IMAGE DE-NOISING AND COMPRESSION USING STATISTICAL BASED THRESHOLDING IN 2D WAVELET TRANSFORM.

By

QAZI MAZHAR UL HAQ

Submitted to the Faculty of Electrical Engineering

National University of Sciences and Technology, Rawalpindi in partial

Fulfillment for the requirements of a M.S Degree in Electrical (Telecom) Engineering

AUGUST 2016

SUPERVISOR CERTIFICATE

It is to certify that the final copy of thesis has been evaluated by me, found as per the specified format and error free.

Date__________

____

Col. Dr. Imran touqir

ABSTRACT

Images are very good information carriers but they depart from their original condition during transmission. They are corrupted by different kinds of noises during communication and transmission. All the images need to be de-noised before processing. The aim is to de-noise an image such that minimum amount of information is lost and maximum amount of noise is reduced. For de-noising different kind of techniques is applied i.e. linear, non-linear, adaptive and non-adaptive techniques are observed. Soft thresholding, hard thresholding, universal thresholding (Visu shrink) are used but the performance is not still good. De-noising through two dimensional discrete wavelet transform is three step process: wavelet decomposition, wavelet thresholding and wavelet reconstruction. The discrete wavelet transform gives the sparse representation of the image which is very best for the optimal threshold value selection. We used statistical based thresholding methods for de-noising which shows improved results than existing techniques.

The goal of this research is to compare the performance of different statistical based thresholding techniques. The results of using these wavelet bases are compared on the basis of peak signal to noise ratio and mean square error. The research shows that use of bi orthogonal wavelets bases is better than orthogonal wavelet bases. We used bi-orthogonal wavelets version 6.8 improved the results by statistical thresholding methods.

ACKNOWLWDGEMENTS

First and foremost I must pay gratitude to Allah Almighty for giving me access to new horizons in the literary research.

I would like to thank Military College of Signals, NUST, for providing admirable research environment at the institute. This painstaking task was not possible without the support of my supervisor Col. Dr. Imran Taoqir who ignited critical abilities in me. I am indebted to him for his timely guidance, profound encouragement and formative criticism. His affectionate and kind consideration towards my research helped me to carry on with my project in odd circumstances. I am also very grateful to my committee members, Lt. Col. Dr. Adil Masood Siddique and Col. Dr. Imran Rashid for their involvement in this research work.

Finally, this work would have never materialized, had there been no consistent prayers, never ending love, support and encouragement of my parents. Thank you for permissiveness of my absence from home in pursuit of this work.

DEDICATION

Dedicated to my

Father and Mother

Whose love, affection and support enabled me to achieve yet another milestone in my professional life.

LIST OF ACRONYMS

1) Bits per pixel	**Bpp**
2) Compression Ratio	**CR**
3) Daubaches	**db**
4) Discrete Fourier Transform	**DFT**
5) Discrete Wavelet Transform	**DWT**
6) Wavelet Thresholding	**WT**
7) Structural similarity index	**SSIM**
8) Fast Fourier Transform	**FFT**
9) Mean Square Error	**MSE**
10) Peak Signal to Noise Ratio	**PSNR**
11) One Dimensional	**1 D**
12) Bi-orthogonal	**Bior**
13) Mean opinion score	**MOS**
14) Two Dimensional	**2 D**

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Background

A very large portion of digital image processing is devoted to image de-noising and image compression. Applied engineers don't believe that images received will not contain noise. Image are corrupted by noise. The de-noising of images is a big problem in digital image processing. The de-noising is the removal of degradation or blurring introduced in images during communication [1]. Blurring and visual artifacts comes from different kind of noise and also due to the imperfect image formation such as relative motion of camera and the original scene [2].

Wavelet theory has great application in digital image processing. Wavelets are developed by Morlet and Grossman. The relationship between wavelets and filter banks is developed by French researchers Meyer, Mallat and Cohen [3], this theory is now used in most of technical work. The main application of wavelet transform is image or signal compression and de-noising. In image de-noising and compression wavelet transform is successful because to de-noise and compress an image it needs the sparse representation of the image which is best done by wavelet transform. Wavelet theory is ensured by the separation of signal and noise component of different discrete wavelet transform (DWT) [4]. Filtering the coefficients, most of the noisy coefficient is dropped. Each method of de-noising using wavelets is passed through the following three prototypal steps: computing the DWT of the image to be de-noised (wavelet decomposition), filtering/thresholding the decomposed image, corresponding inverse wavelet transform.

The Daubechies [5] work is useful for those who have limited knowledge of wavelets and mathematics. Wavelet transform is very useful and interesting tool for investigating images. Wavelet transform overcome the limitation of Fourier transform. Fourier transform shows the signal only in time domain or in frequency domain. It only shows the global information of the signal. A mother wavelet whose mean is zero have all its energy in time domain and is well observed by time varying signals. Scaling and wavelets function are basis function in wavelet transforms. Mallat [6] multi resolution representation theory allowed researchers to create their own family of wavelets on the specific criteria. The usefulness of DWT over other transformation is because it shows time frequency localization.

For Image de-noising we need to know about the degradation process in order to develop an algorithm to improve the quality and minimize the noise in image. When we have a model of the degradation process, we can use the inverse process to bring back the image into its original position. This type of restoration is used in the spacecraft to recoup for degradation and remove the artifacts in telescope of the optical system. Image de-noising has great application in the fields of astronomy where resolution is very bad, and in medical imaging which quality of the images to diagnose the patient, and in forensic science where useful photographs are extremely bad and needs high quality photographic evidences [7].

Now consider a two dimensional (2D) gray scale image which can be represented by f(x,y) an array of data where (x,y) is the pixel coordinates. The pixel intensity is the value on a gray scale level from 0-255. Binary images are represented with a value of 0 for black and 1 for white is the simple form of digital image. This form of images are very useful in study of image processing and computer vision and pattern recognition. The images used in the thesis for experimentation purpose are gray scale images which have no color information and only shows the brightness of the image on gray scale (0-255) 0 is black image and 255 is white image and values between 1-254 have different intensity levels. These are also called intensity images.

Color images are three band monochrome images in which each band is different color RGB, Red Green and Blue are three different bands. All other colors are made from the combination of these three color bands. These are also called RGB images and are 24 bits/pixel images while the binary images are 1 bit/pixel.

1.2 Research motivation

Image de-noising and compression is very important task in digital image processing and computer vision. Digital images are used in different kind of application i.e Satellite images, Radar Images, Medical images, images used in forensic evidence. All these images when processed are dealt by different equipment which induces different kind of noise in it. Before the images are processed by system it need to be de-noised. The de-noising process eliminates noise form images such that minimum amount of information is lost and maximum amount of noise is reduced. If Gaussian noise[8] of variance 10 is added to a grayscale image the peak signal to noise ratio (PSNR) is reduced to about 10%. It need to improve the PSNR and minimize mean square error (MSE).

1.3 Problem formulation and thesis layout

The basic idea behind this thesis is the approximation of the uncorrupted image from a distorted and noisy images, this process is also called de-noising. Image de-noising is done by different kind of techniques to restore the original image from noise contaminated image. Selecting a best applicable technique plays major role in image de-noising. For example the techniques that are developed for satellite images will enhance minor details and will be according to satellite images. So these techniques will not be suitable for medical images. In this thesis a study is done on different kind of techniques that are already used for image de-noising and compression. State of art technique are globally and locally threshold techniques in which local dependent thresholding gives best result. There are two types of techniques used for de-noising process in wavelets thresholding. Adaptive [9] and non-adaptive techniques are used. Which shows very poor performance in image de-noising. Blurring are visual artifacts are the main disadvantages of the state of art techniques. In this thesis two methods are used for image de-noising that are statistically dependent. The threshold value is obtained from data driven method. Finding best threshold value will enhance the result of de-noised image.

Image de-noising is a three step process, a linear wavelet transform, non-linear thresholding, linear inverse wavelet transform.

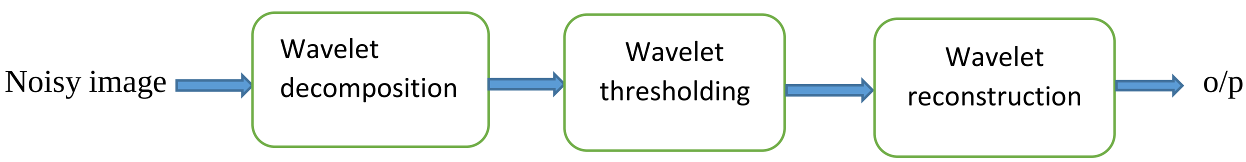


Figure 1.1 Image de-noising three step process

In figure 1.1 wavelet decomposition is to decompose the input noisy image. We can decompose the image up to ‘L’ levels. The wavelet decomposition gives us details and approximation of the input image. The details shows us vertical, horizontal and diagonal details. The approximation details shows the original image coefficients which should not be threshold during thresholding section. The details are thresholding during the threshold section and then reconstruction of the image wavelets occurs which reconstructs the image and gives us the estimated image or de-noised image.

Literature review

Chapter 2 Wavelets analysis

2 Wavelet definition

The name “Wavelets” is referred to oscillatory evanescent wave time-circumscribed elongate, which has the competency to describe the time-frequency plane, with atoms of different time fortifies (see in fig 2.1). Generally, wavelets are purposefully crafted to have categorical properties that make them utilizable for signal processing. They represent a felicitous implement for the analysis of non-stationary or transient phenomena. In mathematics wavelets are those functions whose value over the period is zero and is defined in a finite time interval [10, 11]. In wavelets the data is converted into different frequencies and every frequency represents similar resolution and scale.

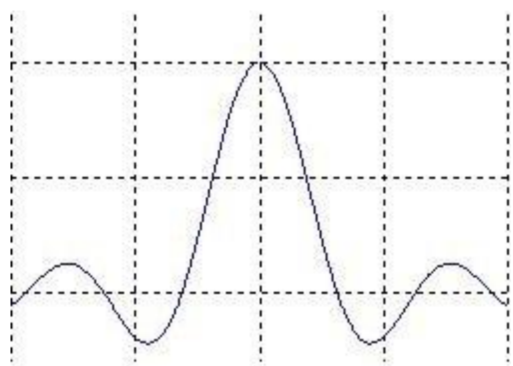


Figure 2.1 Mother wavelet w(t)

2.1 Wavelet characteristics

Wavelet is used to extract information from many kinds of data including EEG signals, Audio signals and Digital images etc. The wavelet ψ is a function with finite limits and average value of zero. The formula for this wavelet is:

$$\int_{-\infty}^{\infty} \psi\ (t)\ dt\ =\ 0 \qquad (2.1)$$

In order to get more extensible information about time and frequency a wavelet family can be constructed by a function of $\psi(t)$ known as Mother Wavelet. Daughter wavelets can formed from mother wavelets with a scaling factor and translation factor $\psi_{u,s}(t)$ here $\psi(t)$ is translation factor and *s* is dilation factor with scale *s*.

$$\psi_{u,s}(t) = \frac{1}{\sqrt{u}}\, \psi \left[\frac{t-s}{u}\right] \;;\; u, s \in R^1 \text{ and } u > 0 \qquad (2.2)$$

2.2 Wavelet analysis

Wavelets analysis is the decomposition of the images to different levels. It is performed to by the projection of the signal over wavelet function. It is multiplication and integration process.

$$< x(t)\psi_{u,s}(t) >= \int x(t)\, \psi u, s\ (t) d(t) \qquad (2.3)$$

We can use different scales and translations of mother wavelet depending on the shape and characteristics of signal. The singularity of the wavelet permit us to use willingly different scales and translations or to change the size of the function or window to make it compatible with resultant resolution in time and frequency domain. We will note the abrupt changes occur in the original image in high resolution time domain and we do that by contracted the mother wavelet. Conversely for high resolution in frequency domain we use a dilated version of the mother wavelet.

2.3 Wavelet history

The first wavelet was introduced in 1909 by Haar. In 1946 Denis Gabor, introduced Gabor transform, which is used in wavelet analysis, a family of functions that can be used for generating new functions or new wavelet family. In 1975 George Zweig, discovered continues wavelet transform. Morlet[7] while doing his research observed that small scales of high frequency is most useful for finding fine details for closely-spaced layers. Grossman, recognized some more ideas in Morlet research on coherent quantum science. In 1982 Grossman and Morlet formulated continues wavelet transform (CWT). Meyer know the importance of this mathematical tool and discovered his own theory in collaboration with Ingrid Duabechies[5], founder of Orthogonal wavelets and Stephen Mallat,[7] founder of filter bank implementation with Discrete Wavelets Transform (DWT).

2.4 Evolution of wavelet transform

The evolution of the wavelet transform comes from the Fourier transform (FT). The wavelet is actually an extension to the limitation of Fourier transform (FT). In Fourier transform bases functions are sine and cosine sinusoids and are predictable while in wavelet transform (WT) the bases functions are symmetric and unpredictable. The whole story of how Fourier transform is made and how wavelet transform overcome the limitation of Fourier transform is discussed in following topics.

2.5 Fourier transform

Joseph Fourier, a French mathematician and physicist proved that any periodic function can be decomposed into small parts of sine and cosine waves or complex functions. Further in half century non periodic waves were discussed. The reconstruction and decomposition of the signals into complex exponential function into different frequencies are following:

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x)\ e^{-2\pi i x \xi}\, dx, \quad \text{for any real value of } \xi \qquad (2.4)$$

$$f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi)\ e^{2\pi i \xi x}\, d\xi, \quad \text{for any real value of x.} \qquad (2.5)$$

In the above equations $\hat{f}$ shows the Fourier transform of signal x in time domain while *f(x)* shows the inverse transform and x is in time domain while ξ in frequency domain (in hertz). We can do the computation of Fourier transform over all time. The scaling property of fourier transform states that if we have scaled version of

$$fs\,(x) = f\,(sx) \qquad (2.6)$$

then, it corresponds to $\hat{f}(\xi) = 1/|s|\ \ f(x)$ (2.7)

We can observe from the above equations that if we decrease the time spread then FT is dilated. If we increase the time spread FT is concentrated. It mean that if we get the time localization we lost frequency localization. Now if we project the signals on complex exponentials we get a good frequency localization. The no time localization is the main disadvantage of Fourier transform makes it unsuitable for many of the application.

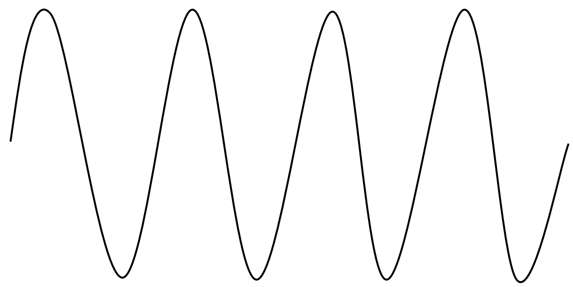


Figure 2.2 Fourier transform of sinusoid

2.6 Short-time fourier transform

Short time Fourier transform (STFT) is Fourier related term states that Fourier transform of longer signal over time is converted into short segments. The Fourier transform (FT) of each segment is computed in order to see how the frequency content changes for each segment of time. This basic idea was introduced by Gabor in 1946. Further in 1977 the interpretation of STFT with filter banks by Allen. Now how we actually apply STFT to signal. To compute STFT we will pass the original signal by a translating window of small size and then apply Fourier transform to the signal, the result can be mathematically written as:

$$\mathbf{STFT}\{x(t)\}(\tau,\omega) \equiv X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)w(t-\tau)e^{-j\omega t}\,dt \tag{2.8}$$

Here *w(t)* is window function mostly known is Hann or Gaussian window centered at zero, *x(t)* is the signal to be analyzed. The above equation is used by sliding a window in time domain on the original signal or passing by a filter in frequency domain. The window function will be of same size in the whole process.

Depending on the time localization of the signal we can use any width of window which is more suitable to our application but we get poor frequency localization

2.7 Wavelet transform

In the above two methods we find the limitation of these two methods now keeping in mind the limitation of Fourier transform (FT) and short time Fourier transform (STFT) which is poor time localization and frequency localization represented in exponential form. Grossman and Morlet [5] in 1984 formulated continues wavelet transform which decomposes the original signal into translated and dilated version of the original signal. The main advantage of the wavelet transform (WT) over other methods is multiresolution analysis (MRA), having different frequencies in different scales and time. Most of the signals have small frequency for long time duration and large frequency for small duration of time. Wavelet transform do the same work for all signals analysis.

If we consider *x is member of* $\boldsymbol{L^2(R)}$ and for analysis we use mother wavelet ψ. We can write the signal in scaled and translated version. So we can write the equation in the following form:

$$[W_\psi f](a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} \overline{\psi\left(\frac{x-b}{a}\right)} f(x)dx \tag{2.9}$$

By looking to the above equation we conclude that wavelet transform is the convolution of the original signal and the wavelet to be derived.

2.8 Comparative visualization

In this topic we will visually see the graphs of the three transforms which we have discussed above. The graph of fourier transform is given in the figure (2.2) which shows very good frequency localization but non existing time localization.

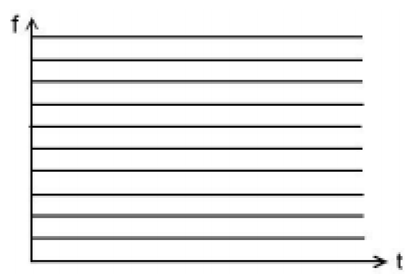


Figure 2.3 Time-frequency localization of Fourier transform

In figure 2.3 Short time Fourier transform time frequency localization is shown. It is based on Heisenberg uncertainty principle which shows time frequency localization is limited. One can do time frequency localization up to some extent. This lead to the fact that time-frequency boxes (rectangles) will be of same length. This localization is better up to some extent.

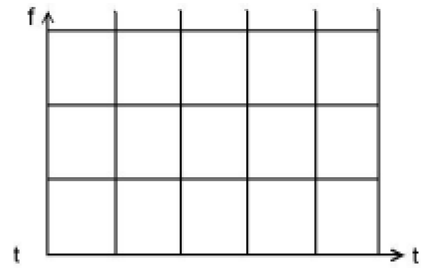


Figure 2.4 Time-frequency localization of short-time fourier transform

In the figure 2.4 a much better time-frequency localization is shown. For low frequencies large time interval is needed and for large frequencies small time interval is used. Because of this particular approach wavelet transform is used and is more suitable for most of the signals.

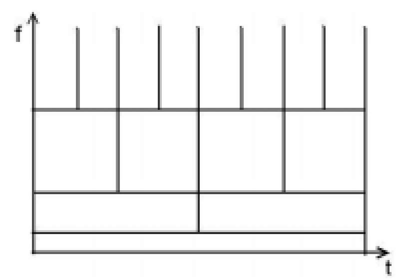


Figure 2.5 Time-frequency localization of wavelet transform

2.9 Perfect reconstruction of the filter bank

Wavelets analysis and synthesis shows that how wavelets decomposes and reconstructs a signal or image. Wavelets decomposes a 1-dimensional (1D) signal into two parts high frequency component and low frequency component. Passing the signal low pass filter (LPF) retains low frequency component and rejects high frequency component while retains high frequency component by passing it into high pass filter (HPF). The result contains two parts a high frequency component and a low frequency component.

The following figure shows how the analysis and reconstruction of filter bank works.

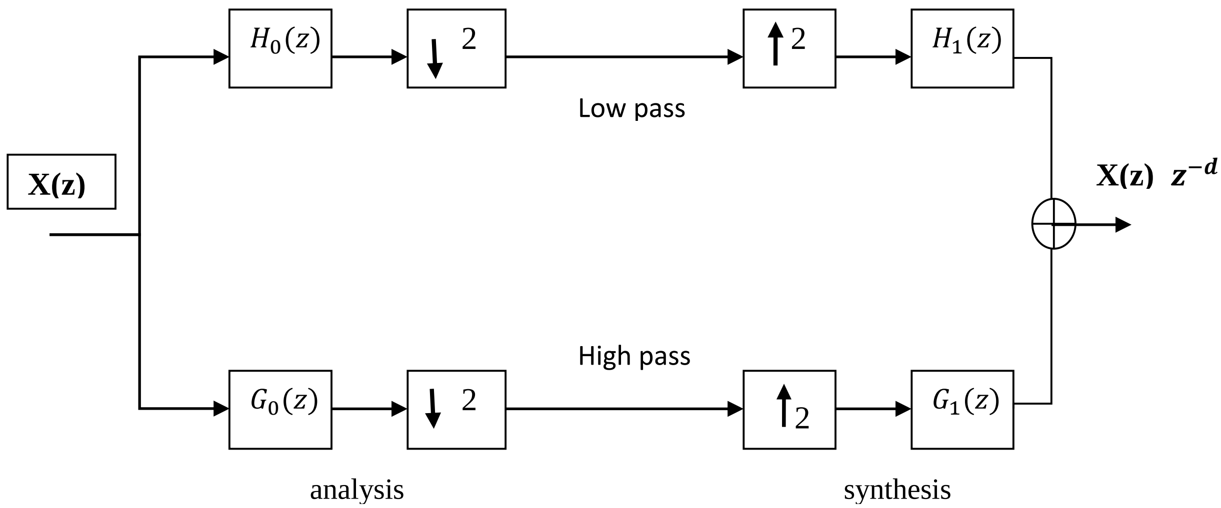


Figure 2.6 One dimensional, one level perfect reconstruction filter bank

in the above diagram the input signal X (z) is passed from two filters i.e. $H_0(z)$ and $G_0(z)$ which are low pass and high pass filter respectively. The signal is further passed from down sampling. These down sampling will give us sparse representation of the signal. This is the first level decomposition. If we want to go into further level of decomposition we can up to N level. We call every level a sub-band. We can decompose it into N sub-bands. Here in this case we decomposed the signal into one level decomposition. The signal is further passed from up sampling and passed from their respective filters i.e. $H_1(z)$ and$G_1(z)$. These signals are further added with each other to get the output signal X (z) with d sec delay.

The analysis introduces aliasing in the signal even though the sampling rate is preserved. The analysis part induces phase and magnitude distortion in the signal. The reconstruction part or the synthesis filter aims to reduce the aliasing and distortion in the signal. The following relation is used for the perfect reconstruction of the signal.

$$G_0(z)\,H_0(z) \;+\; G_1(z)\,H_1(z) \;=\; 2 \qquad (2.10)$$

$$G_0(z)\,H_0(-z) \;+\; G_1(z)\,H_1(-z) = 0 \qquad (2.11)$$

2.10 Classification of wavelets

There is different relation for analysis and reconstruction of orthogonal and bi-orthogonal [13] wavelet basis. Depending on the different properties of theses basis they are classified into two main categories.

a) Orthogonal basis
b) Bi-orthogonal basis

2.11 Features of orthogonal wavelet filter banks

Orthogonal basis are non-symmetric and same in length. They have real coefficient values. In this case following relation between analysis and synthesis exist.

$$H_1(z) = H_0(z^{-1}) \tag{2.12}$$

$$G_1(z) = G_0(z^{-1}) \tag{2.13}$$

Following filters exist between high and low pass filters with N being filters length

$$G_o(z) = -z^{-N} H(-z^{-1}) \tag{2.14}$$

So we can say that we can easily reconstructs the signal with only one filter equation. The regular structure of the filters makes it easier to implement.

2.12 Features of bi-orthogonal wavelet filter banks

The coefficient values of bi-Orthogonal filters are either real numbers or integers. The following condition must be satisfied:

$$G_0(z) = H_1(-z) \tag{2.15}$$

$$G_1(z) = -H_0(z^{-1}) \tag{2.16}$$

There are two filters that represents orthogonal filter bank $H_0(z)$ and $H_1(z)$. Filters with linear phase are needed for orthogonal basis. The values of low pass filter will be symmetric wile that for the other filter the coefficient values may be symmetric or may be not symmetric.

2.13 Wavelets transform

There are two kind of wavelet transform. Continues wavelet transform and DWT. Both of them are discussed following with details.

2.13.1 Continues wavelet transform

In Fourier transform, we represent Fourier analysis mathematically as:

$$F(w) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\, dt \quad (2.17)$$

In the above equation the exponential contains the superposition of real and complex values. Now in the continues wavelet transform is mathematically written as:

$$C\,(Scale\,, position) = \int_{-\infty}^{\infty} f(t)\; \psi\,(Scale\,, position\,, t)\, dt \quad (2.18)$$

Continues wavelet transform is the result of multiplying a signal f (t) with mother wavelet$\psi(t)$. Different scales of this mother wavelet can be obtained by the basic formula of the mother wavelet bases. The projection of the signal with mother wavelet can give us different scales. The formula is given below:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\, \psi \left[\frac{t-b}{a}\right]\; ; \; a, b \in R^1 \text{ and } a > 0 \quad (2.19)$$

Here in the above equation ‘a’ is scaling factor while ‘b’ is translation factor. The resulting $\psi_{a,b}(t)$ is the translated and scaled version of the mother wavelet.

The basic formula for the continues wavelet transform is:

$$W_f(a, b) = \int_{-\infty}^{\infty} x(t)\; \psi_{a,b}(t)\; dt \quad (2.20)$$

Inverse wavelet transform

The decomposed structure of the continues wavelet transform will be reconstructed by inverse wavelet transform. The inverse wavelet transform is mathematically defined as:

$$x(t) = \frac{1}{C} \int_{0}^{\infty} \int_{-\infty}^{\infty} W_f(a, b)\; \psi_{a,b}(t)\; db\, \frac{da}{a^2} \quad (2.21)$$

$$\text{Where } C = \int_{-\infty}^{\infty} \frac{|\psi|^2}{\omega}\, d\omega < \infty \quad (2.22)$$

In the above equation two condition must be satisfied for wavelet transform basis. First, to avoid singularity in eq. 2.22 the integral must be finite with a mean value of zero. Mathematically we can write as follows:

$$\int_{-\infty}^{\infty} \psi(t)\, dt = 0 \qquad (2.23)$$

This property is known as admissibility property.

The second main requirement is the mother wavelet will have finite energy.

$$\int_{-\infty}^{\infty} |\psi(t)|^2\, dt = \infty \qquad (2.24)$$

2.13.2 Discrete wavelet transform

1D DWT

The analysis of 1D orthogonal wavelet transform for any signal that belongs to $L^2(R)$ is mathematically represented as:

$$a_{j,k} = \int x(t) 2^{j/2}\, \phi\left(2^j t - k\right) dt \qquad b_{j,k} = \int x(t) 2^{j/2}\, \psi\left(2^j t - k\right) dt \qquad (2.25)$$

The synthesis equation for inverse discrete wavelet transform (IDWT) is given below.

$$x(t) = 2^{N/2} \sum_k a_{N,k}\, \phi\left(2^N t - k\right) + \sum_{j=N}^{M-1} 2^{j/2} \sum_k b_{j,k}\, \psi\left(2^j t - k\right) \qquad (2.26)$$

$\phi(t)$ Represents scaling function of the orthogonal wavelet while $a_{j,k}$ represents coefficient value.

$\psi(t)$ Represents wavelet function, $b_{j,k}$ represents coefficient value for translation.

The analysis equation for the bi-orthogonal basis are

$$\tilde{a}_{j,k} = \int x(t) 2^{j/2}\, \tilde{\phi}\left(2^j t - k\right) dt \qquad \tilde{b}_{j,k} = \int x(t) 2^{j/2}\, \tilde{\psi}\left(2^j t - k\right) dt \qquad (2.27)$$

The synthesis equation for the bi-orthogonal basis are

$$X(t) = 2^{N/2} \sum_k \tilde{a}_{N,k}\, \phi\left(2^N t - k\right) + \sum_{j=N}^{M-1} 2^{j/2} \sum_k \tilde{b}_{j,k}\, \psi\left(2^j t - k\right) \qquad (2.28)$$

In bi-orthogonal basis $\phi(t)$ represents the scaling function, $a_{j,k}$ represents coefficient value for the input signal or image X (t). $\psi(t)$ Represents wavelet function, $b_{j,k}$ represents coefficient value for wavelet basis.

In DWT 'j' is the scaling function and 'k' is the translation of the wavelet. N is the number of levels which we want to decompose and reconstructs the one dimensional (1D) signal.

2D Discrete Wavelet Transform

As we know that images are the 2D form of 1-D signals. 2D signals are formed by a series of 1D signals. The 2D data is first passed through one dimensional wavelet transform along rows and then it is passed through one dimensional (1D) wavelet transform along columns during analysis section of wavelet transform. The following figure shows DWT and IDWT.

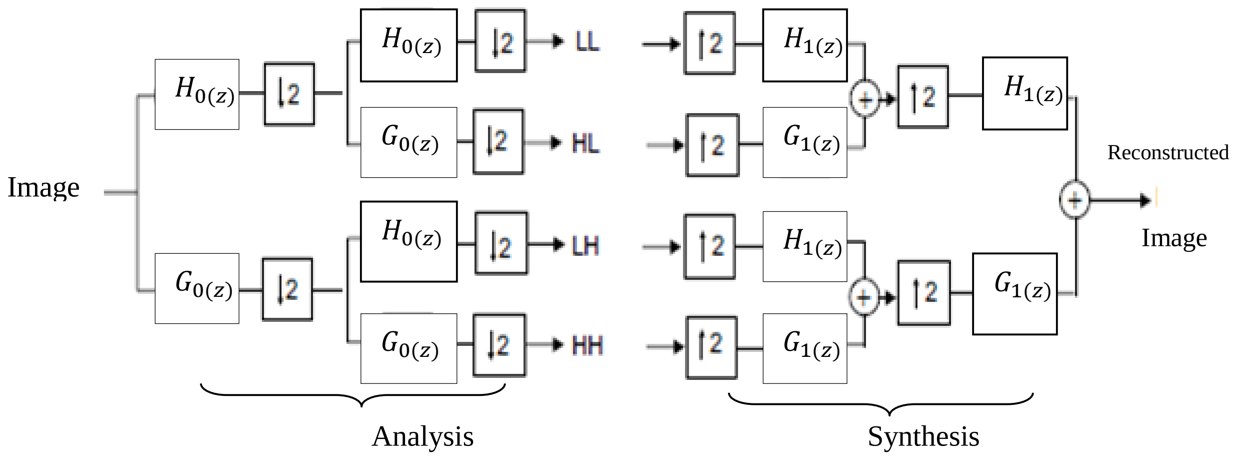


Figure 2.7 One level filter bank for computation of 2-D DWT.

When an image is passed through 2D basis function, transform coefficients are received. The result of multiplication of two one dimensional (1D) basis function are achieved. So far we have four section received in equation 2.29

$$\phi\,(u,v) = \phi(u)\,\phi(v)$$
$$\psi_1\,(u,v) = \psi(u)\,\phi(v)$$
$$\psi_2\,(u,v) = \phi(u)\,\psi(v)$$
$$\psi_3\,(u,v) = \psi(u)\,\psi(v) \tag{2.29}$$

$\phi\,(u,v)$ Scaling function for all images. Wavelet function for images are $\psi_1\,(u,v), \psi_2\,(u,v)$ and $\psi_3\,(u,v)$. After passing through these basis functions coefficients are achieved. As a result, four sub-band are achieved which are as follows:

1. LL sub-band is course approximation
2. LH sub-band contains vertical details
3. HL sub-band contains horizontal details
4. HH sub-band contains diagonal details

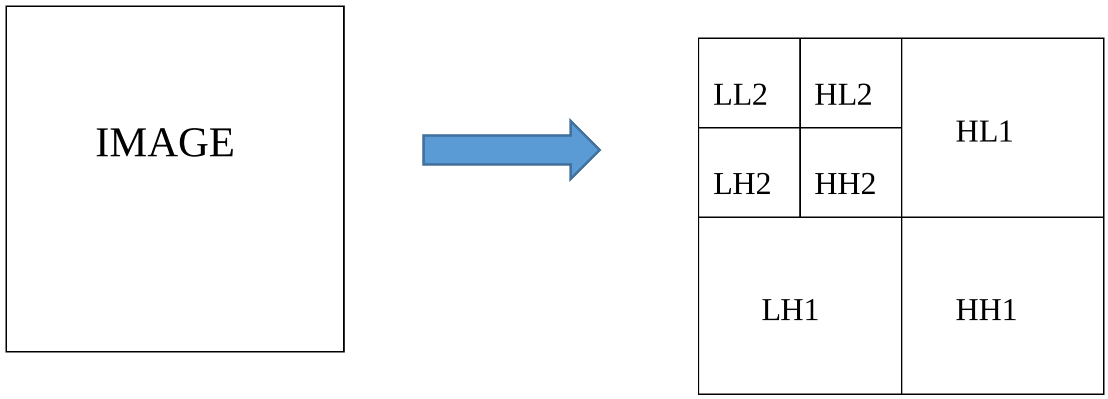


Figure 2.8. Output of 2-D decomposition up to one level.

Wavelet transform De-noising techniques

Chapter 3 Statistical wavelet thresholding techniques

3 Introduction

Various research work has been done on wavelet image de-noising and thresholding shows that wavelet transform is very best for image de-noising and compression. The basic problem in wavelet image de-noising and compression is to find the optimal threshold value. Because of the compaction property of wavelet which shows small number of large coefficient and large number of small coefficients. The small coefficient values also contain noisy coefficients values. The purpose is to remove the noisy coefficient values to large extent without the loss of useful information. To remove these small noisy coefficient values a threshold values is calculated. All the values below the threshold value are set to zero are above the threshold are kept. For removing noisy small coefficient values a lot of methods were observed. All these methods are discussed below along with our statistical method used for image de-noising.

Image de-noising is a three step process which is very simply explained by the figure 3.1. The first step consist of wavelet decomposition, the second step is wavelet thresholding, while the third step is wavelet reconstruction.

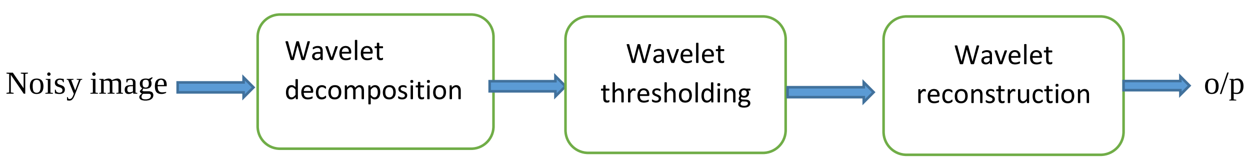


Figure 3.1 a three step de-noising process.

In figure 3.1 wavelet decomposition is to decompose the input noisy image. We can decompose the image up to ‘L’ levels. The wavelet decomposition gives us details and approximation of the input image. The details shows us vertical, horizontal and diagonal details. The approximation details shows the original image coefficients which should not be thresholded during thresholding section. The details are thresholding during the thresholded section and then reconstruction of the image wavelets occurs which reconstructs the image and gives us the estimated image or de-noised image.

3.1 Estimation and de-noising:

De-noising is actually the process of estimation. We can define the problem set of de-noising in the following expression. Suppose that $Y_{j,k}$ is the wavelet coefficient of the noisy signal ‘B’ of length ‘L’ and $X_{j,k}$ is the coefficient of the original image without noise. We write the equation in the following manner.

$$Y_{j,k} = X_{j,k} + N_{j,k} \tag{3.1}$$

$N_{j,k}$ is the noise with Gaussian distribution N(0, sigma). Our goal is find the output image x to estimate the image and suppress the noise for every level and position. We will minimize the risk of difference between input image and output image.

For de-noising and compression different techniques have been implemented. Wavelet thresholding techniques are being wieldy used for image de-noising. There are universal thresholding (UT) invented by [12] Dohono and Johnstone. After it Visu shrink method was implemented for thresholding. After this method an adaptive technique for image de-noising was implemented which is SURE shrink. All these methods are discussed below in the details.

3.2 Threshold approaches

Before we represent our work it should be noted that some thresholding methods exist for finding the value of threshold (lambda). There are two kind of methods *soft thresholding* and *hard thresholding.*

Hard thresholding kills the coefficient value that is smaller than the threshold value and leaves the coefficients that are larger than threshold value while Soft thresholding shrinks the coefficient values towards threshold. Mathematically these techniques can represented as:

The ***hard thresholding*** operator is defined as

$$D(U,\lambda) \quad = \begin{cases} U \text{ for all } |U| > \lambda \} \\ 0 \text{ otherwise } \quad \} \end{cases} \tag{3.2}$$

The ***soft thresholding*** operator on the other hand is defined as

$$D(U,\lambda) = \operatorname{sgn}(U) * \max(0, |U| - \lambda) \tag{3.3}$$

3.3 Universal thresholding:

In the wavelet de-noising literature the universal thresholding is the most widely used one. It is globally approached and can be formulated as follows:

$$\lambda_T = \sigma \sqrt{2logN} \tag{3.4}$$

Where N is the size of the image and σ is noise variance. The λ_T must above the max level but not too large. Also too much large coefficients may not be averted. Also with increase in N Length the threshold also increases due to Gaussian distribution.

Universal thresholding does not require prior information exactly like the Bayesian thresholding. For smooth data like dohono it may be applied easily and conveniently.

When the size of the input signal is so large that it approaches to infinity the universal thresholding is the best candidate in that scenario. Also it is a good approach for statistical smoothness whose asymptotic behavior is better the MSE.

This approach is too much fast and easy. Its implementation is straight forward, however when implemented on an image it produce a de-noised image which lost enough information.

3.4 Statistical thresholding in wavelets:

3.4.1 Method 1:

In this method we find the mean of each detail sub-band ‘μ’. The σ_y is the variance of the degraded image which can be find by robust median estimator

$$\sigma_y^{\ 2} = [\text{median}(|\text{each sub} - \text{band}|)/0.6745] \qquad (3.7)$$

The noisy coefficients are very small and the signal coefficient are very large contains the useful information of the image. After the decomposition of the image to N level. The coefficients of the detail sub bands are stored an array. The values that are greater than $2\sigma_y,\ 3\sigma_y$ are dropped and the other values are kept. i.e

$$\text{y} > \ 2\sigma_y,\ 3\sigma_y; \text{x} = 2\sigma_y \qquad (3.8)$$

$$\text{Else y} = \text{y}$$

Finding the noise variance σ_n and threshold value, finally add the value with mean 'μ'

$$t = \sigma_n{}^2/\sigma_s{}^2 \qquad (3.9)$$

3.4.2 Method 2:

The Statistical Thresholding method is effective for images including Gaussian noise. The observation model is expressed as follows:

$$\text{Y = X + N} \qquad (3.10)$$

Here Y is the wavelet transform of the degraded image, X is the wavelet transform of the original image, and V denotes the wavelet transform of the noise components following the Gaussian distribution N (0, $\sigma_n{}^2$). Here, since X and V are mutually independent, the variances $\sigma_y{}^2$, $\sigma_x{}^2$ and $\sigma_n{}^2$ of y, x and n are given by

$$\sigma_y{}^2 = \sigma_x{}^2 + \sigma_n{}^2 \qquad (3.11)$$

It has been shown that the noise variance can be estimated from the first decomposition level diagonal sub-band HH1 by the robust and accurate median estimator [4].

$$\sigma_n^{\ 2} = [\text{median (|each sub-band|)/0.6745}] \qquad (3.12)$$

The variance of the sub-band of degraded image can be estimated as:

$$\sigma_y{}^2 = 1/M \sum_{m=1}^{M} (A\,m)^2 \qquad (3.13)$$

where Am are the wavelet coefficients of sub-band under consideration, M is the total number of wavelet coefficient in that sub-band. The statistical thresholding technique performs soft thresholding, with adaptive data driven, sub-band and level dependent near optimal threshold given by:

$$T = \begin{cases} \dfrac{\sigma_n{}^2}{\sigma_x{}^2} & if\ \sigma_n{}^2 > \sigma_y{}^2 \\ \max(Am) & otherwise \end{cases} \tag{3.14}$$

Results and Simulations

Chapter 4 Results and simulations

4 Introduction

This chapter discusses about the implementation of the hard thresholding, soft thresholding, visu shrink, Statistical method 1 and statistical method 2. The results of all the methods is compared with statistical method 1 and statistical method 2 algorithm.

It is very essential to compare the input image with the output image. We take the input image and we add different variances of noise with it. The noisy image is subjected to wavelet transform decomposition after passing through this it is passed to wavelet thresholding and after reconstructing transformation we get a de-noised image. Now to find the performance of every method we take some parameters to check which thresholding methods out performs the other thresholding methods. PSNR is used to measure the distortion.

$$\text{PSNR} = 10 \ \log 10_{10} \ \left(\frac{\left(\max\left(f(m,n)\right)\right)^2}{\text{MSE}} \right) \tag{4.1}$$

In the above equation $f(m, n)$ shows the input image. We are dealing here with gray scale images. So we have

$$\left(\max\left(f(m, n)\right)\right) = 255 \tag{4.2}$$

The other parameter used is MSE, it checks the error between original image and output image or the de-noised image. We can describe it mathematically as the difference between original image and estimated image.

$$\text{MSE} = \sum_{MN} \frac{f(m,n) - \tilde{f}(m,n)}{M \times N} \tag{4.3}$$

In the above equation $\tilde{f}(m, n)$ is the de-noised image or the estimated image or reconstructed image.

The other parameter we have used for psycho-visual comparison is structural similarity index (SSIM) which shows the image quality comparison between the original image and the estimated image. This method has recently been developed for images quality comparison. We can mathematically write the formula of SSIM is the measure between two images X and Y of common size N x N is:

$$SSIM(X,Y) = \frac{(2\mu_x\mu_y + C1)\ (2\sigma_{xy} + C2)}{(\mu_x + \mu_y\ + C1)\ (\sigma_x + \sigma_y + C2)} \qquad (4.4)$$

Here

μ_x is the average of *x*

μ_y is the average of *y*

σ_x is the variance of *x*

σ_y is the variance of *y*

σ_{xy} is the covariance of *xy*

C_1 and C_2 are constants

Mean opinion score (MOS) is used to check the human view about the visual quality of images. Every user will look to the image and shows his view about the image from unacceptable to excellent in form of numbering form 1 (worst) to 5 (bad). The table is shown below.

Table 4.1 Mean opinion score (MOS)

MOS	QUALITY
1	Unacceptable
2	Poor
3	Fair
4	Good
5	Excellent

The existing statistical techniques uses statistical parameters for threshold value selection. The existing techniques uses different wavelet bases for that bi-orthogonal wavelet is version 6.8 has much better results that other wavelet bases. So in our simulations we have used wavelet bases bior6.8.

4.1 Test images

For evaluation of our methods five natural test images are used. These five images are shown in the following figure.

(a) Lena

(b) Barbara

(c) House

(d) MCS Library

(e) Cameraman

Figure 4.1 five natural test images

4.2 Visual quality:

MOS is used to check the visual performance of all the thresholding methods. We need to show the images of every techniques. Following are the figures of five test images. These images show the performance of all thresholding techniques.

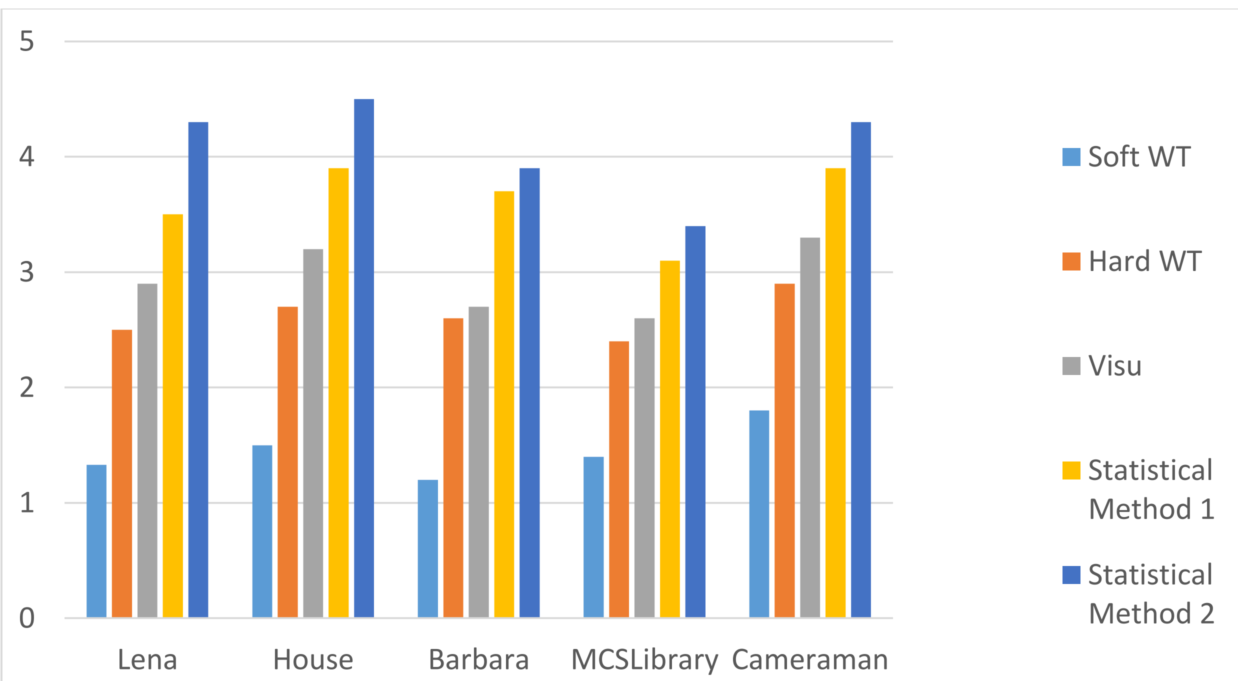
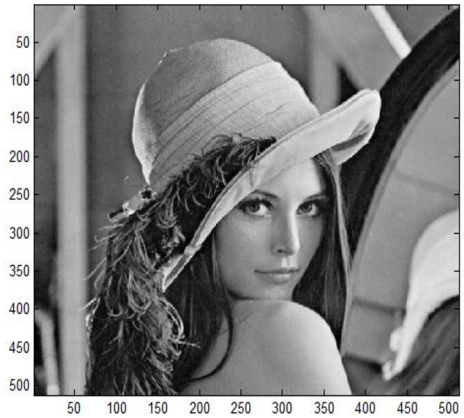
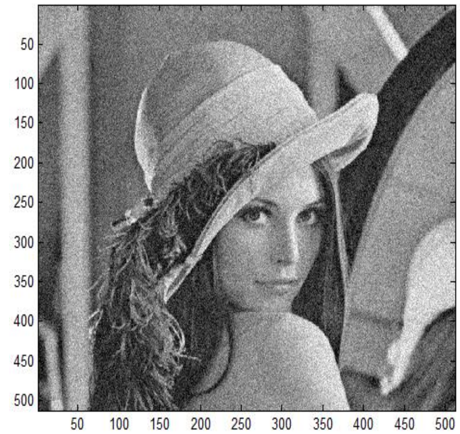


Figure 4.2 MOS for five images on five thresholding techniques.

4.2.1 Lena



(a)

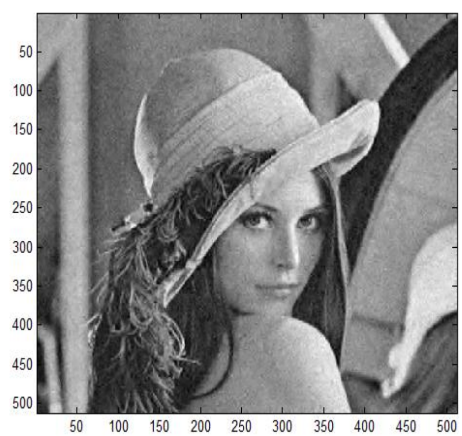


(b)

(c)

(d)

(e)



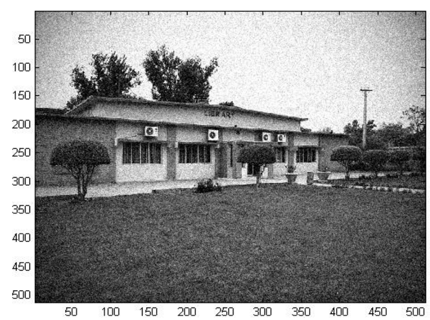
(f)

(g)

Figure 4.3 (a) A clean Lena image (512 x 512) (b) noisy image with variance σ = 20 (c) De-noised image by soft thresholding (d) De-noised image by hard thresholding (e) De-noised image by Visu Shrink (f) De-noised image by Statistical method 1 (g) De-noised image by statistical method 2

4.2.2 MCS Library

(a)



(b)

(c)

(d)

(e)

(f)

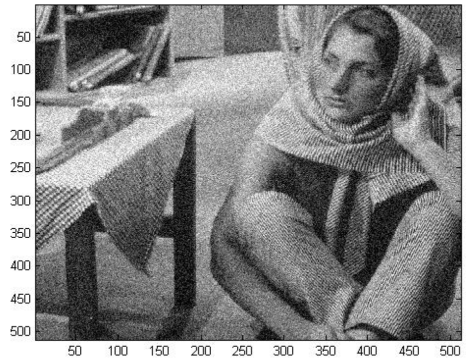
(g)

Figure 4.4 shows an image of MCS library 512 x 512 de-noised by various methods with noise variance of 20.

In the above figures an image of our college library is taken as a test image. in the above figure section (a) shows an original image and (b) shows noisy image with a variance of sigma equal to 20 section (c) shows a de-noised image by soft thresholding section (d) shows a de-noised image by hard thresholding section (e) shows a de-noised image by visu shrink method section (f) shows de-noised image by statistical method 1 section (f) shows de-noised image by statistical method 2

4.2.3 Barbara:

(a)

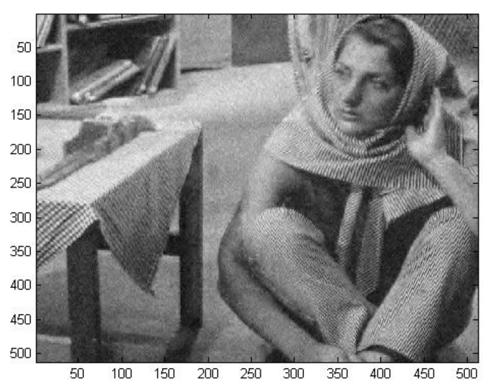


(b)

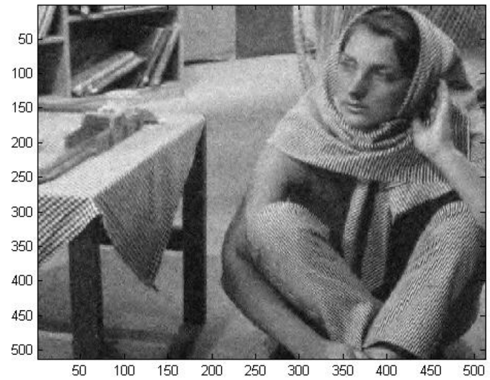
(c)

(d)

(e)



(f)

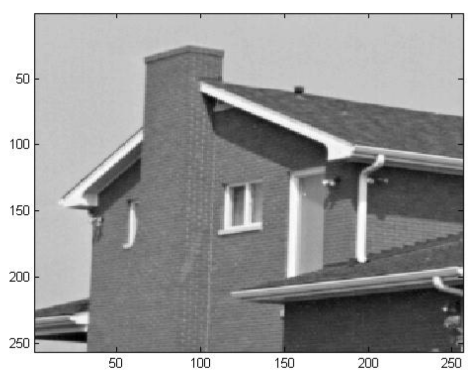


(g)

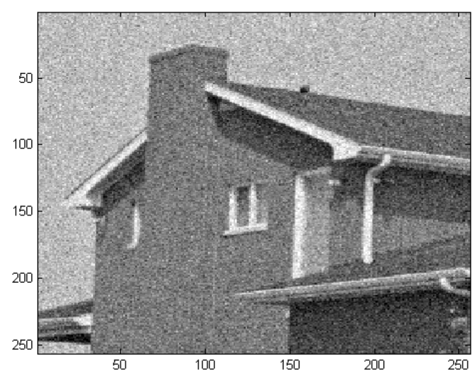
Figure 4.5 shows an image of Barbara 512 x 512 de-noised by various methods with noise variance of 25.

In the above figures Barbara is taken as a test image. in the above figure section (a) shows an original image and (b) shows noisy image with a variance of sigma equal to 25 section (c) shows a de-noised image by soft thresholding section (d) shows a de-noised image by hard thresholding section (e) shows a de-noised image by visu shrink method section (f) shows de-noised image by statistical method 1 section (f) shows de-noised image by statistical method 2

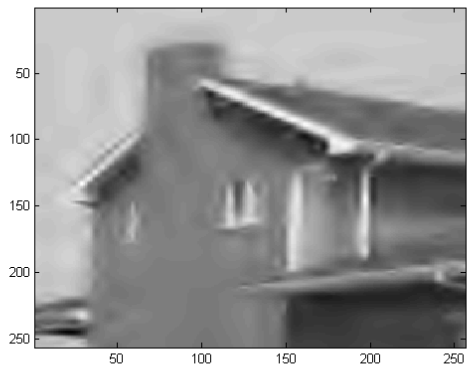
4.2.4 House:

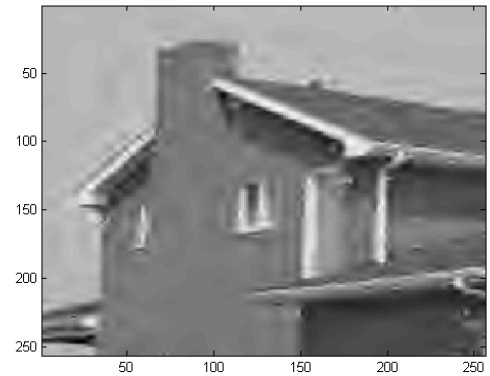


(a)

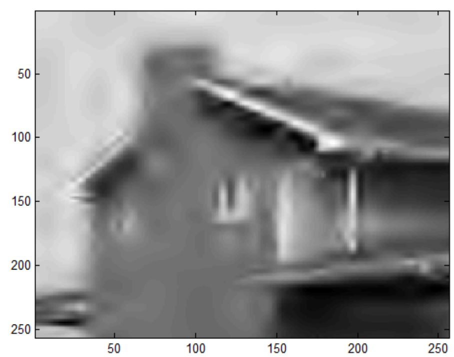


(b)

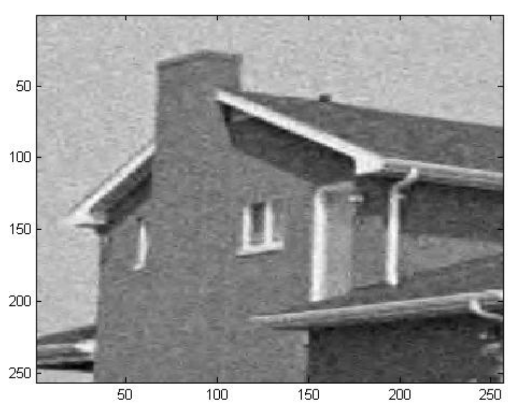




(c) (d)



(e) (f)



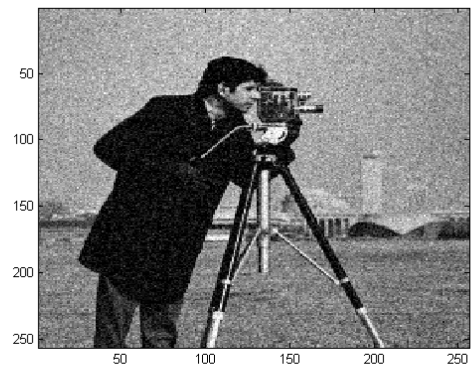
(g)

Figure 4.6 shows an image of House 512 x 512 de-noised by various methods with noise variance of 20.

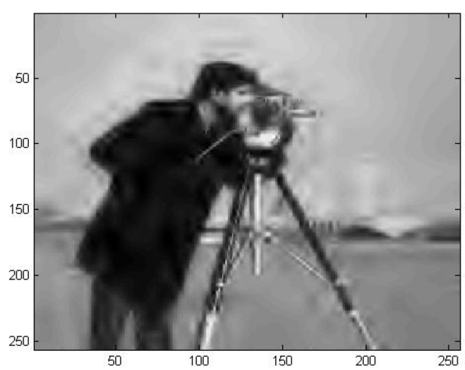
In the above figures House is taken as a test image. in the above figure section (a) shows an original image and (b) shows noisy image with a variance of sigma equal to 20 section (c) shows a de-noised image by soft thresholding section (d) shows a de-noised image by hard thresholding section (e) shows a de-noised image by visu shrink method section (f) shows de-noised image by statistical method 1 section (f) shows de-noised image by statistical method 2

4.2.5 Cameraman:

(a)



(b)



(c)

(d)

(e) (f)

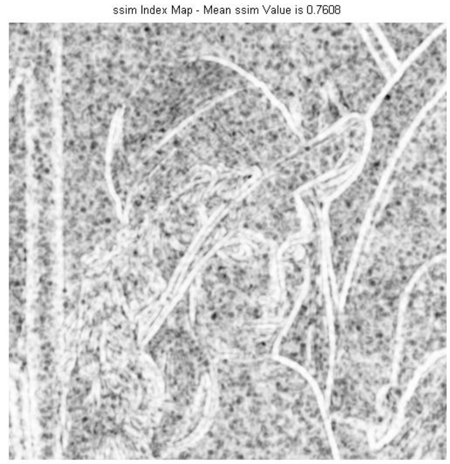
Figure 4.7 shows an image of Cameraman 512 x 512 de-noised by various methods with noise variance of 20.

In the above figures Cameraman is taken as a test image. in the above figure section (a) shows an original image and (b) shows noisy image with a variance of sigma equal to 20 section (c) shows a de-noised image by soft thresholding section (d) shows a de-noised image by hard thresholding section (e) shows a de-noised image by visu shrink method section (f) shows de-noised image by statistical method 1 section (f) shows de-noised image by statistical method 2

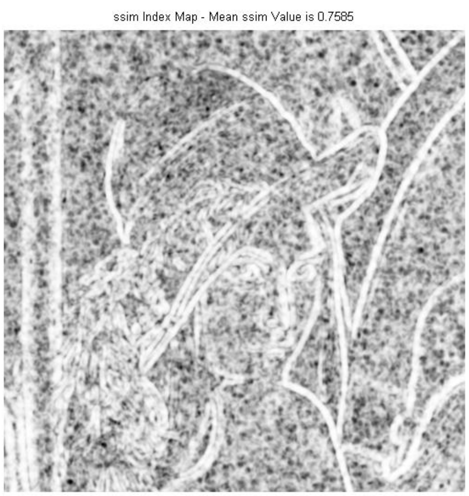
4.3 SSIM:

4.3.1 Lena:

(a) Soft Thresholding. SSIM value is 0.7195. (b) Hard Thresholding. SSIM value is 0.7619.



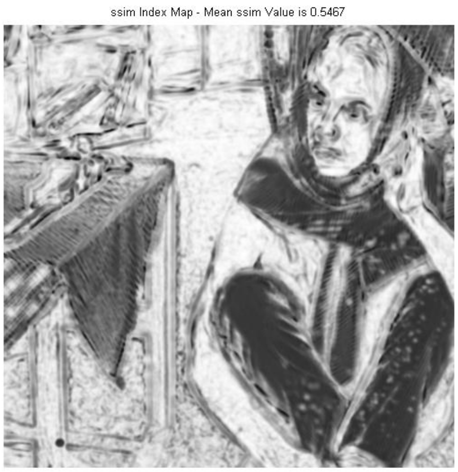
(c) Visu Shrink SSIM value is 0.7644 (d) Statistical Method 1 SSIM value is 0.7608



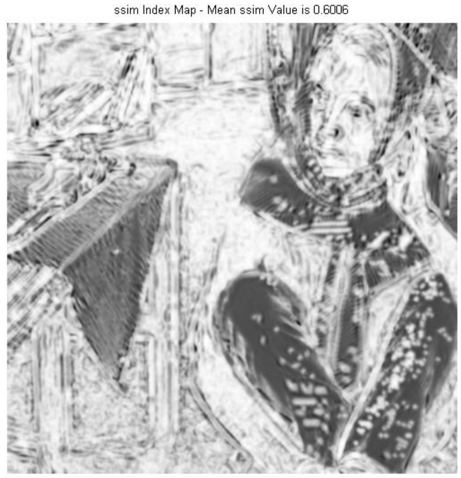
(e) Statistical Method 2 SSIM value is 0.7585.

Figure 4.8 SSIM map and values for the image Lena de-noised with noise variance sigma = 20

4.3.2 Barbara:



(a) Soft Thresholding SSIM value is 0.5467



(b) Hard Thresholding SSIM value is .6006

(c) Visu Shrink SSIM value is 0.6206

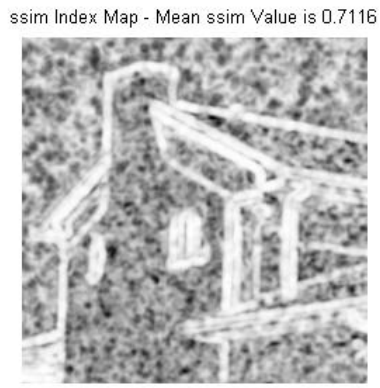
(d) Statistical Thresholding SSIM value is 0.6777

(e) Statistical method 2 SSIM value is 0.6850

Figure 4.9 SSIM map and values for the image Barbara de-noised with noise variance sigma = 20

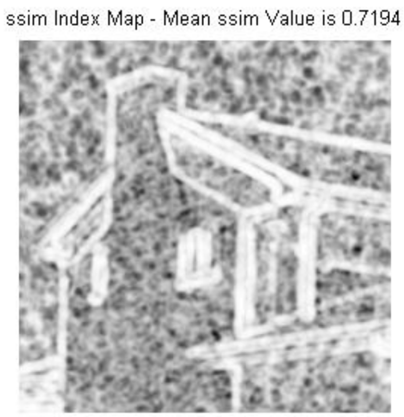
4.3.3 House:

(a) Soft thresholding SSIM value is 0.7258 (b) Hard thresholding SSIM value is 0.7563



(c) Visu Shrink SSIM value is 0.7573

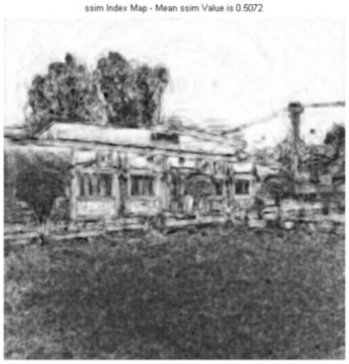
(d) Statistical method 1 SSIM value is 0.7116

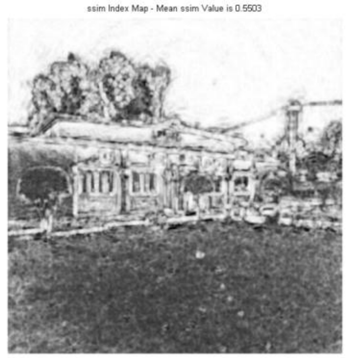


(e) Statistical method 2 SSIM value is 0.7194

Figure 4.10 SSIM map and values for the image House de-noised with noise variance sigma = 20

4.3.4 MCS Library:





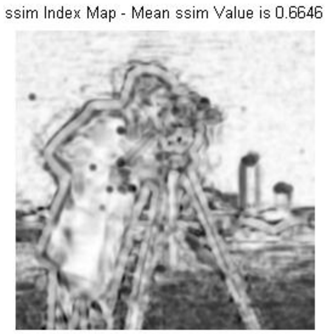
(a) Soft thresholding SSIM value is 0.5072 (b) Hard thresholding SSIM value is 0.5503

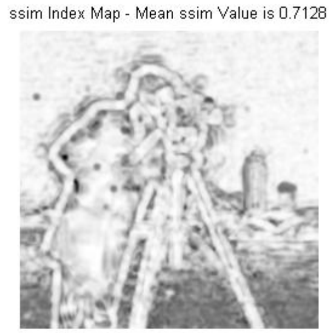
(c) Visu shrink SSIM value is 0.5627 (d) Statistical method SSIM value is 0.6142

(e) Statistical method 2 SSIM value is 0.6181

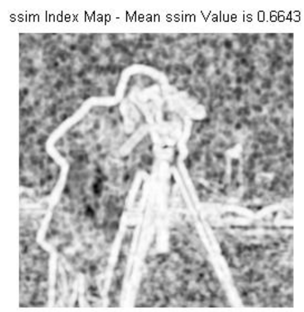
Figure 4.11 SSIM map and values for the image MCS Library de-noised with noise variance sigma=20

4.3.5 Cameraman:

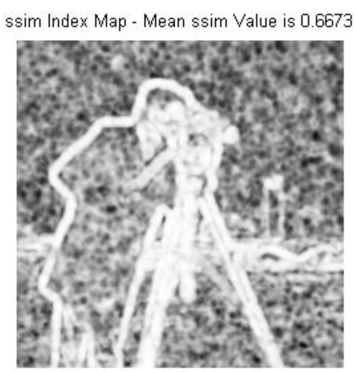




(a) Soft thresholding SSIM value is 0.6646 (b) Hard thresholding SSIM value is 0.7128



(c) Visu shrink SSIM value is 0.7182 (d) Statistical thresholding method 1 SSIM value 0.6643



(e) Statistical thresholding SSIM value is 0.6673

Figure 4.12 SSIM map and values for the image Lena de-noised with noise variance sigma = 20

4.4 Threshold values for each sub-band for image MCS library:

This section discusses the threshold values for each level for both the thresholding methods from level one to level five for each sub-band details i.e. for HH, HL and LH. The threshold value decreases as the decomposition level increases for image MCS library we have find the following threshold values.

Table 4.1 Threshold values for image MCS Library for each detail sub-band from level one to level five for statistical thresholding method 1.

Scale	HH	LH	HL
1(Finest)	88.5173	37.4301	44.2574
2	24.7179	14.7820	21.2625
3	10.9920	5.8531	9.2425
4	5.1413	2.3776	3.1231
5(coarsest)	2.7454	1.1358	1.6891

Table 4.2 threshold values for image MCS Library from level one to level five by statistical thresholding method 2

Scale	HH	HL	LH
1	240.7110	36.6272	44.4087
2	24.4503	14.7553	20.7946
3	10.8098	5.8190	9.2413
4	5.0202	2.3329	3.0196

5	2.6864	1.0784	1.7289

The histograms for these decomposition levels is given below for every level and every sub-band. The horizontal, vertical and diagonal details for every level is given below. The histograms for every level is also given below as we go in decomposition level the image is going more blurred because for reducing the size of the image by N/2 x N/2.

Figure 4.13 shows the original image of MCS Library

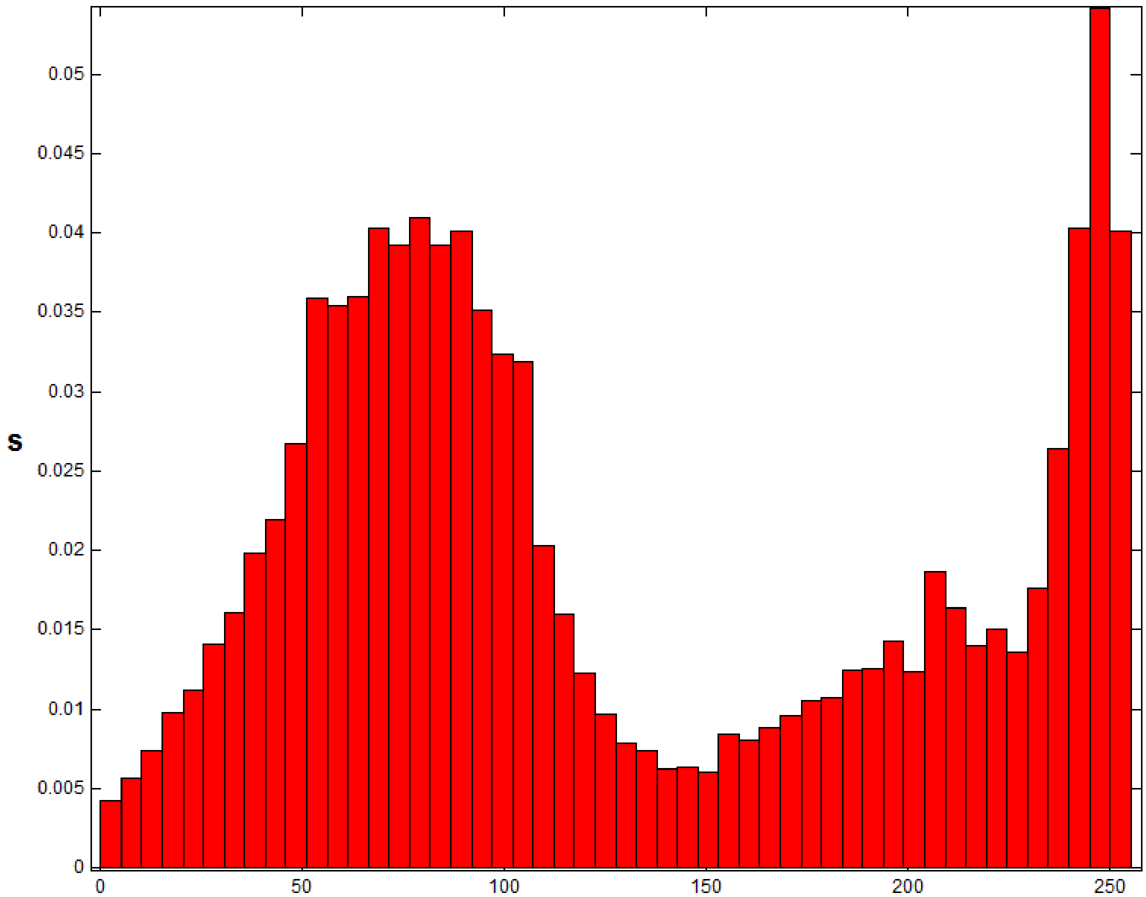


Figure 4.14 shows histogram of original image MCS Library

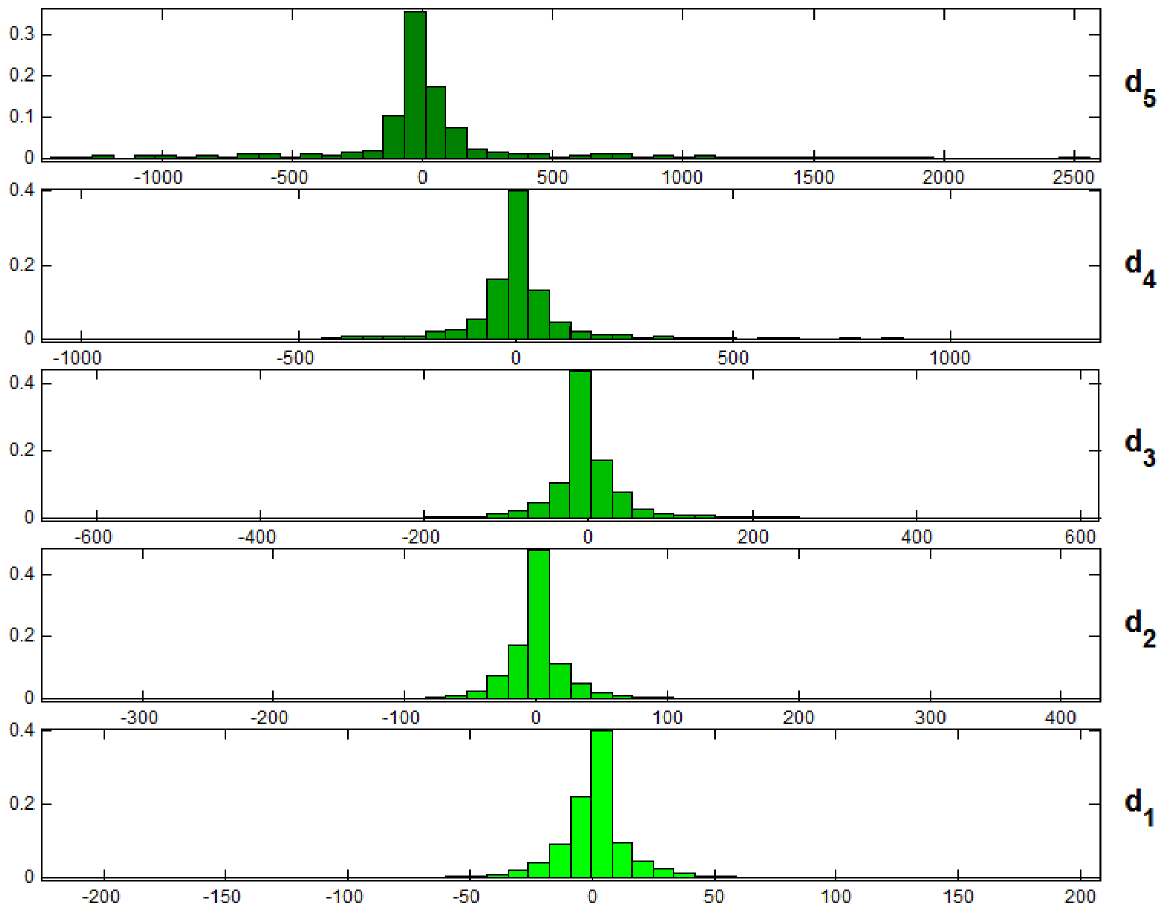


Figure 4.15 shows the histogram of horizontal details

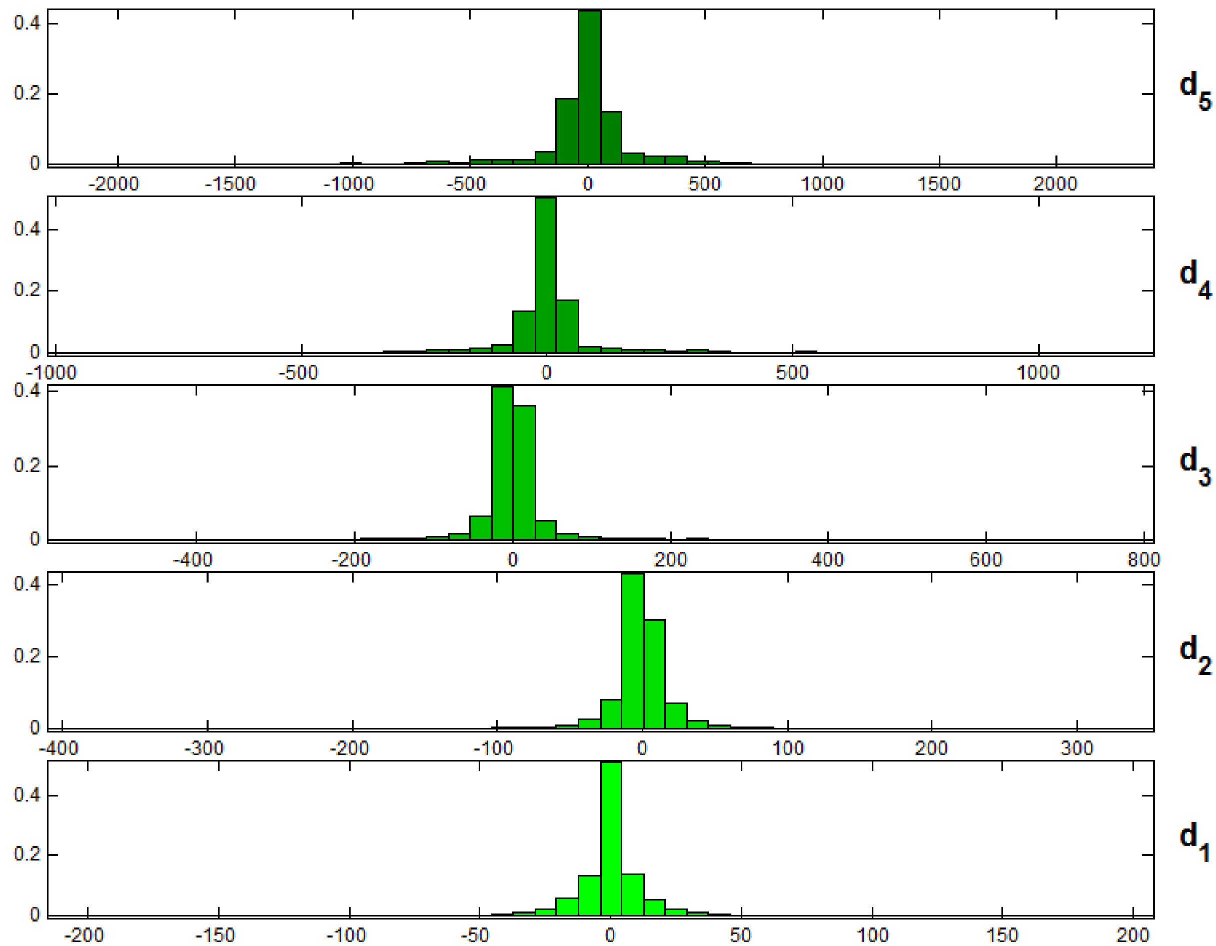


Figure shows 4.16 vertical details of image MCS Library from level 1 to 5

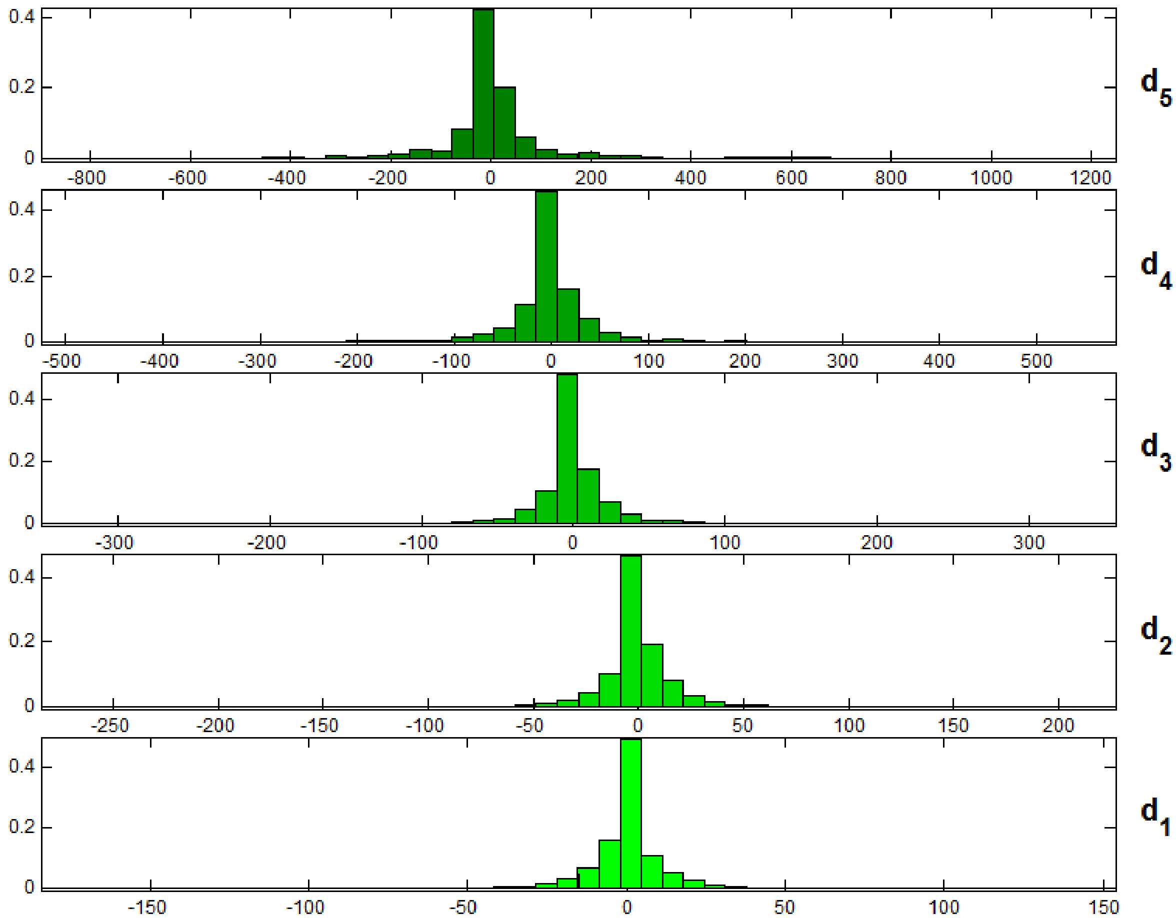


Figure 4.17 shows diagonal details histogram for image MCS Library from level 1 to 5

4.5 Graph of PSNR vs noise variance:

Figure shows the graph of PSNR values versus noise variance sigma = 20 for the image Lena. The image is passed from the five techniques i.e. Soft thresholding, hard thresholding, Visu shrink, statistical thresholding method 1 and statistical thresholding method 2. The graph of noisy image is shown which is improved to some extent by soft thresholding. The hard thresholding has higher values for PSNR than soft thresholding. Visu shrink has improved the PSNR values to more than hard thresholding. The statistical thresholding method 1 crossed the values of PSNR of all techniques. The statistical method 2 has top values than all techniques.

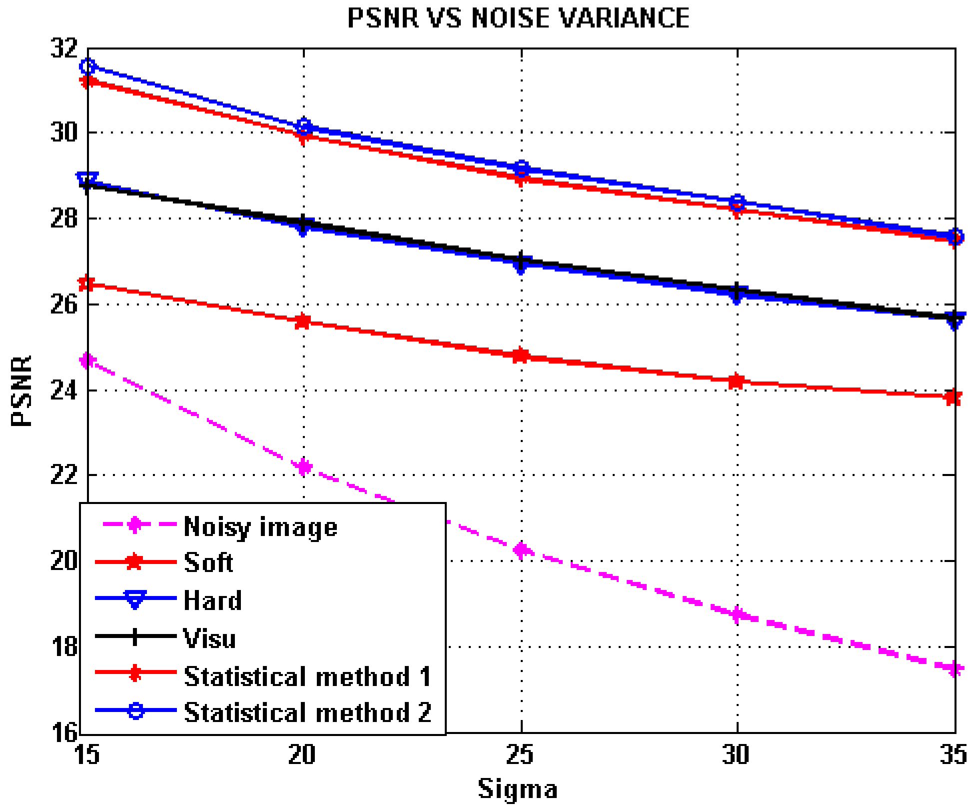


Figure 4.18 PSNR vs noise variance of image Lena for all thresholding techniques

4.6 Graph of MSE vs noise variance:

The objective is to minimize MSE versus different noise variance. The following graph shows MSE of noisy image. In order to compare results MSE of all techniques is compared with that of noisy image. The graph shows that how better each techniques performs.

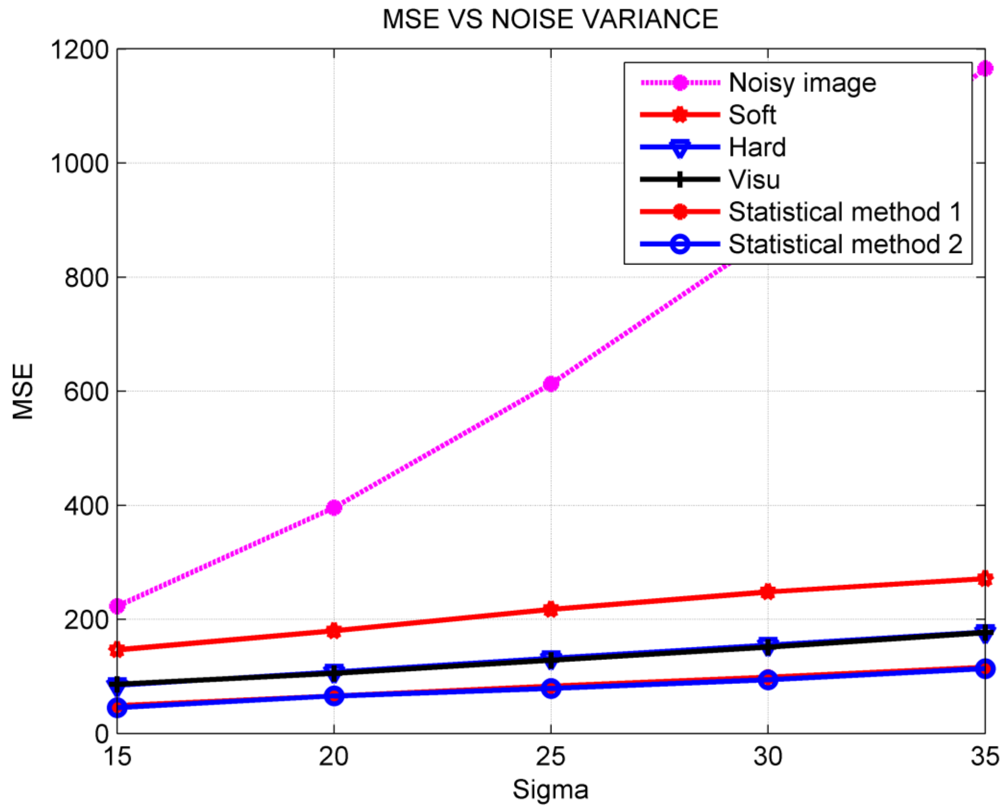


Figure 4.19 Plot of MSE vs noise variance for different techniques

4.7 PSNR and MSE values for all images

4.7.1 PSNR values for different images and different noise level:

The following table describes the PSNR of three images i.e Lena, Barbara and house. The noise variance is taken for sigma values of 15,20,25,30 and 35. These images are passed from different techniques i.e. the typical soft thresholding, hard thresholding, VISU shrink and statistical thresholding method 1 and statistical thresholding method 2.

First of all the input image Lena 512 x 512 is passed from wavelet decomposition and after hen it is passed by different thresholding techniques for noise variance sigma=20 the image is passed by Soft Thresholding, Hard thresholding, Visu shrink, statistical method 1 and statistical method 2. The visual quality shows that statistical method 1 and statistical method 2 are the best methods for thresholding.

Table 4.3 Shows the PSNR values for Images Lena, House, Barbara for noise variance σ = 15,20,25,30,35 by soft, hard, Visu shrink, Proposed method 1, proposed method 2.

Input image	Noise variance sigma	Noisy image	Soft thresholding	Hard thresholding	Visu shrink	Statsitaical method 1	Statistical method 2
Lena	15	24.67	26.46	28.87	28.77	31.22	31.57
House	15	24.67	26.25	28.77	29.09	29.86	30.97
Barbara	15	24.67	22.74	24.50	25.52	28.30	28.73
MCS library	15	24.96	24.97	24.97	24.39	26.79	21.72
Cameraman	15	24.90	23.36	25.90	26.30	27.86	28.76
Lena	20	22.19	25.57	27.79	27.90	29.94	30.13
House	20	22.19	25.31	27.72	27.83	28.77	29.49
Barbara	20	22.19	22.18	23.51	24.13	26.92	27.25
MCS library	20	22.54	22.53	23.02	23.37	25.47	25.79
Cameraman	20	22.40	22.34	24.74	25.13	26.58	27.12
Lena	25	20.25	24.75	26.93	27.04	28.93	29.16
House	25	20.25	24.50	26.80	26.90	27.87	28.38
Barbara	25	20.25	21.73	22.97	23.29	25.86	26.08
MCS library	25	20.67	21.19	22.51	22.64	24.46	24.74
Cameraman	25	20.34	21.67	23.86	24.50	25.47	26.00
Lena	30	18.73	24.18	26.22	26.33	28.19	28.39
House	30	18.73	23.89	25.98	26.17	27.16	27.72
Barbara	30	18.73	21.37	22.57	22.69	25.10	25.23
MCS library	30	19.21	21.13	22.08	22.05	23.72	23.97
Cameraman	30	19.08	20.68	23.08	23.94	24.56	24.94
Lena	35	17.46	23.79	25.63	25.65	27.48	27.56
House	35	17.46	23.34	25.49	25.52	26.57	27.21
Barbara	35	17.46	21.09	22.20	22.29	24.32	24.52
MCS library	35	18.00	20.45	21.72	21.54	23.11	23.97
Cameraman	35	17.80	20.55	22.50	23.15	23.80	24.26

4.7.2 MSE values for different images and different noise level:

Simulation results of Lena, house and Barbara image for soft thresholding, hard thresholding, visu shrink, proposed method 1 and proposed method 2 for different noise variances are given below.

Table 4.4 Shows the MSE values for Images Lena, House, Barbara for noise variance σ = 15,20,25,30,35 by soft, hard, Visu shrink, Proposed method 1, proposed method 2.

Input image	Noise variance	Noisy image	Soft thresholding	Hard thresholding	Visu shrink	Statistical method 1	Statistical method 2
Lena	15	223.12	146.58	84.20	86.14	49.04	45.23
House	15	221.85	153.90	86.22	80.07	67.05	51.97
Barbara	15	221.73	345.60	230.40	182.40	96.15	87.10
MCS Library	15	388.14	206.88	207.92	236.55	136.12	121.06
Cameraman	15	209.10	299.77	166.76	134.23	106.22	86.45
Lena	20	396.02	180.03	107.93	105.23	65.83	65.83
House	20	392.27	191.36	109.73	107.02	86.21	86.21
Barbara	20	391.77	393.60	284.35	251.21	131.87	122.23
MCS Library	20	447.06	362.45	324.09	299.05	184.46	171.21
Cameraman	20	369.67	373.7	217.87	185.45	142.84	126.10
Lena	25	613.03	217.41	131.63	128.29	83.01	78.84
House	25	614.87	230.64	135.59	132.56	106.18	94.30
Barbara	25	601.20	436.40	327.76	304.50	168.38	159.99
MCS Library	25	552.55	493.59	364.46	353.79	232.82	218.12
Cameraman	25	566.15	442.17	266.94	202.54	184.26	163.06
Lena	30	870.42	248.11	155.11	151.37	98.61	94.11
House	30	885.18	265.48	164.00	156.98	125.01	109.74
Barbara	30	846.92	474.26	359.81	349.90	200.63	194.99
MCS Library	30	775.55	533.23	402.35	405.41	275.58	260.11
Cameraman	30	811.99	500.83	319.72	294.01	227.09	208.22
Lena	35	1166	271.51	177.69	176.89	115.98	113.85
House	35	1169	301.23	183.28	182.20	143.09	123.39
Barbara	35	1138	505.08	391.09	383.23	240.10	229.35
MCS Library	35	1031	585.66	437.21	455.70	317.19	305.34
Cameraman	35	1075	572.87	365.10	304.26	270.46	243.38

5 Chapter 5 Conclusion and future work

In this thesis digital image de-noising using statistical based thresholding has been discussed. In the first part of the thesis, we have discussed introduction to wavelet transform. Then after we introduced some important features of DWT. Then the previously used thresholding techniques has been discussed with literature review.

In the latter part we proposed two different thresholding schemes that are based on the statistical properties of the input data. In the first approach statistical based thresholding scheme the image is passed by decomposition and after that we have applied this scheme on sub-band of the image. The advantage of this method is that it is much more adaptive towards the image coefficient values. The second statistical approach has improved the values more from the first statistical thresholding method. This method shows fairly satisfying results in both visual and numerical aspect. The statistical thresholding methods have better PSNR, MSE and visual quality than other techniques. These methods also shows better SSIM mean value and map.

Although our proposed methods has improved the results gives encouraging results, but we believe there is some space for further improvements to achieve more high results for example it is need that a hybrid thresholding scheme to be applied in image de-noising as a thresholding function, which is more coherent, improve the relation of neighbor dependency and representing better hierarchal dependency between the multiple decomposition levels.

6 References:

[1] Iain M.Johnstone David L Donoho. Adapting to smoothness via wavelet shrinkage.

Journal of the Statistical Association, 90(432):1200–1224, Dec 1995.

[2] David L Donoho. Ideal spatial adaptation by wavelet shrinkage. Biometrika,

[3] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1989..

[4] David L Donoho. De-noising by soft thresholding. IEEE Transactions on Information

Theory, 41(3):613–627, May 1995.

[4] Stephane Mallat. A Wavelet Tour of Signal Processing. Academic Press,
1999.

[5] Daubechies, I.,Orthonormal bases of compactly supported wavelets,Communications on Pure and Applied Mathematics, Vol. 41, No. 7, pp. 909-996, 1988.

[6] A theory for multi-resolution signal decomposition: The wavelet representation, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11, No. 7, pp. 674-693, 1989.

[7] Grossman, A. and Morlet , J., Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape.,SIAM Journal of Mathematical Analysis, Vol. 15, No. 4, pp. 723-736, 1984

[8] S. Grace Chang, Bin Yu and M. Vattereli. (2000). *Wavelet Thresholding for Multiple Noisy Image Copies*. IEEE Transaction. Image Processing, vol. 9, pp.1631- 1635.

[9] S. Grace Chang, Bin Yu and M. Vattereli. (2000). *Spatially Adaptive Wavelet Thresholding with Context Modeling for Imaged noising*. IEEE Transaction - Image Processing, volume 9, pp. 1522-1530.

[10] Unser, M. and Aldroubi, A.,A review of wavelets in biomedical applications,Proceedings of the IEEE, Vol. 84, No. 4, pp. 626-638, 1996.

[11] Laine, A.,Wavelets in spatial processing of biomedical images,Annual Review of Biomedical Engineering, Vol. 2, pp. 511-550, 2000

[12] Iain M.Johnstone David L Donoho. Adapting to smoothness via wavelet shrinkage.Journal of the Statistical Association, 90(432):1200–1224, Dec 1995.Tavel, P. 2007 Modeling and Simulation Design. AK Peters Ltd..

APPENDIX

Matlab Code:

```
clear;
clc;
clear all;
close all;
display('           ');
display('           ');

display('           ');
display('               SOME EXPERIMENTS ON IMAGE DENOISING USING WAVELETS
');

display('           ');
display('           ');
display('         RAJA RAO           ');

display('           ');
display('           ');

display('select the image');

display('           1:lena.png');
display('           2:barbara.png');
display('           3:mcslib.png');
display('           4:house.png');
display('           5:JF17.png');
display('           6:cameraman.jpg');
display('           ');
display('           7:hyderabad.png');
display('           8:friendgray.jpg');
display('           ');


ss1=input('enter your choice:   ');
switch ss1
    case 1
       f=imread('lena.png');
       %f=imread('babu.jpg');
    case 2
        f=imread('barbara.png');
    case 3
        f=imread('mcslib.png');
    case 4
        f=imread('house.png');
        case 5
        f=imread('JF17.png');
    case 6
        f=imread('cameraman.jpg');
```

```
    case 7
        f=imread('hyderabad512.png');
    case 8
        f=imread('friendgray.jpg');
end

% subplot(2,2,1), imshow(f);title('original image');

display('enter the type of noise:');
display('    1    for salt & pepper');
display('    2    for gaussian');
display('    3    for poisson');
display('    4    for speckle');

ud=input('enter the value:');

switch ud
    case 1
        display('enter the % of noise(Ex:0.2)');
        ud1=input('pls enter:     ');
        g=imnoise(f,'salt & pepper',ud1);
    case 2


%f=imread('peppers256.png');
%subplot(2,2,1),imshow(f);
display('enter the noise varience:  ');
sig=input('enter between 10 to 50:   ');

va =(sig/256)^2;
g=imnoise(f,'gaussian',0,va);
    case 3
       % display('enter the % of noise(Ex:0.2)');
        %ud1=input('pls enter:     ');
        g=imnoise(f,'poisson');
        case 4
        display('enter the varience of noise(Ex:0.02)');
        ud1=input('pls enter:     ');
        g=imnoise(f,'speckle',ud1);



end
%g=imnoise(f,'salt & pepper',01);
% subplot(2,2,2),imshow(g);title('noisy image');
figure, imagesc(g);colormap(gray);

%[ca,ch,cv,cd] = dwt2(g,'db2');
%c=[ca ch;cv cd];
%subplot(2,2,3),imshow(uint8(c));

x=g;
% Use wdencmp for image de-noising.
% find default values (see ddencmp).
[thr,sorh,keepapp] = ddencmp('den','wv',x);
display('');
```

```
display('select wavelet');
display('enter 1 for haar wavelet');
display('enter 2 for db2 wavelet');
display('enter 3 for db4 wavelet');
display('enter 4 for sym wavelet');
display('enter 5 for sym wavelet');
display('enter 6 for bior1.1 wavelet');
display('enter 7 for bior6.8 wavelet');
display('enter 8 for mexh wavelet');
display('enter 9 for coif  wavelet');
display('enter 10 for meyr wavelet');
display('enter 11 for morl wavelet');
display('enter 12 for  rbio wavelet');
display('press any key to quit');
display('');

ww=input('enter your choice:    ');
switch ww
    case 1
        wv='haar';
    case 2
        wv='db2';
    case 3
        wv='db4' ;
    case 4
        wv='sym2'
    case 5
        wv='sym4';
    case 6
        wv='bior1.1';
    case 7
       wv='bior6.8';
    case 8
        wv='mexh';
    case 9
        wv='coif5';
    case 10
        wv='dmey';
    case 11
        wv='morl';
    case 12
        wv='jpeg9.7';
    otherwise
        quit;
end
display('');
display('enter 1 for soft thresholding');
display('enter 2 for hard thresholding');
display('enter 3 for bayes soft thresholding');
sorh=input('sorh:   ');

display('enter the level of decomposition');
level=input(' enter 1 or 2 :    ');

switch sorh
    case 1
```

```
        sorh='s';
        xd = wdencmp('gbl',x,wv,level,thr,sorh,keepapp);
    case 2
        sorh='h';
        xd = wdencmp('gbl',x,wv,level,thr,sorh,keepapp);
    case 3
        %%%%%%%%%%%%%%%%%%%%%
       % clear all;
%close all;
%clc;

%Denoising using Bayes soft thresholding

%Note: Figure window 1 displays the original image, fig 2 the noisy img
%fig 3 denoised img by bayes soft thresholding


%Reading the image
%pic=imread('elaine','png');
pic=f;
%figure, imagesc(pic);colormap(gray);

%Define the Noise Variance and adding Gaussian noise
%While using 'imnoise' the pixel values(0 to 255) are converted to double in
the range 0 to 1
%So variance also has to be suitably converted
% sig=15;
% V =(sig/256)^2;
npic=g;
%npic=imnoise(pic,'gaussian',0,V);
%figure, imagesc(npic);colormap(gray);

%Define the type of wavelet(filterbank) used and the number of scales in the
wavelet decomp
filtertype=wv;
levels=level;

%Doing the wavelet decomposition
[C,S]=wavedec2(npic,levels,filtertype);

st=(S(1,1)^2)+1;
bayesC=[C(1:st-1),zeros(1,length(st:1:length(C)))];
var=length(C)-S(size(S,1)-1,1)^2+1;

% Calculating sigmahat
sigmahat=median(abs(C(var:length(C))))/0.6745;
% xthr=C;
% sigma=sqrt(var)
%  aa =  find(xthr>=2*sigma);
%  xthr(aa) = 2*sigma;
%   C=xthr;
%

thr1=zeros(3,size(S,1)-2);
for jj=2:size(S,1)-1
```

```
    %for the H detail coefficients
    coefh=C(st:st+S(jj,1)^2-1);
    thr=bayes(coefh,sigmahat);
    thr1(1,jj-1)=thr
    bayesC(st:st+S(jj,1)^2-1)=sthresh(coefh,thr);
    st=st+S(jj,1)^2;

    % for the V detail coefficients
    coefv=C(st:st+S(jj,1)^2-1);
    thr=bayes(coefv,sigmahat);
    thr1(2,jj-1)=thr
    bayesC(st:st+S(jj,1)^2-1)=sthresh(coefv,thr);
    st=st+S(jj,1)^2;

    %for Diag detail coefficients
    coefd=C(st:st+S(jj,1)^2-1);
    thr=bayes(coefd,sigmahat);
    thr1(3,jj-1)=thr
    bayesC(st:st+S(jj,1)^2-1)=sthresh(coefd,thr);
    st=st+S(jj,1)^2;
end


%Reconstructing the image from the Bayes-thresholded wavelet coefficients
bayespic=waverec2(bayesC,S,filtertype);
xd=bayespic;
%Displaying the Bayes-denoised image
figure, imagesc(uint8(xd));colormap(gray);
display('IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 9, NO. 9, SEPTEMBER
2000');

display('IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 9, NO. 9, SEPTEMBER
2000');
display('Adaptive Wavelet Thresholding for Image Denoising and Compression');
display('S. Grace Chang, Student Member, IEEE, Bin Yu, Senior Member, IEEE,
and Martin Vetterli, Fellow, IEEE');



        %%%%%%%%%%%%%%%%%%%%%%%%%%


end



%sorh=sorh;
% de-noise image using global thresholding option.


%f=imread('peppers256.png');
[c,s]=wavefast(g,level,wv);
% subplot(2,2,3),wave2gray(c,s,8);title('decomposed structure');
```

```
figure, imagesc(xd);colormap(gray);


% subplot(2,2,4),xd=uint8(xd);
% imshow(xd);title('denoised image');
%subplot(2,2,4),sub=f-xd;
%sub=abs(1.2*sub);
%imshow(im2uint8(sub));title('difference image');
ff=im2double(f);xdd=im2double(xd);
% figure, imshow(xd);
display('       ');
display('       ');
display('       ');
% snr=wpsnr(ff,xdd)
%
% display('       ');
% display('       ');
% mse=compare11(f,xd)
% noisy=im2double(g);
% SNR_NOISY_IMAGE=wpsnr(ff,noisy)
[MSE PSNR]=Calc_MSE_PSNR(f,xd)
[MSE PSNR]=Calc_MSE_PSNR(f,g)
%%
[ssimval, ssimmap] = ssim(f,xd);

fprintf('The SSIM value is %0.4f.\n',ssimval);

figure, imshow(ssimmap,[]);
title(sprintf('ssim Index Map - Mean ssim Value is %0.4f',ssimval));
```

Visu Shrink

```
clear all;
close all;
clc;

%Implementing Visu Shrink-
%Denoising using universal threshold with both hard and soft thresholding

%Note: Figure window 1 displays the original image, fig 2 the noisy img
%fig 3 denoised img by hard thresholding, fig 4 denoised by soft thresholding


%Reading the image
pic=imread('house.png');

figure, imagesc(pic);colormap(gray);
% subplot(2,2,1), imshow(pic);title('original image');

%Define the Noise Variance and adding Gaussian noise
%While using 'imnoise' the pixel values(0 to 255) are converted to double in
the range 0 to 1
```

```
%So variance also has to be suitably converted
sig=20;
V=(sig/256)^2;

npic=imnoise(pic,'gaussian',0,V);
figure, imagesc(npic);colormap(gray);
% subplot(2,2,2),imshow(npic);title('noisy image');

%Define the type of wavelet(filterbank) used and the number of scales in the
wavelet decomp
filtertype='bior6.8';
levels=5;

%Doing the wavelet decomposition
[C,S]=wavedec2(npic,levels,filtertype);
% subplot(2,2,3),wave2gray(C,S,8);title('decomposed structure');

% Define the threshold(universal threshold)
M=size(pic,1)^2;
UT=sig*sqrt(2*log(M));


%%


%Hard thresholding
%Doing the hard thresholding-threshold only detail coefficients!!
hardC=[C(1:S(1,1)^2), hthresh(C(S(1,1)^2+1:length(C)),UT)];

%Reconstructing the image from the hard-thresholded wavelet coefficients
newpich=waverec2(hardC,S,filtertype);

%Displaying the hard-denoised image
figure, imagesc(newpich);colormap(gray);
% subplot(2,2,4),newpich=uint8(newpich);
figure, imagesc(newpich);colormap(gray);


%Soft thresholding
softC=[C(1:S(1,1)^2), sthresh(C(S(1,1)^2+1:length(C)),UT)];

%Reconstructing the image from the soft-thresholded wavelet coefficients
newpics=waverec2(softC,S,filtertype);

%Displaying the soft-denoised image
% figure, imagesc(newpics);colormap(gray);

% Estimate the denoising effcet (i.e. computing MSE and PSNR)
[MSE, PSNR] = Calc_MSE_PSNR(pic,newpich)
%%
[ssimval, ssimmap] = ssim(pic,newpich);

fprintf('The SSIM value is %0.4f.\n',ssimval);
```

```
figure, imshow(ssimmap,[]);
title(sprintf('ssim Index Map - Mean ssim Value is %0.4f',ssimval));
```