1-1-2007

Minimum noiseless description length (MNDL) thresholding

Azadeh Fakhrzadeh
Ryerson University

MINIMUM NOISELESS DESCRIPTION LENGTH (MNDL) THRESHOLDING

by

Azadeh Fakhrzadeh
Bachelor of Electrical Engineering
Tehran Polytechnic University of Technology
Tehran, Iran, 2004

A thesis
presented to Ryerson University
in partial fulfillment of the
requirement for the degree of
Master of Applied Science
in the Program of
Electrical and Computer Engineering. Toronto, Ontario, Canada, 2007

Author's Declaration

I hereby declare that I am the sole author of this thesis.
I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.
Signature

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.
Signature

Abstract

MNDL Thresholding with Application od Data Denoising

© Azadeh Fakhrzadeh, 2007

Master of Applied Science (MASc)
Department of Electrical and Computer Engineering
Ryerson University

In this thesis, the problem of data denoising is considered and a new data denoising method is developed. This approach is an adaptive, data-driven thresholding method that is based on Minimum Noiseless Description Length (MNDL). MNDL is an approach to subspace selection which estimates bounds on the desired Mean Square Error (MSE). The subspace minimizing these bounds is chosen as the optimum one. In this research, we explore application of MNDL Subspace Selection (MNDL-SS) as a thresholding method. Although the basic idea and desired criterion of MNDL thresholding and MNDL-SS are the same, the challenges in calculation of the desired criterion in MNDL thresholding are very different. In MNDL-SS, the additive noise effects are in the form of samples of a Chi-Square random variable. However, this assumption does not hold for MNDL thresholding anymore. In this research, we developed a new method for calculation of the desired criterion based on characteristics of noise in thresholding. Our simulation results show that MNDL thresholding outperforms the compared methods.

In this thesis, we also explore the area of image denoising. In image denoising approaches, some properties of the image are considered. One of the well known image denoising methods, that outperforms other methods, is BayesShrink. We compare our method with BayesShrink. We show that the results of MNDL thresholding are comparable with BayesShrink in our simulations.

Acknowledgements

I would like to express my gratitude to all those who gave me the possibility to complete this thesis. Especially, I would like to thank the following people in orthocronological order of their presence in my life:

- I would like to thank my mother, Ashraf, who was a constant source of love in my life. She did her best to provide me with a situation to accomplish whatever I was looking for. She was my teacher and raised me with the passion for knowledge. Certainly, without her care and support, I would have not been able to achieve what I did.

- I would like to thank my father, Mehdi, who stood by me in every decision I made in life. He was my mentor in life and taught me how to treat every situation logically. I am thankful to him for everything he did for me.

- I would like to thank my brother, Mohammad, whose quick wit always made me laugh and swept away grief from my heart.

- I would like to thank my sister, Faezeh, who was always kind to me.

- I would like to thank my supervisor, Soosan Beheshti, to whom I owe a huge debt of gratitude. Had not she supported me in my M.Sc. career, I would have not been able to complete this thesis. It was a great honor for me to collaborate with her in various projects. Aside from her scientific character, which was a source of inspiration to me, her humane nature was appraisable. I wish her well in all aspects of her life.

- I would like to thank my friends here in Waterloo who made my life here a wonderful experience. Especially, I would like to thank Zahra Mousavian, Bita Roushanaei, Somayeh Azarnoosh, Narges Simjour, Masoumeh Roudafshani, Mojgan Madandar, Mahtab Kamali, Atefeh Mashatan , Leila Jalali, Azadeh mohebi and Faezeh Jahanmiri.

- I would like to thank my friends at Ryerson, Sina Zareiee , Sudeshna Pal and Omid Talakoub.

- I would like to thank Tahereh Garshasb for reading the manuscript and Narges Simjour and Masoumeh Roudafshani for technical support and consultations.

- The last but not least, I would like to thank my life companion and true friend, Amjad, whose supportive shoulder was always there for me. God breathed through him into my soul once again. He made my life much more enjoyable and without him, this thesis would not be possible.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

We recognize different phenomena by collecting data from them. The data is digitalized and saved in computer for analysis. Defective instruments, problems with the data acquisition process, and interfering natural factors can all degrade the data of interest. Furthermore, noise can be introduced by transmission errors and compression. Thus, denoising is often a necessary step to be taken before data processing. Different approaches have been introduced for denoising. Some of them like Wiener filters, are grouped as linear techniques. While these methods are simple to implement, their results are not always satisfactory. Therefore, researchers have improved the performance of existing methods by developing nonlinear approaches (for example [1] ,[2] and [41]). Although their approaches have succeeded in improving the results, their suggested methods are usually computationally exhaustive and hard to implement. In [3] Donoho and Johnstone introduced an interesting method which is both simple and effective in comparison with other techniques. The main issue in such an approach is to find a proper threshold. The threshold that Donoho and Johnstone introduced is called VisuShrink [3]. This threshold is a non-adaptive universal threshold and depends only on the number of data points and noise variance. Another recently introduced denoising method is called Minimum Description Length (MDL) thresholding [4]. MDL defines the Normalized Maximum Likelihood (NML) as the description length of the noisy data. of the noisy data in each subspace of the basis functions. A threshold which minimizes an estimate of the description of length is then chosen.

Thresholding is also used in image denoising. An improper threshold may introduce artifacts and causes blurring in the image. Different approaches have been suggested on this subject and the famous ones are BayesShrink and SureShrink. SureShrink [5] uses a hybrid of the universal threshold and the SURE (Stein's Unbiased Risk Estimator)threshold. The SURE threshold is chosen by minimizing Stein's estimation. BayesShrink [6] attempts to minimizes the Bayes' Risk Estimator function assuming a Generalized Gaussian prior distribution and thus yields a data adaptive threshold.

In this thesis, we develop a new denoising method called Minimum Noiseless Description Length (MNDL) thresholding. This method is based on MNDL Subspace Selection (MNDL-SS) which has recently been proposed by Beheshti and Dahleh [7]. MNDL-SS chooses the subspace that minimizes the description length of "noiseless" data. For this purpose, MNDL provides bounds on the reconstruction error. The subset of order m represents the bases with only m nonzero coefficients of the estimated denoised signal. For each subset The bounds on the reconstruction error are estimated and the subset that minimizes the upper bound of the reconstruction error is chosen as the best subset for representing the noiseless data.

In MNDL thresholding the desired criterion is the same as MNDL-SS. However, the characteristics of additive noise are different from MNDL-SS. Therefore, the main challenge is to develop a new method for estimating the desired criterion. In this research, we analyze the effect of additive noise in MNDL thresholding. Using the new approach, we calculate the optimum threshold. This threshold is a function of the noise variance σ_w^2, the data length N, and noisy data itself. We use some existing methods which are used as benchmark in recent papers and compare MNDL with them. The results of MNDL are very interesting. MNDL as a signal denoising method outperforms MDL and VisuShrink in our application. Its performance as an image denoising method is much better than SureShrink and is comparable to BayesShrink.

List of Notations:

N: Length of data

y: Noisy data,

$\bar{y}$: Noiseless data

w: additive noise

$\hat{y}$: Noiseless data estimation

θ: Wavelet transform of y

$\bar{\theta}$: Wavelet coefficients of $\bar{y}$

$\hat{\theta}$: Wavelet coefficients of $\hat{y}$

V: Wavelet coefficients of w

$y(i)$: The ith sample of y^N,

y^N :A vector with length of N, $y^N = [y(1), y(2), ..., y(N)]^T$

$Y(f)$: Fourier transform of y

$\eta_T(.)$: is a function applies the threshold T on its input

Y: is a random variable and y is a sample of it

$\| y^N \|_2$: l_2 -norm of vector y^N:$\sqrt{\sum_{i=1}^{N} y^2(i)}$

Chapter 2

Background

2.1 Data Denoising in Wavelet Domain

The poor quality of acquisition systems has always caused data to become corrupted with noise. Therefore, researchers, have been developing certain techniques to minimize the effects of contaminating noise. In data denoising techniques, we estimate the noiseless data using available noisy data, along with some statistical knowledge of additive noise. Different models have been suggested to describe noise; one of such noise models, which is frequently used in literature is Additive White Gaussian Noise (AWGN). This model has been known to be successful in describing the noise and is tractable in both the spatial and frequency domains. Therefore, in this work our focus is on data degraded with AWGN. This additive noise is independent from noiseless data. The $y(i)$ which is the i-th sample of noisy data is defined as

$$y(i) = \bar{y}(i) + w(i), \tag{2.1}$$

where $\bar{y}(i)$ is the i-th sample of noiseless data and $w(i)$ is i-th sample of AWGN noise with zero mean and constant variance σ_w^2. Different denoising techniques estimate the noiseless data differently. Mean Square Error (MSE) is one of the standard criterion used to estimate the noiseless data and is defined as:

$$MSE = E[\| \bar{Y}^N - \hat{Y}^N \|_2^2] \tag{2.2}$$

where $\bar{Y}^N$ is $[\bar{Y}(1), \bar{Y}(2), .., \bar{Y}(N)]^T$, N is the length of data, $\hat{Y}^N$ is an estimate of noiseless data. The general algorithm for most of the data denoising methods is shown in Figure 2.1. As we can see in this figure, the first step in data denoising is to transform the data

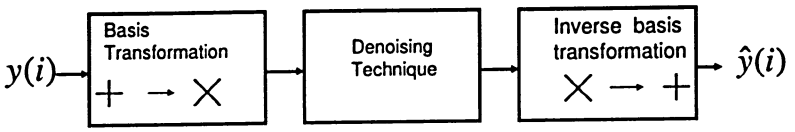

Figure 2.1: General algorithm for data denoising

from time domain into another domain. We do this transformation in hope of seeing further information of data, which is not obvious in time domain. More importantly, we hope to find a new domain in which the noiseless data is represented with fewer coefficients than that of additive noise. Applying a kind of transformation on data, we find the data representation in space S^N (N is the length of the data). The space S^N can be expanded by its orthogonal bases vectors $[s(1), s(2), s(3), ..., s(N)]$ such that

$$< s(i), s(j) >= \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases} \tag{2.3}$$

where $< s(i), s(j) >$ is the inner product of vectors, $s(i)$ and $s(j)$ and is calculated as $< s(i).s(j) >= s^T(i)s(j)$. The noiseless data, using this new base, is defined as follows:

$$\bar{y}^N = \sum_{i=1}^{N} \bar{\theta}(i)s(i) \tag{2.4}$$

where $\bar{\theta}(i)$ is the i-th coefficient of the noiseless data. The noisy data in space S^N is:

$$y^N = \sum_{i=1}^{N} \theta(i)s(i). \tag{2.5}$$

Please note that since S^N is a space with orthogonal bases we still have a Gaussian noise added to noiseless signal in S^N. In other words, the Equation 2.1 is held for the corresponding coefficients:

$$\theta(i) = \bar{\theta}(i) + V(i), \tag{2.6}$$

where $W(i)$ is the i-th noise coefficient in S^N. Different transforms have been suggested. The most commonly used ones are the Fourier and wavelet transforms. The wavelet transform has

some advantages over other transforms and has mostly been used in many areas. Although we can generalize our research, here we focus on the wavelet transform. The next section briefly introduces the wavelet transform and its properties.

2.1.1 The Wavelet Transform

The wavelet transform, has recently been introduced, and because of its unique properties has become very popular. In order to see how the wavelet transform works and what its properties are, let us compare it with the well-known Fourier Transform. The Fourier transform decomposes a signal into the spectrum of its frequency components:

$$X(f) = \int_{-\infty}^{+\infty} x(t)e^{-2\pi jft}dt \tag{2.7}$$

where $x(t)$ is the signal in time domain and $X(f)$ is its spectrum. Applying the inverse Fourier transform on $X(f)$, the time domain function can be synthesized. The inverse Fourier transform is :

$$x(t) = \int_{-\infty}^{+\infty} X(f)e^{-2\pi jft}df \tag{2.8}$$

The Fourier transform provides information about the frequency components of a signal, which can not be easily seen in time domain. Despite being very useful, the Fourier transform does have one drawback. When one looks at the output of the Fourier transform, is not able to discern where each of these frequency components has occurred in time domain. There are situations, in which we would rather have both frequency and time information simultaneously. To solve this problem, Gabor [8] has suggested the Short Term Fourier Transform (STFT). In STFT we move a window over the entire signal and take the Fourier transform in the windowed data. This approach solves the Fourier transform's problem to some extent. According to the Heisenberg's Uncertainty Principle, one cannot measure the frequency and time simultaneously with arbitrary precision, and Windowing-based the Fourier transform provides only a range of frequency components. The STFT with large window gives good frequency resolution and poor time resolution, while the STFT with small window size makes a better time information but poorer frequency information. This

shortcoming of the STFT was the motivation behind the development of a new transform. Therefore, the wavelet transform was born. The wavelet transform is a windowing technique like the STFT. However, rather than using a fixed window size for the entire signal, it employs a varying window size for different parts of the signal. The Fourier analysis consists of breaking up a signal into sinusoidal waves of various frequencies. wavelet analysis is the breaking up of a signal into shifted and scaled versions of the mother wavelet. Wavelets are a better tool for analyzing signals with sharp discontinuities. The reason that the wavelet transform can detect the discontinuity and the Fourier transform cannot, lies in the difference between the basis functions of these two transforms. Sinusoid are smooth and predictable. However, wavelets have a limited duration and are irregular and asymmetric. After taking into consideration all the advantages that the wavelet transform has, we chose it as our transform. In the next section, we briefly discusses the implementation of this transform.

Implementation of the Wavelet Transform

Before discussing implementation, we need to become familiar with wavelet basis functions and see how we can find the new representation of signal using these bases. The continuous wavelet transform is defined as a sum of the multiplication of signal to scaled and shifted mother wavelet (basis functions) over all time. As a result of the wavelet transform we have

$$\theta(l,\tau) = \int_{-\infty}^{+\infty} f(t)\psi^*_{l,\tau}(t)dt \tag{2.9}$$

where $*$ denotes complex conjugation, $\psi_{l,\tau}$ is scaled and shifted wavelet function, $\theta(l,\tau)$ is a wavelet coefficient which is a function of scale, l, and translation, τ. wavelet functions, $\psi_{l,\tau}$, are defined based on mother wavelet $\psi_{l,t}$ as following,

$$\psi_{l,\tau}(t) \equiv \frac{1}{\sqrt{l}}\psi(\frac{t-\tau}{l}) \tag{2.10}$$

The inverse wavelet transform is defined as:

$$f(t) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \theta(l,\tau)\psi^*_{l,\tau}(t)d\tau dl \tag{2.11}$$

The wavelet theory is based on the general properties of the wavelets. Here, we briefly take a look at some wavelet properties. A wavelet function must satisfy the admissibility and the regularity condition. The admissibility is defined as follows:

$$\int_{-\infty}^{+\infty} \frac{|\Psi(f)|^2}{|f|} df < +\infty \tag{2.12}$$

where $|\Psi(f)|$ is the amplitude of Fourier transform of $\psi(t)$. From admissibility, one deduces that the amplitude of Fourier transform of $\psi(t)$ vanishes at zero:

$$|\Psi(f)|^2|_{f=0} = 0 \tag{2.13}$$

The equation above tells us that the wavelet should be like a band pass filter. The Equation 2.13 is also equivalent with

$$\int_{-\infty}^{+\infty} \psi(t)dt = 0 \tag{2.14}$$

which means that the wavelet must be oscillatory (or in other words a wave). Another property of the wavelet function is its regularity. Based on regularity condition, the wavelet function should have some smoothness and concentration in both time and frequency domains. We need to discretize the data and transform tools (wavelets) to do all the computations by

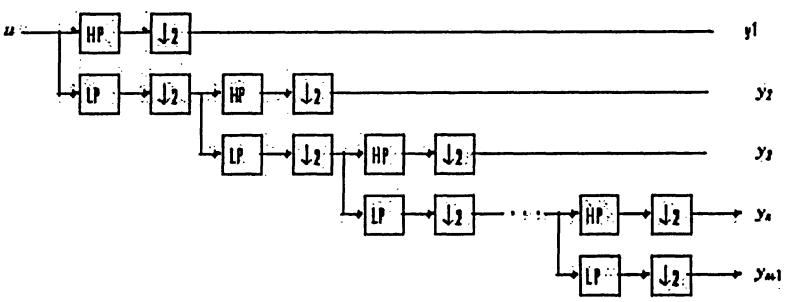

Figure 2.2: Discrete wavelet Transform (DWT) algorithm.

computer. For discretization, data is sampled with Nyquist sampling rate so that no information is lost. After data sampling, the discrete wavelet series is applied on data. To reduce the computation time different algorithms have been introduced, and the most efficient one is Discrete Wavelet Transform (DWT). DWT is implemented with the concept of filter banks,

where a series of high-pass and low-pass filters are applied to a signal. A subsampling step is done after filtering. The procedure is shown in Figure 2.2. Data u is passed from a low-pass and high-pass filter followed by subsampling by two. The high-pass filter output which is called detail subband, has half time resolution. The low-pass filter output which is called approximation sub-band, is used for further decomposition. This method is simply extended for 2D data (like images). For 2D data, the theory of DWT is the same as 1D but how we apply the filters is different. Images are stored in the computer in a matrix format with M rows and N columns. In every level of decomposition, a high-pass and a low-pass filter are first applied on rows. Consequently, we will have two horizontal approximation and detail subbands. Each of these two subbands is then filtered on columns. Therefore, in every level of 2D DWT, we have four subbands. Three of them are details and one of them, which is the output of low-pass low-pass filter, is the approximation subband . Just like 1D the approximation is used for further decomposition levels. Figure 2.3 shows the three levels of decomposition of Lena. After applying the wavelet transform we can employ one of the data

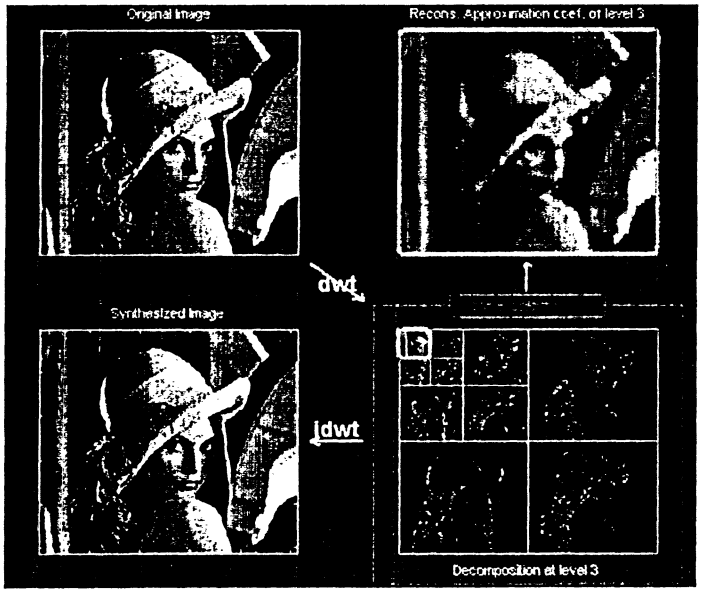

Figure 2.3: 2D DWT with 3 levels of decomposition. The figure was generated using matlab "wavemenue"

denoising methods on wavelet coefficients and estimate the noiseless data coefficients. The inverse wavelet transform is applied on noiseless coefficient estimations and the denoising data is complimented. Wavelet shrinkage is one of the well known denoising methods in the wavelet domain. The denoising method that we suggest in the next chapter, is categorized in this area. Thus, we review wavelet shrinkage in the next section.

2.2 Wavelet Shrinkage or Thresholding

Wavelet Shrinkage Method (WSM), also known as thresholding is a nonlinear and nonparametric signal denoising technique. When orthogonal wavelet bases are used, the coefficients with small absolute values tend to be attributed to the additive noise. By taking advantage of this property, finding a proper threshold, and setting all absolute values of coefficients smaller than the threshold to zero, one can suppress the noise. Different applications for wavelet Shrinkage exist. Weaver et. al. [9] have used it in signal and image processing for the first time. The first thorough mathematical treatment of wavelet thresholding has been done by Donoho and Johnstone [3].

In general, two thresholding methods exist: hard and soft thresholding. Hard thresholding kills or keeps the noisy coefficients by comparing them to the threshold

$$\hat{\theta}(i) = \begin{cases} \theta(i) & \text{if } |\theta(i)| > T_h, \\ 0 & \text{if } |\theta| < T_h. \end{cases} \tag{2.15}$$

where $\hat{\theta}(i)$ is the i-th sample of estimate of noiseless data, and $|\theta(i)|$ is the absolute value of i-th noisy coefficients and T_h is the hard threshold and . Soft thresholding kills the coefficients below T_s and subtracts the threshold from any coefficient that is greater than the threshold:

$$\hat{\theta}(i) = \begin{cases} \text{sign}(\theta)(|\theta - \text{T}_\text{s}|) & \text{if } |\theta| > T_s, \\ 0 & \text{if } |\theta| < T_s. \end{cases} \tag{2.16}$$

where T_S is the soft threshold. Figure 2.4 shows the both thresholding methods.

The threshold that Donoho and Johnstone first suggested is called VisuShrink and is a function of noise variance and data length. They have estimated the Mean Square Error (MSE) as defined in Equation 2.2, which is a measure of the error between the estimated

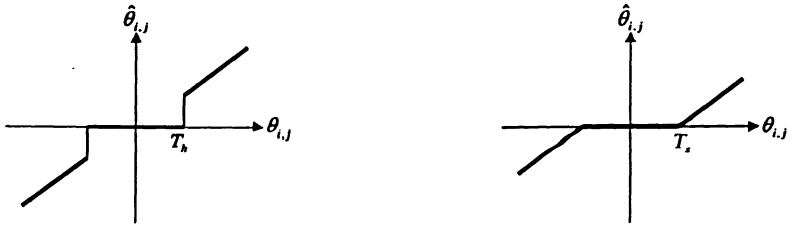

Figure 2.4: Left: Hard thresholding , right: Soft thresholding

and noiseless data as a function of the threshold. VisuShrink is the threshold that minimizes this risk and is defined as follows:

$$T_{visu} = \sigma_w \sqrt{2 \log N} \tag{2.17}$$

where σ_w is the variance of noise and N is the length of data. This threshold, is a hard threshold and is more suitable for smooth signals. Later Walczak and Massart [10] and Saito [11] used the Minimum Description Length (MDL) principle suggested by Rissenan [4] and proposed hard thresholds which are very much like T_{visu}. Their suggested hard theresholds are $\sigma_w \sqrt{\log N}$ and $\sigma_w \sqrt{3 \log N}$ respectively. MDL is a technique from algorithmic information theory. Based on MDL the best hypothesis for a given set of data is the one that leads to the largest compression of the data. Here we refer to denoising method using the threshold $\sigma_w \sqrt{\log N}$, as MDL thresholding. MDL thresholding and VisuShrink are two signal denoising methods that we compare with our own in the hard thresholding section. As we extend our denoising method to images , we briefly give an introduction about popular image denoising methods in the next section.

2.3 Image Denoising

One of the most widely used image denoising method is based on Bayesian Least Square (BLS) estimation [16] [34] . The BLS is defined as follows:

$$\hat{y}(y) = E(\bar{y}|y) = \int_{-\infty}^{+\infty} p_{\bar{Y}|Y}(\bar{y}|y)\bar{y}d\bar{y} = \frac{\int_{-\infty}^{+\infty} p_W(y-\bar{y})p_{\bar{Y}}(\bar{y})\bar{y}d\bar{y}}{\int_{-\infty}^{+\infty} p_W(y-\bar{y})p_{\bar{Y}}(\bar{y})d\bar{y}} \tag{2.18}$$

where y is a sample of random variable Y, $\bar{y}$ is a sample of random variable $\hat{Y}$, $p_{(Y|\bar{Y})}(y|\bar{y})$ denotes the density function of $\bar{y}$ given y, p_W is the probability density function of noise and $p_{\bar{Y}}$ is the prior probability density function of noiseless data. To solve Equation 2.18, in addition to probability of noise coefficients, we must know the probability of the noiseless data. If we assume a Gaussian distribution for noiseless data coefficients the BLS results in a linear solution to estimate the noiseless data. This linear solution which is also called Wiener Filter, is frequently used in the literature.

2.3.1 Wiener Filter

Wiener filter is a linear denoising method. For every pixels of image like $y(i,j)$, the output of the Wiener Filter is :

$$\hat{y}(i,j) = y(i,j)\frac{\sigma_{\bar{y}}}{\sigma_{\bar{y}} + \sigma_w} \tag{2.19}$$

where $\sigma_{\bar{y}}$ is the standard deviation of noiseless data and σ_w is the standard deviation of noise. For Wiener Filter, we need to estimate the variance of unavailable noiseless data. As the image is not stationary, it is better to have a local estimate of variance. To estimate the noiseless data variance, the simplest method which does not need any prior knowledge, is the Maximum Likelihood (ML) estimator. In this estimator, for each data point, $y(i,j)$, an estimate of $\sigma^2_{\bar{y}}(i,j)$ is formed based on its local neighborhood . We choose a square window B centered at $y(i,j)$, then compute the Maximum Likelihood (ML) estimator in B:

$$\hat{\sigma_{\bar{y}}}^2(i,j) = \arg\max_{\sigma^2 \geq 0} \prod_{k,m \in B} p(y(k,m)|\sigma^2) \tag{2.20}$$

$$= \max\left\{0, (\frac{1}{M}\sum_{i,j \in B} y(i,j)) - \sigma_w^2\right\} \tag{2.21}$$

where M is number of data points in B. When the variance is estimated, the denoised data is calculated simply as Equation 2.19 shows. Using an ML estimator Ramchandran et. al. showed in [13] that it is better to choose a different window size for the sharp and smooth regions of image. This method is heavily computational. In [12] they employed the Maximum A Posterior (MAP) estimator rather than the ML estimator. Before estimating $\sigma_{\bar{y}}$, they

estimated a prior marginal distribution $f_{\sigma_{\tilde{y}}}(\sigma_{\tilde{y}}^2)$ as a distribution of noiseless variance. These linear filters are simple in implementation, however, their performance are limited due to their assumptions. For example, linear filters assume that data is stationary, while real time images has a non-stationary nature. As nonlinear methods developed, experiments showed that they have a better rate of convergence than linear systems. Alternative approaches to nonlinear wavelet based denoising can be found in [14]– [45]. In some of them other algorithms rather than DWT is used [24] [23]. Wavelet thresholding is one of most used of such approaches. We discussed this method as signal denoising in the previous section and introduced two popular hard thresholding method in signal denoising. In image denoising the soft thresholding rule is preferred over hard thresholding. First, the optimal soft thresholding estimator yields a smaller mean square error than the optimal hard thresholding estimator. Second, in practice, the soft thresholding method yields more visually enhanced images over hard-thresholding because the latter is discontinuous and yields abrupt artifact in the recovered images, especially when the noise energy is significant. Here we are referring to two of popular soft thresholding methods. Their performance is much better than linear approaches. These methods are used as benchmarks in most of the wavelet shrinkage papers and in this thesis.

2.3.2 SureShrink

Donoho et al. have suggested a threshold approach, referred to as "SURE" threshold [5]. This method is a subband dependent method and treats every subband differently. SureShrink method is based on Stein's Unbiased Risk Estimate (SURE) [47]. Donoho et al. utilized the SURE estimate and computed equation the $MSE(T, \theta)$ as a function of noisy data and threshold in every decomposition subband. The SURE threshold which minimizes MSE is as follows:

$$T_{SURE} = \arg \min_{0<T<\sqrt{2\log(N)}} \mathrm{MSE}(\mathrm{T}, \theta/\sigma_{\mathrm{w}}^2). \tag{2.22}$$

where N is the number of coefficients in the considered subband, and θ is the noisy coefficients in that subband which is normalized by dividing to σ_w^2. SureShrink, applies a hybrid

threshold on an image, meaning that it applies the SURE threshold (as calculated above) to dense subbands, and the T_{visu} threshold in Equation 2.17 to sparse subbands. The condition which is checked to choose between T_{visu} and T_{SURE} threshold is as the following:

$$\hat{\theta}(i,j) = \begin{cases} \eta_{T_{SURE}}(\theta(i,j)) & \text{if } \xi_N^2 \geq \gamma_N \\ \eta_{T_{visu}}(\theta(i,j)) & \text{if } \xi_N^2 < \gamma_N. \end{cases} \tag{2.23}$$

where $\eta_T(.)$ is a function that applies the threshold T on its input, ξ_N^2 and γ_N are defined as the following: Another soft thresholding method is the BayesShrink. In this method, the threshold that minimizes the MSE is chosen through a different approach.

2.3.3 BayesShrink

The BayesShrink method has been introduced by Chang et al. [6]. They suggested a closed form threshold, called Bayes threshold, T_B, which is a function of noise and observed data. Through different graphs, they showed that this threshold is very close to the optimum threshold that minimizes the MSE. When we use the soft thresholded data as an estimate of noiseless data the MSE can be calculated as:

$$\text{MSE(T)} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}(\eta_T(\theta) - \bar{\theta})^2 p_{\Theta|\bar{\Theta}}(\theta|\bar{\theta}) p_{\bar{\Theta}}(\bar{\theta}) d\theta d\bar{\theta} \tag{2.24}$$

where $\eta_T(\theta)$ is a soft thresholded noisy data using $T(\theta)$, The optimum threshold that minimizes the equation above is calculated numerically and is a function of noiseless coefficients variance:

$$T^*(\sigma_{\bar{\theta}}) = \arg\min_T \text{MSE(T)} \tag{2.25}$$

To solve the Equation 2.24 we first need to assume a priori distribution for $\bar{\theta}$. Different distribution suggested [17, 19, 42, 48]. Mallat [48], has introduced Generalized Gaussian Distribution (GGD) as a distribution of image coefficients. Chang et al. have utilized GGD and calculated the optimum threshold in the Equation 2.25 numerically. Then they introduced an ad-hoc threshold which happened to be close to the optimum. Their threshold is calculated as:

$$T_B(\sigma_{\bar{\theta}}) = \frac{\sigma_w^2}{\hat{\sigma}_{\bar{\theta}}}. \tag{2.26}$$

For an unknown noise variance case, they used the Robust Median Estimator to estimate the value of noise standard deviation:

$$\sigma_w(\theta) = \frac{\text{Median}|\theta(\text{i,j})|}{0.675}, \theta(i,j) \in \text{subband D1} \tag{2.27}$$

where $D1$ is the diagonal subband at the decomposition level 1. The variance of noisy data coefficients is estimated as follows:

$$\sigma_\theta^2 = \frac{1}{N}\sum_{i,j=1}^{N} \theta(i,j)^2, \theta(i,j) \in \text{a specific subband} \tag{2.28}$$

where N is the number of all pixels in the considered subband. BayesShrink estimates the noiseless data coefficients variance in Equation 2.26 as follows:

$$\hat{\sigma}_{\bar{\theta}} = \begin{cases} \sqrt{\sigma_\theta^2(i,j) - \sigma_w^2} & \text{if } \sigma_\theta^2 > \sigma_w^2, \\ 0 & \text{if } \sigma_\theta^2 < \sigma_w^2. \end{cases} \tag{2.29}$$

2.4 Conclusion

In this chapter we briefly reviewed several popular denoising methods. The denoising methods are categorized into two groups: linear and non-linear. As an example of linear methods we studied the Wiener filter. Nonlinear approaches have better results than linear. Wavelet shrinkage is a nonlinear approach which is preferred over existing methods due to its simplicity and efficiency. Two of the well-known signal thresholding methods are MDL and VisuShrink. These thresholds are function of data length and noise variance. BayesShrink and SureShrink are two popular image denoising techniques. Both methods try to find a threshold that minimizes MSE but with different approaches. BayesShrink outperforms SureShrink. However, the calculation of BayesShrink threshold is through an ad-hoc approach resulted from experiments.

Chapter 3

New Subspace Selection Method

3.1 Minimum Noiseless Description Length (MNDL)

Minimum Noiseless Description Length (MNDL) is a new approach to subspace selection, which has recently been proposed by Beheshti and Dahleh [7]. In the MNDL approach the length of "noiseless" data is estimated in every subspace. The subspace with minimum data length is chosen as the best subspace for representing data. The existing subspace selection methods decide between competing subspaces based on noise variance and length of data. MNDL, includes the characteristics of data as well in its judgment, and has shown advantages over the existing methods [4]. For now, let us examine how Data Length (DL) is computed. When a Gaussian noise is added to data, the density function of noisy data is

$$f_Y(y^N;\bar{y}^N) = \frac{1}{(\sqrt{2\pi\sigma_w^2})^N} e^{\frac{-\|y^N-\bar{y}^N\|_2^2}{2\sigma_w^2}} \tag{3.1}$$

where $f_Y(y^N;\bar{y}^N)$ is the definition of density function of random variable y^N using the $\bar{y}^N$, $\bar{y}^N$ is the noiseless data and y^N is the noisy data, N is length of data and σ_w^2 is the noise variance. Let us consider y^N which is a sample of Y^N. Using the Shannon code the code length of y^N is:

$$DL(y^N;\bar{y}^N) = -\frac{1}{N}\log_2(f_Y(y^N;\bar{y}^N)) \tag{3.2}$$

$$= \log_2\sqrt{2\pi\sigma_w^2} + \frac{\| y^N - \bar{y}^N \|_2^2}{2\sigma_w^2 N}\log_2 e \tag{3.3}$$

In the equation above, the DL of y^N is calculated by noiseless data; $\bar{y}^N$. However, we know that $\bar{y}^N$ quantity is not available. The noiseless data belongs to a space S_N, assume that the best representation of noiseless data belongs to subspace S_m and is defined such:

$$y_{S_m}^N = \sum_{i=1}^{N} \hat{\theta}_{S_m} s(i) \tag{3.4}$$

where where $S_m \subset S^N$, m denotes the order of this subspace and the $\hat{\theta}_{S_m}$ is defined as:

$$\hat{\theta}_{S_m}(i) = \begin{cases} \theta(i) & \text{if } s_i \in S_m \\ 0 & otherwise. \end{cases} \tag{3.5}$$

We can calculate the DL of y^N using $\hat{y}_{S_m}^N$ is as follows:

$$DL(y^N; \hat{y}_{S_m}^N) = \log_2 \sqrt{2\pi\sigma_w^2} + \frac{\| y^N - \hat{y}_{S_m}^N \|_2^2}{2\sigma_w^2 N} \log_2 e \tag{3.6}$$

The comparison of DL calculated above in order to find the best subspace fails, since it is monastically decreasing and we cannot be minimized. To solve this problem MDL [4] method adds an ad-hoc extra term to the definition of DL in Equation 3.6. However, MNDL proposes has a better solution. MNDL uses the estimate $\hat{y}_{S_m}^N$ by noisy data and provides the DL of "noiseless" data. This new definition of DL is a function of MSE and noise variance:

$$DL(\bar{y}^N; \hat{y}_{S_m}^N) = \log_2 \sqrt{2\pi\sigma_w^2} + \frac{\log_2 e}{2\sigma_w^2} z_{S_m} \tag{3.7}$$

where z_{S_m} is the MSE in Equation 2.2 corresponding to subspace S_m and can be defined as follows:

$$z_{S_m} = \frac{1}{N} \| \bar{y}^N - \hat{y}_{S_m}^N \|_2^2, \tag{3.8}$$

For a known noise variance we need to estimate the MSE to compute the DL in (3.7). Therefore, in MNDL, the first step before subspace selection is finding bounds on MSE. With a proper choice of the competing subspace, this method not only chooses the optimum subset, but can also provide the optimum threshold simultaneously. The goal of this research is to investigate the implementation of MNDL subspace selection (MNDL-SS) as data denoising method.

3.2 Estimation of Mean Square Error(MSE) in MNDL

MNDL judges between subspace using the noiseless data description length given in Equation 3.7. This DL is a function of MSE. MNDL shows that the minimization of DL is equivalent to the minimization of MSE, and uses the MSE as a criterion in subspace selection. However, we know that MSE is not available due to its dependency on original data. MNDL suggests probability bounds on it, using information from observed data. The probability bounds are then used as an MSE estimate. Note that based on Parseval's Theorem, the MSE error in (3.8) can be written in the following form

$$z_{S_m} = \frac{1}{N} \parallel \bar{\theta} - \hat{\theta}_{S_m} \parallel_2^2 . \tag{3.9}$$

where $\bar{\theta}$ represents noiseless coefficients and $\hat{\theta}_{S_m}$ denotes the noiseless estimate coefficients. The $\hat{\theta}_{S_m}$ is a sample of random variable $\hat{\Theta}_{S_m}$, so the z_{S_m} in the equation above is a sample of random variable $Z_{S_m} = \frac{1}{N} \parallel \bar{\theta} - \hat{\Theta}_{S_m} \parallel_2^2$. MNDL uses the expected value of this random variable to provides bounds on z_{S_m}. In the next section the expected value of Z_{S_m} is calculated and in the following sections we can see how this quantity can be used in MSE estimate.

3.2.1 Calculation of MSE and its Expected Value

To provide bounds on MSE, its expected value, is calculated. MNDL computes the expected value of MSE as a sum of the noiseless and noisy parts. The noiseless part is roughly l_2-norm of noiseless coefficients which are not in S_m, and the noisy part is the l_2-norm of noise coefficients in subspace S_m. MNDL estimates these two parts separately. Now, let us examine how MNDL calculates z_{S_m} and its expected value. For the simplicity, we assume that S_m corresponds to only first m bases of S^N,$(s_i \in S_m, i = 1, ..., m)$. The noisy data (in Equation 2.6) in subspace S_m, can be rewritten as follows:

$$y^N = [A_{S_m} \;\; B_{S_m}] \begin{pmatrix} \bar{\theta}_{S_m} \\ \Delta_{S_m} \end{pmatrix} + w^N \tag{3.10}$$

where w^N is a vector of noise coefficients with length of N, the columns of matrix A_{S_m} are $s_i \in S_m, 1 \leq i \leq N$, and the columns of matrix B_{S_m} are basis vectors that are not

in S_m,$s_i \in \bar{S}_m$. The vector $\theta^*_{S_m}$ is the noiseless coefficients in S_m. Therefore, the noiseless coefficients with length of N can be represented as

$$\bar{\theta} = \begin{pmatrix} \bar{\theta}_{S_m} \\ \Delta_{S_m} \end{pmatrix} \tag{3.11}$$

where Δ_{S_m} is a vector of length $N - m$, corresponding to the coefficients of bases that are not in S_m. Using Equations 3.10 and 3.11 the vector of the estimate coefficients $\hat{\theta}_{S_m}$ is

$$\hat{\theta}_{S_m} = \begin{pmatrix} A^H_{S_m} y^N \\ 0_{(N-m\times 1)} \end{pmatrix} = \begin{pmatrix} \bar{\theta}_{S_m} + A^H_{S_m} w^N \\ 0_{(N-m\times 1)} \end{pmatrix} \tag{3.12}$$

where H denotes the Hermitian. Exploiting the vectors defined in Equations 3.10, 3.11 and (3.12) z_{S_m} can be expressed as a function of the basis vectors, additive noise, and the noiseless coefficients:

$$z_{s_m} = \frac{1}{N} \parallel \bar{\theta} - \hat{\theta}_{S_m} \parallel_2^2 = \frac{1}{N} \parallel A^H_{S_m} w^N \parallel_2^2 + \frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2 \tag{3.13}$$

where $\parallel \Delta_{S_m} \parallel_2^2$ is the l_2-norm of discarded coefficients in subspace S_m. As in subspace selection the additive noise effect is independent from data, it is known that Z_{S_m} has a Chi-square structure. Using the Chi-Square random variable the expected value and variance of Z_{S_m} is obtained, as follows:

$$E(Z_{S_m}) = \frac{m}{N}\sigma_w^2 + \frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2 \tag{3.14}$$

$$var(Z_{S_m}) = \frac{2m}{N^2}\sigma_w^2 \tag{3.15}$$

The first part of (3.14) is the expected value of the noisy part of MSE, which we briefly refer to as the noisy part of MSE. The second part which is an unmoulded dynamic data is referred to here as the noiseless part of MSE. With Chi-Square assumption, the noisy part has found the linear form as a function of noise variance and order of subspace m. Using the expected value of Z_{S_m} in (3.14) one can estimate bounds on z_{S_m}. The calculation of these bounds are in the next section.

3.2.2 Probability Bounds on MSE using its Expected Value

Having the expected value and variance of a random variable, the bounds are . The expected value of Z_{S_m} was calculated in Equation 3.14. The random variable Z_{S_m} is near its mean with probability p_1 as follows:

$$p\{|Z_{S_m} - E(Z_{S_m})|\} \leq D_{S_m} = p_1 \tag{3.16}$$

where p is the probability function, D_{S_m} is a function of p_1 and the structure of random variable Z_{S_m}. As Z_{S_m} is a Chi-square random variable, D_{S_m} can be found using the table of Chi-square random variable. Therefore, MSE error is between it lower bound $\underline{z_{S_m}(p_1)}$ and its upper bound as $\overline{z_{S_m(p_1)}}$ as follows:

$$\underline{z_{S_m}(p_1)} \leq z_{S_m} \leq \overline{z_{S_m(p_1)}} \tag{3.17}$$

When by using (3.14) and (3.16) we have:

$$\underline{z_{S_m}(p_1)} = \frac{m}{N}\sigma_w^2 + \frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2 - D_{S_m}(p_1) \tag{3.18}$$

$$\overline{z_{S_m}(p_1)} = \frac{m}{N}\sigma_w^2 + \frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2 + D_{S_m}(p_1) \tag{3.19}$$

These bounds are functions of $\parallel \Delta_{S_m} \parallel_2^2$, while in the real world the $\parallel \Delta_{S_m} \parallel_2^2$ is not known. Hence, to complete our estimate in Equations 3.18 and 3.19 we need to estimate $\parallel \Delta_{S_m} \parallel_2^2$ as well. MNDL introduces a new quantity so-called "Data error", and utilizes it to validate $\parallel \Delta_{S_m} \parallel_2^2$. Data error is an accessible quantity and can be calculated using observed data.

3.2.3 Estimation of MSE using Data Error

Bounds on MSE were estimated in (3.18) and (3.19). These bounds are dependent on noiseless coefficients through $\parallel \Delta_{S_m} \parallel_2^2$. Unfortunately, we only have access to noisy data and $\parallel \Delta_{S_m} \parallel_2^2$ is not known. MNDL employs Data error which is an error between noisy data and its estimate in a specific subspace, to find an approximation for $\parallel \Delta_{S_m} \parallel_2^2$. The Data error is calculated as follows:

$$x_{S_m} = \frac{1}{N} \parallel y^N - \hat{y}_{S_m}^N \parallel_2^2, \tag{3.20}$$

Where using Parseval's theorem we have:

$$x_{S_m} = \frac{1}{N} \parallel \theta - \hat{\theta}_{S_m} \parallel_2^2, \tag{3.21}$$

Since $\hat{\theta}_{S_m}$ is a sample random variable $\hat{\Theta}_{S_m}$, one can conclude that x_{S_m} is a sample of a random variable too. MNDL validates the expected value and variance of this random variable and uses them in estimating $\parallel \Delta_{S_m} \parallel_2^2$. In the next section we see how the second order statistics of X_{S_m} can be helpful in the estimation of $\parallel \Delta_{S_m} \parallel_2^2$.

Estimation of $\parallel \Delta_{S_m} \parallel_2^2$ using Second Order Statistics of X_{S_m}

In this section first, the expected value and variance of X_{S_m} are calculated, and then we can see how $\parallel \Delta_{S_m} \parallel_2^2$ is derived from them. The estimated data can be written as:

$$\hat{y}_{S_m}^N = [A_{S_m} \quad B_{S_m}]\hat{\theta}_{S_m} \tag{3.22}$$

Using the equation above, the Data error is represented as:

$$x_{s_m} = \frac{1}{N} \parallel y^N - \hat{y}_{S_m}^N \parallel_2^2 = \frac{1}{N} \parallel B_{S_m}\Delta_{S_m} + G_{S_m}w^N \parallel_2^2 \tag{3.23}$$

where

$$G_{S_m} = I - A_{S_m}A_{S_m}^H = B_{S_m}B_{S_m}^H \tag{3.24}$$

is a projection matrix. Due to the independency of noise effect from noiseless data, X_{S_m} finds the Chi-square structure of $N - m$-th order and therefore has the following expected value and variance:

$$E(X_{S_m}) = (1 - \frac{m}{N})\sigma_w^2 + \frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2 \tag{3.25}$$

$$var(X_{S_m}) = \frac{2}{N}(1 - \frac{m}{N})(\sigma_w^2)^2 + \frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2 \tag{3.26}$$

The similarity between $E(X_{S_m})$ and $E(Z_{S_m})$ in Equations 3.14 and 3.25, becomes apparent as they both have a linear noisy part with additive a term $\parallel \Delta_{S_m} \parallel_2^2$. The noisy part of $E(Z_{S_m})$ is a decreasing function of m, while the noisy part of $E(X_{S_m})$ is an increasing function of m.

When the variance of noiseless data is small, the expected value of X_{S_m} in Equation 3.25, is very close to Data error, x_{S_m}, therefore the $\frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2$ is estimated as follows:

$$\frac{1}{N} \parallel \hat{\Delta}_{S_m} \parallel_2^2 \approx x_{s_m} - (1 - \frac{m}{N})\sigma_w^2 \tag{3.27}$$

If the variance is significant, a probabilistic validation for $\parallel \Delta_{S_m} \parallel_2^2$ is suggested. Data error x_{S_m}, is a function of $\frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2$ and by bounding it, one can validate $\parallel \Delta_{S_m} \parallel_2^2$. for each value of $\parallel \Delta_{S_m} \parallel_2^2$, we have:

$$p\{|X_{S_m} - E(X_{S_m})| \leq J_{S_m}\} = p_2 \tag{3.28}$$

The bound J_{S_m} is a function of $\parallel \Delta_{S_m} \parallel_2^2$,$\sigma_w$,$m$ and p_2, the value of J_{S_m} is calculated based on these values and Chi-square table. Given x_{S_m}, and validation probability p_2, the $\parallel \Delta_{S_m} \parallel_2^2$ is validated as follows:

$$|x_{S_m} - E(X_{S_m})| \leq J_{S_m}(p_2, \sigma_w^2, m, \parallel \Delta_{S_m} \parallel_2^2) \tag{3.29}$$

The estimated $\frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2$ obtained from the equation above, can be used in Equations 3.18 and 3.19 and the estimation of z_{S_m} is completed.

Bounds on MSE Using Data Error

Using the bounds on $\parallel \Delta_{S_m} \parallel_2^2$ in Equation 3.29, we can provide bounds on the reconstruction error or MSE. With validation probability p_2 and confidence probability p_1 the MSE error bounds are as follows:

$$\underline{z_{S_m}(p_1, y^N, p_2)} \leq z_{S_m} \leq \overline{z_{S_m}(p_1, y^N, p2)} \tag{3.30}$$

where

$$\underline{z_{S_m}(p_1, y^N, p_2)} = \min\left\{0, \min_{\frac{1}{N}\|\Delta_{S_m}\|_2^2 \in (L_{S_m}, U_{S_m})} \{E(Z_{S_m}) - D_{S_m}\}\right\} \tag{3.31}$$

and

$$\overline{z_{S_m}(p_1, y^N, p_2)} = \max_{\frac{1}{N}\|\Delta_{S_m}\|_2^2 \in (L_{S_m}, U_{S_m})} \{E(Z_{S_m}) - D_{S_m}\} \tag{3.32}$$

where L_{S_m} is the lower bound of $\frac{1}{N} \parallel \Delta_{S_m} \parallel_2^2$ and U_{S_m} is its upper bound. Here, bounds on z_{S_m} are calculated using the Chi-square table, noise variance, subspace order, length of data, noisy data for a choice of validation probability and confidence probability. These bounds provide bounds on description length of data and can be used for subspace selection. MNDL also suggests using the Central Limit Theorem (CLT) and estimates two Chi-square random variables, X_{S_m} and Z_{S_m} with Gaussian. Using Gaussian distribution obviates us from Chi-square table. Readers please refer to the paper, [7] for more details.

3.2.4 Subspace Selection Using MSE Estimate

Thus far, probability bounds on MSE were calculated. From Equation 3.7, one can conclude that comparing the description of length in different subspaces, is the same as comparing of reconstruction error. MNDL suggests a comparison of the worst case of reconstruction error or upper bound; $\bar{z}_{S_m}$, and chooses the subspace that minimizes this quantity. Based on this criterion we have:

$$S_{m^*} = \arg\min_{S_m} \overline{z_{S_m}(p_1, y^N, p_2)} \tag{3.33}$$

where S_{m^*} is the optimum threshold. The decision is made based on p_1 and p_2 which are confidence probability and validation probability, respectively. If we form the competing subspaces in a special form, the best subspace selection is equivalent with denoising with thresholding. In other words MNDL can provide best subspace and threshold simultaneously.

3.3 Using MSE Estimate in Denoising by Thresholding

Searching between all 2^N subspaces to find the optimum threshold is almost infeasible. To solve this problem MNDL suggests comparing between the nested subspace of different orders. The absolute value of noisy coefficients $\Theta(i)$s are sorted in decreasing order. This provides a nested set of subspaces. In this case, the subspace S_1 is the subspace corresponding to θ_1, where $\theta_1 = \max_i \theta(i)$ and subspace S_i is spanned by the first ith bases associated with the sorted set. With this approach the number of subspaces is at most N and optimum subspace can provide the optimum threshold too. For the optimum subspace S_m^*, in which

the data description is minimized, the optimum threshold is the minimum value of coefficients associated with S_m^*:

$$T^* = \min_i |\theta(i)|, i \in j | s_j \in S_m^* \tag{3.34}$$

This threshold is a function of observed noisy data. In the next chapter we examine the application of this threshold in data denoising.

Chapter 4

MNDL Thresholding

We have reviewed about MNDL subspace selection (MNDL-SS) in the previous chapter. MNDL-SS estimates the MSE in every subspace and chooses the subspace in which the MSE is at minimum. We have also learned that if we form the competing subspaces in a special way, the MNDL subspace selection is can be used for MNDL thresholding. While in MNDL-SS, we assume that the subspaces are chosen a priori and are not functions of the observed data, in MNDL thresholding, the subspaces are chosen based on the observed data. In other words, in the MNDL thresholding approach, the competing subsets have to be chosen as nested subsets based on the sorted version of absolute value of coefficients. For example, the first subset represents the basis associated with the largest absolute value of the coefficients. The subset with two coefficients includes this basis and the basis with the second largest value from the absolute value of sorted coefficients. The thresholding question is then answered by providing the optimum subset in which MSE is at minimum. Let us make this clear with one example. We consider the Block signal and add a Gaussian noise with variance one to this signal. The noiseless signal, noisy signal and their associated wavelet coefficients are shown in Figure 4.1. Sorted version of absolute value of noisy coefficients are shown in Figure 4.2. The basis associated with the first m coefficients in this graph belongs to S_m, where $1 < m < N$. We refer to m as the order of subspace. Let us assume $m = 50$. To find the estimate of noiseless signal in S_m, we keep the coefficients of noisy signal corresponding to the first 50 coefficients of sorted version, and set the rest of coefficients to zero. This

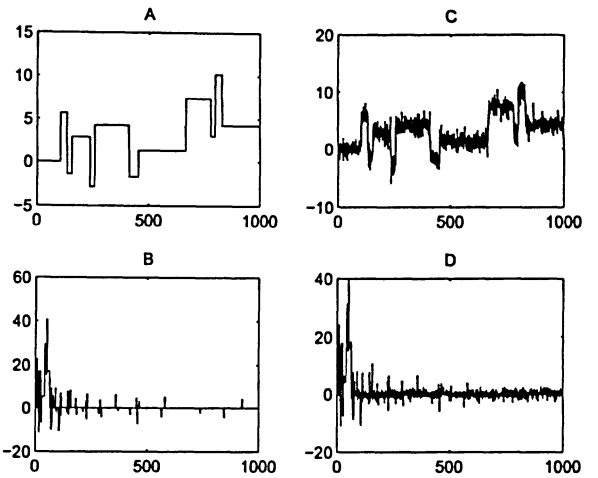

Figure 4.1: (A) Noiseless Block signal, (B) Noiseless Block coefficients (C) Noisy Block signal (noise level is "1") (D) Noisy Block coefficients. Haar wavelet is used with 4-level wavelet transform decompostion.

estimate is shown in Figure 4.3. MSE in this subspace, z_{S_m}, is calculated by finding the error between estimated signal in Figure 4.3 and noiseless signal in part B of Figure 4.1. For different values of m the z_{S_m} is calculated. This error as a function of m is plotted in Figure 4.4. As we can see in this graph, the z_{S_m} has one minimum. This minimum happens at m_{opt}. For this especial case m_{opt} is 80. This corresponds to the fact that the 80 largest coefficients of noisy data is the best representation of noiseless signal. The estimated noiseless signal is shown in Figure 4.5. m_{opt} plays a crucial role in the MNDL thresholding. In MNDL thresholding we are looking for an estimate of the desired z_{S_m} in Figure 4.4 to estimate the m_{opt}. We first tried the MSE estimate obtained from MNDL-SS for MNDL thresholding case and noticed this approach dose not provide a proper estimate of z_{S_m}. Figure 4.6 shows the desired MSE and MNDL-SS estimate. The reason that MNDL-SS estimate is not working for MNDL thresholding is their difference in the effect of additive noise on them. In the following sections we develop a new estimate of MSE for MNDL thresholding, taking into account the new characteristics of noise.

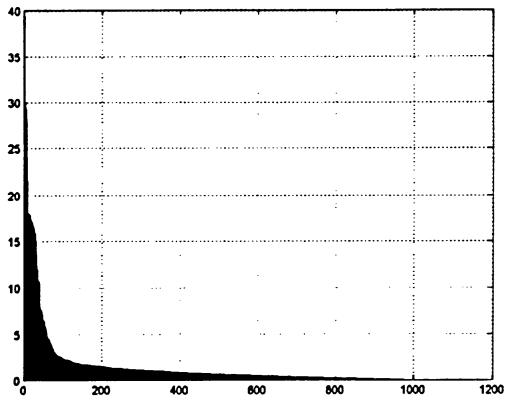

Figure 4.2: Sorted version of absolute values of noisy coefficients

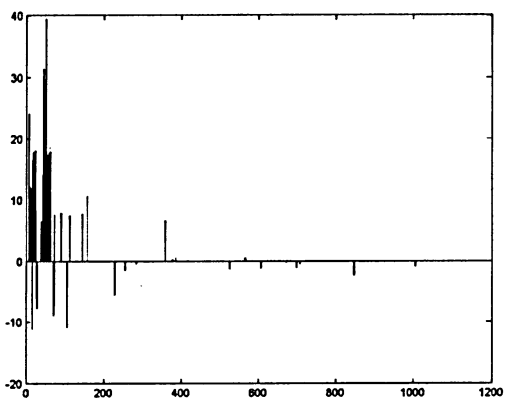

Figure 4.3: Estimate of noiseless signal in subspace S_m, where $m = 50$

4.1 MNDL Hard Thresholding and the Main Challenge

In MNDL hard thresholding, we use the MNDL-SS to estimate theMSE. This estimate is later used to find a proper threshold. In MNDL-SS we estimated the MSE using its expected value. The MSE was first formed as the sum of noisy and noiseless parts in Equation 3.13. Afterwards, the expected value of MSE was calculated by estimating these two parts,

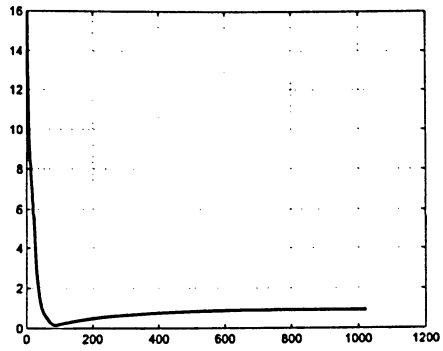

Figure 4.4: MSE as a function of m, tested signal is Block signal, noise level is 1.

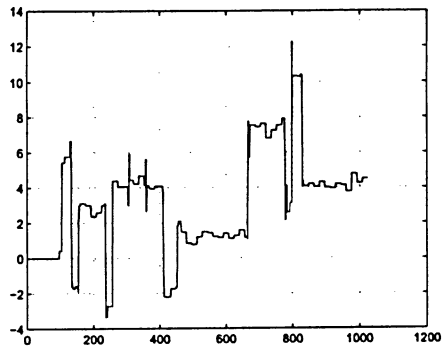

Figure 4.5: The best representation of noiseless signal obtained with minimizing the true MSE.

separately. The expected value of z_{S_m} in Equation 3.13 can be written as follows:

$$E(Z_{S_m}) = E(noise_{zsm}) + \frac{1}{N}\sum_{i=1}^{N}\bar{\theta}(i)^2. \tag{4.1}$$

The first part of equation above is an estimate of noisy part of z_{S_m}, and the second part is the noiseless part. The noiseless part of $E(Z_{S_m})$ was estimated using the expected value of Data error, X_{S_m}. The expected value of X_{S_m} can be defined as:

$$E(X_{S_m}) = E(noise_{xsm}) + \frac{1}{N}\sum_{i=1}^{N}\bar{\theta}(i)^2. \tag{4.2}$$

We can see in Equations 4.1 and 4.2, that the noiseless part of $E(X_{S_m})$ is the same as $E(Z_{S_m})$. Finding the expected value of the noisy part of $E(X_{S_m})$, and deducting it from

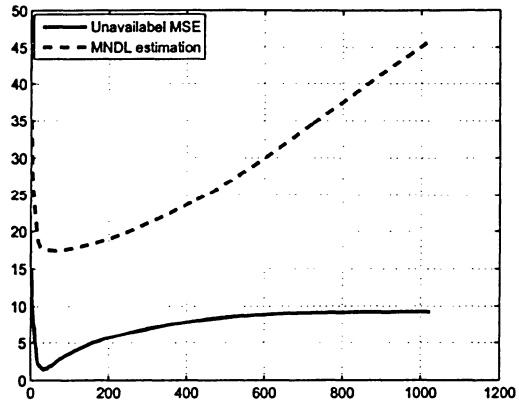

Figure 4.6: Desired MSE and its estimate as a function of m using the MNDL-SS. Noise variance is "1".

$E(X_{S_m})$, one can obtain the noiseless part of z_{S_m}. Now, let us revisit the effect of additive noise in both errors. As we defined in the previous chapter, we have:

$$\theta(i) = \bar{\theta}(i) + V(i) \tag{4.3}$$

where $\theta(i)$ is noisy data coefficient, $\bar{\theta}(i)$ is the noiseless coefficient and $V(i)$ is the noise coefficient, which is a sample of a Gaussian random variable with zero mean and variance σ_w^2. When we compose the competing subspaces of different orders, the MSE is estimated as a function of the subspace order m. The m_{opt} in which MSE is minimum defines the best subspace and threshold simultaneously.

For now, let us consider the case that the length of noisy data is 16. We sort this noisy data as shown in Figure 4.7 (A), then for every m ($0 < m < 16$), we compute the MSE. For when m is equal to 7, the estimate of noiseless data which is the 7 largest coefficients of noisy data is shown in Figure 4.7(B). Part (D) shows the difference between noiseless coefficients in part (C) and its estimate in part (B). It is obvious that, this difference is equal to noiseless coefficients in every point that estimate of noiseless data in part (B) is zero, and is equal to noise coefficients ($V(i)$) in all other points. The mean square of this difference is the MSE

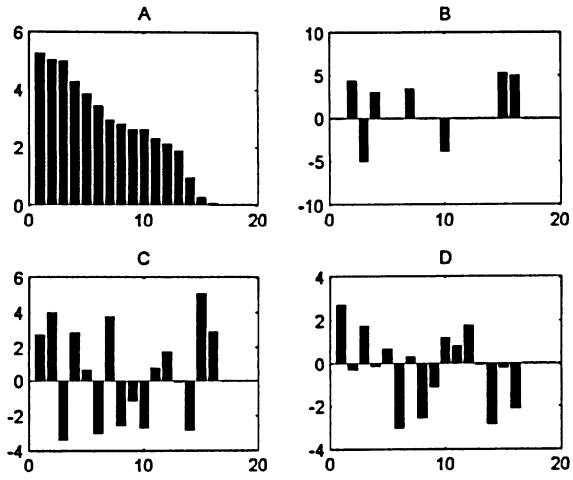

Figure 4.7: (A) Sorted coefficients, (B) Estimate of noiseless coefficients for $m = 7$ (C) Noiseless coefficients (D) the difference between (B) and (C).

for this value of m. Therefore, the expected value of noise-related part of $E(Z_{S_m})$ is

$$E(noise_{zsm}) = \frac{1}{N}\sum_{i=1}^{m} E(V(i)^2) \tag{4.4}$$

The Data error is the mean square error between noisy data and its estimate in Figure 4.7, part (B). Therefore, the expected value of its noisy part is defined as

$$E(noise_{xsm}) = \frac{1}{N}\sum_{i=m}^{N} E(V(i)^2) \tag{4.5}$$

MNDL-SS assumed that $V(i)$s are samples of Chi-square random variable and found the following estimations of Equations 4.5 and 4.5:

$$E(noise_{xsm}) = \frac{m}{N}\sigma_w^2 \tag{4.6}$$

and

$$E(noise_{xsm}) = \frac{N-m}{N}\sigma_w^2 \tag{4.7}$$

However, these estimations are not working for the case of MNDL thresholding. We have plotted the desired noisy part of both errors and MNDL-SS estimations in Figure 4.8. In

MNDL-SS the effect of noise in both errors are independent from the noiseless part. Therefore, we could take x_{S_m} and z_{S_m} as two Chi-square random variables. In MNDL thresholding, the dependency assumption is not exact and we cannot use the Chi-square structure in estimation of (4.1) and (4.2) anymore. Therefore, the main challenge in MNDL thresholding

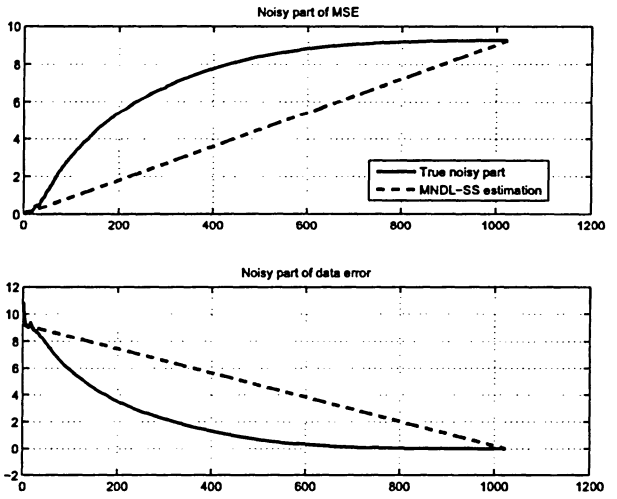

Figure 4.8: Desired noisy part of MSE and its estimate using MNDL-SS (top) and the desired noisy part of Data error and its estimate (bottom) as a function of m. Noise variance is one and signal is Block.

is to calculate the effect of addictive noise in MSE error and Data error.

4.1.1 The Effect of Additive Noise in MSE and Data Error

As discussed, the effect of additive noise is very important in MNDL thresholding. Here we suggest two alternatives to determine the expected value of the noisy part of MSE and Data error. To calculate the expected value, we need to assume a distribution for noise, $V(i)$s, in Equations 4.4 and 4.5. This noise is associated with sorted noisy coefficients. When the noise coefficients are dominant, by sorting noisy data coefficients, we assume that noise will be sorted too. We know that the $V(i)$s, in Equation 4.3, are samples of Gaussian noise. Therefore, we assume the noises effect in MSE and Data error are sorted Gaussian. As sorted Gaussian is not an ordinary distribution, the expected value of $\frac{1}{N}\sum_{i=1}^{m} V(i)^2$ can be

calculated by Monte Carlo method [49]. The result of this approach is reasonable. However, it is very time-consuming. We suggest another method which is based on the exact calculation of Equations 4.4 and 4.5. This method leads to almost the same results of the first method. However its running time is quite shorter.

The Effect of Additive Noise in MSE Using Monte Carlo Method

The first approach in calculating the effect of noise is based on sorted Gaussian distribution with zero mean and variance of ${\sigma_w}^2$. We first focus on the noisy part of MSE. To calculate this quantity, in this section, we generate the noisy part M times and calculate its mean. In each trial, additive white Gaussian noise with variance σ_w^2 is generated and the absolute value of associated coefficients are sorted. Let us denote the sorted noise coefficients of length N with $\vec{G}_i^{sort}[n]$ where i represents the i-th trial. Therefore, the expected value of noisy part of z_{S_m} is estimated as follows:

$$E(noise_{zsm}) = \frac{1}{M}\sum_{i=1}^{M}\sum_{n=1}^{m}(\vec{G}_i^{sort}[n])^2 \tag{4.8}$$

The quantify above is used as an estimate of noisy part of z_{S_m}. We evaluate the estimate in equation above using two signals; Block and Mishmash. We have seen Block signal in Figure 4.1 and Mishmash is shown in Figure 4.9. Figure (4.10) shows the true noisy part of

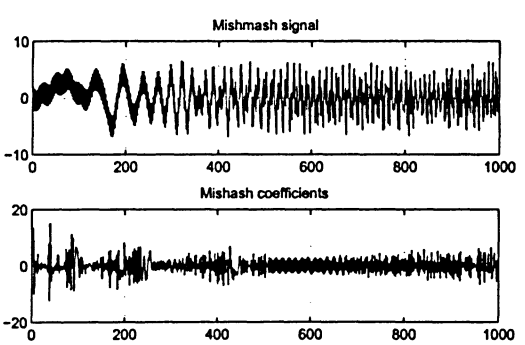

Figure 4.9: Mishmash signal and its associated coefficients.

the reconstruction error, z_{S_m}, the estimate in Equation 4.6 (used by MNDL-SS) and the new

estimate in Equation 4.8 for Block and Mishmash signals. As Figure (4.10) shows, for the Block signal with almost 70 nonzero coefficients, and for the smaller values of m, the noisy part coincides with $\frac{m}{N}\sigma_w^2$. However, for higher values of m, the new approximation from (4.8) performs better than $\frac{m}{N}\sigma_w^2$. For Mishmash signal, with almost 1024 nonzero coefficients, and for noise level one, the true noisy part of reconstruction error is very close to $\frac{m}{N}\sigma_w^2$. For this signal and smaller levels of noise, since the contribution of signal coefficients are more than noise, in thresholding method we sort signal coefficients rather than the noise. Therefore we do not have sorted Gaussian noise anymore. However, for this signal, when the level of noise is high, the estimate using sorted Gaussian works better and Figure 4.10, part (D) confirms this claim. Comparing the results in Figure 4.10 one can conclude that in most cases except for Mishmash with noise of 1, the estimate in Equation 4.8 is a better estimate. We know

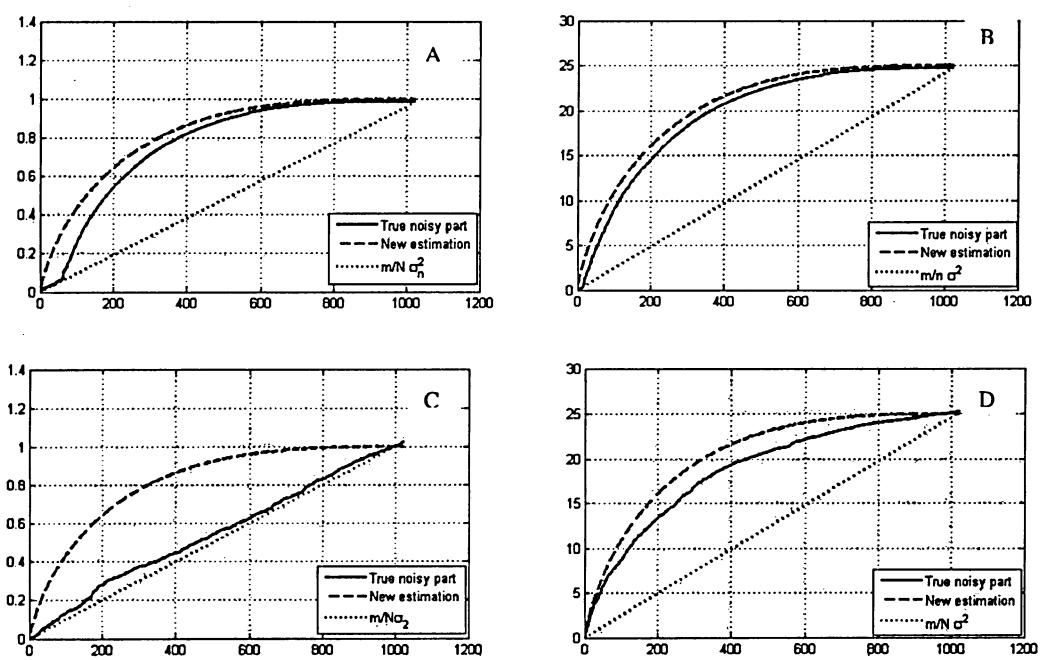

Figure 4.10: The true noisy part of MSE, the estimate $\frac{m}{N}\sigma_w^2$ and the estimate in Equation 4.8 as a function of m. (A) Block signal, (B) Block signal, $\sigma_w = 5$ (c) Mishmash signal, $\sigma_w = 1$ (D) Mishmash signal, $\sigma_w = 5$

that to complete the MSE estimate the next step is to estimate the noiseless part of MSE

using the expected value of Data error. For this we need to calculate the effect of noise in Data error too.

The Effect of Additive Noise in Data Error Using Monte Carlo Method

The noiseless part of MSE is calculated using the expected value of Data error. In this step, we first calculate the expected value of X_{S_m}, as the sum of noisy and noiseless parts (like what we have seen in Equation 4.2. The expected value of Data error in MNDL thresholding is

$$E(X_{S_m}) = \frac{1}{N} E[\sum_{i=m}^{N} (\bar{\theta}(i) + V(i))^2]. \tag{4.9}$$

We need to break up the expected value of X_{S_m} into noiseless and noisy parts. To do so, we extend the equation above:

$$\frac{1}{N}\sum_{i=m}^{N} E[(\bar{\theta}(i) + V(i))^2)] = \frac{1}{N}\sum_{i=m}^{N} E(\bar{\theta}(i))^2 + \frac{1}{N}\sum_{i=m}^{N} E(V(i)^2) + \frac{1}{N}\sum_{i=1}^{N} E(2\bar{\theta}(i)V(i)). \tag{4.10}$$

The first part of above equation is the l_2-norm of the noiseless part or $\frac{1}{N}\Delta_{S_m}$, which we would like to estimate. The second part is the expected value of noisy part of Data error. The last part is a multiplication of noisy and noiseless coefficients. If we assume that $\theta^*(i)$s are independent from $V(i)$s then the expected value of the third part becomes zero. Through the estimating x_{S_m} with its expected value in (4.9) we have

$$\hat{x}_{S_m} = \frac{1}{N}\sum_{i=m}^{N} \bar{\theta}(i)^2 + \frac{1}{N} E(\sum_{i=m}^{N} V(i)^2) \tag{4.11}$$

The second part of above equation is the same as $E[(noise_{x_{S_m}})]$ in Equation 4.5, which we would like to estimate it here. Following the same discussion we had for MSE, one approaches to estimating this part with the sorted Gaussian. Using sorted Gaussian assumption the estimate of this part is:

$$E(noise_{x_{S_m}}) = \frac{1}{M}\sum_{i=1}^{M}\sum_{n=m}^{N} (\vec{G}_i^{sort}[n])^2 \tag{4.12}$$

where $\vec{G}_i^{sort}[n]$ is the absolute value of sorted Gaussian vector and i represents the i-th trial. While for the noisy part of z_{S_m} in (4.8), in every trial, we choose the first m coefficients of

$\vec{G}_i^{sort}$, here we consider $N - m$ coefficients of it. This new estimate along with the linear MNDL estimate for both Block and Mishmash signals are shown in Figure 4.11. In this case, most of the time the new estimate works better.

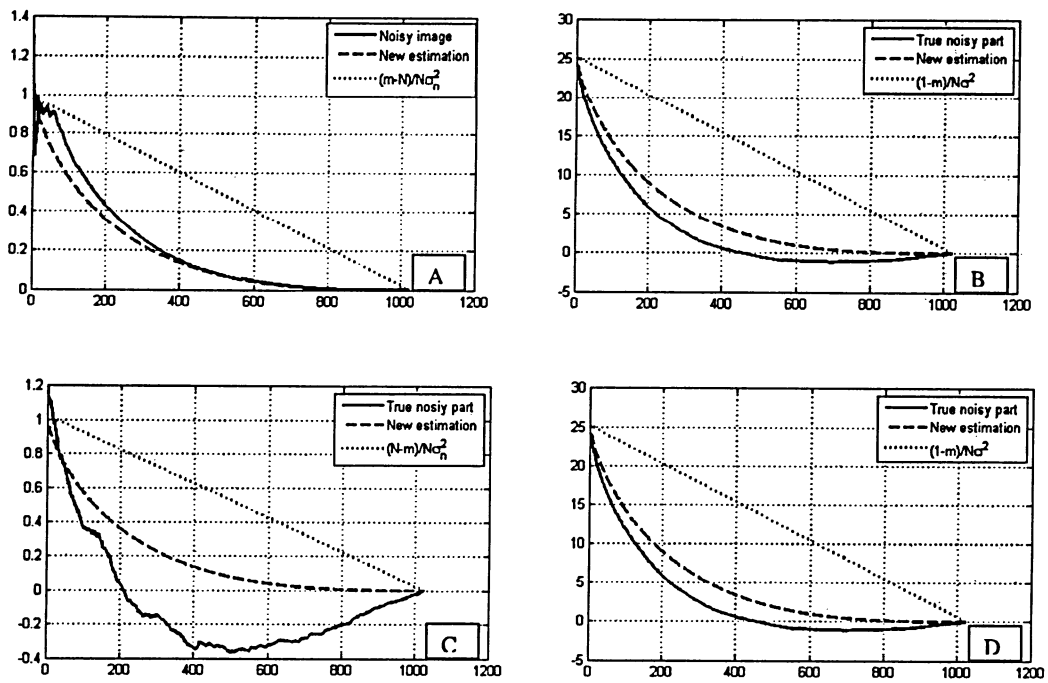

Figure 4.11: The true noisy part of Data error, the estimate $\frac{m}{N}\sigma_w^2$ the new estimate in Equation 4.12 as a function of m. (A) Block signal, $\sigma_w = 1$ (B) Block signal, $\sigma_w = 5$ (c) Mishmash signal, $\sigma_w = 1$ (D) Mishmash signal, $\sigma_w = 5$

Exact Calculation Approach

The effects of additive noise were calculated in the previous parts using the sorted Gaussian. The results of this estimate are reasonable. However, calculating Equations 4.12) and (4.8 by matlab is time consuming. This problem manifests itself when the length of the data is long. Therefore, here we develope another method which is based on the exact calculation of Equations 4.5 and 4.4. The noise part of sorted version of coefficients is a sample of $V(i)$s, where every $V(i)$ has roughly a Gaussian distribution, with mean of $\bar{\theta}(i)$ and variance of σ_w^2. With this assumption the exact solution of expected value of additive noise is as follows:

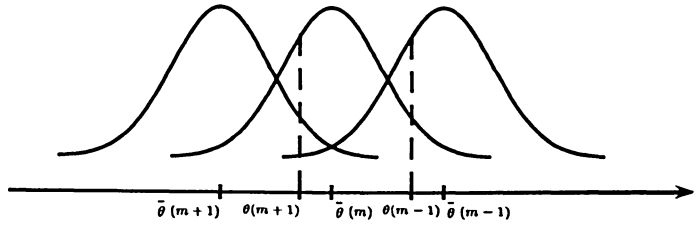

Figure 4.12: The distribution of $V(m)$, $V(m-1)$, $V(m+1)$. The colored region is the permissible region for θ_m

Calculation of expected value of noisy part of MSE, $E(noise_{z_{s_m}})$: To find a solution for Equations 4.5 and 4.4, we first focus on $V(m)$, and then extend our solution for all possible values of $V(i)$s. Figure 4.12 shows the distribution of $V(m)$, $V(m-1)$, $V(m+1)$. With Gaussian assumption, and using the Bayesian theory, the expected value of MSE noisy part under condition $\theta(m+1) < \theta(m) < \theta(m-1)$ is calculated as follows:

$$E[V(m)^2] = \frac{\int_{\theta(m+1)}^{\theta(m-1)} v_m^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-(v_m-\bar{\theta}(m))^2}{2\sigma_w^2}} dv_m}{\int_{\theta(m+1)}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-(v_m-\bar{\theta}(m))^2}{2\sigma_w^2}} dv_m - \int_{\theta(m-1)}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-(v_m-\bar{\theta}(m))^2}{2\sigma_w^2}} dv_m} \tag{4.13}$$

To make the calculation easier we shift the v_m to v_{m+1} and as a result we have:

$$E[V(m)^2] = \frac{\int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} v_m^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m}{\int_{\theta(m+1)-\bar{\theta}(m)}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m - \int_{\theta(m-1)-\bar{\theta}(m)}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m} \tag{4.14}$$

The numerator of Equation 4.14 would be:

$$\int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} v_m^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m = \frac{\sigma_w}{\sqrt{2\pi}}[\theta(m-1)-\bar{\theta}(m)]e^{-\frac{[\theta(m-1)-\bar{\theta}(m)]^2}{2\sigma_w^2}} \tag{4.15}$$

$$-\frac{\sigma_w}{\sqrt{2\pi}}[\theta(m+1)-\bar{\theta}(m)]e^{-\frac{[\theta(m+1)-\bar{\theta}(m)]^2}{2\sigma_w^2}} + \sigma_w^2[Q(\frac{[\theta(m-1)-\bar{\theta}(m)]}{\sigma_w}) - Q(\frac{[\theta(m+1)-\bar{\theta}(m)]}{\sigma_w})]$$

The denominator of Equation 4.14 is calculated as follows:

$$Q(\frac{[\theta(m+1)-\bar{\theta}(m)]}{\sigma_w}) - Q(\frac{[\theta(m-1)-\bar{\theta}(m)]}{\sigma_w}) \tag{4.16}$$

and eventually by dividing nominator to denominator the $E[(V(m))^2]$ in Equation 4.14 is calculated as :

$$E[V(m)^2] = \sigma_w^2 + \varepsilon(diff1, diff2) \tag{4.17}$$

where $diff1 = \theta(m+1) - \bar{\theta}(m)$ and $diff2 = \theta(m-1) - \bar{\theta}(m)$ and $\varepsilon(diff1, diff2)$ is

$$\varepsilon(diff1, diff2) = \frac{f_{diff1} - f_{diff2}}{Q(diff1) - Q(diff2)} \tag{4.18}$$

where f_{diff1} and f_{diff2} are defined as:

$$f_{diff1} = \frac{\sigma_w}{\sqrt{2\pi}} \exp(\frac{-diff1^2}{2\sigma_w^2}) diff1, \tag{4.19}$$

and

$$f_{diff2} = \frac{\sigma_w}{\sqrt{2\pi}} \exp(\frac{-diff2^2}{2\sigma_w^2}) diff2. \tag{4.20}$$

The expected value of noise coefficients, for the case of $V(m)$, was calculated in Equation 4.17. Now we have to extend our calculations for all $V(i)$s when ($i \in [1, 2, ..., N]$) and take the sum over all possible values. To do so, let us to define two vectors, $\vec{diff1}$ and $\vec{diff2}$ as follows:

$$\vec{diff1} = [\theta(3), \theta(4), \theta(5), ...] - [\bar{\theta}(2), \bar{\theta}(3), \bar{\theta}(4), ...] = \vec{\Theta_3} - \vec{\Theta_2}^* \tag{4.21}$$

$$\vec{diff2} = [\theta(1), \theta(2), \theta(3), ...] - [\bar{\theta}(2), \bar{\theta}(3), \bar{\theta}(4), ...] = \vec{\Theta_1} - \vec{\Theta_2}^* \tag{4.22}$$

where the index of $\vec{\Theta_3}$ shows the starting point of the vector. Using $\vec{diff1}$ and $\vec{diff2}$ the noisy part of MSE is such:

$$E(noise_{z_{S_m}}) = \frac{1}{N}(\sum_{i=1}^{m} E(V(i))^2) = \frac{m}{N}\sigma_w^2 + \frac{1}{N}\sum_{i=1}^{m} \varepsilon(\vec{diff1}(1:m), \vec{diff2}(1:m)) \tag{4.23}$$

and the noisy part of Data error is such:

$$E(noise_{x_{S_m}}) = \frac{1}{N}(\sum_{i=m}^{N} E(V(i))^2) = \frac{N-m}{N}\sigma_w^2 + \frac{1}{N}\sum_{i=1}^{m} \varepsilon(\vec{diff1}(m:N), \vec{diff2}(m:N)) \tag{4.24}$$

The last parts of equations above, are dependent on $\vec{diff1}$, $\vec{diff2}$ which are not available. In the following section, we suggest some methods to estimate these two parameters.

Estimation of the Important Elements

In the previous section, the expected value of the noisy part of MSE was calculated as a function of $\vec{diff}1$ and $\vec{diff}2$. Two vectors $\vec{diff}1$ and $\vec{diff}2$ in Equations 4.21 and 4.22 are the difference between the absolute value of sorted noisy coefficients and their associated noiseless coefficients ($\vec{\bar{\Theta}}_2$). These two quantities are not available because of their dependency on noiseless coefficients. Here we try to estimate them using information we obtain from noisy data. Estimate of $\vec{diff}1$ and $\vec{diff}2$, with their expected value is our solution. Following, two alternatives are suggested to find an approximation of expected value of them.

1. **Sorting method**

   This estimate is based on sorted Gaussian distribution. We used this distribution once in Monte Carlo method in section (4.1.1). Here we exploit the vector $\vec{G}_i^{sort}$ that we generated in section (4.1.1) to estimate the$\vec{diff}1$ and $\vec{diff}2$ as follows:

   - Generate a Gaussian vector with variance of noise and length of data.
   - Sort the absolute value of associated coefficients, $\vec{G}_i^{sort}$ (where i representing the number of trial).
   - Find the expected value of 50 trials, $E[\vec{G}_i^{sort}]$, where $1 < i < 50$.
   - Estimate the $\vec{\bar{\Theta}}_2$ as follows:

   $$\hat{\vec{\bar{\Theta}}}_2 \approx \vec{\Theta}_2 - E[\vec{G}_i^{sort}] \tag{4.25}$$

   - Calculate $\vec{diff}1$ and $\vec{diff}2$ by replacing $\hat{\vec{\bar{\Theta}}}_2$ with $\vec{\bar{\Theta}}_2$ in Equations 4.21 and 4.22.

   $$\hat{\vec{diff}}1 = \vec{\Theta}_3 - \hat{\vec{\bar{\Theta}}}_2; \tag{4.26}$$

   $$\hat{\vec{diff}}2 = \vec{\Theta}_1 - \hat{\vec{\bar{\Theta}}}_2; \tag{4.27}$$

   - Calculate The noisy part of MSE in Equation 4.23 and 4.24 using the $\hat{\vec{diff}}1$ and $\hat{\vec{diff}}2$:

   $$\hat{E}(noise_{z_{S_m}}) = \frac{m}{N}\sigma_w^2 + \frac{1}{N}\sum_{i=1}^{m}\varepsilon(\hat{\vec{diff}}1, \hat{\vec{diff}}2) \tag{4.28}$$

$$\hat{E}(noise_{x_{S_m}}) = \frac{N-m}{N}\sigma_w^2 + \frac{1}{N}\sum_{i=m}^{N} \epsilon(\hat{\vec{diff}}1, \hat{\vec{diff}}2) \tag{4.29}$$

2. **Unsorting method**

This method is like "sort" method; however, $\bar{\vec{\Theta}}_{S_m}$ is estimated as follows:

$$\hat{\bar{\vec{\Theta}}}_2 \approx \vec{\Theta}_2 - E[|\vec{G}_i|] \tag{4.30}$$

where $|\vec{G}_i|$ is the absolute value of a Gaussian vector with variance of noise and length of data.

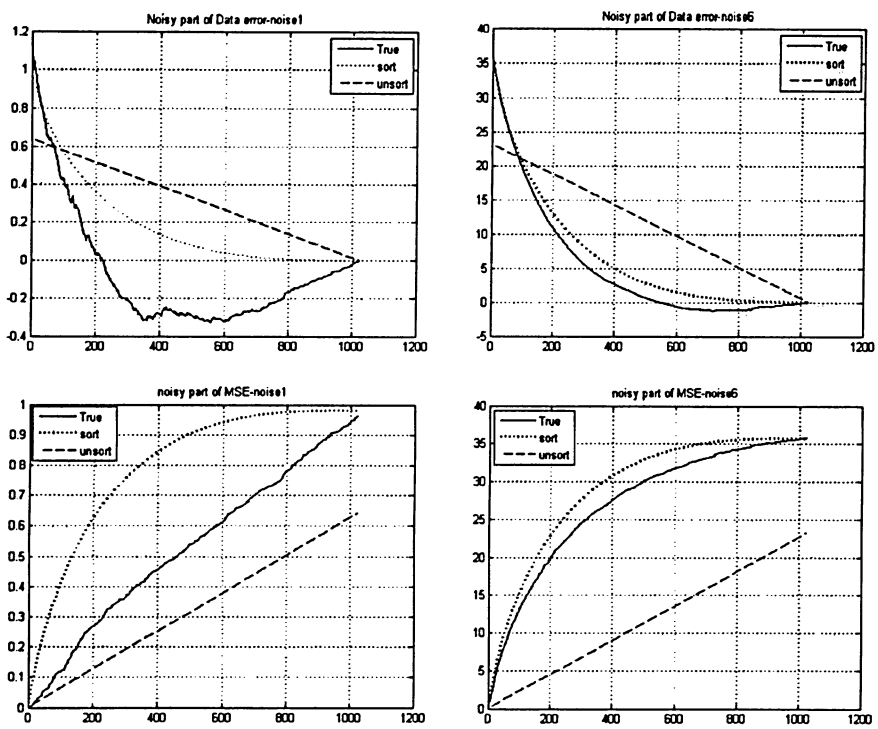

Figure 4.13: True noisy part and its estimates as a function of m, True: the unavailable noisy part, "Sort": when we use sorted Gaussian, "Unsort" : when we use the unsort Gaussian to estimate the $\vec{diff}1$, $\vec{diff}2$.

These two methods are compared in Figure 4.13. In this figure, the true noisy part of MSE and Data error are compared with their estimations. The test signal is Mishmash and the

level of noise is one and six. From Figure 4.13, we can comprehend that in all of the cases, estimate using the Unsort Gaussian fails. We introduced two approaches to estimating the additive noise in both errors. The first one is called Monte Carlo approach, the second one was the exact solution for this problem. The noisy part obtained from exact solution, in Equations 4.29 and 4.28, are compared with the noisy parts from fist method in Equations 4.8 and 4.12, in Figure 4.14. We can see that the results, obtained from both approaches, are almost the same. However, in terms of running time, using the exact solution is much better than using Monte Carlo method. Therefore, we decided to choose the exact solution to estimated the effect of noise. Now that the effect of additive noise was determined, we

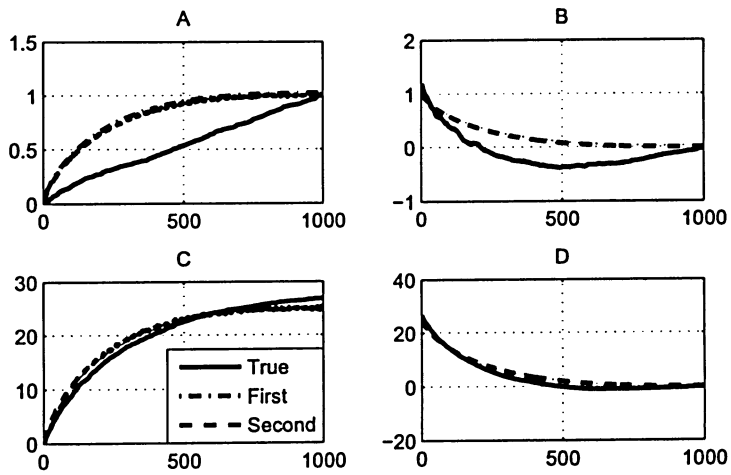

Figure 4.14: The true noisy part of Data error and MSE along with their estimates as function of m using Monte Carlo method "first" and using exact calculation "second" (A) Noisy part of MSE $\sigma_w = 1$ (B) Noisy part of MSE $\sigma_w = 5$ (c) Noisy part of Data error $\sigma_w = 1$ (D) Noisy part of Data error $\sigma_w = 5$. Signal tested is mishmash

can estimate the MSE. First using the Data error, the noiseless part is calculated, then this quantity is added to the noisy part of MSE.

4.1.2 Estimation of the Noiseless Part of MSE

We have suggested two approaches for estimating the $E[(noise_{x_{S_m}})]$ in the previous section. We preferred the exact solution in Equation 4.29, because of its better running time. Using this estimate and Equation 4.24 the noiseless part of MSE can be estimated as follows:

$$\frac{1}{N} \parallel \hat{\Delta}_{S_m} \parallel^2 = \frac{1}{N} \sum_{i=1}^{N} \bar{\theta}(i) = x_{S_m} - E(noise_{x_{S_m}}). \tag{4.31}$$

The desired $\frac{1}{N} \parallel \Delta_{S_m} \parallel^2$ and its estimate using the equation above, along with its estimate using MNDL-SS, have all been plotted in Figure 4.15. The plots are for two different noise standard deviations, "1" and "5" and tested signals are Mishmash and Block. We can see that MNDL thresholding provides a much better estimate for the noiseless part.

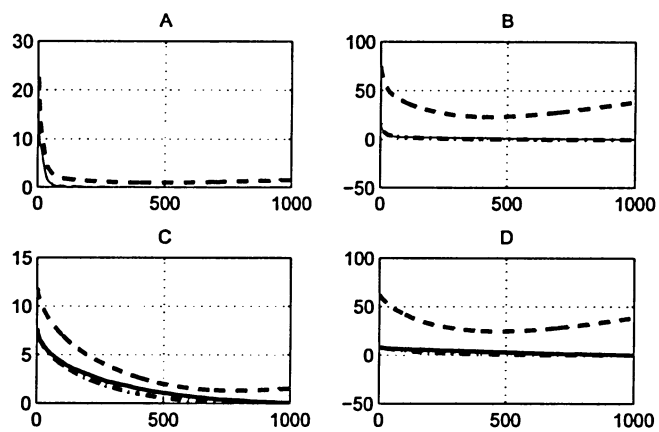

Figure 4.15: Desired unavailable $\frac{1}{N} \parallel \hat{\Delta}_{S_m} \parallel^2$ (solid line), and its estimate as a function of m using MNDL thresholding (-.) and MNDL-SS (- -). (A) Block signal, $\sigma_w = 1$ (B) Block signal, $\sigma_w = 5$ (c) Mishmash signal, $\sigma_w = 1$ (D) Mishmash signal, $\sigma_w = 5$

4.2 Bounds on MSE in Hard Thresholding Method

We summarize the steps of MSE estimate in MNDL thresholding as follows:

- Estimate the noisy part of x_{S_m} using the Equation 4.24, $(E(noise_{x_{S_m}}))$.

- Estimate noiseless part of x_{S_m} which is the same as the noiseless part of z_{s_m} using the Equation 4.31, $(\frac{1}{N} \parallel \hat{\Delta}_{S_m} \parallel^2)$
- Estimate noisy part of z_{S_m} using (4.23), $(E(noise_{z_{S_m}}))$
- MSE is estimated as follows:

$$\hat{z}_{S_m} = \frac{1}{N} \parallel \hat{\Delta_{S_m}} \parallel^2 + E(noise_{z_{S_m}}) \tag{4.32}$$

In Equation 4.32, an estimate of z_{S_m} is calculated as a function of m. The next step in MNDL thresholding is minimizing the z_{S_m} over a different value of m and finding the m_{opt}. The m_{opt}-th coefficient in sorted version of noisy coefficients gives the threshold.

4.3 Simulation Results

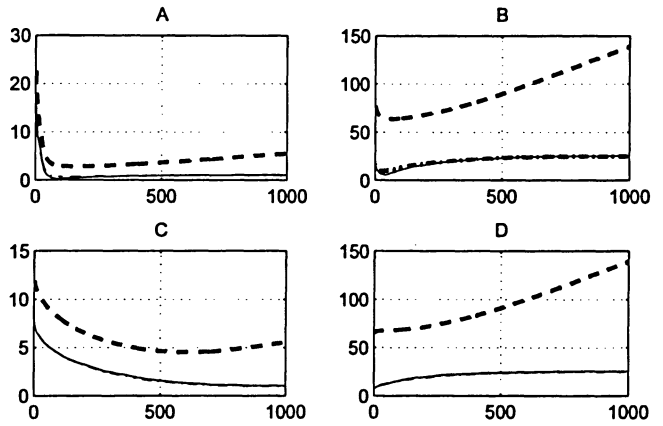

Figure 4.16: Desired unavailable z_{S_m} (solid line), and its estimate using MNDL thresholding (-.) and MNDL-SS (- -) as a function of m. (A) Block signal, $\sigma_w = 1$ (B) Block signal, $\sigma_w = 5$ (c) Mishmash signal, $\sigma_w = 1$ (D) Mishmash signal, $\sigma_w = 5$.

The true z_{S_m} and its estimates using MNDL-SS and MNDL thresholding methods have been plotted in Figure 4.16. This simulation is for two different levels of noise. In both levels, MNDL thresholding estimate performs much better than MNDL-SS. To have a better

	(1)	(2)	(3)
$\sigma_w = 1$	72	77	91
$\sigma_w = 3$	34	29	40
$\sigma_w = 5$	22	18	37

Table 4.1: Comparing m_{opt} and its estimations(1) true value; (2) estimate using MNDL hard thresholding; and (3) estimate using MNDL-SS method.

Block	optimal	MNDL-SS	MNDL Thresholding	MDL	VisuShrink
$\sigma_w = 1$	0.13	0.3	0.28	**0.2**	**0.2**
$\sigma_w = 3$	1.8	2.8	**2**	2.1	2.2
$\sigma_w = 6$	7.9	9.3	**9**	**9**	9.2
$\sigma_w = 10$	14	17.6	17.2	17.7	**15.4**
Mishmash					
$\sigma_w = 1$	0.9	1.2	**0.9**	2	3.3
$\sigma_w = 3$	7.1	7.8	7.5	7.4	**7.3**
$\sigma_w = 6$	7.8	**7.8**	**7.8**	10	7.9
$\sigma_w = 10$	7.86	**7.86**	**7.86**	16	8

Table 4.2: MSE comparison of different thresholding methods with proposed method "MNDL Thresholding". Averaged over ten runs.

comparison between these two methods, the optimum subspace order, m_{opt}, from different approaches are provided in Table 4.1 for different noise variances. In m_{opt} the z_{S_m} is at its minimum. This value directly provides the optimum threshold. A method whose m_{opt} is closer to true z_{S_m}'s is a better method.

Table (4.1) shows that m_{opt} of MNDL thresholding is closer to true m_{opt} than the MNDL-SS for all levels of noise variance. In table (4.2), the MSE of the proposed hard thresholding method is compared to the existing methods; VisuShrink and MDL. The explanation of these method can be found in section (2.2). The comparison is also included the optimal method in which we assume that the noiseless data is known. The method with the smaller MSE is the best one. In every case the smallest MSE is in boldface. The results are the average of 10 runs.

4.4 MNDL Soft Thresholding

In many applications soft thresholding gives a smaller MSE than hard thresholding, especially in image denoising. In soft thresholding, not only we set values smaller than the threshold to zero, but also we reduce the value of coefficients bigger than the threshold by the amount of the threshold. Thus, we need to take into account this changing level of coefficients in our estimations. As we have seen in MNDL hard thresholding, to estimate MSE, we need to determine the effect of additive noise in the expected value of MSE and Data error. In other words, we have to find an estimate for $E(noise_{zsm})$ and $E(noise_{xsm})$ in Equations 4.1 and 4.2 before estimating MSE. In soft thresholding we also, follow the same procedure as hard thresholding. However, we try to involve the effect of soft threshold in every step. To start, let us revisit the expected value of MSE and Data error in the soft thresholding case. Here, the expected value of MSE in subspace S_m is:

$$E(Z_{s_m}) = \frac{1}{N}E(\sum_{i=1}^{m}(V(i) - T_m)^2) + \frac{1}{N}\sum_{i=1}^{N}\bar{\theta}(i)^2 \tag{4.33}$$

where T_m is the the smallest coefficient in subspace S_m. The first part of the equation above is the expected value of MSE noisy part. The difference between expected value of MSE in soft thresholding (Equation 4.33) and in hard Thresholding (Equation 4.1), lies in their noisy part. Here the noise elements are subtracted from soft threshold in every subspace. We have plotted the MSE noisy part for both hard and soft thresholding cases in Figure 4.17. The signal which was used is Block and the noise variance is one. We can see the effect of the soft threshold as an overshoot at starting point of the related graph. The soft threshold also affects the expected value of data error. The expected value of X_{S_m} in soft threshoding is as follows:

$$E(X_{S_m}) = \frac{1}{N}(\sum_{i=1}^{m}E(T_m^2) + \sum_{i=m}^{N}E((\bar{\theta}(i)+V(i))^2)) = \frac{m}{N}T_m^2 + \frac{1}{N}\sum_{i=m}^{N}E((\bar{\theta}(i)+V(i))^2). \tag{4.34}$$

The second part of the equation above is exactly the same as the expected value of Data error in the hard thresholding case, in Equation 4.9. Therefore all the estimations we have used there are valid here too. In the following sections we try to find an estimate for noisy

part of MSE and X_{S_m} in Equations 4.33 and 4.34, and then we use these estimations in MSE calculation. The MSE is later used in providing a proper threshold.

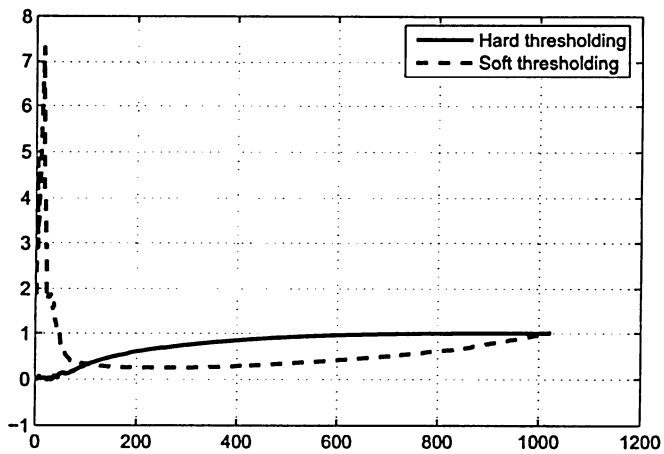

Figure 4.17: Noisy part of MSE error using hard and soft thresholding. The signal used is Block and the noise variance is one .

4.4.1 The Effect of Additive Noise in MSE (Exact Calculation)

As a first step of MNDL thresholding, we want to estimate the expected value of the noisy part of MSE. This quantity is later used as an approximation of MSE's noisy part [1]. Based on Equation 4.33, the expected value of MSE's noisy part is as follows:

$$E(noise_{z_{s_m}}) = \frac{1}{N}\sum_{i=1}^{m} E(V(i) - T_m)^2, \tag{4.35}$$

where $V(i)$s are the associated noisy part of coefficients in subspace S_m, $(V(i) \in \{V(1), V(2), ...V(m)\}$ We introduced two approaches in MNDL hard thresholding to estimate the expected value of MSE noisy part. We have seen that both approaches have the same results. As exact calculation leads to a faster algorithm, here we use this approach too. We use the same approach we had exploited in hard threshoding and find the exact solution of above equation.

[1]The expected value of MSE noisy part is an estimate of it. Therefore, in this thesis we sometimes refer to the "expected value of MSE's noisy part " as the "MSE noisy part"

Calculation of expected value of MSE's noisy part, $E(noise_{z_{s_m}})$: Like MNDL hard thresholding, we first solve the Equation 4.35 for $V(m)$ and then extend our solution for all possible values of $V(i)$s. With Gaussian assumption, and using the Bayesian theory, the expected value of MSE noisy part under condition $\theta(m+1) < \theta(m) < \theta(m-1)$ is calculated as follows:

$$E[(V(m)-T(m))^2] = \frac{\int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} (v_m - T(m))^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m}{\int_{\theta(m+1)-\bar{\theta}(m)}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m - \int_{\theta(m-1)-\bar{\theta}(m)}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m} \tag{4.36}$$

The numerator of equation (4.36) will be:

$$\int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} v_m^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m + \int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} T(m)^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m \tag{4.37}$$

$$- \int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} 2v_m T(m) \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m \tag{4.38}$$

The equation above has three parts; the integral of the first part is

$$\int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} v_m^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m = \frac{\sigma_w}{\sqrt{2\pi}}[\theta(m-1)-\bar{\theta}(m)]e^{-\frac{[\theta(m-1)-\bar{\theta}(m)]^2}{2\sigma_w^2}} \tag{4.39}$$

$$-\frac{\sigma_w}{\sqrt{2\pi}}[\theta(m+1)-\bar{\theta}(m)]e^{-\frac{[\theta(m+1)-\bar{\theta}(m)]^2}{2\sigma_w^2}} + \sigma_w^2[Q(\frac{[\theta(m-1)-\bar{\theta}(m)]}{\sigma_w}) - Q(\frac{[\theta(m+1)-\bar{\theta}(m)]}{\sigma_w})]$$

The integral of the second part would be like:

$$\int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} T(m)^2 \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m = T(m)^2[Q(\frac{[\theta(m+1)-\bar{\theta}(m)]}{\sigma_w}) - Q(\frac{[\theta(m-1)-\bar{\theta}(m)]}{\sigma_w})] \tag{4.40}$$

and the third part is:

$$\int_{\theta(m+1)-\bar{\theta}(m)}^{\theta(m-1)-\bar{\theta}(m)} 2v_m T(m) \frac{1}{\sqrt{2\pi}\sigma_w} e^{\frac{-v_m^2}{2\sigma_w^2}} dv_m = \frac{2T(m)\sigma_w}{\sqrt{2\pi}}[e^{-\frac{[\theta(m+1)-\bar{\theta}(m)]^2}{2\sigma_w^2}} - e^{-\frac{[\theta(m-1)-\bar{\theta}(m)]^2}{2\sigma_w^2}})] \tag{4.41}$$

The numerator is calculated by adding up Equations 4.39 to (4.41). The denominator of Equation 4.36 is calculated as follows:

$$Q(\frac{[\theta(m+1)-\bar{\theta}(m)]}{\sigma_w}) - Q(\frac{[\theta(m-1)-\bar{\theta}(m)]}{\sigma_w}) \tag{4.42}$$

and eventually by dividing the numerator by the denominator the $E[(N_m - T_m)^2]$ in Equation 4.36 is calculated as :

$$E[(N(m) - T(m))^2] = T(m)^2 + \sigma_w^2 + \varepsilon(diff1, diff2) \tag{4.43}$$

where $diff1 = \theta(m+1) - \bar{\theta}(m)$ and $diff2 = \theta(m-1) - \bar{\theta}(m)$ and $\varepsilon(diff1, diff2)$ is

$$\varepsilon(diff1, diff2) = \frac{f_{diff1} + f_{diff2}}{Q(diff1) - Q(diff2)} \tag{4.44}$$

and f_{diff1} and f_{diff2} are defined as:

$$f_{diff1} = \frac{\sigma_w}{\sqrt{2\pi}} \exp(\frac{-diff1^2}{2\sigma_w^2})(diff1) - 2T(m)\frac{\sigma_w}{\sqrt{2\pi}} \exp(\frac{-diff1^2}{2\sigma_w^2}) \tag{4.45}$$

$$f_{diff2} = 2T(m)\frac{\sigma_w}{\sqrt{2\pi}} \exp(\frac{-diff2^2}{2\sigma_w^2}) - \frac{\sigma_w}{\sqrt{2\pi}} \exp(\frac{-diff2^2}{2\sigma_w^2})(diff2) \tag{4.46}$$

The expected value of MSE noisy part, for $V(m)$, was calculated in Equation 4.43. Now we we have to extant our calculations for all $V(i)$s when ($i \in [1, 2, ..., m]$) and take the sum over all possible values. We use two vectors that we defined as $\vec{diff1}$ and $\vec{diff2}$ in the hard thresholding case in Equations 4.21 and 4.22, and generalize our solution for different values of m. Thus, the noisy part of MSE is defined as:

$$E(noise_{z_{S_m}}) = \frac{1}{N}(\sum_{i=1}^{m} E(V(i) - T(m))^2) = \frac{m}{N}T(m)^2 + \frac{m}{N}\sigma_w^2 + \frac{1}{N}\sum_{i=1}^{m}\varepsilon(\vec{diff1}, \vec{diff2}) \tag{4.47}$$

We explore the same estimate of $\vec{diff1}$ and $\vec{diff2}$ in the hard thresholding case (based on Sorting method) and estimate the noisy part of MSE using the above equation. The noisy part that we have estimated here is later used in the estimate of MSE. The MSE is estimated using its expected value in Equation 4.33. As we can see in this equation, to calculate the expected value of MSE, we need to validate the noiseless part as well. However, in order to evaluate the estimate of the noisy part using Equation 4.47, we first estimate the MSE for a known noiseless part (Δ_{S_m}) and in the next section we consider the case with an unknown Δ_{S_m}. We estimate the noisy part using Equation 4.47, then add up this quantity to the noiseless part and estimate MSE. Figure 4.18 shows the desired MSE and its estimate. In MNDL thresholding, we minimize the MSE estimate over all possible values of

Block	m_{opt}	$\hat{m}_{opt}$	T_{opt}	$\hat{T}_{opt}$
$\sigma_w = 1$	293	158	1.2	1.8
$\sigma_w = 6$	170	130	8.9	9.8
Mishmash				
$\sigma_w = 1$	948	809	0.3	0.84
$\sigma_w = 6$	45	110	13.3	10.5

Table 4.3: Comparing m_{opt}, $\hat{m}_{opt}$ and their corresponding threshold when we assume that Δ_{S_m} is known.

m, to find $\hat{m}_{opt}$. We know that the closer $\hat{m}_{opt}$ is to the desired m_{opt} (in which the true MSE is minimized), the better denoising we will have. Therefore we have compared the $\hat{m}_{opt}$, m_{opt} and corresponding thresholds in table (4.3). This table and Figure 4.18 show that our estimate works better when the level of noise is high. Also, the performance of our estimate is dependent on the number of nonzero coefficients. This estimate works better for signals with a few zero coefficients (like Mishmash) than a signal with many zero coefficients(like Block).

4.4.2 The Effect of Additive Noise in Data Error

So far we have estimated the noisy part of MSE in the soft thresholding. The next step is to estimate its noiseless part. The noiseless part of MSE is estimated using the expected value of Data error. We have calculated the expected value of Data error for the soft thresholding case in Equation 4.34. The second part of the expected value of Data error in Equation 4.34 is the same as the expected value of Data error in hard thresholding in Equation 4.9. Therefore, we can exploit the results we had there, for soft thresholding too. Using the estimations in section (4.1.1), the expected value of Data error will be as follows:

$$E(x_{S_m}) = \frac{m}{N}T(m) + \frac{1}{N}\sum_{i=m}^{N}\bar{\theta}(i)^2 + \frac{1}{N}E(\sum_{i=m}^{N}V(i)^2) \tag{4.48}$$

We have $T(m)$ and therefor to derive $\frac{1}{N}\sum_{i=m}^{N}\bar{\theta}(i)^2$ we need to estimate the third part. The third part is exactly the same as noisy part of Data error in hard thresholding and can be

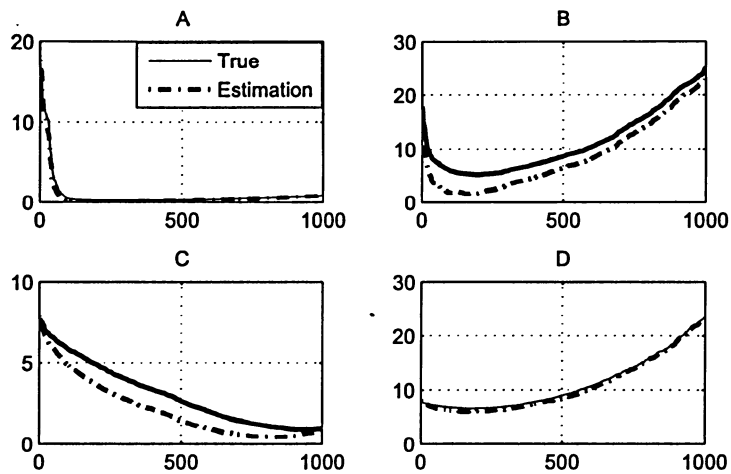

Figure 4.18: Comparison of MSE and its estimate. The estimate is obtained by adding up the known noiseless party to noisy part estimate in Equation 4.50 as a function of m. (A) Block signal, $\sigma_w = 1$ (B) Block signal, $\sigma_w = 5$ (c) Mishmash signal, $\sigma_w = 1$ (D) Mishmash signal, $\sigma_w = 5$

estimated using Equation 4.29. Therefore we have

$$\frac{1}{N}\hat{\Delta}_{S_m} = x_{S_m} - \frac{m}{N}T(m) - \frac{1}{N}E(\sum_{i=m}^{N} V(i)^2) \tag{4.49}$$

To evaluate the performance of our estimate in Equation 4.49, we have plotted the true noiseless part and its estimate in Figure 4.19. In contrast with the noisy part, the noiseless part works better when the level of noise is small and the nonzero coefficients of the signal are a few (like Block signal).

4.5 Bounds on MSE in Soft Thresholding Method

Two different parts of the expected value of MSE (in Equation 4.33), were estimated in sections 4.4.1 and 4.4.2. The MSE expected value can be used as its approximation. We can say that the procedure of estimating MSE in soft thresholding method is as follows:

- The expected value of MSE is computed as a sum of noiseless part and noisy part, Equation 4.33.

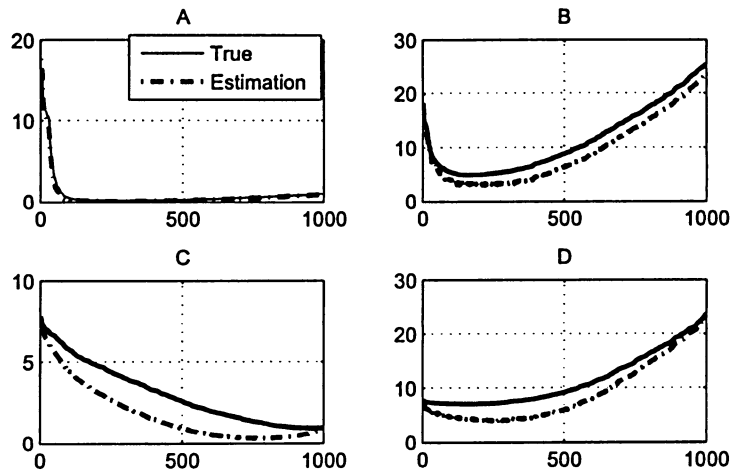

Figure 4.19: The true noiseless part and its estimate in soft thresholding case as a function of m. (A) Block signal, $\sigma_w = 1$ (B) Block signal, $\sigma_w = 5$ (c) Mishmash signal, $\sigma_w = 1$ (D) Mishmash signal, $\sigma_w = 5$

- The noisy part is estimated :
  1. Two parameters, $di\vec{f}f1$ and $di\vec{f}f2$ are estimated, using Equations 4.26 and (4.27).
  2. The $di\hat{\vec{f}}f1$ and $di\hat{\vec{f}}f2$ is employed in Equation 4.47
- The noiseless part is estimated using Equation 4.49
- MSE is estimated as follows:

$$\hat{z}_{S_m} = \frac{1}{N} \parallel \hat{\Delta_{S_m}} \parallel^2 + E(noise_{z_{S_m}}) \tag{4.50}$$

The MSE in (4.50) is calculated as a function of m. The value of m, whose corresponding MSE is minimum, is picked up as the optimum m and the m-th largest absolute value of coefficients is the optimum threshold.

4.6 Comparison of Results

In MNDL thresholding, the threshold is provided by minimizing the estimate of unavailable MSE. The MSE and its estimate for the soft thresholding case are shown in Figure 4.20. The plots are for Mishmash, Block and for two different levels of noise. We can see that

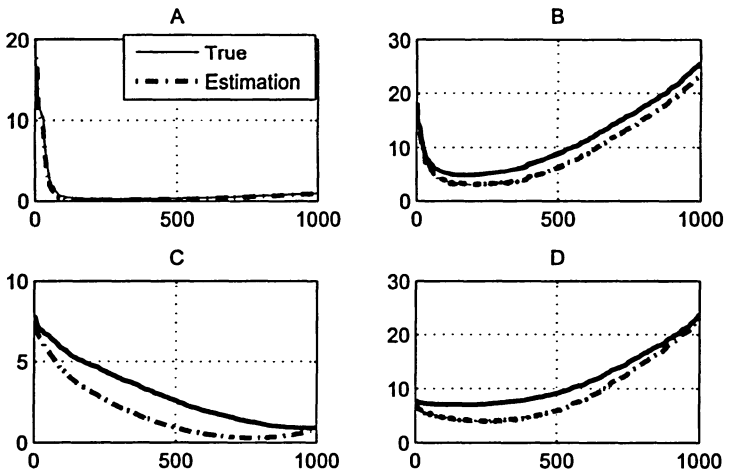

Figure 4.20: The true MSE and its estimate in soft thresholding case as a function of m. (A) Block signal, $\sigma_w = 1$ (B) Block signal, $\sigma_w = 5$ (c) Mishmash signal, $\sigma_w = 1$ (D) Mishmash signal, $\sigma_w = 5$

the estimate of MSE is very close to its true value in this figure. This estimate works well for block signal and with variance of one. The estimate of MSE is computed as a function of m and the value of m which minimizes the MSE is picked up as optimum value. The m_{opt} directly provides the threshold. The m_{opt} in which the true MSE is minimum and its corresponding threshold are compared with our estimations in table (4.4). We denoised the signal with MNDL thresholding then calculated the error between noiseless data and its estimate (MSE) using MNDL. The MSE of our method is compared with optimum denoising method (in which we assume the noiseless data is known), in table (4.5). The comparison includes the different range of noise. We can see in this table that the proposed method

Block	m_{opt}	$\hat{m}_{opt}$	T_{opt}	$\hat{T}_{opt}$
$\sigma_w = 1$	308	285	1.1	1.2
$\sigma_w = 6$	163	78	8.9	12.2
Mishmash				
$\sigma_w = 1$	964	766	0.3	0.98
$\sigma_w = 6$	104	137	10.7	9.7

Table 4.4: Comparing m_{opt} , $\hat{m}_{opt}$ and corresponding thresholds in the soft thresholding case. Averaged over ten runs.

Block	optimal	MNDL-soft
$\sigma_w = 1$	0.3	0.3
$\sigma_w = 3$	2.1	2.2
$\sigma_w = 6$	6.5	6.9
$\sigma_w = 10$	12.1	12.5
Mishmash		
$\sigma_w = 1$	0.89	1.2
$\sigma_w = 3$	4.8	4.9
$\sigma_w = 6$	7.3	7.5
$\sigma_w = 10$	7.7	8.8

Table 4.5: Comparison of MSE of MNDL soft-thresholding method with optimal method. Averaged over ten runs.

has its best performance for an intermediate noise, such as noise with level of 3. It means that our estimate of MSE for this level of noise is better. To estimate the MSE we first formed it as a sum of the noiseless and noisy part and then treated every part separately. The noisy part was estimated in section (4.4.1) and the noiseless part in section (4.4.2). We evaluated our estimations through different graphs and noticed that the estimate of the noiseless part works better when the level of noise is small; on the other hand, the noisy part estimate performs better for high levels of noise. Therefore, for an intermediate level of noise, in which the estimations of both the noisy and noiseless part are reasonable, the proposed method works better. Also comparing table 4.5 with 4.2, we can see that the soft thresholding method, in most of the cases, results in a smaller MSE.

4.7 MNDL Image Denoising

The MNDL soft thresholding is simply extended for $2D$ data, like images. In the image denoising area, soft thresholding works better than hard thresholding in terms of MSE value and visual quality. Besides, experiments have shown that subband-dependent thresholding converges better than universal thresholding methods. In the subband-dependent method, one provides a different threshold for every subband of the wavelet transform. Here we study the MNDL soft thresholding as a subband dependent image denoising method. For this purpose, we estimate MSE in every subband and find the threshold that minimizes it. The procedures of MNDL image denoising method is as follows:

1. Take the Wavelet transform of image.
2. In every subband the MSE is estimated with its expected value as follows:
   - The expected value of MSE is formed as the sum of noisy and noiseless parts, Equation 4.33.
   - The noisy part is estimated using Equation 4.47.
   - The noiseless part is estimated using Equation 4.49.
   - MSE is estimated as a function of m by adding the noiseless part and the noisy part.
3. The MSE is minimized over all possible values of m, and m_{opt} is chosen, where ($m \in \{1, 2, .., N\}$) and N is the number of coefficients in a specific subband.
4. The m_{opt}-th largest absolute value of coefficients is chosen as the optimum threshold.
5. The image is denoised using the provided threshold.
6. Take the inverse Wavelet Transform.

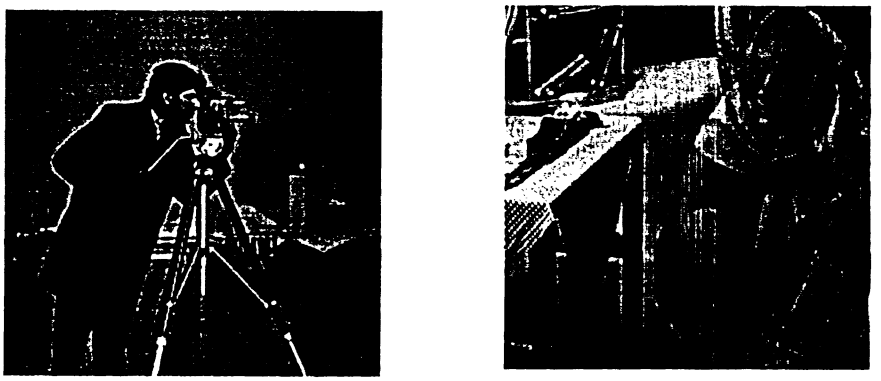

Figure 4.21: Test images, on the left Cameraman and on right the Barbara

	H	V	D
level1	1.2	1.6	0.7
level2	2.5	3.9	1.6
level3	5.7	7.2	3.1
level4	9.2	14.2	6.6

Table 4.6: SNR of every subband of cameraman when the level of noise is "10"

We tested two images, Cameraman as a sample of soft image and Barbara as a sample of a highly detailed image. These images are shown in Figure 4.7 the wavelet transform employs Daubechies's wavelet with eight vanishing moments [50] with four scales of orthogonal decomposition. The MSE and its estimate, in all subbands, are seen in Figure 4.22. As we have discussed in the previous sections the MNDL performance is dependent on the SNR (Signal to Noise Ratio). The SNR of every subband in Figure 4.22 can be seen in table (4.7). In Figure 4.22, the worse estimate is for D1, which is the diagonal subband of level one. In table (4.7), we can see that the SNR of this subband is very small. We know that when the level of noise is small, the estimate of MSE noisy part does not work well and that is why our estimate is far from true in this case. We compared the MSE of the MNDL thresholding with two well-known methods: BayesShrink and SureShrink in table (4.7). The

cameraman	Optimum	MNDL soft thresholding	BayesShrink	SureShrink
$\sigma_w = 5$	16.3	16.9	18	16.5
$\sigma_w = 10$	47	52	50	53
$\sigma_w = 15$	86	92	88	143
Barbara				
$\sigma_w = 5$	16.7	17.8	17.4	16.8
$\sigma_w = 10$	47.8	54.7	48.9	59.17
$\sigma_w = 15$	80.7	87.2	82.1	107

Table 4.7: MSE comparison of proposed method with the existing methods. Averaged over five runs.

BayesShrink method [20], uses the Bayes theorem and assumes the GGD structure for the image coefficients. The SureShrink estimates the MSE using the " SURE" estimate and provides a threshold that minimizes it. These methods are both soft thresholding method and subband-dependent. The proposed method works better than the SureShrink and its result is very close to BayesShrink, which is one of the best known wavelet denoising methods. The MNDL soft thresholding is compared visually with BayesShrink in Figure 4.23. It seems that MNDL works better, the ringing effect in edges of image in MNDL soft thresholding is less than BayesShrink.

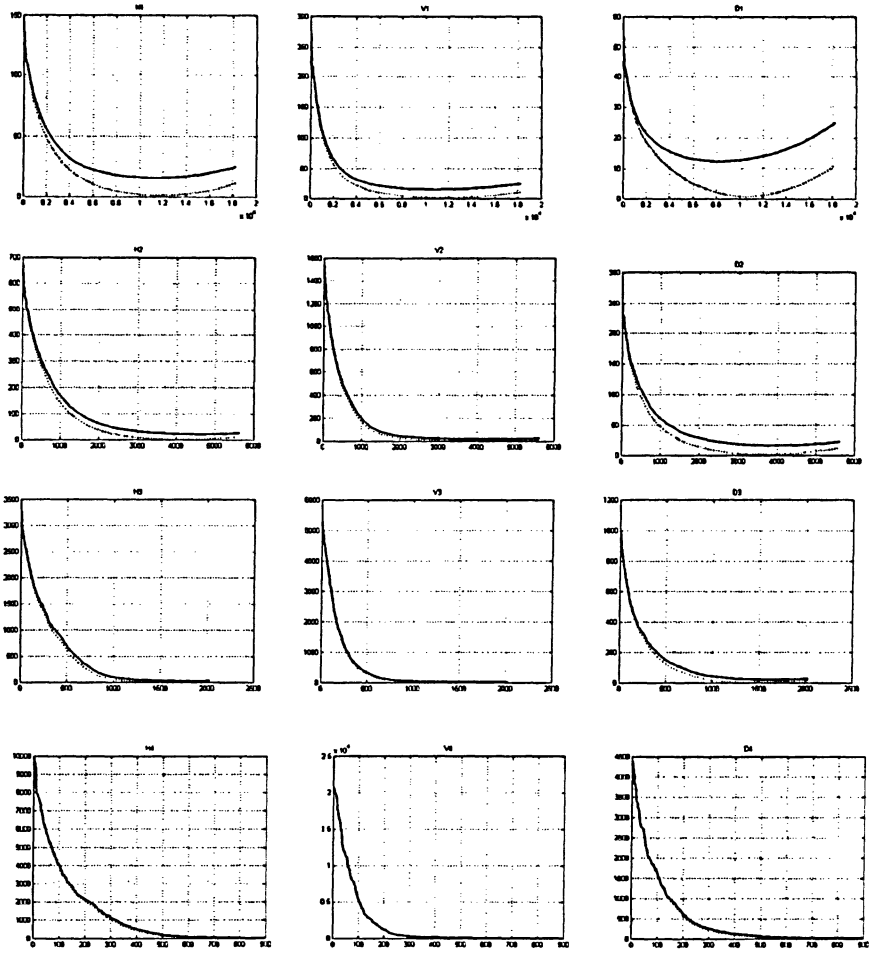

Figure 4.22: The MSE and its estimate as a function of m. the tested image is Cameraman, the level of noise is "3". the first column corresponds to first level of decomposition, the second column is the second level and so on. In every row the subbands are ordered as horizontal (H), Diagonal (D), Vertical(V), from left to right .

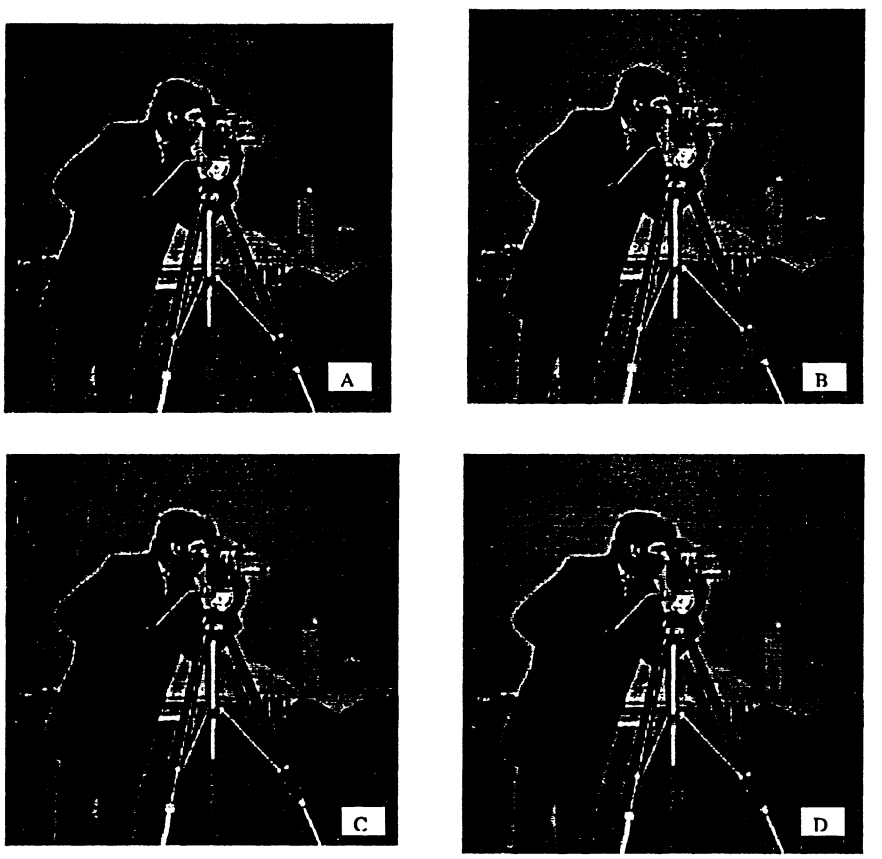

Figure 4.23: (A) noseless image, (B) Noisy image noise level is "5", (C) the estimate of BayesShrink (D) the estimate of of MNDL soft thresholding.

Chapter 5

Conclusion

A new thresholding method based on Minimum Noiseless Description Length subspace selection (MNDL-SS) approach was proposed. MNDL-SS provides bounds on the desired Mean Square Error (MSE) for subspaces of different orders. The approach uses the available Data Error to provide an estimate of the desired MSE for comparison of competing subspaces. In this approach, the structures of the desired MSE and the Data error play important roles. These two quantities are samples of two random variables and the approach heavily relies on the second order statistics of these two random variables. In MNDL thresholding, although the desired criterion is the same as MNDL-SS, the structure of these two random variables are very different from that in MNDL-SS. In this thesis, we develop a new approach to estimate the noisy part of Data Error and MSE in thresholding. We study the applications of MNDL thresholding for both the Soft and Hard thresholding. In either cases, the second order statistics are estimated accordingly in our experiments. We have seen that MNDL Hard thresholding outperformed the well known existing signal denoising methods. Implementing it as Soft thresholding made the results even better than hard thresholding.

We have extended the MNDL soft thresholding for image denoising as well. The new image denoising method was compared with well known methods. SureShrink chooses the threshold minimizing the upper bound of MSE estimation. MNDL thresholding leads to better results. We also included the BayesShrink method in our comparison. BayesShrink is an ad-hoc denoising approach performs very well. The MNDL results are comparable with

this method too.

It is very important to note that in our research we dealt with random variable and important elements of the approach are expected value or samples of those random variables. In MNDL thresholding, we estimated expected value of random variables with samples of that random variable. Although we have tested the validity of these estimates in our work, to generalize this approach, a complete study of the method based on the variance of the error in this estimation is needed. This will be expansion of probabilistic events which has been introduced in MNDL-SS for MNDL thresholding.

MNDL thresholding developed in this thesis is a new approach to denoising which has a great potential in application. This research constructs the foundations of MNDL thresholding. There is a lot to explore in this area, both in theory and application. For example, more research can be done in finding the statistics of random variables involved in this approach and also in considering noiseless signal with specific statistical structure. Due to rich theory of this approach, MNDL thresholding can be explored in the area of quantization.

Bibliography

[1] R. Yang, L. Yin, M. Gabbouj, J. Astola, and Y. Neuvo, "Optimal weighted median filters under structural constraints", *IEEE Transactions Signal Processing*, vol. 43, pp. 591604, Mar. 1995

[2] A. Ben Hamza, P. Luque, J. Martinez, and R. Roman, "Removing noise and preserving details with relaxed median filters",*J. Math Imag Vision*, vol. 11, no. 2, pp. 161177, Oct. 1999.

[3] D. L. Donoho and I. M. Johnstone, "Ideal Spatial Adaption via Wavelet Shrinkage". In Biometrika, vol. 81, pp. 425-455, 1994.

[4] J. Rissanen, "Minimum Description Length Denoising", IEEE Transactions. Information Theory, vol. 46, no. 7, pp. 25372543, Nov. 2000.

[5] D. L. Donoho and I. M. Johnstone, "Adapting to Unknown Smoothness via Wavelet Shrinkage", Journal of the American Statistical Assos., vol. 90, pp.1200-1224, 1995.

[6] S G Chang, B Yu, and M Vetterli, "Spatially Adaptive Wavelet Thresholding with Context Modeling for Image Denoising", Fifth IEEE International Conferance on Image Processing, Vlo. 9 pp. 1529 1531, Oct 1998.

[7] S. Beheshti and Munther A. Dahleh,"A New Information-Theoretic Approach to Signal Denoising and Best Basis Selection", IEEE Transactions on signal Processing, vol. 53, NO. 10, OCTOBER 2005

[8] D. Gabor, "Theory of Communication". In Journal of IEE, vol. 93, pp 429-441, 1946.

[9] J. B. Weaver, Yansun Xu, D. M. Healy, and L. D. Cromwell, "Filtering Noise from Images with Wavelet Transforms", Magnetic Resonance in Medicine, vol.21, pp 288295, 1991.

[10] B. Walczak and D. L. Massart, "Noise Suppresion and Signal Compression using the Wavelet Packet Transform". Chemometrics and Intelligent Laboratory Systems, vol. 36, pp. 8194, Apr. 1997.

[11] N. Saito, "Simultaneous Noise Suppression and Signal Compression using a Library of Orthonormal Bases and the Minimum Description Length Criterion". In Wavelets in Geophysics, pp. 299324, 1994

[12] M. K. Mihak, I. Kozintsev, and K. Ramchandran,"Low-Complexity Image Denoising Based on Statistical Modeling of Wavelet Coefficients". In IEEE Signal Processing Letters , vol. 6, no. 12, December 1999.

[13] M. K. Mihak, I. Kozintsev, and K. Ramchandran, "Spatially Adaptive Statistical Modeling of Wavelet Image Coefficients and Its Application to Denoising,". In IEEE Int. Conf. Acoust. , Speech, Signal Processing, vol. 6, pp. 32533256, Mar. 1999.

[14] Ronald A. DeVore and Bradley J. Lucier, " Fast Wavelet Techniques for Near-Optimal Image Processing". In MILCOM '92, IEEE Military Communications Conference Record, vol.3 pages 11291135. 1992.

[15] P. Moulin, "a awavelet Regularization Method for Diffuse Radar-target Imaging and Speckle-Noise Reduction". In J. Math. Imaging and Vision, vol.3, pp. 123134, 1993.

[16] F Abramovich, T Besbeas, and T Sapatinas, "Empirical Bayes Approach to Block Wavelet Function Estimation". In Computational Statistics and Data Analysis, vol. 39, pp. 435451, 2002.

[17] J. Portilla, Vasily Strela, Martin J. Wainwright, Eero P. Simoncelli "Image Denoising

using Scale Mixtures of Gaussians in the Wavelet Domain". In IEEE Transactions on Image Processing, vol. 12, No. 11 pp. 1338-1351, November 2003.

[18] E P Simoncelli, "Bayesian Denoising of Visual Images in the Wavelet Domain". In Bayesian Inference in Wavelet Based Models, P Muller and B Vidakovic, Eds., chapter 18, Lecture Notes in Statistics, vol. 141 pp. 291308. Spring 1999.

[19] E P Simoncelli, "Statistical Models for Images: Compression, Restoration and Synthesis". In Proc 31st Asilomar Conf on Signals, Systems and Computers, Pacific Grove, CA, pp. 673678 Nov 1997.

[20] S. G.Chang, B.Yu, and M. Vetterli, "Image denoising via lossy compression and wavelet thresholding". In IEEE Transactions. Image Processing, vol. 9, pp. 15321546, Sept. 2000.

[21] R. R. Coifman and D. L. Donoho,"Translation-invariant de-noising,". In Wavelets and Statistics,A. Antoniadis and G. Oppenheim, Eds. Berlin, Germany: Springer-Verlag, 1995.

[22] M. Vetterli and J. Kovacevic, "Wavelets and Subband Coding", in Englewood Cliffs, NJ: Prentice-Hall, 1995.

[23] Xin Li and Michael T. Orchard, "Spatially Aadaptive Imgae Denoising under Overcomplete Expansion". In IEEE Intl Conf on Image Proc, September 2000.

[24] W. T. Freeman and E. H. Adelson. "The Design and Use of Steerable Filters". In IEEE Transactions. Patt. Anal. Mach. Intell, Vol. 13 Num 9, pp 891-906, September 1991.

[25] Pietro Perona. "Steerable-scalable kernels for edge detection and junction analysis". In 2nd European Conf. Computer Vision, pp. 3-18, 1992.

[26] H Greenspan, S Belongie, R Goodman, P Perona, S Rakshit, and C H Anderson. "Overcomplete Steerable Pyramid Filters and Rotation Invariance". In Proceedings CVPR, pp. 222-228 ,1994.

[27] Eero P Simoncelli William T Freeman, "The Steerable Pyramid: A Flexible Architecture For Multi-Scale Derivative Computation ". In 2nd IEEE International Conference on Image Processing. vol III, pp 444-447. October, 1995, Washington, DC.

[28] H. Chipman, E. Kolaczyk, and R. McCulloch, "Adaptive Bayesian Wavelet Shrinkage," J. Amer. Statist. Assoc., vol. 92, No. 440, pp. 14131421, 1997.

[29] M. Clyde, G. Parmigiani, and B. Vidakovic, "Multiple shrinkage and subset selection in wavelets, Biometrika, vol. 85, pp. 391402, 1998.

[30] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Wavelet-Based Statistical Signal Processing using Hidden Markov Models, in ".In IEEE Transactions Signal Processing, vol. 46, pp. 886902, Apr. 1998.

[31] M. Jansen, M. Malfait, and A. Bultheel,"Generalized Cross Validation for Wavelet Thresholding". In Signal Process., vol. 56, pp. 3344, Jan. 1997.

[32] I. M. Johnstone and B.W. Silverman, "Wavelet Threshold Estimators for Data with Correlated Noise". In J. R. Statist. Soc., vol. 59, 1997.

[33] G. Nason, "Choice of the Threshold Parameter in Wavelet Function Estimation," in Wavelets in Statistics, A. Antoniades and G. Oppenheim, Eds. Berlin, Germany, Springer-Verlag, 1995.

[34] F. Ruggeri and B. Vidakovic, "A Bayesian Decision Theoretic Approach to Wavelet Thresholding". In Statist. Sinica, vol. 9, no. 1, pp. 183197, 1999.

[35] B. Vidakovic, "Nonlinear Wavelet Shrinkage with Bayes Rules and Bayes Factors". In J. Amer. Statist. Assoc., vol. 93, no. 441, pp. 173179, 1998.

[36] Y. Wang, "Function estimation via wavelet shrinkage for long-memory data". In Ann. Statist., vol. 24, pp. 466484, 1996.

[37] N. Weyrich and G. T. Warhola, "De-noising using Wavelets and Crossvalidation," Dept. of Mathematics and Statistics, Air Force Inst. of Tech., AFIT/ENC, OH, Tech. Rep. AFIT/EN/TR/94-01, 1994.

[38] Jerome M. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients". In IEEE Transactions on signal processing, Vol. 41, no 12, DEC. 1991

[39] G E P Box and C Tiao, "Bayesian Inference in Statistical Analysis, Addison-Wesley", Reading, MA, 1992.

[40] M Figueiredo and R Nowak, "Wavelet-Based Image Estimation: An Empirical Bayes Approach using Jeffreys Noninformative Prior," IEEE Transactions. Image Proc., vol. 10, no.9, pp. 1322-1331, Sep 2001.

[41] X.-P. Zhang and M. Desai, "Nonlinear Adaptive Noise Suppression Based on Wavelet Transform," submitted for publication.

[42] P Moulin and J Liu, "Analysis of multiresolution image denoising schemes using a generalized Gaussian and complexity priors," IEEE Transactions. Info. Theory, vol. 45, pp. 909919, 1999.

[43] J S Lee, "Digital image enhancement and noise filtering by use of local statistics," IEEE Pat. Anal. Mach. Intell., vol. PAMI-2, pp. 165168, Mar 1980.

[44] H Robbins, "The empirical Bayes approach to statistical decision problems," in *Ann. Math. Statistics*, vol. 35, pp. 120, 1964.

[45] E. P. Simoncelli and E. H. Adelson, "Noise removal via Bayesian wavelet coring", in *Third Intl Conf. on Image Processing*, vol. 1, pp. 379382, September 1996.

[46] , F. Abramovich, T. Sapatinas, and B. W. Silverman, "Wavelet thresholding via a Bayesian approach",in *J. R. Statist. Soc., ser. B*, vol. 60, pp. 725749, 1998.

[47] C. Stein, "Estimation of the mean of a multivariate normal distribution," Ann. Stat., vol. 9, pp. 11351151, 1981.

[48] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," in *IEEE Transactions. Pattern Anal. Machine Intell.*, vol. 11, pp. 674693, July 1989.

[49] A. Fakhrzadeh and S. Beheshti, "Minimum Noiseless Description Length (MNDL) Thresholding", Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Image and Signal Processing, 2007.

[50] I. Daubechies, "Ten Lectures on Wavelets", Vol. 61 of Proc. CBMS-NSF Regional Conference Series in Applied Mathematics. Philadelphia, PA: SIAM, 1992.