter [24-25] has been adapted. Also with the help of modern fuzzy logic, the Fuzzy Inference Rules by Else action filters [5, 14, 15] have been developed. These filters belongs to non-linear procedures that to eliminate noise from images by implementing fuzzy rules**.** A multi pass fuzzy filter containing of three cascaded blocks presented by Russo in [32]. Every block is attached to a fuzzy procedure that efforts to abandon the noise while conserving the image organisation. Jiu [21] presented a fuzzy multilevel median filter which is an expression of the multilevel median filter in the fuzzy domain. It comprises fuzzy rules for the elimination of such a noise. The histogram adaptive filter by Wang and Chu [40] fits to a category of filters that uses the histogram for decreasing noise. Androutsos et al. [3] developed a different category of filters termed Fuzzy vector rank filters built on a grouping of changed distance measures, fuzzy membership values and α -trimmed functions. A multi channel filtering method by merging fuzzy rational and median functions which reserves the edges and structure of the image presented by Khrjii and Gabbouj [27].

A new concept uses alpha trimmed mean with the likeness of pixels for the revealing of such a noise offered by Wenbin [41]. In [8] rank order mean is adapted for formulating a Least Mean Square Scheme to clean out the static as well as nonlinear noise. Also a different procedure is offered in [31] for the recognition of such a noise with the help of enough alteration among the noisy pixels and the noiseless pixels. Exertions being prepared in [23, 30] for the purpose of diminishing blurring generated because of linear filtering. A new approach uses signal adaptive median filtering algorithm is recommended in [16] for the exclusion of noise. This algorithm is based on the concept of similarity level is proposed for pixel values constructed using their global and local arithmetical properties also to signify the correlations between a pixel and its neighbours, and to decide the upper and lower bounds of the similarity level, co-occurrence matrices being used. For the purpose of noise reduction as well as structure conservation use of nonlinear filters being recommended in [9-11]. To preserve finer details in images the median based filters are being adapted in [39]. An iterative edge conserving filtering procedure with the use of the blur metric presented by Mansoor et al. [38] in which first noisy pixels have been categorized into edge and non-edge pixels and different filtering methods are used. A fuzzy two step procedure to reduce noise in colour image offered by Stefan *et al*. [34] which employs the fuzzy gradient values and fuzzy processing for the recognition of corrupted pixels. Mansoor *et al*. [37] proposed a recursive procedure. This procedure approximate the noise level that is needed to compute the filter parameters.

For the reduction of Gaussian noise the median filter is less effective. This downside is managed to some extend by using a different nonlinear filter method that is filter with the help of moving average filters. To substitute concerned pixel value with the mean value of its pre decided neighbourhood pixels, a standard moving average filter is used. One of the most important concern in the elimination of the Gaussian noise is, how to distinguish among sudden changes in pixel values because of noise or because of edges. Numerous efforts had been proposed earlier to resolve this issue. For this issue in [6], fuzzy derivatives had been employed. The GOA [2] proposed a new filter which have been aimed for dropping noise which are of Gaussian kind by approximating a fuzzy gradient in every direction so as to determine whether the local deviation is because of noise or because of finer details of image. Stefan *et al*. [35] concentrated on the fuzzy distance among colour pairs of central pixel and the neighbourhood pixel as a weight for pixels within window to accomplish the weighted average of windows pixels to filter out the Gaussian noise in colour images. Russo [12] suggests a technique to eliminate Gaussian noise that combines a non-linear procedure to preserve fine details by only smoothing of pixels not present on edges and a method to tune parameter alteration with the help of noise approximation. For the elimination of the Gaussian noise present in the images a filter has been is formulated in [13] which is based on multi constraint piecewise linear (PWL) functions. Perceptual classification rules has been used by Xiaofen and Qigang [42] to isolated noise from actual finer details of image. With the help of the the Gaussian filter Shin *et al*. [7] developed a block based noise approximation algorithm. In this algorithm a function of standard deviation of Gaussian noise in the given colour images used to tune coefficients of the Gaussian filter. Although Bilateral filtering perform smoothening the colour image but still try to preserve edge with the help of a nonlinear grouping of nearby pixel values [4]. Choi and Krishnapuram [43] proposed a fuzzy filter scheme to handle Gaussian noises. A specific scheme chooses on the basis of compatibility function at the concerned pixel. This compatibility function is assembled into low, medium and high membership functions and these are multiplied with the three filter outputs of [43] to obtained the collective output to deal with gaussian noise in Hanmandlu *et al*. [28]. With the use of the ideas of both [43] and [28], three sigma and Pi filters are designed in [19].

Another exciting effort [20], in which fuzzy based smoothing of images for Gaussian noise is accomplished by coalescing the results of some filters called as hybrid filters. Most of these approaches discussed here generally work on gray scale images. However, it is feasible to use these approaches to reduce Gaussian noise in colour image. In colour images each colour component i.e. R, G, and B can be treated as different gray scale image and hence could be input to a filter individually and later these filtered components represent colour image together. Except a few, maximum papers on noise elimination have worked on the each component of RGB colour space individually. When dealing with Gaussian noise, it is not easy to differentiate among noise and edges. In colour image treating each component a separate gray image may produce artifacts in the filtered image. A small number of research papers explore the relations among the colour components like Shulte et al. [35] inspected the fuzzy distances among the colour components to explore the conceivable relations among the colour components. This relations among the colour components calculate the correction term by means of fuzzy distances among the colour components. The result of this collaboration among the values of each colour component in the course of filtering should not disturb the colour structure of the image. The second sub filter offered in [35] uses the average of differences among the R, G, B component of the pixel of interest and its neighbouring pixels, which alters the actual colour of the concerned pixel.

1.4 Thesis Layout

Content in this thesis is structured as: general concepts of image processing have been discussed in chapter 2 along with information about Gaussian noise and most commonly used image metrics. In chapter 3 information about filtering in spatial domain along with several filters to reduce Gaussian noise has been provided. In chapter 4 details of the proposed Fuzzy based Filter have been discussed. In chapter 5 results of proposed filter and comparative analysis of its performance has been discuss. Chapter 6 is the last chapter, it conclude the thesis and present future scope of the proposed approach.

CHAPTER 2

GENERAL CONCEPTS

2.1 Basics of Digital Image Processing

A digital image in *spatial* domain may be defined as a 2-D function $f(p, q)$.

$$I = f(p, q) \tag{2.1}$$

Here p and q represent *spatial* coordinates. Value of function for a pair of spatial coordinates (p, q) is termed as intensity I or gray scale level or value for that pair of spatial coordinates (p, q). Also when gray scale has limited levels and spatial domain has finite number of co-ordinates then the image pixel's intensity represented using this gray scale called digital image. In the digital image acquisition process first convert an optical image into a electrical signal which is continuous then sampled and quantized to form a digital image.

Digital image processing can be divided into many sub divisions on the basis of procedures in which :

• Input is an image and outputs also an image

• Inputs are images and outputs are features take out from input images.

On the basis of this division there exist many image processing functions. Below is the list of such a function those are based on the above two classes.

a. Image Acquisition

b. Image Enhancement

c. Image Restoration

d. Transform-domain Processing

e. Image Compression

f. Morphological Image Processing

g. Image segmentation

h. Image Representation and Description

i. Object Recognition

Image Acquisition, Image Enhancement, Image Restoration, Image Compression, Transform-Domain Processing are those image processing functions in which input is an image as well as output is an image. Morphological Image Processing, Image Segmentation, Image Representation and Description, Object Recognition are those image processing functions in which inputs are images and outputs are features take out from input images.

The functions of Image processing can be implemented in two ways, these function may be carried out in frequency domain or in spatial domain. In present thesis image processing is carried out in Spatial Domain specially. This choice may be effected by type of application and which image processing function are carried out. For example on the basis of type of application an efficient intensity transformation of values of the pixels in an image carried out, some examples of such a transformations are Discrete Fourier transform also called DFT, Discrete cosine transform also called DCT, Discrete Hartley transform also called DHT, Discrete wavelet transform also called DWT etc.. These transformation may be carried out for the purpose of Image processing either in the spatial domain or in frequency domain. Also on the basis of type of application Discrete Fourier transformation, discrete cosine transformation, Discrete Hartley transform, Discrete wavelet transformation according to the need of image processing applications.

In the Image Enhancement process we perform such image processing operation which results are more relevant than actual image or captured image for a particular application. Here the word particular is very significant because it convey that process of image enhancement is highly application oriented. Also one more thing is that image enhancement is a highly subjective area. Usually enhancement procedures are employed to highlight specific properties of the images. For example:

a. Contrast Enhancement

b. Altering the level of brightness of an image.

Unlike Image enhancement Image restoration is a highly objective area of image processing while image enhancement is a subjective area. Also Image restoration is one of the leading area of image processing. For the purpose of image restoration we develop mathematical and statistical models of image degradation and then develop their inverse process that is why the techniques of image restoration are highly depends upon mathematical and statistical models of noises which degrade Image. Because of this reduction of noise and undo blurring are major tasks of image restoration process. The objective of image restoration is to reduce the noises in the images such that the difference between actual Image and degraded image become minimum as much as possible that is why the procedures carried out in image restoration are concerned with mathematical modeling of noise and degradations such that an inverse process could be develop to restore the image in actual image.

On the basis of uses many sorts of imaging systems exists. X-ray, Gamma ray, ultraviolet, and ultrasonic are such imaging systems which applied in biomedical devices. The field of space science employs ultraviolet, Infrared & Radio Imaging Systems are used. In the field of Geological Exploration Sonic imaging is used. Radar applications uses microwave imaging. The field of visible light imaging systems which one of the most important imaging system and used in the areas of remote sensing, microscopy, measurements, consumer electronics, and entertainment electronics.

Images which are attained by optical, electro-optical or electronic means may corrupted by detecting atmosphere. This degradation may be in the form of blurring of image, noisy image. Noise in the images are likely to be occur because of sensor behavior. Blurring in the images are likely to be occur because of camera misfocus, relative motion between object and image capturing device, arbitrary environmental instability etc. Also presence of noise in an image could be because of a noisy medium of transmission using which the image had been transmitted. Occurrence of noise might also be because of electronic noise like in process of storing and retrieve in a storage system.

Presence of noise in an image is not a strange problem, it is very commonly occurring phenomena. Corruption of an image can be take place at the time of image acquisition or at the time of image transmission or at the time of image storage or at the time of image retrieval. Noises could be categorized widely in three category substitutive noise, additive noise and multiplicative noise. Substitutive noise also called as impulsive noise such as salt and pepper noise. Additive Gaussian noise is an example of additive noise. Additive Gaussian Noise follows Gaussian probability distribution and additive in nature that is why it called as Additive White Gaussian Noise.

In this thesis our main concern is to suppress additive white Gaussian nose. Additive white Gaussian noise distributes all over the image and corrupts almost all the pixels within image hence it is not easy to quash such a noise completely. In this thesis efforts are made to reduce Gaussian noise in color images. The basic assumption followed in a most of these researches is that the samples of the Gaussian noise are mutually uncorrelated and follow a Gaussian distribution.

In the process of image denoising the information about the type of noise corrupting the image has a significant character. Spatial representation of additive and multiplicative noise can be done as below. Suppose noise free image is $f(x,y)$, noise represented as $\eta(x,y)$ and noisy image represented as $w(x,y)$ where (x,y) represents the pixel location. Noisy Image because of additive noise is represented as

$$w(x,y) = f(x,y) + \eta(x,y) \tag{2.2}$$

In the same way, Noisy Image because of multiplicative noise is represented as

$$w(x,y) = f(x,y) * \eta(x,y) \tag{2.3}$$

These equations represents operations which performed at pixel level in spatial domain.

2.2 Gaussian Noise

Gaussian noise is uniformly distributed over the signal [Um98]. It implies that every pixel in the noisy image is suffered by Gaussian noise. The noise in a pixel is the sum of the actual value of pixel and a random Gaussian distributed noise value. The name Gaussian noise itself specifies that Gaussian noise follows Gaussian distribution. Gaussian distribution represents graphically a bell shaped probability distribution function. Gaussian noise adds a random Gaussian distributed noise value to the actual pixel value. Gaussian distribution's bell shaped probability distribution function mathematically represented as,

$$F(g) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(g-m)^2/2\sigma^2} \tag{2.4}$$

Where g stands for the gray level, m represents the mean of the distribution function. Standard deviation of the noise represented by σ . Graphical representation of the above function shown as.

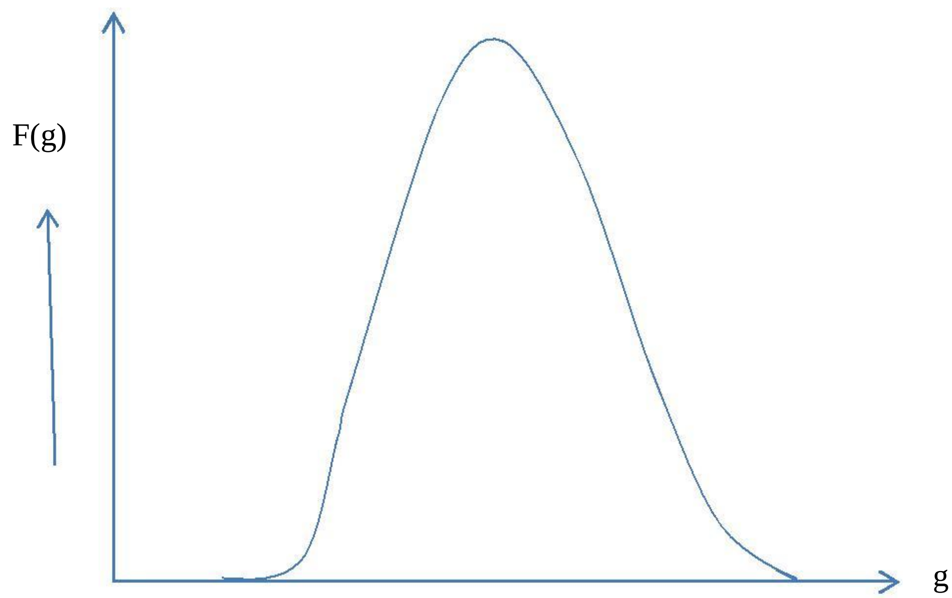

Figure 2.1 Gaussian Noise Distribution

When we introduced Gaussian noise having standard deviation equal to 20 into an image sea life, it would look like as shown in figure 2.2 which has shown below, the noise free image in which noise has been added has been shown in figure 2.3.

Figure 2.2 Fish Image with Gaussian Noise

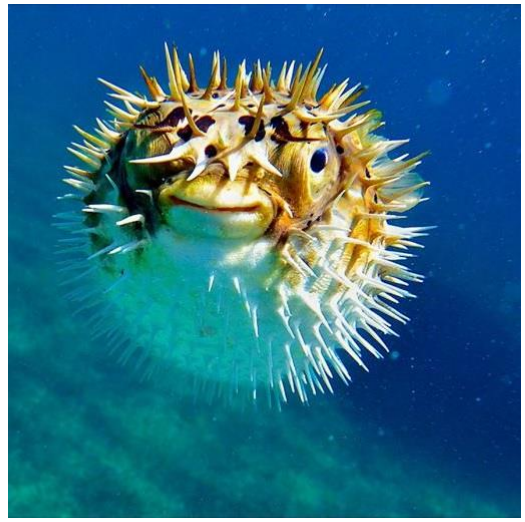

Figure 2.3 Fish image without Noise

We know Gaussian noise follow Gaussian distribution i.e. Normal Distribution. As per according to the normal distribution majority of the pixels suffer from less Gaussian noise level and minority of pixels suffer from high level of Gaussian noise. For Example suppose a vector has 100 elements and if we introduce Gaussian Noise having standard deviation equal to 1 and mean zero. Then elements of the vector will be random numbers from a Gaussian distribution. Gaussian distribution over these 100 elements has been shown in figure 2.4 with the help of plot of histogram of this vector.

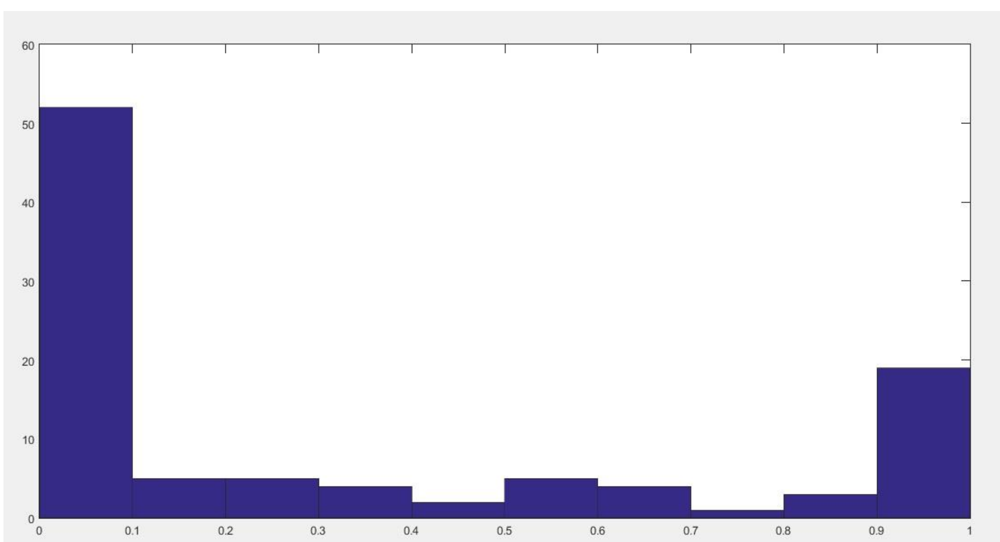

Figure 2.4 Histogram of Vector r

In the figure we can see as the standard deviation is 1 but still there are more than 50 fifty elements have value in between 0 and 0.1 also less than 20 elements have value in between 0.9 and 1. Hence more no. of pixel have less amount of Gaussian noise and there are very less number of pixels which have very high amount of noise. Mathematically we can show it as follows

$$P(\mu - \sigma \leq x \leq \mu + \sigma) = 0.68 \tag{2.5}$$

$$P(\mu - 2\sigma \leq x \leq \mu + 2\sigma) = 0.95 \tag{2.6}$$

$$P(\mu - 3\sigma \leq x \leq \mu + 3\sigma) = 1 \tag{2.7}$$

Where $\mu\ and\ \sigma$ stands for mean and standard deviation respectively. From equation 2.2.1 we can see value of the random variable x lies between $\mu - \sigma\ and\ \mu + \sigma$ for near about 68 percent of time. Only 27 percent of time value of goes beyond $\mu - \sigma\ upto\ \mu - 2\sigma$ and $\mu + \sigma\ and\ \mu + 2\sigma$ and only 5 percent of time value goes beyond it.

With the help of software Gaussian noise could be easily introduce to an input image. The basic assumption followed in a most of these researches is that the samples of the Gaussian noise are mutually uncorrelated and follow a Gaussian distribution (Normal Distribution).

2.3 Image Metrics

An image can be evaluate by both objective evaluation and by subjective evaluation. A human expert can observed the image for the purpose of subjective evaluation. But because of the complication of the human visual system (HVS) exact quality of image can not be determined.

Hence for the purpose of a better quality evaluation various matrices are used specially for objective evaluation. Out of these some matrices for objective evaluation are mean square error, mean absolute error, root mean squared error and peak signal to noise ratio. These are matrices which are discussed here.

For example: let the noise less original image is $O(p,q)$, image with noise is $D(p,q)$ and restored image is $R(p,q)$ where p and q stands for spatial coordinates of the image in digital form.

Let us consider an image containing $M \times N$ pixels in discrete spatial coordinates that are $p = 1,2,3,\dots,M$ and $q = 1,2,3,\dots,N$. Now we will look for Mean Square Error below

2.3.1 Mean Square Error(MSE)

Mean Square Error also represented as MSE is defined as below when there are single colour component i.e. for gray image:

$$MSE = \frac{1}{M \times N} \sum_{c=1}^{M} \sum_{d=1}^{N} \big(R(p,q) - D(p,q)\big)^2 \tag{2.8}$$

For a color image having three color components namely Red, Green and Blue then the Mean Square Error will be as below

$$MSE = \frac{1}{3 \times M \times N} \sum_{r=1}^{3} \sum_{p=1}^{M} \sum_{q=1}^{N} \big(R(p,q,r) - D(p,q,r)\big)^2 \tag{2.9}$$

Root Mean Squared Error also denoted as RMSE is square root of mean square error MSE is defined as

$$RMSE = \sqrt{MSE} \tag{2.10}$$

Mean Absolute Error also represented as MAE, for an gray image is defined as

$$MAE = \frac{1}{3 \times M \times N} \sum_{r=1}^{3} \sum_{p=1}^{M} \sum_{q=1}^{N} |R(p,q,r) - D(p,q,r)| \tag{2.11}$$

For a color image having three color components namely Red, Green and Blue then the Mean Square Error will be as below.

2.3.2 Peak signal to noise ratio (PSNR)

Peak signal to noise ratio also denoted as PSNR is another highly significant objective image quality metric. Peak signal to noise ratio is measured in logarithmic scale in dB. PSNR is measured as a ratio which is ratio of peak signal power to noise power. PSNR can be represented in MSE terms because the MSE also represent noise and the peak signal power, so the PSNR in terms of MSE is defined as:

$$PSNR = 10 * \log_{10} \frac{1}{MSE} \tag{2.12}$$

These image metrics are used for assessment of the quality of a restored image and the competence and effectiveness of a filtering procedure.

2.3.3 Execution Time

Execution Time (T_E) is measurement of efficiency of a filtering methods which are applied to restore the image. Execution Time is represented as the time consumed by a Processor for the execution of filtering algorithm with a constraint that there should not be other program in running stage besides the operating system.

Execution Time be determined by in essence by the fact that how fast is CPU clock, besides this, there are also many other factors on them Execution time relies like memory size, the input data size, the memory access time and the algorithm logic itself.

CHAPTER 3

FILTERING IN THE PRESENCE OF GAUSSIAN NOISE

3.1 Fundamentals of filtering

For the purpose of image denoising Filters are used. In the process of image denoising filters are significantly used or we can say denoising process relies heavily on use of filters. Filtering is a process to modify an image for the purpose of image enhancement and restoration. The fundamental concept which take place for suppressing noise in a noisy image is use of linear filtering, digital convolution and moving window principle. Linear filtering is process which produce output value of a denoised pixel as a linear combination of the neighbouring pixel values of the input pixel. Let suppose $d(x)$ is an input signal needed to be filtered and $r(x)$ is an restored signal. If the used filter fulfils firm conditions like linearity and shift invariance, then the output signal can be stated mathematically in simple form as provided below

$$r(x) = \int d(t)h(x-t)dt \tag{3.1}$$

Where $h(t)$ represents impulse response and it completely determines the filter. The given technique called as convolution and can be stated as $r = d * h$. Also filter can be expressed as below for discrete convolution in spatial domain

$$r(i) = \sum_{-\infty}^{\infty} d(t)h(i-t) \tag{3.2}$$

It states that the output $r(i)$ at a point i is determine by a weighted sum of input pixels which surrounding i and in above equation the weights are determined by $h(t)$. Also to find the output value at the next pixel $i+1$, the function $h(t)$ is shift by one then the weighted sum is calculated again. The complete output is produce by a series of shiftmultiply-sum operations and this formulate a discrete convolution. In the 2-dimensional space $h(t)$ turn out to be $h(t,u)$ and give Equation turn into

$$r(i,j) = \sum_{t=i-k}^{i+k} \sum_{u=j-1}^{j+1} d(t,u)h(i-t,j-u) \tag{3.3}$$

Where the values expressed by $h(t,u)$ are represented the filter weights or filter mask or filter kernel. For the reasons of symmetry $h(t,u)$ is generally chosen to be of size mxn where m and n are both generally odd $(often\ m = n)$. In physical systems, kernel $h(t,u)$ must be non-negative always, which in turn produce some blurring or averaging of the image. If $h(t,u)$ is narrow then the filter produce less blurring. In digital image processing, $h(t,u)$ maybe defined subjectively and that is why $h(t,u)$ gives rise to many types of filters.

3.2 Reduction of Gaussian Noise with Spatial Filtering

As we know Image Restoration is largely a objective process because in restoration we have some priori knowledge of the degradation process. On the basis of this priori knowledge we try out to model the noise filter. Hence in restoration process our main focus on modelling the noise model and apply its inverse process to restore the noise less image.

The restoration process generally involves finding a measurement of goodness. With the help of this measurement of goodness we will try to optimize the result. Here I have discussed some basic spatial filter for reduction of Gaussian noise.

3.2.1 Restoration using Mean Filters

Mean filters usually works on an image through reduction in the deviation with the use intensity difference among neighbourhood pixels in working window. The mean filter calculate the centred pixel value in the concerned window by averaging of all the neighbouring pixel values with that centre value within window. Mean Filter is applied with a convolution mask. It produce output which is an average of sum of the values of all pixels contained within window. Mean filter is a linear filter. Mask of mean filter is a square. Generally a 3×3 square window is used to perform convolution. Also there is one property which should be note that for brightness of the image to be unaffected sum of coefficients of the kernel should be 1. Also if the sum of the coefficients of kernel is 0 then output may be a dark image.

Let us consider a hypothetical $3x3$ mean filter mask as represented in Table 3.1.

h_1	h_2	h_3
h_4	h_5	h_6
h_7	h_8	h_9

Table 3.1 3×3 mask

Let us consider a window having pixel of spatial coordinates $(5,5)$ as centred pixel represented in Table 3.2

w_{44}	w_{45}	w_{46}
w_{54}	w_{55}	w_{56}
w_{64}	w_{65}	w_{66}

Table 3.2 Neighbourhood of $w(5,5)$

Result of convolution using mean filter shown in table 3.1 to the window having pixel with spatial $(5,5)$ as centred is given as

$$h_1 w_{44} + h_2 w_{45} + h_3 w_{46} + h_4 w_{54} + h_5 w_{55} + h_6 w_{56} + h_7 w_{64} + h_8 w_{65} + h_9 w_{66}$$

If in a filter if all weight coefficients are equal then it is termed constant weight filter. Calculation of convolution of an image with the mean kernel referred to as mean filtering

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

Table 3.3 Constant Weight Filter

process. Mean filter generally behave as a low pass filter. This filter may not allow the high frequency constituents present in the processed image. Also it is blurring effect increase when we increase the size of mask like 5×5 or7×7. Also the mean filter discussed here is a simply a arithmetic mean filter but we can also develop its variants like Geometric mean filter, Harmonic mean filter but arithmetic and geometric mean filters are more suited for Gaussian noise.

3.2.2 Order – Statistic Filters

Order – statistics filters are spatial filters which belongs to the class of non-linear filters. These filters output are depends on the pixels ranking present in that image area which are contained by spatial filter. Order statistics filter calculates the value of the central pixel on the basis of the order of the pixels present in filter. One of the well-known example of order statistic filters is median filter. Below I discussed some introductory details about the media filter.

Median Filter

A median filter is a nonlinear filter. We can estimate by its name that Median filter also works on moving window policy same as mean filter. A mask of size $3 \times 3, 5 \times 5$ *or* 7×7 may be usually used for the convolution purpose. In Median filter we compute the median of the pixels present in image area surrounded by filter window. After this we replace the value of central pixel's colour components with the medians of the colour components of the pixels contained in filter kernel for each colour component. Median filter usually produce less blurring than its peer linear filters.

$$r(p,q) = \underset{(i,j)\epsilon h_{p,q}}{median}\{d(i,j)\} \tag{3.4}$$

Median is computed first by sorting the colour components values of the pixels enclosed within filter window either in ascending or in descending order and then middle pixel value is determined as middle pixel value of this.

For example: Let us consider a $3x3$ kernel for the purpose of median filtering. Also consider an image area whose pixel value for a single colour components are given in Table 3.4. We are assuming that highlighted pixel having value 140 is pixel of interest. The mask which have this pixel as central pixel contained nearby pixel with colour component values as 115,150,108,132,107,152,128,134.

125	147	175	111	150
120	115	150	108	118
122	132	140	107	112
112	152	128	134	112
134	155	155	198	145

Table 3.4 Pixel values of Sample Image

Now sort down the pixel values contained in concerned filter window

107,108,115,128,132,134,140,150,152

As mask has 9 values so value at position 5th in sorted sequence will represent the median value of this sequence which is 132 concerned example. Hence the value of concerned pixel which is 140 will be replace by 132.

By median filter working method we can observe that median filter does not create new pixel value as mean filter create it just replace the value from one of the pixel value belong to the window which is median. Hence when it applied on edges, filter does not create new pixel values hence preserve sharpness more than median filter. So we can see Median filter is do less smoothing than mean filter.

Other order statistic filter are max filter, min filter, mid-point filter, alpha trimmed mean filter etc.

3.2.3 Lee Filter

Lee filter [6], created by Jong Sen Lee is an adaptive filter which vary its computation as per as with the local statistical properties of the concerned filter window. Also this filter take care of underlying structure of an image through its adaptive computation by performing smoothing only in the areas which do not have fine details flat and by leaving important fixtures like lines and textures unaffected. Lee filter generally adopt lesser sized masks such as $3 \times 3, 5 \times 5, 7 \times 7$. Within each window, the local mean and variances are estimated. There is a need to estimate local statistical properties which are local mean and variances inside each filter window.

The value central pixel of location (x, y) computed by Lee filter is formulated as:

$$\tilde{f}(x,y) = k(x,y)[g(x,y) - \bar{g}] + \bar{g} \tag{3.8}$$

Where

$$k(x,y) = \begin{cases} 1 - \dfrac{{\sigma_n}^2}{{\sigma_g}^2} & {\sigma_g}^2 > {\sigma_n}^2 \\ 0 & {\sigma_n}^2 > {\sigma_g}^2 \end{cases} \tag{3.9}$$

Lee filter parameters range between zero and one. If parameter is 0 or near to 0, it means area under investigation is flat, means does not have enough fixture and also its opposition stands. The Lee Filter has a distinctive feature which is when the filter window perform filtering process on the section of images having much details like edges, corner etc., then filter favours the pixels of high intensity, those are usually considered as noisy pixels while

Filter window perform filtering process on the section of images having less details than output tends toward the local mean. This phenomena ensure less smoothing in area having important fixture. But we can observe that this method will retain noise in the area having detailed fixtures.

3.2.4 Wiener Filter

Wiener filter is filter which belongs to a family of spatial-domain filter. The concept of Wiener filtering is developed by Norbert Wiener in the year 1942. The Wiener Filter usually used for reduction of additive noise that is why we could use it to perform restoration of an image suffered by additive Gaussian Noise. Norbert Wiener proposed two methods for both frequency domain and for spatial domain which are

(a). Fourier-transform method

(b). Mean-squared method

The Fourier-transform method is used for frequency domain while Mean-squared method used for spatial Domain. The Fourier-transform method perform both denoising and deblurring, while Mean squared method can perform denoising only. When we perform Wiener filtering in frequency domain then it demands information of the noise power spectra and original image in advance while we perform Wiener filtering in spatial domain there is no need such information. Here I would like to discuss some more details about mean square method which is a filter in spatial domain because our work of focus is in spatial domain only. Mean-squared method Wiener filter which is spatial domain filter follows least-squared principle that is the this filter mean-squared error (MSE) between the noise free image and filtered image should be minimized.

Wiener filtering is based on both the global statistics that are mean, variance, etc. of the whole image and local statistics that are mean, variance, etc. of a minor region or sub-part of image because we understand in an image statistical properties of image are not same from a region to another region even inside the same image also both global statistics (mean, variance, etc. of the whole image) as well as local statistics (mean, variance, etc. of a minor region or sub- part of image) are vital. Wiener filtering mathematically represented as

$$r(x,y) = \bar{g} + \frac{{\sigma_d}^2}{{\sigma_d}^2 + {\sigma_\eta}^2}(g(x,y) - \bar{g}) \tag{3.5}$$

Where $\tilde{r}(x,y)$ stands for the restored image, $\bar{g}$ represent the local mean of values of colour components of pixels contained in filter window, ${\sigma_d}^2$ is the local variance of values of colour components of pixels contained in filter window and ${\sigma_\eta}^2$ express variance of noise which degraded image.

For example: Suppose we have a $(2m+1) \times (2n+1)$ filter mask then while we use this mask, local mean $\bar{g}$, of values of colour components of pixels contained in filter mask is computed as

$$\bar{g} = \frac{1}{L} \sum_{s=-m}^{m} \sum_{t=-n}^{n} g(s,t) \tag{3.6}$$

Where L express number of pixels contained by filter mask.

In the same manner Suppose we have a $(2m+1) \times (2n+1)$ filter mask then while we use this mask, local variance ${\sigma_g}^2$, of values of colour components of pixels contained in filter mask is computed as

$${\sigma_g}^2 = \frac{1}{L-1} \sum_{s=-m}^{m} \sum_{t=-n}^{n} (g(s,t) - \bar{g})^2 \tag{3.7}$$

The local variance in noise free image represented as ${\sigma_f}^2$ is computed by subtracting ${\sigma_g}^2$ which is local mean computed in equation 3.6 from priori known noise variance ${\sigma_n}^2$. It relies on the assumption that the noise is independent in spatial space.

Here we can observe that output of the filter will be equal to the local mean if the central pixel value in the filter window is same as local mean else output of the filter is not same as mean. Hence the output of filter change with the local mean also depends on local variance, by this tries to obtain noise free result as much as parameters permits. Wiener filtering tries to estimates the actual image pixel values with least mean-squared error which leads to minimum presence of noise in filtered image.

But the dark side of this filtering process is that it generally not know that what is the variance of the noise so if our estimation for noise variance is not relevant than output may differ greatly from actual.

3.2.5 Bilateral Filter

Bilateral filter [11] belongs to the family of a nonlinear filter. Bilateral filter is presented by Tomasi and Manduchi. This is a type of filter which contributes in the fields of additive noise reduction. Like many other filters Bilateral filter also smooth image sections having less details and also make effort to maintain edges, corners, lines etc. i.e. higher details sections unaffected with the help nonlinear filtering. Bilateral filter is easy to use, no iterative and simple.

Let us see some introductory functioning of Bilateral filter:
The kernel w of Bilateral filter is derived with the help of two sub-kernels

(a). Gray Level Kernel w_g

(b). Distance Kernel w_d

Gray Level Kernel:
To design gray level kernel first we calculate gray level distance between nearby pixel (suppose at location (x_1, y_1) having intensity value $g(x_1, y_1)$) and central pixel (present at location (x, y) having intensity value $g(x, y)$) inside filter window that is :

$$d_g = [|g^2(x_1, y_1) - g^2(x, y)|]^{\frac{1}{2}} \tag{3.10}$$

With the help of this we develop Gray Level Kernal as mathematically described in equation 3.11:

$$w_g = \exp\left(-\frac{1}{2}\left(\frac{d_g}{\sigma_g}\right)^2\right) \tag{3.11}$$

Where σ_g represents distribution function for w_g.

Now we consider Distance kernel:

To design Distance Kernel first we calculate spatial distance between nearby pixel (suppose at location (x_1, y_1) having spatial co-ordinate (x_1, y_1)) and central pixel (present at location (x, y) having spatial co-ordinate (x, y)) inside filter window that is :

$$d_s = \sqrt{(x_1 - x)^2 + (y_1 - y)^2} \tag{3.12}$$

With the help of this we develop Distance Kernel as mathematically described in equation 3.13:

$$w_d = \exp\left(-\frac{1}{2}\left(\frac{d_s}{\sigma_d}\right)^2\right) \tag{3.13}$$

Here σ_d stands for standard deviation of w_d.
Now with the use of Distance Kernel and Grey Level Kernel we can calculate Bilateral Filter as describe in equation 3.14:

$$w_b = w_g w_d \tag{3.14}$$

From Equation 3.14 we have computed Bilateral Filter, to eliminate noise we need to move this Kernel all over the noisy image and have to perform following computation in each filter window.

$$\check{f}(x, y) = \frac{\sum_{s=-a}^{a} \sum_{t=-b}^{b} g(x + s, y + t)}{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w_b(s, t)} \tag{3.13}$$

This filter is frequently used because of its simplicity and easiness.

CHAPTER 4

PROPOSED WORK

4.1 Fuzzy Based Filter to Reduce Gaussian Noise

As we knows Gaussian Noise is an additive noise follows Gaussian distribution or Normal distribution. In a colour image a pixel has three components RGB i.e. Red, Green, Blue respectively. Suppose a pixel in colour space represented by 3 coordinates$(p, q\ , r)$. A noisy pixel is represented as

$$d(p, q, r) = o(p, q, r) + n(p, q, r) \qquad (4.1)$$

Here $d(p, q, r)$ stands for degraded colour image, $o(p, q, r)$ is the original colour image and $n(p, q, r)$ stands for Gaussian noise in colour space.

Methods used to reduce Gaussian Noise Generally compute the weighted average of nearby pixels of the central pixel value. Important Point is to compute the weights of nearby pixel in such a manner as to obtain corrected central pixel value. Many techniques in the literature use same colour component of nearby pixels to calculate the average of all these components for obtain the modified value of central pixel without taking the effect of other colour components of nearby pixels. While considering RGB colours components separately we have also compute an adaptive fuzzy distance between RGB colour pair as red green, red blue and green blue to obtain the weights to the pixels within window. These colour pairs used in computation of weights of pixels in window leads to reduction in MSE and PSNR. Also adaptive fuzzy distance of these colour pair provides similarity between neighbourhood pixel and central pixel. Adaptive fuzzy distance contributes in preserving edges by a way of giving more weight to the pixels which are similar to the central pixel and less weight to the pixels which are not similar to the central pixel in the process of weight computation of neighbourhood pixels.

But this also has a drawback that is if pixel of interest is itself highly noisy and if nearby pixels are not much noisy even then they will be provided the less weights because their colour components will not be similar to the central pixel colour components. This phenomena will generate ensuing artefacts.

To prevent this phenomena to take place we will employee a process before calculating the pairwise fuzzy adaptive distances between neighbouring pixels and pixel of interest. In this process first we calculate the standard deviation of colour components of pixels of the current window. Then for each colour component if the standard deviation is low then this colour components of the pixels are not highly noisy. If it is the case then we compute adaptive fuzzy distance directly using colour components values of the neighbouring pixels otherwise if standard deviation of a window pixels are high then it is the matter of concern. Here we have to worry before calculating the adaptive fuzzy distance between the colour pairs of the concerned window's pixel. In such a case we can not directly find out the adaptive fuzzy distance between the colour pairs of central pixel and neighbouring pixels because it may be either due to highly noisy window or because window contain edge area.

If it is an edge area then it is fine. Go ahead, directly compute the adaptive fuzzy distance between the colour pairs of the concerned window's pixel else it is a highly noisy image area. If concern window standard deviation is high because of a highly noisy area then we should not calculate the adaptive fuzzy distance between the colour pairs of the concerned window's pixel because either central pixel itself may be highly noisy or there are very few or no similar pixel colour component available between central pixel and neighbouring pixels. Hence in such a case we have to improve the values of colour components of pixels of the concern window and then we will be compute the adaptive fuzzy distance between the pixels. If the adaptive fuzzy distance between the central pixel and neighbouring pixels is less the weight provided is high else vice-versa.

Also I have observed that if we calculate the standard deviation of each window of size 3×3 within image then standard deviation for most of the windows is for an image. For example I have taken out a plot of standard deviations of individual window from pixel 6100 to 6600 in an image shown in below figures. In this graph we can observe that standard deviation is low for most of the times and there are only two high spike which shows sudden changes in image pixel values within a window. These spikes stands for the fine details in the image like edges, corner etc. The mean standard deviation for the graph shown in figure 4.1 is 5.7503 and it is for noise less image. In the below graphs X axis represents individual local windows and Y axis represents value of standard deviation of these local windows.

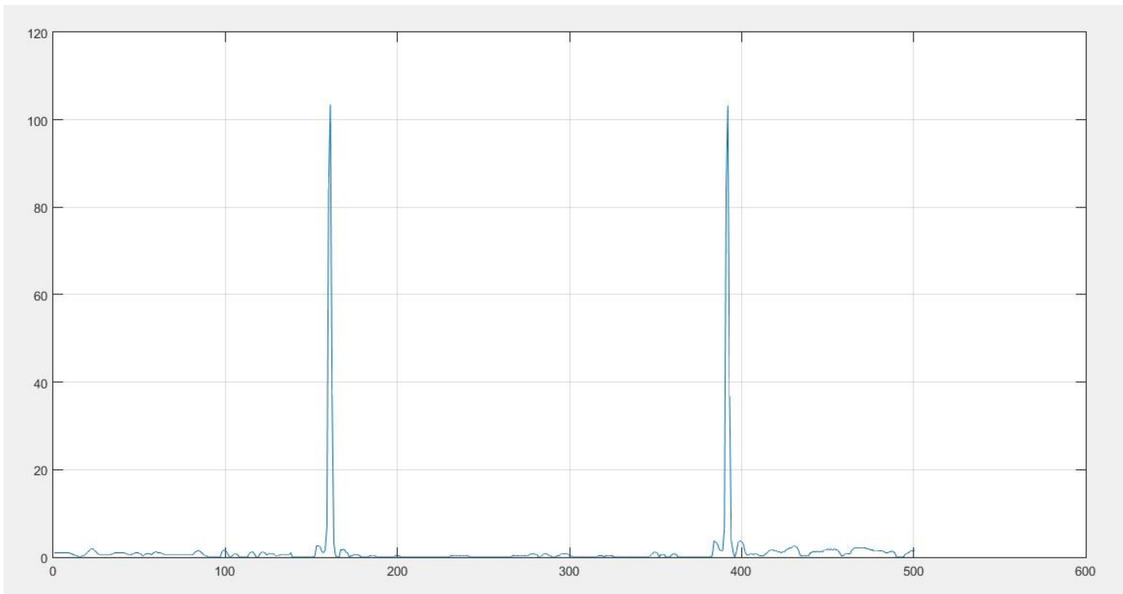

Figure 4.1 Plot for standard deviation of individual local window in a noise free Image

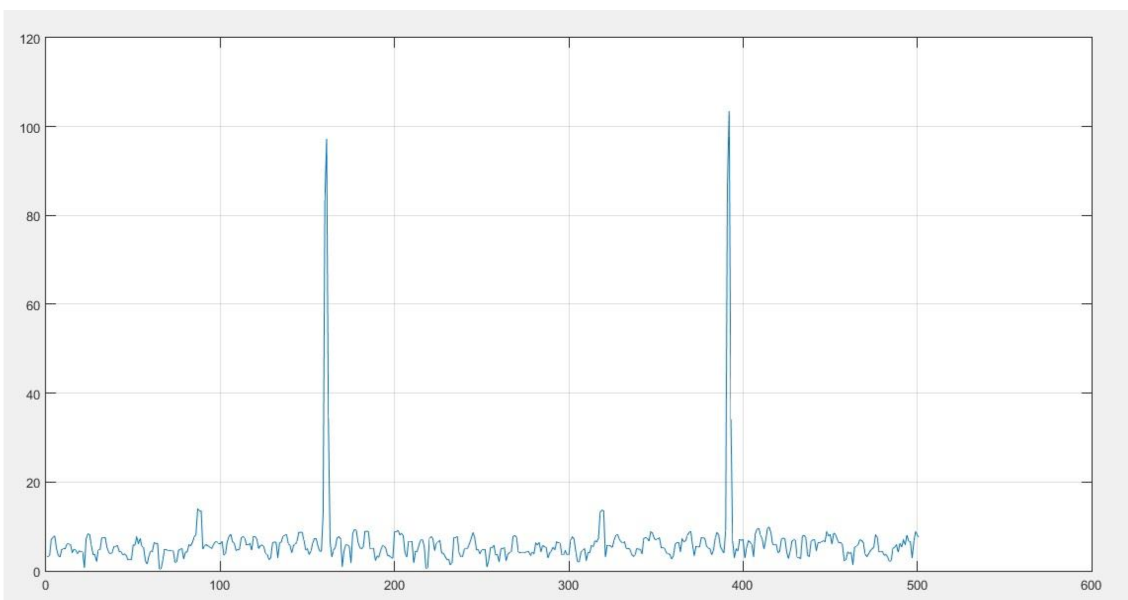

Figure 4.2 Plot for standard deviation of individual local window in a noisy Image

For a noisy image suffering from Gaussian Noise with $\sigma = 10$ it have been observed that value of standard deviation is not much effected for most of the local windows because of Gaussian distribution as explained in chapter 2. For 68 percent of times value of Gaussian random variable is low and for 27 percent of times Gaussian random variable value is medium and only for 5 percent of times value of standard deviation is high. In the following figure we can observe this. If we see figure 4.1 and figure 4.2 then we can observe that values of standard deviation of local windows have been slightly increased and the mean of the standard deviation of all individual local windows of the noisy image is 10.1593 while it was 5.75 for noise less image.

4.1.1 Calculation of Thresholding

Before computation of adaptive fuzzy distance we first compare standard deviation of current local window with the threshold value to know whether the standard deviation of local window is high. Hence we first need to decide a threshold value. To compute threshold we are using otsu's method of thresholding. To do so we first compute standard deviation for all the local windows in the image and store them in a two dimension matrix. This matrix is given as input (like an grayscale image) to the otsu's method and the result obtained through it is used as threshold.

4.1.2 Calculation of Adaptive Fuzzy Distances

In the concern window adaptive fuzzy distance between colour components of the central pixel and neighbouring pixels is calculated using ahead method:

Colour pairs are represented as follows

RG as $(d(p,q,1), d(p,q,2))$

RB as $(d(p,q,1), d(p,q,3))$

GB as $(d(p,q,2), d(p,q,3))$

Pairwise adaptive fuzzy distance between colour component of central pixel and that of nearby pixels is calculated as below:

Let's say adaptive distance between colour pairs of $d(p,q)$ central pixel and $d(p+m, q+n)$ nearby pixel within concern window.

$$AD_{rg}(p+m,q+n) \\ = ed_{rg}(p+m,q+n) - T_r(p+m,q+n) - T_g(p+m,q+n) \quad (4.2.1)$$

$$AD_{rb}(p+m,q+n) \\ = ed_{rb}(p+m,q+n) - T_r(p+m,q+n) - T_b(p+m,q+n) \quad (4.2.2)$$

$$AD_{gb}(p+m,q+n) \\ = ed_{gb}(p+m,q+n) - T_g(p+m,q+n) - T_b(p+m,q+n) \quad (4.2.3)$$

Where $AD_{rg}(p+m,q+n)$ represent Adaptive distance between Red Green colour pair, $AD_{rb}(p+m,q+n)$ represent Adaptive distance between Red Blue colour pair, $AD_{gb}(p+m,q+n)$ represent Adaptive distance between Green Blue colour pair of central pixel $d(m,n)$ and nearby pixel $d(p+m,q+n)$ within concern window, $ed_{rg}(p+m,q+n)$, $ed_{rb}(p+m,q+n)$ and $ed_{gb}(p+m,q+n)$ represents the Euclidean distance respectively between RG, RB and GB pairs, the terms T_r, T_g and T_b make colour pair wise distance adaptive, e.g. if the green components has almost similar value and red components have a large difference then the Euclidean distance would be large. Because of large Euclidean distance, it will give less weight to green component of the neighbourhood pixel while it should be large as both have almost same value. This problem will be solved when we subtract the term T_g in equation 4.2 and in opposite case the term T_r will take care of. Term $ed_{rg}(x+i,y+j)$, $ed_{rb}(p+m,q+n)$ and $ed_{gb}(p+m,q+n)$ are calculated as below:

$$ed_{rg}(p+m,q+n) \\ = \sqrt{(d(p+m,q+n,1) - d(p,q,1))^2 + (d(m,n,2) - d(p,q,2))^2} \quad (4.3.1)$$

$$ed_{rb}(p+m,q+n) = \sqrt{(d(p+m,q+n,1)-d(p,q,1))^2+(d(m,n,3)-d(p,q,3))^2} \tag{4.3.2}$$

$$ed_{gb}(p+m,q+n) = \sqrt{(d(p+m,q+n,2)-d(p,q,2))^2+(d(m,n,3)-d(p,q,3))^2} \tag{4.3.3}$$

T_r, T_g and T_b are calculated as below:

$$T_r(p+m,q+n) = \frac{1}{\left(\frac{d(p+m,q+n,1)-d(p,q,1)}{a}\right)^{2b}} \tag{4.4.1}$$

$$T_g(p+m,q+n) = \frac{1}{\left(\frac{d(p+m,q+n,2)-d(p,q,2)}{a}\right)^{2b}} \tag{4.4.2}$$

$$T_b(p+m,q+n) = \frac{1}{\left(\frac{d(p+m,q+n,3)-d(p,q,3)}{a}\right)^{2b}} \tag{4.4.3}$$

In the same Adaptive distance can be calculated between red blue and green blue colour pairs. To fuzzify these adaptive distance we introduce a membership function called **SHORT** which determine till what extend an adaptive distance is SHORT and is defined as below

$$\mu_t(\lambda) = \begin{cases} exp\left(\frac{\lambda}{t}\right) & for\ \lambda \le t \\ 0 & for\ \lambda > t \end{cases} \tag{4.5}$$

Figure below shows Membership Function for the set "SHORT". Maximum distance between a nearby pixel and pixel of interest in a window is represented by Parameter 't'.

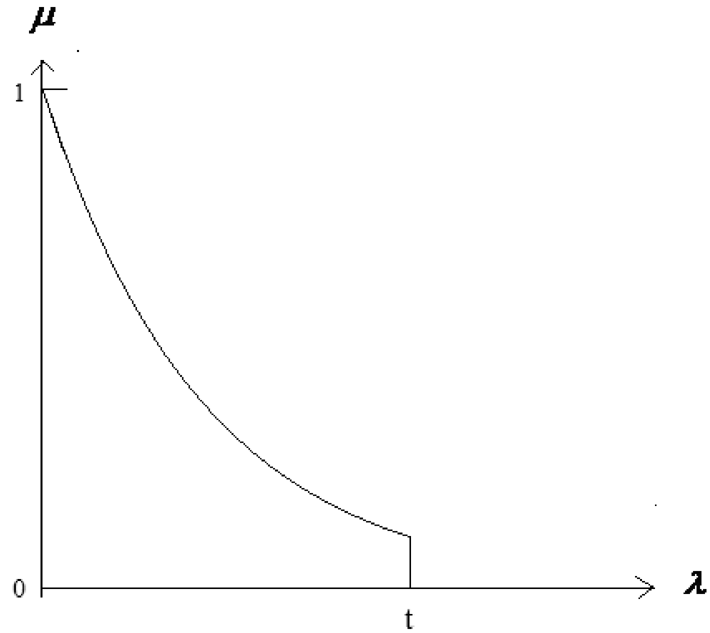

Figure 4.3 Membership Function SHORT

For different colour pairs Parameter 't' is given as below:

$$t_{rg}(p,q) = \max\left(ed_{rg}(p+m,q+n)\right) \quad (4.6.1)$$

$$t_{rb}(p,q) = \max\left(ed_{rb}(p+m,q+n)\right) \quad (4.6.2)$$

$$t_{gb}(p,q) = \max\left(ed_{gb}(p+m,q+n)\right) \quad (4.6.3)$$

4.2 Structure of Filter

Filter is designed in such a manner that it will compute the weighted average of the nearby pixels in the concern window. However the neighbouring pixels weights are calculated using the following fuzzy rules.

FUZZY RULE TO DETERMINE THE WEIGHT OF RED COMPONENT

IF $AD_{rg}(p+m,q+n)$ is SHORT & $AD_{rb}(p+m,q+n)$ is SHORT

THEN WEIGHT $w(p+m,q+n,1)$ is a large value. (4.7.1)

FUZZY RULE TO DETERMINE THE WEIGHT OF GREEN COMPONENT

IF $AD_{rg}(p+m,q+n)$ is SHORT & $AD_{gb}(p+m,q+n)$ is SHORT

THEN WEIGHT $w(p+m,q+n,2)$ is a large value. (4.7.2)

FUZZY RULE TO DETERMINE THE WEIGHT OF BLUE COMPONENT

IF $AD_{rb}(p+m,q+n)$ is SHORT & $AD_{gb}(p+m,q+n)$ is SHORT

THEN WEIGHT $w(p+m,q+n,3)$ is a large value. (4.7.3)

After the calculation of adaptive fuzzy distance with the help of membership function SHORT, implement above fuzzy rules. For example, adaptive fuzzy distance between the green & blue colour component of central pixel (p,q) and a nearby pixel $(p+m,q+n)$ in concern window is presented as $\mu_{gb(p,q)}(AD_{gb}(p+m,q+n))$, where μ_{gb} is membership function of the green & blue colour component.

Neighbouring pixels RGB colour components weights at location $(p+m,q+n)$ in the window of interest using above described fuzzy rules are computed as:

$$w(p+m,q+n,1) = \min\left\{\mu_{rg(p,q)}\left(AD_{rg}(p+m,q+n)\right),\ \mu_{rb(p,q)}\left(AD_{rb}(p+m,q+n)\right)\right\} \quad (4.8.1)$$

$$w(p+m,q+n,2)$$

$$= \min\left\{\mu_{rg(p,q)}\left(AD_{rg}(p+m,q+n)\right),\ \mu_{gb(p,q)}\left(AD_{gb}(p+m,q+n)\right)\right\} \quad (4.8.2)$$

$$w(p+m,q+n,3)$$

$$= \min\left\{\mu_{gb(p,q)}\left(AD_{gb}(p+m,q+n)\right),\ \mu_{rb(p,q)}\left(AD_{rb}(p+m,q+n)\right)\right\} \quad (4.8.3)$$

In the window of interest improved value of red, green and blue components of central pixel is calculated as:

$$R(x,y,1) = \frac{\sum_{m=-k}^{m=k}\sum_{n=-k}^{n=k} w(p+m,q+n,1) * d(p+m,q+n,1)}{\sum_{m=-k}^{m=k}\sum_{n=-k}^{n=k} w(p+m,q+n,1)} \quad (4.9.1)$$

$$R(x,y,2) = \frac{\sum_{m=-k}^{m=k}\sum_{n=-k}^{n=k} w(p+m,q+n,2) * d(p+m,q+n,2)}{\sum_{m=-k}^{m=k}\sum_{n=-k}^{n=k} w(p+m,q+n,2)} \quad (4.9.2)$$

$$R(x,y,3) = \frac{\sum_{m=-k}^{m=k}\sum_{n=-k}^{n=k} w(p+m,q+n,3) * d(p+m,q+n,3)}{\sum_{m=-k}^{m=k}\sum_{n=-k}^{n=k} w(p+m,q+n,3)} \quad (4.9.3)$$

4.3 Algorithm

Step 1: In the noisy image consider a window of size $w \times w$ centred at pixel of interest.

Step 2: Compute the standard deviation of RGB colour components of all pixels within concerned window.

Step 3: If the standard deviation is lower than threshold limit for any colour components go to step 8.

Step 4: Take a pixel of concerned window and for each colour component of this pixel consider a 3 x 3 kernel centred at this pixel and calculate the mean of each of colour component of the pixels of this 3 x 3 kernel.

Step 5: Repeat step 4 for all pixels of the window of interest and store result in w x w matrices for each colour components.

Step 6: Compute the difference between central pixel's colour components result (mean) and neighbouring pixels colour component mean (stored in temporary w x w matrices).

Step 7: If the differences calculated in above step is lower than threshold value for more than half of the neighbouring pixels then consider the values stored in w x w matrices for each colour components in place of values present in noisy image for further computation else continue.

Step 8: Pick up a nearby pixel within the window and calculate adaptive distances for each of three colour pairs red green, red blue and green blue between this pixel and central pixel using Equations 4.2.1, 4.2.2 and 4.2.3.

Step 9: Fuzzify these adaptive distances with the help of membership function defined in Equation 4.5 and parameter 't' with the help of Equations 4.7.1, 4.7.2 and 4.7.3.

Step 10: For Each colour components compute weights of the nearby pixel which has been picked in Step 8 with the help of Equations 4.8.1, 4.8.2 and 4.8.3.

Step 11: Repeat Step 2 to 10 for all neighbourhood pixels in the window.

Step 12: Using Equations 4.9.1, 4.9.2 and 4.9.3 compute the improved value of colour components of central pixel.

Step 13: Repeat for all the pixels contained in image.

CHAPTER 5

RESULTS AND COMPARATIVE ANALYSIS

5.1 Output of the Fuzzy based Gaussian Filter

A colour image digitally viewed as a $M \times N \times 3$ array of pixel values. Pixel at index (x, y) can be treated as a stack of 3 gray scale image for RGB colour components respectively. The range of values of these colour components depends on the data class of component image. If the image is of the class double then the range will be 0 to 1. Also for the class uint8 the range will be 0 to 255 and for the class uint16 the range will be 0 to 65535.

The Gaussian Noise has been introduce in the colour images of 'Lena', 'Fish' and 'Bird'. These images belong to the image class of uint8 having size 256×256 and then proposed algorithm has been applied to restore image. The noiseless image or original image of 'Lena', 'Fish' and 'Bird' has been shown in figure 5.1, 5.2 and in 5.3 respectively. Gaussian noise having standard deviation equal to 10, 20 and 30 have been introduced in these images and then proposed Fuzzy Based Gaussian Noise Reduction filter has been applied to restore the image.

Figure 5.1 Noiseless Lena Image

Figure 5.2 Noise less Fish Image

Figure 5.3 Bird Image without Noise

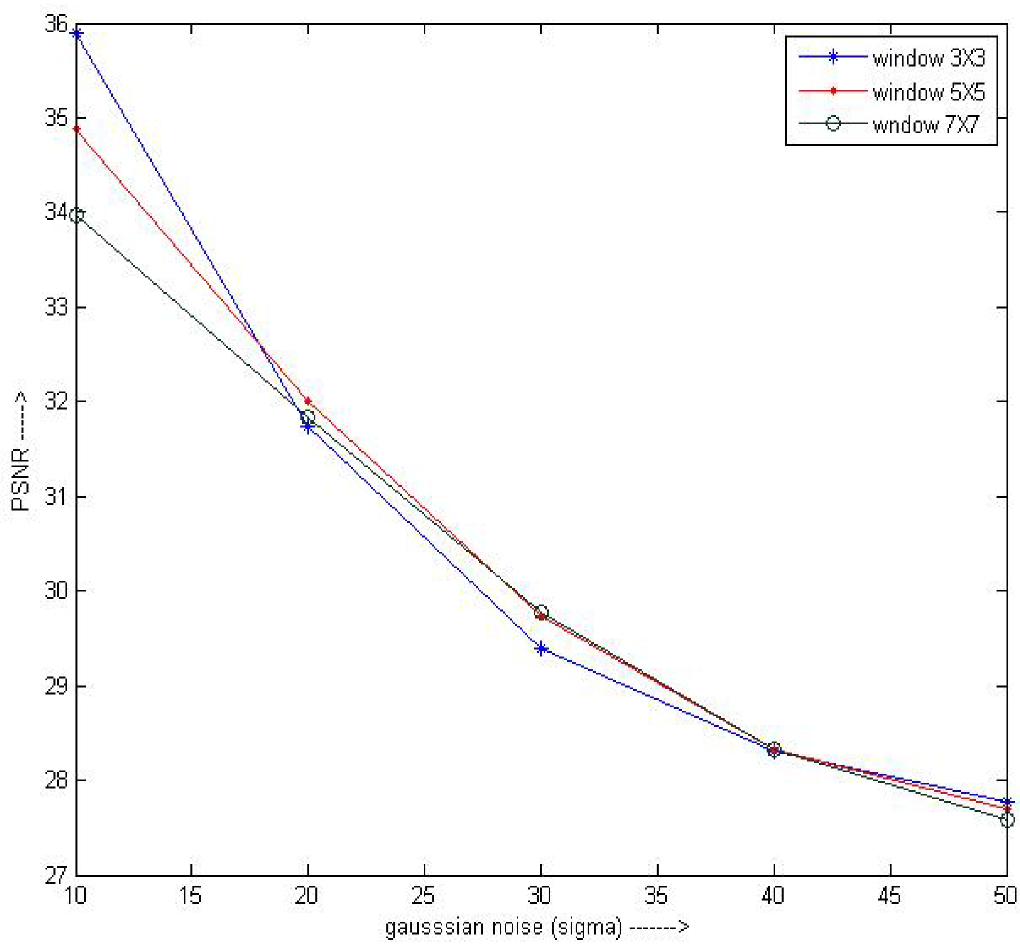

Figure 5.4 PSNR vs. Window sizes for Gaussian noise

For the purpose of accessing the quantitative measure of goodness of proposed image restoration method I have been used image metric MSE and PSNR. The performance of the Gaussian Noise is assess with the help of test colour images with different levels of Gaussian Noise. Experiments are performed by means of filtering windows having sizes $3 \times 3, 5 \times 5$ and 7×7. It has been observed that the window size of 3×3 is more appropriate for the Gaussian noise having noise level until $\sigma = 30$ for colour images. Figure 5.4 represents results for this observation in the form of a plot between PSNR and Gaussian Noise level with the help of standard deviation.

For the images having higher levels of Gaussian noise, filter window of larger size observed to be more appropriate but there exist a drawback of burring effect with higher filter window size that is over smoothing of the edges.

Image	Noisy		FNRC		NRFF		Proposed	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
Lina								
$\sigma = 10$	30.28	33.31	25.22	34.11	21.94	34.72	13.18	36.93
$\sigma = 20$	55.65	30.67	45.24	31.71	45.08	31.59	24.73	34.19
$\sigma = 30$	66.26	29.91	55.18	30.37	53.33	30.48	33.43	32.88
Fish								
$\sigma = 10$	39.38	32.17	32.10	33.07	26.77	33.85	23.95	34.33
$\sigma = 20$	75.99	29.38	75.50	29.35	46.02	31.50	38.78	32.24
$\sigma = 30$	90.83	28.54	86.34	28.77	60.26	30.33	52.10	30.96
Bird								
$\sigma = 10$	39.97	32.11	31.91	33.09	21.21	34.86	15.86	36.12
$\sigma = 20$	74.73	29.39	58.01	30.50	39.61	32.15	33.53	32.87
$\sigma = 30$	89.91	28.59	73.77	29.45	55.14	30.72	51.85	30.98

Table 5.1 Comparison of Performance of Proposed Method using MSE and PNRC

Proposed method performance has been compared with methods developed by Stefan Schulte *et al.* (FNRC) [4] and Dimitri Van *et al.* (NRFF) [1]. Comparison of MSE and PSNR values are used for performance comparison with FNRC and NRFF and Table 5.1 represent the results of MSE and PSNR values of noisy image, FNRC, NRFF and of proposed method. The figure 5.5 shows graph of PSNR vs. Gaussian noise in term of standard deviation for noisy image, FNRC method, NRFF method and proposed method. In this table results of FNRC and NRFF has been obtained from source [44].

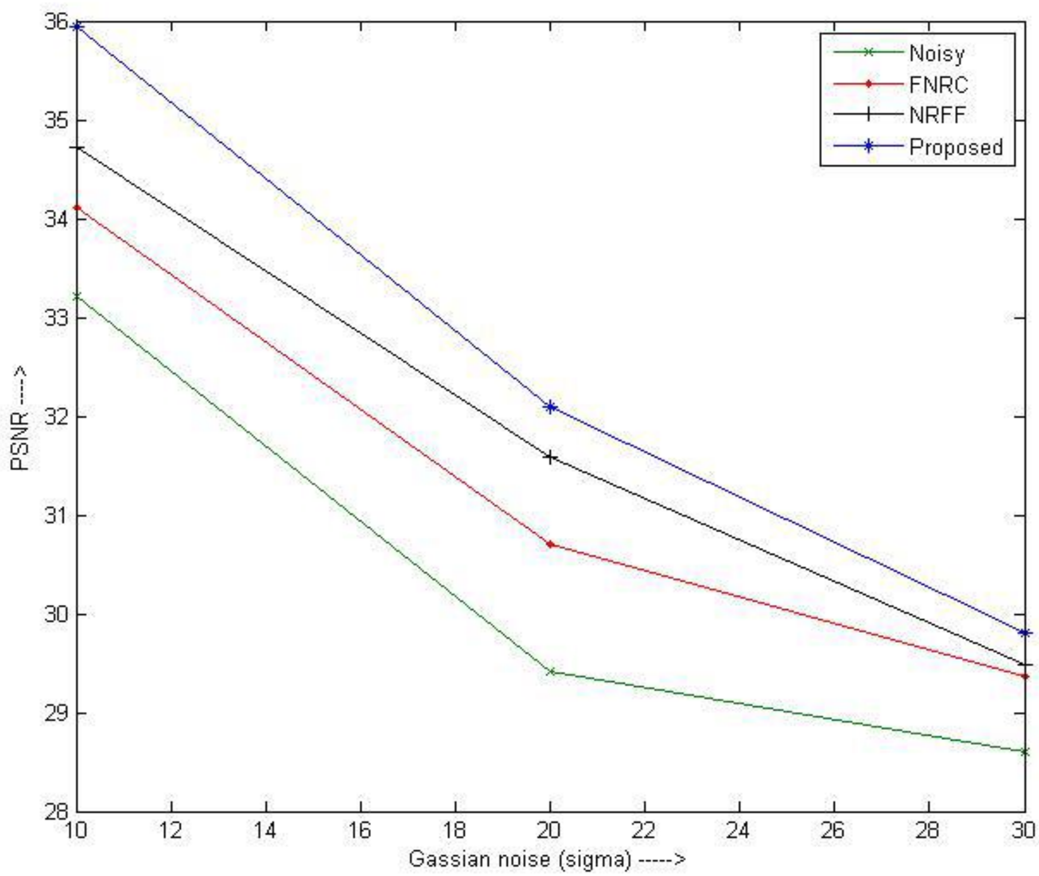

Figure 5.5 A comparative analysis of Gaussian Noise Reduction

In figure 5.6, 5.7, 5.8 and in 5.9 Lena image with Gaussian noise having $\sigma = 20$, filtered Lena image with proposed method, with FNRC and with NRFF has been shown respectively. In figure 5.10, 5.11, 5.12 and in 5.13 Fish image with Gaussian noise having $\sigma = 20$, filtered Fish image with proposed method, with FNRC and with NRFF has been shown respectively. Similarly In figure 5.14, 5.15, 5.16 and in 5.17 Bird image with Gaussian noise having $\sigma = 20$, filtered Bird image with proposed method, with FNRC and with NRFF has been shown respectively.

Figure 5.6 Lena Image with Gaussian Noise

Figure 5.7 Lena Image with NRFF

Figure 5.8 Lena Image with FNRC

Figure 5.9 Lena Image with proposed Method

Figure 5.10 Fish Image with Gaussian Noise

Figure 5.11 Fish Image with NRFF

Figure 5.12 Fish with FNRC

Figure 5.13 Fish with Proposed Method

Figure 5.14 Bird Image with Gaussian Noise

Figure 5.15 Bird Image with NRFF

Figure 5.16 Bird Image with FNRC

Figure 5.17 Bird Image with Proposed Method

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

The efforts for reduction of Gaussian noise in a colour image is a made with the help of proposed fuzzy based Gaussian noise reduction filter. This purpose is fulfilled with the help of examination of the local window standard deviation, the adaptive distances between the pairs of the colour components of a concerned pixel and the nearby pixel and membership function Small to compute fuzzy adaptive distance. The relations among the colour components for neighbourhood pixels to the pixel of interest are established by computing the weights for the neighbourhood pixel by considering pairs of colour components at place of individual colour components.

This standard deviation and adaptive fuzzy distance among the colour pairs based methodology produces a filtered image which preserves all the substantial details to a higher extent and ensure the reduction of noise having various degree of noise levels in a colour image. The performance of the proposed filter has been evaluated in terms of MSE and PSNR on several test colour images. The designed filter defend its performance by reducing Gaussian noise of different levels on the highly complex test colour images. It is observed that the restored image is blurred to some extent but the designed filter try to deblur the image to a higher extent.

Future work will try to find out more appropriate threshold values to reduce the Gaussian noise to a much higher extent. To reduce deblurring proposed thresholds are based on otsu’s method for global thresholding.

REFERENCES

[1] Taguchi, H. Takashima, and Y. Murata, "A Fuzzy Filter for Image Smoothing", Proceedings of International SPIE Conf. on Nonlinear Image Processing V, pp. 332-339, 1994.

[2] Taguchi, H. Takashima, and F. Russo, "Data Dependent Filtering using The Fuzzy Inference", Proceedings of IEEE Instrumentation and Measurement Technology Conference, pp. 752-756, 1995.

[3] D. Androutsos, K.N. Plataniotis, and A.N. Venetsanopoulos, "Colour image processing using vector rank filters", International Conf. on Digital Signal Processing, Vol. 2, pp. 614-619, 1995.

[4] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images", Proc. IEEE Int. Conf. Computer Vision, pp. 839-846, 1998.

[5] D. Van De Vile, M. Nachtegael, D. Vander Weken, W. Philips, I. Lemahieu, and E. E. Kerre, "A new Fuzzy Filter for Gaussian Noise Reduction", Proceedings of International SPIE Conf. on Electronic Imaging, pp. 1-9, 2001.

[6] D. Van De Ville, M. Nachtegael, D. Van der Weken, E.E. Kerre, W. Philips, and I. Lemahie, "Noise Reduction by Fuzzy Image Filtering", IEEE Trans. on Fuzzy Systems, Vol. 11, No. 4, pp. 429-436, 2003.

[7] D-H. Shin, R-H. Park, S. Yang, and J.H. Jung, "Block-Based Noise Estimation Using Adaptive Gaussian Filtering", IEEE Transactions on Consumer Electronics, Vol. 51, No. 1, February 2005.

[8] E. Abreu, M. Lightstone, S. Mitra, and K. Arakawa, "A new efficient approach for the removal of impulse noise from highly corrupted images", IEEE Trans. on Image Processing, Vol. 5, pp. 1012-1025, June 1996.

[9] F. Russo, "A fuzzy approach to digital signal processing: Concepts and applications", Proc. of IEEE Instrumentation and Measurement Technology Conference IMTC92, New York, pp.640-645, 1992.

[10] F. Russo, "Noise cancellation using nonlinear fuzzy filters", Proc. of IEEE Instrumentation and Measurement Technology Conference, Vol. 2, pp. 77-2777, 1997

[11] F. Russo, "A technique for image restoration based on recursive processing and error correction", Proc. of IEEE Instrumentation and Measurement Technology Conference, Vol. 3, pp. 1232- 1236, 2000.

[12] F. Russo, "A Method for Estimation and Filtering of Gaussian Noise in Images", IEEE Trans. on Instrumentation and Measurement, Vol. 52, No. 4, 2003.

[13] F. Russo, "A Method Based on Piecewise Linear Models for Accurate Restoration of Images Corrupted by Gaussian Noise", IEEE Trans. on Instrumentation and Measurement, Vol. 55, No. 6, 2006.

[14] F. Russo, "FIRE operators for Image Processing", Fuzzy Sets and Systems, Vol. 103, pp. 265-275, 1997.

[15] F. Russo, "Noise Cancellation using Non-linear Fuzzy Filters", Proc. of IEEE Instrumentation and Measurement Technology Conference, pp. 772-777, May 1997.

[16] G. Pok, J. Liu and A. S. Nair, "Selective removal of impulse noise based on homogeneity level information", IEEE Trans. on Image Processing, Vol. 12, pp. 85-92, Jan. 2003.

[17] R. Garnett, T. Huegerich, C. Chui, and W. He, "A universal noise removal algorithm with an impulse detector", IEEE Trans. on Image Processing, Vol 14, Issue 11, pp. 1747-1754, November 2005.

[18] H.J. Zimmermann**,** Fuzzy Set Theory and its Applications. (3rd ed.) Kluwer Academic Publishers, Boston, Dordrecht, Lancaster, 1996.

[19] H.S. Kam, M. Hanmandlu, and W.H. Tan, "An adaptive fuzzy filter system for smoothing noisy images", Proc. of Intl. Conf. on Convergent Technologies for Asia-Pacific Region, Vol.4, pp. 1614 – 1617, 2003.

[20] H.S. Kam and M. Hanmandlu, "An improved fuzzy enhancement by adaptive parameter selection", Proc. of IEEE International Conference on Systems, Man & Cybernetics, Vol. 2, pp. 2001-2006 , 2003.

[21] J.Y. Jiu, "Multilevel Median Filter based on Fuzzy Decision", DSP IC E.E. NTU., 1996.

[22] J.Portilla, V. Strela, M. Wainwright, and E.Simoncelli, "Image denoising using Gaussian scale mixtures in the wavelet domain," IEEE Trans. on Image Process., Vol. 12, No. 11, pp. 1338–1351, Nov. 2003.

[23] K. J. Overton and T. E. Weymouth, "A noise reducing preprocessing algorithm", Proceedings of the IEEE Computer Science Conference on Pattern Recognition and Image Processing, pp. 498-507, Chicago, IL, 1979.

[24] K. Arojawa, "Median Filter based on Fuzzy Rules and its Application to Image Restoration", Fuzzy Sets and Systems, Vol. 77, pp. 3-13, 1996.

[25] K. Arojawa, Edited by E.E. Kerre and M. Nachtegael, "Fuzzy Ruled-Based Image Processing with Optimization", Springer- Verlag, pp. 222-247, 2000.

[26] L. Zadeh, "Fuzzy sets", Journal of Information and Control, Vol 8, pp.338.353, 1965.

[27] L. Khriji, M. Gabbouj, "A Class of Multichannel Image Processing Filters", IEICE Trans. on Information and System, Vol. E82-D, No.12, pp.1589-1596, 1999.

[28] M. Hanmandlu, D. Jha, B. Vittal and Vivek Gera, "Image smoothing using fuzzy clustering", Proc. of MMU International Symposium on Information and Communication Technologies (M2USIC' 2001), Kaula Lumpur, Malaysia, pp. 4.1(1)-4.1(5), 16-17 October 2001.

[29] M. Nachtegael, S. Schulte, D. Van der Weken, V. De Witte, and E. E.Kerre, "Fuzzy filters for noise reduction: The case of Gaussian noise," Proc. IEEE Int. Conf. Fuzzy Systems, pp. 201–206, 2005.

[30] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion", IEEE Trans. Pattern Anal. Machine Intell., Vol. 12, pp. 629639, 1990.

[31] P-E. Ng and K-K. Ma, "A Switching Median Filter with Boundary Discriminative Noise detection for Extremely Corrupted Images", IEEE Trans. on Image Processing, Vol.15, No. 6, June 2006.

[32] F. Russo and G. Ramponi, "A noise smoother using cascaded FIRE filters", Proc. of 4th Intl. Conf. on Fuzzy Systems, Vol. 1, pp. 351-358, 1995.

[33] S. Schulte, B. Huysmans, A. Pizurica, E.E. Kerre, and W. Philips, "A new fuzzy-based wavelet shrinkage image denoising technique," Lecture Notes Comput. Sci., vol. 4179, pp. 12–23, 2006.

[34] S. Schulte, V. De Witte, M. Nachtegael, D. Van der Weken, and E.E. Kerre, "Fuzzy Two-Step Filter for Impulse Noise Reduction From Color Images", IEEE Trans. on Image Processing, Vol. 15, No. 11, November 2006.
[35] S. Schulte, V. De Witte, and E.E. Kerre, " A Fuzzy Noise Reduction Method for Color Images", IEEE Transactions on Image Processing ,Vol.16, No. 5. May 2007.
[36] S. Schulte, S. Morillas, V. Gregori, and E.E. Kerre, "A New Fuzzy Color Correlated Impulse Noise Reduction Method", IEEE Trans. on Image Processing, Vol. 16, No. 10, pp. 2565-2575, 2007.
[37] S.M. Roomi and I.M. Lakshmi and V.A. Kumar, "A Recursive Gaussian Weighted Filter for Impulse Noise Removal", GVIP Special Issue on Denoising, pp. 21-25, 2007.
[38] S. Md. Mansoor Roomi and T. Pandy Maheswari and V. Abhai Kumar, "A Detail Preserving Filter for Impulse Noise Detection and Removal", GVIP Special Issue on Denoising, pp. 35-40, 2007.
[39] T. Sun and Y. Neuvo, "Detail-preserving median based filters in image processing", Pattern Recognition Letters. Vol. 15, pp. 341-347, Apr. 1994.
[40] Wang J. H. Yang, Chiu H.C., "An Adaptive Fuzzy Filter for Restoring Highly Corrupted Image by Histogram Estimation", Proc. Natl. Sci. ROC(A), Vol. 23, No. 5, pp. 630-643, 1999.
[41] W. Luo, "An Efficient Detail-Preserving Approach for Removing Impulse Noise in Images", IEEE Signal Processing Letters, Vol. 13, No. 7, July 2006.
[42] X. Zheng and Q. Gao, "Image Noise Removal Using Perceptual Edge Features", GVIP Special Issue on Denoising, pp. 15-20, 2007.
[43] Y. S. Choi and R. Krishnapuram, "A Robust Approach to Image Enhancement Based on Fuzzy Logic", IEEE Trans. Image Proc., Vol. 6, pp. 808-825, June 1997.
[44] O.P. Verma, M. Hanmandlu, A. S. Parihar and V. K. Madasu, "Fuzzy filters for Noise Reduction in Color Images", Graphics, Vision and Image Processing Journal, Vol. 9, Issue 5, ICGST, pp. 29-43,Sep. 2009